



AFRL-AFOSR-VA-TR-2022-0379

Optimization Based Approaches to Network Analytics

Pavlo Krokhmal
UNIVERSITY OF ARIZONA
888 N EUCLID AVE RM 510
TUCSON, AZ, 85719
USA

07/14/2022
Final Technical Report

DISTRIBUTION A: Distribution approved for public release.

Air Force Research Laboratory
Air Force Office of Scientific Research
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE 20220714		2. REPORT TYPE Final		3. DATES COVERED <table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;">START DATE 20190415</td> <td style="width: 50%; border: none;">END DATE 20220414</td> </tr> </table>		START DATE 20190415	END DATE 20220414
START DATE 20190415	END DATE 20220414						
4. TITLE AND SUBTITLE Optimization Based Approaches to Network Analytics							
5a. CONTRACT NUMBER		5b. GRANT NUMBER FA9550-19-1-0161		5c. PROGRAM ELEMENT NUMBER 61102F			
5d. PROJECT NUMBER		5e. TASK NUMBER		5f. WORK UNIT NUMBER			
6. AUTHOR(S) Pavlo Krokhmal							
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) UNIVERSITY OF ARIZONA 888 N EUCLID AVE RM 510 TUCSON, AZ 85719 USA					8. PERFORMING ORGANIZATION REPORT NUMBER		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research 875 N. Randolph St. Room 3112 Arlington, VA 22203				10. SPONSOR/MONITOR'S ACRONYM(S) AFRL/AFOSR RTA2	11. SPONSOR/MONITOR'S REPORT NUMBER(S) AFRL-AFOSR-VA-TR-2022-0379		
12. DISTRIBUTION/AVAILABILITY STATEMENT A Distribution Unlimited: PB Public Release							
13. SUPPLEMENTARY NOTES							
14. ABSTRACT The U.S. Air Force vision of net-centric warfare requires development and deployment of large-scale complex systems that will encompass and interconnect various manned and unmanned weapon platforms and combat units. Efficient and reliable operation of the resulting complex, heterogeneous, multilayered networks (which may include C3 networks, sensory networks, ad-hoc wireless networks connecting battlefield units, etc) will require a deep understanding of the properties and behavior of such networks, particularly in the context of incomplete, conflicting, or uncertain operational data. The objective of this research project was to develop theoretical and computational tools for network analytics, comprising a collection of optimization-based frameworks and methods for comprehensive analysis of complex, heterogeneous, multilayered networks involving imperfect/flawed/uncertain data, common in military operations. The underlying theme of this project has been the development and study of predictive metrics and characterizations of network performance that are specifically defined and constructed through solutions of optimization problems. The conducted research consisted in a fusion of graph-theoretical approaches with the methodologies of discrete/combinatorial, stochastic, and robust optimization, with the focus on obtaining globally optimal solutions for new data-driven network analytic models.							
15. SUBJECT TERMS							
16. SECURITY CLASSIFICATION OF:				17. LIMITATION OF ABSTRACT UU			
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U		18. NUMBER OF PAGES 250			
19a. NAME OF RESPONSIBLE PERSON WARREN ADAMS				19b. PHONE NUMBER (Include area code) 00000000			

2022 AFOSR Final Report

FA9550-19-1-0161

Optimization Based Approaches to Network Analytics

Pavlo Krokhmal (U of Arizona), Sergiy Butenko (TAMU)

July 2022

1 Accomplishments

Complex networked systems manifest themselves in all aspects of society's everyday functions. In particular, the U.S. Air Force vision of net-centric warfare requires development and deployment of large-scale complex systems that will encompass and interconnect various manned and unmanned weapon platforms and combat units. Efficient and reliable operation of the resulting complex, heterogeneous, multilayered networks (which may include cyberinfrastructure networks, Communications, Command, and Control (C3) networks, sensory networks, ad-hoc wireless networks connecting battlefield units, logistics networks, etc.) will require a deep understanding of the properties and behavior of such networks, particularly in the context of incomplete, conflicting, or uncertain operational data.

1.1 Research objectives

The objective of the proposed research was to develop theoretical and computational tools for network analytics, comprising a collection of optimization-based frameworks and methods for comprehensive analysis of complex, heterogeneous, multilayered networks involving imperfect/flawed/uncertain data, which are common in cyberinfrastructure and military operations. The underlying theme of this project was the development and study of predictive metrics and characterizations of network performance that were specifically defined and constructed through solutions of optimization problems.

The accomplished research relied on a fusion of graph-theoretical approaches with the methodologies of discrete/combinatorial, stochastic, and robust optimization, with the focus on obtaining globally optimal solutions for new data-driven network analytic models. The models and methods proposed in the course of this project are quite general, and applicable to many civilian, industrial, social, etc., distributed systems. The research agenda pursued during the course of the project has been divided into the following major thrusts:

1. Fundamental optimization-based framework for network analysis, in particular, in multilayered/multipartite networks. Within this thrust, we worked on development of a systematic framework for analyzing networks using clique relaxations, which provide realistic models of important clusters in networks.
2. Stochastic optimization methods for network analytics. The second research thrust dealt with the situations when the network’s building blocks (nodes, edges, components) were subjected to uncertain failures. It employed models and techniques of stochastic programming to address the concepts that are often applied to characterization of networked systems, such as robustness, vulnerability, resilience, redundancy, scalability, connectivity, and so on through the prism of data-driven optimization.
3. Network performance metrics based on risk-averse optimization. The third thrust extended the general principles of axiomatic risk theory, which was developed primarily in the context of financial applications, to networked systems. Within this thrust, we utilized the theoretical and methodological underpinnings of risk measures to develop novel performance metrics for networked systems.
4. Methods for quantifying a network’s “clusterability”. The fourth thrust addressed some of the most important questions arising in unsupervised network clustering: Is there a “natural” partitioning of a given network into clusters? If there is, how many clusters does it consist of and what properties does each cluster satisfy?

1.2 Accomplishments

Optimization-based quantification of network “clusterability” (Research Thrust 4) represents one of the research thrusts of this project. A theoretical model for network clusterability is employed in the form of the problem of

finding the maximum cardinality of an independent union of clusters (i.e., subgraphs of specific topology) in a graph, a problem that generalizes several classical graph-theoretical problems, such as the maximum clique problem, maximum independent set problem, etc. In this context, a study of clusterability bounds in random graphs has been conducted. Specifically, sufficient conditions ensuring $O(\log n)$ (where n is the number of nodes) upper and lower bounds that hold almost surely are developed and shown to be applicable for the maximum independent union of cliques as well as several clique relaxations. In addition, it is shown that every graph must have an independent union of cliques of cardinality at least $\Omega(\log n)$. It is demonstrated that this bound is asymptotically tight in uniform random graphs, suggesting that these graphs can be viewed as a “least clusterable” class of graphs. The corresponding findings are presented in a paper “Asymptotic Bounds for Clustering Problems in Random Graphs”, currently under review in *Networks*.

Network clusterability plays an important role in analysis of network behavior and properties. In particular, it is critical for understanding the propagation of “cascading” processes in networks, such as blackout failures in electrical grids, spread of viruses, political preferences, influence, etc. A study of “cascading” processes, such as spread of influence, in arbitrary networks has been conducted in the context of influence maximization problem. Tying in the developments within the Research Thrust 1, this study developed a representation for the cascading spread of influence in a given network as a Markovian stochastic process on a “multi-scale” graph, where the vertices of the original network represent lower-scale graphs that describe the dynamics of changes in the vertex’ state. A stochastic upper bound on the rate of influence propagation has been derived, and the problem of influence maximization has been shown to reduce to a convex optimization problem, for which a cutting plane algorithm utilizing max-flow/min-cut separation routine has been developed. The corresponding paper “A scalable Markov chain framework for influence maximization in arbitrary networks” has been published in *IEEE Transactions on Network Science and Engineering*.

In addition, a thorough investigation of the polyhedral properties of the induced cluster subgraphs was performed (Research Thrust 1). In particular, several families of facet-defining valid inequalities for the independent union of cliques (IUC) polytope were derived. Namely, a subset of vertices in a simple undirected graph G inducing a cluster graph is called an independent union of cliques (IUC), and the IUC polytope associated with G is defined as the convex hull of the incidence vectors of all IUCs in the graph. It was shown that the fractional IUC polytope, obtained from re-

laxing integrality of the variables in definition of the original polytope, is a polyhedral outer-approximation of a cubically constrained region in the space of the original variables; hence, it provides a very weak approximation of the (integral) IUC polytope, remarking the importance of incorporating strong valid inequalities in integer (linear) programming solution methods of the Maximum IUC problem. Several facet-defining valid inequalities for the IUC polytope associated with cycle, anti-cycle, star, double-star, fan and wheel graphs, along with the conditions that they remain facet-defining for the IUC polytope associated with a supergraph containing them, as well as some results concerning the corresponding lifting procedures. For each family of the proposed valid inequalities, the computational complexity of the corresponding separation problem was studied; in particular, it has been shown that the separation problem for the families of star, double-star, fan and wheel inequalities are NP-hard. The effectiveness of employing the corresponding cutting planes in an integer linear programming framework for the Maximum IUC problem through computational experiments has been explored. The corresponding paper “Polyhedral Properties of the Induced Cluster Subgraphs” has been published in *Discrete Applied Mathematics*.

The concept of IUC in a bipartite graph is equivalent to that of 1-dependent set, which is a special case of k -dependent set. In another paper, also published in *Discrete Applied Mathematics*, we present a $(1 + \frac{k}{k+2})$ -approximation algorithm for the MAXIMUM k -DEPENDENT SET problem on bipartite graphs for any $k \geq 1$. For a graph with n vertices and m edges, the algorithm runs in $\mathcal{O}(km\sqrt{n})$ time and improves upon the previously best-known approximation ratio of $1 + \frac{k}{k+1}$ established by Kumar et al. [Theoretical Computer Science, 526: 90–96 (2014)]. Our proof also indicates that the algorithm retains its approximation ratio when applied to the (more general) class of König-Egerváry graphs.

Also, continuous cubic formulations for the maximum s -defective clique problem and the maximum s -plex problem are established by generalizing the celebrated Motzkin–Straus formulation for the maximum clique problem to the corresponding clique relaxations. Clique relaxations, such as s -defective clique and s -plex, emerged as attractive, more practical alternatives to cliques in network-based cluster detection models arising in numerous applications. The formulations are used to extend Turan’s theorem and other known lower bounds on the clique number to the considered clique relaxations. In particular, the proposed formulations are used to obtain analytical bounds on the s -defective clique number and the s -plex number of a graph, generalizing three of the standard bounds on the clique number. Results of preliminary numerical experiments with the CONOPT solver

demonstrate that the proposed formulations can be used to quickly compute high-quality solutions for the considered cluster detection problems, such as maximum s-defective clique problem and the maximum s-plex problem. The proposed formulations can also be used to generate interesting test instances for global optimization solvers. The corresponding paper “Continuous Cubic Formulations for Cluster Detection Problems in Networks” has been published online in *Mathematical Programming B*.

The continuous approaches to discrete optimization problems were further explored in our paper titled “A Hierarchy of Standard Polynomial Programming Formulations for the Maximum Clique Problem”, recently accepted for publication in *SIAM Journal on Optimization*. This paper establishes a hierarchy of polynomial programming formulations $(\mathbf{P}^k)$, $k \in \{2, \dots, \omega\}$ for the maximum clique problem, relating the clique number ω of a given graph to the global maximal value of a multilinear polynomial function $f_k(x)$ of degree k over the standard simplex in $\mathbb{R}^{|V|}$. The case of $k = 2$ corresponds to the Motzkin-Straus formulation, and the hierarchical feature is in the fact that the set of local maxima of $(\mathbf{P}^{k+1})$ is a subset of the set of local maxima of $(\mathbf{P}^k)$, $k \in \{2, \dots, \omega - 1\}$. In particular, every local maximum of $(\mathbf{P}^\omega)$ is global. The novel hierarchy established in this paper is fundamentally different from existing hierarchies focusing on convex relaxations, such as Sherali-Adams, Lovász-Schrijver and Lasserre hierarchies. More specifically, the existing hierarchies rely on convex relaxations and proceed by introducing additional variables and constraints in order to improve the quality of the relaxation at each next level of the hierarchy. The convex relaxation obtained at the final level is tight and yields a global optimal solution to the original problem. In contrast to the existing methods, our approach works with the original feasible region; instead, it alters the objective function at each level of the hierarchy. Rather than building a tighter and tighter convex *outer* approximation for the original problem, we construct its equivalent non-convex reformulation aiming to reduce the set of local maxima that are not global. This process can be viewed as an *inner* transformation of the original problem, such that a global maximum of each reformulation is also a global maximum of the original problem, whereas a local maximum of a reformulation yields a heuristic solution to the original problem. The worst-case quality of a local maximum is expected to improve with each next reformulation in the hierarchy, and every local maximum is guaranteed to be global at the final level of the hierarchy. Hence, instead of taking advantage of convexity, as in the previous approaches, we shift the effort towards building an “equi-maximal” reformulation of the original problem, in which every local maximum would be global. This property is

eventually achieved at the final level of the hierarchy. In terms of computational complexity, the improved quality of local maxima in our approach comes with an increased cost of objective function evaluation, which is in contrast to the existing hierarchies, where the increase in computational expense is due to the increase in the number of variables and constraints.

The objective typically employed in existing cluster-detection methods is based on either cardinality or sum-of-weights of vertices, which is appropriate for maximizing the number or sum of “benefits” of the vertices. However, such an objective is not suitable for detecting a cluster maximizing some *efficiency* measure (e.g., a “benefit-to-cost” ratio), which is a common performance metric in many real-life application settings. Our paper titled “On Maximum Ratio Clique Relaxations”, recently published in *Networks*, introduces and studies two clique relaxation models with fractional objectives: the maximum ratio s -plex problem and the maximum ratio s -defective clique problem. The decision version of each problem is shown to be strongly NP -complete; we also discuss some related computational complexity issues. The base optimization models are single-ratio fractional 0–1 problems. We describe two general types of solution methods: the first one is based on mixed-integer linear programs (MILPs) obtained by linearizing the original nonlinear 0–1 models; the other one exploits parametric methods (namely, a binary search and Newton’s method) that solve MILPs in an iterative manner. For the proposed MILPs we derive valid inequalities that are shown to substantially improve the performance of an off-the-shelf MILP solver. Finally, the considered solution approaches are compared via extensive numerical experiments using a set of artificially generated and real-life network instances.

In the context of Research Thrusts 2 and 3, a novel approach to construction of advanced data-driven metrics of network performance that represent solutions of specially formulated stochastic programming model has been proposed. This approach builds upon an analogy with a stochastic programming representation for classes of coherent and convex measures of risk previously developed by the PI, and establishes a metric of network performance as an optimal preemptive allocation of resources to mitigate future uncertain losses due to stochastic risk factors. As one of the first illustrations of the general approach, we developed several Network Flow Resilience metrics, which quantify the resilience of the given network to randomized losses of its throughput capacity, and generalize the classic Maximum Network Flow and Minimum Cost Network Flow models. The developed metrics inherit a number of key properties of the classical network flow models, such as the total unimodularity and the resulting integrality of solutions. It is

also shown that the proposed metrics can be viewed as “network measures of risk”, which compute the risk exposure due to multiple stochastic factors that are nontrivially interacting based on the given network’s topology. As such, they generalize the class of convex measures of risk, which quantify the risk of a single scalar stochastic factor. A number of decomposition algorithms were developed for efficient computation of the proposed metrics of network performance, including a nonstandard nonlinear Benders decomposition algorithm for the nonlinear versions of the corresponding network resilience metrics. The corresponding paper “Risk-Averse Optimization and Resilience of Network Flows” is currently under review in *Networks*.

In the paper “Risk Reduction in Discrete Systems via Diversification and Concentration”, the context of discrete/networked systems is used to shed a new light on the connection between risk theory and optimization. It is commonly acknowledged that risk analysis is an integral part of decision making and optimization under uncertainties, yet the axiomatic foundations of modern risk theory largely ignore the optimization component. By considering the problem in a discrete graph-theoretic setting, we show that the all-important tenet of diversification as a vehicle for risk reduction is actually attributable to a particular “investor’s” attitude, which presumes that zero commitment of resources implies zero losses due to uncertainty. Formally, this “investor’s” attitude is represented by a choice of loss function h such that $h(0, X) = 0$, which arises naturally in financial applications, from which the concept of diversification has originally emerged. Remarkably, it has been shown that the convexity of risk measures, which is central to the modern risk theory, is not required for achieving diversification in this case, provided that the risk is optimized. Further, we show that there exists a dual “regulator’s” attitude, that zero commitment of resources implies unchanged (nonzero) uncertainty/loss (e.g., the risk of a threat remains unchanged if zero resources are spent on mitigation), or, formally $h(0, X) \neq 0$, which enables risk reduction via concentration, or anti-diversification. It has been shown that the risk reduction via diversification and via concentration holds for a broad class of stochastic programming problems. Furthermore, risk-averse formulations that satisfy the “concentration” conditions allow for nontrivial stochastic generalizations of a large class of graph-theoretic and combinatorial optimization problems. Complexity of such risk-averse discrete optimization problems was investigated; in particular, it has been shown that risk-averse minimum spanning tree problem is NP-hard, while risk-averse minimum shortest path is polynomially solvable, while the deterministic versions of these programs are well known to be polynomial. The corresponding combinatorial branch-and-bound and polynomial algorithms

have been proposed.

Within the Research Thrusts 2 and 3, we also considered risk-averse chance-constrained optimization problems on networks. While a very popular method for constructing risk-averse solutions, chance constraints are notorious for being generally nonconvex and thus hard to handle in optimization problems. To this end, a combined sample average approximation (SAA) and a linearization technique is proposed, which circumvents introduction of additional binary variables, which otherwise are commonly introduced when reformulating chance constraints. This method is applied to a two-stage mixed-integer model for optimal placement and planning of distributed generation (DG) units and energy storage system units. Numerical experiments demonstrate that the chance constraints are effective tools for control of load shedding in distributed generation and the proposed approach outperforms traditional SAA chance programming methods in terms computational time. The corresponding paper “Chance constrained optimization based solar microgrid design and dispatch for radial distribution networks” has been published online in *Energy Systems*.

1.3 Dissemination of Results

The results obtained in the course of this project were published/accepted for publication in several reputed journals, including *Mathematical Programming*, *SIAM Journal on Optimization*, *Discrete Applied Mathematics*, *Networks*, *IEEE Transactions on Network Science and Engineering*, and *Energy Systems*. In addition, the PIs and their students have presented the results at the AFOSR Program Review Meetings, Annual INFORMS Meetings, and several other invited/keynote talks in international conferences, research seminars, and summer schools, including

- A plenary talk in the 7th International Conference on Optimization, Simulation and Control (ICOSC 2022), Ulaanbaatar, Mongolia, 06/21/2022 (online);
- An invited seminar talk in the Daniel J. Epstein Department of Industrial & Systems Engineering, University of Southern California, 04/20/2021 (online);
- Two invited talks (by both PIs) in 2021 International Workshop on Modern Optimization and Applications (MOA 2021), Beijing, China, 01/05/2021 (online);

- An invited seminar talk in Fujitsu Co-Creation Research Laboratory, the University of Toronto, 08/25/2020 (online);
- Summer school lectures in ACDL 2020: Advanced Course on Data Science and Machine Learning, Certosa di Pontignano, Siena, Italy, 07/13–07/17/2020 (online);
- A plenary lecture in the Sixth World Congress on Global Optimization (WCGO 2019), Metz, France, 07/08–07/10/2019.

Some of these talks are available online on YouTube.

On June 14, 2021, the PIs have organized the 4th International Workshop on Critical Infrastructure and Network Security (CINS) that was held in conjunction with the SIGMETRICS conference of the Association for Computing Machinery (ACM). The workshop aimed to focus on issues related to the interdependence of infrastructure and ecological, biological, and social networks, in addition to more traditional topics concerning security, robustness, and vulnerability of infrastructure networks. More specifically, the workshop participants will discuss recent progress and will seek to identify promising future research directions on the following topics, among others:

- Optimization and machine learning techniques to predict, restrict and control the damages and losses associated with various risk factors that affect the operations of interdependent networked systems.
- Probability-based risk measures characterizing a network’s vulnerability to edge/node failures.
- Exact, approximate, and heuristic algorithms for designing minimum cost robust network topologies and for detecting robust structures of interest in networks.
- Mathematical models and algorithms for identification of critical network components.
- Mathematical models describing structures that characterize the network’s robustness and vulnerability to a cyber/physical attack or a natural disaster.
- Case studies concerning ecosystems, as well as large-scale real-life supply chain, communication, social, and energy networks.

In addition to the regular talks, the workshop featured four keynote presentations delivered by prominent researchers in the area of network optimization:

- Alfredo Garcia, Texas A&M University
- David Morton, Northwestern University
- Anna Nagurney, University of Massachusetts Amherst
- Pascal Van Hentenryck, Georgia Institute of Technology

Due to the ongoing COVID-19 pandemic, the workshop was held fully virtually. The recorded talks are available on YouTube.

2 Impacts

Development of the principal discipline(s) of the project. In the course this project, we have obtained a number of exciting results in analysis and optimization of networks under uncertainties. In particular, using the setting of discrete and networked systems we elucidated the connection between reduction of risk and diversification, a staple strategy in investment, finance, and, generally, decision making science. We have shown diversification as a vehicle for risk reduction is actually attributable to a particular “investor’s” attitude, which presumes that zero commitment of resources implies zero losses due to uncertainty (e.g., one will not lose any money in the market if she does not make any investments). If the decision-maker has a dual “regulator’s” attitude, namely that zero commitment of resources implies unchanged loss due to uncertainties (e.g., the risk of a threat remains unchanged if zero resources are spent on mitigation), then risk reduction can be achieved via anti-diversification, or concentration. This insight allowed us to set up a framework for formulating nontrivial generalizations of a broad class of graph-theoretical problems under uncertainties, and develop general solution methods for such problems. In addition, we proposed a general approach to constructing theoretically sound metrics of network performance in the presense of uncertainties, and illustrated this approach by developing metrics of network resilience to unforeseen random losses of its throughput capacity. A number of scalable algorithms were proposed for efficient computation of the network’s resilience.

The proposed hierarchy of standard polynomial programming formulations for the maximum clique problem opens a novel direction in the field of

mathematical optimization. In optimization, there are several classical hierarchies (e.g., Sherali-Adams, Lovász-Schrijver, and Lasserre) designed for solving hard problems. These hierarchies are based on different ideas but share a common feature of relying on tighter and tighter convex relaxations in order to eventually obtain a provably optimal solution to the original problem at the final level of the hierarchy. A typical convex relaxation in these hierarchies involves a large number of additional variables and constraints. Our work proposes a fundamentally different type of a hierarchy of formulations, in which the sets of variables and constraints remain the same throughout, while the (nonconvex) objective function is updated at each next level of the hierarchy. Even though the objective function remains nonconvex after every update, we prove that, with each next level in the hierarchy, the set of local optima that are not global gradually shrinks, to eventually produce a formulation in which every local optimum is guaranteed to be global. Quoting one of the referees of *SIAM Journal on Optimization* (where the paper has been accepted for publication),

“I find the hierarchy proposed by the author to be quite refreshing: in an area where much work is devoted to improving current techniques, they propose a novel approach with interesting structural results.”

This proposed departure from the “convexity-dominated mentality” in optimization and applications opens up several new, exciting directions for theoretical, algorithmic, as well as applied developments. We believe this new line of work has a tremendous potential, which we are planning to explore during the next several years.

Other disciplines: One of the developments that may make an impact in other disciplines concerns the obtained insights into the propagation of “cascading” processes in networks, such as blackout failures in electrical grids, spread of viruses, political preferences, rumors, influence, etc. A study of “cascading” processes, such as spread of influence, in arbitrary networks has been conducted in the context of influence maximization problem. We have developed a highly scalable model and numerical algorithms that allow one to study the dynamics of cascading processes in very large-scale networks. We expect that these results could be used in other fields, such as epidemiology, marketing, sociology, etc.

In addition, the cluster detection approached proposed in this project are applicable to numerous domains, ranging from social network analysis to materials science.

The impact on the development of human resources. The project supported several Ph.D. dissertations at the University of Arizona and Texas A&M University. One of the students supported by the project was recognized as a recipient of the very prestigious 2020-2021 Outstanding Engineering Graduate Student Award at the doctoral level. The award is presented annually to just one doctoral student in the Texas A&M University College of Engineering (at the time of the award, the college had 1,700 Ph.D. students). In addition, the CINS workshop the PIs organized (as described in “Dissemination of Results” section) was attended by many students and young researchers.

The impact on teaching and educational experiences. During the course of the project, the PIs developed a significant amount of teaching materials based on the obtained results and/or relevant to the project’s themes. A large portion of these materials are made available to instructors in other institutions and general public. Examples of the developed materials include a recorded presentation dedicated to the results published in one of the papers supported by the project (https://www.youtube.com/watch?v=3Pp0nP_Lz0g) and a comprehensive handbook developed for an undergraduate course focusing on deterministic optimization (https://drive.google.com/file/d/1g6E_cjMkaTn1KTnyvX856gAgZj2T4RLT/view). This and similar manuals developed for graduate courses *Survey of Optimization* and *Linear Programming* are used by other instructors in several universities.

3 Changes

There were no changes in the scope, goals, or direction of the project.

4 Technical Updates

In the course of this project, the PIs have obtained some exciting results in graph and network optimization, which open new inroads into more general problems of 0-1/IP and combinatorial optimization. More specifically, we developed a hierarchy of standard polynomial programming formulations for maximum clique, with the hierarchical aspect lying in the fact that, as we move to a higher-order formulation, the set of local optima that are not global is gradually reduced. The last formulation in the hierarchy has the property that every local maximum is global. The classical hierarchies focus on convex relaxations, whereas the one we developed addresses the

non-convexity in a novel way. As the next step, the PIs consider extending this idea to the general 0-1 IP/combinatorial problems. In another development, we have introduced a generalized framework for formulating and solving graph theoretical and network problems under uncertainties, where the cost/objective is nonlinear/nonseparable and is expressed via an embedded optimization problem that computes the “risk” of the graph structure/solution. We plan to extend the nonlinear hierarchy ideas to this type of problems, as well as generalize these formulations to 0-1/combinatorial setting and consider problems where instead of “risk”, the objective represents “consensus” among multiple agents/entities with possible hidden adversaries.

A HIERARCHY OF STANDARD POLYNOMIAL PROGRAMMING FORMULATIONS FOR THE MAXIMUM CLIQUE PROBLEM*

SERGIY BUTENKO[†], MYKYTA MAKOVENKO[†], AND MILTIADES PARDALOS[†]

Abstract. The maximum clique problem is a classical optimization problem which finds many important applications. Some of the most theoretically interesting and computationally successful approaches to this problem are based on the Motzkin-Straus formulation, expressing the clique number in terms of the maximal value of a standard quadratic program. This intriguing relationship has also motivated significant developments in quadratic optimization, copositive programming, and complexity in nonlinear optimization. Inspired by such developments, this paper establishes a hierarchy of polynomial programming formulations $(\mathbf{P}^k)$, $k \in \{2, \dots, \omega\}$ for the maximum clique problem, relating the clique number ω of a given graph to the global maximal value of a multilinear polynomial function $f_k(x)$ of degree k over the standard simplex in $\mathbb{R}^{|V|}$. The case of $k = 2$ corresponds to the Motzkin-Straus formulation, and the hierarchical feature is in the fact that the set of local maxima of $(\mathbf{P}^{k+1})$ is a subset of the set of local maxima of $(\mathbf{P}^k)$, $k \in \{2, \dots, \omega - 1\}$. In particular, every local maximum of $(\mathbf{P}^\omega)$ is global.

Key words. maximum clique problem, standard polynomial programming, Motzkin-Straus formulation, continuous approaches to discrete optimization

AMS subject classifications. 90C26, 90C35, 05C31

1. Introduction. Consider a simple, undirected graph $G = (V, E)$, where $V = \{1, \dots, n\}$ is the vertex set and $E \subseteq \binom{V}{2}$ is the edge set; here and below, given a set S and a positive integer k , $\binom{S}{k}$ denotes the set of k -element subsets of S . Given $C \subseteq V$, let $G[C]$ denote the subgraph induced by C in G . A set of vertices $C \subseteq V$ is called a *clique* if $G[C]$ is a complete graph. The maximum clique problem asks for a clique of the largest cardinality in G . This cardinality is referred to as the clique number, denoted by $\omega(G)$, which is simplified to ω throughout this paper.

The classical Turán's graph theorem [34] provides the following relation between the number of edges m , the number of vertices n , and the clique number ω :

$$(1.1) \quad m \leq \left(1 - \frac{1}{\omega}\right) \frac{n^2}{2}.$$

This inequality holds at equality for the Turán graph $T(n, \omega)$, which is a complete multipartite graph on n vertices with ω parts of equal or nearly equal size.

Motzkin and Straus [26] proved that

$$(1.2) \quad \frac{1}{2} \left(1 - \frac{1}{\omega}\right) = \max_{x \in \Delta^n} \sum_{\{i,j\} \in E} x_i x_j,$$

where

$$(1.3) \quad \Delta^n = \left\{x \in \mathbb{R}^n \mid \sum_{i \in V} x_i = 1, x_i \geq 0, i \in V\right\}$$

is the standard simplex in $\mathbb{R}^n$. Clearly, Turán's theorem follows from the Motzkin-Straus formulation by setting $x_i = \frac{1}{n} \forall i \in V$ in (1.2), yielding one of the most elegant proofs of the theorem [2].

*Submitted to the editors May 13, 2021.

Funding: Partial support by AFOSR award FA9550-19-1-0161 is gratefully acknowledged.

[†]Wm Michael Barnes '64 Department of Industrial and Systems Engineering, Texas A&M University, College Station, TX 77843-3131 (butenko@tamu.edu, makovenko@tamu.edu, apardalos@tamu.edu).

The maximum clique problem is one of the first problems shown to be NP-hard [20] and is known to be hard to approximate [17, 37]. It is among the most popular problems in operations research and mathematical optimization [3, 7, 36], and the Motzkin-Straus formulation is the cornerstone of research bridging continuous and discrete optimization. The first algorithm for the maximum clique problem based on the Motzkin-Straus formulation was published in [28]. Since then, the formulation has been used to develop new optimality conditions and several successful heuristics for the maximum clique problem based on continuous optimization [5, 11, 14, 15, 19]. This non-traditional approach also inspired some interesting developments in quadratic programming [9, 10, 16, 23], copositive programming [6, 8, 12], and complexity analysis in nonlinear optimization [1, 22, 35]. In addition, generalizations to uniform hypergraphs [31] and hypergraphs with $\{1, 2\}$ -edges [30] have been formulated. More recently, the Motzkin-Straus formulation has been extended to two clique relaxation models, s -defective clique and s -plex [33]. Furthermore, a regularization of the cubic formulation for the maximum s -defective clique proposed in [33] was studied in [4].

Optimality conditions for the Motzkin-Straus quadratic program (QP) have been studied in several works [14, 15, 19, 29]. In particular, the correspondence of its (strict) local maxima to (strictly) maximal cliques and multipartite cliques (see Definition 1.1 below) in the considered graph has been investigated, and several related concepts concerning cliques and multipartite cliques have been defined in connection with optimality conditions for the Motzkin-Straus QP.

DEFINITION 1.1. *A vertex set $C \subseteq V$ is called a multipartite clique if $G[C]$ is a complete multipartite graph. We will use the notation $C = \bigsqcup_{s=1}^p P_s$ for a p -partite clique C with the parts $P_s, s \in [p]$, where $[p]$ denotes the set $\{1, \dots, p\}$.*

A (multipartite) clique is *maximal* if it is not a subset of a larger (multipartite) clique in the graph. A maximal clique C is called *strictly maximal* if no vertex from $V \setminus C$ has $|C| - 1$ neighbors in C .

To state our main contributions, we will need the following definitions.

DEFINITION 1.2. *We will call a multipartite clique $C = \bigsqcup_{s=1}^p P_s$ in G part-maximal if it is not a subset of a $(p+1)$ -partite clique in G .*

DEFINITION 1.3. *We will call a multipartite clique $C = \bigsqcup_{s=1}^p P_s$ in G strongly part-maximal if there does not exist $s \in [p]$ and $D \subseteq P_s$ such that $C \setminus D$ is a subset of a $(p+1)$ -partite clique in G .*

Main contributions. In this paper, we introduce a hierarchy of standard polynomial programming formulations of the maximum clique problem in a given graph G , establishing the relation of the clique number to the global maximal value of a multi-linear polynomial $f_k(x)$ of degree $k \in \{2, \dots, \omega\}$ over the standard simplex Δ^n . More specifically,

$$(1.4) \quad f_k(x) := \sum_{C \in \mathcal{C}_k} \prod_{i \in C} x_i,$$

where $\mathcal{C}_k$ is the set of all cliques on k vertices in G , and we show that

$$\max_{x \in \Delta^n} f_k(x) = \binom{\omega}{k} \left(\frac{1}{\omega} \right)^k$$

in Theorem 3.1. This result includes the Motzkin-Straus formulation as a special case ($k = 2$). The hierarchical aspect of the proposed formulations lies in the fact

that, by increasing the value of k , the set of local maxima that are not global is narrowed down (Theorem 5.5), to eventually yield a formulation with each local maximum being global for $k = \omega$ (Corollary 4.8). To the best of our knowledge, this type of hierarchy is new in mathematical optimization and is fundamentally different from existing hierarchies focusing on convex relaxations, such as Sherali-Adams [32], Lovász-Schrijver [24] and Lasserre [21]. More specifically, the existing hierarchies rely on convex relaxations and proceed by introducing additional variables and constraints in order to improve the quality of the relaxation at each next level of the hierarchy. The convex relaxation obtained at the final level is tight and yields a global optimal solution to the original problem. In contrast to the existing methods, our approach works with the original feasible region; instead, it alters the objective function at each level of the hierarchy. Rather than building a tighter and tighter convex *outer* approximation for the original problem, we construct its equivalent non-convex reformulation aiming to reduce the set of local maxima that are not global. This process can be viewed as an *inner* transformation of the original problem, such that a global maximum of each reformulation is also a global maximum of the original problem, whereas a local maximum of a reformulation yields a heuristic solution to the original problem. The worst-case quality of a local maximum is expected to improve with each next reformulation in the hierarchy, and every local maximum is guaranteed to be global at the final level of the hierarchy. Hence, instead of taking advantage of convexity, as in the previous approaches, we shift the effort towards building an “equi-maximal” reformulation of the original problem, in which every local maximum would be global. This property is eventually achieved at the final level of the hierarchy. In terms of computational complexity, the improved quality of local maxima in our approach comes with an increased cost of objective function evaluation, which is in contrast to the existing hierarchies, where the increase in computational expense is due to the increase in the number of variables and constraints.

We use the Karush-Kuhn-Tucker (KKT) optimality conditions to show that the support of a local maximum of each considered formulation corresponds to a multipartite clique in G (Theorem 4.3). In Theorems 4.4 and 4.11, we further refine this characterization and prove that such a multipartite clique must, in fact, be strongly part-maximal. Finally, we establish the hierarchical structure of local maxima of the proposed formulations (Theorem 5.5). The proof uses the solution of a non-trivial optimization problem treating elementary symmetric polynomials (Lemma 2.2), which could present interest on its own merits.

Organization. The remaining parts of this paper are structured as follows. Section 2 establishes two lemmata to be used in the proofs. Section 3 develops the proposed formulations. Section 4 explores the optimality conditions for the considered formulations. Section 5 studies the relation between local optima for formulations of different orders. Finally, Section 6 provides concluding remarks and outlines directions for future work.

2. Preliminaries. The inequality in Lemma 2.1 will be used in several proofs in the paper.

LEMMA 2.1. *Given three integers, $p \geq k \geq t \geq 0$, where $p \geq 1$, the following inequality holds:*

$$(2.1) \quad \frac{1}{p^{k-t}} \leq \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t} \leq \frac{1}{(k-t)!} \leq 1.$$

Proof. The statement clearly holds for $k = t$, which includes the case of $k = 0$. Now assume that $k > t$ (in particular, $k \geq 1$) and consider a straightforward expansion:

$$\begin{aligned} \frac{1}{p^{k-t}} &\leq \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t} = \frac{(p-t)!}{(p-k)!(k-t)!p^{k-t}} = \frac{(p-k+1)^{\overline{k-t}}}{(k-t)!p^{k-t}} \\ &\leq \frac{1}{(k-t)!p^{k-t}} = \frac{1}{(k-t)!} \leq 1. \end{aligned}$$

(Here $p^{\overline{n}} := p(p+1) \cdots (p+n-1)$ is the rising factorial.) $\square$

To establish the next result, some background on elementary symmetric polynomials is required. Given $x \in \mathbb{R}^p$, we denote by

$$(2.2) \quad e_{k,p}(x) := \sum_{I \in \binom{[p]}{k}} \prod_{i \in I} x_i, \quad k \geq 1$$

the k -th order elementary symmetric polynomial in p variables given by vector $x = (x_1, \dots, x_p)$. In the following lemma, we will use Newton's inequality [27], which states that

$$(2.3) \quad \left(\frac{e_{k,p}(x)}{\binom{p}{k}} \right)^2 \geq \left(\frac{e_{k+1,p}(x)}{\binom{p}{k+1}} \right) \left(\frac{e_{k-1,p}(x)}{\binom{p}{k-1}} \right).$$

It is easy to check that (2.3) can be equivalently written as follows:

$$(2.4) \quad (e_{k,p}(x))^2 \geq e_{k-1,p}(x) e_{k+1,p}(x) \left(\frac{k+1}{k} \right) \left(\frac{p-k+1}{p-k} \right).$$

The equality in (2.4) is achieved if and only if $x_i = x_j$, $\forall i, j \in [p]$.

LEMMA 2.2. Consider the following optimization problem for integers $p > k \geq 2$:

$$(2.5) \quad \begin{aligned} &\text{minimize} && e_{k,p}(x) \\ &\text{subject to} && e_{k-1,p}(x) = \binom{p-1}{k-1}, \\ &&& x \in [0, 1]^p, \end{aligned}$$

where $e_{k,p}(x)$ the k -th order elementary symmetric polynomial in p variables given by x . Then the optimal value of (2.5) is $\binom{p-1}{k}$ and is attained at $x = e_p^j$, a p -vector whose j -th entry is 0 and all other entries are 1, for any $j \in [p]$.

Proof. We treat the cases of $k = 2$ and $k \geq 3$ separately. For $k = 2$ the proof is straightforward, and for $k \geq 3$ we can take advantage of Newton's inequality (2.3).

First, we prove the statement of the lemma for $k = 2$. In this case, the first constraint of (2.5) becomes $\sum_{i=1}^p x_i = p - 1$. Hence, the objective function can be rewritten as

$$(2.6) \quad \frac{1}{2} \sum_{i=1}^p x_i(p-1-x_i) = \frac{1}{2} \left((p-1)^2 - \sum_{i=1}^p x_i^2 \right) \geq \frac{1}{2} ((p-1)^2 - (p-1)) = \binom{p-1}{2},$$

with the equality achieved for any binary x , which must have exactly one 0 entry for feasibility.

In the remainder of the proof, we assume that $k \geq 3$. For $x \in [0, 1]^p$, let

$$Z(x) := \{i \mid x_i = 0\}, \quad M(x) := \{i \mid 0 < x_i < 1\}, \quad E(x) := \{i \mid x_i = 1\},$$

and

$$e_{k,p}^S(x) := \sum_{I \in \binom{[p] \setminus S}{k}} \prod_{i \in I} x_i, \quad k \geq 1.$$

Suppose that x^* is an optimal solution of (2.5). Note that if there is $i \in [p]$ such that $x_i^* = 0$, then $e_{k-1,p}(x^*) = e_{k-1,p}^{\{i\}}(x^*) = \sum_{I \in \binom{[p] \setminus \{i\}}{k-1}} \prod_{j \in I} x_j \leq \binom{p-1}{k-1}$, with equality holding if and only if $x_j^* = 1$ for all $j \neq i$. This implies that $|Z(x^*)| \leq 1$ and the statement of the lemma holds when $|Z(x^*)| = 1$. To complete the proof, we will show that the only remaining case of $|Z(x^*)| = 0$ is impossible. We will use contradiction. Assume that $|Z(x^*)| = 0$, so for all $i \in [p] : x_i^* \neq 0$. For the equality constraint in (2.5) to hold at x^* , $M(x^*)$ must contain at least two elements. In the remainder of the proof, we will first show that in this case all entries x_i^* corresponding to $i \in M(x^*)$ must be equal to each other; then we will use this observation to obtain a contradiction with optimality of x^* .

Suppose there is a pair $\{i, j\} \subseteq M(x^*)$ such that $x_i^* \neq x_j^*$. It is straightforward to check that the linear independence constraint qualification is satisfied at x^* . Therefore, from the KKT conditions for (2.5), there exists $\lambda \in \mathbb{R}$ such that

$$(2.6) \quad e_{k-1,p}^{\{i\}}(x^*) = -\lambda e_{k-2,p}^{\{i\}}(x^*),$$

$$(2.7) \quad e_{k-1,p}^{\{j\}}(x^*) = -\lambda e_{k-2,p}^{\{j\}}(x^*).$$

The last two equations can be rewritten as

$$(2.8) \quad x_j^* e_{k-2,p}^{\{i,j\}}(x^*) + e_{k-1,p}^{\{i,j\}}(x^*) = -\lambda (x_j^* e_{k-3,p}^{\{i,j\}}(x^*) + e_{k-2,p}^{\{i,j\}}(x^*)),$$

$$(2.9) \quad x_i^* e_{k-2,p}^{\{i,j\}}(x^*) + e_{k-1,p}^{\{i,j\}}(x^*) = -\lambda (x_i^* e_{k-3,p}^{\{i,j\}}(x^*) + e_{k-2,p}^{\{i,j\}}(x^*)).$$

By subtracting (2.9) from (2.8) and considering that $x_i^* \neq x_j^*$, we obtain

$$(2.10) \quad e_{k-2,p}^{\{i,j\}}(x^*) = -\lambda e_{k-3,p}^{\{i,j\}}(x^*),$$

which, combined with (2.8) or (2.9), implies that

$$(2.11) \quad e_{k-1,p}^{\{i,j\}}(x^*) = -\lambda e_{k-2,p}^{\{i,j\}}(x^*).$$

Expressing λ from (2.10) and substituting in (2.11) we obtain

$$(2.12) \quad e_{k-1,p}^{\{i,j\}}(x^*) e_{k-3,p}^{\{i,j\}}(x^*) = (e_{k-2,p}^{\{i,j\}}(x^*))^2,$$

which contradicts Newton's inequality (2.4). Therefore, x_i^* has to be equal to x_j^* . Hence, each $x_i^*, i \in [p]$ must be equal to either 1 or some constant $t \in (0, 1)$, and at least two of the entries must be equal to t . Without loss of generality, we can assume

that the first two entries of x^* are equal to t . Denoting the last $p-2$ entries of x^* by $y \in \mathbb{R}^{p-2}$, we can represent x^* as $x^* = (t, t, y)$.

Now let

$$(2.13) \quad x^{(k)}(\varepsilon) := (t + \varepsilon, t_k^*(\varepsilon), y),$$

where $\varepsilon < 1 - t$ and $t_k^*(\varepsilon)$ is chosen to satisfy the following equation:

$$(2.14) \quad e_{k+1,p}(x^*) = e_{k+1,p}(x^{(k)}(\varepsilon)).$$

Note that

$$e_{k+1,p}(x^{(k)}(\varepsilon)) = e_{k+1,p-2}(y) + (t + \varepsilon)t_k^*(\varepsilon)e_{k-1,p-2}(y) + (t_k^*(\varepsilon) + t + \varepsilon)e_{k,p-2}(y)$$

and

$$e_{k+1,p}(x^*) = e_{k+1,p-2}(y) + t^2e_{k-1,p-2}(y) + 2te_{k,p-2}(y).$$

Hence, (2.14) is equivalent to

$$(t + \varepsilon)t_k^*(\varepsilon)e_{k-1,p-2}(y) + (t_k^*(\varepsilon) + t + \varepsilon)e_{k,p-2}(y) = t^2e_{k-1,p-2}(y) + 2te_{k,p-2}(y),$$

which can be rewritten as

$$t_k^*(\varepsilon)((t + \varepsilon)e_{k-1,p-2}(y) + e_{k,p-2}(y)) = t^2e_{k-1,p-2}(y) + (t - \varepsilon)e_{k,p-2}(y).$$

We obtain

$$(2.15) \quad t_k^*(\varepsilon) = \frac{t^2e_{k-1,p-2}(y) + (t - \varepsilon)e_{k,p-2}(y)}{(t + \varepsilon)e_{k-1,p-2}(y) + e_{k,p-2}(y)}.$$

Recall that each entry of y is either 1 or $t \in (0, 1)$; hence, $t_k^*(\varepsilon) > 0$ for a sufficiently small ε . Also, it is easy to see that $t_k^*(\varepsilon) < t < 1$. Now observe that $x^{(k-2)}(\varepsilon)$ is a feasible point for (2.5) due to requirement (2.14) and feasibility of x^* . Also, (2.14) implies that $e_{k,p}(x^*) = e_{k,p}(x^{(k-1)}(\varepsilon))$, that is, the objective value of (2.5) is the same at x^* and $x^{(k-1)}(\varepsilon)$. According to (2.13), $x^{(k-1)}(\varepsilon)$ and $x^{(k-2)}(\varepsilon)$ differ only in the second entry. Clearly, if $t_{k-2}^*(\varepsilon) < t_{k-1}^*(\varepsilon)$, this would imply that $e_{k,p}(x^{(k-2)}(\varepsilon)) < e_{k,p}(x^{(k-1)}(\varepsilon)) = e_{k,p}(x^*)$. Hence, showing that $t_{k-2}^*(\varepsilon) < t_{k-1}^*(\varepsilon)$ would yield a contradiction with optimality of x^* , needed to complete the proof. Using (2.15), we can verify that the inequality $t_{k-1}^*(\varepsilon) > t_{k-2}^*(\varepsilon)$ simplifies to

$$\varepsilon^2((e_{k-2,p-2}(y))^2 - e_{k-1,p-2}(y)e_{k-3,p-2}(y)) > 0,$$

which holds due to Newton's inequality (2.4). This completes the proof. $\square$

3. Standard Polynomial Programming Formulations. Let $\mathcal{C}_k$ denote the set of all cliques of cardinality k in G . We assume that empty set is the only clique with zero vertices. For $U = \{u_1, \dots, u_p\} \subseteq V$, let

$$N_U := \{v \in V \mid \{v, u_i\} \in E \ \forall i \in [p]\}.$$

That is, N_U consists of the vertices that are adjacent to every vertex in U . Also, let $\mathcal{C}_k(U)$ denote the set of all cliques of cardinality k in $G[U]$, and for $W = \{w_1, \dots, w_t\} \subseteq V$, let

$$\mathcal{C}_{k;W}^U := \mathcal{C}_{k;w_1, \dots, w_t}^{u_1, \dots, u_p} := \mathcal{C}_k(N_U \setminus W).$$

That is, both $\mathcal{C}_{k;W}^U$ and $\mathcal{C}_{k;w_1,\dots,w_t}^{u_1,\dots,u_p}$ denote the set of all cliques consisting of k vertices that are adjacent to every vertex in $U = \{u_1, \dots, u_p\}$ and containing no vertex from $W = \{w_1, \dots, w_t\}$. If $U = \emptyset$ or $W = \emptyset$, we will use the notations

$$\mathcal{C}_{k;W} := \mathcal{C}_{k;w_1,\dots,w_t} := \mathcal{C}_k(V \setminus W) \quad \text{and} \quad \mathcal{C}_k^U := \mathcal{C}_k^{u_1,\dots,u_p} := \mathcal{C}_k(N_U),$$

respectively.

In addition, let

$$\pi_C(x) := \prod_{i \in C} x_i.$$

For $x \in \mathbb{R}^{|V|}$, let $S(x) := \{i \in V \mid x_i \neq 0\}$ be the support of x . We will also refer to $S(x)$ as the set of vertices corresponding to x . Given a set of vertices $C \subseteq V$, its characteristic vector x^C is defined as follows:

$$(3.1) \quad x_i^C = \begin{cases} \frac{1}{|C|}, & i \in C, \\ 0, & \text{otherwise.} \end{cases}$$

The introduced notations allow us to invoke various alternative representations of $\sum_{C \in \mathcal{C}_k} \pi_C(x)$ using a set of one or more vertices in our derivations. For example, by observing that for any vertex v the set $\mathcal{C}_k$ of k -vertex cliques is a disjoint union of $\{C \cup \{v\} \mid C \in \mathcal{C}_{k-1}^v\}$ and $\mathcal{C}_{k;v}$, we obtain the following representation:

$$(3.2) \quad \sum_{C \in \mathcal{C}_k} \pi_C(x) = x_v \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x) + \sum_{C \in \mathcal{C}_{k;v}} \pi_C(x).$$

On multiple occasions, we will deal with x^* such that $S(x^*) = \bigsqcup_{s=1}^p P_s$ is a p -partite clique with the parts $P_s, s \in [p]$, where $p \geq k$. We can use the fact that no two vertices from the same part P_s can belong to the same clique to obtain the following alternative representation using P_s (assuming $k \leq p$):

$$(3.3) \quad \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = \sum_{v \in P_s} x_v^* \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + \sum_{C \in \mathcal{C}_{k;P_s}} \pi_C(x^*).$$

In this case, each term $\pi_C(x^*)$ for $C \in \mathcal{C}_k$ is a product of k entries of x^* , each corresponding to a different part P_s of $S(x^*)$. Hence, the sum can be rewritten as follows:

$$(3.4) \quad \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = \sum_{D \in \binom{[p]}{k}} \prod_{s \in D} \sum_{v \in P_s} x_v^*.$$

(This can be easily shown by induction on $p - k$ using (3.3).)

Next we establish one of the main results of the paper, in which the global optimal value of a multilinear polynomial formulated for a given graph over the standard simplex is shown to be related to the clique number of the graph.

THEOREM 3.1. *Consider the following problem of maximizing a multilinear polynomial of degree $k \in \{2, \dots, \omega\}$ over the standard simplex:*

$$(P^k) \quad f_k(G) := \max_{x \in \Delta^n} \sum_{C \in \mathcal{C}_k} \prod_{i \in C} x_i = \max_{x \in \Delta^n} \sum_{C \in \mathcal{C}_k} \pi_C(x).$$

235 Then for any maximum clique C of G , x^C is a global maximum of $(\mathbf{P}^k)$ and

236 (3.5)
$$f_k(G) = \binom{\omega}{k} \left(\frac{1}{\omega} \right)^k.$$

237 *Proof.* Let $x' \in \Delta^n$ be an optimal solution of $(\mathbf{P}^k)$ with the minimum number of
 238 missing edges in the subgraph $G[S(x')]$ induced by its corresponding set of vertices.
 239 First, we will show that $S(x')$ is a clique in G . Assume there is a pair of vertices
 240 $i, j \in S(x')$ such that $\{i, j\} \notin E$. Then, since no clique can include both i and j ,

241
$$f_k(x') = x'_i \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x') + x'_j \sum_{C \in \mathcal{C}_{k-1}^j} \pi_C(x') + \sum_{C \in \mathcal{C}_{k-1}^{i,j}} \pi_C(x').$$

242 Without loss of generality, assume that $\sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x') \geq \sum_{C \in \mathcal{C}_{k-1}^j} \pi_C(x')$. For $v \in V$,
 243 set

244
$$x''_v := \begin{cases} x'_i + x'_j, & v = i, \\ 0, & v = j, \\ x'_v, & v \in V \setminus \{i, j\}. \end{cases}$$

245 Then $\sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x') = \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x'')$, $\sum_{C \in \mathcal{C}_{k-1}^j} \pi_C(x') = \sum_{C \in \mathcal{C}_{k-1}^j} \pi_C(x'')$, $x'' \in \Delta^n$, and

246
$$f(x'') - f(x') = \left(\sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x') - \sum_{C \in \mathcal{C}_{k-1}^j} \pi_C(x') \right) x'_j \geq 0.$$

247 Hence, x'' is an alternative optimal solution, such that $G[S(x'')]$ has at least one less
 248 missing edge than $G[S(x')]$, which contradicts the definition of x' . Thus, $S(x')$ is
 249 a clique in G . This clique must contain at least k vertices, otherwise $f_k(x') = 0 <$
 250 $f_k(x^V)$. Next, we show that $x'_i = x'_j$ for any $i, j \in S(x')$. Assume the opposite, then
 251 there exists ε such that $x'_i - x'_j > \varepsilon > 0$. For $v \in V$, set

252
$$x''_v := \begin{cases} x'_i - \varepsilon, & v = i, \\ x'_j + \varepsilon, & v = j, \\ x'_v, & v \in V \setminus \{i, j\}. \end{cases}$$

253 Since $S(x')$ is a clique, $\sum_{C \in \mathcal{C}_{k-1}^{i,j}} \pi_C(x') = \sum_{C \in \mathcal{C}_{k-1}^{j,i}} \pi_C(x')$; hence, we have:

$$\begin{aligned} f(x'') - f(x') &= (x'_i - \varepsilon)(x'_j + \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{i,j}} \pi_C(x') - x'_i x'_j \sum_{C \in \mathcal{C}_{k-2}^{i,j}} \pi_C(x') \\ &= \varepsilon(x'_i - x'_j - \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{i,j}} \pi_C(x') > 0. \end{aligned}$$

254 This contradicts the optimality of x' . Thus, $x'_i = x'_j$ for any $i, j \in S(x')$, implying
 255 that $x' = x^{S(x')}$ and

256
$$f_k(x') = \binom{|S(x')|}{k} \left(\frac{1}{|S(x')|} \right)^k.$$

257 Finally, observe that $f_k(x')$ increases with the increase in $|S(x')|$, hence it achieves
 258 its maximum at x' if $S(x')$ is a maximum clique of G . $\square$

Observe that the Motzkin-Straus formulation is obtained by setting $k = 2$ in Theorem 3.1. In this case, the corresponding quadratic optimization problem is referred to as the Motzkin-Straus QP. While the focus of this paper is on studying the properties of formulations presented in Theorem 3.1, in the reminder of this section we state several relevant results that follow directly from this theorem. First, note that formulation $(\mathbf{P}^k)$ can be used to obtain a generalization of Turán's graph theorem as follows.

COROLLARY 3.2. *The number c_k of cliques with k vertices in G satisfies the following inequality:*

$$(3.6) \quad c_k \leq \binom{\omega}{k} \left(\frac{n}{\omega}\right)^k.$$

Proof. Set $x_i = \frac{1}{n}$, $i \in V$ in $(\mathbf{P}^k)$, then by Theorem 3.1,

$$\binom{\omega}{k} \left(\frac{1}{\omega}\right)^k \geq c_k \left(\frac{1}{n}\right)^k,$$

which is equivalent to (3.6). $\square$

It should be pointed out that (3.6) is also implied by the following inequality established by Fisher and Ryan [13]:

$$(3.7) \quad \left(\frac{c_1}{\binom{\omega}{1}}\right)^{1/1} \geq \left(\frac{c_2}{\binom{\omega}{2}}\right)^{1/2} \geq \left(\frac{c_3}{\binom{\omega}{3}}\right)^{1/3} \geq \dots \geq \left(\frac{c_\omega}{\binom{\omega}{\omega}}\right)^{1/\omega}.$$

Note that for $k = 2$, we have $c_2 = m$ and (3.6) coincides with the Turán's bound (1.1). In addition, (3.6) is sharp for Moon-Moser graphs [25], which are known to contain the largest possible number of maximal cliques among all n -vertex graphs. Specifically, the Moon-Moser graph on $n = 3p$ vertices is a complete balanced p -partite graph, in which each of the p parts consists of 3 vertices, and hence there are $3^p = 3^{n/3}$ distinct maximum cliques of cardinality p .

Next we establish a relationship between the clique number of a graph and a polynomial function constructed using all the cliques in the graph.

THEOREM 3.3. *Let $\mathcal{C}$ denote the set of all cliques in G . Then*

$$(3.8) \quad \left(1 + \frac{1}{\omega}\right)^\omega = \max_{x \in \Delta^n} \sum_{C \in \mathcal{C}} \pi_C(x).$$

Proof. Let C^* be a maximum clique of G . Then $x^* := x^{C^*}$ is an optimal solution of the problem in $(\mathbf{P}^k)$ for each $k \in \{2, \dots, \omega\}$. Also, for $k = 0$ and $k = 1$, we have $f_k(x) = 1$, hence (3.5) holds for any nonempty graph and any $k \in \{0, 1, \dots, \omega\}$. Thus,

$$\max_{x \in \Delta^n} \sum_{C \in \mathcal{C}} \pi_C(x) = \sum_{k=0}^{\omega} \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = \sum_{k=0}^{\omega} f_k(G) = \sum_{k=0}^{\omega} \binom{\omega}{k} \left(\frac{1}{\omega}\right)^k = \left(1 + \frac{1}{\omega}\right)^\omega,$$

which completes the proof. $\square$

Observe that (3.8) can be used to obtain an upper bound on the clique polynomial [18] at a certain point. For a graph G , the *clique polynomial* $C_G(t)$ is defined

290 as

$$291 \quad (3.9) \quad C_G(t) = \sum_{k=0}^{\omega} c_k t^k.$$

292 Let $f_G(x) = \sum_{C \in \mathcal{C}} \pi_C(x)$ be the objective function in (3.8). For $x = te_n$, where e_n is
 293 the vector of n ones, we obtain the clique polynomial:

$$294 \quad f_G(te_n) = C_G(t).$$

295 Of course, such x will be feasible for the problem in (3.8) if and only if $t = 1/n$, that
 296 is, $x = x^V = \frac{1}{n}e_n$. Thus, we obtain the following corollary.

297 **COROLLARY 3.4.** *The clique polynomial $C_G(t)$ of a simple undirected graph G*
 298 *satisfies the following inequality for $t = \frac{1}{n}$:*

$$299 \quad (3.10) \quad C_G(1/n) \leq \left(1 + \frac{1}{\omega}\right)^{\omega} < e,$$

300 where e is Euler's number.

301 **4. Characterization of Local Maxima.** Consider an arbitrary local maximum
 302 point x^* of $(\mathbf{P}^k)$. Clearly, x^* must satisfy the KKT first order necessary conditions
 303 for some $\lambda^{(k)} \in \mathbb{R}$ and $\mu^{(k)} \in \mathbb{R}^n$, as outlined next.

304 • Stationarity:

$$305 \quad (4.1) \quad \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + \mu_v^{(k)} = \lambda^{(k)}, \quad \forall v \in V.$$

306 • Primal feasibility:

$$307 \quad (4.2) \quad \sum_{v \in V} x_v^* = 1; \quad x_v^* \geq 0, \quad \forall v \in V.$$

308 • Dual feasibility and complementary slackness:

$$309 \quad (4.3) \quad x_v^* \mu_v^{(k)} = 0, \quad \mu_v^{(k)} \geq 0, \quad \forall v \in V.$$

310 In particular, due to (4.3), $\mu_i^{(k)} = 0$ for any $i \in S(x^*)$. Hence, from (4.1) we have

$$311 \quad (4.4) \quad \lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*), \quad \forall i \in S(x^*).$$

312 Also, since $\mathcal{C}_{k-1}^i$ is the disjoint union of $\mathcal{C}_{k-1;j}^i$ and $\{C \cup \{j\} \mid C \in \mathcal{C}_{k-2}^{i,j}\}$ for $j \in V \setminus \{i\}$,
 313 we have

$$314 \quad (4.5) \quad \lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1;j}^i} \pi_C(x^*) + x_j^* \sum_{C \in \mathcal{C}_{k-2}^{i,j}} \pi_C(x^*), \quad \forall i \in S(x^*), j \in V \setminus \{i\}.$$

315

316 In the following lemma, we show that every vertex from $S(x^*)$ belongs to some
 317 k -vertex clique contained in $S(x^*)$.

LEMMA 4.1. *If x^* is a local maximum of $(\mathbf{P}^k)$ for a given $k \in \{2, \dots, \omega\}$, then each $v \in S(x^*)$ belongs to some clique in $\mathcal{C}_k(S(x^*))$.*

Proof. Let $v \in S(x^*)$ and suppose there is no k -vertex clique in $S(x^*)$ containing v . Then $\sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) = 0$ and using (3.2),

$$(4.6) \quad f_k(x^*) = \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = x_v^* \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + \sum_{C \in \mathcal{C}_{k;v}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k;v}} \pi_C(x^*).$$

Since $2 \leq k \leq \omega$, $\mathcal{C}_k \neq \emptyset$. Consider an arbitrary clique $K \in \mathcal{C}_k$. Then it is evident from (4.6) that the direction d given by

$$d_i := \begin{cases} 1, & i \in K \setminus \{v\}, \\ -|K \setminus \{v\}|, & i = v, \\ 0, & \text{otherwise} \end{cases}$$

is a feasible direction of increase for $f_k(x)$ at x^* , a contradiction. $\square$

Next, we show that two adjacent vertices from $S(x^*)$ that have a common non-neighbor in $S(x^*)$ both belong to a k -vertex clique in $G[S(x^*)]$.

LEMMA 4.2. *Suppose x^* is a local maximum of $(\mathbf{P}^k)$ for a given $k \in \{2, \dots, \omega\}$ and there exist $u, v, w \in S(x^*)$ such that $\{u, v\} \in E$ and $\{u, w\}, \{v, w\} \notin E$. Then $u, v \in K$ for some $K \in \mathcal{C}_k(S(x^*))$.*

Proof. Clearly, the statement holds for $k = 2$. Hence, in the remainder of the proof we assume that $k \geq 3$. Suppose there does not exist a clique in $\mathcal{C}_k(S(x^*))$ containing both u and v . By Lemma 4.1, $S(x^*)$ contains a clique $\tilde{C} \in \mathcal{C}_k$ such that $w \in \tilde{C}$. Let $w' \in \tilde{C} \setminus \{w\}$. Then

$$(4.7) \quad \sum_{C \in \mathcal{C}_{k-2}^{w, w'}} \pi_C(x^*) > 0.$$

Consider $d \in \mathbb{R}^n$ such that

$$(4.8) \quad \sum_{i \in V} d_i = 0, \quad d_i = 0 \quad \forall i \in V \setminus \{u, v, w, w'\}.$$

Due to the local optimality of x^* , the KKT conditions (4.1)–(4.3) as well as (4.4)–(4.5) hold for some $\lambda^{(k)} \in \mathbb{R}$ and $\mu^{(k)} \in \mathbb{R}^n$. From (4.3), $\mu_{w'}^{(k)} = 0$. Using (4.1) with $v = w'$ and considering that $\mathcal{C}_{k-1}^{w'}$ is the disjoint union $\mathcal{C}_{k-1;u,v,w}^{w'} \cup \{C \cup \{u\} \mid C \in \mathcal{C}_{k-2}^{u, w'}\} \cup \{C \cup \{v\} \mid C \in \mathcal{C}_{k-2}^{v, w'}\} \cup \{C \cup \{w\} \mid C \in \mathcal{C}_{k-2}^{w, w'}\}$, we have

$$(4.9) \quad \lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1;u,v,w}^{w'}} \pi_C(x^*) + x_u^* \sum_{C \in \mathcal{C}_{k-2}^{u, w'}} \pi_C(x^*) + x_v^* \sum_{C \in \mathcal{C}_{k-2}^{v, w'}} \pi_C(x^*) + x_w^* \sum_{C \in \mathcal{C}_{k-2}^{w, w'}} \pi_C(x^*).$$

Since no two of the vertices $\{u, v, w\}$ can belong to the same k -vertex clique in

346 $G[S(x^*)]$, we have

$$\begin{aligned}
 f_k(x^* + \varepsilon d) &= \sum_{C \in \mathcal{C}_{k;u,v,w,w'}} \pi_C(x^*) + (x_u^* + \varepsilon d_u) \sum_{C \in \mathcal{C}_{k-1;w'}} \pi_C(x^*) + (x_v^* + \varepsilon d_v) \sum_{C \in \mathcal{C}_{k-1;w'}^v} \pi_C(x^*) \\
 &+ (x_w^* + \varepsilon d_w) \sum_{C \in \mathcal{C}_{k-1;w'}^w} \pi_C(x^*) + (x_{w'}^* + \varepsilon d_{w'}) \left(\sum_{C \in \mathcal{C}_{k-1;u,v,w}^{w'}} \pi_C(x^*) + (x_u^* + \varepsilon d_u) \sum_{C \in \mathcal{C}_{k-2}^{u,w'}} \pi_C(x^*) \right. \\
 &\quad \left. + (x_v^* + \varepsilon d_v) \sum_{C \in \mathcal{C}_{k-2}^{v,w'}} \pi_C(x^*) + (x_w^* + \varepsilon d_w) \sum_{C \in \mathcal{C}_{k-2}^{w,w'}} \pi_C(x^*) \right).
 \end{aligned}$$

348 Therefore, using (4.9) and (4.5) for $(i, j) \in \{(u, w'), (v, w'), (w, w')\}$ we obtain:

$$\begin{aligned}
 f_k(x^* + \varepsilon d) - f_k(x^*) &= \varepsilon \lambda^{(k)} (d_u + d_v + d_w + d_{w'}) \\
 &+ \varepsilon^2 d_{w'} \left(d_u \sum_{C \in \mathcal{C}_{k-2}^{u,w'}} \pi_C(x^*) + d_v \sum_{C \in \mathcal{C}_{k-2}^{v,w'}} \pi_C(x^*) + d_w \sum_{C \in \mathcal{C}_{k-2}^{w,w'}} \pi_C(x^*) \right).
 \end{aligned}$$

350 Since $d_u + d_v + d_w + d_{w'} = 0$ due to (4.8), for d to be a direction of improvement, it
 351 is sufficient for it to be a solution to the following system:

$$\begin{aligned}
 &(4.10) \\
 &\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & \sum_{C \in \mathcal{C}_{k-2}^{u,w'}} \pi_C(x^*) & \sum_{C \in \mathcal{C}_{k-2}^{v,w'}} \pi_C(x^*) & \sum_{C \in \mathcal{C}_{k-2}^{w,w'}} \pi_C(x^*) \end{pmatrix} \begin{pmatrix} d_{w'} \\ d_u \\ d_v \\ d_w \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \\ 1 \end{pmatrix}.
 \end{aligned}$$

353 If the matrix above has full row rank, the system has a solution d , which is a direction
 354 of improvement, contradicting the local maximality of x^* . Otherwise, the second row
 355 of the matrix must be proportional to the third row, and since (4.7) holds,

$$\sum_{C \in \mathcal{C}_{k-2}^{u,w'}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-2}^{v,w'}} \pi_C(x^*) > 0,$$

357 which implies that $\exists \hat{C} \in \mathcal{C}_k(S(x^*))$ such that $u, w' \in \hat{C}$ as $k \geq 3$. Since u and v do
 358 not belong to the same k -vertex clique in $G[S(x^*)]$, there must exist $w'' \in \hat{C}$ such
 359 that $\{v, w''\} \notin E$. Now let

$$d'_i := \begin{cases} 1, & i \in \{u, w''\}, \\ -2, & i = v, \\ 0, & \text{otherwise,} \end{cases}$$

361 and observe that

$$\begin{aligned}
 f_k(x^* + \varepsilon d') &= \sum_{C \in \mathcal{C}_{k;u,v,w''}} \pi_C(x^*) + (x_u^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-1;w''}^u} \pi_C(x^*) + (x_v^* - 2\varepsilon) \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) \\
 &+ (x_{w''}^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-1;u}^{w''}} \pi_C(x^*) + (x_u^* + \varepsilon)(x_{w''}^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{u,w''}} \pi_C(x^*).
 \end{aligned}$$

363 Hence, using (4.4) for $i = v$ and (4.5) for $(i, j) = (u, w'')$ and (w'', u) , we obtain:

$$f_k(x^* + \varepsilon d') - f_k(x^*) = \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,w''}} \pi_C(x^*) > 0.$$

Thus, d' is a feasible ascent direction at x^* , a contradiction with the local maximality of x^* . $\square$

Next we show that the support of a local maximum of $(\mathbf{P}^k)$ is a multipartite clique in G . In the proof, we use the fact that a complete multipartite graph is, equivalently, a $\bar{P}_3$ -free graph, where $\bar{P}_3$ is the complement graph of the path on 3 vertices.

THEOREM 4.3. *If x^* is a local maximum of $(\mathbf{P}^k)$ for a given $k \in \{2, \dots, \omega\}$, then $S(x^*)$ is a multipartite clique with at least k parts.*

Proof. Suppose that x^* is a local maximum of $(\mathbf{P}^k)$, but $S(x^*)$ is not a multipartite clique. Due to the local optimality of x^* , the KKT conditions (4.1)–(4.3) as well as (4.4)–(4.5) hold for some $\lambda^{(k)} \in \mathbb{R}$ and $\mu^{(k)} \in \mathbb{R}^n$.

Since $S(x^*)$ is not a multipartite clique, there exist $u, v, w \in S(x^*)$ such that $\{u, v\} \in E$ and $\{u, w\}, \{v, w\} \notin E$. By Lemma 4.2, there exists a k -vertex clique in $G[S(x^*)]$ containing both u and v . This implies that

$$(4.11) \quad \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) > 0.$$

Let $d \in \mathbb{R}^n$ be the direction defined as

$$(4.12) \quad d_i := \begin{cases} -1, & i \in \{u, v\}, \\ 2, & i = w, \\ 0, & \text{otherwise.} \end{cases}$$

Note that $\mathcal{C}_k$ can be represented as the disjoint union of the following five sets:

- the set $\mathcal{C}_{k;u,v,w}$ of cliques in $\mathcal{C}_k$ that contain none of the vertices u, v, w ,
- the set $\{C \cup \{u\} \mid C \in \mathcal{C}_{k-1;v}^u\}$ of cliques in $\mathcal{C}_k$ that contain u but not v ,
- the set $\{C \cup \{v\} \mid C \in \mathcal{C}_{k-1;u}^v\}$ of cliques in $\mathcal{C}_k$ that contain v but not u ,
- the set $\{C \cup \{u, v\} \mid C \in \mathcal{C}_{k-2}^{u,v}\}$ of cliques in $\mathcal{C}_k$ that contain both u and v , and
- the set $\{C \cup \{w\} \mid C \in \mathcal{C}_{k-1}^w\}$ of cliques in $\mathcal{C}_k$ that contain w .

We have

$$\begin{aligned} f_k(x^* + \varepsilon d) &= \sum_{C \in \mathcal{C}_{k;u,v,w}} \pi_C(x^*) + (x_u^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-1;v}^u} \pi_C(x^*) + (x_v^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-1;u}^v} \pi_C(x^*) \\ &\quad + (x_u^* - \varepsilon)(x_v^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) + (x_w^* + 2\varepsilon) \sum_{C \in \mathcal{C}_{k-1}^w} \pi_C(x^*). \end{aligned}$$

Hence, using (4.4) for $i \in \{u, v, w\}$ and (4.5) for $(i, j) \in \{(u, v), (v, u)\}$ we obtain:

$$\begin{aligned} &f_k(x^* + \varepsilon d) - f_k(x^*) \\ &= -\varepsilon \sum_{C \in \mathcal{C}_{k-1;v}^u} \pi_C(x^*) - \varepsilon \sum_{C \in \mathcal{C}_{k-1;u}^v} \pi_C(x^*) + 2\varepsilon \sum_{C \in \mathcal{C}_{k-1}^w} \pi_C(x^*) \\ &\quad + (\varepsilon^2 - \varepsilon x_u^* - \varepsilon x_v^*) \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) \\ &= -\varepsilon \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) - \varepsilon \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + 2\varepsilon \sum_{C \in \mathcal{C}_{k-1}^w} \pi_C(x^*) + \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) \\ &= -\varepsilon \lambda^{(k)} - \varepsilon \lambda^{(k)} + 2\varepsilon \lambda^{(k)} + \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) = \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) > 0, \end{aligned}$$

where the inequality follows from (4.11). Clearly, d is a feasible ascent direction for

the objective function $f_k(x)$ of $(\mathbf{P}^k)$ at x^* , a contradiction with the local maximality of x^* . This proves that $S(x^*)$ is a multipartite clique.

Finally, due to Lemma 4.1, every vertex from $S(x^*)$ belongs to a k -vertex clique in $G[S(x^*)]$. This implies that the complete multipartite graph $G[S(x^*)]$ has at least k parts. $\square$

The following theorem further refines the structure of a local maximum x^* of $(\mathbf{P}^k)$.

THEOREM 4.4. *Suppose that $S(x^*) = \bigcup_{s=1}^p P_s$, where $P_s, s \in [p]$ are the parts of the multipartite clique $S(x^*)$ defined by a local maximum x^* of $(\mathbf{P}^k)$. Then*

$$(4.13) \quad \sum_{u \in P_s} x_u^* = \frac{1}{p}, \quad \forall s \in [p].$$

Moreover,

$$(4.14) \quad f_k(x^*) = \binom{p}{k} \frac{1}{p^k}.$$

Proof. Let $u \in P_r, v \in P_s$ for some distinct $r, s \in [p]$. Consider (4.1) for u and v . Since x^* is a local maximum, the following must hold:

$$(4.15) \quad \lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) \implies \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) - \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) = 0.$$

Using the alternative representation in the form (3.3), we can rewrite (4.15) as

$$(4.16) \quad \sum_{w \in P_s} x_w^* \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) + \sum_{C \in \mathcal{C}_{k-1}^{v,w}} \pi_C(x^*) - \sum_{w \in P_r} x_w^* \sum_{C \in \mathcal{C}_{k-2}^{v,w}} \pi_C(x^*) - \sum_{C \in \mathcal{C}_{k-1}^{u,w}} \pi_C(x^*) = 0.$$

Since $S(x^*)$ is a multipartite clique, $\forall w \in P_s, w' \in P_r$ we have

$$\sum_{C \in \mathcal{C}_{k-1}^{u,w}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1}^{u,w'}} \pi_C(x^*), \quad \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-2}^{u,w'}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-2}^{v,w'}} \pi_C(x^*) \neq 0.$$

Then (4.16) becomes

$$\sum_{w \in P_s} x_w^* \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) - \sum_{w \in P_r} x_w^* \sum_{C \in \mathcal{C}_{k-2}^{v,w}} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*) \left(\sum_{w \in P_s} x_w^* - \sum_{w \in P_r} x_w^* \right) = 0,$$

and, therefore,

$$\sum_{w \in P_s} x_w^* = \sum_{w \in P_r} x_w^* = \frac{1}{p}, \quad \forall i, j \in [p].$$

Finally, the fact that $S(x^*)$ is a multipartite clique implies that (3.4) holds, hence (4.17)

$$f_k(x^*) = \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = \sum_{D \in \binom{[p]}{k}} \prod_{r \in D} \sum_{v \in P_r} x_v^* = \sum_{D \in \binom{[p]}{k}} \prod_{r \in D} \frac{1}{p} = \sum_{D \in \binom{[p]}{k}} \frac{1}{p^k} = \binom{p}{k} \frac{1}{p^k},$$

which completes the proof. $\square$

COROLLARY 4.5. Suppose x^* and $\bar{x}$ are two local maxima of $(\mathbf{P}^k)$ such that $S(x^*)$ is a p -partite clique and $S(\bar{x})$ is a $\bar{p}$ -partite clique. If $p > \bar{p}$ then $f_k(x^*) > f_k(\bar{x})$.

Proof. Note that by Theorem 4.4, $p > \bar{p} \geq k$. It is sufficient to show that the statement holds for $p = \bar{p} + 1$. From (4.17) we have

$$\begin{aligned} f_k(x^*) - f_k(\bar{x}) &= \binom{p}{k} \frac{1}{p^k} - \binom{p-1}{k} \frac{1}{(p-1)^k} \\ &= \frac{(p-1)!}{k!(p-k-1)!} \left[\frac{(p-1)^k - (p-k)p^{k-1}}{(p-k)p^{k-1}(p-1)^k} \right]. \end{aligned}$$

Since $(p-1)^k - (p-k)p^{k-1} = kp^{k-1} - \sum_{i=0}^{k-1} (p-1)^i p^{k-1-i} > kp^{k-1} - \sum_{i=0}^{k-1} p^{k-1} = 0$, the statement of the corollary holds. $\square$

COROLLARY 4.6. If x^* is a local maximum of $(\mathbf{P}^k)$, then the KKT multiplier $\lambda^{(k)}$ in (4.1) is given by

$$(4.18) \quad \lambda^{(k)} = \binom{p-1}{k-1} \frac{1}{p^{k-1}}.$$

Proof. As before, we assume that $P_1, \dots, P_p$ are the parts of the complete multipartite clique $S(x^*)$. Let u be a vertex from P_1 . From (4.1) and the result of Theorem 4.4:

$$\lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) = \sum_{D \in \left(\{2, \dots, p\} \atop k-1\right)} \prod_{s \in D} \sum_{v \in P_s} x_v^* = \binom{p-1}{k-1} \frac{1}{p^{k-1}}.$$

The following corollary provides an alternative proof of Theorem 3.1.

COROLLARY 4.7. If x^* is a global maximum of $(\mathbf{P}^k)$, then

$$f_k(x^*) = \binom{\omega}{k} \frac{1}{\omega^k}.$$

Proof. There is no complete multipartite subgraph with $\omega + 1$ parts in G . Therefore, due to Theorem 4.4 and Corollary 4.5, the statement of the corollary holds. $\square$

COROLLARY 4.8. For $k = \omega$, every local maximum of $(\mathbf{P}^k)$ is global.

Proof. Let x^* be a local maximum of $(\mathbf{P}^k)$ with $k = \omega$. According to Theorem 4.3, $S(x^*)$ is a multipartite clique with at least ω parts. Since the graph cannot contain a multipartite clique with more than ω parts, $f_\omega(x^*) = \frac{1}{\omega^\omega}$ by Theorem 4.4. Hence, x^* is a global maximum by Corollary 4.7. $\square$

COROLLARY 4.9. If x^* is a strict local maximum of $(\mathbf{P}^k)$ then $S(x^*)$ is a clique in G and $x^* = x^{S(x^*)}$.

Proof. Note that (4.17) implies that $f_k(x^*) = f(x')$ for any $x' \in \{x \in \mathbb{R}^n \mid \sum_{u \in P_s} x_u = \frac{1}{p}, s \in [p]\}$. Hence, x^* can be a strict local maximum only if $|P_s| = 1 \forall s \in [p]$, that is, if C is a clique. $\square$

LEMMA 4.10. If x^* is a local maximum of $(\mathbf{P}^k)$ then $S(x^*)$ is a part-maximal multipartite clique in G .

443 *Proof.* By Theorem 4.3, $S(x^*) = \bigsqcup_{s=1}^p P_s$ is a p -partite clique for some $p \geq k$.
 444 Assume it is not part-maximal. Then there exists $v \in V \setminus S(x^*)$ such that $S(x^*) \cup \{v\}$
 445 is a $(p+1)$ -partite clique in G . Consider the following direction $d \in \mathbb{R}^n$:

$$446 \quad d_i := \begin{cases} -1, & i \in P_1, \\ |P_1|, & i = v, \\ 0, & \text{otherwise.} \end{cases}$$

447 Then

$$448 \quad f_k(x^* + \varepsilon d) - f_k(x^*) = \sum_{i \in P_1} (-\varepsilon) \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) + \varepsilon |P_1| \sum_{C \in \mathcal{C}_{k-1; P_1}^v} \pi_C(x^*) + \varepsilon |P_1| \sum_{i \in P_1} (x_i^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^i} \pi_C(x^*).$$

Note that due to the optimality conditions (4.1)-(4.3), since $S(x^*) \cup \{v\}$ is a multipartite clique and $x_v^* = 0$, for $i \in P_1$ we have

$$\lambda^{(k)} = \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1; P_1}^v} \pi_C(x^*).$$

Hence, for $\varepsilon < \min_{i \in P_1} \{x_i^*\}$, we obtain

$$f_k(x^* + \varepsilon d) - f_k(x^*) = -\varepsilon |P_1| \lambda^{(k)} + \varepsilon |P_1| \lambda^{(k)} + \varepsilon |P_1| \sum_{i \in P_1} (x_i^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^i} \pi_C(x^*) > 0.$$

449 Thus, d is a feasible direction of improvement for $(\mathbf{P}^k)$ at x^* , a contradiction. $\square$

450 THEOREM 4.11. *If x^* is a local maximum of $(\mathbf{P}^k)$ then $S(x^*)$ is a strongly part-*
 451 *maximal multipartite clique in G .*

452 *Proof.* By Lemma 4.10, $S(x^*) = \bigsqcup_{s=1}^p P_s$ is a part-maximal p -partite clique for
 453 some $p \geq k$. Assume it is not strongly part-maximal. Then, according to Definition
 454 1.3, for some $r \in [p]$ there exists $D \subset P_r$ such that $S(x^*) \setminus D$ is a subset of a
 455 $(p+1)$ -partite clique in G . Since $S(x^*)$ is part-maximal, D has to be non-empty. We
 456 will consider two possible cases, $D \neq P_r$ and $D = P_r$. We will provide a direction of
 457 improvement for each case to obtain a contradiction with the local maximality of x^* .

458 Case 1: $D \neq P_r$. Then there exists a vertex $u \in V \setminus S(x^*)$ such that u is adjacent
 459 to all vertices in $S(x^*) \setminus D$, while $D \cup \{u\}$ is an independent set. Consider $d \in \mathbb{R}^n$
 460 defined as follows:

$$461 \quad d_i := \begin{cases} -1, & i \in D, \\ |D| & i = u, \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$f_k(x^* + \varepsilon d) = \sum_{i \in D} (x_i^* - \varepsilon) \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) + \varepsilon |D| \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) + \sum_{i \in P_r \setminus D} x_i^* \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*).$$

Note that, according to (4.4), $\sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) = \lambda^{(k)}$ for all $i \in S(x^*)$, including all
 $i \in D$. Also, since $S(x^*)$ is a multipartite clique and $x_u^* = 0$, for any $v \in P_r$ we have
 $\sum_{C \in \mathcal{C}_{k-1; P_r}^u} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) = \lambda^{(k)}$. Hence,

$$\sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) = \sum_{i \in P_r \setminus D} x_i^* \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) + \sum_{C \in \mathcal{C}_{k-1; P_r}^u} \pi_C(x^*) = \lambda^{(k)} \sum_{i \in P_r \setminus D} x_i^* + \lambda^{(k)},$$

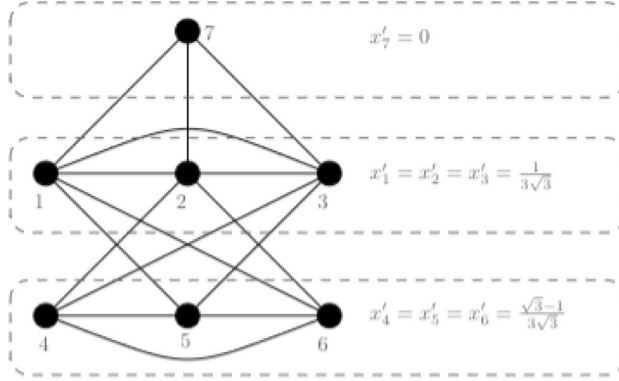


FIG. 1. A graph in which $\{1, \dots, 6\}$ is a strongly part-maximal 3-partite clique, x' satisfies the KKT conditions for $(\mathbf{P}^k)$ with $k = 3$, but x' is not a local maximum of $(\mathbf{P}^k)$.

and

$$f_k(x^* + \varepsilon d) - f_k(x^*) = -\varepsilon |D| \lambda^{(k)} + \varepsilon |D| \left(\lambda^{(k)} \sum_{i \in P_r \setminus D} x_i^* + \lambda^{(k)} \right) = \varepsilon |D| \lambda^{(k)} \sum_{i \in P_r \setminus D} x_i^* > 0.$$

462 Case 2: $D = P_r$. Then there exist two vertices $u, v \in V \setminus S(x^*)$ such that $\{u, v\} \in E$
 463 and both u and v are adjacent to all vertices in $S(x^*) \setminus P_r$. Consider $d \in \mathbb{R}^n$ defined
 464 as follows:

$$465 \quad d_i := \begin{cases} -1, & i \in P_r, \\ |P_r|/2 & i \in \{u, v\} \\ 0, & \text{otherwise.} \end{cases}$$

Then

$$f_k(x^* + \varepsilon d) - f_k(x^*) = \sum_{i \in P_r} (-\varepsilon) \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) + \frac{1}{2} \varepsilon |P_r| \left(\sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) + \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) \right) + \delta,$$

where

$$\delta := \frac{1}{4} \varepsilon^2 |P_r|^2 \sum_{C \in \mathcal{C}_{k-2}^{u,v}} \pi_C(x^*).$$

Noting that for $i \in P_r$

$$\sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) = \lambda^{(k)}, \quad \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) \geq \lambda^{(k)}, \quad \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) \geq \lambda^{(k)},$$

466 and $\delta > 0$, we conclude that $f_k(x^* + \varepsilon d) - f_k(x^*) > 0$. □

467 It is interesting to observe that if a feasible point x' for $(\mathbf{P}^k)$ corresponds to a
 468 strongly part-maximal multipartite clique in G and the KKT conditions are satisfied
 469 at some λ', μ' , this does not imply local maximality of x' for $(\mathbf{P}^k)$. Indeed, consider an
 470 example graph and a feasible x' shown in Figure 1 for $k = 3$. Note that x' satisfies the

KKT conditions with $\lambda' = 1/9$, $\mu' = 0$, and $\{1, 2, 3, 4, 5, 6\}$ forms a 3-partite clique, but x' is not a local maximum, as the direction d given by

$$d_i = \begin{cases} 0, & i \in \{1, 2, 3\}, \\ -1, & i \in \{4, 5, 6\}, \\ 3, & i = 7 \end{cases}$$

is a direction of improvement.

5. Hierarchical Structure of the Local Maxima. In this section, we establish the main structural result of the paper. Namely, in [Theorem 5.5](#) we show that given $k \in \{2, \dots, \omega - 1\}$, any local maximum of $(\mathbf{P}^{k+1})$ is also a local maximum of $(\mathbf{P}^k)$. The proof of this result takes advantage of properties of local maxima established in several lemmata stated and proved before [Theorem 5.5](#).

For a KKT point x^* of $(\mathbf{P}^k)$ with associated KKT multipliers $\lambda^{(k)}$ and $\mu^{(k)}$, we will use the following notations:

$$(5.1) \quad Z_k := \{i \mid \mu_i^{(k)} = 0, x_i^* = 0\}; \quad I_k := S(x^*) \cup Z_k = \{i \mid \mu_i^{(k)} = 0\}.$$

In addition, for a direction d such that $x^* + d$ is a feasible point for $(\mathbf{P}^k)$, let

$$(5.2) \quad Y_k^d := S(d) \setminus I_k = \{i \mid d_i > 0, \mu_i^{(k)} > 0\}.$$

In the following two lemmata, we assume that x^* is a local maximum of $(\mathbf{P}^k)$, the corresponding multipartite clique is given by $S(x^*) = \bigsqcup_{s=1}^p P_s$, where $p \geq k$, and $\lambda^{(k)}$ and $\mu^{(k)}$ are the corresponding KKT multipliers satisfying (4.1)–(4.3). Lemma 5.1 characterizes the set of vertices outside of the support set of a local maximum that have the corresponding KKT multiplier equal to zero. Lemma 5.2 further refines this characterization, and is then used in Lemma 5.3 to show the KKT conditions for $(\mathbf{P}^k)$ are satisfied at a point of local maximum of $(\mathbf{P}^{k+1})$.

LEMMA 5.1. *If $Z_k \neq \emptyset$ then for each vertex $u \in Z_k$, $\{u\} \cup S(x^*)$ is a p -partite clique. Moreover, $S(x^*)$ is a maximal p -partite clique if and only if $Z_k = \emptyset$.*

Proof. First, we show that if $u \in Z_k$ has no neighbor in one of the parts P_r , $r \in [p]$, of $S(x^*)$ then $\{u\} \cup S(x^*)$ is a p -partite clique. Suppose there is a vertex in $S(x^*) \setminus P_r$ that is not adjacent to u , then, recalling (3.4), we have the following inequality:

$$(5.3) \quad \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) < \sum_{D \in \binom{[p] \setminus \{r\}}{k-1}} \prod_{s \in D} \sum_{v \in P_s} x_v^* = \binom{p-1}{k-1} \frac{1}{p^{k-1}}.$$

However, since $\mu_u^{(k)} = 0$, from (4.1) and Corollary 4.6 we have

$$(5.4) \quad \sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) = \lambda^{(k)} = \binom{p-1}{k-1} \frac{1}{p^{k-1}},$$

which contradicts (5.3).

Now assume that u is adjacent to at least one vertex in each part of $S(x^*)$. If u is adjacent to every vertex in $S(x^*)$, then x^* is not a local maximum due to Lemma 4.10.

Therefore, there must exist $r \in [p]$ such that some vertex $w \in P_r$ is adjacent to u , while another vertex $v \in P_r$ is not adjacent to u . Consider $d \in \mathbb{R}^n$ defined as

$$d_i := \begin{cases} -2, & i = v, \\ 1, & i \in \{u, w\}, \\ 0, & \text{otherwise.} \end{cases}$$

Since $\{u, v\}, \{v, w\} \notin E$ and $x_u^* = 0$, we have:

$$\begin{aligned} f_k(x^* + \varepsilon d) &= \sum_{C \in \mathcal{C}_{k;u,v,w}} \pi_C(x^*) + (x_v^* - 2\varepsilon) \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + (x_w^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-1;u}^w} \pi_C(x^*) \\ &\quad + \varepsilon \sum_{C \in \mathcal{C}_{k-1;w}^u} \pi_C(x^*) + \varepsilon(x_w^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*). \end{aligned}$$

Also,

$$f_k(x^*) = \sum_{C \in \mathcal{C}_k} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k;u,v,w}} \pi_C(x^*) + x_v^* \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + x_w^* \sum_{C \in \mathcal{C}_{k-1;u}^w} \pi_C(x^*).$$

Hence,

$$\begin{aligned} f_k(x^* + \varepsilon d) - f_k(x^*) &= -2\varepsilon \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) + \varepsilon \sum_{C \in \mathcal{C}_{k-1;u}^w} \pi_C(x^*) + \varepsilon \sum_{C \in \mathcal{C}_{k-1;w}^u} \pi_C(x^*) \\ &\quad + \varepsilon(x_w^* + \varepsilon) \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*). \end{aligned}$$

Considering that $x_u^* = 0$ and taking into account (4.1), we have

$$\sum_{C \in \mathcal{C}_{k-1;u}^w} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1}^w} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) = \lambda^{(k)}.$$

Also, since $\mu_u^{(k)} = 0$, (4.1) implies that

$$\sum_{C \in \mathcal{C}_{k-1}^u} \pi_C(x^*) = \sum_{C \in \mathcal{C}_{k-1;w}^u} \pi_C(x^*) + x_w^* \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) = \lambda^{(k)}.$$

Thus, using (5.6) and (5.7) in (5.5) we obtain

$$f_k(x^* + \varepsilon d) - f_k(x^*) = -2\varepsilon \lambda^{(k)} + \varepsilon \lambda^{(k)} + \varepsilon \lambda^{(k)} + \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) = \varepsilon^2 \sum_{C \in \mathcal{C}_{k-2}^{u,w}} \pi_C(x^*) > 0.$$

The last inequality holds since u is adjacent to at least one vertex in each part of $S(x^*)$. Patently, (5.8) implies that d is a feasible direction of improvement for $(\mathbf{P}^k)$ at x^* , which contradicts the assumption that x^* is a local maximum.

To establish the second statement of the lemma, note that from the first statement we have $Z_k \neq \emptyset \implies S(x^*)$ is not a maximal p -partite clique. On the other hand, if $S(x^*)$ is not a maximal p -partite clique then there exists $u \in V \setminus S(x^*)$ such that $\{u\} \cup S(x^*)$ is a p -partite clique. From (4.1), $\mu_u^{(k)} = 0$, so $u \in Z_k$ and $Z_k \neq \emptyset$. $\square$

LEMMA 5.2. Z_k can be represented as a disjoint union of $U_s, s \in [p]$, where for each $s \in [p]$, $U_s \cup S(x^*)$ is a p -partite clique in G , with $U_s \cup P_s$ being one of its p parts. Moreover, the restriction of x^* to $S(x^*) \cup Z_k$ is a global maximum of $(\mathbf{P}^k)$ formulated for graph $G[S(x^*) \cup Z_k]$.

Proof. Given $s \in [p]$, let U_s be the set of vertices from Z_k with no neighbors in P_s . Due to Lemma 5.1, each $v \in Z_k$ belongs to exactly one among the sets $U_s, s \in [p]$ and forms a multipartite clique together with $S(x^*) \setminus P_s$. To complete the proof, we need to show that U_s is an independent set. Assume otherwise. Then there exist $u, v \in U_s$ such that $\{u, v\} \in E$ and $\{u, v\} \cup S(x^*) \setminus P_s$ forms a $(p+1)$ -partite clique. Hence, $S(x^*)$ is not a strongly part-maximal multipartite clique, a contradiction with Theorem 4.11. Finally, note that to form a clique, we cannot select vertices from both U_s and P_s . This means that a maximum clique in $G[S(x^*) \cup Z_k]$ has at most p vertices, while x^* corresponds to a multipartite clique with exactly p parts. Therefore, by Theorems 3.1 and 4.4 the restriction of x^* to $S(x^*) \cup Z_k$ is a global maximum of $(\mathbf{P}^k)$ stated for $G[S(x^*) \cup Z_k]$. $\square$

LEMMA 5.3. *Let x^* be a local maximum of $(\mathbf{P}^{k+1})$ for some $k \in \{2, \dots, \omega - 1\}$. Then x^* satisfies the KKT conditions for $(\mathbf{P}^k)$. Moreover, if the KKT multiplier $\mu_i^{(k+1)}$ associated with vertex i for $(\mathbf{P}^{k+1})$ is positive, then so is the KKT multiplier $\mu_i^{(k)}$ for $(\mathbf{P}^k)$, that is, $I_k \subseteq I_{k+1}$.*

Proof. Assume x^* is a local maximum of $(\mathbf{P}^{k+1})$ with the corresponding multipartite clique $S(x^*) = \bigsqcup_{s=1}^p P_s$, where $p \geq k+1$, and the KKT multiplier $\lambda^{(k+1)} = \binom{p-1}{k} \frac{1}{p^k}$ (per Corollary 4.6). Consider any $v \in S(x^*) \cup Z_{k+1}$. According to Lemma 5.2, v belongs to some $P_s \cup U_s$, forming a p -partite clique with $P_r, r \in [p] \setminus \{s\}$. Thus, recalling (3.4), we have

$$\sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) = \sum_{D \in \binom{[p] \setminus \{s\}}{k-1}} \prod_{r \in D} \sum_{v \in P_r} x_v^* = \sum_{D \in \binom{[p] \setminus \{s\}}{k-1}} \frac{1}{p^{k-1}} = \binom{p-1}{k-1} \frac{1}{p^{k-1}}.$$

Hence, the KKT condition (4.1) for $(\mathbf{P}^k)$ clearly holds at x^* for all $v \in I_k$ with $\lambda^{(k)} = \binom{p-1}{k-1} \frac{1}{p^{k-1}}$. To complete the proof, we need to show that for any $v \in V \setminus I_{k+1}$:

$$\mu_v^{(k+1)} = \lambda^{(k+1)} - \sum_{C \in \mathcal{C}_k^v} \pi_C(x^*) > 0 \implies \mu_v^{(k)} = \lambda^{(k)} - \sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) > 0,$$

or, equivalently,

$$\sum_{C \in \mathcal{C}_{k-1}^v} \pi_C(x^*) \geq \lambda^{(k)} \implies \sum_{C \in \mathcal{C}_k^v} \pi_C(x^*) \geq \lambda^{(k+1)}.$$

Let $y^*(v) \in \mathbb{R}^p$ be defined as follows:

$$y_s^*(v) = p \sum_{j \in P_s \cap N_v} x_j^*, \quad s \in [p],$$

where N_v is the set of vertices adjacent to v in G .

Since $\sum_{j \in P_s \cap N_v} x_j^* \leq \sum_{j \in P_s} x_j^* = 1/p$ (by Theorem 4.4), $y_s^*(v) \in [0, 1]^p$. Note that for $t \in \{0, 1\}$ we have:

$$\begin{aligned} \sum_{C \in \mathcal{C}_{k-t}^v} \pi_C(x^*) &= \sum_{D \in \binom{[p]}{k-t}} \prod_{s \in D} \sum_{j \in P_s \cap N_v} x_j^* = \frac{1}{p^{k-t}} \sum_{D \in \binom{[p]}{k-t}} \prod_{s \in D} y_s^*(v) \\ &= \frac{1}{p^{k-t}} e_{k-t,p}(y^*(v)), \end{aligned}$$

where $e_{k-t,p}$ is an elementary symmetric polynomial as defined in (2.2). Now what we want to show becomes

$$(5.9) \quad e_{k-1,p}(y^*(v)) \geq \binom{p-1}{k-1} \implies e_{k,p}(y^*(v)) \geq \binom{p-1}{k}.$$

Since $y_s^*(v) \in [0, 1]^p$, (5.9) follows from Lemma 2.2. $\square$

The proof of the main structural result (Theorem 5.5) will be based on the definition of a local maximum. In addition to the lemmata above, it will use one more technical lemma, established next. Recall the definition of I_k and Y_k^d given in (5.1) and (5.2), respectively.

LEMMA 5.4. *Let x^* be a point satisfying the KKT conditions for $(\mathbf{P}^k)$. Let $\bar{\varepsilon} := \min \{1/(2^n p^{k-2}), 1/(2^{p+1} p^{k-2} n)\}$. Suppose that $\varepsilon \in (0, \bar{\varepsilon}]$ and $d \in \mathbb{R}^n$ is such that $\|d\| < \varepsilon$, $x^* + d$ is feasible for $(\mathbf{P}^k)$, and $Y_k^d \neq \emptyset$. Then*

$$(5.10) \quad \sigma' := \sum_{t=2}^k \sum_{U \in \mathcal{C}_t(I_k)} \pi_U(d) \sum_{C \in \mathcal{C}_{k-t}^U} \pi_C(x^*) \leq \frac{n}{2} \varepsilon \sum_{i \in Y_k^d} d_i.$$

Proof. Since $x^* + d$ is a feasible point,

$$(5.11) \quad \sum_{i \in S(d)} d_i = 0 \quad \text{and} \quad \sum_{i \in I_k} d_i = - \sum_{i \in Y_k^d} d_i.$$

Also, since $\|d\| < \varepsilon$, for $t, q \geq 1$ and any $U \in \mathcal{C}_t(I_k)$, $D \in \mathcal{C}_q(Y_k^d)$ we have

$$(5.12) \quad |\pi_U(d)| \leq \varepsilon^t \leq \varepsilon, \quad 0 \leq \pi_D(d) \leq \varepsilon^q \leq \varepsilon.$$

To derive an upper bound on σ' , we first use the multipartite structure of $S(x^*)$ to write

$$(5.13) \quad \sigma' = \sum_{t=2}^k \sum_{U \in \mathcal{C}_t(I_k)} \pi_U(d) \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t}$$

$$(5.14) \quad = \sum_{U \in \mathcal{C}_2(I_k)} \pi_U(d) \sum_{t=0}^{k-2} \sum_{D \in \mathcal{C}_t^U(I_k)} \pi_D(d) \binom{p-t-2}{k-t-2} \left(\frac{1}{p}\right)^{k-t-2}.$$

For $u, w \in Z_k$, we have no information on whether the edge $\{u, w\}$ is present in $G[I_k]$ if $u \in U_s$, $w \in U_r$, $s \neq r$. However, we will show that by assuming that every such edge is indeed present, we can obtain a valid upper bound on σ' . Indeed, consider $u \in U_s$, $w \in U_r$, $s \neq r$ and let

$$\sigma'[u, w] := d_u d_w \sum_{t=0}^{k-2} \sum_{U \in \mathcal{C}_t^{u,w}(I_k)} \pi_U(d) \binom{p-t-2}{k-t-2} \left(\frac{1}{p}\right)^{k-t-2}.$$

Observe that

$$(5.15) \quad \sigma'[u, w] = d_u d_w \binom{p-2}{k-2} \left(\frac{1}{p}\right)^{k-2} + d_u d_w \sum_{t=1}^{k-2} \sum_{U \in \mathcal{C}_t^{u,w}(I_k)} \pi_U(d) \binom{p-t-2}{k-t-2} \left(\frac{1}{p}\right)^{k-t-2}.$$

Since $u, w \in Z_k$, we have $d_u, d_w \geq 0$ due to the feasibility of $x^* + d$, hence

$$(5.16) \quad d_u d_w \binom{p-2}{k-2} \left(\frac{1}{p}\right)^{k-2} \geq d_u d_w \frac{1}{p^{k-2}}.$$

Also, due to (5.12) and Lemma 2.1,
(5.17)

$$(5.17) \quad d_u d_w \sum_{t=1}^{k-2} \sum_{U \in \mathcal{C}_t^{u,w}(I_k)} \pi_U(d) \binom{p-t-2}{k-t-2} \left(\frac{1}{p}\right)^{k-t-2} \geq -d_u d_w \sum_{t=1}^{k-2} \sum_{U \in \mathcal{C}_t^{u,w}(I_k)} \varepsilon \geq -d_u d_w 2^n \varepsilon.$$

Using (5.16) and (5.17) in (5.15), a lower bound on $\sigma'[u, w]$ can be given as

$$(5.18) \quad \sigma'[u, w] \geq d_u d_w \frac{1}{p^{k-2}} - d_u d_w 2^n \varepsilon = d_u d_w \left[\frac{1}{p^{k-2}} - 2^n \varepsilon \right] \geq 0,$$

assuming

$$(5.19) \quad \varepsilon \leq \frac{1}{2^n p^{k-2}}.$$

Therefore, if the edge $\{u, v\}$ is missing, we cannot decrease the value of $\sigma'[u, w]$ by assuming the edge is present, as long as (5.19) holds. This implies that we can get an upper bound on σ' by assuming $\bigsqcup_{s=1}^p (P_s \cup U_s)$ to be a multipartite clique with the parts $P'_s = P_s \cup U_s$. Hence, denoting $d'_s := \sum_{j \in P'_s} d_j$, we can assume that I_k is a multipartite clique in (5.13) to obtain an upper bound

$$(5.20) \quad \sigma' \leq \sum_{t=2}^k \sum_{I \in \binom{[p]}{t}} \prod_{s \in I} \sum_{j \in P'_s} d_j \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t} = \sum_{t=2}^k e_{t,p}(d') \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t},$$

where $e_{t,p}(d')$ is an elementary symmetric polynomial as defined in (2.2). Now observe that

$$\sum_{s \in S(d')} d'_s = \sum_{s \in [p]} d'_s = \sum_{s \in [p]} \sum_{j \in P_s \cup U_s} d_j = \sum_{j \in I_k} d_j.$$

Hence, in view of (5.11) we have

$$(5.21) \quad \sum_{s \in S(d')} d'_s = - \sum_{i \in Y_k^d} d_i.$$

Also, $0 \leq d_i \leq \varepsilon \forall i \in Y_k^d$ and the definition (5.2) of Y_k^d imply that $0 < \sum_{i \in Y_k^d} d_i \leq n\varepsilon$, thus, we obtain

$$(5.22) \quad \begin{aligned} 2e_{2,p}(d') &= \sum_{s \in S(d')} d'_s \sum_{r \in S(d') \setminus \{s\}} d'_r = \sum_{s \in S(d')} d'_s \left(- \sum_{i \in Y_k^d} d_i - d'_s \right) \\ &= - \left(\sum_{s \in S(d')} d'_s \right) \sum_{i \in Y_k^d} d_i - \sum_{s \in S(d')} d_s'^2 = \left(\sum_{i \in Y_k^d} d_i \right)^2 - \sum_{s \in S(d')} d_s'^2 \\ &\leq \sum_{i \in Y_k^d} d_i n\varepsilon - \sum_{s \in S(d')} d_s'^2. \end{aligned}$$

On the other hand, any term in $e_{t,p}(d')$, where $t > 3$, can be upper bounded by $n\varepsilon \max_{s \in S(d')} d_s'^2$, so using Lemma 2.1 we have

$$(5.23) \quad \sum_{t=3}^k e_{t,p}(d') \binom{p-t}{k-t} \left(\frac{1}{p}\right)^{k-t} \leq \sum_{t=3}^k |e_{t,p}(d')| \leq 2^p n\varepsilon \max_{s \in S(d')} d_s'^2.$$

Therefore, using (5.22) and (5.23) in (5.20), we obtain:

$$(5.24) \quad \sigma' \leq \frac{1}{2} \binom{p-2}{k-2} \left(\frac{1}{p}\right)^{k-2} \left(\sum_{i \in Y_k^d} d_i n\varepsilon - \sum_{s \in S(d')} d_s'^2 \right) + 2^p n\varepsilon \max_{s \in S(d')} d_s'^2.$$

Note that $\max_{s \in S(d')} d_s'^2$ is a part of $\sum_{s \in S(d')} d_s'^2$. Hence,

$$(5.25) \quad \varepsilon \leq \frac{1}{2^{p+1} p^{k-2} n} \implies 2^p n\varepsilon \leq \frac{1}{2} \binom{p-2}{k-2} \left(\frac{1}{p}\right)^{k-2}$$

in (5.24) yields the desired upper bound:

$$(5.26) \quad \sigma' \leq \frac{1}{2} \binom{p-2}{k-2} \left(\frac{1}{p}\right)^{k-2} \sum_{i \in Y_k^d} d_i n\varepsilon \leq \frac{n}{2} \varepsilon \sum_{i \in Y_k^d} d_i. \quad \square$$

We are now ready to establish the main structural result.

THEOREM 5.5. *Suppose x^* is a local maximum of $(\mathbf{P}^{k+1})$ for some $k \in \{2, \dots, \omega - 1\}$. Then x^* is a local maximum of $(\mathbf{P}^k)$.*

Proof. Since x^* is a local maximum for $(\mathbf{P}^{k+1})$, by Theorem 4.11, $S(x^*)$ is a strongly part-maximal multipartite clique $S(x^*) = \bigsqcup_{s=1}^p P_s$, where $p \geq k+1$. Also, due to Lemma 5.3, there exist $\lambda^{(k)}, \mu^{(k)}$ satisfying KKT conditions for x^* relative to $(\mathbf{P}^k)$ and $I_k \subseteq I_{k+1}$. We will establish the statement of the theorem by showing that there exists a sufficiently small $\varepsilon > 0$ such that $f_k(x') \leq f_k(x^*)$ for all feasible x' with $\|x' - x^*\| < \varepsilon$, where $\|\cdot\|$ denotes the standard Euclidean norm. First, note that selecting

$$(5.27) \quad \varepsilon < \min_{i \in S(x^*)} x_i^*$$

guarantees that $S(x^*) \subseteq S(x')$. Let us define $d := x' - x^*$.

Consider

$$(5.28) \quad f_k(x') - f_k(x^*) = \sum_{C \in \mathcal{C}_k} \pi_C(x^* + d) - \sum_{C \in \mathcal{C}_k} \pi_C(x^*)$$

$$(5.29) \quad = \sum_{t=1}^k \sum_{D \in \mathcal{C}_t} \pi_D(d) \sum_{C \in \mathcal{C}_{k-t}^D} \pi_C(x^*)$$

$$(5.30) \quad = \sum_{i \in S(d)} d_i \sum_{C \in \mathcal{C}_{k-1}^i} \pi_C(x^*) + \sum_{t=2}^k \sum_{D \in \mathcal{C}_t} \pi_D(d) \sum_{C \in \mathcal{C}_{k-t}^D} \pi_C(x^*)$$

$$(5.31) \quad = \sum_{i \in S(d)} d_i (\lambda^{(k)} - \mu_i^{(k)}) + \sum_{t=2}^k \sum_{D \in \mathcal{C}_t} \pi_D(d) \sum_{C \in \mathcal{C}_{k-t}^D} \pi_C(x^*)$$

$$(5.32) \quad = \sum_{i \in S(d)} d_i (\lambda^{(k)} - \mu_i^{(k)}) + \sigma,$$

619 where

$$620 \quad (5.33) \quad \sigma := \sum_{t=2}^k \sum_{D \in \mathcal{C}_t} \pi_D(d) \sum_{C \in \mathcal{C}_{k-t}^D} \pi_C(x^*).$$

621 Here (5.31) follows from (4.1), since the KKT conditions hold. Note that $d_i < 0$
 622 implies that $x_i > 0$ and, subsequently, $\mu_i^{(k)} = 0$. Hence, considering (5.11),

$$623 \quad (5.34) \quad \sum_{i \in S(d)} d_i (\lambda^{(k)} - \mu_i^{(k)}) = \lambda^{(k)} \sum_{i \in S(d)} d_i - \sum_{i \in Y_k^d} d_i \mu_i^{(k)} = - \sum_{i \in Y_k^d} d_i \mu_i^{(k)} \leq 0.$$

624 The inequality in (5.34) is satisfied at equality if and only if $Y_k^d = \emptyset$. In this case
 625 $d_i = 0$ for all i such that $\mu_i^{(k)} > 0$ and hence $S(x') \subseteq I_k$. Due to Lemma 5.2 applied
 626 for $(\mathbf{P}^{k+1})$, any clique in $G[I_{k+1}]$ has at most p vertices. This, along with the fact
 627 that $I_k \subseteq I_{k+1}$, implies that $f_k(x') \leq f_k(x^*)$. Hence, in the remainder of the proof
 628 we assume that the inequality in (5.34) is strict, i.e., $Y_k^d \neq \emptyset$.

629 Next we analyze the expression (5.33) for σ , with the aim of proving that the
 630 expression in (5.32) is non-positive for a sufficiently small ε . We rewrite the sum
 631 in (5.33) by assuming that q of the vertices in a clique D are in Y_k^d for $q = 0, \dots, k$,
 632 and the remaining vertices are in I_k . We obtain

$$633 \quad (5.35) \quad \sigma = \sum_{t=2}^k \sum_{D \in \mathcal{C}_t} \pi_D(d) \sum_{C \in \mathcal{C}_{k-t}^D} \pi_C(x^*) = \sigma' + \sigma'' + \sigma''',$$

634 where σ' corresponds to the case where all vertices in D are from I_k , σ'' corresponds
 635 to the case where all vertices in D are from Y_k^d , and σ''' describes the case where D
 636 contains at least one vertex from I_k and Y_k^d , i.e.,

$$637 \quad (5.36) \quad \sigma' := \sum_{t=2}^k \sum_{U \in \mathcal{C}_t(I_k)} \pi_U(d) \sum_{C \in \mathcal{C}_{k-t}^U} \pi_C(x^*),$$

$$638 \quad (5.37) \quad \sigma'' := \sum_{q=2}^k \sum_{D \in \mathcal{C}_q(Y_k^d)} \pi_D(d) \sum_{C \in \mathcal{C}_{k-q}^D} \pi_C(x^*),$$

$$639 \quad (5.38) \quad \sigma''' := \sum_{q=1}^{k-1} \sum_{t=1}^{k-q} \sum_{D \in \mathcal{C}_q(Y_k^d)} \pi_D(d) \sum_{U \in \mathcal{C}_t^D(I_k)} \pi_U(d) \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*).$$

640 By Lemma 5.4,

$$641 \quad (5.39) \quad \sigma' \leq \frac{n}{2} \varepsilon \sum_{i \in Y_k^d} d_i.$$

642 for a sufficiently small $\varepsilon > 0$. Next, we establish upper bounds on σ'' and σ''' defined
 643 in (5.37) and (5.38), respectively. Note that since $0 \leq d_i \leq \varepsilon \forall i \in Y_k^d$, we have:

$$644 \quad (5.40) \quad \pi_D(d) \leq \sum_{i \in Y_k^d} d_i, \quad \forall D \in \mathcal{C}_q(Y_k^d), \quad q \geq 1.$$

Using (5.12) and (5.40), we obtain the following bounds on σ'' and σ''' , respectively:

$$\begin{aligned}
 \sigma'' &= \sum_{q=2}^k \sum_{D \in \mathcal{C}_q(Y_k^d)} \pi_D(d) \sum_{C \in \mathcal{C}_{k-q}^D} \pi_C(x^*) \\
 &\leq \sum_{i \in Y_k^d} d_i \left(\sum_{q=1}^{k-1} \sum_{D \in \mathcal{C}_q^i(Y_k^d)} \pi_D(d) \sum_{C \in \mathcal{C}_{k-q}^{\{i\} \cup D}} \pi_C(x^*) \right) \\
 &\leq \sum_{q=1}^{k-1} \sum_{i \in Y_k^d} d_i \sum_{D \in \mathcal{C}_q^i(Y_k^d)} \varepsilon \sum_{C \in \mathcal{C}_{k-q}^{\{i\} \cup D}} \pi_C(x^*) \\
 &\leq \varepsilon \sum_{q=1}^{k-1} \sum_{i \in Y_k^d} d_i \sum_{D \in \mathcal{C}_q(Y_k^d)} \sum_{C \in \mathcal{C}_{k-q}^D} \pi_C(x^*) \\
 &= \varepsilon \sum_{q=1}^{k-1} \sum_{t=0}^0 \sum_{D \in \mathcal{C}_q(Y_k^d)} \sum_{i \in Y_k^d} d_i \sum_{U \in \mathcal{C}_t^D(I_k)} \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*),
 \end{aligned}$$

645

$$\begin{aligned}
 \sigma''' &= \sum_{q=1}^{k-1} \sum_{t=1}^{k-q} \sum_{D \in \mathcal{C}_q(Y_k^d)} \pi_D(d) \sum_{U \in \mathcal{C}_t^D(I_k)} \pi_U(d) \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*) \\
 &\leq \sum_{q=1}^{k-1} \sum_{t=1}^{k-q} \sum_{D \in \mathcal{C}_q(Y_k^d)} \pi_D(d) \sum_{U \in \mathcal{C}_t^D(I_k)} \varepsilon \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*) \\
 &\leq \varepsilon \sum_{q=1}^{k-1} \sum_{t=1}^{k-q} \sum_{D \in \mathcal{C}_q(Y_k^d)} \sum_{i \in Y_k^d} d_i \sum_{U \in \mathcal{C}_t^D(I_k)} \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*).
 \end{aligned}$$

646

647

648

649 Hence,

$$(5.41) \quad \sigma'' + \sigma''' \leq \varepsilon \sum_{q=1}^{k-1} \sum_{t=0}^{k-q} \sum_{D \in \mathcal{C}_q(Y_k^d)} \sum_{i \in Y_k^d} d_i \sum_{U \in \mathcal{C}_t^D(I_k)} \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*)$$

651 Also, due to Lemma 2.1, and Theorem 4.4 we have

$$(5.42) \quad \sum_{C \in \mathcal{C}_{k-q-t}^{D \cup U}} \pi_C(x^*) \leq \binom{p-q-t}{k-q-t} \left(\frac{1}{p} \right)^{k-q-t} \leq 1.$$

653 Using (5.42) in (5.41) and considering that the number of different subsets of the set
 654 of n nodes is bounded from above by 2^n , we obtain

$$\begin{aligned}
 \sigma'' + \sigma''' &\leq \varepsilon \sum_{i \in Y_k^d} d_i \sum_{q=1}^{k-1} \sum_{D \in \mathcal{C}_q(Y_k^d)} \sum_{t=0}^{k-q} \sum_{U \in \mathcal{C}_t^D(I_k)} 1 \\
 &\leq \varepsilon \sum_{i \in Y_k^d} d_i \sum_{q=1}^{k-1} \sum_{D \in \mathcal{C}_q(Y_k^d)} 2^n \leq 4^n \varepsilon \sum_{i \in Y_k^d} d_i.
 \end{aligned}$$

655

From (5.35), (5.39), and (5.43) we obtain the following upper bound on σ :

$$(5.44) \quad \sigma \leq \left(\frac{n}{2} + 4^n\right) \varepsilon \sum_{i \in Y_k^d} d_i.$$

Therefore, considering (5.32) and (5.34),

$$f_k(x') - f_k(x^*) \leq - \sum_{i \in Y_k^d} d_i \mu_i^{(k)} + \left(\frac{n}{2} + 4^n\right) \varepsilon \sum_{i \in Y_k^d} d_i = - \sum_{i \in Y_k^d} d_i \left(\mu_i^{(k)} - \left(4^n + \frac{n}{2}\right) \varepsilon\right),$$

which is guaranteed to be negative by choosing ε such that

$$(5.45) \quad \mu_i^{(k)} > \varepsilon \left(4^n + \frac{n}{2}\right) \quad \forall i \in Y_k^d \iff \varepsilon < \min_{i \in V \setminus I_k} \frac{\mu_i^{(k)}}{4^n + n/2}.$$

Finally, by combining all the conditions for ε (i.e., (5.27), (5.19), (5.25), (5.45)), for

$$\varepsilon < \min \left\{ \min_{i \in S(x^*)} x_i^*, \frac{1}{2^n p^{k-2}}, \frac{1}{2^{p+1} p^{k-2} n}, \min_{i \in V \setminus I_k} \frac{\mu_i^{(k)}}{4^n + n/2} \right\},$$

x^* satisfies the definition of a local maximum for $(\mathbf{P}^k)$. $\square$

6. Conclusion. We introduced a hierarchy of standard polynomial programming formulations for the maximum clique problem. Namely, given a graph $G = (V, E)$ and $k \in \{2, \dots, \omega\}$, where ω is the clique number of the graph, the maximum clique problem is formulated as a problem $(\mathbf{P}^k)$ of maximizing a multilinear polynomial of degree k over the standard simplex in $\mathbb{R}^{|V|}$. We have shown that the support of a local maximum of each formulation corresponds to a strongly part-maximal multipartite clique in G . Moreover, the local maximality of a point for $(\mathbf{P}^{k+1})$ implies its local maximality for $(\mathbf{P}^k)$. These results invite several interesting directions for future research. In particular, developing a regularized version of the proposed formulations, guaranteeing one-to-one correspondence between local maxima of the formulation and maximal cliques in the graph would be a natural extension of this work. Such results for the Motzkin-Straus formulation and its generalization to s -defective clique were established in [5, 19] and [4], respectively. Also, exploring computational aspects of the proposed formulations, especially with low values of k , could be promising. Our optimism is supported by the proven success of algorithmic approaches utilizing the Motzkin-Straus formulation [5, 11, 14, 15, 19, 28], as well as excellent performance of local optimization methods on cubic formulations for the closely related maximum s -defective clique and maximum s -plex problems, recently reported in [33, 4]. We have obtained some preliminary results along these lines and are currently working on extensive computational experiments, exploiting the proposed formulations, their regularizations, and combinations of thereof. Once finalized, our findings will be reported in a followup paper.

Acknowledgments. We would like to thank the three anonymous referees for their helpful comments and suggestions.

REFERENCES

- [1] A. AHMADI, A. OLSHEVSKY, P. PARRILO, AND J. TSITSIKLIS, *NP-hardness of deciding convexity of quartic polynomials and related problems*, Mathematical Programming, 137 (2013), pp. 453–476.

- [2] M. AIGNER AND G. M. ZIEGLER, *Proofs from THE BOOK, Sixth Edition*, Springer, Berlin, Heidelberg, 2018.
- [3] E. BALAS, V. CHVÁTAL, AND J. NEŠETŘIL, *On the maximum weight clique problem*, Mathematics of Operations Research, 12 (1987), pp. 522–535.
- [4] I. BOMZE, F. RINALDI, AND D. ZEFFIRO, *Fast cluster detection in networks by first-order optimization*, SIAM Journal on Mathematics of Data Science, 4 (2022), pp. 285–305.
- [5] I. M. BOMZE, *Evolution towards the maximum clique*, Journal of Global Optimization, 10 (1997), pp. 143–164.
- [6] I. M. BOMZE, *On standard quadratic optimization problems*, Journal of Global Optimization, 13 (1998), pp. 369–387.
- [7] I. M. BOMZE, M. BUDINICH, P. M. PARDALOS, AND M. PELILLO, *The maximum clique problem*, in Handbook of Combinatorial Optimization, D.-Z. Du and P. M. Pardalos, eds., Kluwer Academic Publishers, 1999, pp. 1–74.
- [8] I. M. BOMZE, M. DÜR, E. DE KLERK, C. ROOS, A. J. QUIST, AND T. TERLAKY, *On copositive programming and standard quadratic optimization problems*, Journal of Global Optimization, 18 (2000), pp. 301–320.
- [9] I. M. BOMZE, M. KAHR, AND M. LEITNER, *Trust your data or not – StQP remains StQP: Community detection via robust standard quadratic optimization*, Mathematics of Operations Research, 46 (2021), pp. 301–316.
- [10] P. BONAMI, A. LODI, J. SCHWEIGER, AND A. TRAMONTANI, *Solving quadratic programming by cutting planes*, SIAM Journal on Optimization, 29 (2019), pp. 1076–1105.
- [11] S. BUSYGIN, *A new trust region technique for the maximum weight clique problem*, Discrete Applied Mathematics, 154 (2006), pp. 2080–2096.
- [12] E. DE KLERK AND D. V. PASECHNIK, *Approximation of the stability number of a graph via copositive programming*, SIAM Journal on Optimization, 12 (2002), pp. 875–892.
- [13] D. C. FISHER AND J. RYAN, *Bounds on the number of complete subgraphs*, Discrete Mathematics, 103 (1992), pp. 313–320.
- [14] L. E. GIBBONS, D. W. HEARN, AND P. M. PARDALOS, *A continuous based heuristic for the maximum clique problem*, in Cliques, Coloring and Satisfiability: Second DIMACS Implementation Challenge, D. S. Johnson and M. A. Trick, eds., vol. 26 of DIMACS Series on Discrete Mathematics and Theoretical Computer Science, American Mathematical Society, 1996, pp. 103–124.
- [15] L. E. GIBBONS, D. W. HEARN, P. M. PARDALOS, AND M. V. RAMANA, *Continuous characterizations of the maximum clique problem*, Mathematics of Operations Research, 22 (1997), pp. 754–768.
- [16] Y. GÖKMEN AND E. YILDIRIM, *On standard quadratic programs with exact and inexact doubly nonnegative relaxations*, Mathematical Programming, 193 (2022), pp. 365–403.
- [17] J. HÅSTAD, *Clique is hard to approximate within $n^{1-\epsilon}$* , Acta Mathematica, 182 (1999), pp. 105–142.
- [18] C. HOEDE AND X. LI, *Clique polynomials and independent set polynomials of graphs*, Discrete Mathematics, 125 (1994), pp. 219–228.
- [19] J. T. HUNGERFORD AND F. RINALDI, *A general regularized continuous formulation for the maximum clique problem*, Mathematics of Operations Research, 44 (2019), pp. 1161–1173.
- [20] R. M. KARP, *Reducibility among combinatorial problems*, in Complexity of Computer Computations, R. E. Miller and J. W. Thatcher, eds., Plenum, New York, 1972, pp. 85–103.
- [21] J. B. LASSERRE, *Global optimization with polynomials and the problem of moments*, SIAM Journal on Optimization, 11 (2001), pp. 796–817.
- [22] C. LING, J. NIE, L. QI, AND Y. YE, *Biquadratic optimization over unit spheres and semidefinite programming relaxations*, SIAM Journal on Optimization, 20 (2010), pp. 1286–1310.
- [23] G. LIUZZI, M. LOCATELLI, AND V. PICCIALLI, *A new branch-and-bound algorithm for standard quadratic programming problems*, Optimization Methods and Software, 34 (2019), pp. 79–97.
- [24] L. LOVÁSZ AND A. SCHRIJVER, *Cones of matrices and set-functions and 0–1 optimization*, SIAM Journal on Optimization, 1 (1991), pp. 166–190.
- [25] J. MOON AND L. MOSER, *On cliques in graphs*, Israel Journal of Mathematics, 3 (1965), pp. 23–28.
- [26] T. S. MOTZKIN AND E. G. STRAUS, *Maxima for graphs and a new proof of a theorem of Turán*, Canad. J. Math., 17 (1965), pp. 533–540.
- [27] C. P. NICULESCU, *A new look at Newton’s inequalities*, Journal of Inequalities in Pure and Applied Mathematics, 1 (2000).
- [28] P. M. PARDALOS AND A. T. PHILLIPS, *A global optimization approach for solving the maximum clique problem*, International Journal of Computer Mathematics, 33 (1990), pp. 209–216.

- [29] M. PELILLO AND A. JAGOTA, *Feasible and infeasible maxima in a quadratic program for maximum clique*, Journal of Artificial Neural Networks, 2 (1996), pp. 411–420.
- [30] Y. PENG, H. PENG, Q. TANG, AND C. ZHAO, *An extension of the Motzkin–Straus theorem to non-uniform hypergraphs and its applications*, Discrete Applied Mathematics, 200 (2016), pp. 170–175.
- [31] S. ROTA BULÒ AND M. PELILLO, *A generalization of the Motzkin–Straus theorem to hypergraphs*, Optimization Letters, 3 (2009), pp. 287–295.
- [32] H. D. SHERALI AND W. P. ADAMS, *A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems*, SIAM Journal on Discrete Mathematics, 3 (1990), pp. 411–430.
- [33] V. STOZHKOVA, A. BUCHANAN, S. BUTENKO, AND V. BOGINSKI, *Continuous cubic formulations for cluster detection problems in networks*, Mathematical Programming, (2020). To appear, DOI: 10.1007/s10107-020-01572-4.
- [34] P. TURÁN, *On an extremal problem in graph theory*, Matematikai és Fizikai Lapok, 48 (1941), pp. 436–452. In Hungarian.
- [35] S. A. VAVASIS, *Nonlinear Optimization: Complexity Issues*, Oxford University Press, Inc., New York, NY, USA, 1991.
- [36] J. WALTEROS AND A. BUCHANAN, *Why is maximum clique often easy in practice?*, Operations Research, 68 (2020), pp. 1866–1895.
- [37] D. ZUCKERMAN, *Linear degree extractors and the inapproximability of max clique and chromatic number*, Theory of Computing, 3 (2007), pp. 103–128.



Continuous cubic formulations for cluster detection problems in networks

Vladimir Stozhkov¹ · Austin Buchanan²  · Sergiy Butenko³ · Vladimir Boginski⁴

Received: 19 March 2020 / Accepted: 23 September 2020

© Springer-Verlag GmbH Germany, part of Springer Nature and Mathematical Optimization Society 2020

Abstract

The celebrated Motzkin–Straus formulation for the maximum clique problem provides a nontrivial characterization of the clique number of a graph in terms of the maximum value of a nonconvex quadratic function over a standard simplex. It was originally developed as a way of proving Turán’s theorem in graph theory, but was later used to develop competitive algorithms for the maximum clique problem based on continuous optimization. Clique relaxations, such as s -defective clique and s -plex, emerged as attractive, more practical alternatives to cliques in network-based cluster detection models arising in numerous applications. This paper establishes continuous cubic formulations for the maximum s -defective clique problem and the maximum s -plex problem by generalizing the Motzkin–Straus formulation to the corresponding clique relaxations. The formulations are used to extend Turán’s theorem and other known lower bounds on the clique number to the considered clique relaxations. Results of preliminary numerical experiments with the CONOPT solver demonstrate that the proposed formulations can be used to quickly compute high-quality solutions for the maximum s -defective clique problem and the maximum s -plex problem. The proposed formulations can also be used to generate interesting test instances for global optimization solvers.

Keywords Clique relaxations · Maximum s -defective clique problem · Maximum s -plex problem · Motzkin–Straus formulation · Turán’s theorem · Cubic optimization · Fractional b -matching

Mathematics Subject Classification 90C26 · 90C27 · 90C35 · 05C35 · 05C50 · 05C69

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s10107-020-01572-4>) contains supplementary material, which is available to authorized users.

Extended author information available on the last page of the article

1 Introduction

Let $G = (V, E)$ be a simple (undirected) graph with vertex set $V = \{1, 2, \dots, n\}$ and edge set $E \subseteq \binom{V}{2}$ consisting of m unordered pairs of adjacent vertices. A subset of vertices $C \subseteq V$ is called a clique if $\{i, j\} \in E$ for every pair of distinct vertices $i, j \in C$. The maximum clique problem is to find a clique of largest cardinality in a graph, and the clique number $\omega(G)$ of G is the cardinality of a maximum clique in G .

Turán [44] established the following inequality between the number of edges m , the number of vertices n , and the clique number $\omega(G)$ of a simple graph G :

$$m \leq \left(1 - \frac{1}{\omega(G)}\right) \frac{n^2}{2}. \quad (1)$$

This inequality, known as Turán's theorem, is one of the most prominent results in extremal graph theory [2].

Motzkin and Straus [33] discovered a nontrivial connection between the clique number of a simple graph G and the global optimal value of a quadratic program associated with G :

$$1 - \frac{1}{\omega(G)} = \max_{x \in \Delta^n} x^T A_G x, \quad (2)$$

where A_G is the adjacency matrix of G , $\Delta^n = \{x \in \mathbb{R}^n \mid e_n^T x = 1, x \geq 0\}$ is the standard simplex in $\mathbb{R}^n$, and e_n is the vector of n ones. Turán's theorem can be proved by setting $x = \frac{1}{n} e_n$ in the Motzkin–Straus formulation. This proof is among the “Proofs from THE BOOK” [3], containing some of the most elegant mathematical proofs.

Pardalos and Phillips [35] proposed the first algorithmic approach for the maximum clique problem based on the Motzkin–Straus formulation. This inspired some interesting developments in the interface of discrete and continuous optimization, aiming to use properties of this (continuous) model in order to solve the (discrete) maximum clique problem [10,13,19,20,24]. In addition, the Motzkin–Straus formulation influenced several other research directions in mathematical optimization, including quadratic programming [12,27], copositive programming [9,11,14], and complexity analysis in nonlinear optimization [1,45].

In this paper, the Motzkin–Straus formulation is extended to two clique relaxation models, s -defective clique and s -plex, defined next. Given a simple graph $G = (V, E)$ and vertex subset $S \subseteq V$, denote by $G[S] = (S, E \cap \binom{S}{2})$ the subgraph induced by S in G . Let s be a fixed positive integer. A subset of vertices $S \subseteq V$ is

- an s -defective clique if $G[S]$ has at least $\binom{|S|}{2} - s$ edges;
- an s -plex if each vertex in S is adjacent to at least $|S| - s$ other vertices from S .

A clique is equivalent to a 0-defective clique and to a 1-plex. Given a graph G and an integer s , the maximum s -defective clique problem asks for an s -defective clique of maximum cardinality $\omega_s^d(G)$ in G , and $\omega_s^d(G)$ is called the s -defective clique number. The maximum s -plex problem and the s -plex number $\omega_s(G)$ are defined similarly.

These and other clique relaxations were introduced in the literature in order to overcome practical limitations of using cliques for modeling tightly knit clusters in networks [36]. In particular, s -defective cliques were introduced in the context of protein interaction networks [47], and s -plexes originally appeared in the social networks literature [39]. Existing approaches for the maximum s -defective clique problem and the maximum s -plex problem, as well as their analogues in the complement graph, include algorithms based on integer linear programming [6,30,41] and combinatorial branch-and-bound methods [22,29,32,43]. In addition, several metaheuristics for the maximum s -plex problem have been proposed [31,34,48].

In this work, we show that the Motzkin–Straus formulation can be generalized to formulate the considered optimization problems in terms of maximizing a cubic function over a polyhedral set described by two types of decision variables, x —corresponding to the graph’s vertices, and y —representing the edges of the complement graph $\bar{G}$. In both cases, the trilinear objective function is quadratic with respect to the vertex variables x and linear with respect to the auxiliary variables y . Also, the projection of the feasible set onto x is the standard simplex Δ^n , while the y variables indicate whether the corresponding missing edge is a part of the subgraph induced by the desired clique relaxation.

The proposed formulations can be thought of as a formalization of the following two-stage process. First, we pick a subset of missing edges that would be feasible to add to G with respect to the considered clique relaxation (i.e., up to s edges for s -defective clique and up to $s - 1$ incident edges per vertex for s -plex). We will refer to such missing edges as *fake edges*. Next, we write the Motzkin–Straus formulation for the maximum clique problem in the graph obtained from G by including the fake edges picked in the first stage. The described process involves binary decisions concerning the missing edges, represented by y -variables. We show that the binary variables can be relaxed to obtain a continuous formulation for each problem. The correctness of the continuous formulation is easy to establish for s -defective clique, but is rather nontrivial for s -plex. In the latter case, we take advantage of a classical result concerning the half-integrality of the fractional b -matching polytope, given in Lemma 1 below.

Given a simple graph $G = (V, E)$ and a nonnegative integer vector $b \in \mathbb{Z}_+^n$, $M \subseteq E$ is called a (1-capacitated) b -matching in G if no vertex $i \in V$ is incident to more than b_i edges in M . The maximum b -matching problem, which is to find a b -matching with the largest number of edges in G (whose cardinality is denoted by $\mu_b(G)$), is polynomial-time solvable [18]. Note that for $b = e_n$ this problem becomes the maximum matching problem. Given a weight w_{ij} associated with each edge $\{i, j\} \in E$, the maximum weight b -matching problem is to find a b -matching with the largest total weight in G . Let $N_G(i) = \{j \in V \mid \{i, j\} \in E\}$ denote the neighborhood of i in G . Then the maximum-weight b -matching problem can be formulated as the following 0-1 program, where y_{ij} and y_{ji} refer to the same variable

and $w_{ij} = w_{ji}$:

$$\begin{aligned} \max \quad & \sum_{\{i,j\} \in E} w_{ij} y_{ij} \\ \text{s.t.} \quad & \sum_{j \in N_G(i)} y_{ij} \leq b_i \quad \forall i \in V \\ & y_{ij} \in \{0, 1\} \quad \forall \{i, j\} \in E. \end{aligned} \quad (3)$$

Clearly, the linear programming (LP) relaxation of this formulation, obtained by replacing $y_{ij} \in \{0, 1\}$ with $y_{ij} \in [0, 1]$ in (3), may have a fractional optimal solution. Consider, for example, the maximum matching problem (i.e., the case where $b = e_n$ and $w_{ij} = 1$ for every edge $\{i, j\} \in E$) on $G = C_5$, the cycle on 5 vertices. Then the optimal solution of the LP relaxation is obtained by setting $y_{ij} = \frac{1}{2}$ for every edge $\{i, j\} \in E$. This observation led to the following well-known result (see, e.g., [26, pp. 44–46]).

Lemma 1 *Let P be the feasible region of the LP relaxation of (3). Any extreme point $\hat{y}$ of P is half-integral, i.e., $\hat{y}_{ij} \in \{0, \frac{1}{2}, 1\}$, $\forall \{i, j\} \in E$. Moreover, the edge-induced subgraph $G\langle F \rangle$, where $F = \{\{i, j\} \in E \mid \hat{y}_{ij} = \frac{1}{2}\}$, consists of disjoint odd cycles.*

Here, the *edge-induced subgraph* $G\langle F \rangle = (V_F, F)$ has F as its edge set while its vertex set $V_F \subseteq V$ includes all vertices incident to the edges of F .

The remainder of this paper is organized as follows. Section 2 establishes the generalizations of Motzkin–Straus formulation to the maximum s -defective clique problem and to the maximum s -plex problem. Section 3 generalizes Turán’s theorem and some known lower bounds on the clique number to s -defective clique and s -plex. Section 4 reports the results of numerical experiments. Finally, Sect. 5 concludes the paper.

2 Generalizations of the Motzkin–Straus formulation

Given a simple graph $G = (V, E)$, let $\bar{G} = (V, \bar{E})$ denote its complement and let $\bar{m} = |\bar{E}| = \binom{n}{2} - m$. For each missing edge $\{i, j\} \in \bar{E}$, we introduce a binary variable y_{ij} , indicating whether both i and j are included in the clique relaxation structure sought. Let $A_{\bar{G}}(y)$ be the matrix with the entry in the i^{th} row and j^{th} column given by y_{ij} if $\{i, j\} \in \bar{E}$, and 0, otherwise. That is, $A_{\bar{G}}(y)$ is obtained from the adjacency matrix $A_{\bar{G}}$ of $\bar{G}$ by replacing each 1 entry with the corresponding y_{ij} . See Fig. 1 for an illustration. We refer to the variables y_{ij} , for $\{i, j\} \in \bar{E}$, as the *fake edge variables*.

2.1 Motzkin–Straus formulation for the maximum s -defective clique problem

Consider the following mixed integer optimization problem.

$$f_s^d(G) = \max_{(x,y) \in \Delta^n \times D_s(G)} f_G(x, y), \quad (4)$$

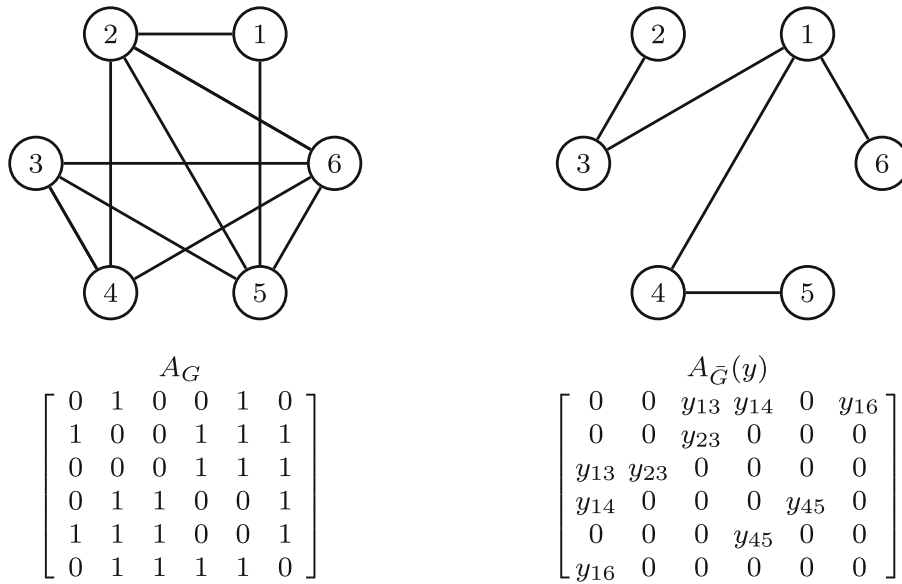


Fig. 1 Illustration of $A_{\bar{G}}(y)$ construction

where

$$f_G(x, y) = x^T (A_G + A_{\bar{G}}(y))x, \quad (5)$$

$\Delta^n = \{x \in \mathbb{R}^n \mid e_n^T x = 1, x \geq 0\}$ is the standard simplex in $\mathbb{R}^n$, and

$$D_s(G) = \{y \in \{0, 1\}^{\bar{E}} \mid e_m^T y \leq s\}. \quad (6)$$

Proposition 1 Let $\omega_s^d(G)$ denote the s -defective clique number of G . Then

$$1 - \frac{1}{\omega_s^d(G)} = f_s^d(G). \quad (7)$$

Proof First, we show that $1 - \frac{1}{\omega_s^d(G)} \leq f_s^d(G)$. Suppose that S is a maximum s -defective clique of cardinality $k = \omega_s^d(G)$, and define $(\hat{x}, \hat{y})$ as follows:

$$\hat{x}_i = \begin{cases} \frac{1}{k}, & \text{if } i \in S, \\ 0, & \text{otherwise;} \end{cases} \quad \hat{y}_{ij} = \begin{cases} 1, & \text{if } i, j \in S \text{ and } \{i, j\} \in \bar{E}, \\ 0, & \text{otherwise.} \end{cases}$$

Then $(\hat{x}, \hat{y}) \in \Delta^n \times D_s(G)$ and

$$1 - \frac{1}{\omega_s^d(G)} = 2 \binom{k}{2} \frac{1}{k^2} = f_G(\hat{x}, \hat{y}) \leq \max_{(x, y) \in \Delta^n \times D_s(G)} f_G(x, y) = f_s^d(G).$$

We prove $1 - \frac{1}{\omega_s^d(G)} \geq f_s^d(G)$ by contradiction. Suppose there exists $(\hat{x}, \hat{y}) \in \Delta^n \times D_s(G)$ such that $1 - \frac{1}{\omega_s^d(G)} < f_G(\hat{x}, \hat{y})$. Let $\hat{E} = \{\{i, j\} \in \bar{E} \mid \hat{y}_{ij} = 1\}$

and $\hat{G} = (V, E \cup \hat{E})$. Then if we fix $y = \hat{y}$, Eq. (4) becomes the Motzkin–Straus formulation for the maximum clique problem in $\hat{G}$. We have

$$1 - \frac{1}{\omega_s^d(G)} < f_G(\hat{x}, \hat{y}) \leq \max_{x \in \Delta^n} f_G(x, \hat{y}) = 1 - \frac{1}{\omega(\hat{G})},$$

implying that $\omega_s^d(G) < \omega(\hat{G})$. However, since $(\hat{x}, \hat{y}) \in \Delta^n \times D_s(G)$, we have $|\hat{E}| \leq s$, hence any clique in $\hat{G}$ is an s -defective clique in G . Thus, $\omega_s^d(G) \geq \omega(\hat{G})$, a contradiction. $\square$

Theorem 1 (Motzkin–Straus formulation for s -defective clique) *The s -defective clique number $\omega_s^d(G)$ of G can be expressed using a global maximum value of a continuous cubic program as follows:*

$$1 - \frac{1}{\omega_s^d(G)} = \max_{(x, y) \in \Delta^n \times D'_s(G)} f_G(x, y), \quad (8)$$

where $f_G(x, y)$ is as in (5), and $D'_s(G)$ is obtained from $D_s(G)$ by relaxing the integrality of y :

$$D'_s(G) = \{y \in [0, 1]^{\bar{m}} \mid e_m^T y \leq s\}. \quad (9)$$

Proof By Proposition 1, it suffices to prove that (8) has an optimal solution with binary y . Note that $f_G(x, y)$ is linear with respect to y and can be expressed as $f_G(x, y) = z(x) + \sum_{\{i, j\} \in \bar{E}} w_{ij}(x) y_{ij}$, where $z(x) = x^T A_G x$ and $w_{ij}(x) = 2x_i x_j$ for $\{i, j\} \in \bar{E}$. Let $(\hat{x}, \hat{y})$ be an optimal solution of (8), where $\hat{y}$ is fractional. Let $F = \{\{i, j\} \in \bar{E} \mid 0 < \hat{y}_{ij} < 1\}$ and $t = \sum_{\{i, j\} \in F} \hat{y}_{ij}$. Let F_t be the set of the first $\lceil t \rceil$ elements of the ordered set listing the edges $\{i, j\} \in F$ in nonincreasing order of $w_{ij}(\hat{x})$. For $\{i, j\} \in \bar{E}$, let

$$y_{ij}^* = \begin{cases} 1, & \text{if } \{i, j\} \in F_t, \\ 0, & \text{if } \{i, j\} \in F \setminus F_t, \\ \hat{y}_{ij}, & \text{if } \{i, j\} \in \bar{E} \setminus F. \end{cases}$$

Then,

$$e_m^T y^* = \sum_{\{i, j\} \in F} y_{ij}^* + \sum_{\{i, j\} \in \bar{E} \setminus F} y_{ij}^* = \lceil t \rceil + \sum_{\{i, j\} \in \bar{E} \setminus F} \hat{y}_{ij} \leq s,$$

so $y^* \in D_s(G)$. Finally, let $x^* = \hat{x}$. Then $f_G(x^*, y^*) \geq f_G(\hat{x}, \hat{y})$, so (x^*, y^*) is optimal for (8) with binary y^* . $\square$

2.2 Motzkin–Straus formulation for the maximum s -plex problem

Consider the following mixed integer optimization problem.

$$f_s(G) = \max_{(x,y) \in \Delta^n \times P_s(G)} f_G(x, y), \quad (10)$$

where $f_G(x, y)$ is defined in (5),

$$P_s(G) = \left\{ y \in [0, 1]^{\bar{m}} \mid B_{\bar{G}} y \leq (s-1)e_n \right\}, \quad (11)$$

and $B_{\bar{G}}$ is the $n \times \bar{m}$ incidence matrix of the complement graph $\bar{G}$.

Proposition 2 *Let $\omega_s(G)$ denote the s -plex number of G . Then*

$$1 - \frac{1}{\omega_s(G)} = f_s(G). \quad (12)$$

Proof The proof is nearly identical to that of Proposition 1. $\square$

To obtain a generalization of the Motzkin–Straus formulation for s -plex, we need to show that the integrality constraints on y in (11) can be relaxed. The resulting optimization problem is

$$f'_s(G) = \max_{(x,y) \in \Delta^n \times P'_s(G)} f_G(x, y), \quad (13)$$

where

$$P'_s(G) = \{y \in [0, 1]^{\bar{m}} \mid B_{\bar{G}} y \leq (s-1)e_n\}. \quad (14)$$

First, we establish some lemmata which require some new notation. For $S \subseteq V$, the notation $x[S]$ refers to the restriction of x to S , i.e., $x[S] = [x_i]_{i \in S}$. Also, $y[\bar{E}(S)]$ refers to the restriction of y to $\bar{E}(S) = \{\{i, j\} \in \bar{E} \mid i, j \in S\}$. Finally, the support of x is denoted by $\text{supp}(x) := \{i \in V \mid x_i > 0\}$.

Lemma 2 *For every feasible solution $(\hat{x}, \hat{y})$ to (13), there is an alternative feasible solution $(\tilde{x}, \hat{y})$ whose objective value is no worse, $f_G(\tilde{x}, \hat{y}) \geq f_G(\hat{x}, \hat{y})$, and whose restriction to $S := \text{supp}(\tilde{x})$ is strictly positive, $(\tilde{x}[S], \hat{y}[\bar{E}(S)]) > 0$. Moreover, the inclusion $\text{supp}(\tilde{x}) \subseteq \text{supp}(\hat{x})$ holds and is strict when $\hat{y} \neq 0$.*

Proof If the restriction of $(\hat{x}, \hat{y})$ to $\hat{S} := \text{supp}(\hat{x})$ is strictly positive, i.e., $(\hat{x}[\hat{S}], \hat{y}[\bar{E}(\hat{S})]) > 0$, then the statement holds trivially by setting $\tilde{x} = \hat{x}$. So, suppose there is a fake edge $\{i, j\} \in \bar{E}(\hat{S})$ with $\hat{y}_{ij} = 0$. We modify $\hat{x}$ to obtain $\tilde{x}$ such that $f_G(\tilde{x}, \hat{y}) \geq f_G(\hat{x}, \hat{y})$, and $\text{supp}(\tilde{x}) \subseteq \text{supp}(\hat{x})$. For $k \in V$, let

$$\tilde{x}_k = \begin{cases} \hat{x}_k, & \text{if } k \neq i, j, \\ \hat{x}_k - \varepsilon, & \text{if } k = i, \\ \hat{x}_k + \varepsilon, & \text{if } k = j, \end{cases}$$

where ε is selected later. We have

$$f_G(\tilde{x}, \hat{y}) - f_G(\hat{x}, \hat{y}) = 2\varepsilon p,$$

where

$$p := \sum_{k \in N(j)} \hat{x}_k - \sum_{k \in N(i)} \hat{x}_k + \sum_{k \in \bar{N}(j) \setminus \{i\}} \hat{y}_{jk} \hat{x}_k - \sum_{k \in \bar{N}(i) \setminus \{j\}} \hat{y}_{ik} \hat{x}_k$$

and $\bar{N}(v) := N_{\bar{G}}(v)$. Note that p does not contain ε because $\hat{y}_{ij} = 0$. Setting

$$\varepsilon = \begin{cases} \hat{x}_i, & \text{if } p \geq 0, \\ -\hat{x}_j, & \text{if } p < 0 \end{cases}$$

yields $\tilde{x}$ such that $f_G(\tilde{x}, \hat{y}) \geq f_G(\hat{x}, \hat{y})$. Meanwhile, the restriction of $(\tilde{x}, \hat{y})$ to $\tilde{S} := \text{supp}(\tilde{x})$, denoted $(\tilde{x}[\tilde{S}], \hat{y}[\bar{E}(\tilde{S})])$, has fewer zero entries than that of $(\hat{x}[\hat{S}], \hat{y}[\bar{E}(\hat{S})])$ in both its x and y portions. Also, $\tilde{S} \subsetneq \hat{S}$. So, if $(\tilde{x}[\tilde{S}], \hat{y}[\bar{E}(\tilde{S})])$ is strictly positive, then we are done. Otherwise, repeat this process recursively, by setting $\hat{x} = \tilde{x}$, until $\hat{y}[\bar{E}[\tilde{S}]]$ has no zero entries. $\square$

Lemma 3 *If $(\hat{x}, \hat{y})$ is an optimal solution of (13) for G , then its restriction $(\hat{x}[S], \hat{y}[\bar{E}(S)])$ to $S = \text{supp}(\hat{x})$ is an optimal solution of (13) for $G[S]$.*

Proof First, see that $\hat{x}[S] \in \Delta^{|S|}$ because $\hat{x} \in \Delta^n$ and $\hat{x}_i = 0$ for all $i \in V \setminus S$. Also, $\hat{y}[\bar{E}(S)]$ belongs to $P'_s(G[S])$ because every $i \in S$ satisfies

$$\sum_{j \in N_{\bar{G}[S]}(i)} \hat{y}_{ij} \leq \sum_{j \in N_{\bar{G}}(i)} \hat{y}_{ij} \leq s - 1.$$

For contradiction purposes, suppose there exists $(\tilde{x}, \tilde{y}) \in \Delta^{|S|} \times P'_s(G[S])$ such that $f_{G[S]}(\tilde{x}, \tilde{y}) > f_{G[S]}(\hat{x}[S], \hat{y}[\bar{E}(S)])$. Add zero entries to $(\tilde{x}, \tilde{y})$ for vertices $V \setminus S$ and fake edges $\bar{E} \setminus \bar{E}(S)$ to obtain $(\tilde{x}^+, \tilde{y}^+) \in \Delta^n \times P'_s(G)$. Then,

$$f_G(\tilde{x}^+, \tilde{y}^+) = f_{G[S]}(\tilde{x}, \tilde{y}) > f_{G[S]}(\hat{x}[S], \hat{y}[\bar{E}(S)]) = f_G(\hat{x}, \hat{y}),$$

contradicting that $(\hat{x}, \hat{y})$ is an optimal solution of (13) for G . $\square$

Lemma 4 *If $(\hat{x}, \hat{y})$ is an optimal solution of (13) and $\hat{y}$ is a fractional extreme point of $P'_s(G)$, then at least one entry of $(\hat{x}, \hat{y})$ is zero.*

Proof For contradiction purposes, suppose that such an optimal solution $(\hat{x}, \hat{y})$ is strictly positive. Note that if we fix $x = \hat{x}$ in (13), we obtain the problem

$$\begin{aligned} \max \quad & \sum_{\{i,j\} \in \bar{E}} \hat{w}_{ij} y_{ij} + \hat{z} \\ \text{s.t.} \quad & \sum_{j \in N_{\bar{G}}(i)} y_{ij} \leq s - 1 \quad \forall i \in V \\ & 0 \leq y_{ij} \leq 1 \quad \forall \{i, j\} \in \bar{E}, \end{aligned} \tag{15}$$

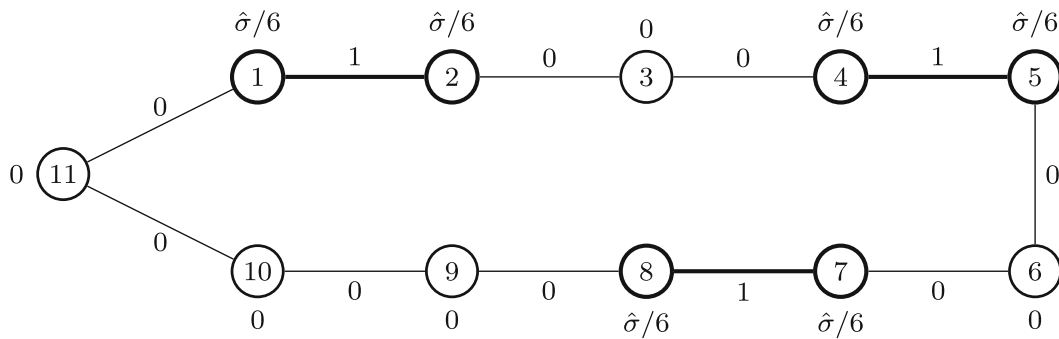


Fig. 2 An illustration of the proof of Lemma 4 for $l = 11$, i.e., $q = 3$ and $r = 2$

where $\hat{w}_{ij} = 2\hat{x}_i\hat{x}_j > 0$ for $\{i, j\} \in \bar{E}$, and $\hat{z} = 2 \sum_{\{i, j\} \in E} \hat{x}_i\hat{x}_j$. This problem is equivalent to the LP relaxation of the maximum b -matching formulation (3) for graph $\bar{G}$ with $b = (s - 1)e_n$ and the edge weights $\hat{w}_{ij}$, $\{i, j\} \in \bar{E}$.

Since $\hat{y}$ is fractional, $F := \{\{i, j\} \in \bar{E} \mid 0 < \hat{y}_{ij} < 1\}$ is nonempty. So, by Lemma 1, $\hat{y}_{ij} = \frac{1}{2}$, $\forall \{i, j\} \in F$, and there exists $H \subseteq F$ such that $\bar{G}(H) = (V_H, H)$ is an isolated odd cycle in $\bar{G}(F)$. Let $l = |V_H|$ and relabel the vertices in V so that $V_H = \{1, 2, \dots, l\}$. We construct $(\tilde{x}, \tilde{y}) \in \Delta^n \times P'_s(G)$ such that $f_G(\tilde{x}, \tilde{y}) > f_G(\hat{x}, \hat{y})$, contradicting the optimality of $(\hat{x}, \hat{y})$.

Let $q = \lfloor l/3 \rfloor$ and $r = l - 3q \in \{0, 1, 2\}$ be the quotient and the remainder from the division of l by 3. That is, $l = 3q + r$ for some $r \in \{0, 1, 2\}$. Let

$$\hat{\sigma} := \sum_{i \in V_H} \hat{x}_i. \quad (16)$$

Note that $\hat{\sigma} > 0$ since $(\hat{x}, \hat{y})$ is strictly positive.

First, assume $q > r$. Since $l = 3q + r$ and $r \in \{0, 1, 2\}$, this assumption is valid for all odd integers $l \geq 3$ except $l = 5$; this exceptional case will be addressed later. We define $(\tilde{x}, \tilde{y})$ as follows.

$$\tilde{x}_i = \begin{cases} \hat{\sigma}/(2q), & \text{if } i \in \{3k + 1, 3k + 2 \mid k = 0, 1, \dots, q - 1\}, \\ 0, & \text{if } i \in \{3k \mid k = 0, 1, \dots, q - 1\} \cup \{3q + j \mid j = 0, 1, \dots, r\}, \\ \hat{x}_i, & \text{otherwise;} \end{cases} \quad (17)$$

$$\tilde{y}_{ij} = \begin{cases} 1, & \text{if } \{i, j\} \in H' := \{\{3k + 1, 3k + 2\} \mid k = 0, 1, \dots, q - 1\}, \\ 0, & \text{if } \{i, j\} \in H \setminus H', \\ \hat{y}_{ij}, & \text{otherwise.} \end{cases} \quad (18)$$

Figure 2 provides an illustration. Note that $2q$ of the first $l = 3q + r$ entries of $\tilde{x}$ defined above are equal to $\hat{\sigma}/(2q)$, and the remaining $q + r$ entries are zeros. Thus, $\sum_{i \in V_H} \tilde{x}_i = \hat{\sigma} = \sum_{i \in V_H} \hat{x}_i$ according to (16). Also, each $j \in V_H$ has exactly two incident edges in H . Denoting these edges by $\{j, j'\}$ and $\{j, j''\}$, we have $\tilde{y}_{jj'} + \tilde{y}_{jj''} \leq$

1 for all $j \in V_H$. Since $\hat{y}_{jj'} + \hat{y}_{jj''} = \frac{1}{2} + \frac{1}{2} = 1$, changing $\hat{y}$ to $\tilde{y}$ does not violate the s -plex constraints (14). Thus, $(\tilde{x}, \tilde{y}) \in \Delta^n \times P'_s(G)$.

Now, we show $f_G(\tilde{x}, \tilde{y}) > f_G(\hat{x}, \hat{y})$. Denote by

$$\begin{aligned} E(V_H) &= \{\{i, j\} \in E \mid i, j \in V_H\}, \\ \bar{E}(V_H) &= \{\{i, j\} \in \bar{E} \mid i, j \in V_H\}, \\ \delta(V_H) &= \{\{i, j\} \in E \mid i \in V_H, j \in V \setminus V_H\}, \\ \bar{\delta}(V_H) &= \{\{i, j\} \in \bar{E} \mid i \in V_H, j \in V \setminus V_H\}. \end{aligned}$$

Since $(\tilde{x}, \tilde{y})$ and $(\hat{x}, \hat{y})$ can only differ in the entries corresponding to the vertices and edges within $\bar{G}\langle H \rangle$, we can write

$$f_G(\tilde{x}, \tilde{y}) - f_G(\hat{x}, \hat{y}) = \left(g(\tilde{x}, \tilde{y}) + h(\tilde{x}, \tilde{y}) \right) - \left(g(\hat{x}, \hat{y}) + h(\hat{x}, \hat{y}) \right),$$

where

$$\begin{aligned} g(x, y) &:= 2 \sum_{\{i, j\} \in E(V_H)} x_i x_j + 2 \sum_{\{i, j\} \in \bar{E}(V_H)} y_{ij} x_i x_j \\ &= 2 \sum_{\{i, j\} \in E(V_H) \cup \bar{E}(V_H)} x_i x_j - 2 \sum_{\{i, j\} \in \bar{E}(V_H)} (1 - y_{ij}) x_i x_j, \end{aligned} \quad (19)$$

$$\begin{aligned} h(x, y) &:= 2 \sum_{\{i, j\} \in \delta(V_H)} x_i x_j + 2 \sum_{\{i, j\} \in \bar{\delta}(V_H)} y_{ij} x_i x_j \\ &= 2 \sum_{\{i, j\} \in \delta(V_H) \cup \bar{\delta}(V_H)} x_i x_j - 2 \sum_{\{i, j\} \in \bar{\delta}(V_H)} (1 - y_{ij}) x_i x_j. \end{aligned} \quad (20)$$

The following equality will be used later.

$$2 \sum_{\{i, j\} \in E(V_H) \cup \bar{E}(V_H)} x_i x_j = \left(\sum_{i \in V_H} x_i \right)^2 - \sum_{i \in V_H} x_i^2. \quad (21)$$

By Lemma 1 and the assumption that $\hat{y}$ is strictly positive, we have $\hat{y}_{ij} = 1$ for every $\{i, j\} \in \bar{\delta}(V_H)$. So, by (20),

$$h(\hat{x}, \hat{y}) = 2 \sum_{\{i, j\} \in \delta(V_H) \cup \bar{\delta}(V_H)} \hat{x}_i \hat{x}_j = 2 \sum_{i \in V_H} \hat{x}_i \sum_{j \in V \setminus V_H} \hat{x}_j = 2\hat{\sigma}(1 - \hat{\sigma}) = h(\tilde{x}, \tilde{y}),$$

and so

$$f_G(\tilde{x}, \tilde{y}) - f_G(\hat{x}, \hat{y}) = g(\tilde{x}, \tilde{y}) - g(\hat{x}, \hat{y}).$$

By the construction of $(\tilde{x}, \tilde{y})$ given in (17) and (18), every $\{i, j\} \in H$ satisfies $\tilde{y}_{ij}=1$ or $\tilde{x}_i\tilde{x}_j = 0$. Also, $\tilde{y}_{ij} = \hat{y}_{ij} = 1$ for every $\{i, j\} \in \bar{E}(V_H) \setminus H$. Thus,

$$\sum_{\{i,j\} \in \bar{E}(V_H)} (1 - \tilde{y}_{ij})\tilde{x}_i\tilde{x}_j = 0.$$

So, by (19), (21), and (17),

$$g(\tilde{x}, \tilde{y}) = 2 \sum_{\{i,j\} \in E(V_H) \cup \bar{E}(V_H)} \tilde{x}_i\tilde{x}_j = \hat{\sigma}^2 \left(1 - \frac{1}{2q}\right).$$

By Lemma 1 and the assumption that $\hat{y}$ is strictly positive, every $\{i, j\} \in H$ has $\hat{y}_{ij} = \frac{1}{2}$ and every $\{i, j\} \in \bar{E}(V_H) \setminus H$ has $\hat{y}_{ij} = 1$. So, by (19) and (21),

$$g(\hat{x}, \hat{y}) = \left(\sum_{i \in V_H} \hat{x}_i\right)^2 - \sum_{i \in V_H} \hat{x}_i^2 - \sum_{\{i,j\} \in H} \hat{x}_i\hat{x}_j = \hat{\sigma}^2 - c(\hat{x}),$$

where $c(\hat{x}) := \sum_{i \in V_H} \hat{x}_i^2 + \sum_{\{i,j\} \in H} \hat{x}_i\hat{x}_j$ satisfies the following inequality:

$$c(\hat{x}) = \frac{1}{2} \sum_{\{i,j\} \in H} (\hat{x}_i + \hat{x}_j)^2 \geq \frac{1}{2l} \left(\sum_{\{i,j\} \in H} (\hat{x}_i + \hat{x}_j)\right)^2 = \frac{2\hat{\sigma}^2}{l}.$$

The inequality above is obtained by invoking the root mean square – arithmetic mean inequality. Therefore,

$$g(\hat{x}, \hat{y}) = \hat{\sigma}^2 - c(\hat{x}) \leq \hat{\sigma}^2 \left(1 - \frac{2}{l}\right).$$

By our assumption that $q > r$, we obtain

$$f_G(\tilde{x}, \tilde{y}) - f_G(\hat{x}, \hat{y}) = g(\tilde{x}, \tilde{y}) - g(\hat{x}, \hat{y}) \geq \hat{\sigma}^2 \left(\frac{2}{l} - \frac{1}{2q}\right) = \hat{\sigma}^2 \left(\frac{q-r}{2ql}\right) > 0.$$

Finally, in the exceptional case where $l = 5$ (and thus $q = 1$ and $r = 2$), the entries of $(\tilde{x}, \tilde{y})$ corresponding to $\bar{G}\langle H \rangle$ can be defined as follows:

$$\tilde{x}_1 = \tilde{x}_2 = \tilde{x}_4 = \hat{\sigma}/3 \quad \tilde{x}_3 = \tilde{x}_5 = 0; \quad (22)$$

$$\tilde{y}_{12} = 1 \quad \tilde{y}_{23} = \tilde{y}_{34} = \tilde{y}_{45} = \tilde{y}_{51} = 0. \quad (23)$$

Then, $g(\tilde{x}, \tilde{y}) = \left(\sum_{i \in V_H} \tilde{x}_i \right)^2 - \sum_{i \in V_H} \tilde{x}_i^2 = \frac{2}{3} \hat{\sigma}^2$, so

$$f_G(\tilde{x}, \tilde{y}) - f_G(\hat{x}, \hat{y}) = g(\tilde{x}, \tilde{y}) - g(\hat{x}, \hat{y}) \geq \hat{\sigma}^2 \left(\frac{2}{3} - 1 + \frac{2}{5} \right) = \frac{\hat{\sigma}^2}{15} > 0,$$

contradicting the optimality of $(\hat{x}, \hat{y})$. The proof is complete. $\square$

Theorem 2 (Motzkin–Straus formulation for s -plex) *The s -plex number $\omega_s(G)$ of G can be expressed using a global maximum value of a continuous cubic program as follows:*

$$1 - \frac{1}{\omega_s(G)} = \max_{(x,y) \in \Delta^n \times P'_s(G)} f_G(x, y), \quad (24)$$

where $f_G(x, y)$ is as in (5), and $P'_s(G)$ is given in (14).

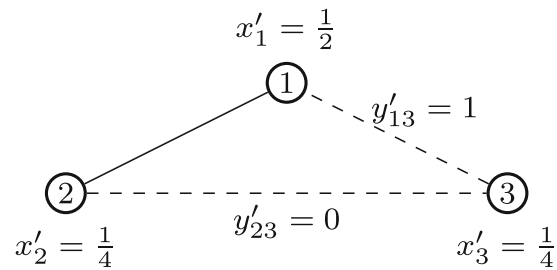
Proof By Proposition 2, it suffices to show that problem (24) always has an optimal solution (x^*, y^*) with $y^* \in \{0, 1\}^{\bar{m}}$. We show this by constructing such an optimal solution, given an arbitrary optimal solution $(\hat{x}, \hat{y})$. The proof consists of the following steps.

1. Apply Lemma 2 to obtain an alternative optimal solution $(\tilde{x}, \hat{y})$ in which $\hat{y}[\bar{E}(\tilde{S})] > 0$ and $\tilde{S} \subseteq \hat{S}$, where $\tilde{S} := \text{supp}(\tilde{x})$ and $\hat{S} := \text{supp}(\hat{x})$.
2. By Lemma 3, $(\tilde{x}[\tilde{S}], \hat{y}[\bar{E}(\tilde{S})])$ is an optimal solution of the Motzkin–Straus formulation (24) for the maximum s -plex problem in $G[\tilde{S}]$.
3. Solve the b -matching LP obtained by fixing $x = \tilde{x}[\tilde{S}]$ to find an optimal solution $\tilde{y}$ that is an extreme point of $P'_s(G[\tilde{S}])$. This gives an alternative optimal solution $(\tilde{x}[\tilde{S}], \tilde{y})$ of (24) for $G[\tilde{S}]$.
4. If $\tilde{y}$ is not strictly positive, repeat steps 1–3 until obtaining an optimal solution $(\tilde{x}[\tilde{S}], \tilde{y})$ of (24) for $G[\tilde{S}]$ with $\tilde{y}$ being an extreme point of $P'_s(G[\tilde{S}])$. If $\tilde{y}$ is fractional, we obtain a contradiction with Lemma 4 for $G[\tilde{S}]$ and the optimal solution $(\tilde{x}[\tilde{S}], \tilde{y})$. We conclude that $\tilde{y}_{ij} = 1$ for all $\{i, j\} \in \bar{E}(\tilde{S})$.
5. Finally, extend $(\tilde{x}[\tilde{S}], \tilde{y})$ to $(x^*, y^*) \in \Delta^n \times P'_s(G)$ by adding zero entries for vertices in $V \setminus \tilde{S}$ and for fake edges in $\bar{E} \setminus \bar{E}(\tilde{S})$.

Every change applied during this procedure preserves optimality, so (x^*, y^*) is an optimal solution of (24) with binary y^* . $\square$

Observe that as a part of the above proof, we showed that we can find $x^* \in \Delta^n$ such that the LP relaxation (15) of the maximum weight b -matching problem has an integer optimal solution. Providing a complete description of the b -matching polytope

Fig. 3 An illustration to Example 1. The dashed lines show the missing edges



(which is the convex hull of the solutions to the binary program (3)) generally requires adding the so-called blossom inequalities:

$$\sum_{\{i,j\} \in E(U)} y_{ij} + \sum_{\{i,j\} \in F} y_{ij} \leq \left\lfloor \frac{b(U) + |F|}{2} \right\rfloor, \forall U \subseteq V, F \subseteq \delta(U) : b(U) + |F| \text{ odd},$$

to the degree inequalities and bound constraints [15], where $b(U) = \sum_{i \in U} b_i$, $E(U) = \{\{i, j\} \in E \mid i, j \in U\}$, and $\delta(U) = \{\{i, j\} \in E \mid i \in U, j \in V \setminus U\}$. However, no blossom inequalities are needed to find a binary optimal solution in our case.

In the proof of Theorem 2, we constructed an optimal solution (x^*, y^*) satisfying two intuitive properties:

1. $S^* := \text{supp}(x^*)$ is a maximum s -plex in G , and
2. y^* is binary, with $y^*_{ij} = 1$ if and only if $\{i, j\} \in \bar{E}(S^*)$.

However, not all global optima satisfy these properties, as Example 1 shows.

Example 1 Any pair of vertices in the graph G from Fig. 3 is a maximum 2-plex, and yet there is a global optimal solution (x', y') of (24) for which $S' := \text{supp}(x') = \{1, 2, 3\}$ is not a 2-plex.

We have seen that even though an optimal solution of (24) may not directly correspond to an s -plex, it can be easily converted into another optimal solution that does. In the next subsection, we show the same to be true for feasible (not necessarily optimal) solutions to the s -defective clique and s -plex formulations.

2.3 Computing clique relaxations from feasible solutions to multi-linear continuous formulations

One of the challenges associated with continuous optimization approaches to combinatorial optimization problems consists in developing effective and efficient strategies for converting a solution obtained for a continuous model into a feasible solution for the combinatorial problem. In particular, it is known that the original Motzkin–Straus formulation allows the so-called spurious local maxima, which do not correspond to characteristic vectors of cliques [37], and we have seen in Example 1 that spurious optima are also possible for the Motzkin–Straus formulation of the maximum s -plex problem. To overcome this issue, regularized versions of the Motzkin–Straus formulation for the maximum clique problem have been developed [8, 24], where every continuous local maximum is guaranteed to correspond to a maximal clique. These

results can potentially be extended to the considered clique relaxations. However, one should keep in mind that all the known regularizations of the Motzkin–Straus formulation for the maximum clique problem result in the loss of the bilinear structure that the original formulation enjoys.

Alternatively, one could take advantage of the trilinear structure of the objective function of the proposed cubic formulations in order to extract an s -defective clique or an s -plex based on a feasible point of the corresponding cubic formulation. This approach essentially follows the logic of the proofs of Theorems 1 and 2. Each of these proofs constructs a global optimal solution (x^*, y^*) of the Motzkin–Straus formulation for the respective cluster detection problem, where y^* is guaranteed to be binary, starting from an arbitrary global maximum $(\hat{x}, \hat{y})$. The cardinality of a maximum s -defective clique (s -plex) is then given by $\lceil 1/(1 - f_G(x^*, y^*)) \rceil = \lceil 1/(1 - f_G(\hat{x}, \hat{y})) \rceil$. Since computing a global maximum of a Motzkin–Straus cubic program is extremely challenging (our results imply NP-hardness of both cubic problems), in practice we are interested in converting any feasible solution $(\hat{x}, \hat{y})$ of a Motzkin–Straus formulation into an s -defective clique or s -plex C of cardinality $|C| \geq \lceil 1/(1 - f_G(\hat{x}, \hat{y})) \rceil$. We break this task into two steps:

- (i) Compute a feasible solution $(\tilde{x}, \tilde{y})$ to the respective Motzkin–Straus formulation such that $\tilde{y}$ is binary and $f_G(\tilde{x}, \tilde{y}) \geq f_G(\hat{x}, \hat{y})$.
- (ii) Use $(\tilde{x}, \tilde{y})$ to construct a feasible solution $C \subseteq V$ of cardinality

$$|C| \geq \lceil 1/(1 - f_G(\tilde{x}, \tilde{y})) \rceil \geq \lceil 1/(1 - f_G(\hat{x}, \hat{y})) \rceil.$$

In the case of s -defective clique, the first step can be completed using the procedure outlined in the proof of Theorem 1. In the case of s -plex, the procedure in the proof of Theorem 2 uses the fact that $(\hat{x}, \hat{y})$ is a global optimal solution to obtain a contradiction in Lemma 4 and hence cannot be used directly. However, a slight modification of this procedure will lead to the desired outcome. Specifically, instead of obtaining a contradiction by Lemma 4 in step 4 of the procedure, we use the modification described in (17), (18) or (22), (23) to obtain $(\tilde{x}, \tilde{y})$ such that $\tilde{y}$ is binary and $f_G(\tilde{x}, \tilde{y}) > f_G(\hat{x}, \hat{y})$.

Given a feasible solution $(\tilde{x}, \tilde{y})$ with binary $\tilde{y}$, we can use the following process to obtain an s -defective clique (s -plex) with at least $\lceil 1/(1 - f_G(\tilde{x}, \tilde{y})) \rceil$ vertices. Let $\tilde{E} = \{\{i, j\} \mid \tilde{y}_{ij} = 1\}$ and $\tilde{G} = (V, E \cup \tilde{E})$. Let $\tilde{C} = \text{supp}(\tilde{x}_i)$. If $\tilde{C}$ is a clique in $\tilde{G}$, then it is an s -defective clique (s -plex) in G and we are done. Otherwise, if $\tilde{C}$ is not a clique in $\tilde{G}$, we use Lemma 2 to construct x' such that $f_G(x', \tilde{y}) \geq f_G(\tilde{x}, \tilde{y})$, and $C' = \text{supp}(x')$ is a clique in $\tilde{G}$. Furthermore, if $x'_i - x'_j > \epsilon > 0$ for some $i, j \in C'$, modifying x' by subtracting ϵ from x'_i and adding ϵ to x'_j will increase the value of $f_G(x', \tilde{y})$ by $(x'_i - \epsilon)(x'_j + \epsilon) - x'_i x'_j = \epsilon(x'_i - x'_j) - \epsilon^2 > 0$. So, by setting

$$\bar{x}_i = \begin{cases} \frac{1}{|C'|}, & \text{if } i \in C', \\ 0, & \text{if } i \in V \setminus C', \end{cases}$$

we have $C := \text{supp}(\bar{x}) = C'$ and $f_G(\bar{x}, \tilde{y}) \geq f_G(x', \tilde{y})$. Thus, we obtain an s -defective clique (s -plex) C in G such that

$$1 - \frac{1}{|C|} = f_G(\bar{x}, \tilde{y}) \geq f_G(x', \tilde{y}) \geq f_G(\tilde{x}, \tilde{y}),$$

implying that $|C| \geq \lceil 1/(1 - f_G(\tilde{x}, \tilde{y})) \rceil$.

3 Extending Turán's theorem and other lower bounds on the clique number to clique relaxations

In this section, we show that the formulations developed in Sect. 2 can be used to generalize Turán's theorem to s -defective clique and s -plex, as well as to extend some known lower bounds on the clique number to the considered clique relaxations.

First, recall Turán's bound (1):

$$m \leq \left(1 - \frac{1}{\omega(G)}\right) \frac{n^2}{2}. \quad (25)$$

Let λ_G be the maximum eigenvalue (spectral radius) of A_G , and let u_P be the Perron unit eigenvector of A_G ($u_P^T u_P = 1$). Also, let $s_P = e_n^T u_P$. In addition, denote by $d = (d_1, d_2, \dots, d_n)^T$, where $d_i = |N_G(i)|$ is the degree of i in G , $\forall i \in V$. The following spectral lower bound on the clique number was established by Wilf [46]:

$$\omega(G) \geq \frac{s_P^2}{s_P^2 - \lambda_G}. \quad (26)$$

Finally, the following degree-based lower bound can be obtained from the Motzkin–Straus formulation by setting $x = d/(2m)$ [42]:

$$1 - \frac{1}{\omega(G)} \geq \frac{d^T A_G d}{(2m)^2}. \quad (27)$$

3.1 Bounds for the s -defective clique number

Theorem 3 (Turán's bound for s -defective clique) *Assume that $m \leq \binom{n}{2} - s$. Then*

$$m \leq \left(1 - \frac{1}{\omega_s^d(G)}\right) \frac{n^2}{2} - s. \quad (28)$$

Proof Let

$$\bar{x} = \frac{1}{n} e_n \text{ and } \bar{y} = \frac{s}{\bar{m}} e_{\bar{m}}. \quad (29)$$

Then, since $s \leq \bar{m}$, $(\bar{x}, \bar{y})$ is feasible to (8), and from Theorem 1 we have

$$1 - \frac{1}{\omega_s^d(G)} \geq \frac{2(m+s)}{n^2},$$

which is equivalent to (28). $\square$

When $s = 0$, the lower bound (28) is equivalent to Turán's classical bound (25). Next, we apply the Motzkin–Straus formulation to derive a spectral lower bound on the s -defective clique number.

Theorem 4 (Wilf's bound for s -defective clique) *Assume that $m \leq \binom{n}{2} - s$. Then*

$$1 - \frac{1}{\omega_s^d(G)} \geq \frac{\lambda_G}{s_P^2} + \frac{s}{\binom{n}{2} - m} \left(1 - \frac{\lambda_G + 1}{s_P^2} \right), \quad (30)$$

Proof We substitute

$$\bar{x} = \frac{u_P}{s_P} \text{ and } \bar{y} = \frac{s}{\bar{m}} e_{\bar{m}} \quad (31)$$

in (8). Then it follows from Theorem 1 that

$$\begin{aligned} 1 - \frac{1}{\omega_s^d(G)} &\geq f_G(\bar{x}, \bar{y}) = \frac{u_P^T A_G u_P}{s_P^2} + \frac{s}{\bar{m}} \left(\frac{u_P^T A_{\bar{G}} u_P}{s_P^2} \right) \\ &= \frac{\lambda_G}{s_P^2} + \frac{s}{\bar{m}} \cdot \frac{1}{s_P^2} \cdot u_P^T (O_n - I_n - A_G) u_P \\ &= \frac{\lambda_G}{s_P^2} + \frac{s}{\bar{m}} \cdot \frac{1}{s_P^2} \cdot (s_P^2 - 1 - \lambda_G) = \frac{\lambda_G}{s_P^2} + \frac{s}{\binom{n}{2} - m} \left(1 - \frac{\lambda_G + 1}{s_P^2} \right), \end{aligned}$$

where O_n is the $n \times n$ matrix with all ones and I_n is the $n \times n$ identity matrix. $\square$

When $s = 0$, the lower bound (30) is equivalent to Wilf's bound (26).

Theorem 5 (Degree-based bound for s -defective clique) *Assume that $m \leq \binom{n}{2} - s$. Then*

$$1 - \frac{1}{\omega_s^d(G)} \geq \frac{d^T A_G d}{(2m)^2} + \frac{s}{\binom{n}{2} - m} \cdot \frac{d^T A_{\bar{G}} d}{(2m)^2}. \quad (32)$$

Proof We substitute

$$\bar{x} = \frac{d}{2m} \text{ and } \bar{y} = \frac{s}{\bar{m}} e_{\bar{m}} \quad (33)$$

in (8) to obtain

$$1 - \frac{1}{\omega_s^d(G)} \geq f_G(\bar{x}, \bar{y}) = \frac{d^T A_G d}{(2m)^2} + \frac{s}{\binom{n}{2} - m} \cdot \frac{d^T A_{\bar{G}} d}{(2m)^2}.$$

□

When $s = 0$, the lower bound (32) is equivalent to the degree-based bound (27) for the clique number.

3.2 Bounds for the s -plex number

Theorem 6 (Turán's bound for s -plex)

$$m \leq \left(1 - \frac{1}{\omega_s(G)}\right) \frac{n^2}{2} - \mu_{s-1}(\bar{G}), \quad (34)$$

where $\mu_{s-1}(\bar{G})$ is the cardinality of a maximum $(1$ -capacitated) b -matching in $\bar{G}$ with $b = (s - 1)e_n$.

Proof We substitute

$$\bar{x} = \frac{1}{n}e_n \text{ and } \bar{y}_{ij} = \begin{cases} 1, & \text{if } \{i, j\} \in M, \\ 0, & \text{otherwise} \end{cases} \quad (35)$$

in (10), where $M \subseteq \bar{E}$ is a maximum b -matching with $b = (s - 1)e_n$ in the complement graph $\bar{G}$. □

When $s = 1$, the lower bound (34) is equivalent to Turán's classical bound (25).

Theorem 7 (Wilf's bound for s -plex) *Assume that V is not an s -plex. Then*

$$1 - \frac{1}{\omega_s(G)} \geq \frac{\lambda_G}{s_P^2} + \frac{s - 1}{n - 1 - \delta(G)} \left(1 - \frac{\lambda_G + 1}{s_P^2}\right), \quad (36)$$

where $\delta(G)$ denotes the minimum degree of G .

Proof We substitute

$$\bar{x} = \frac{u_P}{s_P} \text{ and } \bar{y} = \frac{s - 1}{n - 1 - \delta(G)} e_{\bar{m}} \quad (37)$$

in (10). Since V is not an s -plex, we have $s \leq n - \delta(G) - 1$, so $\bar{y}_{ij} \in [0, 1]$ for every $\{i, j\} \in \bar{E}$. Also, for every $i \in V$,

$$\sum_{j \in N_{\bar{G}}(i)} \bar{y}_{ij} = (s - 1) \frac{n - 1 - d_i}{n - 1 - \delta(G)} \leq s - 1,$$

so $(\bar{x}, \bar{y})$ is feasible. We have

$$\begin{aligned}
 1 - \frac{1}{\omega_s(G)} &\geq f_s(\bar{x}, \bar{y}) = \frac{u_P^T A_G u_P}{s_P^2} + \frac{s-1}{n-1-\delta(G)} \bar{x}^T A_{\bar{G}} \bar{x} \\
 &= \frac{\lambda_G}{s_P^2} + \frac{s-1}{n-1-\delta(G)} \cdot \frac{1}{s_P^2} \cdot u_P^T (O_n - I_n - A_G) u_P \\
 &= \frac{\lambda_G}{s_P^2} + \frac{s-1}{n-1-\delta(G)} \cdot \frac{1}{s_P^2} \cdot (s_P^2 - \lambda_G - 1) \\
 &= \frac{\lambda_G}{s_P^2} + \frac{s-1}{n-1-\delta(G)} \left(1 - \frac{\lambda_G + 1}{s_P^2} \right),
 \end{aligned}$$

where O_n is the $n \times n$ matrix with all ones and I_n is the $n \times n$ identity matrix. $\square$

When $s = 1$, the lower bound (36) becomes Wilf's bound (26) on the clique number.

Theorem 8 (Degree-based bound for s -plex) *Assume that V is not an s -plex. Then*

$$1 - \frac{1}{\omega_s(G)} \geq \frac{d^T A_G d}{(2m)^2} + \frac{s-1}{n-1-\delta(G)} \cdot \frac{d^T A_{\bar{G}} d}{(2m)^2}. \quad (38)$$

Proof We substitute

$$\bar{x} = \frac{d}{2m} \text{ and } \bar{y} = \frac{s-1}{n-1-\delta(G)} e_{\bar{m}} \quad (39)$$

in (10). $\square$

When $s = 1$, bound (38) becomes the degree-based bound (27) on the clique number.

4 Computational experiments

Computing provable global optima of the cubic formulations developed in this paper is a difficult task, even for small-scale instances. However, a local maximum can be found fairly quickly starting from any feasible point, and then converting it into the desired clique relaxation structure, thus yielding a heuristic for the corresponding combinatorial optimization problem. In this section, we report the results of preliminary experiments for such an approach, with local optima computed using the CONOPT solver [4]. The experiments were conducted with the following objectives in mind.

- Assess the quality of the solutions for the maximum s -defective clique and the maximum s -plex problems corresponding to local maxima obtained starting from the different points used to establish the analytical lower bounds in Sect. 3.
- Evaluate the performance of the local optimization solver applied to the Motzkin–Straus formulation starting from random initial points under a specified time limit.

In Sect. 2.3, we explained how a feasible solution $(\hat{x}, \hat{y})$ of a Motzkin–Straus formulation can be converted to a feasible solution C for the graph problem of interest such that $|C| \geq \lceil 1/(1 - f_G(\hat{x}, \hat{y})) \rceil$. It is easy to construct graph instances where the solution C obtained using this combinatorial argument will have a much higher cardinality than the lower bound obtained from the Motzkin–Straus formulation. (Consider, for example, the complement of a graph consisting of many disjoint odd cycles as an input for the maximum s -plex problem.) As a result, the contribution of the combinatorial post-processing to the overall quality of the obtained solution could overshadow that of the initial solution given by a local maximum of the continuous formulation. Hence, to provide a fair assessment of the performance of the continuous approach, we focus primarily on the quality of local maxima obtained for the cubic formulations.

Experimental setting. All experiments were conducted on a Dell PC with 64 GB RAM, Intel® Xeon® CPU E5-2643 0 @ 3.30GHz processor, running a 64-bit Windows 7 Enterprise operating system. The cubic formulations were implemented in AMPL, with the CONOPT 3.17A solver used to produce local optima. The maximum b -matchings used to generate the Turán points for the maximum s -plex problem were computed using the CPLEX 12.9.0.0 solver. The corresponding running times are negligible and are not reported. The Perron eigenvectors used in the Wilf points were computed in MATLAB® R2018A using the `perron` function [40]. The results were recorded and used as inputs in the AMPL implementation.

Test instances and the comparison base. In our experiments, we use fifty graph instances from the Second DIMACS Implementation Challenge [25]. These graphs were originally proposed for testing algorithms for the maximum clique problem, but are also typically used as a standard testbed for clique relaxation detection algorithms [6,22,43]. The instances are described in Table 1, together with results from [22], which will be used to assess the performance of the proposed formulations. In this table, the column “Graph” contains the name of each instance used. The next three columns give the number of vertices ($|V|$) and edges ($|E|$) in the graph, and the number of edges in its complement ($|\bar{E}|$). The latter is important since $\bar{m} = |\bar{E}|$ gives the number of the fake edge variables in the considered formulations. The total number of variables is $n + \bar{m} = |V| + |\bar{E}|$. The fifth column of the table specifies the clique number $\omega(G)$ of the corresponding graph. The remaining columns present the s -defective clique number $\omega_s^d(G)$ (for $s = 1, 2, 3, 4$) and the s -plex number $\omega(G)$ (for $s = 2, 3, 4, 5$), respectively, according to [22]. The figures preceded by “ $\geq$ ” in these columns do not necessarily show the best known lower bound for the corresponding instance and graph invariant; instead, they report the value of the best solution obtained by Russian Doll Search (RDS) algorithm within a 600-second time limit in [22]. Notably, some of the solutions found by RDS within the specified time limit are of worse quality than those found by the same algorithm for smaller values of s (see, e.g., the results for `brock200_1`, for which RDS failed to match the size of the maximum clique for any of the considered s values). Even though better solutions were found by heuristic methods [31,34,48], we use the RDS results as a comparison base in our preliminary experiments since, similarly to [22], we simply assess the performance of a standard general-purpose algorithm (solver), without any additional problem-specific enhancements.

Table 1 Description of the graphs used in the experiments and the results obtained in [22] by running RDS with 600-second time limit

Graph	$ V $	$ E $	$ \bar{E} $	$\omega(G)$	$\omega_s^d(G)$ for $s = \dots$				$\omega_s(G)$ for $s = \dots$			
					1	2	3	4	2	3	4	5
brock200_1	200	14,834	5066	21	≥ 20	≥ 20	≥ 19	≥ 19	≥ 20	≥ 20	≥ 19	≥ 19
brock200_2	200	9876	10,024	12	12	12	13	≥ 13	13	≥ 15	≥ 16	≥ 16
brock200_3	200	12,048	7852	15	15	16	≥ 15	≥ 15	17	≥ 18	≥ 18	≥ 20
brock200_4	200	13,089	6811	17	17	≥ 17	≥ 16	≥ 16	≥ 19	≥ 20	≥ 21	≥ 21
brock400_1	400	59,723	20,077	27	≥ 22	≥ 20	≥ 19	≥ 19	≥ 23	≥ 25	≥ 26	≥ 26
brock400_2	400	59,786	20,014	29	≥ 20	≥ 20	≥ 18	≥ 18	≥ 22	≥ 23	≥ 23	≥ 26
brock400_3	400	59,681	20,119	31	≥ 19	≥ 20	≥ 20	≥ 19	≥ 23	≥ 24	≥ 26	≥ 28
brock400_4	400	59,765	20,035	33	≥ 19	≥ 20	≥ 19	≥ 18	≥ 23	≥ 22	≥ 24	≥ 27
c-fat200-1	200	1534	18,366	12	12	12	12	12	12	12	12	14
c-fat200-2	200	3235	16,665	24	24	24	24	24	24	24	24	24
c-fat200-5	200	8473	11,427	58	58	58	58	58	58	58	58	58
c-fat500-1	500	4459	120,291	14	14	14	14	14	14	14	14	15
c-fat500-2	500	9139	115,611	26	26	26	26	26	26	26	26	26
c-fat500-5	500	23,191	101,559	64	64	64	64	64	64	64	64	64
c-fat500-10	500	46,627	78,123	126	126	126	126	126	126	126	126	126
hamming6-2	64	1824	192	32	32	32	32	32	32	32	40	48
hamming6-4	64	704	1312	4	4	5	6	6	6	8	10	12
hamming8-2	256	31,616	1024	128	128	128	128	128	128	≥ 89	≥ 44	≥ 55
hamming8-4	256	20,864	11,776	16	16	16	16	≥ 15	16	≥ 16	≥ 18	≥ 16
hamming10-2	1024	518,656	5120	512	512	512	512	512	512	≥ 89	≥ 44	≥ 55
hamming10-4	1024	434,176	89,600	40	≥ 17	≥ 18	≥ 18	≥ 15	≥ 22	≥ 16	≥ 18	≥ 16
johnson8-2-4	28	210	168	4	4	5	5	6	5	8	9	12

Table 1 continued

Graph	$ V $	$ E $	$ \bar{E} $	$\omega(G)$	$\omega_s^d(G)$ for $s = \dots$			$\omega_s(G)$ for $s = \dots$				
					1	2	3	4	2	3	4	5
johnson8-4-4	70	1855	560	14	14	14	14	15	14	18	≥ 21	≥ 23
johnson16-2-4	120	5460	1680	8	8	9	≥ 9	≥ 10	≥ 10	≥ 15	≥ 18	≥ 20
johnson32-2-4	496	107,880	14,880	16	≥ 8	≥ 7	≥ 8	≥ 9	≥ 21	≥ 24	≥ 25	≥ 26
keller4	171	9435	5100	11	12	13	≥ 14	≥ 15	15	≥ 21	≥ 16	≥ 18
keller5	776	225,990	74,710	27	≥ 16	≥ 15	≥ 15	≥ 15	≥ 15	≥ 22	≥ 16	≥ 18
MANN_a9	45	918	72	16	17	18	19	20	26	36	36	45
MANN_a27	378	70,551	702	126	≥ 21	≥ 19	≥ 18	≥ 18	≥ 235	≥ 351	≥ 351	≥ 351
MANN_a45	1035	533,115	1980	345	≥ 21	≥ 19	≥ 18	≥ 18	≥ 661	≥ 990	≥ 990	≥ 990
p_hat300-1	300	10,933	33,917	8	9	9	10	10	10	12	≥ 13	≥ 14
p_hat300-2	300	21,928	22,922	25	26	≥ 22	≥ 23	≥ 22	≥ 29	≥ 27	≥ 27	≥ 28
p_hat300-3	300	33,390	11,460	36	≥ 26	≥ 26	≥ 24	≥ 24	≥ 31	≥ 32	≥ 31	≥ 31
p_hat500-1	500	31,569	93,181	9	10	11	≥ 11	≥ 11	12	≥ 13	≥ 14	≥ 14
p_hat500-2	500	62,946	61804	36	≥ 32	≥ 28	≥ 27	≥ 24	≥ 34	≥ 31	≥ 29	≥ 30
p_hat500-3	500	93,800	30,950	50	≥ 30	≥ 28	≥ 28	≥ 28	≥ 35	≥ 35	≥ 33	≥ 35
p_hat700-1	700	60,999	183,651	11	12	12	≥ 11	≥ 11	13	≥ 13	≥ 13	≥ 13
p_hat700-2	700	121,728	122,922	44	≥ 26	≥ 25	≥ 25	≥ 23	≥ 31	≥ 30	≥ 29	≥ 25
p_hat700-3	700	183,010	61,640	62	≥ 29	≥ 28	≥ 25	≥ 24	≥ 32	≥ 29	≥ 29	≥ 30
san200_07_1	200	13,930	5970	30	≥ 18	≥ 18	≥ 18	≥ 19	≥ 29	≥ 41	≥ 52	≥ 73
san200_07_2	200	13,930	5970	18	≥ 15	≥ 15	≥ 15	≥ 16	≥ 24	≥ 34	≥ 46	≥ 56
san200_09_1	200	17,910	1990	70	≥ 36	≥ 34	≥ 35	≥ 34	≥ 67	125	≥ 38	≥ 40
san200_09_2	200	17,910	1990	60	≥ 30	≥ 28	≥ 27	≥ 28	≥ 42	≥ 47	≥ 43	≥ 46
san200_09_3	200	17,910	1990	44	≥ 28	≥ 26	≥ 25	≥ 25	≥ 42	≥ 35	≥ 38	≥ 43
san400_0.5_1	400	39,900	39,900	13	≥ 9	≥ 10	≥ 11	≥ 11	≥ 14	≥ 20	≥ 26	≥ 31

Table 1 continued

Graph	V	E	$\bar{E}$	$\omega(G)$	$\omega_s^d(G)$ for $s = \dots$				$\omega_s(G)$ for $s = \dots$				
					1	2	3	4	2	3	4	5	
san400_0.7_1	400	55,860	23,940	40	≥ 20	≥ 20	≥ 21	≥ 22	≥ 34	≥ 48	≥ 70	≥ 90	
san400_0.7_2	400	55,860	23,940	30	≥ 17	≥ 18	≥ 18	≥ 19	≥ 28	≥ 41	≥ 51	≥ 56	
san400_0.7_3	400	55,860	23,940	22	≥ 15	≥ 16	≥ 16	≥ 17	≥ 23	≥ 33	≥ 44	≥ 54	
samr200_07	200	138,68	6032	18	≥ 19	≥ 18	≥ 17	≥ 17	≥ 20	≥ 21	≥ 22	≥ 24	
samr200_09	200	17,863	2037	42	≥ 27	≥ 28	≥ 27	≥ 27	≥ 33	≥ 37	≥ 40	≥ 43	

First set of experiments: Starting points used in the analytical bounds. Local optima computed by the CONOPT solver strongly depend on the starting point used. In the first set of experiments, we tried three different initial solution choices for both the maximum s -defective clique problem (with $s = 1, 2, 3, 4$) and the maximum s -plex problem (with $s = 2, 3, 4, 5$). Namely, the starting solutions correspond to the points used to obtain the analytical bounds in Sect. 3, described next.

1. *The Turán points:* The points employed in establishing the extensions of Turán's theorem, (29) for s -defective clique, and (35) for s -plex. The corresponding results are presented in detail in Table A1 (of the online supplement) for s -defective clique and in Table A4 for s -plex.
2. *The Wilf points:* The points used to prove the generalizations of Wilf's bound, (31) for s -defective clique and (37) for s -plex. See Tables A2 and A5 for the results for s -defective clique and s -plex, respectively.
3. *The degree-based points:* The points used to obtain the degree-based bounds, (33) for s -defective clique and (39) for s -plex. Tables A3 and A6 show the corresponding results.

Table 2 presents a summary of results for the first set of experiments. More specifically, the table contains the results of pairwise comparisons between the solutions obtained using RDS and CONOPT applied to the cubic formulations with the three different starting points (Turán, Wilf, and degree-based). The diagonal entries in this table show the number of instances on which the corresponding method yielded the best result obtained. A non-diagonal entry a - b in row i and column j ($i, j \in \{\text{RDS, Turán, Wilf, Degree}\}$) indicates that method i beat method j on a instances, and method j beat method i on b instances out of 200 (50 graphs $\times$ four different values of s); thus, the methods i and j were tied on the remaining $200 - (a + b)$ instances.

We observe that the Turán, Wilf, and degree-based points perform similarly for s -defective clique. Meanwhile, on the s -plex instances, there is a clear hierarchy, with the Wilf points outperforming the degree-based points, and the degree-based points outperforming the Turán points. One explanation for this phenomenon is the amount of information they exploit. For example, the Turán points, which use no instance-specific information (besides the number of vertices), perform the worst. One explanation for why the hierarchy is more apparent on the s -plex instances than on the s -defective clique instances is that the s -plex numbers are themselves more varying; recall that the s -defective clique number satisfies $\omega(G) \leq \omega_s^d(G) \leq \omega(G) + s$, while the s -plex number satisfies $\omega(G) \leq \omega_s(G) \leq s\omega(G)$, see [7].

As one would expect, RDS outperforms the cubic formulations in terms of quality of the solutions; however, the fact that the results obtained using CONOPT are better than those for RDS on many of the instances is rather encouraging. Moreover, the time limit used for RDS in [22] was 600 seconds, whereas the local optima of the cubic formulations were computed much quicker with CONOPT. This motivated the second set of experiments, described next.

Second set of experiments: Random starting points under a time limit. The second set of experiments evaluates the performance of the formulations by recording the quality of local maxima found by CONOPT within a 600-second time limit, starting

Table 2 Summary of results for the first set of experiments

	RDS	Turán	Wilf	Degree
<i>s-defective clique</i>				
RDS	121	85–76	86–76	88–76
Turán	76–85	90	34–35	29–31
Wilf	76–86	35–34	89	34–35
Degree	76–88	31–29	35–34	88
<i>s-plex</i>				
RDS	131	107–55	95–64	98–63
Turán	55–107	51	22–94	27–63
Wilf	64–95	94–22	80	60–34
Degree	63–98	85–27	34–60	71

The diagonal entries show the number of instances on which the corresponding method yielded the best result obtained. An entry a - b in row i and column j indicates that method i beat method j on a instances, and method j beat method i on b instances out of 200

from up to 100 randomly generated initial feasible solutions. The chosen time limit matches that used in [22] for RDS. Tables A7 and A8 (of the online supplement) give results for the maximum s -defective clique problem. Table A7 reports the values for the best solution found, the average quality of the solution found, as well as the standard deviation for the objective value, and Table A8 shows the number of runs performed within the 600-second time limit, as well as the average time per run and the standard deviation for the running times. Similarly, Tables A9 and A10 give results for the maximum s -plex problem.

Table 3 provides a comparison with RDS, where “max” and “mean” stand for the best and average solution found using the random starting points, respectively. An entry a - b in this table means that the best or average solution found in this set of experiments was better than that reported for RDS on a instances, and worse on b instances, out of 50 instances for each value of s , or out of 200 instances total for all considered s values. Clearly, the approach based on the cubic formulations outperforms RDS in terms of the best solution found within the allocated time limit. This advantage can be explained by the relatively high level of diversification achieved in the second set of experiments via randomization. The RDS approach favors intensification over diversification in search for the optimal solution. Its performance strongly depends on the ordering of vertices used, which is fixed once it has been decided. The method continues exploring the branch-and-bound tree without revisiting the selected ordering, even when little to no progress has been made for a while. On the other hand, the random starting points used in conjunction with a local optimization solver in our experiments allow to obtain a diverse set of local optima, which increases the chance of finding a high-quality solution. It is notable that even the average quality of the solutions found using random initial points for the cubic formulation is better than that of RDS solutions on a considerable number of instances.

We would like to conclude this section by remarking that the preliminary numerical experiments conducted here are by no means conclusive. Still, we hope that these

Table 3 Summary of results for the second set of experiments

	s -defective clique, $s = \dots$					s -plex, $s = \dots$				
	1	2	3	4	All	2	3	4	5	All
Max	20–8	23–8	25–7	27–4	95–27	21–7	26–13	27–16	23–19	97–55
Mean	18–28	20–28	22–26	24–25	84–107	20–30	23–27	19–30	18–32	70–119

An entry *a*–*b* in the first (second) row indicates that the best (average) solution found was better than that obtained by RDS on *a* instances, and worse on *b* instances

initial experiments show sufficient promise to stimulate future work in this direction, including development of global optimization algorithms based on the proposed cubic formulations. While specialized algorithms for solving non-convex quadratic problems are gradually becoming an integral part of standard optimization engines [23,38], research focusing on cubic programming is still rather scarce. We hope that the formulations developed in this paper will encourage work towards filling this gap. At the very least, the proposed formulations yield challenging and interesting cubic test instances for continuous global optimization solvers; also see [5,16,17].

5 Conclusion

The paper develops continuous cubic formulations for the maximum s -defective clique problem and the maximum s -plex problem, extending the well-known Motzkin–Straus quadratic formulation for the maximum clique to the considered clique relaxations. The proof of the formulation for the case of maximum s -plex establishes interesting connections to classical results regarding the fractional b -matching polytope. The proposed formulations are used to obtain analytical bounds on the s -defective clique number and the s -plex number of a graph, generalizing three of the standard bounds on the clique number. The numerical experiments conducted using the CONOPT solver demonstrate that the developed formulations can be used to quickly obtain high-quality lower bounds for the considered cluster detection problems.

The results presented in this paper invite several directions for future research. In particular, analyzing optimality conditions for the proposed formulations could lead to non-trivial continuous characterizations for the considered discrete problems, as was the case with the Motzkin–Straus formulation for the maximum clique problem [20]. The obtained local optima of the cubic formulations could likely be improved using tailored stochastic global optimization methods, such as a monotonic basin hopping strategy that was successfully used for several applications [21,28]. In addition, the special structure of the proposed formulations could be exploited to develop methods based on bilevel optimization and decomposition techniques. Another direction to investigate is whether meaningful generalizations of the Motzkin–Straus formulation can be obtained for other important clique relaxation models, such as quasi-clique and s -club [36]. Finally, from a graph theory perspective, Turán’s theorem is known to have several elegant proofs [3]. It would be interesting to explore whether the techniques used in each proof could be extended to obtain the generalizations of Turán’s theorem presented here.

Acknowledgements This material is based upon work supported by the National Science Foundation under Grants Nos. 1538493 and 1662757. The support of AFOSR Grant FA9550-19-1-0161 and AFRL Award FA8651-16-2-0009 is also gratefully acknowledged.

References

1. Ahmadi, A., Olshevsky, A., Parrilo, P., Tsitsiklis, J.: NP-hardness of deciding convexity of quartic polynomials and related problems. *Math. Program.* **137**, 453–476 (2013)
2. Aigner, M.: Turán's graph theorem. *Am. Math. Month.* **102**, 808–816 (1995)
3. Aigner, M., Ziegler, G.M.: *Proofs from THE BOOK*, 6th edn. Springer, Berlin (2018)
4. ARKI Consulting & Development A/S CONOPT. <http://www.conopt.com/>. Accessed March 2020
5. Balasundaram, B., Butenko, S.: Constructing test functions for global optimization using continuous formulations of graph problems. *Optim. Methods Softw.* **20**(4–5), 439–452 (2005)
6. Balasundaram, B., Butenko, S., Hicks, I.: Clique relaxations in social network analysis: the maximum k -plex problem. *Oper. Res.* **59**, 133–142 (2011)
7. Balasundaram, B., Chandramouli, S.S., Trukhanov, S.: Approximation algorithms for finding and partitioning unit-disk graphs into co- k -plexes. *Optim. Lett.* **4**(3), 311–320 (2010)
8. Bomze, I.M.: Evolution towards the maximum clique. *J. Global Optim.* **10**, 143–164 (1997)
9. Bomze, I.M.: On standard quadratic optimization problems. *J. Global Optim.* **13**(4), 369–387 (1998)
10. Bomze, I. M., Budinich, M., Pardalos, P. M., Pelillo, M.: The maximum clique problem. In: Du, D.-Z., Pardalos, P. M. (Eds.) *Handbook of Combinatorial Optimization*, pp. 1–74. Kluwer Academy Publishers (1999)
11. Bomze, I.M., Dür, M., de Klerk, E., Roos, C., Quist, A.J., Terlaky, T.: On copositive programming and standard quadratic optimization problems. *J. Global Optim.* **18**, 301–320 (2000)
12. Bonami, P., Lodi, A., Schweiger, J., Tramontani, A.: Solving quadratic programming by cutting planes. *SIAM J. Optim.* **29**, 1076–1105 (2019)
13. Busygin, S.: A new trust region technique for the maximum weight clique problem. *Discret. Appl. Math.* **154**, 2080–2096 (2006)
14. de Klerk, E., Pasechnik, D.V.: Approximation of the stability number of a graph via copositive programming. *SIAM J. Optim.* **12**(4), 875–892 (2002)
15. Edmonds, J.: Maximum matching and a polyhedron with 0–1 vertices. *J. Res. Natl. Bureau Stand.* **69B**, 125–130 (1965)
16. Floudas, C.A., Pardalos, P.M., Adjiman, C., Esposito, W.R., Günius, Z.H., Harding, S.T., Klepeis, J.L., Meyer, C.A., Schweiger, C.A.: *Handbook of Test Problems in Local and Global Optimization*, vol. 33. Springer, Berlin (2013)
17. Furini, F., Traversi, E., Belotti, P., Frangioni, A., Gleixner, A., Gould, N., Liberti, L., Lodi, A., Misener, R., Mittelmann, H., et al.: QPLIB: a library of quadratic programming instances. *Math. Program. Comput.* **11**(2), 237–265 (2019)
18. Gerards, A.M.H.: Matching. In: Ball, M.O. (ed.) *Handbooks in OR and MS*, ch 3, vol. 7, pp. 135–224. Elsevier, New York (1995)
19. Gibbons, L.E., Hearn, D.W., Pardalos, P.M.: A continuous based heuristic for the maximum clique problem. In: Johnson, D.S., Trick, M.A. (eds.) *Cliques, Coloring and Satisfiability: Second DIMACS Implementation Challenge*. DIMACS Series on Discrete Mathematics and Theoretical Computer Science, pp. 103–124. American Mathematical Society, Providence (1996)
20. Gibbons, L.E., Hearn, D.W., Pardalos, P.M., Ramana, M.V.: Continuous characterizations of the maximum clique problem. *Math. Oper. Res.* **22**, 754–768 (1997)
21. Grosso, A., Jamali, A., Locatelli, M., Schoen, F.: Solving the problem of packing equal and unequal circles in a circular container. *J. Global Optim.* **47**(1), 63–81 (2010)
22. Gschwind, T., Irnich, S., Podlinski, I.: Maximum weight relaxed cliques and Russian Doll Search revisited. *Discret. Appl. Math.* **234**, 131–138 (2018)
23. Gurobi Optimization. Gurobi optimizer 9.0. https://www.gurobi.com/wp-content/plugins/hd_documentations/documentation/9.0/refman.pdf (2020)
24. Hungerford, J.T., Rinaldi, F.: A general regularized continuous formulation for the maximum clique problem. *Math. Oper. Res.* **44**, 1161–1173 (2019)
25. Johnson, D.S., Trick, M.A. (eds.): *Cliques, Coloring and Satisfiability: Second DIMACS Implementation Challenge*. DIMACS Series on Discrete Mathematics and Theoretical Computer Science. American Mathematical Society, Providence (1996)
26. Johnson, E.L.: *Programming in networks and graphs*. Technical Report ORC 65-1, Operations Research Center, University of California, Berkeley (1965)
27. Liuzzi, G., Locatelli, M., Piccialli, V.: A new branch-and-bound algorithm for standard quadratic programming problems. *Optim. Methods Softw.* **34**, 79–97 (2019)

28. Locatelli, M., Schoen, F.: Efficient algorithms for large scale global optimization: Lennard-Jones clusters. *Comput. Optim. Appl.* **26**(2), 173–190 (2003)
29. McClosky, B., Hicks, I.V.: Combinatorial algorithms for the maximum k -plex problem. *J. Combin. Optim.* **23**, 29–49 (2012)
30. McClosky, B., Hicks, I.V.: The co-2-plex polytope and integral systems. *SIAM J. Discret. Math.* **23**, 1135–1148 (2009)
31. Miao, Z., Balasundaram, B.: Approaches for finding cohesive subgroups in large-scale social networks via maximum k -plex detection. *Networks* **69**, 388–407 (2017)
32. Moser, H., Niedermeier, R., Sorge, M.: Exact combinatorial algorithms and experiments for finding maximum k -plexes. *J. Combin. Optim.* **24**, 347–373 (2012)
33. Motzkin, T.S., Straus, E.G.: Maxima for graphs and a new proof of a theorem of Turán. *Can. J. Math.* **17**, 533–540 (1965)
34. Nogueira, B., Pinheiro, R.G.S.: A GPU based local search algorithm for the unweighted and weighted maximum s -plex problems. *Ann. Oper. Res.* **284**(1), 367–400 (2020)
35. Pardalos, P.M., Phillips, A.T.: A global optimization approach for solving the maximum clique problem. *Int. J. Comput. Math.* **33**(3–4), 209–216 (1990)
36. Pattillo, J., Youssef, N., Butenko, S.: On clique relaxation models in network analysis. *Eur. J. Oper. Res.* **226**, 9–18 (2013)
37. Pelillo, M., Jagota, A.: Feasible and infeasible maxima in a quadratic program for maximum clique. *J. Artif. Neural Netw.* **2**, 411–420 (1995)
38. Sahinidis, N.V.: BARON: a general purpose global optimization software package. *J. Global Optim.* **8**(2), 201–205 (1996)
39. Seidman, S.B., Foster, B.L.: A graph theoretic generalization of the clique concept. *J. Math. Sociol.* **6**, 139–154 (1978)
40. Seshadri, A.: Perron root computation, version 1.3.0.0. MathWorks File Exchange, <https://www.mathworks.com/matlabcentral/fileexchange/22763>, 2009. Accessed December 2019
41. Sherali, H.D., Smith, J.C.: A polyhedral study of the generalized vertex packing problem. *Math. Program.* **107**(3), 367–390 (2006)
42. Stozhkov, V., Pastukhov, G., Boginski, V., Pasiliao, E.L.: New analytical lower bounds on the clique number of a graph. *Optim. Methods Softw.* **32**(2), 336–368 (2017)
43. Trukhanov, S., Balasubramaniam, C., Balasundaram, B., Butenko, S.: Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations. *Comput. Optim. Appl.* **56**, 113–130 (2013)
44. Turán, P.: On an extremal problem in graph theory. *Matematikai és Fizikai Lapok* **48**, 436–452 (1941). (in Hungarian)
45. Vavasis, S.A.: *Nonlinear Optimization: Complexity Issues*. Oxford University Press Inc, New York, NY (1991)
46. Wilf, H.S.: Spectral bounds for the clique and independence numbers of graphs. *J. Combin. Theory Ser. B* **40**(1), 113–117 (1986)
47. Yu, H., Paccanaro, A., Trifonov, V., Gerstein, M.: Predicting interactions in protein networks by completing defective cliques. *Bioinformatics* **22**, 823–829 (2006)
48. Zhou, Y., Hao, J.K.: Frequency-driven Tabu search for the maximum s -plex problem. *Comput. Oper. Res.* **86**, 65–78 (2017)

Affiliations

Vladimir Stozhkov¹ · Austin Buchanan²  · Sergiy Butenko³ · Vladimir Boginski⁴

✉ Austin Buchanan
buchanan@okstate.edu

Vladimir Stozhkov
vstozhkov@fedex.com

Sergiy Butenko
butenko@tamu.edu

Vladimir Boginski
vladimir.boginski@ucf.edu

- ¹ FedEx Express, Memphis, TN, USA
- ² Oklahoma State University, Stillwater, OK, USA
- ³ Texas A&M University, College Station, TX, USA
- ⁴ University of Central Florida, Orlando, FL, USA

Polyhedral Properties of the Induced Cluster Subgraphs

S. Mohammad Hosseinian, Sergiy Butenko

*Texas A&M University, Department of Industrial and Systems Engineering
College Station, TX 77843-3131, {hosseinian, butenko}@tamu.edu*

Abstract

A cluster graph is a graph whose every connected component is a complete graph. Given a simple undirected graph G , a subset of vertices inducing a cluster graph is called an independent union of cliques (IUC), and the IUC polytope associated with G is defined as the convex hull of the incidence vectors of all IUCs in the graph. The MAXIMUM IUC problem, which is to find a maximum-cardinality IUC in a graph, finds applications in network-based data analysis. In this paper, we derive several families of facet-defining valid inequalities for the IUC polytope. We also give a complete description of this polytope for some special classes of graphs. We establish computational complexity of the separation problem for most of the considered families of valid inequalities and explore the effectiveness of employing the corresponding cutting planes in an integer (linear) programming framework for the MAXIMUM IUC problem through computational experiments.

Keywords: cluster subgraphs, independent union of cliques, network clustering, integer programming

1. Introduction

A graph is called a *cluster graph* if its every connected component is a complete graph. Given a simple undirected graph $G = (V, E)$, with the set of vertices V and the set of edges E , an *independent union of cliques* (IUC) is a subset of vertices $U \subseteq V$ inducing a cluster graph. The MAXIMUM IUC problem is to find an IUC of maximum cardinality in G . The importance of this problem stems from its application in *graph clustering* (partitioning the vertices of a graph into cohesive subgroups), which is a fundamental task in unsupervised data analysis; see [46] and the references therein for a survey of graph clustering applications and methods. A cluster graph is an ideal instance from the standpoint of cluster analysis—as it is readily composed of mutually disjoint “tightly knit” subgroups—and the MAXIMUM IUC problem aims to identify the largest induced cluster graph contained in an input graph G . Such an induced subgraph discloses the natural cores (centers) of the input graph clusters. Besides the practical importance of this problem, the definition of IUC subsumes two fundamental structures in a graph, i.e., cliques and independent sets. The

MAXIMUM CLIQUE and MAXIMUM INDEPENDENT SET problems are among the most popular problems of combinatorial optimization, several variants of which have been studied in the literature; see e.g. [32, 8, 53, 27, 40, 43]. Therefore, the study of the MAXIMUM IUC problem, that admits both of these complementary structures as feasible solutions, is also of a particular theoretical interest.

The MAXIMUM IUC problem has appeared under different names in the literature. Fomin et al. [22] introduced this problem as the MAXIMUM INDUCED CLUSTER SUBGRAPH problem, and proposed an exact $\mathcal{O}(1.6181^n n^{\mathcal{O}(1)})$ -time algorithm for it, where $n = |V|$. Ertem et al. [20] considered the MAXIMUM IUC problem as a relaxation of the MAXIMUM α -CLUSTER problem with the maximum local clustering coefficient, i.e., $\alpha = 1$. They proposed a graph clustering algorithm based on a given maximum IUC, referred to as disjoint 1-clusters in [20], and showed the competence of their method via experiments with real-life social networks. It was not until very recently that the term “independent union of cliques” was used by Ertem et al. [19]. In this work, the authors studied some basic properties of IUCs and analyzed complexity of the MAXIMUM IUC problem on some restricted classes of graphs. They also performed computational experiments using a combinatorial algorithm (Russian Doll Search) and an integer (linear) programming formulation of this problem. To the best of our knowledge, this is the only work containing computational experiments on the MAXIMUM IUC problem, and their results show that this problem is quite challenging for the exact solution methods. It is known that a graph is a cluster graph if and only if it contains no induced subgraph isomorphic to P_3 (the path graph on three vertices) [26]. In this regard, the MAXIMUM IUC problem has also been referred to as the MAXIMUM INDUCED P_3 -FREE SUBGRAPH problem [22]. In our presentation to follow, we refer to induced subgraphs isomorphic to P_3 as *open triangles*.

Evidently, finding a maximum IUC in G is equivalent to finding a minimum number of vertices whose deletion turns G into a cluster graph. The latter is the optimization version of the CLUSTER VERTEX DELETION problem (k -CVD), which asks if an input graph G can be transformed into a cluster graph by deleting at most k vertices, and is known to be NP-hard by Lewis and Yannakakis theorem [36]. The study of this problem started from the viewpoint of parameterized complexity with the work of Gramm et al. [26], who proposed an $\mathcal{O}(2.26^k + nm)$ -time fixed-parameter tractable algorithm for k -CVD, where $n = |V|$ and $m = |E|$. Later, Hüffner et al. [30] and Boral et al. [9] improved this result to $\mathcal{O}(2^k k^9 + nm)$ and $\mathcal{O}(1.9102^k (n + m))$, respectively. Le et al. [35] showed that k -CVD admits a kernel with $\mathcal{O}(k^{5/3})$ vertices, which means there exists a polynomial-time algorithm that converts an n -vertex instance of k -CVD to an equivalent instance with $\mathcal{O}(k^{5/3})$ vertices. Fomin et al. [23] used the parameterized results of [9] to show that the MINIMUM CVD problem (hence, MAXIMUM IUC) is solvable in $\mathcal{O}(1.4765^{n+o(n)})$ time, which is better than the former result of [22]. It is also known that the MINIMUM CVD problem is approximable to within a constant factor. In particular, You et al. [54] proposed a $\frac{5}{2}$ -approximation algorithm for it, and Fiorini et al. [21] improved the approximation ratio to $\frac{7}{3}$. Several other problems involving cluster subgraphs

(not necessarily induced) have also been considered in the literature, including CLUSTER EDITING [47, 18, 6, 33, 7, 24, 2], DISJOINT CLIQUES [31, 1], s -PLEX CLUSTER VERTEX DELETION [51], and s -PLEX CLUSTER EDITING [28].

From a broader perspective, the MAXIMUM IUC problem is to identify a maximum-cardinality *independent set* in a graph-based *independence system*. An independence system $(S, \mathcal{S})$ is a pair of a finite set S together with a family $\mathcal{S}$ of subsets of S such that $I_2 \subseteq I_1, I_1 \in \mathcal{S}$ implies $I_2 \in \mathcal{S}$. Referring to an independence system $(S, \mathcal{S})$, every $I \in \mathcal{S}$ is called an *independent set* and every $D \subseteq S, D \notin \mathcal{S}$, is called a *dependent set*. The minimal (inclusion-wise) dependent subsets of S are called *circuits* of the independence system $(S, \mathcal{S})$. In graph-theory terminology, an independent set (stable set, vertex packing) is a subset of pairwise non-adjacent vertices in a graph $G = (V, E)$. Let $\mathcal{I}$ denote the set of all independent sets (in the sense of graph theory) in G . Then, $(V, \mathcal{I})$ is an independence system whose circuit set is given by E ; hence, the two definitions coincide. Following this concept, V together with the set of all IUCs in G form an independence system whose set of circuits is given by the three-vertex subsets of V inducing open triangles in the graph. This perspective on the MAXIMUM IUC problem is of our interest as some of the results to follow correspond to certain properties of independence systems. It is worth noting that many well-known problems of graph theory, beyond those mentioned here, convey the notion of independence systems associated with a graph; see for example [34, 44].

This paper presents a comprehensive study of the MAXIMUM IUC problem from the standpoint of polyhedral combinatorics. More specifically, we study the facial structure of the IUC polytope associated with a simple undirected graph G , and identify several facet-defining valid inequalities for this polytope. As a result, we obtain the full description of the IUC polytope for some special classes of graphs. We also study the computational complexity of the separation problems for these inequalities, and perform computational experiments to examine their effectiveness when used in a branch-and-cut scheme to solve the MAXIMUM IUC problem. The organization of this paper is as follows: we continue this section by introducing the terminology and notation used in this paper. In Section 2, the IUC polytope is defined and the strength of its fractional counterpart—obtained from relaxing the integrality of the variables in definition of the original polytope—is discussed. In Section 3, we study some facet-producing structures for the IUC polytope and present the corresponding facet-inducing inequalities, as well as convex hull characterization (when possible). In our study, we pay special attention to similarities between the facial structure of the IUC polytope and those of the independent set (vertex packing) and clique polytopes. In Section 4, we investigate the computational complexity of the separation problem for each class of the proposed valid inequalities. Section 5 contains the results of our computational experiments, and Section 6 concludes this paper.

1.1. Terminology and Notation

Throughout this paper, we consider a simple undirected graph $G = (V, E)$, where V is the vertex set and $E \subseteq \{\{i, j\} \mid i, j \in V, i \neq j\}$ is the edge set. Unless otherwise stated, we assume $|V| = n$. Given a subset of vertices $S \subseteq V$, the subgraph *induced* by S is denoted by $G[S]$ and is defined as $G[S] = (S, E(S))$, where $E(S)$ is the set of edges with both endpoints in S . A *clique* is a subset of vertices inducing a complete graph, and an *independent set* (stable set, vertex packing) is a subset of vertices inducing an edgeless graph. A clique (resp. independent set) is called *maximum* if it is of maximum cardinality, i.e., there is no larger clique (resp. independent set) in the graph. Cardinality of a maximum clique in G is called the *clique number* of the graph, and is denoted by $\omega(G)$. Respectively, the *independence number* (stability number) of G is the cardinality of a maximum independent set in the graph, and is denoted by $\alpha(G)$. In a similar manner, a *maximum IUC* in G is an IUC of maximum cardinality, and the *IUC number* of G , denoted by $\alpha^\omega(G)$, is the cardinality of a maximum IUC in the graph. Since every clique as well as every independent set in G is an IUC, the IUC number of the graph is bounded from below by its clique number and independence number, i.e., $\max\{\omega(G), \alpha(G)\} \leq \alpha^\omega(G)$. The (open) *neighborhood* of a vertex $i \in V$ is the set of vertices $j \in V \setminus \{i\}$ adjacent to i , and is denoted by $N(i)$. The *closed neighborhood* of i is defined as $N[i] = N(i) \cup \{i\}$. The *incidence vector* of a subset of vertices $S \subseteq V$ is a binary vector $x \in \mathbb{R}^n$ such that $x_i = 1, \forall i \in S$, and $x_i = 0, \forall i \in V \setminus S$. Conventionally, the incidence vector of a single vertex $i \in V$ is denoted by e_i . We use $\mathbf{0}$, $\mathbf{1}$ and $\mathbf{2}$ to denote the vectors of all zero, one and two, respectively. The corresponding dimensions will be clear from the context.

We assume familiarity of the reader with fundamentals of polyhedral theory, and just briefly mention the concept of *lifting* [42, 52, 55], as several proofs to follow are based on lifting arguments. Let $\mathcal{P}$ be the convex hull of the set of 0–1 feasible solutions to an arbitrary system of linear inequalities with a nonnegative coefficient matrix, i.e., $\mathcal{P} = \text{conv}\{x \in \{0, 1\}^n \mid Ax \leq b\}$. Corresponding to a subset of variables indexed by $N' \subseteq N = \{1, \dots, n\}$, consider the polytope $\mathcal{P}_{(x_{N'}=0)}$ obtained from $\mathcal{P}$ by setting $x_j = 0, \forall j \in N'$. Let $N'' = N \setminus N'$, and suppose that the inequality $\sum_{i \in N''} \pi_i x_i \leq \pi_0$ induces a facet of $\mathcal{P}_{(x_{N'}=0)}$. Then, lifting a variable $x_j, j \in N'$, into this inequality leads to an inequality $\pi_j x_j + \sum_{i \in N''} \pi_i x_i \leq \pi_0$, where

$$\pi_j = \pi_0 - \max \left\{ \sum_{i \in N''} \pi_i x_i \mid \sum_{i \in N''} A_i x_i \leq b - A_j; x_i \in \{0, 1\}, \forall i \in N'' \right\}, \quad (1)$$

which is facet-defining for $\mathcal{P}_{(x_{N' \setminus \{j\}}=0)}$. In (1), A_i denotes the coefficient vector of a variable x_i in the system of inequalities, i.e., the i -th column of A . The importance of this result is due to the fact that some facets of the original polytope $\mathcal{P}$ can be obtained from the facets of the restricted polytopes through sequential (or simultaneous) lifting procedures. In particular, some facets of the polytope associated with a graph G for a certain problem may be identified

through lifting the facets of the lower-dimensional polytopes corresponding to its subgraphs. Finally, note that (1) is equivalent to

$$\pi_j = \pi_0 - \max \left\{ \sum_{i \in N''} \pi_i x_i \mid x \in \mathcal{P} \text{ and } x_j = 1 \right\}, \quad (2)$$

which is easier to use when $\mathcal{P}$ is defined to be the convex hull of the incidence vectors of the sets of vertices in a graph holding a certain property.

2. The IUC Polytope

The IUC polytope associated with a simple undirected graph G , denoted by $\mathcal{P}_{IUC}(G)$, is the convex hull of the incidence vectors of all IUCs in G , which can be characterized by the circuit set of the corresponding independence system as follows [38]:

$$\mathcal{P}_{IUC}(G) = \text{conv}\{x \in \{0, 1\}^n \mid x_i + x_j + x_k \leq 2, \forall \{i, j, k\} \in \Lambda(G)\}, \quad (3)$$

where $\Lambda(G)$ denotes the set of three-vertex subsets of V inducing open triangles in G . Patently, $\alpha^\omega(G) = \max\{\sum_{i \in V} x_i \mid x \in \mathcal{P}_{IUC}(G)\}$, and the MAXIMUM WEIGHT IUC problem, which is to find an IUC with the maximum total weight in a vertex-weighted graph, can be formulated as

$$\text{Maximize } \sum_{i \in V} w_i x_i \text{ subject to } x \in \mathcal{P}_{IUC}(G), \quad (4)$$

with w_i denoting the weight of a vertex $i \in V$. We call each inequality $x_i + x_j + x_k \leq 2, \{i, j, k\} \in \Lambda(G)$, an open-triangle (OT) inequality. The following theorems establish the basic properties of $\mathcal{P}_{IUC}(G)$ including the strength of the OT inequalities.

Theorem 1. *$\mathcal{P}_{IUC}(G)$ is a full-dimensional polytope, and for every $i \in V$, the inequalities $x_i \geq 0$ and $x_i \leq 1$ define its facets.*

PROOF. Note that $\mathbf{0} \in \mathcal{P}_{IUC}(G)$. Also, every single vertex, as well as every pair of vertices in G is an IUC, by definition. Full-dimensionality of $\mathcal{P}_{IUC}(G)$ is established by noting that $\mathbf{0}$ and $e_i, \forall i \in V$, are $n+1$ affinely independent points in $\mathbb{R}^n$ belonging to this polytope. Besides, for every $i \in V$, $\mathbf{0}$ and $e_j, \forall j \in V \setminus \{i\}$, belong to $\mathcal{F}_i^0 = \{x \in \mathcal{P}_{IUC}(G) \mid x_i = 0\}$, which indicates that $\mathcal{F}_i^0$ is a facet of $\mathcal{P}_{IUC}(G)$. Similarly, for every $i \in V$, the points e_i and $e_{ij} = e_i + e_j, \forall j \in V \setminus \{i\}$, are affinely independent and belong to $\mathcal{F}_i^1 = \{x \in \mathcal{P}_{IUC}(G) \mid x_i = 1\}$, hence $\mathcal{F}_i^1$ is a facet of $\mathcal{P}_{IUC}(G)$.

It follows from full-dimensionality of $\mathcal{P}_{IUC}(G)$ that every facet-defining inequality of this polytope is unique to within a positive multiple. Thus, in order to prove that a proper face $\mathcal{F}$ of $\mathcal{P}_{IUC}(G)$ defined by a valid inequality $\pi(x) \leq \pi_0$ is a facet, it suffices to show that any supporting hyperplane of $\mathcal{P}_{IUC}(G)$ containing $\mathcal{F}$ is a (non-zero) scalar multiple of $\pi(x) = \pi_0$.

Theorem 2. *The inequality $x_i + x_j + x_k \leq 2$, $\{i, j, k\} \in \Lambda(G)$, is facet-defining for $\mathcal{P}_{\text{IUC}}(G)$ if and only if there does not exist a vertex $v \in V \setminus \{i, j, k\}$ such that $H = \{i, j, k, v\}$ induces a chordless cycle in G .*

PROOF. Clearly, $\mathcal{F} = \{x \in \mathcal{P}_{\text{IUC}}(G) \mid x_i + x_j + x_k = 2\}$ is a proper face of $\mathcal{P}_{\text{IUC}}(G)$. So, suppose that it is contained in a supporting hyperplane $\mathcal{H} : \sum_{v \in V} a_v x_v = b$ of this polytope. Observe that,

$$e_i + e_j \in \mathcal{F} \subseteq \mathcal{H} \quad \therefore \quad a_i + a_j = b.$$

Similarly, $a_i + a_k = b$ and $a_j + a_k = b$, which imply $a_i = a_j = a_k = a$ and $b = 2a$. Now, consider $G[H]$ the subgraph induced by $H = \{i, j, k, v\}$ for an arbitrary vertex $v \in V \setminus \{i, j, k\}$. As $G[H]$ is not a chordless cycle, it contains at most three open triangles, and those have at least one vertex in common. Let k be such a vertex, then

$$e_i + e_j + e_v \in \mathcal{F} \subseteq \mathcal{H} \quad \therefore \quad a_i + a_j + a_v = b \quad \therefore \quad a_v = 0, \quad \forall v \in V \setminus \{i, j, k\}.$$

As the interior of $\mathcal{P}_{\text{IUC}}(G)$ is nonempty, $a \neq 0$ and $\mathcal{H}$ is a (non-zero) scalar multiple of $x_i + x_j + x_k = 2$. To complete the proof, note that if $H = \{i, j, k, v\}$ induces a chordless cycle in G , then $x_i + x_j + x_k \leq 2$ is dominated by the inequality $x_i + x_j + x_k + x_v \leq 2$, which is facet-defining for $\mathcal{P}_{\text{IUC}}(G)$ by Theorem 3 of the next section.

In Section 3 we will show that Theorem 2 is a special case of a more general statement for the so-called *star inequality*, defined later.

Obviously, the MAXIMUM IUC problem can be directly solved using off-the-shelf MIP solvers through the following formulation:

$$\alpha^\omega(G) = \max \sum_{i \in V} x_i \tag{5a}$$

$$\text{s.t.} \quad x_i + x_j + x_k \leq 2, \quad \forall \{i, j, k\} \in \Lambda(G), \tag{5b}$$

$$x_i \in \{0, 1\}, \quad \forall i \in V. \tag{5c}$$

However, the LP relaxation of (5) is generally too weak. In fact, given the feasibility of the fractional assignment $x_i = \frac{2}{3}, \forall i \in V$, the optimal solution value of the LP relaxation problem is at least as large as $\frac{2n}{3}$, while the computational results of Ertem et al. [19] show that the IUC number of graphs with moderate densities tends to stay in a close range of their relatively small independence and clique numbers. This is due to the fact that the fractional IUC polytope $\mathcal{P}_{\text{IUC}}^F(G) = \{x \in [0, 1]^n \mid x_i + x_j + x_k \leq 2, \forall \{i, j, k\} \in \Lambda(G)\}$ is a polyhedral relaxation of a *cubically* constrained region in the space of original variables. Note that the IUC property may be enforced through trilinear products of the variables in a 0 – 1 program, i.e., by replacing the OT inequalities of (5) with $x_i x_j x_k = 0, \forall \{i, j, k\} \in \Lambda(G)$. In such a cubic formulation, relaxing the integrality of variables will not change the set of optimal solutions, because given a set of variables $x_i = 0, \forall i \in V_0 \subset V$, to satisfy the constraints, the optimality

conditions necessitate $x_i = 1, \forall i \in V \setminus V_0$. Hence, $\alpha^\omega(G)$ can be obtained by solving the following continuous problem:

$$\alpha^\omega(G) = \max_{x \in [0,1]^n} \left\{ \sum_{i \in V} x_i \mid x_i x_j x_k = 0, \forall \{i, j, k\} \in \Lambda(G) \right\}. \quad (6)$$

It is known that the (convex and concave) envelopes of a trilinear monomial term $w = xyz$ when $0 \leq x, y, z \leq 1$ are as follows [37]:

$$w \leq x, \quad w \leq y, \quad w \leq z, \quad w \geq x + y + z - 2, \quad w \geq 0,$$

which, along with $w_{ijk} = x_i x_j x_k = 0, \forall \{i, j, k\} \in \Lambda(G)$, indicates that $\mathcal{P}_{IUC}^F(G)$ is a polyhedral outer-approximation of the feasible region of (6). As a result, the gap between $\alpha^\omega(G)$ and the optimal solution value of the LP relaxation problem of (5) is due to accumulation of the gap between the trilinear terms and their overestimating envelopes. A similar result holds between continuous and integer (linear) programming formulations of the MAXIMUM INDEPENDENT SET and MAXIMUM CLIQUE problems, except that the corresponding continuous formulations of those problems are quadratically constrained. The cubic nature of the IUC formulation justifies the weakness of $\mathcal{P}_{IUC}^F(G)$ even compared to the fractional independent set (vertex packing) and clique polytopes, which are known to be loose in general. This further reveals the importance of exploring the facial structure of the IUC polytope and applying strong cutting planes in integer (linear) programming solution methods of the MAXIMUM IUC problem.

3. Facet-producing Structures

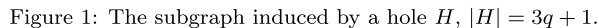
In this section, we present some facet-inducing valid inequalities for the IUC polytope associated with chordless cycle (and its edge complement), star (and double-star), fan and wheel graphs. We study the conditions under which these inequalities remain facet-defining for the IUC polytope of an arbitrary graph that contains the corresponding structure as an induced subgraph, and present some results with respect to lifting procedure for those that do not have this property. In some cases, we will present the full description of the IUC polytope associated with these graphs.

3.1. Chordless cycle and its complement

The following theorem concerns the facial structure of the IUC polytope associated with a (chordless) cycle graph. In an arbitrary graph $G = (V, E)$, a subset of vertices $H \subseteq V$, $|H| \geq 4$, inducing a chordless cycle is called a *hole*. In labeling the vertices of a hole H , we assume that adjacent vertices have consecutive labels with the convention $|H| + 1 \equiv 1$.

Theorem 3. (Hole Inequality) *Let $H \subseteq V$ be a hole of cardinality $|H| = 3q + r$ in $G = (V, E)$, where q is a positive integer and $r \in \{0, 1, 2\}$. Then, the inequality*

$$\sum_{i \in H} x_i \leq 2q + \left\lfloor \frac{2r}{3} \right\rfloor \quad (7)$$



PROOF. Consider $G[H]$ the subgraph induced by H . This subgraph contains $|H|$ distinct open triangles, and each vertex $i \in H$ appears in exactly three of them. Summation of the corresponding OT inequalities leads to

After dividing both sides of the last inequality by 3, integrality of the left-hand side implies validity of (7) for $\mathcal{P}_{\mathcal{M}SC}(G)$ as a Gomory-Chvátal cut.

- CASE 1: $r = 1$

$$U_1 = \{1, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\},$$

$$U_2 = \{2, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\}.$$

The sets of black vertices in Figures 1(a) and 1(b) depict U_1 and U_2 , respectively. Observe that, $x_{U_1}, x_{U_2} \in \mathcal{F} = \{x \in \mathcal{P}_{TUC}(G[H]) \mid \sum_{v \in H} x_v = 2q\}$, hence $\mathcal{F}$ is a proper face of $\mathcal{P}_{TUC}(G[H])$. Suppose that $\mathcal{H} : \sum_{v \in H} a_v x_v = b$

is an arbitrary supporting hyperplane of $\mathcal{P}_{IUC}(G[H])$ containing $\mathcal{F}$. Then, $x_{U_1}, x_{U_2} \in \mathcal{H}$ implies that $a_1 = a_2$. The same argument using similar partitions can be used to show that this result holds for every two adjacent vertices in $G[H]$, thus $a_v = a, \forall v \in H$, and $b = 2qa$. The proof is complete by noting that the interior of $\mathcal{P}_{IUC}(G[H])$ is nonempty, so $a \neq 0$.

– CASE 2: $r = 2$

Similar to the former case, let x_{U_1} and x_{U_2} be the incidence vectors of

$$\begin{aligned} U_1 &= \{3q + 2, 1, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\}, \\ U_2 &= \{3q + 2, 2, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\}. \end{aligned}$$

Then, the same argument leads to the result that the inequality $\sum_{v \in H} x_v \leq 2q + 1$ induces a facet of $\mathcal{P}_{IUC}(G[H])$.

(ii) To conclude the proof, we show that (7) is facet-defining for $\mathcal{P}_{IUC}(G)$ if $q = r = 1$, i.e., $|H| = 4$. By (i), $\sum_{i \in H} x_i \leq 2$ defines a facet of $\mathcal{P}_{IUC}(G[H])$. Let $\sum_{i \in H} x_i + \sum_{v \in V \setminus H} a_v x_v \leq 2$ be the inequality obtained from an arbitrary sequential lifting of $\sum_{i \in H} x_i \leq 2$ (to $\mathbb{R}^n$). Observe that, for every vertex $v \in V \setminus H$, there always exists two vertices $i, j \in H$ such that $\{i, j, v\} \notin \Lambda(G)$, thus, $\max\{\sum_{i \in H} x_i \mid x \in \mathcal{P}_{IUC}(G) \text{ and } x_v = 1\} = 2$. This implies $a_v = 0, \forall v \in V \setminus H$, hence $\sum_{i \in H} x_i \leq 2$ induces a facet of $\mathcal{P}_{IUC}(G)$.

It is easy to see the similarity between (7) and the *odd-hole* inequality for the vertex packing (independent set in the sense of graph theory) polytope associated with G . Padberg [41] proved that, given a hole H in G with an odd number of vertices, the inequality $\sum_{i \in H} x_i \leq \frac{1}{2}(|H| - 1)$ is facet-defining for the vertex packing polytope associated with $G[H]$. Later, Nemhauser and Trotter [38] noted that the odd-hole inequality is a special case of a more general result concerning independence systems, stated as follows:

Theorem 4. [38] *Suppose α_0 is the cardinality of a maximum independent set in an independence system $\mathcal{S} = (S, \mathcal{S})$, and $\mathcal{S}$ contains $n = |S|$ maximum independent sets $I_1, I_2, \dots, I_n$ with corresponding affinely independent (incidence) vectors $x^1, x^2, \dots, x^n$. Then $\sum_{i \in S} x_i \leq \alpha_0$ is facet-defining for the polytope associated with $\mathcal{S}$ (i.e., the convex hull of the incidence vectors of all members of $\mathcal{S}$ in $\mathbb{R}^n$).*

Similar to the odd-hole inequality for the vertex packing polytope, Theorem 3(i) spots a facet-inducing inequality implied by Theorem 4 for the independence system defined by the IUC property on $G[H]$. Note that Theorem 4 itself is of little practical value as it provides no means to identify the independence systems satisfying such properties.

Furthermore, Theorems 2 and 3(ii) correspond to *maximal clique* inequality of independence systems. An independence system $(S, \mathcal{S})$ is called k -regular if every circuit of it is of cardinality k . Referring to a k -regular independence system $(S, \mathcal{S})$, $C \subseteq S$ is called a *clique* if $|C| \geq k$ and all $\binom{|C|}{k}$ subsets of C are circuits of $(S, \mathcal{S})$. Clearly, this definition coincides with the graph-theoretic

definition of a clique. Nemhauser and Trotter [38] also proved that the inequality $\sum_{i \in C} x_i \leq k - 1$ is facet-inducing for the polytope associated with a k -regular independence system $(S, \mathcal{S})$ if $C \subseteq S$ is a maximal (inclusion-wise) clique. A special case of this result for the vertex packing polytope associated with a graph G had been originally shown by Padberg [41]. Let $\mathcal{U}$ denote the set of all IUCs in G , and $(V, \mathcal{U})$ be the corresponding independence system. Patently, every $\{i, j, k\} \in \Lambda(G)$ is a clique in $(V, \mathcal{U})$, and such a clique is maximal only if it is not contained in a clique of cardinality 4. It is not hard to see that a clique of size 4 in $(V, \mathcal{U})$ must be a hole in G . Therefore, Theorem 2 corresponds to maximality of $\{i, j, k\} \in \Lambda(G)$ as a clique of the corresponding independence system. Besides, by the proof of Theorem 3(ii), it became clear that $(V, \mathcal{U})$ cannot have a clique of cardinality greater than 4 because, for every 4-hole H and every vertex $v \in V \setminus H$, there always exist two vertices $i, j \in H$ such that $\{i, j, v\} \notin \Lambda(G)$. Thus, a hole of cardinality 4 in G is actually a maximum clique in $(V, \mathcal{U})$, and Theorem 3(ii) corresponds to its maximality. This result is of a special value for the MAXIMUM IUC problem as all facets of this type can be identified in polynomial time. Recall that the vertex packing polytope associated with G may possess an exponential number of such facets.

Given this result, we extend the concept of *fractional clique/independence number* of a graph G to the independence system $(V, \mathcal{U})$. The fractional clique number of G is the maximum sum of nonnegative weights that can be assigned to its vertices such that the total weight of every independent set in the graph is at most 1. Naturally, the fractional clique number of G is equivalent to the fractional independence number of its complement graph, where the total weight of every clique is at most 1. We define the *fractional IUC number* of a graph G as the maximum sum of nonnegative weights that can be assigned to its vertices such that the total weight of every clique in $(V, \mathcal{U})$, i.e., the vertex set of every open triangle and 4-hole in G , is at most 2. Clearly, the fractional IUC number of G provides an upper bound on its IUC number, which is computable in polynomial time as opposed to the fractional clique and independence numbers of a graph which are NP-hard to compute [27].

Next, we show that if G itself is a cycle graph, the inequalities identified by Theorems 2 and 3 along with the variable bounds are sufficient to describe the IUC polytope associated with G . To this end, we first state a preliminary result regarding the rank of the coefficient matrix of the OT inequalities of a cycle graph.

Lemma 1. *Let $Ax \leq \mathbf{2}$ denote the system of OT inequalities corresponding to a cycle graph G on $n \geq 4$ vertices. Then, A is full-rank if and only if $n \neq 3q$, for every positive integer q .*

PROOF. As it was pointed out in the proof of Theorem 3, G contains exactly n open triangles, hence A is a square matrix of order n . Consider the system of equations $\sum_{i=1}^n \alpha_i A_i = 0$, where A_i denotes the i -th column of A . Clearly, the equations corresponding to $\{i, i+1, i+2\} \in \Lambda(G)$ and $\{i+1, i+2, i+3\} \in \Lambda(G)$ in this system imply $\alpha_i = \alpha_{i+3}, \forall i \in \{1, \dots, n\}$, given the convention $n+i \equiv i$;

$i = 1, 2, 3$. Now, suppose $n = 3q$, for some positive integer q . This implies

$$\alpha_1 = \alpha_4 = \dots = \alpha_{n-2}, \quad \alpha_2 = \alpha_5 = \dots = \alpha_{n-1}, \quad \alpha_3 = \alpha_6 = \dots = \alpha_n,$$

and any combination of real numbers satisfying $\alpha_1 + \alpha_2 + \alpha_3 = 0$ is a solution to this system. Therefore, the columns of A are linearly dependent and A is not full-rank. On the other hand, if $n = 3q + 1$, then $\alpha_i = \alpha_{i+3}, \forall i \in \{1, \dots, n\}$, leads to

$$\alpha_1 = \alpha_4 = \dots = \alpha_n = \alpha_3 = \alpha_6 = \dots = \alpha_{n-1} = \alpha_2 = \alpha_5 = \dots = \alpha_{n-2},$$

which implies $\alpha_i = 0, \forall i \in \{1, \dots, n\}$. Clearly, the same result holds when $n = 3q + 2$, which establishes linear independency of columns of A and completes the proof.

Theorem 5. *Let G be a cycle graph on $n \geq 4$ vertices. Let q and r be the quotient and remainder of a division of n by 3, respectively (i.e., $n = 3q + r$), and $n + j \equiv j$ for $j = 1, 2$. Then,*

$$\mathcal{P}_{\text{IUC}}(G) = \{x \in [0, 1]^n \mid \begin{array}{l} x_i + x_{i+1} + x_{i+2} \leq 2, \forall i \in \{1, \dots, n\}, \\ \sum_{i \in V} x_i \leq 2q + \lfloor 2r/3 \rfloor. \end{array} \} \quad (8)$$

PROOF. Let $\mathcal{P}$ denote the polytope defined by the right-hand side of (8). Note that the last inequality in the definition of $\mathcal{P}$ is the hole inequality (7) written for $H = V$. Given the validity of (7) for $\mathcal{P}_{\text{IUC}}(G)$, we just need to show that every extreme point of $\mathcal{P}$ is integral. We start with the case where n is a multiple of 3, so (7) is redundant in the description of $\mathcal{P}$. Recall that $\hat{x}$ is an extreme point of $\mathcal{P}$ if and only if it lies on some n linearly independent supporting hyperplanes of this polytope. By Lemma 1, the coefficient matrix of the set of OT inequalities is rank deficient, thus at least one variable is tight at its bound in every extreme point of $\mathcal{P}$. Suppose x_i , for some $i \in \{1, \dots, n\}$, is such a variable in an arbitrary extreme point $\hat{x}$ of $\mathcal{P}$, and consider the corresponding facets $\mathcal{P}_{(x_i=0)} = \{x \in \mathcal{P} \mid x_i = 0\}$ and $\mathcal{P}_{(x_i=1)} = \{x \in \mathcal{P} \mid x_i = 1\}$. Observe that, the inequalities $x_{i-2} + x_{i-1} + x_i \leq 2$, $x_{i-1} + x_i + x_{i+1} \leq 2$, and $x_i + x_{i+1} + x_{i+2} \leq 2$ are redundant in description of $\mathcal{P}_{(x_i=0)}$, thus at most $n - 3$ (linearly independent) supporting hyperplanes of this facet are provided by the set of OT inequalities. Moreover, in description of $\mathcal{P}_{(x_i=1)}$, the OT inequalities involving x_i reduce to $x_{i-2} + x_{i-1} \leq 1$, $x_{i-1} + x_{i+1} \leq 1$, and $x_{i+1} + x_{i+2} \leq 1$. This implies redundancy of two other OT inequalities, that is $x_{i-3} + x_{i-2} + x_{i-1} \leq 2$ and $x_{i+1} + x_{i+2} + x_{i+3} \leq 2$, in description of this facet, which further indicates that OT inequalities can provide at most $n - 2$ (linearly independent) supporting hyperplanes for $\mathcal{P}_{(x_i=1)}$. Therefore, in either case, x_i is not the only integral component of $\hat{x}$. It is easy to verify that fixing any other variable at its (lower or upper) bound either makes more OT inequalities redundant in description of the corresponding faces, or forces another variable to be integral, which in turn reduces the number of OT inequalities in description of lower-dimensional faces of $\mathcal{P}$ containing $\hat{x}$. Clearly, this chain continues up to the zero-dimensional

face of $\mathcal{P}$ containing $\hat{x}$, i.e., $\hat{x}$ itself, where all coordinates are integral. This completes the proof for the case $n = 3q$, for every positive integer q .

Next, we examine the case where n is not a multiple of 3. Let $\hat{x}$ be an extreme point of $\mathcal{P}$, and assume (7) is not binding at this point. As A is full-rank, the system of OT equalities has a unique fractional solution given by $x_i = \frac{2}{3}, \forall i \in \{1, \dots, n\}$. Since this solution violates (7), $\hat{x}$ must have an integer component, which implies $\hat{x}$ is integral in its entirety by our former argument. Now, suppose (7) is binding at $\hat{x}$. In this case, exactly $n - 1$ OT inequalities must be binding at any fractional extreme point of $\mathcal{P}$ (if any), because all OT inequalities can not be active simultaneously with (7) and less number of them leaves at least one variable to be integer, which leads to integrality of $\hat{x}$. Let $n = 3q + 1$, for some positive integer q and, without loss of generality, assume $x_n + x_1 + x_2 \leq 2$ is the non-binding constraint. Then, the unique solution of the corresponding system of equations is given by

$$\hat{x}_1 = \hat{x}_4 = \dots = \hat{x}_n = \hat{x}_2 = \hat{x}_5 = \dots = \hat{x}_{n-2} = 0, \quad \hat{x}_3 = \hat{x}_6 = \dots = \hat{x}_{n-1} = 2,$$

which violates the variable bounds. Alternatively, if $n = 3q + 2$, the solution of this system of equations is given by

$$\hat{x}_1 = \hat{x}_4 = \dots = \hat{x}_{n-1} = \hat{x}_3 = \hat{x}_6 = \dots = \hat{x}_{n-2} = 1, \quad \hat{x}_2 = \hat{x}_5 = \dots = \hat{x}_n = 0,$$

which is integral. Consequently, such a fractional extreme point $\hat{x}$ does not exist, which completes the proof.

The proof of Theorem 5 also implies that the family of OT inequalities together with the variable bounds is sufficient to describe the IUC polytope associated with a path (chain) graph. A path graph on n vertices is denoted by P_n . We present this result through the following corollary.

Corollary 1. *Let $G = P_n, n \geq 3$. Then,*

$$\mathcal{P}_{IUC}(G) = \{x \in [0, 1]^n \mid x_i + x_{i+1} + x_{i+2} \leq 2, \forall i \in \{1, \dots, n-2\}\}. \quad (9)$$

PROOF. Consider a cycle graph $G' = (V', E')$ generated from $G = P_n$ by adding some additional vertices (and the corresponding edges) such that $|V'| = 3q$, for some positive integer q . Let V_0 denote the set of appended vertices, and observe that the right-hand side of (9) is precisely the description of the face of $\mathcal{P}_{IUC}(G')$ defined by $x_v = 0, \forall v \in V_0$. Thus, each extreme point of this polytope is integral by Theorem 5 and the incidence vector of an IUC in G . The proof is complete.

Given the definition of IUC, it is also natural to consider the correspondence between the IUC polytope and the clique polytope associated with a graph G , i.e., the convex hull of the incidence vectors of all cliques in G . Patently, the odd-hole inequality for the vertex packing polytope translates to the *odd-anti-hole* inequality for the clique polytope associated with G . An *anti-hole* is a subset of vertices $A \subseteq V$, $|A| \geq 4$, such that the (edge) complement of the

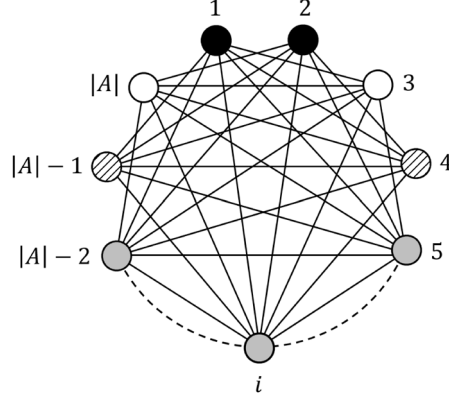


Figure 2: The subgraph induced by an anti-hole.

corresponding induced subgraph is a chordless cycle, i.e., A is a hole in the complement graph of G . The following theorem states that, under a slightly stronger condition, the same inequality is valid for the IUC polytope associated with G and possesses the same facial property on the polytope associated with the corresponding induced subgraph.

Theorem 6. (Anti-hole Inequality) *Let $A \subseteq V$, $|A| \geq 6$, be an anti-hole in $G = (V, E)$. Then, the inequality*

$$\sum_{i \in A} x_i \leq \left\lfloor \frac{|A|}{2} \right\rfloor \quad (10)$$

is valid for $\mathcal{P}_{\text{IUC}}(G)$. Furthermore, it induces a facet of $\mathcal{P}_{\text{IUC}}(G[A])$ if and only if $|A|$ is odd.

PROOF. Consider a labeling of the vertices of an anti-hole A , $|A| \geq 6$, such that $\{i, i+1\}, \forall i \in \{1, \dots, |A|-1\}$, and $\{|A|, 1\}$ identify the pairs of non-adjacent vertices in $G[A]$. For a fixed sequence of labels, note that $\{1, 2, i, i+1\}, \forall i \in \{4, \dots, |A|-2\}$, is a hole of cardinality 4, thus the inequality $x_1 + x_2 + x_i + x_{i+1} \leq 2, \forall i \in \{4, \dots, |A|-2\}$, is valid for $\mathcal{P}_{\text{IUC}}(G)$. Consider the summation of those $|A|-5$ valid inequalities. Observe that each vertex $j \in \{1, 2\}$ appears $|A|-5$ times, $j \in \{4, |A|-1\}$ once, and $j \in \{5, \dots, |A|-2\}$ twice, in the left-hand side of the resultant inequality. These partitions are illustrated by black, hatched, and gray vertices in Figure 2, respectively. The white vertices in Figure 2 are those that are not present in this inequality. Since the starting vertex of the labeling sequence was arbitrary, $|A|$ valid inequalities of this type may be written for $\mathcal{P}_{\text{IUC}}(G)$. Considering all those inequalities, every vertex will appear exactly twice in a black position, twice in a white, twice in a hatched, and $|A|-6$ times in a gray position. As a result, the summation of all 4-hole inequalities

associated with $G[A]$ results in

$$\left(2(|A| - 5) + 2 + 2(|A| - 6)\right) \sum_{i \in A} x_i \leq 2|A|(|A| - 5).$$

Therefore, $\sum_{i \in A} x_i \leq \frac{1}{2}|A|$ is valid for $\mathcal{P}_{IUC}(G)$, which further implies validity of (10) as the Gomory-Chvátal cut corresponding to this inequality. Clearly, when $|A|$ is even, (10) is a linear combination of the 4-hole inequalities associated with $G[A]$, hence not facet-defining for the corresponding polytope. Thus, it remains to show that it induces a facet of $\mathcal{P}_{IUC}(G[A])$ if $|A|$ is odd.

Given an odd anti-hole A , consider $\mathcal{F} = \{x \in \mathcal{P}_{IUC}(G[A]) \mid \sum_{v \in A} x_v = \frac{1}{2}(|A| - 1)\}$. Associated with a fixed labeling of A as described above, let x_C be the incidence vector of the set of even vertices, i.e., $C = \{2, 4, \dots, |A| - 1\}$. Observe that C is a clique in $G[A]$ and $x_C \in \mathcal{F}$, which implies $\mathcal{F}$ is a proper face of $\mathcal{P}_{IUC}(G[A])$. Suppose that $\mathcal{F}$ is contained in a supporting hyperplane $\mathcal{H} : \sum_{i \in A} a_i x_i = b$ of $\mathcal{P}_{IUC}(G[A])$. Let $C' = (C \cup \{1\}) \setminus \{2\}$ and note that C' is also a clique, hence $x_{C'} \in \mathcal{F}$. Then, $x_C, x_{C'} \in \mathcal{H}$ implies that $a_1 = a_2$. Since the start vertex of the labeling was arbitrary, this result holds for all pairs of vertices with consecutive labels. That is, $a_i = a, \forall i \in A$, and $b = \frac{1}{2}a(|A| - 1)$. The proof is complete noting that the interior of $\mathcal{P}_{IUC}(G[A])$ is nonempty, thus $a \neq 0$.

Theorem 6 is closely related to a basic property of IUCs. Ertem et al. [19] showed that, in an arbitrary graph, a (maximal) clique is a maximal IUC if and only if it is a dominating set. In an anti-cycle graph $G = (A, E)$, each maximal clique is maximum, and in case $|A| \geq 6$, it is also a maximum IUC by Theorem 6. It is easy to verify that the anti-cycle graphs on 4 and 5 vertices are the only ones in which a maximal clique is not a dominating set.

In the subsequent sections, we will show that the IUC polytope associated with an anti-cycle graph has many other facets than those induced by the 4-hole inequalities and (10), due to the *fan* substructures that appear in this graph. We postpone our discussion in this regard to Section 3.3.

Finally, it should be mentioned that the odd-hole and odd-anti-hole inequalities for the vertex packing polytope have been generalized by Trotter [50] via introducing a general class of facet-producing subgraphs, called *webs*, that subsumes cycle and anti-cycle graphs as special cases. Given the connection between the vertex packing and IUC polytopes, webs may lead to further facet-inducing inequalities for the IUC polytope as well. We leave this topic for future research and will not pursue it in this paper.

Lifting hole and anti-hole inequalities

Let $G[H]$, $|H| \geq 5$, be an induced chordless cycle in $G = (V, E)$, where $|H| = 3q + r$ and $r \neq 0$. Then by Theorem 3, inequality (7) induces a facet of $\mathcal{P}_{IUC}(G[H])$, but it is not necessarily facet-defining for $\mathcal{P}_{IUC}(G)$. Consider lifting a variable x_v , $v \in V \setminus H$, into (7) to generate a facet-defining inequality for $\mathcal{P}_{IUC}(G[H \cup \{v\}])$ of the form $a_v x_v + \sum_{i \in H} x_i \leq 2q + \lfloor \frac{2r}{3} \rfloor$. The value of a_v in such

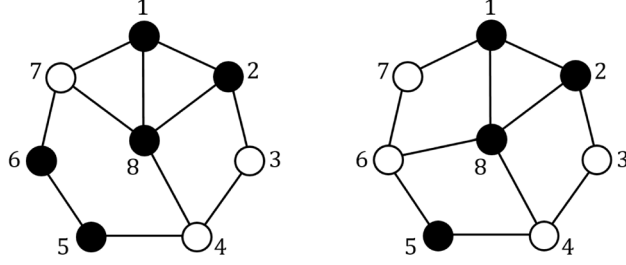


Figure 3: Example graphs for lifting the hole inequality.

an inequality depends on cardinality of $H \cap N(v)$ as well as distances (lengths of the shortest paths) among the vertices of $N(v)$ in $G[H]$. As a case in point, consider the graphs of Figure 3, and lifting x_8 into the inequality $\sum_{i=1}^7 x_i \leq 4$ corresponding to $H = \{1, \dots, 7\}$. In both graphs, $|H \cap N(8)| = 4$ and the difference is due to the distance between the vertices 4 and 7 (equivalently, 1 and 7) in the left-hand-side graph, and the vertices 4 and 6 (equivalently, 1 and 6) in the right-hand-side graph. The sets of black vertices illustrate the maximum size IUCs containing vertex 8 in these graphs. Clearly, the coefficient of x_8 vanishes in lifting $\sum_{i=1}^7 x_i \leq 4$ for the left-hand-side graph, indicating that this inequality is facet-defining for $\mathcal{P}_{IUC}(G[H \cup \{8\}])$. On the other hand, for the IUC polytope associated with the right-hand-side graph, this inequality is dominated by the lifted (facet-defining) inequality $x_8 + \sum_{i=1}^7 x_i \leq 4$.

Obviously, the structure of $N(v)$ in $G[H]$ becomes irrelevant at extreme values of $|H \cap N(v)|$. This result can be slightly strengthened by showing that the lengths of the paths among the neighbors of v in $G[H]$ are redundant when $|H \cap N(v)|$ is restricted to 3 and 2 for $r = 1$ and $r = 2$, respectively. This leads to the following theorem concerning the lifting procedure of the hole inequality to higher-dimensional spaces.

Theorem 7. *Consider the hole inequality (7) where $r \neq 0$. In every sequential lifting of (7), the coefficient of a variable x_v , $v \in V \setminus H$, vanishes under the following conditions: (i) $|H \cap N(v)| \leq 3$ if $r = 1$, (ii) $|H \cap N(v)| \leq 2$ if $r = 2$.*

PROOF. Note that, in an arbitrary sequential lifting of the hole inequality (7), even if some variables not-satisfying the theorem conditions have already been lifted into (7) before x_v , existence of $2q + \lfloor \frac{2r}{3} \rfloor$ vertices in H forming an IUC with v is sufficient for the coefficient of x_v to vanish. Thus, we just need to establish the existence of these vertices under the theorem conditions. Such structures for all possible distributions of neighbors of v in $G[H]$ are provided in Appendix A.

In Section 3.3, we will discuss implication of this theorem about the facial structure of the IUC polytope associated with a *defective wheel* graph.

We conclude this section by stating a similar result concerning the anti-hole inequality. By construction of a maximum IUC in an anti-cycle graph, presented

it the proof of Theorem 6, it is easy to verify that the coefficient of a variable $x_v, v \in V \setminus A$, vanishes in lifting the odd-anti-hole inequality (10) to higher-dimensional spaces if $|A \cap N(v)| \in \{0, 1, |A| - 1, |A|\}$. The following theorem mainly asserts that distribution of the neighbors of v in $G[A]$ is not relevant in lifting x_v into (10) when $|A \cap N(v)| \in \{2, |A| - 2\}$, and the corresponding coefficient vanishes under this condition as well.

Theorem 8. *Consider the odd-anti-hole inequality (10). In every sequential lifting of (10), the coefficient of a variable $x_v, v \in V \setminus A$, vanishes if $|A \cap N(v)| \leq 2$ or $|A \cap N(v)| \geq |A| - 2$.*

PROOF. We show that if $|A \cap N(v)| \leq 2$, then $G[A]$ always contains a clique C of cardinality $|C| = \frac{1}{2}(|A| - 1)$ such that $C \cap N(v) = \emptyset$, and if $|A \cap N(v)| \geq |A| - 2$, then $G[A \cup \{v\}]$ has a clique of size $\frac{1}{2}(|A| - 1) + 1$. The cases $|A \cap N(v)| \leq 1$ and $|A \cap N(v)| \geq |A| - 1$ are trivial. Let $|A \cap N(v)| = 2$, and suppose u and w are the neighbors of v in A . Consider a clockwise labeling of A starting from u as described in the proof of Theorem 6. Observe that, if the label of w is odd, then the set of vertices with even labels form a clique of size $\frac{1}{2}(|A| - 1)$, neither of which is connected to v . On the other hand, if the label of w is even, it will become odd in a counterclockwise labeling of A starting from u , as A itself is of odd cardinality. The clique of interest is then given by the even-labeled vertices in the counterclockwise labeling. The proof for the case $|A \cap N(v)| = |A| - 2$ is identical except that u and w are defined to be the vertices in A not adjacent to v . The proof is complete.

3.2. Star and complete bipartite graph

A *star* graph on n vertices is a tree composed of a central vertex, called *hub*, connected to $n - 1$ leaves. We use the notation $S = \{h\} \cup I$ to refer to the vertex set of a star graph. Here, h is the hub vertex and I denotes the set of leaves, remarking that they form an independent set in the graph.

Theorem 9. (Star Inequality) *Let $S = \{h\} \cup I$ be a subset of vertices inducing a star graph in $G = (V, E)$. Then, the inequality*

$$\sum_{i \in I} x_i + (|I| - 1)x_h \leq |I| \quad (11)$$

is valid for $\mathcal{P}_{IUC}(G)$. Furthermore, (i) it induces a facet of $\mathcal{P}_{IUC}(G[S])$, and (ii) it is facet-defining for $\mathcal{P}_{IUC}(G)$ if and only if there does not exist a vertex $v \in V \setminus N[h]$ such that $S' = \{v\} \cup I$ induces another star graph in G .

PROOF. Validity of (11) is evident. Let $\hat{x}$ be the incidence vector of an arbitrary IUC in G . If $\hat{x}_h = 0$, then (11) is trivially valid. Otherwise, at most one vertex from I may belong to this IUC, since $\{h, i, j\} \in \Lambda(G), \forall i, j \in I$.

(i) To prove that (11) is facet-defining for $\mathcal{P}_{IUC}(G[S])$, it is sufficient to point out that $x_I = \sum_{i \in I} e_i$ and $x_{\{h, i\}} = e_h + e_i, \forall i \in I$, are $|I| + 1$ affinely independent points belonging to $\mathcal{F} = \{x \in \mathcal{P}_{IUC}(G[S]) \mid \sum_{i \in I} x_i + (|I| - 1)x_h = |I|\}$.

(ii) To see the necessity of this condition, note that if there exist two star subgraphs in G induced by $S = \{h\} \cup I$ and $S' = \{v\} \cup I$, for some $v \in V \setminus N[h]$, then the star inequality corresponding to S is dominated by $\sum_{i \in I} x_i + (|I| - 1)x_h + x_v \leq |I|$, which is obtained from lifting x_v into (11).

To prove sufficiency, assume that there does not exist a vertex $v \in V \setminus N[h]$ such that $S' = \{v\} \cup I$ induces a star graph in G . Let $\sum_{v \in V \setminus S} a_v x_v + \sum_{i \in I} x_i + (|I| - 1)x_h \leq |I|$ be the inequality obtained from an arbitrary sequential lifting of (11). We need to show that $a_v = 0, \forall v \in V \setminus S$. In this regard, consider the following cases:

- $v \in N(h)$
If v is also adjacent to a vertex $i \in I$, then $\{h, i, v\}$ is a clique. Otherwise, $\{v\} \cup I$ is an independent set in G .
- $v \notin N(h)$
Since $\{v\} \cup I$ does not induce a star subgraph, there exists a vertex $i \in I$ not adjacent to v . Thus, $G[\{h, i, v\}]$ is the union of an isolated vertex v and an edge $\{h, i\}$.

Therefore, there always exists an IUC in $G[S \cup \{v\}]$ containing v whose incidence vector satisfies $\sum_{i \in I} x_i + (|I| - 1)x_h = |I|$. This result implies that $a_v = 0, \forall v \in V \setminus S$, hence (11) is facet-defining for $\mathcal{P}_{IUC}(G)$ under the theorem's condition.

It is interesting to note that no further condition, such as maximality of I in the neighborhood of h , is required for (11) to be facet-defining for $\mathcal{P}_{IUC}(G)$. In fact, in absence of a vertex $v \in V \setminus N[h]$ described in the theorem statement, every $I' \subseteq I, |I'| \geq 2$, corresponds to a distinct facet of $\mathcal{P}_{IUC}(G)$ defined by (11). In particular, every minimal (inclusion-wise) of such subsets corresponds to a facet-defining OT inequality. Recall that an OT inequality corresponding to $\{h, i, j\}$, for some $\{i, j\} \subseteq I$, is facet-defining for $\mathcal{P}_{IUC}(G)$ if and only if $\{h, i, j, v\}, \forall v \in V \setminus \{h, i, j\}$, is not a 4-hole, or equivalently, $\{v, i, j\}$ does not induce a star subgraph.

The proof of Theorem 9 also declares that if two star subgraphs with the same set of leaves exist in G , then the family of star inequalities (11) corresponding to each one of them can be lifted to a higher-dimensional space to generate facet-inducing inequalities for the corresponding *double-star* (complete bipartite $K_{|I|, 2}$) subgraph. The following theorem shows that these double-star inequalities are actually facet-inducing for $\mathcal{P}_{IUC}(G)$, which also generalizes the facial property of the 4-hole inequality.

Theorem 10. (Double-star Inequality) *Let $S_h = \{h\} \cup I$ and $S_u = \{u\} \cup I$ be two sets of vertices inducing star subgraphs in G , where h and $u \in V \setminus N[h]$ denote two (non-adjacent) hub vertices, and I is the joint set of leaves. Then, the inequalities*

$$\sum_{i \in I} x_i + (|I| - 1)x_h + x_u \leq |I| \quad \text{and} \quad \sum_{i \in I} x_i + x_h + (|I| - 1)x_u \leq |I| \quad (12)$$

induce facets of $\mathcal{P}_{IUC}(G)$.

PROOF. We present the proof for $\sum_{i \in I} x_i + (|I| - 1)x_h + x_u \leq |I|$. The other follows from the symmetry of this structure. To this end, we just need to show that, for every $v \in V \setminus (\{h, u\} \cup I)$, there always exists an IUC containing v in the subgraph $G[\{h, u, v\} \cup I]$ whose incidence vector satisfies $\sum_{i \in I} x_i + (|I| - 1)x_h + x_u = |I|$. This result will be sufficient to indicate that the coefficient of a variable $x_v, \forall v \in V \setminus (\{h, u\} \cup I)$ vanishes in every sequential lifting of $\sum_{i \in I} x_i + (|I| - 1)x_h + x_u \leq |I|$, proving that this inequality is facet-defining for $\mathcal{P}_{IUC}(G)$. The existence of such IUCs when $v \in N(h)$ is established by the proof of Theorem 9. On the other hand, if $v \notin N(h)$, then $\{h, u, v\}$ is an IUC, that is either an independent set or the union of an isolated vertex with two adjacent vertices. This completes the proof.

It is easy to see that if $G = (S, E)$, $S = \{h\} \cup I$, is a star graph, then a subset of vertices in G is an IUC if and only if it is a *co-1-defective clique*. An s -defective clique is a subset of vertices C such that the corresponding induced subgraph contains at least $\binom{|C|}{2} - s$ edges [43]. Correspondingly, a co- s -defective clique is a subset of vertices whose induced subgraph has at most s edges. Sherali and Smith [49] have shown that the family of inequalities defined by (11) for all $I' \subseteq I, |I'| \geq 2$, along with the variable bounds, is sufficient to describe the co-1-defective clique polytope associated with a star graph. Given the equivalence of these two structures in a star graph, this result also holds for the IUC polytope associated with G , which is presented through the following theorem.

Theorem 11. *The family of inequalities (11) together with the variable bounds provide a complete description for the IUC polytope associated with a star graph $G = (S, E)$.*

PROOF. See Proposition 3 in [49] and its proof. ¹

We further show that the family of inequalities given by (12) plays the same role in description of the IUC polytope associated with a complete bipartite graph $G = K_{|I|, |J|}$. Our proof technique is different from the one of Sherali and Smith [49] as it does not rely on higher-dimensional representation of the original polytope. The class of complete bipartite graphs is of a special interest because it has been shown by Pyatkin et al. [45] that a complete bipartite graph $K_{\lfloor \frac{n}{2} \rfloor, \lceil \frac{n}{2} \rceil}$ on n vertices accommodates the maximum possible number of open triangles among all the graphs with the same number of vertices.

Theorem 12. *Let $G = (V, E)$ be a complete bipartite graph, where the bipartition of vertices is given by $V = I \cup J$. Then, a complete description of the IUC polytope associated with G is given by the family of inequalities (12) for all double-star subgraphs of G together with the variable bounds.*

¹In [49], the authors use the term *generalized vertex packing* to refer to co- s -defective clique. We have used the latter term to avoid confusion with the concept of *generalized independent set*, which has been the subject of some recent studies [16, 29].

PROOF. We assume $|I|, |J| \geq 2$ because otherwise G is a star graph. Let $\mathcal{P}$ be the polytope defined by the family of inequalities (12) for all double-star subgraphs of G together with the variable bounds:

$$\mathcal{P} = \{x \in \mathbb{R}^n \mid 0 \leq x_v \leq 1, \forall v \in V; \quad (13a)$$

$$\forall \{j', j''\} \subseteq J, \forall I' \subseteq I, |I'| \geq 2 :$$

$$\sum_{i \in I'} x_i + (|I'| - 1)x_{j'} + x_{j''} \leq |I'|, \quad (13b)$$

$$\sum_{i \in I'} x_i + x_{j'} + (|I'| - 1)x_{j''} \leq |I'|; \quad (13c)$$

$$\forall \{i', i''\} \subseteq I, \forall J' \subseteq J, |J'| \geq 2 :$$

$$\sum_{j \in J'} x_j + (|J'| - 1)x_{i'} + x_{i''} \leq |J'|, \quad (13d)$$

$$\sum_{j \in J'} x_j + x_{i'} + (|J'| - 1)x_{i''} \leq |J'|. \quad (13e)$$

We aim to show that $\mathcal{P}_{IUC}(G) = \mathcal{P}$. Validity of the set of inequalities (13) for the IUC polytope associated with G indicates that $\mathcal{P}_{IUC}(G) \subseteq \mathcal{P}$. In order to prove that the reverse also holds, we show that every extreme point of $\mathcal{P}$ is integral, hence it is the incidence vector of an IUC in G , which implies $\mathcal{P} \subseteq \mathcal{P}_{IUC}(G)$. The proof is by contradiction. Let $\hat{x}$ be a fractional extreme point of $\mathcal{P}$, and define $V_0 = \{v \in V \mid \hat{x}_v = 0\}$ and $V_1 = \{v \in V \mid \hat{x}_v = 1\}$. First, observe that every inequality in (13) involving x_v , for some $v \in V_0$, is redundant in description of the face $\mathcal{P}_{(x_{V_0}=0)} = \{x \in \mathcal{P} \mid x_v = 0, \forall v \in V_0\}$. To see this, consider a fixed vertex $\tilde{j} \in J \cap V_0$. If $|J| = 2$, i.e., $J = \{\tilde{j}, j'\}$, then (13) reduces to

$$\mathcal{P} = \{x \in \mathbb{R}^n \mid 0 \leq x_{j'} \leq 1; \quad 0 \leq x_i \leq 1, \forall i \in I;$$

$$\sum_{i \in I'} x_i + (|I'| - 1)x_{j'} \leq |I'|, \forall I' \subseteq I, |I'| \geq 2\},$$

which is precisely the description of the IUC polytope associated with the star subgraph $G[\{\tilde{j}\} \cup I]$. If $|J| \geq 3$, then each inequality of type (13b)-(13c) corresponding to $\{\tilde{j}, j'\}$ is dominated by an inequality of the same type corresponding to $\{j', j''\}, \forall j'' \in J \setminus \{\tilde{j}, j'\}$. Besides, each inequality of type (13d)-(13e) involving $x_{\tilde{j}}$ can be written as a linear combination of $x_{i'} \leq 1$ (or $x_{i''} \leq 1$) and another inequality of the same type corresponding to $J' \setminus \{\tilde{j}\}$. Therefore,

$$\mathcal{P}_{(x_{V_0}=0)} = \{x \in \mathbb{R}^n \mid x_v = 0, \forall v \in V_0; \quad (14a)$$

$$0 \leq x_v \leq 1, \forall v \in V \setminus V_0; \quad (14b)$$

$$\forall \{j', j''\} \subseteq J \setminus V_0, \forall I' \subseteq I \setminus V_0, |I'| \geq 2 :$$

$$\sum_{i \in I'} x_i + (|I'| - 1)x_{j'} + x_{j''} \leq |I'|, \quad (14c)$$

$$\sum_{i \in I'} x_i + x_{j'} + (|I'| - 1)x_{j''} \leq |I'|; \quad (14d)$$

$$\forall \{i', i''\} \subseteq I \setminus V_0, \forall J' \subseteq J \setminus V_0, |J'| \geq 2 : \\ \sum_{j \in J'} x_j + (|J'| - 1)x_{i'} + x_{i''} \leq |J'|, \quad (14e)$$

$$\sum_{j \in J'} x_j + x_{i'} + (|J'| - 1)x_{i''} \leq |J'|. \quad (14f)$$

Now, consider the following three possibilities for the cardinality of V_1 : $|V_1| = 0$, $|V_1| = 1$, and $|V_1| \geq 2$.

- CASE 1: $|V_1| = 0$ ($\hat{x}_v$ is fractional for every $v \in V \setminus V_0$).
Then, there exists a sufficiently small $\epsilon > 0$ for which $\hat{x}^+$ and $\hat{x}^-$ with

$$\begin{aligned} \hat{x}_i^+ &= \hat{x}_i + \epsilon, \forall i \in I \setminus V_0; \hat{x}_j^+ = \hat{x}_j - \epsilon, \forall j \in J \setminus V_0; \hat{x}_v^+ = \hat{x}_v, \forall v \in V_0 \\ \hat{x}_i^- &= \hat{x}_i - \epsilon, \forall i \in I \setminus V_0; \hat{x}_j^- = \hat{x}_j + \epsilon, \forall j \in J \setminus V_0; \hat{x}_v^- = \hat{x}_v, \forall v \in V_0 \end{aligned}$$

belong to $\mathcal{P}_{(x_{V_0}=0)} \subseteq \mathcal{P}$. Since $\hat{x} = \frac{1}{2}(\hat{x}^+ + \hat{x}^-)$, this contradicts extremity of $\hat{x}$.

- CASE 2: $|V_1| = 1$
Without loss of generality, assume $V_1 = \{\tilde{j}\}$, for some $\tilde{j} \in J$. Note that, in description of the face $\mathcal{P}_{(x_{V_0}=0, x_{\tilde{j}}=1)}$, every inequality in (14d) corresponding to $\{\tilde{j}, j'\}, j' \in J \setminus (V_0 \cup \{\tilde{j}\})$, is dominated by the corresponding inequality of type (14c), as $x_{j'} \leq 1$. Furthermore, each inequality of type (14c) involving $x_{\tilde{j}}$ and corresponding to a proper subset of I is in turn dominated by the one corresponding to I itself, i.e.,

$$\sum_{i \in I} x_i + x_{j'} \leq 1, \forall j' \in J \setminus (V_0 \cup \{\tilde{j}\}). \quad (15)$$

Now, it is clear that every inequality of type (14c)-(14d) corresponding to $\{j', j''\} \subseteq J \setminus (V_0 \cup \{\tilde{j}\})$ and I can be written as a linear combination of an inequality in (15), $x_{j'} \leq 1$, and $x_{j''} \leq 1$, and the ones corresponding to proper subsets of I are dominated by such linear combinations. Hence, (14c)-(14d) reduces to (15) in description of $\mathcal{P}_{(x_{V_0}=0, x_{\tilde{j}}=1)} = \{x \in \mathcal{P} | x_{\tilde{j}} = 1; x_v = 0, \forall v \in V_0\}$. Besides, in the description of this face, every inequality of type (14e)-(14f) corresponding to a set $J' = \tilde{J}$, $\tilde{j} \notin \tilde{J}$, is dominated by the inequality corresponding to $J' = \tilde{J} \cup \{\tilde{j}\}$, thus (14e)-(14f) can be replaced by

$$\forall \{i', i''\} \subseteq I \setminus V_0, \forall J' \subseteq J \setminus (V_0 \cup \{\tilde{j}\}), |J'| \geq 1 : \\ \sum_{j \in J'} x_j + |J'|x_{i'} + x_{i''} \leq |J'|, \quad (16a)$$

$$\sum_{j \in J'} x_j + x_{i'} + |J'|x_{i''} \leq |J'|. \quad (16b)$$

Furthermore, all inequalities in (16) are dominated by linear combinations of the inequalities corresponding to $|J'| = 1$, i.e.,

$$x_{j'} + x_{i'} + x_{i''} \leq 1, \quad \forall j' \in J \setminus (V_0 \cup \{\tilde{j}\}), \quad \forall \{i', i''\} \subseteq I \setminus V_0, \quad (17)$$

and (17) is, in turn, dominated by (15). This implies that

$$\begin{aligned} \mathcal{P}_{(x_{V_0}=0, x_{\tilde{j}}=1)} &= \{x \in \mathbb{R}^n \mid x_{\tilde{j}} = 1; \ x_v = 0, \ \forall v \in V_0; \\ &\quad 0 \leq x_v \leq 1, \ \forall v \in V \setminus (V_0 \cup \{\tilde{j}\}); \\ &\quad \sum_{i \in I} x_i + x_{j'} \leq 1, \ \forall j' \in J \setminus (V_0 \cup \{\tilde{j}\})\}. \end{aligned}$$

Let i' and i'' be two fixed vertices in I , and recall that $0 < \hat{x}_{i'}, \hat{x}_{i''} < 1$. Correspondingly, let $\hat{x}^+$ and $\hat{x}^-$ be the two distinct points defined as follows for a small-enough value of $\epsilon > 0$:

$$\begin{aligned} \hat{x}_{i'}^+ &= \hat{x}_{i'} + \epsilon; \quad \hat{x}_{i''}^+ = \hat{x}_{i''} - \epsilon; \quad \hat{x}_v^+ = \hat{x}_v, \forall v \in V \setminus \{i', i''\}, \\ \hat{x}_{i'}^- &= \hat{x}_{i'} - \epsilon; \quad \hat{x}_{i''}^- = \hat{x}_{i''} + \epsilon; \quad \hat{x}_v^- = \hat{x}_v, \forall v \in V \setminus \{i', i''\}. \end{aligned}$$

Then, $\hat{x} = \frac{1}{2}(\hat{x}^+ + \hat{x}^-)$ while $\hat{x}^+, \hat{x}^- \in \mathcal{P}_{(x_{V_0}=0, x_{\tilde{j}}=1)}$, which contradicts extremity of $\hat{x}$.

– CASE 3: $|V_1| \geq 2$

The set of inequalities (14) indicate that if $I \cap V_1 \neq \emptyset$ and $J \cap V_1 \neq \emptyset$, then each one of I and J must have exactly one vertex in V_1 . Suppose $V_1 = \{i', j'\}$, for some $i' \in I$ and $j' \in J$. Then, (14) immediately implies $\hat{x}_v = 0, \forall v \in V \setminus (V_0 \cup V_1)$, which contradicts $\hat{x}$ being fractional. On the other hand, if $I \cap V_1 = \emptyset$, then (14c)-(14d) imply $\hat{x}_i = 0, \forall i \in I$, and (14e)-(14f) become trivial and redundant. As a result,

$$\begin{aligned} \mathcal{P}_{(x_{V_0}=0, x_{V_1}=1)} &= \{x \in \mathbb{R}^n \mid x_v = 1, \forall v \in V_1; \\ &\quad x_v = 0, \ \forall v \in V_0; \\ &\quad 0 \leq x_v \leq 1, \ \forall v \in V \setminus (V_0 \cup V_1)\}, \end{aligned}$$

all extreme points of which are integral. Patently, the same result holds when $J \cap V_1 = \emptyset$, which contradicts fractionality of $\hat{x}$.

Hence, such a fractional extreme point $\hat{x}$ does not exist, and the proof is complete.

It is obvious that the equivalence of IUC and co-1-defective clique extends to the family of complete bipartite graphs. This observation leads to the following corollary concerning the co-1-defective clique polytope.

Corollary 2. *The family of inequalities (13) gives a complete description of the co-1-defective clique polytope associate with a complete bipartite graph $K_{|I|, |J|}$, where $|I|, |J| \geq 2$.*

3.3. Fan and Wheel

A *fan* graph on n vertices is composed of one vertex adjacent to $n - 1$ vertices inducing a path (chain) subgraph. Following our notation in the previous section, we denote the vertex set of a fan graph by $F = \{h\} \cup P$, where h is the hub vertex and P denotes the vertex set of the corresponding path subgraph. Recall that a path graph itself is not facet-producing for the IUC polytope by Corollary 1. Conventionally, we assume that the edges of $G[P]$ are given by $\{i, i + 1\}, \forall i \in \{1, \dots, |P| - 1\}$.

Theorem 13. (Fan Inequality) *Let $F = \{h\} \cup P$, $|P| \geq 4$, be a subset of vertices inducing a fan graph in $G = (V, E)$. Given $|P| = 3q + r$, where q is a positive integer and $r \in \{0, 1, 2\}$, the inequality*

$$\sum_{i \in P} x_i + \left(2(q - 1) + \left\lfloor \frac{2(r + 1)}{3} \right\rfloor \right) x_h \leq 2q + \left\lfloor \frac{2(r + 1)}{3} \right\rfloor \quad (18)$$

is valid for $\mathcal{P}_{IUC}(G)$, and induces a facet of $\mathcal{P}_{IUC}(G[F])$ if and only if $r \neq 2$.

PROOF. First, note that the right-hand side of (18) is the cardinality of a maximum IUC in $G[P]$, hence the inequality

$$\sum_{i \in P} x_i \leq 2q + \left\lfloor \frac{2(r + 1)}{3} \right\rfloor \quad (19)$$

is valid for $\mathcal{P}_{IUC}(G)$. More formally, (19) is the Gomory-Chvátal cut corresponding to the inequality obtained from the summation of all OT inequalities of $G[P]$ and $2x_1 + x_2 + x_{|P|-1} + 2x_{|P|} \leq 6$ divided by 3. Besides, every maximal IUC containing h in $G[F]$ is a clique of size 3, since every pair of non-adjacent vertices in $G[P]$ with h induces an open triangle in this subgraph. Therefore, $\max\{\sum_{i \in P} x_i \mid x \in \mathcal{P}_{IUC}(G[F]) \text{ and } x_h = 1\} = 2$, and the coefficient of x_h due to lifting into (19) is $2q + \left\lfloor \frac{2(r+1)}{3} \right\rfloor - 2$. This establishes validity of (18).

Next, we show that $r \neq 2$ is sufficient for (18) to induce a facet of the IUC polytope associated with $G[F]$. For $r = 0$, consider the proper face $\mathcal{F} = \{x \in \mathcal{P}_{IUC}(G[F]) \mid \sum_{i \in P} x_i + 2(q - 1)x_h = 2q\}$ of $\mathcal{P}_{IUC}(G[F])$ and an arbitrary supporting hyperplane $\mathcal{H} : a_h x_h + \sum_{i \in P} a_i x_i = b$ that contains $\mathcal{F}$. Let x_{U_1} and x_{U_2} be the incidence vectors of two maximum IUCs in $G[P]$ given as follows:

$$\begin{aligned} U_1 &= \{1, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\}, \\ U_2 &= \{2, 3, 3p - 1, 3p, \forall p \in \{2, \dots, q\}\}. \end{aligned} \quad (20)$$

These are the sets of black vertices in the path graphs obtained from eliminating the vertex in the last partition in the graphs of Figure 1. Clearly, $x_{U_1}, x_{U_2} \in \mathcal{F} \subseteq \mathcal{H}$, which immediately leads to $a_1 = a_2$. In addition, $a_i = a_{i+2}, \forall i \in \{1, \dots, |P| - 2\}$, as the incidence vector of every clique $\{h, i, i + 1\}, \forall i \in \{1, \dots, |P| - 1\}$, in $G[F]$ belongs to $\mathcal{F} \subseteq \mathcal{H}$. This indicates that $a_i = a, \forall i \in P$, $b = 2aq$, and $a_h = 2a(q - 1)$. Finally, $\mathcal{P}_{IUC}(G[F])$ having non-empty interior

implies $a \neq 0$, so $\mathcal{F}$ is a facet of $\mathcal{P}_{IUC}(G[F])$. For the case $r = 1$, using the same argument based on

$$\begin{aligned} U_1 &= \{3q + 1, 1, 2, 3p - 2, 3p, \forall p \in \{2, \dots, q\}\}, \\ U_2 &= \{3q + 1, 1, 3, 3p - 2, 3p, \forall p \in \{2, \dots, q\}\}, \end{aligned}$$

we may show that the inequality $\sum_{i \in P} x_i + (2q - 1)x_h \leq 2q + 1$ is facet-inducing for $\mathcal{P}_{IUC}(G[F])$. This completes the sufficiency part of the proof.

It remains to prove that (18) is not facet-defining for $\mathcal{P}_{IUC}(G[F])$ if $r = 2$. Observe that $G[F]$ has exactly $|P| - 1$ IUCs containing h whose incidence vectors belong to $\mathcal{F} = \{x \in \mathcal{P}_{IUC}(G[F]) \mid \sum_{i \in P} x_i + 2qx_h = 2q + 2\}$, that is $\{h, i, i+1\}, \forall i \in \{1, \dots, |P| - 1\}$. Thus, every other extreme point of $\mathcal{P}_{IUC}(G[F])$ that lies on $\mathcal{F}$ must be the incidence vector of a maximum IUC in $G[P]$. On the other hand, if $r = 2$, $G[P]$ has a unique maximum IUC, whose incidence vector satisfies $x_1 = x_2 = \dots = x_{|P|-1} = x_{|P|} = 1$. Therefore, exactly $|P|$ extreme points of $\mathcal{P}_{IUC}(G[F])$ belong to $\mathcal{F}$, which implies $\dim(\mathcal{F}) \leq |P| - 1 = |F| - 2$. Recall that, every point $x \in \mathcal{F}$ is a convex combination of these extreme points. Hence, $\mathcal{F}$ is not a facet of $\mathcal{P}_{IUC}(G[F])$, and the proof is complete.

Every fan graph contains quadratically many induced subgraphs of the same type, most of which satisfy the conditions of Theorem 13. The following theorem states that the fan inequalities corresponding to those subgraphs are facet-inducing for the IUC polytope associated with the supergraph.

Theorem 14. *Let $G = (F, E)$, $F = \{h\} \cup P$, be a fan graph. Then, every fan inequality (18) corresponding to $F' = \{h\} \cup P'$, $P' \subseteq P$, satisfying the cardinality condition of Theorem 13 induces a facet of $\mathcal{P}_{IUC}(G)$.*

PROOF. Without loss of generality, let $P' = \{j, j+1, \dots, k-1, k\}$ and consider lifting a variable $x_i, i \in P \setminus P'$ into the fan inequality (18) corresponding to $F' = \{h\} \cup P'$. Note that $P' \cap N(i) = \emptyset, \forall i \in P \setminus \{j-1, \dots, k+1\}$, thus every maximum IUC in $G[P']$ together with $i \in P \setminus \{j-1, \dots, k+1\}$ forms an IUC in G . Besides, observe in the proof of Theorem 13 that $G[P']$ always has a maximum IUC in which j is an isolated vertex. As $P' \cap N(j-1) = \{j\}$, the union of this set and the vertex $j-1$ is also an IUC in G . By symmetry, the same result holds for the vertex $k+1$. This implies that the coefficient of every variable $x_i, \forall i \in P \setminus P'$, vanishes in an arbitrary sequential lifting of the fan inequality corresponding to F' , which completes the proof.

Next, we present similar results concerning the IUC polytope associated with a *wheel* graph. A wheel graph induced by $W = \{h\} \cup H$ consists of a hub vertex h connected to all vertices of a hole H . As before, we assume that adjacent vertices in $G[H]$ have consecutive labels, and $|H| + 1 \equiv 1$. Novelty of the following result mainly concerns the case that $|H|$ is a multiple of 3.

Theorem 15. (Wheel Inequality) *Let $W = \{h\} \cup H$, $|H| \geq 4$, be a subset of vertices inducing a wheel graph in $G = (V, E)$. Given $|H| = 3q + r$, where q*

is a positive integer and $r \in \{0, 1, 2\}$, the inequality

$$\sum_{i \in H} x_i + \left(2(q-1) + \left\lfloor \frac{2r}{3} \right\rfloor \right) x_h \leq 2q + \left\lfloor \frac{2r}{3} \right\rfloor \quad (21)$$

is valid for $\mathcal{P}_{IUC}(G)$, and induces a facet of $\mathcal{P}_{IUC}(G[W])$ if and only if $r \neq 0$ or q is odd.

PROOF. Validity of (21) as well as its facial property when $r \neq 0$ are evident, as (21) is obtained from lifting x_h into the hole inequality (7). Similar to a fan graph, $\max\{\sum_{i \in H} x_i \mid x \in \mathcal{P}_{IUC}(G[W]) \text{ and } x_h = 1\} = 2$, hence the coefficient of x_h in such a lifted hole inequality is $2q + \left\lfloor \frac{2r}{3} \right\rfloor - 2$.

Let $r = 0$, and consider $\mathcal{F} = \{x \in \mathcal{P}_{IUC}(G[W]) \mid \sum_{i \in H} x_i + 2(q-1)x_h = 2q\}$ as well as a supporting hyperplane $\mathcal{H} : a_h x_h + \sum_{i \in H} a_i x_i = b$ of $\mathcal{P}_{IUC}(G[W])$ containing it. The incidence vector of every clique $\{h, i, i+1\}, \forall i \in \{1, \dots, |H|\}$, in $G[W]$ belongs to $\mathcal{F} \subseteq \mathcal{H}$, thus $a_i = a_{i+2}, \forall i \in \{1, \dots, |H|\}$. If q is odd—equivalently, H is of odd cardinality—this result further implies that $a_{|H|} = a_2$, which immediately leads to $a_i = a, \forall i \in \{1, \dots, |H|\}$, $b = 2aq$, and $a_h = 2a(q-1)$. Since the interior of $\mathcal{P}_{IUC}(G[W])$ is non-empty, $a \neq 0$ and the sufficiency part of the proof is complete.

To prove the necessity, we examine the case where $r = 0$ and q is even. Similar to the proof of Theorem 13, our argument is based on the number of extreme points of $\mathcal{P}_{IUC}(G[W])$ that lie on $\mathcal{F}$, and the dimension of the affine subspace containing them. Patently, $G[W]$ has exactly $|H|$ IUCs containing h whose incidence vectors belong to $\mathcal{F}$, i.e., $\{h, i, i+1\}, \forall i \in \{1, \dots, |H|\}$. Recall from Theorem 3 that the right-hand-side of (21) is the IUC number of $G[H]$, hence every other extreme point of $\mathcal{P}_{IUC}(G[W])$ lying on $\mathcal{F}$ must be the incidence vector of a maximum IUC in $G[H]$. It is easy to verify that, if $|H|$ is a multiple of 3, $G[H]$ always has three distinct maximum IUCs (of cardinality $2q$) as follows:

$$\begin{aligned} U_1 &= \{3p-2, 3p-1, \forall p \in \{1, \dots, q\}\}, \\ U_2 &= \{3p-1, 3p, \forall p \in \{1, \dots, q\}\}, \\ U_3 &= \{3p-2, 3p, \forall p \in \{1, \dots, q\}\}. \end{aligned}$$

Hence, exactly $|H| + 3$ extreme points of $\mathcal{P}_{IUC}(G[W])$ belong to $\mathcal{F}$.

It is known that the adjacency matrix of a cycle graph with even number of vertices is rank deficient, which immediately implies that the affine subspace of $\mathbb{R}^{|H|}$ spanned by the points $x_{\{i, i+1\}} = e_i + e_{i+1}, \forall i \in \{1, \dots, |H|\}$, is at most $|H| - 2$ dimensional. Besides, the incidence vectors of U_1, U_2 , and U_3 in $\mathbb{R}^{|H|}$ can be easily written as linear combinations of these points, hence the dimension of the subspace of $\mathbb{R}^{|H|}$ spanned by $\{x_{U_1}, x_{U_2}, x_{U_3}, x_{\{i, i+1\}}, \forall i \in \{1, \dots, |H|\}\}$, is no more than $|H| - 2$. Since all extreme points of $\mathcal{P}_{IUC}(G[W])$ lying on $\mathcal{F}$ are generated from a point in this set by adding the coordinate corresponding to h , the subspace of $\mathbb{R}^{|W|}$ spanned by them is at most $|H| - 1 = |W| - 2$ dimensional. Finally, because every point of $\mathcal{F}$ is a convex combination of these extreme points, dimension of this face is no more than $|W| - 2$, therefore $\mathcal{F}$ is not a facet. The proof is complete.

The result of Theorem 14 naturally extends to the induced fans of a wheel graph. The only exception is given by the subgraphs obtained from eliminating a single vertex (and incident edges) from the corresponding chordless cycle. Let $F = \{h\} \cup P$, where $P = H \setminus \{v\}$, be the vertex set of such a fan subgraph, while $|P|$ satisfies the facet-defining conditions of Theorem 13, that is $|P| = 3q + 1$ or $|P| = 3q$. Observe that, in this case, $|H|$ also satisfies the facet-defining conditions of Theorem 15, that is $|H| = 3q + 2$ or $|H| = 3q + 1$. Then, it can be easily shown that the wheel inequality (21) corresponding to $W = \{h\} \cup H$ coincides with the lifted fan equality (18) corresponding to $F = \{h\} \cup P$.

Patently, this result also applies to a *defective wheel* graph. A defective wheel is obtained from a (complete) wheel graph by eliminating some edges connecting the hub vertex to the corresponding hole. In addition to its fan subgraphs, a defective wheel always has at least one chordless cycle that contains its hub vertex. Examples of defective wheels are given in Figure 3. Let H_h denote a hole containing the hub vertex h in a defective wheel graph. In reference to Theorem 7 of Section 3.1, observe that every vertex not in H_h has at most two neighbors in this set. The following corollary summarizes the implications of Theorems 14 and 7 regarding the facial structure of the IUC polytope associated with a defective wheel graph.

Corollary 3. *Let $G = (V, E)$ be a defective wheel graph. Then,*

- (i) *inequality (18) corresponding to a fan subgraph satisfying the cardinality condition of Theorem 13 is facet-defining for $\mathcal{P}_{IUC}(G)$. Besides,*
- (ii) *inequality (7) corresponding to a hole containing the hub vertex that satisfies the cardinality condition of Theorem 3 induces a facet of $\mathcal{P}_{IUC}(G)$.*

Finally, note that the IUC polytope associated with a fan graph, likewise a wheel graph, possesses exponentially many facets induced by the star inequality (11) as well.

Lifting fan and wheel inequalities

We consider lifting the fan and wheel inequalities under the conditions that they are facet-defining for the IUC polytope associated with the corresponding induced subgraph. Similar to Section 3.1, our focus is on the distribution of $N(v)$, $v \in V \setminus F$ (resp. $v \in V \setminus W$), within $G[F]$ (resp. $G[W]$) and the cases where the lifting coefficients can be determined without solving a computationally expensive lifting problem.

First, observe that the result of Theorem 7, concerning the hole inequality (7), also applies to the wheel inequality (21) since an IUC of cardinality $2q + \lfloor \frac{2r}{3} \rfloor + 1$ in $G[W \cup \{v\}]$, $v \in V \setminus W$, can always be attained through the vertices of H (together with v) under the theorem conditions. However, the results concerning redundancy of the distribution of neighbors of v does not extend to the fan inequality (18) as well as the wheel inequality (21) when $r = 0$ (and q is odd). Consider, for example, the case where $|P| = 7$ and $P \cap N(v) = \{1, 3\}$, or $|H| = 9$ and $H \cap N(v) = \{1, 5\}$. In addition, it is trivial that $|P \cap N(v)| \leq 1$

(resp. $|H \cap N(v)| \leq 1$) is sufficient for the coefficient of $x_v, v \in V \setminus F$ (resp. $v \in V \setminus W$), to vanish in the latter cases, regardless of the role of the hub vertex h . In general, however, the lifting coefficient of a variable x_v is highly influenced by connectivity of v and h . Particularly, if $\{h, v\} \in E$, existence of two adjacent vertices in $P \cap N(v)$ (resp. $H \cap N(v)$) is sufficient for the corresponding lifting coefficient to vanish. On the other hand, if $\{h, v\} \notin E$, existence of two adjacent vertices in $P \setminus N(v)$ (resp. $H \setminus N(v)$) brings about the same result. The following theorem concerns redundancy of the distribution of $N(v)$ within $G[F]$ given the contribution of the hub vertex h . Note that, the coefficient of x_v in every sequential lifting of the corresponding inequality will not exceed 2 because $\{h, v\}$ is always an IUC in the graph.

Theorem 16. *Consider the fan inequality (18) where $r \neq 2$. In every sequential lifting of this inequality, the coefficient of a variable $x_v, v \in V \setminus F$, vanishes under the following conditions: (i) $h \in N(v)$ and $|P \cap N(v)| \geq \left\lceil \frac{|P|}{2} \right\rceil + 1$, or (ii) $h \notin N(v)$ and $|P \cap N(v)| \leq \left\lfloor \frac{|P|}{2} \right\rfloor - 1$.*

PROOF. It suffices to observe that the cardinality of $P \cap N(v)$ in each case guarantees the existence of two adjacent vertices $i, i+1 \in P$ such that $\{h, v, i, i+1\}$ is an IUC in $G[F \cup \{v\}]$. In the former case, this IUC is a clique, and in the latter, it is the union of a clique and an isolated vertex.

Evidently, a similar result holds for the wheel inequality, which we skip here.

We finish this section with presenting a family of facet-defining inequalities for the IUC polytope associated with an anti-cycle graph due to its fan substructures. We assume the vertices of an anti-cycle graph are labeled as described in the proof of Theorem 6, and $|A| + 1 \equiv 1$.

Theorem 17. *Let $G = (A, E)$ be an anti-cycle graph on $|A| \geq 7$ vertices. Then, the inequality*

$$\sum_{j=i}^{i+3} x_j + x_k + x_{k+1} \leq 3, \quad \forall i \in A, \quad \forall k \in \{i+4, \dots, |A|+i-2\}, \quad (22)$$

induces a facet of $\mathcal{P}_{IUC}(G)$.

PROOF. Observe that $P = \{i, i+1, i+2, i+3\}, \forall i \in A$, induces a path (chain) graph on 4 vertices, and $G[\{k\} \cup P], \forall k \in \{i+5, \dots, |A|+i-2\}$, is a fan in G . In fact, for a fixed vertex i , the vertices $i+4$ and $|A|+i-1$ are the only ones in $A \setminus P$ which are not adjacent to all vertices of P . Therefore, the inequality $\sum_{j=i}^{i+3} x_j + x_k \leq 3, \forall k \in \{i+5, \dots, |A|+i-2\}$, is valid for $\mathcal{P}_{IUC}(G)$ and induces a facet of $\mathcal{P}_{IUC}(G[\{k\} \cup P])$, by Theorem 13. First, consider lifting the variable x_{k+1} into $\sum_{j=i}^{i+3} x_j + x_k \leq 3$, for some $k \in \{i+5, \dots, |A|+i-3\}$. Since $\{k, k+1\} \notin E$ and $P \cap N(k+1) = P$, the coefficient of x_{k+1} in the corresponding lifted inequality will be 1 and the inequality $\sum_{j=i}^{i+3} x_j + x_k + x_{k+1} \leq 3$ is facet defining for $\mathcal{P}_{IUC}(G[\{k, k+1\} \cup P])$. Besides, every other vertex $v \in A \setminus (\{k, k+1\} \cup P)$

has at least one neighbor in $\{k, k+1\}$ and $|P \cap N(v)| \geq 3$, hence the coefficient of x_v vanishes in every sequential lifting of this inequality, by Theorem 16. Therefore, the inequality $\sum_{j=i}^{i+3} x_j + x_k + x_{k+1} \leq 3$ induces a facet of $\mathcal{P}_{IUC}(G)$, for every $k \in \{i+5, \dots, |A|+i-3\}$. It remains to show that, for every $i \in A$, the inequalities $\sum_{j=i}^{i+3} x_j + x_{i+4} + x_{i+5} \leq 3$ and $\sum_{j=i}^{i+3} x_j + x_{|A|+i-2} + x_{|A|+i-1} \leq 3$ are facet-defining for $\mathcal{P}_{IUC}(G)$. We present the proof for the former inequality, i.e.,

$$\sum_{j=i}^{i+5} x_j \leq 3, \quad (23)$$

for an arbitrary vertex i ; the latter then follows by symmetry of the graph structure. It is easy to verify that (23) is obtained from lifting x_{i+4} into the fan inequality $\sum_{j=i}^{i+3} x_j + x_{i+5} \leq 3$, hence facet-defining for the IUC polytope associated with the corresponding induced subgraph. Besides, every vertex $k \in \{i+7, \dots, |A|+i-1\}$ is adjacent to the vertex $i+5$, thus the corresponding coefficient vanishes in lifting x_k into (23), by Theorem 16. Finally, observe that $\{i, i+2, i+4, i+6\}$ is a clique in G , therefore, $\max\{\sum_{j=i}^{i+5} x_j \mid x \in \mathcal{P}_{IUC}(G) \text{ and } x_{i+6} = 1\} = 3$, and the coefficient of x_{i+6} also vanishes in lifting this variable into (23). The proof is complete.

It is important to note the pivotal role of the (complete) fan structures and the corresponding fan inequalities in the foregoing result. Observe that the anti-cycle graph on $|A| = 6$ vertices does not have an induced (complete) fan, and correspondingly, the inequality $\sum_{j=1}^6 x_j \leq 3$ is not facet-inducing for the corresponding IUC polytope, previously shown by Theorem 6.

4. The Separation Problems

In this section, we study the computational complexity of the separation problem for each family of the valid inequalities presented in the previous section for the IUC polytope. Given a family of (linear) inequalities $\mathcal{F}$ and a point x^* , the corresponding separation problem is to decide whether x^* belongs to the polyhedron defined by $\mathcal{F}$, i.e., if x^* satisfies all inequalities in $\mathcal{F}$, and if not, find an inequality which is violated by x^* . Efficient separating subroutines are essential in devising branch-and-cut algorithms for integer (linear) programs; however, our results in this part are negative in that we can show that the separation problems for most of the proposed valid inequalities for the IUC polytope are NP-hard.

Complexity of the separation problem for the family of hole, as well as anti-hole, inequalities for the IUC polytope is yet to be determined, although there is a rich body of literature concerning identification of holes and anti-holes in graphs. In fact, the focus of this line of research has been mostly on holes and anti-holes of odd cardinality due to the Berge's strong perfect graph conjecture [3]. According to this conjecture, a graph G is perfect, i.e., the chromatic number of every induced subgraph of G equals its clique number, if and only if

G does not contain an odd hole or odd anti-hole. Chudnovsky et al. [14] proved this conjecture, and Chudnovsky et al. [13] proposed an $\mathcal{O}(n^9)$ -time algorithm to test if a graph has an odd hole or odd anti-hole; hence deciding if it is perfect. However, it remained an open problem whether or not it is polynomial-time decidable if a graph has an odd hole (equivalently, an odd anti-hole) until very recently that Chudnovsky et al. [15] presented an $\mathcal{O}(n^9)$ -time algorithm for it. It is also known that the problem of deciding if a graph has a hole (equivalently, an anti-hole) of even cardinality is tractable [17], and the most efficient algorithm proposed for this problem has a time complexity of $\mathcal{O}(n^{11})$ [11]. On the other hand, it is NP-complete to decide if a graph has an odd (resp. even) hole containing a given vertex [4, 5, 11]. It is clear—due to feasibility of $x^* = \frac{2}{3}\mathbf{1}$ to the fractional IUC polytope—that the separation problem for the family of hole inequalities (7) for the IUC polytope is at least as hard as determining if a graph has a hole whose cardinality is not a multiple of 3, for which we are not aware of an efficient algorithm. Similarly, the feasibility of $x^* = \frac{1}{2}\mathbf{1}$ to the fractional IUC polytope implies that the separation problem for the family of anti-hole inequalities (10) for the IUC polytope is at least as hard as deciding if a graph has an odd hole. It should be mentioned that the separation problem for the family of odd-hole inequalities for the vertex packing polytope (equivalently, odd-anti-hole inequalities for the clique polytope) is polynomial-time solvable [27]; however, this result does not automatically extend to the family of odd-anti-hole inequalities for the IUC polytope.

In the rest of this section, we show that the separation problems for the families of star, double-star, fan and wheel inequalities for the IUC polytope are all NP-hard.

Theorem 18. *The separation problem for the family of star inequalities (11) for the IUC polytope is NP-hard.*

PROOF. The reduction is from the INDEPENDENT SET problem (k -IS), which asks if a simple undirected graph $G = (V, E)$ has an independent set of cardinality at least k . Without loss of generality, we assume $k \geq 3$. Corresponding to an instance (G, k) of k -IS, consider an instance (G', x^*) of the separation problem for the family of star inequalities (11), where $G' = (V', E')$ is constructed by appending a new vertex h to G adjacent to all vertices of V , i.e.,

$$\begin{aligned} V' &= V \cup \{h\}, \\ E' &= E \cup \{\{h, i\}, \forall i \in V\}, \end{aligned}$$

and x^* is defined as follows:

$$\begin{aligned} x_h^* &= 1, \\ x_i^* &= \frac{1}{k-1}, \quad \forall i \in V. \end{aligned}$$

Note that $k \geq 3$ implies $x^* \in \mathcal{P}_{IUC}^F(G')$. We claim that (G, k) is a positive instance of k -IS if and only if x^* violates a star inequality in G' . Let I^* be

an independent set of cardinality k in G . Then, $S = \{h\} \cup I^*$ induces a star subgraph in G' , and

$$\sum_{i \in I^*} x_i^* + (|I^*| - 1)x_h^* = \frac{k}{k-1} + (k-1) = k + \frac{1}{k-1} > k = |I^*|,$$

which indicates violation of the star inequality associated with $G'[S]$. To prove the reverse, suppose that x^* violates the inequality associated with a star subgraph induced by $S' = \{v\} \cup I'$ in G' , for some $I' \subseteq V'$ and $v \in V' \setminus I'$. Evidently, $I' \subseteq V$; since $x_v^* \leq 1 = x_h^*, \forall v \in V'$, this also indicates violation of the star inequality associated with $G'[\{h\} \cup I']$, i.e.,

$$\sum_{i \in I'} x_i^* + (|I'| - 1)x_h^* = \frac{|I'|}{k-1} + (|I'| - 1) > |I'|,$$

which implies $|I'| > k-1$, or equivalently $|I'| \geq k$. The proof is complete.

Theorem 19. *The separation problem for the family of double-star inequalities (12) for the IUC polytope is NP-hard.*

PROOF. The proof is similar to Theorem 18. Corresponding to an instance (G, k) of k -IS, consider an instance $(G' = (V', E'), x^*)$ of the separation problem for the family of double-star inequalities (12) constructed as follows:

$$\begin{aligned} V' &= V \cup \{h, u\}, \\ E' &= E \cup \{\{h, i\}, \forall i \in V\} \cup \{\{u, i\}, \forall i \in V\}, \\ x_h^* &= 1, \\ x_i^* &= \frac{1}{k}, \forall i \in V \cup \{u\}. \end{aligned}$$

Then, the same argument leads to the fact that (G, k) is a positive instance of k -IS if and only if x^* violates a double-star inequality in G' .

Theorem 20. *The separation problem for the family of fan inequalities (18) for the IUC polytope is NP-hard.*

PROOF. We present a reduction from the INDUCED PATH problem (k -IP), which is to decide if a simple undirected graph $G = (V, E)$ has an induced path subgraph (equivalently, a chordless path) on at least k vertices. Similar to k -IS, this problem is among the classical NP-complete problems listed by Garey and Johnson [25]. The proof idea is similar to that for Theorem 18. Let $k \geq 5$, and corresponding to an instance (G, k) of k -IP, consider an instance $(G' = (V', E'), x^*)$ of the separation problem for the family of fan inequalities (18) constructed as follows:

$$\begin{aligned} V' &= V \cup \{h\}, \\ E' &= E \cup \{\{h, i\}, \forall i \in V\}, \\ x_h^* &= 1, \\ x_i^* &= \frac{2}{k-1}, \forall i \in V. \end{aligned}$$

Let P^* be the vertex set of a chordless path in G on $k = 3q_k + r_k$ vertices. Then, $F = \{h\} \cup P^*$ induces a fan subgraph in G' , and

$$\begin{aligned} \sum_{i \in P^*} x_i^* + \left(2(q_k - 1) + \left\lfloor \frac{2(r_k + 1)}{3} \right\rfloor \right) x_h^* &= \frac{2k}{k-1} + 2q_k - 2 + \left\lfloor \frac{2(r_k + 1)}{3} \right\rfloor \\ &= \frac{2}{k-1} + 2q_k + \left\lfloor \frac{2(r_k + 1)}{3} \right\rfloor \\ &> 2q_k + \left\lfloor \frac{2(r_k + 1)}{3} \right\rfloor, \end{aligned}$$

which indicates violation of the fan inequality associated with $G'[F]$. Now, suppose that x^* violates a fan inequality in G' . Given the magnitude of the components of x^* , this also implies violation of the inequality associated with a fan subgraph $G'[\{h\} \cup P']$, for some $P' \subseteq V$, i.e.,

$$\sum_{i \in P'} x_i^* + \left(2(q' - 1) + \left\lfloor \frac{2(r' + 1)}{3} \right\rfloor \right) x_h^* > 2q' + \left\lfloor \frac{2(r' + 1)}{3} \right\rfloor,$$

where $|P'| = 3q' + r'$. This leads to

$$\sum_{i \in P'} x_i^* = \frac{2|P'|}{k-1} > 2,$$

or equivalently, $|P'| \geq k$, and completes the proof.

Theorem 21. *The separation problem for the family of wheel inequalities (21) for the IUC polytope is NP-hard.*

PROOF. The proof is almost identical to the proof of Theorem 20. The difference is that, instead of k -IP, the reduction is from the INDUCED CYCLE problem, which is also known to be NP-complete [25]. The INDUCED CYCLE problem (k -IC) is to decide if a simple undirected graph $G = (V, E)$ has a chordless cycle on at least k vertices. Then, using the same argument on an instance $(G' = (V', E'), x^*)$ of k -IC constructed as stated in the proof of Theorem 20, one can show that the separation problem for the family of wheel inequalities (21) is NP-hard.

It is worth noting that, from the viewpoint of parameterized complexity, the INDEPENDENT SET, INDUCED PATH, and INDUCED CYCLE problems are all $W[1]$ -complete [12] implying that it is unlikely that there exists a fixed-parameter-tractable (fpt) algorithm for either of them. This result immediately implies that the separation problems for the families of star, double-star, fan and wheel inequalities remain intractable even if the separation problem is defined with respect to a subset of inequalities from the corresponding family with a fixed (small) number of variables.

5. Computational Experiments

In this section, we present the results of our computational experiments. We have conducted two sets of experiments to evaluate the effectiveness of incorporating the proposed valid inequalities in integer (linear) programming solution approaches for the MAXIMUM IUC problem.

In the first set of experiments, we investigated the effectiveness of each class of valid inequalities, as well as their combinations, using a set of graphs whose structures were known *a priori*. We generated a set of 20 undirected graphs, each of which is essentially a collection of 21 *principal* subgraphs that are sparsely connected to each other. Every principal subgraph of each instance has been randomly selected to be a cycle, anti-cycle, star, double-star, fan or wheel, and its number of vertices is given by the floor of a number drawn from a normal distribution with mean 10 and standard deviation 1. This choice of parameters yields graphs with around 200 vertices. The principal subgraphs have been connected to one another by a sparse set of random edges. To do this, an edge has been generated between every pair of vertices from two different principal subgraphs with a probability 0.01 independent of the others. Table 1 displays the characteristics of these instances, including the number of open triangles $|\Lambda|$ and the number of principal subgraphs for each structure type. In this table, “A-cycle” and “D-star” stand for anti-cycle and double-star, respectively. We solved the MAXIMUM IUC problem on each instance using the

Table 1: First set of experiments: characteristics of the test instances.

Name	$ V $	$ E $	$ \Lambda $	Number of principal subgraphs					
				Cycle	A-cycle	Star	D-star	Fan	Wheel
rnd#1	204	550	2,464	3	2	2	3	7	4
rnd#2	213	630	2,857	2	4	6	1	3	5
rnd#3	196	510	2,273	3	2	3	5	6	2
rnd#4	194	470	2,056	6	2	3	3	2	5
rnd#5	211	623	3,078	4	5	5	3	3	1
rnd#6	194	617	2,673	2	8	2	4	2	3
rnd#7	193	488	2,014	2	3	3	4	6	3
rnd#8	198	506	2,226	5	2	4	4	4	2
rnd#9	194	496	2,293	3	2	5	3	3	5
rnd#10	194	472	1,998	2	3	5	6	3	2
rnd#11	203	567	2,644	4	3	1	5	5	3
rnd#12	188	466	2,143	2	1	2	5	8	3
rnd#13	202	514	2,537	4	2	4	6	4	1
rnd#14	212	497	2,070	7	2	5	2	3	2
rnd#15	206	575	2,536	4	3	3	2	5	4
rnd#16	201	550	2,608	2	3	2	7	2	5
rnd#17	204	586	2,751	4	4	3	5	2	3
rnd#18	200	581	2,813	3	5	4	3	2	4
rnd#19	204	562	2,422	5	4	2	1	5	4
rnd#20	203	550	2,268	5	3	2	3	7	1

Table 2: First set of experiments: improvement (in solution time) for each family of the valid inequalities.

Name	opt.	Solution time (CPU sec.)				Impv. per ineq. (%)		
		Base	+HA	+SD	+FW	+HA	+SD	+FW
rnd#1	111	3,094.30	989.39	1,804.45	715.48	13.6	5.2	7.0
rnd#2	118	1,103.08	765.27	791.47	369.09	5.1	3.5	8.3
rnd#3	114	90.77	87.08	92.72	21.45	0.8	−0.2	9.5
rnd#4	110	197.38	180.75	177.79	108.90	1.1	1.1	6.4
rnd#5	118	1,218.80	471.90	510.82	486.17	6.8	5.3	15.0
rnd#6	103	9,820.76	250.19	5,012.75	2,299.58	9.7	4.9	15.3
rnd#7	110	173.19	61.19	120.13	82.52	12.9	2.8	5.8
rnd#8	115	115.95	54.36	75.58	87.51	7.6	2.9	4.1
rnd#9	113	95.45	83.50	51.37	22.74	2.5	4.2	9.5
rnd#10	114	48.18	11.76	43.24	18.51	15.1	0.6	12.3
rnd#11	114	438.31	197.32	471.63	182.44	7.9	−0.7	7.3
rnd#12	110	73.44	20.98	53.08	13.16	23.8	2.3	7.5
rnd#13	123	24.86	14.33	13.22	11.37	7.1	2.9	10.9
rnd#14	124	156.53	89.51	84.94	90.49	4.8	5.1	8.4
rnd#15	116	654.23	320.22	521.43	289.63	7.3	2.9	6.2
rnd#16	109	5,259.73	2,170.50	1,271.24	1,342.31	11.7	4.7	10.6
rnd#17	111	3,870.85	1,096.95	944.77	1,126.95	9.0	5.8	14.2
rnd#18	110	1,085.21	216.91	602.61	376.10	10.0	4.4	10.9
rnd#19	110	10,496.40	1,971.21	3,847.48	708.52	9.0	15.8	10.4
rnd#20	114	540.62	90.82	382.95	359.84	10.4	3.6	4.2

base formulation (5), i.e., considering only the OT inequalities, as well as using this formulation strengthened by the valid inequalities corresponding to its principal subgraphs. For each principal subgraph, we generated only one inequality, except for the double-star subgraphs where both inequalities of (12) were generated. Recall that anti-cycle, star, double-star, fan and wheel graphs may generate several (potentially facet-defining) valid inequalities of the same or different types; yet, we restricted the incorporated inequalities to the whole structures, i.e., the principal subgraphs themselves, so we could have a better measure of the effectiveness of each family. We performed the experiments using ILOG/CPLEX 12.7 solver on a Dell Precision Workstation T7500[®] machine with eight 2.40 GHz Intel Xeon[®] processors and 12 GB RAM. We used the default settings of the solver except for the automatic cut generation, which was off in our first set of experiments.

Table 2 presents the solution time for each instance using the base formulation (5) as well as the strengthened formulations by the individual classes of the proposed valid inequalities. In this table, “opt.” denotes the optimal solution value for each instance. The column “Base” reports the solution time under the base formulation (5). The next three columns, named “+HA”, “+SD”, and “+FW”, present the solution time for each instance when the base formulation is strengthened by the hole (7) and anti-hole (10) inequalities, star (11) and

double-star (12) inequalities, and fan (18) and wheel (21) inequalities associated with its principal subgraphs, respectively. The last three columns of this table show the percentage improvement in the solution time for each instance per appended valid inequality; that is,

$$\frac{\text{Base solution time} - \text{Improved solution time}}{\text{Base solution time} \times \text{Number of incorporated inequalities}} \times 100.$$

The boldface numbers in these columns show the highest improvement for each instance. Note that the highest per-inequality improvement does not necessarily imply the best solution time, due to different number of principal subgraphs of each type. On 10 instances (out of 20), the most per-inequality improvement was attained by the FW family (fan and wheel inequalities). With a similar performance, the HA family (hole and anti-hole inequalities) generated the highest per-inequality improvement on 9 instances. On the other hand, the SD family (star and double-star inequalities) showed an inferior performance compared to the two former families. In fact, on two instances **rnd#3** and **rnd#11**, incorporating these inequalities into the base formulation even resulted in a slight increase in the solution time. This observation may be justified by noting that star and double-star subgraphs generate exponentially many (with respect to the size of the subgraph) known facet-defining valid inequalities for the IUC polytope associated with the supergraph, whereas this number for the other two families is much smaller, and we have incorporated only a couple of these inequalities from each family.

In the next phase of our experiments, we also examined the effect of adding different combinations of these valid inequalities on the solution time of the test instances. Table 3 presents the corresponding results. In this table, “+HA&SD” shows the solution time when the base formulation has been strengthened by the combination of HA and SD families. Similarly, “+HA&FW” and “+SD&FW” show the results for the combination of HA and FW families, and SD and FW families, respectively. The last column of this table, i.e., “+All”, presents the solution time for each instance obtained by incorporating all three HA, SD, and FW families in the base formulation (5). Note that the total number of appended inequalities in this case is about 1% of the number of OT inequalities for each instance. The boldface numbers in this table show the best solution time obtained for each instance. Even by incorporating a rather small number of valid inequalities, the solution times are substantially better than those obtained due to incorporating individual families of the valid inequalities, presented in Table 2. Interestingly, for all instances, the best solution time was attained when the FW family was incorporated, and only on 11 instances (out of 20), the best solution time was due to incorporating all valid inequalities.

The results of our first set of experiments reveal the merit of the proposed valid inequalities. In practice, however, the structure of an input graph is not known *a priori*. Therefore, integer (linear) programming solution methods of the MAXIMUM IUC problem should rely on subgraph-detection routines in order to generate these valid inequalities. In our second set of experiments, we investigated efficacy of the proposed valid inequalities under this setting. We

Table 3: First set of experiments: results of incorporating different combinations of valid inequalities.

Name	Solution time (CPU sec.)				
	Base	+HA&SD	+HA&FW	+SD&FW	+All
rnd#1	3,094.30	477.81	345.94	291.17	167.73
rnd#2	1,103.08	923.40	637.46	237.37	354.18
rnd#3	90.77	29.96	19.81	25.13	16.91
rnd#4	197.38	158.29	103.24	91.26	103.08
rnd#5	1,218.80	326.88	356.16	583.89	242.28
rnd#6	9,820.76	274.15	119.29	1,346.66	157.43
rnd#7	173.19	32.46	36.02	58.75	23.43
rnd#8	115.95	26.17	19.85	84.22	25.78
rnd#9	95.45	68.12	17.80	19.70	15.05
rnd#10	48.18	11.85	15.05	9.29	7.47
rnd#11	438.31	152.38	123.44	158.43	131.15
rnd#12	73.44	28.09	10.14	14.47	14.18
rnd#13	24.86	7.28	1.99	8.93	8.40
rnd#14	156.53	97.76	50.99	87.31	47.58
rnd#15	654.23	287.86	147.35	216.37	103.77
rnd#16	5,259.73	722.94	665.09	414.69	157.23
rnd#17	3,870.85	846.85	635.40	510.96	332.18
rnd#18	1,085.21	138.07	127.95	223.97	102.35
rnd#19	10,496.40	1,391.08	283.62	546.51	354.59
rnd#20	540.62	93.92	61.29	191.92	68.03

generated a set of (Erdős-Rényi) random graphs $G(n, p)$ with $n = 100$ and p varying from 0.05 to 0.95. In these graphs, n is the number of vertices and an edge exists between every pair of vertices with a probability p independent of the others. The characteristics of our second set of test instances, including the number of open triangles $|\Lambda|$ for each instance, are shown in Table 4. In this table, the numerical part of the name of each instance shows the corresponding value of p , which is also the expected edge density of the graph. In light of the results of our first set of experiments, as well as the complexity results of the previous section, we just considered 4-hole, wheel and fan cutting planes in this part of our experiments, and applied them all to the root node of the branch-and-cut tree. It should be mentioned that developing a full-blown branch-and-cut algorithm for the MAXIMUM IUC problem is out of the scope this paper, and the purpose of our experiments was to test effectiveness of employing these inequalities using a straightforward strategy, described below.

For each instance, we identified the entire set of 4-holes, but we employed a subset of them to generate the 4-hole cutting planes. Note that the expected number of 4-holes in random graphs with moderate densities is much higher than the expected number of open triangles. Besides, every 4-hole inequality (7) dominates four OT inequalities, which can be eliminated from the original formulation upon adding the 4-hole cuts. In this regard, our solution strategy was

Table 4: Second set of experiments: cut-generation characteristics.

Name	$ V $	$ E $	$ \Lambda $	# Valid inequalities (cuts)				Cut gener. (CPU sec.)
				OT	4-hole	Wheel	Fan	
RND_0.05	100	255	1,204	956	70	0	0	0.05
RND_0.10	100	503	4,558	1,867	868	9	0	0.06
RND_0.15	100	810	10,823	1,235	3,444	378	211	0.34
RND_0.20	100	1,058	17,337	511	6,126	1,614	965	1.40
RND_0.25	100	1,261	23,406	274	8,378	2,463	1,323	3.57
RND_0.30	100	1,541	32,444	30	10,945	3,338	1,466	9.623
RND_0.35	100	1,812	41,242	7	13,683	3,382	1,616	20.19
RND_0.40	100	1,966	45,987	2	14,988	3,470	1,528	27.58
RND_0.45	100	2,253	54,553	1	17,523	3,506	1,494	46.07
RND_0.50	100	2,470	60,337	0	19,323	3,456	1,544	63.04
RND_0.55	100	2,773	67,027	0	21,103	3,604	1,396	87.25
RND_0.60	100	2,981	70,308	0	21,897	3,687	1,313	101.20
RND_0.65	100	3,223	71,881	0	22,766	3,589	1,384	108.66
RND_0.70	100	3,457	71,135	0	22,923	3,747	1,184	104.21
RND_0.75	100	3,748	67,644	0	21,707	2,328	700	91.00
RND_0.80	100	3,994	61,000	0	19,990	350	62	66.70
RND_0.85	100	4,259	50,326	0	16,822	2	24	38.02
RND_0.90	100	4,460	39,011	2	13,635	0	0	17.86
RND_0.95	100	4,726	19,977	762	7,509	0	0	2.53

to find a small subset of 4-hole inequalities that could cover a large portion (possibly all) of OT inequalities. Clearly, finding a minimum number of 4-holes to dominate all possible open triangles in a graph is a set covering problem, which is NP-hard in general. We employed a well-known greedy heuristic for this problem to generate the 4-hole cuts; at every iteration, we selected a 4-hole with the maximum number of uncovered open triangles, and stopped when this number was zero for all the remaining 4-holes. Obviously, not all open triangles are guaranteed to be part of a 4-hole in a graph, thus not all OT inequalities could be replaced with the generated 4-hole cuts. Table 4 shows the number of generated 4-hole cuts for each instance, as well as the number of OT inequalities that were not covered by them and remained in the problem formulation.

We also generated a set of wheel and fan cuts for each instance. To do this, for every vertex $i \in V$, we considered the subgraph induced by its neighbors, i.e., $G[N(i)]$, and detected a set of chordless cycles and paths using a simple enumeration method. Clearly, a graph may contain an exponential number of such structures, thus we restricted the search on each subgraph to at most 50 cycles and paths or 1.00 CPU second, whichever came first before the search was complete. In order to generate high-quality wheel and fan cuts, we only considered (chordless) cycles and paths with at least 7 vertices that satisfied the facet-defining conditions of Theorems 13 and 15. The number of generated wheel and fan cuts for each instance is presented in Table 4. The last column of this table shows the aggregate time spent on generating all 4-hole, wheel and fan

Table 5: Second set of experiments: integer (linear) programming solution results.

Name	opt.	LP rlx. opt. value		# B&C nodes		Sol. time (CPU sec.)	
		Base	+VI	Base	+VI	Base	+VI
RND_0.05	53	66.67	59.22	3,241	0	22.51	5.63
RND_0.10	39	66.67	49.71	213,358	17,005	1,778.93	218.46
RND_0.15	28	66.67	46.60	2,017,952	160,544	35,357.90	2,284.37
RND_0.20	24	66.67	44.92	1,112,661	180,755	23,173.20	4,248.66
RND_0.25	21	66.67	44.30	1,159,561	192,961	34,036.80	5,123.73
RND_0.30	18	66.67	44.29	724,553	227,952	35,861.80	8,733.08
RND_0.35	16	66.67	44.33	615,306	191,929	30,004.00	8,444.93
RND_0.40	15	66.67	44.44	468,742	125,469	26,178.80	6,928.29
RND_0.45	13	66.67	44.44	295,550	126,449	19,691.90	7,250.91
RND_0.50	12	66.67	44.44	222,235	104,277	18,260.00	8,861.15
RND_0.55	11	66.67	44.44	176,862	70,025	13,810.80	6,621.82
RND_0.60	11	66.67	44.44	111,914	49,889	9,102.53	5,930.36
RND_0.65	13	66.67	44.44	66,995	33,969	5,615.65	4,927.96
RND_0.70	15	66.67	44.44	57,423	31,844	5,273.13	5,414.28
RND_0.75	19	66.67	44.48	31,135	19,714	5,163.23	4,490.54
RND_0.80	21	66.67	46.63	74,719	29,461	7,943.90	4,605.04
RND_0.85	25	66.67	49.75	46,781	22,872	4,574.90	2,404.52
RND_0.90	31	66.67	50.00	18,562	7,787	1,897.80	492.21
RND_0.95	46	66.67	51.50	0	0	34.24	5.16

cuts for each instance. It should be mentioned that, for all instances, the CPU time taken to generate the wheel and fan inequalities was negligible compared to the required time to generate the 4-hole cuts. In fact, the maximum time spent by the algorithm to generate the entire set of wheel and fan cuts for an instance remained below 1.20 CPU seconds.

The results of the second set of experiments are presented in Table 5. We solved each instance using the base formulation (5), as well as its enhancement obtained by adding the generated valid inequalities, denoted by “+VI” in Table 5. The second column of this table shows the optimal solution value of the MAXIMUM IUC problem for each instance. The next two columns present the optimal solution value of the corresponding LP relaxation problems. The last four columns of this table compare the size of the branch-and-cut (B&C) trees and solution times. In this set of experiments, the automatic cut-generation of the solver was also on. The results clearly show the effectiveness of the proposed valid inequalities. Significant improvements in the solution times are observed for the graphs with the edge density of less than 65%, as well as those with the edge density of 80% and more, due to incorporating the generated cutting planes. The +VI solution times were slightly better than those for the base formulation on RND_0.65 and RND_0.75, and slightly worse on RND_0.70. The better solution time for each instance is shown in bold in Table 5. For all instances, the number of B&C nodes was considerably smaller under the +VI strategy.

6. Conclusion

In this paper, we presented a study of the IUC polytope associated with a simple undirected graph $G = (V, E)$, defined as the convex hull of the incidence vectors of all subsets of V inducing cluster subgraphs of G . It was shown that the fractional IUC polytope, obtained from relaxing integrality of the variables in definition of the original polytope, is a polyhedral outer-approximation of a cubically constrained region in the space of the original variables; hence, it provides a very weak approximation of the (integral) IUC polytope, remarking the importance of incorporating strong valid inequalities in integer (linear) programming solution methods of the MAXIMUM IUC problem. We derived several facet-defining valid inequalities for the IUC polytope associated with cycle, anti-cycle, star, double-star, fan and wheel graphs, along with the conditions that they remain facet-defining for the IUC polytope associated with a supergraph containing them, as well as some results concerning the corresponding lifting procedures. We also presented a complete description of the IUC polytope for some classes of these graphs. In our presentation, we spotted the similarities between the facial structure of the IUC polytope and those of the independent set (vertex packing) and clique polytopes. For each family of the proposed valid inequalities, we studied the computational complexity of the corresponding separation problem. In particular, we showed that the separation problem for the families of star, double-star, fan and wheel inequalities are NP-hard. We also examined the effectiveness of the proposed valid inequalities when employed in an integer (linear) programming solution method of the MAXIMUM IUC problem through computational experiments. Our results showed great improvement in solution time of this problem, revealing the merit of the proposed valid inequalities from the computational standpoint.

As an extension of this research, one may search for other facet-producing structures for the IUC polytope. As stated previously, an immediate step could be to investigate webs that subsume cycle and anti-cycle graphs. Developing detailed branch-and-cut algorithms for the MAXIMUM IUC problem based on effective heuristic separation procedures for the proposed valid inequalities in this paper is another direction for future research. Another interesting direction is to explore LP-based scale reduction techniques for the MAXIMUM IUC problem. It should be mentioned that, because of the cubic nature of the IUC formulation (5) as explained before, $\mathcal{P}_{IUC}^F(G)$ does not generally hold the favorable structural properties of the fractional independent set (vertex packing) and clique polytopes such as persistency [39]. However, we observed through a set of computational experiments that the frequency of non-persistent optimal solutions to LP relaxation of the IUC formulation (5) is not high. In this regard, it is of interest to identify the conditions under which an LP relaxation optimal solution of (5) shows persistency. Finally, we note that several other integer (linear) programming formulations for the MAXIMUM IUC problem can be obtained from the cubic formulation (6) through different linearization techniques, such as the method of Sherali and Adams [48] or the one of Boros and Hammer [10] in higher-dimensional spaces than the space of the original variables.

These formulations will have theoretically stronger LP relaxations than (5), hence it is also interesting to compare their performances in practice.

Acknowledgements. Partial support by AFOSR award FA9550-19-1-0161 is gratefully acknowledged.

References

- [1] B. P. W. Ames and S. A. Vavasis. Convex optimization for the planted k-disjoint-clique problem. *Mathematical Programming*, 143(1):299–337, Feb 2014.
- [2] L. Bastos, L. S. Ochi, F. Protti, An. Subramanian, I. C. Martins, and R. G. S. Pinheiro. Efficient algorithms for cluster editing. *Journal of Combinatorial Optimization*, 31(1):347–371, 2016.
- [3] C. Berge. Färbung von Graphen, deren sämtliche bzw. deren ungerade Kreise starr sind (zusammenfassung). *Wissenschaftliche Zeitschrift, Martin Luther Universität Halle-Wittenberg, Mathematisch-Naturwissenschaftliche*, 10:114–115, 1961.
- [4] D. Bienstock. On the complexity of testing for odd holes and induced odd paths. *Discrete Mathematics*, 90(1):85–92, 1991.
- [5] D. Bienstock. Corrigendum to: On the complexity of testing for odd holes and induced odd paths. *Discrete Mathematics*, 102(1):109, 1992.
- [6] S. Böcker. A golden ratio parameterized algorithm for cluster editing. *Journal of Discrete Algorithms*, 16:79–89, 2012.
- [7] S. Böcker and J. Baumbach. Cluster editing. In P. Bonizzoni, V. Brattka, and Löwe B., editors, *The Nature of Computation. Logic, Algorithms, Applications. CiE 2013*, volume 7921 of *Lecture Notes in Computer Science*, pages 33–44. Springer, Berlin, Heidelberg, 2013.
- [8] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. The maximum clique problem. In D.-Z. Du and P. M. Pardalos, editors, *Handbook of Combinatorial Optimization*, pages 1–74, Dordrecht, The Netherlands, 1999. Kluwer Academic Publishers.
- [9] A. Boral, M. Cygan, T. Kociumaka, and M. Pilipczuk. A fast branching algorithm for cluster vertex deletion. *Theory of Computing Systems*, 58(2):357–376, 2016.
- [10] E. Boros and P. L. Hammer. Pseudo-boolean optimization. *Discrete applied mathematics*, 123(1-3):155–225, 2002.
- [11] H. Chang and H. Lu. A faster algorithm to recognize even-hole-free graphs. *Journal of Combinatorial Theory, Series B*, 113:141–161, 2015.

- [12] Y. Chen and J. Flum. On parameterized path and chordless path problems. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, pages 250–263. IEEE, 2007.
- [13] M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour, and K. Vušković. Recognizing Berge graphs. *Combinatorica*, 25(2):143–186, 2005.
- [14] M. Chudnovsky, N. Robertson, P. Seymour, and R. Thomas. The strong perfect graph theorem. *Annals of Mathematics*, 164:51–229, 2006.
- [15] M. Chudnovsky, A. Scott, P. Seymour, and S. Spirkl. Detecting an odd hole. *Journal of the ACM*, 67(1): Article 5, 2020.
- [16] M. Colombi, R. Mansini, and M. Savelsbergh. The generalized independent set problem: Polyhedral analysis and solution approaches. *European Journal of Operational Research*, 260(1):41–55, 2017.
- [17] M. Conforti, G. Cornuéjols, A. Kapoor, and K. Vušković. Even-hole-free graphs part II: Recognition algorithm. *Journal of Graph Theory*, 40(4):238–266, 2002.
- [18] F. Dehne, M. A. Langston, X. Luo, S. Pitre, P. Shaw, and Y. Zhang. The cluster editing problem: Implementations and experiments. In H.L. Bodlaender and M.A. Langston, editors, *Parameterized and Exact Computation. IWPEC 2006*, volume 4169 of *Lecture Notes in Computer Science*, pages 13–24, Berlin, Heidelberg, 2006. Springer.
- [19] Z. Ertem, E. Lykhowyd, Y. Wang, and S. Butenko. The maximum independent union of cliques problem: complexity and exact approaches. *Journal of Global Optimization*, 76:545–562, 2020.
- [20] Z. Ertem, A. Veremyev, and S. Butenko. Detecting large cohesive subgroups with high clustering coefficients in social networks. *Social Networks*, 46:1–10, 2016.
- [21] S. Fiorini, G. Joret, and O. Schaudt. Improved approximation algorithms for hitting 3-vertex paths. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 238–249. Springer, 2016.
- [22] F. V. Fomin, S. Gaspers, D. Kratsch, M. Liedloff, and S. Saurabh. Iterative compression and exact algorithms. *Theoretical Computer Science*, 411(7-9):1045–1053, 2010.
- [23] F. V. Fomin, S. Gaspers, D. Lokshantov, and S. Saurabh. Exact algorithms via monotone local search. *Journal of the ACM*, 66(2): Article 8, 2019.
- [24] F. V. Fomin, S. Kratsch, M. Pilipczuk, M. Pilipczuk, and Y. Villanger. Tight bounds for parameterized complexity of cluster editing with a small number of clusters. *Journal of Computer and System Sciences*, 80(7):1430–1447, 2014.

- [25] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W.H. Freeman and Company, New York, 1979.
- [26] J. Gramm, J. Guo, F. Hüffner, and R. Niedermeier. Automated generation of search tree algorithms for hard graph modification problems. *Algorithmica*, 39(4):321–347, 2004.
- [27] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, Berlin, 2nd edition, 1993.
- [28] J. Guo, C. Komusiewicz, R. Niedermeier, and J. Uhlmann. A more relaxed model for graph-based data clustering: s -plex cluster editing. *SIAM Journal on Discrete Mathematics*, 24(4):1662–1683, 2010.
- [29] S. Hosseini and S. Butenko. Algorithms for the generalized independent set problem based on a quadratic optimization approach. *Optimization Letters*, 13:1211–1222, 2019.
- [30] F. Hüffner, C. Komusiewicz, H. Moser, and R. Niedermeier. Fixed-parameter algorithms for cluster vertex deletion. *Theory of Computing Systems*, 47(1):196–217, 2010.
- [31] K. Jansen, P. Scheffler, and G. Woeginger. The disjoint cliques problem. *RAIRO-Operations Research*, 31(1):45–66, 1997.
- [32] R. M. Karp. Reducibility among combinatorial problems. In R.E. Miller, J.W. Thatcher, and Bohlinger J.D., editors, *Complexity of Computer Computations*, IBM Research Symposia Series, pages 85–103. Plenum Press, New York, 1972.
- [33] C. Komusiewicz and J. Uhlmann. Cluster editing with locally bounded modifications. *Discrete Applied Mathematics*, 160(15):2259–2270, 2012.
- [34] M. S. Krishnamoorthy and N. Deo. Node-deletion NP-complete problems. *SIAM Journal on Computing*, 8(4):619–625, 1979.
- [35] T.-N. Le, D. Lokshtanov, S. Saurabh, S. Thomassé, and M. Zehavi. Subquadratic kernels for implicit 3-hitting set and 3-set packing problems. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 331–342. SIAM, 2018.
- [36] J. M. Lewis and M. Yannakakis. The node-deletion problem for hereditary properties is NP-complete. *Journal of Computer and System Sciences*, 20(2):219–230, 1980.
- [37] C. A. Meyer and C. A. Floudas. Trilinear monomials with positive or negative domains: Facets of the convex and concave envelopes. In C.A. Floudas and P. Pardalos, editors, *Frontiers in Global Optimization*, volume 74 of *Nonconvex Optimization and Its Applications*, pages 327–352. Kluwer Academic Publishers, Boston, MA, 2004.

- [38] G. L. Nemhauser and L. E. Trotter. Properties of vertex packings and independence system. *Mathematical Programming*, 6:48–61, 1974.
- [39] G. L. Nemhauser and L. E. Trotter. Vertex packing: structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975.
- [40] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. Wiley, New York, 1999.
- [41] M. W. Padberg. On the facial structure of set packing polyhedra. *Mathematical Programming*, 5:199–215, 1973.
- [42] M. W. Padberg. A note on zero-one programming. *Operations Research*, 23(4):833–837, 1975.
- [43] J. Pattillo, N. Youssef, and S. Butenko. On clique relaxation models in network analysis. *European Journal of Operational Research*, 226(1):9–18, 2013.
- [44] M. Pilipczuk. Exact algorithms for induced subgraph problems. In M.Y. Kao, editor, *Encyclopedia of Algorithms*. Springer, Boston, MA, 2015.
- [45] A. Pyatkin, E. Lykhovyd, and S. Butenko. The maximum number of induced open triangles in graphs of a given order. *Optimization Letters*, 13:1927–1935, 2019.
- [46] S. E. Schaeffer. Graph clustering. *Computer science review*, 1(1):27–64, 2007.
- [47] R. Shamir, R. Sharan, and D. Tsur. Cluster graph modification problems. *Discrete Applied Mathematics*, 144(1-2):173–182, 2004.
- [48] H. D. Sherali and W. P. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal of Discrete Mathematics*, 3:411–430, 1990.
- [49] H. D. Sherali and J. C. Smith. A polyhedral study of the generalized vertex packing problem. *Mathematical Programming*, 107(3):367–390, 2006.
- [50] L. E. Trotter. A class of facet producing graphs for vertex packing polyhedra. *Discrete Mathematics*, 12:373–388, 1975.
- [51] R. Van Bevern, H. Moser, and R. Niedermeier. Approximation and tidying – a problem kernel for s -plex cluster vertex deletion. *Algorithmica*, 62(3-4):930–950, 2012.
- [52] L. A. Wolsey. Facets and strong valid inequalities for integer programs. *Operations Research*, 24(2):367–372, 1976.
- [53] Q. Wu and J. Hao. A review on algorithms for maximum clique problems. *European Journal of Operational Research*, 242(3):693–709, 2015.

- [54] J. You, J. Wang, and Y. Cao. Approximate association via dissociation. *Discrete Applied Mathematics*, 219:202–209, 2017.
- [55] E. Zemel. Lifting the facets of zero–one polytopes. *Mathematical Programming*, 15(1):268–277, 1978.

Appendix A Proof of Theorem 7

Consider a sequential partitioning of H as described in the proof of Theorem 3. In a clockwise walk, let v_i^p denote the i -th vertex in the p -th partition, where $i \in \{1, 2, 3\}$, $p \in \{1, \dots, q+1\}$, and $p = q+1$ is the partition with less than 3 vertices. The cases where $|H \cap N(v)| \leq 1$ are trivial, so we start with $|H \cap N(v)| = 2$. Let u and w be the neighbors of v in H , i.e., $H \cap N(v) = \{u, w\}$, and $d = d_{G[H]}(u, w)$ denote the distance (length of the shorter path) between u and w in $G[H]$. Without loss of generality, we assume $u = v_1^1$ and the shorter path between u and w in $G[H]$ is due to a clockwise walk from u toward w . We treat $d \in \{1, 2\}$ separately as those are the only cases where u and w are located in the same partition. For each case, we present an IUC of size $2q + \lfloor \frac{2r}{3} \rfloor + 1$ in $G[H \cup \{v\}]$. In the following presentation, each set of vertices is a clique, and j denotes the partition that contains w .

$$\begin{aligned}
d = 1, r = 1 : & \quad \{v, u = v_1^1, w = v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\} \\
d = 1, r = 2 : & \quad \{v, u = v_1^1, w = v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\}; \{v_1^{q+1}\} \\
d = 2, r = 1 : & \quad \{v\}; \{v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\}; \{v_1^{q+1}\} \\
d = 2, r = 2 : & \quad \{v\}; \{v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\}; \{v_1^{q+1}, v_2^{q+1}\} \\
d \bmod 3 = 0, r = 1 : & \quad \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, q\} \\
d \bmod 3 = 0, r = 2 : & \quad \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, q\}; \{v_2^{q+1}\} \\
d \bmod 3 = 1, r = 1 : & \quad \{v, w = v_2^j\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \quad \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}\} \\
d \bmod 3 = 1, r = 2 : & \quad \{v, w = v_2^j\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \quad \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}, v_2^{q+1}\} \\
d \bmod 3 = 2, r = 1 : & \quad \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \{v_2^j\}; \\
& \quad \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}\} \\
d \bmod 3 = 2, r = 2 : & \quad \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \{v_2^j\}; \\
& \quad \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}, v_2^{q+1}\}
\end{aligned}$$

Next, we consider $|H \cap N(v)| = 3$ and $r = 1$. Let $H \cap N(v) = \{u, w, z\}$. As before, suppose that $u = v_1^1$, and in a clockwise walk starting from u , first w is reached in $d_1 = d_{G[H]}(u, w)$ steps, and then z is reached in $d_2 = d_{G[H]}(w, z)$ steps from w . We also assume $d_1 \leq d_2$. In the following presentation, j and k denote the partitions containing w and z , respectively. First, we consider the cases where u and w are located in the same partition, i.e., $d_1 \in \{1, 2\}$. Note

that u , w and z are placed in one partition only if $d_1 = d_2 = 1$.

$$\begin{aligned}
d_1 = 1, d_2 = 1 : & \{v, u = v_1^1, w = v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\} \\
d_1 = 1, d_2 \bmod 3 = 0 : & \{v, u = v_1^1, w = v_2^1\}; \{v_1^2\}; \\
& \{v_3^p, v_1^{p+1}\}, \forall p \in \{2, \dots, q-1\}; \{v_3^q\} \\
d_1 = 1, d_2 \bmod 3 = 1 : & \{v, u = v_1^1, w = v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\} \\
d_1 = 1, d_2 \bmod 3 = 2 : & \{v, u = v_1^1, w = v_2^1\}; \{v_2^p, v_3^p\}, \forall p \in \{2, \dots, q\} \\
d_1 = 2, d_2 \bmod 3 = 0 : & \{v\}; \{v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, q\}; \{v_1^{q+1}\} \\
d_1 = 2, d_2 \bmod 3 = 1 : & \{v, z = v_1^k\}; \{v_2^1\}; \{v_1^p, v_2^p\}, \forall p \in \{2, \dots, k-1\}; \\
& \{v_3^p, v_1^{p+1}\}, \forall p \in \{k, \dots, q\} \\
d_1 = 2, d_2 \bmod 3 = 2 : & \{v\}; \{v_2^1\}; \{v_1^2\}; \{v_3^p, v_1^{p+1}\}, \forall p \in \{2, \dots, q\}
\end{aligned}$$

Finally, the following IUCs correspond to the cases where u , w and z are all located in separate partitions.

$$\begin{aligned}
d_1 \bmod 3 = 0, d_2 \bmod 3 = 0 : & \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, q\} \\
d_1 \bmod 3 = 0, d_2 \bmod 3 = 1 : & \{v, z = v_2^k\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, k-1\}; \\
& \{v_1^p, v_2^p\}, \forall p \in \{k+1, \dots, q\}; \{v_1^{q+1}\} \\
d_1 \bmod 3 = 0, d_2 \bmod 3 = 2 : & \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, k-1\}; \{v_2^k\}; \\
& \{v_1^p, v_2^p\}, \forall p \in \{k+1, \dots, q\}; \{v_1^{q+1}\} \\
d_1 \bmod 3 = 1, d_2 \bmod 3 = 0 : & \{v, w = v_2^j\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_1^{j+1}\}; \{v_3^p, v_1^{p+1}\}, \forall p \in \{j+1, \dots, q\}; \\
d_1 \bmod 3 = 1, d_2 \bmod 3 = 1 : & \{v, w = v_2^j\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}\} \\
d_1 \bmod 3 = 1, d_2 \bmod 3 = 2 : & \{v, u = v_1^1\}; \{v_3^p, v_1^{p+1}\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_3^p, v_1^{p+1}\}, \forall p \in \{j, \dots, k-2\}; \{v_3^{k-1}\}; \\
& \{v_2^p, v_3^p\}, \forall p \in \{k, \dots, q\} \\
d_1 \bmod 3 = 2, d_2 \bmod 3 = 0 : & \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_2^j\}; \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, q\}; \{v_1^{q+1}\} \\
d_1 \bmod 3 = 2, d_2 \bmod 3 = 1 : & \{v, z = v_1^k\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_2^j\}; \{v_1^p, v_2^p\}, \forall p \in \{j+1, \dots, k-1\}; \\
& \{v_3^p, v_1^{p+1}\}, \forall p \in \{k, \dots, q\} \\
d_1 \bmod 3 = 2, d_2 \bmod 3 = 2 : & \{v\}; \{v_2^p, v_3^p\}, \forall p \in \{1, \dots, j-1\}; \\
& \{v_2^j\}; \{v_1^{j+1}\}; \{v_3^p, v_1^{p+1}\}, \forall p \in \{j+1, \dots, q\}
\end{aligned}$$

The proof is complete.

An Improved Approximation for MAXIMUM k -DEPENDENT SET on Bipartite Graphs

Syedmohammadhossein Hosseini, Sergiy Butenko

Texas A&M University, Department of Industrial and Systems Engineering
College Station, TX 77843-3131, {hosseini, butenko}@tamu.edu

Abstract

We present a $(1 + \frac{k}{k+2})$ -approximation algorithm for the MAXIMUM k -DEPENDENT SET problem on bipartite graphs for any $k \geq 1$. For a graph with n vertices and m edges, the algorithm runs in $\mathcal{O}(km\sqrt{n})$ time and improves upon the previously best-known approximation ratio of $1 + \frac{k}{k+1}$ established by Kumar et al. [Theoretical Computer Science, 526: 90–96 (2014)]. Our proof also indicates that the algorithm retains its approximation ratio when applied to the (more general) class of König-Egerváry graphs.

Keywords: maximum k -dependent set, approximation algorithms, bipartite graphs, König-Egerváry graphs

1. Introduction

Given a simple, undirected graph $G = (V, E)$, where V is the set of vertices and E is the set of edges, and a constant k , a set of vertices $S \subseteq V$ is called a *k -dependent set* if the degree of every vertex in the subgraph induced by S is at most k . The MAXIMUM k -DEPENDENT SET problem is to find a maximum-cardinality k -dependent set in G . This problem, first introduced in [6], is a well-known generalization of the classical MAXIMUM INDEPENDENT SET problem (the case of $k = 0$) and is equivalent to the MAXIMUM s -PLEX problem [2, 17] on the complement graph $\bar{G}$ of G , where $s = k + 1$. For any $k \geq 0$, the MAXIMUM k -DEPENDENT SET problem is NP-hard on general graphs [6], and by the theorem of Feige and Kogan [8], it cannot be approximated (in polynomial time) within a factor of $n^{1-\epsilon}$, for any $\epsilon > 0$, unless NP=ZPP. Due to these hardness results, the MAXIMUM k -DEPENDENT SET problem has been also studied on special classes of graphs. It is known that, for any $k \geq 1$, the problem is polynomial-time solvable on cographs, split graphs, and graphs with bounded treewidth, hence trees, but it remains NP-hard on unit-disk graphs, planar graphs, and bipartite graphs [5, 10]. A constant-factor approximation algorithm and a polynomial time approximation scheme (PTAS) for this problem on unit-disk graphs have been presented in [3] and [10], respectively. Due to the Lipton-Tarjan separation theorem [16], the problem is also known to admit PTAS on planar graphs. In

this paper, we focus on the class of bipartite graphs. Given the NP-hardness result of Dessmark et al. [5], we are interested in solving the following problem approximately:

MAXIMUM k -DEPENDENT SET IN BIPARTITE GRAPHS (MAX- k -DSBG)

Given: A bipartite graph $G = (V, E)$ and an integer $k \geq 1$.
Find: A maximum-cardinality k -dependent set in G .

We present a $(1 + \frac{k}{k+2})$ -approximation algorithm for MAX- k -DSBG, which is an improvement upon the best previously known approximation ratio of $1 + \frac{k}{k+1}$, established by Kumar et al. [15].

To the best of our knowledge, the work of Kumar et al. [15] is the only nontrivial approximation algorithm for MAX- k -DSBG in the literature. More specifically, Kumar et al. [15] have studied approximability of the MAXIMUM Π_{k_1, k_2} SUBGRAPH and MINIMUM Π_{k_1, k_2} VERTEX DELETION problems. Given a bipartite graph $G = (V_1 \cup V_2, E)$ with the parts V_1 and V_2 , let Π_{k_1, k_2} be a property characterized by a finite set $\mathcal{H}$ of forbidden induced subgraphs, where any $H \in \mathcal{H}$ contains at most k_1 vertices in one of the parts V_1 and V_2 and at most k_2 vertices in the other of the parts. Then the MAXIMUM Π_{k_1, k_2} SUBGRAPH problem is to find a maximum-cardinality subset of vertices satisfying the property Π_{k_1, k_2} (that is, containing no induced subgraph isomorphic to a graph from $\mathcal{H}$). The MINIMUM Π_{k_1, k_2} VERTEX DELETION is the corresponding node deletion problem. Kumar et al. [15] have shown that both these problems are APX-complete and proposed approximation algorithms based on the technique of iterative rounding applied to a sequence of optimal solutions of the linear programming relaxations. The approximation factors they established are given by $\max\{k_1, k_2\}$ for the MINIMUM Π_{k_1, k_2} VERTEX DELETION and $2 - \frac{1}{\max\{k_1, k_2\}}$ for the MAXIMUM Π_{k_1, k_2} SUBGRAPH, respectively. It is evident that MAX- k -DSBG is equivalent to MAXIMUM Π_{k_1, k_2} SUBGRAPH if the corresponding set $\mathcal{H}$ of forbidden subgraphs consists of a *star* graph $K_{1, k+1}$, i.e., a single vertex in one part adjacent to $k+1$ vertices in the other part of the graph. Due to this equivalence, the method of Kumar et al. [15] provides a $(1 + \frac{k}{k+1})$ -approximation algorithm for MAX- k -DSBG. Here, we present is a purely combinatorial approximation algorithm for this problem that achieves the approximation ratio of $1 + \frac{k}{k+2}$ and runs in time bounded by $\mathcal{O}(km\sqrt{n})$ on a graph with n vertices and m edges. Our method is surprisingly simple.

We use the following concepts and notations. Let $n := |V|$ and $m := |E|$ denote the number of vertices and edges in G , respectively. For a subset of vertices S , $G[S]$ denotes the subgraph induced by S in G . A subset of edges $M \subseteq E$ is called a matching if no two edges in M have a common incident vertex. A set of vertices $C \subseteq V$ is a vertex cover if every edge in E has at least one of its end-vertices in C . König-Egerváry matching theorem [13] (see [19] for a short proof) states that the maximum cardinality of a matching in a bipartite graph G is equal to the minimum cardinality of a vertex cover. The reader is referred to [1] and [21] for background on polynomial-time algorithms

Algorithm 1 AN APPROXIMATION ALGORITHM FOR MAX- k -DSBG

Require: An integer constant $k \geq 1$ and a bipartite graph $G = (V, E)$

```
1: for  $i = 1$  to  $k$  do
2:   find a maximum-cardinality matching  $M \subseteq E$  in  $G = (V, E)$ 
3:    $E \leftarrow E \setminus M$ 
4: end for
5: find a maximum independent set  $S$  in the residual graph  $G$ 
6: return  $S$ 
```

for problems in graphs/networks and approximation algorithms, respectively.

2. The Algorithm

In this section, we present our algorithm, establish its approximation ratio, and analyze its time complexity.

The proposed method consists of two simple stages. The first stage is comprised of k recursive iterations, each of which computes a maximum-cardinality matching and removes it from the graph. This is followed by the second stage, which computes a maximum independent set S in the residual graph and outputs it as an approximate solution for MAX- k -DSBG. The process is formally described in Algorithm 1. Notice that, in line 3 of the algorithm, some *edges* of the graph are deleted, not their *end-vertices*. An illustration of the process for $k = 1$ on an example graph is provided in Figure 1.

To establish the correctness of the algorithm, observe that a vertex $v \in S$ can be incident to at most k (deleted) edges in $G[S]$; hence, S is a feasible solution to MAX- k -DSBG. We proceed to establish the approximation ratio of the algorithm.

Let $\mathcal{M}_i$, $i \in \{1, \dots, k\}$, be the *accumulative* set of edges deleted through the first i iterations of lines 2 and 3 of Algorithm 1, and let G_i denote the corresponding residual graph, i.e., the graph obtained from G by removing $\mathcal{M}_i$. Furthermore, let I_i , $i \in \{1, \dots, k\}$, be a maximum independent set of G_i , and let I_k^* be an arbitrary maximum k -dependent set in G ; by this notation, $S \equiv I_k$. Patently, the edge set of the subgraph induced by I_k^* , i.e., $E(G[I_k^*])$, can be decomposed into a collection of disjoint matchings in G ; removing the set of edges belonging to these matchings from the graph, and then, finding a maximum independent set in the residual graph generates an optimal solution to MAX- k -DSBG on G . The intuition behind our proof is based on observation that the performance of Algorithm 1 depends on the number of edges from $E(G[I_k^*])$ that are not included in the maximum-cardinality matchings detected through k iterations of the algorithm.

Lemma 1. $|I_k| \geq n - \frac{|\mathcal{M}_k|}{k}$, for every fixed $k \geq 1$.

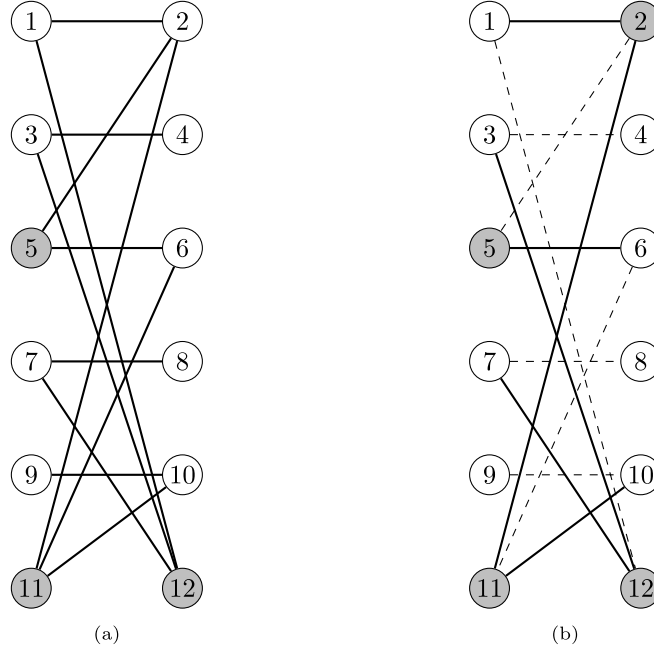


Figure 1: An example for $k = 1$: (a) the set of white vertices is a maximum 1-dependent set, (b) a maximum-cardinality matching M is illustrated by the dashed edges, and the gray vertices depict a minimum vertex cover of the residual graph after deleting M ; the set of white vertices is the output of the algorithm.

PROOF. For $k = 1$,

$$|I_1| \geq \alpha(G) = n - |\mathcal{M}_1|,$$

where $\alpha(G)$ denotes the independence number of G , and the equality is due to the König-Egerváry matching theorem. For $k \geq 2$, the proof is by induction. The base case is already established, so for a fixed $k \geq 2$ and some $i \in \{2, \dots, k\}$, suppose that

$$|I_{i-1}| \geq n - \frac{|\mathcal{M}_{i-1}|}{i-1}.$$

Then, using the induction hypothesis and the relation $|M(G_{i-1})| = n - |I_{i-1}|$ given by the König-Egerváry matching theorem, we obtain

$$|\mathcal{M}_i| = |\mathcal{M}_{i-1}| + |M(G_{i-1})| \geq (i-1)(n - |I_{i-1}|) + (n - |I_{i-1}|) = i(n - |I_{i-1}|),$$

where $M(G_{i-1})$ denotes a maximum-cardinality matching in G_{i-1} . Finally, the fact that G_i is obtained by removing $M(G_{i-1})$ from G_{i-1} , implies that $|I_{i-1}| \leq |I_i|$; hence,

$$|\mathcal{M}_i| \geq i(n - |I_i|).$$

This completes the proof. $\square$

Let $E'_k := E(G[I_k^*]) \setminus \mathcal{M}_k$ denote the set of edges with both end-vertices in the optimal solution I_k^* that are not deleted through executions of lines 2 and 3 of Algorithm 1.

Lemma 2. $|I_k| \geq \frac{|I_k^*|}{2} + \frac{|E'_k|}{k}$, for every fixed $k \geq 1$.

PROOF. Let $D_k^* := V \setminus I_k^*$ be the set of vertices not included in the optimal solution I_k^* , and $E''_k := E(G[I_k^*]) \setminus E'_k = E(G[I_k^*]) \cap \mathcal{M}_k$. Then,

$$|\mathcal{M}_k| \leq |E''_k| + k|D_k^*| = (|E(G[I_k^*])| - |E'_k|) + k|D_k^*| \leq \frac{k|I_k^*|}{2} - |E'_k| + k|D_k^*|,$$

where the first inequality is due to the fact that every edge from $\mathcal{M}_k$ not belonging to E''_k has at least one end-vertex in D_k^* , and at most k edges containing a vertex $v \in D_k^*$ can belong to $\mathcal{M}_k$, i.e., at most one edge from each of the k computed matchings. The second inequality is implied by the fact that the degrees of the vertices of I_k^* in $G[I_k^*]$ are bounded by k . This result, along with Lemma 1, leads to

$$|I_k| \geq n - \frac{|\mathcal{M}_k|}{k} \geq (n - |D_k^*|) - \frac{|I_k^*|}{2} + \frac{|E'_k|}{k} = |I_k^*| - \frac{|I_k^*|}{2} + \frac{|E'_k|}{k} = \frac{|I_k^*|}{2} + \frac{|E'_k|}{k}.$$

□

Lemma 3. $\frac{|I_k|}{|I_k^*|} \geq \frac{1}{2} + \frac{1}{2(k+1)}$, for every fixed $k \geq 1$.

PROOF. Suppose $|I_k^*| \leq \frac{2|E'_k|(k+1)}{k}$, or equivalently $\frac{|E'_k|}{k} \geq \frac{|I_k^*|}{2(k+1)}$. This, together with Lemma 2, implies

$$|I_k| \geq \frac{|I_k^*|}{2} + \frac{|E'_k|}{k} \geq \frac{|I_k^*|}{2} + \frac{|I_k^*|}{2(k+1)} = |I_k^*| \left(\frac{1}{2} + \frac{1}{2(k+1)} \right).$$

Now, consider the case $|I_k^*| > \frac{2|E'_k|(k+1)}{k}$, or equivalently $|E'_k| < \frac{k|I_k^*|}{2(k+1)}$.

Observe that, at most $|E'_k|$ vertices (one for each edge in $G_k[I_k^*]$) should be deleted from I_k^* to turn it into an independent set in G_k . Thus, $|I_k| \geq |I_k^*| - |E'_k|$, which, under the case assumption, leads to

$$|I_k| \geq |I_k^*| - |E'_k| > |I_k^*| - \frac{k|I_k^*|}{2(k+1)} = |I_k^*| \left(1 - \frac{k}{2(k+1)} \right) = |I_k^*| \left(\frac{1}{2} + \frac{1}{2(k+1)} \right).$$

□

Theorem 1. Algorithm 1 is a $(1 + \frac{k}{k+2})$ -approximation algorithm for MAX- k -DSBG that runs in $\mathcal{O}(km\sqrt{n})$ time.

PROOF. By Lemma 3 and the feasibility of an output of Algorithm 1 to MAX- k -DSBG,

$$\frac{|I_k^*|}{|S|} = \frac{|I_k^*|}{|I_k|} \leq 1 + \frac{k}{k+2},$$

which establishes the approximation ratio.

Finding a maximum-cardinality matching M in a bipartite graph $G = (V, E)$ takes no more than $\mathcal{O}(m\sqrt{n})$ time by the Hopcroft-Karp algorithm [11]. Deleting M itself takes no more than a linear time. Thus, the loop of Algorithm 1 runs in a time bounded by $\mathcal{O}(km\sqrt{n})$. Besides, by the König-Egerváry matching theorem, the Hopcroft-Karp algorithm can be also used to find a maximum independent set in the residual graph G_k . Therefore, the time complexity of Algorithm 1 is no worse than $\mathcal{O}(km\sqrt{n})$. $\square$

It is worth noting that 1-dependent set and *independent union of cliques* (IUC) are equivalent in bipartite graphs, so Algorithm 1 (for $k = 1$) provides a $\frac{4}{3}$ -approximation algorithm for the MAXIMUM IUC problem [7, 12] on this class of graphs. We should also point out that finding a maximum-cardinality matching can be done by randomized algorithms with a better time complexity than the Hopcroft-Karp method; see e.g. [18].

Bipartite graphs are a subclass of König-Egerváry graphs. A graph is called a König-Egerváry (KE) graph if the maximum cardinality of a matching is equal to the minimum cardinality of a vertex cover in the graph. It is known that the MAXIMUM INDEPENDENT SET problem is polynomial-time solvable on KE graphs [4, 20]. Given this result, as well as the fact that our proof relies only on the König-Egerváry matching theorem, Algorithm 1 is indeed a $(1 + \frac{k}{k+2})$ -approximation algorithm for the MAXIMUM k -DEPENDENT SET problem on the (more general) class of KE graphs. A forbidden-subgraph characterization of KE graphs is presented in [14].

3. Examples Achieving the Approximation Ratio Bound

In this section, we present a family of bipartite graphs, on which Algorithm 1 may achieve the $1 + \frac{k}{k+2} = \frac{2(k+1)}{k+2}$ approximation bound.

For a fixed $k \geq 1$, let $G = (V_1 \cup V_2, E)$ be a bipartite graph on $2(k+2)$ vertices, where $V_1 = A \cup \{u\}$, $V_2 = B \cup \{w\}$, $A = \{a_1, \dots, a_{k+1}\}$, and $B = \{b_1, \dots, b_{k+1}\}$. We distinguish u and w from the rest of the vertices, as they are the only ones not included in the optimal solution of MAX- k -DSBG on G . The construction of G is as follows: u and w are adjacent; every vertex in A (resp. B) is adjacent to w (resp. u) as well as all vertices in B (resp. A) except for the one with the same index. That is,

$$\begin{aligned} E = & \{\{u, w\}\} \cup \{\{a_i, b_{i'}\} \mid i \in \{1, \dots, k+1\}, i' \in \{1, \dots, k+1\} \setminus \{i\}\} \\ & \cup \{\{a_i, w\} \mid i \in \{1, \dots, k+1\}\} \cup \{\{b_i, u\} \mid i \in \{1, \dots, k+1\}\}. \end{aligned}$$

Observe that, by this construction, $A \cup B$ is the (unique) optimal solution of MAX- k -DSBG on G , with the optimal solution value $|A \cup B| = 2(k+1)$.

Besides, $G[A \cup B]$ has k disjoint perfect matchings, each of which constitutes a perfect matching for G with $\{u, w\} \in E$.

We show that the output of Algorithm 1 on G may contain exactly $k + 2$ vertices; we present the process for even and odd values of k separately.

Let k be even, and consider the performance of Algorithm 1 on G as follows: at every iteration $j \in \{1, \dots, k\}$, the algorithm removes the (perfect) matching

$$M_j = (\{\{a_i, b_{i+j}\} \mid i \in \{1, \dots, k+1\}\} \setminus \{a_j, b_{2j}\}) \cup \{\{a_j, w\}, \{b_{2j}, u\}\} \quad (1)$$

from G , with the convention $b_{i+k+1} \equiv b_i$. Note that, at every iteration $j \in \{1, \dots, k\}$, the algorithm includes the edges $\{a_j, w\}$ and $\{b_{2j}, u\}$ in M_j and leaves the edges $\{a_j, b_{2j}\}$ and $\{u, w\}$ intact. As a result, after k iterations of the loop of Algorithm 1, the set of edges $\tilde{E} = \{\{a_j, b_{2j}\}, \forall j \in \{1, \dots, k\}\}$ will remain in the residual graph, i.e., the graph obtained from G by removing $\mathcal{M}_k = \bigcup_{j=1}^k M_j$. Observe that, given k is even, there exists no index $j \in \{1, \dots, k\}$ such that $2j = 2j' + k + 1$ for some index $j' \in \{1, \dots, j-1\}$. This implies that no two edges in $\tilde{E}$ share an end-vertex, so every minimum vertex cover of the residual graph contains one end-vertex for each edge in $\tilde{E}$. Besides, it is easy to verify that, in addition to $\tilde{E}$, the edge set of the residual graph contains (exactly) three edges, i.e., $\{a_{k+1}, w\}$, $\{w, u\}$, and $\{u, b_{k+1}\}$, which form the (isolated) path $P = \{a_{k+1}, w, u, b_{k+1}\}$. Every minimum vertex cover of this graph must contain two out of four vertices from P . This implies that every independent set of the residual graph is of cardinality $2(k+2) - (k+2) = k+2$.

Now, let k be odd, and consider the abovementioned process with a slight modification: at every iteration $j \in \{1, \dots, \frac{k+1}{2}\}$, the algorithm removes the matching defined by (1), and at every iteration $j \in \{\frac{k+3}{2}, \dots, k\}$, it removes the following (perfect) matching from G :

$$M_j = (\{\{a_i, b_{i+j}\}, \forall i \in \{1, \dots, k+1\}\} \setminus \{a_{j+1}, b_{2j+1}\}) \cup \{\{a_{j+1}, w\}, \{b_{2j+1}, u\}\}.$$

Notice that $b_{k+3} \equiv b_2$ for every odd value of k ; at the iteration $j = \frac{k+3}{2}$, instead of the edge $\{a_{\frac{k+3}{2}}, b_2\}$, the algorithm excludes the edge $\{a_{\frac{k+3}{2}+1}, b_3\}$ from the matching M_j , and it follows this pattern for the rest of the iterations. This ensures that the set of edges connecting A and B in the residual graph, denoted by $\tilde{E}$, are isolated, i.e., no two edges among them have a common end-vertex. Besides, similar to the previous case, the residual graph always includes the path $P = \{a_{\frac{k+3}{2}}, w, u, b_1\}$, which indicates that every independent set of this graph is of cardinality $2(k+2) - (k+2) = k+2$.

Figures 2 and 3 illustrate the process for $k = 3$ and $k = 4$, respectively. In these figures, the (perfect) matching identified by the algorithm at each iteration is depicted by dashed and dotted edges; the dashed lines identify the edges with both end-vertices in the optimal solution, and the dotted lines show the edges incident to u or w . The edges in $\tilde{E}$ are depicted by thick solid lines.

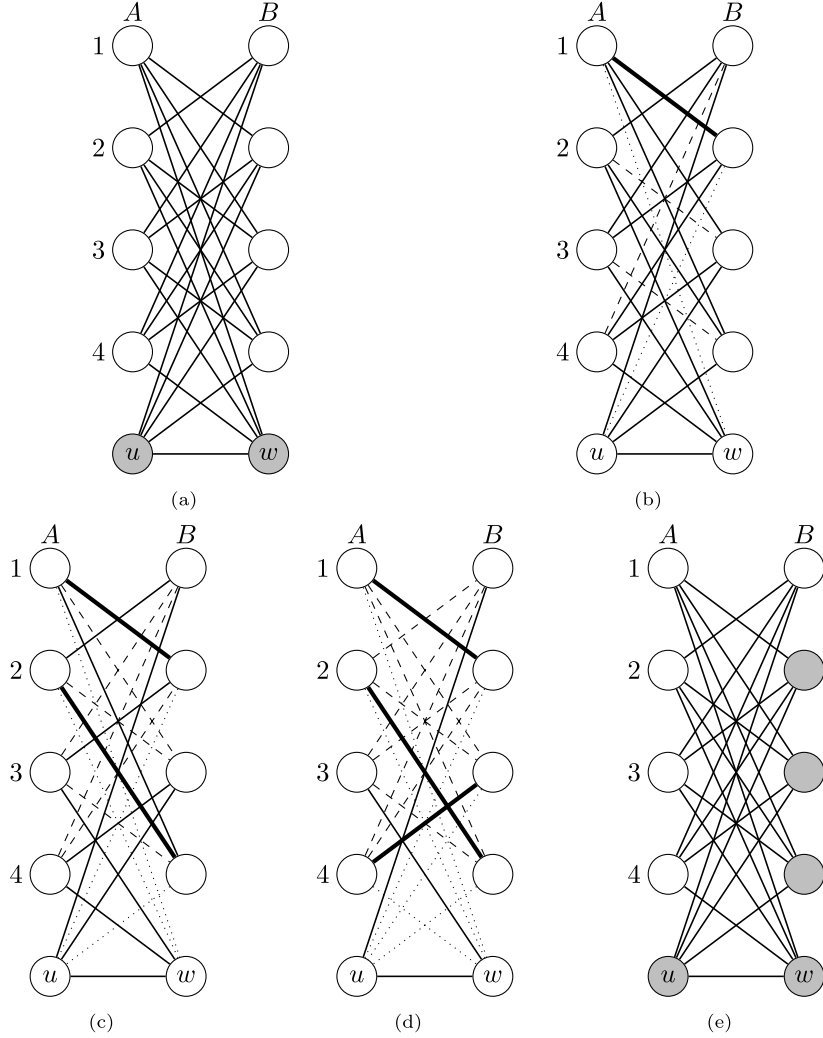


Figure 2: Worst-case example for $k = 3$: (a) the original graph; the set of white vertices is a maximum 3-dependent set, (b) the graph after removing $\mathcal{M}_1$, (c) the graph after removing $\mathcal{M}_2$, (d) the graph after removing $\mathcal{M}_3$, (e) the original graph; the gray vertices depict a minimum vertex cover of the residual graph, i.e., after removing $\mathcal{M}_3$, and the set of white vertices is the output of the algorithm.

4. Conclusion

The MAXIMUM k -DEPENDENT SET problem is NP-hard on the class of bipartite graphs, for any $k \geq 1$. This paper presents a $(1 + \frac{k}{k+2})$ -approximation algorithm for this problem that runs in $\mathcal{O}(km\sqrt{n})$ time on a (bipartite) graph with n vertices and m edges and improves upon its previously best-known approximation ratio of $1 + \frac{k}{k+1}$. Our proof also indicates that the algorithm retains

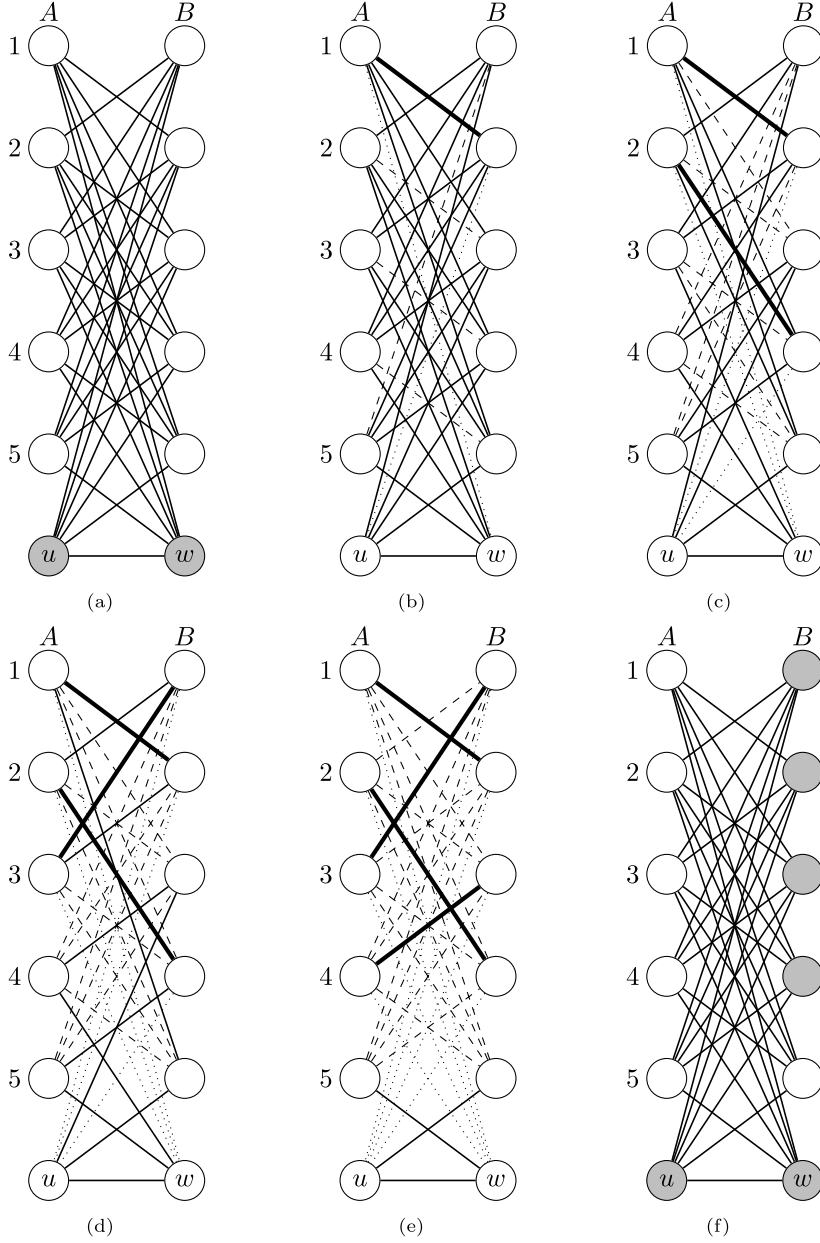


Figure 3: Worst-case example for $k = 4$: (a) the original graph; the set of white vertices is a maximum 4-dependent set, (b) the graph after removing $\mathcal{M}_1$, (c) the graph after removing $\mathcal{M}_2$, (d) the graph after removing $\mathcal{M}_3$, (e) the graph after removing $\mathcal{M}_4$, (f) the original graph; the gray vertices depict a minimum vertex cover of the residual graph, i.e., after removing $\mathcal{M}_4$, and the set of white vertices is the output of the algorithm.

its approximation ratio when applied to König-Egerváry (KE) graphs, which is a superclass of bipartite graphs. Furthermore, we present a family of bipartite graphs, on which the algorithm achieves the approximation bound.

As an extension of this work, one may replace the loop of Algorithm 1, i.e., removing k maximum-cardinality matchings, with removing a single maximum-cardinality k -matching and study the approximation ratio of the new algorithm. A k -matching in a graph is a subset of edges such that every vertex in the graph is incident to at most k of them. The maximum k -matching problem, which is to find a k -matching with the largest number of edges in a given graph, is polynomial-time solvable [9]. The structure of the examples presented in this paper suggests such an algorithm may avoid the pitfall of the current method in missing the “correct” edges, hence achieving an approximation ratio better than $1 + \frac{k}{k+2}$. In the presence of multiple maximum-cardinality matchings in a graph, one may also try to quantify the probability of missing the “correct” edges—based on the employed matching subroutine—which may lead to the design of a randomized variant of the current algorithm with a better expected approximation ratio. The proposed algorithm always generates a feasible solution for the MAXIMUM k -DEPENDENT SET problem on general graphs. Also, the minimum cardinality of a vertex cover is an upper bound on the maximum cardinality of a matching in every graph. Let $\theta(G)$ denote the gap between these two invariants in a graph G . Then, for the classes of graphs with polynomial-time computable maximum independent set in the residual, it is interesting to establish the approximation ratio of the algorithm presented in this paper as a function of $\theta(G)$ and k .

Acknowledgements. Partial support by AFOSR award FA9550-19-1-0161 is gratefully acknowledged. We would also like to thank the four anonymous referees for their insightful comments and suggestions.

References

- [1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Pearson, 1993.
- [2] B. Balasundaram, S. Butenko, and I. V. Hicks. Clique relaxations in social network analysis: The maximum k -plex problem. *Operations Research*, 59:133–142, 2011.
- [3] B. Balasundaram, S. S. Chandramouli, and S. Trukhanov. Approximation algorithms for finding and partitioning unit-disk graphs into co- k -plexes. *Optimization Letters*, 4:311–320, 2010.
- [4] R. W. Deming. Independence numbers of graphs—an extension of the König-Egerváry theorem. *Discrete Mathematics*, 27:23–33, 1979.
- [5] A. Dessmark, K. Jansen, and A. Lingas. The maximum k -dependent and f -dependent set problem. In *International Symposium on Algorithms and Computation*, pages 88–97. Springer, 1993.

- [6] H. Djidjev, O. Garrido, C. Levkopoulos, and A. Lingas. On the maximum k -dependent set problem. Technical Report LU-CS-TR:92-91, Department of Computer Science, Lund University, Sweden, 1992.
- [7] Z. Ertem, E. Lykhovyd, Y. Wang, and S. Butenko. The maximum independent union of cliques problem: complexity and exact approaches. *Journal of Global Optimization*, 76:545–562, 2020.
- [8] U. Feige and S. Kogan. The hardness of approximating hereditary properties. Technical report, Department of Computer Science and Applied Mathematics, Weizmann Institute, Rehovot, Israel, 2005.
- [9] A.M.H. Gerards. Matching. In M.O. Ball, editor, *Handbooks in OR and MS*, volume 7, chapter 3, pages 135–224. Elsevier, 1995.
- [10] F. Havet, R. J. Kang, and J. Sereni. Improper coloring of unit disk graphs. *Networks*, 54:150–164, 2009.
- [11] J. Hopcroft and R. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2:225–231, 1973.
- [12] S. Hosseini and S. Butenko. Polyhedral properties of the induced cluster subgraphs. *Discrete Applied Mathematics*, 297:80–96, 2021.
- [13] D. König. Gráfok és mátrixok. *Matematikai és Fizikai Lapok*, 38:116–119, 1931.
- [14] E. Korach, T. Nguyen, and B. Peis. Subgraph characterization of red/blue-split graph and König-Egerváry graphs. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 842–850. Cite-seer, 2006.
- [15] M. Kumar, S. Mishra, N. S. Devi, and S. Saurabh. Approximation algorithms for node deletion problems on bipartite graphs with finite forbidden subgraph characterization. *Theoretical Computer Science*, 526:90–96, 2014.
- [16] R. Lipton and R. Tarjan. A separator theorem for planar graphs. *SIAM Journal on Applied Mathematics*, 36:177–189, 1979.
- [17] B. McClosky and I. V. Hicks. The co-2-plex polytope and integral systems. *SIAM Journal on Discrete Mathematics*, 23:1135–1148, 2009.
- [18] M. Mucha and P. Sankowski. Maximum matchings via gaussian elimination. In *45th Annual IEEE Symposium on Foundations of Computer Science*, pages 248–255. IEEE, 2004.
- [19] R. Rizzi. A short proof of matching theorem. *Journal of Graph Theory*, 33:138–139, 2000.

- [20] F. Sterboul. A characterization of the graphs in which the transversal number equals the matching number. *Journal of Combinatorial Theory Series B*, 27:228–229, 1979.
- [21] V. V. Vazirani. *Approximation Algorithms*. Springer, 2003.

A scalable Markov chain framework for influence maximization in arbitrary networks

Juan S. Borrero, Majid Akhgar, and Pavlo A. Krokhmal

Abstract—We consider a Markov chain model of influence in a network where inactive nodes can be activated by their active in-neighbors or by external factors. The activation time is a random variable whose parameters, or *influence rates*, depend on control variables whose values can be changed by the decision-maker (DM) seeking to allocate her limited budget to minimize the time until all nodes are active. We show that the optimization of either the expectation or the tail probabilities of the network activation time (NAT) is an intractable (EXPTIME) problem. Thus, we focus on a stochastic upper bound on the NAT and show that both its expectation and tail probabilities are minimized when the Markov chain's minimum sojourn rate is maximized. We provide performance guarantees for the *network activation time optimization problem* (NATP) with respect to the optimization of the expectation problem and show that the NATP becomes an efficiently solvable convex optimization problem when we have concave influence rates and a convex cost function. We also provide an efficient cut generation algorithm, which can solve many synthetic and real-life instances with more than 10^4 nodes and 10^5 arcs in times ranging from a few seconds to a few minutes.

Index Terms—Influence maximization, Markov Chains, convex optimization, minimum cut, cutting planes



1 INTRODUCTION

SOME of the most interesting phenomena observed in both physical and social networks have to do with sweeping, or cascading, changes in the status of the network's nodes, such as the spread of viruses or infections, propagation of rumors, ideas, political preferences, adoption of new technology, etc. Many of such processes can be modeled under the general umbrella of *influence propagation* in networks, with a specific subclass of *influence maximization* problems that seek to maximize the rate of spread of influence or the number of influenced nodes. Influence maximization problems have applications in viral marketing, information diffusion in online social networks and in autonomous systems, among others [1]. Influence maximization models could also be used to identify the network elements whose deactivation or deletion minimizes the spread of the cascade, and in this sense they could also be used to find vaccination strategies in epidemic models [2].

Influence Maximization Problem and related models.

One of the most common influence optimization models is the Influence Maximization Problem (IMP) [3]. In its basic formulation, a decision-maker (DM) needs to change the influence status of $K > 0$ nodes (i.e.,

“activate” these nodes) in order to maximize the expected number of influenced (active) nodes at the end of the cascade. There are two modes of influence: in the threshold model, at a given stage an inactive node becomes active if some function of its active in-neighbors exceeds a uniformly distributed random threshold; in the independent cascade model, at any given stage a newly activated node has a single chance to activate its out-neighbors and each activation is successful with a fixed probability (which might be arc-dependent). A constant time approximation algorithm for these NP-hard problems was developed in [3] based on the optimization of submodular functions. The main difficulty, however, is that at each iteration the algorithm has to evaluate (or estimate) a complicated stochastic function; evaluating this function is a #P problem [4]. The IMP model has many modeling enhancements and approximate and heuristic solution techniques, see the surveys in [5] and [1].

Recently, integer optimization methods have been employed to solve the IMP. Wu and Küçükyavuz [6] frame the IMP models as two-stage stochastic programs and propose a sample average approximation cut generation algorithm based on stochastic programming techniques. Güney [7] provides an integer programming formulation of the independence cascade version of the IMP and discusses the merits of solving it using linear programming approximations. We note that integer optimization methods have also been applied in more complex influence models that have additional requirements, see for example [8].

Deterministic alternatives to the IMP have also been approached using integer programming. Chen [9] and Akckerman et al. [10] discuss the target-set selection problem, which is a deterministic version of the threshold model in [3]; see also Raghavan and Zhang [11] who

- J.S. Borrero is with the School of Industrial Engineering & Management, Oklahoma State University, Stillwater, OK, 74078.
E-mail: juan.s.borrero@okstate.edu.
- M. Akhgar is with the School of Industrial Engineering & Management, Oklahoma State University, Stillwater, OK, 74078.
E-mail: majid.akhgarfarsani@okstate.edu.
- P. A. Krokhmal is with the Department of Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, 85721.
E-mail: krokhmal@arizona.edu.

discuss improved integer programming formulations and branch and cut approaches to solve the more general weighted target-set selection problem. Fischetti et al. [12] consider a deterministic threshold model with general threshold functions where the objective is to find the least costly incentive strategy to activate a fixed proportion of nodes of the network. This model, which can be considered as a generalization of the weighted-target set selection problem, is formulated as two different integer programs and the authors provide cut generation methods to solve the resulting exponentially large formulations. Further developments on this model, including complexity results and polynomial-time algorithms for particular graphs, are discussed in [13] and solved by branch-and-cut techniques in [14]. Nannicini et al. [15] introduce a robust version of the linear threshold of the IMP where uncertainty affects the node thresholds. They provide mixed-integer linear formulations for the deterministic and robust versions with an exponential number of variables which are solved using column generation and branch-and-cut. We note that although the above models do not deal with probability distributions or sampling, they remain very challenging to solve. For instance, the basic target-set selection problem is approximable only up to a polylogarithmic factor on arbitrary networks [9]. Moreover, even with the state-of-the-art integer programming methods developed in [12] and [11], instances of the order of 10^3 and 10^4 nodes and arcs can take hours to solve and for many of such instances optimality cannot be guaranteed within three hours. With our approach, although the influence models are not directly comparable, we can solve similar and larger instances significantly faster.

We note that IMP-based models result in combinatorial optimization problems that seek to identify a set of nodes to activate (either directly or indirectly via incentives) so as to maximize the number of active nodes. While IMP-based approaches have advantages such as having only a few (types of) parameters or being readily adaptable to deterministic or stochastic versions, they also have modeling limitations. For instance, IMP-based models are largely concerned only about the activation of nodes. Thus, other parameters that control the propagation of influence, such as the probability that a node successfully activates other nodes, or the thresholds levels in threshold-based models, are not typically modifiable by the decision maker.

Moreover, there is no scalable exact method to solve IMP-based models and large instances are solved using heuristics, which do not provide theoretical optimality guarantees [16]. Indeed, for any given seed selection, evaluation of the expected number of active nodes at the end of the cascade is $\#P$ -hard [4]. Moreover, a high quality $(1 \pm \epsilon)$ -approximation with probability at least $(1 - \alpha)$ of this expected value requires at least $\Omega(\frac{n^2}{\epsilon^2} \ln 1/\alpha)$ samples [1]. Consequently, algorithms with performance guarantees do not scale. For example, in [6] the authors

employed a sampling average approximation method to approximate the optimal solution on instances of the *independent cascade* model of the IMP. For instances of the order of 10^4 nodes, their method took more than three hours using a rather small number of seeds ($K = 5$). Moreover, the solution times increase sharply in terms of K , which limits the applicability and scalability of their method to settings with a very low number of initial seeds.

Influence models based on a Markovian framework. Alternative approaches to influence maximization use a Markovian framework that is more common in epidemic models, see [17], [18]. Within this framework, the change of influence status of nodes in the network is driven by a Markov chain, such that activation of a node represents a transition of the Markov chain to a new state, and the likelihood and timing of such activations are determined by the corresponding transition rates of the stochastic process. Modeling the influence in terms of Markov processes is common in epidemic models such as the SIR (Susceptible-Infected-Recovered), or SIS (Susceptible-Infected-Susceptible), see [19], [2], [20], [21] and the references therein; see also the general control models in [22], [23], where explicit conditions and formulas are derived to control a broad class of stochastic systems under parametric uncertainty.

Influence problems based on Markov epidemic models typically simplify the dynamics of the influence diffusion via mean-field approximations [19] or dynamic message-passing equations [17] in order to find optimal solutions. While these models might also yield activation policies in continuous time, the extent to which the approximations adequately represent or bound the actual (non-approximated) activation times is not known in general.

The proposed influence framework. Although all the models discussed above are built upon simple and intuitive cascade dynamics, their numerical tractability generally presents significant challenges in even the most basic cases. This implies that exact optimal solutions, or even approximate solutions with optimality guarantees, are very difficult to find for arbitrary large scale and real-world networks, and, as discussed in [6], in practice one often has to resort to heuristic methods for scalability.

The framework presented in this paper seeks to address these limitations. The model follows the Markovian approach and focuses on allocating resources to modify the rates of a Markov influence process in order to minimize the network activation time (NAT), which is defined as the time it takes for all the nodes of the network to become active. This approach allows for more flexibility than the standard IMP, because the DM can control the extent to which each node and each arc are contributing to the propagation process. In our framework, in contrast to other Markovian models of influence, we do not simplify the dynamics of the Markov chain. Instead, we derive an upper bound for the NAT in

terms of the minimum sojourn rate of the Markov chain¹, derive an optimization problem based on the bound, and provide theoretical performance guarantees of the optimal value of the upper bound with respect to the minimum possible NAT.

Our proposed framework is built upon the model discussed in [18]. Here, we extend its modeling capabilities, its theoretical underpinnings, and its solution algorithms, to provide a more streamlined, comprehensive, and useful modeling framework for influence maximization in arbitrary networks. Specifically, in the influence maximization model of [18] nodes in an arbitrary network are initially inactive (uninfluenced) but can become active via the influence of active in-neighbors (referred to as *network influence*) or by the influence of external factors (referred to as *external influence*). These activation (influence) events are governed by a Markov chain, whose transition rates are linearly parametrized, such that the corresponding parameters can be modified, at certain costs, by the DM. The objective of the DM is to use her limited resources to adjust the parameters of influence/activation times distributions in order to minimize the NAT.

It turns out, however, that minimizing the expectation or the tail probabilities of the NAT is a highly intractable problem because it requires solving an exponentially large system of differential equations. Hence, instead of tackling this problem directly, in [18] the authors solve an optimization problem that minimizes an eigenvalue-based upper bound of the NAT. The resulting *cascading rate optimization problem* is shown to be equivalent to a generalized maximum flow problem in the network, or under certain assumptions, to a minimum spanning arborescence problem (MSA), and is solved in polynomial time using linear programming (LP) or MSA algorithms.

The Markov chain framework for influence maximization we propose in this paper extends the influence maximization model of [18] on several fronts. Particularly, our work makes the following specific contributions:

- (i) We generalize the model to handle arbitrary rate and cost functions with an arbitrary (but finite) number of control variables, and an arbitrary initial activation state. Under concavity and convexity assumptions we show that the minimization of the upper bound reduces to a convex optimization problem and hence it can be efficiently solved under general tractability conditions.
- (ii) By using a stochastic upper bound rather than an eigenvalue-based upper bound, we provide an alternative derivation of the upper-bounding technique for the NAT. The stochastic upper bound is a sum of iid copies of the minimum sojourn time of the Markov chain, and hence the resulting upper bound optimization problem can be framed as maximizing the minimum activation (transition) rate of the

Markov chain. In addition, we derive simple explicit expressions for the expectation and the distribution of the upper bound and provide explicit formulas for the expectation and the distribution of the NAT. These last two formulas require exponential time to evaluate and thus we show that the problems of minimizing either the expectation or the tail probabilities of the NAT are not even in *NP*.

- (iii) We show that the ratio between the exact optimal solution of the expected activation time and the solution resulting from minimizing the stochastic upper bound is upper-bounded by an $O(n\Delta)$ function, where n is the number of nodes of the network and Δ is the maximum in-degree of the network. Moreover, we show that under additional assumptions this upper bound can be tightened.
- (iv) We introduce a cutting plane algorithm to solve the upper bound optimization problem which makes calls to a minimum-cut separation routine. This algorithm is significantly faster than the generalized maximum flow approach and can solve real-life and synthetic instances of the order of 10^4 and 10^5 nodes and arcs in a matter of seconds to a few minutes under several different parameter configurations.

Modeling the propagation of *positive cascades*, such as ideas or useful information, through a population of people is one of the main applications of the proposed model. As a case in point, suppose that a company has already identified a group of customers with a high likelihood of participation in an upcoming auction. It is assumed that all potential customers will be informed; so, the company aims to inform them about the auction in the *minimum possible time*. To this end, the company can use two alternatives. In the first, which is considered as the *network influence*, it can offer incentives such as discounts, gift cards or even free products to customers who share the information about the auction with other targeted customers. In the second case, which is considered as the *external influence*, the company informs customers by public advertisements such as billboards, social media ads, among others. On the other hand, it can also decide to spend its budget on a combination of both options. Our model, therefore, can help the company decide how to assign its budget for each customer, namely whether the customer should be influenced using either public advertisement, viral-marketing, or a combination of both.

The rest of the paper is organized as follows. In Section 2 we present the Markov chain model of influence. Section 3 introduces the mathematical formulation of our model and discusses the tractability issues, the stochastic upper bounds, and the performance guarantees. Section 4 discusses the two algorithms to solve the upper bound optimization problem. Section 5 presents the results of our computational experiments and Section 6 contains the conclusions. The proofs not discussed in the main text can be found in Appendix A.

1. The sojourn time at a given state is the (exponentially distributed) time that it takes the Markov chain to leave the state. The sojourn rate at a given state is the rate of the corresponding sojourn time, see [24].

2 MARKOV CHAIN MODEL OF INFLUENCE PROPAGATION

Consider a simple directed network² $G = (N, A)$ with $n = |N|$ nodes and $m = |A|$ arcs. A given node can be either active (1) or inactive (0). Influence propagates in the network as follows: Node $k \in N$ can be activated via *network influence* by its active in-neighbors $\ell \in N^-(k)$, where $N^-(k) = \{\ell \in N : (\ell, k) \in A\}$ denotes the set of in-neighbors of k . The time it takes an active in-neighbor ℓ to activate node k is exponentially distributed with the rate $d^{\ell k}(\zeta)$, where $d^{\ell k}$ is a non-negative function of the control variables $\zeta = (\zeta_v : v \in \mathcal{V})$ and $\mathcal{V}$ is a finite set. An inactive node can also receive *external (non-network)* influence. The time it takes external influence to activate the inactive node $k \in N$ is exponentially distributed with the rate $g^k(\zeta)$, and g^k is a known non-negative function. We refer to these exponentially-distributed times as *influence times*.

Besides the non-negativity of the influence rates, we make the following assumptions about the influence model:

- A1.** All influence times are exponentially distributed.
- A2.** All influence times are statistically independent.
- A3.** Any active node simultaneously tries to activate all of its inactive neighbors.
- A4.** Before the DM takes any action, all the control variables take the value of zero, $\zeta = \mathbf{0}$, where $\mathbf{0}$ is a vector of zeros of the appropriate dimension. Moreover, all rate functions $g^k(\zeta)$ and $d^{\ell k}(\zeta)$ are assumed to be non-decreasing in ζ element-wise and finite for all $k \in N$ and $(\ell, k) \in A$.
- A5.** All nodes of the network can become active.

Assumption **A1** means that each influence time has the memoryless property [24]; that is, the distribution of the remaining time to activation does not depend on the time elapsed since the influence attempt began. This assumption is the continuous-time analogue of saying that the outcomes of successive attempts to activate a node are independent. Such assumptions are standard in epidemic models [21] and influence models (see e.g., the basic threshold model in [3]) and allow for an analytical treatment of the model. Assumption **A2** implies that the activation attempts from different in-neighbors and the external influence are independent of each other. This type of assumption, along with Assumption **A3**, are also standard in the influence propagation literature, see, for example, the independence cascade model in [3]. Assumption **A4** is technical and is made to normalize the role of the rate functions in our model. Finally, Assumption **A5** is also made to avoid trivial instances, because if a node cannot become active then the NAT is equal to infinity.

For any value of the control vector ζ , let $X(t, \zeta)$ be the activation status of the network at time t if the control

variables take the value ζ . For any node k , let $r_k = 0$ if k is inactive and $r_k = 1$ otherwise. The state space of $X(\zeta) = \{X(t, \zeta), t \geq 0\}$ is $\mathcal{S} = \{(r_1, \dots, r_n) : r_k \in \{0, 1\} \forall k \in N\} \subseteq \{0, 1\}^n$ and it can be shown that $X(\zeta)$ is a continuous-time Markov chain (CTMC) with an absorbing state $(1, \dots, 1)$, which we denote by $\mathbf{1}$ [18]. The generator matrix $Q(\zeta) = (q_{rr'}(\zeta) : r, r' \in \mathcal{S})$ of the CTMC $X(\zeta)$ is given by

$$q_{rr'}(\zeta) = \begin{cases} g^k(\zeta) + \sum_{(\ell, k) \in A} r_\ell d^{\ell k}(\zeta), & \text{if } r' = r + e_k \text{ and } r_k = 0 \\ & \text{for some } k \in N \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

where e_k is the n -dimensional unit vector with a one in the k -th entry. The diagonal elements of $Q(\zeta)$ are equal to the negative sum of the elements in the corresponding row, i.e.,

$$q_{rr}(\zeta) = - \sum_{r' \in \mathcal{S}, r' \neq r} q_{rr'}(\zeta). \quad (2)$$

$$= - \sum_{k \in N} (1 - r_k) \left(g^k(\zeta) + \sum_{(\ell, k) \in A} r_\ell d^{\ell k}(\zeta) \right) \quad (3)$$

To simplify the notation, hereafter we denote $q_r(\zeta) = |q_{rr}(\zeta)|$. We denote states as vectors for a more intuitive notation, instead of the more common integer labeling of states.

Observe that for any $t \geq 0$, if $X(t, \zeta) = r$ and $r_k = 0$, then the time it takes node k to become active is exponentially distributed with the rate $q_{r, r+e_k}(\zeta)$. Therefore, the expected activation time of node k from state r is $1/q_{r, r+e_k}(\zeta)$ and the variance is $1/(q_{r, r+e_k}(\zeta))^2$. Using the basic properties of the exponential distribution, it can be shown that the probability that node k becomes active before any other inactive node in r is given by $q_{r, r+e_k}(\zeta)/q_r(\zeta)$. Therefore, by modifying the values of the rates $g^k(\zeta)$ and $d^{\ell k}(\zeta)$ the DM explicitly controls the probabilistic behavior of the activation time of any node in the network.

Depending on the specific assumptions regarding the functions $g^k(\zeta)$ and $d^{\ell k}(\zeta)$, $k \in N$, $(\ell, k) \in A$, the outlined Markov chain model can represent continuous time and stochastic variations of different types of influence maximization problems:

- Target-set selection-like [3], [9], [17]: In these models, the DM has to select nodes to start a cascade in the network but she cannot modify the mechanism or the probabilities at which influence spreads between nodes. This class of problems can be modeled within the described Markov Chain framework by letting $d^{\ell k}(\zeta)$ be independent of ζ for all $(\ell, k) \in A$, and by letting $g^k(\mathbf{0}) = 0$, $g^k(\zeta) \geq 0$ for $\zeta \neq \mathbf{0}$, for all $k \in N$.
- Target-arc selection-like: Here, the DM has to select arcs to start a cascade in the network but she cannot modify the mechanism or the probabilities at which nodes receive external influence. This class of problems can be modeled by letting $g^k(\zeta)$ be

² A network is simple if it does not contain multiple arcs of the same orientation between the same pair of nodes, and if it does not have arcs with both endpoints at the same node.

independent of ζ for all $k \in N$, and by letting $d^{\ell k}(\mathbf{0}) = 0$, $d^{\ell k}(\zeta) \geq 0$ for $\zeta \neq \mathbf{0}$, for all $(\ell, k) \in A$.

- Network design for influence propagation [18]: In this model it is assumed that there are no existing influence spread capabilities in the network. The objective is to determine what arcs should be used to spread influence and what nodes should be subject to external influence. This type of design problems can be modeled by letting the functions $g^k(\zeta)$ and $d^{\ell k}(\zeta)$ satisfy $g^k(\mathbf{0}) = d^{\ell k}(\mathbf{0}) = 0$, and $g^k(\zeta), d^{\ell k}(\zeta) \geq 0$ for $\zeta \neq \mathbf{0}$, for all $k \in N$ and $(\ell, k) \in A$.

Using the CTMC described in this section, we next formally state the optimization problem we aim to solve in this paper. For convenience, the list of notations used therein is presented in Table 12 in Appendix D.

3 THE NETWORK ACTIVATION TIME MINIMIZATION PROBLEM

At the initial time $t = 0$ the network is in a given state $s \in \mathcal{S}$ and the DM has to determine what values to assign to the control variables in ζ to minimize the time until all the nodes are active. We make the assumption that the cost of assigning a value of ζ is given by $c(\zeta)$, where c is a known non-negative and component-wise non-decreasing cost function. The DM cannot increase the values of ζ indefinitely, because she has a total budget of B ; without loss of generality, we assume $B = 1$.

Let $T_s(\zeta)$ denote the network activation time (NAT), i.e., the time it takes for all nodes to become active given that the initial state is s and that the control variables take the value ζ . In other words, $T_s(\zeta)$ is the time it takes state 1 to absorb the chain $X(\zeta)$ given that $X(0, \zeta) = s$. Observe that because $T_s(\zeta)$ is a random variable and the DM can only control the parameters associated with the probabilistic distributions underlying $T_s(\zeta)$, the minimization of $T_s(\zeta)$ must be formulated in terms of deterministic equivalents related to $T_s(\zeta)$. For instance, the DM could try to find the value of ζ that minimizes the tail probabilities $P[T_s(\zeta) > t]$ for a given t or could try to find the value of ζ that minimizes the expectation $E[T_s(\zeta)]$.

Unfortunately, as shown next, general analytical expressions for the cumulative distribution function (cdf) of $T_s(\zeta)$ or the expectation $E[T_s(\zeta)]$ involve sums with exponentially many terms. That is, the problems of minimizing $P[T_s(\zeta) > t]$ or minimizing $E[T_s(\zeta)]$ are not even in the class NP. Consequently, an efficient optimization approach that directly relies on an explicit expression for distribution of $T_s(\zeta)$ is unlikely to exist.

3.1 Optimizing the distribution or expectation of the NAT is hard

In this section, for simplicity, we omit ζ from the notation and only include the dependence on ζ when necessary. For any $\mathbf{r} \in \mathcal{S}$ we define $\tilde{\mathcal{R}}(\mathbf{r})$ as the set of non-absorbing states reachable from $\mathbf{r}$ after one transition and $\tilde{\mathcal{E}}(\mathbf{r})$ as

the set of non-absorbing states reachable from $\mathbf{r}$ after any number of transitions. Formally,

$$\tilde{\mathcal{R}}(\mathbf{r}) = \{\mathbf{r}' \in \mathcal{S} \setminus \{\mathbf{1}\} : q_{\mathbf{r}\mathbf{r}'} > 0\} \text{ and} \quad (4)$$

$$\tilde{\mathcal{E}}(\mathbf{r}) = \{\mathbf{r}' \in \mathcal{S} \setminus \{\mathbf{r}, \mathbf{1}\} : p_{\mathbf{r}\mathbf{r}'}(t) > 0 \text{ for some } t \geq 0\}. \quad (5)$$

In the previous equation, $p_{\mathbf{r}\mathbf{r}'}(t)$ denotes the probability that the CTMC is in state $\mathbf{r}'$ after t time units given that it starts from $\mathbf{r}$: $p_{\mathbf{r}\mathbf{r}'}(t) = P[X(t) = \mathbf{r}' | X(0) = \mathbf{r}]$ for all $\mathbf{r}, \mathbf{r}' \in \mathcal{S}$ and $t \geq 0$. Note that the above definitions imply that $\tilde{\mathcal{R}}(\mathbf{r}) \subseteq \tilde{\mathcal{E}}(\mathbf{r})$ and that $\mathbf{r} \notin \tilde{\mathcal{E}}(\mathbf{r})$. Finally, we define $n(\mathbf{r})$ as the number of state transitions that are required for the chain to be absorbed by 1 given that it started from state $\mathbf{r}$. From the dynamics of the CTMC it is readily seen that $n(\mathbf{r})$ is precisely the number of inactive nodes in $\mathbf{r}$, i.e., $n(\mathbf{r}) = \sum_{k \in N} (1 - r_k)$.

For any $t \geq 0$, $j \geq 1$, $i = 1, \dots, j-1$, and any sequence of j states $(\mathbf{r}^j, \mathbf{r}^{j-1}, \dots, \mathbf{r}^1)$ such that $\mathbf{r}^{\ell-1} \in \tilde{\mathcal{R}}(\mathbf{r}^\ell)$ for all $\ell = 2, \dots, j$, define

$$L((\mathbf{r}^j, \dots, \mathbf{r}^i), t) = \int_0^t e^{-q_j u} L((\mathbf{r}^{j-1}, \dots, \mathbf{r}^i), t-u) du \quad (6)$$

where for a single-state sequence $(\mathbf{r}^j) = \mathbf{r}^j$ we define

$$L(\mathbf{r}^j, t) = e^{-q_j t} \quad \forall t \geq 0. \quad (7)$$

In the above equations, we write $q_j = q_{\mathbf{r}^j}$ for notational simplicity.

Proposition 1. Let $j = 1, \dots, n(s)$ be given and let $\mathbf{r}^j \in \mathcal{S}$ be any state such that $n(\mathbf{r}^j) = j$. Then,

$$P[T_{\mathbf{r}^j} > t] = L(\mathbf{r}^j, t) + \sum_{\mathbf{r}^{j-1} \in \tilde{\mathcal{R}}(\mathbf{r}^j)} q_{j,j-1} L((\mathbf{r}^j, \mathbf{r}^{j-1}), t) \quad (8a)$$

$$+ \sum_{\mathbf{r}^{j-1} \in \tilde{\mathcal{R}}(\mathbf{r}^j)} \sum_{\mathbf{r}^{j-2} \in \tilde{\mathcal{R}}(\mathbf{r}^{j-1})} q_{j,j-1} q_{j-1,j-2} L((\mathbf{r}^j, \mathbf{r}^{j-1}, \mathbf{r}^{j-2}), t) \quad (8b)$$

⋮

$$+ \sum_{\mathbf{r}^{j-1} \in \tilde{\mathcal{R}}(\mathbf{r}^j)} \dots \sum_{\mathbf{r}^1 \in \tilde{\mathcal{R}}(\mathbf{r}^2)} \prod_{i=2}^j q_{i,i-1} L((\mathbf{r}^j, \mathbf{r}^{j-1}, \dots, \mathbf{r}^1), t), \quad (8c)$$

where for simplicity $q_{i,i-1} = q_{\mathbf{r}^i, \mathbf{r}^{i-1}}$ for any $i \geq 1$. In particular, if $j = 1$ then $\tilde{\mathcal{R}}(\mathbf{r}^1) = \emptyset$ and the right-hand side of above formula reduces to $e^{-q_1 t}$.

Remark 1. We note that expression (8) involves a number of summands that is exponential in terms of j . Indeed, consider computing $L((\mathbf{r}^j, \dots, \mathbf{r}^i), t)$ and assume that $(\mathbf{r}^j, \dots, \mathbf{r}^i)$ has p states. Then $L((\mathbf{r}^j, \dots, \mathbf{r}^i), t)$ corresponds to $p-1$ nested integrals. It is possible to analytically compute this iterated integral, in particular it is the sum of 2^{p-1} (potentially different) terms of the form

$$\prod_{(u,v) \in R_p} \frac{1}{q_u - q_v} e^{-q_i t}$$

for some $i = 1, \dots, j$ and for some subset R_p of $p-1$ pairs such that each pair (u, v) satisfies that $u > v$. We note that the above summands can also have negative signs in front of them.

As a consequence, (8c) involves computing $j!2^{j-1}$ potentially different summands in the worst case (i.e., in settings where all the rate functions are different and their products and differences do not cancel out).

Expression (8) also suggests that $E[T_{r^j}]$ contains a number of summands which is exponential in terms of j . This assertion follows from the fact that $E[T_{r^j}] = \int_0^\infty P[T_{r^j} > t] dt$ and from the observations given above about $L((r^j, \dots, r^j), t)$. There are, however, alternative methods to compute the expectation that does not involve the cdf, as given below.

Proposition 2. Let $j = 1, \dots, n(s)$ and let $r^j \in \mathcal{S}$ be any state such that $n(r^j) = j$. Denote $u_{r^j} = E[T_{r^j}]$. Then u_{r^j} is given by

$$u_{r^j} = \frac{1}{q_j} + \sum_{r^{j-1} \in \tilde{\mathcal{R}}(r^j)} \frac{q_{j,j-1}}{q_j q_{j-1}} \quad (9a)$$

$$+ \sum_{r^{j-1} \in \tilde{\mathcal{R}}(r^j)} \sum_{r^{j-2} \in \tilde{\mathcal{R}}(r^{j-1})} \frac{q_{j,j-1} q_{j-1,j-2}}{q_j q_{j-1} q_{j-2}} \quad (9b)$$

$$+ \dots \quad (9c)$$

$$+ \sum_{r^{j-1} \in \tilde{\mathcal{R}}(r^j)} \dots \sum_{r^1 \in \tilde{\mathcal{R}}(r^2)} \frac{\prod_{i=0}^{j-2} q_{j-i,j-i-1}}{\prod_{i=0}^{j-1} q_{j-i}}, \quad (9d)$$

where for simplicity $q_{i,i-1} = q_{r_i, r_{i-1}}$ for any $i \geq 1$.

We close this section by noting that CTMC theory offers other methods to compute the cdf and the expectation of T_s . For instance, one could compute the cdf by $P[T_s > t] = \sum_{r \in \mathcal{S}, r \neq 1} p_{sr}(t)$. Nevertheless, computing the probabilities $p_{sr}(t)$ would require solving a system of exponentially many differential equations (see Equation (39) in Appendix A), or computing the matrix exponential e^{Qt} [24], [25]. Any such method would require at least an exponential number of operations in the worst case.

3.2 A stochastic upper bound for the NAT $T_s(\zeta)$

An alternative to optimizing the exact probabilistic quantities of $T_s(\zeta)$ is to minimize upper bounds on $P[T_s(\zeta) > t]$ or $E[T_s(\zeta)]$. In this paper we pursue such an approach by maximizing $m_s(\zeta)$, the minimum rate of the sojourn times of all the states reachable from s , including s itself, i.e.,

$$m_s(\zeta) = \min\{q_r(\zeta) : r \in \mathcal{E}(s)\}, \quad (10)$$

where we define $\mathcal{E}(s) = \tilde{\mathcal{E}}(s) \cup \{s\}$. In order to explain the rationale behind maximizing $m_s(\zeta)$ over all feasible $\zeta \in \mathcal{B}$, where $\mathcal{B}$ is the set of feasible values for the control variables,

$$\mathcal{B} = \{\zeta \in \mathbb{R}_+^{|\mathcal{V}|} : c(\zeta) \leq 1\}, \quad (11)$$

consider the following arguments. Fix $\zeta \in \mathcal{B}$ and let $U_s(\zeta)$ be the sojourn time of the chain in s , that is, the total time that the chain resides in state s before transitioning to any other state. From the standard theory

of CTMCs [24], it is known that $U_s(\zeta) \sim \exp(q_s(\zeta))$ and that the random variables in the sequence $\{U_s(\zeta) : s \in \mathcal{S}\}$ are independent. Observe that

$$T_s(\zeta) = \sum_{i=0}^{|n(s)|-1} U_{r_i}(\zeta), \quad (12)$$

where $r_i \in \mathcal{E}(s)$ is the state of the chain after the i -th transition from s , in particular $r_0 = s$ and $n(r_i) = n(s) - i$. On the other hand, let $\{V_i(\zeta) : i = 0, \dots, n(s) - 1\}$ be a sequence of independent and identically distributed (iid) $\exp(m_s(\zeta))$ random variables. Because $U_{r_i}(\zeta) \sim \exp(q_{r_i}(\zeta))$ for any r_i and $r_i \in \mathcal{E}(s)$, it follows that

$$U_{r_i}(\zeta) \preceq_{st} V_i(\zeta) \quad \forall i = 0, \dots, n(s) - 1, \quad (13)$$

where $\preceq_{st}$ denotes the first order stochastic dominance [26]. Equations (12) and (13) then imply that

$$T_s(\zeta) \preceq_{st} \sum_{i=0}^{|n(s)|-1} V_i(\zeta) := W_s(\zeta), \quad (14)$$

which in turn implies that

$$P[T_s(\zeta) > t] \leq P[W_s(\zeta) > t] \text{ for all } t \geq 0 \text{ and} \\ E[T_s(\zeta)] \leq E[W_s(\zeta)], \quad (15)$$

see [26]. Importantly, as $W_s(\zeta)$ is a sum of $n(s)$ iid $\exp(m_s(\zeta))$ random variables, $W_s(\zeta)$ has an Erlang distribution with parameters $n(s)$ and $m_s(\zeta)$. Consequently, we have proven the following result:

Proposition 3. Suppose $s \in \mathcal{S}$ and $\zeta \in \mathcal{B}$ are given. Let $m_s(\zeta) = \min\{q_r(\zeta) : r \in \mathcal{E}(s)\}$. Then $T_s(\zeta) \preceq_{st} W_s(\zeta)$ where $W_s(\zeta)$ is an Erlang distributed random variable with parameters $n(s)$ and $m_s(\zeta)$. Therefore,

$$P[T_s(\zeta) > t] \leq e^{-m_s(\zeta)t} \sum_{j=0}^{|n(s)|-1} \frac{(m_s(\zeta)t)^j}{j!} \text{ and} \quad (16)$$

$$E[T_s(\zeta)] \leq \frac{n(s)}{m_s(\zeta)}.$$

Observe that the expressions on the right-hand sides of both inequalities in (16) are strictly decreasing in terms of the value of $m_s(\zeta)$, hence both of them are minimized when $m_s(\zeta)$ takes the maximum allowable value. In other words, the DM can minimize an upper bound for both $P[T_s(\zeta) > t]$ and $E[T_s(\zeta)]$ by maximizing the value of $m_s(\zeta)$ across all feasible values of the control variables ζ .

We note that it is possible to derive tighter bounds than those in inequalities (16) by other means (see, e.g., Equation (39) in Appendix A). By minimizing such tighter bounds, a larger decrease of $P[T_s(\zeta) > t]$ or $E[T_s(\zeta)]$ that does not necessarily maximize $m_s(\zeta)$ could be achieved. However, maximizing $m_s(\zeta)$ instead of other upper bounds has two advantages: (i) It leads to a simple class of optimization problems (as shown in Section 4) that can provide quality, or approximation, guarantees and (ii) by maximizing $m_s(\zeta)$ the DM is

hedging against the worst possible realization of uncertainty in the CTMC. Indeed, observe that the worst-case scenario from the stand point of the DM is that the chain lands on the state r such that $q_r(\zeta) = m_s(\zeta)$. Hence, by maximizing $m_s(\zeta)$ the decision-maker is guaranteeing the fastest possible propagation (in distribution) across all states independent of the realization.

As a consequence of the foregoing discussion, in this paper we attempt to minimize $T_s(\zeta)$ by minimizing an upper bound of $T_s(\zeta)$, which in turn is based on maximizing $m_s(\zeta)$ as defined in (10). Thus, if at time $t = 0$ the process is in state s , we define the problem of minimizing the upper bound, referred to hereafter as the *network activation time problem* (NATP), as finding an optimal solution to

$$z_s = \max\{m_s(\zeta) : \zeta \in \mathcal{B}\}. \quad (17)$$

In Section 4 we show an explicit mathematical programming formulation of this problem, discuss conditions that allow the problem to be solvable in polynomial time, and provide two solution algorithms. Next, we discuss how an optimal solution of the NATP performs, in the worst-case, with respect to the optimal value of the expected activation time problem.

3.3 Performance bounds on the value of the NATP

As a consequence of Propositions 1 and 2, the problem

$$\zeta^E \in \operatorname{argmin}\{E[T_s(\zeta)] : \zeta \in \mathcal{B}\}, \quad (18)$$

which we hereafter refer to as the *expected network activation problem* (E-NATP), is in the class of EXPTIME problems and therefore an efficient algorithm to solve it is unlikely to exist. In fact, the exponential growth is so severe that this problem is practically impossible to solve even for networks with a few dozens of nodes. A relevant theoretical question is thus, how different is $E[T_s(\zeta^*)]$, where ζ^* is an optimal solution of the NATP (17), from the exact optimal value $E[T_s(\zeta^E)]$. In order to develop an upper bound on the ratio of the two expectations, let $M_s(\zeta)$ be the largest sojourn rate across all states reachable from s , including s itself, thus

$$M_s(\zeta) = \max\{q_r(\zeta) : r \in \mathcal{E}(s)\}. \quad (19)$$

Similar arguments as the ones used in Proposition 3 show that $Y_s(\zeta) \leq_{st} T_s(\zeta)$, where $Y_s(\zeta)$ has an Erlang distribution with parameters $n(s)$ and $M_s(\zeta)$. Therefore, $E[Y_s(\zeta)] \leq E[T_s(\zeta)]$ for any $\zeta \in \mathcal{B}$ and we have that:

$$\begin{aligned} \frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]} &\leq \frac{E[W_s(\zeta^*)]}{E[T_s(\zeta^E)]} \leq \frac{E[W_s(\zeta^E)]}{E[T_s(\zeta^E)]} \\ &\leq \frac{E[W_s(\zeta^E)]}{E[Y_s(\zeta^E)]} \leq \frac{M_s(\zeta^E)}{m_s(\zeta^E)}, \end{aligned} \quad (20)$$

where the last inequality follows because $E[W_s(\zeta^E)] = n(s)/m_s(\zeta^E)$ and $E[Y_s(\zeta^E)] = n(s)/M_s(\zeta^E)$. From (20) we conclude that

$$\frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]} \leq \max\left\{\frac{M_s(\zeta)}{m_s(\zeta)} : \zeta \in \mathcal{B}\right\}, \quad (21)$$

and an upper bound on the ratio can be given by the maximum value of the ratio between $M_s(\zeta)$ and $m_s(\zeta)$ across all feasible values $\zeta \in \mathcal{B}$.

Proposition 4. Assume that the functions $g^k(\zeta)$ and $d^{\ell k}(\zeta)$, $k \in N$, $(\ell, k) \in A$, are bounded over $\mathcal{B}$; that neither the rate nor the cost functions depend on the size of the network; and suppose that an optimal solution in (21) exists. Let b be an upper bound on the value of the rate functions, thus $g^k(\zeta) \leq b$ and $d^{\ell k}(\zeta) \leq b$ for all $k \in N$, $(\ell, k) \in A$ and $\zeta \in \mathcal{B}$. Let $\Delta^-(s)$ and $\delta^-(s)$ be upper and lower bounds, respectively, on the number of in-neighbors of any node in the network inactive in s ; that is, $\delta^-(s) \leq |N^-(k)| \leq \Delta^-(s)$ for all $k \in N$ such that $s_k = 0$. Then, there exist $b' > 0$ such that

$$\frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]} \leq \frac{bn(s)(1 + \Delta^-(s))}{b'}. \quad (22)$$

Moreover, if in the optimal solution of (21) all rate functions in the denominator m_s are strictly greater than zero, then there exist $b' > 0$ such that

$$\frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]} \leq \frac{bn(s)(1 + \Delta^-(s))}{b'(1 + \delta^-(s))}. \quad (23)$$

In particular, equations (22) and (23) imply that the ratio is upper bounded by either a $O(n(s)\Delta^-(s))$ function or a $O\left(\frac{n(s)\Delta^-(s)}{1 + \delta^-(s)}\right)$ function.

Bounds for the eigenvalues of general matrices can also be used to provide bounds for the maximum in (21). To this end, let $Q_s(\zeta)$ be the restriction of the generator matrix $Q(\zeta)$ to the states in $\mathcal{E}(s)$. Observe that $Q_s(\zeta)$ is a $(2^{n(s)} - 1) \times (2^{n(s)} - 1)$ upper triangular matrix, and moreover, that $m_s(\zeta)$ and $M_s(\zeta)$ are the smallest and largest eigenvalues, respectively, of $-Q_s(\zeta)$. Therefore, $M_s(\zeta)/m_s(\zeta)$ is the ratio between the largest eigenvalue and the smallest eigenvalue of $-Q_s(\zeta)$. However, applying general results such as those in [27], [28] would yield unnecessarily loose upper bounds, i.e., of the order of $O(2^{n/2}n^{1/2})$. Only under very stringent conditions on the value of the rates, such results provide $O(1)$ upper bounds:

Proposition 5. Let $q(\zeta)$ be the diagonal of $-Q_s(\zeta)$ as a vector and let $\|x\|_p$ denote the p -norm of x . Suppose that for all $\zeta \in \mathcal{B}$ we have that

$$\|q(\zeta)\|_1^2 \geq \frac{(2^{n(s)} - 1)^2}{2^{n(s)}} \|q(\zeta)\|_2^2. \quad (24)$$

If neither the rate nor the cost functions depend on the size of the network, then $\frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]}$ is bounded from above by an $O(1)$ function, i.e., a function that does not depend on the size of the network.

The applicability of Proposition 5 is limited due to the assumptions on the norm of the diagonal in (24). In fact, from basic properties of the 1- and 2-norm we have that $\|q(\zeta)\|_1^2 \leq (2^{n(s)} - 1)\|q(\zeta)\|_2^2$ where equality holds if and only if all the entries in $q(\zeta)$ are equal. Because the difference between $(2^{n(s)} - 1)$ and $\frac{(2^{n(s)} - 1)^2}{2^{n(s)}}$ is $1 - 2^{-n(s)}$,

then the above observations imply that condition (24) would hold only if all the entries in $q(\zeta)$ are virtually the same.

4 SOLUTION METHODS FOR THE NATP

In this section we present two methods to solve problem (17) and discuss their main features. The first method is based on the derivation of a polynomially-sized mathematical programming formulation while the second is a cut generation algorithm. Under concavity and convexity assumptions on the functions defining the rates and the costs of the model, respectively, we show that the problem becomes a *convex optimization problem* and hence is efficiently solvable under mild additional assumptions. The algorithms are based on an observation that (17) can be reformulated as the following mathematical programming problem

$$z_s = \max \lambda \quad (25a)$$

$$\text{s. t. } \lambda \leq \sum_{k \in N} (1 - r_k) \left(g^k(\zeta) + \sum_{(\ell, k) \in E} r_\ell d^{\ell k}(\zeta) \right) \quad \forall r \in \mathcal{E}(s) \quad (25b)$$

$$c(\zeta) \leq 1 \quad (25c)$$

$$\zeta \in \mathbb{R}_+^{|\mathcal{V}|}. \quad (25d)$$

Note that this formulation has $|\mathcal{V}|+1$ continuous variables and $2^{n(s)}$ constraints. Because the formulation has an exponential number of constraints, it cannot be solved directly by an off-the-shelf solver if the network is sufficiently large.

4.1 A polynomially-sized formulation

For any state $r \in \mathcal{S}$ define $J_r \subseteq N$ as the set of active nodes in state r , i.e.,

$$J_r = \{k \in N : r_k = 1\}. \quad (26)$$

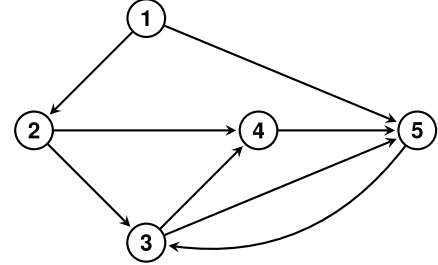
Let $G_s = (N_s, A_s)$ be an auxiliary weighted network based on G where all active nodes in s collapse to a single node that we denote by R_s . The set A_s consists of the edges in A that have no active endpoints in s plus new edges from R_s to all the inactive nodes in s . That is, $N_s = \{R_s\} \cup (N \setminus J_s)$ and $A_s = \{(R_s, k) : k \in N \setminus J_s\} \cup \tilde{A}_s$, where

$$\tilde{A}_s = A \cap [(N \setminus J_s) \times (N \setminus J_s)]. \quad (27)$$

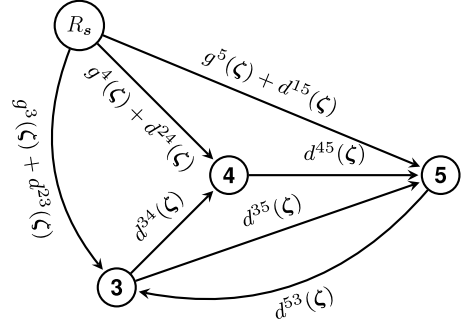
The weights of G_s depend on the value of ζ and are defined by

$$w_s((u, v), \zeta) = \begin{cases} g^k(\zeta) + \sum_{(\ell, k) \in A, \ell \in J_s} d^{\ell k}(\zeta), & \text{if } (u, v) = (R_s, k) \\ & \text{for some } k \in N \\ d^{\ell k}(\zeta), & \text{if } (u, v) = (\ell, k) \\ & \text{for some } (\ell, k) \in \tilde{A}_s. \end{cases} \quad (28)$$

An example of a network and the corresponding auxiliary graph is shown in Figure 1.



(a) The original network G



(b) The auxiliary network G_s

Fig. 1. An example of an auxiliary network with $n = 5$ and $s = (1, 1, 0, 0, 0)$.

For any $J \subseteq N_s$ define $C_s(J)$ as the cut induced on A_s by J , thus

$$C_s(J) = \{(u, v) \in A_s : u \in J, v \in N_s \setminus J\}, \quad (29)$$

and let $w_s(J, \zeta)$ be the weight of the cut

$$w_s(J, \zeta) = \sum_{(u, v) \in C_s(J)} w_s((u, v), \zeta). \quad (30)$$

Finally, denote by $\mathcal{J}_s$ the class of proper subsets of N_s that contain R_s , i.e.,

$$\mathcal{J}_s = \{J \subset N_s : R_s \in J\}. \quad (31)$$

We have the following relationship between the cuts induced by the elements of $\mathcal{J}_s$ and problem (17); its proof follows by applying the definitions. The result implies that problem (17) can be viewed as finding a value of the control variables $\zeta \in \mathcal{B}$ that maximizes the minimum cut in A_s across all cuts defined by elements of $\mathcal{J}_s$.

Proposition 6. *Let $s \in \mathcal{S}$ be given and consider the auxiliary network G_s as defined above. Then, the NATP (17) can be written as*

$$z_s = \max \{ \min \{ w_s(J, \zeta) : J \in \mathcal{J}_s \} : \zeta \in \mathcal{B} \}. \quad (32)$$

Formulation (32) allow us to reframe the problem as a single level generalized max-flow problem. Indeed, for any $k \in N_s \setminus \{R_s\}$, define $\mathcal{J}_s^k$ as the class of subsets of $\mathcal{J}_s$ that do not contain k , i.e.,

$$\mathcal{J}_s^k = \{J \in \mathcal{J}_s : k \notin J\}.$$

Because $\mathcal{J}_s = \bigcup_{k \in N_s \setminus \{R_s\}} \mathcal{J}_s^k$ then for the inner min-cut problem in Equation (32) we have that

$$\begin{aligned} & \min\{w_s(J, \zeta): J \in \mathcal{J}_s\} \\ &= \min \left\{ \min\{w_s(J, \zeta): J \in \mathcal{J}_s^k\}: k \in N_s \setminus \{R_s\} \right\} \\ &= \max \left\{ f: f \leq \min\{w_s(J, \zeta): J \in \mathcal{J}_s^k\} \right. \\ & \quad \left. \forall k \in N_s \setminus \{R_s\}, f \in \mathbb{R} \right\}. \end{aligned}$$

Observe that $\min\{w_s(J, \zeta): J \in \mathcal{J}_s^k\}$ is precisely the value of the minimum $R_s - k$ cut in G_s . Therefore, the max-flow min-cut Theorem [29] implies that $\min\{w_s(J, \zeta): J \in \mathcal{J}_s^k\} = \phi_{sk}(\zeta)$, where $\phi_{sk}(\zeta)$ is the maximum flow that can be sent from R_s to k in G_s assuming that the capacities of the arcs are given by their weights $w_s((u, v), \zeta)$. Hence, (32) becomes:

$$z_s = \max f \quad (33a)$$

$$\text{s. t. } f \leq \phi_{sk}(\zeta) \quad \forall k \in N_s \setminus \{R_s\} \quad (33b)$$

$$c(\zeta) \leq 1 \quad (33c)$$

$$f \in \mathbb{R}, \zeta \in \mathbb{R}^{|V|}. \quad (33d)$$

Formulation (33) implies that the solution of the NATP aims at finding the edge capacities of G_s that maximize the minimum maximum flow that R_s can send to all the nodes in $N_s \setminus R_s$. In addition, formulation (33) allows us to derive the desired polynomially sized formulation in terms of the parameters of the problem, as shown in the next proposition.

Proposition 7. *Let $s \in \mathcal{S}$ be given. Then, the NATP problem (17) has the following mathematical programming formulation:*

$$z_s = \max_{x, f, \zeta} f \quad (34a)$$

$$\text{s. t. } f - \sum_{(R_s, v) \in A_s} x_{R_s v}^k \leq 0 \quad \forall k \in N_s \setminus \{R_s\} \quad (34b)$$

$$\sum_{(u, v) \in A_s} x_{uv}^k - \sum_{(v, u) \in A_s} x_{vu}^k \leq 0 \quad (34c)$$

$$\forall v \in N_s, k \in N_s \setminus \{R_s\}$$

$$x_{R_s v}^k - g^v(\zeta) \leq 0 \quad \forall v, k \in N_s \setminus \{R_s\} \quad (34d)$$

$$x_{uv}^k - d^{uv}(\zeta) \leq 0 \quad \forall (u, v) \in \tilde{A}_s, k \in N_s \setminus \{R_s\} \quad (34e)$$

$$c(\zeta) \leq 1 \quad (34f)$$

$$x_{uv}^k \geq 0 \quad \forall (u, v) \in A_s, k \in N_s \setminus \{R_s\} \quad (34g)$$

$$\zeta \in \mathbb{R}^{|V|}, f \in \mathbb{R}.$$

In formulation (34), constraints (34d)–(34f) are nonlinear, thus (34) is not polynomially solvable across all type of rate and cost functions. However, it is clear that if $g^k(\zeta)$ and $d^{\ell k}(\zeta)$ are concave functions of ζ and $c(\zeta)$ is convex, then (34) can be written as $-\min\{-f: f_i(\zeta) \leq 0 \ i = 1, \dots, M\}$ where M is polynomial in the size of the network, f_i , $i = 1, \dots, M$, are convex functions and $\hat{\zeta} = (f, x, \zeta)$. These observations are summarized as follows:

Proposition 8. *Let $s \in \mathcal{S}$ be given, assume that $g^k(\zeta)$ and $d^{\ell k}(\zeta)$ are concave functions of ζ for all $k \in N_s \setminus \{R_s\}$ and $(\ell, k) \in E_s$, and that $c(\zeta)$ is a convex function of ζ . Then problem (34) is a convex optimization problem.*

As a consequence of Proposition 8, we have that the NATP (17) can be solved efficiently in polynomial time within a given precision for a broad class of concave rate functions and convex cost functions, see e.g., [30], [31]. In particular, if the rate and cost functions are linear then (17) becomes a linear program (LP) which can be solved exactly in polynomial time.

To conclude this section, we note that an additional consequence of the introduction of the auxiliary graph G_s is that the performance upper bounds discussed in Section 3.3 can be given in terms of the max-cuts and min-cuts of the graph G_s . Indeed, observe that

$$\frac{E[T_s(\zeta^*)]}{E[T_s(\zeta^E)]} \leq \max \left\{ \frac{\max\{w_s(J, \zeta): J \in \mathcal{J}_s\}}{\min\{w_s(J, \zeta): J \in \mathcal{J}_s\}}: \zeta \in \mathcal{B} \right\}. \quad (35)$$

An interesting avenue of further research is to improve the bounds for the ratio between $E[T_s(\zeta^*)]$ and $E[T_s(\zeta^E)]$ for specific type of networks and rate functions based on (35). For instance, simple calculations show that if all rate functions are positive such that g^k and $d^{\ell k}$ depend on neither k nor (ℓ, k) , respectively, and if G is a complete graph, then $E[T_s(\zeta^*)]/E[T_s(\zeta^E)]$ is a $O(n)$ function.

4.2 A cut generation algorithm

While formulation (34) establishes that the NATP can be solved fast in theory for a broad class of rate and cost functions, in practice this *monolithic* formulation does not scale adequately because it requires a quadratic number of variables and constraints in terms of the size of the network. This section discusses an alternative cut generation method to solve problem (17) that does not require the introduction of additional variables or constraints. At iteration $i \geq 0$ the algorithm keeps a subset of states $\mathcal{E}^i(s) \subseteq \mathcal{E}(s)$ and solves a relaxation of (25) that only considers the constraints associated with the states in $\mathcal{E}^i(s)$. The value of the relaxation is denoted by z_s^i , i.e.,

$$z_s^i = \max \left\{ \lambda: \lambda - q_r(\zeta) \leq 0 \quad \forall r \in \mathcal{E}^i(s), \zeta \in \mathcal{B} \right\}. \quad (36)$$

Suppose that ζ^i is an optimal solution of (36) and consider the problem $p_s(\zeta^i)$ given by

$$p_s(\zeta^i) = \min\{q_r(\zeta^i): r \in \mathcal{E}(s)\}. \quad (37)$$

If $z_s^i \leq p_s(\zeta^i)$, then ζ^i is feasible in (17) and therefore z_s^i and ζ^i are the optimal value and an optimal solution, respectively, of (17). Else, if $z_s^i > p_s(\zeta^i)$, then there exists a state $\hat{s} \in \mathcal{E}(s) \setminus \mathcal{E}^i(s)$ such that $z_s^i > q_{\hat{s}}(\zeta^i)$. Constraint $\lambda - q_{\hat{s}}(\zeta) \leq 0$ is then added to the master problem (36) and the algorithm repeats the above steps at iteration $i + 1$. The pseudocode of the cut generation algorithm is given in Algorithm 1. Note that this algorithm converges

to the optimal solution in a finite number of iterations because $\mathcal{E}(s)$ is finite and at each iteration a new constraint of (17) is added to the master.

Algorithm 1: A cut generation algorithm to find z_s .

Input : Node set N , arc set A , state s , initial set of states $\mathcal{E}^0 \subseteq \mathcal{E}(s)$
Output: An optimal solution ζ^* to problem z_s

```

1 Set  $i = 0$ ,  $z_s^0 = 0$ ,  $p_s(\zeta^0) = -\infty$ ;
2 while  $z_s^i > p_s(\zeta^i)$  do
3   Solve the master problem
      $z_s^i = \max \left\{ \lambda : \lambda - q_r(\zeta) \leq 0 \ \forall r \in \mathcal{E}^i, \ \zeta \in \mathcal{B} \right\}$ 
     and let  $\zeta^i$  be an optimal solution;
4   Solve the subproblem
      $p_s(\zeta^i) = \min \{ q_r(\zeta^i) : r \in \mathcal{E}(s) \}$ ;
5   if  $z_s^i > p_s(\zeta^i)$  then
6     Find  $\hat{s} \in \mathcal{E}(s) \setminus \mathcal{E}^i$  such that  $z_s^i > q_{\hat{s}}(\zeta^i)$ ;
7     Add the constraint  $\lambda - q_{\hat{s}}(\zeta) \leq 0$  to the
       master problem;
8     Put  $\mathcal{E}^{i+1} = \mathcal{E}^i \cup \{\hat{s}\}$  and  $i = i + 1$ ;
9   end
10 end
11 return  $\zeta^i$  and  $z_s^i$ 

```

While the solvability of the master problem depends on the assumptions that are made with respect to the rate and cost functions, problem $p_s(\zeta)$ is always a minimum cut problem. Indeed, following Section 4.1, this problem can be written as follows in terms of the network G_s :

$$p_s(\zeta^i) = \min \{ w_s(J, \zeta^i) : J \in \mathcal{J}_s \}. \quad (38)$$

That is, $p_s(\zeta^i)$ is the value of the minimum cut in network G_s under the edge weights induced by ζ^i , with the constraint that the root node R_s has to be on the source side of the cut. This type of constrained minimum cuts are sometimes called rooted-out minimum cuts.

The rooted-out minimum cut problem can be solved in strongly polynomial time independent of the rate functions. In the computational experiments in Section 5 we consider two approaches: within the first approach, following Equation (33a), we solve $n(s)$ different source-sink min cut problems by selecting R_s as the source and letting each of the inactive nodes in state s be the sink. This approach requires applying a source-sink min cut/max flow algorithm such as the Goldberg-Tarjan preflow push-relabel algorithm [32] $n(s)$ times, which would imply a time complexity of $O(n(s)^2 m(s) \log(n(s)^2/m(s)))$, where $m(s) = |E_s|$ is the number of edges in G_s . Alternatively, we solve (38) directly by employing the Hao-Orlin algorithm [33], which is designed to solve unrestricted min cut problems and would imply a time complexity of $O(n(s)m(s) \log(n(s)^2/m(s)))$.

A relevant implementation issue for Algorithm 1 concerns the set of states $\mathcal{E}^0$ that are used in the first iteration for the master problem. It is desirable for $\mathcal{E}^0$ to have few

but *dominating states*, where we say that r dominates r' if and only if $q_r(\zeta) \leq q_{r'}(\zeta)$ for all $\zeta \in \mathcal{B}$. Clearly, if there exists a set of states $\mathcal{E}^*(s)$ that dominates all other states in $\mathcal{E}(s)$, then we could compute z_s by replacing $\mathcal{E}(s)$ by $\mathcal{E}^*(s)$ in (25). In general, it is not straightforward to determine such a set $\mathcal{E}^*(s)$; however, if additional assumptions are made about the network topology and the rate functions, it is possible to derive the following results:

Proposition 9. Suppose that G is a complete network, that $g^k(\zeta) = g(\zeta)$ for all $k \in N$, and that $d^{\ell k}(\zeta) = d(\zeta)$ for all $(\ell, k) \in E$. If $s \neq \mathbf{0}$ then $\mathcal{E}^*(s) = \{r \in \mathcal{S} : n(r) = 1\}$. If $s = \mathbf{0}$ and $g(\zeta) \leq d(\zeta)$ for all $\zeta \in \mathcal{B}$ then $\mathcal{E}^*(\mathbf{0}) = \{\mathbf{0}\}$; else, if $g(\zeta) > d(\zeta)$ for all $\zeta \in \mathcal{B}$ then $\mathcal{E}^*(\mathbf{0}) = \{r \in \mathcal{S} : n(r) = 1\}$.

For complete networks with homogeneous rate functions, Proposition 9 implies that all states are dominated either by states in which all but one node have become active, or by s in case that $s = \mathbf{0}$ (i.e., if initially all nodes inactive). This result, and particularly its proof, gives insights into what might happen in general instances because these sets represent two extremes of the influence process. On the one hand, s is the state with the most inactive nodes in $\mathcal{E}(s)$ and thus is the one that receives the largest rate of external influence. On the other, states r such that $n(r) = 1$ are subject to the largest rates of network influence across $\mathcal{E}(s)$. Hence, if the external influence rates tend to be smaller than the network influence rates, it is likely that s dominates a large number of states; conversely, if the network influence rates tend to be smaller than the external influence rates, it is likely that the states with $n(r) = 1$ dominate many states in $\mathcal{E}(s)$. The above observations motivate us to define $\mathcal{E}^0 = \{s\} \cup \{r : n(r) = 1\}$ in Algorithm 1.

We close this section by noting that Algorithm 1 might require to add many cuts in the worst case. Each iteration, however, is computationally cheap as long as the rate functions are concave and the cost function is convex. In addition, as mentioned above, independent of the rates functions, the separation problem is a rooted-out minimum cut problem, which can be solved in strongly polynomial time.

5 COMPUTATIONAL EXPERIMENTS

We consider three types of experiments with different objectives. In Section 5.1 we consider small instances and compare the expected activation time using the optimal solution of (17) with the minimizer of the expected time to activation $E[T_s(\zeta^E)]$, and with other benchmark policies. Our objective is to determine how different are the values of these problems. Next, in Section 5.2 we analyze small instances under different assumptions on the functional form of the rates of influence to observe if there are structural changes in the optimal solution of the NATP. Finally, in Section 5.3 we compute the optimal solution of our model in large scale instances to illustrate the scalability of our approach.

5.1 The NATP solution vs the E-NATP solution

In this section we compare the expected activation time $E[T_s(\zeta^*)]$ obtained by the solution ζ^* of the NATP (17) with the expected activation $E[T_s(\zeta^E)]$ given by the solution of E-NATP (18). As an additional benchmark, we compute the expected activation time using a policy that does nothing (i.e., that does not invest to increase the influence rates thus $\zeta = 0$) and a policy that invest the budget uniformly across all control variables. We denote this last solution by ζ^U . Recall that problem (18) is not in *NP*, see Proposition 2. Therefore, problem (18) is practically impossible to solve exactly for real-sized instances and is only solvable on very small networks. The instances in this section and in Section 5.2 were solved using Matlab 2019b [34] on a 64-bit Windows 10 Pro laptop with 8GB of RAM and an eighth-generation i7 1.8GHz processor.

We assume that the rate functions are given by the following models:

- Linear model [18]: There are $|N|+|A|$ control variables θ^k , $k \in N$, and $\eta^{\ell k}$, $(\ell, k) \in A$. The external rates are given by $g^k(\zeta) = h_0^k + h_1^k \theta^k$ for all $k \in N$ and the network rates are given by $d^{\ell k}(\zeta) = f_0^{\ell k} + f_1^{\ell k} \eta^{\ell k}$ for all $(\ell, k) \in A$. The values of the coefficients for the *base case* are given by $h_0^k = 3 \times 10^{-4} u_0^k$, $h_1^k = 0.3 u_1^k$, $f_0^{\ell k} = 3 \times 10^{-4} u_0^{\ell k}$ and $f_1^{\ell k} = 0.3 u_1^{\ell k}$ where u_0^k, u_1^k are drawn independently at random from the uniform discrete distribution with support $\{1, \dots, 10\}$. The cost function is homogeneous and the budget is equal to 1.
- Logarithmic model: There are $|N|+|A|$ control variables θ^k , $k \in N$, and $\eta^{\ell k}$, $(\ell, k) \in E$. The external rates are given by $g^k(\zeta) = \ln(1 + h_0^k + h_1^k \theta^k)$ for all $k \in N$ and the network rates are given by $d^{\ell k}(\zeta) = \ln(1 + f_0^{\ell k} + f_1^{\ell k} \eta^{\ell k})$ for all $(\ell, k) \in E$. The values of the coefficients for the base case are generated in the same way as in the linear model. The cost constraint is as before with the difference that we assume a budget of 10. This modification is made to allow the control variables to take values larger than one and consequently the logarithmic and linear model produce rates with a similar order of magnitude.

In order to conduct the comparisons, we consider uniform random networks [35] and Barabasi-Albert (BA) scale-free networks [36] with five nodes. For the uniform networks we vary the value of the connection probability p while for the scale-free networks we change the number of nodes c in the core. Once a network topology is fixed, we randomly generate the base case coefficients h and f of the linear model and use the same coefficients for the logarithmic model. Three additional experiments are conducted, one where the values of the f coefficients are divided by 2 with respect to the base case and another where these coefficients are multiplied by 2. Intuitively, the first case implies that the network influence is less efficient while in the second case the

network influence is more efficient than in the base case. In the final experiment, we assume that all values of f_1 and h_1 are equal to 0.3 and that all the values of f_0 and h_0 are equal to 3×10^{-4} . The NATP is solved by feeding formulation (25) to the function `fmincon` in Matlab. The optimal solution of E-NATP is found by using the function `globalsearch` in Matlab. The average results across five replications are in Table 1.

The results in Table 1 show that the NATP solution gives an adequate approximation to the exact optimal solution across all instances and parameter types. The expected network activation time given by the solution of NATP is roughly 20% larger than the exact optimal NAT on average, ranging from a minimum of 2% to a maximum of 39%. This is a reasonable performance, especially taking into account that the exact method takes almost three minutes to solve each instance while the solution of the NATP takes less than a second. The E-NATP problem, moreover, is not scalable, and additional experiments demonstrated that for networks with ten nodes a solution cannot be found within an hour. The uniform policy, on the other hand, performs worse than the NATP solution as it yields NATs two to four times larger than those given by exact solution for many instances.

5.2 Sensitivity to the model of the rates

In this section we examine how the structure of the optimal solution of problem (17) changes depending on the functional form of the rate functions. To this end, we consider the linear and logarithmic models described in the previous section. Because the value of the rates in the solutions of the model are not easy to compare a priori, we focus on comparing the number of nonzero control variables in the optimal solution.

We consider again uniform and BA networks with different values of p and c , respectively, and the same cases for the parameters as in the previous section. For the experiments we consider networks with ten vertices, thus there are ten external influence control variables θ^k and at most 180 network influence control variables $\eta^{\ell k}$. The average results are summarized in Table 2 across five replications for each configuration.

The results in Table 2 suggest that there are structural differences between linear and logarithmic rate models. The optimal solutions of the linear model tend to have more nonzero external control variables while the solutions of the logarithmic model tend to have more nonzero network control variables. Particularly, in the logarithmic model the number of nonzero external variables is close to one on average, which suggests that in this model there are barely any investments in increasing the external rates. In addition, it can also be observed that the number of nonzero control variables for the linear model is smaller than the corresponding number for the logarithmic model.

We believe that the above observations are a consequence of the concavity of logarithmic rates. Because

TABLE 1

Average optimal expected network activation time $E[T_s(\zeta^E)]$ and the % increase in the expectation by using uniform investment (ζ^U) and the solution of the NATP (ζ^*). The % increase by using the do-nothing policy was greater than 5000% everywhere and thus not included.

		Linear Model			Logarithmic Model		
Uniform		% incr. ζ^U	% incr. ζ^*	$E[T_s(\zeta^E)]$	% incr. ζ^U	% incr. ζ^*	$E[T_s(\zeta^E)]$
$p = 0.75$	Base/2	199%	10%	8.33	90%	10%	1.61
	Base	166%	28%	6.75	74%	18%	1.34
	Base $\times$ 2	167%	25%	4.79	64%	12%	1.09
	Equal	125%	19%	37.43	88%	33%	4.82
$p = 1$	Base/2	143%	11%	10.02	68%	14%	1.76
	Base	133%	29%	7.18	56%	13%	1.39
	Base $\times$ 2	153%	21%	4.63	50%	5%	1.10
	Equal	98%	22%	37.35	65%	39%	4.82
Scale-free							
$c = 2$	Base/2	288%	2%	7.16	129%	7%	1.52
	Base	242%	12%	6.12	106%	13%	1.33
	Base $\times$ 2	213%	13%	4.88	91%	18%	1.14
	Equal	204%	12%	37.60	155%	25%	4.83
$c = 3$	Base/2	227%	10%	7.95	105%	10%	1.59
	Base	204%	24%	6.25	86%	14%	1.36
	Base $\times$ 2	204%	19%	4.52	74%	14%	1.14
	Equal	162%	15%	37.53	119%	30%	4.83

TABLE 2

Average number of nonzeros per type of control variable across five replications.

Uniform		# $\theta^k > 0$		# $\eta^{\ell k} > 0$		Total > 0	
		Linear	Log	Linear	Log	Linear	Log
$p = 0.5$	Base/2	5.6	1.2	6.4	19.8	12	21
	Base	2.4	2.2	11.2	28	13.6	30.2
	Base $\times$ 2	1.2	2	12.8	34.4	14	36.4
	Base/2	5.6	0.2	6.4	29.6	12	29.8
$p = 0.75$	Base	2.4	0.6	11.6	40.2	14	40.8
	Base $\times$ 2	2.4	0.8	13.2	47.4	15.6	48.2
	Base/2	5.8	0.4	7.6	32.8	13.4	33.2
	Base	2.6	1.6	13.4	48.2	16	49.8
$p = 1$	Base $\times$ 2	2.2	1.6	14	58	16.2	59.6
Scale-free							
$c = 2$	Base/2	7.8	0.2	2.4	12	10.2	12.2
	Base	5.2	0.8	7.2	18.4	12.4	19.2
	Base $\times$ 2	3	1.4	11	21.2	14	22.6
	Base/2	8.2	0.2	3	13	11.2	13.2
$c = 3$	Base	5	1	8	17.8	13	18.8
	Base $\times$ 2	3.2	1.2	10.6	20.2	13.8	21.4
	Base/2	7	0.4	3.8	17	10.8	17.4
	Base	3.8	1.4	8.6	22.6	12.4	24
$c = 5$	Base $\times$ 2	2.2	1.6	11.6	26.2	13.8	27.8

$\ln(\cdot)$ is a concave function, it has the *diminishing returns* property, thus the larger the increase of the control variable the less the increase of the corresponding (external or network) rates. Thus, an optimal solution with logarithmic rates would seek to increase the value of many control variables rather than investing all the budget in increasing just a few.

Table 2 also shows that increasing the values of the coefficients of the network rates increases the number of network control variables that are nonzero. This result agrees with intuition, as larger values of coefficients

mean that the network rates increase faster in terms of the control variables. An opposite behavior is observed for the external influence rates of the linear model; however, for the logarithmic model an increase in the values of the network coefficients does increase slightly the number of nonzero external variables.

5.3 Large Scale Experiments

In this section we focus on the linear model for the rates described in Section 5.1 and run experiments across large-scale instances. We use Algorithm 1 as described in Section 4.2 using the Hao-Orlin algorithm to solve the separation problem. Repeating the Goldberg-Tarjan method for the separation is also considered, however, the Hao-Orlin approach yields significantly better results (see Appendix B, where, in addition, we show that the cut approach is significantly better than directly solving the max-flow formulation in (34)). All LPs are solved using CPLEX 12.9.0 and the rooted-out min-cut problems are solved using the LEMON graph library [37]. Because we are using linear influence rates, Network Design settings can be solved using a minimum spanning arborescence problem [18]. We use the LEMON library implementation of the MSA algorithm in [38] to solve those instances. The algorithms were coded in C++ and the experiments were carried out on a 64-bit Windows 10 Pro laptop with 8GB of RAM and an eighth-generation i7 1.8GHz processor.

We consider two types of network topologies:

- Real networks taken from [11]. They include networks from many online repositories [39], [40], [41], [42]. We refer to these networks as RZ instances.

Whenever the instance is undirected, we transform each undirected edge $\{k, \ell\}$ into two directed arcs.

- Directed versions of small-world Watts–Strogatz (WS) networks [43]. These instances are generated randomly from three parameters: The number of nodes n , the average number of neighbors per node δ , and the rewiring probability β . We use values from $n = 1,200$ to $n = 10,000$, let $\delta \in \{4, 8, 12, 16\}$, and assume $\beta \in \{0.1, 0.3\}$. The number of arcs of these networks is given by $m = n\delta$, thus the largest instances have $n = 10,000$ nodes and $m = 160,000$ arcs. These type of instances, albeit smaller, have recently been used by [12] in influence maximization problems. By construction these networks are undirected; we transform each undirected edge into two directed arcs as explained for the RZ case.

Once a graph has a fixed topology, we consider eight different parameters sets that reflect different influence models as described in Section 2:

- General: We have that $f_0^{\ell k} = Mu_0^{\ell k}/(n+m)$, $f_1^{\ell k} = Mu_1^{\ell k}$, $h_0^k = Mu_0^k/(n+m)$, and $h_1^k = Mu_1^k$. Here, M is a scaling constant given by the largest power of ten smaller than $m+n$ and $u_0^{\ell k}$, $u_1^{\ell k}$, u_0^k , and u_1^k are drawn independently at random from a uniform discrete distribution between 1 and 10 for each node $k \in N$ and arc $(\ell, k) \in A$.
- Target-set selection: As in the General type but setting $f_1^{\ell k} = 0$ for all $(\ell, k) \in A$.
- Target-arc selection: As in the General type but setting $h_1^k = 0$ for all $k \in N$.
- Network Design: As in the General type but setting $f_0^{\ell k} = h_0^k = 0$ for all $k \in N$ and $(\ell, k) \in A$.

For each of the above *non-constant* types we have a corresponding *constant* type where the numbers do not depend on the node or arc, thus $u_0^{\ell k} = u_0^A$, $u_1^{\ell k} = u_1^A$, $u_0^k = u_0^N$, and $u_1^k = u_1^N$ for all $(\ell, k) \in A$, $k \in N$.

For the RZ instances, five replications are conducted on the same network topology. On each replication, the parameters of the General type are generated first and then these parameters are reused for the other parameter types by making the corresponding values equal to zero. For the WS instances five replications are also conducted. On each replication we generate a different network and the parameters on each replication are generated as in the RZ instances. We let each instance run for at most two hours. Once the *relative gap* $(z_s^i - p_s(\zeta^i))/p_s(\zeta^i)$ at iteration i of Algorithm 1 becomes less than 10^{-3} the algorithm stops and reports the best solution found.

The results for the RZ instances, along with the instances name and sizes, are given in Table 3 and Table 4 for the non-constant and constant parameter types, respectively. The results for the WS instances are presented in Table 5 and Table 6 for the non-constant and constant parameter, respectively. The results in these tables assume that all nodes are initially inactive ($s = 0$).

From Tables 3 and 4 it can be observed that Algorithm 1 is very effective at solving virtually all instances

across all parameter types. For the instances where the algorithm does not add cuts, i.e., those in which at optimality the slowest transition happens in a state in $\{s\} \cup \{r: n(r) = 1\}$, the algorithm takes less than a few seconds to solve. Likewise, solving network design instances using the MSA approach take a fraction of a second. In the instances for which adding cuts is necessary, the solution times are of the order of a few minutes. Only for the largest instance (Douban), an optimal solution was not found across any of the replications within two hours for three out of eight of the parameter types. Regardless, for those cases the algorithm found solutions with a fairly small relative gap of about 1%.

The results for the RZ instances show that our modeling and algorithmic framework is highly scalable and can solve influence maximization problems in real world large-scale networks. The model is also fairly sensitive to the structure of the parameters. Models with the General and Target-arc parameter types are more time consuming to solve than models with the Target-set and Network Design (ND) parameter types. While the fast solution times of the ND type instances can be attributed to a different algorithm, for the Target-set type instances the reduction in computational time occurs because the state in which the minimum is attained at optimality belongs to $\mathcal{E}^0$ practically always. The results also show that non-constant parameters lead to longer solution times in general (with the exception of some target-arc problems). Such behavior is expected, i.e., having a heterogeneous population with heterogeneous connections should result in instances that are more challenging to solve. In addition, from the tables it can be seen that the number of cuts that are added is significantly smaller than the total number of constraints in the original LP formulation (25). As an illustration, consider the AS01 instance in Table 3. In theory, the original LP contains 2^{10670} constraints; in practice, an optimal was found with less than 2^9 of those constraints.

The results for the WS instances in Tables 5 and 6 reinforce the conclusions that are made for the RZ instances. In this case, most instances are solved in less than a minute and only the larger ones required a few minutes of computational time; the Target-set type instances do not add cuts; and the non-constant instances are more challenging to solve because the algorithm needs to add many more cuts outside $\mathcal{E}^0$. The results for WS instances also illustrate that the computational cost (both with respect to the solution time and the number cuts) seems to increase linearly with the number of nodes. This observation further confirms the scalability of our model and of Algorithm 1 in solving influence maximization problems.

A counter intuitive observation that can be made for the WS instances is that for General and Target-arc parameter types, networks with more arcs (for a fixed number of nodes) require fewer cuts and less time to be solved. These instances were also solved by setting $\mathcal{E}^0 = \emptyset$ at the beginning of the algorithm and

TABLE 3

Average time in seconds and average number of cuts across five replications for the RZ instances assuming non-constant parameters and $s = 0$. (*) means that no replication ended before two hours and we report instead the average relative gap.

Instance	n	m	General		Target-set		Target-arc		Network D.
			Time	Cuts	Time	Cuts	Time	Cuts	Time
Advagato	5,042	78,454	166.373	186	0.668	0	340.309	304	0.013
Anybeat	12,645	98,264	778.292	353	0.773	0	2,010.050	544	0.022
AS01	10,670	44,004	385.785	340	1.014	0	765.665	462	0.012
AS02	10,900	62,360	451.312	333	0.920	0	728.352	465	0.016
B-Alpha	3,783	24,186	0.309	0	0.255	0	0.042	0	0.005
B-OTC	5,881	35,592	0.460	0	0.470	0	0.064	0	0.008
Escorts	10,106	78,032	1,063.640	486	0.424	0	1,878.040	829	0.028
Facebook1	39,439	100,444	8.861	0	4.676	0	6.186	0	0.034
Facebook2	2,888	5,962	0.242	0	0.121	0	0.072	0	0.003
G04	10,876	39,994	2.011	0	0.984	0	0.107	0	0.015
G05	8,846	31,839	1.055	0	0.647	0	0.085	0	0.011
G06	8,717	31,525	1.624	0	0.826	0	0.095	0	0.011
G08	6,301	20,777	0.800	0	0.522	0	0.070	0	0.008
G09	8,114	26,013	0.895	0	0.561	0	0.093	0	0.010
Gplus	23,613	78,364	8.246	1	2.360	0	7.126	1	0.023
Hamsterster	1,778	24,952	21.574	74	0.268	0	27.389	117	0.005
Ning	9,727	81,140	1,507.450	655	0.687	0	1,581.750	1100	0.020
Douban	154,908	654,324	(*) (0.73%)	67	1,534	6	(*) (0.85%)	55	0.194

TABLE 4

Average time in seconds and average number of cuts across five replications for the RZ instances assuming constant parameters and $s = 0$. (*) means that no replication ended before two hours and we report instead the average relative gap.

Instance	n	m	General		Target-set		Target-arc		Network D.
			Time	Cuts	Time	Cuts	Time	Cuts	Time
Advagato	5,042	78,454	120.370	326	0.220	0	579.096	1,407	0.013
Anybeat	12,645	98,264	23.537	22	0.575	0	529.055	485	0.016
AS01	10,670	44,004	2.262	2	0.420	0	639.356	912	0.014
AS02	10,900	62,360	2.471	2	0.389	0	879.003	1,009	0.019
B-Alpha	3,783	24,186	0.187	0	0.091	0	0.058	0	0.005
B-OTC	5,881	35,592	0.327	0	0.203	0	0.147	0	0.008
Escorts	10,106	78,032	2.412	2	0.524	0	1,706.040	2,353	0.022
Facebook1	39,439	100,444	5.412	0	0.311	0	5.373	0	0.024
Facebook2	2,888	5,962	0.117	0	0.061	0	0.056	0	0.003
G04	10,876	39,994	1.171	1	0.345	0	0.104	0	0.014
G05	8,846	31,839	0.425	0	0.307	0	0.082	0	0.010
G06	8,717	31,525	0.417	0	0.345	0	0.096	0	0.011
G08	6,301	20,777	0.310	0	0.143	0	0.062	0	0.008
G09	8,114	26,013	0.419	0	0.228	0	0.078	0	0.010
Gplus	23,613	78,364	3.248	0	1.787	0	4.491	1	0.021
Hamsterster	1,778	24,952	36.998	230	0.074	0	21.352	216	0.005
Ning	9,727	81,140	2.295	2	0.427	0	2,475.110	2,849	0.020
Douban	154,908	654,324	171.666	2	52.988	0	(*) (1.045%)	96	0.169

in this case the instances with more arcs took more time and required more cuts to be solved. This behavior indicates that for the WS instances we considered in our experiments, the initial set of cuts given by the states in $\mathcal{E}^0$ tends to generate improved cuts for denser instances.

Finally, similar experiments as above are performed in Appendix C assuming that the initial state $s \neq 0$. As it can be checked there, the results follow the same behavior as in the case when $s = 0$ with the difference of taking a fraction of the time because the optimization problems are smaller.

6 CONCLUSIONS

We discuss a Markovian framework for modeling stochastic influence maximization problems in networks. Our approach assumes the decision maker can modify the external and network influence rates that control the activation times of the nodes; these modifications have a cost and the decision maker has a finite budget. We derive exact formulas for the distribution and the expectation of the network activation time and show that the respective optimization problems are not in the class of NP problems. Consequently, we frame the problem as minimizing a stochastic upper bound of the

TABLE 5
Average time in seconds and average number of cuts across five replications for the WS instances assuming non-constant parameters and $s = 0$.

n	m	δ	β	General		Target-set		Target-arc		Network D.
				Time	Cuts	Time	Cuts	Time	Cuts	Time
1,200	4,800	4	0.1	26.294	170	0.071	0	46.829	320	0.002
			0.3	19.532	125	0.084	0	32.293	229	0.001
	9,600	8	0.1	20.370	108	0.093	0	21.821	139	0.002
			0.3	16.879	88	0.096	0	17.378	113	0.002
	14,400	12	0.1	14.552	70	0.102	0	13.158	77	0.002
			0.3	12.366	57	0.107	0	10.792	63	0.002
	19,200	16	0.1	12.560	55	0.117	0	11.418	59	0.003
			0.3	10.497	45	0.113	0	9.824	51	0.002
2,000	8,000	4	0.1	62.346	294	0.048	0	104.805	529	0.003
			0.3	49.142	231	0.051	0	73.556	380	0.003
	16,000	8	0.1	44.500	184	0.131	0	51.348	238	0.003
			0.3	39.272	146	0.133	0	41.094	174	0.003
	24,000	12	0.1	39.786	125	0.149	0	35.109	143	0.004
			0.3	34.094	112	0.136	0	31.017	121	0.003
	32,000	16	0.1	30.957	97	0.146	0	28.752	101	0.004
			0.3	28.273	85	0.167	0	27.671	90	0.004
3,000	12,000	4	0.1	89.725	377	0.160	0	229.482	768	0.004
			0.3	74.982	329	0.173	0	178.437	575	0.004
	24,000	8	0.1	93.194	286	0.159	0	112.215	368	0.004
			0.3	79.434	236	0.176	0	89.433	286	0.005
	36,000	12	0.1	78.931	187	0.184	0	79.441	213	0.005
			0.3	66.729	156	0.170	0	69.225	180	0.008
	48,000	16	0.1	80.116	134	0.217	0	64.918	148	0.012
			0.3	89.920	124	0.211	0	62.718	125	0.009
5,000	20,000	4	0.1	527.773	682	0.134	0	1,247.490	1,343	0.007
			0.3	303.773	525	0.131	0	853.480	1,035	0.006
	40,000	8	0.1	292.488	445	0.294	0	413.505	607	0.008
			0.3	247.274	361	0.286	0	308.816	470	0.009
	60,000	12	0.1	290.546	339	0.292	0	309.296	374	0.019
			0.3	213.438	252	0.308	0	244.570	297	0.011
	80,000	16	0.1	258.639	236	0.334	0	240.203	258	0.011
			0.3	241.692	213	0.349	0	234.778	229	0.014
10,000	40,000	4	0.1	1,915.280	1,193	0.433	0	6,158.270	2,716	0.012
			0.3	1,609.660	954	0.396	0	3,494.230	2,007	0.015
	80,000	8	0.1	2,197.250	849	0.516	0	2,819.170	1,234	0.015
			0.3	1,724.230	736	0.559	0	2,728.080	938	0.015
	120,000	12	0.1	1,567.640	626	1.098	0	1,927.380	760	0.016
			0.3	1,384.800	561	0.969	0	1,495.160	621	0.016
	160,000	16	0.1	1,395.680	467	1.218	0	1,911.580	522	0.018
			0.3	1,345.280	405	1.157	0	1,708.560	460	0.019

network activation time, which also can be interpreted as maximizing the minimum sojourn rate of the Markov chain. This upper bound optimization problem is called the network activation time problem (NATP).

We provide performance guarantees for the exact expectation problem in terms of the optimal solution of the NATP. In this regard, we compare the expected network activation times $E[T_s(\zeta^*)]$ and $E[T_s(\zeta^E)]$ given by solutions of NATP and E-NATP, respectively, under four different sets of parameters for both the linear and the logarithmic rate functions over two groups of instances. The results in Table 1 show that the expected network activation time due to the NATP solution is roughly 20% longer on average than the exact optimal network activation time given by the E-NATP solution. Moreover, the E-NATP is not scalable and is computationally expensive, e.g., for a network with 10 nodes a solution cannot be found in an hour.

We also conduct a sensitivity analysis with respect to the model of rates. Table 2 confirms the intuition that the larger the coefficients of the rate functions, the faster the network rates increase in terms of the control variables. For the external influence rates in the linear model, the opposite behavior is observed; however, the number of nonzero external variables increase slightly as the coefficients of the network rates do increase. Generally speaking, our results show that depending on the functional form of the rates, the structure of the optimal solution can be different.

Then, we show that the NATP becomes a convex optimization problem and hence can be solved efficiently under concavity and convexity assumptions on the influence rates and the costs. We provide two approaches to solve the problem: A polynomially-sized mathematical programming formulation and a cut generation algorithm. For the computational

TABLE 6

Average time in seconds and average number of cuts across five replications for the WS instances assuming constant parameters and $s = 0$. (*) means that no replication ended before two hours and we report instead the average relative gap.

n	m	δ	β	General		Target-set		Target-arc		Network D.
				Time	Cuts	Time	Cuts	Time	Cuts	Time
1,200	4,800	4	0.1	0.061	0	0.045	0	0.564	11	0.002
			0.3	0.145	1	0.047	0	1.154	23	0.002
	9,600	8	0.1	0.323	4	0.053	0	0.311	4	0.006
			0.3	0.192	2	0.051	0	0.185	2	0.004
	14,400	12	0.1	2.005	25	0.059	0	2.053	26	0.009
			0.3	1.669	20	0.060	0	1.650	20	0.009
	19,200	16	0.1	2.299	25	0.084	0	2.246	24	0.010
			0.3	1.199	11	0.074	0	1.119	11	0.013
2,000	8,000	4	0.1	1.645	18	0.071	0	1.615	20	0.003
			0.3	2.868	32	0.070	0	2.602	33	0.003
	16,000	8	0.1	7.051	61	0.084	0	6.677	66	0.014
			0.3	7.653	65	0.084	0	6.656	64	0.004
	24,000	12	0.1	4.395	30	0.104	0	4.436	34	0.019
			0.3	4.088	27	0.100	0	3.225	23	0.019
	32,000	16	0.1	4.518	26	0.104	0	4.020	26	0.020
			0.3	2.948	17	0.111	0	3.016	18	0.029
	12,000	4	0.1	2.595	18	0.117	0	1.993	17	0.004
			0.3	4.800	38	0.126	0	4.863	40	0.004
3,000	24,000	8	0.1	12.583	84	0.148	0	13.107	86	0.033
			0.3	15.694	82	0.164	0	14.058	85	0.007
	36,000	12	0.1	10.541	41	0.180	0	7.085	37	0.039
			0.3	11.246	39	0.176	0	8.276	39	0.041
	48,000	16	0.1	11.834	33	0.204	0	8.149	34	0.047
			0.3	8.248	22	0.207	0	5.278	21	0.073
5,000	20,000	4	0.1	107.522	552	0.189	0	149.028	599	0.005
			0.3	117.785	568	0.203	0	1,603.660	2,670	0.005
	40,000	8	0.1	27.408	103	0.223	0	44.800	104	0.061
			0.3	17.818	75	0.225	0	39.923	77	0.014
	60,000	12	0.1	12.146	50	0.279	0	41.155	54	0.103
			0.3	10.982	44	0.304	0	40.569	52	0.096
	80,000	16	0.1	11.153	40	0.324	0	36.217	42	0.121
			0.3	9.439	28	0.359	0	24.873	30	0.189
10,000	40,000	4	0.1	8.670	18	0.341	0	(*) 0.28%	6,494	0.013
			0.3	5.234	6	0.397	0	(*) 0.34%	7,427	0.010
	80,000	8	0.1	27.632	32	0.404	0	1,585.500	1,651	0.080
			0.3	34.338	49	0.399	0	2,074.020	2,056	0.030
	120,000	12	0.1	37.581	46	0.444	0	105.637	138	0.295
			0.3	41.415	56	0.461	0	206.419	224	0.114
	160,000	16	0.1	42.170	35	0.557	0	180.177	174	0.374
			0.3	31.519	34	0.573	0	77.913	80	0.612

experiments over both Watts-Strogatz (WS) and real world instances, we design eight different parameter sets of rates including four *non-constant* and four *constant* types. They reflect different influence models, namely General, Target-set selection, Target-arc selection, and Network Design. The results show that the cutting plane algorithm can solve many large scale synthetic and real world instances with more than 10^4 and 10^5 nodes and arcs in a matter of a few seconds to a few minutes. Our results for WS type networks also show that the solution time grows almost linearly with the size of the instances, suggesting that our approach provides a scalable method to solve influence maximization problems in networks.

ACKNOWLEDGMENTS

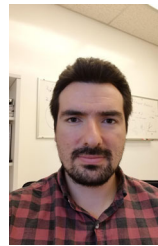
The research of the first author in this paper is partially supported by the Office of Naval Research under

Grant N00014-19-1-2329. The last author was partially supported by the Air Force Office of Scientific Research grant FA9550-19-1-0161.

REFERENCES

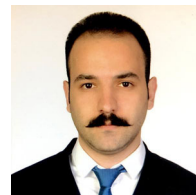
- [1] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," *Theory of Computing*, vol. 11, no. 4, pp. 105–147, 2015.
- [2] H. W. Hethcote, "The mathematics of infectious diseases," *SIAM review*, vol. 42, no. 4, pp. 599–653, 2000.
- [3] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003, pp. 137–146.
- [4] C. Wang, W. Chen, and Y. Wang, "Scalable influence maximization for independent cascade model in large-scale social networks," *Data Mining and Knowledge Discovery*, vol. 25, no. 3, pp. 545–576, 2012.

- [5] W. Chen, L. V. Lakshmanan, and C. Castillo, "Information and influence propagation in social networks," *Synthesis Lectures on Data Management*, vol. 5, no. 4, pp. 1–177, 2013.
- [6] H.-H. Wu and S. Küçükyavuz, "A two-stage stochastic programming approach for influence maximization in social networks," *Computational Optimization and Applications*, vol. 69, no. 3, pp. 563–595, 2018.
- [7] E. Güney, "An efficient linear programming based method for the influence maximization problem in social networks," *Information Sciences*, vol. 503, pp. 589–605, 2019.
- [8] M. Samadi, R. Nagi, A. Semenov, and A. Nikolaev, "Seed activation scheduling for influence maximization in social networks," *Omega*, vol. 77, pp. 96–114, 2018.
- [9] N. Chen, "On the approximability of influence in social networks," *SIAM Journal on Discrete Mathematics*, vol. 23, no. 3, pp. 1400–1415, 2009.
- [10] E. Ackerman, O. Ben-Zwi, and G. Wolfowitz, "Combinatorial model and bounds for target set selection," *Theoretical Computer Science*, vol. 411, no. 44–46, pp. 4017–4022, 2010.
- [11] S. Raghavan and R. Zhang, "A branch-and-cut approach for the weighted target set selection problem on social networks," *Informatics Journal on Optimization*, vol. 1, no. 4, pp. 304–322, 2019.
- [12] M. Fischetti, M. Kahr, M. Leitner, M. Monaci, and M. Ruthmair, "Least cost influence propagation in (social) networks," *Mathematical Programming*, vol. 170, no. 1, pp. 293–325, 2018.
- [13] D. Güneş, S. Raghavan, and R. Zhang, "Least-cost influence maximization on social networks," *INFORMS Journal on Computing*, 2019.
- [14] —, "A branch-and-cut approach for the least cost influence problem on social networks," *Networks*, vol. 76, no. 1, pp. 84–105, 2020.
- [15] G. Nannicini, G. Sartor, E. Traversi, and R. Wolfler-Calvo, "An exact algorithm for robust influence maximization," *Mathematical Programming*, 2020.
- [16] H. T. Nguyen, M. T. Thai, and T. N. Dinh, "Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks," in *Proceedings of the 2016 International Conference on Management of Data*, 2016, pp. 695–710.
- [17] A. Y. Lokhov and D. Saad, "Optimal deployment of resources for maximizing impact in spreading processes," *Proceedings of the National Academy of Sciences*, vol. 114, no. 39, pp. E8138–E8146, 2017.
- [18] J. S. Borrero, O. A. Prokopyev, and P. Krokhmal, "Optimization of cascading processes in arbitrary networks with stochastic interactions," *IEEE Transactions on Network Science and Engineering*, vol. 6, no. 4, pp. 773–787, 2019.
- [19] V. M. Preciado, M. Zargham, C. Enyioha, A. Jadbabaie, and G. Pappas, "Optimal resource allocation for network protection: A geometric programming approach," *IEEE Transactions on Control of Network Systems*, vol. 1, no. 1, pp. 99–108, 2014.
- [20] C. Nowzari, V. M. Preciado, and G. J. Pappas, "Analysis and control of epidemics: A survey of spreading processes on complex networks," *IEEE Control Systems Magazine*, vol. 36, no. 1, pp. 26–46, 2016.
- [21] I. Z. Kiss, J. C. Miller, P. L. Simon *et al.*, "Mathematics of epidemics on networks," *Cham: Springer*, vol. 598, 2017.
- [22] J. Dong and G.-H. Yang, "H2 state feedback control synthesis of continuous-time uncertain markov jump linear systems," in *2007 American Control Conference*. IEEE, 2007, pp. 2417–2421.
- [23] —, "Robust h2 control of continuous-time markov jump linear systems," *Automatica*, vol. 44, no. 5, pp. 1431–1436, 2008.
- [24] V. G. Kulkarni, *Modeling and analysis of stochastic systems*. Crc Press, 2016.
- [25] S. I. Resnick, *Adventures in stochastic processes*. Springer Science & Business Media, 1992.
- [26] M. Shaked and J. G. Shanthikumar, *Stochastic orders*. Springer Science & Business Media, 2007.
- [27] H. Wolkowicz and G. P. Styan, "Bounds for eigenvalues using traces," *Linear algebra and its applications*, vol. 29, pp. 471–506, 1980.
- [28] —, "More bounds for eigenvalues using traces," *Linear Algebra and Its Applications*, vol. 31, pp. 1–17, 1980.
- [29] G. B. Dantzig and D. R. Fulkerson, "On the max flow min cut theorem of networks," 1955.
- [30] A. Ben-Tal and A. Nemirovski, *Lectures on modern convex optimization: analysis, algorithms, and engineering applications*. Siam, 2001, vol. 2.
- [31] A. Nedić and A. Ozdaglar, "Approximate primal solutions and rate analysis for dual subgradient methods," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1757–1780, 2009.
- [32] A. V. Goldberg and R. E. Tarjan, "A new approach to the maximum-flow problem," *Journal of the ACM (JACM)*, vol. 35, no. 4, pp. 921–940, 1988.
- [33] J. Hao and J. B. Orlin, "A faster algorithm for finding the minimum cut in a directed graph," *Journal of Algorithms*, vol. 17, no. 3, pp. 424–446, 1994.
- [34] MATLAB, version 9.7 (R2019b). Natick, Massachusetts: The MathWorks Inc., 2019.
- [35] P. Erdős and A. Rényi, "On the evolution of random graphs," *Publ. Math. Inst. Hung. Acad. Sci.*, vol. 5, no. 1, pp. 17–60, 1960.
- [36] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of modern physics*, vol. 74, no. 1, p. 47, 2002.
- [37] B. Dezső, A. Jüttner, and P. Kovács, "Lemon—an open source c++ graph template library," *Electronic Notes in Theoretical Computer Science*, vol. 264, no. 5, pp. 23–45, 2011.
- [38] D. R. Fulkerson, "Packing rooted directed cuts in a weighted directed graph," *Mathematical Programming*, vol. 6, no. 1, pp. 1–13, 1974.
- [39] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection," <http://snap.stanford.edu/data>, jun 2014.
- [40] R. Rossi and N. Ahmed, "The network data repository with interactive graph analytics and visualization," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [41] O. Lesser, L. Tenenboim-Chekina, L. Rokach, and Y. Elovici, "Intruder or welcome friend: Inferring group membership in online social networks," in *International conference on social computing, behavioral-cultural modeling, and prediction*. Springer, 2013, pp. 368–376.
- [42] J. Kunegis, "Konec network dataset," <http://konec.uni-koblenz.de/about>, mar 2020.
- [43] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, vol. 393, no. 6684, p. 440, 1998.



Juan S. Borrero received the BS degree in math and the MS degree in industrial engineering from the Universidad de los Andes in Bogotá, Colombia, and the PhD degree in industrial engineering from the University of Pittsburgh. He is an assistant professor with the School of Industrial Engineering & Management, Oklahoma State University. His research interests include decision making in adversarial settings under uncertainty, bilevel optimization, models for the spread of influence in networks, and operations

research applied to defense and manufacturing problems



Majid Akhgar received the BS degree in Industrial Engineering from the Bu-Ali Sina University, Iran. He is a PhD student with the School of Industrial Engineering & Management, Oklahoma State University. His research interests include network optimization, models for the spread of influence in networks, stochastic modeling, and supply chain network design problems.



Pavlo Krokhmal received the PhD degree in operations research from the University of Florida, and the PhD degree in mechanics and applied mathematics from Kyiv University, Ukraine. He is a professor of systems and industrial engineering with the University of Arizona. His research interests include stochastic optimization and risk analysis, probabilistic combinatorial optimization, cooperative control and multi-agent decision making under uncertainty, and data mining. His research in decision making and optimization under uncertainty has been supported by the NSF, the Air Force Office of Scientific Research, the Defense Threat Reduction Agency, the Air Force Research Lab, and he is a recipient of the AFOSR Young Investigator Award, National Research Council Senior Research Associateship Award, and several AFOSR Air Force Summer Faculty Fellowship Awards. He is co-Editor-in-Chief of Optimization Letters.

APPENDIX A

In this appendix we show all the formal proofs.

Proof of Proposition 1. A modification of the argument leading to the backwards equations of $\{X(t), t \geq 0\}$ implies that

$$P[T_{r^j} > t] = e^{-q_j t} + \sum_{r^{j-1} \in \tilde{\mathcal{R}}(r^j)} \frac{q_{j,j-1}}{q_j} \int_0^t q_j e^{-q_j u} P[T_{r^{j-1}} > t - u] du, \quad (39)$$

see Chapter 5 in [25]. The desired formula is proved by using induction over j , using Equation (39), and rearranging the resulting terms.

Proof of Proposition 2. By exploiting the Markov property of the CTMC and by the basic properties of the exponential distribution, it can be shown that the expectations satisfy the following system of linear equations, see [24]:

$$u_r = \frac{1}{q_r} + \sum_{r' \in \tilde{\mathcal{R}}(r)} \frac{q_{rr'}}{q_r} u_{r'} \quad \forall r \in \tilde{\mathcal{E}}(s) \cup \{s\}. \quad (40)$$

Equation (9) follows from (40) by induction, starting with $j = 1$.

Proof of Proposition 4. Let $\hat{\zeta} \in \mathcal{B}$ attain the maximum in (21) and let $r, r' \in \mathcal{E}(s)$ be the states where $M_s(\hat{\zeta})$ and $m_s(\hat{\zeta})$ attain their respective maximum and minimum. Therefore,

$$\max \left\{ \frac{M_s(\zeta)}{m_s(\zeta)} : \zeta \in \mathcal{B} \right\} = \frac{q_r(\hat{\zeta})}{q_{r'}(\hat{\zeta})}. \quad (41)$$

Observe that

$$\begin{aligned} q_r(\hat{\zeta}) &= \sum_{k \in N} (1 - r_k) \left(g^k(\hat{\zeta}) + \sum_{(\ell, k) \in E} r_\ell d^{\ell k}(\hat{\zeta}) \right) \\ &\leq n(s)(b + \Delta^-(s)b) = bn(s)(1 + \Delta^-(s)). \end{aligned}$$

Since the optimal value is finite, $q_{r'}(\hat{\zeta}) > 0$ and the above equation yields (22). Suppose, on the other hand, that all the rate functions in $m_s(\hat{\zeta})$ are greater than zero. Then there exist $b' > 0$ such that

$$\begin{aligned} q_{r'}(\hat{\zeta}) &= \sum_{k \in N} (1 - r'_k) \left(g^k(\hat{\zeta}) + \sum_{(\ell, k) \in E} r'_\ell d^{\ell k}(\hat{\zeta}) \right) \\ &\geq b' + b' \delta^-(s) = b'(1 + \delta^-(s)), \end{aligned}$$

which implies that (23) holds.

Proof of Proposition 5. The proof follows from applying the upper and lower bounds on the largest and smallest eigenvalues of $-Q_s(\zeta)$ that are given in [27], [28].

Note that the maximum flow problem $\phi_{sk}(\zeta)$ is given by

$$\phi_{sk}(\zeta) = \max \sum_{(R_s, v) \in A_s} x_{R_s v}^k \quad (42a)$$

$$\text{s. t.} \quad \sum_{(u, v) \in A_s} x_{uv}^k - \sum_{(v, u) \in A_s} x_{vu}^k \leq 0 \quad \forall v \in N_s \quad (42b)$$

$$x_{R_s v}^k - g^v(\zeta) \leq 0 \quad \forall v \in N_s \quad (42c)$$

$$x_{uv}^k - d^{uv}(\zeta) \leq 0 \quad \forall (u, v) \in \tilde{A}_s \quad (42d)$$

$$x_{uv}^k \geq 0 \quad \forall (u, v) \in A_s. \quad (42e)$$

Therefore, constraint (33b) holds for $k \in N_s \setminus \{R_s\}$ if and only if there exist $x^k \in \mathbb{R}_+^{|A_s|}$ satisfying equations (42b)–(42d) such that $f \leq \sum_{(R_s, v) \in A_s} x_{R_s v}^k$. This observation implies the desired result.

Proof of Proposition 9. The proof follows from the observation that in this setting, if $r \in \{r' \in \mathcal{S} : n(r') = r\}$, then $q_r(\zeta) = f(r, \zeta) = rg(\zeta) + r(n-r)d(\zeta)$. Observe that $f(r, \zeta)$ is a concave function of r , hence for any ζ and $a < b$ we have that $\min\{f(r, \zeta) : a \leq r \leq b\}$ is either $f(a, \zeta)$ or $f(b, \zeta)$. The result follows from these observations after simple computations.

APPENDIX B

In this appendix we compare the performance of the generalized max-flow formulation in (34); Algorithm 1 by repeatedly solving the min-cut problem in terms of the max-flow Goldberg-Trajan algorithm (GT); and Algorithm 1 solving the rooted-out min-cut problem directly using Hao-Orlin algorithm (HO), using some of the instances described in Section 5.3. For each instances considered we run two replications with the General parameters, and assume the linear model for the rates. We note that the implementations in LEMON of the GT and HO algorithms have a worst-case time complexity of $O(n(s)^2 \sqrt{m(s)})$. Recall, however, that in our setting the complexity of GT's algorithm has to be further multiplied by $n(s)$. The results of the comparisons are shown in Table 7.

The results of Table 7 clearly show that Algorithm 1 with Hao-Orlin method is the better alternative out of the three approaches and that the cut generation methods are significantly better than the monolithic max-flow formulation. The generalized-max flow solution approach, while polynomial, does not translate into a scalable method in practice, its main drawback being that the size of the formulation depends quadratically on the size of the network. In contrast, while the cut generation algorithms are very fast in practice. In what follows, we use the version of Algorithm 1 with the Hao-Orlin min-cut separation to run the rest of the experiments.

APPENDIX C

In this appendix we show the tables for the RZ and WS instances of Section 5.3 assuming that $s \neq 0$. In Tables 8

TABLE 7

Comparison of three approaches. Times in seconds. Notation for the WS instances is $n.k.\beta$. NEM stands for instances where the solver stopped because of not enough memory, (*) stands for an instance that did not finish after on hour.

Instance	#Nodes	#Arcs	Algorithm 1+HO		Algorithm 1+GT		Max-Flow in Eq. (34)
			Time	#Cuts	Time	#Cuts	Time
WS 300.4.3	300	1,200	0.823	38	3.600	38	262.726
WS 300.8.3	300	2,400	0.540	19	2.360	19	653.361
WS 300.12.3	300	3,600	0.717	14	1.935	14	1,627.170
WS 300.16.3	300	4,800	0.545	10	1.886	10	3,108.010
Advagato	5,042	78,454	231.187	193	3,600(*)	27	NEM
B-OTC	5,881	35,592	0.719	0	45.384	0	NEM
B-Alpha	3,783	24,186	0.408	0	4.381	0	NEM
Facebook2	2,888	5,962	0.212	0	50.232	0	NEM
Hamsterster	1,778	24,952	20.004	68	682.401	68	NEM

TABLE 8

Average time in seconds and average number of cuts across five replications for the RZ instances assuming non-constant parameters and $s \neq 0$. (*) means that no replication ended before two hours and we report instead the average relative gap.

Instance			General		Target-set		Target-arc		Network D.
	n	m	Time	Cuts	Time	Cuts	Time	Cuts	Time
Advagato	5,042	78,454	80.413	110	0.388	0	115.305	165	0.012
Anybeat	12,645	98,264	155.744	125	0.498	0	300.219	227	0.017
AS01	10,670	44,004	181.191	209	0.552	0	285.398	320	0.011
AS02	10,900	62,360	269.057	239	0.580	0	372.450	337	0.014
B-Alpha	3,783	24,186	0.217	0	0.179	0	0.046	0	0.004
B-OTC	5,881	35,592	0.357	0	0.185	0	0.078	0	0.006
Escorts	10,106	78,032	548.831	351	0.731	0	774.257	517	0.020
Facebook1	39,439	100,444	0.782	0	0.590	0	0.606	0	0.008
Facebook2	2,888	5,962	0.077	2	0.025	0	0.067	2	0.001
G04	10,876	39,994	1.802	0	0.596	0	0.106	0	0.010
G05	8,846	31,839	1.024	0	0.428	0	0.095	0	0.008
G06	8,717	31,525	1.276	0	0.447	0	0.095	0	0.009
G08	6,301	20,777	0.549	0	0.283	0	0.056	0	0.006
G09	8,114	26,013	0.686	0	0.355	0	0.069	0	0.006
Gplus	23,613	78,364	4.408	2	0.654	0	1.756	1	0.012
Hamsterster	1,778	24,952	13.999	45	0.186	0	14.917	67	0.005
Ning	9,727	81,140	374.465	280	0.549	0	581.820	504	0.018
Douban	154,908	654,324	(*) (0.22%)	200	10.681	0	(*) (0.13%)	317	0.115

and 9 we show the results for the RZ instances assuming that $s \neq 0$; specifically we assume that initially the *leaf* nodes (i.e., nodes with only one incoming arc) are active. In general lines, these results follow the same patterns of Tables 3 and 4, respectively. Mostly the running times and the number of cuts are a fraction of the corresponding numbers for the instances with $s = 0$, and only for a few instances (e.g., Escorts and Ning for the constant parameter types) the version with $s \neq 0$ is harder to solve.

APPENDIX D

This Section provides Table 12, that summarizes the notations used throughout the paper.

For the WS instances, we select one third of the nodes of each network at random and assume that all of them are active. Tables 10 and 11 show the results, which give similar insights as the tables for the RZ instances, that is, having less inactive nodes translates to faster solution times and generally fewer cuts. Similar results are observed for the WS instances with $s \neq 0$.

TABLE 9

Average time in seconds and average number of cuts across five replications for the RZ instances assuming constant parameters and $s \neq 0$.

Instance	n	m	General		Target-set		Target-arc		Network D.
			Time	Cuts	Time	Cuts	Time	Cuts	Time
Advagato	5,042	78,454	249.856	592	0.425	0	547.818	1,394	0.012
Anybeat	12,645	98,264	16.740	27	0.447	0	492.030	476	0.018
AS01	10,670	44,004	2.846	5	0.402	0	396.192	1,090	0.011
AS02	10,900	62,360	23.076	46	0.479	0	573.890	1,238	0.014
B-Alpha	3,783	24,186	0.119	0	0.111	0	0.038	0	0.004
B-OTC	5,881	35,592	0.197	0	0.178	0	0.065	0	0.005
Escorts	10,106	78,032	1.816	2	0.548	0	1,631.700	2,653	0.019
Facebook1	39,439	100,444	0.528	0	0.563	0	0.414	0	0.008
Facebook2	2,888	5,962	0.105	2	0.041	0	0.058	2	0.001
G04	10,876	39,994	0.442	0	0.416	0	0.106	0	0.010
G05	8,846	31,839	0.344	0	0.273	0	0.084	0	0.009
G06	8,717	31,525	0.341	0	0.281	0	0.085	0	0.010
G08	6,301	20,777	0.198	0	0.163	0	0.063	0	0.006
G09	8,114	26,013	0.251	0	0.207	0	0.084	0	0.007
Gplus	23,613	78,364	1.281	1	0.567	0	1.272	2	0.012
Hamsterster	1,778	24,952	26.316	205	0.109	0	25.081	203	0.006
Ning	9,727	81,140	118.926	245	0.499	0	1,374.950	2,625	0.017
Douban	154,908	654,324	20.927	1	8.012	0	2,502.080	304	0.113

TABLE 10
Average time in seconds and average number of cuts across five replications for the WS instances assuming non-constant parameters and $s \neq 0$.

n	m	δ	β	General		Target-set		Target-arc		Network D.
				Time	Cuts	Time	Cuts	Time	Cuts	Time
1,200	4,800	4	0.1	7.569	79	0.101	0	19.152	129	0.002
			0.3	6.594	65	0.102	0	14.601	101	0.002
	9,600	8	0.1	7.466	55	0.110	0	12.732	68	0.003
			0.3	6.896	52	0.110	0	11.495	59	0.003
	14,400	12	0.1	4.120	26	0.134	0	8.707	39	0.003
			0.3	3.411	22	0.126	0	6.752	32	0.003
	19,200	16	0.1	3.180	18	0.132	0	4.498	19	0.003
			0.3	3.230	18	0.132	0	4.442	19	0.003
2,000	8,000	4	0.1	18.090	125	0.077	0	39.740	185	0.003
			0.3	16.270	106	0.081	0	34.404	151	0.003
	16,000	8	0.1	14.160	79	0.200	0	31.783	116	0.003
			0.3	13.370	74	0.203	0	23.594	87	0.004
	24,000	12	0.1	8.353	41	0.214	0	15.580	50	0.004
			0.3	6.818	33	0.211	0	13.603	44	0.004
	32,000	16	0.1	9.334	41	0.219	0	15.829	45	0.004
			0.3	6.898	29	0.218	0	10.542	34	0.004
3,000	12,000	4	0.1	47.179	179	0.120	0	49.998	283	0.002
			0.3	38.132	140	0.126	0	40.326	228	0.002
	24,000	8	0.1	35.976	108	0.150	0	35.421	164	0.003
			0.3	33.824	100	0.156	0	28.686	132	0.003
	36,000	12	0.1	26.854	65	0.165	0	22.625	87	0.003
			0.3	21.116	51	0.172	0	17.092	65	0.003
	48,000	16	0.1	25.612	49	0.197	0	20.168	63	0.003
			0.3	20.038	39	0.190	0	15.808	49	0.003
5,000	20,000	4	0.1	64.487	271	0.098	0	129.060	484	0.002
			0.3	60.253	240	0.104	0	122.766	442	0.004
	40,000	8	0.1	66.619	173	0.211	0	103.307	283	0.004
			0.3	60.334	153	0.210	0	77.951	218	0.006
	60,000	12	0.1	61.249	120	0.246	0	69.127	146	0.004
			0.3	40.923	80	0.238	0	63.351	129	0.004
	80,000	16	0.1	50.042	80	0.268	0	52.089	88	0.005
			0.3	42.677	67	0.297	0	46.971	79	0.005
10,000	40,000	4	0.1	440.667	508	0.380	0	1,141.250	1,063	0.008
			0.3	325.659	392	0.430	0	1,056.850	904	0.008
	80,000	8	0.1	390.891	312	0.706	0	444.162	410	0.011
			0.3	321.245	261	0.779	0	415.442	351	0.013
	120,000	12	0.1	315.991	193	0.775	0	357.285	265	0.012
			0.3	279.480	176	0.833	0	322.008	228	0.013
	160,000	16	0.1	321.037	157	0.860	0	329.110	196	0.013
			0.3	227.336	113	0.887	0	239.673	142	0.013

TABLE 11
Average time in seconds and average number of cuts across five replications for the WS instances assuming constant parameters and $s \neq 0$.

n	m	δ	β	General		Target-set		Target-arc		Network D.
				Time	Cuts	Time	Cuts	Time	Cuts	Time
1,200	4,800	4	0.1	0.045	0	0.039	0	6.220	124	0.002
			0.3	1.882	43	0.036	0	2.178	52	0.002
	9,600	8	0.1	1.124	22	0.048	0	1.106	22	0.003
			0.3	0.916	18	0.047	0	0.918	18	0.002
	14,400	12	0.1	0.193	2	0.058	0	0.202	2	0.003
			0.3	0.297	4	0.057	0	0.281	4	0.003
	19,200	16	0.1	0.297	3	0.064	0	0.297	3	0.005
			0.3	0.182	1	0.060	0	0.175	1	0.004
2,000	8,000	4	0.1	27.561	330	0.055	0	45.876	501	0.003
			0.3	24.277	282	0.061	0	24.413	291	0.003
	16,000	8	0.1	2.724	34	0.076	0	2.490	32	0.003
			0.3	2.266	29	0.072	0	2.302	27	0.003
	24,000	12	0.1	0.678	6	0.090	0	0.624	5	0.004
			0.3	0.614	5	0.085	0	0.569	5	0.004
	32,000	16	0.1	2.983	27	0.127	0	2.951	26	0.009
			0.3	3.081	27	0.102	0	2.985	27	0.007
3,000	12,000	4	0.1	65.519	786	0.053	0	52.771	644	0.002
			0.3	54.367	593	0.056	0	118.144	1,001	0.002
	24,000	8	0.1	4.918	64	0.073	0	4.148	55	0.002
			0.3	4.165	52	0.074	0	3.032	39	0.002
	36,000	12	0.1	9.748	102	0.083	0	9.836	102	0.003
			0.3	10.787	110	0.081	0	10.549	109	0.004
	48,000	16	0.1	5.649	49	0.090	0	6.190	56	0.012
			0.3	10.080	88	0.092	0	7.548	67	0.008
5,000	20,000	4	0.1	100.748	728	0.088	0	117.733	827	0.003
			0.3	124.417	806	0.087	0	260.168	1,395	0.003
	40,000	8	0.1	110.522	646	0.116	0	238.572	1,132	0.004
			0.3	96.995	542	0.113	0	98.064	586	0.004
	60,000	12	0.1	41.563	204	0.135	0	24.145	139	0.007
			0.3	44.139	212	0.158	0	37.052	203	0.010
	80,000	16	0.1	21.512	92	0.211	0	18.144	89	0.019
			0.3	26.204	103	0.214	0	23.256	108	0.016
10,000	40,000	4	0.1	8.748	27	0.424	0	902.067	1,610	0.007
			0.3	4.631	16	0.348	0	838.736	1,385	0.007
	80,000	8	0.1	36.896	93	0.497	0	30.842	76	0.009
			0.3	8.487	18	0.591	0	22.166	52	0.009
	120,000	12	0.1	31.995	56	0.708	0	24.604	54	0.010
			0.3	30.812	39	0.728	0	26.073	54	0.012
	160,000	16	0.1	49.831	53	0.848	0	25.349	46	0.038
			0.3	69.410	71	0.872	0	45.393	85	0.017

TABLE 12
List of the most important notations in the paper.

Notations	Definition
$G = (N, A)$	Any arbitrary directed network with $n = N $ nodes and $m = A $ arcs.
$N^-(k)$	Set of in-neighbors of node k .
$\mathcal{V}$	Finite set of control variables.
ζ	Vector of control variables, i.e., $\zeta = (\zeta_v: v \in \mathcal{V})$.
$d^{\ell k}(\zeta)$	The time it takes an active in-neighbor ℓ to activate node k is exponentially distributed with rate $d^{\ell k}(\zeta)$, where $d^{\ell k}$ is a non-negative function of the control variables ζ .
$g^k(\zeta)$	The time it takes external influence to activate the inactive node $k \in N$ is exponentially distributed with rate $g^k(\zeta)$, and g^k is a known non-negative function.
$X(t, \zeta)$	Activation status of the network at time t if the control variables take the value ζ .
r_k	State variable which is equal to 0 if node k is inactive; and it is 1 if node k is active.
$X(\zeta)$	Continuous time Markov chain (CTMC) that tracks the status of all nodes.
$\mathcal{S}$	State space of CTMC $X(\zeta)$.
$\mathbf{1}$	Absorbing state of the CTMC $X(\zeta)$.
$Q(\zeta)$	Generator matrix of the CTMC $X(\zeta)$.
e_k	n -dimensional unit vector with a one in the k -th entry.
$q_{rr'}$	Element of the generator matrix corresponding to row state r and column state r' .
$q_r(\zeta)$	Notation for $ q_{rr}(\zeta) $.
B	Total budget for increasing the control variables.
$T_s(\zeta)$	Network activation time given that the initial state is s and that the control variables take the value ζ .
$\tilde{\mathcal{R}}(r)$	Set of non-absorbing states reachable from r after one transition.
$\tilde{\mathcal{E}}(r)$	Set of non-absorbing states reachable from r after any number of transitions.
$p_{rr'}(t)$	Probability that the CTMC is in state r' after t time units given that it starts from r .
$n(r)$	Number of state transitions that are required for the chain to be absorbed by $\mathbf{1}$ given that it started from state r .
$m_s(\zeta)$	Minimum rate of the sojourn times of all the states reachable from s , including s itself.
$\mathcal{B}$	Set of feasible values for the control variables.
$M_s(\zeta)$	The largest sojourn rate across all states reachable from s , including s itself.
$U_s(\zeta)$	Sojourn time of the chain in state s .
$V_i(\zeta)$	A set of independent and identically distributed (iid) $\exp(m_s(\zeta))$ random variables.
$W_s(\zeta)$	A sum of $n(s)$ iid $\exp(m_s(\zeta))$ random variables.
ζ^*	An optimal solution of NATP.
ζ^E	An optimal solution of E-NATP.
$\Delta^-(s)$	Upper bound on the number of in-neighbors of any node in the network inactive in s .
$\delta^-(s)$	Lower bound on the number of in-neighbors of any node in the network inactive in s .
$Q_s(\zeta)$	Restriction of the generator matrix $Q(\zeta)$ to the states in $\mathcal{E}(s)$.
$J_r \subseteq N$	Set of active nodes in state r .
$G_s = (N_s, A_s)$	An auxiliary weighted network based on G where all active nodes in s collapse to a single node that we denote by R_s .
A_s	Set of edges in A that have no active endpoints in s plus new edges from R_s to all the inactive nodes in s .
$w_s((u, v), \zeta)$	Weights of G_s if the control variables take the value of ζ .
$C_s(J)$	Cut induced on A_s by the subset of nodes J .
$w_s(J, \zeta)$	Weight corresponding to cut $C_s(J)$.
$\mathcal{J}_s$	Class of proper subsets of N_s that contain R_s .
$\mathcal{J}_s^k$	Class of subsets of $\mathcal{J}_s$ that do not contain k .
$\phi_{sk}(\zeta)$	Maximum flow that can be sent from R_s to k in G_s assuming that the capacities of the arcs are given by their weights $w_s((u, v), \zeta)$.
$c(\zeta)$	Cost of increasing the value of the control variables from $\mathbf{0}$ to ζ .
x_{uv}^k	Flow decision-variable from node u to v if node k is the sink node.
$p_s(\zeta^i)$	Value of the minimum cut in network
θ^k	External influence control variable for node k .
η^{lk}	Network influence control variable for arc (ℓ, k) .

Risk-Averse Optimization and Resilient Network Flows

Masoud Eshghali and Pavlo A. Krokhmal*

Department of Systems and Industrial Engineering
University of Arizona, USA
{masoudeshghali, krokhmal}@email.arizona.edu

Abstract

We propose an approach to constructing metrics of network resilience, where resilience is understood as the network’s amenability to restoring its optimal or near-optimal operations subsequent to unforeseen (stochastic) disruptions of its topology or operational parameters, and illustrated it on the examples of the resilient maximum network flow problem and the resilient minimum cost network problem. Specifically, the network flows in these problems are designed for resilience against unpredictable losses of network carrying capacity, and the mechanism of attaining a degree of resilience is through preallocation of resources toward (at least partial) restoration of the capacities of the arcs. The obtained formulations of resilient network flow problems possess a number of useful properties, e.g., similarly to the standard network flow problems, the network flow is integral if the arc capacities, costs, etc., are integral. It is also shown that the proposed formulations of resilient network flow problems can be viewed as “network measures of risk”, similar in properties and behavior to convex measures of risk. Efficient decomposition algorithms have been proposed for both the resilient maximum network flow problem and the resilient minimum cost network flow problem, and a study of the network flow resilience as a function of network’s structure has been conducted on networks with three types of topology: that of uniform random graphs, scale-free graphs, and grid graphs.

Keywords: Resilient maximum network flow problem, resilient minimum cost network flow problem, stochastic network, convex risk measures, Benders decomposition

1 Introduction and Motivation

Network models and representations find applications in a variety of fields, including physics, biology, transportation and logistics, energy, telecommunications, social media, and so on. A simple network consists of a set of nodes, which are the principal components of the system, and a set of arcs that represent relations and/or connections between nodes. Industrial networked systems are most often designed with the purpose of transmitting material goods or information among their nodes. As such, a networked system may be regarded with respect to how efficient it is at performing its intended function, and a variety of network performance metrics can be used to describe various aspects of network functionality.

*Corresponding author.

Typical characterizations of real-life networked and/or distributed systems that can be found in the literature include *robustness*, *vulnerability*, *resilience*, *redundancy*, *scalability*, *connectivity*, and so on [29]. Remarkably, despite the widespread usage of these and other terms in technical literature, the quantitative or mathematical models designed *specifically* for quantification of these performance metrics are rather scarce. Most commonly, various graph-theoretical constructs have been used as proxies for the aforementioned network performance metrics. For example, *strength* $\sigma(G)$ of a graph G [15, 28] can be regarded as a measure of network’s connectivity due to the Tutte-Nash-Williams theorem, according to which $\lfloor \sigma(G) \rfloor$ is the *tree packing number* of G , i.e., the maximum number of edge-disjoint spanning trees that can be contained in G .

The goal of the present paper is to outline a general approach to construction of mathematically rigorous formulations for the aforementioned network performance metrics, and illustrate it by constructing a measure of network’s *resilience* with respect to network flow operations. We note that “resilience” and many other characterizations of network performance, including those mentioned above, have common (and implicit) context: namely, they refer to the *network’s ability to sustain operations under unforeseen future changes in operating conditions, network’s own topology, etc.* Within this common context, we can further identify the following two aspects:

- *uncertainty aspect*: the uncertainties in data and/or parameters must be explicitly taken into account to obtain a quantifiable metric of network’s performance;
- *decision aspect*: response to unforeseen changes necessarily involves decision making.

This interplay of uncertainty and decision making lends itself naturally to the framework of stochastic optimization. Of course, the stochastic optimization literature contains an extremely large volume of models and algorithms that deal with networks under uncertainties, too numerous to cite here. The focus of the present endeavor is on constructing mathematical programming models of network resilience that are methodologically meaningful and computationally tractable, and thus can fit the description of “performance metric”. Furthermore, we base our approach on analogies with methodologies and formalism of modern risk theory and risk-averse optimization.

Specifically, in this work we focus on development of metrics of *resilience of network flow operations*, where “resilience of network flow” is understood as the network’s amenability to restoring its optimal or near-optimal flow operations subsequent to unforeseen (stochastic) disruptions. Network flows, and more generally, network optimization [1, 2, 19] is an integral part of operations research discipline that studies how network systems can be successfully designed and managed. Network optimization provides the basic models that is used in transportation, supply chain management, and logistic [7, 8] and so on. In particular, the classical maximum network flow problem consists in determining the maximum volume of flow that can be sent from a given source node s to another prescribed sink node t , while the minimum cost network flow problem seeks to satisfy the system of flow demand and supply values at the network’s nodes at the lowest possible cost. The importance of these two classical network flow problems is underlined by the fact that they contain a number of other well known network problems as special cases. Among such examples are the shortest path problem, the transportation problem, and the assignment problem. The shortest path problem consists in finding the minimum cost path between two particular nodes, which is a special case of the minimum cost network problem in the case when the arc capacities are equal to one and there is just one supply node and one demand node with a unit of demand. The transportation problem features n supply nodes and m demand nodes, while the arcs in the network have unbounded capacities. The assignment problem can be viewed as a transportation problem in which every node has demand or supply of one unit. The maximum network flow problem appears as a subproblem or a building block in many operations research and computer science models, and while the first polynomial time algorithm for the maximum network flow problem was proposed by Ford and Fulkerson [24], and the problem continues to attract the attention of researchers who have since then put forward a number of efficient maximum network flow algorithms [4, 26].

As it has been mentioned, the concept of resilience is considered in the present work with respect to unforeseen disruptions in network’s operation. Exogenous uncertainties in network models, such as those affecting the overall topology, capacity, flow distribution and cost, and so on, have been studied extensively in the context of network problems, using the paradigms of stochastic optimization [5, 18, 22, 25, 32, 35, 41], robust optimization [5, 12, 46], and chance programming [17, 36, 44]. In the present work, we pursue construction of network flow resilience metrics within the framework of risk-averse stochastic optimization based on the formalism of risk measures [33].

As such, the authors view this work’s contributions to the literature as follows:

- (i) We propose the resilient maximum network flow and resilient minimum cost network flow models that quantify the network’s amenability to restoring the optimal network flow following unforeseen stochastic losses to network’s throughput capacities. The developed models are sufficiently simple to serve as “metrics” of resilience and possess a number of useful and meaningful methodological properties. The models are generally nonlinear and employ the concept of utility function for quantification of the effects of stochastic disruptions on the network performance. The rationale behind formulation of the proposed network flow resilience models follows the ideas used for construction of a class of statistical functionals known as “certainty equivalent measures of risk.”
- (ii) It is demonstrated that the developed models of resilient network flows possess properties analogous to those of a class of *convex measures of risk*, and as such can be considered as “network measures of risk”, which quantify the risk exposure of a network to a set of stochastic factors that with nontrivial interactions.
- (iii) Efficient cutting plane (Benders decomposition) methods are proposed for the proposed mathematical programming models of network flow resilience that offer substantial computational advantage over state-of-the-art off-the-shelf solvers.
- (iv) Lastly, we conduct a numerical study elucidating the effects of network topology on the resilience of maximum network flow.

The remainder of the paper is organized as follows. Section 2.1 provides the requisite background on the concepts of measures of risk that serve as a motivation for development of the models of network resilience for maximum network flow and minimum cost network flow in sections that are introduced and analyzed in the rest of this section. Section 3 presents several decomposition algorithms for the constructed mathematical programming formulations of resilient network flow models. Section 4 contains numerical studies of the computational efficiency of developed algorithms as well as an investigation of the influence of a network’s structure, or topology, on its resilience with respect to stochastic disruptions of its throughput capacity. Finally, conclusions and acknowledgements are presented in Sections 5 and 6, respectively.

2 A Risk-Based Approach to Constructing Resilience Metrics

In this section we delineate the proposed approach to constructing data-driven metrics of network performance that is based on risk-averse stochastic optimization, and in particular, uses analogies with construction of certain families of statistical functionals known as *measures of risk*, such as the *optimized certainty equivalents* [10] and *certainty equivalent measures of risk* [50].

2.1 Certainty Equivalent Measures of Risk

The concept of a *measure of risk* as a quantification of the adverse realizations of uncertainties with respect to the outcome of a decision making process traces back to the seminal work of Markowitz [37]. His idea of bi-criteria, mean-variance optimization in portfolio selection has since then flourished into the *risk-reward paradigm* of decision science, which maintains that every decision under uncertainties can be viewed in terms of a tradeoff between the risks and the rewards it incurs. The risk-reward paradigm provides a fertile modeling ground for a vast array of applications, and represents an alternative to another pillar of decision making science, the utility theory. The utility theory, originated by D. Bernoulli [11], has received its current formal axiomatic foundation in the work of von Neumann and Morgenstern [51], where exact conditions (“axioms”) on the *preferences* of a decision maker were formulated that guaranteed the existence of a utility function which equivalently described these preferences. The rigorous mathematical framework of the utility theory is one of its appealing features; a similar axiomatic approach to the risk-reward paradigm in the context of quantification of risk-averse preferences was pioneered by Artzner et al. [3] with the introduction of the *coherent measures of risk*, and was later extended to *convex measures*

of risk [23], deviation measures [43], error measures, etc. In this work, we are primarily concerned with convex measures of risk, which include the coherent measures of risk as a special case.

Given a probability space $(\Omega, \mathcal{F}, \mathbb{P})$, a measure of risk is a mapping $\rho : \mathcal{X} \mapsto \mathbb{R}$, where $\mathcal{X}$ is a space of $\mathcal{F}$ -measurable functions $X : \Omega \mapsto \mathbb{R}$, which further satisfies a number of specific properties, or axioms. Depending on whether the random element $X = X(\omega)$, $\omega \in \Omega$, represents wealth or a gain, or a cost or a loss (in other words, whether the larger or smaller realizations of $X(\omega)$ are preferable), the form of some of these axioms will change. In this work, we assume that $X(\omega)$ constitutes a loss or a cost incurred due to a random event $\omega \in \Omega$, whereby the axioms of convex measures of risk take the form

(C1) *monotonicity*: $\rho(X_1) \leq \rho(X_2)$ whenever $X_1 \leq X_2$, $X_{1,2} \in \mathcal{X}$;

(C2) *convexity*: $\rho(\lambda X_1 + (1 - \lambda)X_2) \leq \lambda\rho(X_1) + (1 - \lambda)\rho(X_2)$ for any $\lambda \in [0, 1]$ and all $X_1, X_2 \in \mathcal{X}$;

(C3) *translation invariance*: $\rho(X + a) = \rho(X) + a$ for any $a \in \mathbb{R}$, $X \in \mathcal{X}$.

The monotonicity axiom provides that lower costs or losses imply lower risks. The convexity axiom is of fundamental importance from both mathematical and methodological points of view, for it enables efficient optimization and control of risks, as well as reduction of risk via diversification, a staple strategy in risk management. In the context of the latter, (C2) guarantees that it is impossible to create a combination of stochastic factors X_1, X_2 such that its risk exceeds the corresponding individual risks, i.e., the inequality

$$\rho(\lambda X_1 + (1 - \lambda)X_2) > \max\{\rho(X_1), \rho(X_2)\}$$

is impossible if ρ is convex, but may become satisfiable if the convexity requirement of risk measure ρ is relaxed (see [21] for more discussion of the interplay of risk reduction and diversification). The translational invariance property (C3) relates to the financial setting that was considered in the original work [3], where it guaranteed that adding a certain amount of risk-free asset to a portfolio can eliminate the portfolio's risk. More generally, the translation invariance axiom postulates that adding a constant amount to a stochastic loss factor changes the risk of that factor by the same amount. Coherent measures of risk constitute a special case of convex risk measures that further are positively homogeneous,

(C4) *positive homogeneity*: $\rho(\lambda X) = \lambda\rho(X)$ for any $\lambda > 0$, $X \in \mathcal{X}$,

or, in other words, that guarantee that risk scales linearly with losses.

As it has been mentioned, in this paper we pursue the task of constructing metrics of network resilience by following the line of arguments analogous to that used in development of constructive representations for certain families of coherent and convex measures of risk, and which employ a connection with the utility theory. Specifically, we consider the *certainty equivalent measures of risk* [50]

$$\rho(X) = \min_{\eta \in \mathbb{R}} \eta + (1 - \alpha)^{-1} v^{-1} \mathbb{E} v(X - \eta)_+, \quad \alpha \in [0, 1), \quad (1)$$

which are a special case of the general representation for coherent and convex measures of risk [34, 50]

$$\rho(X) = \min_{\eta \in \mathbb{R}} \eta + \phi(X - \eta). \quad (2)$$

The rationale behind expression (1) is as follows. Let $X = X(\omega)$ be the amount of a cost or a loss that may occur sometime in the future, and suppose that right now it is possible to preallocate a fixed amount η worth of resources to be used for covering the future cost X . Then, at the time when the actual realization of X is observed, the decision maker will have to deal with the amount of losses not covered by η , namely the amount $(X - \eta)_+ = \max\{0, X - \eta\}$. In order to quantify deterministically the future uncertain cost $(X - \eta)_+$, the decision maker can employ the concept of *certainty equivalents*, well known in utility theory [51]. Namely, a rational decision maker with utility function $u : \mathbb{R} \mapsto \mathbb{R}$ will be indifferent between a random wealth/profit amount $Y = Y(\omega)$ and its corresponding deterministic, or certainty equivalent amount $\text{CE}(Y) = u^{-1}(\mathbb{E}[u(Y)])$.

Since in this work we adopt the convention that random elements represent costs/losses, one can introduce the *deutility function* $v(t) = -u(-t)$, or utility function adopted to losses. In other words, just as utility function $u(Y)$ quantifies the decision maker's satisfaction with wealth amount Y , the deutility function $v(X)$ quantifies her dissatisfaction with loss $X = -Y$. Thus, given a deutility function $v(t) = -u(-t)$, the decision maker's certainty equivalent of random cost $(X - \eta)_+$ is equal to

$$v^{-1} \mathbb{E} v(X - \eta)_+ \equiv v^{-1} \left(\mathbb{E} [v((X - \eta)_+)] \right).$$

Then the total deterministic cost associated with a future random loss X is given by

$$\eta + (1 - \alpha)^{-1} v^{-1} \mathbb{E} v(X - \eta)_+,$$

where a “safety” factor $(1 - \alpha)^{-1} \geq 1$, $\alpha \in [0, 1)$, is added. Since the choice of the preallocation amount η is at the discretion of the decision maker, the risk $\rho(X)$ of the future uncertain cost/loss X can be associated with the minimum deterministic cost as given above, leading to expression (1). The risk measures of the form (1) were introduced in [50]; similar arguments were employed by [10] in discussion of *optimized certainty equivalents*.

The conditions under which the risk measure (1) satisfies the axioms for convex or coherent measures of risk were discussed in [50]. In particular, due to the fact that loss $(X - \eta)_+$ is nonnegative, it suffices to consider the so-called one-sided deutility functions, which are also convex and nondecreasing:

$$v(t) = v(t_+) = \begin{cases} v(t) > 0, & t > 0 \\ 0, & t \leq 0 \end{cases} \quad (3)$$

Conditions that guarantee convexity of risk measure ρ in (1) can be obtained from the sufficient conditions for convexity of certainty equivalents given in [10, 38]. Namely, if function $v \in C^3(\mathbb{R})$ is strictly convex and v'/v'' is convex, then the certainty equivalent $v^{-1} \mathbb{E} v$ is also convex.

In particular, the choice of a linear deutility function $v(t) = t_+$ yields the celebrated Conditional Value-at-Risk (CVaR) measure [42],

$$\text{CVaR}_\alpha(X) = \min_{\eta \in \mathbb{R}} \eta + (1 - \alpha)^{-1} \mathbb{E}(X - \eta)_+, \quad \alpha \in [0, 1), \quad (4)$$

whereas selecting a power deutility function $v(t) = (t_+)^p$ with $p \geq 1$ leads to the Higher-Moment Coherent Measures of Risk (HMCR) [34]

$$\text{HMCR}_{p,\alpha}(X) = \min_{\eta \in \mathbb{R}} \eta + (1 - \alpha)^{-1} \|(X - \eta)_+\|_p, \quad p \in [1, \infty], \alpha \in [0, 1), \quad (5)$$

and an exponential deutility function $v(t) = \lambda^{t_+} - 1$ for $\lambda > 1$ results in the family of Log-Exponential Convex Measures of Risk (LogExpCR) [50]:

$$\text{LogExpCR}_{\lambda,\alpha}(X) = \min_{\eta \in \mathbb{R}} \eta + (1 - \alpha)^{-1} \log_\lambda \mathbb{E} \lambda^{(X - \eta)_+}, \quad \lambda > 1, \alpha \in [0, 1). \quad (6)$$

The successes of the formal axiomatic approach to development of risk averse decision making have manifested themselves in development of number of classes of axiomatically defined statistical functionals that quantify various aspects of decision making under uncertainties, such as *convex measures of risk*, *deviation measures*, *error measures*, etc., see comprehensive reviews in [33, 49], and applications ranging from finance to healthcare, energy, etc. [16, 30, 31, 47]. Next we outline an approach to constructing resilience metrics for network flow problems under uncertainties that is based on the analogy with the described above interpretation of certainty equivalent measures of risk.

2.2 Resilient Maximum Network Flow Model

In this section, we illustrate the proposed approach to construction of resilience metrics of network operations by developing a model of resilient maximum network flow. The standard deterministic maximum network flow

problem is usually formulated as follows. Let $G = (V, E)$ be a directed graph, where V is the set of nodes (vertices) and E is the set of arcs (edges) as ordered pairs (i, j) for distinct nodes $i, j \in V$. For each arc $(i, j) \in E$, let u_{ij} denote the maximum capacity of that arc, and x_{ij} denote the amount of flow along this arc. Let $A^+ \in \mathbb{R}^{|V| \times |E|}$ be the node-arc incidence matrix of the digraph G :

$$a_{ij} = \begin{cases} +1, & \text{if arc } j \text{ leaves node } i, \\ -1, & \text{if arc } j \text{ enters node } i, \\ 0, & \text{otherwise.} \end{cases}$$

Then, the maximum flow problem can be written as

$$\begin{aligned} \max \quad & \mathbf{a}_s^\top \mathbf{x} \\ \text{s. t.} \quad & A\mathbf{x} = \mathbf{0} \\ & \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \end{aligned} \tag{7}$$

where $\mathbf{x} \in \mathbb{R}^{|E|}$ is the vector of flows along the arcs in E , vector $\mathbf{a}_s^\top$ is the row of matrix A^+ corresponding to source node s , and matrix A is obtained from A^+ by removing the rows corresponding to the source node s and sink node t .

Now, assume that operational or topological disruptions within the network manifest themselves through stochastic reductions of the arc capacities:

$$\mathbf{0} \leq \mathbf{x} \leq \mathbf{u} - \mathbf{X}(\omega), \tag{8}$$

where $\mathbf{X}(\omega) = (X_{ij}(\omega))_{(i,j) \in E}$ is the vector of “capacity losses” such that

$$\mathbf{0} \leq \mathbf{X}(\omega) \leq \mathbf{u}, \quad \omega \in \Omega, \tag{9}$$

and whose joint distribution is known. We also make an assumption traditional in stochastic optimization literature that the space of random events Ω is finite, $\Omega = \{\omega_1, \dots, \omega_N\}$, where $\mathbb{P}(\omega_k) = p_k > 0$, $k = 1, \dots, N$, $\sum_{k=1}^N p_k = 1$. This assumption also allows the vector $\mathbf{X}(\omega)$ to possess a sufficient degree of integrability.

Note that the upper bound on the network flow in expression (8) is ill-defined at the moment since it does not specify which $\omega \in \Omega$ it is valid for. A well-defined description of constraints in (8) and, ultimately, a meaningful formulation of the maximum network flow problem with randomized flow bounds can be obtained in a number of different ways. Perhaps, the most straightforward and rather uninteresting solution is to require the upper bound in (8) to hold “on average”, or in expectation, which immediately yields the standard deterministic maximum network flow formulation (7) with flow bounds $\mathbf{0} \leq \mathbf{x} \leq \mathbf{u} - \mathbb{E}[\mathbf{X}(\omega)]$. Also, under certain conditions, it could be of value to consider maximization of expected flow by allowing the network flow to respond stochastically to stochastic capacity losses, $\mathbf{x} = \mathbf{x}(\omega)$, whereby the problem can be written as

$$\begin{aligned} \max \quad & \mathbb{E}[\mathbf{a}_s^\top \mathbf{x}(\omega)] \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - \mathbf{X}(\omega), \quad \omega \in \Omega. \end{aligned}$$

The above formulation is known in stochastic optimization literature as “wait-and-see” solution and, obviously, it is equivalent to solving $|\Omega|$ scenario-based deterministic maximum flow problems

$$\left. \begin{aligned} \max \quad & \mathbf{a}_s^\top \mathbf{x}(\omega) \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0} \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - \mathbf{X}(\omega) \end{aligned} \right\}, \quad \omega \in \Omega. \tag{10}$$

We will return to formulation (10) later in this section.

Among approaches that offer more significant interest and challenge, robust optimization as well as chance-constrained programming can be employed, requiring that the upper bound in (8) holds with a prescribed probability β , i.e., $\mathbb{P}(\mathbf{x} \leq \mathbf{u} - \mathbf{X}(\omega)) \geq \beta$, where $\beta = 1$ or $\beta \in (0, 1)$, respectively. In both these cases, an optimal network flow solution of the corresponding optimization problem could be characterized by its “robustness” or “reliability” with respect the randomized losses of capacity, in the sense of how well the network be able to sustain the flow (without changing it) under the conditions of reduced network capacities.

The approach proposed and pursued in this work aims at quantifying network flow’s “resilience,” generally understood as an ability to restore, perhaps partially, the flow in the network subsequent to the flow’s reduction caused by some unforeseen reductions in the network’s throughput capacities. Noting that certain expenditures or investments are typically required in order to restore functionality of a compromised system, let us suppose that it is possible to preallocate in advance $\boldsymbol{\eta} \in \mathbb{R}_+^{|E|}$ worth of resources to be used towards offsetting the observed realizations of stochastic capacity losses $\mathbf{X}(\omega)$, thereby aiding in recovery of network flow after the disruption caused by the capacity losses $\mathbf{X}(\omega)$. Upon application of the preallocated resources to negate the observed capacity losses, the corresponding realization $\mathbf{x}(\omega)$ of the flow in the network has to satisfy the constraints

$$\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega, \quad (11)$$

as well as the flow balance equations $A\mathbf{x}(\omega) = \mathbf{0}$, $\omega \in \Omega$. Then, the *resilient maximum network flow problem* can be formulated as

$$Z_\alpha^*(\mathbf{X}) = \max \quad -(1 - \alpha)\mathbf{1}^\top \boldsymbol{\eta} + u^{-1} \mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}(\omega)) \quad (12a)$$

$$\text{s. t.} \quad A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \quad (12b)$$

$$\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega \quad (12c)$$

$$\boldsymbol{\eta} \geq \mathbf{0}, \quad (12d)$$

where $\mathbf{1}$ is a vector of ones, $\mathbf{1} = (1, \dots, 1)^\top$. In the objective function (12), the total amount of capacity preallocations over the network $\mathbf{1}^\top \boldsymbol{\eta}$ includes the “discount” $\alpha \in [0, 1]$ associated with preemptive action in the face of possible future losses, which can also be viewed as an incentive to take preemptive action. Given a utility function $u(\cdot)$, the certainty equivalent term $u^{-1} \mathbb{E}u(\cdot)$ in (12a) represents the current deterministic value of a future uncertain network flow, and the term $(1 - \alpha)\mathbf{1}^\top \boldsymbol{\eta}$ is taken with a negative sign since preallocation of capacity resources prevents the utilization of these resources at the current moment.

To maintain computational tractability of model (12), we assume that the utility function u in (12) is such that the certainty equivalent $u^{-1} \mathbb{E}u(\cdot)$ is increasing and concave. It is known that for functions $u \in C^{(3)}$ that are strictly increasing and strictly concave, the certainty equivalent $u^{-1} \mathbb{E}u(\cdot)$ is concave if and only if the function u'/u'' is convex, see [9, 38]. In general, the certainty equivalent term in (12) can be replaced by a function $\Phi : \mathcal{X} \mapsto \mathbb{R}$ that is increasing and concave.

Formulation (12) can be viewed as a two-stage stochastic programming problem [13, 40], where the preallocation capacities $\boldsymbol{\eta}$ and the stochastic flows $\mathbf{x}(\omega)$ represent first- and second-stage variables, respectively. As such, problem (12) quantifies the *resilience* of network flows in G by providing a measure of the degree to which the amount network flow can be restored, through resource preallocation, after observing a stochastic reduction $\mathbf{X}(\omega)$ of capacities of the network arcs in E . This resilient maximum network flow problem has a number of properties that further illuminate its methodological and practical merits.

First, we note that the resilient maximum network flow model (12) inherits the well-known integrality property of the standard maximum network flow problem (7), namely that an optimal network flow is integer-valued if the arc capacities are integer-valued.

Proposition 2.1 (Integrality) *If both the arc capacity vector $\mathbf{u}$ and the stochastic capacity loss vector $\mathbf{X}(\omega)$ are integral for any $\omega \in \Omega$, then an optimal solution $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ of the resilient maximum network flow model (12) is also integral, provided that utility function u is linear.*

Proof: By rewriting problem (12) in the case of a linear utility function as

$$Z_\alpha^*(\mathbf{X}) = \max \quad -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \mathbb{E}[\mathbf{a}_s^\top \mathbf{x}(\omega)] \quad (13a)$$

$$\text{s. t.} \quad A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \quad (13b)$$

$$\mathbf{x}(\omega) + \mathbf{w}(\omega) \leq \mathbf{u}, \quad \omega \in \Omega \quad (13c)$$

$$\mathbf{w}(\omega) + \boldsymbol{\eta} \geq \mathbf{X}(\omega), \quad \omega \in \Omega \quad (13d)$$

$$\boldsymbol{\eta}, \mathbf{x}(\omega), \mathbf{w}(\omega) \geq \mathbf{0}, \quad \omega \in \Omega \quad (13e)$$

it is clear that the feasible region of (13) can be presented in the form

$$\{\mathbf{z} : B\mathbf{z} \leq \mathbf{b}, \mathbf{z} \geq \mathbf{0}\},$$

where $\mathbf{z} = (\boldsymbol{\eta}^\top, \mathbf{x}(\omega_1)^\top, \mathbf{w}(\omega_1)^\top, \dots, \mathbf{x}(\omega_N)^\top, \mathbf{w}(\omega_N)^\top)^\top$, and matrix B consists of blocks A , $-A$, I , $-I$, and O , where I is an identity matrix, and O is a zero matrix of appropriate dimensions. Given that matrix A is totally unimodular, matrix B can be constructed from blocks $\pm A$, $\pm I$, O using the basic operations that preserve total unimodularity, see, e.g., [39]. Hence, the vertices of polyhedron $\{B\mathbf{z} \leq \mathbf{b}, \mathbf{z} \geq \mathbf{0}\}$ are integral if $\mathbf{b}$ is integral. $\square$

The statement of Proposition 2.1 will also hold when the utility function u is strictly convex and u'/u'' is convex, i.e., such that the certainty equivalent $u^{-1}\mathbb{E}u(\cdot)$ is convex [48].

It is worth noting that an alternative formulation of resilient maximum network problem may be considered, where only the total amount M of capacity preallocations is decided in advance, to be distributed among arcs $(i, j) \in E$ in an optimal way after observing the actual realizations $X_{ij}(\omega)$ of the capacity losses in the network:

$$\begin{aligned} Z_\alpha^*(\mathbf{X}) = \max \quad & -(1-\alpha)M + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}(\omega)) \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta}(\omega))_+, \quad \omega \in \Omega \\ & \mathbf{1}^\top \boldsymbol{\eta}(\omega) = M, \quad \omega \in \Omega \\ & \boldsymbol{\eta}(\omega) \geq \mathbf{0}, \quad \omega \in \Omega \end{aligned} \quad (14)$$

In contrast to problem (12), the above formulation (14) does not preserve integrality of the optimal solution.

Proposition 2.2 (Boundedness) *Problem (12) is bounded and its optimal objective value is nonnegative:*

$$0 \leq Z_\alpha^*(\mathbf{X}) < \infty. \quad (15)$$

Proof: Note that $(\boldsymbol{\eta}', \mathbf{x}'(\omega))$, where $\boldsymbol{\eta}' = \mathbf{0}$ and $\mathbf{x}'(\omega) = \mathbf{0}$ is a feasible solution of (12) with the corresponding objective value of 0, whence $Z_\alpha^*(\mathbf{X}) \geq 0$. It is obvious that $Z_\alpha^*(\mathbf{X}) < \infty$ since $\boldsymbol{\eta} \geq \mathbf{0}$ and $\mathbf{x}$ is bounded. $\square$

Proposition 2.3 (Monotonicity with respect to α) *The optimal value of problem (12) is non-decreasing as a function of $\alpha \in [0, 1]$: $Z_{\alpha_1}^*(\mathbf{X}) \leq Z_{\alpha_2}^*(\mathbf{X})$, $\alpha_1 < \alpha_2$. Moreover, both the total capacity preallocations $\mathbf{1}^\top \boldsymbol{\eta}$ and the certainty equivalents of the network flow $u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}(\omega))$ are non-decreasing in α at optimality.*

Proof: Let $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ and $(\boldsymbol{\eta}^{**}, \mathbf{x}^{**}(\omega))$ be optimal solutions of problem (12) with $\alpha = \alpha_1$ and $\alpha = \alpha_2$, respectively, such that $\alpha_1 < \alpha_2$. Observing that $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ is a feasible solution of (12) with $\alpha = \alpha_2$, we have

$$\begin{aligned} Z_{\alpha_2}^*(\mathbf{X}) &= -(1-\alpha_2)\mathbf{1}^\top \boldsymbol{\eta}^{**} + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)) \geq -(1-\alpha_2)\mathbf{1}^\top \boldsymbol{\eta}^* + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) \\ &= -(1-\alpha_1)\mathbf{1}^\top \boldsymbol{\eta}^* + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) + (\alpha_2 - \alpha_1)\mathbf{1}^\top \boldsymbol{\eta}^* \geq Z_{\alpha_1}^*(\mathbf{X}), \end{aligned}$$

where the last inequality is due to $\alpha_2 > \alpha_1$ and $\boldsymbol{\eta}^* \geq \mathbf{0}$. Similarly, since $(\boldsymbol{\eta}^{**}, \mathbf{x}^{**}(\omega))$ is a feasible solution of (12) with $\alpha = \alpha_1$, we have

$$-(1 - \alpha_1)\mathbf{1}^\top \boldsymbol{\eta}^* + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) \geq -(1 - \alpha_1)\mathbf{1}^\top \boldsymbol{\eta}^{**} + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega))$$

and

$$-(1 - \alpha_2)\mathbf{1}^\top \boldsymbol{\eta}^{**} + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)) \geq -(1 - \alpha_2)\mathbf{1}^\top \boldsymbol{\eta}^* + u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)).$$

Rearranging terms in the above expressions, we obtain

$$(1 - \alpha_2)\mathbf{1}^\top (\boldsymbol{\eta}^{**} - \boldsymbol{\eta}^*) \leq u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)) - u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) \leq (1 - \alpha_1)\mathbf{1}^\top (\boldsymbol{\eta}^{**} - \boldsymbol{\eta}^*)$$

which immediately yields

$$(\alpha_2 - \alpha_1)\mathbf{1}^\top (\boldsymbol{\eta}^{**} - \boldsymbol{\eta}^*) \geq 0.$$

For $\alpha_1 < \alpha_2$ the latter inequality then yields $\mathbf{1}^\top \boldsymbol{\eta}^* \leq \mathbf{1}^\top \boldsymbol{\eta}^{**}$, while the former inequality provides that

$$u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) \leq u^{-1}\mathbb{E}u(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)),$$

which verifies the statements of the proposition. $\square$

Proposition 2.3 implies that the largest possible objective value in problem (12) is achieved when $\alpha = 1$, i.e., the resources $\boldsymbol{\eta}$ to counteract the losses $\mathbf{X}(\omega)$ can be preallocated for free. In such a case, obviously, one can select $\eta_i = \max_\omega X_i(\omega)$, whereby the resilient maximum flow problem (12) reduces to its deterministic version.

Corollary 2.4 *If $\alpha = 1$ and the utility function is linear, there exists an optimal solution $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ of (12) such that*

$$\eta_{ij}^* = \max_\omega X_{ij}(\omega) \quad \forall (i, j) \in E \quad \text{and} \quad \mathbf{x}^*(\omega) = \mathbf{x}^* \in \arg \max \{ \mathbf{a}_s^\top \mathbf{x} \mid A\mathbf{x} = \mathbf{0}, \mathbf{0} \leq \mathbf{x} \leq \mathbf{u} \}$$

i.e., $\mathbf{x}^*(\omega) = \mathbf{x}^*, \omega \in \Omega$, is a solution of the deterministic maximum flow problem without capacity losses (7).

Proof: When $\alpha = 1$, problem (12) with a linear utility function becomes

$$\begin{aligned} \max \quad & \mathbb{E}[\mathbf{a}_s^\top \mathbf{x}(\omega)] \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega \\ & \boldsymbol{\eta} \geq \mathbf{0}. \end{aligned} \tag{16}$$

Obviously, the optimal objective value of (16) attains its largest possible magnitude when the flow bound constraints take the form $\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u}$, $\omega \in \Omega$, which can be achieved by selecting $\eta_{ij} = \max_\omega X_{ij}(\omega)$, $(i, j) \in E$ in (16). In this case, the optimal values of network flows become $\mathbf{x}^*(\omega) = \mathbf{x}^*, \omega \in \Omega$, where $\mathbf{x}^*$ is an optimal solution of the deterministic maximum network flow problem (7). $\square$

In this case of “free” resources, the network exhibits maximum resilience, in the sense that the network flow can be completely restored to its pre-disruption levels given by solution of the deterministic problem (7). In other words, *if there is no cost to preallocating resources, any network flow is fully resilient.*

On the other hand, in the case when there is no “discount” for preallocating the resources, $\alpha = 0$, then forgoing the preallocation, $\boldsymbol{\eta} = \mathbf{0}$, is optimal, provided that the utility function is linear. An optimal flow, $\mathbf{x}^*(\omega)$, is in this case given by the solutions of the scenario based maximum flow problems with stochastic arc capacities (10):

Corollary 2.5 *If the utility function is linear, then forgoing the resource preallocation is optimal when $\alpha = 0$. Specifically, there exists an optimal solution $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ of problem (12) with $\alpha = 0$ where $\boldsymbol{\eta}^* = \mathbf{0}$ and*

$$\mathbf{x}^*(\omega) \in \arg \max \{ \mathbf{a}_s^\top \mathbf{x}(\omega) \mid A\mathbf{x}(\omega) = \mathbf{0}, \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - \mathbf{X}(\omega) \}, \quad \omega \in \Omega,$$

i.e., $\mathbf{x}^*(\omega)$ is a solution of the maximum expected flow problem with stochastic arc capacities (10).

Proof: Let $Z_0^{**}(\mathbf{X})$ and $\mathbf{x}^{**}(\omega)$ denote the optimal value and an optimal flow in problem (12) with $\alpha = 0$ in the case when capacity preallocations are not permitted, i.e., with an additional constraint $\boldsymbol{\eta} = \mathbf{0}$. Obviously, in this case one has, taking into account the assumed linearity of the utility function,

$$Z_0^{**}(\mathbf{X}) = \mathbb{E}(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)).$$

At $\alpha = 0$ and in the presence of preallocation-forbidding constraint $\boldsymbol{\eta} = \mathbf{0}$, the problem decomposes into $|\Omega|$ scenario-specific problems (10), whence

$$\mathbf{x}^{**}(\omega) \in \arg \max \{ \mathbf{a}_s^\top \mathbf{x}(\omega) \mid A\mathbf{x}(\omega) = \mathbf{0}, \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - \mathbf{X}(\omega) \}, \quad \omega \in \Omega.$$

Now, let $Z_0^*(\mathbf{X})$ and $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ denote the optimal value and an optimal solution of (12) with $\alpha = 0$ such that capacity preallocations are permitted. It is clear that $(\boldsymbol{\eta}^{**}, \mathbf{x}^{**}(\omega))$, where $\boldsymbol{\eta}^{**} = \mathbf{0}$, is a feasible solution of (12), whereby $Z_0^{**}(\mathbf{X}) \leq Z_0^*(\mathbf{X})$. Next, observe that adding $M = \mathbf{1}^\top \boldsymbol{\eta}^*$ units of capacity preallocations can increase the “no-preallocations” feasible network flow $\mathbf{a}_s^\top \mathbf{x}^{**}(\omega)$ by at most M units under any given scenario ω , from which it follows that

$$\mathbf{a}_s^\top \mathbf{x}^*(\omega) \leq \mathbf{a}_s^\top \mathbf{x}^{**}(\omega) + M, \quad \forall \omega \in \Omega.$$

Therefore, we obtain

$$\begin{aligned} Z_0^{**}(\mathbf{X}) &\leq Z_0^*(\mathbf{X}) = -\mathbf{1}^\top \boldsymbol{\eta}^* + \mathbb{E}(\mathbf{a}_s^\top \mathbf{x}^*(\omega)) \\ &\leq -M + \mathbb{E}(\mathbf{a}_s^\top \mathbf{x}^{**}(\omega) + M) = Z_0^{**}(\mathbf{X}), \end{aligned}$$

which establishes the statement of the corollary. $\square$

This observation reinforces the proposed interpretation of resilience as the capability of (at least partially) recovering the flows in a network after an unforeseen reduction in the network capacity, depending on the amount of available resources. *Without additional resources (or, as in the above case, without an incentive for preemptive allocation of resources), resilience cannot be achieved.*

2.3 Resilient Maximum Network Flow Model as a Measure of Risk

Next, we show that the optimal value $Z_\alpha^*(\mathbf{X})$ of the resilient network flow model (12) as a function of the vector of stochastic capacity losses $\mathbf{X} = \mathbf{X}(\omega)$ possesses properties analogous to those of convex measures of risk. In order to examine the properties of model (12) as a measure of risk associated with stochastic losses $\mathbf{X}(\omega)$, we make the following two observations.

Remark 2.1 Note that we explicitly imposed the nonnegativity constraints $\boldsymbol{\eta} \geq \mathbf{0}$ due to modeling considerations: as the stochastic capacity disruptions $\mathbf{X}(\omega)$ are assumed to be nonnegative, the resources to offset these disruptions should also be nonnegative. Mathematically, however, nonnegativity of $\boldsymbol{\eta}$ is not required for the problem to remain meaningful; for example, note that if $\boldsymbol{\eta} \in \mathbb{R}^{|E|}$ (12), the problem remains bounded, since as soon as for some $(i, j) \in E$ the value of η_{ij} is such that $(X_{ij}(\omega) - \eta_{ij})_+ > u_{ij}$, the problem becomes infeasible. Therefore, in the context of considering model (12) as a measure of risk, we relax the nonnegativity constraint (12d) and allow the vector of resource preallocations to take values $\boldsymbol{\eta} \in \mathbb{R}^{|E|}$.

Remark 2.2 Observe that the possibility of resource preallocations in the resilient maximum flow model (12) allows for relaxing the upper and lower bounds on the capacity losses $\mathbf{X}(\omega)$ in (9):

$$-\infty < X_{ij}(\omega) < \infty, \quad \omega \in \Omega, \quad (i, j) \in E, \quad (17)$$

in the sense that values of $X_{ij}(\omega) > u_{ij}$ for some $(i, j) \in E$ do not induce infeasibility in the resilient maximum flow problem (12), unlike in the stochastic maximum flow problem (10). Therefore, in the remainder of this section we will generally presume that the stochastic capacity reductions $\mathbf{X}(\omega)$ are unbounded as in (17).

With these two considerations in mind, it can then be shown that the optimal value of (12) is monotonic with respect to capacity losses, namely that higher capacity losses $\mathbf{X}(\omega)$ lead to lower values of $Z_\alpha^*(\mathbf{X})$.

Proposition 2.6 (Monotonicity with respect to capacity losses) *Let two realizations $\mathbf{X}^1(\omega)$, $\mathbf{X}^2(\omega)$ of stochastic capacity losses be such that*

$$\mathbf{X}^1(\omega) \leq \mathbf{X}^2(\omega) \quad \forall \omega \in \Omega, \quad (18)$$

then the corresponding optimal values of (12) satisfy

$$Z_\alpha^*(\mathbf{X}^1) \geq Z_\alpha^*(\mathbf{X}^2). \quad (19)$$

Proof: Let $P(\mathbf{X}^1)$ and $P(\mathbf{X}^2)$ denote the instances of problem (12) corresponding to the values of stochastic capacity losses $\mathbf{X}^1(\omega)$ and $\mathbf{X}^2(\omega)$, respectively. Let $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ be an optimal solution of $P(\mathbf{X}^2)$. Observe that if we show that an optimal solution of $P(\mathbf{X}^2)$ is also a feasible solution for $P(\mathbf{X}^1)$, it will prove that $Z_\alpha^*(\mathbf{X}^1) \geq Z_\alpha^*(\mathbf{X}^2)$. Indeed, since $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ is optimal for $P(\mathbf{X}^2)$, it satisfies

$$\mathbf{x}^*(\omega) + (\mathbf{X}^2(\omega) - \boldsymbol{\eta}^*)_+ \leq \mathbf{u} \quad \forall \omega \in \Omega.$$

Then from (18) it follows that

$$(\mathbf{X}^1(\omega) - \boldsymbol{\eta}^*)_+ \leq (\mathbf{X}^2(\omega) - \boldsymbol{\eta}^*)_+ \quad \forall \omega \in \Omega,$$

whereby

$$\mathbf{x}^*(\omega) + (\mathbf{X}^1(\omega) - \boldsymbol{\eta}^*)_+ \leq \mathbf{u} \quad \forall \omega \in \Omega,$$

i.e., $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ is feasible for $P(\mathbf{X}^1)$. □

Proposition 2.7 (Concavity with respect to capacity losses) *The optimal value of (12) is concave as a function of stochastic capacity losses,*

$$Z_\alpha^*(\lambda \mathbf{X}^1 + (1 - \lambda) \mathbf{X}^2) \geq \lambda Z_\alpha^*(\mathbf{X}^1) + (1 - \lambda) Z_\alpha^*(\mathbf{X}^2), \quad 0 \leq \lambda \leq 1. \quad (20)$$

Proof: This statement is a straightforward extension of the well-known fact that the optimal value of, e.g., a linear programming problem with maximization objective is concave in the right hand side vector of its constraints, see, e.g., [13]. □

Proposition 2.8 *For any $\mathbf{m} \in \mathbb{R}^{|E|}$ it holds that*

$$Z_\alpha^*(\mathbf{X} + \mathbf{m}) = -(1 - \alpha) \mathbf{1}^\top \mathbf{m} + Z_\alpha^*(\mathbf{X}) \quad (21)$$

Proof: Consider problem (12) without the nonnegativity constraint on $\boldsymbol{\eta}$:

$$\begin{aligned} Z_\alpha^*(\mathbf{X} + \mathbf{m}) &= \max_{\boldsymbol{\eta}} \quad -(1 - \alpha) \mathbf{1}^\top \boldsymbol{\eta} + u^{-1} \mathbb{E} u(\mathbf{a}_s^\top \mathbf{x}(\omega)) \\ \text{s. t.} \quad &A \mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ &\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) + \mathbf{m} - \boldsymbol{\eta})_+, \quad \omega \in \Omega \\ &\boldsymbol{\eta} \in \mathbb{R}^{|E|} \end{aligned}$$

By substituting $\boldsymbol{\eta} = \boldsymbol{\eta}' + \mathbf{m}$, we obtain

$$\begin{aligned} Z_\alpha^*(\mathbf{X} + \mathbf{c}) &= \max_{\boldsymbol{\eta}'} \quad -(1 - \alpha) \mathbf{1}^\top (\boldsymbol{\eta}' + \mathbf{m}) + u^{-1} \mathbb{E} u(\mathbf{a}_s^\top \mathbf{x}(\omega)) \\ \text{s. t.} \quad &A \mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ &\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta}')_+, \quad \omega \in \Omega \\ &\boldsymbol{\eta}' \in \mathbb{R}^{|E|} \end{aligned}$$

i.e., $Z_\alpha^*(\mathbf{X} + \mathbf{m}) = -(1 - \alpha)\mathbf{1}^\top \mathbf{m} + Z_\alpha^*(\mathbf{X})$. □

Propositions 2.6, 2.7, 2.8 imply that the quantity

$$\zeta(\mathbf{X}) = -\frac{1}{1 - \alpha} Z_\alpha^*(\mathbf{X}),$$

where $Z_\alpha^*(\mathbf{X})$ is the optimal value of resilient maximum flow problem (12) without nonnegativity constraints on the resource preallocations $\boldsymbol{\eta}$ is a convex measure of risk with respect to the vector of stochastic capacity losses, in that it satisfies the monotonicity, translation invariance, and convexity properties:

- (A1) $\zeta(\mathbf{X}^1) \leq \zeta(\mathbf{X}^2)$ for any $\mathbf{X}^1(\omega) \leq \mathbf{X}^2(\omega)$ a.s.;
- (A2) $\zeta(\lambda \mathbf{X}^1 + (1 - \lambda)\mathbf{X}^2) \leq \lambda \zeta(\mathbf{X}^1) + (1 - \lambda)\zeta(\mathbf{X}^2)$ for any $\mathbf{X}^1(\omega), \mathbf{X}^2(\omega)$ and $0 \leq \lambda \leq 1$;
- (A3) $\zeta(\mathbf{X} + \mathbf{m}) = \zeta(\mathbf{X}) + \mathbf{1}^\top \mathbf{m}$ for any $\mathbf{X}(\omega)$ and $\mathbf{m} \in \mathbb{R}^{|E|}$.

In view of the above, the optimal value $Z_\alpha^*(\mathbf{X})$ of problem (12) without nonnegativity constraints can be considered as a “network measure of risk” that quantifies the risk exposure of network’s operation (namely, the maximum network flow) with respect to a set of stochastic factors that embody the unpredictable losses of arcs’ carrying capacities.

2.4 Resilient Minimum Cost Network Flow Model

In this section we illustrate the proposed idea of constructing metrics of network resilience on the example of the minimum cost flow problem. Using the notations introduced above, the standard minimum cost network flow model can be stated as

$$\begin{aligned} \min \quad & \mathbf{c}^\top \mathbf{x} \\ \text{s. t.} \quad & A^+ \mathbf{x} = \mathbf{q} \\ & \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \end{aligned} \tag{22}$$

where, as before, A^+ is the node-arc incidence matrix, $\mathbf{u}$ is the vector of capacities of arcs in E , and $\mathbf{c}$ is the cost vector, whose elements represent the costs of sending one unit of flow along the arcs in E , and the vector $\mathbf{q}$ of supplies/demands at network’s nodes is such that $q_i > 0$ (respectively, $q_i < 0$) if i is a supply (respectively, demand) node, and $q_i = 0$ otherwise. Obviously, $\mathbf{q}$ has to satisfy

$$\mathbf{1}^\top \mathbf{q} = 0.$$

As before, assume that operational arc capacities $\mathbf{u}$ are subject to stochastic capacity losses $\mathbf{X}(\omega)$ such that

$$\mathbf{0} \leq \mathbf{X}(\omega) \leq \mathbf{u}, \quad \omega \in \Omega.$$

Following the same line of arguments as presented above, we formulate the resilient minimum cost network flow problem as follows:

$$\bar{Z}_\alpha^*(\mathbf{X}) = \min (1 - \alpha)\mathbf{c}^\top \boldsymbol{\eta} + v^{-1} \mathbb{E} v(\mathbf{c}^\top \mathbf{x}(\omega)) \tag{23a}$$

$$\text{s. t.} \quad A^+ \mathbf{x} = \mathbf{q}, \quad \omega \in \Omega \tag{23b}$$

$$\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega \tag{23c}$$

$$\boldsymbol{\eta} \geq \mathbf{0}. \tag{23d}$$

As before, $\boldsymbol{\eta}$ denotes the amount of preallocated resources to offset the uncertain capacity losses $\mathbf{X}(\omega)$, such that the network flow $\mathbf{x}(\omega)$ realized upon observing the outcome of capacity reductions $\mathbf{X}(\omega)$ has to satisfy the bounds (23c) as well as the flow balance constraints (23b). Given that the smaller values of the stochastic cost $\mathbf{c}^\top \mathbf{x}(\omega)$ of the network flow are preferred, its deterministic equivalent value is quantified using the deutility function based

certainty equivalent $v^{-1}\mathbb{E}v(\mathbf{c}^\top \mathbf{x}(\omega))$, which is to be minimized together with the discounted cost of capacity preallocations $(1 - \alpha)\mathbf{c}^\top \boldsymbol{\eta}$. It is assumed that the deutility function is such that the certainty equivalent term in the objective (23) is convex.

The resilient minimum cost network flow model (23) possesses a number of properties analogous to those of the resilient network flow model discussed in Sections 2.2–2.3. In particular, it similarly possesses the integrality of solution, provided that the arc capacities, capacity losses, and supply/demand values at the nodes are integer, and the deutility function is linear or concave.

Proposition 2.9 (Integrality) *If the arc capacity vector $\mathbf{u}$, the supply/demand vector $\mathbf{q}$, and the stochastic capacity loss vector $\mathbf{X}(\omega)$ are integral for any $\omega \in \Omega$, and the deutility function v is linear, then an optimal solution $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ of the resilient minimum network flow model (23) is also integral.*

Analogously to the resilient maximum network flow model, the optimal value of the resilient min-cost flow problem (23) is monotonic with respect to the “discount coefficient” α , but since the objective function (23) represents the cost of the network flow, it is non-increasing in α .

Proposition 2.10 (Monotonicity with respect to α) *The optimal value of problem (23) is non-increasing as a function of $\alpha \in [0, 1]$: $\bar{Z}_{\alpha_1}^*(\mathbf{X}) \geq \bar{Z}_{\alpha_2}^*(\mathbf{X})$, $\alpha_1 < \alpha_2$. In particular, both the costs of capacity preallocations $\mathbf{c}^\top \boldsymbol{\eta}^*$ and the certainty equivalents of the network flow cost $v^{-1}\mathbb{E}v(\mathbf{c}^\top \mathbf{x}^*(\omega))$ are non-increasing in α at optimality.*

According to Proposition 2.10, the smallest possible objective value in problem (23) is attained when the resources $\boldsymbol{\eta}$ for counteracting the capacity losses $\mathbf{X}(\omega)$ can be preallocated for free, $\alpha = 1$. Similarly to above, in this case one can select $\eta_i = \max_{\omega} X_i(\omega)$, whereby the resilient min-cost flow problem (23) reduces to its deterministic version.

Corollary 2.11 *If $\alpha = 1$ and the deutility function is linear, there exists an optimal solution $(\boldsymbol{\eta}^*, \mathbf{x}^*(\omega))$ of (23) such that*

$$\eta_{ij}^* = \max_{\omega} X_{ij}(\omega) \quad \forall (i, j) \in E \quad \text{and} \quad \mathbf{x}^*(\omega) = \mathbf{x}^* \in \arg \min \{ \mathbf{c}^\top \mathbf{x} \mid A\mathbf{x} = \mathbf{q}, \mathbf{0} \leq \mathbf{x} \leq \mathbf{u} \}$$

i.e., $\mathbf{x}^*(\omega) = \mathbf{x}^*$, $\omega \in \Omega$, is a solution of the deterministic minimum cost network flow problem without capacity losses (22).

In contrast to the resilient maximum flow model described in Section 2.2, in the case when there is no “discount” for preallocating the resources, $\alpha = 0$, forgoing the resource preallocation, $\boldsymbol{\eta} = \mathbf{0}$, is not optimal in general. Indeed, consider a simple network consisting of two nodes connected by an arc, such that the first node is a supply node with outflow value $q_1 = q > 0$, the second node is a demand node with inflow value $q_2 = -q$, and the capacity of the arc is $u_{12} = q$. In this situation, for any positive capacity loss $X_{12}(\omega) > 0$, zero resource preallocation $\eta_{12} = 0$ cannot be feasible and therefore optimal.

In line with the discussion in Section 2.3, constraints (9) on the capacity loss vector $\mathbf{X}(\omega)$ as well as the nonnegativity constraints on the vector $\boldsymbol{\eta}$ of capacity preallocations can be relaxed, in the sense that problem (23) remains well defined and bounded after that. In such a case, the objective value of the resilient minimum cost network flow model (23) behaves as a convex measure of risk. More formally, the quantity

$$\bar{\zeta}(\mathbf{X}) = \frac{1}{1 - \alpha} \bar{Z}_{\alpha}^*(\mathbf{X}),$$

where $\bar{Z}_{\alpha}^*(\mathbf{X})$ is the optimal value of resilient maximum flow problem (23) without nonnegativity constraints on the resource preallocations $\boldsymbol{\eta}$, is a convex measure of risk with respect to the vector of stochastic capacity losses by virtue of the fact that it satisfies the monotonicity, translation invariance, and convexity properties,

- (B1) $\bar{\zeta}(\mathbf{X}^1) \leq \bar{\zeta}(\mathbf{X}^2)$ for any $\mathbf{X}^1(\omega) \leq \mathbf{X}^2(\omega)$ a.s.;
- (B2) $\bar{\zeta}(\lambda \mathbf{X}^1 + (1 - \lambda) \mathbf{X}^2) \leq \lambda \bar{\zeta}(\mathbf{X}^1) + (1 - \lambda) \bar{\zeta}(\mathbf{X}^2)$ for any $\mathbf{X}^1(\omega)$, $\mathbf{X}^2(\omega)$ and $0 \leq \lambda \leq 1$;
- (B3) $\bar{\zeta}(\mathbf{X} + \mathbf{m}) = \bar{\zeta}(\mathbf{X}) + \mathbf{c}^\top \mathbf{m}$ for any $\mathbf{X}(\omega)$ and $\mathbf{m} \in \mathbb{R}^{|E|}$,

as evidenced by Propositions 2.12–2.14 below, where it is assumed that the capacity losses $\mathbf{X}(\omega)$ are unbounded per (17) and the preallocations $\boldsymbol{\eta}$ in (23) are unrestricted.

Proposition 2.12 (Monotonicity with respect to capacity losses) *The optimal value $\bar{Z}_\alpha^*(\mathbf{X})$ of problem (23) is non-decreasing as a function of stochastic capacity losses $\mathbf{X} = \mathbf{X}(\omega)$, $\omega \in \Omega$. Namely, if two realizations $\mathbf{X}^1(\omega)$, $\mathbf{X}^2(\omega)$ of stochastic capacity losses are such that*

$$\mathbf{X}^1(\omega) \leq \mathbf{X}^2(\omega), \quad \omega \in \Omega, \quad (24)$$

then the corresponding optimal values of (23) satisfy

$$\bar{Z}_\alpha^*(\mathbf{X}^1) \leq \bar{Z}_\alpha^*(\mathbf{X}^2). \quad (25)$$

Proposition 2.13 (Convexity with respect to capacity losses) *The optimal value $\bar{Z}_\alpha^*(\mathbf{X})$ of (23) is a convex function of stochastic capacity losses $\mathbf{X} = \mathbf{X}(\omega)$, $\omega \in \Omega$, namely*

$$\bar{Z}_\alpha^*(\lambda \mathbf{X}^1 + (1 - \lambda) \mathbf{X}^2) \leq \lambda \bar{Z}_\alpha^*(\mathbf{X}^1) + (1 - \lambda) \bar{Z}_\alpha^*(\mathbf{X}^2), \quad 0 \leq \lambda \leq 1. \quad (26)$$

Proposition 2.14 *For any $\mathbf{m} \in \mathbb{R}^{|E|}$ it holds that*

$$\bar{Z}_\alpha^*(\mathbf{X} + \mathbf{m}) = (1 - \alpha) \mathbf{c}^\top \mathbf{m} + \bar{Z}_\alpha^*(\mathbf{X}) \quad (27)$$

Similarly to the analysis in Section 2.3, the introduced quantity $\bar{\zeta}(\mathbf{X})$ can be viewed as a network measure of risk that quantifies the risk exposure of the minimum cost flow in a network with respect to a set of stochastic factors whose contributions to the overall risk picture are rather nontrivial and are dictated by the topological structure of the network.

3 Decomposition Methods for Resilient Network Flow Problems

This section proposes efficient solution methods for the introduced above resilient maximum network flow problem and resilient minimum cost network flow problem. As a general approach, we pursue the Benders decomposition techniques, due to the fact that the capacity preallocation vector $\boldsymbol{\eta}$ can be considered as a complicating variable, or, in the terminology of stochastic programming, the first-stage variable, and the network flow vector $\mathbf{x}(\omega)$ can be regarded as a second-stage variable. Below we discuss separate variants of Benders decomposition algorithm for resilient network problems with linear and nonlinear utility functions, respectively.

3.1 Resilient Maximum Flow Problem

3.1.1 Linear Resilient Maximum Flow Problem

First, let us consider the resilient maximum network flow problem (12) with a linear utility function:

$$\begin{aligned} \max \quad & -(1 - \alpha) \mathbf{1}^\top \boldsymbol{\eta} + \mathbb{E}[\mathbf{a}_s^\top \mathbf{x}(\omega)] \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega \\ & \boldsymbol{\eta} \geq \mathbf{0}. \end{aligned} \quad (28)$$

By regarding the capacity preallocations $\boldsymbol{\eta}$ as the first-stage variables and the network flows $\mathbf{x}(\omega)$ as second-stage variables in the resilient maximum network flow problem (28), for a fixed $\omega \in \Omega$ we define

$$\begin{aligned} \Theta(\boldsymbol{\eta}, \omega) = \max \quad & \mathbf{a}_s^\top \mathbf{x}(\omega) \\ \text{s. t.} \quad & A\mathbf{x}(\omega) = \mathbf{0} \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+. \end{aligned} \quad (29)$$

Since (29) represents a standard maximum network flow problem, the value of $\Theta(\boldsymbol{\eta}, \omega)$ can be computed efficiently for given values of $\boldsymbol{\eta}$ and ω ; namely, due to the maximum flow-minimum cut duality, we have that

$$\Theta(\boldsymbol{\eta}, \omega_k) = \sum_{(i,j) \in M(\boldsymbol{\eta}, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right),$$

where $M(\boldsymbol{\eta}, \omega_k) \subset E$ is the set of arcs in a minimum cut under scenario ω_k , given the preallocated capacities $\boldsymbol{\eta}$. Then the original problem (28) can be written as

$$\max \left\{ -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \mathbb{E}[\Theta(\boldsymbol{\eta}, \omega)] \mid \boldsymbol{\eta} \geq \mathbf{0} \right\},$$

which in a discrete probability space, $\mathbb{P}(\omega_k) = \pi_k > 0$, can be presented in the form

$$\max \left\{ -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \theta \mid \theta \leq \sum_{k=1}^N \pi_k \Theta(\boldsymbol{\eta}, \omega_k), \boldsymbol{\eta} \geq \mathbf{0} \right\}. \quad (30)$$

Based on (30), the master problem can be written as

$$\max \quad -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \theta \quad (31a)$$

$$\text{s. t.} \quad \theta \leq \sum_{k=1}^N \pi_k \sum_{(i,j) \in M(\boldsymbol{\eta}^v, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad v = 1, \dots, r-1 \quad (31b)$$

$$\boldsymbol{\eta} \geq \mathbf{0} \quad (31c)$$

Alternatively, a multicut version of the master problem can be formulated as

$$\max \quad -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \boldsymbol{\pi}^\top \boldsymbol{\theta} \quad (32a)$$

$$\text{s. t.} \quad \theta_k \leq \sum_{(i,j) \in M(\boldsymbol{\eta}^{\nu_k}, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad \nu_k = 1, \dots, s_k, k = 1, \dots, N \quad (32b)$$

$$\boldsymbol{\eta} \geq \mathbf{0}, \quad (32c)$$

where $\boldsymbol{\pi} = (\pi_1, \dots, \pi_N)^\top$, $\boldsymbol{\theta} = (\theta_1, \dots, \theta_N)^\top$.

Let $(\theta^r, \boldsymbol{\eta}^r)$ be an optimal solution of the master problem (31) problem at iteration r . If it holds that

$$\theta^r > \sum_{k=1}^N \pi_k \sum_{(i,j) \in M(\boldsymbol{\eta}^r, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij}^r)_+ \right), \quad (33)$$

then the cut

$$\theta \leq \sum_{k=1}^N \pi_k \sum_{(i,j) \in M(\boldsymbol{\eta}^r, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij}^r)_+ \right), \quad (34)$$

is sent to the master (31), the iteration counter is incremented, $r := r + 1$, and the master problem is resolved again. Once the condition (33) is failed, the corresponding solution $(\theta^r, \boldsymbol{\eta}^r)$ is optimal for problem (30), and, correspondingly, $\boldsymbol{\eta}^r$ is optimal for (28).

In the case of multi-cut master (32), let $(\boldsymbol{\theta}^r, \boldsymbol{\eta}^r)$ be its solution at iteration r . If there exists $k \in \{1, \dots, N\}$ such that

$$\theta_k^r > \sum_{(i,j) \in M(\boldsymbol{\eta}^r, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij}^r)_+ \right), \quad (35)$$

then one sets $s_k := s_k + 1$, $\boldsymbol{\eta}^{s_k} = \boldsymbol{\eta}^r$, and the cut

$$\theta_k \leq \sum_{(i,j) \in M(\boldsymbol{\eta}^{s_k}, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad (36)$$

is added to (32). If condition (35) is not satisfied for all $k = 1, \dots, K$, then $\boldsymbol{\eta}^r$ is optimal for (30) and (28).

3.1.2 Nonlinear Resilient Maximum Network Flow Problem

Now let us consider the case when the utility function u in (12) is such that the certainty equivalent $u^{-1} \mathbb{E}u(\cdot)$ is a concave function, whereby the resilient maximum network flow problem (12) is a convex programming problem. The described above Benders decomposition method does not readily extend to this nonlinear case, due to the fact that cuts of the form (34) explicitly rely on a linear aggregation of values $\Theta(\boldsymbol{\eta}, \omega)$. The traditional nonlinear Benders decomposition approach to convex nonlinear programming problems (see, e.g., [14]) in application to problem (12) does not provide the expected computational efficiency comparing to solving the full formulation (12) using an off-the-shelf solver. Therefore, below we present a multi-cut variation of Benders decomposition method described in the previous section that relies on max flow-min cut duality for computational efficiency.

Similarly to above, by introducing the subproblem value function $\Theta(\boldsymbol{\theta}, \omega)$ per (29), the original nonlinear problem (12) can be represented in the form

$$\max \quad -(1 - \alpha) \mathbf{1}^\top \boldsymbol{\eta} + \theta \quad (37a)$$

$$\text{s. t.} \quad \theta \leq u^{-1} \left(\sum_{k=1}^N \pi_k u(y_k) \right) \quad (37b)$$

$$y_k \leq \Theta(\boldsymbol{\eta}, \omega_k), \quad k = 1, \dots, N, \quad (37c)$$

$$\boldsymbol{\eta} \geq \mathbf{0}, \quad (37d)$$

where, as before, $\pi_k = \mathbb{P}(\omega_k)$, $k = 1, \dots, N$. Note that, in accordance with our assumptions on the utility function $u(\cdot)$, constraint (37b) is convex. Then, master problem can be formulated as

$$\max \quad -(1 - \alpha) \mathbf{1}^\top \boldsymbol{\eta} + \theta \quad (38a)$$

$$\text{s. t.} \quad \theta \leq u^{-1} \left(\sum_{k=1}^N \pi_k u(y_k) \right) \quad (38b)$$

$$y_k \leq \sum_{(i,j) \in M(\boldsymbol{\eta}^v, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad v = 1, \dots, (r-1), \quad k = 1, \dots, N \quad (38c)$$

$$\boldsymbol{\eta} \geq \mathbf{0}. \quad (38d)$$

Let $(\theta^r, \boldsymbol{\eta}^r)$ be an optimal solution of the master problem (38) problem at iteration r . If it holds that

$$\theta^r > u^{-1} \left(\sum_{k=1}^N \pi_k u \left(\sum_{(i,j) \in M(\boldsymbol{\eta}^r, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij}^r)_+ \right) \right) \right), \quad (39)$$

then the cuts

$$y_k \leq \sum_{(i,j) \in M(\boldsymbol{\eta}^r, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad k = 1, \dots, N, \quad (40)$$

are sent to the master (38), the iteration counter is incremented, $r := r + 1$, and the master problem is resolved again. Once the condition (39) is failed, the corresponding solution $(\theta^r, \boldsymbol{\eta}^r)$ is optimal for problem (37), and, correspondingly, $\boldsymbol{\eta}^r$ is optimal for (12).

3.2 Linear Resilient Minimum Cost Network Flow Problem

Similarly, we regard $\boldsymbol{\eta}$ as complicating variables in and $f_{ij}(\omega)$ as second-stage variables for the resilient minimum cost network flow problem (23). By defining

$$\begin{aligned} \Theta(\boldsymbol{\eta}, \omega) = \min \quad & \mathbf{c}^\top \mathbf{x}(\omega) \\ \text{s. t.} \quad & A^+ \mathbf{x}(\omega) = \mathbf{q} \\ & \mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \end{aligned} \quad (41)$$

problem (23) can be written as

$$\min \{ (1 - \alpha) \mathbf{c}^\top \boldsymbol{\eta} + \mathbb{E} [\Theta(\boldsymbol{\eta}, \omega)] \mid \boldsymbol{\eta} \geq \mathbf{0} \},$$

which, in a discrete probability space, $\mathbb{P}(\omega_k) = \pi_k > 0$, can be further reduced to

$$\min \left\{ (1 - \alpha) \mathbf{c}^\top \boldsymbol{\eta} + \theta \mid \theta \geq \sum_{k=1}^N \pi_k \Theta(\boldsymbol{\eta}, \omega_k), \boldsymbol{\eta} \geq \mathbf{0} \right\}. \quad (42)$$

Based on (42), the master problem can be written as

$$\min \quad (1 - \alpha) \mathbf{c}^\top \boldsymbol{\eta} + \theta \quad (43a)$$

$$\text{s. t.} \quad \theta \geq \sum_{k=1}^N \pi_k \left(\mathbf{q}^\top \mathbf{v}_k^v - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta})_+)^T \mathbf{y}_k^v \right), \quad v = 1, \dots, r-1 \quad (43b)$$

$$\boldsymbol{\eta} \geq \mathbf{0} \quad (43c)$$

Similarly to the previous subsection, a multicut version of the master can be formulated as

$$\min \quad (1 - \alpha) \mathbf{c}^\top \boldsymbol{\eta} + \boldsymbol{\pi}^\top \boldsymbol{\theta} \quad (44a)$$

$$\text{s. t.} \quad \theta_k \geq \mathbf{q}^\top \mathbf{v}_k^{v_k} - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta})_+)^T \mathbf{y}_k^{v_k}, \quad v_k = 1, \dots, s_k, \quad k = 1, \dots, N \quad (44b)$$

$$\boldsymbol{\eta} \geq \mathbf{0}, \quad (44c)$$

where, as before, $\boldsymbol{\pi} = (\pi_1, \dots, \pi_N)^\top$, $\boldsymbol{\theta} = (\theta_1, \dots, \theta_N)^\top$.

Let $(\theta^r, \boldsymbol{\eta}^r)$ be an optimal solution of (43) at iteration r . Then, if the following inequality holds,

$$\theta^r < \sum_{k=1}^N \pi_k \left(\mathbf{q}^\top \mathbf{v}_k^r - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta}^r)_+)^T \mathbf{y}_k^r \right), \quad (45)$$

where $(\mathbf{v}_k^r, \mathbf{y}_k^r)$ is an optimal solution of the dual of (41):

$$\begin{aligned} \Theta(\boldsymbol{\eta}^r, \omega_k) = \max \quad & \mathbf{q}^\top \mathbf{v} - (\mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+)^T \mathbf{y} \\ \text{s. t.} \quad & A^T \mathbf{v} - I \mathbf{y} \leq \mathbf{c} \\ & \mathbf{y} \geq \mathbf{0}, \end{aligned}$$

we add the cut

$$\theta \geq \sum_{k=1}^N \pi_k \left(\mathbf{q}^\top \mathbf{v}_k^r - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta})_+)^T \mathbf{y}_k^r \right), \quad (46)$$

to the master problem (43) and set $r = r + 1$. If condition (45) is not satisfied, then $\boldsymbol{\eta}^r$ is an optimal solution of (23).

Analogously, let $(\boldsymbol{\theta}^r, \boldsymbol{\eta}^r)$ be an optimal solution of the multicut master problem (44). If there exist $k \in \{1, \dots, N\}$ such that

$$\theta_k^r < \mathbf{q}^\top \mathbf{v}_k^r - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta}^r)_+)^{\top} \mathbf{y}_k^r, \quad (47)$$

then we set $s_k = s_k + 1$ and add the cut

$$\theta_k \geq \mathbf{q}^\top \mathbf{v}_k^r - (\mathbf{u} - (\mathbf{X}(\omega_k) - \boldsymbol{\eta}^r)_+)^{\top} \mathbf{y}_k^r \quad (48)$$

to the multicut version of the master (44); if conditions (47) are not satisfied for all $k = 1, \dots, N$, then $\boldsymbol{\eta}^r$ is an optimal solution of (44) and (23).

4 Numerical Experiments and Results

The performance of the proposed Benders decomposition algorithms was evaluated against CPLEX solver. All computations were conducted on a PC with 3.40GHz CPU and 32GB of RAM, running 64-bit Windows 10 environment. The algorithms presented in Section 3 have been coded in C++, and IBM CPLEX version 12.10 has been used to solve the complete mathematical programming formulations of the corresponding problems. Tables 1 and 3 report the computational running times on of the corresponding solution methods for various network instances within a 2,000 second time limit. The best average running times are highlighted in bold, and symbol “—” indicates a failure to find an optimal solution within the prescribed time limit.

4.1 Resilient Maximum Network Flow Problem

The numerical studies of the solution methods for the resilient maximum network flow problem were conducted on network instances of order $|V| = 96, 160, 500, 640, 960$, and 1440 from *MP-Testdata* repository [45], see [27] for details. The scenario sets with $|\Omega| = 3$ to 2000 for different problems were generated. In each scenario set, the stochastic capacity losses along the network’s arcs were generated as i.i.d. samples from the uniform $U[0, 1]$ distribution multiplied by the arc’s original capacity.

4.1.1 Linear Resilient Maximum Network Flow Problem

In the case of the resilient maximum network flow problem with a linear utility function (28), both the single-cut and multi-cut Benders decomposition algorithms were implemented in C++, and their running times were compared against the runtimes of CPLEX solver used to solve the full mathematical programming formulations of the corresponding problem instances. Table 1 presents the mean computational times, in seconds, averaged over five randomly generated instances of a problem of the given size. As it has been mentioned above, “—” represents a failure to obtain an optimal solution of the problem due to either running out of memory or exceeding the 2,000 second time limit. It follows from the results in Table 1 that both the single-cut and multi-cut versions of the Benders decomposition method offer significant computational savings over the CPLEX solver. It is also worth noting that the multi-cut variant of the Benders decomposition algorithm mostly performs better than the single-cut version on instances with a small number of scenarios, and vice versa.

4.1.2 Nonlinear Resilient Maximum Network Flow Problem With a Power Utility Function

In order to illustrate the performance of the algorithm proposed in Section 3.1 for the nonlinear convex resilient maximum network flow problem, we consider formulation (12) with a power utility function $u(t) = t^{1/p}$, $t \geq 0$,

Table 1: Average running times (in seconds) obtained by using different solution methods for the resilient maximum network flow problem with a linear utility function (28). All running times are averaged over 5 instances, and symbol “—” indicates that an optimal solution could not be found within the time limit of 2000 seconds.

$ V $	$ \Omega $	α	SINGLE-CUT	MULTI-CUT	CPLEX
98	3	0.4	1	1	14.2
	10	0.7	2	1.8	33.4
	50	0.95	9.4	8.6	143.2
	150	0.4	25.8	—	785.2
	1000	0.7	172.8	—	—
	2000	0.95	381.4	—	—
160	3	0.4	3	2.8	25.8
	10	0.7	9	8.2	47
	50	0.95	26.6	38.6	209
	150	0.4	82.2	82.4	813.2
	1000	0.7	586	—	—
500	3	0.4	42.8	32.4	162.6
	10	0.7	64.8	91.2	225.6
	50	0.95	323.6	505.6	902.8
	100	0.4	703	727.8	—
640	3	0.4	49.4	47.8	258.2
	10	0.7	151.2	148.8	339
	20	0.7	303.6	299.8	469
	100	0.95	1575.2	1599.6	—
960	3	0.4	88.2	86.2	628.6
	10	0.7	270.6	380.2	779.2
	20	0.95	1540	906.8	3210.4
	50	0.4	1512.4	—	—
1440	3	0.4	194.6	177	1840.8
	10	0.7	658	665.6	—
	20	0.95	1354	1297.4	—

where $p > 1$. Clearly, $u(t)$ in this case is an increasing concave utility function on $t \geq 0$, and the certainty equivalent $u^{-1} \mathbb{E} u(\mathbf{a}_s \mathbf{x}(\omega))$ is then a concave function of $\mathbf{x}$. The resilient maximum flow model then becomes

$$Z_{\alpha}^*(\mathbf{X}) = \max \quad -(1 - \alpha) \mathbf{1}^T \boldsymbol{\eta} + \left(\mathbb{E}[(\mathbf{a}_s^T \mathbf{x}(\omega))^{1/p}] \right)^p \quad (49a)$$

$$\text{s. t.} \quad A\mathbf{x}(\omega) = \mathbf{0}, \quad \omega \in \Omega \quad (49b)$$

$$\mathbf{0} \leq \mathbf{x}(\omega) \leq \mathbf{u} - (\mathbf{X}(\omega) - \boldsymbol{\eta})_+, \quad \omega \in \Omega \quad (49c)$$

$$\boldsymbol{\eta} \geq \mathbf{0}. \quad (49d)$$

Under a finite scenario model, $\Omega = \{\omega_1, \dots, \omega_N\}$, where $\pi_k = \mathbb{P}(\omega_k) > 0$, $\sum_{k=1}^N \pi_k = 1$, the above problem can be written in a mathematical programming form as follows

$$\begin{aligned}
Z_\alpha^*(\mathbf{X}) = \max \quad & -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \theta \\
\text{s. t.} \quad & A\mathbf{x}_k = \mathbf{0}, \quad k = 1, \dots, N \\
& \theta \leq (y_1^{1/p} + \dots + y_N^{1/p})^p \\
& y_k = \pi_k^p \mathbf{a}_s^\top \mathbf{x}_k, \quad k = 1, \dots, N \\
& \mathbf{0} \leq \mathbf{x}_k \leq \mathbf{u} - (\mathbf{X}_j - \boldsymbol{\eta})_+, \quad k = 1, \dots, N \\
& \boldsymbol{\eta} \geq \mathbf{0}.
\end{aligned} \tag{50}$$

Observe that the set

$$\{(\theta, y_1, \dots, y_N) \in \mathbb{R}_+^{N+1} : \theta \leq (y_1^{1/p} + \dots + y_N^{1/p})^p\}$$

is a convex cone in $\mathbb{R}_+^{N+1}$, and for $p > 1$ admits a lifted representation via N power cones:

$$\begin{aligned}
y_j &\geq \frac{w_j^p}{\theta^{p-1}}, \quad j = 1, \dots, N, \\
\theta &\leq w_1 + \dots + w_N,
\end{aligned} \tag{51}$$

which leads to the following reformulation of (50):

$$\begin{aligned}
Z_\alpha^*(\mathbf{X}) = \max \quad & -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \theta \\
\text{s. t.} \quad & A\mathbf{x}_k = \mathbf{0}, \quad k = 1, \dots, N \\
& \theta \leq w_1 + \dots + w_N \\
& w_k^p \leq y_k \theta^{p-1}, \quad k = 1, \dots, N \\
& y_k = \pi_k^p \mathbf{a}_s^\top \mathbf{x}_k, \quad k = 1, \dots, N \\
& \mathbf{0} \leq \mathbf{x}_k \leq \mathbf{u} - (\mathbf{X}_j - \boldsymbol{\eta})_+, \quad k = 1, \dots, N \\
& \boldsymbol{\eta} \geq \mathbf{0},
\end{aligned} \tag{52}$$

where the power cone constraints $w_k^p \leq y_k \theta^{p-1}$, $k = 1, \dots, N$, in the case of $p = 2$ become rotated second-order cone constraints

$$w_k^2 \leq y_k \theta, \quad k = 1, \dots, N,$$

and in the case $p = 3$ they can be represented by second-order cone constraints of the form

$$w_k^2 \leq v_k \theta, \quad v_k^2 \leq w_k y_k, \quad k = 1, \dots, N.$$

The corresponding formulation of master problem then becomes

$$\begin{aligned}
\max \quad & -(1-\alpha)\mathbf{1}^\top \boldsymbol{\eta} + \theta \\
\text{s. t.} \quad & \theta \leq w_1 + \dots + w_N \\
& y_k \geq \frac{w_k^p}{\theta^{p-1}}, \quad k = 1, \dots, N \\
& y_k \leq \pi_k^p \sum_{(i,j) \in M(\boldsymbol{\eta}^v, \omega_k)} \left(u_{ij} - (X_{ij}(\omega_k) - \eta_{ij})_+ \right), \quad v = 1, \dots, (r-1), \quad k = 1, \dots, N \\
& \boldsymbol{\eta} \geq \mathbf{0}.
\end{aligned} \tag{53}$$

Since problems (52) and (53) can be formulated as second-order cone programming (SOCP) problems for $p = 2$ and $p = 3$, they can be solved using a number of off-the-shelf solvers, including CPLEX. In our numerical experiments, we compared running times of solving the SOCP reformulation of problem (52) using CPLEX with

the nonlinear decomposition algorithm proposed in Section 3.1, where the master problem (53) was reformulated as a SOCP problem and solved by CPLEX. Table 2 presents the mean computational times, in seconds, averaged over five randomly generated instances of a problem of the given size. According to the results in Table 2, the nonlinear decomposition algorithm allows for substantial computational savings in comparison to solving the full mathematical programming formulation of the problem with the CPLEX solver.

Table 2: Average running times (in seconds) obtained by using different solution methods for the nonlinear resilient maximum network flow problem with a power utility function, $u(t) = t^{1/p}$, $t \geq 0$. All running times are averaged over 5 instances, and symbol “—” indicates that an optimal solution could not be found within the time limit of 2,000 seconds.

$ V $	$ \Omega $	α	$p = 2$		$p = 3$	
			MULTI-CUT	CPLEX	MULTI-CUT	CPLEX
96	3	0.4	0.8	16.4	0.8	16.6
	10	0.7	4.0	41.2	4.0	40.0
	50	0.95	13.4	215.0	14.2	222.6
	150	0.4	32.2	966.4	30.8	872.2
	1000	0.7	264.6	—	462.4	—
	2000	0.95	507.4	—	578.0	—
160	3	0.4	5.2	36.8	5.0	36.0
	10	0.7	17.2	74.6	15.0	72.8
	50	0.95	43.6	375.8	38.6	399.2
	150	0.4	144.6	—	90.2	—
	1000	0.7	602.2	—	595.6	—
500	3	0.4	36.6	173.2	34.8	183.0
	10	0.7	106.8	237.6	105.0	235.4
	50	0.95	344.0	975.4	355.2	938.8
	100	0.4	726.2	—	711.4	—
640	3	0.4	57.4	304.2	56.8	281.4
	10	0.7	161.2	341.0	159.4	343.6
	20	0.7	339.8	487.6	337.6	514.2
	100	0.95	1679.6	—	1775.0	—
960	3	0.4	502.8	854.2	493.0	839.4
	10	0.7	797.4	1507.6	775.8	1497.2
	20	0.95	1081.2	—	1035.2	—
	50	0.4	1746.8	—	1663.6	—
1440	3	0.4	380.0	—	379.4	—
	10	0.7	817.6	—	787.0	—
	20	0.95	1510.2	—	1393.8	—

4.2 Resilient Minimum Cost Network Flow Problem

Case studies of the computational approaches to the resilient minimum cost network flow problem involved network instances with $|V| = 96, 200, 400, 500, 640, 960$ nodes from the same repository. Similarly to before, scenario sets for the arc capacity losses were generated with the number of scenarios $|\Omega|$ ranging from 3 to 1000,

where the arc capacity losses were obtained as $U[0, 1]$ i.i.d. samples multiplied by the arc's original capacity value. The results of numerical experiments are reported in Table 3 for values of parameter $\alpha = 0.4, 0.7, 0.95$, where all running times are averaged over 5 randomly generated instances.

As it follows from the data in Table 3, the single-cut Benders algorithm still affords substantial computational advantage over the off-the-shelf solver CPLEX. But, unlike in the case of the resilient maximum network flow problem, the multi-cut version of the Benders method performs significantly worse than the single-cut variant, and arguably worse than CPLEX as it fails to reach an optimal solution except for the problems with smallest scenario sets.

Table 3: Average running times (in seconds) obtained by using different solution methods for the linear resilient minimum cost network flow problem. All running times are averaged over 5 instances, and symbol “—” indicates that an optimal solution could not be found within the time limit of 2000 seconds.

$ V $	$ \Omega $	α	SINGLE-CUT	MULTI-CUT	CPLEX
96	3	0.4	4.4	4.4	21.2
	10	0.7	15.2	350.4	45.8
	50	0.95	176.6	—	232.6
	150	0.4	611.8	—	942
	500	0.7	1328.6	—	—
200	3	0.4	7	5.4	46.4
	10	0.7	13.2	12.4	119.8
	50	0.95	80.8	—	934.4
	150	0.4	130.2	—	—
	500	0.7	534.4	—	—
	1000	0.95	1582	—	—
400	3	0.4	38	99	199.6
	10	0.7	148	—	1475.4
500	3	0.4	43.4	55	548.4
	10	0.7	91.2	—	—
	50	0.95	590	—	—
	100	0.4	1974.2	—	—
640	3	0.4	152.4	207	—
	10	0.7	372.2	—	—
	40	0.95	1985.2	—	—
960	3	0.4	177.2	897.6	—
	10	0.7	796.4	—	—

4.3 Analysis of the Effects of Network's Structure on Its Resilience

In this subsection we investigate the question of the effect of network's structure, or topology, on the network's resilience on the example of the resilient maximum network flow problem (12) with a linear utility function, which reduces to the form (28). To this end, we generated three types of networks, namely those having the topology of a uniform random graph, also known as Erdős-Rényi random graphs, a scale-free graph topology, and a grid network structure.

The Erdős-Rényi uniform random graph on the node set V is obtained by connecting each pair of nodes with

an arc at a constant probability $p \in (0, 1)$, independently of other arcs in the graph [20]. The uniform random graphs, despite their theoretical significance, possess a binomial degree distribution, and as such do not provide good models for many real-life networked systems, which are known to contain several nodes of very high degree with the remaining nodes having relatively few neighbors. Such networks fit well the model of scale-free, or power law networks, i.e., networks whose degree distribution follows a power law. The well-known Barabasi-Albert preferential attachment algorithm [6] represents one of the primary methods for generating scale-free graphs. Grid networks are characterized by high clustering and a small average distance length between nodes. A grid network is typically constructed by partitioning nodes into a number of subsets, such that each node is only independently connected to nodes in the same subset or the neighboring subset.

The numerical studies of the resilience of the maximum network flow were conducted on randomly generated uniform (Erdős-Rényi), scale-free (Barabasi-Albert), and grid networks with 100 nodes and the same density of about $d = 0.1$. To make sure that the instances of the resilient maximum network flow problem shown above that were solved on these network topologies were otherwise as similar as possible, the following procedure had been followed. First, a Barabasi-Albert random network was generated, its density computed, and a uniform random Erdős-Rényi network and a grid network were then generated with the same density. Arc capacities for each of the generated networks were sampled at random from the set $\{1, \dots, 20\}$.

In order to choose the source-sink node pairs in each of the constructed networks that would provide for “most comparable” maximum network flow problem instances differing primarily in the underlying network structure, the following procedure was implemented. A maximum network flow problem was solved for every pair of nodes in a network, and among the pairs with highest flow throughput in all three networks we selected the source-sink pairs that had identical maximum network flow and the same distance between them, as given by the number of arcs in the shortest path. In other words, as a result of this selection procedure, the constructed problem instances had an identical value of the deterministic maximum network flow, and the same distance between the source and sink nodes (typically, 4 to 5), such that the differences in resilience of network flow among the three networks could be attributed primarily to the underlying network structure. The described procedure of constructing a triple of network instances was repeated five times to create a sample of 5 tuples, each containing randomly generated network instances with uniform, scale-free, and grid topologies.

Next, for each tuple of network instances we generated three scenario sets of capacity losses $\mathbf{X}(\omega)$. Namely, in the first scenario set, which we call “independent”, the capacity losses were generated as i.i.d. samples from the uniform $U[0, 1]$ distribution multiplied by the deterministic capacity of the arc. The second scenario set models the circumstances in which the capacity losses in arcs increase proportionally with the degree of the arc’s origin node. In other words, if an arc leaves a high-degree node, it is likely to experience higher capacity losses than an arc leaving a low-degree node. We refer to this scenario set as “increasing”. Lastly, the third scenario set models an opposite situation, when an arc leaving a high-degree node is likely to sustain smaller capacity losses than an arc leaving a low-degree node; this scenario set is called “decreasing”. Each instance of problem (28), for every network topology and scenario of capacity losses, was solved with values of the discount factor $\alpha \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 0.95\}$, and the obtained optimal values of the “discounted” capacity preallocations $(1 - \alpha)\mathbf{1}^\top \boldsymbol{\eta}^*$ and expected network flow $\mathbb{E}[\mathbf{a}_s^\top \mathbf{x}^*(\omega)]$ were normalized by the value deterministic maximum flow in the network, corresponding to the case of $\alpha = 1$ (see Corollary 2.4). Figures 1–3 present obtained results, averaged over five randomly generated instances.

Namely, Figure 1 displays resilience characteristics of networks through the relationship between the normalized optimal discounted amount of preallocated capacity and the expected network flow in the case when the capacity losses are i.i.d. and do not depend on the node degree (scenario set “independent”). In accordance with Proposition 2.3, the optimal expected maximum flow in all three network increases with α ; note that data points in the graphs correspond to values of $\alpha = 0, 0.1, \dots, 0.95$ in the order from left to right, e.g., the leftmost datapoints correspond to $\alpha = 0$ and therefore represent the optimal expected maximum flows in problems with stochastic arc capacities and no preallocations, see Corollary 2.5. Respectively, the rightmost data points correspond to value $\alpha = 0.95$, while the value $\alpha = 1$ would result in a normalized expected optimal flow being equal to one, and discounted optimal preallocation amount equal to zero (Corollary 2.5). The graphs in Figure 1 show that for values of discount factor from $\alpha = 0$ to about $\alpha = 0.6$ the expected maximum flow in the networks with uniform,

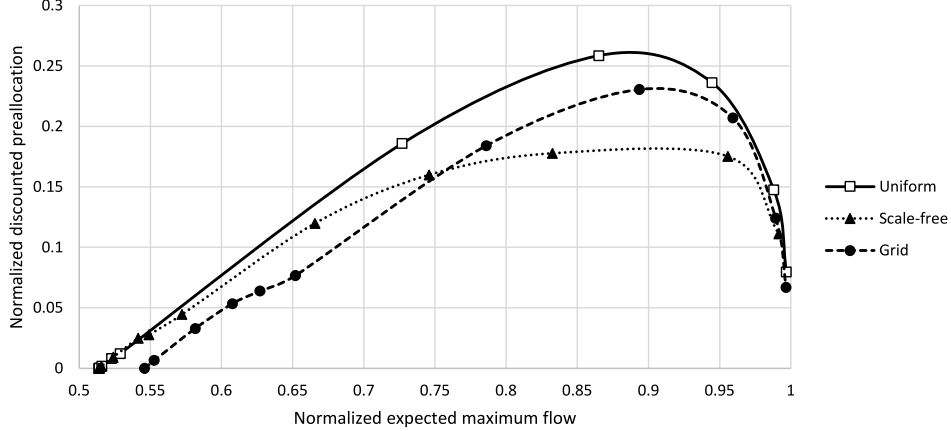


Figure 1: The normalized expected maximum flow and discounted capacity preallocation amounts in networks of different topologies under the assumption that capacity losses do not depend on the degrees of nodes. The data points correspond to values of the discount factor $\alpha = 0, 0.1, \dots, 0.9, 0.95$ in the order from left to right.

scale-free, and grid topologies grows approximately proportionately to discounted preallocation amounts, and at approximately the same rate for all three types of network structure. The grid network structure appears to be the most resilient within this range of discount factor values, as it allows for higher values of expected maximum flow at lower preallocation amounts. For values of discount factor α greater than 0.7, the scale-free network structure exhibits most resilience; in particular, it can be seen that for a certain range of discount factor values, a significant increase in the expected maximum flow can be achieved with very little increase in discounted preallocated capacity amount.

Figure 2 presents the dependency between the discounted preallocation capacity amounts and expected maximum network flow in the case when capacity losses are likely to be larger in the arcs that leave high-degree nodes, i.e., under scenario set “increasing”. Under such circumstances, clearly, the scale-free, or power-law topology of the network is the least resilient, in the sense that larger discounted preallocated capacity amounts are required for achieving a partial restoration of the expected network flow, as graphs in Figure 2 show. This outcome is not unexpected, since, in accordance with the power-law degree distribution, the scale-free network sustains higher cumulative capacity losses comparing to networks of other topologies. The network topology that is most conducive to high resilience of the maximum network flow in this case is the grid topology.

Finally, the case when random capacity losses are likely to be lower in arcs leaving high-degree nodes (scenario set “decreasing”) is presented in Figure 3. Under this scenario set, the patterns are similar to those observed under “independent” scenario set in Figure 1. Namely, scale-free network topology appears to be more conducive to achieving high resiliency when the value of the discount factor α is around 0.7 or higher. For values of α between approximately 0.3 and 0.7, the grid network topology is the most conducive to resilience of network flows; for values of the discount factor between 0 and 0.3, both the uniform and grid network structures appear to result in the same degree of resiliency.

5 Conclusions

In this work we have proposed an approach to constructing metrics of network resilience, and illustrated it on the examples of the resilient maximum network flow problem and the resilient minimum cost network problem. Specifically, the network flows in these problems are designed for resilience against unpredictable losses of network arc capacities, and the mechanism of attaining a degree of resistance is through preallocation of resources to (at least partially) restoring the capacities of the arcs. It is shown that the resulting formulations of resilient network

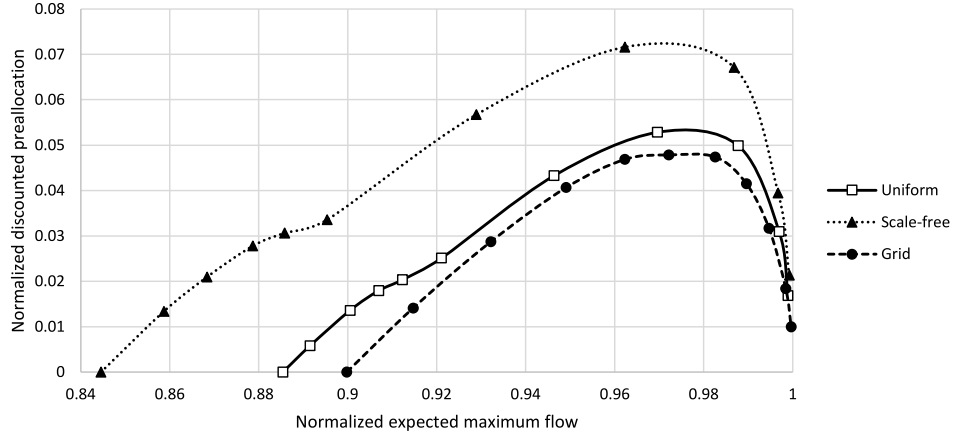


Figure 2: The normalized expected maximum flow and discounted capacity preallocation amounts in networks of different topologies under the assumption that capacity losses are likely to be higher in arcs that leave high-degree nodes. The data points correspond to values of the discount factor $\alpha = 0, 0.1, \dots, 0.9, 0.95$ in the order from left to right.

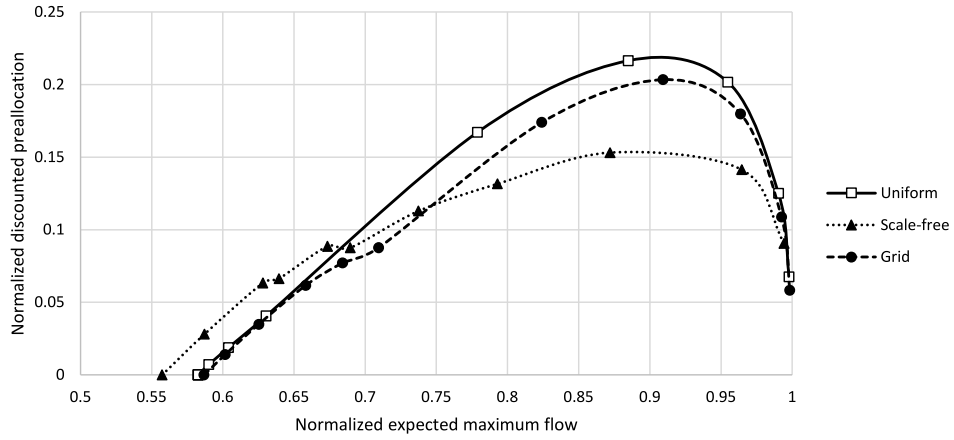


Figure 3: The normalized expected maximum flow and discounted capacity preallocation amounts in networks of different topologies under the assumption that capacity losses are likely to be lower in arcs that leave high-degree nodes. The data points correspond to values of the discount factor $\alpha = 0, 0.1, \dots, 0.9, 0.95$ in the order from left to right.

flow problems possess a number of useful properties; for instance, similarly to the standard network flow problems, the solution is integral if the arc capacities, costs, etc., are integral. It is also shown that the proposed formulations of resilient network flow problems can be viewed as “network measures of risk”, similar in properties and behavior to convex measures of risk. Efficient decomposition algorithms have been proposed for both the resilient maximum network flow problem and the resilient minimum cost network flow problem, and a study of the effects of network structure on resiliency of maximum network flow has been conducted on networks with three types of topology: that of uniform random graphs, scale-free graphs, and grid graphs.

6 Acknowledgements

This work was supported in part by AFOSR grant FA9550-19-1-0161.

References

- [1] Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993) *Network flows: Theory, Algorithms, and Applications*, Pearson, Cambridge, MA.
- [2] Ahuja, R. K., Magnanti, T. L., Orlin, J. B., and Reddy, M. (1995) “Applications of network optimization,” *Handbooks in Operations Research and Management Science*, **7**, 1–83.
- [3] Artzner, P., Delbaen, F., Eber, J.-M., and Heath, D. (1999) “Coherent measures of risk,” *Mathematical finance*, **9** (3), 203–228.
- [4] Asano, T. and Asano, Y. (2000) “Recent developments in maximum flow algorithms,” *Journal of the Operations Research Society of Japan*, **43** (1), 2–31.
- [5] Atamtürk, A. and Zhang, M. (2007) “Two-stage robust network flow and design under demand uncertainty,” *Operations Research*, **55** (4), 662–673.
- [6] Barabási, A.-L. and Albert, R. (1999) “Emergence of scaling in random networks,” *Science*, **286** (5439), 509–512.
- [7] Basten, R. J., van der Heijden, M. C., and Schutten, J. M. (2011) “Practical extensions to a minimum cost flow model for level of repair analysis,” *European Journal of Operational Research*, **211** (2), 333–342.
- [8] Basten, R. J., Van der Heijden, M. C., and Schutten, J. M. J. (2011) “A minimum cost flow model for level of repair analysis,” *International Journal of Production Economics*, **133** (1), 233–242.
- [9] Ben-Tal, A. and Teboulle, M. (1986) “Expected Utility, Penalty Functions, and Duality in Stochastic Nonlinear Programming,” *Management Science*, **32** (11), 1445–1466.
- [10] Ben-Tal, A. and Teboulle, M. (2007) “An old-new concept of convex risk measures: The optimized certainty equivalent,” *Mathematical Finance*, **17** (3), 449–476.
- [11] Bernoulli, D. (2011) “Exposition of a new theory on the measurement of risk,” in: “The Kelly capital growth investment criterion: Theory and practice,” 11–24, World Scientific.
- [12] Bertsimas, D. and Sim, M. (2003) “Robust discrete optimization and network flows,” *Mathematical programming*, **98** (1), 49–71.
- [13] Birge, J. R. and Louveaux, F. (1997) *Introduction to Stochastic Programming*, Springer, New York.
- [14] Conejo, A. J., Castillo, E., Mínguez, R., and García-Bertrand, R. (2006) *Decomposition Techniques in Mathematical Programming*, Springer, Berlin.

- [15] Cunningham, D. (1985) “Optimal attack and reinforcement of a network,” *Journal of the ACM*, **32**, 549–561.
- [16] Dehlendorff, C., Kulahci, M., Merser, S., and Andersen, K. K. (2010) “Conditional Value at Risk as a Measure for Waiting Time in Simulations of Hospital Units,” *Quality Technology & Quantitative Management*, **7** (3), 321–336.
- [17] Ding, S. (2015) “The α -maximum flow model with uncertain capacities,” *Applied Mathematical Modelling*, **39** (7), 2056–2063.
- [18] Doulliez, P. (1971) “Probability distribution function for the capacity of a multiterminal network,” *Revue française d’informatique et de recherche opérationnelle. Série verte*, **5** (V1), 39–49.
- [19] Du, D.-Z. and Pardalos, P. M. (1993) *Network optimization problems: algorithms, applications and complexity*, volume 2, World Scientific.
- [20] Erdős, P. and Rényi, A. (1960) “On the evolution of random graphs,” *Publ. Math. Inst. Hung. Acad. Sci.*, **5** (1), 17–60.
- [21] Eshghali, M. and Krokhmal, P. (2022) “Risk Reduction in Discrete Systems via Diversification and Concentration,” *Working paper*.
- [22] Feng, Y., Niazadeh, R., and Saberi, A. (2020) “Two-stage matching and pricing with applications to ride hailing,” *Available at SSRN*.
- [23] Föllmer, H. and Schied, A. (2004) *Stochastic finance: An introduction in discrete time*, Walter de Gruyter, Berlin, 2nd edition.
- [24] Ford Jr, L. R. and Fulkerson, D. R. (2015) *Flows in networks*, volume 54, Princeton university press.
- [25] Frank, H. and Frisch, I. (1970) “Analysis and design of survivable networks,” *IEEE Transactions on Communication Technology*, **18** (5), 501–519.
- [26] Goldberg, A. V. (1998) “Recent developments in maximum flow algorithms,” in: “Scandinavian Workshop on Algorithm Theory,” 1–10, Springer.
- [27] Goldfarb, D. and Grigoriadis, M. D. (1988) “A computational comparison of the Dinic and network simplex methods for maximum flow,” *Annals of Operations Research*, **13** (1), 81–123.
- [28] Gusfeld, D. (1983) “Connectivity and edge disjoint spanning trees,” *Information Processing Letters*, **16**, 87–89.
- [29] Hokstad, A., Utne, I. B., and Vatn, J. (Eds.) (2012) *Risk and Interdependencies in Critical Infrastructures: A Guideline for Analysis*, Springer.
- [30] Hosseini, S. D. and Verma, M. (2018) “Conditional value-at-risk (CVaR) methodology to optimal train configuration and routing of rail hazmat shipments,” *Transportation Research Part B: Methodological*, **110**, 79–103.
- [31] Kim, J., Lee, J., and Joo, S.-K. (2009) “Assessment of Conditional Value at Risk (CVaR) in transmission investment using Cross Entropy method,” in: “2009 Transmission Distribution Conference Exposition: Asia and Pacific,” 1–4.
- [32] Kong, N. and Schaefer, A. J. (2006) “A factor 12 approximation algorithm for two-stage stochastic matching problems,” *European Journal of Operational Research*, **172** (3), 740–746.
- [33] Krokhmal, P., Zabaranin, M., and Uryasev, S. (2011) “Modeling and optimization of risk,” *Surveys in Operations Research and Management Science*, **16** (2), 49 – 66.

- [34] Krokmal, P. A. (2007) “Higher moment coherent risk measures,” *Quantitative Finance*, **7** (4), 373–387.
- [35] Lee, E. and Singla, S. (2020) “Maximum matching in the online batch-arrival model,” *ACM Transactions on Algorithms (TALG)*, **16** (4), 1–31.
- [36] Majumder, S., Saha, B., Anand, P., Kar, S., and Pal, T. (2018) “Uncertainty based genetic algorithm with varying population for random fuzzy maximum flow problem,” *Expert Systems*, **35** (4), e12264.
- [37] Markowitz, H. M. (1952) “Portfolio Selection,” *Journal of Finance*, **7** (1), 77–91.
- [38] Müller, A. (2007) “Certainty equivalents as risk measures,” *Brazilian Journal of Probability and Statistics*, **21** (1), 1–12.
- [39] Nemhauser, G. and Wolsey, L. (1988) *Integer and Combinatorial Optimization*, John Wiley & Sons, Inc., Hoboken, NJ.
- [40] Prékopa, A. (1995) *Stochastic Programming*, Kluwer Academic Publishers.
- [41] Rebennack, S., Prokopyev, O. A., and Singh, B. (2020) “Two-stage stochastic minimum s- t cut problems: Formulations, complexity and decomposition algorithms,” *Networks*, **75** (3), 235–258.
- [42] Rockafellar, R. T. and Uryasev, S. (2002) “Conditional Value-at-Risk for General Loss Distributions,” *Journal of Banking & Finance*, **26** (7), 1443–1471.
- [43] Rockafellar, R. T., Uryasev, S., and Zabarankin, M. (2006) “Generalized Deviations in Risk Analysis,” *Finance and Stochastics*, **10** (1), 51–74.
- [44] Shi, G., Sheng, Y., and Ralescu, D. A. (2017) “The maximum flow problem of uncertain random network,” *Journal of Ambient Intelligence and Humanized Computing*, **8** (5), 667–675.
- [45] Skorobohatyj, G. (2004) “MP-Testdata,” <http://elib.zib.de/pub/mp-testdata/>, accessed: 2021-08-01.
- [46] Špoljarec, M. and Manger, R. (2021) “Solving Robust Variants of Integer Flow Problems with Uncertain Arc Capacities,” *Promet-Traffic&Transportation*, **33** (1), 77–89.
- [47] Sun, B., Krokmal, P., and Chen, Y. (2018) “Risk-Averse Capacity Planning for Renewable Energy Production,” *Energy Systems*, **9**, 223–256.
- [48] Tardella, F. (2011) “The fundamental theorem of linear programming: extensions and applications,” *Optimization*, **60** (1–2), 283–301.
- [49] Uryasev, S. and Rockafellar, R. T. (2013) “The fundamental risk quadrangle in risk management, optimization and statistical estimation,” *Surveys in Operations Research and Management Science*, **18**, 33–53.
- [50] Vinel, A. and Krokmal, P. A. (2017) “Certainty equivalent measures of risk,” *Annals of Operations Research*, **249** (1-2), 75–95.
- [51] von Neumann, J. and Morgenstern, O. (1944) *Theory of Games and Economic Behavior*, Princeton University Press, Princeton, NJ, 1953rd edition.



Chance-constrained optimization-based solar microgrid design and dispatch for radial distribution networks

Hossein Dashti¹ · Jianqiang Cheng² · Pavlo Krokhmal²

Received: 16 March 2020 / Accepted: 4 December 2020

© The Author(s), under exclusive licence to Springer-Verlag GmbH, DE part of Springer Nature 2021

Abstract

We consider a solar microgrid design and dispatch problem using an adaptive stochastic optimization framework. First, we propose a two-stage mixed-integer model for optimal placement and planning of distributed generation (DG) units and energy storage system units. We incorporate time series modeling into stochastic optimization approach to characterize the solar irradiance uncertainty. In the first stage, design decisions (e.g., location and sizing of DGs) are made and in the second stage, dispatch decisions (e.g., how much to generate, how much to store) are made such that electricity demand is met in a reliable and cost effective way. Chance constraints are employed to control the real load shedding within a predefined probability level. Then, we propose a combined sample average approximation (SAA) and linearization technique to solve this problem more efficiently. The advantage of this approach is that no additional binary variables are introduced while reformulating the chance constraints. Computational time, quality of solution, and load shedding percentage are compared with the traditional SAA. Moreover, we carry out a comprehensive out-of-sample simulation on a real-world case study in the state of Arizona assessing the effectiveness of our approach. Numerical experiments demonstrate that the chance constraints are effective tools for control of load shedding in distributed generation and the proposed approach outperforms traditional methods both in terms of true load shedding percentage and computational time.

Keywords Distributed generation (DG) · Solar photovoltaics (PV) · Chance constrained programming (CCP) · Sample average approximation (SAA) · Optimal power flow (OPF) · Energy storage systems (ESS)

✉ Pavlo Krokhmal
krokhmal@arizona.edu

Extended author information available on the last page of the article

1 Introduction and motivation

Distributed generation (also referred to as decentralized generation, or DG) of electricity from renewable energy sources is widely regarded as one of the primary strategies for achieving the goal of global energy sustainability and reducing the dependence on fossil fuels. According to the International Energy Agency, DG is defined as a generating plant serving a consumer on premise, or one providing support to a distribution network while being connected to the grid at distribution level voltages [1]. Distributed generation units are relatively smaller generating plants located at the user nodes such as households and connected to a distribution network. During last decades, distributed generation using renewable and non-renewable resources has been supported by several national and international policies to promote renewable energy integration and decrease greenhouse gas emissions. Moreover, implementation and integration of highly efficient microgrids is expected to promote diversification of energy resources, reduce peak demand and on-peak operating costs, improve voltage profile and voltage stability, reactive power control, lower energy losses, and reduce costs of energy transmission and distribution. Also, such measures are expected to promote higher-quality service for consumers. In addition, availability of DGs in modular units facilitates site location for placing of smaller generators, optimization of construction times, and reduction of capital costs [2].

Decisions on placement of DG units are normally made by the owners and investors, based upon the properties of installation site, fuel availability, and environmental conditions. The sizing and location of DG units should be chosen carefully with the goal of making a full use of DG capabilities while limiting the potentially negative effects on the overall system operation, and are usually accomplished via mixed-integer programming (MIP) methods. A number of different MIP formulations have been studied in the DG literature, including the ones aimed at reduction of energy losses, reliability improvement, voltage reduction, economic viability, and so on [3]. The interested reader is referred to Georgilakis and Hatziaargyriou [2] and Prakash and Khatod [4] for a comprehensive taxonomy of optimal DG placement models.

When it comes to efficient operations of grids with distributed generation, optimal power flow (OPF) is considered one of the core elements. Carpentier [5] introduced the OPF problem in 1962 in the context of an economic dispatch (ED) model. Since then, deterministic OPF formulations have been studied extensively in the power systems literature (see, e.g., [6–10]). Specifically, Baran and Wu [11, 12] considered placement and sizing of capacitors along with OPF equations on a radial distribution system. An OPF model and a sectioning algorithm for dispatching the capacitors and reactive outputs of DGs was given in Zhu et al. [13] and Ochoa et al. [14] studied reactive power support for distributed generation. Additionally, Moradi and Abedini [15] developed a combination of metaheuristics for optimal location and sizing of DGs in distribution systems with a similar goal of minimizing the power losses in the network. Hung and Nadarajah [16] investigated the problem of DG placement so as to reduce losses and achieve an

optimal power factor. Dall'Anese et al. [17] studied OPF model for microgrids within a semidefinite programming (SDP) formulation to solve the problem to minimize the power losses or the cost of power supplied by DG units. Levron et al. [18] incorporated energy storage systems in microgrids to obtain an optimal power flow solution and optimize total power consumption in a microgrid. Zhang et al. [19] studied the problem of voltage control in distribution networks with renewable energy resources (RERs) where they proposed a distributed algorithm for the SDP relaxation of the OPF problem. Hence, incorporating the OPF equations in microgrid design and dispatch problem with DGs is essential.

It is widely acknowledged that power system operations are affected by many uncertain factors, such as energy resources, power supply and demand, and fuel prices. On one hand, conservative decisions with high operating costs can result from an overestimation of such uncertain factors. On the other hand, an aggressive approach may yield a decision violating the system constraints. In the past few decades, a variety of techniques have been introduced to cope with uncertain factors in power systems. In particular, probabilistic approaches that model the uncertain parameters as stochastic variables with given probability density functions (PDFs) are prominently popular. None of the aforementioned works, however, took into account the uncertainty of RERs (i.e., wind speed, solar irradiance) in optimal power generation problem which can significantly impact the performance, reliability and cost effectiveness of distributed systems. There are a few studies that address this issue via stochastic optimization approaches. Atwa et al. [20] discussed a probabilistic generation-load model for optimal allocation of renewable distributed generation units to minimize annual energy loss. Wang et al. [21] employed a two-stage robust optimization based approach for optimal DG placement in microgrids for profit maximization, which used uncertainty sets for power output of wind or solar power generation instead of the corresponding probability distributions. The obtained MIP formulation was solved with a column and constraint generation algorithm. In a related work, Wang et al. [3] considered stochastic optimal DG placement with the purpose of conservation voltage reduction. They used a combined SAA and multiple replication procedure technique to solve the two-stage problem. Bazrasfshan and Gatsis [22] used a risk-averse method to optimally place and size photovoltaic (PV) inverters in distribution networks, considering the uncertainties in both solar irradiance and load. However, they solved a small-scale static problem where dispatching stage does not have any time horizon. A risk-averse capacity planning wind farm location problem involving nonlinear measures of risk was considered in Sun et al. [23].

Optimization problems under uncertainty have often been tackled by chance constrained programming (CCP), a popular method introduced by Charnes and Cooper in 1959 [24]. Within the CCP framework, randomized constraints in the system are required to be satisfied at a predefined probability level. Hence, it can be very effective in balancing the profitability and reliability of a power system [25]. Recently, different objectives have oriented the application of CCP-based methods to power systems. One example was the solution proposed by [26] and applied to a chance constrained stochastic unit commitment problem. Namely, it was required that the power load be satisfied at a prescribed probability level over

the given time horizon. Liu et al. [27] proposed a CCP-based framework to optimally site and size DG units in distribution systems under uncertain RERs, variable fuel prices and future demand growth. Chance constraints were utilized to control the current limit violations in the feeder as a security measure. To solve the CCP problem, they developed a genetic-algorithm-based approach with Monte Carlo simulation. Zhang and Li [28] introduced the CCP approach to OPF formulation under load power uncertainty and implemented a back-mapping approach and a linear approximation of the nonlinear equations in their model. By adopting their approach, Cao et al. [29] formulated a CCP-based multi-objective OPF model under wind power output uncertainty to minimize active power loss, overload and voltage violation risk with respect to several contingencies. Since it is difficult to handle joint chance constraints in the OPF problem, both studies considered individual chance constraint formulation for state variables. Jin et al. [30] presented a multi-criteria optimization model to design a DG system with wind turbines under power intermittency and load uncertainty. A genetic algorithm was used to obtain optimal design and maintenance schedules where the total loss of load probability was controlled with an individual chance constraint. Recently, Hejazi and Mohsenian-Rad [31] proposed a non-parametric CCP-based optimization approach for planning of energy storage systems (ESS) in power distribution networks. Chance constraints were leveraged to control the current and voltage limit violations. In this paper, we formulate a stochastic microgrid design and dispatch problem with OPF and ESS under RER's uncertainty for radial distribution networks as a multi-season two-stage joint chance constrained optimization problem. The main goal is to control real power load shedding to increase system reliability. The main contributions of this work can be summarized as follows:

1. We propose a two-stage solar stochastic microgrid design and dispatch problem with OPF and ESS where the first stage corresponds to siting and sizing of PV and ESS units, and the second stage represents generation planning with multi-season horizon.
2. Uncertainty and variability of PV outputs are characterized using seasonal autoregressive integrated moving average (SARIMA) models.
3. Chance constrained programming is employed to control load shedding in distributed generation.
4. We develop a combined SAA and linearization technique to solve the problem efficiently.

The remainder of this paper is organized as follows. A two-stage stochastic microgrid design and dispatch model is presented in Sect. 2. We employ chance constraints to control real power load shedding (i.e., real power shortage) probability to increase power system reliability. Additionally, the ARIMA modeling is used to characterize the PV output uncertainty. In Sect. 3, the combined SAA and linearization technique is developed to solve the problem. Then, in Sect. 4 numerical experiments are conducted using data from a real-life case study in the state of Arizona. Finally, Sect. 5 discusses future work and concludes the paper.

2 Mathematical formulation

2.1 Nomenclature

A. Sets

$\mathcal{N}^+ = \{1, \dots, N\}$	: the set of all nodes except substation (i.e., node 0);
$\mathcal{T} = \{1, \dots, T\}$	: the set of all time periods;
$\mathcal{T}_s = \{1, \dots, T_s\}$	: the set of time periods in the season s ;
$\mathcal{B} = \{1, \dots, B\}$	: the set of all years in the planning horizon;
$\mathcal{S} = \{1, \dots, S\}$	: the set of all seasons;
$\mathcal{M} = \{1, \dots, M\}$	: the set of all scenarios;
$\mathcal{A}_i$	: the ancestor node of the node i ;
$\mathcal{C}_i$	: the set of child nodes of the node i ;
$\mathcal{C}_0$	: the set of child nodes of the substation;

B. Parameters

B^{PV}	: the maximum number of available Photovoltaic (PV) units;
$a^{s_1}/a^{s_2}/a^{s_3}$	: the small/medium/large size modules for the PV units;
c_i^{setup}	: the setup cost of installing a PV unit at the node i ;
c_i^{area}	: the area cost of installing PV unit at the node i ;
c_i^{ESS}	: the capacity cost of installing an energy storage system (ESS) at the node i ;
c_i^{MT}	: the fuel cost of generation using a micro-turbine (MT) at the node i ;
c_i^{LS}	: the penalty of load shedding;
c_b	: the annual cost factor for the year b ;
p_i^{max}	: the maximum real power output of the micro-turbine i ;
RU_i/RD_i	: ramp-up/ramp-down rate of a micro-turbine at the node i ;
R_i/X_i	: the resistance/reactance on the line i ;
v_0	: the nominal substation voltage;
γ	: the small voltage deviation threshold;
$\tilde{h}_{it}$	: the solar irradiance at the node i at the period t ;
p_{it}^c/q_{it}^c	: the real/reactive power load (demand) at the node i at the period t ;
r_i	: the solar panel yield or efficiency installed at node i ;
η	: the ESS charge/discharge rate;
IC^{min}/IC^{max}	: the minimum/Maximum capacity for the ESS units.
q_i^s	: the reactive power injected by the shunt capacitor at the node i ;
α, β	: the shape parameters of Beta distribution for solar irradiance;
μ, σ	: the mean and standard deviation of Beta distribution for solar irradiance;
δ	: the threshold used in the chance constraints;

C. Decision Variables

First stage

- $x_i^{s1}/x_i^{s2}/x_i^{s3}$: the binary variables equal to one if the PV unit installed at the node i is small/medium/large, zero otherwise;
 IC_i : the ESS inventory capacity installed at the node i ;
 u_i : the area (size) of PV unit installed in node i ;

Second stage

- p_{it}^g/q_{it}^g : the real/reactive power generated by the PV unit at the node i at the period t ;
 v_{it} : the magnitude of voltage phasor at the node i at the period t ;
 l_{it} : the magnitude of current phasor on the line i at the period t ;
 P_{it}/Q_{it} : the real/reactive power flow into the node i at the period t ;
 I_{it} : the ESS storage level at the node i at the period t ;
 z_{it} : the real load shedding (≥ 0) or the real power curtailment (≤ 0) at the node i at the period t ;
 Z_{it} : the real load shedding at the node i at the period t .

We present the two-stage stochastic microgrid design and dispatch problem as follows:

$$\min \sum_{i \in \mathcal{N}^+} (c_i^{setup}(x_i^{s1} + x_i^{s2} + x_i^{s3}) + c_i^{area}(x_i^{s1}a^{s1} + x_i^{s2}a^{s2} + x_i^{s3}a^{s3}) + c_i^{ESS}IC_i) + E[f(x, IC, \tilde{h})] \quad (1a)$$

$$\text{s.t. } x_i^{s1} + x_i^{s2} + x_i^{s3} \leq 1, \quad \forall i \in \mathcal{N}^+, \quad (1b)$$

$$\sum_{i \in \mathcal{N}^+} x_i^{s1} + x_i^{s2} + x_i^{s3} \leq B^{PV}, \quad (1c)$$

$$(x_i^{s1} + x_i^{s2} + x_i^{s3})IC^{min} \leq IC_i \leq (x_i^{s1} + x_i^{s2} + x_i^{s3})IC^{max}, \quad \forall i \in \mathcal{N}^+, \quad (1d)$$

$$x_i^{s1}, x_i^{s2}, x_i^{s3} \in \{0, 1\}, \quad IC_i \geq 0, \quad \forall i \in \mathcal{N}^+, \quad (1e)$$

where $f(x, IC, \tilde{h})$ represents the operating cost for given PV and ESS installation, capacity (i.e., $x_i^{s1}, x_i^{s2}, x_i^{s3}, IC_i$), and realized solar irradiance h_{it} which is equal to the optimal objective function value of the following problem:

$$f(x, IC, \tilde{h}) = \min \sum_{b \in \mathcal{B}} c_b \left(\sum_{t \in T} \sum_{i \in \mathcal{N}^+} c_i^{MT} p_{it}^{MT} + c^{LS} Z_{it} \right) \quad (2a)$$

$$\text{s.t. } \sum_{j \in \mathcal{C}_i} (P_{jt} + R_j l_{jt}) = P_{it} - p_{it}^c + p_{it}^{PV} - (I_{i,t} - I_{i,t-1}) + p_{it}^{MT} + z_{it}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2b)$$

$$\sum_{j \in \mathcal{C}_i} (Q_{jt} + X_j l_{jt}) = Q_{it} - q_{it}^c + q_{it}^{PV} + q_i^s, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2c)$$

$$\sum_{j \in \mathcal{C}_0} (P_{jt} + R_j l_{jt}) = P_{0t}, \quad \forall t \in \mathcal{T}, \quad (2d)$$

$$\sum_{j \in \mathcal{C}_0} (Q_{jt} + X_j l_{jt}) = Q_{0t}, \quad \forall t \in \mathcal{T}, \quad (2e)$$

$$v_{it} + (R_i P_{it} + X_i Q_{it}) / v_0 = v_{\mathcal{A}_t}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2f)$$

$$-\gamma^2 \leq v_{it} - v_0 \leq \gamma^2, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2g)$$

$$p_{it}^{PV} = (x_i^{s_1} a^{s_1} + x_i^{s_2} a^{s_2} + x_i^{s_3} a^{s_3}) \tilde{h}_{it} r_i, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2h)$$

$$I_{it} \leq IC_i, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2i)$$

$$-\eta IC_i \leq I_{i,t+1} - I_{it} \leq \eta IC_i, \quad \forall i \in \mathcal{N}^+, t = 1, \dots, T-1, \quad (2j)$$

$$p_{it}^{MT} \leq p_i^{max}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2k)$$

$$-RD_i \leq p_{i,t+1}^{MT} - p_{it}^{MT} \leq RU_i, \quad \forall i \in \mathcal{N}^+, \forall t = 1, \dots, T-1, \quad (2l)$$

$$Z_{it} = \max\{0, z_{it}\}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}, \quad (2m)$$

$$p_{it}^{MT}, p_{it}^{PV}, q_{it}^s, v_{it}, l_{it}, P_{it}, Q_{it} \geq 0, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}. \quad (2n)$$

In formulation (1)–(2) we seek to minimize the total capital cost, such as solar panel and energy storage system installation and capacity cost, as well as the expected multi-year operating cost, including the micro-turbine generation cost and the real load shedding penalty $f(x, IC, \tilde{h})$. Constraints (1b) restrict the investor to choosing only one size option for nodes where a PV is going to be installed. Constraint (1c) represents the budget on the total number of PV installations. Installation of the ESS units in the node i depends on the PV installation through constraints (1d). In other words, no standalone ESS installation is allowed in this model. Integrality of the first stage variables is declared by constraints (1e). Constraints (2b)–(2f) are a DC relaxation of AC power flow (i.e., DistFlow [12]) equations in

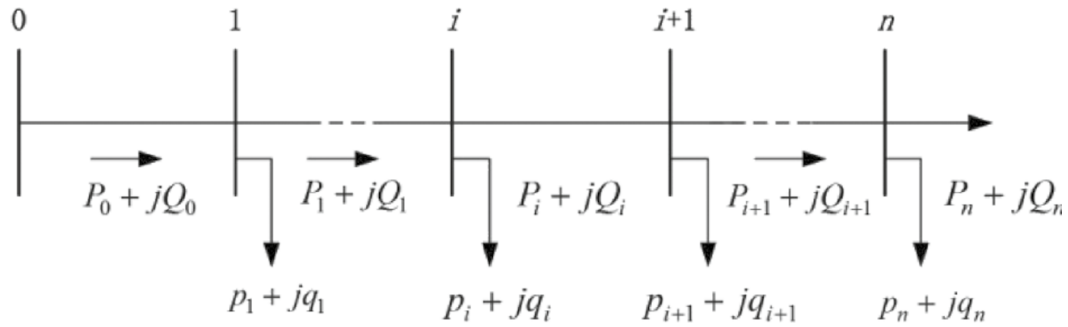


Fig. 1 Diagram of a radial distribution network. *Source:* [33]

distributed systems. Note that p_{it}^{PV} is the power generated by PV units which are subject to uncertainties. Also, q_{it}^{PV} is reactive power generated by the PV and q_i^s is reactive power injected by var compensation devices, such as shunt capacitors. Additionally, p_{it}^{MT} is the power generated by auxiliary micro-turbines and $(I_{i,t} - I_{i,t-1})$ is power charge (if nonnegative) or discharge (if nonpositive) of the ESS units. Without loss of generality, we only consider real load shedding in this network. The nonlinear terms in the OPF equations represent the losses and they tend to be negligible comparing to the branch power terms P_i and Q_i in practice. We refer the reader to Turitsyn et al. [32] where the authors show that replacing the nonlinear terms with linear equations has minimal impact on the results. Hence, the linearized DistFlow equations are widely used and justified for distributed networks and microgrids in the literature [3, 21, 33–35]. A one line diagram of radial network is depicted in Fig. 1. It is worth mentioning that one can convert a meshed network to a radial network by adding dummy nodes and breaking the loops. Also, v_0 is the magnitude of the voltage phasor at the substation and is fixed to 1 p.u. Constraints (2g) guarantee that the voltage deviation along the feeder is controlled in a reasonable range and remains close to nominal substation voltage at all time, where γ is typically 0.03 to 0.05 p.u. Constraints (2h) represent simplified PV power output equations based on PV panel size and solar irradiance which are adopted from [36, 37], and [38]. Constraints (2i) [respectively, (2j)] describe bounds of stored power (respectively, charge and discharge) for the ESS units. Constraints (2l) show the ramp-rate limitation of micro-turbines. Real load shedding and power curtailment relationship is presented in (2m). Finally, nonnegativity constraints of second stage variables are presented in (2n).

In this paper, mathematical programming formulations (1) and (2) are called the first-stage and the second-stage problems, respectively. Note that the second-stage problem becomes a linear optimization problem when the first-stage variables (x, IC) and random variables $\tilde{h}$ are treated as parameters.

2.2 Chance constraints for load shedding

Since the solar irradiance $\tilde{h}$ is random, the solar output variables are random as well. Therefore, it is possible that extremely high costs may be incurred to have the real power supply-demand balance constraints in (2b) satisfied exactly. For example, in

practice purchasing slack real power from utilities at node 0 where the network is connected to a grid would be one option. In order to control the real power load shedding, we employ chance constraints and examine its effectiveness on a case study. Let us use auxiliary variables $u_i = (x_i^{s_1} a^{s_1} + x_i^{s_2} a^{s_2} + x_i^{s_3} a^{s_3})$ to represent size of PV units installed. Then, solar output equation (2h) would be equivalent to $p_{it}^{PV} = u_i \tilde{h}_{it} r_i$, $\forall i \in \mathcal{N}^+, t \in \mathcal{T}$. Now, if we replace this in equations (2b) we get

$$\sum_{j \in \mathcal{C}_i} (P_{jt} + R_j l_{jt}) = P_{it} - p_{it}^c + u_i \tilde{h}_{it} r_i - (I_{i,t} - I_{i,t-1}) + p_{it}^{MT} + z_{it}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}. \quad (3)$$

For brevity of notation, we introduce variables O_{it} to replace

$$\sum_{j \in \mathcal{C}_i} (P_{jt} + R_j l_{jt}) - P_{it} + (I_{i,t} - I_{i,t-1}) - p_{it}^{MT}.$$

Then, equations (3) will take the form

$$u_i \tilde{h}_{it} r_i - p_{it}^c + z_{it} = O_{it}, \quad \forall i \in \mathcal{N}^+, t \in \mathcal{T}. \quad (4)$$

Note that $z_{it} \geq 0$ represents load shedding while $z_{it} \leq 0$ shows power curtailment. Therefore, joint chance constraints to control real load shedding with a given probability level can be defined as

$$\Pr \{u_i \tilde{h}_{it} r_i - p_{it}^c - O_{it} \geq 0, \forall i \in \mathcal{N}^+, t \in \mathcal{T}\} \geq 1 - \delta, \quad \delta \in (0, 1). \quad (5)$$

In other words, using chance constraints we can control the probability of real load shedding to be less than δ .

2.3 Uncertainty of solar irradiance and a SARIMA modeling approach

It is commonly assumed in the literature [3, 20, 39] that a finite number M of scenarios representing the possible solar irradiance realizations characterizes the uncertainty of $\tilde{h}$ in a distribution network. More precisely, at node $i \in \mathcal{N}^+$ at time period t under scenario $m \in \mathcal{M}$, solar irradiance will take values h_{itm} . Moreover, assume the probability of occurrence for scenario m to be π_m , such that $\sum_{m \in \mathcal{M}} \pi_m = 1$, $0 \leq \pi_m \leq 1$.

It is also often assumed that solar irradiance in each time period follows a Beta distribution [3, 39–41]:

$$f(h_{it}) = \frac{h_{it}^{\alpha-1} (1 - h_{it})^{\beta-1}}{B(\alpha, \beta)}, \quad (6a)$$

$$\mu_{it} = \frac{\alpha}{\alpha + \beta}, \quad (6b)$$

$$\sigma_{it}^2 = \frac{\alpha\beta}{(\alpha + \beta)^2(\alpha + \beta + 1)}, \quad (6c)$$

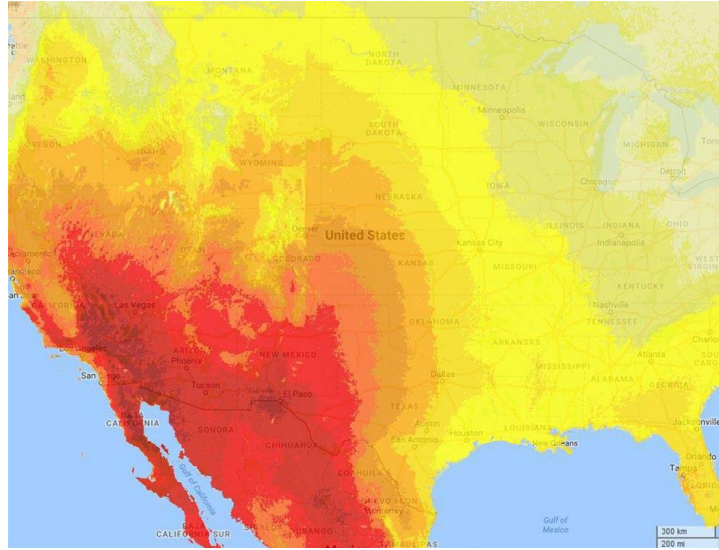


Fig. 2 Solar prospector for direct normal irradiance (DNI) in USA

where $B(\alpha, \beta)$ is the Beta function with parameters α and β .

However, in order to model the uncertainty of solar irradiance more accurately, one can employ adaptive forecasting techniques such as time series instead of assuming a distribution for the corresponding random variables, which can be statistically derived from the historical data of solar irradiance. For example, following [42, 43], and [44], we can construct a predictive seasonal ARIMA model [45] for h as follows. Let $\phi(L)$ be the autoregressive operator as a polynomial in the lag operator: $\phi(L) = 1 - \phi_1 L - \dots - \phi_p L^p$, and let $\Phi(L)$ be the seasonal autoregressive operator. Additionally, let $\theta(L)$ be the moving average operator: $\theta(L) = 1 + \theta_1 L + \dots + \theta_q L^q$, and let $\Theta(L)$ be the seasonal moving average operator. Let superscripts d and D denote the order of differencing and seasonal differencing, respectively. Lastly, let S represent the time span of cyclical patterns; for example $S = 24$ in hourly solar irradiance data. The general SARIMA model for h at a user node would be of the form:

$$(1 - L)^d (1 - L^S)^D h_t = (\theta_t(L) \Theta_t(L) / \phi_t(L) \Phi_t(L)) \varepsilon_t, \quad (7a)$$

where ε_t denotes the error term or white noise such that $\varepsilon_t \sim N(0, \bar{\sigma}_t^2)$.

$\text{ARIMA}(p, d, q) \times (P, D, Q)_S$ is a short hand notation for the above model. The model coefficients and variance of the white noise (i.e., $\bar{\sigma}_t^2$) can be estimated using maximum likelihood techniques. We used the hourly direct normal irradiance (DNI) data from the solar prospector by the National Renewable Energy Laboratory (NREL) [46] for a network in Arizona as visualized in Fig. 2. We apply the statistical package R [47] to the 3-years data to estimate the aforementioned coefficients and the variance of error terms in the SARIMA model. The models for the hourly data were fit as $\text{ARIMA}(1, 0, 0)(0, 1, 0)_{24}$. In other words, they are differenced at the 24 hour time period, but not first hour differenced. This fit was selected with the coefficient of determination $R^2 = 0.96$ which represents a good fit in practice [48]. Figure 3 shows a typical segment (a week) of the normalized series using the fitted

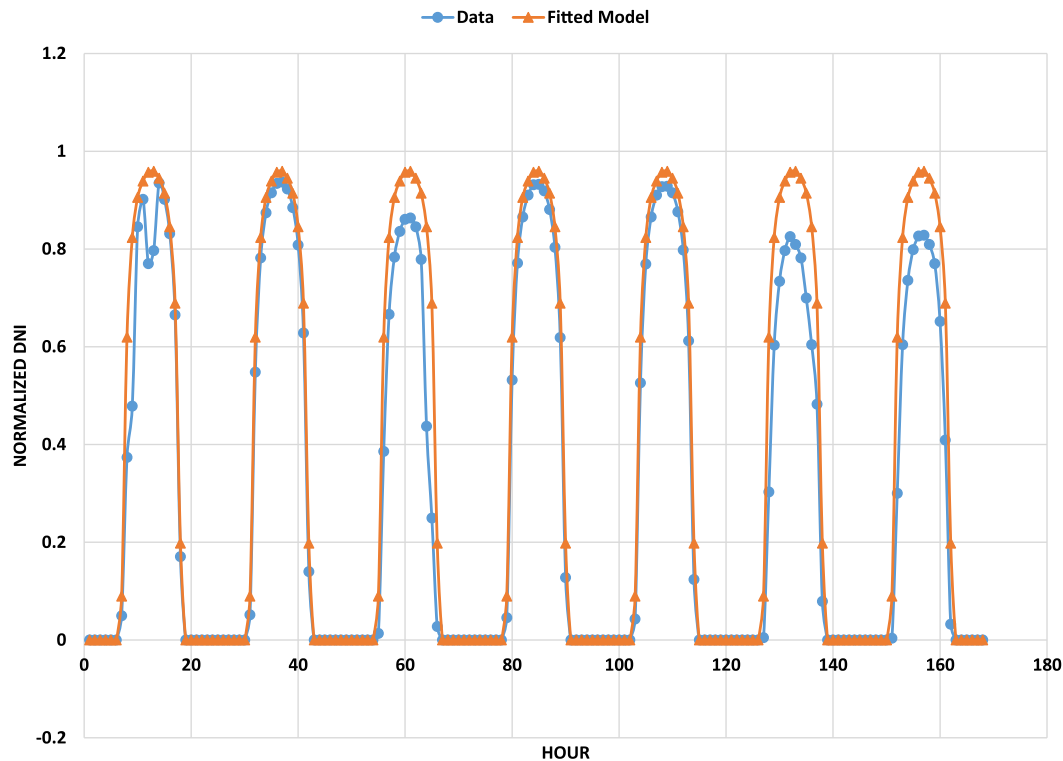


Fig. 3 The ARIMA model fit for hourly DNI data

model. The model to predict solar irradiance at period t for a node i can be written as

$$h_{it} = h_{i,t-24} - \theta(h_{i,t-25} - h_{i,t-1}) + \varepsilon_{it}. \quad (8)$$

In this case, one can obtain estimates of the coefficients from the SARIMA models to generate solar irradiance scenarios at different nodes.

3 Solution approach

Formulation (1)–(2) combined with chance constraints (5) is impractical to solve directly. Thus, in this section we develop a combined SAA and linearization technique to solve the two-stage chance-constrained model more efficiently. Following [20] and [49], four seasons are chosen to represent a year of study, and a typical day is selected for each season to capture the uncertainty of the solar irradiance during that season. The day representing each season contains 24 time periods, each corresponding to a specific hourly interval for the whole season. Hence, each year has 96 time periods. Furthermore, we follow [50] and aggregate the load-generation for each season to lower the quantity of chance constraints to be satisfied as well as to conserve the relationship between supply and demand. Obviously, with this process the obtained solution to OPF will be an

approximation, but it will make it possible to solve the problem within a reasonable time. As a result, the aggregated chance constraints for season $s \in \mathcal{S}$ are as follows

$$\Pr \{u_i \tilde{h}_{si} r_i - p_{si}^c - O_{si} \geq 0, \forall i \in \mathcal{N}^+, \forall s \in \mathcal{S}, \quad (9)$$

where $\tilde{h}_{si}$, p_{si}^c , and O_{si} represent aggregated solar irradiance, real load, and power flow at node i for season s , respectively.

A chance constrained optimization problem can be solved by a number of approaches. The SAA [51] is proven to be an effective method with applications in unit commitment [52] and distributed generation planning [3]. The idea behind SAA is to approximate the expectation term in the objective function (1a) by its sample average. When the sample size is large enough, according to the law of large numbers, the value of the approximation converges to the value of the original quantity. Meanwhile, the optimal solution set of the reformulated problem would be same as the one of the original problem. One popular way to reformulate chance constraints (9) with SAA is using the “big-M” method. This approach requires introduction of binary variables y_m for each scenario $m \in \mathcal{M}$. Then, one can replace chance constraints (9) with following inequalities in problem (2) for season $s \in \mathcal{S}$ to control the real load shedding probability:

$$z_{sim} \leq M^{big} y_{sm}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \forall m \in \mathcal{M}, \quad (10a)$$

$$\frac{\sum_{m \in \mathcal{M}} y_{sm}}{M} \leq \delta, \quad \forall s \in \mathcal{S}, \quad (10b)$$

$$y_{sm} \in \{0, 1\}, \quad \forall s \in \mathcal{S}, \forall m \in \mathcal{M}, \quad (10c)$$

where M^{big} is a sufficiently large number. With this method, one can generate samples $\{\hat{h}_{sim}\}_{m \in \mathcal{M}}$ of $\tilde{h}_{si}$ from the ARIMA model in (8). The drawback of this technique is the increasing number of binary variables as the number of scenarios increases to get a better approximation of the expected value in objective function. Also, the choice of M^{big} value can significantly affect computational time due to possibly poor properties of MIP relaxations of (10). In this work, we use (10) as an example and benchmark for the proposed combined SAA and linearization technique which is free of such binary variables and M^{big} .

Assuming independence among solar irradiance in different nodes and introducing auxiliary variable w_{si} , we follow [53] to rewrite chance constraints (9) as

$$\Pr \{u_i \tilde{h}_{si} r_i - p_{si}^c \geq O_{si}\} \geq (1 - \delta)^{w_{si}}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (11a)$$

$$\sum_{i \in \mathcal{N}^+} w_{si} = 1, \quad \forall s \in \mathcal{S}, \quad (11b)$$

$$w_{si} \geq 0, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+. \quad (11c)$$

With the seasonal aggregation $\tilde{h}_{si} = \sum_{t \in \mathcal{T}_s} \tilde{h}_{it}$, $\forall s \in \mathcal{S}$, one can show that $\tilde{h}_{si} \sim N(\bar{\mu}_{si}, \bar{\sigma}_{si})$, where $\bar{\mu}_{si}, \bar{\sigma}_{si}$ can be calculated from the ARIMA estimates of (8) based on the fact that sum of Normal random variables follows a Normal distribution as well. Therefore, we can rewrite the chance constraints (11a) as

$$\Pr \left\{ \frac{-u_i \tilde{h}_{si} r_i + u_i \bar{\mu}_{si} r_i}{u_i \bar{\sigma}_{si} r_i} \leq \frac{-p_{si}^c - O_{si} + u_i \bar{\mu}_{si} r_i}{u_i \bar{\sigma}_{si} r_i} \right\} \geq (1 - \delta)^{w_{si}}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (12)$$

which is equivalent to

$$F \left(\frac{-p_{si}^c - O_{si} + u_i \bar{\mu}_{si} r_i}{u_i \bar{\sigma}_{si} r_i} \right) \geq (1 - \delta)^{w_{si}}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+,$$

where $F(\cdot)$ is the cumulative distribution function (CDF) of the standard normal distribution. By taking inverse of this function, we get the inequalities

$$\frac{-p_{si}^c - O_{si} + u_i \bar{\mu}_{si} r_i}{\bar{\sigma}_{si} r_i} \geq u_i F^{-1}((1 - \delta)^{w_{si}}), \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+. \quad (13)$$

Since $F^{-1}((1 - \delta)^{w_{si}})$ is a convex function in w_{si} for $\delta \leq 0.5$ (see [53]), a piecewise linear approximation of (13) can be introduced as follows:

$$\frac{-p_{si}^c - O_{si} + u_i \bar{\mu}_{si} r_i}{\bar{\sigma}_{si} r_i} \geq u_i (\bar{a}_k w_{si} + \bar{b}_k), \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \forall k \in \mathcal{K}, \quad (14)$$

where $\mathcal{K}$ is a set of linear segments, $\bar{a}_k$ and $\bar{b}_k$ are the slope and intercept of each segment, respectively.

Recall that $u_i = (x_i^{s_1} a^{s_1} + x_i^{s_2} a^{s_2} + x_i^{s_3} a^{s_3})$, hence the right hand side of inequalities (14) contains bilinear terms $x_i^{s_1} w_{si}, x_i^{s_2} w_{si}, x_i^{s_3} w_{si}$. We can linearize these terms based on the fact that variables x_i are binary and variables w_{si} are lower and upper bounded due to constraints (11b)–(11c). Let us introduce variables $d_{si}^{s_1}, d_{si}^{s_2}, d_{si}^{s_3}$ such that $d_{si}^{s_1} = x_i^{s_1} w_{si}$, $d_{si}^{s_2} = x_i^{s_2} w_{si}$ and $d_{si}^{s_3} = x_i^{s_3} w_{si}$, respectively. Thus, one can linearize inequalities (14) as follows

$$\begin{aligned} \frac{-p_{si}^c - O_{si} + u_i \bar{\mu}_{si} r_i}{\bar{\sigma}_{si} r_i} &\geq \bar{a}_k (d_{si}^{s_1} a^{s_1} + d_{si}^{s_2} a^{s_2} + d_{si}^{s_3} a^{s_3}) + \bar{b}_k (x_i^{s_1} a^{s_1} + x_i^{s_2} a^{s_2} + x_i^{s_3} a^{s_3}), \\ &\forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \forall k \in \mathcal{K}, \end{aligned} \quad (15a)$$

$$0 \leq d_{si}^{s_1} \leq x_i^{s_1}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15b)$$

$$0 \leq d_{si}^{s_2} \leq x_i^{s_2}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15c)$$

$$0 \leq d_{si}^{s_3} \leq x_i^{s_3}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15d)$$

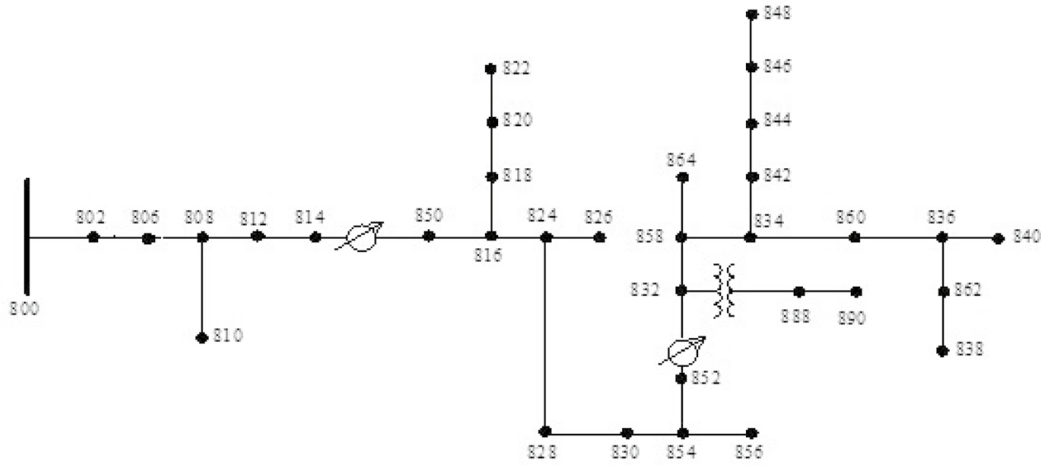


Fig. 4 IEEE 34-BUS test feeder under study

$$w_{si} - (1 - x_i^{s_1}) \leq d_{si}^{s_1} \leq w_{si}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15e)$$

$$w_{si} - (1 - x_i^{s_2}) \leq d_{si}^{s_2} \leq w_{si}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15f)$$

$$w_{si} - (1 - x_i^{s_3}) \leq d_{si}^{s_3} \leq w_{si}, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+, \quad (15g)$$

$$\sum_{i \in \mathcal{N}^+} w_{si} = 1, \quad \forall s \in \mathcal{S}, \quad (15h)$$

$$w_{si} \geq 0, \quad \forall s \in \mathcal{S}, \forall i \in \mathcal{N}^+. \quad (15i)$$

Now, one can replace chance constraints (9) with linearized inequalities (15) in second stage problem (2) to control load shedding. Next, we describe the numerical tests setup and results.

4 Case study

4.1 Numerical experiment setup

In this section, we conduct a computational case study on a radial network in the state of Arizona to show how effective the proposed approach is in controlling load shedding for distributed generation to increase reliability of microgrids. We also compare solution quality and computational time of SAA and the proposed approach.

The IEEE 34-bus test feeder¹ [54] with a nominal voltage of 24.9 kV is employed here for numerical experiments. A one line diagram of this feeder is shown in Fig. 4.

¹ Data available online at <http://ewh.ieee.org/soc/pes/dsacom/testfeeders/index.html>.

Table 1 Per unit line impedances, peak real and reactive load

Line	R_i	X_i	Bus	p_i^c	q_i^c
1	0.0005	0.0005	2	0.11	0.058
2	0.0004	0.0004	4	0.032	0.016
3	0.0065	0.0066	9	0.068	0.034
4	0.0012	0.0013	10	0.27	0.14
5	0.0076	0.0077	11	0.01	0.004
6	0.0060	0.0061	12	0.08	0.04
7	0.0001	0.0001	13	0.008	0.004
8	0.0007	0.0004	14	0.064	0.034
10	0.0059	0.0031	17	0.164	0.086
11	0.0030	0.0022	18	0.056	0.028
12	0.0013	0.0007	19	0.134	0.082
13	0.0002	0.0002	21	0.828	0.64
14	0.0060	0.0044	22	0.09	0.046
15	0.0000	0.0000	23	0.166	0.118
16	0.0017	0.0013	27	0.008	0.004
17	0.0008	0.0006	28	0.03	0.014
18	0.0014	0.0011	29	0.412	0.242
19	0.0003	0.0002	31	0.004	0.002
20	0.0001	0.0001			
21	0.0004	0.0003			
22	0.0011	0.0008			
23	0.0002	0.0001			
24	0.0000	0.0000			
25	0.0108	0.0080			
26	0.0002	0.0001			
27	0.0068	0.0053			
28	0.0014	0.0011			
29	0.0006	0.0004			
30	0.0001	0.0001			
31	0.0007	0.0004			
32	0.0109	0.0408			
33	0.0021	0.0022			

Table 2 Per unit solar PV characteristics

c_i^{setup}	c_i^{area}	c^{LS}	a^{s_1}	a^{s_2}	a^{s_3}	r_i
12,000	12,000	5.0	0.7	1.0	1.3	0.406

For the single-phase network under study, line impedances, peak real and reactive loads are slightly modified and the equivalent per unit (p.u.) values are shown in Table 1. There are two shunt capacitors on nodes 21 and 23 with capacity of 0.6 and 0.9, respectively. We place three auxiliary micro-turbines in node 3, 8, and 16 to

Table 3 Per unit ESS characteristics

c_i^{ESS}	η	IC^{min}	IC^{max}
1000	0.10	0.05	0.16

Table 4 Per unit micro-turbine (MT) characteristics

MT	c_i^{MT}	RU_i	RD_i	P_i^{max}
3	0.9	0.04	0.04	0.22
8	1.0	0.05	0.05	0.23
16	1.1	0.06	0.06	0.24

Table 5 Piecewise approximation coefficients

Segment	$\bar{a}_k$	$\bar{b}_k$
1	− 6.9352	2.9053
2	− 2.0566	2.4478
3	− 0.9492	2.1068
4	− 0.6463	1.9165

help balance the load-generation because of zero solar irradiance during night hours. Characteristics of PV, ESS, and MT units are provided in Tables 2, 3 and 4, respectively. In this study, we use piecewise approximation for right hand side of chance constraints. The coefficients for $\delta = 0.10$ are provided in Table 5.

Finally, we create the seasonal load profiles based on the IEEE RTS-96 system [55], which are shown in Fig. 5 as percentage of the daily peak load (MW). In order to account for the weekday–weekend load difference, we use the maximum load to consider the worse case scenario. All problem instances were implemented in AMPL [56] and solved using CPLEX 12.10 solver on a Windows machine with Intel Core i7 CPU @ 2.60 GHz and 8 GB memory.

4.2 Numerical experiment results

In this case study, we simulate to install up to 5 PV units with different sizes and control real load shedding while dispatching energy for 1 year (i.e., 4 seasons) with $T = 96$ and $T_s = 24$, and extend the operation cost to a horizon of 10 years. For micro-grid design and dispatch problem, we (1) conduct time series analysis based on historical data for the years 2012–2015 from a station near Tucson, Arizona; the data is normalized to $[0,1]$ with $h_i^{max} = 1 \text{ kW/m}^2$, (2) construct the ARIMA models (8) for each season, and (3) employ the combined SAA and linearization technique described in Sect. 3 to solve the two-stage chance constrained problem with $\delta = 0.10$ under 3 different scenario set sizes $M = 100, 200$, and 300. For demonstration purposes, we also solve the two-stage problem with and without chance constraints (CC) using the traditional SAA method under the same problem settings.

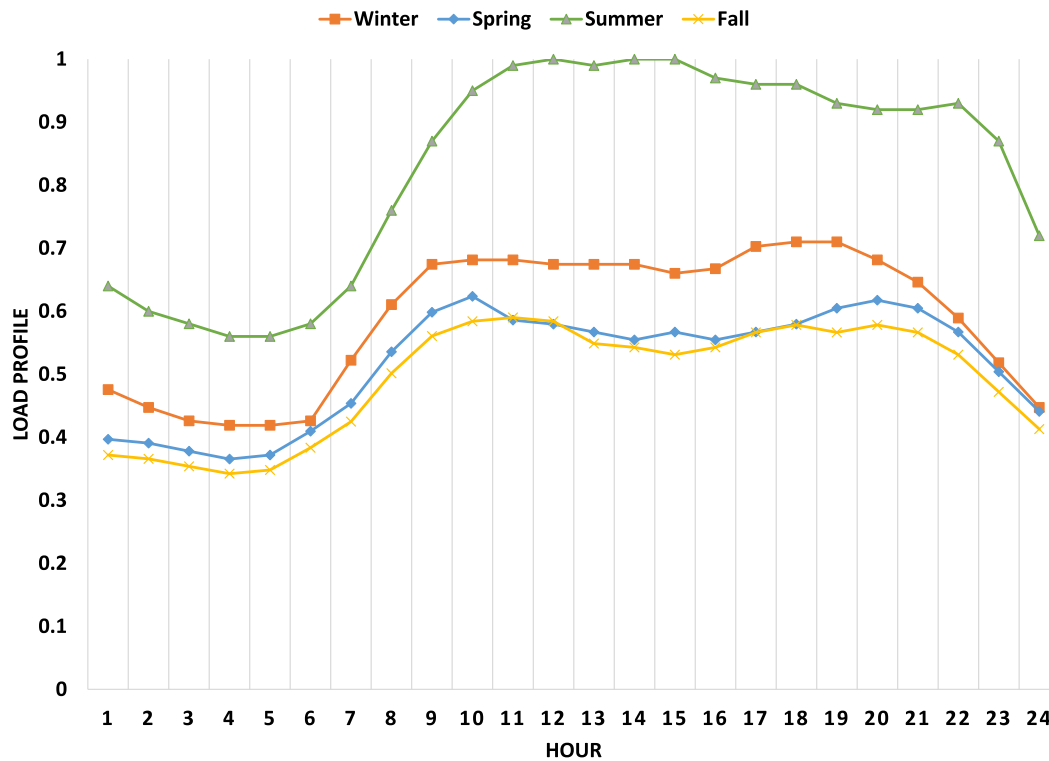


Fig. 5 Seasonal load profiles for the system under study

In this work, we evaluate the effect of chance constraints in controlling real load shedding to increase the reliability of the system under study as well as cost-effectiveness and performance of the two solution approaches. We report results of objective value, its increase, CPU time (with a limit of 6 hours), microgrid design solutions, and breakdown of total cost to capital and dispatch costs for the two-stage problem without chance constraints (model I), with chance constraints and traditional SAA method using “big-M” method shown in (10) as benchmark (model II), and with chance constraints and our proposed combined SAA-linearization approach (model III) in Table 6. Note that in order to capture true load shedding probability one has to run the obtained solutions in Table 6 against a sufficiently large number of scenarios. Hence, Table 7 represents load shedding probabilities $LS1, \dots, LS4$ which are the true load shedding for 4 seasons as well as average total cost under 10,000 Monte Carlo (MC) simulation runs. In this way we evaluate the solution of three models under different scenarios to obtain the true values of the real load shedding percentage.

Based on Table 7, we observe that without chance constraints (model I) load shedding probability is 100% in season 1 (i.e., winter) due to lower solar irradiation and over 29% in season 3 (i.e., summer) due to higher load profile under 300 scenarios. This issue is addressed using chance constraints (models II and III) to control the load shedding within $\delta = 0.10$ threshold with a slight increase in objective function value which is due to installation of more/larger PV and/or ESS units. Note that even though the total cost increase in model III is higher than that in model II (by about 9%), the resulting solution is more robust in terms

Table 6 Objective value, CPU time, and first-stage solutions (location and sizing) of models I, II, and III under in-sample scenarios

Model	Scenarios	Total cost	Capital cost	Dispatch cost	CPU time(s)	Location and sizing (solar)	IC (ESS)	Cost increase
I	100	139,653.7	83,280	56,373.7	771.33	{2(L), 6(L), 26(L)}	{0.16, 0.16, 0.16}	–
	200	138,768.6	83,280	55,488.6	2715.88	{2(L), 4(L), 7(L)}	{0.16, 0.16, 0.16}	–
	300	141,373.1	83,280	58,093.1	6175.44	{2(L), 3(L), 12(L)}	{0.16, 0.16, 0.16}	–
II	100	150,505.8	111,004	39,501.8	8157.4	{2(L), 6(L), 7(L), 26(L)}	{0.16, 0.16, 0.124, 0.16}	7.77%
	200	148,159	110,840	37,319	Time limit	{2(L), 4(L), 7(L), 21(L)}	{0.16, 0.16, 0.12, 0}	6.77%
	300	152,353.2	110,917	41,436.2	Time limit	{2(L), 3(L), 9(L), 12(L)}	{0.16, 0.037, 0.16, 0.16}	7.77%
III	100	164,930.9	134,934	29,996.9	1101.43	{2(L), 6(L), 7(L), 15(M), 26(L)}	{0.16, 0.04, 0.16, 0.16, 0.014}	18.10%
	200	160,472.8	129,142	31,330.8	8005.73	{2(L), 4(L), 7(L), 9(S), 21(L)}	{0.16, 0.16, 0.16, 0.16, 0.10}	15.64%
	300	165,393.8	134,906	30,487.8	8831.37	{2(L), 3(L), 6(L), 9(M), 12(L)}	{0.16, 0.16, 0, 0.16, 0.16}	16.99%

Table 7 Seasonal load shedding percentage (LS1,...,LS4) under 10,000 scenarios from Monte Carlo simulation

Model	Scenario solution	LS1 (%)	LS2 (%)	LS3 (%)	LS4 (%)	Average total cost
I	100	100	0	97.73	0	152,444
	200	99.99	0	7.25	0	140,895
	300	100	0	29.27	0	143,884
II	100	58.76	0	0	0	156,441
	200	11.97 ^a	0 ^a	0 ^a	0 ^a	150,844 ^a
	300	36.59 ^a	0 ^a	0 ^a	0 ^a	155,116 ^a
III	100	0.96	0	0	0	172,186
	200	0.01	0	0	0	164,141
	300	0.13	0	0	0	169,672

^aDenotes incumbent solutions

of load shedding control due to installing more PV units. For example, in an optimal solution of model II under 300 solar irradiance scenarios we have 4 large PV units (with a total size of 5.2) and 4 ESS units (with total size of 0.517) installed on nodes {2, 3, 9, 12}, whereas model III suggests downgrading PV unit in node 9 to a medium size and adding additional large solar unit in node 6 (with a total size of 6.2) and 4 ESS units (with a total size of 0.48) in nodes {2, 3, 9, 12}. Moreover, the load shedding percentage under model III remains below 1%. Such results demonstrate that model III is promoting solar PV installation and significantly reduces load shedding in a cost effective manner. Additionally, model III solves larger instances like 300 scenario one in reasonable time whereas model II fails to find an optimal solution within the rather generous time limit of 6 hours, which further demonstrates the effectiveness of the proposed combined SAA and linearization techniques. Note that load shedding probabilities for model II under 200 and 300 scenario are different because MC simulations are run based on the incumbent solutions obtained within the time limit. It is worth mentioning that optimality gap for all problem instances was set to 0.001 and the reported values are averaged over 5 instances for each scenario set.

In order to show sensitivity of the proposed microgrid design model to the significance level δ , we ran model III under 100 scenarios with different thresholds. The results in Table 8 show that when significance level δ is increased to 0.40 from the original 0.10, the model downgrades a large solar panel at node 7 to a medium one and replaces a medium solar panel at node 15 with a small one at node 18. Hence, the capital cost and the total cost has gone down to 125,528 and 162,854.40, respectively. On the other hand, when we decrease the significance level δ to 0.05 from the original 0.10, the model upgrades a medium solar panel at node 15 to larger size. As a result, the capital cost and the total cost has gone up to 138,507 and 165,662.90, respectively. It is worth mentioning that ESS units with different capacities are installed under such changes of significance level δ as well.

Table 8 Sensitivity of microgrid design model III to the significance level δ

Model	Scenarios	δ	Total cost	Capital cost	Dispatch cost	Location and sizing (solar)	IC (ESS)	Cost increase (%)
III	100	0.40	162,854.4	125,528	37,326.4	{2(L), 6(L), 7(M), 18(S), 26(L)}	{0.16, 0.16, 0.16, 0.15, 0.098}	16.61
	100	0.10	164,930.9	134,934	29,996.9	{2(L), 6(L), 7(L), 15(M), 26(L)}	{0.16, 0.04, 0.16, 0.16, 0.014}	18.10
	100	0.05	165,662.9	138,507	27,155.9	{2(L), 6(L), 7(L), 15(L), 26(L)}	{0.16, 0.16, 0.027, 0, 0.16}	18.62

5 Conclusions and future research directions

In this paper, we proposed a two-stage chance constrained optimization model for solar microgrid design and dispatch. Particularly, we focused on the uncertainty of the solar irradiance and considered time series modeling instead of making assumptions on the distribution of solar irradiance. Additionally, we proposed a combined SAA and linearization technique to solve the problem more efficiently. Compared to the traditional SAA, this method does not require additional binary variables. Based on real-life data, we conducted a case study on a radial network in Arizona. The numerical results indicated that the chance constraints are an effective way to control real power load shedding and the proposed approach outperformed traditional SAA method both in terms of computational time and real load shedding controllability. Moreover, two important future research directions are to explore the possibility of extending the formulation (1) and (2) for economic considerations by buying and selling power to a utility grid at the connection point; and, employ Benders' decomposition algorithm to speed up our solution approach for large-scale problems.

Acknowledgements This work was supported in part by the Bisgrove Scholars program (sponsored by Science Foundation Arizona) Grant BSP 0818-17, AFOSR Grants FA9550-19-1-0161, and DTRA Grant HDTRA1-16-1-0054.

References

1. Yadav, A., Srivastava, L.: Optimal placement of distributed generation: an overview and key issues. In: 2014 International Conference on Power Signals Control and Computations (EPSCICON), pp. 1–6. IEEE (2014)
2. Georgilakis, P.S., Hatziargyriou, N.D.: Optimal distributed generation placement in power distribution networks: models, methods, and future research. *IEEE Trans. Power Syst.* **28**(3), 3420–3428 (2013)
3. Zhaoyu, W., Bokan, C., Jianhui, W., Miroslav, M.B.: Stochastic DG placement for conservation voltage reduction based on multiple replications procedure. *IEEE Trans. Power Deliv.* **30**(3), 1039–1047 (2015)
4. Prakash, P., Khatod, D.K.: Optimal sizing and siting techniques for distributed generation in distribution systems: a review. *Renew. Sustain. Energy Rev.* **57**, 111–130 (2016)
5. Carpentier, J.: Contribution a l'etude du dispatching economique. *Bulletin de la Societe Francaise des Electriciens* **3**(1), 431–447 (1962)
6. Dommel, H.W., Tinney, W.F.: Optimal power flow solutions. *IEEE Trans. Power Appar. Syst.* **87**(10), 1866–1876 (1968)
7. Momoh, J.A., Adapa, R., El-Hawary, M.E.: A review of selected optimal power flow literature to 1993. i. Nonlinear and quadratic programming approaches. *IEEE Trans. Power Syst.* **14**(1), 96–104 (1999)
8. Momoh, J.A., El-Hawary, M.E., Adapa, R.: A review of selected optimal power flow literature to 1993. ii. Newton, linear programming and interior point methods. *IEEE Trans. Power Syst.* **14**(1), 105–111 (1999)
9. Abido, M.A.: Optimal power flow using particle swarm optimization. *Int. J. Electr. Power Energy Syst.* **24**(7), 563–571 (2002)
10. Geidl, M., Andersson, G.: Optimal power flow of multiple energy carriers. *IEEE Trans. Power Syst.* **22**(1), 145–155 (2007)
11. Baran, M.E., Wu, F.F.: Optimal capacitor placement on radial distribution systems. *IEEE Trans. Power Deliv.* **4**(1), 725–734 (1989)

12. Baran, M., Wu, F.F.: Optimal sizing of capacitors placed on a radial distribution system. *IEEE Trans. Power Deliv.* **4**(1), 735–743 (1989)
13. Zhu, Y., Tomsovic, K.: Optimal distribution power flow for systems with distributed energy resources. *Int. J. Electr. Power Energy Syst.* **29**(3), 260–267 (2007)
14. Ochoa, L.F., Keane, A., Harrison, G.P.: Minimizing the reactive support for distributed generation: enhanced passive operation and smart distribution networks. *IEEE Trans. Power Syst.* **26**(4), 2134–2142 (2011)
15. Moradi, M.H., Abedini, M.: A combination of genetic algorithm and particle swarm optimization for optimal dg location and sizing in distribution systems. *Int. J. Electr. Power Energy Syst.* **34**(1), 66–74 (2012)
16. Hung, D.Q., Mithulanathan, N.: Multiple distributed generator placement in primary distribution networks for loss reduction. *IEEE Trans. Industr. Electron.* **60**(4), 1700–1708 (2013)
17. Dall'Anese, E., Zhu, H., Giannakis, G.B.: Distributed optimal power flow for smart microgrids. *IEEE Trans. Smart Grid* **4**(3), 1464–1475 (2013)
18. Levron, Y., Guerrero, J.M., Beck, Y.: Optimal power flow in microgrids with energy storage. *IEEE Trans. Power Syst.* **28**(3), 3226–3234 (2013)
19. Zhang, B., Lam, A.Y.S., Domínguez-García, A.D., Tse, D.: An optimal and distributed method for voltage regulation in power distribution systems. *IEEE Trans. Power Syst.* **30**(4), 1714–1726 (2015)
20. Atwa, Y.M., El-Saadany, E.F., Salama, M.M.A., Seethapathy, R.: Optimal renewable resources mix for distribution system energy loss minimization. *IEEE Trans. Power Syst.* **25**(1), 360–370 (2010)
21. Zhaoyu, W., Bokan, C., Jianhui, W., Jinho, K., Miroslav, M.B.: Robust optimization based optimal DG placement in microgrids. *IEEE Trans. Smart Grid* **5**(5), 2173–2182 (2014)
22. Bazrafshan, M., Gatsis, N.: Risk-averse placement and sizing of photovoltaic inverters in radial distribution networks. In: 2015 49th Asilomar Conference on Signals, Systems and Computers, pp. 885–889. IEEE (2015)
23. Sun, B., Krokhmal, P., Chen, Y.: Risk-averse capacity planning for renewable energy production. *Energy Syst.* **9**, 1–34 (2018)
24. Charnes, A., Cooper, W.W.: Chance-constrained programming. *Manag. Sci.* **6**(1), 73–79 (1959)
25. Li, P., Arellano-Garcia, H., Wozny, G.: Chance constrained programming approach to process optimization under uncertainty. *Comput. Chem. Eng.* **32**(1), 25–45 (2008)
26. Ozturk, U.A., Mazumdar, M., Norman, B.A.: A solution to the stochastic unit commitment problem using chance constrained programming. *IEEE Trans. Power Syst.* **19**(3), 1589–1598 (2004)
27. Liu, Z., Wen, F., Ledwich, G.: Optimal siting and sizing of distributed generators in distribution systems considering uncertainties. *IEEE Trans. Power Deliv.* **26**(4), 2541–2551 (2011)
28. Zhang, H., Li, P.: Chance constrained programming for optimal power flow under uncertainty. *IEEE Trans. Power Syst.* **26**(4), 2417–2424 (2011)
29. Cao, Y., Tan, Y., Li, C., Rehtanz, C.: Chance-constrained optimization-based unbalanced optimal power flow for radial distribution networks. *IEEE Trans. Power Deliv.* **28**(3), 1855–1864 (2013)
30. Jin, T., Tian, Y., Zhang, C.W., Coit, D.W.: Multicriteria planning for distributed wind generation under strategic maintenance. *IEEE Trans. Power Deliv.* **28**(1), 357–367 (2013)
31. Hejazi, H., Mohsenian-Rad, H.: Energy storage planning in active distribution grids: a chance-constrained optimization with non-parametric probability functions. *IEEE Trans. Smart Grid* **9**, 1972–1985 (2016)
32. Turitsyn, K., Šulc, P., Backhaus, M.C.: Distributed control of reactive power flow in a radial distribution circuit with high photovoltaic penetration. In: Power and Energy Society General Meeting, 2010 IEEE, pp. 1–6. IEEE (2010)
33. Baran, M.E., Wu, F.F.: Network reconfiguration in distribution systems for loss reduction and load balancing. *IEEE Trans. Power Deliv.* **4**(2), 1401–1407 (1989)
34. Yeh, H.-G., Gayme, D.F., Low, S.H.: Adaptive var control for distribution circuits with photovoltaic generators. *IEEE Trans. Power Syst.* **27**(3), 1656–1663 (2012)
35. Wang, Z., Chen, B., Wang, J., et al.: Decentralized energy management system for networked microgrids in grid-connected and islanded modes. *IEEE Trans. Smart Grid* **7**(2), 1097–1105 (2016)
36. Koutroulis, E., Kolokotsa, D., Potirakis, A., Kalaitzakis, K.: Methodology for optimal sizing of stand-alone photovoltaic/wind-generator systems using genetic algorithms. *Sol. Energy* **80**(9), 1072–1088 (2006)
37. Boyson, W.E., Galbraith, G.M., King, D.L., Gonzalez, S.: Performance model for grid-connected photovoltaic inverters. Technical report, Sandia National Laboratories (2007)
38. Dall'Anese, E., Giannakis, G.B.: Optimal distributed generation placement in distribution systems via semidefinite relaxation. In: Asilomar Conference on Signals, Systems and Computers, pp. 369–373. IEEE (2013)

39. Niknam, T., Zare, M., Aghaei, J.: Scenario-based multiobjective volt/var control in distribution networks including renewable energy sources. *IEEE Trans. Power Deliv.* **27**(4), 2004–2019 (2012)
40. Karaki, S.H., Chedid, R.B., Ramadan, R.: Probabilistic performance assessment of autonomous solar-wind energy conversion systems. *IEEE Trans. Energy Convers.* **14**(3), 766–772 (1999)
41. Zhaoxia, S., Weiwei, L., Jinfeng, Z., Qiangmin, L., Tianci, L.: A planning method for siting and sizing of distributed generation based on chance-constrained programming. In: 2015 5th International Conference on Electric Utility Deregulation and Restructuring and Power Technologies (DRPT), pp. 527–531. IEEE (2015)
42. Mora-Lopez, L.L., Sidrach-de-Cardona, M.: Multiplicative ARMA models to generate hourly series of global irradiation. *Solar Energy* **63**(5), 283–291 (1998)
43. Boland, J.: Time series modelling of solar radiation. In: Badescu, V. (ed.) *Modeling Solar Radiation at the Earth's Surface*, pp. 283–312. Springer, Berlin (2008)
44. Reikard, G.: Predicting solar radiation at high resolutions: a comparison of time series forecasts. *Sol. Energy* **83**(3), 342–349 (2009)
45. Box, G.E.P., Jenkins, G.M., Reinsel, C., Ljung, G.M.: *Time Series Analysis: Forecasting and Control*. Wiley, Hoboken (2015)
46. National renewable energy laboratory solar prospector (2012). Available online: <http://maps.nrel.gov/prospector>. Accessed 25 Feb 2017
47. Team, R Core.: *R: A language and environment for statistical computing*. Vienna, Austria: R foundation for statistical computing; 2014 (2014)
48. Adel Mellit and Alessandro Massi Pavan: A 24-h forecast of solar irradiance using artificial neural network: application for performance prediction of a grid-connected pv plant at trieste, italy. *Sol. Energy* **84**(5), 807–821 (2010)
49. Ochoa, L.F., Dent, C.J., Harrison, G.P.: Distribution network capacity assessment: variable DG and active networks. *IEEE Trans. Power Syst.* **25**(1), 87–95 (2010)
50. Ochoa, L.F., Harrison, G.P.: Minimizing energy losses: optimal accommodation and smart operation of renewable distributed generation. *IEEE Trans. Power Syst.* **26**(1), 198–205 (2011)
51. Pagnoncelli, B.K., Ahmed, S., Shapiro, A.: Sample average approximation method for chance constrained programming: theory and applications. *J. Optim. Theory Appl.* **142**(2), 399–416 (2009)
52. Wang, Q., Guan, Y., Wang, J.: A chance-constrained two-stage stochastic program for unit commitment with uncertain wind power output. *IEEE Trans. Power Syst.* **27**(1), 206–215 (2012)
53. Cheng, J., Lissner, A.: A second-order cone programming approach for linear programs with joint probabilistic constraints. *Oper. Res. Lett.* **40**(5), 325–328 (2012)
54. Kersting, W.H.: Radial distribution test feeders. *IEEE Trans. Power Syst.* **6**(3), 975–985 (1991)
55. Pinheiro, J.M.S., Dornellas, C.R.R., Th Schilling, M., Melo, A.C.G., Mello, J.C.O.: Probing the new IEEE reliability test system (rts-96): HI-II assessment. *IEEE Trans. Power Syst.* **13**(1), 171–176 (1998)
56. Fourer, R., Gay, D.M., Kernighan, B.W.: *AMPL: A Mathematical Programming Language*, p. 07974. AT&T Bell Laboratories, Murray Hill (1987)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Hossein Dashti¹ · Jianqiang Cheng² · Pavlo Krokhmal²

Hossein Dashti
hossein.dashti@aa.com

Jianqiang Cheng
jqcheng@arizona.edu

¹ Operations Research and Advanced Analytics Group, American Airlines Inc., Fort Worth, TX, USA

² Department of Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, USA

RESEARCH ARTICLE

WILEY

On maximum ratio clique relaxations

Yehor Blokhin¹ | Sergiy Butenko¹  | Petar Momcilovic¹  | Oleg A. Prokopyev² ¹Department of Industrial and Systems Engineering, Texas A&M University, College Station, Texas, USA²Department of Industrial Engineering, University of Pittsburgh, Pittsburgh, Pennsylvania, USA**Correspondence**Oleg A. Prokopyev, Department of Industrial Engineering, University of Pittsburgh, 1031 Benedum Hall, Pittsburgh, PA 15261, USA.
Email: droleg@pitt.edu**Funding information**

Air Force Office of Scientific Research, Grant/Award Number: FA9550-19-1-0161; National Science Foundation, Grant/Award Number: DMS 1818700; CRDF Global, Grant/Award Number: G-202102-67302.

Abstract

This article introduces and studies two clique relaxation models with fractional objectives: the maximum ratio s -plex problem and the maximum ratio s -defective clique problem. The decision version of each problem is shown to be strongly NP -complete; we also discuss some related computational complexity issues. The base optimization models are single-ratio fractional 0-1 problems. We describe two general types of solution methods: the first one is based on mixed-integer linear programs (MILPs) obtained by linearizing the original nonlinear 0-1 models; the other one exploits parametric methods (namely, a binary search and Newton's method) that solve MILPs in an iterative manner. For the proposed MILPs, we derive valid inequalities that are shown to substantially improve the performance of an off-the-shelf MILP solver. Finally, the considered solution approaches are compared via extensive numerical experiments using a set of artificially generated and real-life network instances.

KEYWORDS

clique relaxation, fractional 0-1 programming, linearization, mixed integer programming, network, parametric methods

1 | INTRODUCTION

Network analysis deals with network representations of data describing various complex systems, aiming to characterize a system's behavior by exploring the structural properties of the corresponding network model [27]. Naturally, large tightly knit groups of nodes, referred to as clusters, tend to play an important role in this methodology, making cluster detection a central task in network analysis. One of the classical examples of this approach is found in social network analysis, studying interactions between individuals, organizations, and other entities. Clusters, or cohesive subgroups, in social networks represent groups of entities with common features, which can be thought of as fundamental building blocks of the studied system. It is in this context that the concept of a *clique* was first introduced in the literature by Luce and Perry in 1949 [23].

Given a simple, undirected graph, a clique is a subset of vertices such that any two vertices from the subset are adjacent to each other. It should be noted that the original definition of a clique required *maximality by inclusion*, due to its application-specific considerations. In modern graph-theoretic terms, the structure originally defined in [23] is referred to as a *maximal clique*, which is a clique that is not a subset of a larger clique in the graph. Intuitively, the larger is a clique, the more important role it is expected to play in overall network's structure. This consideration motivates the classical *maximum clique problem*, which is to find a clique of the largest cardinality in the graph; see, for example, [6, 29]. More generally, given non-negative rational weights associated with the graph's vertices, the *maximum weight clique problem* is to find a clique that maximizes the total sum of weights of its vertices.

The maximum (weight) clique problem is among the most popular and well-studied problems in discrete optimization, which arises in a wide variety of applications; see, for example, surveys in [9, 46]. Nevertheless, in some of the applied contexts the following two features of the maximum weight clique model are often found inadequate:

- (i) The structural definition of a clique, requiring every possible edge in the subset to be present, is too restrictive for modeling many real-life clusters.
- (ii) The sum-of-weights objective is appropriate for maximizing the sum of “benefits” of the vertices. However, it is not suitable for detecting a cluster maximizing some *efficiency* measure (e.g., a “benefit-to-cost” ratio), which is a common performance metric in many real-life application settings.

Each of these two drawbacks is addressed separately in the previous work related to cliques. On the one hand, the overly restrictive nature of a clique led researchers to define numerous *clique relaxation* structures, relaxing one or more *elementary clique-defining properties*; see a detailed discussion on this issue in [31]. On the other hand, the limitations of the maximum-sum objective were remedied by the introduction of the *maximum ratio clique problem* [37]. However, to the best of our knowledge, no previous research dealt with maximum ratio clique relaxation models. This article starts filling this gap by introducing a general maximum ratio clique relaxation model and studying its two specific cases, the *maximum ratio s-plex problem* and the *maximum ratio s-defective clique problem*. Below in this section, we motivate this particular choice of clique relaxations, formally define the optimization problems of interest, and then position the contributions of this article among the existing work.

As mentioned above, clique relaxations were introduced to overcome the practical limitations of the utterly rigid clique requirements. This motivated many alternative definitions of cohesive subgroups, primarily in the field of social network analysis, focusing on various structural properties expected of a cohesive subgroup. The specific structural properties enforced, in a relaxed form, in these definitions include *familiarity* (each vertex has many neighbors and only a few non-neighbors in the group), *reachability* (a low diameter, facilitating fast communication between the group members) and *robustness* (high connectivity, making it difficult to destroy the group by removing members). The models formalizing these properties in graph-theoretic terms include degree-based relaxations called *s-plex* [36] and *k-core* [35], edge density-based models known as *s-defective clique* [49] and γ -*quasi-clique* [1], and the distance-based clique relaxations called *s-clique* [22] and *s-club* [3, 26].

In [31], these and other clique relaxation concepts are examined in a methodical fashion, and a systematic framework for defining clique relaxation models is proposed. According to this framework, all of the considered clique relaxations can be defined by applying some simple rules to the so-called *elementary clique-defining properties*, providing equivalent characterizations of a clique in terms of various graph-theoretic concepts (distance, diameter, degree, edge density, etc.). It is also observed that the way a clique relaxation is described based on an elementary clique-defining property may prescribe some important structural properties it is guaranteed to possess. One such property, *heredity on induced subgraphs*, has significant implications on several important aspects concerning the problem of finding a maximum cardinality subset of vertices satisfying the given clique relaxation definition, including the existence of a meaningful complementary structure, the computational complexity of the problem, as well as our ability to extend the standard algorithms for the maximum clique problem to the problem of interest [42].

Formally, a graph property Π is called *hereditary on induced subgraphs*, if for any graph G with property Π deleting any subset of vertices does not produce a graph violating Π . In addition, property Π is said to be *nontrivial* if a single-vertex graph satisfies Π , but not every graph satisfies Π . Furthermore, a property is *interesting* if there are arbitrarily large graphs satisfying Π . If Π is nontrivial, interesting, and hereditary on induced subgraphs, then the *maximum Π -set problem*, which is to find a maximum-cardinality subset of vertices inducing a subgraph satisfying Π , is known to be *NP-hard* [48]. At the same time, the presence of the heredity property invites adaptation of standard combinatorial branch-and-bound algorithms successfully used for the maximum clique problem [42].

Two of the most popular hereditary clique relaxations, *s-plex* and *s-defective clique*, are in the focus of this study. They are formally defined next. We consider a simple, undirected graph $G = (V, E)$ with the set of vertices $V = \{1, \dots, n\}$ and the set of edges $E \subseteq \binom{V}{2}$. We use $\{i, j\}$ to denote an edge in E between vertices i and j , where $i, j \in V$. For a subset of vertices $C \subseteq V$, the subgraph induced by C in G is denoted by $G[C]$, that is, $G[C] := (C, E \cap \{\{i, j\} \in E \mid i, j \in C\})$.

Definition 1. A subset of vertices $C \subseteq V$ is called

- an *s-plex* if every vertex in C is adjacent to at least $|C| - s$ vertices in $G[C]$;
- an *s-defective clique* if $G[C]$ has at least $\binom{|C|}{2} - s$ edges.

The corresponding *maximum s-plex* and the *maximum s-defective clique problems* are known to be *NP-hard* and studied in [4, 42].

Following the definition of the maximum ratio clique problem (MRCP) presented in [37], we introduce the general maximum ratio Π -set problem. Recall that $C \subseteq V$ is a Π -set if $G[C]$ satisfies property Π . A Π -set is maximal if it is not a subset of a larger Π -set. Then:

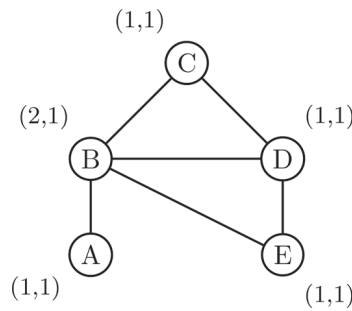


FIGURE 1 An illustrative example for Definitions 1 and 2. The weights (a_i, b_i) , $i \in V$, are depicted for each vertex $i \in \{A, B, C, D, E\}$. On the one hand, observe that $\{B, C, D\}$ and $\{B, D, E\}$ form the cliques of maximum cardinality. On the other hand, $\{A, B\}$ is the most “efficient” maximal clique, that is, an optimal solution of the MRCP problem. Similarly, $\{B, C, D, E\}$ is a 2-plex (1-defective clique) of maximum cardinality, while $\{A, B, D\}$ is an optimal solution of the MRsPP with $s = 2$ (the MRsDCP with $s = 1$)

Definition 2. Given a simple, undirected graph $G = (V, E)$, where each vertex $i \in V$ is assigned two non-negative rational weights, a_i and b_i , the maximum ratio Π -set problem is to find a maximal Π -set C in G that maximizes the quantity

$$f(C) = \frac{\sum_{i \in C} a_i}{\sum_{i \in C} b_i}. \quad (1)$$

If $b_i = 1$ for all $i \in V$, then the problem is referred to as the maximum average weight Π -set problem.

Observe that this definition allows a possibility of a zero denominator, in which case the objective is $+\infty$ (the $0/0$ case can be avoided by eliminating all vertices i such that $a_i = b_i = 0$). Furthermore, we note that following the study in [37] we require maximality by inclusion in Definition 2. In particular, it allows us to avoid trivial solutions that arise from having the fractional objective; also, recall our earlier discussion on this issue in the context of [23].

In this article, we focus on two special cases of the maximum ratio Π -set problem:

- the *maximum ratio s -plex problem (MRsPP)*, where a Π -set is an s -plex for some positive integer constant s ; and
- the *maximum ratio s -defective clique problem (MRsDCP)*, where a Π -set is an s -defective clique for some non-negative integer constant s .

Similarly, we define the *maximum average weight s -plex problem (MAsPP)* and *maximum average weight s -defective clique problem (MAsDCP)*, whenever $b_i = 1$ for all $i \in V$. Note that MRCP from [37] is a special case of the our problems as it corresponds to MRsPP with $s = 1$ or MRsDCP with $s = 0$. We illustrate Definitions 1 and 2 in Figure 1.

Generally speaking, the problems studied in this article belong to a broad class of combinatorial optimization problems with fractional objective functions. We refer the reader to recent surveys on fractional optimization [8, 40] and the references therein.

Several approaches have been considered for the MRCP problem in the literature. In particular, algorithms based on mixed-integer linear programming (MILP), a binary search, and Newton’s method are discussed in [37]. Also, an approach based on DC programming (where the term “DC” is used to denote “difference of convex functions”) is proposed in [25], and a variable neighborhood search (VNS) heuristic is developed in [17].

In this work, we extend the approaches proposed for the MRCP in [37] to the MRsPP and MRsDCP problems. Specifically:

- We study the theoretical computational complexity of the maximum ratio versions for several popular clique relaxation models and other structures; see Section 2. We establish a general *NP*-hardness result and then explore several other related computational complexity issues including approximability and uniqueness.
- We then study two general classes of solution methods for the considered class of fractional combinatorial optimization problems. The first class is based on mixed-integer programming (MIP) techniques that exploit linearization ideas to provide MILP formulations solvable by an off-the-shelf solver; see Section 3. We also derive several types of valid inequalities that can be used to strengthen our MILPs and improve the solver’s performance. Specialized linearization approaches for the average weight case of our problems are also considered.
- The other class of solution methods is based on the classical parametric solution scheme from fractional optimization; see Section 4. The key idea is to find a root of a particular parameterized function in iterative manner using either the binary search or the Newton’s method. In each iteration, these methods solve an MILP model using an off-the-shelf solver.
- Finally, we compare the performance of the considered solution methods using randomly generated and real-life networks instances, including the stock market and wind energy graphs, as well as instances from the 2nd and 10th DIMACS Implementation Challenges; see Section 5. Some interesting qualitative observations are also provided.

2 | COMPUTATIONAL COMPLEXITY

Given a simple undirected graph $G = (V, E)$, where each vertex $i \in V$ is assigned two non-negative rational weights, a_i and b_i , and a non-negative rational number c , the *decision version* of the maximum ratio Π -set problem is to determine if there exists a maximal Π -set C in G for which the quantity $f(C) := \frac{\sum_{i \in C} a_i}{\sum_{i \in C} b_i} \geq c$.

First, we establish a general complexity result for the maximum ratio Π -set problem, applicable to a wide variety of graph properties in general and clique relaxations in particular. We refer the reader to [16] for a classical reference on the theoretical computational complexity.

Proposition 1. *Consider a graph property Π such that the following two requirements hold:*

- (R1) *the maximum Π -set problem is NP-hard and*
- (R2) *exactly one of the following two properties is satisfied:*
 - (a) *If C is a Π -set and $v \in V \setminus C$ is adjacent to all vertices in C , then $C \cup \{v\}$ is a Π -set.*
 - (b) *If C is a Π -set and $v \in V \setminus C$ is adjacent to no vertex in C then $C \cup \{v\}$ is a Π -set.*

Then the maximum ratio Π -set problem and the maximum average weight Π -set problem are strongly NP-hard. Moreover, if checking Π -set maximality by inclusion can be done in polynomial time then the decision versions of these problems are strongly NP-complete.

Proof. Given an instance $G = (V, E)$ of the maximum Π -set problem, we construct an instance of the maximum ratio Π -set problem as follows. We introduce a new vertex v and build a graph $G' = (V', E')$, where $V' = V \cup \{v\}$ and

$$E' = \begin{cases} E \cup \{\{u, v\} \mid \forall u \in V\}, & \text{if property (a) is satisfied,} \\ E, & \text{if property (b) is satisfied.} \end{cases}$$

Furthermore, we assign the vertex weights a_i and b_i for each $i \in V'$ as follows: $a_i = 1$ for all $i \in V$, $a_v = 0$ and $b_i = 1$ for all $i \in V'$. Since a Π -set is required to be maximal for feasibility, the vertex v belongs to every feasible solution of the maximum ratio Π -set problem.

Assume there exist two distinct maximal Π -sets in G , given by C_1 and C_2 . (Note that if there exists only one maximal Π -set in G , then the proof is trivial; recall Definition 2.) By our construction $C'_1 := C_1 \cup \{v\}$ and $C'_2 := C_2 \cup \{v\}$ form two distinct maximal Π -sets in G' . Hence, for the maximum ratio Π -set problem in G' we have:

$$f(C'_1) - f(C'_2) = \frac{|C_1|}{|C_1| + 1} - \frac{|C_2|}{|C_2| + 1} = \frac{|C_1| - |C_2|}{(|C_1| + 1)(|C_2| + 1)},$$

which implies that $|C_1| > |C_2|$ if and only if $f(C'_1) - f(C'_2) > 0$. Therefore, finding an optimal solution to the maximum ratio Π -set problem in G' is equivalent to finding a maximum Π -set in G . Also, recall that the vertex weights (i.e., the values of a_i and b_i) in the constructed instance belong to the set $\{0, 1\}$. In view of these observations, we conclude that the maximum ratio Π -set problem is strongly NP-hard. Finally, our construction (namely, the fact that $b_i = 1$ for all $i \in V'$) implies that the maximum average weight Π -set problem is also strongly NP-hard. ■

Proposition 1 is as a generalization of the corresponding result in [37], which covers only the MRCP. The reduction used in the proof above exploits the construction idea proposed in [37], with some appropriate modifications to capture (R1) and (R2).

It is important to point out that most clique relaxations satisfy property (a) in Proposition 1, while complementary structures, generalizing independent set, tend to satisfy property (b). In particular, Proposition 1 yields strong NP-hardness of the maximum ratio s -plex, s -defective clique, s -club, and γ -quasi-clique problems, among many others. Furthermore, for the decision versions of the MRsPP and MRsDCP as well as their average weight versions MAsPP and MAsDCP, respectively, the strong NP-completeness result clearly holds.

Corollary 1. *The decision versions of the MRsPP, MRsDP, MAsPP, and MAsDP problems are strongly NP-complete.*

Note that Proposition 1 does not imply strong NP-completeness of the decision versions of the maximum ratio Π -set problems for clique relaxations that do not possess the heredity in the induced subgraphs, such as s -club and γ -quasi-clique; recall our discussion on the hereditary property in Section 1. In fact, checking maximality by inclusion is known to be NP-hard for s -club; see [24].

A graph for which every maximal independent set is a maximum independent set is called *well-covered*. It is known that recognizing whether a graph is not well-covered is NP-complete [34]. This result is exploited in [37] to show that verifying

whether there exist two maximal cliques, C_1 and C_2 , such that their corresponding weighted ratios are different, that is, $|f(C_1) - f(C_2)| > 0$, is also a difficult problem.

Somewhat related results are also known for other classes of fractional combinatorial optimization problems. In particular, for *unconstrained* fractional 0-1 problems the study in [10] shows that recognizing whether the globally optimal solution is unique, is a Δ_2P -complete problem. Furthermore, it is known that an unconstrained fractional 0-1 problem remains *NP*-hard even if it is known that the globally optimal solution is unique [32]. (We note that for the single-ratio version of the problem, the denominator should be allowed to take both positive and negative values; otherwise, the problem is easy.) Next, we establish a similar in spirit result for our class of problems.

Proposition 2. *Consider a graph property Π such that (R1) and (R2) hold. The maximum ratio Π -set and the maximum average weight Π -set problems remain *NP*-hard even if it is known that the globally optimal solution is unique.*

Proof. We use the same construction as in the proof of Proposition 1 with slightly different vertex weights with an additional construction idea from [32]. Specifically, we assign the vertex weights a_i and b_i for each $i \in V'$ as follows: $a_i = M + 2^i$ for all $i \in V$ and $a_v = 0$, and $b_i = 1$ for all $i \in V'$, where $n = |V|$ and $M = (n + 1)(2^n - 1) + 1$. Note that we assume that vertices in V are indexed from 1 to n , that is, $V = \{1, \dots, n\}$, and $V' = V \cup \{v\}$.

As in the proof of Proposition 1, we consider two distinct maximal Π -sets in G , given by C_1 and C_2 . Also, recall that a Π -set is required to be maximal for feasibility; hence, the vertex v belongs to every feasible solution of the maximum ratio Π -set problem in G' . Then $C'_1 := C_1 \cup \{v\}$ and $C'_2 := C_2 \cup \{v\}$ form two distinct maximal Π -sets in G' and we have:

$$\begin{aligned} f(C'_1) - f(C'_2) &= \frac{M|C_1| + \sum_{i \in C_1} 2^i}{|C_1| + 1} - \frac{M|C_2| + \sum_{i \in C_2} 2^i}{|C_2| + 1}, \\ &= \frac{M(|C_1| - |C_2|) + \left(\sum_{i \in C_1} 2^i(|C_2| + 1) - \sum_{i \in C_2} 2^i(|C_1| + 1) \right)}{(|C_1| + 1)(|C_2| + 1)}. \end{aligned} \quad (2)$$

Next, we observe that

$$\left| \sum_{i \in C_1} 2^i(|C_2| + 1) - \sum_{i \in C_2} 2^i(|C_1| + 1) \right| \leq (n + 1)(2^n - 1),$$

which implies that if $|C_1| > |C_2|$ then $f(C'_1) - f(C'_2) > 0$ given our setting for the value of M . Therefore, a globally optimal solution for the maximum ratio Π -set problem in G' corresponds to a maximum Π -set in G .

Finally, we observe that if C_1 and C_2 are two distinct maximal Π -sets in G of the same size (i.e., $C_1 \neq C_2$ and $|C_1| = |C_2|$), then $f(C'_1) - f(C'_2) \neq 0$; see the numerator in (2). This observation implies that the maximum ratio Π -set problem in G' has a unique globally optimal solution. Hence, the proposed reduction proves the statement of the proposition. ■

Corollary 2. *The MRsPP, MRsDP, MAsPP, and MAsDP problems remain *NP*-hard even if it is known that the corresponding globally optimal solutions are unique.*

Next, we show that if it is difficult to find an approximate solution for the maximum Π -set problem, then the corresponding maximum ratio problem is also difficult to approximate.

Proposition 3. *Consider a graph property Π such that (R1) and (R2) hold. If the maximum Π -set problem is *NP*-hard to approximate within some factor ρ , then the maximum ratio Π -set problem is *NP*-hard to approximate within the same factor.*

Proof. We use the same construction as in the proof of Proposition 1 with slightly different vertex weights. Specifically, we assign the vertex weights a_i and b_i for each $i \in V'$ as follows: $a_i = 1$ for all $i \in V$ and $a_v = 0$, while $b_i = 0$ for all $i \in V$ and $b_v = 1$.

Then we consider any maximal Π -sets in G , given by C . By our construction $C' := C \cup \{v\}$ forms a maximal Π -set in G' . Hence, for the maximum ratio Π -set problem in G' we have:

$$f(C') = \frac{|C| + 0}{0 + 1} = |C|, \quad (3)$$

which implies that finding either an exact solution or an approximate one for the maximum ratio Π -set problem is at least as difficult as solving the corresponding maximum Π -set problem. ■

It is known that approximating maximum clique within a factor better than $O(|V|^{1-\epsilon})$ is *NP*-hard for any $\epsilon > 0$; see [50]. Therefore:

Corollary 3. *Finding an approximate solution for the MRCP problem within a factor better than $O(|V|^{1-\epsilon})$ is *NP*-hard for any $\epsilon > 0$.*

Finally, we note that our reduction in Proposition 3 (in contrast to our reductions in the proofs of Propositions 1 and 2) does not apply to the maximum average weight Π -set problem. Hence, approximability of this version of the problem remains an open research question, which provides an interesting avenue for future research.

3 | MIP MODELS AND ENHANCEMENTS

In Section 3.1, we formulate our base single-ratio fractional 0-1 model. In Section 3.2, we show how to linearize this nonlinear 0-1 model to obtain MILPs solvable by an off-the-shelf solver. In Section 3.3, we derive valid inequalities that strengthen our MILPs and improve the solver's performance; some of these inequalities are based on simple upper bounds on the sizes of the considered clique relaxations. Finally, in Section 3.4, we consider additional specialized versions of our MILPs for the average weight cases of our problems.

3.1 | Base fractional 0-1 model

A typical integer programming formulation for a maximum Π -set problem involves binary decision variables $x \in \{0, 1\}^{|V|}$ such that $x_i = 1$ if and only if vertex i is included in the Π -set solution. For $x \in \mathbb{R}^{|V|}$, let $S(x) := \{i \in V : x_i > 0\}$ denote the support of x . That is, given a binary x , we refer to $S(x)$ as the set of vertices corresponding to x or defined by x . Assuming that X_Π is the set of all feasible solutions (i.e., $x \in X_\Pi$ if and only if $S(x)$ is a Π -set), the maximum Π -set problem can be formulated as the problem of maximizing the sum of entries of x over all $x \in X_\Pi$.

To obtain a formulation of the maximum ratio Π -set problem, we need to add *maximality by inclusion constraints* to the description of X_Π in order to obtain a description of the feasible set $\bar{X}_\Pi$, such that $x \in \bar{X}_\Pi$ if and only if $S(x)$ is a maximal Π -set. Then the maximum ratio Π -set problem can be formulated as follows:

$$\begin{aligned} \max \quad & \frac{\sum_{i \in V} a_i x_i}{\sum_{i \in V} b_i x_i}, \\ \text{s.t.} \quad & x \in \bar{X}_\Pi, \end{aligned} \quad (4)$$

where set $\bar{X}_\Pi$ can be described by linear constraints and binary variables; this description may also involve some auxiliary variables in addition to x . Then the resulting nonlinear 0-1 model can be linearized to obtain MILP formulations. We describe such MILPs next.

3.2 | MILPs for the general case

Let $A = [a_{ij}]_{i,j \in V}$ be the adjacency matrix of G , where $a_{ij} = 1$ for all $\{i, j\} \in E$; all other entries of A are zero including $a_{ii} = 0$ for all $i \in V$. Let $\bar{d}_i = |V \setminus (\mathcal{N}(i) \cup \{i\})| = |V \setminus \mathcal{N}(i)| - 1$ be the degree of vertex i in the complement graph $\bar{G} = (V, \bar{E})$, where $\mathcal{N}(i)$ denotes the neighborhood of i in G , that is, $\mathcal{N}(i) := \{j \in V : \{i, j\} \in E\}$, and $\bar{E} = \{\{i, j\} : \{i, j\} \notin E\}$.

Integer programming formulations for the maximum s -plex and the maximum s -defective clique problems are described in [4, 39], respectively. To derive our formulations, we first need to consider fractional objective functions and then introduce additional linear constraints to enforce the maximality-by-inclusion requirement; recall Definition 2. Note that another class of fractional problems on graphs is also considered in [44], where similar in spirit linearized 0-1 models are described; however, the study in [44] does not enforce the maximality-by-inclusion requirement.

Specifically, MRsPP can be formulated as the following fractional 0-1 problem:

$$f_p^*(s) = \max \frac{\sum_{i \in V} a_i x_i}{\sum_{i \in V} b_i x_i}, \quad (5a)$$

$$\text{s.t.} \quad \sum_{j \in V} (1 - a_{ij}) x_j + (\bar{d}_i - s) x_i \leq \bar{d}_i, \quad \forall i \in V, \quad (5b)$$

$$\sum_{j \in V} (1 - a_{ij}) x_j + (s - 1) x_i \geq s, \quad \forall i \in V, \quad (5c)$$

$$x_i \in \{0, 1\}, \quad \forall i \in V, \quad (5d)$$

where variable x_i , $i \in V$, indicates whether vertex i is included into the solution s -plex. Note that constraints (5b) are the s -plex-defining constraints; see [4]. That is, $S(x)$ is an s -plex for a binary x if and only if x satisfies (5b). In addition, constraints (5c) ensure that the s -plex induced by $S(x)$ is maximal. Indeed, if $x_i = 1$ then the constraint trivially holds. If $x_i = 0$, then the constraint requires that vertex i has at least s non-neighbors in $S(x)$, implying that the s -plex cannot be expanded by including vertex i .

Similarly, MRsDCP can be formulated as the following fractional 0-1 problem:

$$f_d^*(s) = \max \frac{\sum_{i \in V} a_i x_i}{\sum_{i \in V} b_i x_i}, \quad (6a)$$

$$\text{s.t. } \sum_{\{i,j\} \in \bar{E}} y_{ij} \leq s, \quad (6b)$$

$$x_i + x_j - y_{ij} \leq 1, \quad \forall \{i,j\} \in \bar{E}, \quad (6c)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + \sum_{\{k,j\} \in \bar{E}} y_{kj} + s x_i \geq s + 1, \quad \forall i \in V, \quad (6d)$$

$$x_i \in \{0, 1\}, \quad \forall i, j \in V, \quad (6e)$$

$$y_{ij} \in \{0, 1\}, \quad \forall \{i,j\} \in \bar{E}, \quad (6f)$$

where variable x_i , $i \in V$, indicates whether vertex i is included into an s -defective clique sought and y_{ij} , $\{i,j\} \in \bar{E}$, indicates if the corresponding induced subgraph contains the non-adjacent pair of vertices $\{i,j\}$. Binary vectors x and y of appropriate dimensions satisfy constraints (6b) and (6c) if and only if $S(x)$ is an s -defective clique in G ; see [39]. In addition, similar to the model above we can verify that $S(x)$ is enforced to be maximal via constraints (6d).

Then following the general approach for fractional 0-1 problems proposed in [47], we linearize our fractional 0-1 models to obtain equivalent MILP formulations. Specifically, define:

$$\xi := \frac{1}{\sum_{i \in V} b_i x_i} \quad \text{and} \quad z_i := \xi x_i, \quad \forall i \in V, \quad (7)$$

and let L and U be some lower and upper bounds for ξ , respectively. For example, in our computations, we set:

$$L := \frac{1}{\sum_{i \in V} b_i} \quad \text{and} \quad U := \frac{1}{\min\{b_i : i \in V\}}. \quad (8)$$

Then z_i can be expressed using the following classical set of linear constraints; see, for example, [2]:

$$z_i \geq L x_i, \quad z_i \leq U x_i, \quad z_i \leq \xi - L(1 - x_i), \quad z_i \geq \xi - U(1 - x_i), \quad \forall i \in V. \quad (9)$$

Applying the linearization variables and constraints described above, we obtain the following MILP formulation for MRsPP:

$$f_p^*(s) = \max \sum_{i \in V} a_i z_i, \quad (10a)$$

$$\text{s.t. } \sum_{i \in V} b_i z_i = 1, \quad (10b)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (\bar{d}_i - s)x_i \leq \bar{d}_i, \quad \forall i \in V, \quad (10c)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (s - 1)x_i \geq s, \quad \forall i \in V, \quad (10d)$$

$$z_i - L x_i \geq 0, \quad \xi + L x_i - z_i \geq L, \quad \forall i \in V, \quad (10e)$$

$$z_i - U x_i \leq 0, \quad \xi + U x_i - z_i \leq U, \quad \forall i \in V, \quad (10f)$$

$$L \leq \xi \leq U, \quad (10g)$$

$$x_i \in \{0, 1\}, \quad z_i \geq 0, \quad \forall i \in V. \quad (10h)$$

Similarly, MRsDCP can be formulated as the following MILP:

$$f_d^*(s) = \max \sum_{i \in V} a_i z_i, \quad (11a)$$

$$\text{s.t. } \sum_{i \in V} b_i z_i = 1, \quad (11b)$$

$$\sum_{\{i,j\} \in \bar{E}} y_{ij} \leq s, \quad (11c)$$

$$x_i + x_j - y_{ij} \leq 1, \quad \forall \{i, j\} \in \bar{E}, \quad (11d)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + \sum_{\{k, j\} \in \bar{E}} y_{kj} + sx_i \geq s + 1, \quad \forall i \in V, \quad (11e)$$

$$z_i - Lx_i \geq 0, \quad \xi + Lx_i - z_i \geq L, \quad \forall i \in V, \quad (11f)$$

$$z_i - Ux_i \leq 0, \quad \xi + Ux_i - z_i \leq U, \quad \forall i \in V, \quad (11g)$$

$$L \leq \xi \leq U, \quad (11h)$$

$$x_i \in \{0, 1\}, \quad z_i \geq 0, \quad \forall i \in V, \quad (11i)$$

$$y_{ij} \in \{0, 1\}, \quad \forall \{i, j\} \in \bar{E}. \quad (11j)$$

The linearized models (10) and (11) can be handled by an off-the-shelf MILP solver, for example, Gurobi [18]. Next, we describe additional valid inequalities that can be used to strengthen our models (namely, the quality of their LP relaxations) and improve the solver's performance.

3.3 | Upper bounds and valid inequalities

First, we derive two simple upper bounds on the sizes of any s -plex and s -defective clique in a graph, which are then exploited in our MILPs for providing additional valid inequalities.

Proposition 4 (Bound on the size of an s -plex). *Let $S \subseteq V$ be an s -plex in $G = (V, E)$. Then*

$$|S| \leq \beta_p^u(s) := \left\lfloor \frac{s + \sqrt{s^2 + 8|E|}}{2} \right\rfloor. \quad (12)$$

Proof. By its definition, in an s -plex, say S , every vertex is adjacent to at least $|S| - s$ vertices. Hence:

$$\frac{|S|(|S| - s)}{2} \leq |E|,$$

which can be solved as a quadratic inequality with respect to $|S|$ and results in (12). ■

Proposition 5 (Bound on the size of an s -defective clique). *Let $S \subseteq V$ be an s -defective clique in $G = (V, E)$. Then*

$$|S| \leq \beta_d^u(s) := \left\lfloor \frac{1 + \sqrt{1 + 8(s + |E|)}}{2} \right\rfloor. \quad (13)$$

Proof. By its definition, in an s -defective clique, say S , the number of edges should satisfy:

$$\frac{|S|(|S| - 1)}{2} - s \leq |E|,$$

which can be solved as a quadratic inequality with respect to $|S|$ and results in (13). ■

The key idea in the proofs provided above (i.e., linking edge density in the subgraph with the total number of edges in the graph, and then solving the resulting quadratic inequality) is similar in spirit to the approach applied for deriving the upper bounds on the sizes of the maximum quasi-clique and maximum 2-club in a graph; see [30, 43]. It is clear from the constructions used in the proofs that the quality of these upper bounds depends on the graph sparsity (and the quality should improve as the sparsity increases); we verify this observation computationally in Section 5.

Valid inequalities based on upper bounds. Inequalities (12) and (13) naturally produce the following additional constraints:

$$\sum_{i \in V} x_i \leq \beta_p^u, \quad (14a)$$

$$\sum_{i \in V} x_i \leq \beta_d^u, \quad (14b)$$

for MILPs (10) and (11), respectively. To simplify the notation, we drop the functional dependence of β_p^u and β_d^u on s in the right-hand side (14); recall (12) and (13).

Valid inequalities using variables corresponding to the ratio. Next, we apply the general method commonly used in nonlinear mixed 0-1 programming; see, for example, [15, 38, 41]. Specifically, for (10), we derive additional valid inequalities when multiplying both sides of (10c)–(10d) and (14a) by variable ξ and substituting ξx_i with z_i ; recall (7). Hence, we obtain the following three sets of linear constraints for (10):

$$\sum_{j \in V} (1 - a_{ij})z_j + (\bar{d}_i - s)z_i \leq \bar{d}_i \xi, \quad \forall i \in V, \quad (15a)$$

$$\sum_{j \in V} (1 - a_{ij})z_j + (s - 1)z_i \geq s \xi, \quad \forall i \in V, \quad (15b)$$

$$\sum_{i \in V} z_i \leq \beta_p^u \xi. \quad (15c)$$

For (11) there exist several possible ways of generating valid inequalities using the outlined general concept. Based on our preliminary computational experiments, we propose the following approach. After multiplying both sides of (11e) by ξ we have:

$$\sum_{j \in V} (1 - a_{ij})z_j + \sum_{\{k,j\} \in \bar{E}} y_{kj} \xi + sz_i \geq (s + 1)\xi, \quad \forall i \in V, \quad (16)$$

where the left-hand side of (16) contains a nonlinear term. Then we introduce auxiliary variables $t := \sum_{\{i,j\} \in \bar{E}} y_{ij}$ and $\omega := \xi t$.

Clearly, $t \geq 0$ by its definition; note also that additional upper and lower bounds for t and ξ are given by (11c) and (11h), respectively. Then the McCormick envelopes (see, e.g., [21]) describe a relaxation of w with the following set of linear constraints:

$$Lt \leq \omega \leq Ut \quad \text{and} \quad s\xi + Ut - Us \leq \omega \leq s\xi + Lt - Ls. \quad (17)$$

These derivations result in a set of valid inequalities for (11) given by the following set of constraints:

$$Lt \leq \omega \leq Ut \quad \text{and} \quad s\xi + Ut - Us \leq \omega \leq s\xi + Lt - Ls, \quad (18a)$$

$$\sum_{j \in V} (1 - a_{ij})z_j + \omega + sz_i \geq (s + 1)\xi, \quad \forall i \in V, \quad (18b)$$

$$\sum_{i \in V} z_i \leq \beta_d^u \xi, \quad (18c)$$

where (18b) is obtained from (16) using (17) and (18c) is derived from (14b) similar to (15c).

3.4 | MILPs for the average weight case

In this section, we consider two additional linearization methods for the special case of our class of problems, where $b_i = 1$ for all $i \in V$. Both methods are based on the general value-disjunction idea (see, e.g., [45]), namely, the fact that the term $\sum_{i \in V} x_i$ can take only integer values.

The *first linearization method* exploits the approach following the one used in [5], where variable ξ from (7) is represented as:

$$\xi := \sum_{k=1}^{\beta^u} \frac{1}{k} v_k, \quad \sum_{k=1}^{\beta^u} v_k = 1 \quad \text{and} \quad \sum_{k=1}^{\beta^u} k v_k = \sum_{i \in V} x_i, \quad (19)$$

where $v_k \in \{0, 1\}$ for all $k \in \{1, \dots, \beta^u\}$ and $\beta^u \in \{\beta_p^u, \beta_d^u\}$ depending on the considered problem. As in Section 3.3, to simplify the notation we drop the functional dependence of β_p^u and β_d^u on s ; recall (12) and (13), respectively.

Then for linearizing the objective functions in (5a) and (6a), we introduce the same set of 0-1 variables $z_i := \xi x_i$ as in (7), which is representable by (9). However, we need to redefine the appropriate lower and upper bounds; in our computations we use

$$L := \frac{1}{\beta^u} \quad \text{and} \quad U := 1, \quad (20)$$

respectively. Consequently, we obtain the following MILP formulation for MAsPP:

$$f_p^*(s) = \max \sum_{i \in V} a_i z_i, \quad (21a)$$

$$\text{s.t.} \quad \sum_{k=1}^{\beta_p^u} v_k = 1, \quad (21b)$$

$$\xi = \sum_{k=1}^{\beta_p^u} \frac{1}{k} v_k, \quad (21c)$$

$$\sum_{k=1}^{\beta_p^u} k v_k = \sum_{i \in V} x_i, \quad (21d)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (\bar{d}_i - s)x_i \leq \bar{d}_i, \quad \forall i \in V, \quad (21e)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (s - 1)x_i \geq s, \quad \forall i \in V, \quad (21f)$$

$$z_i - Lx_i \geq 0, \quad \xi + Lx_i - z_i \geq L, \quad \forall i \in V, \quad (21g)$$

$$z_i - Ux_i \leq 0, \quad \xi + Ux_i - z_i \leq U, \quad \forall i \in V, \quad (21h)$$

$$L \leq \xi \leq U, \quad (21i)$$

$$x_i \in \{0, 1\}, \quad v_k \in \{0, 1\}, \quad z_i \geq 0, \quad \forall i \in V, \quad (21j)$$

where we fully outline the linearized reformulation for completeness of our discussion. When comparing (10) and (21) we observe that the main difference between these MILPs is in their representations of the denominators; see (10b) and (21b)–(21d).

Our valid inequalities from Section 3.3, for example, (15a)–(15b), are also applicable for strengthening formulation (21). Finally, the MILP model for MAsDCP using (19) can be derived in a similar manner; we omit it for brevity.

The *second linearization method* is based on a binary representation of the term $\sum_{i \in V} x_i$. Specifically, define $\hat{\beta}^u := \lceil \log_2 \beta^u \rceil$ and let

$$\sum_{k=0}^{\hat{\beta}^u} 2^k w_k = \sum_{i \in V} x_i, \quad (22)$$

where $w_k \in \{0, 1\}$ for all $k \in \{0, \dots, \hat{\beta}^u\}$. Next, we introduce a new variable y given by:

$$y := \frac{\sum_{i \in V} a_i x_i}{\sum_{i \in V} x_i},$$

which, in view of (22), can be rewritten as the following nonlinear constraint:

$$y \sum_{k=0}^{\hat{\beta}^u} 2^k w_k = \sum_{i \in V} a_i x_i. \quad (23)$$

For the left-hand side of (23) we introduce additional variables $t_k := y w_k$, $k \in \{0, \dots, \hat{\beta}^u\}$, and the resulting nonlinear terms are linearizable using the same approach as in (9). Note that one possible setting for the lower and upper bounds for y (also used in our computations) is given by:

$$L := \frac{\sum_{i \in S} a_i}{|S|} \quad \text{and} \quad U := \max_{i \in V} a_i, \quad (24)$$

respectively, where S is an arbitrary maximal s -plex; S can be computed by simply solving a feasibility version of our MILP model.

Consequently, we obtain the following MILP formulation for MAsPP:

$$f_p^*(s) = \max \quad y, \quad (25a)$$

$$\text{s.t.} \quad \sum_{k=0}^{\hat{\beta}^u} 2^k w_k = \sum_{i \in V} x_i, \quad (25b)$$

$$\sum_{k=0}^{\hat{\beta}^u} 2^k t_k = \sum_{i \in V} a_i x_i, \quad (25c)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (\bar{d}_i - s)x_i \leq \bar{d}_i, \quad \forall i \in V, \quad (25d)$$

$$\sum_{j \in V} (1 - a_{ij})x_j + (s - 1)x_i \geq s, \quad \forall i \in V, \quad (25e)$$

$$L \leq y \leq U, \quad (25f)$$

$$t_k - Lw_k \geq 0, \quad y + Lw_k - t_k \geq L, \quad \forall k \in \{0, \dots, \hat{\beta}^u\}, \quad (25g)$$

$$t_k - Uw_k \leq 0, \quad y + Uw_k - t_k \leq U, \quad \forall k \in \{0, \dots, \hat{\beta}^u\}, \quad (25h)$$

$$x_i \in \{0, 1\}, \quad w_k \in \{0, 1\}, \quad t_k \geq 0, \quad \forall k \in \{0, \dots, \hat{\beta}^u\}. \quad (25i)$$

It is important to point out that one potential advantage of (25) is that it may need fewer binary variables than (21). In particular, when comparing (19) and (22), we observe that these linear 0-1 representations of the term $\sum_{i \in V} x_i$ require additional $O(\beta^u)$ and $O(\log \beta^u)$ binary variables, respectively. In Section 5 we provide computational comparisons of (21), (25) and the MILPs from Section 3.2.

We note that the outlined linearization method via (22) and (23) can be viewed as a special case of the approach discussed in [7], where a general class of multiple-ratio fractional 0-1 problems is considered. Finally, the MILP model for MAsDCP using (22) and (23) can be derived in a similar manner; it is omitted for brevity.

4 | PARAMETRIC METHODS

Single-ratio fractional programs can be solved by applying the parametric solution scheme [20]. We refer the reader to [8, 33] for detailed discussions of these methods (viz., the binary search and Newton's method) in the context of fractional combinatorial optimization. These methods are also used for detecting particular graph bipartitions with structural requirements in [44] and solving the MRCP problem in [37]. Hence, we adopt them for solving our class of problems with some subtle modifications as outlined below.

Formally, in our setting, the key idea behind these approaches is to consider the parametric problem of the form:

$$\begin{aligned} P(\lambda) := \max \quad & \sum_{i \in V} a_i x_i - \lambda \sum_{i \in V} b_i x_i, \\ \text{s.t.} \quad & x \in \bar{X}_\Pi, \end{aligned} \quad (26)$$

where $\lambda \in \mathbb{R}_+$ is some given parameter. We observe that (26) reduces to the problem of finding a *maximal* Π -set that has the maximum total weight of its nodes. Note that this problem is different from the maximum weight Π -set problem if negative weights of nodes are allowed. The weight of each node $i \in V$ is given by $a_i - \lambda b_i$, which can be both positive and negative depending on the value of the parameter λ .

Let $x^*(\lambda)$ be an optimal solution to (26), then $S(x^*(\lambda))$ is the corresponding maximal Π -set. It can be shown (see further details in [8, 20, 33]) that an optimal solution of the maximum ratio Π -set problem (4) can be obtained by finding parameter λ^* such that the optimal objective function value of the corresponding parametric problem $P(\lambda^*)$ is either zero, or sufficiently close to zero. This observation is one of the most fundamental results in fractional optimization literature, which can be applied for general single-ratio fractional programs (where the denominator and the numerator are not necessarily linear functions) as long as the denominator is strictly positive [13, 33]; recall our assumption on the values of a_i and b_i , $i \in V$, in Definition 2. For simplicity, in the remainder of this section we assume that $b_i > 0$ for all $i \in V$.

To summarize, solving (4) reduces to finding a root of function $P(\lambda)$. There are two common approaches in the literature for solving this problem, namely, the binary search and the Newton's method; see details below. Both of these approaches involve solution of a sequence of parametric problems as in (26). For our computations in Section 5, we solve (26) using an off-the-shelf solver with MILPs that are similar to those outlined in Section 3.2.

In particular, for s -plex we solve a modified version of (5), where we replace the objective function (5a) by a linear one as in (26); similarly, for s -defective clique we solve a modified version of (6), where we also replace the objective function. That is, feasible set $\bar{X}_\Pi$ in (26) is described by constraints (5b)–(5d) for MRsPP and by constraints (6b)–(6f) for MRsDCP, respectively. Naturally, some of our valid inequalities from Section 3.3, namely, constraints given by (14), are also applicable and hence, are also exploited in our computations.

Next, in order to have a relatively self-contained article, we briefly overview the pseudo-codes of the algorithms and some technical details that arise in the context of our problem. Note that since our base model (4) is a generalization of the MRCP problem, we mostly follow the notation used in [37]. Finally, for a detailed discussion of the convergence results for both methods we refer the reader to [33].

Binary search. The method is outlined in Algorithm 1. For the update structure of the appropriate lower and upper bounds we adopt the approach used in [37], which, in turn, follows [20]. The initial maximal Π -set in G , denoted by $\tilde{S}$ in Algorithm 1, can be computed, for example, using some heuristic; however, in our computations we simply solve feasibility versions of our MILP models.

Let C_1 and C_2 be two distinct maximal Π -sets in G such that

$$\frac{\sum_{i \in C_1} a_i}{\sum_{i \in C_1} b_i} \neq \frac{\sum_{i \in C_2} a_i}{\sum_{i \in C_2} b_i}.$$

Assume that a_i and b_i are positive integers for all $i \in V$, which is a common assumption in the related literature. Then following a standard approach in fractional optimization (see, e.g., an example in [28]), consider a gap (with respect to the objective function values) between C_1 and C_2 :

$$\left| \frac{\sum_{i \in C_1} a_i}{\sum_{i \in C_1} b_i} - \frac{\sum_{i \in C_2} a_i}{\sum_{i \in C_2} b_i} \right| = \left| \frac{\sum_{i \in C_1} a_i \sum_{i \in C_2} b_i - \sum_{i \in C_2} a_i \sum_{i \in C_1} b_i}{\sum_{i \in C_1} b_i \cdot \sum_{i \in C_2} b_i} \right| \geq \frac{1}{(\beta^u \max_{i \in V} b_i)^2}, \quad (27)$$

Algorithm 1. Binary search for the maximum ratio Π -set problem

Input: Graph $G = (V, E)$, with positive weights a_i and b_i for all $i \in V$, any maximal Π -set $\tilde{S}$ in G , and precision parameter $\epsilon > 0$

Output: A maximal Π -set S , $\lambda = \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$ such that $0 \leq \lambda^* - \lambda < \epsilon$, where λ^* denotes the optimal objective function value of the maximum ratio Π -set problem, and the number of iterations k

```

 $\lambda_1 \leftarrow \frac{\sum_{i \in \tilde{S}} a_i}{\sum_{i \in \tilde{S}} b_i}$  and  $\lambda_2 \leftarrow \max_{i \in V} \frac{a_i}{b_i}$ 
Compute  $P(\lambda_1)$  and  $x^*(\lambda_1)$  as well as  $P(\lambda_2)$  and  $x^*(\lambda_2)$  by solving (26)
 $LB \leftarrow \frac{\sum_{i \in S(x^*(\lambda_1))} a_i}{\sum_{i \in S(x^*(\lambda_1))} b_i}$  and  $UB \leftarrow \frac{\lambda_2 P(\lambda_1) - \lambda_1 P(\lambda_2)}{P(\lambda_1) - P(\lambda_2)}$ ;           /* initial bounds */
 $\lambda \leftarrow \lambda_1$  and  $S \leftarrow S(x^*(\lambda_1))$ ;                                     /* current solution */
 $k \leftarrow 1$ ;                                                                /* iteration counter */
while  $UB - LB \geq \epsilon$  do
     $\lambda \leftarrow \frac{1}{2}(LB + UB)$ 
    Compute  $P(\lambda)$  and  $x^*(\lambda)$  by solving (26)
    if  $P(\lambda) \geq 0$  then
         $\lambda_1 \leftarrow \lambda$  and  $S \leftarrow S(x^*(\lambda))$ 
         $LB \leftarrow \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$  and  $UB \leftarrow \frac{\lambda_2 P(\lambda_1) - \lambda_1 P(\lambda_2)}{P(\lambda_1) - P(\lambda_2)}$ 
    else
         $\lambda_2 \leftarrow \lambda$ 
         $UB \leftarrow \frac{\lambda_2 P(\lambda_1) - \lambda_1 P(\lambda_2)}{P(\lambda_1) - P(\lambda_2)}$ 
    end
     $k \leftarrow k + 1$ 
end
return  $S$ ,  $\lambda = \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$ ,  $k$ 

```

where β^u denotes an upper bound on a size of an optimal maximal Π -set in G , for example, $\beta^u \in \{\beta_p^u, \beta_d^u\}$ from Section 3.3 depending on the considered problem; alternatively, β^u can be simply set to $|V|$. Therefore, the derivations in (27) imply that if parameter ϵ in Algorithm 1 is set to any value smaller than $(\beta^u \max_{i \in V} b_i)^{-2}$, then Algorithm 1 is guaranteed to return a globally optimal solution to (4). Otherwise, the approach returns an ϵ -approximate solution. Note that if the integrality assumption does not hold, then a similar bound can be derived after multiplying all parameters by a sufficiently large constant parameter.

Newton's method. The method is outlined in Algorithm 2. Let S and C^* be maximal Π -sets returned by Algorithm 2 and an optimal solution of (4), respectively. Then, as observed in [37],

$$0 \leq \frac{\sum_{i \in C^*} a_i}{\sum_{i \in C^*} b_i} - \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i} = \frac{\sum_{i \in C^*} a_i - \lambda \sum_{i \in C^*} b_i}{\sum_{i \in C^*} b_i} < \frac{\epsilon}{\min_{i \in V} b_i}, \quad (28)$$

where λ denotes the objective function value corresponding to S and ϵ is the input parameter; see Algorithm 2. Combining (27) and (28) with the assumption that a_i and b_i are positive integers for all $i \in V$, we conclude that if parameter ϵ in Algorithm 2 is set to any value smaller than $(\beta^u \max_{i \in V} b_i)^{-2} \min_{i \in V} b_i$, then Algorithm 2 is guaranteed to return a globally optimal solution to (4). Otherwise, the approach returns an $(\epsilon / \min_{i \in V} b_i)$ -approximate solution.

5 | COMPUTATIONAL EXPERIMENTS

This section presents the result of our computational study with randomly generated and real-life network instances. The discussion is organized as follows. In Section 5.1, we describe the network instances used and make some qualitative observations concerning optimal solutions obtained in several illustrative examples. In Section 5.2, we study the performance of the proposed MILP models. In particular, we focus on exploring the efficacy of our valid inequalities and the scalability of the developed linearized models. Then, in Section 5.3, we compare the performance of our MILP models enhanced with the valid inequalities against the parametric methods. Finally, in Section 5.4, we evaluate the specialized MILP models for the average weight cases of our problems.

Algorithm 2. Newton's method for the maximum ratio Π -set problem

Input: Graph $G = (V, E)$, with positive weights a_i and b_i for all $i \in V$, any maximal Π -set $\tilde{S}$ in G , and precision parameter $\epsilon > 0$

Output: A maximal Π -set S and $\lambda = \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$ such that $0 \leq \lambda^* - \lambda < \epsilon / \min_{i \in V} b_i$, where λ^* denotes the optimal objective function value of the maximum ratio Π -set problem, and the number of iterations k

```

 $\lambda \leftarrow \frac{\sum_{i \in \tilde{S}} a_i}{\sum_{i \in \tilde{S}} b_i}$ 
Compute  $P(\lambda)$  and  $x^*(\lambda)$  by solving (26)
 $S \leftarrow S(x^*(\lambda))$  and  $\lambda \leftarrow \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$ ;           /* current solution */
 $k \leftarrow 1$ ;                                           /* iteration counter */
while  $P(\lambda) \geq \epsilon$  do
    Compute  $P(\lambda)$  and  $x^*(\lambda)$  by solving (26)
     $S \leftarrow S(x^*(\lambda))$  and  $\lambda \leftarrow \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}$ 
     $k \leftarrow k + 1$ 
end
return  $S, \lambda = \frac{\sum_{i \in S} a_i}{\sum_{i \in S} b_i}, k$ 

```

The experiments discussed below were performed using an HP workstation with CPU 3.60 GHz and RAM 16 GB. All MIP models were solved using Gurobi Optimizer 9.1 [18] with Python 3.8 interface and NetworkX library [19]. For the parametric methods we use $\epsilon = 10^{-3}$ as the stopping criterion; the MIP solver is used with the default parameter settings.

5.1 | Test instances and optimal solutions obtained

The test bed we use for the experiments consists of uniform random graphs (Sets A and B, described below) as well as real-life instances originally proposed for the MRCP in [37] (Sets C–E). More specifically, we consider the following sets of instances.

- *Set A* consists of 60 uniform random graphs, with a_i and b_i drawn from the discrete uniform distribution between 1 and 100. A uniform random graph $G(n, p)$ is defined by the number of vertices n and the probability p that an arbitrary pair of vertices forms an edge [14]. We consider $(n, p) \in \{100, 200, 300\} \times \{0.1, 0.5\}$ and generate 10 random graphs for each considered combination.
- *Set B* consists of 60 uniform random graphs, with a_i and b_i for each vertex constrained to be within 1.5% of each other. The random graphs are generated in the same fashion as in Set A.
- *Set C* (market graphs) was initially constructed in [37] based on stock market data. In these graphs, the financial instruments are presented as vertices of the graph. The price of each stock at the beginning and at the end of a 500-day period corresponds to b_i and a_i , where i is the corresponding stock. The edge between two nodes in these graphs exists *iff* the correlation of price fluctuations for each pair of stocks is negative.
- *Set D* (wind graphs) uses wind speed data for establishing locations of wind energy farms. The set consists of 3 graphs with 500 vertices representing 250 prospective wind farm locations and 2 available options for wind turbine height (80 and 100 m). The edge between two nodes exists *iff* the wind speed correlation between two corresponding locations and height is negative. For each node i of a wind graph, a_i is the average wind speed at the corresponding location/height, and b_i is proportional to the average annual cost of the wind turbine for the corresponding location/height. A more detailed description of these instances is given in [37].
- *Set E* consists of graphs from the 2nd and 10th DIMACS Implementation Challenges [11, 12]:
 - brock200_1, brock200_2, brock200_3, brock200_4: instances from the generator that attempts to hide cliques in a graph where the expected clique size is much smaller.
 - c-fat-200-1, c-fat-500-1, c-fat-500-2: problems based on fault diagnosis problems.
 - karate ($|V| = 34, |E| = 78$): a social network of friendships between 34 members of a karate club.
 - dolphins ($|V| = 62, |E| = 159$): an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand.
 - polbooks ($|V| = 105, |E| = 441$): a network of books about U.S. politics published around the time of the 2004 presidential election and sold by the online bookseller Amazon.com; edges between books correspond to frequent co-purchasing of the books by the same buyers.

TABLE 1 Description of the test instances, their optimal objective function values, and the corresponding maximal s -plex sizes for $s = 1, 2, 3$

				$f_p^*(s)$			Size		
Graph	$ V $	$ E $	Density (%)	$s = 1$	$s = 2$	$s = 3$	$s = 1$	$s = 2$	$s = 3$
Set A: Uniform random graphs with uncorrelated weights									
r-100-0.1-A	100	504.80	10.20	11.56	6.67	5.58	2.00	3.50	4.60
r-200-0.1-A	200	1998.60	10.04	14.36	11.65	9.10	2.30	4.00	5.00
r-300-0.1-A	300	4481.50	9.99	15.86	13.54	11.90	2.90	4.00	5.00
r-100-0.5-A	100	2470.90	49.92	5.61	3.87	3.55	5.00	8.00	9.80
r-200-0.5-A	200	9952.50	50.06	8.28	5.48	4.83	6.30	8.70	10.90
r-300-0.5-A	300	22,414.30	49.98	9.26	6.10	5.61	6.60	9.20	12.00
Set B: Uniform random graphs with constrained vertex ratio									
r-100-0.1-B	100	491.80	9.92	1.14	1.09	1.07	2.20	3.10	4.40
r-200-0.1-B	200	2004.70	10.07	1.26	1.13	1.12	2.10	3.90	5.00
r-300-0.1-B	300	4485.60	10.00	1.22	1.12	1.18	2.30	4.00	5.00
r-100-0.5-B	100	2480.90	50.12	1.09	1.05	1.05	5.10	7.80	9.70
r-200-0.5-B	200	9937.00	49.96	1.08	1.06	1.05	5.90	8.50	11.00
r-300-0.5-B	300	22,403.00	49.95	1.12	1.09	1.07	6.30	9.00	11.40
Set C: Stock market graphs									
market-1	500	23,116	18.53	21.21	14.55	13.55	4	6	9
market-2	500	26,431	21.19	23.69	20.00	16.65	5	7	9
market-3	500	27,704	22.21	13.92	10.02	8.03	4	6	9
market-4	500	25,819	20.70	18.74	14.42	12.48	5	6	9
market-5	500	26,196	21.00	14.19	11.15	8.13	4	6	9
Set D: Wind energy graphs									
wind-2004	500	10,277	8.24	96,113.00	93,752.33	92,782.33	3	6	6
wind-2005	500	10,516	8.43	94,686.58	94,380.00	92,912.80	2	4	6
wind-2006	500	9681	7.76	98,471.50	95,451.00	94,108.90	2	4	6
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$									
brock200_1	200	14,834	74.54	6.88	5.02	4.46	12	17	21
brock200_2	200	9876	49.63	15.30	10.17	8.07	6	8	12
brock200_3	200	12,048	60.54	13.89	10.70	7.66	8	11	15
brock200_4	200	13,089	65.77	8.63	7.45	5.58	8	13	16
c-fat-200-1	200	1534	7.71	1.22	1.22	1.22	10	10	10
c-fat-500-1	500	4459	3.57	1.26	1.26	1.26	12	12	12
c-fat-500-2	500	9139	7.32	1.07	1.07	1.07	26	26	26
karate	34	78	13.90	16.50	10.67	6.95	3	4	5
dolphins	62	159	8.41	11.60	3.67	8.00	2	4	4
polbooks	105	441	8.08	12.25	6.79	6.79	3	5	5
adjnoun	112	425	6.84	19.55	7.92	17.08	2	3	4
football	115	613	9.35	9.55	7.70	7.75	2	3	4
celegans	453	2025	2.41	6.32	5.49	4.88	2	5	4
email	1133	5451	0.85	57.15	14.75	50.55	2	4	4

Note: For random graphs (Sets A and B), the average values for each set of 10 instances are reported.

- adjnoun($|V| = 112$, $|E| = 425$): an adjacency network of common adjectives and nouns in the novel titled *David Copperfield* by Charles Dickens.
- football($|V| = 115$, $|E| = 613$): a network representing American football games between Division IA colleges during regular Fall 2000 season.
- jazz($|V| = 198$, $|E| = 2742$): jazz musicians network.
- celegans($|V| = 453$, $|E| = 2025$): C. elegans metabolic network.
- email($|V| = 1133$, $|E| = 5451$): a network of email interchanges between members of the Univeristy Rovira i Virgili (Tarragona).

The weights for these graphs are set to $a_i = i$ and $b_i = |V| - i + 1$ for all $i \in V$. For the average case study, we use some of these graphs with $a_i = i$ and $b_i = 1$ for all $i \in V$.

Tables 1 and 2 describe the basic characteristics of the considered instances, such as the number $|V|$ of nodes, the number $|E|$ of edges, and the edge density, as well as the optimal objective values and the sizes of optimal solutions found for the

TABLE 2 Description of the test instances, their optimal objective function values, and the corresponding maximal s -defective clique sizes for $s = 0, 1, 2$

Graph	V	E	Density (%)	$f_d^*(s)$			Size		
				$s = 0$	$s = 1$	$s = 2$	$s = 0$	$s = 1$	$s = 2$
Set A: Uniform random graphs with uncorrelated weights									
r-100-0.1-A	100	504.80	10.20	11.56	31.78	20.54	2.00	2.20	3.00
r-200-0.1-A	200	1998.60	10.04	14.36	36.65	39.00	2.30	2.50	3.00
r-300-0.1-A	300	4481.50	9.99	15.86	38.40	44.20	2.90	2.50	3.00
r-100-0.5-A	100	2470.90	49.92	5.61	27.94	9.33	5.00	2.20	5.20
r-200-0.5-A	200	9952.50	50.06	8.28	10.47	15.03	6.30	5.90	5.90
r-300-0.5-A	300	22,414.30	49.98	9.26	13.92	17.28	6.60	6.30	6.40
Set B: Uniform random graphs with constrained vertex ratio									
r-100-0.1-B	100	491.80	9.92	1.14	1.38	1.40	2.20	2.10	3.00
r-200-0.1-B	200	2004.70	10.07	1.26	1.48	1.44	2.10	2.20	3.00
r-300-0.1-B	300	4485.60	10.00	1.22	1.36	1.38	2.30	2.50	3.00
r-100-0.5-B	100	2480.90	50.12	1.09	1.38	1.13	5.10	2.00	5.50
r-200-0.5-B	200	9937.00	49.96	1.08	1.10	1.16	5.90	5.60	5.70
r-300-0.5-B	300	22,403.00	49.95	1.12	1.23	1.20	6.30	6.30	6.40
Set C: Stock market graphs									
market-1	500	23,116	18.53	21.21	49.74	66.45	4	4	4
market-2	500	26,431	21.19	23.69	44.64	55.38	5	5	4
market-3	500	27,704	22.21	13.92	19.27	26.28	4	5	4
market-4	500	25,819	20.70	18.74	29.03	37.50	5	4	4
market-5	500	26,196	21.00	14.19	28.16	42.86	4	4	4
Set D: Wind energy graphs									
wind-2004	500	10,277	8.24	96,113.00	96,236.67	97,002.00	3	3	3
wind-2005	500	10,516	8.43	94,686.58	99,876.00	99,425.00	2	2	3
wind-2006	500	9681	7.76	98,471.50	101,611.50	100,302.67	2	2	3
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$									
brock200_1	200	14,834	74.54	6.88	8.85	9.77	12	10	12
brock200_2	200	9876	49.63	15.30	19.44	23.61	6	6	6
brock200_3	200	12,048	60.54	13.89	17.27	19.10	8	7	8
brock200_4	200	13,089	65.77	8.63	12.93	13.78	8	7	10
c-fat-200-1	200	1534	7.71	1.22	79.40	85.14	10	2	3
c-fat-500-1	500	4459	3.57	1.26	199.40	213.71	12	2	3
c-fat-500-2	500	9139	7.32	1.07	199.40	213.71	26	2	3
karate	34	78	13.90	16.50	13.00	10.60	3	4	4
dolphins	62	159	8.41	11.60	41.00	14.75	2	2	3
polbooks	105	441	8.08	12.25	52.00	52.00	3	2	3
adjnoun	112	425	6.84	19.55	74.33	27.25	2	2	3
football	115	613	9.35	9.55	76.33	42.50	2	2	3
celegans	453	2025	2.41	6.32	301.67	150.33	2	2	3
email	1133	5451	0.85	57.15	—	—	2	—	—

Note: For random graphs (Sets A and B) the average values for each set of 10 instances are reported. No results were obtained for email instance due to the out-of-memory error.

maximum ratio s -plex problem with $s = 1, 2, 3$ (Table 1) and the maximum ratio s -defective clique problem with $s = 0, 1, 2$ (Table 2). Note that for the random graphs (Sets A and B), we specify the average values for each batch of 10 instances. This is also the case for the remaining tables in this article that involve the random graphs.

To conclude this subsection, we make some qualitative observations concerning the optimal solutions reported in Tables 1 and 2. In most instances, the maximum ratio s -defective clique problem has better optimal objective value than the maximum ratio clique or s -plex problems. Notable exceptions include karate, adjnoun, and dolphins graphs. For email, adjnoun, football, and dolphins instances, even though the objective value is the highest for $s = 1$, the objective for $s = 3$ is higher than one for $s = 2$. However, some of the resulting subgraphs in case of $s = 3$ have only 4 vertices, which makes those subgraphs possibly not sufficiently “cohesive,” since a vertex with just one neighbor can still be in the 3-plex.

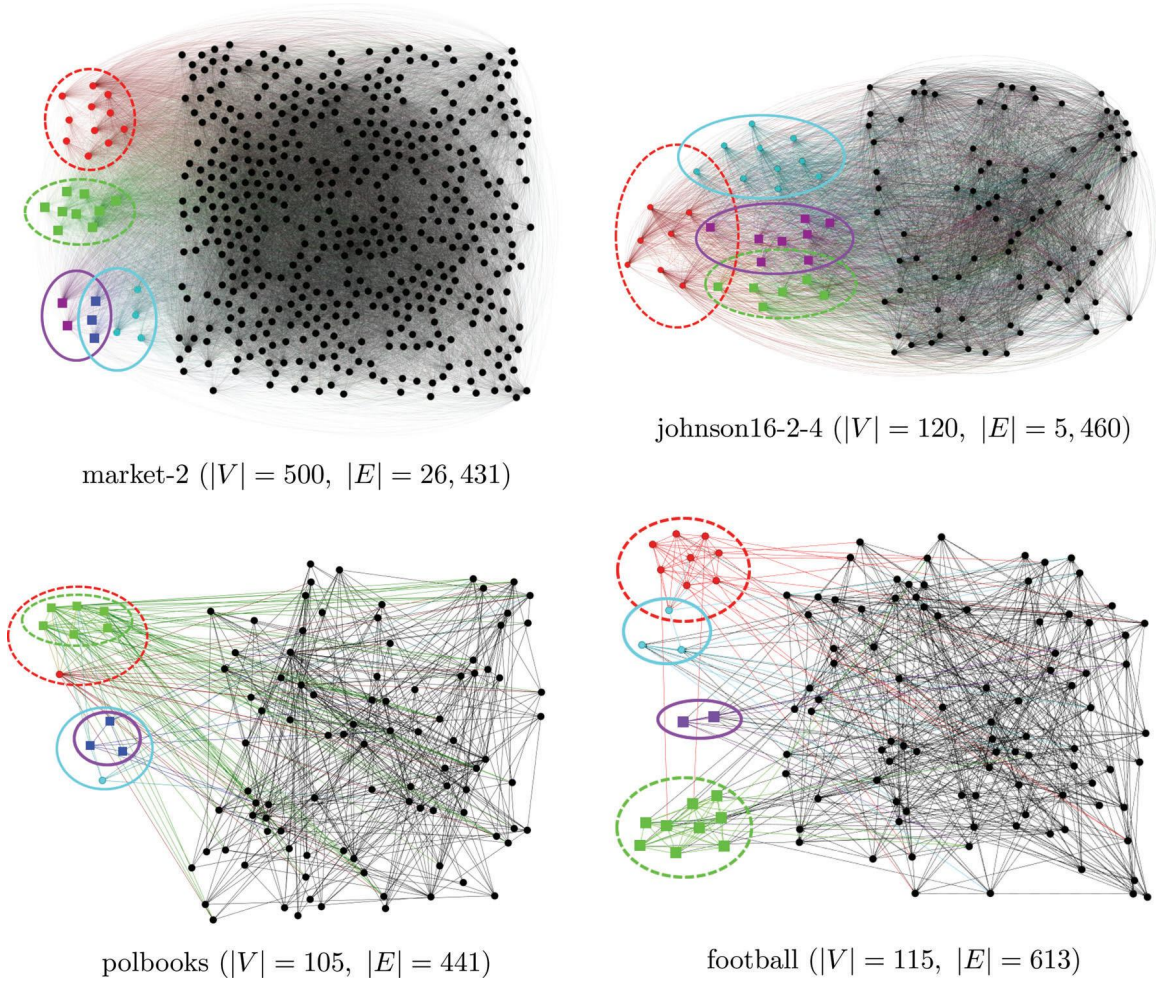


FIGURE 2 Comparison of cardinality and structure of the maximum clique and 2-plex as well as maximum ratio maximal clique and 2-plex. The maximum clique and 2-plex are circled in green and red dashed lines, respectively. The maximum ratio maximal clique and 2-plex are circled in violet and light blue lines. The nodes that belong to the considered cliques are represented with squares, whereas the nodes that belong only to the considered 2-plexes are represented with circles. Blue color is used for the nodes that belong to both the maximum ratio maximal clique and 2-plex

Generally speaking, the considered problems produce optimal solutions that can be significantly different, and each of the obtained clusters could be of potential interest from an applied perspective. This point is highlighted in Figure 2 by illustrating the maximum clique (1-plex), maximum 2-plex, maximum ratio clique, and maximum ratio 2-plex in four different graphs from the considered set of instances. It can be seen from this illustration that the level of overlap between these optimal structures may vary significantly from one instance to another.

5.2 | Exploring the effect of valid inequalities on performance of MILPs

In this subsection, we compare the performance of MILPs (10) and (11) with and without the valid inequalities (14a) and (15) as well as (14b) and (18), respectively. The results are reported in Table 3 (for the maximum ratio 2-plex) and Table 4 (for the maximum ratio 2-defective clique problem). More specifically, we report the quality of LP relaxations as well as running times.

The results demonstrate a significant improvement in performance by using the proposed valid inequalities for both problems. In fact, valid inequalities were not helpful only in 7 out of 284 considered cases: the three *c-fat* instances and *celegans* graph for the maximum ratio 2-plex problem and the two *r-300-0.1-A* graphs and *dolphins* instance for the maximum ratio 2-defective clique problem. In all the remaining cases the inclusion of valid inequalities led to speed-up, in some cases by several orders of magnitude.

Notably, we observe in Table 4 that for *market* and *wind* instances addition of the valid inequalities helped to close the MIP gap for the maximum ratio 2-defective clique problem within the chosen time limit of 7200 s. Furthermore, in both Tables 3 and 4, we also observe significant improvements in the quality of LP relaxations for many instances, which explains superior performance resulting from inclusion of the valid inequalities. Based on these observations, we use the MILPs enhanced by the proposed valid inequalities in the more comprehensive experiments reported in the next subsection.

TABLE 3 Performance of MILP (10) for the maximum ratio 2-plex problem with and without valid inequalities (14a) and (15).

Graph	V	E	β^u	Size	Without valid inequalities		With valid inequalities	
					f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time
Set A: Uniform random graphs with uncorrelated weights								
r-100-0.1-A	100	504.80	32.30	3.50	6.87	0.69	1.96	0.28
r-200-0.1-A	200	1998.60	63.60	4.00	7.21	2.02	2.94	1.21
r-300-0.1-A	300	4481.50	95.30	4.00	5.60	14.93	3.22	2.95
r-100-0.5-A	100	2470.90	70.80	8.00	13.29	3.60	2.58	1.34
r-200-0.5-A	200	9952.50	141.70	8.70	9.80	91.85	3.46	10.01
r-300-0.5-A	300	22,414.30	212.40	9.20	11.46	852.20	4.02	44.46
Set B: Uniform random graphs with constrained vertex ratio								
r-100-0.1-B	100	491.80	31.90	3.10	1.51	0.56	1.24	0.34
r-200-0.1-B	200	2004.70	63.90	3.90	1.52	33.68	1.25	1.62
r-300-0.1-B	300	4485.60	95.30	4.00	1.47	140.64	1.25	3.36
r-100-0.5-B	100	2480.90	71.10	7.80	1.50	3.49	1.08	1.19
r-200-0.5-B	200	9937.00	141.30	8.50	1.81	104.33	1.14	8.62
r-300-0.5-B	300	22,403.00	212.00	9.00	1.75	723.72	1.19	43.11
Set C: Stock market graphs								
market-1	500	23,116	216	6	14.95	578.88	6.09	68.18
market-2	500	26,431	230	7	10.73	339.91	3.74	33.77
market-3	500	27,704	236	6	8.69	540.29	3.19	29.55
market-4	500	25,819	228	6	8.42	369.79	3.97	174.92
market-5	500	26,196	229	6	10.38	401.35	4.69	51.55
Set D: Wind energy graphs								
wind-2004	500	10,277	147	6	1.07	252.25	1.03	20.01
wind-2005	500	10,516	146	4	1.06	62.23	1.04	8.56
wind-2006	500	9681	140	4	1.09	69.11	1.05	14.79
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$								
brock200_1	200	14,834	173	17	38.05	5002.58	3.18	408.68
brock200_2	200	9876	141	8	19.24	72.35	3.49	6.02
brock200_3	200	12,048	156	11	18.62	112.97	18.06	14.72
brock200_4	200	13,089	162	13	26.68	336.56	25.85	34.77
c-fat-200-1	200	1534	56	10	8.43	0.32	8.03	0.94
c-fat-500-1	500	4459	95	12	13.62	3.20	12.98	10.07
c-fat-500-2	500	9139	136	24	14.37	1.73	14.07	5.32
karate	34	78	13	4	2.95	0.04	1.00	0.01
dolphins	62	159	18	4	16.30	0.41	8.21	0.19
polbooks	105	441	33	5	15.45	0.76	6.05	0.44
adjnoun	112	425	30	3	13.54	0.51	5.95	0.50
football	115	613	36	3	14.65	0.78	5.82	0.40
celegans	453	2025	71	5	82.20	7.53	21.79	18.63
email	1133	5451	105	4	76.75	1428.63	38.35	196.97

Note: Column " f_{LP}^*/f_{MIP}^* " presents the ratio of the optimal objective function value f_{LP}^* of the LP relaxation to the optimal objective function value f_{MIP}^* of (10). The solution times are in seconds; the best results are in bold.

Additionally to illustrate contributions of each subset of valid inequalities we compare the performance of MILPs (10) and (11) when only a particular subset of valid inequalities is used. We report these results in Table 5 (for the maximum ratio 2-plex problem) and Table 6 (for the maximum ratio 2-defective clique problem). We observe in these tables that there are multiple instances, where only a subset of valid inequalities results in a substantial improvement of the MIP running time; also, constraints (14) by themselves are not effective due to the low quality of the upper bounds used in their right-hand sides. However, it could be argued that for the most difficult instances, it is preferable to use all of our valid inequalities.

Finally, when comparing the results between the maximum ratio 2-plex and maximum ratio 2-defective clique formulations, it is interesting to note that the running times tend to be lower for the former, while the LP relaxations tend to be stronger for the latter. The tighter LP relaxation bounds for the 2-defective clique can be explained by observing that the projection of the feasible set of (11) onto x -variables yields a set that is expected to be "smaller" than the feasible set of (10) for $s = 2$. Even though there

TABLE 4 Performance of MILP (11) for the maximum ratio 2-defective clique problem with and without valid inequalities (14b) and (18)

Graph	V	E	β^u	Size	Without valid inequalities		With valid inequalities	
					f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time
Set A: Uniform random graphs with uncorrelated weights								
r-100-0.1-A	100	504.80	31.80	3.00	2.64	109.81	1.32	33.51
r-200-0.1-A	200	2998.60	63.30	3.00	2.35	849.59	1.41	161.58
r-300-0.1-A	300	4481.50	94.60	3.00	1.76	2589.30	1.16	1521.61 ² (0.17%)
r-100-0.5-A	100	2470.90	70.20	5.20	5.98	8.48	1.75	2.62
r-200-0.5-A	200	9952.50	141.10	5.90	4.23	103.11	1.81	91.92
r-300-0.5-A	300	22,414.30	211.60	6.20	4.62	471.37	2.05	172.65
Set B: Uniform random graphs with constrained vertex ratio								
r-100-0.1-B	100	491.80	31.40	3.00	1.17	48.22	1.17	20.29
r-200-0.1-B	200	2004.70	63.30	3.00	1.19	604.37	1.07	143.02
r-300-0.1-B	300	4485.60	94.70	3.00	1.35	4133.01 ⁵ (0.19%)	1.07	1925.47 ¹ (0.09%)
r-100-0.5-B	100	2480.90	70.30	5.50	6.73	56.67	1.06	2.53
r-200-0.5-B	200	9937.00	141.00	5.60	1.63	829.48	1.63	68.93
r-300-0.5-B	300	22,403.00	211.70	6.40	1.59	6003.11 ⁴ (0.52%)	1.15	189.21
Set C: Stock market graphs								
market-1	500	23,116	215	4	3.43	TE (3.78%)	2.04	417.35
market-2	500	26,431	230	4	4.81	TE (2.77%)	1.54	460.28
market-3	500	27,704	235	4	7.54	TE (2.73%)	1.24	343.64
market-4	500	25,819	227	4	3.24	TE (3.44%)	1.89	795.64
market-5	500	26,196	229	4	2.70	TE (2.22%)	1.58	352.15
Set D: Wind energy graphs								
wind-2004	500	10,277	146	3	1.04	TE (0.03%)	1.02	456.68
wind-2005	500	10,516	145	3	1.01	TE (0.01%)	1.003	4459.78
wind-2006	500	9681	139	3	1.04	TE (0.02%)	1.01	3890.83
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$								
brock200_1	200	14,834	172	12	20.02	2794.08	2.26	208.19
brock200_2	200	9876	141	6	8.38	632.18	2.18	15.53
brock200_3	200	12,048	156	11	10.33	940.61	2.01	16.68
brock200_4	200	13,089	162	13	14.27	1646.21	2.49	46.46
c-fat-200-1	200	1534	55	3	2.34	323.86	1.00	2.90
c-fat-500-1	500	4459	94	3	2.34	1435.97	1.00	39.57
c-fat-500-2	500	9139	135	3	2.34	1303.65	1.00	40.18
karate	34	78	13	4	3.09	0.79	1.55	0.40
dolphins	62	159	18	3	4.13	8.53	2.78	25.29
polbooks	105	441	33	3	2.00	14.78	1.00	0.81
adjnoun	112	425	29	3	4.07	72.66	2.73	45.27
football	115	613	35	3	2.68	29.39	1.80	9.97
celegans	453	2025	70	3	3.01	TE (1.06%)	2.01	5845.15
email	1133	5451	104	–	OM	–	OM	–

Note: All times are seconds; the time limit is 7200 s. Let f_{LP}^* be the optimal objective value of the LP relaxation of (11), and f_{MIP}^* be the optimal objective value of (11) or the value of the best solution found if the time limit was exceeded, in which case “TE” (MIP gap) is used as time for Sets C–E. For random graphs, if the time limit was exceeded for at least one instance in the batch, the average MIP gap is presented. The number of such instances out of 10 is shown in the superscript after the average solution time. “OM” indicates out of memory error.

may exist some 2-defective cliques that are not 2-plexes, it is more common to observe 2-plexes that are not 2-defective cliques. As for the higher running times observed for 2-defective cliques, they are a consequence of a significantly higher number of binary variables and constraints in (11) compared to (10). This issue will also be discussed in the next subsection.

5.3 | Comparing the MILP approach and the parametric methods

In this subsection, we present the numerical results for the three proposed approaches: MILP (with all the valid inequalities discussed above), the binary search, and Newton’s method for $s = 1, 2, 3$ for the MRsPP and for $s = 0, 1, 2$ for the MRsDCP.

TABLE 5 Performance of MILP (10) for the maximum ratio 2-plex problem when only a subset of valid inequalities is used that is, only (14a), only (14a) and (15c), and only (15a)–(15b)

Graph	β^u	Size	Only (14a)	Time	Only (14a) and (15c)	Time	Only (15a)–(15b)	Time
			f_{LP}^*/f_{MIP}^*		f_{LP}^*/f_{MIP}^*		f_{LP}^*/f_{MIP}^*	
Set A: Uniform random graphs with uncorrelated weights								
r-100-0.1-A	32.30	3.50	8.77	0.69	9.12	0.67	3.78	0.35
r-200-0.1-A	63.60	4.00	7.19	2.35	7.07	4.29	2.94	1.26
r-300-0.1-A	95.30	4.00	6.04	16.46	5.77	18.03	3.18	3.38
r-100-0.5-A	70.80	8.00	13.29	3.67	13.29	2.83	2.57	1.59
r-200-0.5-A	141.70	8.70	11.52	97.78	11.10	95.39	3.56	8.91
r-300-0.5-A	212.40	9.20	11.46	887.92	11.46	766.54	4.02	36.03
Set B: Uniform random graphs with constrained vertex ratio								
r-100-0.1-B	31.90	3.10	1.63	0.54	1.57	0.63	1.18	0.32
r-200-0.1-B	63.90	3.90	1.52	33.85	1.52	15.80	1.25	1.54
r-300-0.1-B	95.30	4.00	1.53	143.55	1.52	139.61	1.25	3.75
r-100-0.5-B	71.10	7.80	1.50	3.53	1.50	3.20	1.08	1.27
r-200-0.5-B	141.30	8.50	1.81	104.30	1.81	101.89	1.14	9.47
r-300-0.5-B	212.00	9.00	1.75	737.84	1.75	666.12	1.19	43.96
Set C: Stock market graphs								
market-1	216	6	14.95	603.14	14.95	562.56	6.09	763.78
market-2	230	7	10.73	355.13	10.73	910.80	3.74	28.26
market-3	236	6	8.69	541.48	8.69	486.40	3.19	37.13
market-4	228	6	8.42	369.33	8.42	600.83	3.97	30.97
market-5	229	6	10.38	401.48	10.38	51.79	4.69	59.11
Set D: Wind energy graphs								
wind-2004	147	6	1.07	253.76	1.07	296.31	1.03	19.93
wind-2005	146	4	1.06	61.77	1.06	65.61	1.04	9.28
wind-2006	140	4	1.09	69.32	1.09	83.55	1.05	22.28
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$								
brock200_1	173	17	38.05	5258.58	38.05	4154.92	3.18	490.30
brock200_2	141	8	19.24	72.17	19.24	60.52	3.49	5.13
brock200_3	156	11	18.17	112.09	18.17	113.56	2.34	13.78
brock200_4	162	13	25.94	339.22	25.94	310.25	3.17	36.43
c-fat-200-1	56	10	8.43	0.34	8.43	0.32	1.00	1.02
c-fat-500-1	95	12	12.98	3.27	13.05	2.87	12.44	9.73
c-fat-500-2	136	24	14.07	1.71	14.21	1.50	14.65	5.44
karate	13	4	1.00	0.04	2.91	0.04	1.00	0.01
dolphins	18	4	16.30	0.35	16.12	0.32	8.21	0.22
polbooks	33	5	15.16	0.59	16.02	0.72	6.05	0.52
adjnoun	30	3	13.85	0.49	13.75	0.26	5.95	0.44
football	36	3	14.65	0.69	14.65	0.74	5.82	0.34
celegans	71	5	82.19	8.23	82.19	203.72	21.79	28.72
email	105	4	76.75	1423.63	76.75	419.51	38.35	718.86

Note: Column " f_{LP}^*/f_{MIP}^* " presents the ratio of the optimal objective function value f_{LP}^* of the LP relaxation to the optimal objective function value f_{MIP}^* of (10). The solution times are in seconds.

The valid inequalities (14) based on upper bounds are also used for solving (26) in both the binary search and Newton's method. Note that this computational study was used to obtain the optimal objective function values and sizes of the optimal maximal s -plexes and s -defective cliques in Tables 1 and 2. The running times taken by each method are presented in Table 7 (for the MRsPP) and Table 8 (for the MRsDCP).

The key takeaways from this computational study are summarized next.

- The overall performance of the considered methods is rather promising. We were able to compute optimal solutions of all the test bed instances for the MRsPP, and of all but one (email) instances for the MRsDCP.
- Solving MRsDCP takes more time due to a much larger number of binary variables and constraints (6d). The number of those constraints depends on the number of non-adjacent pairs of vertices and is large in low-density graphs. Note that real-life graphs are typically sparse.

TABLE 6 Performance of MILP (10) for the maximum ratio 2-defective clique problem when only a subset of valid inequalities is used that is, only (14b), only (14b) and (18c), and only (18a)–(18b)

Graph	β^u	Size	Only (14b)		Only (14b) and (18c)		Only (18b)	
			f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time
Set A: Uniform random graphs with uncorrelated weights								
r-100-0.1-A	31.80	3.00	2.64	115.61	2.64	108.07	1.32	35.07
r-200-0.1-A	63.30	3.00	2.35	896.28	2.35	678.42	1.41	121.96
r-300-0.1-A	94.60	3.00	1.76	1897.29	1.76	1824.18	1.16	1328.38
r-100-0.5-A	70.20	5.20	5.98	90.17	5.98	75.99	1.75	2.58
r-200-0.5-A	141.10	5.90	4.23	589.77	4.23	641.14	1.81	18.50
r-300-0.5-A	211.60	6.20	4.62	1058.81	4.37	3817.89	2.05	208.09
Set B: Uniform random graphs with constrained vertex ratio								
r-100-0.1-B	31.40	3.00	1.17	49.34	1.17	52.59	1.17	18.09
r-200-0.1-B	63.30	3.00	1.19	578.55	1.19	361.65	1.07	124.67
r-300-0.1-B	94.70	3.00	1.36	3648.22 ³ (0.67%)	1.36	5248.64 ³ (0.41%)	1.07	2091.61
r-100-0.5-B	70.30	3.00	4.48	60.61	4.48	74.82	1.06	5.59
r-200-0.5-B	141.00	5.50	1.63	789.67	1.63	754.11	1.63	69.83
r-300-0.5-B	211.70	6.40	1.60	4782.71 ⁴ (1.13%)	1.60	4198.07 ³ (0.47%)	1.15	127.31
Set C: Stock market graphs								
market-1	215	4	3.43	TE (3.70%)	3.43	TE (3.11%)	2.04	519.98
market-2	230	4	4.81	TE (2.60%)	4.81	TE (1.98%)	1.54	732.55
market-3	235	4	7.54	TE (1.22%)	7.54	TE (1.02%)	1.24	413.80
market-4	227	4	3.24	TE (3.41%)	3.24	TE (3.37%)	1.89	1077.45
market-5	229	4	2.70	TE (3.03%)	2.70	TE (3.43%)	1.58	378.34
Set D: Wind energy graphs								
wind-2004	146	3	1.04	TE (0.01%)	1.04	TE (0.01%)	1.02	718.13
wind-2005	145	3	1.01	TE (0.02%)	1.01	TE (0.03%)	1.003	5266.13
wind-2006	139	3	1.04	TE (0.01%)	1.04	TE (0.02%)	1.01	6235.69
Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$								
brock200_1	172	12	20.02	2875.61	20.02	2366.17	2.26	223.18
brock200_2	141	6	8.38	631.433	8.38	745.64	2.18	14.76
brock200_3	156	11	10.33	984.33	10.33	1011.22	2.01	18.87
brock200_4	162	13	14.27	1823.10	14.27	1728.80	2.49	56.99
c-fat-200-1	55	3	2.34	318.88	2.34	393.15	1.00	2.96
c-fat-500-1	94	3	2.34	1401.24	2.34	1298.33	1.00	34.41
c-fat-500-2	135	3	2.34	1298.49	2.34	1317.10	1.00	45.72
karate	13	4	3.09	0.73	3.09	0.69	1.55	0.35
dolphins	18	3	4.13	8.05	4.13	7.89	2.78	24.03
polbooks	33	3	2.00	14.92	2.00	14.61	1.00	1.27
adjnoun	29	3	4.07	73.45	4.07	107.01	2.73	73.28
football	35	3	2.68	29.93	2.68	25.93	1.80	27.78
celegans	70	3	3.01	TE (0.20%)	3.01	OM	2.01	6726.81
email	104	–	OM	–	OM	–	OM	–

Note: Column “ f_{LP}^*/f_{MIP}^* ” presents the ratio of the optimal objective function value f_{LP}^* of the LP relaxation to the optimal objective function value f_{MIP}^* of (11). The solution times are in seconds. For random graphs, if the time limit was exceeded for at least one instance in the batch, the average MIP gap is presented. The number of such instances out of 10 is shown in the superscript after the average solution time. “OM” indicates out of memory error.

- Generally, the time required to solve the MRsPP increases with s . In case of MRsDCP there is no clear trend in this regard.
- While none of the three methods clearly dominates another, the MILP approach shows better overall performance than the parametric methods for the maximum ratio s -plex problem. In particular, it outperforms the other two methods in terms of the total time for Sets A–D for all considered s values, with the exception of Set C for $s = 3$. However, the MILP approach is outperformed by both parametric methods on Set E.

TABLE 7 Running times (in seconds) when solving the maximum ratio s -plex problem; average times are presented for random graphs (Sets A and B)

Graph	MILP time			Binary search time (iterations)			Newton's method time (iterations)		
	$s = 1$	$s = 2$	$s = 3$	$s = 1$	$s = 2$	$s = 3$	$s = 1$	$s = 2$	$s = 3$
<i>Set A: Uniform random graphs with uncorrelated weights</i>									
r-100-0.1-A	0.22	0.28	0.51	0.21 (2.9)	0.46 (3.0)	0.54 (2.6)	0.17 (3.6)	0.34 (2.7)	0.38 (3.3)
r-200-0.1-A	0.65	1.21	2.19	1.03 (2.7)	2.62 (2.4)	3.03 (2.7)	0.90 (3.6)	2.38 (3.3)	2.58 (3.0)
r-300-0.1-A	1.77	2.95	4.52	3.98 (3.3)	7.55 (2.8)	18.47 (2.7)	2.94 (3.4)	7.08 (3.0)	14.70 (3.4)
r-100-0.5-A	0.29	1.34	3.79	0.32 (2.8)	1.16 (2.5)	2.51 (2.4)	0.27 (3.3)	0.72 (2.8)	1.42 (2.7)
r-200-0.5-A	1.47	10.01	43.99	6.73 (3.0)	16.62 (3.2)	50.51 (2.9)	2.33 (3.6)	12.85 (3.4)	27.80 (3.2)
r-300-0.5-A	5.48	44.46	273.75	22.34 (3.1)	146.68 (3.1)	686.70 (2.9)	8.72 (3.5)	129.96 (3.3)	559.34 (3.5)
Set A total	98.80	602.50	3287.50	346.10	1750.90	7617.60	153.30	1533.30	6062.20
<i>Set B: Uniform random graphs with constrained vertex ratio</i>									
r-100-0.1-B	0.18	0.34	0.56	0.18 (2.4)	0.45 (2.6)	0.56 (2.7)	0.18 (3.2)	0.41 (3.4)	0.49 (3.8)
r-200-0.1-B	0.55	1.62	2.26	1.12 (2.5)	2.61 (2.7)	3.75 (2.6)	1.18 (3.9)	2.29 (3.5)	3.14 (3.8)
r-300-0.1-B	1.57	3.36	4.91	3.21 (2.6)	9.91 (3.3)	15.52 (2.8)	2.97 (3.6)	8.23 (3.7)	13.89 (4.0)
r-100-0.5-B	0.24	1.19	3.38	0.31 (2.8)	1.15 (2.6)	2.75 (3.0)	0.27 (3.1)	0.78 (3.3)	2.13 (3.7)
r-200-0.5-B	2.08	8.62	76.26	8.07 (3.3)	17.16 (3.4)	87.41 (3.0)	2.43 (3.6)	127.30 (3.6)	42.98 (3.5)
r-300-0.5-B	5.89	43.11	503.82	16.51 (3.2)	132.05 (2.8)	730.34 (3.3)	7.46 (3.7)	125.57 (3.7)	610.61 (3.9)
Set B total	105.10	582.40	5911.90	294.00	1633.30	8403.30	144.90	2645.80	6732.40
<i>Set C: Stock market graphs</i>									
market-1	21.88	68.18	2021.02	71.32 (4)	153.88 (5)	161.73 (5)	68.01 (4)	112.85 (4)	140.73 (4)
market-2	10.12	33.77	128.81	34.71 (3)	52.10 (3)	241.27 (3)	31.10 (4)	27.18 (3)	48.84 (3)
market-3	9.96	29.52	364.90	45.27 (4)	114.95 (4)	387.53 (5)	42.43 (4)	46.82 (3)	115.80 (4)
market-4	11.02	34.56	174.92	44.25 (4)	106.92 (4)	241.04 (4)	37.40 (3)	43.43 (4)	67.53 (4)
market-5	10.50	51.55	168.99	45.87 (4)	71.88 (4)	190.94 (5)	31.55 (3)	30.77 (3)	72.23 (4)
Set C total	63.48	217.58	2858.64	241.42	499.73	1222.51	210.49	261.05	445.13
<i>Set D: Wind energy graphs</i>									
wind-2004	4.97	18.61	19.98	14.55 (2)	82.12 (2)	49.91 (2)	13.43 (3)	67.95 (2)	46.32 (3)
wind-2005	5.24	8.19	28.96	21.41 (3)	34.54 (2)	43.52 (3)	10.53 (3)	28.95 (3)	24.19 (2)
wind-2006	4.55	14.79	22.88	41.15 (2)	40.28 (3)	14.65 (2)	64.97 (3)	37.19 (3)	14.50 (2)
Set D total	14.76	41.59	71.82	77.11	156.94	108.08	88.93	134.09	85.01
<i>Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$</i>									
brock200_1	13.59	238.12	4561.07	3.41 (3)	127.66 (3)	2533.49 (3)	1.72 (3)	49.10 (3)	2467.35 (3)
brock200_2	1.03	4.21	30.47	2.95 (2)	9.39 (3)	38.76 (2)	2.49 (3)	7.73 (3)	23.09 (2)
brock200_3	2.02	14.72	101.09	1.72 (2)	43.40 (2)	322.33 (2)	0.98 (3)	41.65 (3)	679.92 (3)
brock200_4	4.44	34.77	738.38	4.10 (3)	16.48 (2)	340.94 (3)	3.15 (4)	9.53 (2)	308.8 (3)
c-fat-200-1	0.77	0.94	1.08	0.25 (1)	0.43 (1)	0.33 (1)	0.16 (2)	0.24 (2)	0.31 (2)
c-fat-500-1	7.97	10.07	9.62	4.47 (1)	4.74 (1)	4.62 (1)	2.66 (2)	3.28 (2)	3.24 (2)
c-fat-500-2	4.75	5.32	4.85	3.13 (2)	3.81 (2)	3.68 (2)	1.68 (2)	2.16 (2)	2.16 (2)
karate	0.02	0.01	0.10	0.01 (1)	0.01 (1)	0.07 (1)	0.01 (2)	0.01 (2)	0.04 (2)
dolphins	0.08	0.19	0.50	0.25 (2)	0.18 (2)	0.17 (1)	0.05 (3)	0.14 (2)	0.19 (2)
polbooks	0.25	0.44	1.58	0.08 (1)	1.05 (5)	1.21 (2)	0.07 (2)	0.65 (3)	1.57 (4)
adjnoun	0.23	0.46	0.64	0.38 (2)	0.59 (2)	0.36 (1)	0.26 (2)	0.97 (3)	0.47 (2)
football	0.28	0.40	1.98	0.69 (2)	1.23 (2)	0.31 (1)	0.57 (3)	1.31 (3)	0.11 (1)
celegans	17.20	18.63	34.75	60.81 (4)	28.15 (3)	48.25 (3)	37.05 (3)	13.41 (2)	25.58 (3)
email	37.85	195.33	177.06	32.83 (1)	257.04 (3)	796.87 (2)	11.59 (1)	194.58 (3)	614.69 (4)
Set E total	90.48	523.61	5663.17	115.08	494.16	4091.39	62.44	324.76	4127.52
Total	372.62	1967.68	17,793.03	1073.71	4535.03	21,442.88	660.06	4899.00	17,452.26

Note: The lines "Set X total," $X \in \{A, B, C, D, E\}$ show the total time taken to solve all instances from Set X. The last line displays the total time taken to solve all the instances in the test bed. The best time among the three methods for each s is shown in bold in every line. For the parametric methods we also display the number of iterations in parentheses.

TABLE 8 Running times (in seconds) for the maximum ratio s -defective clique problem with $s = 0, 1, 2$; average times are presented for Sets A and B

Graph	MILP time			Binary search time (iterations)			Newton's method time (iterations)		
	$s = 0$	$s = 1$	$s = 2$	$s = 0$	$s = 1$	$s = 2$	$s = 0$	$s = 1$	$s = 2$
<i>Set A: Uniform random graphs with uncorrelated weights</i>									
r-100-0.1-A	0.46	1.44	33.51	0.70 (2.7)	21.82 (2.2)	17.32 (2.3)	0.82 (3.4)	22.04 (4.6)	16.18 (3.5)
r-200-0.1-A	2.70	21.33	161.58	8.23 (2.7)	103.13 (2.4)	159.54 (2.6)	6.07 (3.3)	97.43 (3.4)	155.09 (3.8)
r-300-0.1-A	8.49	684.14	1521.61	29.23 (2.8)	1947.54 (2.4)	1418.55 (2.3)	21.78 (3.4)	2758.45 (3.9)	1524.50 (4.1)
r-100-0.5-A	0.60	2.14	2.42	0.70 (2.7)	12.71 (3.0)	9.25 (3.0)	0.56 (3.4)	9.48 (3.3)	8.81 (3.5)
r-200-0.5-A	3.06	29.07	91.92	6.39 (3.0)	54.44 (3.0)	134.67 (3.2)	4.95 (3.6)	48.98 (4.0)	135.12 (4.6)
r-300-0.5-A	10.21	408.52	172.65	22.72 (3.0)	4273.63 (2.3)	3524.19 (3.4)	18.09 (3.7)	3604.75 (4.0)	488.23 (4.0)
Set A total	255.20	11,466.40	19,836.90	679.70	64,132.70	52,635.20	522.70	65,411.30	23,279.30
<i>Set B: Uniform random graphs with constrained vertex ratio</i>									
r-100-0.1-B	0.42	0.44	20.29	1.05 (2.6)	25.01 (3.0)	17.65 (2.5)	0.79 (3.2)	23.94 (3.5)	7.88 (3.7)
r-200-0.1-B	2.63	33.52	143.02	7.68 (2.4)	103.64 (2.4)	144.91 (2.2)	6.63 (3.6)	120.43 (3.9)	140.27 (3.9)
r-300-0.1-B	7.82	1301.06	1925.16	26.93 (2.6)	3742.33 (2.7)	1236.73 (3.4)	21.55 (3.4)	2754.19 (3.7)	721.70 (4.3)
r-100-0.5-B	0.51	3.14	6.02	0.64 (2.6)	14.42 (2.8)	6.66 (3.1)	0.53 (3.3)	12.54 (3.6)	6.18 (3.7)
r-200-0.5-B	3.32	517.29	68.93	7.73 (3.2)	64.61 (3.4)	72.23 (3.3)	5.68 (3.7)	61.93 (4.2)	73.54 (4.4)
r-300-0.5-B	9.32	1288.38	189.21	25.37 (3.3)	6358.67 (3.3)	617.38 (3.2)	20.89 (3.8)	5673.35 (4.3)	790.45 (4.1)
Set B total	240.20	31,438.30	23,526.30	694.00	103,086.80	20,955.60	560.70	86,465.80	17,400.20
<i>Set C: Stock market graphs</i>									
market-1	65.46	469.98	417.35	163.70 (4)	5179.29 (4)	3015.11 (4)	74.56 (3)	TE	2877.88 (4)
market-2	44.67	438.85	460.28	119.53 (3)	5863.50 (4)	4410.55 (5)	64.54 (3)	TE	4758.18 (5)
market-3	41.12	544.31	343.64	167.72 (4)	4822.32 (5)	4970.85 (4)	106.53 (4)	TE	5651.41 (4)
market-4	44.48	563.31	795.64	146.05 (4)	5891.39 (4)	TE	69.24 (3)	4483.71 (3)	TE
market-5	39.61	307.42	352.15	172.66 (4)	7050.09 (4)	3697.09 (2)	72.27 (3)	6581.42 (4)	3901.91 (4)
Set C total	235.34	2323.66	2369.06	769.66	28,806.59	23,293.60	387.14	32,665.13	24,389.38
<i>Set D: Wind energy graphs</i>									
wind-2004	25.59	880.27	456.68	65.61 (1)	1073.63 (3)	3215.50 (2)	45.78 (2)	TE	2845.33 (3)
wind-2005	28.16	35.98	4459.78	103.83 (3)	656.55 (2)	3231.39 (1)	62.70 (3)	571.26 (3)	3378.83 (2)
wind-2006	32.45	36.70	3890.83	92.41 (2)	TE	TE	75.02 (3)	TE	TE
Set D total	86.20	952.95	8807.29	261.85	8930.18	13,646.89	183.50	14,971.26	13,424.16
<i>Set E: DIMACS instances with $a_i = i$, $b_i = V - i + 1$</i>									
brock200-1	8.40	27.06	208.19	5.29 (3)	97.80 (2)	89.98 (3)	2.12 (3)	26.94 (3)	61.74 (4)
brock200-2	3.90	15.54	15.53	4.68 (2)	111.06 (3)	86.53 (2)	2.70 (3)	106.88 (3)	81.71 (4)
brock200-3	2.95	12.31	16.68	4.37 (2)	111.32 (2)	259.46 (2)	3.24 (3)	123.41 (4)	289.53 (4)
brock200-4	6.54	16.01	46.46	5.65 (3)	141.20 (2)	63.38 (4)	4.45 (4)	135.11 (4)	43.81 (3)
c-fat-200-1	7.26	11.48	2.90	8.37 (1)	2596.51 (2)	268.56 (2)	4.60 (2)	4276.8 (3)	347.10 (3)
c-fat-500-1	138.91	187.47	39.57	175.46 (1)	TE	TE	104.68 (2)	TE	TE
karate	0.02	0.29	0.30	0.03 (1)	0.24 (1)	0.23 (1)	0.02 (2)	0.17 (2)	0.20 (2)
dolphins	0.10	0.11	25.29	0.16 (1)	3.60 (2)	1.00 (1)	0.11 (2)	3.53 (3)	0.71 (2)
polbooks	0.53	0.46	0.81	1.01 (2)	31.60 (2)	19.24 (2)	0.74 (3)	33.06 (4)	17.69 (3)
adjnoun	0.44	0.59	45.27	0.89 (1)	14.53 (2)	29.94 (2)	0.64 (2)	14.05 (3)	24.81 (3)
football	0.72	0.60	9.97	1.57 (2)	14.52 (2)	29.09 (2)	2.00 (4)	15.27 (2)	24.61 (3)
celegans	59.40	29.93	5845.15	131.53 (4)	3786.09 (1)	6008.42 (2)	72.77 (3)	3991.32 (3)	5616.42 (4)
email	OM	OM	OM	OM	OM	OM	OM	OM	OM
Set E total	229.17	301.85	6256.12	339.01	14,108.47	14,055.83	198.07	15,926.54	13,708.33
Total	1046.11	46,483.16	60,795.67	2744.22	219,064.74	124,587.12	1852.11	215,440.03	92,201.37

Note: The lines “Set X total,” $X \in \{A, B, C, D, E\}$ show the total time taken to solve all instances from Set X . The last line displays the total time taken to solve all instances. The best time among the three methods for each s is shown in bold in every line. For the parametric methods we also display the number of iterations in parentheses.

TABLE 9 Performance comparison of the MILP formulations for the maximum average 2-plex problem

Graph	MILP-G		MILP-A1		MILP-A2		Binary search	Newton's method
	f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time	Time	Time
<i>Random graphs</i>								
r-100-0.1-A	1.05	0.25	13.79	3.90	1.07	0.24	0.38 (2.1)	0.29 (2.8)
r-200-0.1-A	1.05	1.24	26.30	41.51	1.06	0.86	1.58 (1.5)	1.52 (2.2)
r-300-0.1-A	1.03	2.99	38.27	227.02	1.04	3.64	15.19 (1.9)	10.25 (2.9)
r-100-0.5-A	1.12	1.47	21.07	36.96	1.17	1.37	1.26 (2.3)	0.89 (3.0)
r-200-0.5-A	1.09	10.43	39.90	6036.45 ³ (0.24%)	1.14	7.42	15.56 (2.2)	12.84 (2.8)
r-300-0.5-A	1.08	56.41	56.65	TE (2.34%)	1.09	26.58	73.65 (2.4)	64.97 (3.0)
r-100-0.1-B	1.09	0.32	14.04	3.64	1.08	0.22	0.41 (2.0)	0.28 (2.6)
r-200-0.1-B	1.06	1.46	27.02	48.14	1.06	1.07	2.28 (2.1)	1.90 (2.5)
r-300-0.1-B	1.03	3.91	37.79	209.89	1.04	3.42	16.25 (1.9)	12.15 (2.7)
r-100-0.5-B	1.13	1.39	21.08	35.71	1.17	0.99	1.26 (2.2)	0.82 (2.6)
r-200-0.5-B	1.09	12.19	38.90	6147.57 ⁴ (0.37%)	1.15	6.79	18.73 (2.2)	14.93 (2.9)
r-300-0.5-B	1.07	51.81	56.78	TE (2.47 %)	1.09	24.77	128.13 (2.5)	135.75 (3.0)
<i>DIMACS graphs</i>								
brock200_1	1.13	213.54	44.88	TE (0.52%)	1.89	74.99	125.80 (3)	44.76 (3)
brock200_2	1.07	6.80	38.18	2601.08	1.09	7.63	10.24 (2)	7.67 (3)
brock200_3	1.05	15.08	39.50	TE (1.00%)	1.08	5.33	52.10 (2)	35.60 (3)
brock200_4	1.09	32.84	41.32	TE (1.14%)	1.13	14.89	18.51 (2)	9.34 (2)
c-fat-200-1	1.64	0.88	14.65	3.50	1.75	0.18	0.33 (1)	0.30 (2)
c-fat-500-1	1.69	7.10	24.53	10.39	1.77	1.91	4.60 (1)	3.17 (2)
c-fat-500-2	1.80	4.94	36.25	8.25	1.89	1.14	4.54 (3)	2.03 (2)
karate	1.00	0.01	179.08	0.28	1.05	0.03	0.01 (1)	0.01 (2)
dolphins	1.23	0.15	9.64	0.80	1.25	0.13	0.19 (2)	0.07 (2)
polbooks	1.11	0.27	15.23	5.01	1.14	0.32	0.97 (4)	0.53 (3)
adjnoun	1.10	0.34	14.26	3.54	1.12	0.33	0.73 (2)	0.61 (3)
football	1.11	0.33	16.20	5.70	1.12	0.35	0.86 (2)	1.19 (3)
celegans	1.17	36.33	38.59	212.85	1.18	20.33	32.00 (3)	16.21 (2)
email	1.07	223.82	36.13	1634.29	1.07	117.52	241.33 (3)	216.20 (3)

Note: Column “MILP-G” reports the results for MILP (10) with valid inequalities; column “MILP-A1” reports the results for MILP (21) with valid inequalities; column “MILP-A2” reports the results for MILP (25) without valid inequalities. All running times are in seconds. The best time among the five methods is shown in bold in every line. For the parametric methods we also display the number of iterations in parentheses.

- The superiority of the MILP approach is more pronounced for the maximum ratio s -defective clique problem. In this case, it is considerably faster than the parametric methods on most instances. Recall that the parametric methods require solving MILP (26) at each iteration, which in case of s -defective clique has a binary variable and a constraint for each pair of non-adjacent vertices.

5.4 | Exploring the performance of general and customized methods for the average weight case

In this subsection, we present and compare the results for the maximum average 2-plex and 2-defective cliques problems solved using the general purpose methods analyzed in the previous subsection, as well as the two customized approaches developed in Section 3.4. More specifically, we consider three different MILP models and two parametric methods.

The first approach uses the general-purpose formulations (10) and (11) with the valid inequalities (14a) and (15) as well as (14b) and (18), respectively. Those valid inequalities are effective for decreasing the solver's running time; recall our discussion in Section 5.2. The second approach is based on formulation (21), which is tailored for the maximum average weight s -plex (and similarly derived MILP for s -defective clique) with the valid inequalities (14a) and (15) (and with (14b) and (18) for s -defective clique). In this case, those valid inequalities also help decreasing running times, as was concluded based on our preliminary experiments (omitted for brevity). The third MILP is (25), which is also custom-tailored for the average weight case. Unlike in the previous models, no valid inequalities are used, as they were not observed to improve running times in our preliminary experiments (not reported here as well). Finally, the two parametric methods are the general-purpose binary search and Newton's methods discussed above.

TABLE 10 Performance comparison of the MILP formulations for the maximum average 2-defective clique

Graph	MILP-G		MILP-A1		MILP-A2		Binary search	Newton's method
	f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time	f_{LP}^*/f_{MIP}^*	Time	Time	Time
<i>Random graphs</i>								
r-100-0.1-A	1.02	16.19	14.38	271.49	1.03	13.34	17.39 (1.8)	22.18 (3.4)
r-200-0.1-A	1.01	33.36	27.52	TE (0.38%)	1.04	175.91	248.57 (2.2)	132.43 (3.3)
r-300-0.1-A	1.00	43.34	40.43	TE (3.47%)	1.01	418.80	1321.51 (3.0)	1696.56 (4.0)
r-100-0.5-A	1.03	1.77	24.43	648.69	1.04	3.71	8.57 (2.4)	6.48 (3.7)
r-200-0.5-A	1.03	15.66	47.87	TE (2.11%)	1.04	128.15	100.23 (2.5)	89.04 (3.4)
r-300-0.5-A	1.03	86.37	71.52	TE (3.47%)	1.03	232.69	439.48 (2.9)	946.90 (3.7)
r-100-0.1-B	1.01	4.27	14.65	272.72	1.04	12.23	19.69 (1.9)	17.74 (3.0)
r-200-0.1-B	1.02	17.13	27.65	TE (0.41%)	1.02	113.64	152.93 (2.0)	137.35 (3.2)
r-300-0.1-B	1.01	94.16	40.61	TE (3.23%)	1.03	693.33	1644.62 (2.0)	1547.96 (3.6)
r-100-0.5-B	1.02	1.96	24.52	838.64	1.05	3.45	7.17 (2.4)	7.81 (3.3)
r-200-0.5-B	1.03	16.09	47.63	TE (1.92%)	1.04	120.27	149.04 (3.1)	132.06 (3.6)
r-300-0.5-B	1.02	72.66	71.57	TE (3.21%)	1.03	226.84	768.40 (3.0)	715.68 (3.8)
<i>DIMACS graphs</i>								
brock200_1	1.05	213.54	54.18	TE (1.86%)	1.10	92.48	146.02 (3)	124.98 (4)
brock200_2	1.02	14.91	47.68	TE (1.98%)	1.04	26.09	79.20 (2)	73.15 (4)
brock200_3	1.03	14.92	50.15	TE (2.16%)	1.05	87.33	303.21 (3)	295.58 (4)
brock200_4	1.04	41.66	51.76	TE (1.85%)	1.07	110.13	65.02 (4)	44.63 (3)
c-fat-200-1	1.00	2.99	24.70	TE (2.90%)	1.01	10.69	329.96 (2)	408.38 (3)
c-fat-500-1	1.00	37.90	43.57	TE (7.87%)	1.004	385.46	TE	TE
c-fat-500-2	1.00	38.59	59.96	TE (9.36%)	1.003	320.39	TE	TE
karate	1.03	0.31	6.02	2.19	1.05	0.33	0.22 (1)	0.17 (2)
dolphins	1.04	5.34	8.53	54.23	1.05	4.47	0.99 (1)	0.72 (2)
polbooks	1.00	1.10	14.47	297.20	1.01	11.96	21.56 (2)	17.40 (3)
adjnoun	1.02	31.04	13.79	481.91	1.03	63.52	29.88 (2)	24.06 (3)
football	1.01	3.11	2.88	731.51	1.01	26.63	29.25 (2)	26.47 (3)
celegans	1.003	2190.53	33.22	TE (6.88%)	1.004	704.49	TE	TE
email	–	OM	–	OM	–	OM	OM	OM

Note: Column “MILP-G” reports the results for MILP (11) with valid inequalities; column “MILP-A1” reports the results for MILP analogous to (21) (modified for 2-defective clique) with valid inequalities; column “MILP-A2” reports the results for MILP analogous to (25) (modified for 2-defective clique) without valid inequalities. All running times are in seconds. The best time among the five methods is shown in bold in every line. For the parametric methods we also display the number of iterations in parentheses.

For this set of experiments, we use DIMACS instances and random instances with $a_i = i$ and $b_i = 1$ for all $i \in V$. The Set C and D instances (market and wind graphs, respectively) are omitted since for these instances the node weights had a specific real-life meaning, which would be distorted by replacing them with weights suitable for the average weight case. The results are presented in Table 9 (for MA2PP) and Table 10 (for MA2DCP).

The results for the maximum average 2-plex problem show that MILP (25) (denoted as MILP-A2 in the table) performs better than the other considered methods both on random and DIMACS graphs. For some instances, the general-purpose MILP (10) (denoted as MILP-G in the table) or Newton’s method performed better than (25). The performance of (21) (denoted as MILP-A1 in the table) is clearly inferior to that of the other considered methods due to the weak quality of its LP relaxation.

For the maximum average 2-defective clique problem, the general-purpose MILP approach (11) is a clear winner, with only a few instances solved faster by either MILP (25) or Newton’s method. However, it should be pointed out that for two of the instances that MILP (25) solves fastest (brock200_1 and celegans), it beats the general-purpose MILP by a large margin. Once again, similarly to the case of 2-plex, the approach based on (21) is clearly inferior to all the other considered methods due to the weak quality of its LP relaxation.

6 | CONCLUDING REMARKS

This article studies graph-theoretic relaxations of the maximal ratio clique problem, specifically the maximum ratio s -plex problem (MRsPP) and the maximum ratio s -defective clique problem (MRsDCP). A general NP -hardness result, which can be

applied to these problems, was established. We discussed and implemented three methods to solve MRsPP and MRsDCP: an MILP approach, as well as two parametric approaches, the binary search and Newton's method. We compared the performance of the proposed approaches through extensive numerical experiments with randomly generated and real-life network instances.

As a result, we concluded that (i) all three methods are capable of solving most considered instances to optimality (or near-optimality) within a reasonable time; (ii) the MILP approach slightly outperformed the parametric methods for the MRsPP; (iii) the MILP approach performed significantly better than the parametric methods for the MRsDCP.

We also discussed two methods tailored for a special case of the considered problems, the maximum average s -plex (s -defective clique) problem. We concluded that one of these methods is competitive with the general-purpose methods. Since all of the considered approaches involve modeling the set of all s -plexes (s -defective cliques) in an MILP framework, including stronger valid inequalities for the s -plex (s -defective clique) polytope could further enhance the performance of the considered methods. Finally, a combinatorial approach based on implicit enumeration of all maximal s -plexes (s -defective cliques) in a graph could be an interesting alternative to explore for future research, in particular, for sparse networks.

ACKNOWLEDGMENTS

The work of Sergiy Butenko was partially supported by the AFOSR award FA9550-19-1-0161 and CRDF Global award G-202102-67302. The work of Oleg A. Prokopyev was partially supported by the NSF award DMS 1818700. The authors would like to thank the associate editor and two anonymous referees for their extremely helpful and constructive comments that allowed us to improve this article.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

ORCID

Sergiy Butenko  <https://orcid.org/0000-0002-6662-9552>

Petar Momcilovic  <https://orcid.org/0000-0003-2147-3985>

Oleg A. Prokopyev  <https://orcid.org/0000-0003-2888-8630>

REFERENCES

- [1] J. Abello, M. G. C. Resende, and S. Sudarsky, "Massive quasi-clique detection," *LATIN 2002: Theoretical informatics*, S. Rajsbaum (ed.), Springer-Verlag, London, UK, 2002, pp. 598–612.
- [2] W. P. Adams and R. J. Forrester, *Linear forms of nonlinear expressions: New insights on old ideas*, Oper. Res. Lett. **35** (2007), no. 4, 510–518.
- [3] R. D. Alba, *A graph-theoretic definition of a sociometric clique*, J. Math. Sociol. **3** (1973), 113–136.
- [4] B. Balasundaram, S. Butenko, and I. V. Hicks, *Clique relaxations in social network analysis: The maximum k -plex problem*, Oper. Res. **59** (2011), no. 1, 133–142.
- [5] D. Bertsimas, N. Korolko, and A. M. Weinstein, *Identifying exceptional responders in randomized trials: An optimization approach*, INFORMS J. Optim. **1** (2019), no. 3, 187–199.
- [6] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo, "The maximum clique problem," *Handbook of combinatorial optimization*, D.-Z. Du and P. M. Pardalos (eds.), Kluwer Academic Publishers, Dordrecht, Netherlands, 1999, pp. 1–74.
- [7] J. S. Borrero, C. Gillen, and O. A. Prokopyev, *A simple technique to improve linearized reformulations of fractional (hyperbolic) 0–1 programming problems*, Oper. Res. Lett. **44** (2016), no. 4, 479–486.
- [8] J. S. Borrero, C. Gillen, and O. A. Prokopyev, *Fractional 0–1 programming: Applications and algorithms*, J. Glob. Optim. **69** (2017), 255–282.
- [9] S. Butenko and W. E. Wilhelm, *Clique-detection models in computational biochemistry and genomics*, Eur. J. Oper. Res. **173** (2006), no. 1, 1–17.
- [10] V. G. Deineko, B. Klinz, and G. J. Woeginger, *Uniqueness in quadratic and hyperbolic 0–1 programming problems*, Oper. Res. Lett. **41** (2013), no. 6, 633–635.
- [11] DIMACS. Cliques, coloring, and satisfiability: Second dimacs implementation challenge. <http://dimacs.rutgers.edu/Challenges/>, 1995.
- [12] DIMACS. 10th DIMACS implementation challenge. <https://www.cc.gatech.edu/dimacs10/index.shtml>, 2011.
- [13] W. Dinkelbach, *On nonlinear fractional programming*, Manag. Sci. **13** (1967), no. 7, 492–498.
- [14] P. Erdős and A. Rényi, *On random graphs I*, Publ. Math. Debr. **6** (1959), 290–297.
- [15] H. K. Fung, S. Rao, C. A. Floudas, O. A. Prokopyev, P. M. Pardalos, and F. Rendl, *Computational comparison studies of quadratic assignment like formulations for the in silico sequence selection problem in de novo protein design*, J. Comb. Optim. **10** (2005), no. 1, 41–60.
- [16] M. R. Garey and D. S. Johnson, *Computers and intractability: A guide to the theory of NP-completeness*, W. H. Freeman and Company, New York, NY, 1979.
- [17] D. Goeke, M. Moeini, and D. Poganiuch, *A variable neighborhood search heuristic for the maximum ratio clique problem*, Comput. Oper. Res. **87** (2017), 283–291.
- [18] Gurobi Optimization. Gurobi optimizer reference manual. <http://www.gurobi.com>, 2019, Accessed July 2021.
- [19] A. Hagberg, P. Swart, and D. S. Chult, *Exploring network structure, dynamics, and function using NetworkX*, Technical report, Los Alamos National Lab.(LANL), Los Alamos, NM, 2008.
- [20] T. Ibaraki, *Parametric approaches to fractional programs*, Math. Program. **26** (1983), 345–362.

- [21] M. Locatelli and F. Schoen, *Global optimization: Theory, algorithms, and applications*, Society for Industrial and Applied Mathematics, 3600 University City Science Center, Philadelphia, PA, 2013.
- [22] R. D. Luce, *Connectivity and generalized cliques in sociometric group structure*, *Psychometrika* **15** (1950), 169–190.
- [23] R. D. Luce and A. D. Perry, *A method of matrix analysis of group structure*, *Psychometrika* **14** (1949), 95–116.
- [24] M. Mahdavi Pajouh and B. Balasundaram, *On inclusionwise maximal and maximum cardinality k -clubs in graphs*, *Discret. Optim.* **9** (2012), 84–97.
- [25] M. Moeini, “*The maximum ratio clique problem: A continuous optimization approach and some new results*,” *Modelling, computation and optimization in information systems and management sciences*, *Advances in Intelligent Systems and Computing*, Vol **359**, H. Le Thi, T. Pham Dinh, and N. Nguyen (eds.), Springer, Cham, 2015.
- [26] R. J. Mokken, *Cliques, clubs and clans*, *Qual. Quant.* **13** (1979), 161–173.
- [27] M. Newman, *Networks*, Oxford University Press, Oxford, UK, 2018.
- [28] J. B. Orlin and R. K. Ahuja, *New scaling algorithms for the assignment and minimum mean cycle problems*, *Math. Program.* **54** (1992), 41–56.
- [29] P. M. Pardalos and J. Xue, *The maximum clique problem*, *J. Glob. Optim.* **4** (1994), 301–328.
- [30] J. Pattillo, A. Veremyev, S. Butenko, and V. Boginski, *On the maximum quasi-clique problem*, *Discret. Appl. Math.* **161** (2013), no. 1–2, 244–257.
- [31] J. Pattillo, N. Youssef, and S. Butenko, *On clique relaxation models in network analysis*, *Eur. J. Oper. Res.* **226** (2013), 9–18.
- [32] O. A. Prokopyev, H. Huang, and P. M. Pardalos, *On complexity of unconstrained hyperbolic 0-1 programming problems*, *Oper. Res. Lett.* **33** (2005), no. 3, 312–318.
- [33] T. Radzik, “*Fractional combinatorial optimization*,” *Handbook of combinatorial optimization*, Vol **1**, D.-Z. Du and P. M. Pardalos (eds.), Dordrecht, The Netherlands: Kluwer Academic Publishers, 1998, pp. 429–478.
- [34] R. S. Sankaranarayanan and L. K. Stewart, *Complexity results for well-coscovered graphs*, *Networks* **22** (1992), no. 3, 247–262.
- [35] S. B. Seidman, *Network structure and minimum degree*, *Soc. Netw.* **5** (1983), 269–287.
- [36] S. B. Seidman and B. L. Foster, *A graph theoretic generalization of the clique concept*, *J. Math. Sociol.* **6** (1978), 139–154.
- [37] S. Sethuraman and S. Butenko, *The maximum ratio clique problem*, *Comput. Manag. Sci.* **12** (2013), 197–218.
- [38] H. D. Sherali and W. P. Adams, *A reformulation-linearization technique for solving discrete and continuous nonconvex problems*, Vol **31**, Springer Science & Business Media, Boston, MA, 2013.
- [39] H. D. Sherali and J. C. Smith, *A polyhedral study of the generalized vertex packing problem*, *Math. Program.* **107** (2006), no. 3, 367–390.
- [40] I. M. Stancu-Minasian, *A eighth bibliography of fractional programming*, *Optimization* **66** (2017), 439–470.
- [41] M. Tawarmalani, S. Ahmed, and N. V. Sahinidis, *Global optimization of 0-1 hyperbolic programs*, *J. Glob. Optim.* **24** (2002), 385–416.
- [42] S. Trukhanov, C. Balasubramaniam, B. Balasundaram, and S. Butenko, *Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations*, *Comput. Optim. Appl.* **56** (2013), 113–130.
- [43] A. Veremyev, V. Boginski, E. L. Pasiliao, and O. A. Prokopyev, *On integer programming models for the maximum 2-club problem and its robust generalizations in sparse graphs*, *Eur. J. Oper. Res.* **297** (2022), no. 1, 86–101.
- [44] C. Vogiatzis and J. L. Walteros, *Integer programming models for detecting graph bipartitions with structural requirements*, *Networks* **71** (2018), no. 4, 432–450.
- [45] L. A. Wolsey and G. L. Nemhauser, *Integer and combinatorial optimization*, Vol **55**, John Wiley & Sons, Hoboken, NJ, 1999.
- [46] Q. Wu and J. Hao, *A review on algorithms for maximum clique problems*, *Eur. J. Oper. Res.* **242** (2015), no. 3, 693–709.
- [47] T. Wu, *A note on a global approach for general 0-1 fractional programming*, *Eur. J. Oper. Res.* **101** (1997), no. 1, 220–223.
- [48] M. Yannakakis, *Node-and edge-deletion NP-complete problems*, *Proc. 10th Annu. ACM Sympos. Theory Comput. STOC '78*, ACM Press, New York, NY, 1978, pp. 253–264.
- [49] H. Yu, A. Paccanaro, V. Trifonov, and M. Gerstein, *Predicting interactions in protein networks by completing defective cliques*, *Bioinformatics* **22** (2006), 823–829.
- [50] D. Zuckerman, *Linear degree extractors and the inapproximability of max clique and chromatic number*, *Proc. 38th Annu. ACM Sympos. Theory Comput. ACM Press, New York, NY; 2006*, pp. 681–690.

How to cite this article: Y. Blokhin, S. Butenko, P. Momeilovic, and O. A. Prokopyev, *On maximum ratio clique relaxations*, *Networks*. (2022), 1–26. <https://doi.org/10.1002/net.22097>

Asymptotic Bounds for Clustering Problems in Random Graphs

Eugene Lykhovyd¹ | Sergiy Butenko¹ | Pavlo Krokhamal²

¹Wm. Michael Barnes '64 Department of Industrial and Systems Engineering, Texas A&M University, College Station, TX 77843-3131, USA

²Department of Systems and Industrial Engineering, University of Arizona, Tucson, AZ 85721-0020, USA

Correspondence

Sergiy Butenko, TAMU-3131, College Station, TX 77843-3131, USA
Email: butenko@tamu.edu

Funding information

Air Force Office of Scientific Research, Grant Number: FA9550-19-1-0161

Graph clustering is an important problem in network analysis. This problem can be approached by first finding a large cluster subgraph (i.e., a subgraph in which every connected component is a complete graph), perhaps in a relaxed form (connected components may have missing edges), and then assigning each of the remaining vertices to one of the connected components of the cluster subgraph according to some optimization criteria. The more vertices can be included in the initial cluster subgraph (also referred to as independent union of clusters), the more “clusterable” the graph is. This paper proposes a framework for establishing asymptotic bounds on the cardinality of independent unions of clusters in Erdős-Rényi random graphs $G(n, p)$ with constant p , referred to as uniform random graphs. In particular, sufficient conditions ensuring $O(\log n)$ (where n is the number of nodes) upper bounds with probability 1 are developed and shown to be applicable for the maximum independent union of cliques as well as some clique relaxations. In addition, it is shown that every graph must have an independent union of cliques of cardinality at least $\Omega(\log n)$. Since this bound is asymptotically tight on uniform random graphs, this suggests that these graphs can be viewed as a “least clusterable” class of graphs.

KEYWORDS

network analysis, network clusterability, independent union of cliques, uniform random graphs, asymptotic bounds

1 | INTRODUCTION

Massive amounts of data generated by real-life complex systems can oftentimes be modeled as networks, with nodes (vertices) representing the system's components and arcs (edges) describing their pairwise relations inferred from the available data. Such a representation enables the powerful *network analysis* tools [13], which can be used to retrieve useful information concerning the underlying complex system by analyzing the structural properties of the network model. One of the key tasks in network analysis is *unsupervised clustering*, aiming to partition a network into (non-overlapping or overlapping) “communities”, or “tightly knit” groups of nodes. The main approaches used to complete this task can be roughly classified into two categories: (i) computing a clustering that optimizes a certain performance metric, such as modularity [24, 4, 5]; (ii) partitioning the nodes into a minimum number of clusters of a given type, e.g., cliques or clique relaxations [20, 27]. In both cases, the number of clusters obtained in an optimal partitioning constitutes one of the most important outcomes of the clustering process and can be thought of as an essential global structural characteristic of the network.

Information about the “natural” number of clusters in a network, when available, could be utilized in developing supervised clustering (partitioning) techniques, which assume that the number of clusters is given as an input. To estimate this number, one could apply *cluster editing* [14, 10] or *cluster vertex deletion* [21] techniques, which can be described as follows. Suppose the edges of the network are obtained from a given data set by inferring binary relations with some noise or errors. One way to “clean” this data is to edit (remove or add) the minimum amount of edges so that each connected component in the residual network forms a cluster of a given kind, e.g., a clique (cluster editing). In a more conservative approach, one could aim to obtain the desired connected components by removing the minimum number of vertices (cluster vertex deletion). In both cases, the connected components of the residual network can be thought of as “seed clusters” that could be further expanded by assigning each vertex from outside to one of these clusters.

This paper focuses on a maximization version of the latter problem, where the objective is to find a maximum-cardinality subset of vertices such that each connected component in the corresponding induced subgraph forms a cluster with a desired structural property Π . We will refer to such a problem as the *maximum independent union of Π -sets (IUP) problem*. A special case of this problem, the *maximum independent union of cliques (IUC) problem* has recently been studied in [17]. In particular, its $\mathcal{NP}$ -hardness was established for several graph classes, and a combinatorial branch-and-bound and IP-based approaches have been proposed for solving it.

In this paper, we investigate the maximum IUP problem on uniform random graphs, also called Erdős-Rényi (ER) graphs [11, 15]. In particular, we propose a method for establishing logarithmic asymptotic upper and lower bounds on the cardinality of a maximum IUP that hold with probability 1, or almost surely (a.s.). This method is used to obtain logarithmic bounds for the maximum IUC and the maximum independent union for several clique relaxations (s -defective cliques, s -plexes, and γ -quasicliques), defined later. On the one hand, this is somewhat surprising, given that both cliques and independent sets yield feasible solutions for the corresponding problems. On the other hand, this result is supported by the observation that uniform random graphs are “un-clusterable” by design, so one would not expect to cover a large portion of such graphs with an independent union of tightly-knit clusters. In fact, this observation is mathematically formalized in this paper by proving that the uniform random graphs are graphs with the asymptotically smallest possible cardinality of a maximum independent union of cliques.

Studying optimization problems in the context of random graphs is of interest for several reasons. From a theoretical viewpoint, asymptotic bounds allow to estimate the magnitude of optimal objective value for random models of real-life massive graphs that cannot be tackled computationally. From a practical perspective, realizations of such graphs are typically used to assess scalability of algorithmic approaches, and the obtained theoretical bounds facilitate this process. Moreover, logarithmic upper bounds immediately imply a sub-exponential (quasi-polynomial) algorithm for the problem of interest on uniform random graphs.

The remainder of the paper is organized as follows. Section 2 provides necessary background information and introduces the concept of independent unions of Π -sets that constitutes the focus of this work. Section 3 introduces the concept of a defining function of a graph property in a random graph, which forms a basis of the methodology used in this paper, and then provides a method for establishing logarithmic upper bounds on the cardinality of independent union of Π -sets in uniform random graphs, with the main result of the paper stated in Theorem 10. Section 4 applies the proposed method to several graph topologies, including cliques, s -defective cliques, s -plexes, and γ -quasi cliques. In particular, the analysis for IUC involves an interesting connection to the Bell numbers. Section 5 uses Ramsey numbers to establish a logarithmic lower bound on the Π -independence number for any property Π that is satisfied by cliques. Finally, Section 6 concludes this paper.

2 | BACKGROUND AND MAIN CONCEPTS

Let $G = (V, E)$ be a simple graph with the vertex set $V = \{1, \dots, n\}$ and the edge set $E \subseteq \binom{V}{2}$. We call two vertices i and j *adjacent (neighbors)* if $\{i, j\} \in E$, in which case the edge $\{i, j\} \in E$ is called *incident* to i and j . A graph G can be represented by its adjacency matrix $A = [a_{ij}]_{i,j=1}^n$, whose element $a_{ij} = 1$ if $\{i, j\} \in E$, and $a_{ij} = 0$, otherwise, for $i, j = 1, \dots, n$. The edge density of G is $\rho(G) = m/\binom{n}{2}$, where $m = |E|$ is the number of edges in G .

A graph $G' = (V', E')$ is a subgraph of $G = (V, E)$, if $V' \subseteq V$ and $E' \subseteq E$. Given a subset of vertices $S \subseteq V$, the *subgraph induced by S* , denoted as $G[S]$, is obtained by deleting all vertices in $V \setminus S$ and the edges incident to at least one of them.

A path of length r between vertices v and w in G is an ordered set of distinct vertices and edges $\{v \equiv v_0, e_0, v_1, e_1, \dots, v_{r-1}, e_{r-1}, v_r \equiv w\}$ such that $e_i \equiv \{v_i, v_{i+1}\} \in E$. We call two vertices *connected* if there is at least one path between them in G . A subset of vertices $V' \subseteq V$ is called a *connected component* of G if every two vertices in V' are connected and every two vertices $v, w \in V$ such that $v \in V'$ and $w \notin V'$ are not connected. That is, a connected component is a maximal connected subgraph.

A graph with $E = \binom{V}{2}$ (i.e., a graph with all possible edges) is called *complete* and is denoted by K_n . A *clique* in G is a subset of vertices $C \subseteq V$ such that $G[C]$ is a complete graph. The maximum cardinality of a clique in G is called the *clique number*, denoted by $\omega(G)$. A subset of pairwise non-adjacent vertices $I \subseteq V$ is called an *independent (stable) set* in G . The maximum cardinality of an independent set in G is called the *independence (stability) number* of G , denoted by $\alpha(G)$.

Cliques possess many cohesiveness properties expected of a network cluster in their “ideal” form. In particular, cliques have the highest possible reachability (direct connection between any pair of vertices), familiarity (no non-neighbors within the group), and edge density, among other properties. The ideal nature of a clique makes it excessively constraining from an applied perspective, which motivated the researchers to introduce a number of alternative concepts, enforcing some of the desired clique properties in a relaxed form. In this paper we will consider some of such models, referred to as *clique relaxations* [27]. Given a positive integer constant s , a subset of vertices S is called an *s -defective clique* [38, 32] if $G[S]$ has at most s missing edges (i.e., $G[S]$ has at least $\binom{|S|}{2} - s$ edges). S is an s -

plex [30, 6] if each vertex from S has at least $|S| - s$ neighbors in $G[S]$. Given a constant $\gamma \in (0, 1]$, $S \subseteq V$ is called a γ -quasi-clique [2, 1, 26, 34] if $\rho(G[S]) \geq \gamma$.

A graph property Π is said to be *nontrivial*, if every graph with one vertex satisfies Π , and not every graph satisfies Π . A graph property Π is said to be *interesting*, if there are graphs with arbitrary large number of vertices satisfying Π . Finally, a graph property Π is said to be *hereditary on induced subgraphs*, if for every $S \subseteq V$ such that $G[S]$ satisfies Π , the property Π is also satisfied by $G[S_0]$ for any nonempty $S_0 \subseteq S$. Note, that requiring the set of vertices of a graph to form a clique, independent set, s -defective clique, or IUC are examples of graph properties that are nontrivial, interesting, and hereditary on induced subgraphs. Given a graph property Π that is nontrivial, interesting, and hereditary on induced subgraphs, finding a maximum-cardinality subset of vertices satisfying Π is known to be $\mathcal{NP}$ -hard [36].

2.1 | Independent union of Π -sets

The focus of this paper is on graph structures formally defined next.

Definition An independent component of graph $G = (V, E)$ is a subset of vertices $I \subseteq V$ such that there are no edges between I and $V \setminus I$ in G .

Note that a connected component is also an independent component, but an independent component does not have to be connected.

Definition For a graph property Π , we call $S \subseteq V$ a Π -set if $G[S]$ satisfies Π . A subset of vertices $U \subseteq V$ is called an *independent union of Π -sets* or an $\mathcal{I}(\Pi)$ -set if there exists a partitioning of U into independent components of $G[U]$ such that every component satisfies Π . The corresponding graph property is denoted by $\mathcal{I}(\Pi)$. The *maximum $\mathcal{I}(\Pi)$ problem* is to find an $\mathcal{I}(\Pi)$ -set of maximum cardinality in G . This cardinality is called the Π -independence number of G , denoted $\alpha^\Pi(G)$.

For example, a subset of vertices $U \subseteq V$ is an *independent union of cliques (IUC)* if every connected component of $G[U]$ is a complete graph [18, 17]. In other words, if the property Π^ω requires that the graph's set of vertices forms a clique, then the set of vertices of a graph satisfying $\mathcal{I}(\Pi^\omega)$ is an IUC. In this case, the Π^ω -independence number is the *IUC number* of G , namely the maximum cardinality of an IUC in G , denoted by $\alpha^\omega(G)$. Note that a clique C is an IUC consisting of a single clique and an independent set I is an IUC containing $|I|$ one-vertex cliques. Thus, $\alpha^\omega(G) \geq \alpha(G)$ and $\alpha^\omega(G) \geq \omega(G)$. An *open triangle* P_3 is a path on 3 vertices and a *closed triangle* K_3 is a 3-vertex complete graph. Clearly, $U \subseteq V$ is an IUC in G if and only if $G[U]$ contains no open triangle as its induced subgraph [17].

In another example, if Π^α requires that the set of vertices is an independent set, then $\mathcal{I}(\Pi^\alpha) \equiv \Pi^\alpha$ (an independent union of independent sets is an independent set).

We will call a graph property Π *connected* if every graph G satisfying Π is connected. If Π is connected then requiring every independent component to satisfy Π in the definition above is equivalent to requiring every connected component to satisfy Π .

Clearly, if a graph property Π is hereditary on induced subgraphs, so is $\mathcal{I}(\Pi)$.

Next, we establish some basic relations between the IUC number and $\alpha^\Pi(G)$ for properties describing several clique relaxations.

Proposition 1 1. Every s -defective clique C in G contains a clique with at least $|C| - s$ vertices.

2. Every s -plex C in G contains a clique with at least $\lceil |C|/s \rceil$ vertices.

3. Every γ -quasi-clique C in G contains a clique with at least $\lceil |C|/(|C| - \gamma(|C| - 1)) \rceil$ vertices.

Proof 1. Consider an s -defective clique C in G . Note that the graph $\bar{G}[C]$ has at most s edges, so its minimum vertex cover C' has at most s vertices. Then $C \setminus C'$ is a clique with at least $|C| - s$ vertices in G .

2. The second statement was proved in [7, Proposition 1]; here we offer a much simpler proof. According to Turán's theorem [3], we have $\omega(G) \geq \frac{|V|^2}{|V|^2 - 2|E|}$ for any simple, undirected graph $G = (V, E)$. Applying this theorem for $G[C]$ and taking into account that the number of edges in $G[C]$ is at least $|C|(|C| - s)/2$, we obtain $\omega(G[C]) \geq |C|/s$.

3. The result follows from Turán's theorem by observing that $G[C]$ has at least $\gamma|C|(|C| - 1)$ edges.

Proposition 2 1. Let Π^s : the set of vertices is an s -defective clique. Then

$$\alpha^{\Pi^s}(G) \leq \lfloor (1 + s/2)\alpha^\omega(G) \rfloor. \quad (1)$$

2. Let $\Pi^{s'}$: the set of vertices is an s -plex. Then

$$\alpha^{\Pi^{s'}}(G) \leq s\alpha^\omega(G). \quad (2)$$

Proof 1. Let U be a maximum independent union of s -defective cliques. We construct an IUC C by finding a maximum clique of each connected component of $G[U]$ and including it in C . If a connected component has 1 or 2 vertices, its whole vertex set is included in C . Otherwise, if a connected component has $k > 2$ vertices and $s' \leq s$ missing edges, its maximum clique has at least $r = \max\{2, k - s'\}$ vertices by Proposition 1. The fraction of vertices from this connected component included in C is at least $r/(r + s) \geq 2/(2 + s)$. Thus, overall, at least $2|U|/(2 + s)$ of vertices from U are included in C .

2. Let U be a maximum independent union of s -plexes. We construct an IUC C consisting of maximum cliques of connected components of $G[U]$. Clearly, each connected component of $G[U]$ is an s -plex. Thus, according to Proposition 1 a connected component of $G[U]$ on k vertices contains a clique with at least k/s vertices. Thus, C must have at least $|U|/s$ vertices.

Note that for the property Π^γ requiring the set of vertices to be a γ -quasi-clique, establishing a meaningful relation between $\alpha^{\Pi^\gamma}(G)$ and $\alpha^\omega(G)$ for arbitrary $\gamma \in (0, 1)$ appears to be more challenging than for the clique relaxations considered in Proposition 2.

3 | ASYMPTOTIC ALMOST SURE BOUNDS FOR Π -INDEPENDENCE NUMBER IN UNIFORM RANDOM GRAPHS

In this paper we deal with the Erdős-Rényi (ER) random graph model [15, 11]. According to this model, let the set Γ contain all graphs on n vertices, $n \in \mathbb{N}$, such that an edge between any pair of vertices in a graph exists with a prescribed probability $p \in [0, 1]$ independently from all other edges. Throughout the paper, by $G(n, p)$ we denote a uniform random graph with n vertices that is an element of Γ . Asymptotic behavior of maximum clique, independent set and other related structures, such as quasi-cliques, in uniform random graphs has been studied in the literature [11, 12, 33, 8]. To the best of our knowledge, this work provides the first study of independent unions of cliques and clique relaxations in random graphs.

3.1 | The defining function of a graph property in a uniform random graph

We start by introducing the concept of a *defining function* for a given graph property Π that facilitates the development of main results in this paper.

Definition A defining function $\zeta(\Pi; n, p)$ for a graph property Π is the probability that a uniform random graph $G(n, p)$ satisfies Π .

We consider examples of defining functions for the following graph properties:

- Π^α : the set of vertices of the graph is an independent set;
- Π^ω : the set of vertices of the graph is a clique;
- Π^s : the set of vertices of the graph is an s -defective clique;
- $\Pi^{s'}$: the set of vertices of the graph is an s -plex;
- Π^γ : the set of vertices of the graph is a γ -quasi-clique.

Then we have:

$$\zeta(\Pi^\alpha; n, p) = (1 - p)^{\binom{n}{2}}, \quad (3)$$

$$\zeta(\Pi^\omega; n, p) = p^{\binom{n}{2}}, \quad (4)$$

$$\zeta(\Pi^s; n, p) = \sum_{i=0}^s \binom{\binom{n}{2}}{i} p^{\binom{n}{2}-i} (1-p)^i, \quad (5)$$

$$\zeta(\Pi^\gamma; n, p) = \sum_{i=0}^{\lfloor (1-\gamma)\binom{n}{2} \rfloor} \binom{\binom{n}{2}}{i} p^{\binom{n}{2}-i} (1-p)^i. \quad (6)$$

Proposition 3 Given a graph property Π , the following assertions hold for the defining functions of Π and $\mathcal{I}(\Pi)$.

$$\zeta(\mathcal{I}(\Pi); n, p) \geq \zeta(\Pi; n, p). \quad (7)$$

If Π is nontrivial, then

$$\zeta(\Pi; 1, p) = 1 \quad (8)$$

and

$$\zeta(\mathcal{I}(\Pi); n, p) \geq (1-p)^{\binom{n}{2}}. \quad (9)$$

If Π is hereditary on induced subgraphs, then

$$\zeta(\Pi; n, p) \leq \zeta(\Pi; n-1, p) \text{ for } n \geq 2. \quad (10)$$

Proof Inequality (7) holds since a Π -set is also an $\mathcal{I}(\Pi)$ -set for any graph property Π . Equation (8) follows directly from the definition of a nontrivial graph property. The fact that Π is nontrivial also implies that an independent set is

a Π -set, so (9) follows from (3). Finally, to establish (10), let V denote the set of vertices of $G(n, p)$, where $n \geq 2$. Fix $v \in V$. Then

$$\begin{aligned}\zeta(\Pi; n, p) &= \mathbb{P}\{G[V] \text{ satisfies } \Pi\} \\ &= \mathbb{P}\{G[V] \text{ satisfies } \Pi \text{ and } G[V \setminus \{v\}] \text{ satisfies } \Pi\} \\ &\leq \mathbb{P}\{G[V \setminus \{v\}] \text{ satisfies } \Pi\} \\ &= \zeta(\Pi; n-1, p).\end{aligned}$$

This completes the proof.

3.2 | Almost sure asymptotic bounds for independent unions of Π sets

Next, we present the main lemma used to prove logarithmic upper bounds for the maximum $\mathcal{I}(\Pi)$ problems. Following the approach by Bollobas and Erdős [12], let X_k be a random variable representing the number of subgraphs on k vertices satisfying $\mathcal{I}(\Pi)$ in $G(n, p)$. Then,

$$\mathbb{P}(X_k \geq 1) \leq \mathbb{E}[X_k] = \binom{n}{k} \zeta(\mathcal{I}(\Pi); k, p). \quad (11)$$

In order to estimate the maximum cardinality of the set of vertices of a subgraph satisfying $\mathcal{I}(\Pi)$ in $G(n, p)$ with high probability, we want to find maximum k such that $\mathbb{E}[X_k] \geq 1$ [12]. Conversely, if $\mathbb{E}[X_{f(n)}] \rightarrow 0$ when $n \rightarrow \infty$, we can say that the size of the maximum $\mathcal{I}(\Pi)$ -set is bounded from above by $f(n)$ with high probability.

Lemma 4 Consider a uniform random graph $G(n, p)$ and a nontrivial, interesting property Π . Let

$$\zeta(\mathcal{I}(\Pi); n, p) \leq a \binom{n}{2} Q_n \quad (12)$$

for some $a \in (0, 1)$ and $\{Q_n : n \geq 1\}$, such that

$$Q_{n+1} \leq CnQ_n \quad (13)$$

for some $C > 0$ and all $n \geq N$, where N is a positive integer. Then, with the exception of a finite number of $n \in \mathbb{N}$, the Π -independence number $\alpha^\Pi(G(n, p))$ satisfies

$$\alpha^\Pi(G(n, p)) \leq (2 + \epsilon) \log_{\frac{1}{a}}(n) \quad (14)$$

with probability 1 for any $\epsilon > 0$.

Proof From the recurrence relation for Q_n , it follows that $Q_n \leq C^{n-1}(n-1)!q_N$, where $q_N = Q_N C^{1-N}/(N-1)!$ is a constant. Then,

$$\zeta(\mathcal{I}(\Pi); n, p) \leq q_N C^{n-1} (n-1)! a \binom{n}{2}$$

and

$$\mathbb{E}[X_k] = \binom{n}{k} \zeta(\mathcal{I}(\Pi); k, p) \leq \binom{n}{k} q_N C^{k-1} (k-1)! a^{\binom{k}{2}} \leq \frac{q_N}{Ck} \left(Cna^{\frac{k-1}{2}} \right)^k.$$

Substituting k with $f(n) = (2 + \epsilon) \log_{\frac{1}{a}}(n)$, we obtain

$$\mathbb{E}[X_{f(n)}] \leq q_N \frac{1}{Cf(n)} \left(C_1 \cdot n^{-\epsilon/2} \right)^{f(n)},$$

where $C_1 = Ca^{-\frac{1}{2}}$. For large enough values of n , it holds that $\frac{\text{const}}{n} \leq \frac{1}{n^{1/2}}$, whence the term $(C_1 n^{-\epsilon/2})^{f(n)}$ in the above inequality can be bounded as

$$(C_1 n^{-\epsilon/2})^{f(n)} \leq n^{-(\epsilon/4)f(n)} \leq n^{-2}, \quad n \gg 1,$$

since for sufficiently large n one also has $\frac{\epsilon}{4}f(n) \geq 2$. Thus, the cardinality of a maximum $\mathcal{I}(\Pi)$ -set in $G(n, p)$ is bounded from above by $f(n)$ with probability 1, by the Borel-Cantelli lemma.

Later in this section we establish the conditions ensuring that Lemma 4 guarantees a logarithmic upper bound. To establish the lower bound, existing results for the maximum clique and the maximum independent set problems can be utilized [12, 22].

Lemma 5 *Let Π be an interesting, nontrivial property. With the exception of a finite number of $n \in \mathbb{N}$, the Π -independence number of a uniform random graph $G(n, p)$ satisfies the inequality*

$$\alpha^\Pi(G(n, p)) \geq (2 - \epsilon) \log_{\frac{1}{1-p}}(n) \quad (15)$$

with probability 1 for any $\epsilon > 0$. Furthermore, if Π holds for complete graphs, then

$$\alpha^\Pi(G(n, p)) \geq (2 - \epsilon) \log_{\frac{1}{p}}(n) \quad (16)$$

with probability 1 for any $\epsilon > 0$.

Proof The first statement follows immediately from the fact that an independent set is an $\mathcal{I}(\Pi)$ -set for any nontrivial graph property Π . Thus, the bound on the independence number established in [12] applies. The second statement is a direct corollary of a clique being an $\mathcal{I}(\Pi)$ -set.

Lemma 6 *If Π is nontrivial, interesting, and hereditary on the induced subgraphs, then the cardinality of a maximum Π -set in $G(n, p)$ is $\Omega(\log n)$ with probability 1.*

Proof Follows from the logarithmic lower bounds on the clique and independence number of $G(n, p)$ [12, 22] and from the observation that any Π satisfying the lemma's conditions must be satisfied by either all cliques and all independent sets [36].

Now we are in position to present the main results of the paper. First, note that the defining function for a property $\mathcal{I}(\Pi)$ may be hard or impossible to establish in a closed form, even for cases when $\zeta(\Pi; n, p)$ is trivial to

compute. For example, for clique, the defining function is given in (4). However, as will be shown in Theorem 14 below, the defining function for IUC is expressed in terms of Bell numbers and hence probably cannot be written in a closed form.

The following statement describes a recurrence relation that can be used to express defining functions for independent unions in random graphs.

Theorem 7 *Given a nontrivial, interesting property Π with the defining function $\zeta(\Pi; n, p)$, the defining function of $\mathcal{I}(\Pi)$ satisfies the recurrence relation*

$$\zeta(\mathcal{I}(\Pi); n, p) \leq \sum_{k=1}^n \binom{n-1}{k-1} \zeta(\Pi; k, p) (1-p)^{k(n-k)} \zeta(\mathcal{I}(\Pi); n-k, p), \quad (17)$$

where $\zeta(\mathcal{I}(\Pi); 0, p) = \zeta(\mathcal{I}(\Pi); 1, p) = 1$. If Π is connected then (17) holds at equality.

Proof Assume Π is connected. Fix a vertex v_0 . Let V denote the set of vertices of $G(n, p)$. If $G(n, p)$ satisfies $\mathcal{I}(\Pi)$ then the connected component of $G(n, p)$ that contains v_0 satisfies the property Π . Assume that this connected component has k vertices given by a Π -set S_k^Π . The $k-1$ vertices to form the set S_k^Π together with v_0 can be selected in $\binom{n-1}{k-1}$ different ways. Let

$\mathbb{P}_1(n, k, p)$ denote the probability that S_k^Π is a Π -set in $G(n, p)$;

$\mathbb{P}_2(n, k, p)$ be the probability that $V \setminus S_k^\Pi$ is an $\mathcal{I}(\Pi)$ -set in $G(n, p)$;

$\mathbb{P}_3(n, k, p)$ denote the probability that there are no edges between S_k^Π and $V \setminus S_k^\Pi$ in $G(n, p)$.

Then the defining function of $\mathcal{I}(\Pi)$ can be written as

$$\zeta(\mathcal{I}(\Pi); n, p) = \sum_{k=1}^n \binom{n-1}{k-1} \mathbb{P}_1(n, k, p) \mathbb{P}_2(n, k, p) \mathbb{P}_3(n, k, p).$$

We have (see Figure 1 for an illustration):

$$\mathbb{P}_1(n, k, p) = \zeta(\Pi; k, p),$$

$$\mathbb{P}_2(n, k, p) = \zeta(\mathcal{I}(\Pi); n-k, p),$$

$$\mathbb{P}_3(n, k, p) = (1-p)^{k(n-k)},$$

yielding (17) at equality. If Π is not connected, the equality transforms into inequality, because v_0 can belong to multiple k -vertex Π -sets.

Corollary 8 *Inequality (17) can be rewritten as*

$$\zeta(\mathcal{I}(\Pi); n, p) \leq \sum_{i=0}^{n-1} \binom{n-1}{i} \zeta(\Pi; n-i, p) (1-p)^{i(n-i)} \zeta(\mathcal{I}(\Pi); i, p). \quad (18)$$

Observe that if we substitute $i(n-i) = \binom{n}{2} - \binom{i}{2} - \binom{n-i}{2}$ in (18), one might see the resulting expression as a convolution of certain generating functions [35].

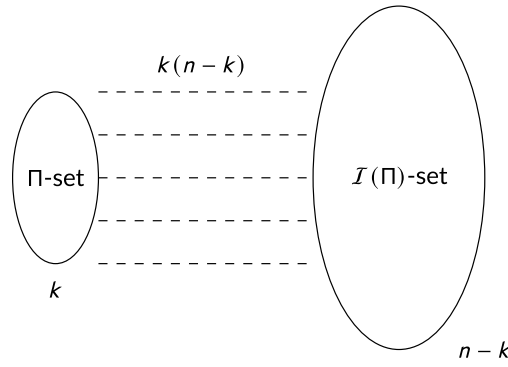


FIGURE 1 Illustration for the recurrence relation.

Lemma 9 *Let*

$$\bar{\zeta}(\mathcal{I}(\Pi); n, p) = \sum_{k=1}^n \binom{n-1}{k-1} \zeta(\Pi; k, p) (1-p)^{k(n-k)} \bar{\zeta}(\mathcal{I}(\Pi); n-k, p), \quad (19)$$

where $\bar{\zeta}(\mathcal{I}(\Pi); 0, p) = \bar{\zeta}(\mathcal{I}(\Pi); 1, p) = 1$. Then $\bar{\zeta}(\mathcal{I}(\Pi); n, p) \geq \zeta(\mathcal{I}(\Pi); n, p)$ for any $n \geq 0$.

Proof We use induction. The inequality clearly holds for $n = 0$. Assume it holds for any $n \leq N$. Then

$$\begin{aligned} \bar{\zeta}(\mathcal{I}(\Pi); N+1, p) &= \sum_{k=1}^{N+1} \binom{N}{k-1} \zeta(\Pi; k, p) (1-p)^{k(N+1-k)} \bar{\zeta}(\mathcal{I}(\Pi); N+1-k, p) \\ &\geq \sum_{k=1}^{N+1} \binom{N}{k-1} \zeta(\Pi; k, p) (1-p)^{k(N+1-k)} \zeta(\mathcal{I}(\Pi); N+1-k, p) \\ &\geq \zeta(\mathcal{I}(\Pi); N+1, p), \end{aligned}$$

where the first inequality is due to the induction assumption and the second inequality is due to (17).

Next we state the main result of this paper.

Theorem 10 *The following asymptotic bounds on the Π -independence number $\alpha^\Pi(G(n, p))$ of a uniform random graph $G(n, p)$ are valid, with the exception of, perhaps, a finite number of $n \in \mathbb{N}$:*

1. *If $\zeta(\Pi; t+1, p)/\zeta(\Pi; t, p) = O(t(1-p)^t)$ as $t \rightarrow \infty$, then*

$$\alpha^\Pi(G(n, p)) \leq (2 + \epsilon) \log_{\frac{1}{1-p}}(n) \quad (20)$$

with probability 1 for every $\epsilon > 0$.

2. *If $p \geq 1/2$ and $\zeta(\Pi; t+1, p)/\zeta(\Pi; t, p) = O(tp^t)$ as $t \rightarrow \infty$, then*

$$\alpha^\Pi(G(n, p)) \leq (2 + \epsilon) \log_{\frac{1}{p}}(n) \quad (21)$$

with probability 1 for every $\epsilon > 0$.

Proof By Lemma 9, if we show that $\bar{\zeta}(\mathcal{I}(\Pi); n, p)$ satisfies (12) for some Q_n , so will $\zeta(\mathcal{I}(\Pi); n, p)$.

1. Denote by

$$Q'_n = (1 - p)^{-\binom{n}{2}} \bar{\zeta}(\mathcal{I}(\Pi); n, p). \quad (22)$$

Then (12) is satisfied with $a = 1 - p$ and $Q_n = Q'_n$, but (13) also needs to be ensured in order for Lemma 4 to apply. Equality (18) can be rewritten as

$$Q'_n = \sum_{i=0}^{n-1} \binom{n-1}{i} \zeta_1(n-i) Q'_i, \quad (23)$$

where

$$\zeta_1(n) = \frac{\zeta(\Pi; n, p)}{(1 - p)^{\binom{n}{2}}}, \quad (24)$$

i.e., the ratio of the probability that $G(n, p)$ satisfies Π to the probability that graph's vertex set forms an independent set.

From (23) we obtain:

$$Q'_{n+1} = \sum_{i=0}^{n-1} \binom{n-1}{i} \frac{n}{n-i} \zeta_1(n-i+1) Q'_i + Q'_n. \quad (25)$$

If $\zeta(\Pi; t+1, p)/\zeta(\Pi; t, p) = O(t(1-p)^t)$, then

$$\frac{1}{n-i} \zeta_1(n-i+1) \leq C' \zeta_1(n-i)$$

for some $C' > 0$. Hence, using (23),

$$Q'_{n+1} \leq (1 + C'n) Q'_n \leq 2C'n Q'_n$$

(assuming $C'n \geq 1$). Thus, (13) holds and the result follows immediately from Lemma 4.

2. Similarly, let

$$\bar{\zeta}(\mathcal{I}(\Pi); n, p) = p^{\binom{n}{2}} Q''_n. \quad (26)$$

Then (12) is satisfied with $a = p$ and $Q_n = Q''_n$. Equality (18) can be rewritten as

$$Q''_n = \sum_{i=0}^{n-1} \binom{n-1}{i} \zeta_2(n-i) \left(\frac{1-p}{p}\right)^{i(n-i)} Q''_i, \quad (27)$$

where

$$\zeta_2(n) = \frac{\zeta(\Pi; n, p)}{p^{\binom{n}{2}}}, \quad (28)$$

that is, the ratio of the probability that $G(n, p)$ satisfies Π to the probability that graph's vertex set forms a clique. Let $r = (1 - p)/p$, then

$$Q''_{n+1} = \sum_{i=0}^{n-1} \binom{n-1}{i} \left(\frac{n}{n-i} \zeta_2(n-i+1) r^i \right) r^{i(n-i)} Q''_i + r^n Q''_n.$$

If $\zeta(\Pi; t+1, p)/\zeta(\Pi; t, p) = O(tp^t)$, then

$$\frac{1}{n-i} \zeta_2(n-i+1) r^i \leq C'' \zeta_2(n-i)$$

for some $C'' > 0$. Hence,

$$Q''_{n+1} \leq (1 + C''n) Q''_n \leq 2C''n Q''_n$$

(assuming $C''n \geq 1$). Thus, (13) holds and the result follows immediately from Lemma 4.

Corollary 11 *The results of Theorem 10 still hold if $\zeta(\Pi; t+1, p)/\zeta(\Pi; t, p)$ is replaced with $\hat{\zeta}(\Pi; t+1, p)/\hat{\zeta}(\Pi; t, p)$, where $\hat{\zeta}(\Pi; t, p) \geq \zeta(\Pi; t, p)$ for any natural t .*

Proof Apply Theorem 10 with $\hat{\zeta}(\Pi; t, p)$ and notice that (12) holds true for $\zeta(\mathcal{I}(\Pi); n, p)$.

4 | SPECIAL CASES: ASYMPTOTIC ALMOST SURE BOUNDS FOR INDEPENDENT UNIONS OF CLIQUES AND CLIQUE RELAXATIONS

4.1 | Independent union of cliques

We consider the property Π^ω , requiring the set of vertices of a graph to form a clique. We have $\zeta(\Pi^\omega; n, p) = p^{\binom{n}{2}}$, and the fraction of interest is

$$\frac{\zeta(\Pi^\omega; n+1, p)}{\zeta(\Pi^\omega; n, p)} = p^n.$$

The results of Theorem 10 apply, so $\alpha^\omega(G(n, p))$ is $O(\log n)$ with probability 1.

The defining function $\zeta(\mathcal{I}(\Pi^\omega); n, p)$ is presented for $n = 0, \dots, 6$ in Table 1. In addition, Figure 2 plots the probability of the vertex set of $G(n, p)$ to form a clique and an IUC as a function of p for $n = 3, 4, 5$, and 6.

For $n = 0, 1, 2$ the vertices of $G(n, p)$ always form an IUC and thus the probability $\zeta(\mathcal{I}(\Pi^\omega); n, p) = 1$. For $n = 3$ the graph's vertex set is an IUC if and only if it is not an open triangle. For larger n the expression quickly becomes very complicated. However, from Figure 2 one might see that for $p > 0.5$, the probability of the set of vertices of $G(n, p)$ to form an IUC quickly approaches the probability of forming a clique. Interestingly, the breaking point is not $2/3$, when the expected number of open triangles is maximized [28]. This observation will be established formally in the following theorem.

Theorem 12 *For $1/2 < p \leq 1$ we have $\zeta(\mathcal{I}(\Pi^\omega); n, p) \sim \zeta(\Pi^\omega; n, p) = p^{\binom{n}{2}}$ as $n \rightarrow \infty$.*

TABLE 1 $\zeta(\mathcal{I}(\Pi^\omega); n, p)$ for $n = 0, \dots, 6$.

n	$\zeta(\mathcal{I}(\Pi^\omega); n, p)$
0 – 2	1
3	$1 - 3p^2 + 3p^3$
4	$1 - 12p^2 + 32p^3 - 39p^4 + 24p^5 - 5p^6$
5	$1 - 30p^2 + 130p^3 - 270p^4 + 318p^5 - 195p^6 + 10p^7 + 75p^8 - 50p^9 + 12p^{10}$
6	$1 - 60p^2 + 360p^3 - 945p^4 + 792p^5 + 2775p^6 - 12090p^7 + 25035p^8 - 34240p^9 + 33204p^{10} - 23130p^{11} + 11385p^{12} - 3780p^{13} + 765p^{14} - 71p^{15}$

Proof We need to show that $\lim_{n \rightarrow \infty} \zeta(\mathcal{I}(\Pi^\omega); n, p) / p^{\binom{n}{2}} = 1$. Suppose

$$\zeta(\mathcal{I}(\Pi^\omega); t, p) = p^{\binom{t}{2}} Q_t(p),$$

where $Q_0(p) = 1$. Then substituting to (18) we have

$$p^{\binom{n}{2}} Q_n(p) = p^{\binom{n}{2}} \sum_{t=0}^{n-1} \binom{n-1}{t} \left(\frac{1-p}{p} \right)^{t(n-t)} Q_t(p).$$

Let $r = (1-p)/p$, then $0 \leq r < 1$. It is sufficient to prove that $Q_n(p) \rightarrow 1$ when $n \rightarrow \infty$. Note that

$$Q_n(r) = 1 + \sum_{t=1}^{n-1} \binom{n-1}{t} r^{t(n-t)} Q_t(r), \quad (29)$$

where, with a slight abuse of notation, we denote the dependence of Q_n on $p = 1/(1+r)$ as $Q_n(r)$. Denoting $D_n(r) = \sum_{t=1}^{n-1} \binom{n-1}{t} r^{t(n-t)} Q_t(r)$, we obtain

$$D_n(r) = \sum_{t=1}^{n-1} \binom{n-1}{t} r^{t(n-t)} (1 + D_t(r)). \quad (30)$$

We need to show $D_n(r) \rightarrow 0, n \rightarrow \infty$. Writing (30) for $n+1$ we have

$$D_{n+1}(r) = \sum_{t=1}^{n-1} \binom{n-1}{t} \left(\frac{n}{n-t} r^t \right) r^{t(n-t)} (1 + D_t(r)) + r^n (1 + D_n(r)). \quad (31)$$

Observe that

$$\frac{n}{n-t} r^t \leq \frac{n}{n-1} r, \quad t = 1, \dots, n-1. \quad (32)$$

Using (32) and factorizing (30) in (31) we can bound D_{n+1} as

$$D_{n+1}(r) \leq \left(\frac{n}{n-1} r + r^n \right) D_n(r) + r^n. \quad (33)$$

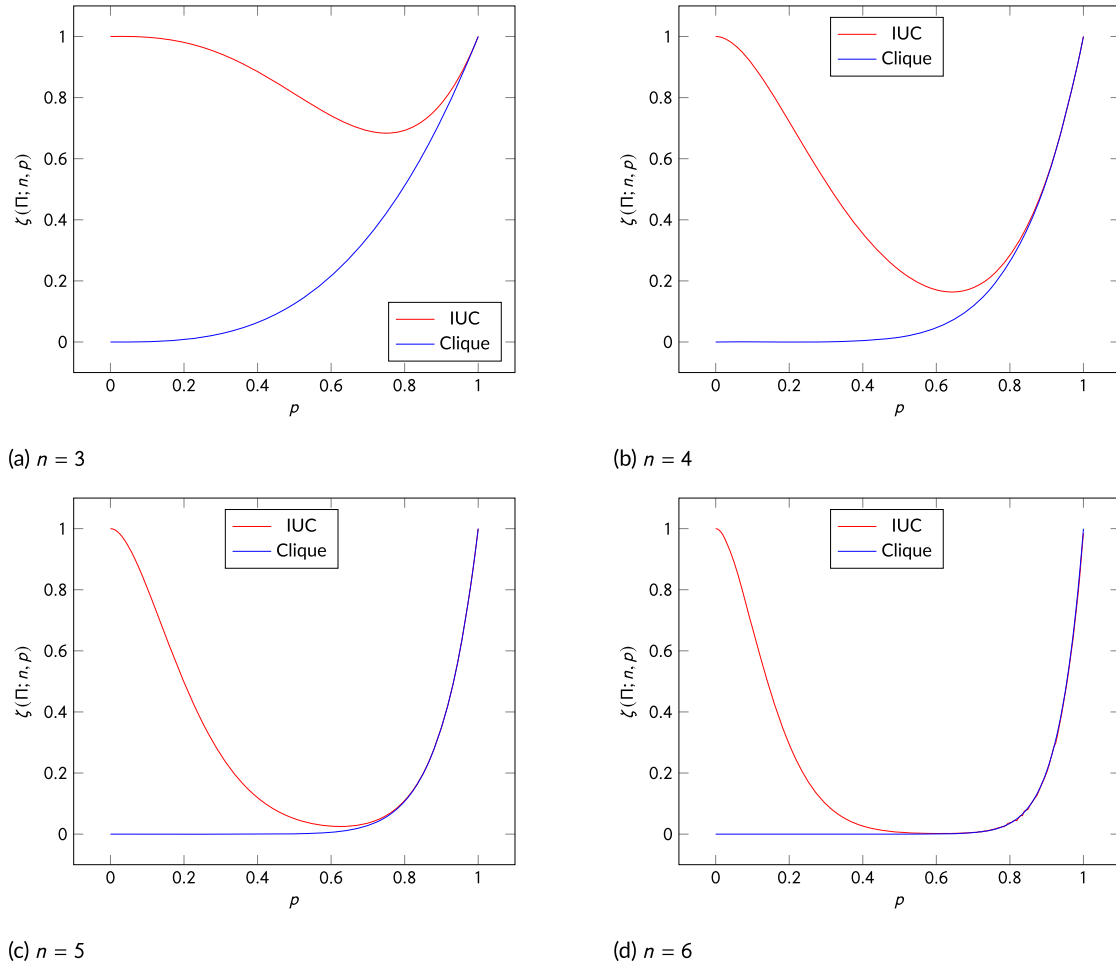


FIGURE 2 Probability $\zeta(\Pi; n, p)$ of the vertex set of $G(n, p)$ to form an IUC ($\Pi = \mathcal{I}(\Pi^\omega)$) and a clique ($\Pi = \Pi^\omega$) for $n = 3, 4, 5$, and 6 .

For $r < 1$ there exists b such that $r < b < 1$. The coefficient in parentheses in (33) tends to r when $n \rightarrow \infty$, thus there exists a natural N such that for every $n \geq N$

$$D_{n+1}(r) \leq bD_n(r) + r^n. \quad (34)$$

Thus, we have

$$D_{n+m}(r) \leq b^m D_n(r) + r^n \sum_{i=0}^{m-1} b^i \leq b^m D_n(r) + \frac{r^n}{1-b}. \quad (35)$$

Fix $\epsilon > 0$. Since $\lim_{n \rightarrow \infty} \frac{r^n}{1-b} = 0$, there exists $N_1 \in \mathbb{N}$ such that $\frac{r^n}{1-b} \leq \frac{\epsilon}{2}$ for every $n \geq N_1$. Also, $\lim_{m \rightarrow \infty} b^m D_{N_1}(r) = 0$, so there exists $N_2 \in \mathbb{N}$ such that $b^m D_{N_1}(r) \leq \frac{\epsilon}{2}$ for every $m \geq N_2$. Therefore, (35) implies that for every $n \geq N_1 + N_2$ we have $D_n(r) \leq \epsilon$. This proves that $D_n(r) \rightarrow 0$ when $n \rightarrow \infty$ for any $0 \leq r < 1$, whereby $Q_n(p) \rightarrow 1$, $n \rightarrow \infty$ for any $1/2 < p \leq 1$ by virtue of (29).

Using techniques from [12] we obtain the following corollary of Theorem 12. Alternatively, this statement can be deduced from the bounds furnished by Theorem 10 and Lemma 5.

Corollary 13 For $1/2 < p \leq 1$ we have $\alpha^\omega(G(n, p)) \sim \omega(G(n, p))$ as $n \rightarrow \infty$ with probability 1.

For $p = 0.5$, the defining function for IUC can be expressed in terms of the n -th Bell number, which is defined as the number of partitions of a set of n elements [19].

Theorem 14 For $p = 0.5$, we have

$$\zeta(I(\Pi^\omega); n, 0.5) = 2^{-\binom{n}{2}} B_n, \quad (36)$$

where B_n is the n -th Bell number.

Proof Substitute $\zeta(\Pi^\omega; n, 0.5) = 2^{-\binom{n}{2}}$ to (18). Then

$$\zeta(I(\Pi^\omega); n, 0.5) = 2^{-\binom{n}{2}} \sum_{t=0}^{n-1} \binom{n-1}{t} 2^{\binom{t}{2}} \zeta(I(\Pi^\omega); t, 0.5). \quad (37)$$

Define $B_n = 2^{\binom{n}{2}} \zeta(I(\Pi^\omega); n, 0.5)$. Then B_n satisfies the recursive relation

$$B_n = \sum_{t=0}^{n-1} \binom{n-1}{t} B_t \quad (38)$$

with $B_0 = 1$. This relation is precisely the relation for the n -th Bell number [35], so $\zeta(I(\Pi^\omega); n, 0.5) = 2^{-\binom{n}{2}} B_n$.

Theorem 14 can alternatively be proved as follows. Notice, that for $p = 0.5$, every realization of a random graph $G(n, p)$ has the same probability $2^{-\binom{n}{2}}$ to appear. Hence, in this case, the probability of a random graph to form an IUC is equal to $2^{-\binom{n}{2}} k$, where k is the number of different IUCs on n vertices. This number can be thought of as the number of different partitions of a set of n vertices into subsets, so that each subset defines a clique in the corresponding IUC. Thus, $k = B_n$ by the definition of the n -th Bell number.

The following upper bound on the n -th Bell number was derived in [9]:

$$B_n < \left(\frac{0.792n}{\ln(n+1)} \right)^n. \quad (39)$$

Thus, according to (11), the random variable X_k representing the number of k -vertex IUCs in $G(n, p)$ satisfies

$$\mathbb{E}(X_k) = \binom{n}{k} 2^{-\binom{k}{2}} B_k \leq \binom{n}{k} 2^{-\binom{k}{2}} \left(\frac{0.792k}{\ln(k+1)} \right)^k. \quad (40)$$

Figure 3 shows the maximum k for which the upper bound on $\mathbb{E}(X_k)$ given in (40) is guaranteed to be at least 1 according to (40) and compares it to the maximum k for which the expected number of k -vertex cliques in $G(n, p)$ is at least 1. As can be seen from this figure, the curve for IUC is quite close to the logarithmic upper bound curve.

Clearly, for $p = 0$ the whole set V of vertices forms an independent set (and hence, an IUC) with probability 1. Next we show that the probability that the vertex set of $G(n, p)$ forms an IUC is infinitely larger than that of forming an independent set for $0 < p \leq 1$, so we observe the so-called phase transition effect at $p = 0$.

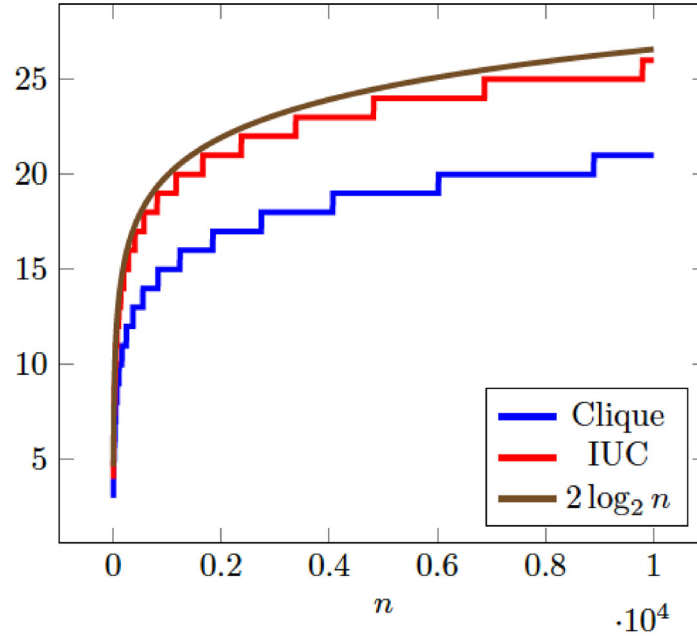


FIGURE 3 Theoretical estimates of the maximum clique and IUC sizes for $G(n, 0.5)$.

Theorem 15 For $0 < p \leq 1$ we have $\zeta(I(\Pi^\omega); n, p)/(1-p)^{\binom{n}{2}} \rightarrow \infty$ as $n \rightarrow \infty$.

Proof For $p > 0.5$ and $p = 0.5$ the result follows directly from Theorems 12 and 14, respectively. Assume $0 < p < 0.5$. Let

$$\zeta(I(\Pi^\omega); t, p) = (1-p)^{\binom{t}{2}} Q_t,$$

where $Q_0 = 1$, and we suppress Q_n 's dependence on p for brevity. Then substituting this and $\zeta(\Pi^\omega; n-i, p) = p^{\binom{n-i}{2}}$ into (18) we obtain

$$(1-p)^{\binom{n}{2}} Q_n = \sum_{i=0}^{n-1} \binom{n-1}{i} p^{\binom{n-i}{2}} (1-p)^{\binom{n}{2} - \binom{n-i}{2}} Q_i.$$

Denoting $\bar{r} = p/(1-p)$, we get

$$(1-p)^{\binom{n}{2}} Q_n = (1-p)^{\binom{n}{2}} \sum_{i=0}^{n-1} \binom{n-1}{i} \bar{r}^{\binom{n-i}{2}} Q_i.$$

Thus, we need to prove that $Q_n \rightarrow \infty$ as $n \rightarrow \infty$ for

$$Q_n = \sum_{i=0}^{n-1} \binom{n-1}{i} \bar{r}^{\binom{n-i}{2}} Q_i.$$

For this purpose, we show that

$$Q_n \geq 1 + \binom{n}{2} \bar{r} \quad (41)$$

for any $n \geq 0$. We use induction on n . The statement clearly holds for $n = 0, 1$. Assume (41) holds for some $n \geq 2$. Then for Q_{n+1} we have

$$\begin{aligned} Q_{n+1} &= \sum_{i=0}^n \binom{n}{i} \bar{r}^{\binom{n+1-i}{2}} Q_i \\ &= Q_n + n \bar{r} Q_{n-1} + \sum_{i=0}^{n-2} \binom{n}{i} \bar{r}^{\binom{n+1-i}{2}} Q_i \\ &\geq 1 + \binom{n}{2} \bar{r} + n \bar{r} \left(1 + \binom{n-1}{2} \bar{r} \right) \\ &\geq 1 + \binom{n+1}{2} \bar{r}. \end{aligned}$$

This completes the proof.

4.2 | Independent union of s -defective cliques

Given a positive integer constant s , we consider the property Π^s , requiring the set of vertices to form an s -defective clique.

Theorem 16 *The cardinality of a maximum $\mathcal{I}(\Pi^s)$ -set in $G(n, p)$ is $O(\log n)$ with probability 1.*

Proof We show that Theorem 10 applies in this case. We have

$$\zeta(\Pi^s; n, p) \leq \hat{\zeta}(\Pi^s; n, p) = \binom{\binom{n}{2}}{s} p^{\binom{n}{2}-s}.$$

Consider the fraction

$$\frac{\hat{\zeta}(\Pi^s; n+1, p)}{\hat{\zeta}(\Pi^s; n, p)} = p^n \frac{\binom{\binom{n+1}{2}}{s}}{\binom{\binom{n}{2}}{s}}.$$

We have

$$\frac{\binom{\binom{n+1}{2}}{s}}{\binom{\binom{n}{2}}{s}} = \frac{((\binom{n}{2} + n)! (\binom{n}{2} - s)!)}{((\binom{n}{2} + n - s)! (\binom{n}{2})!)} = \prod_{i=0}^{s-1} \frac{\binom{n}{2} + n - i}{\binom{n}{2} - i}.$$

We can show that $\frac{\binom{n}{2} + n - i}{\binom{n}{2} - i} \leq 2$, $i \in \{0, \dots, s-1\}$ for sufficiently large n . Thus,

$$\frac{\hat{\zeta}(\Pi^s; n+1, p)}{\hat{\zeta}(\Pi^s; n, p)} \leq 2^s np^n$$

and the cardinality of a maximum independent union of s -defective cliques in $G(n, p)$ is $O(\log n)$ with probability 1.

Alternatively, the logarithmic upper bound for a union of s -defective cliques could be established as a corollary of (1) in Proposition 2 and the logarithmic bound on the IUC number.

4.3 | Independent union of s -plexes

Let $\Pi^{s'}$ be the graph property requiring the set of vertices to form an s -plex.

Theorem 17 *The cardinality of a maximum $\mathcal{I}(\Pi^{s'})$ -set in $G(n, p)$ is $O(\log n)$ with probability 1.*

Proof The result follows from (2) in Proposition 2 and the logarithmic bound on the IUC number.

For $p < 0.5$, the statement can also be established using Theorem 10 as follows. Observe that for an s -plex S the corresponding induced subgraph $G[S]$ has at most $\lfloor sn/2 \rfloor$ missing edges. Obviously, an s -plex is also an $\hat{s}$ -defective clique with $\hat{s} = sn$. Hence,

$$\zeta(\Pi^{s'}; n, p) \leq \hat{\zeta}(\Pi^{s'}; n, p) = \binom{\binom{n}{2}}{sn} p^{\binom{n}{2} - sn}.$$

We have

$$\frac{\hat{\zeta}(\Pi^{s'}; n+1, p)}{\hat{\zeta}(\Pi^{s'}; n, p)(1-p)^n} = p^{-s} \left(\frac{p}{1-p} \right)^n \frac{\binom{\binom{n+1}{2}}{s(n+1)}}{\binom{\binom{n}{2}}{sn}}.$$

The fraction on the right can be rewritten as

$$\begin{aligned} \frac{\binom{\binom{n+1}{2}}{s(n+1)}}{\binom{\binom{n}{2}}{sn}} &= \frac{\prod_{i=1-sn-s+n}^n \left(\binom{n}{2} + i \right)}{\prod_{i=1}^s (sn+i) \prod_{i=1-sn}^0 \left(\binom{n}{2} + i \right)} = \frac{\prod_{i=1}^n \left(\binom{n}{2} + i \right)}{\prod_{i=1}^s (sn+i) \prod_{i=1-sn}^{n-sn-s} \left(\binom{n}{2} + i \right)} \\ &\leq \frac{\prod_{i=1}^{n-s} \left(\binom{n}{2} + i \right) \prod_{i=n-s+1}^n \left(\binom{n}{2} + i \right)}{\prod_{i=1}^{n-s} \left(\binom{n}{2} - sn + i \right)} \\ &\leq \left(1 + \frac{3s}{n-s} \right)^{n-s} \prod_{i=1-s}^0 \left(\binom{n}{2} + n + i \right) \leq \exp\{3s\} \cdot f_s(n), \end{aligned}$$

where $f_s(n) = \prod_{i=1-s}^0 \left(\binom{n}{2} + n + i \right)$ is $O(n^{2s})$. Hence, the first condition in Theorem 10 holds for $p < 0.5$.

4.4 | Independent union of γ -quasi-cliques

Let Π^γ be the property requiring the set of vertices to form a γ -quasi-clique.

Theorem 18 *The cardinality of a maximum $\mathcal{I}(\Pi^\gamma)$ -set in $G(n, p)$, where $p < 0.5$ and γ satisfies the inequality*

$$\frac{p^\gamma}{1-p} \cdot \frac{1}{\gamma^\gamma (1-\gamma)^{1-\gamma}} \leq 1, \quad (42)$$

is $O(\log n)$ with probability 1.

Proof A γ -quasi clique can be seen as a defective clique, missing at most $\lfloor \gamma \binom{n}{2} \rfloor$ edges. Then the defining function satisfies

$$\zeta(\Pi^\gamma; n, p) \leq \hat{\zeta}(\Pi^\gamma; n, p) = \left(\binom{n}{2} \right) p^{\lfloor \gamma \binom{n}{2} \rfloor}.$$

We have

$$\frac{\hat{\zeta}(\Pi^\gamma; n+1, p)}{\hat{\zeta}(\Pi^\gamma; n, p)(1-p)^n} \approx \left(\frac{p^\gamma}{1-p} \right)^n \frac{\left(\binom{n+1}{2} \right) p^{\lfloor \gamma \binom{n+1}{2} \rfloor}}{\left(\binom{n}{2} \right) p^{\lfloor \gamma \binom{n}{2} \rfloor}}. \quad (43)$$

By Stirling approximation,

$$\binom{k}{i} \sim \sqrt{\frac{k}{2\pi i(k-i)}} \frac{k^k}{i^i (k-i)^{(k-i)}}.$$

In particular, for $i = \lfloor \gamma k \rfloor$, we have $\binom{k}{\lfloor \gamma k \rfloor} \sim \left(\gamma^\gamma (1-\gamma)^{(1-\gamma)k} \sqrt{2\pi\gamma(1-\gamma)k} \right)^{-1}$.

Hence, the binomial fraction in (43) can be estimated as

$$\left(\gamma^\gamma (1-\gamma)^{1-\gamma} \right)^{-n}.$$

The first condition in Theorem 10 is satisfied if (42) holds.

Figure 4 illustrates the ranges of γ values that satisfy (42) for $p < 0.5$. The solutions for (42) were computed numerically using the bisection method. In order to obtain an analytic estimate, we can solve the stronger inequality

$$\frac{p^\gamma}{1-p^\gamma} \cdot \frac{1}{\gamma^\gamma (1-\gamma)^{1-\gamma}} \leq 1$$

with the expression

$$p \leq \left(\frac{\gamma^\gamma (1-\gamma)^{1-\gamma}}{1 + \gamma^\gamma (1-\gamma)^{1-\gamma}} \right)^{1/\gamma} \quad (44)$$

(plotted using the dashed curve in Figure 4).

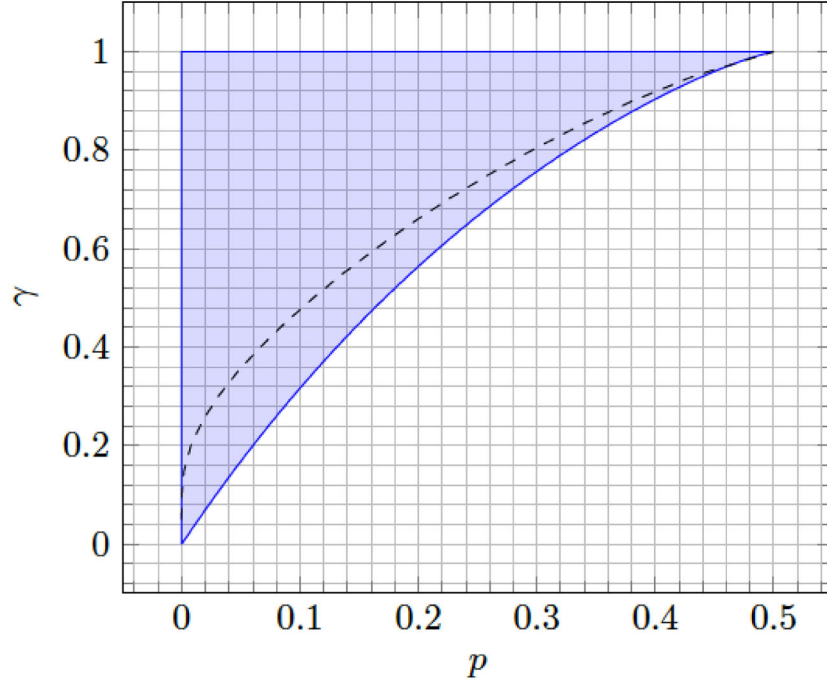


FIGURE 4 Acceptable γ values with respect to probability p . The dashed curve shows the analytic approximation (44).

5 | TIGHTNESS OF THE LOWER BOUND ON THE Π -INDEPENDENCE NUMBER AND “CLUSTERABILITY” OF UNIFORM RANDOM GRAPHS

Both an independent set and a clique are feasible solutions for the maximum IUC problem, as well as for the maximum IUC problem corresponding to any property Π satisfied by cliques. In this section, we assume that Π is satisfied by cliques, which is the case for, e.g., all clique relaxations. The maximum clique problem can have a solution as small as a singleton vertex, when the graph has no edges. Similarly, the same is true about the smallest possible solution for the maximum independent set problem in a complete graph. In both of these cases, the size of the maximum IUC is the largest possible, since the graph’s set of vertices forms an IUC . Hence, determining the smallest possible size of a maximum IUC in a graph on n vertices appears to be a much more challenging question. Lemmata 5 and 6 partially answer this question for the case of uniform random graphs. To address it for arbitrary graphs, we can employ the *Ramsey numbers*, which are well studied in combinatorics [29]. Recall that a Ramsey number $R(s, t)$ is the smallest number such that every graph $G = (V, E)$ with $|V| \geq R(s, t)$ must contain either a clique of cardinality s or an independent set of cardinality t .

It is easy to see that for a graph of order $n \geq R(s, s)$, the cardinality of its maximum IUC must be at least s . The classical result by Erdős and Szekeres [16] states that

$$R(s, s) \leq \binom{2(s-2)}{s-2} + 1 \leq 4^{s-2} + 1.$$

So, for a graph of cardinality $n = 4^{s-2} + 1$, we have

$$\log_2(n - 1) \leq 2(s - 2),$$

implying that the cardinality of a maximum IUP is at least $(\log_2 n)/2$. Since the cardinality of the maximum IUP is at most $O(\log n)$ for ER random graphs, the bound is tight within a constant factor.

The fact that the logarithmic lower bound on the Π -independence number is asymptotically tight for uniform random graphs suggests that this class of graphs has the lowest possible “clusterability” in the sense that the number of vertices belonging to the seed clusters obtained by solving the maximum IUP problem is the smallest possible (within a constant factor). This is the case, in particular, for all the properties Π corresponding to clique and clique relaxations (s -defective clique, as well as s -plex and γ -quasi-clique for $p < 0.5$) for which we were able to establish logarithmic upper bounds on the Π -independence number. Hence, we can think of the uniform random graphs as essentially “unclusterable” with respect to the IUP-based clustering methods for the considered properties. This is in agreement with a common sense assumption that no reasonable clustering method is expected to yield a meaningful clustering on ER uniform random graphs. In fact, one of the most popular clustering performance measures, modularity [24], is based on contrasting the fraction of the edges falling within the clusters to the expected fraction of the edges falling within the clusters in a random graph with the same degree sequence as that of the graph being clustered.

It should be noted that for some graph properties, corresponding to other clique relaxations of practical importance, logarithmic bounds do not hold. Consider, for example, an s -club, which is a subset of vertices inducing a subgraph with diameter at most s [23, 31, 25]. For $s = 1$, s -club is a clique, and for $s > 1$, any vertex together with its closed neighborhood is an s -club. In particular, a node with the maximum degree and its neighbors form a feasible solution to the maximum s -club problem. In $G(n, p)$, the expected degree of a vertex is np , thus the size of the maximum s -club has no logarithmic upper bound. In addition, s -club is also a feasible solution to the maximum independent union of s -clubs problem. Thus, $G(n, p)$ has an independent union of s -clubs of cardinality $\Omega(n)$ with high probability. Having $\Omega(n)$ vertices of $G(n, p)$ in a single cluster indicates that a clustering method based on a maximum independent union of s -clubs may not be meaningful. Minimum s -club partitioning [37] appears to be much more reasonable from an applied perspective for this particular clique relaxation.

6 | CONCLUSIONS

This paper develops a method for establishing a logarithmic upper bound on the Π -independence number. Table 2 summarizes the cases of Π for which we successfully applied the main result, Theorem 10. Each of the corresponding properties was motivated by important applications studied in the literature. In the cases of s -plex and γ -quasi-clique, we were able to apply the proposed method only for sparse graphs (with $p < 0.5$ and some additional restrictions on γ in the case of γ -quasi-clique), however, based on some numerical experiments we conducted, we conjecture that the logarithmic bound may hold for $p \geq 0.5$ as well. This shows the limitations of the proposed methodology; new techniques are needed in order to address the dense cases effectively.

References

- [1] J. Abello, P. Pardalos, and M. Resende, “On maximum clique problems in very large graphs,” *External memory algorithms and visualization*, J. Abello and J. Vitter (eds.), American Mathematical Society, Providence, RI, 1999, pp. 119–130.

TABLE 2 Summary of the problems for which logarithmic bounds were obtained for sparse ($p < 0.5$) and dense ($p \geq 0.5$) random graphs.

Π	Sparse	Dense
clique	✓	✓
s -defective clique	✓	✓
s -plex	✓	✓ ⁺
γ -quasi clique	✓ [*]	

*: for certain values of γ

⁺: proof is not based on Theorem 10

- [2] J. Abello, M. Resende, and S. Sudarsky, *Massive quasi-clique detection*, LATIN 2002: Theoretical Informatics, Springer-Verlag, London, 2002, pp. 598–612.
- [3] M. Aigner, *Turán's graph theorem*, Am. Math. Monthly **102** (1995), 808–816.
- [4] D. Aloise, S. Cafieri, G. Caporossi, P. Hansen, S. Perron, and L. Liberti, *Column generation algorithms for exact modularity maximization in networks*, Phys. Rev. E **82** (Oct 2010), 046112.
- [5] D.A. Bader, H. Meyerhenke, P. Sanders, and D. Wagner (eds.), *Graph Partitioning and Graph Clustering*, American Mathematical Society, Providence, RI, 2013.
- [6] B. Balasundaram, S. Butenko, and I. Hicks, *Clique relaxations in social network analysis: the maximum k -plex problem*, Oper. Res. **59** (2011), 133–142.
- [7] B. Balasundaram, S.S. Chandramouli, and S. Trukhanov, *Approximation algorithms for finding and partitioning unit-disk graphs into co- k -plexes*, Optim. Lett. **4** (2010), 311–320.
- [8] P. Balister, B. Bollobás, J. Sahasrabudhe, and A. Veremyev, *Dense subgraphs in random graphs*, Discr. Appl. Math. **260** (2019), 66–74.
- [9] D. Berend and T. Tassa, *Improved bounds on Bell numbers and on moments of sums of random variables*, Prob. Math. Stat. **30** (2010), 185–205.
- [10] S. Böcker and J. Baumbach, "Cluster editing," *The Nature of Computation. Logic, Algorithms, Applications*. CiE 2013, P. Bonizzoni, V. Brattka, and B. Löwe (eds.), Springer, Berlin, Heidelberg, 2013, pp. .
- [11] B. Bollobás, "Random graphs," *Modern Graph Theory*, Springer, New York, 1998, pp. 215–252.
- [12] B. Bollobás and P. Erdős, *Cliques in random graphs*, Mathematical Proceedings of the Cambridge Philosophical Society, Vol. 80, 1976, pp. 419–427.
- [13] U. Brandes and T. Erlebach (eds.), *Network Analysis: Methodological Foundations* Vol. 3418 of *Lecture Notes in Computer Science*, Springer, Berlin, Heidelberg, 2005.
- [14] J. Chen and J. Meng, *A $2k$ kernel for the cluster editing problem*, J. Comput. System Sci. **78** (2012), 211–220.
- [15] P. Erdős and A. Rényi, *On the evolution of random graphs*, Publ. Math. Inst. Hung. Acad. Sci. **5** (1960), 17–60.
- [16] P. Erdős and G. Szekeres, *A combinatorial problem in geometry*, Compositio Mathematica **2** (1935), 463–470.

- [17] Z. Ertem, E. Lykhovyd, Y. Wang, and S. Butenko, *The maximum independent union of cliques problem: complexity and exact approaches*, J. Global Optim. **76** (2020), 545–562.
- [18] Z. Ertem, A. Veremyev, and S. Butenko, *Detecting large cohesive subgroups with high clustering coefficients in social networks*, Social Networks **46** (2016), 1–10.
- [19] M. Gardner, *Bells-versatile numbers that can count partitions of a set, primes and even rhymes*, Sci. Am. **238** (1978), 24–32.
- [20] P. Hansen and B. Jaumard, *Cluster analysis and mathematical programming*, Math. Program. **79** (Oct 1997), 191–215.
- [21] F. Hüffner, C. Komusiewicz, H. Moser, and R. Niedermeier, *Fixed-parameter algorithms for cluster vertex deletion*, Theory Comput. Syst. **47** (2010), 196–217.
- [22] D.W. Matula, *The largest clique size in a random graph*, Tech. report CS 7608, Department of Computer Science, Southern Methodist University, 1976.
- [23] R. Mokken, *Cliques, clubs and clans*, Qual. Quantity **13** (1979), 161–173.
- [24] M.E.J. Newman, *Modularity and community structure in networks*, Proc. National Acad. Sci. United States America **103** (2006), 8577–8696.
- [25] F.M. Pajouh, B. Balasundaram, and I.V. Hicks, *On the 2-club polytope of graphs*, Oper. Res. **64** (2016), 1466–1481.
- [26] J. Pattillo, A. Veremyev, S. Butenko, and V. Boginski, *On the maximum quasi-clique problem*, Discr. Appl. Math. **161** (2013), 244–257.
- [27] J. Pattillo, N. Youssef, and S. Butenko, *On clique relaxation models in network analysis*, Eur. J. Oper. Res. **226** (2013), 9–18.
- [28] A. Pyatkin, E. Lykhovyd, and S. Butenko, *The maximum number of induced open triangles in graphs of a given order*, Optim. Lett. (2019), To appear, DOI: 10.1007/s11590-018-1330-2.
- [29] S.P. Radziszowski, *Small Ramsey numbers*, Electron. J. Combin. (2004), Dynamical Survey DS1, 1–42.
- [30] S.B. Seidman and B.L. Foster, *A graph theoretic generalization of the clique concept*, J. Math. Sociology **6** (1978), 139–154.
- [31] S. Shahinpour and S. Butenko, *“Distance-based clique relaxations in networks: s-cliques and s-clubs,” Models, Algorithms, and Technologies for Network Analysis*, B.I. Goldengorin, V.A. Kalyagin, and P.M. Pardalos (eds.), Springer Science + Business Media, New York, 2013, pp. 149–174.
- [32] S. Trukhanov, C. Balasubramaniam, B. Balasundaram, and S. Butenko, *Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations*, Comput. Optim. Appl. **56** (2013), 113–130.
- [33] A. Veremyev, V. Boginski, P.A. Krokhmal, and D.E. Jeffcoat, *Dense percolation in large-scale mean-field random networks is provably “explosive”*, PloS One **7** (2012), e51883.
- [34] A. Veremyev, O.A. Prokopyev, S. Butenko, and E.L. Pasiliao, *Exact MIP-based approaches for finding maximum quasi-cliques and dense subgraphs*, Comput. Optim. Appl. **64** (2016), 177–214.
- [35] H.S. Wilf, *Generatingfunctionology: Third Edition*, A K Peters, Ltd., Wellesley, Massachusetts, 2005.
- [36] M. Yannakakis, *Node-and edge-deletion NP-complete problems*, Proceedings of the tenth annual ACM symposium on Theory of computing, 1978, pp. 253–264.
- [37] O. Yezerska, F.M. Pajouh, A. Veremyev, and S. Butenko, *Exact algorithms for the minimum s-club partitioning problem*, Ann. Oper. Res. **276** (2019), 267–291.
- [38] H. Yu, A. Paccanaro, V. Trifonov, and M. Gerstein, *Predicting interactions in protein networks by completing defective cliques*, Bioinformatics **22** (2006), 823–829.