



AFRL-RI-RS-TR-2022-054

# **MACHINE LEARNING-EVOLVED BIOPHYSICAL MODELS FOR ENGINEERING GENETIC CIRCUITS AND PATHWAYS**

---

PENNSYLVANIA STATE UNIVERSITY

*MARCH 2022*

FINAL TECHNICAL REPORT

***APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED***

STINFO COPY

**AIR FORCE RESEARCH LABORATORY  
INFORMATION DIRECTORATE**

■ AIR FORCE MATERIEL COMMAND

■ UNITED STATES AIR FORCE

■ ROME, NY 13441

## NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09. This report is available to the general public, including foreign nations. Copies may be obtained from the Defense Technical Information Center (DTIC) (<http://www.dtic.mil>).

AFRL-RI-RS-TR-2022-054 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

FOR THE CHIEF ENGINEER:

/ S /

CARL R. THOMAS  
Work Unit Manager

/ S /

GREGORY J. HADYNSKI  
Assistant Technical Advisor  
Computing & Communications Division  
Information Directorate

This report is published in the interest of scientific and technical information exchange, and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

## REPORT DOCUMENTATION PAGE

|                                                                                                                                                                                                                                                                                                                                                                                        |                             |                                |                                                   |                                                     |                                                                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------|--------------------------------|---------------------------------------------------|-----------------------------------------------------|-----------------------------------------------------------------------------|
| <b>1. REPORT DATE</b>                                                                                                                                                                                                                                                                                                                                                                  |                             | <b>2. REPORT TYPE</b>          |                                                   | <b>3. DATES COVERED</b>                             |                                                                             |
| MARCH 2022                                                                                                                                                                                                                                                                                                                                                                             |                             | FINAL TECHNICAL REPORT         |                                                   | <b>START DATE</b><br>SEPTEMBER 2022                 | <b>END DATE</b><br>OCTOBER 2021                                             |
| <b>4. TITLE AND SUBTITLE</b><br>MACHINE LEARNING-EVOLVED BIOPHYSICAL MODELS FOR ENGINEERING GENETIC CIRCUITS AND PATHWAYS                                                                                                                                                                                                                                                              |                             |                                |                                                   |                                                     |                                                                             |
| <b>5a. CONTRACT NUMBER</b><br>FA8750-17-C-0254                                                                                                                                                                                                                                                                                                                                         |                             | <b>5b. GRANT NUMBER</b><br>N/A |                                                   | <b>5c. PROGRAM ELEMENT NUMBER</b><br>61101E         |                                                                             |
| <b>5d. PROJECT NUMBER</b>                                                                                                                                                                                                                                                                                                                                                              |                             | <b>5e. TASK NUMBER</b>         |                                                   | <b>5f. WORK UNIT NUMBER</b><br>R2DN                 |                                                                             |
| <b>6. AUTHOR(S)</b><br>Howard M. Salis, Travis L. LaFleur, Ayaan Hossain, Alexander C. Reis, Sean M. Halper, Grace E. Vezeau, Daniel P. Cetnar                                                                                                                                                                                                                                         |                             |                                |                                                   |                                                     |                                                                             |
| <b>7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)</b><br>Pennsylvania State University<br>Office of Sponsored Research<br>110 Technology Center Building<br>University Park PA 16802-7000                                                                                                                                                                                          |                             |                                |                                                   | <b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>     |                                                                             |
| <b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b><br>DARPA<br>675 North Randolph Street<br>Arlington VA 22203-2114                                                                                                                                                                                                                                                        |                             |                                | <b>10. SPONSOR/MONITOR'S ACRONYM(S)</b><br><br>RI |                                                     | <b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b><br><br>AFRL-RI-RS-TR-2022-054 |
| <b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b><br>Approved for Public Release; Distribution Unlimited. This report is the result of contracted fundamental research deemed exempt from public affairs security and policy review in accordance with SAF/AQR memorandum dated 10 Dec 08 and AFRL/CA policy clarification memorandum dated 16 Jan 09.                                    |                             |                                |                                                   |                                                     |                                                                             |
| <b>13. SUPPLEMENTARY NOTES</b>                                                                                                                                                                                                                                                                                                                                                         |                             |                                |                                                   |                                                     |                                                                             |
| <b>14. ABSTRACT</b><br><br>This project sought to further Synthetic Biology by engineering large genetic circuits and metabolic pathway models that hold true in real world experiments. Hybrid Biophysics-Machine Learning Models were developed to achieve predictive design.                                                                                                        |                             |                                |                                                   |                                                     |                                                                             |
| <b>15. SUBJECT TERMS</b><br>Synthetic Biology, Machine Learning, Statistical Thermodynamics, Computational Optimization, Biophysics, Genetic System Design, Metabolic Pathway, Explosive Sensors, CRISPR, Ribosome Binding Site, Riboswitch, Promoter, mRNA decay, Protein Expression, Operon, DNA synthesis, Oligopool Synthesis, Barcoding, DNA assembly, Next-generation Sequencing |                             |                                |                                                   |                                                     |                                                                             |
| <b>16. SECURITY CLASSIFICATION OF:</b>                                                                                                                                                                                                                                                                                                                                                 |                             |                                |                                                   | <b>17. LIMITATION OF ABSTRACT</b>                   |                                                                             |
| <b>a. REPORT</b><br><br>U                                                                                                                                                                                                                                                                                                                                                              | <b>b. ABSTRACT</b><br><br>U | <b>c. THIS PAGE</b><br><br>U   | <b>SAR</b>                                        |                                                     | <b>18. NUMBER OF PAGES</b><br><br>100                                       |
| <b>19a. NAME OF RESPONSIBLE PERSON</b><br>CARL R. THOMAS                                                                                                                                                                                                                                                                                                                               |                             |                                |                                                   | <b>19b. PHONE NUMBER (Include area code)</b><br>N/A |                                                                             |

## Table of Contents

|                                                                                                                                       |     |
|---------------------------------------------------------------------------------------------------------------------------------------|-----|
| LIST OF FIGURES.....                                                                                                                  | ii  |
| LIST OF TABLES.....                                                                                                                   | iii |
| 1.0 SUMMARY .....                                                                                                                     | 1   |
| 2.0 INTRODUCTION .....                                                                                                                | 2   |
| 3.0 METHODS, ASSUMPTIONS, AND PROCEDURES .....                                                                                        | 2   |
| 3.1 AUTOMATED MODEL-PREDICTIVE DESIGN OF SYNTHETIC PROMOTERS TO CONTROL<br>TRANSCRIPTIONAL PROFILES IN BACTERIA.....                  | 2   |
| 3.2 AN AUTOMATED MODEL TEST SYSTEM FOR SYSTEMATIC DEVELOPMENT AND IMPROVEMENT OF<br>GENE EXPRESSION MODELS.....                       | 13  |
| 3.3 THE SYNTHESIS SUCCESS CALCULATOR: PREDICTING THE RAPID SYNTHESIS OF DNA<br>FRAGMENTS WITH MACHINE LEARNING.....                   | 15  |
| 3.4 AUTOMATED DESIGN OF THOUSANDS OF HIGHLY NON-REPETITIVE GENETIC PARTS FOR<br>ENGINEERING EVOLUTIONARY ROBUST GENETIC SYSTEMS ..... | 18  |
| 4.0 RESULTS AND DISCUSSION .....                                                                                                      | 21  |
| 4.1 AUTOMATED MODEL-PREDICTIVE DESIGN OF SYNTHETIC PROMOTERS TO CONTROL<br>TRANSCRIPTIONAL PROFILES IN BACTERIA.....                  | 21  |
| 4.2 AN AUTOMATED MODEL TEST SYSTEM FOR SYSTEMATIC DEVELOPMENT AND IMPROVEMENT OF<br>GENE EXPRESSION MODELS.....                       | 37  |
| 4.3 THE SYNTHESIS SUCCESS CALCULATOR: PREDICTING THE RAPID SYNTHESIS OF DNA<br>FRAGMENTS WITH MACHINE LEARNING.....                   | 51  |
| 4.4 AUTOMATED DESIGN OF THOUSANDS OF HIGHLY NON-REPETITIVE GENETIC PARTS FOR<br>ENGINEERING EVOLUTIONARY ROBUST GENETIC SYSTEMS ..... | 61  |
| 5.0 CONCLUSIONS .....                                                                                                                 | 78  |
| 5.1 A PREDICTIVE MODEL OF TRANSCRIPTIONAL INITIATION .....                                                                            | 78  |
| 5.2 AUTOMATED MODEL TESTING TO QUANTIFY ERROR & INFORMATION CONTENT .....                                                             | 78  |
| 5.3 A PREDICTIVE MODEL OF DNA SYNTHESIS SUCCESS .....                                                                                 | 79  |
| 5.4 AUTOMATED DESIGN OF LARGE TOOLBOXES OF HIGHLY NON-REPETITIVE PARTS .....                                                          | 79  |
| 6.0 REFERENCES .....                                                                                                                  | 80  |
| APPENDIX A: PUBLICATIONS .....                                                                                                        | 87  |

|                                                                       |               |
|-----------------------------------------------------------------------|---------------|
| <b>PUBLISHED PAPERS .....</b>                                         | <b>87</b>     |
| <b>UNPUBLISHED PAPERS.....</b>                                        | <b>88</b>     |
| <br><b>APPENDIX B: ABBREVIATIONS AND GLOSSARY OF TERMINOLOGY.....</b> | <br><b>89</b> |

## List of Figures

|                  |                                                                                   |           |
|------------------|-----------------------------------------------------------------------------------|-----------|
| <b>Figure 1</b>  | <b>Massively Parallel Transcription Rate and Start Site Measurements.</b>         | <b>22</b> |
| <b>Figure 2</b>  | <b>A 3-State Thermodynamic System.</b>                                            | <b>23</b> |
| <b>Figure 3</b>  | <b>Designs and Plasmid Map</b>                                                    | <b>24</b> |
| <b>Figure 4</b>  | <b>Replicate Correlations</b>                                                     | <b>26</b> |
| <b>Figure 5</b>  | <b>Transcription Start Site Profiles</b>                                          | <b>27</b> |
| <b>Figure 6</b>  | <b>Model Development and Validation using Machine Learning.</b>                   | <b>29</b> |
| <b>Figure 7</b>  | <b>Validation of Sequence-to-Function Model on Diverse Promoters.</b>             | <b>32</b> |
| <b>Figure 8</b>  | <b>Model Accuracy on IPTG-inducible Promoters</b>                                 | <b>33</b> |
| <b>Figure 9</b>  | <b>Model Accuracy on Selected Non-Repetitive Promoters.</b>                       | <b>34</b> |
| <b>Figure 10</b> | <b>Overcoming Context Effects to Design and Debug Transcriptional Profiles.</b>   | <b>35</b> |
| <b>Figure 11</b> | <b>The inputs and outputs of the model test system.</b>                           | <b>38</b> |
| <b>Figure 12</b> | <b>Systematic assessment of six sequence-to-function models</b>                   | <b>41</b> |
| <b>Figure 13</b> | <b>Information theory to quantify dataset uncertainty and a model's capacity.</b> | <b>43</b> |
| <b>Figure 14</b> | <b>Identifying the sources of model error.</b>                                    | <b>47</b> |
| <b>Figure 15</b> | <b>The Synthesis Success Calculator</b>                                           | <b>52</b> |
| <b>Figure 16</b> | <b>Feature reduction using the 1076 sequence dataset.</b>                         | <b>54</b> |
| <b>Figure 17</b> | <b>Training and assessment of the Synthesis Success Calculator.</b>               | <b>56</b> |
| <b>Figure 18</b> | <b>Synthesis Success Calculator Examples</b>                                      | <b>57</b> |
| <b>Figure 19</b> | <b>Non-Repetitive Genetic Parts and their Activities</b>                          | <b>62</b> |
| <b>Figure 20</b> | <b>The Non-Repetitive Parts Calculator</b>                                        | <b>64</b> |
| <b>Figure 21</b> | <b>Non-Repetitive Bacterial Promoters</b>                                         | <b>68</b> |
| <b>Figure 22</b> | <b>Design and Characterization of 1678 Non-Repetitive Yeast Promoters</b>         | <b>71</b> |
| <b>Figure 23</b> | <b>Analysis of Non-repetitive Bacterial Promoters</b>                             | <b>73</b> |
| <b>Figure 24</b> | <b>Analysis of Non-repetitive Yeast Promoters</b>                                 | <b>76</b> |

## List of Tables

|                |                                                              |           |
|----------------|--------------------------------------------------------------|-----------|
| <b>Table 1</b> | <b>Primers, RNA Adapter, and DNA Fragment Sequences.</b>     | <b>25</b> |
| <b>Table 2</b> | <b>Promoter Calculator Model Features.</b>                   | <b>28</b> |
| <b>Table 3</b> | <b>Promoter Calculator Model Selection.</b>                  | <b>30</b> |
| <b>Table 4</b> | <b>Promoter Calculator Model Performance Metrics.</b>        | <b>31</b> |
| <b>Table 5</b> | <b>The model test system accuracy and precision metrics.</b> | <b>40</b> |

## 1.0 Summary

### **Predictive Design of Genetic Systems is the Grand Challenge of Synthetic Biology.**

In the Synthetic Biology field, it has been a **Grand Challenge** to rationally engineer large genetic circuits and metabolic pathways *without trial-and-error experimentation*. To overcome this challenge, new approaches to model development are needed to create models that can accurately predict the function of a genetic system (before they are constructed), when utilizing a variety of genetic parts in different environmental & host conditions, using model predictions then to rationally design genetic systems with desired behaviors. Developing accurate models has often been difficult because: (i) there are too many unknown parameters; (ii) parameter values are difficult to precisely measure; (iii) parameter values may depend on the environment/host; (iv) too many internal model states are not observable; (v) and undescribed interactions may confound the model's predictions.

### **Hybrid Biophysics-Machine Learning Models are the Solution to Predictive Design**

We report the development of a new **hybrid machine learning-biophysics approach to model development** that combined our world-leading expertise in the biophysical (BP) modeling of genetic systems with state-of-the-art machine learning (ML) algorithms. We applied this approach to develop several **hybrid BP-ML models**, specifically to:

- 1.0 Predict transcription initiation rates from arbitrary DNA sequences to control mRNA levels
- 2.0 Quantify a model's information content and the sources of model error using a new type of automated model testing framework
- 3.0 Predict when a DNA fragment will be successfully synthesized with a fast turnaround time
- 4.0 Design large toolboxes of highly non-repetitive genetic parts with several useful examples

Altogether, the newly developed predictive models can be combined with pre-existing models to predict gene expression levels from arbitrary DNA sequences. Using this capability, we developed design algorithms able to generate genetic system sequences *de novo* with desired transcription rates, translation rates, and mRNA decay rates. We also include several additional design rules, for example, to ensure that the genetic system may be readily constructed with a fast turnaround time, using standard DNA synthesis and assembly techniques. More information on these advances is below:

## 2.0 Introduction

It has been a grand challenge to transition synthetic biology into a mature engineering discipline, where genetic systems (e.g., biosensors, genetic circuits, and metabolic pathways) are reliably designed, built, and tested to reprogram cellular behavior with desired outcomes. Quantitative models play a central role in synthetic biology's design-build-test-learn cycle by predicting the function of a candidate genetic system, before it is constructed, and therefore reducing trial-and-error experimentation. Gene expression models have been developed to predict the sequence-function relationships for several gene regulatory parts, enabling the automated design of genetic systems with desired protein expression levels. Gene expression predictions have also been combined with system-level models of genetic circuits and metabolic pathways to predict how changes in system architecture, component expression levels, and host genome control organismal sensing, decision-making and the biosynthesis of desired chemicals. Overall, for Synthetic Biology to become a mature engineering discipline, we need accurate gene expression models for each gene expression step.

## 3.0 Methods, Assumptions, and Procedures

### 3.1 Automated Model-Predictive Design of Synthetic Promoters to Control Transcriptional Profiles in Bacteria

#### Library Design

14206  $\sigma^{70}$  promoters were rationally designed to include 2420 UP elements, 4096 -35 hexamers, 229 spacers, 2048 -10 extended motifs, 4096 -10 hexamers, 735 discriminators, and 582 initial transcribed regions (ITR). The 2420 UP element sequences were designed in two subsets: (i) the first set varied adenine and thymine (AT) content from 0-100% within a background sequence containing combinations of consensus, anti-consensus, and mutated -10 and -35 hexamers; and (ii) the second set introduced short cytosine and adenine repeats into the distal and proximal binding sites, while varying AT content within a background sequence containing the same combinations of consensus, anti-consensus, and mutated hexamers. The -10 hexamer set was designed to include the 4096 6-nt sequences in the -10 hexamer, a consensus -35 hexamer, and a 17-bp spacer. The -35 hexamer set was designed to include the 4096 possible 6-nt sequences in the -35 hexamer, a consensus -10 hexamer, and 17-bp spacer. The 229 spacers were designed by varying the spacer length from 1- to 32-bp, in between a consensus -35 and a consensus -10 hexamer, while varying the spacer nucleotide composition at each length. The -10 extended promoter variants were designed to include the 256 possible 4-nt sequence at positions -14 to -17 within 8 different background sequences containing combinations of consensus, anti-consensus, and mutated -10 and -35 hexamer sequences. The 735 discriminator variants were designed in two sets with a background sequence containing a consensus -10 hexamer: (i) a first set that varied the guanine-cytosine (GC) content from 0-100% and the discriminator length from 6 to 8 bp with 5

randomly generated sequences satisfying each criteria; and (ii) a second set that introduced all possible 3-nt sequences in the first half of the discriminator and varied the GC content in the remaining 4-nt. The 582 ITR variants were designed to vary the number of purine bases, the GC content, and the presence of -10 hexamer-like pause sequences within the ITR. Unless otherwise stated, promoter variants were designed within a background containing consensus hexamers and a canonical 17-bp spacer. A complete list of the promoter sequences is available in the Supplementary Data.

## **Oligo Pool Design and Optimization**

An oligopool containing a mixture of 170-nt oligonucleotides was synthesized (Genscript). Each oligonucleotide design contained a promoter variant, four restriction enzyme cut sites, two primer binding sites, and a unique 20 nucleotide barcode sequence (Figure S5A). A custom design algorithm was used to create barcodes and primer binding sites. Barcodes were designed to maximize pair-wise hamming distance between each other and with respect to the unique k-mer list generated from the 14206 designed promoters and the *E. coli* genome. Primer binding site sequences were designed to have target melting temperatures of 55°C, low probabilities of forming primer dimers, and minimal primer structure. Primer binding site sequences were also designed with maximum dissimilarity to barcode sequences, promoter sequences, and the *E. coli* genome sequence.

## **Library Cloning and Growth**

PCR was carried out using the oligopool as DNA template to create an insertion cassette library, using Q5 High-Fidelity DNA Polymerase (NEB), 20 cycles, and primers 1 and 2. DNA was extracted and purified via gel electrophoresis. The purified insertion cassette library and plasmid pFTV1 were double-digested using BamHI and HindIII (NEB), purified, ligated, purified again, and then transformed into NEB 5-alpha Electrocompetent *E. coli* (NEB). All ligations used T7 DNA ligase (NEB) unless otherwise stated. Transformation recovery was 60 min at 37°C using 1 mL of pre-warmed SOC media. Transformation efficiency was  $1.2 \times 10^7$  CFU per mL recovery broth. The transformed cell library was used to inoculate duplicate cultures using 5 mL LB media supplemented with 50 ug/mL chloramphenicol. Shaking cultures were incubated overnight at 37°C. One culture was used for cryostocking. Plasmid purification was performed on the other culture using an EZNA plasmid mini kit (Zymo). The plasmid library was sequentially digested with SpeI and EcoRI, followed by rSAP treatment. Digested plasmid pool was then purified via gel extraction.

To generate DNA templates for carrying out in vitro transcription and TSS mapping, the digested plasmid pool was ligated to a SpeI/EcoRI-digested DNA fragment containing a ribosome binding site (RBS) and mRFP1 coding sequence, followed by purification and transformation into NEB 5-alpha Electrocompetent *E. coli* cells. The expression plasmid library was then harvested from overnight shaking cultures grown in LB media supplemented with 50 ug/mL chloramphenicol. For quantification of library coverage, PCR was carried out on the plasmid library to generate promoter-barcode amplicons,

followed by next-generation sequencing (Illumina MiSeq), barcode mapping, and promoter counting. 1ug aliquots of the expression plasmid library were used as DNA template for the in vitro transcription reactions.

### ***in vitro* Transcription Rate Measurements**

Replicate in vitro transcription reactions were carried out for 3 hours at 37°C, combining 1 ug of the expression plasmid library, 0.5 mM of each NTP, 1X E. coli RNA Polymerase Reaction Buffer, and 1 uL of E. coli RNA Polymerase Holoenzyme (NEB). DNA was removed by adding 2.5 units of TURBO DNase and incubating at 37°C for 30 min. The RNA product was purified using a RNA Clean & Concentrator Kit (Zymo).

For quantification of RNA variant levels after transcription (RNA-Seq), cDNA first-strand synthesis reactions were first carried out on the harvested RNA, using 200 units of SuperScript IV (Invitrogen), 1X buffer, and transcript-specific primer . PCR was carried out to generate barcode-containing amplicons (121 bp), using 5 ng of cDNA as template, primers 2 and 3, and 25 cycles of amplification, followed by next-generation sequencing (Illumina HiSeq 2500, 150 bp paired-end), barcode mapping, and barcode counting. For quantification of DNA variant levels in the expression plasmid library (DNA-Seq), PCR was first carried out on 10 ng of the expression plasmid library to generate promoter-barcode amplicons (847 bp long), using primers 1 and 2 and 25 cycles of amplification. The promoter-barcode regions were then condensed into shorter amplicons (271 bp) by treating the PCR product with T4 Polynucleotide Kinase (NEB), carrying out DNA circularization by ligation using T4 DNA Ligase (NEB), purifying the monomer DNA circles via gel extraction, and then PCR amplifying using primers 3 and 4 with 25 cycles. The promoter-barcode amplicon library was sequenced using Illumina HiSeq 2500 (150 bp, paired-end), followed by barcode mapping, promoter identification, and counting. Three independent in vitro transcription reactions, RNA-Seq, and DNA-Seq assays were carried out.

### **Transcriptional Start Site Mapping**

Starting with RNA harvested from the in vitro transcription reactions, the locations of transcriptional start sites (TSSs) were determined by treating the RNA with TURBO DNase to remove template DNA, purifying, treating with RNA 5' polyphosphatase (Epicentre) to remove 5' phosphates, purifying, ligating the RNA to an 5' RNA adapter using T4 RNA Ligase (NEB), and purifying again. The treated RNA product was then converted to cDNA by treating with SuperScript IV (Invitrogen) and the transcript-specific primer 2. PCR was then carried out using 10ng of cDNA as template, primers 2 and 5, and 30 cycles of amplification. The linear DNA product was then circularized by first treating it with T4 Polynucleotide Kinase, followed by ligating using T4 DNA Ligase. Monomer DNA circles were gel extracted. PCR was then carried out to generate the barcode-TSS amplicon library, using primers 3 and 6 in 25 cycles of amplification. The barcode-TSS amplicon library was then sequenced using Illumina HiSeq 2500 (150 bp, paired-end).

## Analysis of Next-generation Sequencing Data

Barcode mapping, barcode counting, and promoter identification were carried out using custom software developed to analyze data from massively parallel reporter assays. The abundances of the cDNA-derived RNA transcript variants were quantified by counting the numbers of unique barcodes that were within 1 edit distance of the expected barcode sequences. The numbers of each DNA variant within the expression plasmid library were determined by first identifying associated barcode sequences that perfectly matched the expected barcode sequences, extracting the promoter sequence region, and mapping it to the expected promoter sequence according to barcode association. DNA counts were included if the promoter sequence perfectly mapped to the expected sequence. The RNA transcript counts and DNA counts were used to determine the relative transcription initiation rate of each promoter variant  $i$  in replicate  $j$  according to the formula:

$$r_{i,j} = \frac{RNA_{i,j}}{DNA_{i,j}} \times \frac{\sum_{i=0}^n DNA_{i,j}}{\sum_{i=0}^n RNA_{i,j}} \quad (S1)$$

where  $RNA_{i,j}$  is the cDNA-derived barcode count for variant  $i$  in replicate  $j$ ,  $DNA_{i,j}$  is the DNA count containing promoter variant  $i$  in replicate  $j$ , and  $n$  is the total number of variants in replicate  $j$ . Transcriptional start site locations were determined using the TSS-barcode reads. For each read, the barcode was mapped and used to identify the associated reference promoter. The TSS location was then mapped by first identifying the position of a known constant flanking sequence, extracting the adjacent TSS-containing sequence, and carrying out a Smith-Waterman alignment using the associated promoter sequence as a reference. TSS locations were only counted when the TSS-containing sequence contained only one start location that perfectly matched the expected sequence.

## Data Filtering

For model training, the set of promoter variants with a single, predominant transcription start site was defined by the intersection of three criteria: (i) promoter variants must have with at least 50 RNA-Seq and 50 DNA-Seq counts across all triplicate measurements; (ii) promoter variants must have one predominant TSS with at least twice the counts as the next most predominant TSS; and (iii) promoter variants must have a predominant TSS within a 10-bp window surrounding the anticipated TSS location. Data filtering yielded 5193 promoter variants that satisfied these criteria. For model testing, the set of promoter variants with high-precision transcription rate measurements was defined using a single criterion: all promoter variants must have coefficients of variation (CVs) that are less than 0.40. CVs are calculated by determining the standard deviation of the transcription rate measurements and dividing by the mean of the transcription rate measurements across the three independent measurements. Data filtering yielded 5388 promoter variants that satisfied this criterion.

## A Biophysical Model of Bacterial Transcriptional Initiation

We describe the process of prokaryotic transcriptional initiation at a single promoter as a 3-state system. In state 0, RNAP/ $\sigma$  is not bound to the promoter DNA (free state). In state 1, RNAP/ $\sigma$  is bound to the promoter DNA, creating a complex in the closed conformational state. In state 2, RNAP/ $\sigma$  is bound to the promoter DNA with a stable transcriptional bubble, which includes the transition from the closed-to-open conformation, the initial melting of promoter DNA to create an unstable transcriptional bubble, and the formation of a stable transcriptional bubble (R-loop) via initial transcription of the ITR region. The transitions and transition rates in this system include: [1] the association rate of RNAP/ $\sigma$  recruitment to promoter DNA, transitioning the system from state 0 to 1 ( $r_1$ ); [2] the disassociation rate of RNAP/ $\sigma$  after it's bound, transitioning the system from state 1 to 0 ( $r_{-1}$ ); [3] the rate of the closed-to-open conformational change, DNA melting, and stable R-loop formation, transitioning the system from state 1 to 2 ( $r_2$ ); [4] the rate of transcriptional bubble collapse and abortive initiation, transitioning the system from state 2 to 1 ( $r_{-2}$ ); and [5] the rate of productive transcriptional initiation, transitioning the system from state 2 to 0, whereby RNAP/ $\sigma$  transitions to the transcriptional elongation process, escapes from the promoter DNA, and frees up the promoter DNA to be available for binding to another copy of RNAP/ $\sigma$  ( $r_3$ ).

This coarse-grained representation of the transcription initiation process purposefully excludes several intermediate states and internal transitions. As discussed below, if there is no accumulation of internal states during transcriptional initiation (e.g. via significant pausing), then the addition of more internal states to the model is not beneficial and does not alter the free energy model formulation or coefficient values.

A chemical master equation (CME) can be formulated to describe the time-evolution of the probabilities of each state, considering the transition rates between each state. The CME for this system is:

$$\begin{aligned}\frac{dP_0}{dt} &= -r_1P_0 + r_{-1}P_1 + r_3P_2 \\ \frac{dP_1}{dt} &= r_1P_0 - (r_{-1} + r_2)P_1 + r_{-2}P_2 \\ \frac{dP_2}{dt} &= r_2P_1 - (r_{-2} + r_3)P_2\end{aligned}$$

We also designate that the transcription initiation rate ( $TX_{\text{actual}}$ ) for a promoter variant will be the rate of promoter escape ( $r_3$ ) multiplied by the probability of state 2 ( $P_2$ ). Then, by assuming that the system has reached a “dynamic” steady-state where the probabilities of each state are not changing in time, we can set the derivatives in the CME to zero, solve for the probability of state 2 ( $P_2$ ) in terms of the transition rates and the probability of state 0 ( $P_0$ ), and then multiply by the transition rate  $r_3$  to yield the transcription initiation rate  $TX_{\text{actual}}$ :

$$TX_{actual} = \frac{r_3 r_1 r_2 P_0}{r_{-1} r_{-2}}$$

From transition state theory, we then recognize that the ratio containing the sequence-dependent transition rates can be reformulated in terms of a Gibbs free energy change between states 0 and 2, which we call  $\Delta G_{total}$ , as part of Boltzmann's relationship. Here, Boltzmann's constant  $b$  is a thermodynamic parameter that converts free energy differences into changes in state probabilities. In ideal (neutral, dilute) systems,  $b$  is  $1/RT$  ( $R$  is the gas constant,  $T$  is temperature), but the value of  $b$  is not  $1/RT$  in complex (charged, non-dilute) systems. We also recognize that our transcription rate measurements ( $TX_{measured}$ ) only provide a proportional measurement of the actual mRNA concentrations inside the system. We therefore create a proportionality constant  $K$  to denote this. We then observe that the steady-state probability of state 0 ( $P_0$ ) is indistinguishable from this proportionality constant. We therefore lump them together to yield the simple equation:

$$TX_{measured} = K \frac{r_3 r_1 r_2 P_0}{r_{-1} r_{-2}} = K e^{-\beta \Delta G_{total}} \quad (S2)$$

We then formulate a free energy model to calculate  $\Delta G_{total}$  from sequence information. Our model considers each sequence-dependent interaction and then sums them together as linear, additive contributions. The free energy model is:

$$\Delta G_{total} = \Delta G_{-10} + \Delta G_{-35} + \Delta G_{spacer} + \Delta G_{UP} + \Delta G_{disc} + \Delta G_{ITR} \quad (S3)$$

where  $\Delta G_{-10}$  is the contribution from the -10 hexamer sequence,  $\Delta G_{-35}$  is the contribution from the -35 hexamer sequence,  $\Delta G_{spacer}$  is the contribution from the DNA spacer's length and rigidity,  $\Delta G_{disc}$  is the contribution from the discriminator sequence,  $\Delta G_{ITR}$  is the contribution from the ITR sequence, and  $\Delta G_{UP}$  is the contribution from the UP sequence.

We do not know the proportionality constant  $K$ , but by comparing the measured TX rates of the promoter variants to a designated reference promoter, we can remove the proportionality constant from the model training and prediction. To do this, we evaluate Equation S2 for a sample promoter variant (with measurement  $TX_{measured}$  and prediction  $\Delta G_{total}$ ) and the reference promoter (with measurement  $TX_{measured,ref}$  and designated  $\Delta G_{total,ref}$ ) and then subtract their left-hand-sides and right-hand-sides, yielding the equation:

$$\log \left( \frac{TX_{measured}}{TX_{measured,ref}} \right) = -\beta (\Delta G_{total} - \Delta G_{total,ref}) \quad (S4)$$

The choice of a reference promoter is somewhat arbitrary, but it should have a well-measured transcription rate. Here, we selected a reference promoter with a measured transcription rate closest to the log-mean-center of the dataset. The model itself is trained to predict the differences in  $\Delta G_{total}$  between the promoter variants' and the reference

promoter ( $\Delta G_{\text{total}} - \Delta G_{\text{total,ref}}$ ), which leads to an interaction energy scale that spans both negative and positive values. In general, interaction energies have negative values when they are stronger than the average interaction strength. Interaction energies have positive values when they are weaker than the average interaction strength.

Next, we consider the assumptions made during model formulation and the experimental design aspects that ensure their validity. Assumption validity exists on a continuum and our objective here is to identify the best experimental conditions to maximize their validity. First, there is a pool of available RNAP/ $\sigma$  in the system that competitively binds to many DNA sites, including all promoter variants. This assumption is most valid when carrying out transcription measurements on many designed promoter variants (as in our study) or inside cellular conditions where many promoters co-exist, but would be least true if carrying out transcription rate measurements on a single promoter variant. In our in vitro transcription assay, the amounts of RNAP/ $\sigma$  and total DNA are also kept constant throughout the measurements.

Second, the amount of RNAP/ $\sigma$  in the system is small enough, as compared to the amount of available DNA, that the promoter variants are not “saturated” and thus state 0 (the unbound state) is the predominant state. This assumption is most valid in our in vitro transcription assays and within typical in vivo (cellular) conditions, where the total amount of RNAP/ $\sigma$  is much smaller than the total amount of DNA. However, it would be least valid if excess amounts of RNAP/ $\sigma$  were added to the in vitro transcription assays.

Third, we assume that transcriptional pausing does not occur in appreciable amounts across the promoter variants, which would lead to accumulation of state 1 and a reduction in state 0. The presence of pause sequences could make this assumption less true; however, we did not find any motif sequences in the discriminator or ITR regions whose inhibitory effect on TX rate could not be explained by other factors (e.g. R-loop thermodynamic stability).

Fourth, the measured transcription rate of each promoter variant is proportional to the promoter escape rate (transition rate  $r_3$ ) multiplied by the probability of state 2 (stable promoter state), which is  $r_3 P_2$ . We observe successful transcription when RNAP/ $\sigma$  transcribes each promoter variant’s unique barcode sequence, which is located in the 3’ UTR. By placing the barcode sequence in the 3’ UTR and not nearby the promoter, we remove the potential for barcode sequence-dependent changes in promoter escape that would make this assumption less true. We also utilize a short gene (mRFP1) to prevent appreciable amounts of RNAP/ $\sigma$  fall-off during transcription.

## **Model Training using Machine Learning**

Machine learning was used to train, test, and validate several linear models (Ridge Regression, LASSO, and ElasticNet) and parameterize the unknown interaction energies. To do this, we first enumerated a list of sequence motifs and biophysical characteristics that have the potential to contribute to the energetics of transcriptional initiation as quantified in our free energy model. There are 472 features in the exhaustive unpruned

list. We divided our filtered dataset (the set of promoter variants with a single predominant TSS, see the Data Filtering section above) into a training set (4673 promoters, 90%) and an unseen test set (520 promoters, 10%). Transcription initiation rate measurements were log-transformed and converted to Z-scores, using the mean and standard deviation of the log-transformed transcription rates. We discuss the relationship between Z-score normalization and the biophysical modeling in a section below.

We then carried out model training and hyperparameter optimization (alpha, 1 parameter) using 10-fold cross-validation on the training set. The best models were then evaluated on the unseen test set. Accuracy metrics included the mean absolute error (MAE), mean squared error (MSE), and the squared Pearson correlation ( $R^2$ ). Feature importance was then carried out using one-feature drop analysis. Model training and testing was repeated several times, each time dropping individual features or sets of related features. The differences in model accuracy were calculated and features were pruned if they did not appreciably increase model accuracy. Feature drop analysis identified several features whose values were highly correlated with other features in the dataset, for example, AT content in the UP region and the minor groove width in the UP region. The final set of pruned features was identified by including all features that yielded appreciable improvements in model accuracy. When faced with equivalent feature set choices, we selected the smaller set of features with fewer unknown coefficients, which often relied on biophysical calculations. We also compared each model's trained interaction energies against the canonical interactions known to have the most effect on transcription initiation rate, which we refer to as positive controls, and selected models that achieved 100% of these positive controls. Notably, the differences in interaction energies for the highest performing models were small, but the small differences could affect the rank-order of the top/bottom 10 sequence motifs, which affected their positive control successes. We show the accuracies of the top performing linear models using either the full or pruned feature sets. We selected the Ridge Regression model for use in this study, though optimized versions of each linear model type yielded high accuracies on the unseen test set.

The final set of pruned features included: (i) The energetic contributions from binding to the -35 and -10 hexamer were represented using one-hot encoding of the presence of each possible 3-nt motif within each hexamer. All possible 3-nt motif sequences (64 3-mers) are included and their presence or absence is a categorical feature, totaling 256 features. (ii) The energetic contributions from the spacer region were decomposed into two sets of features, including the spacer length (15 to 20 bp, each represented as a categorical feature) and the spacer's DNA rigidity (a single numerical feature calculated based on the sequence-dependent persistence length of double-stranded DNA) [1]. (iii) The energetic contributions from binding to the -10 extended motif were represented using one-hot encoding of the presence of each possible 2-nt motif located upstream of the -10 hexamer, totaling 16 categorical features. (iv) The energetic contributions from the discriminator region were represented using one-hot encoding of each possible 3-nt motif in the first 3 nucleotides of the discriminator region, totaling 64 categorical features. (v) The energetic contributions from the UP region were represented using the numerical value of the calculated minor groove width in the distal and proximal UP sites (each 10 bp long) [2]. (vi) The energetic contributions from the ITR were represented by calculating

the thermodynamic stability of the R-loop within the first 15 nucleotides of the ITR region (1 numerical feature), which is the free energy of the duplexed DNA-DNA complex subtracted by the free energy of the duplexed RNA-DNA complex. Numerical feature values are divided by the maximum possible value for normalization.

Additionally, during the initial model training, feature encoding relied on the positional identification of sequence motifs using the following procedure. The predominant transcriptional start site was determined from TSS Mapping measurements. The promoter variant sequence was then scanned, taking into account that discriminator lengths varied from 5 to 10 nt and spacer lengths varied from 15 to 20 nt. The positions of the -10 and -35 hexamer motifs were then identified by scanning across the sequence, evaluating a position weight matrix (PWM) tailored for each motif, and calculating a motif score. The positions with the highest motif score were labeled as the -10 or -35 hexamer. If two or more locations had equivalent motif scores, the motif location with the most optimal spacer length (17 nt) was selected. This procedure was tailored to function alongside our experimental promoter library design, which included the 4096 possible -35 hexamer motif sequences alongside a consensus -10 hexamer as well as the 4096 possible -10 hexamer motif sequences alongside a consensus -35 hexamer. Even though these designs used a canonical spacer length of 17 nt, an alternate procedure that assumed that the spacer length was always a constant yielded scenarios where sequence changes led to shifts in the locations of the actual -10 or -35 motif and corresponding changes in motif sequence. Failing to account for these shifts led to mismatched interaction energies during model training, for example, non-canonical motifs having interaction energies similar to canonical motifs, created by the formation of a canonical motif with a shifted location.

## The Relationship between Biophysical Modeling and Z-score Normalization

To create our training and test datasets, we log-transformed the transcription rate measurements and applied Z-score normalization on the outcomes. Z-score normalization is a routinely used approach for improving the machine learning training process, whereby the outcomes in a dataset are transformed and re-scaled. The Z-score is:

$$Z = \frac{(X - \mu)}{\sigma}$$

where X is the observed outcome,  $\mu$  is the arithmetic mean of the observed outcomes, and  $\sigma$  is the standard deviation of the observed outcomes. Here, we also use a statistical thermodynamic model to predict the transcription initiation rate of a DNA sequence, according to:

$$\log\left(\frac{TX}{TX_{ref}}\right) = -\beta(\Delta G_{total} - \Delta G_{total,ref})$$

We show that these two relationships are directly related to one another, enabling us to

determine the apparent thermodynamic parameter  $b$ , which is a model constant that converts free energies into state probabilities.

Specifically, we first log-transform the transcription rate measurements such that  $X$  is  $\log(TX)$  for a particular promoter variant,  $m$  is the mean of the log-transformed measurements, and  $s$  is the standard deviation of the log-transformed measurements. Within the dataset, we selected the reference promoter such that its log-transformed measured transcription rate was very close to the mean of the log-transformed measurements ( $m$ ). Therefore, the  $X - m$  in Z-score normalization is equivalent to  $\log\left(\frac{TX}{TX_{ref}}\right)$ . We then use this choice to equate the Z-score and biophysical model equations, such that

$$\sigma Z = (X - \mu) = \log\left(\frac{TX}{TX_{ref}}\right) = -\beta(\Delta G_{total} - \Delta G_{total,ref})$$

Here, we see that the standard deviation  $s$  of the log-transformed measurements is equivalent to the negatively valued thermodynamic beta ( $-b$ ) and that the Z-score used to train the model is equivalent to the apparent difference in free energies ( $\Delta G_{total} - \Delta G_{total,ref}$ ). In other words, the standard Z-score transformation procedure for model training is equivalent to the use of state referencing in thermodynamic modeling of complex systems.

A key effect of the Z-score normalization procedure is to collapse the range of the actual and learned outcomes to aid in the model training process. This has the effect of collapsing the maximum magnitude of the interaction energies. However, we can revert the learned interaction energies back to their original scale by evaluating the biophysical model equation, which uses the learned thermodynamic beta. Notably, the values are similar across all the in vivo datasets, but somewhat distinct for the in vitro dataset. There are physiochemical reasons for this difference, likely related to composition and molecular crowding differences between the in vivo and in vitro systems.

## Individual Promoter Construction and Isogenic Characterization

Individual promoters were constructed to test the effects of changing UP and ITR regions on transcription rates and to test the automated design of promoters using model predictions. Promoters were constructed by synthesizing pairs of overlapping oligonucleotides and using PCR assembly to create insertion DNA cassettes. Cassettes and the mRFP1-expression plasmid pFTV1 were double-digested, purified via gel extraction, ligated, and transformed. Plasmid products were sequence verified using Sanger sequencing. Isogenic E. coli DH10B cells were transformed with each plasmid, following by quantification of their mRFP1 expression levels during exponential growth, using spectrophotometry and flow cytometry measurements. Cells were grown overnight in LB media supplemented with 50  $\mu$ g/ml chloramphenicol, followed by a 1:100 dilution into 200  $\mu$ l M9 minimal media supplemented with 50  $\mu$ g/ml chloramphenicol within 96-well optical bottom microtiter plates. OD600 and mRFP1 fluorescence measurements

were recorded every 10 minutes until cultures reached an OD600 of 0.20. Cultures were then serially diluted 1:10 into pre-warmed supplemented M9 media and grown again until they reached an OD600 of 0.20. 10 ul aliquots were then extracted and added to 190 ul PBS with 2mg/mL kanamycin to stop protein production. Flow cytometry was then carried out on the fixed cells to record their fluorescence distribution (BD LSR Fortessa). The mRFP1 fluorescence level was the arithmetic mean of the measured fluorescence distribution subtracted by autofluorescence, which was the arithmetic mean of the fluorescence distribution of wild-type E. coli DH10B cells.

## **Automated Promoter Sequence Design**

We used an optimization algorithm (simulated annealing) to automatically design 16 promoter sequences with systematically varied transcription initiation rates, using the statistical thermodynamic model prediction within the optimization algorithm's objective function. Designed promoter sequences were allowed to have variable TSS positions and use the surrounding upstream/downstream DNA sequence as part of the promoter. The surrounding upstream/downstream DNA sequences were kept constant for all promoter designs. The average promoter length was 77 bp and the average pairwise Hamming distance (the number of mismatched bp) across the 16 promoters was 57 bp, yielding an average sequence similarity of only 25.9%.

## **Identification of Cryptic Promoters for Genetic Circuit Debugging**

The statistical thermodynamic model was used to predict the transcription initiation rates across the 11-promoter genetic circuit (6793 bp), as studied and characterized in Borujeni et al. [3], and identify desired and undesired/cryptic transcriptional start sites (TSSs) on both its sense and anti-sense DNA strands. We labeled a location as a TSS when its predicted transcription initiation rate was at least 3-fold higher than the average transcription initiation rate per each strand. We used RNA-Seq measurements and RNAP flux inferences from Borujeni et al. [3] to assess the accuracy of these TSS predictions. To do this, we counted the number of actual TSS peaks that were within 10 bp of the model-predicted TSSs and divided by the total number of model-predicted TSSs. This window threshold is reasonable considering that RNA-Seq measurements using 150 bp reads have limited positional precision and the RNAP flux inference calculation was shifted by up to 20 bp for known promoters with predominant TSSs.

**Data and materials availability:** A Python source code implementation of the statistical thermodynamic model of transcriptional initiation is available at <https://github.com/hsalis/SalisLabCode>. Next-generation sequencing read data files in fastQ format are available at NCBI with accession identifier PRJNA754118.

## 3.2 An automated model test system for systematic development and improvement of gene expression models

### Compiling the genetic system database

To create the database, we compiled sequences and expression measurements from the supplemental data of several publications. We included detailed information for all sequences including the position of the start codon, bacteria host, experimental conditions, and characterization method and data. For the 1014IC dataset, mean fluorescence (or luminescence) levels, standard deviation across at least three replicates, and RT-qPCR measurements (when available) were included. For the 8848FS dataset, raw data includes binned read counts, RNA read counts, and DNA read counts for two replicates. The reconstructed relative fluorescence values (protein levels) and the mRNA levels were recalculated as described in the original paper [4]. Apparent translation rates for the FlowSeq datasets are calculated as the protein level divided by the mRNA level.

DNA sequences for the genetic systems were either provided in the supplementary information of the publications, acquired directly from the original authors, or recreated manually from information found in the publication [5]. 5'-untranslated regions (UTRs) of the mRNAs were determined by identifying the most common transcription start site (TSS) of the promoter of the genetic system. Each TSS was either estimated by selected the first nucleotide following the annotated promoter, or in the case of any promoters characterized by the 8848FS dataset, the TSS is defined as the most frequent upstream position of transcription initiation as identified by RNA read counts. Start codons specified by the original publications are saved in the database.

Data was first stored in a pandas dataframe with null values for entries where information or characterization data was not provided. The supporting model test system code includes functions to allow easy manipulation of the dataframe, such as extracting and analyzing genetic systems that share common properties (see SynBioMTS/dbms.py).

### Filtering the FlowSeq datasets

We applied a combination of filters on the FlowSeq dataset to remove characterized sequences with low read counts or skewed binned read counts. Following the rules filters in Kosuri *et al.* [4], we removed sequences with insufficient protein data, or if the reconstructed fluorescence level was above or below the estimated measurement range. The thresholds were defined as two-fold the minimum protein level (1357 reconstructed RFU) and 99% of the maximum protein level (204,060 reconstructed RFU). Characterized sequences that had fewer than 10 DNA read counts (both replicates), fewer than 50 DNA read counts (either replicate), or fewer than 20 RNA read counts (either replicate) were excluded. Lastly, promoters that had transcription start sites that started after the barcode sequence were excluded. These filters removed 3319 sequences (26.2%), resulting in the 8848FS dataset.

## Calculating total sequence entropy

We calculated the sequence diversity of a dataset ( $H_{seq}$ ) by determining the sum of the Shannon entropy across all nucleotide positions from the transcriptional start site to 35 nucleotides after the start codon. Considering that sequences have different length 5' UTRs, we padded the 5' ends of all sequences with "X" characters until all sequences have the same length. We then calculated Shannon entropies using a 5-letter alphabet with a maximum entropy,  $\log_2(5) = 2.32$  bits, at each position.

$$H_{seq} = - \sum_j \sum_{i=1}^5 p_{ij} \log_2 p_{ij}$$

## Calculating Kullback-Leibler divergence

We calculated the model's Kullback-Leibler (KL) divergence, which is the information gain achieved if the model of interest ( $P$ ) is used instead of a random model ( $Q$ ). We defined a random model ( $Q$ ) as having a uniform error distribution across the observed range. We then calculate the relative probability of model error ( $P$ ) across 100 log-spaced bins from  $10^{-3}$  to  $10^3$ . The  $D_{KL}$  value is computed using the equation below, removing bins where  $P(i)$  is zero. The normalized  $D_{KL}$  value is divided by the maximum possible  $D_{KL}$  for a perfect model, which is 4.61 here.

$$D_{KL}(P||Q) = \sum_i P(i) \log \left( \frac{P(i)}{Q(i)} \right)$$

## Calculating area under the ROC curve (AUROC)

Measured protein expression levels are partitioned into HIGH and LOW binary outcomes according to a selected cutoff value. The test metric is the predicted translation initiation rate, using the same cutoff to determine LOW and HIGH predicted outcomes. To generate a Receiver Operating Characteristic curve, the cutoff is then systematically varied between the minimum and maximum of the measured protein expression levels, and the True Positive Rates (TPR) and False Positive Rates (FPR) are calculated for each cutoff value. The area under the ROC curve (AUROC) is then calculated using the trapezoidal rule (see `calc_AUROC` method).

## Using the gene expression models

All models were used with settings to maximize the predictive accuracy of each model, to best assess the proposed determinants. For all versions of RBS Calculator, the nine, 3'-most nucleotides of the 16S rRNA of each species was used as the aSD sequence, the temperature was set at the experimental temperature, a fixed mRNA post-cutoff of 35 nt was used, and all other parameters were left as default. To our knowledge, no source code is available for UTR Designer, so we recreated the model by modifying RBS

Calculator v1.0 with the described changes [6] and ran the model with all the same parameters as RBS Calculator v1.0. For all versions of RBS Calculator and UTR Designer, we selected the most highly translated, in-frame start codon as the start site to predict the translation rate. RBS Designer was released as a Windows installer package. We wrote a Python wrapper to run the batch version of the program on the full database (see RBSBatch.py in the repository). All default settings were used and the correct rRNA sequences were used for RBS Designer. Noncanonical start codons (e.g. “CUG”) were skipped because RBS Designer only supports prediction for genes with an AUG start. For EMOPEC, we used the predicted Expression value for comparisons, rather than the Expression<sub>percent</sub> value as described in Bonde *et al.* [5]. All default settings were used for EMOPEC. All model output was saved to the database for future study.

### 3.3 The Synthesis Success Calculator: Predicting the Rapid Synthesis of DNA Fragments with Machine Learning

#### Software Implementation

The Synthesis Success Calculator was developed in Python 2.7.15 using the Anaconda scientific computing stack [7]. DNA folding predictions were performed using Vienna RNA 2.4.1[8]. Random forests were produced using scikit-learn [9]. Forest analysis and interpretation was done using the treeintepreter [10] python package and functions from the rfpimp package (<https://github.com/parrt/random-forest-importances>). External sequences were read and partitioned using Biopython [11]. An implementation of the analysis pipeline used here can be found on GitHub (<https://github.com/hsalis/SalisLabCode>). The latest version of the Synthesis Success Calculator can be accessed at <https://salislab.net/software>.

#### Dataset Assembly

DNA fragment sequences were sourced from our labs order history over the past 8 years, a recent genome refactoring effort for *S. aureus* [12], and a recent pressure test to design, build and test metabolic pathways within 90 days [13]. The two external sources were selected due to their explicit, unambiguous discussion of synthesis delays and redesigns.

As the negative controls, DNA fragment sequences were systematically designed to contain DNA properties known to be associated with synthesis failure. To do this, fragment sequences were initialized as non-repetitive, randomized sequences with total GC contents within the acceptable range, followed by purposeful introduction of specific sequence determinants with varying quantities. These sequences were fed to pre-existing filters hosted by contemporary commercial service providers and selected for inclusion in the negative control set if service providers would not attempt to synthesize it. For the positive controls, we extracted subsequences from readily synthesized DNA fragments

with a varied selection of distinct DNA properties, and confirmed that these subsequences also passed pre-existing filters provided by commercial service providers.

## Scoring Metrics

To score the performance of the classifier, we used the geometric mean of the  $F_1$  score, Matthew's Correlation Coefficient (MCC), and the Cohen's Kappa (CK). The  $F_1$  score measures the harmonic mean of precision and recall, and is defined as

$$F1 = \frac{2TP}{2TP + FP + FN}$$

where TP is the true positive rate, FP is the false positive rate, and FN is the false negative rate.

Matthew's Correlation Coefficient measures the classifier performance, but does so while better accounting for differences in class sizes and is defined as

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where TN is the true negative rate.

Cohen's Kappa is a scoring metric that accounts for randomness in the classification by measuring the inter-categorical rating reliability across binary outcomes. It is defined as

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

where observed agreement  $p_o$  is defined as:

$$p_o = \frac{TP + TN}{TP + TN + FP + FN}$$

and expected agreement  $p_e$  is defined as:

$$p_e = \frac{(TP + FN) * (TP + FP) + (TN + FP) * (TN + FN)}{(TP + TN + FP + FN)^2}$$

## Model Hyperparameter Optimization

Before training the random forest classifier, we first performed a 3-round iterative hyperparameter optimization, using the full 1076 sequence dataset to find the best hyperparameters to maximize the generalizability of the classifier. We performed iterative 10-fold cross-validation searches for a given set of candidate hyper parameters, using

the geometric mean of the  $F_1$  score, Matthews Correlation coefficient and Cohen's Kappa as a lumped scoring metric, and took the set of hyperparameters that yielded the best out-of-bag accuracy score from the best performing model of each search.

The first hyperparameter search was 100 iterations of a 200-iteration randomized-search with 10-fold cross-validation, the second was 100 iterations of grid-search with 10-fold cross-validation, and the third search was 200 iterations of grid-search with 10-fold cross-validation, with each subsequent round shrinking the parameter space. During the first round, we varied the number of features seen by each decision tree (`max_features`), the number of trees in the forest (`n_estimators`), the number of samples a branch node needs to see in order to split into subsequent nodes (`min_samples_split`), the number of samples a leaf node needs to receive (`min_samples_leaf`), and the weight for each class relative to their representation in the dataset (`class_weight`). Subsequent rounds only varied `n_estimators`, `min_samples_split`, and `min_samples_leaf`.

## **Model Feature Reduction**

Feature reduction was performed using a 100-iteration drop-column feature importance selection strategy. After one round of hyperparameterization on the full dataset, we split the dataset into an 80-20 train-test split, then scored the model fit for 100 random forests using the set hyperparameters. We then removed one of the features from the dataset, refit the same 100 random forests to the reduced dataset, and calculated their performance scores. We then determined the difference in performance score between using all features versus the reduced 1-feature dropped set. These calculations were repeated for every feature. If the performance score improved when a feature was absent (a negative mean loss), then the feature was removed from the feature set. After feature reduction, 29 DNA properties were removed from the feature set, leaving only 9 DNA properties retained for downstream training of the random forest classifier.

## **Model Training and Validation**

After hyperparameterization and feature reduction, we trained the random forest using a 575-250-251 train-validate-test split for 40,000 uniquely seeded forests. Class fractions for each sub-dataset were maintained throughout the splits. After the 40,000 iterations, we assessed the performance of the forests on the validation sequences, according to the geometric mean of the  $F_1$  score, Matthew's Correlation Coefficient and Cohen's Kappa, selecting the forest with the highest performance. We then examined the predicted class probability of sequences in the training and validation sets, relative to their known class. Misclassified sequences with extreme outliers in their predicted class properties (greater than 75% for failures, and less than 25% for successes) were added to the training set to create a rebalanced, retraining set. We then carried out additional training of the random forest classifier using the retraining set, containing these oversampled sequences.

## **Sequence Determinant Analysis**

Sequence determinants that contributed to predicted synthesis failures were extracted using treeinterpreter. The treeinterpreter package calculates the probability prediction of a given tree as

$$T(x) = c_{full} + \sum_{k=1}^K contrib(x, k)$$

where  $c_{full}$  is the default value (bias) of the root node of the tree,  $K$  is the number of features, and  $contrib(x, k)$  is the contribution of the  $k^{th}$  feature in feature vector  $x$  along the decision path leading to the leaf node of  $x$ . When applied across an entire forest, the probability is stated as

$$P(x) = \frac{1}{J} \sum_{j=1}^J c_{jfull} + \sum_{k=1}^K \frac{1}{J} \sum_{j=1}^J contrib(x, k)$$

where  $J$  is the number of trees in the forest. Using these equations, we quantify the contribution of each sequence determinant to the predicted synthesis outcome.

## Sequence Optimization

To perform targeted recoding on *E. coli* protein coding sequences, we first used treeinterpreter to extract the 3 largest contributing features to the prediction that the sequences were not compatible with rapid synthesis. We then highlighted the nucleotides that contributed to these important features, and allocated different amounts of codon recoding (5%, 10%, 15%, and 20% of the codons) to those highlighted nucleotides. We used a codon table based on the synonymous codon usage of *E. coli* proteins with a high predicted translation initiation rate to recode the highlighted nucleotide regions. After recoding, the coding sequences were reassessed by the classifier to see how well the recoding addressed the predicted synthesis probabilities and whether any of the previously recalcitrant sequences were now predicted to be readily synthesized.

## 3.4 Automated Design of Thousands of Highly Non-Repetitive Genetic Parts for Engineering Evolutionary Robust Genetic Systems

**Non-repetitive promoter library design.** The non-repetitive  $\sigma^{70}$  promoter library was designed using the Non-Repetitive Parts Calculator in Maker mode. We used 32 separate IUPAC degenerate sequence constraints with varied numbers of hexamer mismatches and a maximum shared repeat of  $L_{max}$  of 10 base pairs. To do this, Maker mode is run iteratively using previously designed non-repetitive part sequences as the background sequence for the next iteration. Model functions were used to exclude XbaI and BamHI restriction sites, hexamer-like motifs outside the -35, -10 regions, and polymeric sequences. Designed part sequences must also have GC% between 30 and 70%.

**Library cloning and growth.** A 12472-variant oligopool was designed and synthesized (CustomArray), using maximally unique primer binding sites and maximally unique barcodes (hamming distances 5 or greater). The oligo pool was PCR amplified for 20 cycles using Q5 High-Fidelity DNA Polymerase (New England Biolabs) and primers PCR\_FWD and PCR\_REV. The PCR products were digested with BamHI and XbaI (New England Biolabs), and subsequently ligated using T4 DNA Ligase (New England Biolabs) into pre-digested flexible test vector, pFTV1. Ligated vectors were then transformed in duplicate into NEB-5 $\alpha$  cells and recovered for 60 minutes in 1 mL SOC. Serial dilutions were performed from 100  $\mu$ L of recovery broth and plated on selective LB agar plates (50  $\mu$ g/mL chloramphenicol) to estimate transformation efficiencies for each replicate (~4 million CFUs/mL). The remaining 900  $\mu$ L of recovery broth was transferred to duplicate Erlenmeyer flasks containing 50 mL LB Miller with 50  $\mu$ g/mL chloramphenicol for 16 hours of growth in a shaking incubator at 37°C. The library was diluted to an OD<sub>600</sub> of 0.05 in 50 mL of M9 minimal media with 0.4% glucose, 0.5 mM leucine and antibiotic selection, grown to an OD<sub>600</sub> of 0.2, and re-diluted to an OD<sub>600</sub> of 0.005 for a final growth of ~15 hours to ensure that the cells were characterized in exponential growth phase and at steady state (with constant relative mRNA transcript abundances).

**DNA-Seq and RNA-Seq.** Total RNA from 5 mL of the 50 mL cultures with an OD 0.2 were extracted using Norgen Biotek Total RNA Purification Kit. Extracted total RNA was treated with Invitrogen TURBO DNA-free Kit. RNA integrity and DNA removal were verified by running an analytical agarose gel. Total RNA was then concentrated to 10  $\mu$ L using the Qiagen RNeasy MinElute Cleanup Kit. 5  $\mu$ g total RNA samples were depleted of rRNA using Illumina Ribo-Zero Removal Kit and cleaned up using the Qiagen RNeasy MinElute Cleanup Kit. cDNA was synthesized from the rRNA depleted samples using Invitrogen SuperScript IV First-Strand Synthesis System. Reverse primer RNA-Seq\_REV was used for gene-specific reverse transcription. cDNA was treated with RNase H to degrade all RNA hybridized to the cDNA. Plasmid DNA from the same 50 mL cultures were harvested using QIAprep Spin Miniprep Kit following manufacturer's instructions. Promoters from the extracted plasmids were PCR amplified using DNA-Seq\_FWD and DNA-Seq\_REV, and barcodes from transcript cDNA were amplified using RNA-Seq\_FWD and RNA-Seq\_REV primer. All amplicons were verified to be correct length on an analytical agarose gel. The two DNA and RNA biological replicates were sequenced on an Illumina HiSeq (Quintara Biosciences) to generate at least 40 million 150-bp paired-end reads per sample.

**Calculation of relative transcription rates.** Only DNA-Seq and RNA-Seq amplicon reads with completely intact primers (Levenshtein edit distance of 0) were filtered for downstream use. All reads were then mapped to designed promoters and associated barcodes and binned according to the computed Levenshtein edit distance of match. Reads with multiple reference matches within the same edit distance or edit distance greater than 1 were discarded from analysis. Replicate read count correlation analysis was performed and counts were standardized using Lowess normalization. The transcription rates of designed promoters were computed using the equation:

$$Transcription\ Rate(Promoter_j) = 0.5 \times \sum_{i=1}^2 \frac{RNA_{i,j}}{\sum RNA_i} / \frac{DNA_{i,j}}{\sum DNA_i}$$

where  $DNA_{i,j}$  and  $RNA_{i,j}$  represent the DNA-Seq and RNA-Seq read counts of the  $j^{th}$  promoter in  $i^{th}$  replicate. Transcription rates were then normalized with respect to J23100 promoter using the equation:

$$Normalized\ Transcription\ Rate(Promoter_j) = \frac{Transcription\ Rate(Promoter_j)}{Transcription\ Rate(J23100)}$$

**Data analysis.** Dynamic range of the library was computed by taking the ratio of the sum of maximum observed transcription rate and its standard deviation to the difference between minimum observed transcription rate and its standard deviation. Combinatorial analysis was done using standard Levenshtein edit distance and Hamming distance definitions and applied on all possible pairs of promoters and barcodes. Depth of usability analysis was computed by sorting all promoters according to their transcription rates and then computing the range of rates around each measured transcription rate to report the number of promoters that were interchangeable within a threshold. Breadth of usability analysis was done by selecting promoter subsets of size 2 to 4,350 and then reporting the maximum and minimum dynamic range associated with subset size. Fitting distributions to histograms, ordinary least squares linear regression, Lowess normalization, LASSO regression ( $\alpha = 0.01$ ) and ANOVA for model comparisons were done using standard functions available in NumPy, SciPy and scikit-learn. UMAP analysis was done on vectorized promoter sequences and their measured transcription rate using the standard program available from its author ([github.com/lmcinnes/umap](https://github.com/lmcinnes/umap)). WebLogo analysis of sequences were done using the standard webservice available from University of California, Berkeley ([weblogo.berkeley.edu/logo.cgi](http://weblogo.berkeley.edu/logo.cgi)), while DNA entropy computations were done following standard definitions. All visualizations were developed using seaborn and matplotlib.

## 4.0 Results and Discussion

### 4.1 Automated Model-Predictive Design of Synthetic Promoters to Control Transcriptional Profiles in Bacteria

Transcription is the gene expression process responsible for producing all RNA and is a common engineering target for creating novel products, including microbial chemical factories, toxin-sensing genetic circuits, and mRNA vaccines [14-16]. However, while DNA assembly techniques enable the construction of custom-designed genetic systems [17], it remains challenging to a priori predict and control a system's gene expression profile [18], for example, by initiating transcription with desired rates at specific DNA start sites, while minimizing transcription from all other DNA sequence regions. Currently, transcriptional control relies on empirical characterization of promoters as modular genetic parts [19]. Applying modular design to transcriptional control ignores other sources of transcription, for example, inside coding regions, as well as local and long-distance interactions that alter transcription rates and start sites in unexpected ways [20]. Poorly controlled transcriptional profiles can lead to malfunctioning genetic systems, including the undesired production of anti-sense RNA and truncated proteins as well as the misbalancing of protein expression levels that lead to lower system activities [3, 21].

A key challenge is to quantitatively predict how polymerase initiation complex – RNA polymerase (RNAP) and a sigma factor ( $\sigma$ ) in bacteria – interacts with arbitrary DNA sequences [22, 23]. Beyond canonical sequence motifs, it remains unclear how the strengths of multiple interactions collectively determine transcription initiation rates and start sites, particularly when bound to RNAP/ $\sigma^{70}$ , which initiates transcription at the majority of bacterial promoters. To help resolve this challenge, we carried out massively parallel experiments on designed promoter sequences to systematically measure the interactions controlling site-specific transcription at  $\sigma^{70}$  promoters (**Figure 1A**). With this data, we developed a statistical thermodynamic model that calculates how RNAP/ $\sigma^{70}$  interacts with arbitrary DNA to predict transcription initiation rates at each position. The model has only 346 interaction energy parameters, but accurately predicts the transcription rates of 17396 bacterial promoters with diverse sequences. We show how the model enables the automated design and debugging of transcriptional profiles in engineered genetic systems.

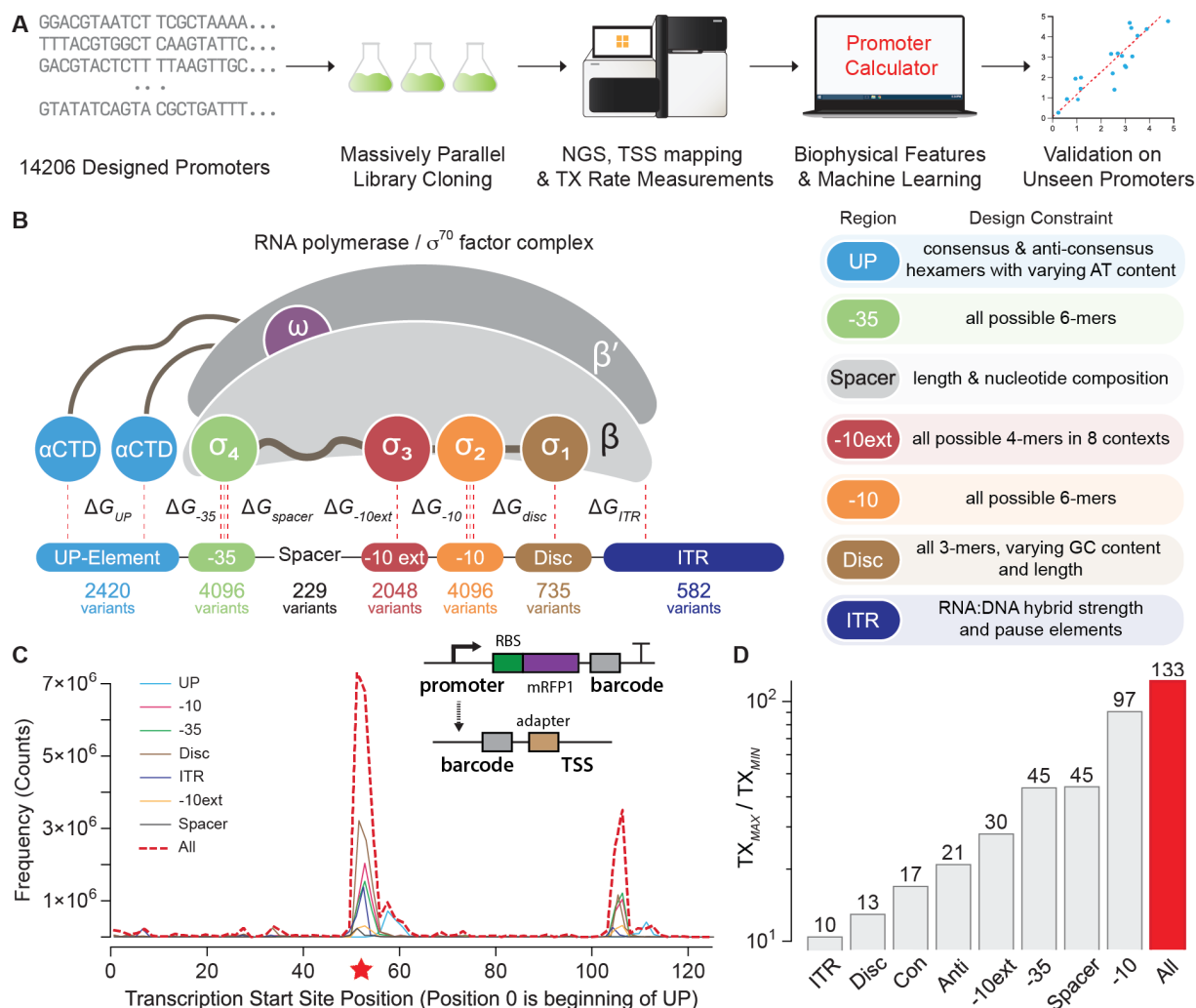


**Figure 1: Massively Parallel Transcription Rate and Start Site Measurements.** (A) Model development combined promoter design, barcoded oligopool synthesis, library cloning & culturing, and next-generation sequencing to measure the transcription start site and transcription rate of each promoter variant. (B) The interaction strengths between RNAP/ $\sigma^{70}$  and promoter DNA control transcription initiation rates. 14206 promoter variants were designed to quantify how sequence modifications affect each interaction. Sequence design criteria are shown. (C) The frequencies of observed transcription start sites are shown for each set of promoter variants. The star indicates the predominant start site. The inset schematic shows the system architecture, the locations of the promoter and barcode variants, and the cDNA architecture after library preparation. (D) The transcription rates' dynamic ranges are shown for each set of promoter variants, considering only the predominant start site. Con: UP-varied promoters with consensus hexamers. Anti: UP-varied promoters with anti-consensus hexamers.

To begin, we designed 14206 promoter variants with varied motif sequences to systematically perturb the interactions that affect RNAP/ $\sigma^{70}$  binding and transcriptional initiation (Figure 1B). These interactions occur at DNA sites known by their canonical positions [24, 25], including (i) an upstream 6-nucleotide site called the -35 motif; (ii) a 20-nucleotide region that appears upstream of the -35 motif, called the UP element; (iii) a downstream 6-nucleotide site called the -10 motif; (iv) a spacer region that separates the -10 and -35 motifs; (v) a 4-nucleotide site upstream of the -10 motif, called the -10

extended motif (-10ext); (vi) a typically 6-nucleotide region in between the -10 motif and transcriptional start site, called the discriminator (Disc); and (vii) the first 20 transcribed nucleotides, called the initial transcribed region (ITR).

Initial RNAP/ $\sigma^{70}$  binding to a promoter is controlled by the interaction Gibbs free energies ( $\Delta G$ ) at the UP, -35, -10 extended, and -10 motifs as well as the torsional stress controlled by the length of the spacer region [24-29]. Bound RNAP/ $\sigma^{70}$  then undergoes a conformational change that catalyzes double-stranded DNA separation, creating a transcription bubble that initially encompasses half the -10 motif, the Disc, and the first two nucleotides of the ITR [25, 26]. RNA polymerization begins at a transcription start site (TSS) determined by where the catalytic site in the  $\beta$  subunit contacts the DNA template, canonically at position +1. The transcription bubble is then stabilized by interactions in the Disc [30] and the formation of an R-loop, whereby the newly synthesized RNA strand immediately hybridizes to the DNA template [31]. The ITR sequence controls the R-loop's thermodynamic stability. Finally, transcription initiation is successful once enough DNA is pulled into the stable transcription bubble that the accumulated stress exceeds the interaction strength between RNAP/ $\sigma^{70}$  and promoter DNA, causing promoter escape and a transition to processive RNA synthesis [31-33].

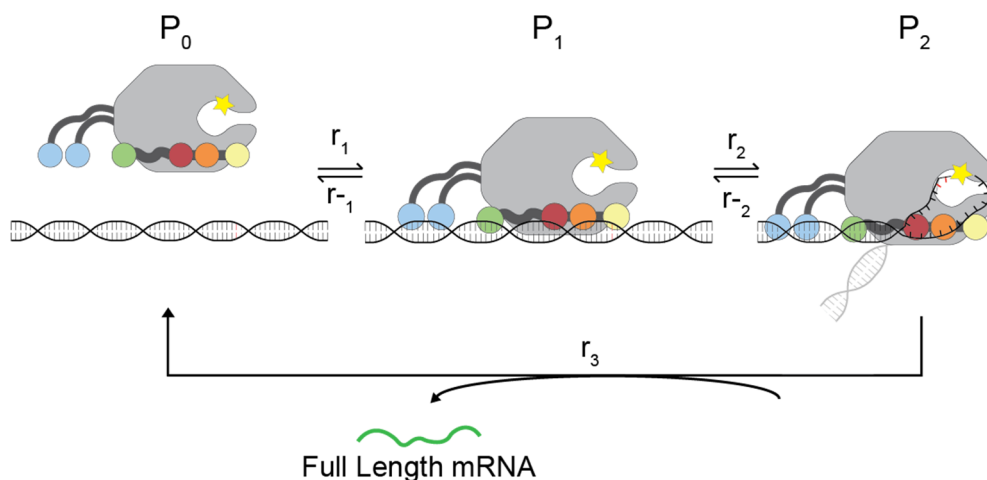


**Figure 2: A 3-State Thermodynamic System.** A coarse-grained representation of prokaryotic transcription initiation.  $P_0$  is the free RNAP and promoter DNA,  $P_1$  is the closed complex, and  $P_2$  is the open complex. Rates are denoted by  $r$ , where negative subscripts denote reverse rates and positive subscripts denote forward rates.

From these interactions, we formulated a statistical thermodynamic model of transcriptional initiation that accounts for competitive binding of RNAP/ $\sigma^{70}$  to all DNA, the multiple sequence contacts that control RNAP/ $\sigma^{70}$  recruitment at each promoter, and the multiple internal states during transcription initiation. So long as the internal states do not become abundant, for example, by significant transcriptional pausing [34], the model indicates that we can decompose how a promoter's sequence controls the interaction energies into a sum of free energies that can be related to the transcription initiation rate (TX), according to:

$$\Delta G_{total} = \Delta G_{UP} + \Delta G_{-35} + \Delta G_{spacer} + \Delta G_{-10ext} + \Delta G_{-10} + \Delta G_{disc} + \Delta G_{ITR} \quad \text{Equation 1}$$

$$\log\left(\frac{TX}{TX_{ref}}\right) = -\beta(\Delta G_{total} - \Delta G_{total,ref}) \quad \text{Equation 2}$$

where  $\Delta G_{total}$  is the difference in free energy between an unbound promoter and a promoter- RNAP/ $\sigma^{70}$  complex with a stable transcriptional bubble, which is used to predict a promoter's TX rate in comparison to a reference promoter sequence with calculated  $\Delta G_{total,ref}$  and measured  $TX_{ref}$ .  $\beta$  is a measurable model constant that converts free energies into state probabilities. Here, we arbitrarily set  $\Delta G_{total,ref}$  to zero so that stronger (weaker) interaction free energies have more negative (positive) values in comparison.

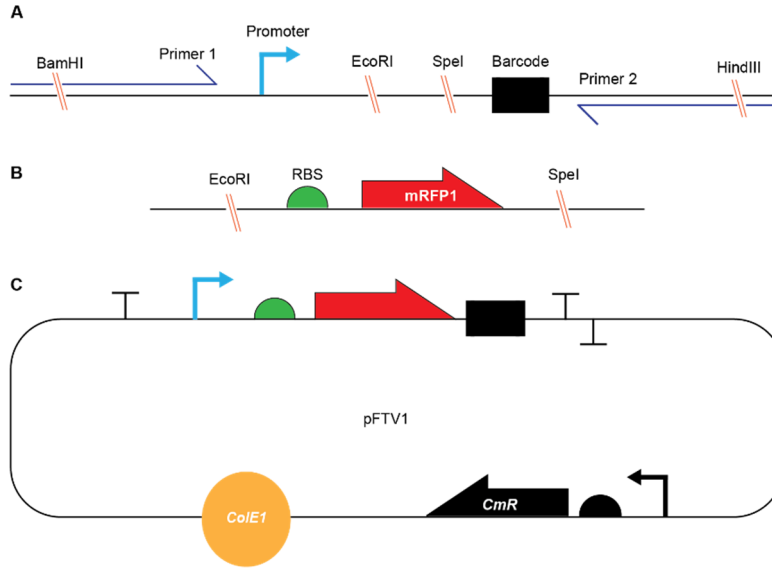


**Figure 3: Designs and Plasmid Map. (A)**, Oligo Pool Library design with variable promoters. **(B)**, Gene-block containing 5' cleaving ribozyme, ribosome binding site and a reporter protein. **(C)**, Final assembled plasmid used for promoter library characterization.

We designed the 14206 promoter sequences to measure how each motif sequence alters these free energies to create a sequence-complete model, including all possible -10 hexamers, all possible -35 hexamers, all possible -10 extended motifs within 8 consensus and anti-consensus background configurations, 229 sequence spacers with varied lengths and nucleotide compositions, 2420 UP elements with varied AT content within 4 consensus and anti-consensus background configurations, 735 discriminator elements with varied lengths and GC content, and 582 ITR sequences with varied R-loop stabilities (**Figure 1B**). Using oligopool synthesis and two-step library cloning, we constructed a barcoded plasmid pool that uses each promoter in a common genetic context, expressing a single protein with a moderate translation initiation rate (about 5000 on the RBS Calculator v2.1 scale [18]), with unique barcode sequences positioned in the 3' untranslated region to avoid confounding effects (**Figure 3**).

**Table 1:** Primers, RNA Adapter, and DNA Fragment Sequences

| Part Name    | Part Sequence                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| primer 1     | TGCTGGATCCCTACTCTGAG                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| primer 2     | CTATAAGCTTGGTCCGACGG                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| primer 3     | TCCCACAACGAAGACTACACC                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| primer 4     | CTCTTTGATAACGTCTTCAGAGC                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| primer 5     | AACAGGATCCACGCACTCTA                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| primer 6     | CGGACGACCTTCACCTTCA                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| gene-block 1 | CTGTGCGTTAGGTGATGCCGCGGCTGTACCGCCCATGTGCTCACGAATTCGATCAA<br>ATTTTCGAGGTTCCATATGGCGAGCTCTGAAGACGTTATCAAAGAGTTCATGCGTTTC<br>AAAGTTCGTATGGAAGGTTCCGTTAACGGTCACGAGTTCGAAATCGAAGGTGAAGG<br>TGAAGGTCGTCCGTACGAAGGTACCCAGACCGCTAAACTGAAAGTTACCAAAGGTG<br>GTCCGCTGCCGTTTCGCTTGGGACATCCTGTCCCCGCAGTTCAGTACGGTTCCAAA<br>GCTTACGTAAACACCCGGCTGACATCCCGGACTACCTGAAACTGTCCTTCCCGGA<br>AGGTTTCAAATGGGAACGTGTTATGAACTTCGAAGACGGTGGTGTGTTACCGTTAC<br>CCAGGACTCCTCCCTGCAAGACGGTGAGTTCATCTACAAAGTTAAACTGCGTGGTA<br>CCAAGTTCCCGTCCGACGGTCCGTTATGCAGAAAAAACCATGGGTTGGGAAGCT<br>TCCACCGAACGTATGTACCCGGAAGACGGTGCTCTGAAAGGTGAAATCAAAATGCG<br>TCTGAAACTGAAAGACGGTGGTCACTACGACGCTGAAGTTAAAACCACTACATGG<br>CTAAAAAACCGGTTTCAGCTGCCGGGTGCTTACAAAACCGACATCAAAGTGGACATC<br>ACCTCCCACAACGAAGACTACACCATCGTTGAACAGTACGAACGTGCTGAAGGTGCG<br>TCACTCCACCGGTGCTTAATAAACTAGTAAACGCAGTTACCCCATAGGCT |
| RNA adapter  | AACAGGAUCCACGCACUCUANN                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |

We then carried out in vitro transcription reactions, TSS mapping, and next-generation sequencing to measure the TX rate of each promoter at each TSS, utilizing a minimal system containing only RNAP/ $\sigma 70$ , the barcoded plasmid pool, NTPs, and buffer. TSS mapping was performed by harvesting product RNA, ligating a 5' RNA adapter, converting to cDNA, circularizing, using PCR to generate amplicons containing the barcode and TSS, and obtaining over 323 million barcode-TSS mapped reads from Illumina sequencing (**Table 1**). Across all promoters, we found 182 TSSs with at least 1000 mapped reads, revealing a primary TSS region and a less frequent, secondary TSS region arising from a downstream cryptic promoter. TX rates were then measured by carrying out DNA-Seq and RNA-Seq on triplicate in vitro transcription reactions, separately harvesting DNA and RNA, converting RNA to cDNA, using PCR to generate barcode-containing amplicons, and obtaining between 81.3 and 130.5 million barcoded

reads per reaction. From the RNA/DNA read count ratios, we obtained TX rates for 13480 promoter variants (mean and standard deviation), excluding any with fewer than 50 read counts in any replicate. Replicate read count measurements were highly reproducible ( $R^2 = 0.89$  to  $0.99$ , **Figure 4**) with 5388 high-precision TX rates (coefficient of variation  $< 0.40$ ).

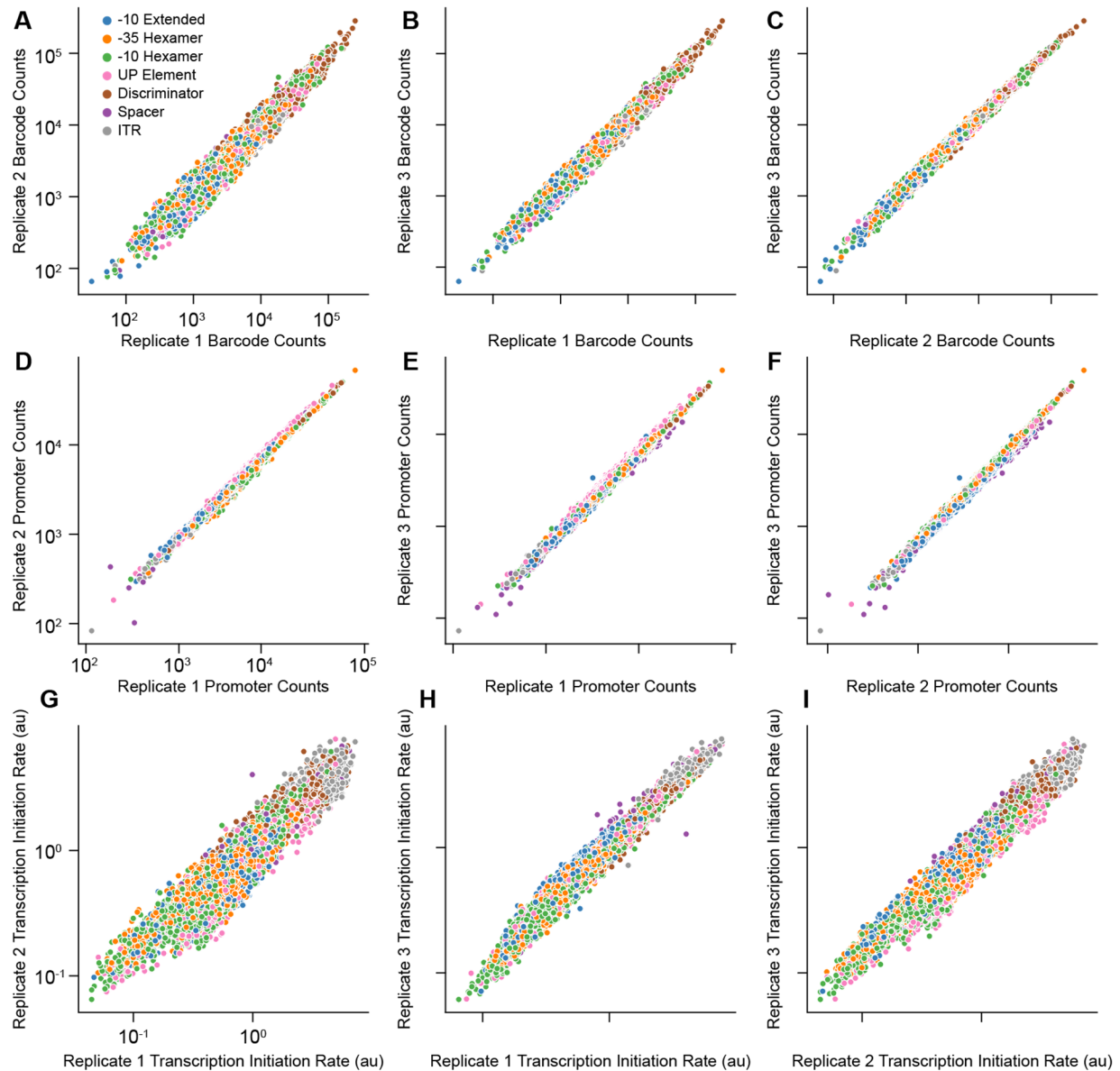


**Figure 4: Replicate Correlations.** RNA read count correlations for (A) replicate 1 vs replicate 2, (B) replicate 1 vs replicate 3 and (C) replicate 2 vs replicate 3 ( $R^2 = 0.94$ ,  $0.97$ , and  $0.99$  respectively). DNA read count correlations for (D) replicate 1 vs replicate 2, (E) replicate 1 vs replicate 3 and (F) replicate 2 vs replicate 3 ( $R^2 = 0.99$ ,  $0.99$ , and  $0.99$  respectively). Transcription Initiation Rate correlations for (G) replicate 1 vs replicate 2, (H) replicate 1 vs replicate 3 and (I) replicate 2 vs replicate 3 ( $R^2 = 0.89$ ,  $0.94$ , and  $0.97$  respectively). Blue dots represent -10 extended promoter variants, orange dots represent -35 hexamer promoter variants, green dots represent -10 hexamer promoter variants, pink dots represent UP element promoter variants, brown dots represent discriminator promoter variants, purple dots represent spacer promoter variants, and grey dots represent ITR promoter variants.

We began model training by identifying 5193 promoter variants where RNAP/ $\sigma$ 70 bound to a single site with one predominant TSS, enabling us to unambiguously pinpoint their motif sequences (**Figure 1C**, **Figure 5**). The TX rates for these single-site promoters altogether varied by 133-fold with sizable effects from each individual motif (**Figure 1D**). Importantly, endogenous RNAses were not present in the in vitro transcription reactions, enabling us to vary the Disc and ITR sequences without changing the mRNA's stability, which is not possible in equivalent in vivo measurements. We then specified 472 sequence, structural, and energetic properties to relate how each motif sequence contributes to each binding free energy term (**Table 2**). For example, for any UP sequence, we calculate the minor groove width of the distal and proximal UP sites [2]; for any ITR sequence, we calculate the thermodynamic stability of the R-loop [35, 36]; for any spacer sequence, we calculate the local DNA rigidity [1]. As categorical properties, we split the -35, -10, and Disc motifs into six 3-nucleotide regions and include 384 3-mers. We split the -10ext motif into 2-nucleotide regions and include 32 2-mers. We also include the spacer length as a categorical property.

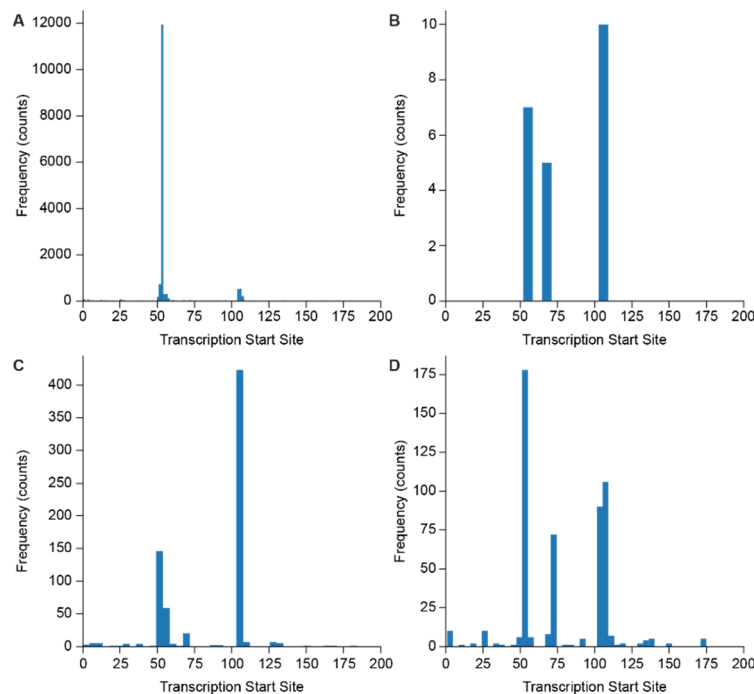


**Figure 5: Transcription Start Site Profiles.** (A), The measured TSS distribution of a variant that met all filtering criteria and exhibited a strong on-target peak. (B), The measured TSS distribution of a variant that did not meet the minimum read threshold. (C), The measured TSS distribution of a variant that had an off-target peak (position ~105) with greater than twice the counts of the on-target peak (position 53). (D), The measured TSS distribution of a variant with multiple strong TSS peaks

**Table 2:** Model Features

| Feature Name, Feature Type              | Number of Parameters | Number Retained <sup>a</sup> |
|-----------------------------------------|----------------------|------------------------------|
| -10 hexamer 3-mers                      | 128                  | 128                          |
| -35 hexamer 3-mers                      | 128                  | 128                          |
| Discriminator 3-mers                    | 128                  | 64                           |
| -10 extended 2-mers                     | 32                   | 16                           |
| Discriminator Length                    | 7                    | 0                            |
| Discriminator GC Content                | 1                    | 0                            |
| Discriminator Purine Content            | 1                    | 0                            |
| TSS 2-mers                              | 16                   | 0                            |
| Spacer Length                           | 6                    | 6                            |
| Spacer Stacking Free Energy             | 1                    | 0                            |
| Spacer Torsional Energy                 | 1                    | 0                            |
| UP Groove Width (Distal and Proximal)   | 2                    | 2                            |
| ITR Purine Content                      | 1                    | 0                            |
| ITR Pause-Element Location              | 15                   | 0                            |
| UP A-tract Length (Distal and Proximal) | 2                    | 0                            |
| UP AT Content                           | 1                    | 0                            |
| DNA Rigidity                            | 1                    | 1                            |
| R-loop Strength                         | 1                    | 1                            |

<sup>a</sup> Retained parameters are the only ones present in the final model

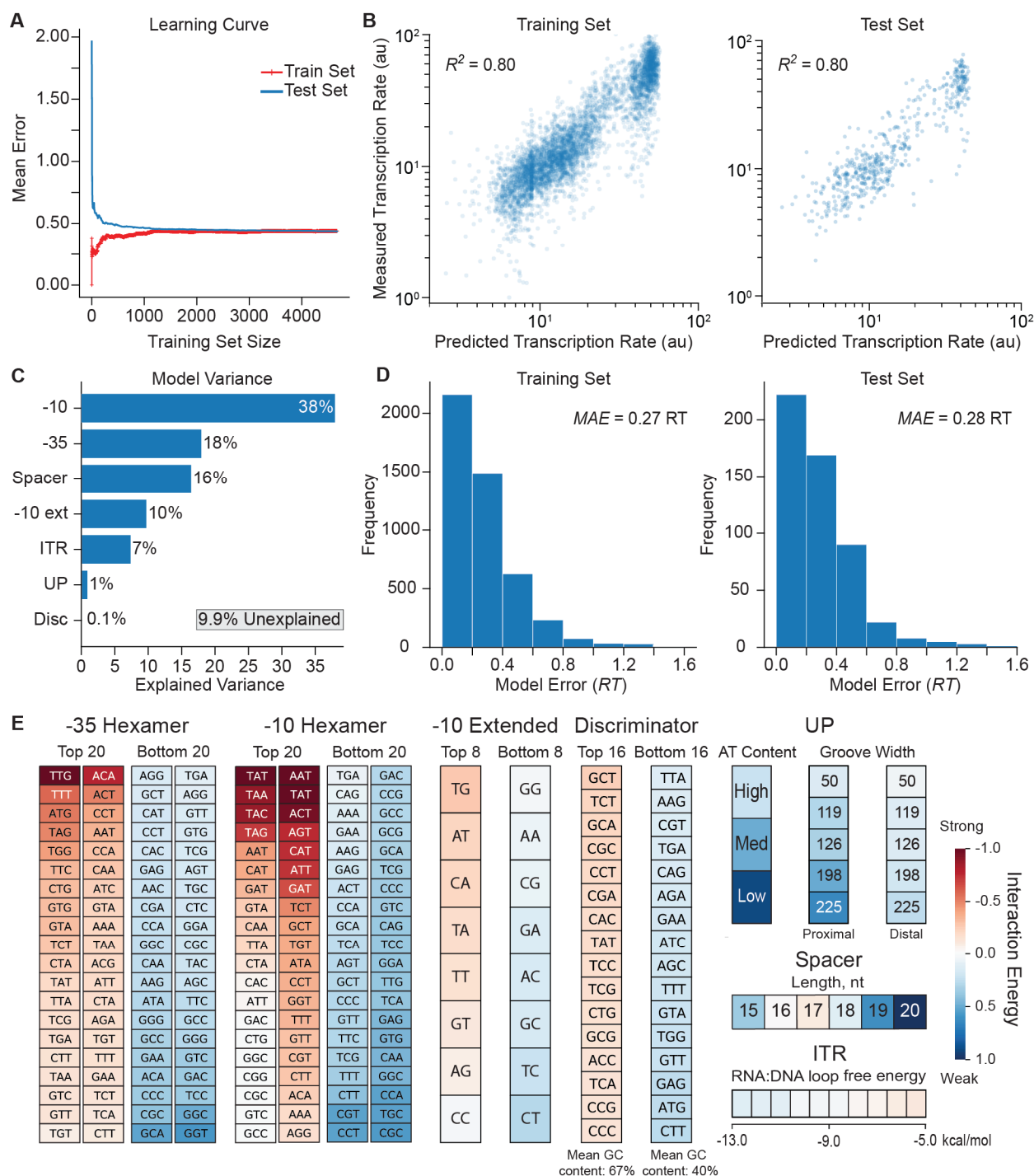


**Figure 6: Model Development and Validation using Machine Learning.** (A) A learning curve shows the training and testing of a ridge regression model to identify the unknown interaction energies. (B) Model-predicted transcription rates are compared to measured transcription rates for both (left) the training set and (right) the test set. (C) The explained variances for each promoter interaction are shown. (D) Histograms of model error are shown for the training and test sets, using energy units. MAE is mean absolute error. (E) The learned interaction energies are shown for the strongest and weakest ones.

We then randomly split our dataset into a training set (4673 promoters, 90%), carried out 10-fold cross-validation to identify the optimal hyperparameters of a machine learning

model, and tested its accuracy on the remaining unseen test set (520 promoters, 10%). We evaluated several models (**Table 3**) that use regularization to identify the optimal coefficients of a linear, additive model that mirrors our free energy parameterization (Equation 1) with dataset normalization that mirrors our statistical thermodynamic model (Equation 2). We carried out feature reduction to remove extraneous properties, which lowered the number of fitted coefficients. The pruned properties included the first 2-nucleotide region of the -10ext motif and the last 3-nucleotides of the Disc motif, which had no discernable effect on TX rate in this dataset. Overall, we found that a ridge regression model with 346 fitted coefficients yielded a convergent learning curve (**Figure 6A**) and highly accurate predictions ( $R^2 = 0.80$ , **Figure 6B**) with similarly low error distributions (**Figure 6C**) across both the training and unseen test datasets. We then performed ANOVA analysis to quantify how each predicted free energy term contributed to the promoters' measured TX rates in our dataset and found that 83% of the TX rate variance could be explained by varying the interactions that control initial RNAP/ $\sigma$ 70 recruitment, using the UP, -35, -10, -10ext, and spacer motifs (**Table 3**, **Figure 6D**). In contrast, 7.1% of the TX rate variance was explained by differences in the interactions controlling DNA melting, R-loop formation, and promoter escape, which are affected by the Disc and ITR regions.

**Table 3:** Model Selection

| Model Type | # Variables | R2 Train | MAE Train | MSE Train | R2 Test | MAE Test | MSE Test | Correct / Total Positive Controls <sup>a</sup> |
|------------|-------------|----------|-----------|-----------|---------|----------|----------|------------------------------------------------|
| Ridge      | 472         | 0.82     | 0.265     | 0.120     | 0.80    | 0.270    | 0.126    | 4/8                                            |
| Lasso      | 472         | 0.82     | 0.264     | 0.120     | 0.80    | 0.270    | 0.126    | 4/8                                            |
| ElasticNet | 472         | 0.82     | 0.264     | 0.120     | 0.80    | 0.270    | 0.126    | 4/8                                            |
| Ridge      | 346         | 0.80     | 0.274     | 0.133     | 0.80    | 0.281    | 0.131    | 8/8                                            |
| Lasso      | 346         | 0.80     | 0.274     | 0.134     | 0.80    | 0.281    | 0.131    | 8/8                                            |
| ElasticNet | 346         | 0.80     | 0.274     | 0.133     | 0.80    | 0.281    | 0.131    | 8/8                                            |

<sup>a</sup> Positive controls were defined to match known consensus promoter sequence motifs [24]

**Table 4:** Model Performance Metrics

| Data Set        | Metric              | Value |
|-----------------|---------------------|-------|
| La Fleur et al. | Pearson Correlation | 0.79  |
| La Fleur et al. | Spearman Rank Order | 0.87  |
| La Fleur et al. | Mean Square Error   | 0.23  |
| La Fleur et al. | Mean Absolute Error | 0.33  |
| Urtecho et al.  | Pearson Correlation | 0.60  |
| Urtecho et al.  | Spearman Rank Order | 0.78  |
| Urtecho et al.  | Mean Square Error   | 0.41  |
| Urtecho et al.  | Mean Absolute Error | 0.54  |
| Hossain et al.  | Pearson Correlation | 0.45  |
| Hossain et al.  | Spearman Rank Order | 0.68  |
| Hossain et al.  | Mean Square Error   | 0.69  |
| Hossain et al.  | Mean Absolute Error | 0.66  |
| Yu et al.       | Pearson Correlation | 0.65  |
| Yu et al.       | Spearman Rank Order | 0.69  |
| Yu et al.       | Mean Square Error   | 0.51  |
| Yu et al.       | Mean Absolute Error | 0.55  |

A key benefit of our hybrid biophysical-machine learning approach is that the fitted coefficients quantify the physical interactions between RNAP/ $\sigma$ 70 and each promoter motif (**Figure 6E**), including all motif sequences and enabling direct comparisons to already known canonical interactions as positive controls. For example, without human supervision, our approach correctly identified the canonical -35 motif (TTGACA), -10 motif (TATAAT), extended -10 motif (TG), and the optimal spacer length (17 base pairs). We also found that AT-rich distal and proximal UP sites had more favorable interactions and that GC-rich Disc motifs were enriched, which were previously observed [29, 30]. However, most promoters do not contain canonical motif sequences; their TX rates are controlled by a mixture of weaker interactions. A key novelty of our model is its complete

set of interaction energies, covering all possible promoter sequences, which provides the ability to predict the TX rate of any  $\sigma 70$  promoter.

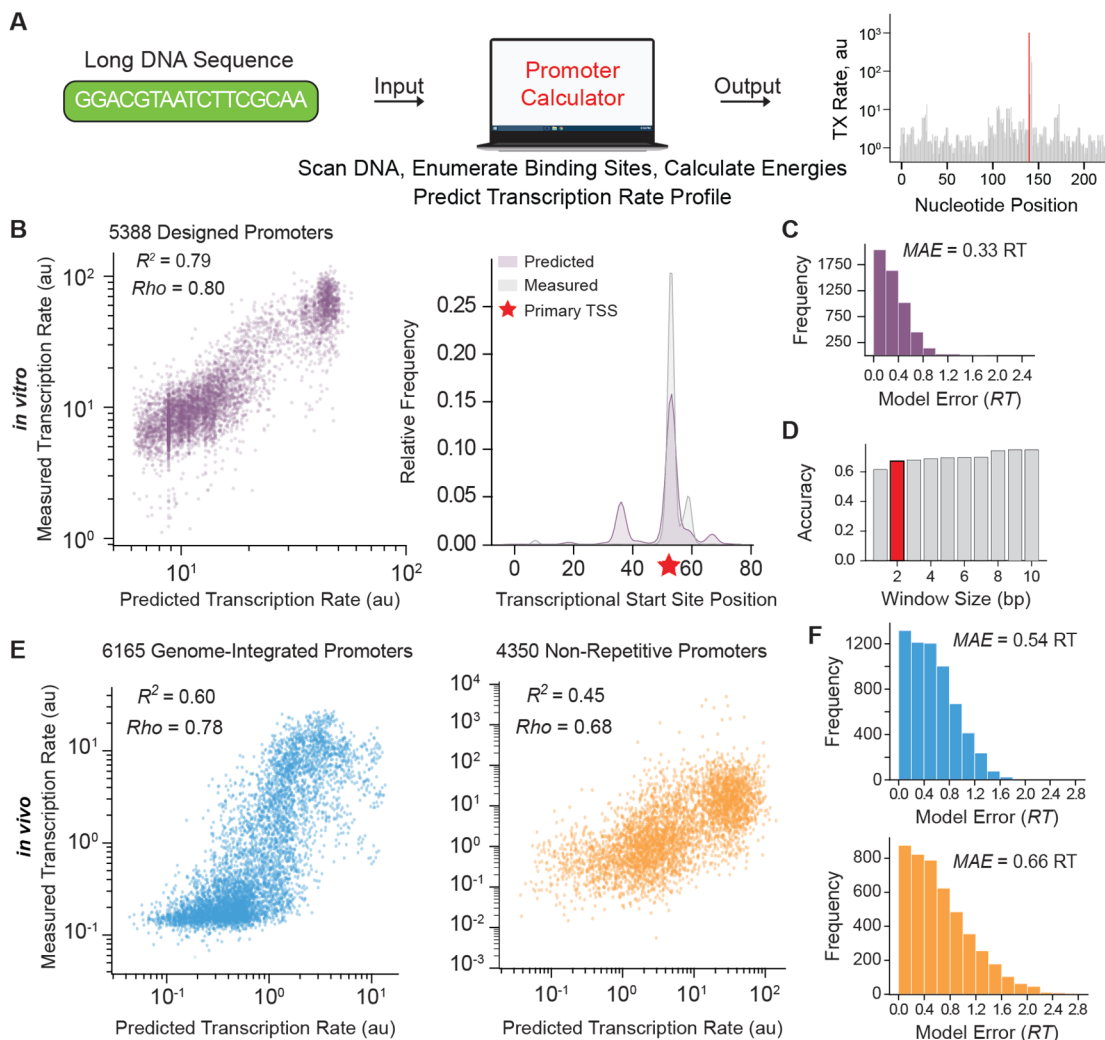


**Figure 7: Validation of Sequence-to-Function Model on Diverse Promoters.** (A) Arbitrary DNA sequences are inputted into the model to predict its transcriptional profile (transcription rates vs. nucleotide positions) without start site information. (B) Model predictions for 5388 designed promoters (this study) are compared to *in vitro* transcription rate and start site measurements. (C) The histogram of the scanning model's error is shown when evaluated on the designed promoters. (D) The model's start site prediction accuracy is shown within defined windows. (E) Model predictions on unseen test sets, including (left) 4350 non-repetitive promoters [37] and (right) 6185 genome-integrated modular promoters [38], are compared to *in vivo* transcription rate measurements. (F) Histograms of the model's error when evaluated on (top) 4350 non-repetitive promoters and (bottom) 6185 genome-integrated modular promoters.

We then tested the model's generalizability and accuracy across 17396 well-characterized promoters, using sequence information alone to predict their transcriptional profiles, specifically their TX rates for each potential TSS position (**Figure 7A**). Here, we do not input the promoters' actual TSS positions and motifs, and instead combine the interaction energies with statistical thermodynamics to identify the most likely RNAP/ $\sigma 70$  binding site configuration. For each potential TSS position, we scan the surrounding DNA

sequence and enumerate the several ways that RNAP/ $\sigma$ 70 may bind to it, varying the spacer and Disc lengths with corresponding changes in motif sequences. We apply the model's interaction energies to calculate the  $\Delta G_{\text{total}}$  for each configuration (**Equation 1**) and determine the configuration with the most negative  $\Delta G_{\text{total}}$ , which is then used to predict the TX rate (**Equation 2**). We repeat these calculations for each potential TSS position within the inputted DNA sequence.

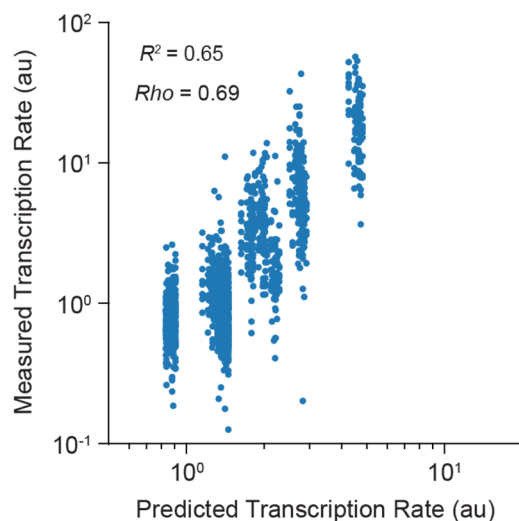


**Figure 8: Model Accuracy on IPTG-inducible Promoters.** Predicted vs. Measured transcription rates for 1493 genome-integrated promoters that contain lacO sites for IPTG induction, characterized *in vivo* by Yu et al. with 1 mM IPTG added [39].

We carried out these additional tests on four datasets: (1) 5388 designed promoters with high-precision TX rate and TSS measurements, characterized here using *in vitro* RNAP/ $\sigma$ 70 transcription assays; (2) 6165 promoters using combinations of motifs to express a ribozyme-insulated transcript, characterized using genome-integrated test circuits inside *E. coli* cells [38]; (3) 4350 non-repetitive promoters using highly diverse sequences to express a transcript, characterized using test circuits carried on multi-copy plasmids inside *E. coli* cells [37]; and (4) 1493 inducible promoters that contain transcription factor binding sites and express a ribozyme-insulated transcript, also characterized using genome-integrated test circuits inside *E. coli* cells [39]. We first confirmed that the statistical thermodynamic model accurately predicted the TX rates of the *in vitro* dataset #1 without utilizing TSS measurements ( $R^2 = 0.79$ ,  $r = 0.80$ , **Figure 7B**) with 4954 promoters (92%) having predicted TX rates to within 2-fold of their actual TX rates (**Figure 7C**) and TSS positions predicted to within 2 bp (67% accuracy, **Figure 7D**). Next, we tested the model's predictions on the three *in vivo* datasets, which were not used during model development. We found that the statistical thermodynamic model retained accuracy ( $R^2 = 0.60$ ,  $r = 0.78$ ;  $R^2 = 0.45$ ,  $r = 0.68$ ;  $R^2 = 0.65$ ,  $r = 0.69$ ; **Figure 7E** and **Figure 8**) with low model errors (**Figure 7F**), even though *in vivo* mRNA levels are affected by other potentially confounding factors, for example, variation in DNA copy number and mRNA stability.

To evaluate the model's predictions using additional types of measurements, we then selected ten of the non-repetitive promoters in dataset #3 and characterized their expression levels in test genetic circuits inside *E. coli* cells, utilizing RT-qPCR to measure their mRNA levels and flow cytometry to measure their fluorescent reporter protein levels. We found that the model's predicted TX rates for these promoters were highly proportional to their measured mRNA levels ( $R^2 = 0.75$ ,  $r = 0.87$ ) and fluorescent protein expression levels ( $R^2 = 0.78$ ,  $r = 0.87$ ) over a 96000-fold dynamic range (**Figure 9**).

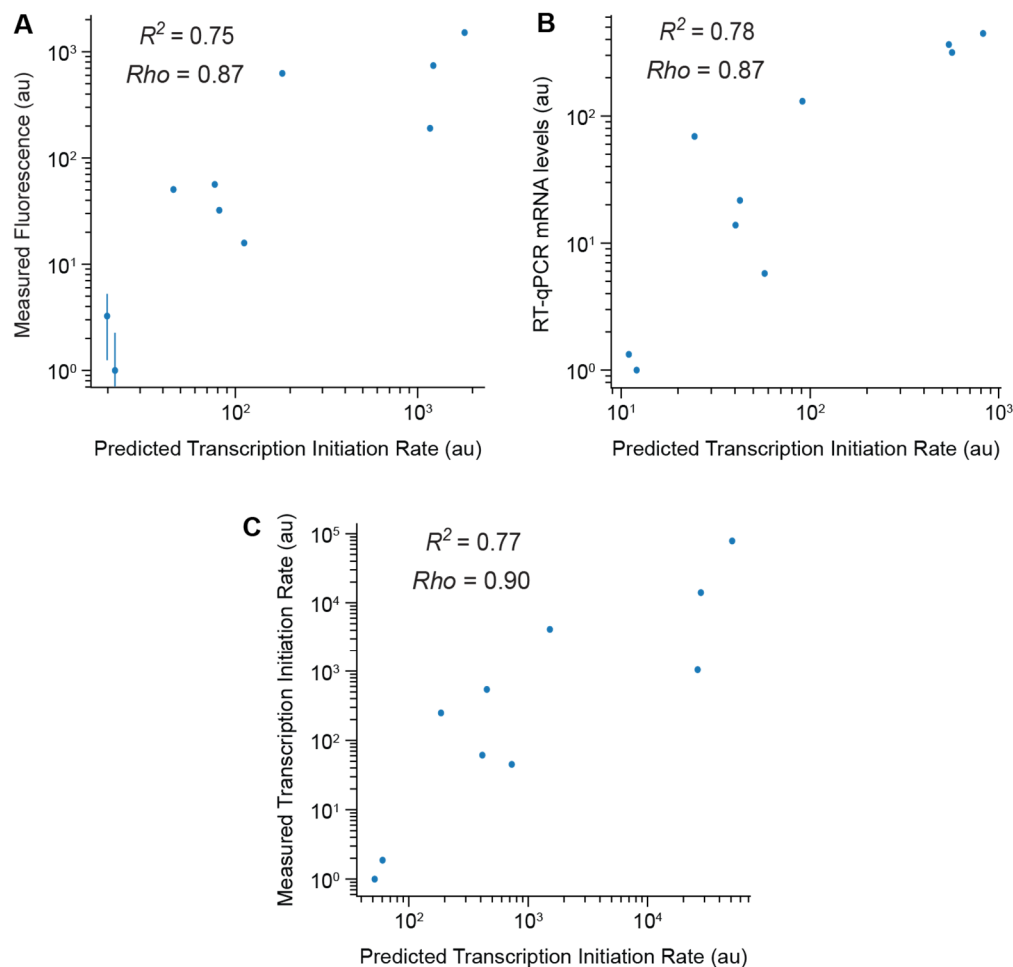


**Figure 9: Model Accuracy on Selected Non-Repetitive Promoters with 3 Types of Measurements.** (A), Predicted vs. Measured TX rates for the 10 promoters further characterized by Hossain et al. using flow cytometry. Error bars were created from duplicate measurements. (B), Predicted vs. Measured TX rates for the 10 promoters further characterized by Hossain et al. using RT-qPCR. (C), Predicted vs. Measured TX rates for the 10 promoters further characterized by Hossain et al. using read-based measurements.

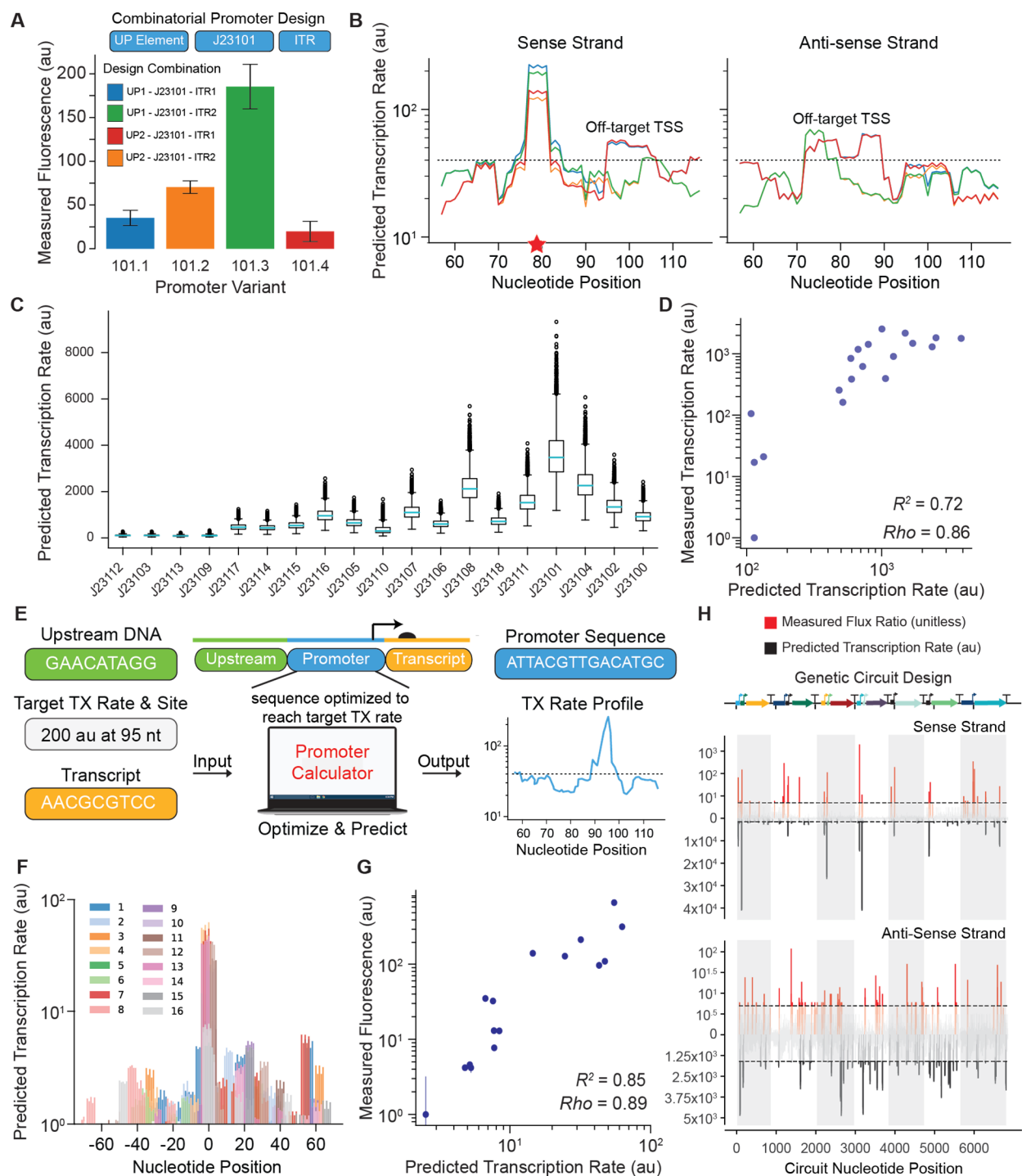


**Figure 10: Overcoming Context Effects to Design and Debug Transcriptional Profiles** (A) Measured *in vivo* transcription rates for the J23101 promoter genetic part with varied upstream (UP) and downstream (ITR) sequences, showing transcriptional context effects. (B) Transcription rate profiles are predicted for each J23101 promoter variant, showing predominant and off-target transcription start sites for the (left) sense and (right) anti-sense strands. The star shows the expected start site. (C) Transcriptional context effects are quantified for common promoter genetic parts, showing predicted transcription rates when varying upstream and downstream sequences. (D) Model predictions are compared to *in vivo* transcription rate measurements for these promoters when specifying their upstream and downstream sequences. (E)

Promoter sequences are automatically designed with desired transcription rates and start sites, including the upstream and downstream sequences (**F**) Predicted transcriptional profiles for 16 automatically designed promoters with varied transcription rate specifications. (**G**) Model predictions are compared to *in vivo* transcription rate measurements for the 16 designed promoters. Dots and bars are the mean and standard deviation from duplicate measurements. (**H**) Predicted  $\sigma 70$  transcriptional profiles of an 11-promoter genetic circuit [3] are compared to *in vivo* transcription rate and start site measurements on the (top) sense and (bottom) anti-sense strand. Transcription flux ratios are the measured differences in adjacent mRNA levels from RNA-Seq. White and grey shadows correspond to each transcribed cistron. The dotted black line shows the minimum transcription rates that define a start site.

Across microbial biotechnology, promoters are routinely treated as reusable, swappable genetic parts to control protein expression levels [19]. However, the model predicts that the DNA sequences surrounding a promoter will greatly affect its TX rate, specifically the UP and ITR regions, which would explain why some promoters express certain proteins at much lower levels. We next tested the model's ability to predict, explain, and overcome these genetic context effects. First, we designed UP and ITR sequences that the model predicted would greatly alter a promoter's TX rate, inserted them around the commonly used J23101 promoter, and measured their *in vivo* TX rates using a fluorescent protein reporter inside *E. coli* cells. Changing only the UP region or only the ITR region reduced the promoter's TX rate by up to 5.3-fold, while changing both regions reduced the promoter's TX rate by up to 9.4-fold (**Figure 10A**). The model predicted that these modifications led to the emergence of off-target TSSs that may interfere with transcriptional initiation [21] (**Figure 10B**). We next evaluated whether these genetic context effects are potentially widespread by carrying out Monte Carlo simulations to predict the range of TX rates expected when varying the UP and ITR regions of several commonly used promoters (**Figure 10C**). Promoters with higher average TX rates exhibited much higher sensitivity to changing the UP and ITR regions with an overall average coefficient of variation of 0.40. We then tested whether the model is able to correctly account for genetic context effects and predict the promoters' TX rates in a particular context. To do this, we selected a classic community dataset whereby the *in vivo* TX rates of these commonly used promoters were previously characterized inside *E. coli* cells and within a specific genetic context [17]. We then inputted the specific UP, core promoter, and ITR sequences from this dataset into our model and confirmed accurate TX rate predictions ( $R^2 = 0.72$ ,  $r = 0.86$ , **Figure 10D**), further illustrating the model's generality.

Next, we developed an automated optimization algorithm that designs promoter sequences with desired transcriptional profiles, taking into account both the upstream DNA sequence and the transcribed mRNA sequence (**Figure 10E**). The algorithm identifies optimal promoter sequences that provide user-defined TX rates at desired TSSs, while minimizing the TX rate at undesired, off-target TSSs. We tested this algorithm by designing, constructing, and characterizing 16 synthetic promoter sequences with targeted, systematically varied TX rates. The designed promoters were 77 base pairs long and shared only 26% sequence similarity (average pairwise hamming distance of 57). The model predicted single predominant TSSs at the desired locations, though it becomes progressively more difficult to achieve single-TSS predominance as the targeted TX rate is lowered (**Figure 10F**). We then measured the promoters' TX rates, each expressing a ribozyme-insulated transcript on a multi-copy plasmid inside *E. coli*

cells during exponential growth, and found that the model's predictions were highly accurate ( $R^2 = 0.85$ ,  $r = 0.89$ , **Figure 10G**), enabling fine control of site-specific transcription rates over a 682-fold range, while taking into account the surrounding genetic context.

Finally, we demonstrated how the statistical thermodynamic model facilitates the debugging of large genetic circuits by predicting their transcriptional profiles and identifying cryptic promoters that could disrupt the circuit's function. As a demonstration, we selected a recently characterized genetic circuit that uses 11 engineered promoters and 7 transcription factors to carry out digital logic [3] (**Figure 10H**). The model predicted single-TSS peaks for 9 of the engineered promoters and identified 51 cryptic promoters (18 sense and 33 anti-sense). These predictions were qualitatively confirmed by system-wide RNA-Seq measurements and transcriptional flux inferences with an overall accuracy of 55% for cryptic promoter identification. Future design software can leverage these predictions to redesign protein coding regions and remove cryptic promoters before genetic circuits are constructed.

In conclusion, we created a parsimonious (human-understandable) biophysical model of bacterial transcription initiation with demonstrated accuracy across thousands of  $\sigma 70$  promoters with highly diverse sequences, enabling site-specific control over TX rates in engineered genetic systems. Our model shows how multiple weak interactions contribute to RNAP/ $\sigma 70$  transcriptional control, including its promiscuous activity on DNA sequences without cognate binding motifs [40]. Our bottom-up model-building approach is readily extendable to other RNAP/ $\sigma$  complexes and demonstrates how advances in machine learning can enhance, rather than replace, existing thermodynamic formalisms with the overall goal of creating a universal system-wide language for engineering gene regulation in synthetic genetic systems.

## 4.2 An automated model test system for systematic development and improvement of gene expression models

It has been a grand challenge to transition synthetic biology into a mature engineering discipline, where genetic systems (e.g. biosensors [41-43], genetic circuits [44-46], and metabolic pathways [47-49]) are reliably designed, built, and tested to reprogram cellular behavior with desired outcomes. Quantitative models play a central role in synthetic biology's design-build-test cycle by predicting the function of a candidate genetic system, before it's constructed, and therefore reduce trial-and-error experimentation. Gene expression models have been developed to predict the sequence-function relationships for several gene regulatory parts [42, 50-55], enabling the automated design of genetic systems with desired protein expression levels. Gene expression predictions have also been combined with system-level models of genetic circuits and metabolic pathways to predict how changes in system architecture, component expression levels, and host genome control organismal sensing, decision-making and the biosynthesis of desired chemicals [48, 56-59]. Overall, more accurate gene expression models are necessary to correctly build genetic systems with many interacting components [60].

In mature engineering disciplines, automated test systems are routinely used to verify that models, design algorithms, and software systems generate predictions and outcomes within specified performance requirements [61]. Model test systems are run whenever an existing model is modified or when new models are proposed to ensure consistent improvements in accuracy across the widest possible range of inputs and to verify identical predictions across different software implementations. Model test systems also accelerate the discovery of new interactions by identifying the factors that contribute to model error. Within the life sciences, the CASP [62], DREAM [63], and IMPROVER [64] competitions have served a somewhat analogous purpose, where researchers are challenged to apply computational modeling to solve complex problems, for example, predicting protein structure from sequence, identifying disease genetic traits, and reverse-engineering gene regulatory networks. A key theme of these contests is that truly novel mechanisms are far more discoverable once state-of-the-art models are challenged to predict the outcome of a large and diverse experimental dataset.

Here, we present the first automated test system for gene expression models and apply it on a compiled database of 9862 characterized genetic systems with highly diverse DNA sequences and measured functions in diverse host organisms. Beyond statistical analysis, we use information theory to critically assess the amount of information added by a model's predictions, automated model testing to accept or reject a large set of mechanistic hypotheses, and machine learning to identify new sequence determinants and mechanisms. We demonstrate these capabilities on six sequence-to-function models of bacterial translation initiation [5, 6, 65-68], providing the first systematic test of their accuracies, and explaining the mechanistic origins of the models' error. Ultimately, we identified several mechanisms missing from the state-of-the-art model, which led to the development of a novel model of bacterial translation initiation with significant improvements in accuracy. Overall, we show that the model test system accelerates the systematic development of improved gene expression models.

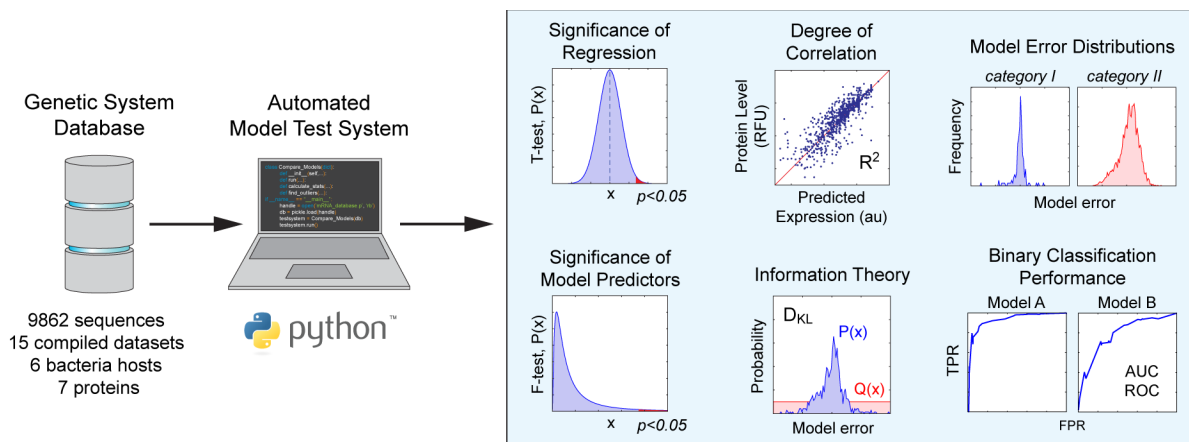


**Figure 11 | The inputs and outputs of the model test system.** The database contains characterized genetic systems and relevant metadata, including their genetic part sequences, host organism specifications, experimental conditions, and publication source in machine- and human-readable formats. The model test system systematically evaluates the predictions of proposed models and returns each

model's accuracy statistics, error distributions, and information theory metrics. These calculations are then automatically utilized to evaluate hypotheses, categorize sequences, and identify sources of model error.

## **An automated model test system for gene expression models**

We first compiled a database of 9862 characterized genetic systems, collected from 15 publications [4, 5, 42, 65, 67-77], whereby 7 unique heterologous reporter proteins (mRFP1, super folder GFP, CFP, YFP, LacZ, Luciferase, and NanoLuc) were expressed using engineered promoters and ribosome binding sites in 6 different bacterial hosts (*E. coli*, *B. subtilis* 168, *C. glutamicum*, *S. typhimurium* LT2, *B. thetaiotaomicron*, and *P. fluorescens*). The genetic systems' reporter expression levels were individually characterized using flow cytometry or spectrophotometry, or using FlowSeq, a technique that uses binned cell sorting, bar-coding, and next-generation sequencing to measure the transcription and translation rates of a library of genetic systems [4]. For FlowSeq measurements, we applied automatic filters to remove measurements with insufficient or skewed read counts. We then grouped together genetic systems first by their characterization method and second by their commonalities (same promoter, same reporter protein, and same host) to create two high-quality databases: 1014 individually characterized genetic systems (1014IC) with 22 sub-groups and 8848 FlowSeq characterized genetic systems (8848FS) from a single experiment.

We then developed an automated model test system that reads each genetic system database and uses its sequence information to predict the genetic systems' protein expression levels, using one of many selectable drop-in quantitative gene expression models (**Figure 11**). These gene expression models were developed with the expectation of a linear relationship between the predicted and measured expression levels, though the relationship's proportionality constant will vary by sub-group, for example, due to the utilization of different reporter proteins in different host organisms. We therefore developed an approach to automatically identify the correct proportionality constant for each sub-group, utilizing median absolute deviation (MAD) criteria to reject outliers and five common statistical metrics (**Table 5**) to quantify each model's overall accuracy and precision. Motivated by our observations below, we then augmented the model test system to perform a more advanced information theoretic analysis that additionally accounts for the diversity of the genetic systems' sequences and the precision of the experimental measurement techniques. By calculating both commonly used and newly proposed metrics for model accuracy, the model test system enables researchers to view each metric with the necessary context to make appropriate comparisons.

**Table 5** | The model test system accuracy and precision metrics for seven models of bacterial translation initiation rate, evaluated on the 1014IC and 8848FS datasets. The range for  $R^2$  and the normalized Kullback-Leibler divergence ( $D_{KL}$ ) is (0,1). The range for AUROC is (0.5,1). The root mean square error (RMSE) is always positive. The model's one-sided error cumulative distribution function is provided at selected thresholds. NP is the number of genetic systems where the model is not capable of calculating a prediction.

| 1014 Individually characterized (1014IC) |       |      |          |       |           | % within X-fold error |    |    |     |
|------------------------------------------|-------|------|----------|-------|-----------|-----------------------|----|----|-----|
| Model                                    | $R^2$ | RMSE | $D_{KL}$ | AUROC | MC (bits) | 2                     | 5  | 10 | NP  |
| RBS Calculator v1.0                      | 0.51  | 1.75 | 0.41     | 0.85  | 3707      | 35                    | 66 | 79 | 66  |
| RBS Calculator v1.1                      | 0.66  | 1.51 | 0.46     | 0.89  | 5911      | 44                    | 77 | 88 | 23  |
| RBS Calculator v2.0                      | 0.71  | 1.39 | 0.48     | 0.93  | 7313      | 50                    | 81 | 90 | 23  |
| RBS Calculator v2.1                      | 0.74  | 1.31 | 0.49     | 0.93  | 7857      | 49                    | 82 | 91 | 0   |
| UTR Designer                             | 0.50  | 1.77 | 0.43     | 0.85  | 4490      | 38                    | 69 | 80 | 66  |
| RBS Designer                             | 0.42  | 1.90 | 0.35     | 0.85  | 2385      | 33                    | 57 | 69 | 101 |
| EMOPEC                                   | 0.31  | 2.14 | 0.40     | 0.76  | 3053      | 31                    | 64 | 77 | 1   |
| 8848 FlowSeq characterized (8848FS)      |       |      |          |       |           | % within X-fold error |    |    |     |
| Model                                    | $R^2$ | RMSE | $D_{KL}$ | AUROC | MC (bits) | 2                     | 5  | 10 | NP  |
| RBS Calculator v1.0                      | 0.25  | 1.03 | 0.21     | 0.80  | 1851      | 39                    | 74 | 88 | 18  |
| RBS Calculator v1.1                      | 0.26  | 1.03 | 0.22     | 0.80  | 1822      | 40                    | 75 | 88 | 0   |
| RBS Calculator v2.0                      | 0.26  | 1.02 | 0.22     | 0.78  | 1421      | 36                    | 71 | 86 | 0   |
| RBS Calculator v2.1                      | 0.28  | 1.01 | 0.22     | 0.78  | 1982      | 40                    | 78 | 92 | 0   |
| UTR Designer                             | 0.23  | 1.04 | 0.21     | 0.80  | 2082      | 42                    | 79 | 91 | 19  |
| RBS Designer                             | 0.23  | 1.05 | 0.27     | 0.73  | 511       | 23                    | 52 | 70 | 0   |
| EMOPEC                                   | 0.09  | 1.14 | 0.21     | 0.64  | 2782      | 47                    | 86 | 95 | 1   |

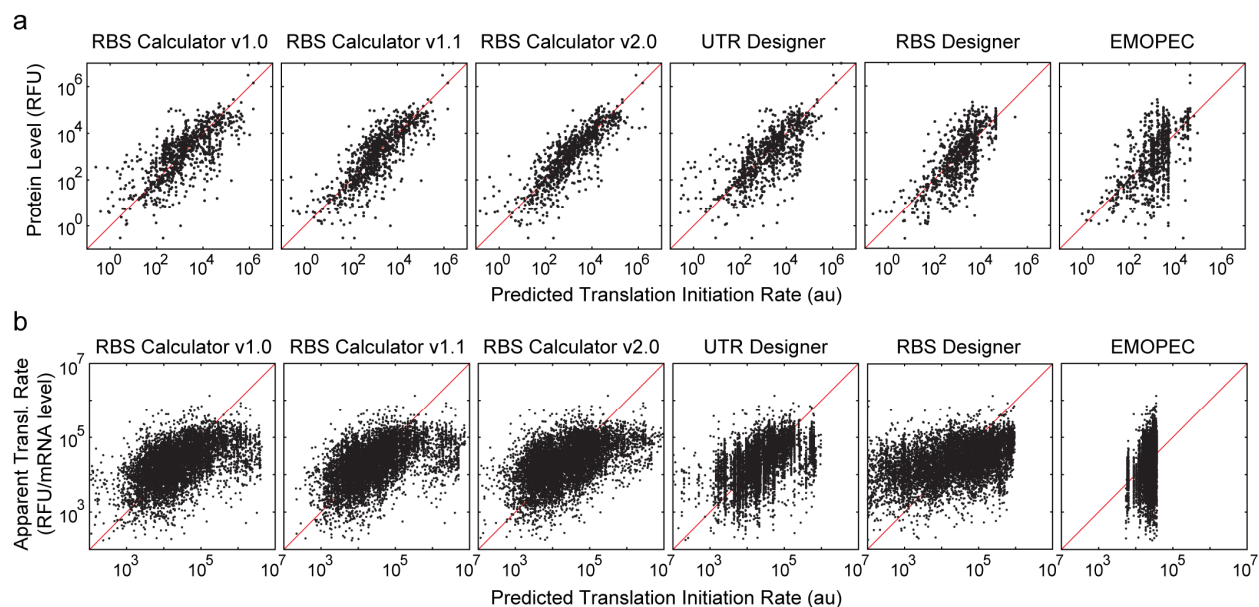


**Figure 12 | Systematic assessment of six sequence-to-function models** of bacterial translation initiation rate evaluated on the (a) 1014IC and (b) 8848FS datasets. Test metrics are listed in **Table 5**.

## A systematic comparison of translation initiation rate models

Several known ribosome-mRNA interactions work together to control an mRNA's translation initiation rate, including preliminary binding of the 30S ribosomal subunit to upstream standby sites in the mRNA (interaction #1) [68, 78], unfolding of inhibitory mRNA structures (interaction #2) [79], hybridization between the ribosome's 16S rRNA and the mRNA at the Shine-Dalgarno sequence (interaction #3) [80], hybridization between the tRNA<sup>fMet</sup> and start codon (interaction #4) [81], and entropic stretching or compression of the ribosome caused by non-optimal distances between the Shine-Dalgarno and start codon (interaction #5) [82]. However, there is considerable debate over the importance of these interactions, the best way to calculate their strengths, and whether there are additional interactions that control translation rate. Thus, several models of translation initiation have been proposed with both conceptual and implementation differences.

Using the automated model test system, we evaluated the accuracies and sources of error for six different models of bacterial translation initiation rate [5, 6, 65-68] on both the 1014IC and the 8848FS datasets. The RBS Designer [66] uses dynamical systems theory to model ribosome binding, calculating the ribosome binding site's exposure probability (interaction #2) and the free energy of mRNA-rRNA hybridization (interaction #3) to predict an mRNA's translation initiation rate. The RBS Calculator (v1.0) uses statistical thermodynamics and a free energy model to calculate the ribosome's total binding free energy to an mRNA sequence, and predict the mRNA's translation initiation rate, accounting for interactions #1 to #5. The RBS Calculator v1.1 [67] uses a modified free energy model implementation that includes the unfolding of inhibitory mRNA structures within the protein's coding sequence (interaction #2) and a new approach for

calculating hybridization free energies that becomes important when mRNAs have non-canonical Shine-Dalgarno sequences (interaction #3). The RBS Calculator v2.0 [68] incorporates new biophysical rules governing the ribosome's interactions at mRNA standby sites, determined through characterization of rationally designed mRNAs (interaction #1). The UTR Designer [6] is a modified version of the RBS Calculator v1.0 that introduces a direct and indirect pathway to unfolding inhibitory mRNA structures (interaction #2). EMOPEC [5] is an empirical model that uses a look-up sequence table of measured hybridization free energies (interaction #3) and the model of non-optimal spacing from RBS Calculator v1.0 (interaction #5) to predict an mRNA's translation initiation rate.

The model test system revealed clear differences in model accuracies and highlighted the utility of applying multiple test metrics to evaluate model performance (**Table 5**). When evaluated on the 1014IC dataset, the successive modifications to the RBS Calculator model resulted in accuracy improvements that were reflected across all test metrics. To compare, the test metrics largely agreed that the UTR Designer, RBS Designer, and EMOPEC models have lower accuracies (**Figure 12A**). Overall, the RBS Calculator v2.0 model could predict 50% of the 1014IC mRNAs' translation initiation rates within 2-fold and 90% within 10-fold with extremely high classification accuracy, an AUROC of 0.93. Notably, all models had variable accuracies on data sub-groups utilizing different reporters, measurement techniques, and host organisms, motivating the further development of automated testing strategies.

When evaluated on the 8848FS dataset, however, there was an overall disagreement among the test metrics over the accuracies of each model (**Figure 12B**, **Table 5**). The Pearson  $R^2$ ,  $D_{KL}$ , and AUROC metrics show that all six models had similarly sharp reductions in accuracy with only slight differences between them, compared to lower RMSEs indicative of higher accuracy. Notably, these metrics also had a poor correspondence to the models' error distributions. In particular, while EMOPEC's test metrics showed the lowest model accuracy, its model error distribution had the highest percentage of well-predicted sequences, clearly contradicted by a comparative graphical analysis (**Figure 12B**). Overall, these results show that once experimental datasets grow beyond a certain size, current approaches to quantifying a model's accuracy strongly depend on the chosen test metric, creating a subjective choice in what should be an objective process.

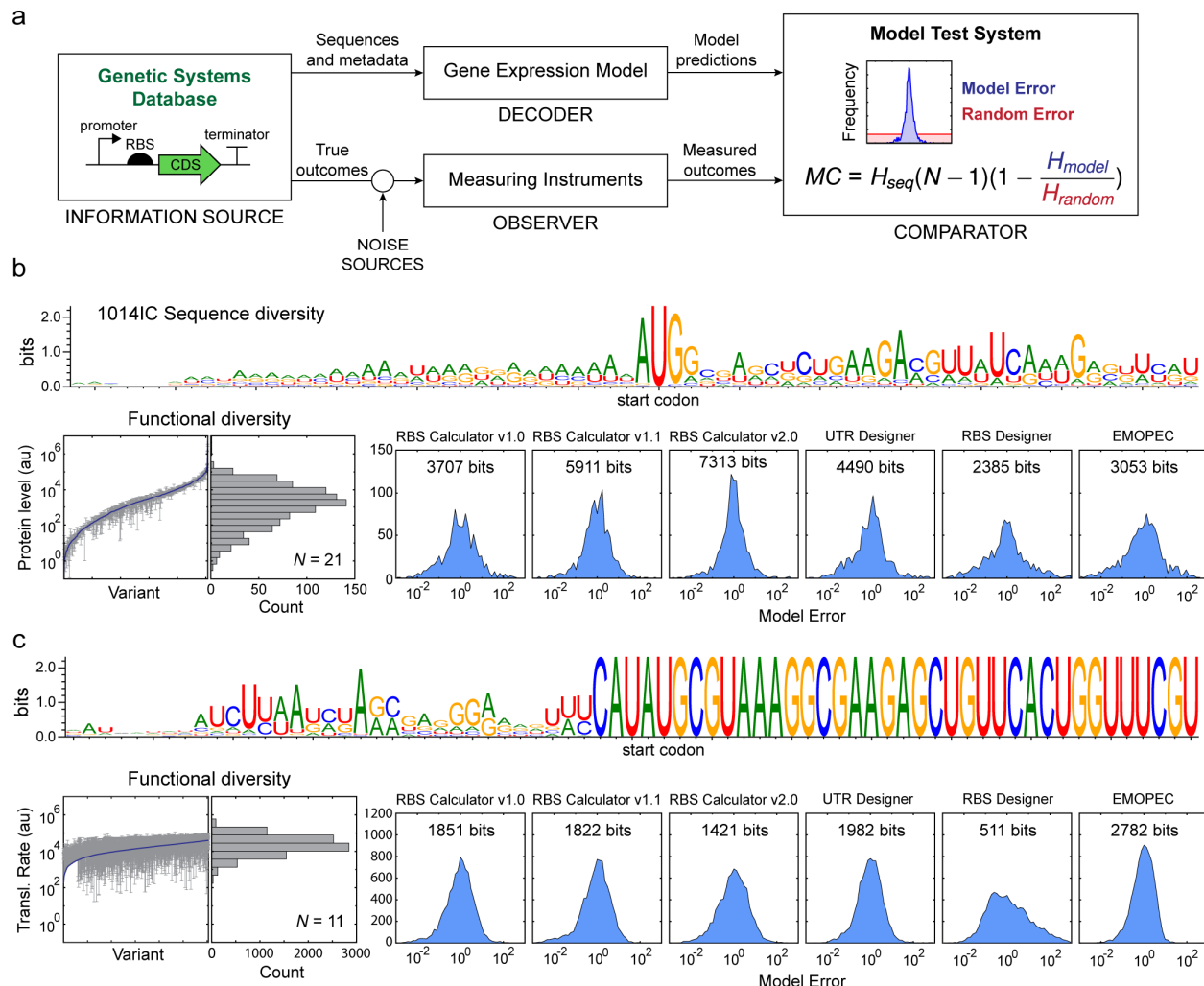


**Figure 13 | Information theory to quantify dataset uncertainty and a model's capacity ( $MC$ ) to predict outcomes.** (a) A schematic showing the flow of information from a list of genetic systems (the source) to the model test system (the comparator), using digital systems to predict outcomes (the decoder) and the physical systems to measure outcomes (the observer).  $N$  is the number of distinguishable measurement outcomes.  $MC$  is the reduction in uncertainty from information source to model error comparator. The sequence logos [83], functional diversities, error distributions, and model capacities for (b) the 1014IC and (c) the 8848FS dataset.

## Information theory to determine model capacity

Using smaller scale examples, we first investigated how the same model evaluated on different data sub-groups had substantially different apparent accuracies. For example, the EMOPEC model could accurately predict the gene expression levels of a 10-sequence *E. coli*-LacZ sub-group ( $R^2 = 0.85$ ) where mutation of only 3 nucleotide positions resulted in a 2400-fold change in translation rate. Similarly, the RBS Calculator v2.0 model was extremely accurate when evaluated on a 14-sequence *P. fluorescens*-mRFP1 sub-group ( $R^2 = 0.90$ ) where mutation of 4 nucleotide positions resulted in a 10900-fold change in translation rate. However, even though these models tested well on

datasets with a small number of nucleotide mutations, they did not achieve similar accuracies on larger, more diverse datasets (**Table 5**). These results inspired us to measure the difficulty of correctly predicting the expression levels of a genetic system dataset. From an information theoretic perspective, a model's accuracy is determined by its ability to maximally reduce uncertainty in the predicted outcomes. It is therefore equally important to consider both the starting amount of uncertainty in the dataset and the ending amount of uncertainty in the gene expression level predictions.

We therefore derived a new test metric, called Model Capacity (*MC*), which quantifies a model's ability to accurately predict gene expression levels by tracking the flow of information from source to comparator and measuring the model's ability to reduce uncertainty in the predicted outcomes, taking into account the genetic system dataset's sequence diversity, the experimental measurements' functional diversity, and the model's error distribution (**Figure 13A**). To calculate *MC*, we quantify a dataset's sequence diversity ( $H_{\text{seq}}$ ) utilizing Shannon's entropy, we determine a dataset's number of distinguishable outcomes ( $N$ ) by dividing the measurements' dynamic range by its average precision, and we compute the Shannon entropy of the model's error distribution, compared to a random model, both using  $N$  outcomes. Overall, our newly proposed *MC* metric provides an objective way to assess both the model and the dataset together, facilitating correct cross-data model comparisons.

For example, in agreement with currently used test metrics, the successive improvements to the RBS Calculator model led to higher *MC*s of 3707, 5911, and 7313 bits when evaluated on 1014IC. The *MC*s for the UTR Designer, EMOPEC, and RBS Designer models also closely tracked the other test metrics (4490, 3053, and 2385 bits, respectively) on this dataset (**Figure 13B**). However, only the *MC* metric can assess whether a dataset has sufficient sequence and functional diversity to truly test a model's accuracy. For example, both the RBS Calculator v2.0 and EMOPEC models have very low *MC*s when evaluated on the 14-sequence *P. fluorescens*-mRFP1 data sub-group (33 vs. 31 bits, respectively). Similarly, the RBS Calculator v2.0 and EMOPEC models have even lower *MC*s when evaluated on the 10-sequence *E. coli*-LacZ data sub-group (14 vs. 9 bits, respectively). The *MC* metric correctly assessed that these small datasets are too homogenous to provide a distinguishable comparison in model accuracy; the relative differences in *MC* are too small, compared to the absolute *MC* values possible when evaluated on the more diverse 1014IC dataset.

Notably, a dataset with more sequences does not always lead to higher sequence diversity. For example, the smaller 1014IC dataset has a larger sequence diversity ( $H_{\text{seq}} = 2243$  bits), compared to 8848FS ( $H_{\text{seq}} = 1448$  bits), because it contains longer 5' UTRs with mutations at more distributed positions (**Figure 13BC**). Moreover, different characterization techniques can also have substantially different dynamic ranges and measurement precisions. For example, while the FlowSeq technique simultaneously measures gene expression levels from thousands of sequences, the current approach sorts fluorescent cells into a relatively small number of defined bins, resulting in a smaller number of distinguishable outcomes, compared to measuring reporter expression levels within isogenic cultures ( $N = 11$  for FS8848 vs.  $N = 21$  for 1014IC). Together,  $H_{\text{SEQ}}$  and

$N$  measure the maximum possible uncertainty in a dataset, and therefore control the maximum possible  $MC$  for a perfect model evaluated on that dataset ( $MC_{\max}$  of 47100 bits for 1014IC, and 14476 bits for 8848FS). As a result, even the most accurate model ( $MC$ ), when evaluated on 8848FS (EMOPEC) has a lower  $MC$ , compared to its evaluation on 1014IC (2782 vs 3053 bits respectively), a symptom of its limited sequence and functional diversity. Future efforts to test model predictions should therefore seek to design and characterize datasets with higher  $MC_{\max}$ .

## **Automated hypothesis testing to evaluate mechanisms and model implementations**

Beyond assessments of model accuracies, we next expanded our model test system to automatically apply statistical analysis to accept or reject proposed mechanistic hypotheses. Here, we demonstrate that capability by evaluating 100 mechanistic hypotheses covering the wide-ranging debate over the sequence determinants that control translation initiation rate. Each hypothesis was first automatically converted into a distinct model implementation, followed by evaluating its predictions on the 1014IC dataset. Hypotheses were accepted only when an  $F$ -test identified that the model implementation's error distribution had a statistically significant reduction in variance ( $F$ -test  $p < 0.01$ ) and when its accuracy test metrics were improved (higher  $R^2$  and  $MC$ ), compared to the RBS Calculator v2.0 baseline model.

Surprisingly, when proposed as hypotheses, several commonly accepted anecdotes were rejected. It has been generally accepted that the ribosome binding site, and specifically, the Shine-Dalgarno sequence, is 6 nucleotides long, suggesting that only these nucleotides hybridize with the 3' end of the 16S ribosomal RNA (SD: 5'-AGGAGG-3', interaction #3). However, by constructing model implementations with 55 different possible 16S rRNA sequences, covering all possible Shine-Dalgarno sequences from 4 to 13 nucleotides long, our analysis revealed that the *E. coli* Shine-Dalgarno sequence is 9 nucleotides long (SD: 5'-UAAGGAGGU-3',  $R^2 = 0.70$ ,  $F = 0.85$ ,  $p = 0.013$ ). Several studies have also claimed that only the Shine-Dalgarno interaction or only the presence of mRNA structures are responsible for controlling a mRNA's translation rate. Instead, a recursive feature reduction method showed that both interactions must be considered to accurately predict translation rates. A model that only incorporates Shine-Dalgarno interactions does not accurately predict translation rates ( $R^2 = 0.54$ ,  $F = 3.3$ ,  $p = 3.6 \times 10^{-75}$ ). Likewise, a model that only quantifies the energy needed to unfold mRNA structures is not accurate either ( $R^2 = 0.44$ ,  $F = 6.2$ ,  $p = 4.6 \times 10^{-159}$ ). Here, the model test system demonstrated its ability to reject false claims with extremely high statistical significance.

Separately, translation rate model predictions have relied on the accurate calculation of RNA folding free energies and structures, though it is not known how these calculations affect the translation model's accuracy. Several options are available; translation rate models may use different RNA free energy parameters, they may calculate either a minimum-free-energy (mfe) or ensemble centroid RNA structure, and they may incorporate additional energetic contributions, such as dangling nucleotide interactions. We therefore formulated and evaluated all combinations of these options as

hypotheses. We found that utilizing the ensemble centroid RNA structure ( $R^2 = 0.58$ ,  $F = 4.7$ ,  $p < 1.8 \times 10^{-119}$ ) or incorporating the dangling nucleotide interaction ( $R^2 = 0.69$ ,  $F = 1.2$ ,  $p = 0.002$ ) resulted in less accurate predictions. We also found that the most recently developed RNA free energy parameter set [84] provided a statistically significant improvement in model accuracy ( $R^2 = 0.72$ ,  $F = 1.2$ ,  $p = 0.008$ ).

Importantly, the model test system also determines when a proposed hypothesis is not testable on a given dataset. If a proposed mechanism or interaction is predicted to have the same, or similar, effect on all predicted outcomes within a data sub-group, then accepting or rejecting the hypothesis is not possible. We found two interesting examples of such hypotheses that could motivate future efforts. First proposed in 1990, the downstream-box (DB) hypothesis suggests that a portion of an mRNA's CDS can hybridize to the 16S rRNA<sub>1469:1483</sub> in *E. coli* to accelerate its translation initiation rate [85]. This hypothesis was untestable because all data sub-groups share similar CDS regions, and therefore no matter how the interaction is proposed to take place, the predicted translation initiation rates within each data sub-group are similarly scaled. Second, it has been proposed that some organisms, including *Bacteroides thetaiotaomicron*, do not utilize a Shine-Dalgarno sequence to initiate translation [71, 86]. We therefore evaluated the importance of the Shine-Dalgarno (interaction #3) when evaluated on the 143 mRNAs characterized in *B. thetaiotaomicron* and found that model accuracies were nearly identical with and without this interaction ( $R^2 = 0.51$  vs.  $0.49$ ,  $F = 1.05$ ,  $p = 0.39$ ). Here, the 143 mRNAs do not have sufficient sequence and functional diversity to test this hypothesis; specifically, the maximum change in interaction #3's energy was relatively small, compared to the overall 1014IC dataset (4.2 vs. 18.6 kcal/mol). Overall, this capability of the model test system can accelerate mechanistic discoveries by directing the design and characterization of datasets to test previously untestable hypotheses.

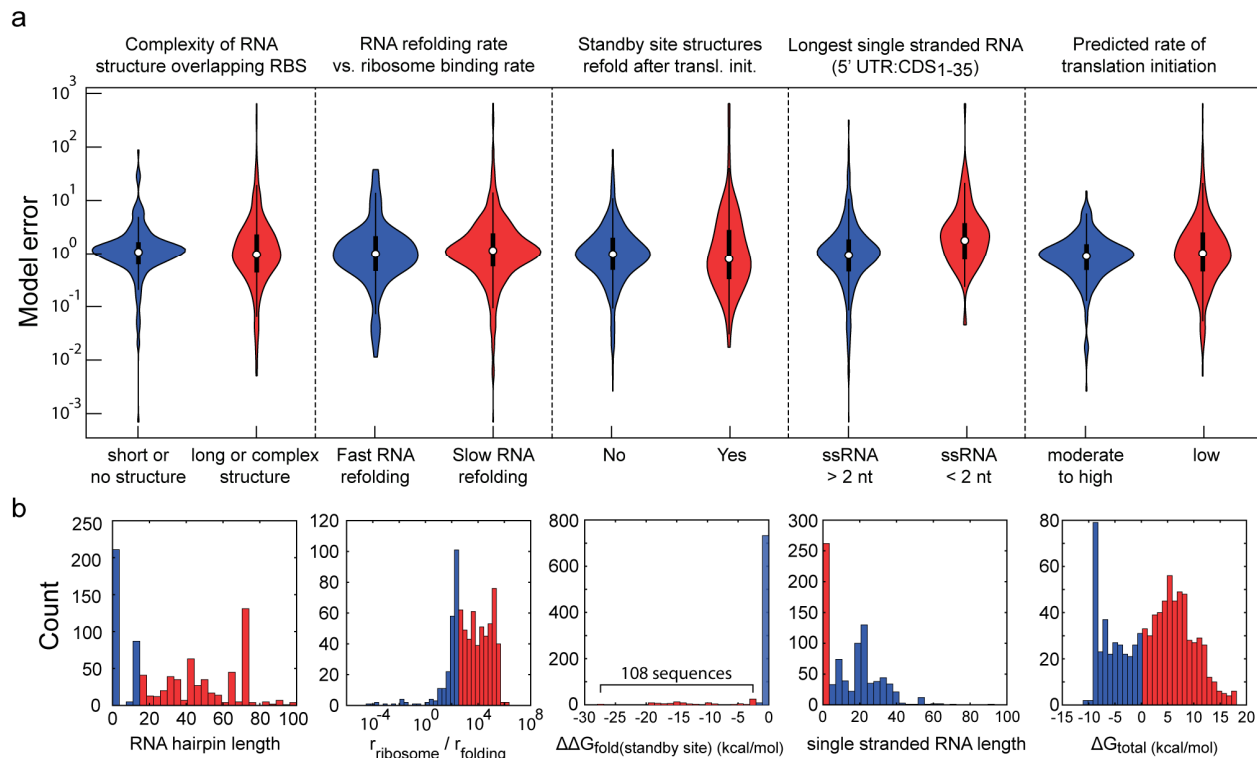


**Figure 14 | Identifying the sources of model error.** (a) Violin plots comparing the error distributions from RBS Calculator v2.0 model predictions across five selected categories of mRNA sequences as shown. (b) Histograms illustrating the thresholds and counts of sequences for each of the categories as shown in (a).

## Sequence-function-error associations to identify new gene expression determinants

Once a gene expression model encompasses all well-characterized interactions controlling gene expression, it can become a significant challenge to discover and describe novel gene regulatory mechanisms that control gene expression levels. By combining the model test system with machine learning techniques, we next demonstrated how to systematically discover poorly characterized interactions controlling gene expression levels within the 1014IC dataset. We first enumerated a long list of candidate biophysical, biochemical, and geometric properties that can all be evaluated from sequence information and have the potential to influence gene expression. Our list included the characteristics of calculated structures of portions of mRNA, including the lengths, nucleotide compositions, and folding free energies of key structural features (hairpin loops, bulges, mismatches, single-stranded regions); the kinetic folding rates of these RNA structures as determined by the algorithm KFOLD [87]; and calculated characteristics of the ribosome-mRNA interaction, such as the structures of upstream standby sites [68] and the compositions of the spacer region [76]. The model test system then uses a feature selection method to automatically iterate over all candidate properties to identify ones where changes in the sequence-function relationship are associated with a statistically significant increase in the gene expression model's error, here using the RBS Calculator v2.0 as the baseline. Overall, we identified five properties associated with

increased model error that have sufficient statistical significance to warrant further investigation ( $p < 0.05$ ) (**Figure 14**).

First, we found that long mRNA structures (>29 nucleotides long) that overlapped with the Shine-Dalgarno sequence greatly reduced the model's ability to correctly predict the mRNA's translation rate ( $F = 43.0$ ,  $p = 2.8 \times 10^{-210}$ ). These structures are inhibitory because they must be unfolded before the ribosome can initiate translation, and therefore any miscalculation in their unfolding free energies will have a large effect on the translation rate prediction. Relatedly, we found that whenever inhibitory mRNA structures required a large amount of time to fold (>322 KFOLD time units), or when the ribosome's binding rate was faster than the RNA structure's refolding rate, the model's predictions were significantly less accurate ( $F = 12.3$ ,  $p = 5.5 \times 10^{-97}$ ; and  $F = 40.1$ ,  $p = 3.2 \times 10^{-115}$  respectively). We also found that when standby site structures were predicted to refold to a lower energetic state following ribosome binding, then the model's predictive accuracy was significantly reduced ( $F = 126.6$ ,  $p = 8.4 \times 10^{-79}$ ). During the cycling of translation initiation, it is possible that these structures do not have sufficient time to refold to the minimum free energy structure, or are kinetically trapped.

Finally, we found that highly structured 5' untranslated regions with fewer than 2 consecutive single-stranded RNA nucleotides had a significant confounding effect on predicted translation rates ( $F = 22.5$ ,  $p = 5.0 \times 10^{-181}$ ). Single-stranded RNA regions are potential binding sites for RNase E, and therefore the absence of such sites could increase mRNA stability and affect gene expression levels [88]. Similarly, we found that whenever mRNAs had low predicted translation initiation rates, then the model also had higher error ( $F = 450$ ,  $p < 10^{-100}$ ), in agreement with evidence that poorly translated mRNAs become unprotected by ribosomes and therefore are more likely to be degraded by RNase E [89]. Altogether, the model test system identified 671 mRNAs (66% of the 1014IC dataset) where the changes in mRNA degradation and stability have confounded the translation rate model's predictions, explaining a key source of model error.

## **The Model Test System Enables Systematic Development of Improved Models**

We then leveraged the compiled genetic system database, confirmed mechanistic hypotheses, and identified sources of model error to develop and validate a new version of the RBS Calculator free energy model (v2.1) with a significant improvement in accuracy. The model test system enabled us to rapidly propose and evaluate new formulas for quantifying interaction energies that control ribosome binding and translation initiation rate, particularly new interactions that only became apparent when evaluated on a large genetic system database. Specifically, we added a new free energy term ( $\Delta G_{\text{stack}}$ ) to quantify stacking interactions between adjacent RNA nucleotides, which had a large effect on ribosome binding sites with homopolymer sequences [76] (**Supplementary Figure 5**); we improved the formulas that quantify how ribosomes in gram-positive organisms bind to structured standby sites and stretch/compress with varying ribosome binding site spacer lengths [90]; we improved the calculation of apparent tRNA<sup>fMet</sup> binding free energies at non-canonical start codons, using 35 recently characterized mRNAs [77] and we updated the translation rate model's RNA free energy parameters to the

Andronescu2007 set [84]. Importantly, while these improvements relied on characterization of mRNA translation rates using different plasmid or genomic copy numbers, promoters, reporters, and measurement techniques, the model test system could correctly extract the maximum amount of information from the measurements to parameterize and validate model predictions. The resulting RBS Calculator v2.1 model predictions were more accurate than v2.0 across the entire 1014IC dataset ( $R^2 = 0.74$  vs.  $0.71$ , MC = 7857 vs. 7312 bits) with a statistically significant reduction in the model error distribution's variance ( $F = 0.74$ ,  $p = 9.8 \times 10^{-7}$ ) and a smaller number of calculation errors on non-canonical sequences (0 vs. 23).

Beyond these improvements, the model test system identified additional confounding factors, such as changes in mRNA folding kinetics and mRNA degradation rates, that are not considered in an equilibrium thermodynamic model of translation initiation rate, but nonetheless increase its model error. To avoid these sources of error, we then utilized the model test system's categorization of sequence-error associations (**Figure 14**) to propose a safe operating zone where RBS sequences could be more accurately designed using the model. Using the mRNA sequence alone, a model prediction is considered outside the safe operating zone when the mRNA contains a long or slow-folding RNA structure ( $>29$  nucleotides long or  $>320$  KFOLD time units), when standby site structures are predicted to refold ( $\Delta\Delta G_{\text{fold, standby site}} > 0$ ), when mRNAs contain long single-stranded RNA regions ( $>5$  nucleotides), and when the predicted translation initiation rate is low, resulting in greater deprotection of mRNAs ( $\Delta G_{\text{total}} > 0$ ). Altogether, the RBS Calculator v2.1 model's accuracy was significantly improved when restricting its predictions to sequences within the safe operating zone. The model predicted the translation rates of 64%, 91% and 95% mRNAs to within 2-fold, 5-fold, and 10-fold of measurements, respectively, and with a statistically significant reduction in the model error distribution's variance ( $F = 0.47$ ,  $p = 9.9 \times 10^{-12}$ ).

The field of Synthetic Biology has considerably matured since its origins, though several cycles of designing, building, and testing organisms are still required to engineer complex genetic systems [56]. Gene expression models have greatly reduced the number of engineering cycles by designing sequences to control protein expression levels, and thereby rationally direct genetic system optimization [67]. However, more accurate gene expression models are needed to engineer complex genetic systems with many interacting components. For several reasons, it has been a significant challenge to systematically develop more accurate gene expression models. First, genetic systems are often characterized using different promoters, reporters, host organisms, and techniques, resulting in disparate datasets that require careful analysis to extract information. Second, several interactions collectively control sequence-expression relationships; correctly quantifying their strengths is a multi-variable, high dimensional problem. Finally, a model's apparent accuracy depends on the dataset used to evaluate it, and statistical test metrics often disagree on accuracy improvements.

To overcome these challenges, we created an automated model test system that extracts the maximum amount of information from large genetic system databases, systematically assesses gene expression model accuracies, evaluates the validity of

mechanistic hypotheses testing the importance of specific interactions, and identifies sequence-dependent physical properties that control a model's error (**Figure 11**). We applied the model test system to carry out the first systematic assessment of six gene expression models that predict the translation initiation rates of mRNAs, showing that models have clear differences in accuracy across five common test metrics and two large datasets (**Figure 12, Table 5**). To explain these differences, we utilized information theory to quantify the uncertainty in both the dataset and the model predictions together, resulting in the first test metric, called Model Capacity (MC), that is capable of correct cross-dataset comparisons (**Figure 13**). Using advanced statistics and machine learning techniques, we then evaluated the validity of 100 mechanistic hypotheses and identified five sources of model error to distinguish model implementations and identify areas for model improvements (**Figure 14**). Using this analysis, we then rapidly developed a new version of RBS Calculator (v2.1) that incorporates several newly quantified interactions to yield statistically significant improvements in accuracy. These improvements in model accuracy considerably reduce trial-and-error experimentation, particularly when engineering complex genetic systems. For example, when using RBS Calculator v2.1 to engineer a 30-protein system with a 5-fold targeted expression operating space, only 2 design trials per protein are needed to yield a 78% chance of first-cycle success, compared to 5% when using RBS Calculator v2.0. Similarly, the success rate is increased to 98% when using 3 design trials per protein, compared to 41%. Overall, the model test system enables the accelerated analysis of model predictions on large datasets to focus model development on poorly characterized interactions that contribute most to model error.

The model test system is open-source software (<https://github.com/reisalex/SynBioMTS>), developed with the explicit goal of making it easy to expand the genetic system database, calculate additional test metrics, assess new gene expression models, evaluate additional hypotheses, and identify new sequence-dependent properties that correlate with model error. We also created an Excel template to enable researchers to add characterized genetic systems to the growing database. Expanded genetic system databases will be made available periodically, which we envision will be used to develop and improve predictive sequence-to-function models for many additional gene regulatory mechanisms, including transcriptional regulation, termination, mRNA decay, gene expression coupling, and post-transcriptional regulation by small RNAs, riboswitches, and ribozymes.

### 4.3 The Synthesis Success Calculator: Predicting the Rapid Synthesis of DNA Fragments with Machine Learning

DNA synthesis and assembly have become a central technology in modern synthetic biology, enabling the construction of synthetic genetic systems with defined DNA sequences, including recoded genes [91], engineered proteins [92], genetic circuits, metabolic pathways and synthetic genomes [12, 93-95]. Whenever a new genetic system does not exist in nature, the synthetic DNA fragments needed to build the genetic system are commonly purchased from commercial service providers. While prices depend on the synthetic DNA fragments' length and purity as well as other business factors, here we focus solely on the amount of time needed to physically synthesize and assemble them, called the turnaround time. The turnaround time for synthetic DNA fragments is significant as it is often the rate-limiting step in a typical design-build-test-learn cycle for engineering genetic systems [12, 13]. For example, multiple synthetic DNA fragments are commonly needed to build a desired genetic system, and therefore a delay or cancellation in making one DNA fragment will affect the build time of the entire genetic system. Faster and more reliable turnaround times would therefore greatly accelerate the engineering of genetic systems.

When DNA synthesis and assembly takes place normally, turnaround times are currently about 5 to 9 business days for non-clonal DNA fragments (up to 3000 bp) and 11 to 25 business days for purified DNA fragments (up to 5000 bp). However, the presence of several sequence determinants can cause DNA synthesis and assembly to fail, yielding a mixture of low-quality synthetic DNA with several undesired byproducts. In some cases, the process conditions can be adjusted to improve DNA product yields, though with a greatly increased turnaround time. In other cases, the process yields are so low that the requested DNA fragments are never delivered. Therefore, it would be highly beneficial to identify the sequence determinants that interfere with DNA synthesis and assembly, and develop algorithmic approaches to locating them within a desired genetic system's sequence. Designing genetic systems for rapid synthesis as well as maximal function greatly reduces the turnaround times for building genetic systems.

Here, we developed and validated a machine learning classifier to quantify the effects of several sequence determinants on synthesis outcomes and to accurately predict when a DNA fragment could be synthesized with a short turnaround time. During evaluation, the classifier also identifies the locations of the sequence elements associated with DNA synthesis failure with a quantified priority for removal (**Figure 15A**). To do this, we identified 38 DNA sequence properties that could potentially interfere with DNA synthesis and assembly and compiled a database containing 1076 DNA fragment sequences with known synthesis outcomes. We then developed an automated train-validate-test pipeline to develop a random forest classifier that was both generalized and accurate. Altogether, the finalized random forest classifier, called the Synthesis Success Calculator, achieved an  $F_1$  score of 0.928 on 251 unseen DNA fragment sequences (**Figure 15B**). Surprisingly, we found only 9 sequence properties associated with a DNA fragment's synthesis success; depending on their values, there is a high chance of synthesis failure. We demonstrate the potential applications of the Synthesis Success

Calculator by recoding recalcitrant protein coding sequences using minimum amounts of synonymous codon optimization, by evaluating how much of the *E. coli* genome can be readily synthesized, and by estimating the percentage of readily synthesized plasmids in the NCBI database.

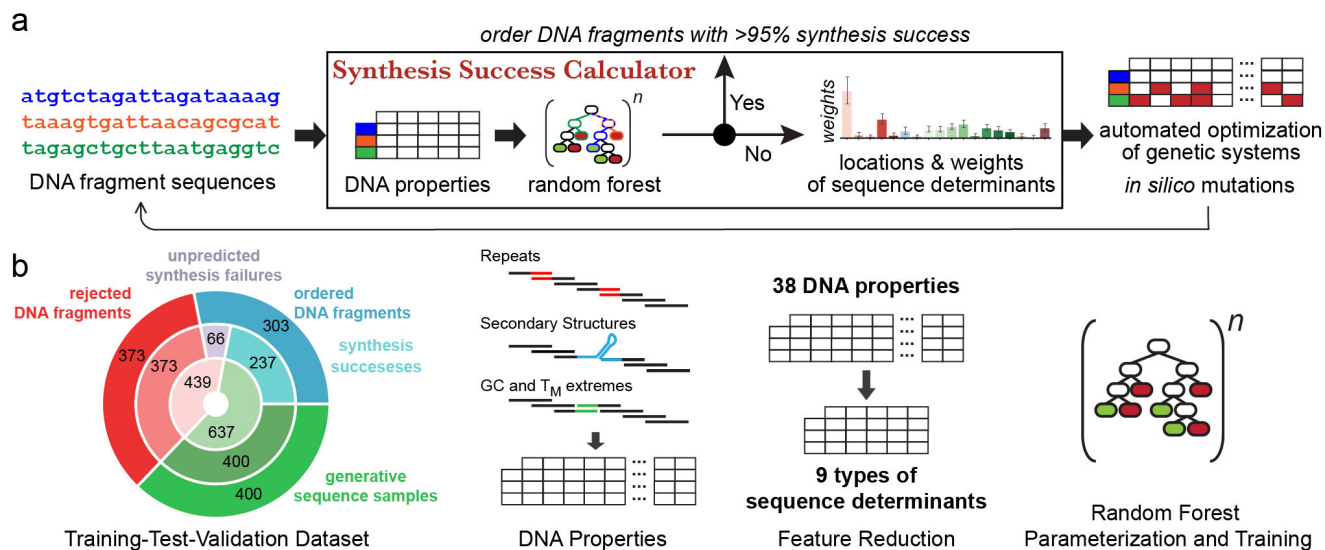


**Figure 15: The Synthesis Success Calculator** (a) The Synthesis Success Calculator uses calculated DNA properties and a random forest classifier to predict when a DNA fragment sequence can be synthesized with a fast turnaround time. It also determines the locations and importances of any sequence determinants associated with synthesis failure. Design algorithms can use this information to optimize the genetic system's sequence for rapid synthesis and maximal function. (b) We created a dataset of 1076 sequences and applied a train-validation-test protocol to parameterize the Synthesis Success Calculator, employing feature reduction to identify the sequence determinants associated with synthesis failure.

Across the synthetic biology ecosystem, algorithms have been developed to break down desired DNA fragments into the optimal set of oligonucleotides for synthesis, for example, Primerize [96], DNAWorks [97], Gene2Oligo [98], and GeneDesign [99]. Other algorithms consider how repetitive sequence elements affect the synthesis and evolutionary robustness of a genetic system [100], for example, the EFM Calculator [101]. Commercial service providers have also developed proprietary rules for evaluating synthesis complexity, which have been encoded in the Genome Calligrapher [102] and BOOST [103] algorithms. However, the Synthesis Success Calculator is distinct from previous efforts as it satisfies several community-wide requirements. First, the Synthesis Success Calculator is fully *transparent* with a non-proprietary dataset, an open-source Python implementation, and a model trained to prioritize researchers' needs first, for example, by limiting false positives. Second, it includes a pipeline for automatic retraining on any inputted sequence dataset, enabling researchers to identify new sequence determinants affecting synthesis outcomes, for example, when using future chemistries. Third, as a Python module, the Synthesis Success Calculator can be directly embedded within other genetic system design algorithms. To support that capability, the Synthesis Success Calculator also quantifies the sequence determinants' effects on synthesis success so that design algorithms can more easily find the shared sequence design space where a genetic system achieves both rapid synthesis and maximal function.

We began by enumerating 38 sequence properties that could influence the DNA synthesis and assembly process through a mechanistic interaction. There are 15 properties related to the presence of repetitive sequences with different positions, lengths and densities. Repetitive sequences confound the DNA assembly process by allowing undesired hybridization or annealing interactions, resulting in byproduct formation. There are 7 sequence properties related to the formation of single-stranded secondary structures, which inhibit hybridization between complementary strands during DNA assembly. There are 11 properties related to extreme double-stranded DNA melting temperatures or GC contents, particularly at the terminal ends of DNA fragments, which make it difficult to achieve uniform annealing of oligonucleotides and amplification of full-length DNA fragments. There are 4 properties related to the presence of single-stranded DNA tertiary structures, including G quadruplexes, C-rich i-motifs, and polynucleotide runs. The last property is the DNA fragment's length.

We then constructed a database of DNA sequences to train, validate, and test the random forest classifier. We first collected 303 DNA fragment sequences with known synthesis success outcomes, taken from our internal order history as well as from contemporary publications that included their synthesized DNA constructs as supplementary information [12, 13]. These DNA fragments were ordered from several different commercial service providers. 66 of these DNA fragments were labeled as synthesis failures because they were either cancelled or significantly delayed beyond the provider's turnaround time. Notably, even though these DNA fragments passed the commercial service providers' pre-existing filters for synthesis complexity, there was still a 22% frequency of synthesis failure. The synthesis failure rates are further compounded when multiple DNA fragments were needed to build a genetic system; when ordering 4 DNA fragments, there is only a 37% chance of obtaining them all with a short turnaround time. However, because these DNA fragments did pass the commercial service providers' filters, they are missing important sequence determinants that are known to create synthesis failure. Therefore, to develop a generalized machine learning model, it becomes necessary to augment the database with a diversified and balanced set of DNA fragment sequences that serve as either positive or negative controls.

As negative controls, we added 373 sequences that were systematically designed to violate the commercial service providers' filters for high sequence complexity, including the presence of long repetitive DNA sequences, fast folding hairpin structures, palindromic subsequences, polynucleotide runs, sequences with extreme GC content, and sequences with extreme DNA melting temperatures. Sequences in this dataset are rejected by contemporary commercial service providers and therefore are labeled as synthesis failures. As positive controls, we selected 400 subsequences from the successfully synthesized DNA fragments (354 to 1863 base pairs), that each contained a smaller number of individual DNA properties. The purpose of these positive controls is to isolate each DNA property that is associated with synthesis success in order to quantify its importance. Without the negative controls, the random forest classifier could not learn about the DNA properties with known negative effects. Without the positive controls, the random forest classifier would not be able to learn the importance of individual DNA properties. The presence of these controls also focuses the random forest classifier on

learning why 22% of the synthesized DNA fragments were not successful. Altogether, the database contains 1076 sequences with a well-balanced combination of evaluated DNA properties and synthesis outcomes.

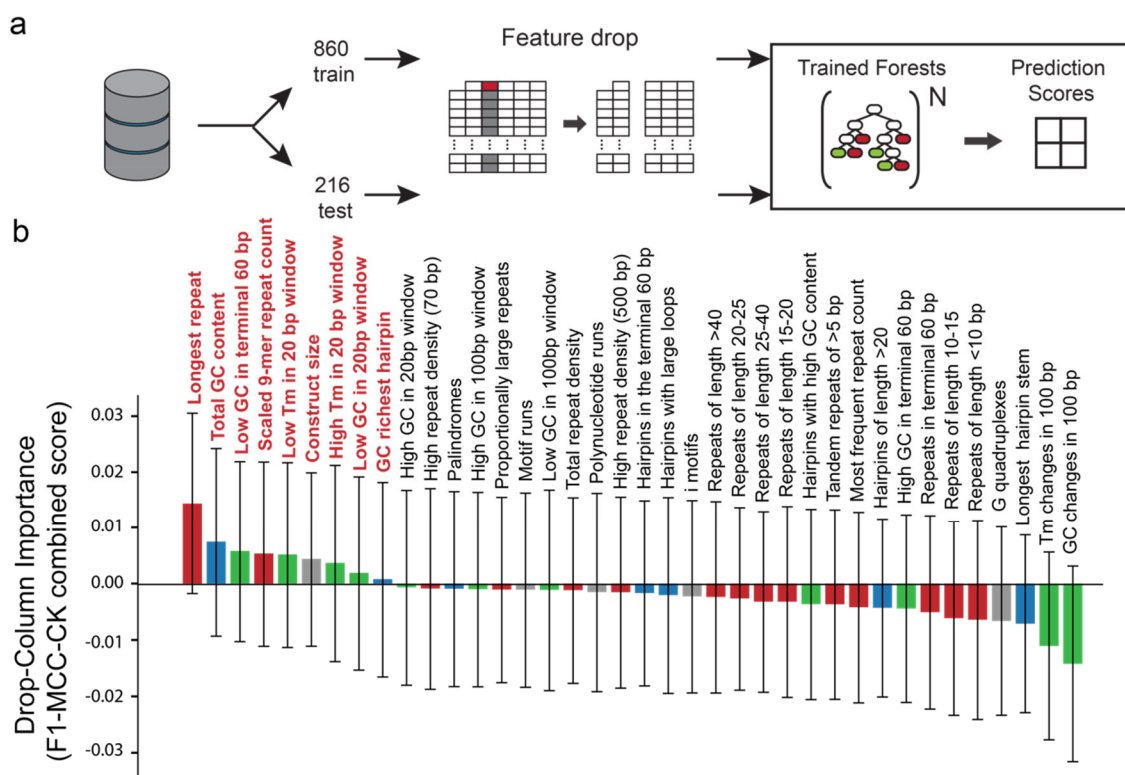


**Figure 16. Feature reduction using the 1076 sequence dataset.** (a) The workflow used to quantify the importance of each feature in the random forest classifier. (b) Drop column importance of each feature to the classification of 216 test set sequences, colored according to their general categories as repeats (red), secondary structures (blue), GC content (green), or miscellaneous (grey). Error bars represent standard deviations across 100 randomly initialized random forests (n = 100).

We selected a random forest classifier as our machine learning model for several reasons. Random forest models use an ensemble of decision trees to predict how combinations of properties (“features”) and their values determine an outcome. Compared to other machine learning models, random forests require fewer datapoints for optimal training; here, fewer means hundreds to thousands and not millions of data points. When their hyperparameters are optimized, for example, the number of features used per tree, the number of trees per forest, and the thresholds for creating new branches, random forest models do not over-fit on a dataset (“no memorization”), but their predictions can readily generalize across the dataset, including the unseen test set. Using feature drop analysis, random forest models can also be trained to reject properties that have limited effect on the outcome. Finally, when a random forest predicts an outcome, we can readily visualize and understand how it made that decision, including the locations of the DNA properties in the fragment sequence that are associated with synthesis failure.

We started by developing an automated test-train-validate pipeline commonly used for training machine learning models, utilizing the scikit-learn Python package [9]. We ran three rounds of hyperparameter optimization to develop a baseline random forest classifier, utilizing either randomized or grid search and employing 10-fold cross-validation to measure model performance. Model accuracy is quantified according to its  $F_1$  score, Matthews Correlation coefficient (MCC) and Cohen's Kappa score (CK), which together provide a metric of the model's precision, recall, and information content. Using these baseline hyperparameters, we then carried out a feature reduction strategy, called drop-column, that quantifies the contribution of each DNA property to model accuracy. To do this, the train-validate-test pipeline is carried out anew in the presence versus absence of each individual DNA property, using the same 80-20 train-test split and 100 random initial conditions, followed by comparing the model accuracies (**Figure 16A**). A DNA property's drop-column importance is the average difference in model accuracy when excluding a DNA property across 100 iterations of the train-test process. Overall, we found that only 9 DNA properties were important enough to lead to a reduction in model performance (**Figure 16B**). Removing any of other 29 DNA properties during the train-test process resulted in a lower  $F_1$ -MCC-CK combined score, for example, by reducing the information content in the model. Here, the most important DNA property was the length of the longest repetitive sequence within the DNA fragment, though overall GC content, extreme double-stranded DNA melting temperatures, GC-rich single-stranded DNA hairpins, and DNA fragment size were also important. These DNA properties are verified as sequence determinants of synthesis success.

We then began a new train-validate-test pipeline to train the random forest model, but now considering only our reduced set of 9 sequence determinants across the 1076 DNA fragment sequences in the database. We carried out training on 40,000 randomly initialized random forest configurations utilizing a training set of 575 DNA fragment sequences, followed by a performance assessment on a validation set of 250 DNA fragment sequences (**Figure 17A**). The resulting random forest classifier had a  $F_1$  score of 0.928, an MCC of 0.851, and a CK of 0.851 when evaluated on this validation set (**Figure 17B**). We then performed a more rigorous assessment of the classifier's performance by evaluating its predictions on an unseen test set of 251 DNA fragment sequences, yielding a  $F_1$  score of 0.908, an MCC of 0.810, and a CK of 0.810. However, we noted in this test set that there 13 positive predictions (synthesis success) were labeled as negative outcomes (synthesis failure), yielding a false discovery rate of 8.6%. The false discovery rate is the number of incorrectly predicted synthesis successes (false positives) over the total number of predicted synthesis successes. As our objective is to minimize the volatility of the DNA synthesis and assembly process, we then pursued a rebalancing strategy, called oversampling, to develop a random forest classifier with a lower false discovery rate.

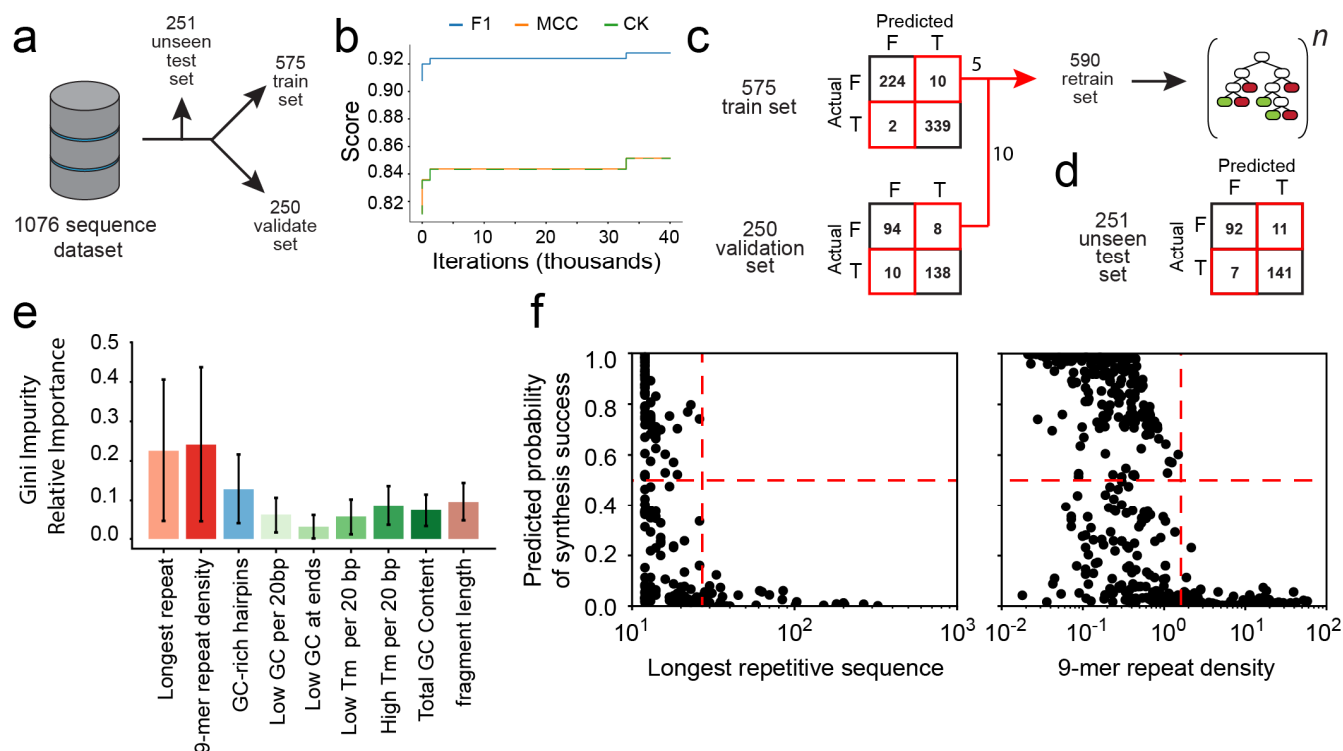


**Figure 17: Training and assessment of the Synthesis Success Calculator.** (a) A train-validate-test pipeline was used to parameterize a random forest classifier. (b) After hyperparameter optimization, training was performed on 40,000 random forest configurations. F1 is the F<sub>1</sub> score, MCC is the Matthew's correlation coefficient, and CK is Cohen's Kappa. (c) The predicted and actual synthesis outcomes are shown when evaluated on the train and validation sequence sets, using the best random forest configuration from (b). Oversampling is carried out to create a more balanced retrain set. (d) The predicted and actual synthesis outcomes are shown when evaluated on the unseen test set, using the retrained random forest classifier. Synthesis outcomes are either success (T) or delayed/cancelled (F). (e) The relative importance values of each sequence determinant in the retrained random forest classifier are shown. Error bars show the standard deviation of importance values across all trees in the forest ( $n = 1512$ ). (f) The predicted probability of synthesis success is shown across sequence determinants quantifying the presence of repetitive DNA sequences. Dashed red lines show the decision threshold for synthesis success (horizontal) and the sequence determinant values associated with synthesis failure (vertical).

To apply oversampling, we first identified 5 DNA fragment sequences in the training set that contained sequence determinants that were infrequently present, compared to other values (**Figure 17C**). We also identified 10 DNA fragment sequences in the validation set that contained extreme sequence determinant values that were not well-represented in the training set. We then added these 15 fragment sequences to the training set to create a rebalanced training set with 585 fragment sequences. We then ran additional iterations of training using this rebalanced training set (**Figure 17C**). The resulting random forest classifier had an excellent F<sub>1</sub> score of 0.928, a MCC of 0.851, and a CK of 0.851 when evaluated on the unseen test set, matching the performance on the validation set and yielding an improved false discovery rate of 7.2% (**Figure 17D**). This finalized random forest classifier was designated version 1.0 of the Synthesis Success Calculator.

Random forest classifiers are not black boxes; after training them, we can analyze the decision trees and sequence determinant cutoffs that lead to a successful classifier. Here, we found that the Synthesis Success Calculator considered the presence of repetitive sequences as the property most associated with synthesis failure (**Figure 17E**). Two sequence determinants – the length of the longest repetitive sequence and the number of repetitive 9-mers per 100 base pairs – had the highest Gini importances, which quantify how frequently they were used to make a decision within the random forests' decision trees.

We also examined how the Synthesis Success Calculator's sequence determinant cutoffs controlled its predictions. For example, the Synthesis Success Calculator applied a fairly sharp cut-off to the longest repetitive sequence in a DNA fragment; DNA fragments with a 26 base pair (bp) repetitive sequence were predicted to be readily synthesized, but any repetitive sequence longer than 26 bp would tilt the classifier's prediction towards synthesis failure (**Figure 17F**). Similarly, if the density of repetitive sequences was too high, the Synthesis Success Calculator would predict synthesis failure; for example, if any 100 bp of the DNA fragment contained 16 or more overlapping repetitive sequences, each 9 bp long (9-mer repeats), then the classifier predicted synthesis failure. Overall, the Synthesis Success Calculator would also predict synthesis success if a DNA fragment's GC content was between 29% and 63%.

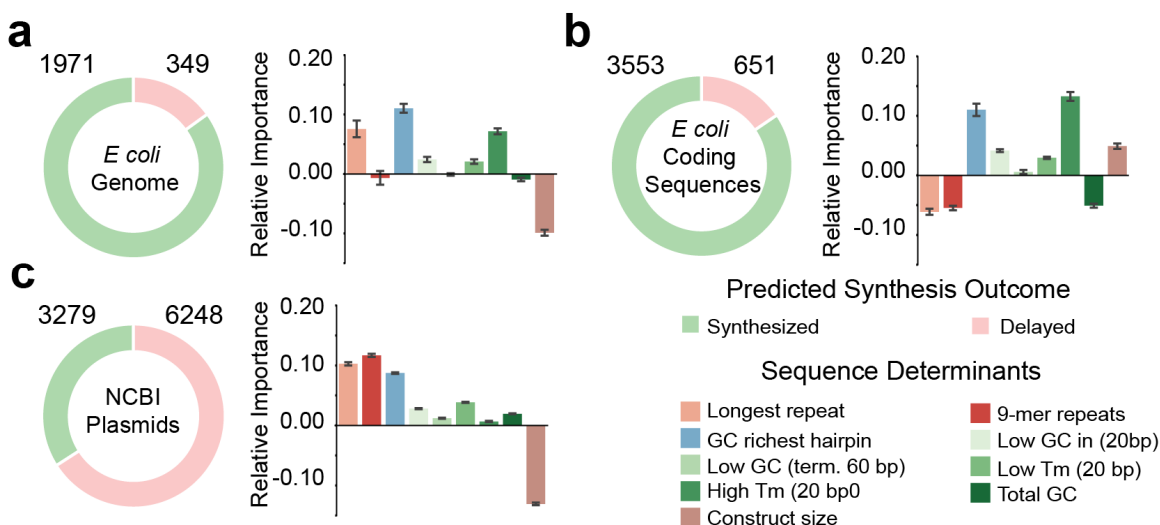


**Figure 18: Synthesis Success Calculator Examples.** (a) Synthesis success or failure were predicted across 2320 DNA fragments (each 2500 bp) extracted from the *Escherichia coli* MG1655 genome. Considering only the predicted synthesis failures, the relative importance values for each sequence determinant are shown. Error bars represent the standard deviation of sequence determinant values across predicted synthesis failures. (b) The same calculations were performed for 4204 unique protein coding sequences found in the *Escherichia coli* MG1655 genome (42 to 7077 bp long). (c) Synthesis success or failure were predicted across 9527 DNA fragments extracted from 4474 plasmids found in the NCBI database. The color code is shown.

Next, we applied the Synthesis Success Calculator across three illustrative examples and identified how sequence determinants affected the synthesis outcomes,

assisted by the Python package *treeinterpreter* [10, 104]. First, we uniformly broke up the *Escherichia coli* MG1655 genome (NCBI Accession #NC000913.3) into 2320 DNA fragments, each 2500 bp long, and evaluated their synthesis success. According to the classifier, 84.9% of the DNA fragments were predicted to be readily synthesized with a short turnaround time (**Figure 18A**). The remaining DNA fragments (15.1%) were predicted to be difficult to synthesize, due to the presence of single-stranded DNA secondary structures with high GC content, regions with very high double-stranded DNA melting temperatures, and regions with a long repetitive sequence. In the second example, we extracted 4204 unique protein coding sequences from same genome (42 to 7077 bp long) and predicted how readily we could synthesize a DNA fragment containing each coding sequence. The classifier predicted that 84.5% of these DNA fragments could be readily synthesized with a short turnaround time. The classifier identified several failure-associated sequence determinants in the remaining DNA fragments (15.5%), including regions of high double-stranded DNA melting temperature, regions containing GC-rich DNA hairpins, and less importantly, the lengths of the DNA fragment (**Figure 18B**). Interestingly, long repetitive DNA sequences or high densities of repetitive sequences were not predicted to be significant causes of synthesis failure. Overall, we were generally surprised that the classifier deemed *E. coli* genomic DNA to be so readily synthesized; however, these predictions are consistent with recent synthetic *E. coli* genome engineering efforts, where the supermajority of genome-like DNA fragments were synthesized within the anticipated turnaround time.

For the third example, we randomly selected 4474 plasmid sequences from the NCBI database, broke them into 9527 DNA fragments (202 to 3500 bp), and applied the Synthesis Success Calculator to predict their synthesis success. In contrast to the previous examples, the classifier predicted that only 34.4% of plasmid sequences could be readily synthesized with a short turnaround time (**Figure 18C**). The classifier predicted that the remaining DNA fragments (65.5%) contained many long repetitive sequences and many regions of high repeat density (9mer repeats) as well as many GC-rich DNA structures. Notably, even though these plasmid-derived DNA fragments have varying lengths, an analysis of the classifier's predictions showed that fragment length did not negatively impact their synthesis success (**Figure 18C**).

Next, we demonstrated how to embed the Synthesis Success Calculator's predictions into a design algorithm that optimizes a genetic system's sequence for improved synthesis success. As an illustrative example, we considered the 651 *E. coli* protein coding sequences that were predicted to be difficult to synthesize (**Figure 18B**) and applied the Synthesis Success Calculator to highlight the most important nucleotide sequence regions associated with synthesis failure. Here, we created a ranked list of these sequence regions, according to the calculated importances of the sequence determinants. We then varied the coding sequences' synonymous codon usages, but only within these highlighted regions, and at a frequency determined by their importance. Synonymous codon changes were then selected if they improved the Synthesis Success Calculator's predicted probability of synthesis success. By leveraging the Synthesis Success Calculator's importance calculations, we found that we only needed to recode 5% of the natural protein coding sequences to push 478 of them (73.4%) into the readily

synthesized category. Interestingly, further recoding of the natural coding sequences only provided marginal improvements in predicted synthesis success; recoding 10% or 15% of the natural protein coding sequences resulted in synthesis success rates of 78.3% or 81.4%, respectively. Overall, the Synthesis Success Calculator was able to pinpoint the most important sequence determinants and identify the sequence regions containing them, enabling a more precise redesign of DNA fragment sequences to ensure their successful synthesis and assembly.

Finally, after beginning routine usage of the Synthesis Success Calculator for other Synthetic Biology projects, we observed that our designed DNA fragments had a greater chance of synthesis success. In particular, when multiple genetic parts were encoded within a large DNA fragment, we found that using the Synthesis Success Calculator enabled us to swap out genetic parts that inadvertently created regions with repetitive DNA or extreme melting temperatures. In one example, we designed and ordered 50 synthetic DNA fragments (2859 to 4682 bp long) that each contained multiple promoters, CRISPR RNAs, and reporter proteins, used to build multi-regulator genetic circuits; 42 fragments were successfully synthesized on the first pass (84% success rate), enabling us to immediately proceed with DNA assembly and genome integration. In contrast, collaborators working in parallel on a similar genetic circuit project ordered 18 DNA fragments through the same service provider, but did not use our algorithm for design. Only 3 of the DNA fragments were synthesized and delivered after two attempts (16.6% success rate).

DNA synthesis and assembly are cornerstone technologies in Synthetic Biology, enabling the construction of genetic systems with desired sequences. Regardless of the experimental techniques used, there will always be a set of sequences that remain difficult to synthesize and assemble. Here, we trained and validated the Synthesis Success Calculator, a random forest classifier, that predicts when a DNA fragment can be readily synthesized with a short turnaround time while utilizing contemporary synthesis technologies that have been commercialized, optimized, and scaled-up for higher economies of scale. During that training process, we found that there are only 9 DNA properties associated with synthesis failure with statistical significance, starting from a much longer list of properties and metrics, many of them commonly used by commercial service providers. However, using only those sequence determinants, the Synthesis Success Calculator was able to correctly predict synthesis success with an  $F_1$  score of 0.928 and a false discovery rate of 7.2% on an unseen test set. Applying the Synthesis Success Calculator, building large genetic systems can now proceed on a more reliable schedule; there is an 74% chance of ordering 4 DNA fragments and receiving them all with a short turnaround time, compared to 37% using pre-existing filters.

To broadly facilitate its usage, we also developed an interactive web interface to the Synthesis Success Calculator, which is available at <https://salislab.net/software>. After inputting a DNA fragment sequence, the interface reports the predicted outcome (synthesis success or failure), the random forest classifier's predicted probability of success, and the importance values of the sequence determinants associated with synthesis failure. The interface also highlights the nucleotide regions associated with

synthesis failure, illustrating their locations and values using several types of graphs and visual aids.

As new synthesis and assembly techniques are developed, we anticipate that the sequence determinants that control synthesis success will also change, including their cut-off values. We therefore automated the training and testing of the Synthesis Success Calculator, making it easy to input sequence databases, train new random forest classifiers, and extract the important sequence determinants. New DNA properties may also be added to the list of potential sequence determinants. Here, our compiled database contains DNA fragment sequences ordered by several commercial service providers as they utilize similar techniques that all rely on the same DNA properties; nevertheless, it is possible to create separate databases for each distinct synthesis and assembly technology and correspondingly develop separate classifiers. To facilitate this process, the Python source code for the Synthesis Success Calculator is available at <https://github.com/hsalis/SalisLabCode>.

As demonstrated here, a key purpose of the Synthesis Success Calculator is to enable the automated design of genetic systems for high performance as well as synthesis success. However, when genetic systems are designed using several rules, it can become difficult to find the shared sequence space where all design rules are satisfied, including synthesis success. For example, if a model highlighted too many sequence regions as associated with synthesis failure, then it becomes difficult to modify all those sequences, while retaining the genetic system's desired functionalities. To overcome this challenge, we restricted the Synthesis Success Calculator to only considering sequence determinants with confirmed associations to synthesis failure, reducing the number of sequence changes needed to ensure synthesis success. To exemplify this distinction, we applied the Synthesis Success Calculator's predictions on a set of recalcitrant protein coding sequences and found that, for 478 out of 651 proteins, modifying only 5% of their sequences was sufficient to predict synthesis success.

With these confirmed sequence determinants in hand, it is also possible to create well-characterized toolboxes of genetic parts that are specially designed to promote synthesis success. For example, we previously designed and characterized 64 non-repetitive promoters and 28 non-repetitive sgRNA handles, enabling the co-expression of over 20 sgRNAs within Extra Long sgRNA Arrays without introducing repetitive DNA sequences [105]. We also designed and characterized 3500 non-repetitive bacterial promoters, varying transcription rates by over 100,000-fold, that can all be used simultaneously without introducing more than 10 bp repeat sequence [106]. Separately, we designed and characterized another 1722 non-repetitive yeast promoters, varying transcription rates by 25,000-fold, that can all be used simultaneously without introducing more than a 15 bp repeat sequence [106].

Future design algorithms can combine the Synthesis Success Calculator's predictions with such synthesis-ready toolboxes of genetic parts to design large genetic systems that have few, if any, sequence determinants associated with synthesis failure.

In this way, we can engineer large genetic systems with sophisticated functionalities, while ensuring our ability to synthesize, assemble, and build them.

#### 4.4 Automated Design of Thousands of Highly Non-Repetitive Genetic Parts for Engineering Evolutionary Robust Genetic Systems

In engineered systems, from ancient bridges to modern CPUs, repetition is often a symbol of elegant design. Copying and re-using the same architectural unit simplifies the relevant physics and can make it easier to rationally engineer desired functionalities. This strategy is commonly used when engineering genetic systems inside living cells, where designers select genetic parts from toolboxes and assemble them together to carry out a desired system-level cellular function, for example, to create cell-based therapies [107, 108], microbial chemical factories [109, 110], living biomaterials [111], or autonomous biosensors [112]. When designers select genetic parts with similar sequences or simply re-use the same genetic part across multiple locations, they introduce repetitive genetic parts into their system. However, in contrast to other disciplines, utilizing repetition to engineer cells can have catastrophic outcomes.

Repetitive genetic parts confound several steps of the design-build-test-learn engineering cycle by creating undesired products during DNA fragment synthesis and assembly [113-115], by triggering homologous recombination and creating genetic instability [116-119], and by preventing the correct alignment of next-generation sequencing reads [120, 121], thereby reducing our ability to troubleshoot genetic system function. For example, in a 90-day pressure test to engineer cells to produce 10 high-value chemicals, repetitive genetic parts greatly increased the time needed to receive synthesized DNA fragments and triggered the deletion of four enzymes within a barbamide biosynthesis pathway [122]. Repetitive genetic parts were also reported to trigger genetic instability in a variety of genetic engineering projects, including combinatorial CRISPR knock-outs for elucidation of regulatory network interactions [123, 124], metabolic pathway engineering in *Synechocystis* to produce fatty alcohols [125], and large-scale genetic circuit engineering in *E. coli* to carry out sophisticated digital logic [126] or to detect blue, green, and red light [127].

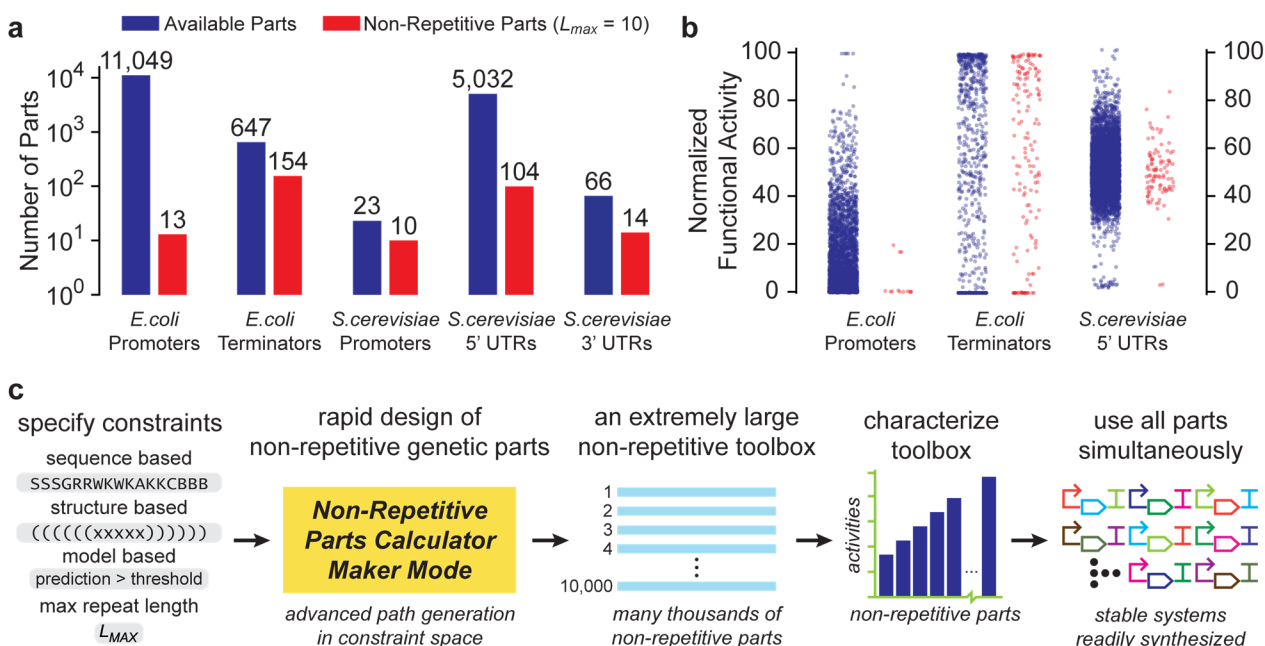


**Figure 19. Non-Repetitive Genetic Parts and their Activities.** (a) The number of non-repetitive genetic parts from existing, well-characterized toolboxes are shown. (b) Normalized functional activities for non-repetitive genetic parts are compared to existing toolboxes. (c) A workflow showing the design, construction, and characterization of non-repetitive genetic parts using the Non-Repetitive Parts Calculator.

Overall, the longest repetitive sequence within a genetic system determines the likelihood of encountering a failure mode. A pair of genetic parts with a 21 base pair repetitive sequence is sufficient to trigger homologous recombination in *E. coli*, which can excise the DNA in between the two parts and break the system function [118]. In other organisms (e.g. *S. cerevisiae*, *B. subtilis*, and mammalian viral vectors), even shorter repetitive regions are sufficient; a 12-18 base pair repeat is long enough for strand invasion and homologous recombination [116, 128, 129]. The process of assembling DNA fragments from oligonucleotides is particularly sensitive to repetitive sequences [113]; a 20 base pair repeat lowers the first-pass single-product success rate to 40% whereas a fragment with at most a 11 base pair repeat can be readily synthesized. But perhaps most limiting, many service providers will simply refuse to synthesize DNA fragments if they contain too many repetitive sequences. Others will provide the service at a higher price with a much longer turnaround time. If designers could select from thousands of highly non-repetitive genetic parts with desired activities, it would enable rapid engineering of much larger and more sophisticated genetic systems, while also greatly improving their evolutionary robustness. New design approaches are needed to solve synthetic biology's *repeat challenge*.

We therefore developed and experimentally validated a suite of algorithms that automatically identifies and designs large toolboxes of highly non-repetitive genetic parts, collectively called the Non-Repetitive Parts Calculator. In Finder mode, the algorithm identifies very large sets of non-repetitive genetic parts from an existing toolbox, while in Maker mode, the algorithm designs very large toolboxes of non-repetitive genetic parts from specified design constraints, a degenerate DNA or RNA sequence, an RNA secondary structure, and/or arbitrary model-based criteria. Non-repetitiveness is a global

property of the entire genetic part toolbox, and it is quantified by a repeat distribution (the number of repeats of length  $L$ ) or more simply by the maximum shared repeat length,  $L_{\max}$ . Both algorithms generate toolboxes according to a user-specified  $L_{\max}$  to control the desired level of non-repetitiveness. For example, when a toolbox of genetic parts is designed using an  $L_{\max}$  of 10 base pairs, every genetic part in the toolbox can be simultaneously utilized in the same genetic system without introducing a repetitive sequence longer than 10 base pairs, which is necessary to ensure successful DNA synthesis and assembly.

Initially, we created Finder mode to identify toolboxes of non-repetitive genetic parts from the thousands of genetic parts that have been previously characterized. Across several studies, we compiled a list of well-characterized genetic parts, including 11049 *E. coli*  $\sigma^{70}$  promoters [4, 50, 130, 131], 647 intrinsic transcriptional terminators [132, 133], 23 *S. cerevisiae* Pol II promoters [134, 135], 5032 *S. cerevisiae* 5' untranslated regions [136, 137], and 66 *S. cerevisiae* 3' untranslated regions [138]. The Finder algorithm revealed that most genetic part toolboxes have several commonly shared repetitive sequences that disqualify them from simultaneous utilization in a genetic system (**Figure 19A**). In particular, out of 11049 *E. coli* promoters, there are only 13 with a  $L_{\max}$  of 10 base pairs, and their measured transcription rates are too low for most applications (**Figure 19B**). Beyond our compiled list, more than half of the genetic parts in the Registry of Standard Biological Parts were predicted to be genetically unstable, due to plentiful repeat sequences [119]. Overall, there are simply too few genetic parts to engineer large genetic systems without being forced to introduce undesirable repetitive sequences.

These surprising observations motivated the development of the Non-Repetitive Parts Calculator's Maker mode (**Figure 19C**), which uses a newly developed path generation algorithm to rapidly carry out toolbox design (**Figure 20**). We then experimentally validated the Non-Repetitive Parts Calculator by designing and characterizing multiple toolboxes of highly non-repetitive, CRISPRi-enabled  $\sigma^{70}$  promoters, including 3550 promoter sequences with an  $L_{\max}$  of only 10 base pairs (**Figure 21**). These toolboxes have unprecedented depth and breadth, collectively varying mRNA levels by more than 1 million fold, and providing 10,000-fold tunable control for thousands of simultaneously expressed RNAs and proteins (**Figure 4**). Moreover, we used our measurements to identify several sequence determinants that modulate transcription initiation rates (**Figure 4**). We also present several additional design examples and benchmarking results to demonstrate how the Non-Repetitive Parts Calculator can rapidly design thousands of genetic parts without introducing repetitive DNA (**Supplementary Information**), thereby enabling the next-generation of large-scale genetic systems engineering, including the construction of evolutionary robust synthetic genomes.

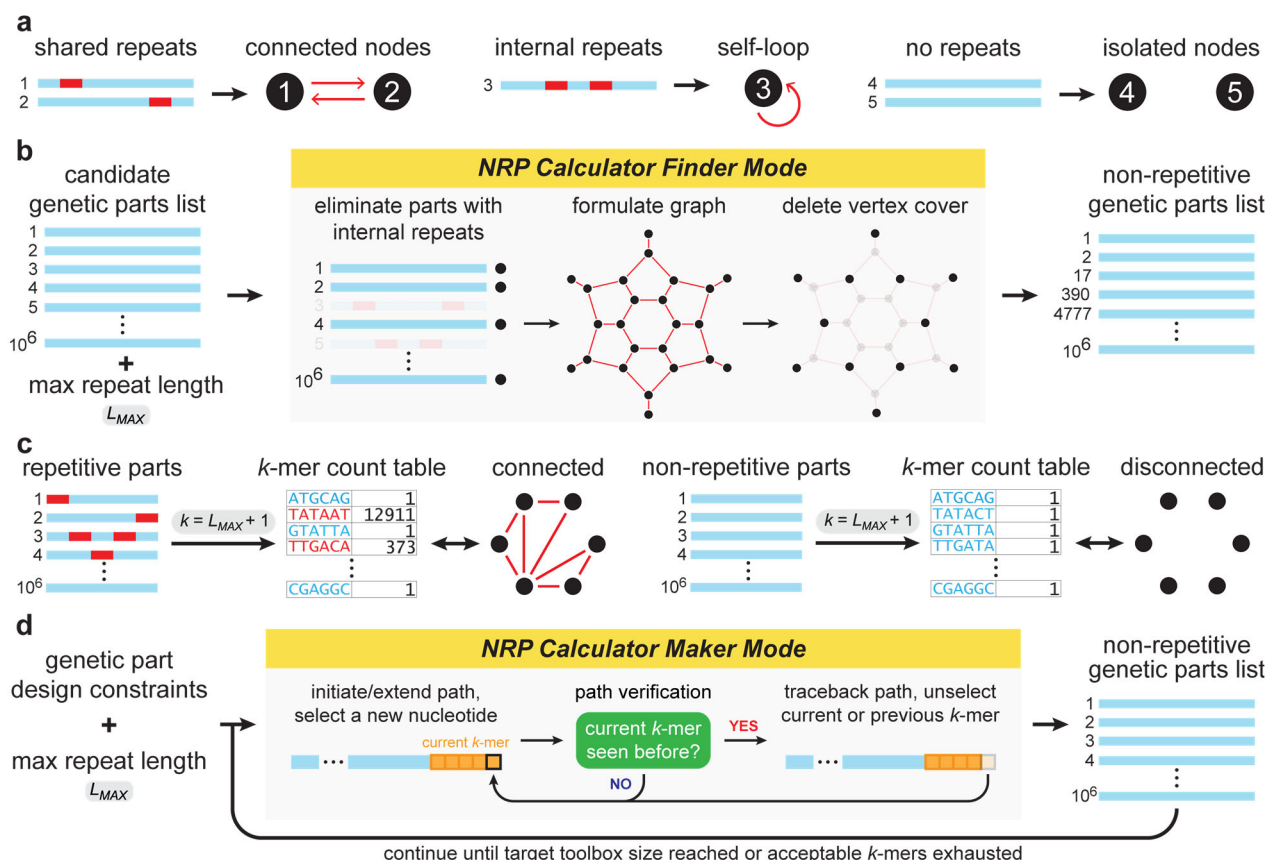


**Figure 20. The Non-Repetitive Parts Calculator.** (a) Genetic part toolboxes are represented by nodes and edges whereby two nodes share an edge whenever they contain a shared repetitive sequence (inverted or direct). Genetic parts with internal repeats are represented by a node with a self-looped edge. (b) An overview of the Non-Repetitive Parts Calculator in Finder Mode showing the identification and deletion of the minimum vertex cover in the repeat graph. (c) A hash table contains the number of occurrences of each  $k$ -mer sequence appearing within a set of genetic part sequences. Genetic parts are non-repetitive when their  $k$ -mer sequences have at most one occurrence in the hash table. (d) An overview of the Non-Repetitive Parts Calculator in Maker mode showing path generation, validation, and traceback for shared repeats.

## Identifying Non-Repetitive Genetic Parts from Existing Toolboxes.

Consider an arbitrary list of genetic part sequences. Our objective is to identify the largest fraction of genetic parts that collectively do not share any repetitive sequence longer than  $L_{max}$  base pairs. We formulated this problem using a graph theoretic formalism, whereby each genetic part is represented by a node, and whenever two genetic parts have a shared repeat sequence, their nodes are connected by an edge, creating a repeat graph (**Figure 20A**). Genetic parts with internal repetitive sequences are represented by a single node with a looped edge. An optimal solution is to find a minimum vertex cover and then delete it from the graph, creating a set of nodes that do not have any edges, which is called a disconnected graph (**Figure 20B**). The repeat graph's minimum vertex cover represents the smallest number of genetic parts that contribute at least one shared repeat. Once these genetic parts are removed, what remains is a smaller toolbox of non-repetitive genetic parts with a maximum shared repeat of  $L_{max}$  base pairs.

Solving for the minimum vertex cover is one of Karp's 21 classical NP-complete problems [139], though it has never been used in synthetic biology for design. An exhaustive search is infeasible when applied to large genetic part toolboxes; for the *E. coli* promoter toolbox, we would need to enumerate over  $10^{3000}$  vertex covers before finding the smallest one. While there are standard approximation algorithms, they scale poorly on larger genetic part toolboxes and do not yield large sets of non-repetitive genetic parts. For example, when applied to our example genetic part toolboxes (**Figure 19A**), the classical 2-approximation minimum vertex cover algorithm [140] identified only 9 non-repetitive *E. coli* promoters and 80 *S. cerevisiae* 5' UTRs ( $L_{\max} = 10$  bp). Doubling the size of the toolbox required 6.9-fold more compute time to solve for the minimum vertex cover. The key challenge here is that most genetic part toolboxes have many shared repeats for a small  $L_{\max}$ , creating highly dense graphs. We therefore developed a new 2-approximation algorithm, called Finder mode, that is specifically tailored towards highly dense graphs by maximizing elimination of highly connected nodes, while minimizing the elimination of pendant nodes. Across existing genetic part toolboxes and randomized sequence datasets, Finder mode is consistently faster than the standard linear time 2-approximation algorithm and always identifies a larger toolbox of non-repetitive genetic parts (**Figure 19A**).

However, it was clear that starting with a large number of existing genetic parts, and then identifying the non-repetitive ones – a top-down approach – was highly inefficient. Instead, we developed a bottom-up design algorithm (Maker mode) that combines constraints-based design of genetic parts with efficient path generation to rapidly design very large toolboxes of highly non-repetitive genetic parts. Rather than solving for the minimum vertex cover of a large graph, Maker mode very efficiently designs genetic parts so that the resulting toolbox never contains a shared repeat, creating a repeat graph that remains disconnected at every step.

## Constraint-based Design of Genetic Parts.

Genetic parts exhibit their activity through biophysical interactions that rely on DNA or RNA sequence motifs, the presence or absence of specific RNA structures, and/or higher-order sequence or structural characteristics. For example, a constitutive  $\sigma^{70}$  *E. coli* promoter sequence will have a high transcription initiation rate when it contains a conserved -35 and -10 hexamer, separated by a 17 base pair spacer. Likewise, a bacterial transcriptional terminator will have a high efficiency when it contains a fast-folding, stable RNA hairpin, followed by a U-rich tract. Such essential characteristics can be flexibly distilled into a set of criteria that every generated genetic part sequence must satisfy. Maker mode accepts three types of genetic part constraints: a degenerate DNA or RNA sequence using the IUPAC code; an essential RNA secondary structure using *dot-bracket-x* notation; and a model-based constraint that can be customized to quantify the presence of higher-order interactions or to facilitate the synthesis and construction of the genetic system by excluding sequences (e.g. restriction sites and polymeric sequences). All three constraints may be simultaneously used to specify a set of genetic part sequences with desired functionalities.

In one sense, these genetic part constraints are explicit hypotheses that distill one's knowledge of gene regulation and biophysics into the simplest possible computable form. In another sense, they are a type of classifier that separates the genetic part sequence space into only two categories: sequences expected to have some amount of genetic part activity versus sequences expected to have minimal to none activity. However, the genetic part constraints are not a quantitative prediction of functional activity; experimental characterization is still needed to validate them. In general, it is advantageous to incorporate as much degeneracy into the genetic part constraints as possible to design larger toolboxes.

## **Bottom-up Design of Non-Repetitive Genetic Parts.**

Consider an arbitrary set of specified genetic part constraints and a desired maximum shared repeat length  $L_{\max}$ . The overall objective is to design the largest possible toolbox of genetic part sequences that satisfy both the specified constraints as well as the desired level of non-repetitiveness. By analogy to the Non-Repetitive String problem, we show that this problem is NP-complete. However, by formulating the design of genetic part toolboxes as the generation of valid paths through sequence space, one nucleotide at a time, one part at a time, we developed a radically new, memory-efficient algorithm, called Maker mode. As illustrative examples of Maker mode's efficiency, we designed non-repetitive toolboxes for several types of genetic parts commonly used in Synthetic Biology, including 512 minimal yeast Pol II promoters [134], 749 insulated ribosome binding sites [54], 1026 toe-hold RNA switches [141], and 32896 intrinsic prokaryotic transcriptional terminators [142]. Maker mode only needed between 5 to 304 seconds to design each toolbox and its computational time scales linearly with toolbox size.

To do this, we developed fast and efficient approaches to generating valid sequences and verifying that they do not contain any repetitive sequence longer than  $L_{\max}$  base pairs. From sequence constraints, we initialize a sequence design tree that enumerates all valid nucleotide choices at each position. We design a candidate genetic part sequence by simply walking the design tree, randomly selecting an allowed nucleotide, and appending it. We then create a hash table that counts the occurrences of each unique  $k$ -mer sequence within the genetic parts' sequences, where  $k = L_{\max} + 1$  (**Figure 20C**). For every nucleotide appended to a candidate path, we verify that its  $k$ -mer sequences do not intersect with the hash table. After a candidate genetic part sequence is verified, we update the hash table so that its  $k$ -mer counts are included. However, if a candidate part contains an already used  $k$ -mer sequence, we explore alternative paths within the design tree, first altering the last nucleotide choice, then altering the last two nucleotide choices, and continuing until a candidate part sequence is verified to be non-repetitive (**Figure 20D**). The process of designing genetic part candidates terminates once all valid  $k$ -nucleotide choices are exhausted or until a user-specified fail count has been exceeded.

Interestingly, as Maker mode only relies on verifying the absence of  $k$ -mer sequences, we could greatly reduce memory requirements by utilizing a bloom filter to carry out part verification, while ensuring constant-time hash table operations. Designing

10,000 non-repetitive genetic parts (each 100 bp) with a  $L_{\max}$  of 10 base pairs only requires 3 megabytes of storage, which scales linearly with toolbox size. With this approach, Maker mode can also design genetic parts to be non-repetitive with respect to any background sequence, for example, a host organism's genome sequence, or any pre-existing toolbox of genetic parts, by simply initializing the bloom filter with their k-mers.

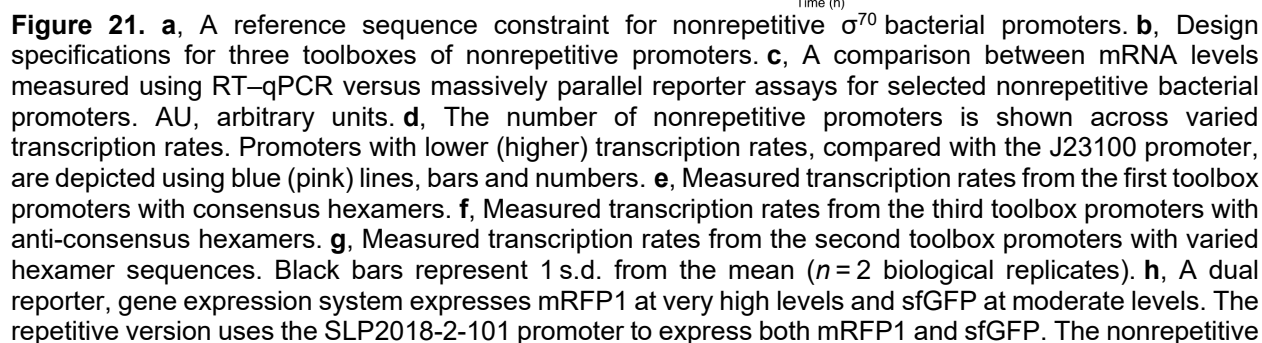
When RNA structural constraints are provided, we also utilize the Vienna RNA folding algorithm to initialize candidate genetic part sequence and verify that it folds into the desired structure [143]. If the predicted structure does not satisfy the constraint, we use the generated sequence as the initial condition to solving the inverse RNA folding problem, followed by selection of a similar non-repetitive sequence that satisfies both sequence and structural constraints. When model-based constraints are provided, they are either applied locally, on a per-nucleotide basis concurrent with part generation, or globally, after the part has been generated. Whenever a genetic part sequence fails to satisfy structural or model-based constraints, the part's design tree is traced back to the offending location and restarted, followed by exploration of other choices (**Figure 20D**).

### **Design and Characterization of 4350 Non-Repetitive Promoters.**

We applied the Non-Repetitive Parts Calculator in Maker mode to design 4350 highly non-repetitive, CRISPR-tunable  $\sigma^{70}$  promoters, followed by experimentally characterizing their transcription rates in *E. coli*. Our overall design goal was to provide tunable control over transcription rates for thousands of co-expressed genes. To do this, we first formulated a reference DNA sequence design constraint for  $\sigma^{70}$  promoters that includes a core promoter sequence for constitutive transcription and a downstream sgRNA/crRNA binding site for dCas9<sub>SP</sub>-mediated transcriptional regulation (**Figure 21A**). The core promoter sequence consists of a 20 base pair upstream region, a consensus -35 hexamer, a 17 base pair spacer, a consensus -10 hexamer, and a 6-base pair discriminator, while the sgRNA binding site contains a non-template PAM motif (CCN) and a 20-base pair sgRNA target site. The sgRNA binding site also serves as the initial transcribed region (ITR) for these promoters. This sequence constraint is highly degenerate with 64 “any” (N) base pairs, leading to an overall sequence space with over  $10^{38}$  variants. We then added model-based constraints to exclude undesired sequences (polymeric sequences that are difficult to synthesize, consensus hexamer-like sequences outside the -35/-10 to avoid creating multiple promoters, and relevant restriction sites) and to generate promoter sequences that match the host genome's GC content (here, 30-70%). Finally, we specified a maximum repeat size ( $L_{\max}$ ) of 10 base pairs to ensure that all promoters could be used simultaneously without introducing synthesis difficulties (**Figure 21B**).

Using these sequence and model constraints as a baseline, we designed three toolboxes with separate goals (**Figure 21B**). The first toolbox contains 800 promoter sequences with consensus -35 and -10 hexamers, expected to have high transcription rates. The second toolbox contains 3500 promoter sequences that have systematically mutated hexamer sequences (500 sequences each with between 0 to 6 consensus

**Figure 21.** **a**, A reference sequence constraint for nonrepetitive  $\sigma^{70}$  bacterial promoters. **b**, Design specifications for three toolboxes of nonrepetitive promoters. **c**, A comparison between mRNA levels measured using RT-qPCR versus massively parallel reporter assays for selected nonrepetitive bacterial promoters. AU, arbitrary units. **d**, The number of nonrepetitive promoters is shown across varied transcription rates. Promoters with lower (higher) transcription rates, compared with the J23100 promoter, are depicted using blue (pink) lines, bars and numbers. **e**, Measured transcription rates from the first toolbox promoters with consensus hexamers. **f**, Measured transcription rates from the third toolbox promoters with anti-consensus hexamers. **g**, Measured transcription rates from the second toolbox promoters with varied hexamer sequences. Black bars represent 1 s.d. from the mean ( $n = 2$  biological replicates). **h**, A dual reporter, gene expression system expresses mRFP1 at very high levels and sfGFP at moderate levels. The repetitive version uses the SLP2018-2-101 promoter to express both mRFP1 and sfGFP. The nonrepetitive



version instead uses SLP2018-1-573 to express sfGFP and has an overall maximum repeat length of only 15 bp. **i**, Bulk mRFP1 fluorescence levels per OD<sub>600</sub> were recorded over a 24-h exponential growth period for both the repetitive and the nonrepetitive genetic systems. **j**, Numbers and types of mutations found in isogenic plasmid isolates from cultures with repetitive (R) and nonrepetitive (NR) genetic systems.

We then used a massively parallel reporter assay to construct and characterize the 4350 highly non-repetitive promoters, combining barcoding, oligopool synthesis, library-based cloning, and next-generation sequencing (DNA-Seq, RNA-Seq) to measure their transcription rates in *E. coli* 5α cells. As an internal reference, we also included the standard J23100 promoter within the same library. Complete library coverage and precision measurements were ensured by using Maker mode to design highly non-repetitive barcode sequences, obtaining at least 4 million transformants per biological replicate, culturing cells for at least 6 hours in exponential growth conditions, and acquiring at least 64 million DNA-Seq reads and at least 95 million RNA-Seq reads per biological replicate. Each promoter was covered by at least 21 DNA-Seq and 40 RNA-Seq read counts with median counts of 5861 and 3804 with an overall mapping efficiency of 89% and 96%, respectively. Read count reproducibility was extremely high across biological duplicates (Pearson's  $R = 0.99$  and  $0.98$ , respectively). We then determined the promoters' transcription rates by taking the ratio between the RNA-Seq and DNA-Seq read counts and dividing by the J23100 promoter's transcription rate for normalization. The measured transcription rates were also highly reproducible across biological duplicates ( $R = 0.98$ , **Figure 21C**) with 3229 promoters passing the Rose criterion of >5 signal-to-noise ratio.

Collectively, the 4350 highly non-repetitive promoters have an exceptionally balanced breadth and depth of transcription rates that vary across a 1150000-fold range (**Figure 21D**). For example, 2922 promoters are stronger than the common J23100 promoter with up to a 5790-fold higher transcription rate. The weakest promoter has a 200-fold lower transcription rate than J23100. On average, there are 621 non-repetitive promoters within each 10-fold increment in transcription rate. As expected, Toolbox 1 has the highest number of strong promoters (97.5% greater than J23100) whereas Toolbox 3 has predominantly weak promoters (80% lower than J23100) (**Figure 21EF**). As designed, the promoters in Toolbox 2 have systematically varied transcription rates, depending on the number of mismatches in their hexamer sequences (**Figure 21G**). Qualitatively, 96% of the 0-mismatch promoters are stronger than J23100, compared to only 27% for the 6-mismatch promoters. Across all three toolboxes, 3550 promoters may be used simultaneously with an  $L_{\max}$  of 10 base pairs. When utilizing all 4350 promoters simultaneously, the  $L_{\max}$  is 19 base pairs.

We then constructed and characterized the 4,350 highly nonrepetitive promoters, combining barcoding, oligopool synthesis, library-based cloning and NGS (DNA sequencing (DNA-seq) and RNA sequencing (RNA-seq)) to measure their transcription rates in *E. coli* 5α cells. Overall, all promoter variants were covered by at least 21 DNA-seq and 40 RNA-seq counts with high reproducibility. We determined the promoters' transcription rates by taking the ratio between their RNA-seq and DNA-seq read counts and dividing by the J23100 promoter's transcription rate for normalization. These massively parallel measurements were then independently confirmed by selecting 10

nonrepetitive promoters, separately cloning them into the same expression vector, including their sgRNA-binding sites, and measuring their mRNA levels using quantitative PCR with reverse transcription (RT-qPCR) (Pearson's  $r = 0.94$ ; **Figure 21C**).

The 4,350 highly nonrepetitive bacterial promoters have an exceptionally high breadth and depth of transcription rates that vary across an 820,000-fold range (**Figure 21D**); 2,922 promoters are stronger than the common J23100 promoter, with up to a 4,731-fold higher transcription rate. The weakest promoter has a 173-fold lower transcription rate than J23100. On average, there are 621 nonrepetitive promoters within each 10-fold increment in transcription rate. As expected, the first toolbox has the highest number of strong promoters whereas the third toolbox has predominantly weak promoters (**Figure 21EF**). As designed, promoters in the second toolbox have systematically varied transcription rates, depending on the number of mismatches in their hexamer sequences (**Figure 21G**). Across all three toolboxes, 3,550 promoters may be used simultaneously with an  $L_{\max}$  of 10 bp. When using all 4,350 promoters simultaneously, the  $L_{\max}$  is 19 bp.

### Nonrepetitive genetic systems with improved stability

We then investigated how using nonrepetitive parts affects the stability of an engineered genetic system with extremely high selection pressure. We constructed and characterized repetitive and nonrepetitive versions of a two-protein expression system. Both versions used the SLP2018-2-101 promoter (1,476-fold higher transcription rate than J23100), the riboJ ribozyme and a designed RBS with a high translation rate (260,000 arbitrary units on the RBS Calculator v2.1 scale) to express very high levels of mRFP1 (red fluorescent protein) on a ColE1 origin plasmid (**Figure 21H**). In the repetitive version, we reused the same promoter to express superfolder green fluorescent protein (sfGFP) in a separate cistron, creating a 78-bp direct repeat. In the nonrepetitive version, we used the SLP2018-1-573 promoter to express sfGFP. All other genetic parts were selected to minimize the unintended presence of repetitive DNA, yielding an  $L_{\max}$  of 15 bp. Notably, stable cloning, stocking and purification of the repetitive plasmid were only possible using lower-temperature culturing.

We characterized the stability of these genetic systems in *E. coli* Turbo cells (New England Biolabs) grown at 37 °C over a 52-h period, measuring their mRFP1 and sfGFP fluorescence levels, followed by spot plating of cultures to purify isogenic plasmids for Sanger sequencing. The expression levels of mRFP1 from the repetitive genetic system rapidly decreased, reaching autofluorescence levels within 24 h of growth in liquid culture, whereas the nonrepetitive genetic system maintained very high mRFP1 expression levels over the entire growth period (**Figure 21I**). Sequencing revealed that 100% ( $n = 30$ ) of the repetitive genetic systems contained the same mRFP1 deletion precisely in between the reused promoter sequences, consistent with excision by homologous recombination (**Figure 21J**). In contrast, only one of the nonrepetitive genetic systems (2% of  $n = 42$ ) had a large deletion in mRFP1, mediated by a 13-bp repetitive sequence inside the noncoding regions surrounding it. Instead, we found that 62% of the nonrepetitive genetic systems were free of mutation. The remaining 36% contained a variety of spontaneous mutations, including early stop codons and frameshifts in mRFP1, the insertion of an IS1 transposase into the designed RBS and promoter modifications (**Figure 21J**), resulting in

minority subpopulations with reduced mRFP1 expression. Notably, both repetitive and nonrepetitive genetic systems maintained their sfGFP expression.

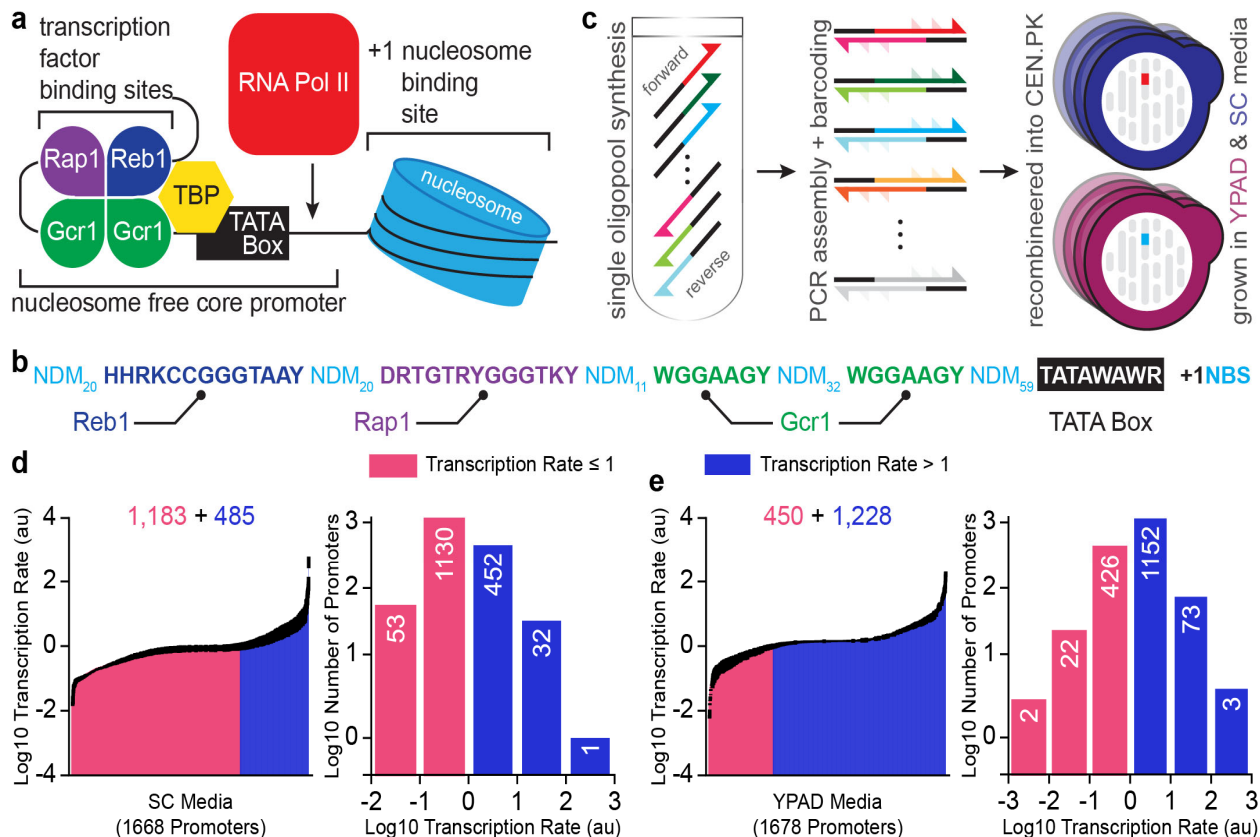


**Figure 22: Design and Characterization of 1678 Non-Repetitive Yeast Promoters.** (a) a, The desired architecture for the yeast promoters, including four TF-binding sites, a TATA box, a nucleosome-free core region and a downstream high-affinity NBS. (b) b, The degenerate sequence constraints used by the Nonrepetitive Parts Calculator to design the toolbox of nonrepetitive yeast promoters. TF-binding motifs are shown. NDMs contain polymeric A-rich or T-rich sequences that do not favor nucleosome binding. The +1 NBS is a 135-bp degenerate sequence that strongly binds nucleosomes. (c) c, Promoters were constructed using one-pot PCR assembly and barcoding, followed by integration into the yeast genome using one-pot recombination. (d,e) d,e, Measured transcription rates for 1,722 nonrepetitive yeast promoters using either SC (d) or YPAD (e) growth medium. Histograms show the number of promoters within selected transcription rate ranges.

### Design, one-pot assembly and testing of 1,722 nonrepetitive yeast promoters

We then applied the Nonrepetitive Parts Calculator in Maker mode to design 1,931 highly nonrepetitive eukaryotic promoters with an  $L_{\max}$  of 15 bp, followed by characterization of their transcription rates in *S. cerevisiae* (Figure 22A). Our sequence design constraints combined four degenerate transcription factor (TF) binding sites for Rap1, Reb1 and Gcr1, a degenerate TATA box site and several flanking nucleosome-depleting motifs (NDMs), followed by a degenerate nucleosome binding site (NBS) motif (Figure 22B). These promoter sequences were longer than 300 bp, requiring them to be encoded by two overlapping oligonucleotides, with additional design constraints to facilitate their one-pot PCR assembly and barcoding, followed by one-pot recombination to integrate them into the *S. cerevisiae* genome (Figure 22C). We first carried out DNA-seq measurements on genomic amplicons, associating 1,825 identified promoters with barcodes. In separate

experiments, we then cultured strains using either synthetic complete (SC) or YPAD medium, followed by extractions and NGS, yielding 1,722 transcription rate measurements with high reproducibility across biological replicates ( $r = 0.98$ ). The yeast promoters' transcription rates varied across a 26,980-fold or 25,820-fold range in SC and YPAD media (**Figure 22DE**), respectively, with considerable medium-to-medium variation ( $r = 0.15$ ). Overall, transcription rates were higher in YPAD medium.

Collectively, these bacterial and yeast promoters provide a remarkable level of expression control without introducing repetitive DNA (**Figure 23**). For depth of control, there is a 2,000-fold range where 20 bacterial genes can be co-expressed with similar transcription rates (**Figure 23A**). For breadth of control, over 2,900 bacterial genes can be co-expressed across a 10,000-fold range (**Figure 23B**). Over 489 yeast genes can be co-expressed with transcriptional uniformity (**Figure 22DE**). These promoters are also exceptionally far apart in sequence space; there are at least 30 mutations separating any two bacterial promoters (**Figure 23C**) and at least 100 mutations separating any two yeast promoters. Every pair of sgRNA binding sites within the bacterial promoters is functionally orthogonal with at least 13 mismatches between them.

Overall, the toolboxes also provide an unprecedented level of depth when selecting multiple non-repetitive promoters to co-express RNAs or proteins at desired transcription rates at any value across a large range (**Figure 22**). For example, there are at least 20 promoters with similar transcription rates (within 10%) across a 2000-fold range that can all be used simultaneously without introducing repetitive sequences (**Figure 22A**). Alternatively, one can utilize these non-repetitive promoters to co-express over 2900 RNAs or proteins, while achieving transcription rates across a 10000-fold range (**Figure 22B**). Interestingly, while the algorithm's metric for non-repetitiveness only considers contiguously shared repeat sequences, the toolboxes' promoters also had exceptionally high pair-wise edit distances as a free side-effect. There are 33 to 50 mismatches between 90% of promoter sequences in these toolboxes (each 78 base pairs long), which is an even more stringent metric for non-repetitiveness (**Figure 22C**). We also obtained the same level of non-repetitiveness across sgRNA binding site sequences using our algorithmic design; any two sgRNA binding sites have pair-wise hamming distances of  $15 \pm 2$  base pairs, making them functionally orthogonal.

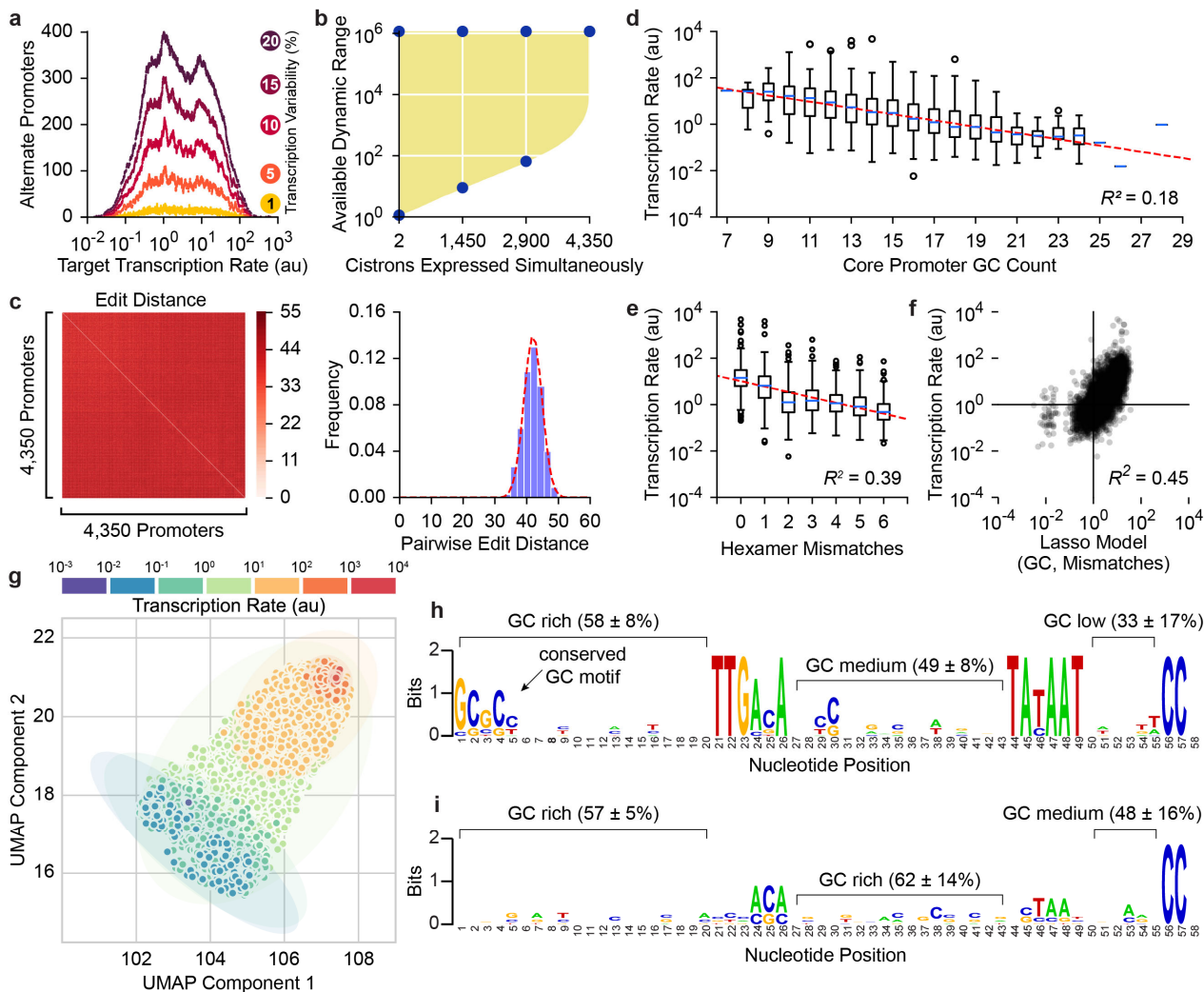


**Figure 23. Analysis of Non-repetitive Bacterial Promoters.** (a) Availability of alternative promoters at specific transcription rates within some tolerated variability. (b) Availability of transcription rate dynamic range for simultaneously expressing a selected number of genes or cistrons. (c) The Levenshtein edit distance matrix and distribution between every pair of non-repetitive promoters. (d) Transcription rate variability with increasing GC count within the core promoter sequence. (e) Transcription rate variability with increasing number of mutations in the hexamers across the first and second promoter toolbox. (f) Comparing transcription rates predicted by a trained Lasso model against experimentally measured transcription rates. (g) Spectral clustering of promoter sequences and their transcription rates using UMAP. (h-i) Weblogo analysis of top and bottom 10 promoters with the highest and lowest transcription rates.

## Sequence Determinants and Transcription Rate Models

Intriguingly, as a byproduct of our algorithmic design, the promoter toolboxes also have exceptional sequence diversity (0.0014 bits average entropy per varied position), providing a highly informative dataset for elucidating the sequence determinants controlling transcription rate and developing sequence-function models. First, as widely known [144], the -35 and -10 hexamers play an important role in controlling transcription initiation rates by mediating RNAP/ $\sigma^{70}$  holoenzyme recruitment. Indeed, promoters with a higher number of mismatches to the consensus hexamer sequences have overall lower transcription rates. By itself, the number of hexamer mismatches is a potent sequence

determinant; a linear regression model using this one variable can explain 39% of the observed changes in transcription rates in this dataset ( $r^2 = 0.39$ ,  $p$ -value  $< 10^{-324}$ , **Figure 22E**). However, across Toolbox 2 promoters with the same number of hexamer mismatches, there is still considerable variation in transcription rates; 29000-fold across 0-mismatch promoters, decreasing to 4100-fold across 6-mismatch promoters (**Figure 21G**), indicating the presence of additional sequence determinants.

As previously suggested [145-147], we next investigated whether the GC composition of a promoter sequence can affect its transcription rate, potentially mediated through several mechanisms. For example, a promoter's GC composition is an indirect proxy for the energy needed to pull apart double-stranded DNA during melting and R-loop formation. GC composition also affects the persistence length of double-stranded DNA and the accessibility of its major and minor grooves, thereby affecting its ability to form DNA kinks and essential contacts during RNAP holoenzyme recruitment and open complex formation[148]. We found that a higher number of GC base pairs in the core promoter sequence, encompassing the hexamers and spacer region, negatively affected the promoter's transcription rate (**Figure 22D**). Considering only this sequence determinant, a linear regression model could explain 18% of the observed changes in transcription rate within this dataset ( $R^2 = 0.18$ ,  $p$ -value  $= 10^{-191}$ , **Figure 22D**). Notably, the GC composition of other individual elements had no statistically significant effect on the promoters' transcription rates. However, because the consensus hexamer sequences are AT-rich, it's possible that GC composition is simply a proxy for the number of hexamer mismatches.

We then carried out a finer-grained analysis by leveraging machine learning to quantify the importance of each base pair within the hexamers as compared to the core promoter's GC count. The resulting 13-variable Lasso model treats mismatches in each hexamer base pair categorically and identifies a linear model that can explain 45% of the observed changes in transcription rate, which is somewhat more than the simple hexamer model. This Lasso model also has an exceptionally high effect size and can rank the promoters with 71% accuracy across a 11500-fold range ( $R^2 = 0.45$ ,  $p$ -value  $< 10^{-324}$ , Cohen's  $f^2 = 0.82$ , **Figure 22F**). A key benefit of Lasso models is the interpretability of their coefficients, due to regularization, enabling quantitative comparisons of feature importance. Based on these coefficients, the -35 hexamer's first and fourth positions are observed to be the most important together with the -10 hexamer's first three positions. To compare, the importance of GC count is much less overall, though it remains statistically significant and inversely related to transcription rate. However, there are additional sequence determinants that are not perturbed here, for example, all non-repetitive promoters in these toolboxes have a 17 base pair spacer region. In addition, only 10% of the non-repetitive promoters contain mutations in the sixth position of the -10 hexamer, causing the Lasso model to deem this position as unimportant.

Collectively, our promoter sequence design constraints successfully generated sequences that support transcriptional initiation (**Figure 21**), though with highly varied transcription rates. We next determined how narrower design constraints could be

formulated to specify promoters with higher transcription rates. To do this, we first applied UMAP spectral clustering [149] to identify multiple clusters of promoters that share similar transcription rates and possess sequence similarities across two reduced components that are learned from the sequence-function dataset. The resulting map shows a continuum of contiguous clusters, each representing a promoter sequence space, that become smaller as transcription rates increase (**Figure 22G**). From this map, we observe that there are many promoter sequences that can support a low transcription rate, while there are much fewer promoter sequences that can support a high transcription rate, consistent with a model that treats a promoter's binding energies as additive. We then examined the sequence similarities of the top 10 promoters with the highest transcription rates (**Figure 22H**) as compared to the bottom 10 promoters with the lowest transcription rates (**Figure 22I**). As expected, the top 10 promoters have few mismatches in their -35/-10 hexamers (60% 0-mismatches, 30% 1-mismatches, and 10% 3-mismatches) whereas the bottom 10 promoters have many hexamer mismatches (70% with 4 or more mismatches). Likewise, the top 10 promoters had lower GC compositions in their core promoter and discriminator sequences, compared to the bottom 10 promoters, suggesting that the GC-mediated interactions must be optimized to achieve very high transcription rates. However, even though AT-rich upstream elements bind to the  $\alpha$ CTD of RNAP [148] and normally increase a promoter's transcription rate, the top 10 promoters share a common 4-base pair GC motif in their upstream elements, suggesting that an excessively strong binding interaction there could negatively impact a promoter's transcription rate, for example, by preventing promoter escape.

We then applied a recently developed neural network model, trained with over 100 million random yeast promoters (CGDB model [150]), to explain how up to 245 TFs bound to our nonrepetitive yeast promoter sequences and regulated their transcription rates. The CGDB model is distinct in that the architecture and parameters of its neural network layers were designed to model physical interactions. In one layer, TF-binding occupancies are predicted using trained position score matrices (**Figure 24C**). In another layer, the effects of the bound TFs on chromatin remodeling and RNAP recruitment are predicted (TF effects), using learned activity and potentiation coefficients (**Figure 24E**). The final output layer combines several layered calculations to predict a promoter's overall transcription rate, incorporating learned multi-TF and position-dependent regulatory effects. However, after running these calculations on our nonrepetitive yeast promoters, we found no correlation between the model-predicted transcription rates (final output layer) and the measured transcription rates

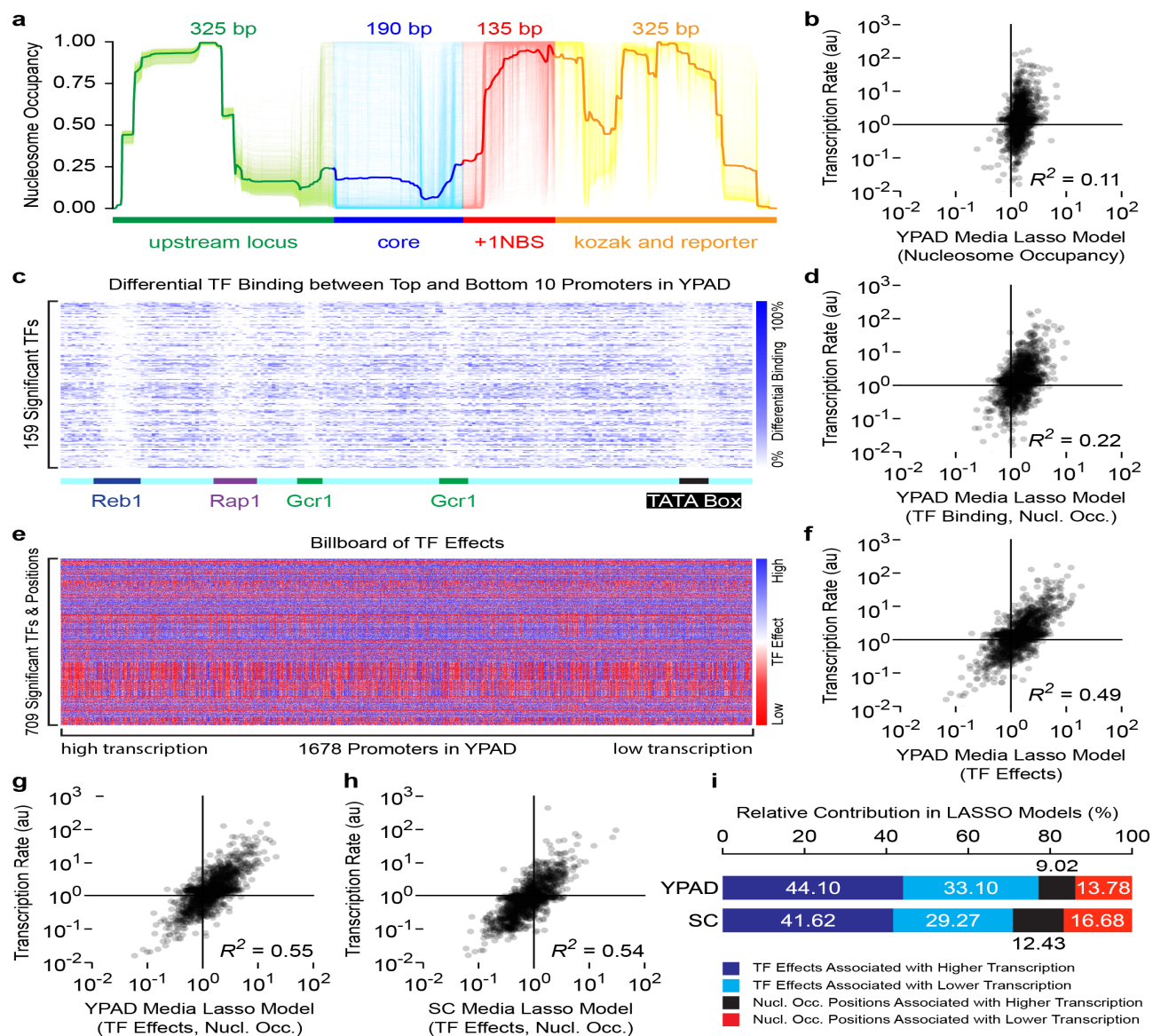


**Figure 24. Analysis of Non-repetitive Yeast Promoters.** **a**, Predicted nucleosome occupancies using the NuPoP model. Light lines are occupancies from individual nonrepetitive yeast promoter sequences. Dark lines are averaged nucleosome occupancies across all sequences. **b**, Transcription rate predictions from a Lasso model trained using nucleosome occupancies at 325 positions for all ( $n = 1,678$ ) measured promoter transcription rates in YPAD medium. **c**, Predicted differences in TF-binding occupancy between the top 10 versus the bottom 10 nonrepetitive yeast promoters, sorted by measured transcription rate. The purposely included TF-binding motifs are labeled. **d**, Transcription rate predictions from a Lasso model trained using 159 summed TF-binding occupancies and 325 nucleosome-binding occupancies (nucl. occ.), for all ( $n = 1,678$ ) measured promoter transcription in YPAD growth medium. **e**, A billboard of predicted TF effects evaluated on 1,678 nonrepetitive yeast promoters, sorted by their measured transcription rates. **f**, Transcription rate predictions from a Lasso model trained using only the most significant 709 TF effects and 1,678 measured promoter transcription rates in YPAD growth medium. **g,h**, Transcription rate predictions from Lasso models trained using 709 TF effects and 325 nucleosome occupancies, for all 1,678 measured promoter transcription rates in YPAD medium (**g**) or 1,668 measured promoter transcription rates in SC (**h**) growth medium. **i**, The overall contributions of TF and nucleosome binding to Lasso model predictions in **g** and **h**.

( $R^2 = 0.005$ ,  $P = 0.003$ ), possibly due to large differences in the promoter datasets, including length (80 versus 325 bp), architecture and the surrounding chromatin structure.

Instead, we found that predictions from the CGDB model's internal calculations explained the nonrepetitive promoters' measured transcription rates, a notable success in neural network transfer learning. First, for comparison, combining the NuPoP nucleosome occupancy and TF-binding occupancy predictions, yielded a Lasso model that could explain only a small portion of the promoters' transcription rates ( $R^2 = 0.22$ ,  $P < 10^{-94}$ ; **Figure 24CD**). However, 709 TF effect predictions from the CGDB model yielded a Lasso model with much better explanatory power ( $R^2 = 0.49$ ,  $P < 10^{-244}$ ; **Figure 24EF**). By combining nucleosome binding and TF effect predictions together, a Lasso model could explain 54% and 55% of the observed 25,000-fold change in transcription rates across our nonrepetitive yeast promoters in SC and YPAD growth media, respectively ( $P < 10^{-284}$ ; **Figure 24GH**). Nucleosome interactions had a greater effect on transcription rates when cells were grown in SC medium (**Figure 24I**), possibly due to lower TF expression levels. Most notably, even though our nonrepetitive yeast promoters had distinct high-affinity TF-binding sites, this modeling confirms that over 100 TFs will bind in a distributed fashion to lower-affinity sites, greatly altering the promoters' overall transcription rates.

Overall, the Nonrepetitive Parts Calculator and its first demonstrative examples facilitate the design and engineering of very large genetic systems with greatly improved stability. Our 4,350 highly nonrepetitive bacterial promoters and 1,722 highly nonrepetitive yeast promoters can immediately be used to engineer many enzyme pathways and gene clusters for biosynthesis of valuable chemicals, such as fuels, flavors, fragrances and pharmaceuticals. Our 4,350 nonrepetitive bacterial promoters also have built-in sgRNA-binding sites, enabling the engineering of many-regulator genetic circuits to process signals, perform pattern recognition and execute actionable decisions. More generally, the Nonrepetitive Parts Calculator can design many types of nonrepetitive parts, according to a flexible set of design specifications that can include model predictions as constraints. We envision a future whereby computational design automatically incorporates nonrepetitive genetic parts to specify extremely large genetic systems, including newly designed genomes, according to a researcher's objective functions. Although it may feel excessive today to have access to thousands of nonrepetitive genetic parts, their availability is essential to future synthetic biology applications.

## 5.0 Conclusions

It has been a grand challenge to transition synthetic biology into a mature engineering discipline, where genetic systems (e.g., biosensors, genetic circuits, and metabolic pathways) are reliably designed, built, and tested to reprogram cellular behavior with desired outcomes. Quantitative models play a central role in synthetic biology's design-build-test-learn cycle by predicting the function of a candidate genetic system, before it is constructed, and therefore reducing trial-and-error experimentation. Gene expression models have been developed to predict the sequence-function relationships for several gene regulatory parts, enabling the automated design of genetic systems with desired protein expression levels. Gene expression predictions have also been combined with system-level models of genetic circuits and metabolic pathways to predict how changes in system architecture, component expression levels, and host genome control organismal sensing, decision-making and the biosynthesis of desired chemicals. Overall, for Synthetic Biology to become a mature engineering discipline, we need accurate gene expression models for each gene expression step.

### 5.1 A Predictive Model of Transcriptional Initiation

Transcription rates are regulated by the interactions between RNA polymerase, sigma factor, and promoter DNA sequences in bacteria. However, it remains unclear how non-canonical sequence motifs collectively control transcription rates. Here, we combined massively parallel assays, biophysics, and machine learning to develop a 346-parameter model that predicts site-specific transcription initiation rates for any  $\sigma^{70}$  promoter sequence, validated across 17396 bacterial promoters with diverse sequences. We applied the model to predict genetic context effects, design  $\sigma^{70}$  promoters with desired transcription rates, and identify undesired promoters inside engineered genetic systems. The model provides a biophysical basis for understanding gene regulation in natural genetic systems and precise transcriptional control for engineering synthetic genetic systems.

### 5.2 Automated Model Testing to Quantify Error & Information Content

Gene expression models greatly accelerate the engineering of synthetic metabolic pathways and genetic circuits by predicting sequence-function relationships and reducing trial-and-error experimentation. However, developing models with more accurate predictions is a significant challenge, even though they are essential to engineering complex genetic systems. We present a model test system that combines advanced statistics, machine learning, and a database of 9862 characterized genetic systems to automatically quantify model accuracies, accept or reject mechanistic hypotheses, and identify areas for model improvement. We also introduce Model Capacity, a new information theoretic metric that enables correct model comparisons across datasets. We

demonstrate the model test system by comparing six models of translation initiation rate, evaluating 100 mechanistic hypotheses, and uncovering new sequence determinants that control protein expression levels. We then applied these results to develop a biophysical model of translation initiation rate with significant improvements in accuracy. Automated model test systems will dramatically accelerate the development of gene expression models, and thereby transition synthetic biology into a mature engineering discipline.

### 5.3 A Predictive Model of DNA Synthesis Success

The synthesis and assembly of long DNA fragments has greatly accelerated synthetic biology and biotechnology research. However, long turnaround times or synthesis failures create unpredictable bottlenecks in the design-build-test-learn cycle. We developed a machine learning model, called the Synthesis Success Calculator, to predict whether a long DNA fragment can be readily synthesized with a short turnaround time. The model also identifies the sequence determinants associated with the synthesis outcome. We trained a random forest classifier using biophysical features and a compiled dataset of 1076 DNA fragment sequences to achieve high predictive performance ( $F_1$  score of 0.928 on 251 unseen sequences). Feature importance analysis revealed that repetitive DNA sequences were the most important contributor to synthesis failures. We then applied the Synthesis Success Calculator across large sequence datasets and found that 84.9% of the *Escherichia coli* MG1655 genome, but only 34.4% of sampled plasmids in NCBI, could be readily synthesized. Overall, the Synthesis Success Calculator can be applied on its own to prevent synthesis failures or embedded within optimization algorithms to design large genetic systems that can be rapidly synthesized and assembled.

### 5.4 Automated Design of Large Toolboxes of Highly Non-Repetitive Parts

Engineered genetic systems can fail when designers are forced to re-use similar genetic parts, due to the introduction of repetitive DNA sequences. To solve this challenge, we developed a novel algorithm, the Non-Repetitive Parts Calculator, that rapidly designs thousands of highly non-repetitive genetic parts from specified constraints, utilizing a newly developed linear-time, memory-efficient approach. As demonstrations, we designed and characterized 4350 highly non-repetitive bacterial  $\sigma^{70}$  promoters with transcription rates that varied across a 1150000-fold range and 1678 highly non-repetitive yeast promoters with transcription rates that varied across a 25000-fold range, specified by synthetic design constraints with desired functionalities. We show how utilizing non-repetitive genetic parts eliminates homologous recombination as a source of genetic instability. We also applied machine learning to identify the binding interactions that control these promoters' transcription rates. Overall, the Non-Repetitive Parts Calculator provides a generalized algorithmic solution to engineering organisms without introducing repetitive DNA, thereby improving their evolutionary robustness.

## 6.0 References

1. Geggier, S. and A. Vologodskii, *Sequence dependence of DNA bending rigidity*. Proc Natl Acad Sci U S A, 2010. **107**(35): p. 15421-6.
2. Heddi, B., et al., *Intrinsic flexibility of B-DNA: the experimental TRX scale*. Nucleic Acids Res, 2010. **38**(3): p. 1034-47.
3. Espah Borujeni, A., et al., *Genetic circuit characterization by inferring RNA polymerase movement and ribosome usage*. Nat Commun, 2020. **11**(1): p. 5001.
4. Kosuri, S., et al., *Composability of regulatory sequences controlling transcription and translation in Escherichia coli*. Proceedings of the National Academy of Sciences, 2013. **110**: p. 14024-14029.
5. Bonde, M.T., et al., *Predictable tuning of protein expression in bacteria*. Nature Methods, 2016. **13**: p. 233-236.
6. Seo, S.W., et al., *Predictive design of mRNA translation initiation region to control prokaryotic translation efficiency*. Metabolic Engineering, 2013. **15**: p. 67-74.
7. Grüning, B., et al., *Bioconda: sustainable and comprehensive software distribution for the life sciences*. Nature Methods, 2018. **15**(7): p. 475-476.
8. Lorenz, R., et al., *ViennaRNA Package 2.0*. Algorithms Mol Biol, 2011. **6**: p. 26.
9. Pedregosa, F., et al., *Scikit-learn: Machine learning in Python*. Journal of machine learning research, 2011. **12**(Oct): p. 2825-2830.
10. Saabas, A. *Interpreting random forests*. 2014 October 19, 2014 [cited 2019 March 5th]; Available from: <http://blog.datadive.net/interpreting-random-forests/>.
11. Cock, P.J., et al., *Biopython: freely available Python tools for computational molecular biology and bioinformatics*. Bioinformatics, 2009. **25**(11): p. 1422-3.
12. Lau, Y.H., et al., *Large-scale recoding of a bacterial genome by iterative recombineering of synthetic DNA*. Nucleic Acids Res, 2017. **45**(11): p. 6971-6980.
13. Casini, A., et al., *A Pressure Test to Make 10 Molecules in 90 Days: External Evaluation of Methods to Engineer Biology*. J Am Chem Soc, 2018. **140**(12): p. 4302-4316.
14. Pardi, N., et al., *mRNA vaccines - a new era in vaccinology*. Nat Rev Drug Discov, 2018. **17**(4): p. 261-279.
15. Jung, J.K., et al., *Cell-free biosensors for rapid detection of water contaminants*. Nat Biotechnol, 2020. **38**(12): p. 1451-1459.
16. Keasling, J., et al., *Microbial production of advanced biofuels*. Nature Reviews Microbiology, 2021: p. 1-15.
17. Anderson, J.C., et al., *BglBricks: A flexible standard for biological part assembly*. J Biol Eng, 2010. **4**(1): p. 1.
18. Reis, A.C. and H.M. Salis, *An automated model test system for systematic development and improvement of gene expression models*. ACS Synthetic Biology, 2020. **9**(11): p. 3145-3156.
19. Grunberg, T.W. and D. Del Vecchio, *Modular analysis and design of biological circuits*. Current opinion in biotechnology, 2020. **63**: p. 41-47.
20. Mejía-Almonte, C., et al., *Redefining fundamental concepts of transcription initiation in bacteria*. Nature Reviews Genetics, 2020. **21**(11): p. 699-714.
21. Gorochofski, T.E., et al., *Genetic circuit characterization and debugging using RNA-seq*. Mol Syst Biol, 2017. **13**(11): p. 952.
22. Brewster, R.C., D.L. Jones, and R. Phillips, *Tuning promoter strength through RNA*

- polymerase binding site design in Escherichia coli*. PLoS computational biology, 2012. **8**(12): p. e1002811.
23. Van Brempt, M., et al., *Predictive design of sigma factor-specific promoters*. Nature communications, 2020. **11**(1): p. 1-13.
  24. Ruff, E.F., M.T. Record, Jr., and I. Artsimovitch, *Initial events in bacterial transcription initiation*. Biomolecules, 2015. **5**(2): p. 1035-62.
  25. Zhang, Y., et al., *Structural basis of transcription initiation*. Science, 2012. **338**(6110): p. 1076-80.
  26. Davis, C.A., et al., *Real-time footprinting of DNA in the first kinetically significant intermediate in open complex formation by Escherichia coli RNA polymerase*. Proc Natl Acad Sci U S A, 2007. **104**(19): p. 7833-8.
  27. Murakami, K.S., et al., *Structural basis of transcription initiation: an RNA polymerase holoenzyme-DNA complex*. Science, 2002. **296**(5571): p. 1285-90.
  28. Ross, W. and R.L. Gourse, *Sequence-independent upstream DNA-alphaCTD interactions strongly stimulate Escherichia coli RNA polymerase-lacUV5 promoter association*. Proc Natl Acad Sci U S A, 2005. **102**(2): p. 291-6.
  29. Ross, W., A. Ernst, and R.L. Gourse, *Fine structure of E. coli RNA polymerase-promoter interactions: alpha subunit binding to the UP element minor groove*. Genes Dev, 2001. **15**(5): p. 491-506.
  30. Haugen, S.P., et al., *Fine structure of the promoter-sigma region 1.2 interaction*. Proc Natl Acad Sci U S A, 2008. **105**(9): p. 3292-7.
  31. Henderson, K.L., et al., *Mechanism of transcription initiation and promoter escape by E. coli RNA polymerase*. Proc Natl Acad Sci U S A, 2017. **114**(15): p. E3032-E3040.
  32. Kapanidis, A.N., et al., *Initial transcription by RNA polymerase proceeds through a DNA-scrunching mechanism*. Science, 2006. **314**(5802): p. 1144-1147.
  33. Hsu, L.M., et al., *Initial transcribed sequence mutations specifically affect promoter escape properties*. Biochemistry, 2006. **45**(29): p. 8841-54.
  34. Saba, J., et al., *The elemental mechanism of transcriptional pausing*. Elife, 2019. **8**: p. e40981.
  35. SantaLucia, J., *A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics*. Proceedings of the National Academy of Sciences, 1998. **95**(4): p. 1460-1465.
  36. Sugimoto, N., et al., *Thermodynamic parameters to predict stability of RNA/DNA hybrid duplexes*. Biochemistry, 1995. **34**(35): p. 11211-11216.
  37. Hossain, A., et al., *Automated design of thousands of nonrepetitive parts for engineering stable genetic systems*. Nat Biotechnol, 2020. **38**(12): p. 1466-1475.
  38. Urtecho, G., et al., *Systematic Dissection of Sequence Elements Controlling sigma70 Promoters Using a Genomically Encoded Multiplexed Reporter Assay in Escherichia coli*. Biochemistry, 2019. **58**(11): p. 1539-1551.
  39. Yu, T.C., et al., *Multiplexed characterization of rationally designed promoter architectures deconstructs combinatorial logic for IPTG-inducible systems*. Nat Commun, 2021. **12**(1): p. 325.
  40. Yona, A.H., E.J. Alm, and J. Gore, *Random sequences rapidly evolve into de novo promoters*. Nature Communications, 2018. **9**(1): p. 1530.
  41. Taylor, N.D., et al., *Engineering an allosteric transcription factor to respond to new ligands*. Nature methods, 2016. **13**(2): p. 177.

42. Espah Borujeni, A., et al., *Automated physics-based design of synthetic riboswitches from diverse RNA aptamers*. Nucleic Acids Research, 2016. **44**: p. 1-13.
43. Skjoedt, M.L., et al., *Engineering prokaryotic transcriptional activators as metabolite biosensors in yeast*. Nature chemical biology, 2016. **12**(11): p. 951-958.
44. Brophy, J.A. and C.A. Voigt, *Principles of genetic circuit design*. Nature methods, 2014. **11**(5): p. 508-520.
45. Din, M.O., et al., *Synchronized cycles of bacterial lysis for in vivo delivery*. Nature, 2016. **536**(7614): p. 81.
46. Gupta, A., et al., *Dynamic regulation of metabolic flux in engineered bacteria using a pathway-independent quorum-sensing circuit*. Nat Biotechnol, 2017.
47. Smanski, M.J., et al., *Synthetic biology to access and expand nature's chemical diversity*. Nature reviews. Microbiology, 2016. **14**(3): p. 135.
48. Ng, C.Y., et al., *Rational design of a synthetic Entner–Doudoroff pathway for improved and controllable NADPH regeneration*. Metabolic engineering, 2015. **29**: p. 86-96.
49. Bentley, G.J., et al., *Engineering Escherichia coli to produce branched-chain fatty acids in high percentages*. Metabolic engineering, 2016. **38**: p. 148-158.
50. Brewster, R.C., D.L. Jones, and R. Phillips, *Tuning Promoter Strength through RNA Polymerase Binding Site Design in Escherichia coli*. PLoS Computational Biology, 2012. **8**.
51. Brewster, R.C., et al., *The transcription factor titration effect dictates level of gene expression*. Cell, 2014. **156**(6): p. 1312-1323.
52. Chen, Y.-J., et al., *Characterization of 582 natural and synthetic terminators and quantification of their design constraints*. Nature Methods, 2013. **10**: p. 659-663.
53. Cambray, G., et al., *Measurement and modeling of intrinsic transcription terminators*. Nucleic acids research, 2013. **41**(9): p. 5139-5148.
54. Espah Borujeni, A. and H.M. Salis, *Translation initiation is controlled by RNA folding kinetics via a ribosome drafting mechanism*. Journal of the American Chemical Society, 2016. **138**(22): p. 7016-7023.
55. Rodrigo, G., et al., *Model-based design of RNA hybridization networks implemented in living cells*. Nucleic Acids Research, 2017.
56. Smanski, M.J., et al., *Functional optimization of gene clusters by combinatorial design and assembly*. Nature biotechnology, 2014. **32**(12): p. 1241-1249.
57. Nielsen, A.A., et al., *Genetic circuit design automation*. Science, 2016. **352**(6281): p. aac7341.
58. Kushwaha, M. and H.M. Salis, *A portable expression resource for engineering cross-species genetic circuits and pathways*. Nature communications, 2015. **6**.
59. Hu, C.Y., J.D. Varner, and J.B. Lucks, *Generating effective models and parameters for RNA genetic circuits*. ACS synthetic biology, 2015. **4**(8): p. 914-926.
60. Reeve, B., et al., *Predicting translation initiation rates for designing synthetic biology*. Frontiers in bioengineering and biotechnology, 2014. **2**.
61. Jacobson, I., et al., *The unified software development process*. Vol. 1. 1999: Addison-wesley Reading.
62. Moult, J., *A decade of CASP: progress, bottlenecks and prognosis in protein structure prediction*. Current Opinion in Structural Biology, 2005. **15**: p. 285-289.
63. Prill, R.J., et al., *Towards a Rigorous Assessment of Systems Biology Models: The DREAM3 Challenges*. PLoS ONE, 2010. **5**.

64. Meyer, P., et al., *Industrial methodology for process verification in research (IMPROVER): toward systems biology verification*. Bioinformatics, 2012. **28**(9): p. 1193-1201.
65. Salis, H.M., E.A. Mirsky, and C.A. Voigt, *Automated design of synthetic ribosome binding sites to control protein expression*. Nature Biotechnology, 2009. **27**: p. 946-950.
66. Na, D., S. Lee, and D. Lee, *Mathematical modeling of translation initiation for the estimation of its efficiency to computationally design mRNA sequences with desired expression levels in prokaryotes*. BMC Systems Biology, 2010. **4**: p. 1-16.
67. Farasat, I., et al., *Efficient search, mapping, and optimization of multi-protein genetic systems in diverse bacteria*. Molecular Systems Biology, 2014. **10**: p. 1-18.
68. Espah Borujeni, A., A.S. Channarasappa, and H.M. Salis, *Translation rate is controlled by coupled trade-offs between site accessibility, selective RNA unfolding and sliding at upstream standby sites*. Nucleic Acids Research, 2014. **42**: p. 2646-2659.
69. Espah Borujeni, A., et al., *Precise quantification of translation inhibition by mRNA structures that overlap with the ribosomal footprint in N-terminal coding sequences*. Nucleic Acids Res, 2017.
70. Tian, T. and H.M. Salis, *A predictive biophysical model of translational coupling to coordinate and control protein expression in bacterial operons*. Nucleic Acids Research, 2015. **43**: p. 7137-7151.
71. Mimee, M., et al., *Programming a Human Commensal Bacterium, Bacteroides thetaiotaomicron, to Sense and Respond to Stimuli in the Murine Gut Microbiota*. Cell Systems, 2015. **1**: p. 62-71.
72. Goodman, D.B., G. Church, and S. Kosuri, *Causes and Effects of N-Terminal Codon Bias in Bacterial Genes*. Science, 2013. **342**: p. 475-479.
73. Espah Borujeni, A. and H.M. Salis, *Translation Initiation is Controlled by RNA Folding Kinetics via a Ribosome Drafting Mechanism*. Journal of the American Chemical Society, 2016. **138**: p. 7016-7023.
74. Beck, H.J., I.M. Fleming, and G.R. Janssen, *5'-Terminal AUGs in Escherichia coli mRNAs with Shine-Dalgarno Sequences: Identification and Analysis of Their Roles in Non-Canonical Translation Initiation*. PloS one, 2016. **11**(7): p. e0160144.
75. Beck, H.J. and G.R. Janssen, *Novel translation initiation regulation mechanism in Escherichia coli ptrB mediated by a 5'-terminal AUG*. Journal of Bacteriology, 2017: p. JB. 00091-17.
76. Egbert, R.G. and E. Klavins, *Fine-tuning gene networks using simple sequence repeats*. Proceedings of the National Academy of Sciences, 2012. **109**(42): p. 16817-16822.
77. Hecht, A., et al., *Measurements of translation initiation from all 64 codons in E. coli*. Nucleic acids research, 2017. **45**(7): p. 3615-3626.
78. Marzi, S., et al., *Structured mRNAs regulate translation initiation by binding to the platform of the ribosome*. Cell, 2007. **130**(6): p. 1019-31.
79. Kozak, M., *Regulation of translation via mRNA structure in prokaryotes and eukaryotes*. Gene, 2005. **361**: p. 13-37.
80. Kozak, M., *Initiation of translation in prokaryotes and eukaryotes*. Gene, 1999. **234**: p. 187-208.
81. Vellanoweth, R.L. and J.C. Rabinowitz, *The influence of ribosome-binding-site elements on translational efficiency in Bacillus subtilis and Escherichia coli in vivo*. Molecular Microbiology, 1992. **6**(9): p. 1105-1114.

82. Hongyun Chen, M.B., Ravindra Kumar and Ernest Jay, *Determinant of the optimal aligned spacing between the SD sequence and the start codon of E coli mRNAs*. Nucleic Acids Res, 1994. **22**(23): p. 4953-4957.
83. Crooks, G.E., et al., *WebLogo: A Sequence Logo Generator*. Genome Research, 2004. **14**: p. 1188-1190.
84. Andronescu, M., et al., *Computational approaches for RNA energy parameter estimation*. RNA, 2010. **16**(12): p. 2304-2318.
85. Sprengart, M.L., H.P. Fatscher, and E. Fuchs, *The initiation of translation in E. coli: apparent base pairing between the 16srRNA and downstream sequences of the mRNA*. Nucleic Acids Research, 1990. **18**(7): p. 1719-1723.
86. Wegmann, U., N. Horn, and S.R. Carding, *Defining the Bacteroides Ribosomal Binding Site*. Applied and Environmental Microbiology, 2013. **79**: p. 1980-1989.
87. Dykeman, E.C., *An implementation of the Gillespie algorithm for RNA kinetics with logarithmic time update*. Nucleic Acids Research, 2015. **43**: p. 5708-5715.
88. Mackie, G.A., *RNase E: at the interface of bacterial RNA processing and decay*. Nature Reviews Microbiology, 2013. **11**: p. 45-57.
89. Deana, A. and J.G. Belasco, *Lost in translation: the influence of ribosomes on bacterial mRNA decay*. Genes & development, 2005. **19**(21): p. 2526-2533.
90. Espah Borujeni, A., *The development of equilibrium and non-equilibrium models of translation and riboswitch regulation: Towards the automated design of cellular sensors*, in *Chemical Engineering*. 2015, The Pennsylvania State University.
91. Patterson, S.S., et al., *Codon optimization of bacterial luciferase (lux) for expression in mammalian cells*. J Ind Microbiol Biotechnol, 2005. **32**(3): p. 115-23.
92. Lutz, S., *Beyond directed evolution--semi-rational protein engineering and design*. Curr Opin Biotechnol, 2010. **21**(6): p. 734-43.
93. Boeke, J.D., et al., *GENOME ENGINEERING. The Genome Project-Write*. Science, 2016. **353**(6295): p. 126-7.
94. Gibson, D.G., et al., *Creation of a bacterial cell controlled by a chemically synthesized genome*. Science, 2010. **329**(5987): p. 52-6.
95. Hutchison, C.A., 3rd, et al., *Design and synthesis of a minimal bacterial genome*. Science, 2016. **351**(6280): p. aad6253.
96. Tian, S., et al., *Primerize: automated primer assembly for transcribing non-coding RNA domains*. Nucleic Acids Res, 2015. **43**(W1): p. W522-6.
97. Hoover, D.M. and J. Lubkowski, *DNAWorks: an automated method for designing oligonucleotides for PCR-based gene synthesis*. Nucleic Acids Res, 2002. **30**(10): p. e43.
98. Rouillard, J.M., et al., *Gene2Oligo: oligonucleotide design for in vitro gene synthesis*. Nucleic Acids Res, 2004. **32**(Web Server issue): p. W176-80.
99. Richardson, S.M., et al., *Automated Design of Assemblable, Modular, Synthetic Chromosomes*. Parallel Processing and Applied Mathematics, Part II, 2010. **6068**: p. 280-+.
100. Tang, N.C. and A. Chilkoti, *Combinatorial codon scrambling enables scalable gene synthesis and amplification of repetitive proteins*. Nat Mater, 2016. **15**(4): p. 419-24.
101. Jack, B.R., et al., *Predicting the Genetic Stability of Engineered DNA Sequences with the EFM Calculator*. ACS Synth Biol, 2015. **4**(8): p. 939-43.
102. Christen, M., S. Deutsch, and B. Christen, *Genome Calligrapher: A Web Tool for Refactoring Bacterial Genome Sequences for de Novo DNA Synthesis*. ACS Synth Biol,

2015. **4**(8): p. 927-34.
103. Oberortner, E., et al., *Streamlining the Design-to-Build Transition with Build-Optimization Software Tools*. ACS Synth Biol, 2017. **6**(3): p. 485-496.
  104. Palczewska, A., et al., *Interpreting random forest models using a feature contribution method*. 2013: p. 112-119.
  105. Reis, A.C., et al., *Simultaneous repression of multiple bacterial genes using nonrepetitive extra-long sgRNA arrays*. Nature biotechnology, 2019: p. 1-8.
  106. Hossain, A., et al., *Automated Design of Thousands of Highly Non-Repetitive Genetic Parts for Engineering Evolutionary Robust Genetic Systems*. Nature biotechnology. **in review**.
  107. Isabella, V.M., et al., *Development of a synthetic live bacterial therapeutic for the human metabolic disease phenylketonuria*. Nature Biotechnology, 2018. **36**: p. 857.
  108. June, C.H., et al., *CAR T cell immunotherapy for human cancer*. Science, 2018. **359**(6382): p. 1361-1365.
  109. Luo, X., et al., *Complete biosynthesis of cannabinoids and their unnatural analogues in yeast*. Nature, 2019. **567**(7746): p. 123.
  110. Whitaker, W.B., et al., *Engineering the biological conversion of methanol to specialty chemicals in Escherichia coli*. Metabolic engineering, 2017. **39**: p. 49-59.
  111. Moser, F., et al., *Light-Controlled, High-Resolution Patterning of Living Engineered Bacteria Onto Textiles, Ceramics, and Plastic*. Advanced Functional Materials, 2019: p. 1901788.
  112. Roggo, C. and J.R. van der Meer, *Miniaturized and integrated whole cell living bacterial sensors in field applicable autonomous devices*. Current opinion in biotechnology, 2017. **45**: p. 24-33.
  113. Hughes, R.A. and A.D. Ellington, *Synthetic DNA Synthesis and Assembly: Putting the Synthetic in Synthetic Biology*. Cold Spring Harbor Perspectives in Biology, 2017. **9**: p. a023812.
  114. Gibson, D.G., *Synthesis of DNA fragments in yeast by one-step assembly of overlapping oligonucleotides*. Nucleic Acids Research, 2009. **37**: p. 6984-6990.
  115. Kosuri, S. and G.M. Church, *Large-scale de novo DNA synthesis: technologies and applications*. Nature Methods, 2014. **11**: p. 499.
  116. Hua, S.B., et al., *Minimum length of sequence homology required for in vivo cloning by homologous recombination in yeast*. Plasmid, 1997. **38**: p. 91-6.
  117. Fujitani, Y., K. Yamamoto, and I. Kobayashi, *Dependence of frequency of homologous recombination on the homology length*. Genetics, 1995. **140**: p. 797-809.
  118. Shen, P. and H.V. Huang, *Homologous recombination in Escherichia coli: dependence on substrate length and homology*. Genetics, 1986. **112**: p. 441-57.
  119. Jack, B.R., et al., *Predicting the genetic stability of engineered DNA sequences with the EFM calculator*. ACS synthetic biology, 2015. **4**(8): p. 939-943.
  120. Gorochofski, T.E., et al., *Genetic circuit characterization and debugging using RNA-seq*. Molecular systems biology, 2017. **13**(11): p. 952.
  121. Gander, M.W., et al., *Digital logic circuits in yeast with CRISPR-dCas9 NOR gates*. Nature communications, 2017. **8**: p. 15459.
  122. Casini, A., et al., *A Pressure Test to Make 10 Molecules in 90 Days: External Evaluation of Methods to Engineer Biology*. Journal of the American Chemical Society, 2018. **140**: p. 4302-4316.

123. Shen, J.P., et al., *Combinatorial CRISPR-Cas9 screens for de novo mapping of genetic interactions*. Nature methods, 2017. **14**: p. 573-576.
124. Najm, F.J., et al., *Orthologous CRISPR-Cas9 enzymes for combinatorial genetic screens*. Nature biotechnology, 2018. **36**: p. 179-189.
125. Kaczmarzyk, D., et al., *Diversion of the long-chain acyl-ACP pool in Synechocystis to fatty alcohols through CRISPRi repression of the essential phosphate acyltransferase PIsX*. Metabolic engineering, 2018. **45**: p. 59-66.
126. Nielsen, A.A.K., et al., *Genetic circuit design automation*. Science, 2016. **352**: p. aac7341.
127. Fernandez-Rodriguez, J., et al., *Engineering RGB color vision into Escherichia coli*. Nature chemical biology, 2017. **13**: p. 706-708.
128. Peeters, B.P., et al., *Structural plasmid instability in Bacillus subtilis: effect of direct and inverted repeats*. Molecular and General Genetics MGG, 1988. **212**(3): p. 450-458.
129. Yao, X.-D. and D.H. Evans, *Effects of DNA structure and homology length on vaccinia virus recombination*. Journal of virology, 2001. **75**(15): p. 6923-6932.
130. Urtecho, G., et al., *Systematic Dissection of Sequence Elements Controlling  $\sigma$ 70 Promoters Using a Genomically-Encoded Multiplexed Reporter Assay in E. coli*. Biochemistry, 2018: p. acs.biochem.7b01069.
131. Alper, H., et al., *Tuning genetic control through promoter engineering*. Proceedings of the National Academy of Sciences, 2005. **102**: p. 12678-12683.
132. Chen, Y.J., et al., *Characterization of 582 natural and synthetic terminators and quantification of their design constraints*. Nature Methods, 2013. **10**: p. 659-664.
133. Cambray, G., et al., *Measurement and modeling of intrinsic transcription terminators*. Nucleic Acids Res, 2013. **41**: p. 5139-5148.
134. Redden, H. and H.S. Alper, *The development and characterization of synthetic minimal yeast promoters*. Nature Communications, 2015. **6**: p. 7810.
135. Curran, K.A., et al., *Design of synthetic yeast promoters via tuning of nucleosome architecture*. Nature Communications, 2014. **5**: p. 4002.
136. Weenink, T., R.M. McKiernan, and T. Ellis, *Rational Design Of RNA Structures That Predictably Tune Eukaryotic Gene Expression*. bioRxiv, 2017: p. 137877.
137. Cuperus, J.T., et al., *Deep learning of the regulatory grammar of yeast 5'untranslated regions from 500,000 random sequences*. Genome Research, 2017. **27**: p. 2015-2024.
138. Curran, K.A., et al., *Short Synthetic Terminators for Improved Heterologous Gene Expression in Yeast*. ACS Synthetic Biology, 2015. **4**: p. 824-832.
139. Karp, R.M., *Reducibility among combinatorial problems*, in *Complexity of computer computations*. 1972, Springer. p. 85-103.
140. Bar-Yehuda, R. and S. Even, *A linear-time approximation algorithm for the weighted vertex cover problem*. Journal of Algorithms, 1981. **2**: p. 198-203.
141. Green, A.A., et al., *Toehold switches: de-novo-designed regulators of gene expression*. Cell, 2014. **159**(4): p. 925-939.
142. Chen, Y.-J., et al., *Characterization of 582 natural and synthetic terminators and quantification of their design constraints*. Nature methods, 2013. **10**(7): p. 659.
143. Lorenz, R., et al., *ViennaRNA Package 2.0*. Algorithms for Molecular Biology, 2011. **6**: p. 26.
144. Browning, D.F. and S.J. Busby, *The regulation of bacterial transcription initiation*. Nature Reviews Microbiology, 2004. **2**(1): p. 57.

145. Johns, N.I., et al., *Metagenomic mining of regulatory elements enables programmable species-selective gene expression*. Nature methods, 2018. **15**(5): p. 323.
146. Meysman, P., et al., *Structural Properties of Prokaryotic Promoter Regions Correlate with Functional Features*. PLoS ONE, 2014. **9**: p. e88717.
147. Khuu, P., et al., *Phylogenomic analysis of the emergence of GC-rich transcription elements*. Proceedings of the National Academy of Sciences, 2007. **104**: p. 16528-16533.
148. Vassilyev, D.G., et al., *Crystal structure of a bacterial RNA polymerase holoenzyme at 2.6 Å resolution*. Nature, 2002. **417**(6890): p. 712.
149. Becht, E., et al., *Dimensionality reduction for visualizing single-cell data using UMAP*. Nature biotechnology, 2019. **37**(1): p. 38.
150. de Boer, C.G., et al., *Deciphering eukaryotic gene-regulatory logic with 100 million random promoters*. Nature biotechnology, 2020. **38**(1): p. 56-65.

## Appendix A: Publications

### Published Papers

Reis, A. C., Halper, S. M., Vezeau, G. E., Cetnar, D. P., Hossain, A., Clauer, P. R., & Salis, H. M. (2019). Simultaneous repression of multiple bacterial genes using nonrepetitive extra-long sgRNA arrays. *Nature biotechnology*, 37(11), 1294-1301.

Hossain, Ayaan, Eriberto Lopez, Sean M. Halper, Daniel P. Cetnar, Alexander C. Reis, Devin Strickland, Eric Klavins, and Howard M. Salis. "Automated design of thousands of nonrepetitive parts for engineering stable genetic systems." *Nature biotechnology* 38, no. 12 (2020): 1466-1475.

Cetnar, Daniel P., and Howard M. Salis. "Systematic quantification of sequence and structural determinants controlling mRNA stability in bacterial operons." *ACS Synthetic Biology* 10, no. 2 (2021): 318-332.

Halper, S. M., Hossain, A., & Salis, H. M. (2020). Synthesis success calculator: predicting the rapid synthesis of DNA fragments with machine learning. *ACS Synthetic Biology*, 9(7), 1563-1571.

Reis, Alexander C., and Howard M. Salis. "An automated model test system for systematic development and improvement of gene expression models." *ACS Synthetic Biology* 9, no. 11 (2020): 3145-3156.

Halper, Sean M., Daniel P. Cetnar, and Howard M. Salis. "An automated pipeline for engineering many-enzyme pathways: Computational sequence design, pathway expression-flux mapping, and scalable pathway optimization." In *Synthetic Metabolic Pathways*, pp. 39-61. Humana Press, New York, NY, 2018.

Vezeau, Grace E., and Howard M. Salis. "Tuning Cell-free Composition Controls the Time-delay, Dynamics, and Productivity of TX-TL Expression." *bioRxiv* (2021).

La Fleur, Travis L., Ayaan Hossain, and Howard M. Salis. "Automated Model-Predictive Design of Synthetic Promoters to Control Transcriptional Profiles in Bacteria." bioRxiv (2021).

### Unpublished Papers

Cetnar, Daniel P., Ayaan Hossain, Grace E. Vezeau, and Howard M. Salis.  
"Comprehensive modeling and design of 5' UTRs for RNA stability using a 62,000 unique 5' UTR Library" *in preparation*

## Appendix B: Abbreviations and Glossary of Terminology

**DNA:** Deoxyribonucleic acid. A linear polymer containing a phosphate backbone and four different nucleotide monomers, joined together by phosphodiester bonds. Nucleotides are adenine (A), guanine (G), cytosine (C), and thymine (T). Exists in single-stranded or double-stranded form. Double-stranded DNA is non-covalently bound together by hydrogen bonds, using Watson-Crick base pairings (A to T, G to C).

The macromolecule responsible for carrying all genetic information: the instructions that enable organisms to perform all cellular processes, including metabolism and self-replication.

Errors in DNA replication or damage to DNA are a cause of major human disease, including cancer.

**RNA:** Ribonucleic acid. A linear polymer with two key chemical differences, compared to DNA. First, the ribose sugars in the nucleotide monomers contain a hydroxyl (-OH) group at the 2' position. The hydroxyl group increases the reactivity of RNA, compared to DNA. Second, the nucleotide uracil is used in place of thymine. Similar to DNA, ribonucleotides within RNA participate in Watson-Crick base pairings. In RNA, uracil forms two hydrogen bond interactions with adenine. A single strand of RNA can form Watson-Crick base pairings with itself, creating RNA structures. Two RNA molecules can hybridize to form double-stranded RNA molecules.

Inside cells, RNA carries out diverse tasks. They are used to prime DNA replication (RNA primers), carry temporary copies of the genetic information (messenger RNAs), as regulators of protein expression (regulatory RNAs), as the functional linkage between RNA and protein (transfer RNAs), as the superstructure of the ribosome (ribosomal RNA), and even as catalysts (ribozymes).

RNA can be converted back to DNA by the enzyme reverse transcriptase.

**protein:** A polypeptide. A linear polymer made up of at least 20 different amino acid units, joined together by peptide bonds. Their amino acid sequence is determined by the organism's DNA sequence, according to the genetic code.

Amino acids vary in size, charge, hydrophobicity, and shape. Proteins form complex three-dimensional structures that determine their function. Their structure is determined by hydrophobic, van Der Waals, electrostatic, and hydrogen bond interactions.

Proteins are the workhorse of the cell with diverse functions. They are structural subunits in large macromolecular complexes. They are catalysts of chemical reactions (enzymes). They bind specifically to small molecules (ions, sugars, carbohydrates, fatty acids, ...), segments of DNA or RNA, and other proteins.

### The Central Dogma

**DNA self-replication:** The enzyme-catalyzed, largely error-free copying of double-stranded DNA

Inside cells, a multi-enzyme complex carries out this process, consisting of a DNA helicase, an RNA primase, a DNA polymerase, and a DNA ligase.

Outside of cells, DNA replication can be performed in a Polymerase Chain Reaction (PCR), using DNA polymerase and DNA primers.

**Transcription:**

The production of RNA inside the cell, using DNA as a template, catalyzed by the enzyme RNA polymerase. The segment of DNA where transcription is initiated is called the promoter. Transcription has three phases: initiation, elongation, and termination. During initiation, a polymerase initiation complex (PIC), containing RNA polymerase and other factors, assembles on a promoter and initiates RNA synthesis. Elongation of the growing RNA molecule by continues until a special RNA structure, called a transcriptional terminator, forms that pulls the growing RNA molecule out of the catalytic site of RNA polymerase.

Transcription is a regulated process, where production of RNA is controlled by extracellular and intracellular signals. This is one of the primary ways an organism reacts to its environment. Many auxiliary proteins are responsible for carrying out the regulation of transcription. In bacteria, transcription is regulated by sigma factors and transcription factors. In eukaryotes, transcription is more tightly controlled by nucleosomes, general transcription factors, enhancer-binding proteins, and chromatin modifiers.

**Translation:**

The production of protein inside the cell, using messenger RNA (mRNA) as a template, catalyzed by the ribosome. Translation has three phases: initiation, elongation, and termination. During initiation, the ribosome supramolecular complex, containing the large and small subunit as well as initiation factors, assembles on a segment of mRNA, and begins protein synthesis at a start codon (AUG, GUG, UUG, or CUG), which encodes the amino acid methionine. A codon is a nucleotide triplet.

Elongation begins when a tRNA, carrying an amino acid, binds to a codon in the mRNA within the P-site of the ribosome. The amino acid is then covalently incorporated into the growing polypeptide chain. The ribosome will continue to move forward, in a ratcheting motion, until it reaches a stop codon (UAA, UGA, or UAG) and terminates polypeptide synthesis.

There are 64 possible codons and 64 corresponding tRNAs. The interaction between codons in the mRNA and anti-codons in the tRNA define the genetic code, which determines the amino acid is incorporated into the growing polypeptide when the ribosome reads each codon. There are only 20 natural amino acids and multiple tRNAs may carry the same amino acid, leading to degeneracy (many-to-one mapping) in the genetic code.

The nucleotide sequence between the start and stop codon of a mRNA is called the protein coding sequence (CDS) or an open reading frame (ORF).

In bacteria: a messenger RNA can contain multiple protein coding sequences, each with its own ribosome binding site. This is called an operon. The segment of RNA where the ribosome binds is called the ribosome binding site. The ribosome binding site can appear anywhere inside the mRNA.

In eukaryotes: mRNA is covalently modified at the 5' and 3' ends, enabling it to form a circular loop. The ribosome binds to the 5' end of the mRNA and ratchets ("scans") forward until it reaches a start codon. The ribosome is paused in front of the start codon by a RNA sequence, called the Kozak sequence, enabling it to initiate translation. Eukaryotic mRNAs typically encode only a single protein coding sequence. However, there are special Internal Ribosome Entry Sites (IRES)

sequences that enable ribosome scanning to begin mid-way on the mRNA, albeit at a lower efficiency.

|                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Eukaryotic splicing: | Eukaryotic mRNAs contain both coding (exons) and non-coding (introns) portions of the protein coding sequence. After mRNA transcription, the nuclear pore complex is responsible for cutting out the intron sequences and exporting the mature mRNA into the cytoplasm. Eukaryotic cells can regulate splicing, selecting which exons are incorporated into the mature mRNA. This process is often used to swap domains in a large protein, in response to changes in environmental signals. |
| mRNA degradation:    | messenger RNAs are destroyed by hydrolysis, catalyzed by RNases. Certain RNA structures can promote or prevent degradation by RNases.                                                                                                                                                                                                                                                                                                                                                        |
| Protein degradation: | proteins are degraded by the proteasome, a multi-protein catalytic machine that breaks down proteins into amino acids. In bacteria, peptide tags can guide the protein to the proteasome for faster degradation. In eukaryotes, proteins are covalently modified (ubiquitinated), which acts to recruit the protein to the proteolytic machinery for rapid degradation.                                                                                                                      |

#### Important proteins & complexes involved in gene expression

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DNA polymerase:        | the enzyme responsible for replicating DNA. Always adds a nucleotide to the end of an existing DNA molecule at its 3' OH end. Requires a primer to initiate DNA synthesis. May have proof-reading activities that improve the fidelity of DNA replication. Thermostable DNA polymerases are used in PCR.                                                                                                                                                                                                                                                                                                        |
| RNA polymerase:        | RNAP, the enzyme responsible for carrying out transcription. In bacteria, binds to sigma factors that guide it to specific promoter sequences. In eukaryotes, RNAP is part of a large multi-protein complex that regulates transcription at a promoter.                                                                                                                                                                                                                                                                                                                                                         |
| the ribosome:          | <p>large supramolecular complex, made of two subunits (large and small). Responsible for carrying out protein synthesis inside cells.</p> <p>In bacteria, the ribosome contains 55 ribosomal proteins and 3 large ribosomal RNAs (5S, 16S and 23S).</p> <p>In eukaryotes, the ribosome contains 82 proteins and four ribosomal RNAs (5S, 5.8S, 28S, and 18S). The S stands for Svedberg unit, a metric to quantify the rate of sedimentation.</p> <p>Contains the A-, P-, and E- sites, where amino acid-carrying tRNAs enter the ribosome, deliver their amino acid for covalent polymerization, and exit.</p> |
| Ribonucleases:         | <p>referred to as "RNases". Degrades, chews, or cuts RNA by hydrolysis. Responsible for processing ribosomal RNAs, degrading mRNAs and regulatory RNAs, or involved in eukaryotic splicing.</p> <p>Various modes of action, including 5' exonuclease activity, endonuclease activity on double-stranded RNA regions, and endonuclease activity at specific RNA structures.</p>                                                                                                                                                                                                                                  |
| transcription factors: | proteins that bind to specific segments of DNA (the DNA sites are called operators or transcription factor binding sites). Transcription factors interact with RNA polymerase or other transcription factors to increase (activate) or decrease (repress) the rate of transcription. The DNA-binding activity of transcription factors                                                                                                                                                                                                                                                                          |

can be turned "on" or "off" by binding to small molecules, RNA, or other proteins, or by covalent modifications (e.g. phosphorylation). Multiple transcription factors can bind cooperatively or anti-cooperatively to their DNA sites, creating forms of logic.

histones, nucleosomes & chromatin: histones are the core protein subunits of the nucleosome. 8 histones form 1 nucleosome particle. 147 base pairs of DNA are wrapped around one nucleosome. Multiple DNA-nucleosome particles can compact together to form chromatin. Highly condensed DNA-nucleosome fibers are called heterochromatin. The binding of DNA to nucleosomes and the DNA's accessibility to other proteins, such as RNA polymerase, can regulate its transcription rate.

### Genetic parts that control gene expression

promoter: a segment of DNA where the RNA polymerase and other proteins in the polymerase initiation complex assemble in order to initiate transcription

transcriptional terminator: a segment of RNA that forms a strong hairpin secondary structure, followed by a polyU sequence. Transcriptional terminators fold very quickly inside the RNA polymerase enzyme, causing the end of the growing RNA chain to be yanked from the RNAP's catalytic site and ending transcription.

ribosome binding site (bacteria): in bacteria, a segment of RNA that binds to the ribosome and is responsible for controlling the rate of translation initiation

Kozak sequence (eukaryotes): in eukaryotes, a segment of RNA that causes the ribosome to pause, allowing it to recognize a start codon and initiate translation

codons: a nucleotide triplet that instructs the ribosome to incorporate a specific amino acid into the end of a growing polypeptide. A protein coding sequence always begins with a start codon (AUG, GUG, UUG, or CUG) and ends with a stop codon (UAA, UGA, or UAG).

### Manipulating recombinant DNA

plasmids: a circular double-stranded DNA molecule that carries an origin of DNA replication and a selective marker. A carrier for recombinant DNA for diverse purposes. Applications: expression of recombinant proteins inside micro-organisms; encoding synthetic metabolic pathways for production of chemicals; and studying gene regulation or other cellular processes. Certain bacteria and yeast species are more likely to stably replicate plasmids.

restriction endonucleases: enzymes that bind to specific segments of double-stranded DNA and cut. Some restriction enzymes create overhangs of single-stranded DNA ("sticky-ends").

DNA ligases: enzymes that covalently attach two double-stranded segments of DNA together. Often used in conjunction with restriction endonucleases.

oligonucleotides: short, single-stranded DNA molecules. Used as DNA primers in PCR.

melting: the act of heating a solution of double-stranded DNA, separating the DNA into single-stranded DNA molecules

|                                  |                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| annealing:                       | the act of cooling a solution of single-stranded DNA molecules, allowing Watson-Crick base pairs to form that will create double-stranded DNA molecules                                                                                                                                                                                                                                                                                         |
| polymerase chain reaction (PCR): | an iterative 3-step process that selectively amplifies ("copies") a portion of a DNA molecule. PCR combines a thermostable DNA polymerase, DNA primers, and a DNA template together through cycles of DNA melting, annealing of DNA primers and DNA template, and followed by DNA replication. By designing the DNA primers, selective portions of a larger DNA molecule can be copied, and/or nucleotides in the DNA sequence can be modified. |
| DNA kinases, phosphatases:       | a DNA kinase is an enzyme that phosphorylates the 5' hydroxyl end of a DNA molecule                                                                                                                                                                                                                                                                                                                                                             |
| DNA molecule                     | a DNA phosphatase is an enzyme that removes the 5' phosphate group of a DNA molecule                                                                                                                                                                                                                                                                                                                                                            |
| modulate ligase activity.        | DNA ligation requires a 5' phosphate group; these enzymes can thus modulate ligase activity.                                                                                                                                                                                                                                                                                                                                                    |
| gel electrophoresis:             | an apparatus for separating DNA molecules by size according to a                                                                                                                                                                                                                                                                                                                                                                                |
| electrophoretic gradient.        | The apparatus contains a cathode and anode, separated by a buffer reservoir, where an agarose or acrylamide gel is placed. DNA samples are inserted into the porous gel and will migrate from the anode to cathode (negatively charged DNA is attractive to the positively charged cathode). The rate of DNA mobility inside the porous gel matrix is inversely proportional to the DNA molecules' length.                                      |
| thermocycler:                    | a thermal block that holds PCR reactions and performs cycles of heating and cooling, from 0 to 100°C, according to a desired "PCR program".                                                                                                                                                                                                                                                                                                     |
| NCBI                             | National Center for Biotechnology Information                                                                                                                                                                                                                                                                                                                                                                                                   |
| UP                               | upstream                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| GC                               | guanine-cytosine                                                                                                                                                                                                                                                                                                                                                                                                                                |
| NEB                              | New England Biolabs                                                                                                                                                                                                                                                                                                                                                                                                                             |
| TSS                              | transcription start site                                                                                                                                                                                                                                                                                                                                                                                                                        |
| LB                               | lysogeny broth                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| cDNA                             | complementary DNA                                                                                                                                                                                                                                                                                                                                                                                                                               |
| AUROC                            | area under the receiver operating characteristic                                                                                                                                                                                                                                                                                                                                                                                                |
| ROC                              | receiver operating characteristic                                                                                                                                                                                                                                                                                                                                                                                                               |
| aSD                              | anti-Shine-Dalgarno                                                                                                                                                                                                                                                                                                                                                                                                                             |
| EMOPEC                           | Empiric Model and Oligos for Protein Expression Changes                                                                                                                                                                                                                                                                                                                                                                                         |
| IUPAC                            | The International Union of Pure and Applied Chemistry                                                                                                                                                                                                                                                                                                                                                                                           |

|          |                                                                       |
|----------|-----------------------------------------------------------------------|
| ANOVA    | analysis of variance                                                  |
| UMAP     | Uniform Manifold Approximation and Projection for Dimension Reduction |
| NTP      | nucleoside triphosphate                                               |
| TX       | transcription                                                         |
| CASP     | Critical Assessment of protein Structure Prediction                   |
| DREAM    | Dialogue on Reverse Engineering Assessment and Methods                |
| IMPROVER | Industrial Methodology for Process Verification in Research           |
| EFM      | Evolutionary Failure Mode                                             |
| CRISPR   | clustered regularly interspaced short palindromic repeats             |
| SgRNA    | single guide RNA                                                      |
| CPU      | central processing unit                                               |
| NuPoP    | Nucleosome Positioning Prediction                                     |
| ACS      | American Chemical Society                                             |