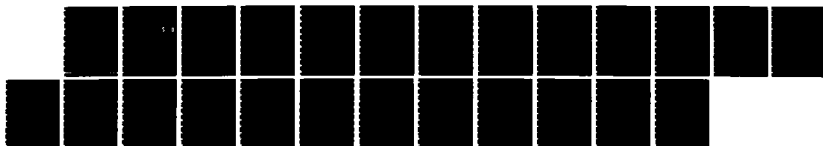
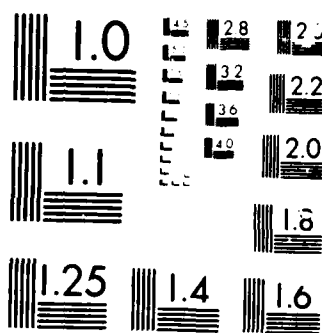


AD-A176 415 ON THE GENERATION OF ORGANIZATIONAL ARCHITECTURES USING 1/1
PETRI NETS(U) MASSACHUSETTS INST OF TECH CAMBRIDGE LAB
FOR INFORMATION AND D. P A REMY ET AL DEC 86
UNCLASSIFIED LIDS-P-1634 N80014-84-K-0519 F/G 9/2 NL





AD-A176 415

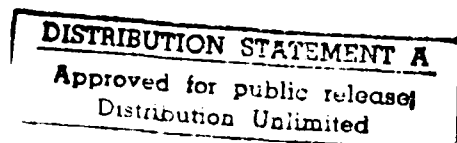
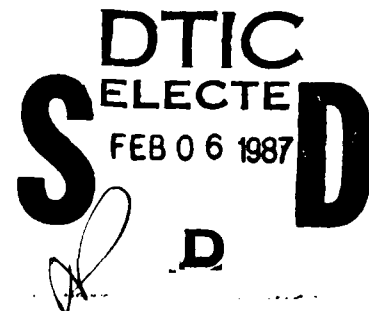
DECEMBER 1986

LIDS-P-1634

ON THE GENERATION OF ORGANIZATIONAL ARCHITECTURES
USING PETRI NETS *

by

Pascal A. Remy **
Alexander H. Levis **



ABSTRACT

A methodology is presented for generating architectures for decisionmaking organizations that satisfy some generic structural properties, as well as more specific designer's requirements. Petri Nets are used as the basic technique to represent organizational architectures. The allowable set of interactions among the organization members is first defined, and a mathematical framework is developed to represent the interactions between organization members. The set of organizational architectures satisfying both the structural and the designer's requirements is then analyzed. This set is delimited by its minimal and maximal elements and a technique is given to generate the entire set from its boundaries. Simple paths are used as the incremental unit leading from one organizational form in the set to its neighboring ones. The methodology has been implemented on a personal computer; a description of the different modules of the program is provided.

* This work was carried out at the MIT Laboratory for Information and Decision Systems with partial support by the U.S. Office of Naval Research under Contract No. N00014-84-K-0519 and by the Joint Directors of Laboratories under Contract No. ONR/N00014-85-K-0782.

** Room 35-410, Laboratory for Information and Decision Systems, MIT, Cambridge, MA 02139.

OTIC FILE COPY

INTRODUCTION

Most of the theoretical developments in decision and control theory have addressed the problem of analyzing the performance of a given organizational form. In this case, the organizational structure is fixed and well defined. Some changes in the topology of the organization may occasionally be made in order to improve its performance, but they always remain incremental. There is a need for a methodology to generate in some orderly manner organizational architectures that are not just variants of the same structure.

Up to now, information processing and decisionmaking organizations have been modeled and analyzed using Petri Nets [1],[2],[3],[4]. Petri Nets are indeed a powerful and convenient tool to represent and study a given organizational structure. When it comes to the problem of designing organizational forms - i.e. when no structure is given a priori -, the general Petri Net theory alone is of little use because of its generality. The scope of organizational forms that can be modeled by Petri Nets is only limited by the imagination of the designer. To make the problem tractable, a framework needs to be defined that will restrict the class of organizational structures under consideration. The complete design methodology is outlined below.

A mathematical model of interactions between decisionmakers is first defined, using, as a starting point, the four stage representation of the single interacting decisionmaker described by Levis [5]. This mathematical model defines the framework within which the design problem will be articulated. The model allows the organization designer to characterize with an arbitrary level of precision the class of organizations he - or she - is considering. The specificity of the designer's requirements will determine the degrees of freedom left. In addition to the designer's requirements, the organizational structures to be generated must satisfy a set of structural constraints reflecting some generic properties. The design problem consists in finding the set of all the organizational architectures that satisfy both the designer's requirements and the structural constraints. To investigate this set, a partial order is defined that allows for a classification of organizations. Using this order, the set is delimited by its minimal and maximal elements. A technique is then presented to generate the entire set from its boundaries. Simple paths are used as the incremental unit with which organizational structures are generated. It is shown that the Petri Nets representing the structures so obtained are event graphs.

The overall procedure has been implemented on a personal computer and a program

with a user interface is available. It allows the organization designer to go step by step through the entire design methodology.

ORGANIZATIONAL CLASSES

This section introduces the general framework defining the class of organizational forms under consideration.

The Four Stage Model of the Interacting Decisionmaker

The first step of a methodology for designing decisionmaking organizations is the modeling of a single decisionmaker. A somewhat simplified version of the four stage model is reproduced in Figure 1. Note that the notation of Petri Net theory is used.

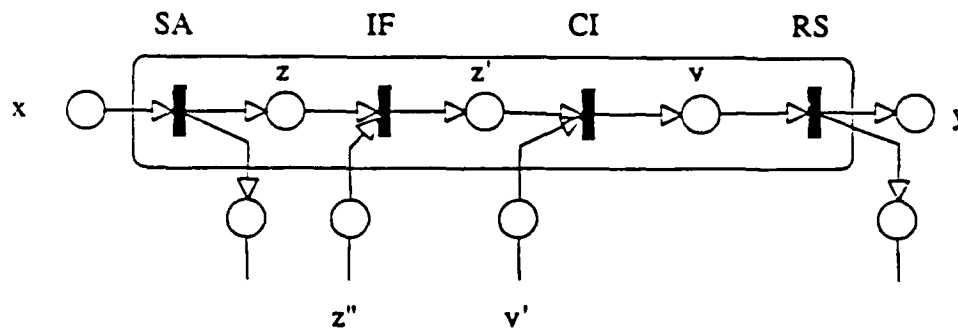


Figure 1 Four stage model of a decisionmaker*

The decisionmaker receives a signal x - from the external environment or from another organization member. The situation assessment (SA) stage contains algorithms that process the incoming signal to obtain the assessed situation z . The assessed situation z may be shared with other members. Concurrently, the decisionmaker can receive a signal z'' from another part of the organization; z'' and z are then merged together in the information

* The figures in this paper were drawn with Design © Meta Software, Inc. Cambridge, MA, 02138.

fusion (IF) stage to produce z' . The possibility of receiving commands from other organization members is reflected in the variable v' . The command interpretation (CI) stage will combine z' and v' to produce the variable v that contains z' and the appropriate strategy to use in the response selection (RS) stage. Finally, the RS stage contains algorithms that will produce the output y .

This model shows explicitly at which stage a decisionmaker can interact either with the external environment or with other organization members. A decisionmaker need not have all four stages. If any two stages are present, however, their intermediate stages must also be present, e. g., if the SA and CI stages are present, then the IF stage must also be present.

Interactions among Decisionmakers

The set of all allowable interactions is represented in Figure 2. Links from DM^i to DM^j only have been represented. Symmetrical links from DM^j to DM^i are of course valid interactions.

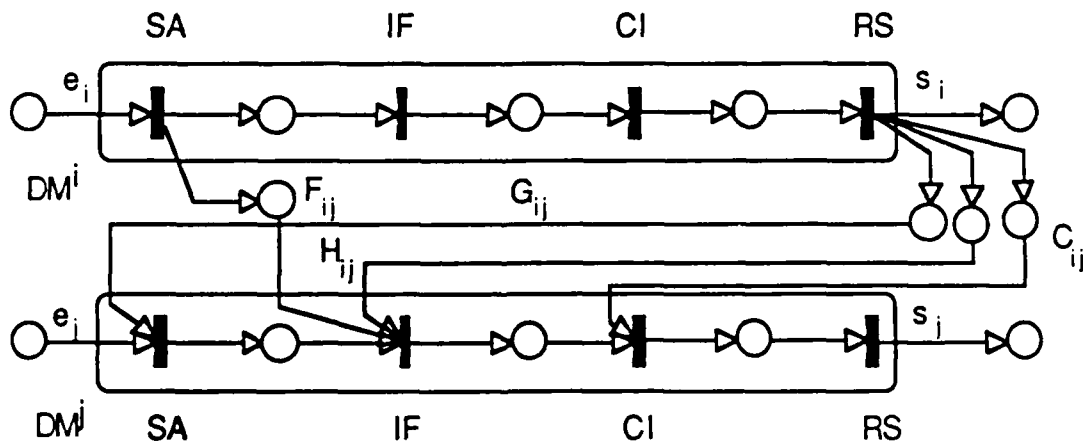


Figure 2 Allowable interactions

There are four possible links from a decisionmaker to another one and the maximum number of links, k_{max} , in a n -decisionmaker organization is therefore

$$k_{\max} = 4n^2 - 2n. \quad (1)$$

Mathematical Representation of Interactions

The previous analysis leads to a mathematical representation of interactions between decisionmakers. The labels $e_i, s_i, F_{ij}, G_{ij}, H_{ij}, C_{ij}$ of Figure 2 will be integer variables taking values in $\{0,1\}$ where 1 will indicate that the corresponding directed link is actually present in the organization, while 0 will reflect the absence of the link.

These variables will be aggregated into two vectors $\underline{e}$ and $\underline{s}$, and four matrices F, G, H , and C . The interaction structure of a n -decisionmaker organization will therefore be represented by the following six arrays.

- Two $n \times 1$ vectors $\underline{e}$ and $\underline{s}$, representing the interactions between the external environment and the organization:

$$\underline{e} \equiv [e_i]; \quad \underline{s} \equiv [s_i]; \quad i = 1, 2, \dots, n. \quad (2)$$

- Four $n \times n$ matrices F, G, H, C representing the interactions between decisionmakers inside the organization:

$$F \equiv [F_{ij}]; \quad G \equiv [G_{ij}]; \quad H \equiv [H_{ij}]; \quad C \equiv [C_{ij}] \quad i = 1, 2, \dots, n \quad \text{and} \quad j = 1, 2, \dots, n. \quad (3)$$

The six-tuple $(\underline{e}, \underline{s}, F, G, H, C)$ will be called a **Well Defined Net (WDN)** of dimension n , where n is the number of decisionmakers in the organization. The set of all Well Defined Nets of dimension n will be denoted Ψ^n . It is clear that Ψ^n is isomorphic to the set $\{0,1\}^{k_{\max}}$, where $k_{\max}$ is given by eq.(1). The dimension of Ψ^n is therefore

$$2^{k_{\max}} = 2^{4n^2 - 2n}. \quad (4)$$

The notion of a **subnet** of a WDN can be defined as follows. Let $\Pi = (\underline{e}, \underline{s}, F, G, H, C)$ and $\Pi' = (\underline{e}', \underline{s}', F', G', H', C')$ be two WDNs. The WDN P is a subnet of P' if and only if

$$\underline{e}' \leq \underline{e} \quad F' \leq F \quad G' \leq G$$

$$\underline{s}' \leq \underline{s} \quad H' \leq H \quad C' \leq C$$

where the inequality between arrays is interpreted element by element.

In other words, Π' is a subnet of Π if any interaction in Π' , i.e. a 1 in any of the arrays $\underline{e}', \underline{s}', F', G', H', C'$, is also an interaction in Π . The union of two subnets Π_1 and Π_2 of a WDN Π , is a new net that contains all the interactions that appear in either Π_1 or Π_2 or both.

PETRI NET REPRESENTATION OF ORGANIZATIONAL CLASSES

The mathematical model of interactions presented above restricts the class of organizations under consideration to the class of Well Defined Nets (WDNs). The next step of the methodology consists of translating the matrix representation associated with a WDN into a Petri Net representation. The analytical tools of Petri Net theory will then become available; they will be used to generate organizational architectures.

Transitions

Each stage of the four stage model of the single interacting decisionmaker will be represented by a single transition. A decisionmaker will therefore have at most four transitions and a n-decisionmaker organization will contain a maximum of 4n internal transitions. Two supplementary transitions are necessary to represent the external environment acting at both ends of the organization, so that the maximum number of transitions in the Petri Net representation of a WDN will be 4n+2. The transition corresponding to stage i ($1 \leq i \leq 4$) of decisionmaker DM^j ($1 \leq j \leq n$) will be labeled t_{ij} . The input and output transitions will be respectively t_0 and t_5 .

Places

A distinction has to be made among the places of the Petri Net representation of a WDN. **Interactional** places will refer to those places that correspond to interactions

between two different decisionmakers or between a decisionmaker and the external environment. **Internal** places will correspond to connections that remain within the boundaries of a single decisionmaker. Finally two places, representing the external environment, will be given a special status: the **source** and the **sink** of the organization.

There is a direct one to one correspondence between interactional places and the non zero elements of the matrix representation of a WDN. Each 1 of the arrays representing a WDN corresponds to a link between two stages of two different decisionmakers or to a link between a decisionmaker and the external environment. In the Petri Net representation of the WDN, there will be a place connecting the two consecutive stages or connecting the external environment and the appropriate stage. Once interactional places have been defined, internal places are uniquely determined: they fill the gaps between interactional places to ensure that a DM cannot be partitionned into two separate pieces.

Example

The foregoing developments are illustrated on the following example. Figure 3 gives the matrix representation of Π_1 , a 3-dimensional WDN, while Figure 4 presents the Petri Net representation of Π_1 .

$$\begin{array}{lll}
 e = [0 & 1 & 1] & F = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} & G = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
 s = [0 & 1 & 1] & H = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & C = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}
 \end{array}$$

Figure 3 Matrix representation of Π_1

Incidence Matrix

A Petri Net can be represented by an integer matrix reflecting its topological structure. This matrix, called incidence matrix [6], is the basis of all algebraic computations that can be made on a Petri Net to analyze its properties. The incidence matrix of the Petri

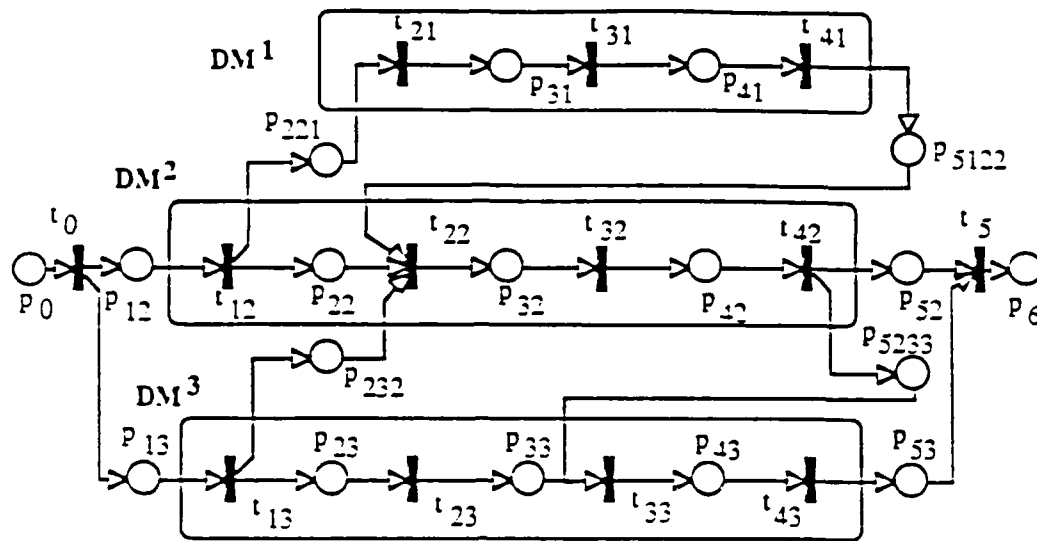


Figure 4 Petri Net representation of Π_1

Net Π_1 is represented in Figure 5. To underline the bloc structure of this matrix, transitions and places have been combined into vectors according to the stages they are related to. Places for instance, have been combined into six groups, corresponding to vertical slices in Figure 4.

	t_0	t_1	t_2	t_3	t_4	t_5
P_0	-1	0 0	0 0 0	0 0 0	0 0 0	0
P_{12}	1	-1 0	0 0 0	0 0 0	0 0 0	0
P_{13}	1	0 -1	0 0 0	0 0 0	0 0 0	0
P_{221}	0	1 0	-1 0 0	0 0 0	0 0 0	0
P_{22}	0	1 0	0 -1 0	0 0 0	0 0 0	0
P_{232}	0	0 1	0 -1 0	0 0 0	0 0 0	0
P_{23}	0	0 1	0 0 -1	0 0 0	0 0 0	0
P_{31}	0	0 0	1 0 0	-1 0 0	0 0 0	0
P_{32}	0	0 0	0 1 0	0 -1 0	0 0 0	0
P_{33}	0	0 0	0 0 1	0 0 -1	0 0 0	0
P_{41}	0	0 0	0 0 0	1 0 0	-1 0 0	0
P_{42}	0	0 0	0 0 0	0 1 0	0 -1 0	0
P_{43}	0	0 0	0 0 0	0 0 1	0 0 -1	0
P_{5122}	0	0 0	0 -1 0	0 0 0	1 0 0	0
P_{52}	0	0 0	0 0 0	0 0 0	0 1 0	-1
P_{5232}	0	0 0	0 0 0	0 0 -1	0 1 0	0
P_{53}	0	0 0	0 0 0	0 0 0	0 0 1	-1
P_6	0	0 0	0 0 0	0 0 0	0 0 0	1

Figure 5 Incidence matrix Δ_1 of Π_1

A WDN can, therefore, be represented in two different ways:

- (1) The matrix representation, i.e. the set of arrays $\{E, S, F, G, H, C\}$.
- (2) The Petri Net representation, given by the graph or the incidence matrix of the net, with the associated labeling of the transitions.

These two representations of a WDN are equivalent, i.e. a one to one correspondence exists between them. Note that the indicated labeling of the transitions is sufficient to interpret correctly the incidence matrix and retrieve the underlying structure of interactions among the decisionmakers.

The following proposition gives a theoretical explanation to the fact that all but the first and the last rows of the incidence matrix of a WDN have exactly one "-1" and one "1".

Proposition 1

Let the source and the sink places of the Petri Net representing a WDN be combined into a unique place. If the resulting Petri Net is strongly connected, it is an event graph.

The proof of Proposition 1 is straightforward. Each internal or interactional place of a WDN has exactly one input and one output transition. The sink of a WDN has one input but no output transitions, while the opposite stands for the source. If source and sink are merged into one place, every place in the net will have, therefore, one input and one output transition. Since the net is strongly connected, it is an event graph.

Note that considering the source and the sink of a WDN as the same place has no bearing on the internal topology of the net. The assumption becomes however important when the dynamic behavior of a WDN is studied. The merging of source and sink limits indeed the amount of information a given organization can process simultaneously. The initial marking of the place representing the external environment will define this bound (see [4] for a detailed discussion of these issues). Those considerations are however not within the scope of this paper.

DECISIONMAKING ORGANIZATIONS

The notion of Well Defined Net (WDN) has been introduced to characterize the class of organizations under consideration. While WDNs constitute the framework within which

organizations will be designed, each WDN is not a valid organizational structure. This section defines additional constraints to restrict the set of WDNs. First, there are some WDNs corresponding to combinations of interactions between decisionmakers that do not have a physical interpretation. Those WDNs should be eliminated, if realistic organizational forms are to be generated. The **structural constraints** define what kinds of combinations of interactions need to be ruled out. Second, any realistic design procedure should allow the designer to introduce specific structural characteristics appropriate to the particular design problem. **User-defined constraints** are introduced to address this issue. As an important side effect of the introduction of constraints, the dimensionality of the problem will be reduced, thus enhancing its computational tractability.

Structural Constraints

Structural constraints refer to the set of conditions that the class of organizations considered here must fulfill. They are contrasted to user-defined constraints which are a set of specific conditions defined by the organization designer for a particular application. Four different structural constraints are formulated that apply to all organizational structures being considered.

- (R₁) A directed path should exist from the source to every node of the structure and from every node to the sink.
- (R₂) The structure should have no loop, i.e., the organizational structures are acyclical.
- (R₃) There can be at most one link from the RS stage of a DM to each one of the other DMs, i.e., for each i and j , only one element of the triplet $\{G_{ij}, H_{ij}, C_{ij}\}$ can be nonzero.
- (R₄) Information fusion can take place only at the IF and CI stages. Consequently, the SA stage of each DM can have only one input.

The set of structural constraints is defined as $R_s = \{R_1, R_2, R_3, R_4\}$.

The constraint R_1 defines connectivity as it pertains to this problem. It eliminates structures that do not represent a single integrated organization and ensures that the flow of

information is continuous within an organization. Note that constraint R_1 ensures that the Petri Net representing an organization whose source and sink have been merged together, is strongly connected. Constraint R_2 allows acyclical organizations only. The acyclical hypothesis has been first formulated by Levis [5]. This restriction is made to avoid deadlock and circulation of messages within the organization and is characteristic of the class of organizations - those that carry out well defined decisionmaking tasks under severe time constraints - that have motivated this work. Constraint R_3 states that a decisionmaker can send its output of the RS stage to another given decisionmaker only once. It does indeed not make much sense to send the same output to the same decisionmaker at several different stages. Constraint R_4 prevents a decisionmaker to receive more than one input at the SA stage. The logic behind this limitation is that information cannot be merged at the SA stage. The IF stage has been specifically introduced to perform such a fusion.

User-defined Constraints

To introduce constraints that will reflect the specific application he is considering, the organization designer can place the appropriate 0's and 1's in the arrays $\{e, s, F, G, H, C\}$ defining a WDN. The other elements will remain unspecified and will constitute the degrees of freedom of the design. The set of user-defined constraints will be denoted R_u , while the complete set of constraints will be denoted R .

Terminology

A WDN that fulfills the set of user-defined constraints R_u will be called an **Admissible Organizational Form (AOF)**. The set of all AOFs will be denoted $\Phi(R_u)$.

An AOF that fulfills the set of constraints R_s will be called a **Feasible Organization (FO)**. Note that a Feasible Organization is a WDN that fulfills the complete set of constraints R . The set of all Feasible Organizations will be denoted $\Phi(R)$.

The following inclusions hold.

$$\Psi^n \supset \Phi(R_U) \supset \Phi(R)$$

CHARACTERIZATION OF THE SET $\Phi(R)$

A Feasible Organization (FO) has been defined as a Well Defined Net (WDN) that satisfies both the structural and the user-defined constraints. The design problem is to determine the set of all Feasible Organizations corresponding to a specific set of constraints. It is assumed throughout this section that the user-defined constraints R_U are given.

Universal and Kernel Nets

The **Universal Net** associated with the constraints R_U - $\Omega(R_U)$ - is the WDN obtained by replacing all undetermined elements of $\{\underline{e}, \underline{s}, F, G, H, C\}$ by 1. Similarly, the **Kernel Net** - $\omega(R_U)$ - is the WDN obtained by replacing the same undetermined elements by 0.

The set $\Phi(R_U)$ of all Admissible Organizational Forms is characterized by the following proposition.

Proposition 2

The set $\Phi(R_U)$ is the subset of Ψ^n that satisfies the two following conditions:

- any element of $\Phi(R_U)$ is a subnet of the Universal Net $\Omega(R_U)$.
- the Kernel Net $\omega(R_U)$ is a subnet of any element of $\Phi(R_U)$.

Alternatively,

$$\Phi(R_U) = \{ \Pi \in \Psi^n \mid \omega(R_U) \leq \Pi \leq \Omega(R_U) \}$$

The notion of subnet introduced earlier defines an order (denoted $\leq$) on the set Ψ^n of all WDNs of dimension n . The concepts of maximal and minimal elements can therefore be

defined. A maximal element of the set $\Phi(R)$ of all Feasible Organizations will be called a **Maximally Connected Organization (MAXO)**. Similarly, a minimal element of $\Phi(R)$ will be called a **Minimally Connected Organization (MINO)**. The set of all MAXOs (resp. MINOs) will be denoted $\Phi_{\max}(R)$ (resp. $\Phi_{\min}(R)$).

Maximally and minimally connected organizations can be interpreted as follows. A MAXO is a WDN such that it is not possible to add a single link without violating the set of constraints R (i.e. without crossing the boundaries of the subset $\Phi(R)$). Similarly, a MINO is a WDN such that it is not possible to remove a single link without violating the set of constraints R . The following proposition is a direct consequence of the definition of maximal and minimal elements.

Proposition 3

For any given Feasible Organization Π , there is at least one MINO $\Pi_{\min}$ and at least one MAXO $\Pi_{\max}$ such that $\Pi_{\min} \leq \Pi \leq \Pi_{\max}$. Alternatively,

$$\{\Pi \in \Psi^n \mid \exists (\Pi_{\min}, \Pi_{\max}) \in \Phi_{\min}(R) \times \Phi_{\max}(R) \Pi_{\min} \leq \Pi \leq \Pi_{\max}\} \supset \Phi(R)$$

Note that the previous inclusion is not an equality in the general case. There is indeed no guarantee that a WDN located between a MAXO and a MINO will fulfill the constraints R . To address this problem, the concept of a simple path will be used.

Simple Path

Let Π be a WDN that satisfies constraint R_1 and whose source and sink have been merged together into a single external place. A **simple path** of Π is a directed elementary circuit which includes the external place.

According to Proposition 1, the Petri Net representing Π is an event-graph. A simple path is therefore a minimal support S -invariant of Π whose component corresponding to the external place is equal to 1 [4]. Note that if the latter property is not satisfied, the S -invariant

is an internal loop of the net. In [7], the concept of S-component associated with an S-invariant is defined: it is the subnet of the initial Petri Net whose places are exactly the places of the support of the S-invariant. An S-component of a WDN is therefore itself a WDN. Consequently, the simple paths of a given WDN are themselves WDNs. We will

denote by $Sp(R_U)$ the set of all simple paths of the Universal Net $\Omega(R_U)$. We will write

$$Sp(R_U) = \{sp_1, \dots, sp_r\},$$

where the sp_i ($1 \leq i \leq r$) are WDNs satisfying $sp_i \leq \Omega(R_U)$.

Union of Simple Paths: the Set $USp(R_U)$

If the cardinal of $Sp(R_U)$ is r , we can write $Sp(R_U) = \{sp_i, 1 \leq i \leq r\}$. Since simple paths are WDNs, the set $Sp(R_U)$ is included in the set of all WDNs, Ψ^n . We will denote by $USp(R_U)$ the set of all possible unions of elements of $Sp(R_U)$, augmented with the null element $\emptyset$ of Ψ^n .

$$USp(R_U) = \{\Pi \in \Psi^n \mid \exists (sp_{i1}, \dots, sp_{iq}) \in Sp(R_U)^q \quad \Pi = sp_{i1} \cup \dots \cup sp_{iq}\} \cup \{\emptyset\}$$

$USp(R_U)$ is the set of all combinations of simple paths of the Universal Net $\Omega(R_U)$.

The union of two elements of $USp(R_U)$ will be the WDN composed of all the simple paths included in **either one** of the two considered elements. Proposition 4 justifies the introduction of the set $USp(R_U)$.

Proposition 4

Every WDN, element of the set $USp(R_U)$, satisfies the connectivity constraint R_1 . Reciprocally, a Feasible Organizational Form that fulfills the constraint R_1 is an element of $USp(R_U)$. In formal language:

$$\{\Pi \in \Psi^n \mid R_1[\Pi] = 1\} \supset USp(R_U) \supset \{\Pi \in \Phi(R_U) \mid R_1[\Pi] = 1\}$$

$R_1[\Pi] = 1$ means that Π satisfies the constraint R_1 .

Characterization of $\Phi(R)$

We are now ready to state the following proposition characterizing the set $\Phi(R)$ of all feasible organizations.

Proposition 5

Let Π be a WDN of dimension n . Π will be a Feasible Organization if and only if

- Π is a union of simple paths of the Universal Net $\Omega(R_u)$, i.e., $\Pi \in \text{USp}(R_u)$.
- Π is bounded by at least one MINO and one MAXO.

In formal language:

$$\Phi(R) = \{ \Pi \in \text{USp}(R_u) \mid \exists (\Pi_{\min}, \Pi_{\max}) \in \Phi_{\min}(R) \times \Phi_{\max}(R) \Pi_{\min} \leq \Pi \leq \Pi_{\max} \}$$

Proposition 5 gives a characterization of the set $\Phi(R)$ just like Proposition 4 gives a characterization of the set $\Phi(R_u)$. While Ψ^n is used in the equality characterizing $\Phi(R_u)$, $\text{USp}(R_u)$ is used to characterize $\Phi(R)$. In the former case, the link is the incremental unit leading from a WDN to its immediate superordinate, while in the latter the simple path plays the role of the building unit. In generating organizational structures with simple paths, the connectivity constraint R_1 is automatically satisfied.

APPLICATION

Let us consider the set of user-defined constraints represented in Figure 6 and corresponding to 5-dimensional WDNs. The "x" in the arrays of Figure 6 correspond to the unspecified elements; their number is equal to 12. The "#" indicate inadmissible links, while the 0's and 1's indicate the forced absence or presence, respectively, of links.

The organization under consideration has five members. Decisionmakers DM^1 and DM^2 act as the sensors of the organization. They both receive information from the external environment. They may or may not share this information with each other and with the other members of the organization. Decisionmaker DM^1 , however, has to send this information to decisionmaker DM^3 , the coordinator. At the other end of the organization, decisionmakers DM^4 and DM^5 act as actuators. They are both directly related to the external environment. They will receive orders from the coordinator DM^3 and they may receive information from DM^1 and DM^2 . They may also share their results, i.e., the concrete action they are taking, with each other and with the coordinator.

$$\begin{array}{lcl}
e = [1 \ 1 \ 0 \ 0 \ 0] & F = & \begin{bmatrix} \# & x & 0 & 0 & 0 \\ x & \# & 0 & 0 & 0 \\ 0 & 0 & \# & 0 & 0 \\ 0 & 0 & 0 & \# & 0 \\ 0 & 0 & 0 & 0 & \# \end{bmatrix} & G = & \begin{bmatrix} \# & 0 & 0 & 0 & 0 \\ 0 & \# & 0 & 0 & 0 \\ 0 & 0 & \# & 0 & 0 \\ 0 & 0 & 0 & \# & 0 \\ 0 & 0 & 0 & 0 & \# \end{bmatrix} \\
\\
s = [0 \ 0 \ 0 \ 1 \ 1] & H = & \begin{bmatrix} \# & 0 & 1 & x & x \\ 0 & \# & x & x & x \\ 0 & 0 & \# & 0 & 0 \\ 0 & 0 & x & \# & x \\ 0 & 0 & x & x & \# \end{bmatrix} & C = & \begin{bmatrix} \# & 0 & 0 & 0 & 0 \\ 0 & \# & 0 & 0 & 0 \\ 0 & 0 & \# & 1 & x \\ 0 & 0 & 0 & \# & 0 \\ 0 & 0 & 0 & 0 & \# \end{bmatrix}
\end{array}$$

Figure 6 User-defined constraints R_u

The universal net $\Omega(R_u)$ is obtained by setting to 1 all the unspecified elements of the arrays, i.e., setting all x's equal to 1. The net $\Omega(R_u)$ is represented in Figure 7. Places are labeled sequentially but transitions are labeled according to the stage they represent and the decisionmaker they belong to. Figure 7 corresponds to representation (2) of a WDN: a WDN is characterized by the graph of its Petri Net representation with the associated labeling of the transitions. The boldface links and places of Figure 7 correspond to the Kernel net $\omega(R_u)$, which is obtained by setting all x's equal to 0.

A computer-aided design procedure has been implemented on an IBM PC/AT with 512K RAM and a 20 MB hard disk drive. A graphical interface has been implemented that allows the user to specify the six arrays for organizations with up to 5 members.

Forty simple paths for the net $\Omega(R_u)$ were generated by the computer program. Two versions of the algorithm have been implemented. One of them uses the algorithm proposed by Martinez and Silva [8] to generate S-invariants, while the other version used an algorithm developed by Jin [1]. Identical results were produced. For this application, 10 MINOs and 3 MAXOs were obtained. The Petri Net representation of two of the 10 MINOs is reproduced in Figure 8, while the three MAXOs are represented in Figure 9.

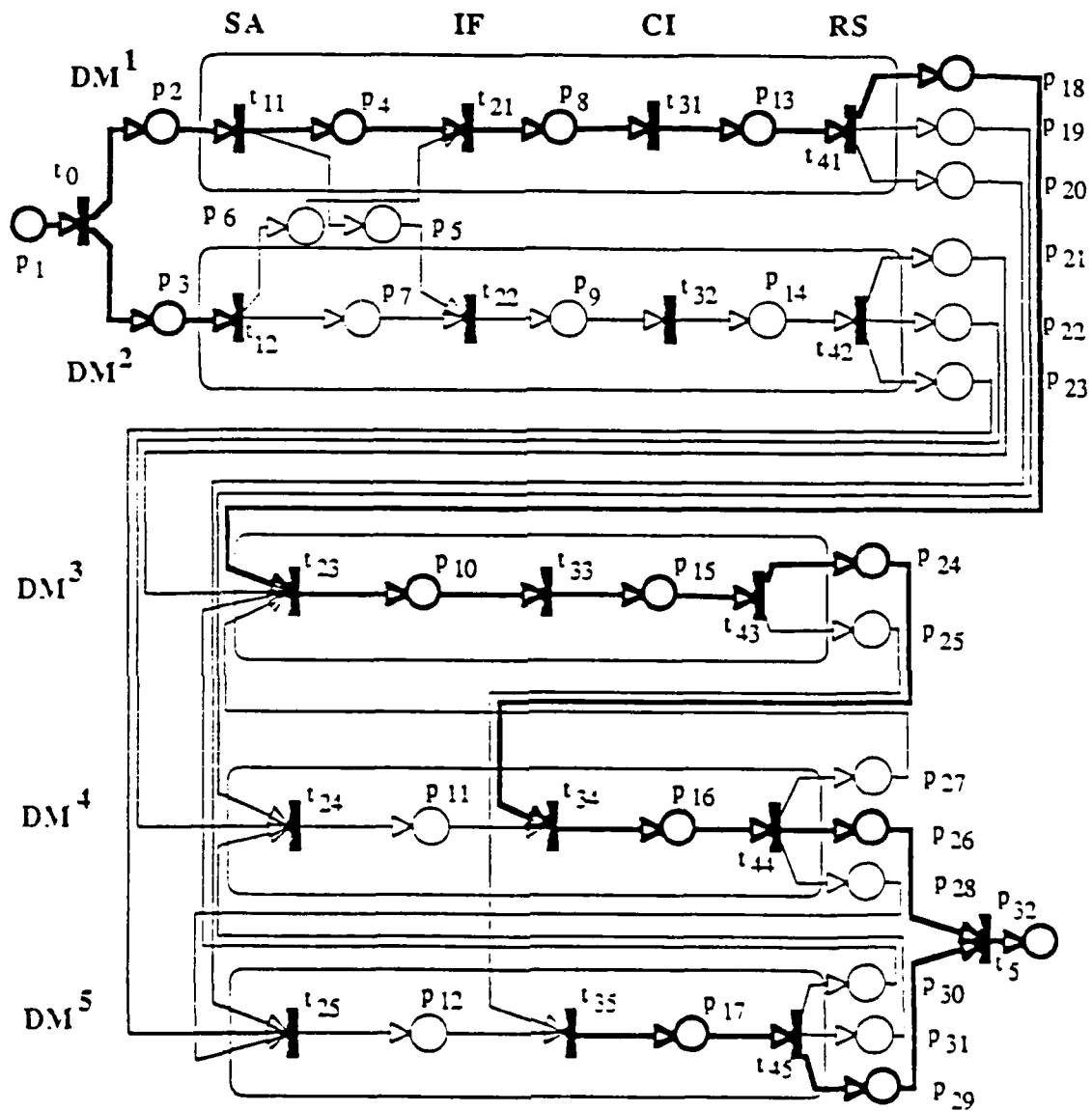


Figure 7 Universal Net $\Omega(R_u)$

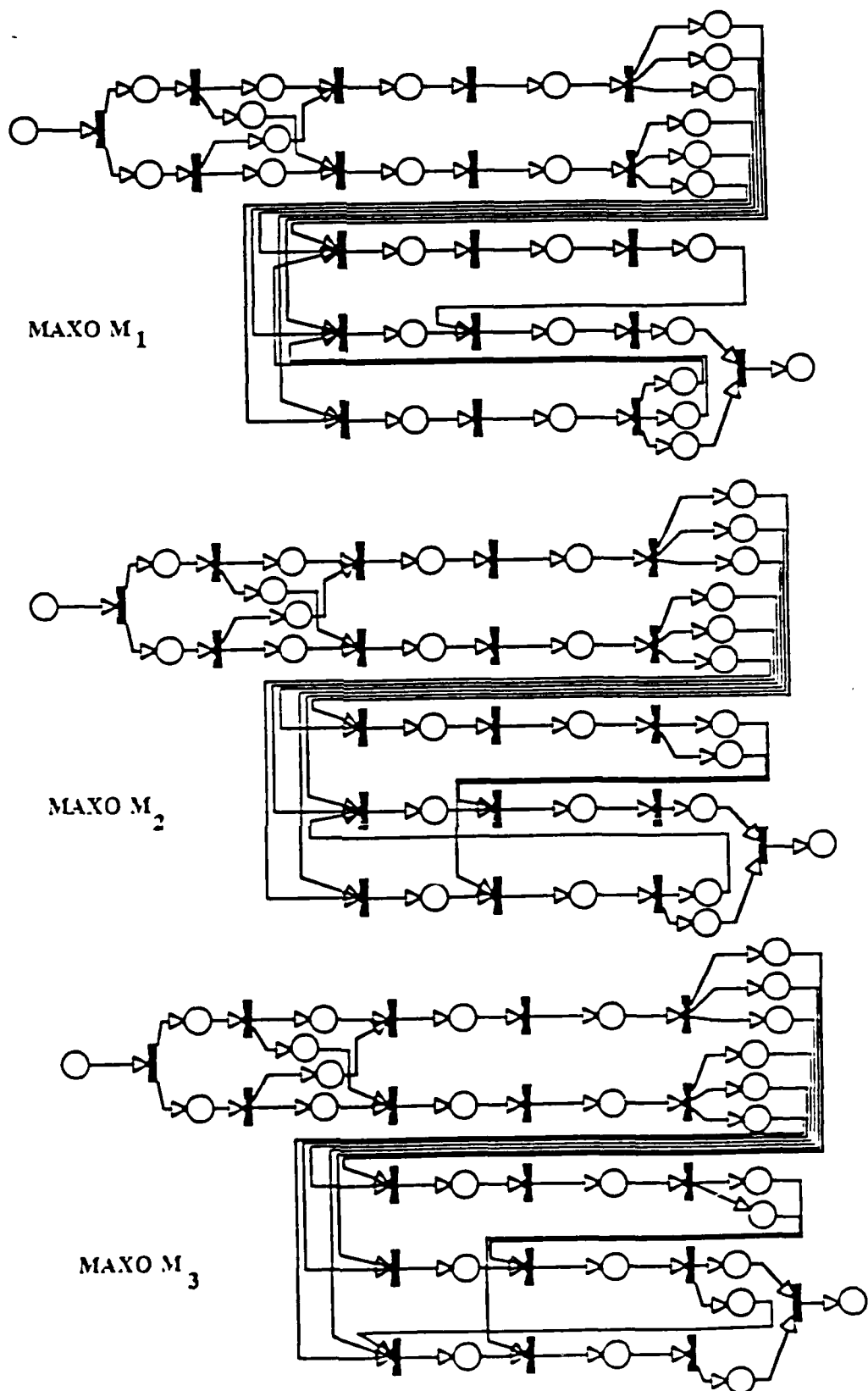


Figure 9 Petri Net representation of the MAXOs

Structure of the Set $\Phi(R)$

The inclusion relation ($\leq$) between MINOs and MAXOs is represented in Figure 10. This graph is the skeleton of the diagram of $\Phi(R)$. Note that the set $\Phi(R)$ seems to be divided into three groups of organizations. One group is related to the MAXO M_3 only and originates from the MINOs m_3, m_6 and m_9 . Another group is related to the MAXOs M_3 and M_2 and originates from the MINOs m_2, m_5 , and m_8 . Lastly, a third group is related to the three MAXOs and originates from the MINOs m_1, m_4, m_7 , and m_{10} . Therefore, MAXO M_3 can be reached from every MINO, while MAXOs M_1 and M_2 can only be

reached from specific MINOs. This division of the set $\Phi(R)$ into categories would require further theoretical development before any meaningful result could be derived.

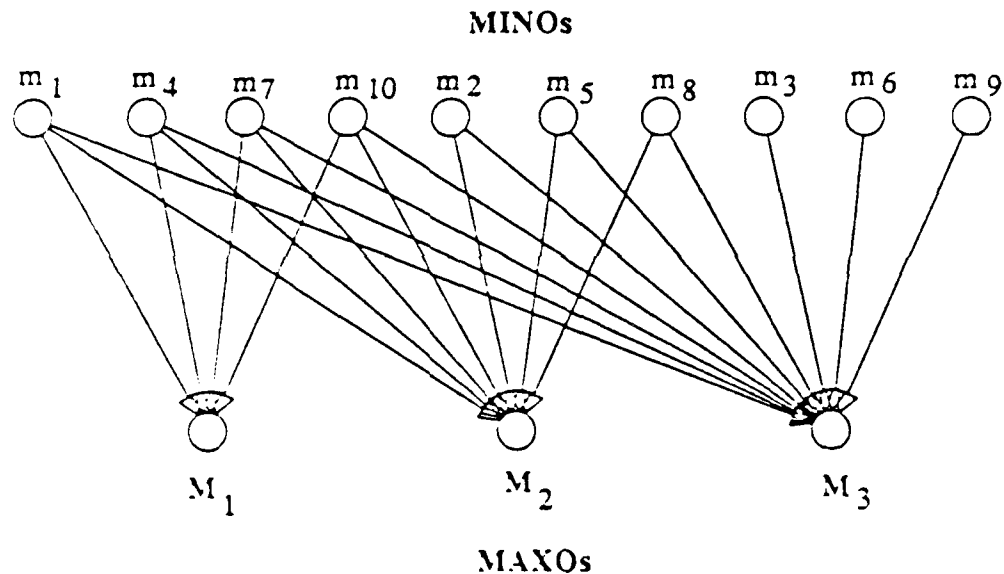


Figure 10 Skeleton of the diagram of the set $\Phi(R)$

The complete diagram of $\Phi(R)$ would be obtained in making explicit the links connecting every pair (MINO,MAXO) of Figure 10. In other words, one would need to determine all the organization chains existing between every pair (MINO,MAXO).

This example shows how a problem can be formulated within the framework developed in this paper. Once the user-defined constraints are specified, the new algorithm, named ARCGEN, generates the MINOs and the MAXOs which characterize the set of all organizational structures that satisfy the designer's requirements. This reduces the workload

of the organization's designer: instead of looking at the entire set, the designer will concentrate on its boundaries, the MINOs and the MAXOs. Note that, although the original problem has very high dimensionality, 2^{90} , there are only 10 MINOs and 3 MAXOs. The organization designer can, therefore, concentrate his analysis on those 13 organizational structures. His task is thus much simplified, if compared to the original one. The first step of the analysis consists of putting the MINOs and the MAXOs in their actual context, to give them a physical interpretation. If the organization designer is interested in a given pair of MINO and MAXO, because they contain interactions that are deemed desirable for the specific application, he can further investigate the chains connecting those two organizations within the set $\Phi(R)$.

In summary, the methodology presented provides the organization designer with a rational way to handle a problem whose combinatorial complexity is very large.

CONCLUSIONS

In this paper, a methodology is presented for generating organizational architectures that satisfy some generic structural properties, as well as more specific designer's requirements. An analytical framework is developed to formulate first and then analyze the problem.

Given the designer's requirements and the set of pre-established structural constraints, the design problem consists of finding the set $\Phi(R)$ of all Feasible Organizations, i.e., the set of organizational structures that satisfy both the designer's requirements and the structural constraints. The set $\Phi(R)$ is characterized by its boundaries, i.e., by its minimal and maximal elements. Those elements, the MINOs and the MAXOs, will correspond to the organizational structures with the minimum and the maximum number of interactional links among organization members. The complete set of Feasible Organizations is then generated by considering all the organizational structures that lie between the MINOs and the MAXOs. The notion of simple path is used as the incremental unit leading from an organization to another. By adding simple paths to every MINO until a MAXO is found, one will scan the complete set of Feasible Organizations.

In this paper, a framework is described briefly (for details, see [9]) within which the organization design problem can be articulated. Assumptions about the model are clearly and precisely defined, thus making the scope and the limitations of the methodology easily

identifiable. In relaxing some of those assumptions, the generality of the model can be extended [9]. The organization designer is provided with a rational way to translate his requirements into specifications. This is a major feature of the methodology, since in most design procedures, the step leading from a physical to an analytic description of a concrete application, is the hardest to make. Lastly, the methodology introduces a technique to reduce considerably the number of organizational classes that the designer will eventually have to investigate. Starting from a high level of complexity, the design problem is therefore brought down to a tractable level. Petri Nets are used throughout the design procedure as the underlying theoretical paradigm. Indeed, the use of S-invariants as the incremental unit leading from an organization to its immediate superordinate is one of the most important feature of the methodology.

REFERENCES

- [1] V.Y-Y. Jin, "Delays for distributed decisionmaking organizations", MS Thesis, Report LIDS-TH-1459, Laboratory for Information and Decision Systems, MIT, 1985.
- [2] G. Bejjani and A.H.Levis, "Information storage and access in decisionmaking organizations", Report LIDS-P-1466, Laboratory for Information and Decision Systems, MIT, 1985.
- [3] D.Tabak and A.H.Levis, "Petri Net Representation of Decision Models", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.SMC-15, No.6, 812-818, November/December 1985.
- [4] H.Hillion, "Performance Evaluation of Decisionmaking Organizations using Timed Petri Nets", MS Thesis, Report LIDS-TH-1590, Laboratory for Information and Decision Systems, MIT, 1986.
- [5] A.H. Levis, "Information processing and decisionmaking organizations: a mathematical description", *Large Scale Systems*, 7., 155-163, 1984.
- [6] G.Memmi and G.Roucairol, "Linear Algebra in Net Theory", in *Net Theory and*

Applications, Proceedings of the Advanced Course on General Net Theory of Processes and Systems, Hamburg 1979, W. Brauer, Ed., Springer-Verlag, Berlin, 1980.

[7] J.Sifakis, "Structural Properties of Petri Nets", in *Mathematical Foundations of Computer Science*, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1978.

[8] J.Martinez and M.Silva, "A Simple and Fast Algorithm to Obtain all Invariants of a Generalized Petri Net", in *Application and Theory of Petri Nets*, C.Giraud, W.Reisig, Eds., Springer-Verlag, Berlin, 1980.

[9] P.A. Remy, "On the Generation of Organizational Architectures using Petri Nets", MS Thesis, Report LIDS-TH-1630, Laboratory for Information and Decision Systems, MIT, December 1986.

END

2-87

OTIC