amcs

A HIERARCHICAL DECOMPOSITION OF DECISION PROCESS PETRI NETS FOR MODELING COMPLEX SYSTEMS

JULIO CLEMPNER

Center for Computing Research
National Polytechnic Institute (CIC-IPN), Av. Juan de Dios Batiz s/n, Edificio CIC, Col. Nueva Industrial Vallejo
07738 Mexico City, Mexico
e-mail: `julio@clempner.name`

We provide a framework for hierarchical specification called Hierarchical Decision Process Petri Nets (HDPPNs). It is an extension of Decision Process Petri Nets (DPPNs) including a hierarchical decomposition process that generates less complex nets with equivalent behavior. As a result, the complexity of the analysis for a sophisticated system is drastically reduced. In the HDPPN, we represent the mark-dynamic and trajectory-dynamic properties of a DPPN. Within the framework of the mark-dynamic properties, we show that the HDPPN theoretic notions of (local and global) equilibrium and stability are those of the DPPN. As a result in the trajectory-dynamic properties framework, we obtain equivalent characterizations of that of the DPPN for final decision points and stability. We show that the HDPPN mark-dynamic and trajectory-dynamic properties of equilibrium, stability and final decision points coincide under some restrictions. We propose an algorithm for optimum hierarchical trajectory planning. The hierarchical decomposition process is presented under a formal treatment and is illustrated with application examples.

Keywords: hierarchy, decomposition, structuring mechanisms, re-usable components, decision process, DPPN, stability, Lyapunov methods, optimization.

1. Introduction

The most critical point in the development of complex systems depends largely on the ability to choose a conceptual model to represent the problem domain in a coherent and natural fashion. Formal models that capture and organize knowledge hierarchically can facilitate solutions to this problem.

Decision Process Petri Nets (DPPNs) are used for complex systems representations, taking advantage of the well-known properties of Petri nets, namely, formal semantic, graphical display and decision process (Clempner 2005b; 2005c). However, DPPNs lack, as any other Petri nets, at least two important characteristics, on the one hand a way of adding a structure and, on the other, a way of decomposing larger nets into smaller ones. One of the basic approaches to accomplishing this task is the hierarchical decompositions.

In order to provide hierarchical features to the DPPN, in this work we introduced Hierarchical Decision Process Petri Nets (HDPPNs). The idea has been advocated and tested for the modeling of complex processes by a number of researchers in Petri nets (Bellman, 2008; Buchholz, 1994; Dai *et al.*, 2009; Gomes and Barros, 2005; Huber *et al.*, 1990; Jensen, 1992). The proposed hierarchical decomposition forms allow the replacement of transitions by more complex nets which describe a refinement of the view.

From a practical point of view, to model a complex system it is convenient to concentrate on some activities which are regarded as being essential for system functionality, and to abstract the activity behavior in the early design stages. When the design evolves, every single transition in a net may be refined by a new net in order to specify the respective activity in greater detail.

Under this method a complex system could be divided into various levels of detail in a top-down approach. The hierarchical decomposition can be used as a structuring mechanism to organize the development of a system in an efficient and coherent manner. It allows the decomposition of the system into different levels of detail giving increased modeling adaptability. The intention of this approach is to define an equivalence relation able to construct a hierarchical partition. At the top level it provides a higher level of abstraction and a complete view of the sys-

tem without a great specification, and at the lowest level it provides a high degree of design detail.

However, the hierarchy in HDPPN formalism is used not only for net efficiency or model specification. One of the most important problems that DPPN theory confronts is the analysis and utility function calculation of sophisticated systems, which is usually huge for real applications. The complexity of the analysis of the DPPN can be reduced significantly if it is hierarchically decomposed. The hierarchical decomposition process generates simple nets with equivalent behavior. As a result, the net is divided into small sets and the complexity of the analysis of the DPPN is reduced considerably.

The main point of the HDPPN is its ability to represent the mark-dynamic and the trajectory-dynamic properties of a hierarchical decision process application. We will identify the mark-dynamic properties of the HDPPN as related to only place-transitions Petri nets, and we will relate the trajectory-dynamic properties of the HDPPN as related with the utility function at each place that depends on a probabilistic routing policy of the DPPN. Within the mark-dynamic properties framework, we show that the HDPPN theoretic notions of local and global stability are those of the DPPN. In the trajectory-dynamic properties framework, we define the utility function as a Lyapunov-like function that is able to track hierarchically the net and to converge to an equilibrium point (Clempner *et al.*, 2005a). By selection of appropriate Lyapunov-like functions under certain desired criteria, it is possible to optimize the utility. In addition, we used the notions of local and global stability in the sense of Lyapunov to characterize the stability properties of the HDPPN. The HDPPN uses a non-negative utility function (as the DPPN does) that converges in decreasing form to a (set of) final decision states. We show that if the HDPPN is finite and non-blocking, then we have that a final decision state is an equilibrium point iff it is an optimum point. We present an algorithm for optimum hierarchical trajectory planning used to find the optimum point. The algorithm consists in finding a firing transition sequence such that an optimum decision state is hierarchically reached in the HDPPN. For this propose the algorithm uses the graphical representation provided by the place-transitions Petri net and the utility function.

The paper is structured in the following manner. The next section presents the necessary mathematical background and terminology needed to understand the rest of the work. Section 3 discusses the main results of this paper, providing a definition of the HDPPN and giving a detailed analysis of the equilibrium, stability and optimum point conditions for the mark-dynamic and the trajectory-dynamic parts of the HDPPN. An algorithm for calculating the optimum trajectory used to find the optimum point is proposed. For illustration purposes, we show how the standard notions defined in HDPPN theory are ap-

plied to a practical example. Finally, some concluding remarks and future work are provided in Section 4. For completeness, appendices related to the mark-dynamic, trajectory-dynamic and the convergence of the HDPPN mark-dynamic and trajectory-dynamic properties are included.

2. Preliminaries

In this section, we present some well-established definitions and properties (Lakshmikantham *et al.*, 1990; 1991) which will be used later.

Notation. $\mathbb{N} = \{0, 1, 2, \dots\}$, $\mathbb{R}_+ = [0, \infty)$, $\mathbb{N}_{n_0+} = \{n_0, n_0 + 1, \dots, n_0 + k, \dots\}$, $n_0 \geq 0$. Given $x, y \in \mathbb{R}^d$, we usually denote the relation "$\leq$" to mean componentwise inequalities with the same relation, i.e., $x \leq y$ is equivalent to $x_i \leq y_i, \forall i$. A function $f(n, x)$, $f : \mathbb{N}_{n_0+} \times \mathbb{R}^d \to \mathbb{R}^d$ is called nondecreasing in x if, given $x, y \in \mathbb{R}^d$ such that $x \geq y$ and $n \in \mathbb{N}_{n_0+}$, $f(n, x) \geq f(n, y)$.

Consider systems of first-order difference equations given by

$$x(n+1) = f[n, x(n)], \quad x(n_o) = x_0, \qquad (1)$$

where $x(n) \in \mathbb{R}^d$ and $f : \mathbb{N}_{n_0+} \times \mathbb{R}^d \to \mathbb{R}^d$ is continuous in $x(n)$.

Definition 1. The n-vector valued function $\Phi(n, n_0, x_0)$ is said to be a *solution* of (1) if $\Phi(n_0, n_0, x_0) = x_0$ and $\Phi(n+1, n_0, x_0) = f(n, \Phi(n, n_0, x_0))$ for all $n \in \mathbb{N}_{n_0+}$.

Definition 2. The system (1) is said to be

(i) *practically stable* if, given (λ, A) with $0 < \lambda < A$, we have

$$|x_0| < \lambda \Rightarrow |x(n, n_0, x_0)| < A, \forall n \in \mathbb{N}_{n_0+}, n_0 \geq 0,$$

(ii) *uniformly practically stable* if it is practically stable for every $n_0 \geq 0$.

The following class of function is defined.

Definition 3. A continuous function $\alpha : [0, \infty) \to [0, \infty)$ is said to *belong to class $\mathcal{K}$* if $\alpha(0) = 0$ and it is strictly increasing.

2.1. Methods for practical stability.
Consider (cf. Lakshmikantham *et al.* 1990; 1991) the vector function $v(n, x(n))$, $v : \mathbb{N}_{n_0+} \times \mathbb{R}^d \to \mathbb{R}^p_+$ and define the variation of v relative to (1) by

$$\Delta v = v(n+1, x(n+1)) - v(n, x(n)). \qquad (2)$$

Then, the following result concerns the practical stability of (1).

Theorem 1. *Let* $v : \mathbb{N}_{n_0+} \times \mathbb{R}^d \to \mathbb{R}^p_+$ *be a continuous function in* x. *Define the function* $v_0(n, x(n)) = \sum_{i=1}^p v_i(n, x(n))$ *such that it satisfies the estimates*

$$b(|x|) \le v_0(n, x(n)) \le a(|x|) \quad \text{for } a, b \in \mathcal{K}$$

and

$$\Delta v(n, x(n)) \le w(n, v(n, x(n)))$$

for $n \in \mathbb{N}_{n_0+}$, $x(n) \in \mathbb{R}^d$, *where* $w : \mathbb{N}_{n_0+} \times \mathbb{R}^p_+ \to \mathbb{R}^p$ *is a continuous function in the second argument.*

Assume that $g(n, e) \triangleq e + w(n, e)$ is nondecreasing in e, $0 < \lambda < J$ are given and, finally, that $a(\lambda) < b(A)$ is satisfied. Then, the practical stability properties of

$$e(n + 1) = g(n, e(n)), e(n_0) = e_0 \ge 0 \qquad (3)$$

imply the corresponding practical stability properties of the system (1).

Corollary 1. *In Theorem 1*

1. *If* $w(n, e) \equiv 0$, *we obtain uniform practical stability of (1) which implies structural stability (Murata, 1989).*

2. *If* $w(n, e) = -c(e)$, *for* $c \in \mathcal{K}$, *we obtain uniform practical asymptotic stability of (1).*

2.2. Petri nets. Petri nets are a tool for systems analysis. Petri net theory allows a system to be modeled by a Petri net, a mathematical representation of the system. The analysis of the Petri net can then, hopefully, reveal important information about the structure and dynamic behavior of the modeled system. This information can then be used to evaluate the modeled system and suggest improvements or changes.

A Petri net is a quintuple, $PN = \{P, Q, F, W, M_0\}$, where $P = \{p_1, p_2, \ldots, p_m\}$ is a finite set of places, $Q = \{q_1, q_2, \ldots, q_n\}$ is a finite set of transitions, $F \subseteq (P \times Q) \cup (Q \times P)$ is a set of arcs, $W : F \to \mathbb{N}_{1+}$ is a weight function, $M_0 : P \to \mathbb{N}$ is the initial marking, $P \cap Q = \varnothing$ and $P \cup Q \ne \varnothing$.

A Petri net structure without any specific initial marking is denoted by N. A Petri net with the given initial marking is denoted by (N, M_0). Notice that if $W(p, q) = \alpha$ (or $W(q, p) = \beta$), this is often represented graphically by α, (β) arcs from p to q (q to p), each with no numeric label.

Let $M_k(p_i)$ denote the marking (i.e., the number of tokens) at place $p_i \in P$ at time k, and let $M_k = [M_k(p_1), \ldots, M_k(p_m)]^T$ denote the marking (state) of PN at time k. A transition $q_j \in Q$ is said to be enabled at time k if $M_k(p_i) \ge W(p_i, q_j)$ for all $p_i \in P$ such that $(p_i, q_j) \in F$. It is assumed that at each time k there exists at least one transition to fire, i.e., it is not possible to block the net. If a transition is enabled, then it can fire. If

an enabled transition $q_j \in Q$ fires at time k, then the next marking for $p_i \in P$ is given by

$$M_{k+1}(p_i) = M_k(p_i) + W(q_j, p_i) - W(p_i, q_j).$$

Let $A = [a_{ij}]$ denote an $n \times m$ matrix of integers (the incidence matrix), where $a_{ij} = a_{ij}^+ - a_{ij}^-$ with $a_{ij}^+ = W(q_i, p_j)$ and $a_{ij}^- = W(p_j, q_i)$. Let $u_k \in \{0, 1\}^n$ denote a firing vector, where if $q_j \in Q$ is fired, then its corresponding firing vector is $u_k = [0, \ldots, 0, 1, 0, \ldots, 0]^T$ with a "1" in the j-th position in the vector and zeros everywhere else. The matrix equation (nonlinear difference equation) describing the dynamical behavior represented by a Petri net is

$$M_{k+1} = M_k + A^T u_k, \qquad (4)$$

where if at step k, $a_{ij}^- < M_k(p_j)$ for all $p_j \in P$, then $q_i \in Q$ is enabled, and if this $q_i \in Q$ fires, then its corresponding firing vector u_k is utilized in the difference equation (4) to generate the next step. Notice that if M' can be reached from some other marking M and if we fire some sequence of d transitions with the corresponding firing vectors $u_0, u_1, \ldots, u_{d-1}$, we obtain that

$$M' = M + A^T u, \ u = \sum_{k=0}^{d-1} u_k. \qquad (5)$$

Definition 4. The set of all the markings (states) reachable from some starting marking M is called the *reachability set* and is denoted by $R(M)$.

Let $(\mathbb{N}_{n_0+}, d)$ be a metric space where $d : \mathbb{N}_{n_0+} \times \mathbb{N}_{n_0+} \to \mathbb{R}_+$ is defined by

$$d(M_1, M_2) = \sum_{i=1}^m \zeta_i |M_1(p_i) - M_2(p_i)|,$$

$$\zeta_i > 0, \quad i = 1, \ldots, m,$$

and consider the matrix difference equation which describes the dynamical behavior of the discrete event system modeled by the Petri net (5). Then the following proposition holds (Passino *et al.*, 1994).

Proposition 1. *Let* PN *be a Petri net. It is uniformly practically stable if there exists a strictly positive* m *vector* Φ *such that*

$$\Delta v = u^T A \Phi \le 0 \Leftrightarrow A \Phi \le 0. \qquad (6)$$

Moreover, a PN *exhibits uniform practical asymptotic stability if the following equation holds:*

$$\Delta v = u^T A \Phi \le -c(e).$$

2.3. Decision processes Petri nets. We introduce the concept of Decision Process Petri Nets (DPPNs) (Clempner, 2005b) by locally randomizing the possible choices, for each individual place of the Petri net.

Definition 5. A *decision process Petri net* is a septuple DPPN = $\{P, Q, F, W, M_0, \pi, U\}$, where

- $P = \{p_0, p_1, p_2, \ldots, p_m\}$ is a finite set of places,

- $Q = \{q_1, q_2, \ldots, q_n\}$ is a finite set of transitions,

- $F \subseteq I \cup O$ is a set of arcs where $I \subseteq (P \times Q)$ and $O \subseteq (Q \times P)$ such that $P \cap Q = \varnothing$ and $P \cup Q \neq \varnothing$,

- $W : F \rightarrow \mathbb{N}_1^+$ is a weight function,

- $M_0 : P \rightarrow \mathbb{N}$ is the initial marking,

- $\pi : I \rightarrow \mathbb{R}_+$ is a routing policy representing the probability of choosing a particular transition (routing arc), such that for each $p \in P$,
$$\sum_{q_j : (p, q_j) \in I} \pi((p, q_j)) = 1,$$

- $U : P \rightarrow \mathbb{R}_+$ is a utility function.

$U_k(.)$ denotes the utility at place $p_i \in P$ at time k, and let $U_k = [U_k(.), \ldots, U_k(.)]^T$ denote the utility state of the DPPN at time k. $FN : F \rightarrow \mathbb{R}_+$ is the number of arcs from place p to transition q (the number of arcs from transition q to place p). The rest of DPPN functionality is as described in the PN preliminaries.

Consider an arbitrary $p_i \in P$, and for each fixed transition $q_j \in Q$ that forms an output arc $(q_j, p_i) \in O$, we look at all the previous places p_h of the place p_i denoted by the list (set) $p_{\eta_{ij}} = \{p_h : h \in \eta_{ij}\}$, where $\eta_{ij} = \{h : (p_h, q_j) \in I \,\&\, (q_j, p_i) \in O\}$, that materialize all the input arcs $(p_h, q_j) \in I$ and form the sum

$$\sum_{h \in \eta_{ij}} \Psi(p_h, q_j, p_i) \cdot U_k(p_h), \qquad (7)$$

where

$$\Psi(p_h, q_j, p_i) = \pi(p_h, q_j) \cdot \frac{FN(q_j, p_i)}{FN(p_h, q_j)}$$

and the index sequence j is the set $\{j : q_j \in (p_h, q_j) \cap (q_j, p_i)$ and p_h running over the set $p_{\eta_{ij}}\}$.

Proceeding with all the q_js, we form the vector indexed by the sequence j identified by $(j_0, j_1, \ldots, j_f)$ as follows:

$$\left[\sum_{h \in \eta_{ij_0}} \Psi(p_h, q_{j_0}, p_i) \cdot U_k(p_h), \sum_{h \in \eta_{ij_1}} \Psi(p_h, q_{j_1}, p_i) \right.$$
$$\left. \cdot U_k(p_h), \ldots, \sum_{h \in \eta_{ij_f}} \Psi(p_h, q_{j_f}, p_i) \cdot U_k(p_h) \right].$$
$$(8)$$

Intuitively, the vector (8) represents all the possible trajectories through the transitions q_js, where $(j_1, j_2, \ldots, j_f)$, to a place p_i for a fixed i.

The aim of this example is to present a business process application as a motivation example represented by the DPPN showing the optimum strategy and the stability properties of the net.

Example 1. Let us consider an insurance agency. The agency sells policies for different companies. The main products are life and automobile policies. Let us consider the process for a car accident. The insurance company depends on the adjustor appraisal to evaluate the damages. To maintain company profitability, the adjustor must evaluate the case so that only the minimal necessary repairs are considered. In this sense, the adjustor evaluation is expected to be in favor of the insurance company because of his/her dependence on the latter. However, the adjustor must be careful, because the insurance company wants to offer good service in order to keep the client. As a result, the automobile owner depends on the appraisal of the adjustor for an appropriate accident evaluation. The automobile owner can also be assisted by an authorized garage to obtain a fair evaluation of the car's damage. Notice that the garage must satisfy both the client and the insurance company, given that the garage income depends on the car owner and on the insurance company. If the accident includes physical damage, the client and passengers must be directed to an accredited hospital for medical treatment.

Three different strategies can be presented to manage a car accident in order to optimize the company's profitability (Hammer and Champy , 1993). To improve the operation cost, small accidents can be directly evaluated by the adjustor or the authorized garage, and reported to the insurance company. Accidents of considerable size must be managed centrally by the insurance company. The partially ordered DPPN (Fig. 1) has the following specifications:

Places

P_0: claim settled

P_1: handled accident info centrally

P_2: handled accident info by authorized garage

P_3: handled accident info by adjustor

P_4: verified policy covering centrally

P_5: verified policy covering by authorized garage

P_6: verified policy covering by adjustor

P_7: corroborated accident details

P_8: evaluated damage centrally

P_9: got medical treatment cost

P_{10}: determined accident in range

P_{11}: send info to be handle centrally

P_{12}: got accident info by adjustor

P_{13}: assessed client antecedents

P_{14}: determined accident covering centrally

P_{15}: got accident info by authorized garage

P_{16}: evaluated damage by authorized garage

P_{17}: determined accident in range

P_{18}: send info to be handle centrally

P_{19}: adjusted policy and made covering offer centrally

P_{20}: made covering offer by authorized garage

P_{21}: evaluated damage by adjustor

P_{22}: made covering offer by adjustor

Transitions

q_1: handle accident info centrally

q_2: handle accident info by authorized garage

q_3: handle accident info by adjustor

q_4: verify policy covering centrally

q_5: verify policy covering by authorized garage

q_6: verify policy covering by adjustor

q_7: corroborate accident details

q_8: evaluate damage

q_9: get medical treatment cost

q_{10}: determine accident in range

q_{11}: send info to be handle centrally

q_{12}: get accident info by adjustor

q_{13}: assess client antecedents

q_{14}: determine accident covering centrally

q_{15}: get accident info by authorized garage

q_{16}: evaluate damage by authorized garage

q_{17}: determine accident in range

q_{18}: send info to be handle centrally

q_{19}: adjust policy and make covering offer centrally

q_{20}: make covering offer by authorized garage

q_{21}: evaluated damage by adjustor

q_{22}: make covering offer by adjustor

(i) *Stability*

From the incidence matrix A of the DPPN of Fig. 1 and choosing

$$\Phi = [1, 1/2, 1, 1, 1/2, 1, 1, 1/2, 1/2, 1/2, 1, 1,$$
$$1, 1/2, 1, 1, 2, 1, 1, 5/2, 3, 1, 1/2],$$

$\Phi > 0$, we obtain that $A\Phi \le 0$ concluding stability.

(ii) *Optimum strategy*

Define the Lyapunov like function L in terms of the entropy $H(p_i) = -p_i \ln p_i$ as

$$L = \max_{i=1,\dots,|\alpha|} (-\alpha_i \ln \alpha_i),$$

(I) The optimum strategy $\sigma^\triangle$ for accidents of considerable size that must be manages centrally by the assurance company is represented by

$$U_{k=0}(p_0) = 1,$$
$$U_{k=0}^{\sigma_{hj}}(p_1) = L[\sigma_{01}(p_1) \cdot U_{k=0}^{\sigma_{01}}(p_0)]$$
$$= L[1/3 \cdot 2 \cdot 1] = \max H[2/3] = 0.270,$$
$$U_{k=0}^{\sigma_{hj}}(p_4) = L[\sigma_{14}(p_4) \cdot U_{k=0}^{\sigma_{14}}(p_1)]$$
$$= L[1 \cdot 0.270] = \max H[0.270] = 0.353,$$
$$U_{k=0}^{\sigma_{hj}}(p_7) = L[\sigma_{47}(p_7) \cdot U_{k=0}^{\sigma_{47}}(p_4)]$$
$$= L[2/5 \cdot 0.353] = \max H[0.141] = 0.276,$$
$$U_{k=0}^{\sigma_{hj}}(p_8) = L[\sigma_{48}(p_8) \cdot U_{k=0}^{\sigma_{48}}(p_4)]$$
$$= L[1/5 \cdot 0.353] = \max H[0.070] = 0.187,$$
$$U_{k=0}^{\sigma_{hj}}(p_9) = L[\sigma_{49}(p_9) \cdot U_{k=0}^{\sigma_{49}}(p_4)]$$
$$= L[2/5 \cdot 0.353] = \max H[0.141] = 0.276,$$
$$U_{k=0}^{\sigma_{hj}}(p_{13}) = L[\sigma_{7,13}(p_{13}) \cdot U_{k=0}^{\sigma_{7,13}}(p_7)]$$
$$= L[1 \cdot 0.276] = \max H[0.276] = 0.355,$$
$$U_{k=0}^{\sigma_{hj}}(p_{14}) = L[\sigma_{8,14}(p_{14}) \cdot U_{k=0}^{\sigma_{8,14}}(p_8)$$
$$+ \sigma_{9,14}(p_{14}) \cdot U_{k=0}^{\sigma_{9,14}}(p_9)]$$
$$= L[1 \cdot 0.187 + 1 \cdot 0.276]$$
$$= \max H[0.463] = 0.356,$$
$$U_{k=0}^{\sigma_{hj}}(p_{19}) = L[\sigma_{13,19}(p_{19}) \cdot U_{k=0}^{\sigma_{13,19}}(p_{13})$$
$$+ \sigma_{14,19}(p_{19}) \cdot U_{k=0}^{\sigma_{14,19}}(p_{14})]$$
$$= L[1 \cdot 0.355 + 1/2 \cdot 0.356]$$
$$= \max H[0.533] = 0.335,$$

where the firing transition vector is u.

For this case, the adjustor or the garage must abort the process because the accident is out of their range obtaining that

$$U_{k=0}^{\sigma_{hj}}(p_{11}) = L[\sigma_{5,11}(p_{11}) \cdot U_{k=0}^{\sigma_{5,11}}(p_5)]$$
$$= L[4/5 \cdot 0.367] = \max H[0.293] = 0.359,$$
$$U_{k=0}^{\sigma_{hj}}(p_{18}) = L[\sigma_{12,18}(p_{18}) \cdot U_{k=0}^{\sigma_{12,18}}(p_{12})]$$
$$= L[3/4 \cdot 0.367] = \max H[0.275] = 0.355,$$

$$u = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
\hline
q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 & q_9 & q_{10} & q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} & q_{17} & q_{18} & q_{19} & q_{20} & q_{21} & q_{22} \\
\hline
\end{array}$$

$$u' = \begin{array}{|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|c|}
\hline
0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\
\hline
q_1 & q_2 & q_3 & q_4 & q_5 & q_6 & q_7 & q_8 & q_9 & q_{10} & q_{11} & q_{12} & q_{13} & q_{14} & q_{15} & q_{16} & q_{17} & q_{18} & q_{19} & q_{20} & q_{21} & q_{22} \\
\hline
\end{array}$$

concluding $U_{k=0}^{\sigma_{hj}}(p_{19}) < U_{k=0}^{\sigma_{hj}}(p_{18}) < U_{k=0}^{\sigma_{hj}}(p_{11})$, i.e., $U_{k=0}^{\sigma_{hj}}(p_{18}), U_{k=0}^{\sigma_{hj}}(p_{19})$ are more expensive than $U_{k=0}^{\sigma_{hj}}(p_{11})$.

(II) The optimum strategy $\sigma'^{\triangle}$ for small accidents that must be managed ideally by the company centrally is represented by

$$U_{k=0}^{\sigma_{hj}}(p_2) = L[\sigma_{02}(p_2) \cdot U_{k=0}^{\sigma_{02}}(p_0)]$$
$$= L[1/3 \cdot 1] = \max H[1/3 \cdot 1] = 0.366,$$
$$U_{k=0}^{\sigma_{hj}}(p_5) = L[\sigma_{25}(p_5) \cdot U_{k=0}^{\sigma_{25}}(p_2)]$$
$$= L[1 \cdot 0.366] = \max H[1 \cdot 0.366] = 0.367,$$
$$U_{k=0}^{\sigma_{hj}}(p_{10}) = L[\sigma_{5,10}(p_{10}) \cdot U_{k=0}^{\sigma_{5,10}}(p_5)]$$
$$= L[1/5 \cdot 0.367]$$
$$= \max H[1/5 * 0.367] = 0.191,$$
$$U_{k=0}^{\sigma_{hj}}(p_{15}) = L[\sigma_{10,15}(p_{15}) \cdot U_{k=0}^{\sigma_{10,15}}(p_{10})]$$
$$= L[6/8 \cdot 0.191]$$
$$= \max H[6/8 \cdot 0.191] = 0.278,$$
$$U_{k=0}^{\sigma_{hj}}(p_{16}) = L[\sigma_{10,16}(p_{16}) \cdot U_{k=0}^{\sigma_{10,16}}(p_{10})]$$
$$= L[1/8 \cdot 0.191]$$
$$= \max H[1/8 \cdot 0.191] = 0.089,$$
$$U_{k=0}^{\sigma_{hj}}(p_{20}) = L[\sigma_{15,20}(p_{20}) \cdot U_{k=0}^{\sigma_{15,20}}(p_{15})$$
$$+ \sigma_{16,20}(p_{20}) \cdot U_{k=0}^{\sigma_{16,20}}(p_{16})]$$
$$= L[1 \cdot 0.278 + 1 \cdot 0.089]$$
$$= \max H[1 \cdot 0.278 + 1 \cdot 0.089] = 0.367,$$

the firing transition vector being u'.

Intuitively, the result is correct, because the best option for the insurance company after a car accident happens is to send an adjustor before the customer takes the car to the garage.

(III) The strategy σ'' for small accidents that must be managed by the adjustor is represented by

$$U_{k=0}^{\sigma_{hj}}(p_3) = L[\sigma_{03}(p_3) \cdot U_{k=0}^{\sigma_{03}}(p_0)]$$
$$= L[1/3 \cdot 1] = \max H[1/3 \cdot 1] = 0.366,$$
$$U_{k=0}^{\sigma_{hj}}(p_6) = L[\sigma_{36}(p_6) \cdot U_{k=0}^{\sigma_{36}}(p_3)]$$
$$= L[1 \cdot 0.366] = \max H[1 \cdot 0.366] = 0.367,$$
$$U_{k=0}^{\sigma_{hj}}(p_{12}) = L[\sigma_{6,12}(p_{12}) \cdot U_{k=0}^{\sigma_{6,12}}(p_6)]$$
$$= L[1 \cdot 0.367] = \max H[1 \cdot 0.367] = 0.367,$$

$$U_{k=0}^{\sigma_{hj}}(p_{17}) = L[\sigma_{12,17}(p_{17}) \cdot U_{k=0}^{\sigma_{12,17}}(p_{12})]$$
$$= L[1/4 \cdot 0.367]$$
$$= \max H[1/4 \cdot 0.367] = 0.219,$$
$$U_{k=0}^{\sigma_{hj}}(p_{21}) = L[\sigma_{17,21}(p_{21}) \cdot U_{k=0}^{\sigma_{17,21}}(p_{17})]$$
$$= L[1 \cdot 0.219] = \max H[1 \cdot 0.219] = 0.332,$$
$$U_{k=0}^{\sigma_{hj}}(p_{22}) = L[\sigma_{21,22}(p_{22}) \cdot U_{k=0}^{\sigma_{21,22}}(p_{21})]$$
$$= L[2 \cdot 0.332] = \max H[2 \cdot 0.332] = 0.271,$$

the firing transition vector being u''.

Notice that, since $U_{k=0}^{\sigma_{hj}}(p_{20})$ is greater than $U_{k=0}^{\sigma_{hj}}(p_{22})$, small accidents must be handled by the adjustor whenever possible.

3. Hierarchical decision process Petri net

Let $\mathrm{DPPN} = \{P, Q, F, W, M_0, \pi, U\}$ be a decision process Petri net and let $f : P \cup Q \to 2^{P \cup Q}$ be a refinement function such that for each $s \in P \cup Q$ the symbol $f(s)$ defines the immediate descendant element of s.

Let $\equiv_f$ be the equivalence relation on $P \cup Q$ induced by f such that

$$\forall s_1, s_2 \in P \cup Q : s_1 \equiv_f s_2 \iff f(s_1) =_f f(s_2). \quad (9)$$

Then the collection of equivalence classes $(P \cup Q/ \equiv_f) = \{\mathcal{C}(s)|s \in P \cup Q\}$, where $\mathcal{C}$ denotes class, is a poset. Thus, $(P \cup Q/ \equiv_f)$ is linearly ordered and, consequently, it is a lattice. The structure $(P \cup Q/ \equiv_f)$ is indeed trivial: all elements in $P \cup Q$ belonging to the same net under f are identified in this quotient set.

On the other hand, let us consider the relation $\leq_f$ as follows:

$$\forall s_1, s_2 \in P \cup Q : s_1 \leq_f s_2 \iff f(s_1) \leq_f f(s_2). \quad (10)$$

This relation is reflexive and transitive, but it is not antisymmetric in most cases[1]. Thus, $\leq_f$ is not an ordering in $P \cup Q$.

At this point let us recall some basic notions on orderings. A binary relation $\leq$ over a set X is a *partial order* if it satisfies the following three properties: reflexivity, antisymmetry and transitivity. A *total order* is a partial order that satisfies a fourth property known as comparability, where every element is related to every element in one way or another. A set and a partial order on that set define a *partially ordered set*, or *poset* for short. A *quasi-order* is

---

[1]It is antisymetric if and only if f is one-to-one.

$u'' =$	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	1	1	0	0	1	1
	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}	q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}
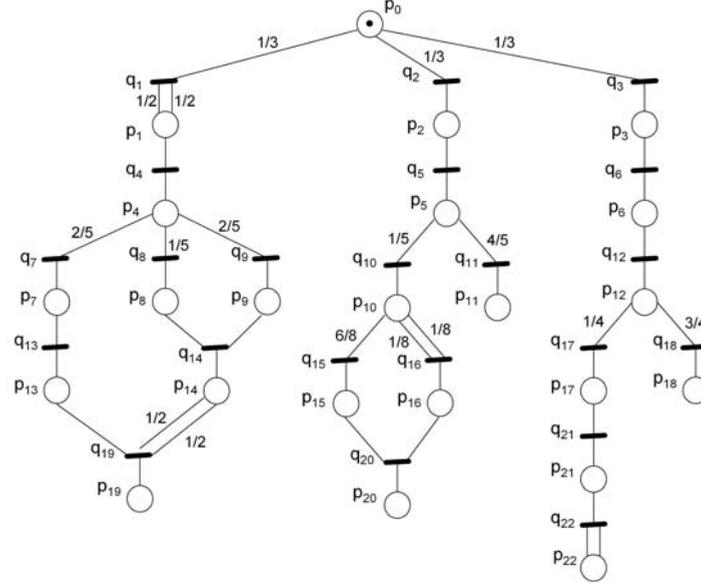
Fig. 1. BRP DPPN.

a relation $\leq$ that satisfies reflexivity and transitivity. Formally, let $(X, \leq)$ be a poset and let $S \subseteq X$. Then an element $b \in S$ is a *minimal* element of S if there is no element $a \in S$ that satisfies $a \leq b$. Similarly, an element $b \in S$ is a *maximal* element of S if there is no element $a \in S$ that satisfies $b \leq a$. It is important to mention maximal elements that are in general not the greatest elements of a subset S. Formally, we have that an element $b \in S$ is the *greatest element* of S if, for every element $a \in S$, $a \leq b$. Dually, an element $b \in S$ is the *least element* of S if for every element $a \in S$, $b \leq a$. Note that the least element of a poset is unique if one exists because of the antisymmetry of $\leq$. A *strict partial order* is a binary relation which is irreflexive, asymmetric and transitive. Strict partial orders correspond to Directed Acyclic Graphs (DAGs), such that every strict partial order is a DAG, and the transitive closure of a DAG is both a strict partial order and also a DAG itself.

For any $s \in P \cup Q$, define the *successors* of s:

$$t \in suc(s) \quad \text{iff} \quad s \neq t, s \leq_f t$$

and

$$\forall t_1 : s \leq_f t_1 \leq_f t \implies (t_1 =_f s) \vee (t_1 =_f t). \quad (11)$$

For any $s \in P \cup Q$, define the *predecessors* of s:

$$t \in pre(s) \quad \text{iff} \quad t \neq s, t \leq_f s$$

and

$$\forall t_1 : t \leq_f t_1 \leq_f s \implies (t_1 =_f t) \vee (t_1 =_f s). \quad (12)$$

Therefore, let $P \cup Q$ be ordered by the following relationship:

$$\forall s_1, s_2 \in P \cup Q,$$
$$s_1 < s_2 \Leftrightarrow (s_1 <_f s_2) \vee (s_1 \equiv_f s_2) \vee (s_2 <_f s_1). \quad (13)$$

Thus, f is inducing a hierarchical structure on the DPPN.

Therefore, we can introduce the hierarchical partition $\{\text{DPPN}_\xi\}_{\xi \in \Xi}$ (where Ξ is a finite set) of the DPPN induced by f, such that each pair $(s, t) \in P_\xi \cup Q_\xi : (s, t)$ is an edge iff $t \in suc(s)$ in the DPPN_ξ (or, equivalently, $s \in pre(t)$). We say that f is consistent if the hierarchical structure has no cycles. From now on, we will consider only consistent functions.

Definition 6. A *hierarchical decision process Petri net* HDPPN is the graph whose set of nodes are the partition $\{\text{DPPN}_\xi\}_{\xi \in \Xi}$ induced by a refinement function f.

The *minimal elements* are those with no predecessors, i.e., nodes with a null inner degree in the HDPPN. The *maximal elements* are those with no successors, i.e., node with a null outer degree in the HDPPN.

Let us define the *upper distance* d^+ as follows:

$$d^+(s, t) = 1 \iff t \in suc(s),$$
$$d^+(s, t) = 1 + r$$
$$\iff \exists t_1 : d^+(s, t_1) = r \,\&\, d^+(t_1, t) = 1.$$

Similarly, the *lower distance* d^- is

$$d^-(s,t) = 1 \Longleftrightarrow t \in pre(s),$$
$$d^-(s,t) = 1 + r$$
$$\Longleftrightarrow \exists t_1 : d^-(s,t_1) = r \ \& \ d^-(t_1,t) = 1.$$

Thus $d^+(s,t) = d^-(t,s)$.

The *upper height* of a node s is $h^+(s) = \text{Max}\{d^+(s_1,s)|s_1 \text{ is minimal}\}$. The *lower height* of a node s is $h^-(s) = \text{Max}\{d^-(s_1,s)|s_1 \text{ is maximal}\}$.

Let P_ξ and Q_ξ be the sets of places and transitions of the DPPN_ξ. Places and transitions in the HDPPN are enumerated consecutively and will receive the number of the corresponding DPPN_ξ if necessary, i.e., $p_{\xi i}$ corresponds to the place i at the DPPN_ξ; otherwise, we will identify the place only as p_i.

Let $M_{\xi k}(p_{\xi i})$ denote the marking (i.e., the number of tokens) at place $p_{\xi i} \in P_\xi$ at time k, and let

$$M_{\xi k} = [M_{\xi k}(p_{\xi 1}), \ldots, M_{\xi k}(p_{\xi m})]^T$$

denote the marking (state) of the DPPN_ξ at time k. A transition $q_{\xi j} \in Q_\xi$ is said to be enabled at time k if $M_{\xi k}(p_{\xi 1}) \geq W_\xi(p_{\xi 1}, q_{\xi j})$ for all $p_{\xi i} \in P_\xi$ such that $(p_{\xi i}, q_{\xi j}) \in F_\xi$. It is assumed that at each time k there exists at least one transition to fire, i.e., it is not possible to block the net. If a transition is enabled, then it can fire. If an enabled transition $q_{\xi j} \in Q_\xi$ fires at time k, then the next marking for $p_{\xi i} \in P_\xi$ is given by

$$M_{\xi k+1}(p_{\xi i})$$
$$= M_{\xi k}(p_{\xi i}) + W_\xi(q_{\xi j}, p_{\xi i}) - W_\xi(p_{\xi i}, q_{\xi j}).$$

Let $A_\xi = [a_{ij}]$ denote an $n \times m$ matrix of integers (the incidence matrix), where $a_{ij} = a_{ij}^+ - a_{ij}^-$ with $a_{ij}^+ = W_\xi(q_{\xi i}, p_{\xi j})$ and $a_{ij}^- = W_\xi(p_{\xi j}, q_{\xi i})$. Let $u_k \in \{0,1\}^n$ denote a firing vector where, if $q_{\xi j} \in Q_\xi$ is fired, then its corresponding firing vector is $u_k = [0, \ldots, 0, 1, 0, \ldots, 0]^T$ with the "1" in the j-th position in the vector and zeros everywhere else. The matrix equation (nonlinear difference equation) describing the dynamical behavior represented by a Petri net is

$$M_{\xi k+1} = M_{\xi k} + A_\xi^T u_k, \qquad (14)$$

where, if at step k, $a_{ij}^- < M_{\xi k}(p_{\iota j})$ for all $p_{\xi i} \in P_\xi$, then $q_{\xi i} \in Q_\xi$ is enabled and, if this $q_{\xi i} \in Q_\xi$ fires, then its corresponding firing vector u_k is utilized in the difference equation (4) to generate the next step. Notice that, if M_ξ' can be reached from some other marking M_ξ and if we fire some sequence of d transitions with corresponding firing vectors $u_0, u_1, \ldots, u_{d-1}$, we obtain that

$$M_\xi' = M_\xi + A_\xi^T u, \quad u = \sum_{k=0}^{d-1} u_k. \qquad (15)$$

In Fig. 2 we have partial routing policies π that generate a transition from state p_1 to state p_2, where $p_1, p_2 \in P$:

- *Case 1.* The probability that q_1 generates a transition from state p_1 to p_2 is 1/3. But, because q_1 transition to state p_2 has two arcs, the probability to generate a transition from state p_1 to p_2 is increased to 2/3.

- *Case 2.* We set by convention the probability that q_1 generates a transition from state p_1 to p_2 to 1/3 (1/6 plus 1/6). However, because the transition q_1 to state p_2 has only one arc, the probability to generate a transition from state p_1 to p_2 is decreased to 1/6.

- *Case 3.* Finally, we have the trivial case when there exists only one arc from p_1 to q_1 and from q_1 to p_2.

Remark 1. In the previous definition we were considering nets with a single initially marked place.

Remark 2. The previous definition in no way changes the behavior of the place-transitions Petri net; the *routing policy is used to calculate the utility value at each place of the net.*

Remark 3. It is important to note that the utility value can be re-normalized after each transition or time k of the net.

$U_k(\cdot)$ denotes the utility at place $p_{\xi i} \in P_\xi$ at time k and let $U_k = [U_k(\cdot), \ldots, U_k(\cdot)]^T$ denote the utility state of the HDPPN at time k. $FN_\xi : F_\xi \to \mathbb{R}_+$ is the number of arcs from place p to transition q at level ξ (the number of arcs from transition q to place p). The rest of HDPPN functionality is as described above.

Consider an arbitrary $p_{\xi i} \in P_\xi$. For each fixed transition $q_{\xi j} \in Q_\xi$ that forms an output arc $(q_{\xi j}, p_{\xi i}) \in O_\xi$, we look at all the previous places $p_{\xi h}$ of the place $p_{\xi i}$ denoted by the list (set) $p_{\xi \eta_{ij}} = \{p_{\xi h} : h \in \eta_{ij}\}$, where $\eta_{ij} = \{h : (p_{\xi h}, q_{\xi j}) \in I \ \& \ (q_{\xi j}, p_{\xi i}) \in O\}$, that materialize all the input arcs $(p_{\xi h}, q_{\xi j}) \in I_\xi$ and form the sum

$$\sum_{h \in \eta_{ij}} \Psi(p_{\xi h}, q_{\xi j}, p_{\xi i}) \cdot U_k(p_{\xi h}), \qquad (16)$$

where

$$\Psi(p_{\xi h}, q_{\xi j}, p_{\xi i}) = \pi(p_{\xi h}, q_{\xi j}) \cdot \frac{FN_\xi(q_{\xi j}, p_{\xi i})}{FN_\xi(p_{\xi h}, q_{\xi j})}$$

and the index sequence j is the set $\{j : q_{\xi j} \in (p_{\xi h}, q_{\xi j}) \cap (q_{\xi j}, p_{\xi i}) \ \& \ p_{\xi h} \text{ running over the set } p_{\xi \eta_{ij}}\}$.
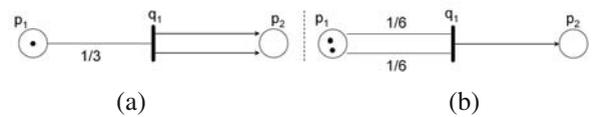
Fig. 2. Routing policy: Case 1 (a), Case 2 (b).

Proceeding with all the $q_{\xi j}$s we form the vector indexed by the sequence j identified by $(j_0, j_1, \ldots, j_f)$ as follows:

$$
\begin{bmatrix}
\sum_{h \in \eta_{ij_0}} \Psi(p_{\xi h}, q_{\xi j_0}, p_{\xi i}) \cdot U_k(p_{\xi h}), \\
\sum_{h \in \eta_{ij_1}} \Psi(p_{\xi h}, q_{\xi j_1}, p_{\xi i}) \cdot U_k(p_{\xi h}), \ldots, \\
\sum_{h \in \eta_{ij_f}} \Psi(p_{\xi h}, q_{\xi j_f}, p_{\xi i}) \cdot U_k(p_{\xi h}).
\end{bmatrix}. \tag{17}
$$

Intuitively, the vector (17) represents all the possible trajectories to a place $p_{\xi i}$ for a fixed i and ξ through the transitions $q_{\xi j}$s where $(j_1, j_2, \ldots, j_f)$.

Continuing the construction of the definition of the utility function U, let us introduce the following definition.

Definition 7. Let $L : \mathbb{R}^n \to \mathbb{R}_+$ be a continuous map. Then, L is a *Lyapunov-like function* (see (Kalman and Bertram, 1960)) iff it satisfies the following properties:

1. $\exists x^*$ such that $L(x^*) = 0$,

2. $L(x) > 0$ for $\forall x \neq x^*$,

3. $L(x) \to \infty$ as $x \to \infty$,

4. $\Delta L = L(x_{i+1}) - L(x_i) < 0$ for all $x_i, x_{i+1} \neq x^*$.

Then, we formally define the utility function U as follows.

Definition 8. Let HDPPN be a hierarchical decision process Petri net. The *utility function U* is represented by the equation

$$
U_k^{q_{\xi j}}(p_{\xi i})
$$
$$
= \begin{cases}
U_k(p_0) & \text{if } i = 0,\, k = 0, \\
L(\alpha) & \text{if } i > 0,\, k = 0 \,\&\, i \geq 0,\, k > 0, \\
U_k^{q_{\xi j}}(p'_{\xi i}) & \text{if } i > 0,\, k = 0 \,\&\, i \geq 0,\, k > 0,
\end{cases} \tag{18}
$$

where

$$
\alpha = \begin{bmatrix}
\sum_{h \in \eta_{ij_0}} \Psi(p_{\xi h}, q_{\xi j_0}, p_{\xi i}) \cdot U_k^{q_{\xi j_0}}(p_{\xi h}), \\
\sum_{h \in \eta_{ij_1}} \Psi(p_{\xi h}, q_{\xi j_1}, p_{\xi i}) \cdot U_k^{q_{\xi j_1}}(p_{\xi h}), \ldots, \\
\sum_{h \in \eta_{ij_f}} \Psi(p_{\xi h}, q_{\xi j_f}, p_{\xi i}) \cdot U_k^{q_{\xi j_f}}(p_{\xi h})
\end{bmatrix}, \tag{19}
$$

the place $p'_{\xi i} \in f(p_{\xi i})$ is the initial marked place of the DPPN$_\xi$, the function $L : D \subseteq \mathbb{R}^n_+ \to \mathbb{R}_+$ is a Lyapunov-like function which optimizes the utility through all possible transitions (i.e., through all the possible trajectories

defined by the different $q_{\xi j}$s), D is the decision set formed by the j's; $0 \leq j \leq f$ of all those possible transitions ($q_{\xi j}$ $p_{\xi i}$) $\in O$,

$$
\Psi(p_{\xi h}, q_{\xi j}, p_{\xi i}) = \pi(p_{\xi h}, q_{\xi j}) \cdot \frac{FN(q_{\xi j}, p_{\xi i})}{FN(p_{\xi h}, q_{\xi j})},
$$

η_{ij} is the index sequence of the list of previous places to $p_{\xi i}$ through transition $q_{\xi j}$, $p_{\xi h}$ ($h \in \eta_{ij}$) is a specific previous place of $p_{\xi i}$ through transition $q_{\xi j}$.

Remark 4.

- Note that the Lyapunov-like function L guarantees that an optimal course of action is followed (taking into account all the the possible paths defined). In addition, the function L establishes a preference relation because by definition L is asymptotic. This condition gives the decision maker the opportunity to select a path that optimizes the utility.

- The iteration over k for U is as follows:

  1. For $i = 0$ and $k = 0$ the utility is $U_0(p_0)$ at place p_0 and for the rest of the places p_i the utility is zero.

  2. For $i \geq 0$ and $k > 0$ the utility is $U_k^{q_{\xi j}}(p_{\xi i})$ at each place $p_{\xi i}$, and it is computed by taking into account the utility value of the previous places $p_{\xi h}$ for k and $k - 1$ (when needed).

Property 1. *The continuous function $U(\cdot)$ satisfies the following properties:*

1. *There is a $p^\triangle \in P$ such that*

   (a) *if there exists an infinite sequence $\{p_i\}_{i=1}^\infty \in P$ with $p_n \underset{n \to \infty}{\to} p^\triangle$ such that $0 \leq \cdots < U(p_n) < U(p_{n-1}) \cdots < U(p_1)$, then $U(p^\triangle)$ is the infimum, i.e., $U(p^\triangle) = 0$,*

   (b) *if there exists a finite sequence $p_1, \ldots, p_n \in P$ with $p_1, \ldots, p_n \to p^\triangle$ such that $C = U(p_n) < U(p_{n-1}) \cdots < U(p_1)$, then $U(p^\triangle)$ is the minimum, i.e., $U(p^\triangle) = C$, where $C \in \mathbb{R}$, $(p^\triangle = p_n)$.*

2. *$U(p) > 0$ or $U(p) > C$, where $C \in \mathbb{R}$, $\forall p \in P$ such that $p \neq p^\triangle$.*

3. *For all p_i and $p_{i-1} \in P$ such that $p_{i-1} \leq_U p_i$ we have $\Delta U = U(p_i) - U(p_{i-1}) < 0$.*

4. *The routing policies decrease monotonically, i.e., $\pi_i \geq \pi_j$ (notice that the indexes i and j are taken $j > i$ along a trajectory to the infimum or the minimum).*

Remark 5. In Property 1, Point 3, we state that $\Delta U = U(p_i) - U(p_{i-1}) < 0$ for determining the asymptotic condition of the Lyapunov-like function. However, it is easy to show that such a property is convenient for deterministic systems. In Markov decision process systems it is necessary to include probabilistic decreasing asymptotic conditions to guarantee the asymptotic condition of the Lyapunov-like function.

Property 2. *The utility function $U : P \to \mathbb{R}_+$ is a Lyapunov-like function.*

Remark 6. From Properties 1 and 2, we have the following:

- $U(p^\triangle) = 0$ or $U(p^\triangle) = C$ means that a final state is reached. Without lost generality we can say that $U(p^\triangle) = 0$ by means of a translation to the origin.

- In Property 1 we determine that the Lyapunov-like function $U(p)$ approaches to a infimum/minimum when p is large thanks to Property 4 of Definition 7.

- Property 1, Point 3, is equivalent to the following statement: $\exists \varepsilon > 0$ such that $|U(p_i) - U(p_{i-1})| > \varepsilon$, $\forall p_i, p_{i-1} \in P$ such that $p_{i-1} \leq_U p_i$.

Property 3. *The marking and the enabling conditions as well as and utility function calculation of the* DPPN *and the* HDPPN *are equivalent.*

The previous formalisms of the HDPPN are described as follows. The DPPN is refined in a subset of local DPPN$_\xi$ determined by a partition and hierarchically structure through a refinement function f. Each local decision process Petri net DPPN$_\xi$ describes more detailed local behavior of a given transition of its immediately high level DPPN$_\xi$. The function f defines the immediate descendant element of the net, and therefore the hierarchy can be considered as a continuous net. The inverse function f^{-1} defines the parent of a given node of the net. The utility is calculated recursively by the hierarchical tracking of the HDPPN. It is important to note that the HDPPN can be transformed into a flat DPPN simply by replacing each DPPN$_\xi$ in its immediate parent.

For optimization reasons, we want to contemplate asynchronous behavior of each DPPN$_\xi$ in the hierarchical structure realized by tokens moving at the different levels of the HDPPN. We can conceptualize a DPPN$_\xi$ as embedded by a set of input and output ports that determine it scope. The input or output ports are places. Port places can be only connected with transitions. The input place port accepts tokens fired by its immediate high level DPPN$_\xi$. When a token reaches a place, it is reserved for the firing of a given transition according to the routing policy determined by U. A transition q must fire as soon as all the places $p_i \in P$ contain enough tokens reserved for the transition q. Once the transition fires, it consumes the corresponding tokens and immediately produces an amount

of tokens in each subsequent places $p_j \in P$. When $\pi(\iota) = 0$ for $\iota \in I$, this means that there are no arcs in the place-transitions Petri net.

Definition 9. A *local equilibrium point* with respect to local decision process Petri net DPPN$_\xi \in$ HDPPN is a place $p_{\xi i} \in P_\xi$ such that $M_{\xi l}(p_{\xi i}) = S < \infty, \forall l \geq k$ and $p_{\xi i}$ is the last place of the net.

Definition 10. A *global equilibrium point* with respect to a hierarchical decision process Petri net HDPPN is a place $p^* \in P$ such that $M_l(p^*) = S < \infty, \forall l \geq k$ and p^* is the last place of the net.

Definition 11. A *local final decision point* $p_{\xi f} \in P_\xi$ with respect a DPPN$_\xi \in$ HDPPN is a place $p \in P_\xi$ where the infimum or a minimum is attained, i.e., $U(p) = 0$ or $U(p) = C$.

Definition 12. A *global final decision point* $p_f \in P$ with respect to a hierarchical decision process Petri net HDPPN is a place $p \in P$ where the infimum or the minimum is attained, i.e., $U(p) = 0$ or $U(p) = C$.

Remark 7. A *local final decision point* $p_{\xi f}$ is a global final decision point if the utility function attains the infimum or the minimum.

Definition 13. A *local optimum point* $p^{\xi\triangle} \in P_\xi$ with respect to a DPPN$_\xi \in$ HDPPN is a local final decision point $p_{\xi f} \in P_\xi$ where the best choice is selected 'according to some local criteria'.

Definition 14. A *global optimum point* $p^\triangle \in P$ with respect to a hierarchical decision process Petri net HDPPN is a global final decision point $p_f \in P$ where the best choice is selected 'according to some global criteria'.

Property 4. *Every hierarchical decision process Petri net* HDPPN *has a final decision point.*

Remark 8. If there are $p_1, \ldots, p_n \in P$, such that $U(p_1) = \cdots = U(p_n) = 0$, then $p_1, \ldots, p_n$ are optimum points.

Definition 15. A *global strategy* with respect to a hierarchical decision process Petri net $HDPPN$ is identified by σ and consists of the routing policy transition sequence represented in the HDPPN graph model such that some point $p \in P$ is reached.

Definition 16. A *local strategy* with respect to a hierarchical decision process Petri net HDPPN is identified by σ_ξ and consists of the routing policy transition sequence represented in the $DPPN_\xi \in HDPPN$ graph model such that some point $p \in P_\xi$ is reached.

Definition 17. An *optimum global strategy* with respect to a hierarchical decision process Petri net HDPPN is identified by $\sigma^\triangle$ and consists of the routing policy transition sequence represented in the HDPPN graph model such that an optimum point $p^\triangle \in P$ is reached.

Definition 18. An *optimum local strategy* with respect to a hierarchical decision process Petri net HDPPN is identified by $\sigma_\xi^\triangle$ and consists of the routing policy transition sequence represented in the $DPPN_\xi \in HDPPN$ graph model such that a local optimum point $p^{\xi\triangle} \in P_\xi$ is reached.

Equivalently, we can represent (18, 19) as follows:

$$U_k^{\sigma_{\xi h j}}(p_{\xi i})$$
$$= \begin{cases} U_k(p_0) & \text{if } i = 0, \ k = 0, \\ L(\alpha) & \text{if } i > 0, \ k = 0 \ \& \ i \geq 0, \ k > 0, \\ U_k^{\sigma_{\xi h j}}(p'_{\xi i}) & \text{if } i > 0, \ k = 0 \ \& \ i \geq 0, \ k > 0, \end{cases}$$
$$(20)$$

$$\alpha = \Big[\sum_{h \in \eta_{ij_0}} \sigma_{\xi h j_0}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_0}}(p_{\xi h}),$$
$$\sum_{h \in \eta_{ij_1}} \sigma_{\xi h j_1}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_1}}(p_{\xi h}), \dots,$$
$$\sum_{h \in \eta_{ij_f}} \sigma_{\xi h j_f}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_f}}(p_{\xi h}) \Big],$$
$$(21)$$

where $\sigma_{\xi h j}(p_{\xi i}) = \Psi(p_{\xi h}, q_{\xi j}, p_{\xi i})$. The rest is as previously defined.

For notational simplicity, we will represent the utility function U as follows:

1. for any transition and any strategy:

$$U_k(p_i) \triangleq U_k^{q_{\xi j}}(p_i) \triangleq U_k^{\sigma_{\xi h j}}(p_{\xi i}) \triangleq U_k^{\sigma_{h j}}(p_i),$$

2. for an optimum transition and optimum strategy:

$$U_k^\triangle(p_{\xi i}) \triangleq U_k^{q_{\xi j}^\triangle}(p_{\xi i}) \triangleq U_k^{\sigma_{\xi h j}^\triangle}(p_{\xi i}) \triangleq U_k^{\sigma_{h j}^\triangle}(p_i).$$

The reader will easily identify which notation is used depending on the context.

Definition 19. Let HDPPN be a hierarchical decision process Petri net. A *trajectory* ω is an (finite or infinite) ordered subsequence of places

$$p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \cdots$$

such that a given strategy σ holds.

Definition 20. Let HDPPN be a hierarchical decision process Petri net. An *optimum trajectory* ω is an (finite or infinite) ordered subsequence of places

$$p_{\varsigma(1)} \leq_{U_k^\triangle} p_{\varsigma(2)} \leq_{U_k^\triangle} \cdots \leq_{U_k^\triangle} p_{\varsigma(n)} \leq_{U_k^\triangle} \cdots$$

such that the optimum strategy $\sigma^\triangle$ holds.

Theorem 2. *Let* HDPPN *be a non-blocking hierarchical decision process Petri net (unless $p \in P$ is a global equilibrium point). Then we have*

$$U_k^\triangle(p^\triangle) \leq U_k(p), \quad \forall \sigma, \sigma^\triangle.$$

Proof. We have that

$$U_k^{\sigma_{\xi h j}}(p_{\xi i})$$
$$= \begin{cases} U_k(p_0) & \text{if } i = 0, \ k = 0, \\ L(\alpha) & \text{if } i > 0, \ k = 0 \ \& \ i \geq 0, \ k > 0, \\ U_k^{\sigma_{\xi h j}}(p'_{\xi i}) & \text{if } i > 0, \ k = 0 \ \& \ i \geq 0, \ k > 0, \end{cases}$$

$$\alpha = \Big[\sum_{h \in \eta_{ij_0}} \sigma_{\xi h j_0}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_0}}(p_{\xi h}),$$
$$\sum_{h \in \eta_{ij_1}} \sigma_{\xi h j_1}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_1}}(p_{\xi h}), \dots,$$
$$\sum_{h \in \eta_{ij_f}} \sigma_{\xi h j_f}(p_{\xi i}) \cdot U_k^{\sigma_{\xi h j_f}}(p_{\xi h}) \Big].$$

Then, starting from p_0 and proceeding with the iteration, eventually the trajectory ω given by $p_0 = p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \cdots$ which converges to $p^\triangle$, i.e., the optimum trajectory, is obtained. Since at the optimum trajectory the optimum strategy $\sigma^\triangle$ holds, we have that $U_k^\triangle(p^\triangle) \leq U_k(p), \quad \forall \sigma, \sigma^\triangle$. ■

Remark 9. The inequality $U_k^\triangle(p^\triangle) \leq U_k(p)$ means that the utility is optimum when the optimum strategy is applied.

Theorem 3. *The behavior of the* HDPPN *is equivalent to the behavior of the* DPPN.

Proof. This follows from Definition 6, Definition 8 and is ensured by Theorem 2. ■

3.1. Optimum trajectory planning. Given a non-blocking (unless $p \in P$ is an equilibrium point) hierarchical decision process Petri net HDPPN, the optimum trajectory planning consists in finding a firing transition sequence u such that the optimum target state M_t with the optimum point is achieved. The target state M_t belongs to the reachability set $R(M_0)$ and satisfies the assumption that it is the last and final task processed by the HDPPN with some fixed starting state M_0 with utility U_0.

Theorem 4. *Let* $DPPN_\xi \in$ HDPPN *be a decision process Petri net. The optimum local trajectory planning problem is solvable.*

Proof. From what is shown in Theorem 2, for each step we find $U_k^\triangle(p_{\xi\varsigma(1)}), \dots, U_k^\triangle(p_{\xi\varsigma(i)}), \dots, U_k^\triangle(p^{\xi\triangle})$. Define a mapping

$$u_r(U_k^{q_{\xi j}^\triangle}(p_{\xi\varsigma(i)})) = [0, \dots, 0, 1, 0, \dots, 0], \quad (22)$$

with one in position j and zero everywhere else, and set $u_\xi = \sum_r u_r((U_k^{q_{\xi j}^\triangle}(p_{\xi\varsigma(i)}))$, where the index r runs over all the transitions associated with the subsequence $\varsigma(i)$ such that $p_{\xi\varsigma(i)}$ converge to $p^{\xi\triangle}$. Then, by construction, a local optimum point is attained. ∎

Remark 10. The order in which the transitions are fired is given by the order of the transitions inherited from the order of the subsequence $p_{\xi\varsigma(i)}$.

Theorem 5. *Let* HDPPN *be a hierarchical decision process Petri net. The optimum global trajectory planning problem is solvable.*

Proof. Let us consider u_ξ as in the previous theorem for each $DPPN_\xi$ and set $u = \bigcup_{\xi \in \Xi} u_\xi$. Then, by construction, the optimum point is attained. ∎

Example 2. Let us extend Example 1 using a hierarchical structure.

The partially ordered HDPPN (Fig. 3) has the following specifications:

Places

P_0: claim settled

P_1: handled accident info centrally

P_2: handled accident info by authorized garage

P_3: handled accident info by adjustor

P_4: adjusted policy and made covering offer centrally

P_5: made covering offer by authorized garage

P_6: made covering offer by adjustor.

Transitions

q_1: handle accident info centrally

q_2: handle accident info by authorized garage

q_3: handle accident info by adjustor

q_4: adjust policy and make covering offer centrally

q_5: make covering offer by authorized garage

q_6: make covering offer by adjustor.
The partially ordered HDPPN (Fig. 4) has the following specifications:

Places

P_7: handled accident info centrally

P_8: verified policy covering centrally

P_9: corroborated accident details

P_{10}: evaluated damage centrally
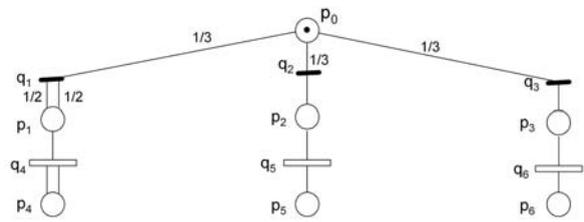
Fig. 3. BPR HDPPN.

P_{11}: got medical treatment cost

P_{12}: assessed client antecedents

P_{13}: determined accident covering centrally

P_{14}: adjusted policy and made covering offer centrally.

Transitions

q_8: verify policy covering centrally

q_9: corroborate accident details

q_{10}: evaluate damage

q_{11}: get medical treatment cost

q_{12}: assess client antecedents

q_{13}: determine accident covering centrally

q_{14}: adjust policy and make covering offer centrally.
The partially ordered HDPPN (Fig. 5) has the following specifications:

Places

P_{15}: handled accident info by authorized garage

P_{16}: verified policy covering by authorized garage

P_{17}: determined accident in range

P_{18}: send info to be handle centrally

P_{19}: got accident info by authorized garage

P_{20}: evaluated damage by authorized garage

P_{21}: made covering offer by authorized garage.

Transitions

q_{15}: handle accident info by authorized garage

q_{16}: verify policy covering by authorized garage

q_{17}: determine accident in range

q_{18}: send info to be handle centrally

q_{19}: get accident info by authorized garage

q_{20}: evaluate damage by authorized garage

q_{21}: make covering offer by authorized garage.

The partially ordered HDPPN (Fig. 6) has the following specifications:

Places

P_{22}: handled accident info by adjustor

P_{23}: verified policy covering by adjustor

P_{24}: got accident info by adjustor

P_{25}: determined accident in range

P_{26}: send info to be handle centrally

P_{27}: evaluated damage by adjustor

P_{28}: made covering offer by adjustor

Transitions

q_{22}: handle accident info by adjustor

q_{23}: verify policy covering by adjustor

q_{24}: get accident info by adjustor

q_{25}: determine accident in range

q_{26}: send info to be handle centrally

q_{27}: evaluated damage by adjustor

q_{28}: make covering offer by adjustor.

Fig. 4. HDPPN accident centrally.

From the incidence matrix of the partially ordered Petri net shown in Figs. 4–6, we conclude what follows:

(i) *Stability*

For the DPPN in Fig. 4 we have

Fig. 5. HDPPN accident by garage.

$$\begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -2 & 1 \end{bmatrix}.$$

Choosing $\Phi = [1,1,1,1/2,1,1,1/2,1/2]$, $\Phi > 0$, we obtain that $A\Phi \le 0$ concluding stability. For the DPPN in Fig. 5 it follows that

$$\begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & -2 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & -1 & 1 \end{bmatrix}.$$

Choosing $\Phi = [1,1,1,1/2,1,1,1/2]$, $\Phi > 0$, we obtain that $A\Phi \le 0$ concluding stability. For the DPPN in Fig. 6 we get

$$\begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 2 \end{bmatrix}.$$

Choosing $\Phi = [1,1,1,1/2,1,1,1/2]$, $\Phi > 0$, we obtain $A\Phi \le 0$ concluding stability.

(ii) *Optimum strategy*

Define the Lyapunov like function L in terms of the entropy $H(p_i) = -p_i \ln p_i$ as

$$L = \max_{i=1,\dots,|\alpha|} (-\alpha_i \ln \alpha_i).$$

Then

$$U_{k=0}(p_0) = 1,$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_1) &= L[\sigma_{01}(p_1) \cdot U_{k=0}^{\sigma_{01}}(p_0)] \\
&= L[1/3 \cdot 2 * 1] = \max H[0.666] = 0.270,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_2) &= L[\sigma_{02}(p_2) \cdot U_{k=0}^{\sigma_{02}}(p_0)] \\
&= L[1/3 \cdot 1] = \max H[0.333] = 0.366,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_3) &= L[\sigma_{03}(p_3) \cdot U_{k=0}^{\sigma_{03}}(p_0)] \\
&= L[1/3 \cdot 1] = \max H[0.333] = 0.366.
\end{aligned}$$

(I) The optimum strategy $\sigma^{\triangle}$ for accidents of considerable size that must be managed centrally by the assurance company is represented by

$$U_{k=0}^{\sigma_{hj}}(p_7) = 0.270,$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_8) &= L[\sigma_{7,8}(p_8) \cdot U_{k=0}^{\sigma_{7,8}}(p_7)] \\
&= L[1 \cdot 0.270] = \max H[0.270] = 0.353,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_9) &= L[\sigma_{8,9}(p_9) \cdot U_{k=0}^{\sigma_{89}}(p_8)] \\
&= L[2/5 \cdot 0.353] = \max H[0.141] = 0.276,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{10}) &= L[\sigma_{8,10}(p_{10}) \cdot U_{k=0}^{\sigma_{8,10}}(p_8)] \\
&= L[1/5 \cdot 0.353] = \max H[0.070] = 0.187,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{11}) &= L[\sigma_{8,11}(p_{11}) \cdot U_{k=0}^{\sigma_{8,11}}(p_8)] \\
&= L[2/5 \cdot 0.353] = \max H[0.141] = 0.276,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{12}) &= L[\sigma_{9,12}(p_{12}) \cdot U_{k=0}^{\sigma_{9,12}}(p_9)] \\
&= L[1 \cdot 0.276] = \max H[0.276] = 0.355,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{13}) &= L[\sigma_{10,13}(p_{13}) \cdot U_{k=0}^{\sigma_{10,13}}(p_{10}) \\
&\quad + \sigma_{11,13}(p_{13}) \cdot U_{k=0}^{\sigma_{11,13}}(p_{11})] \\
&= L[1 \cdot 0.187 + 1 \cdot 0.276] \\
&= \max H[0.463] = 0.356,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{14}) &= L[\sigma_{12,14}(p_{14}) \cdot U_{k=0}^{\sigma_{12,14}}(p_{14}) \\
&\quad + \sigma_{13,14}(p_{14}) \cdot U_{k=0}^{\sigma_{13,14}}(p_{13})] \\
&= L[1 \cdot 0.355 + 1/2 \cdot 0.356] \\
&= \max H[0.533] = 0.335.
\end{aligned}$$

The firing transition vector is

$$u = \begin{array}{|c|c|c|c|c|c|c|}
\hline
1 & 1 & 1 & 1 & 1 & 1 & 1 \\
\hline
q_8 & q_9 & q_{10} & q_{11} & q_{12} & q_{13} & q_{14} \\
\hline
\end{array}.$$

For this case the adjustor or the garage must abort the process because the accident is out of their range obtaining

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{18}) &= L[\sigma_{16,18}(p_{18}) \cdot U_{k=0}^{\sigma_{16,18}}(p_{16})] \\
&= L[4/5 \cdot 0.367] = \max H[4/5 \cdot 0.367] \\
&= 0.359,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{26}) &= L[\sigma_{24,26}(p_{26}) \cdot U_{k=0}^{\sigma_{24,26}}(p_{24})] \\
&= L[3/4 \cdot 0.367] = \max H[3/4 \cdot 0.367] \\
&= 0.355.
\end{aligned}$$

Accordingly,

$$U_{k=0}^{\sigma_{hj}}(p_{14}) < U_{k=0}^{\sigma_{hj}}(p_{26}) < U_{k=0}^{\sigma_{hj}}(p_{18}),$$

i.e., $U_{k=0}^{\sigma_{hj}}(p_{18}), U_{k=0}^{\sigma_{hj}}(p_{26})$ are more expensive than $U_{k=0}^{\sigma_{hj}}(p_{14})$.

(II) The optimum strategy $\sigma'^{\triangle}$ for small accidents that must be managed ideally by the garage is represented by

$$U_{k=0}^{\sigma_{hj}}(p_{15}) = 0.366,$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{16}) &= L[\sigma_{15,16}(p_{16}) \cdot U_{k=0}^{\sigma_{15,16}}(p_{15})] \\
&= L[1 \cdot 0.366] = \max H[0.366] = 0.367,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{17}) &= L[\sigma_{16,17}(p_{17}) \cdot U_{k=0}^{\sigma_{16,17}}(p_{16})] \\
&= L[1/5 \cdot 0.367] = \max H[0.073] = 0.191,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{19}) &= L[\sigma_{17,19}(p_{19}) \cdot U_{k=0}^{\sigma_{17,19}}(p_{17})] \\
&= L[6/8 \cdot 0.191] = \max H[0.143] = 0.278,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{20}) &= L[\sigma_{17,20}(p_{20}) \cdot U_{k=0}^{\sigma_{17,20}}(p_{17})] \\
&= L[1/8 \cdot 0.191] \\
&= \max H[1/8 \cdot 0.191] = 0.089,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{21}) &= L[\sigma_{19,21}(p_{21}) \cdot U_{k=0}^{\sigma_{19,21}}(p_{19}) \\
&\quad + \sigma_{20,21}(p_{21}) \cdot U_{k=0}^{\sigma_{20,21}}(p_{20})] \\
&= L[1 \cdot 0.278 + 1 \cdot 0.089] \\
&= \max H[0.367] = 0.367.
\end{aligned}$$

The firing transition vector is

$$u' = \begin{array}{|c|c|c|c|c|c|}
\hline
1 & 1 & 1 & 1 & 1 & 1 \\
\hline
q_{16} & q_{17} & q_{18} & q_{19} & q_{20} & q_{21} \\
\hline
\end{array}.$$

(III) The strategy σ'' for small accidents that must be managed by the adjustor is represented by

$$U_{k=0}^{\sigma_{hj}}(p_{22}) = 0.366,$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{23}) &= L[\sigma_{22,23}(p_{23}) \cdot U_{k=0}^{\sigma_{22,23}}(p_{22})] \\
&= L[1 \cdot 0.366] = \max H[1 \cdot 0.366] = 0.367,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{24}) &= L[\sigma_{23,24}(p_{24}) \cdot U_{k=0}^{\sigma_{23,24}}(p_{23})] \\
&= L[1 \cdot 0.367] = \max H[1 \cdot 0.367] = 0.367,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{25}) &= L[\sigma_{24,25}(p_{25}) \cdot U_{k=0}^{\sigma_{24,25}}(p_{24})] \\
&= L[1/4 \cdot 0.367] \\
&= \max H[1/4 \cdot 0.367] = 0.219,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{27}) &= L[\sigma_{25,27}(p_{27}) \cdot U_{k=0}^{\sigma_{25,27}}(p_{25})] \\
&= L[1 \cdot 0.219] = \max H[1 \cdot 0.219] = 0.332,
\end{aligned}$$

$$\begin{aligned}
U_{k=0}^{\sigma_{hj}}(p_{28}) &= L[\sigma_{27,28}(p_{28}) \cdot U_{k=0}^{\sigma_{27,28}}(p_{28})] \\
&= L[2 \cdot 0.332] = \max H[2 \cdot 0.332] = 0.271.
\end{aligned}$$

The firing transition vector is

$$u'' = \begin{array}{|c|c|c|c|c|c|}
\hline
1 & 1 & 1 & 1 & 1 & 1 \\
\hline
q_{22} & q_{23} & q_{24} & q_{25} & q_{27} & q_{28} \\
\hline
\end{array}.$$

Observe that, since $U_{k=0}^{\sigma_{hj}}(p_{21})$ is greater than $U_{k=0}^{\sigma_{hj}}(p_{28})$, small accidents must be handled by the garage whenever possible.

4. Conclusions and future work

A formal framework for hierarchical decision process Petri nets was presented for modular modeling of complex systems. The hierarchical refinement process generates simple nets with equivalent behavior. Consequently, the complexity of the analysis of the DPPN is considerably reduced. The analysis of the HDPPN is supported by the hierarchical structure. In this sense, several steps can be performed locally, thereby drastically reducing the size of its upper levels. An algorithm for optimum hierarchical trajectory planning used to identify the optimum point was described. The traditional notions of local and global stability in the sense of Lyapunov used to characterize the stability properties of the HDPPN were explored. Illustrative examples where the properties of the HDPPN were shown to hold were addressed.

References

Bellman, R.E. (1957). *Dynamic Programming*, Princeton University Press, Princeton, NJ.

Bouyakoub, S. and Belkhir, A. (2008). H-SMIL-Net: A hierarchical Petri net model for SMIL documents, *10-th International Conference on Computer Modeling and Simulation, Cambridge, UK*, pp. 106–111.

Buchholz, P. (1994). Hierarchical high level Petri nets for complex system analysis, *in* R. Valette(Ed.) *Application and Theory of Petri Nets*, Lecture Notes in Computer Science, Vol. 815, Springer, Zaragoza, pp. 119–138.

Clempner, J., Medel, J. and Cârsteanu, A. (2005a). Extending games with local and robust Lyapunov equilibrium and stability condition, *International Journal of Pure and Applied Mathematics* **19**(4): 441–454.

Clempner, J. (2005b). Optimizing the decision process on Petri nets via a Lyapunov-like function, *International Journal of Pure and Applied Mathematics* **19**(4): 477–494.

Clempner, J. (2005c). Colored decision process Petri nets: Modeling, analysis and stability, *International Journal of Applied Mathematics and Computer Science* **15**(3): 405–420.

Dai, X. , Li, A.J. and Meng, Z. (2009). Hierarchical Petri net modelling of reconfigurable manufacturing systems with improved net rewriting systems, *International Journal of Computer Integrated Manufacturing* **22**(2): 158–177.

Gomes, L. and Barros, J.P. (2005). Structuring and composability issues in Petri nets modeling, *IEEE Transactions on Industrial Informatics* **1**(2): 112–123.

Hammer, M. and Champy, J. (1993). *Reengineering the Corporation: A Manifesto for Business Revolution*, HarperBusiness, New York, NY.

Howard, R.A. (1960). *Dynamic Programming and Markov Processes*, MIT Press, Cambridge, MA.

Huber, P., Jensen, K. and Shapiro, R. (1990). *Hierarchies in colored Petri nets*, Lecture Notes in Computer Science Vol. 483, Springer-Verlag, pp. 313–341.

Jensen, K. (1992). *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use, Vol. 1: Basic Concepts*, EATCS Monographs in Theoretical Computer Science, Springer-Verlag, New York, NY.

Kalman, R.E. and Bertram, J.E. (1960). Control system analysis and design via the second method of Lyapunov, *Journal of Basic Engineering* **82**: 371–393.

Lakshmikantham, V., Leela, S. and Martynyuk, A.A. (1990). *Practical Stability of Nonlinear Systems,* World Scientific, Singapore.

Lakshmikantham, V., Matrosov, V.M. and Sivasundaram, S. (1991). *Vector Lyapunov Functions and Stability Analysis of Nonlinear Systems*, Kluwer Academic Publishers, Dordrecht.

Murata, T. (1989). Petri nets: Properties, analysis and applications, *Proceedings of the IEEE* **77**(4): 541–580.

Passino, K.M., Burguess, K.L. and Michel, A.N. (1995). Lagrange stability and boundedness of discrete event systems, *Journal of Discrete Event Systems: Theory and Applications* **5**(5): 383–403.

Puterman, M.L. (1994). *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Wiley, New York, NY.

Julio Clempner has ten years of experience in the field of management consulting. Currently, he specializes in applications of high technology. He has also conducted projects related to project management, software analysis, design, development, information technology strategic planning, balanced ScoreCard, evaluation of software and business process reengineering for infusing advanced computing technologies into diverse lines of businesses. Dr. Clempner is now involved in providing e-business solutions including e-business strategy, management of technology, operations strategy, project management, continuous quality improvement, and managing total quality transformation. His research interests are focused on justifying and introducing the Lyapunov equilibrium point in shortest-path decision processes and shortest-path game theory. This interest has led to several streams of research. One stream concerns the use of Markov decision processes for formalizing the previous ideas, changing Bellman's equation by a Lyapunov-like function which is a trajectory-tracking function, and also an optimal cost-to-target function for tracking the net. A second stream concerns the use of Petri nets as a language for modeling decision processes and game theory introducing colors, hierarchy, etc. Petri nets are used for process representation taking advantage of the formal semantic and the graphical display. The final stream examines the possibility to meet modal logic, decision processes and game theory.

Appendix A

HDPPN mark-dynamic properties

We will identify the mark-dynamic properties of the HDPPN as those properties related to the DPPN.

Theorem 6. *The hierarchical decision process Petri net* HDPPN *is uniformly practically stable iff there exists a* Φ *strictly positive m vector such that* $\Delta v = u^T A \Phi \leq 0$.

Proof. (Necessity) It follows directly from Proposition 1. (Sufficiency) Let us suppose by contradiction that $u^T A \Phi > 0$ with Φ fixed. From $M' = M + u^T A$, we have that $M' \Phi = M \Phi + u^T A \Phi > M \Phi$. Then, it is possible to construct an increasing sequence $M \Phi < M' \Phi < \cdots < M^n \Phi < \ldots$ which grows up without bound. Therefore, the HDPPN is not uniformly practically stable. ■

Theorem 7. *The local decision process Petri net* DPPN$_\xi \in$ HDPPN *is local uniformly practically stable iff there exists a strictly positive m vector* Φ *such that* $\Delta v = u^T A \Phi \leq 0$.

Proof. This result follows directly from the previous theorem for a local condition. ■

Remark 11. It is important to underline that the only places where the DPPN$_\xi$ and the HDPPN will be allowed to get blocked are those which correspond to equilibrium points.

Theorem 8. *The hierarchical decision process Petri net* HDPPN *is uniformly practically stable iff every local Decision Process Petri net* DPPN$_\xi \in$ HDPPN *is local uniformly practically stable.*

Proof. (Necessity) Let us suppose by contradiction that for an HDPPN we have that $u^T A \Phi > 0$ with Φ fixed. From $M' = M + u^T A$ we have that $M' \Phi = M \Phi + u^T A \Phi > M \Phi$. Then, it is possible to construct an increasing sequence $M \Phi < M' \Phi < \cdots < M^n \Phi < \ldots$ which grows up without bound. Therefore, it is possible to chose a subsequence $M_\xi \Phi_\xi < M'_\xi \Phi_\xi < \cdots < M^l_\xi \Phi < \ldots$ for $l < n$ that belongs to the DPPN$_\xi$ which grows up without bound. Therefore, DPPN$_\xi$ is not uniformly practically stable.
(Sufficiency) It follows directly from the previous theorem. ■

Appendix B

HDPPN trajectory-dynamic properties

We shall identify the trajectory-dynamic properties of the HDPPN as those related to the utility at each place of the DPPN. In this sense, we shall relate an optimum point to the best possible performance choice. Formally, we introduce the following definition.

Proposition 2. *Let* HDPPN *be a decision process Petri net and let* $p^\triangle \in P$ *a global optimum point. Then* $U(p^\triangle) \leq U(p)$, $\forall p \in P$ *such that* $p \leq_U p^\triangle$.

Proof. From the previous theorem we have $U(p^{\xi\triangle}) \leq U(p_{\xi i})$ $\forall p_{i\xi} \in P_\xi$. Then, by the definition of the HDPPN, the relationship $U(p^\triangle) \leq U(p)$ holds. ■

Proposition 3. *Let* DPPN$_\xi \in$ HDPPN *be a local decision process Petri net and let* $p^{\xi\triangle} \in P_\xi$ *be a local optimum point. Then* $U(p^{\xi\triangle}) \leq U(p_{\xi i})$, $\forall p_{\xi i} \in P_\xi$, *such that* $p_{\xi i} \leq_U p^{\xi\triangle}$.

Proof. We have that $U(p^{\xi\triangle})$ is equal to the minimum or an infimum. Therefore, $U(p^{\xi\triangle}) \leq U(p_{\xi i})$ $\forall p_{\xi i} \in P_\xi$ such that $p_{\xi i} \leq_U p^{\xi\triangle}$. ■

Theorem 9. *The hierarchical decision process Petri net* HDPPN *is global uniformly practically stable iff* $U(p_{i+1}) - U(p_i) \leq 0$.

Proof. (Necessity) Let us choose $v = Id(U(p_i))$. Then $\Delta v = U(p_{i+1}) - U(p_i) \leq 0$. By the autonomous version of Theorem 1 and Corollary 1, the HDPPN is stable.

(Sufficiency) We want to show that the HDPPN is practically stable, i.e., given $0 < \lambda < A$ we must show that $|U(p_i)| < A$. We know that $U(p_0) < \lambda$ and, since U is non-decreasing, we have $|U(p_i)| < |U(p_0)| < \lambda < A$. ■

Theorem 10. *The local decision process Petri net* DPPN$_\xi \in$ HDPPN *is local uniformly practically stable iff* $U(p_{\xi,i+1}) - U(p_{\xi i}) \leq 0$.

Proof. From the previous theorem we know the uniform practical stable condition, and since in particular, if $U(p_{\xi,i+1}) - U(p_{\xi i}) \leq 0$ the local decision process Petri net $DPPN_\xi$ is local uniformly practically stable. ■

Remark 12. It is important to note that the HDPPN is uniformly practically stable in the large, but not necessarily for every local system.

Appendix C

Convergence of the HDPPN mark-dynamic and trajectory-dynamic properties

Theorem 11. *Let* HDPPN *be a hierarchical decision process Petri net. If* $p^* \in P$ *is a global equilibrium point, then it is a global final decision point.*

Proof. Let us suppose that p^* is a global equilibrium point. We want to show that its utility has reached an infimum or a minimum. Since p^* is a global equilibrium point, by definition, it is the last place of the net and its marking cannot be modified. But this implies that the routing policy attached to the transition(s) that follows p^*

is zero (if there is such a transition(s), i.e., worst case). Therefore, its utility cannot be modified and, since the utility is a decreasing function of p_i, an infimum or a minimum is attained. Then, p^* is a global final decision point. ∎

Theorem 12. *Let* HDPPN *be a finite and non-blocking hierarchical decision process Petri net (unless $p \in P$ is a global equilibrium point). If $p_f \in P$ is a global final decision point, then it is a global equilibrium point.*

Proof. If p_f is a global final decision point, since the HDPPN is finite, there exists k such that $U_k(p_f) = C$.

Let us suppose that p_f is not a global equilibrium point.
Case 1. Then, it is not bounded. So, it is possible to increment the marks of p_f in the net. Therefore, it is possible to modify its utility. As a result, it is possible to obtain a lower utility than C.
Case 2. Then, it is not the last place in the net. So, it is possible to fire some output transition to p_f in such a way that its marking is modified. Therefore, it is possible to modify the utility over p_f. As a result, it is possible to obtain a utility lower than C. ∎

Corollary 2. *Let* HDPPN *be a finite and non-blocking hierarchical decision process Petri net (unless $p \in P$ is a global equilibrium point). Then, a global optimum point $p^\triangle \in P$ is a global equilibrium point.*

Proof. From the previous theorem we know that a global final decision point is a global equilibrium point and, since, in particular, $p^\triangle$ is global final decision point, then it is a global equilibrium point. ∎

Remark 13. The finite and non-blocking (unless $p \in P$ is a global equilibrium point) condition over the HDPPN cannot be relaxed:

1. Suppose that the HDPPN is not finite, i.e., p is in a cycle. Then the Lyapunov-like function converges to zero, as $k \to \infty$, i.e., $L(p) = 0$ but the HDPPN has no final place. Therefore, it is not a global equilibrium point.

2. Suppose that the HDPPN blocks at some place (not a global equilibrium point) $p_b \in P$. Then the Lyapunov-like function has a minimum at place p_b, let us say $L(p_b) = C$, but p_b is not a global equilibrium point, because it is not necessarily the last place of the net.

Corollary 3. *Let* HDPPN *be a non blocking hierarchical decision process Petri net (unless $p \in P$ is a global equilibrium point), and let $\sigma^\triangle$ be a optimum global strategy. Set*

$$L = \min_{i=1,\dots,|\alpha|} \{\alpha_i\}.$$

Then $U_k^\triangle(p)$ is equal to

$\sigma_{0j_m}^\triangle(p_{\varsigma(0)})$	$\sigma_{1j_m}^\triangle(p_{\varsigma(0)})$	$\dots$	$\sigma_{nj_m}^\triangle(p_{\varsigma(0)})$	$U_k(p_0)$
$\sigma_{0j_n}^\triangle(p_{\varsigma(1)})$	$\sigma_{1j_n}^\triangle(p_{\varsigma(1)})$	$\dots$	$\sigma_{nj_n}^\triangle(p_{\varsigma(1)})$	$U_k(p_1)$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\sigma_{0j_v}^\triangle(p_{\varsigma(i)})$	$\sigma_{1j_v}^\triangle(p_{\varsigma(i)})$	$\dots$	$\sigma_{nj_v}^\triangle(p_{\varsigma(i)})$	$U_k(p_i)$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

$$\underbrace{\qquad\qquad\qquad}_{\sigma^\triangle} \quad \underbrace{\quad}_{U}$$

(23)

where p is a vector whose elements are those places which belong to the optimum trajectory ω given by $p_0 \leq p_{\varsigma(1)} \leq_{U_k} p_{\varsigma(2)} \leq_{U_k} \cdots \leq_{U_k} p_{\varsigma(n)} \leq_{U_k} \dots$ which converges to $p^\triangle$.

Proof. Since at each step of the iteration $U_k^\triangle(p_i)$ is equal to one of the elements of vector α, we have that the representation that describes the dynamical utility behavior of tracking the optimum strategy $\sigma^\triangle$ is

$\sigma_{0j_m}^\triangle(p_{\varsigma(0)})$	$\sigma_{1j_m}^\triangle(p_{\varsigma(0)})$	$\dots$	$\sigma_{nj_m}^\triangle(p_{\varsigma(0)})$	$U_k(p_0)$
$\sigma_{0j_n}^\triangle(p_{\varsigma(1)})$	$\sigma_{1j_n}^\triangle(p_{\varsigma(1)})$	$\dots$	$\sigma_{nj_n}^\triangle(p_{\varsigma(1)})$	$U_k(p_1)$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\sigma_{0j_v}^\triangle(p_{\varsigma(i)})$	$\sigma_{1j_v}^\triangle(p_{\varsigma(i)})$	$\dots$	$\sigma_{nj_v}^\triangle(p_{\varsigma(i)})$	$U_k(p_i)$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

$$\underbrace{\qquad\qquad\qquad}_{\sigma^\triangle} \quad \underbrace{\quad}_{U}$$

where $j_m, j_n, \dots, j_v, \dots$ represent the indexes of the optimal routing policy, defined by the $q'_j s$. ∎

Plane symmetry involves moving all points around the plane so that their positions relative to each other remain the same, although their absolute positions may change. By analogy, let us introduce the following definition.

Definition 21. A hierarchical decision process Petri net HDPPN is said to be *symmetric* if it is possible to decompose it into some finite number (greater than 1) of local decision process Petri nets $\{\text{DPPN}_\xi\}$ in such a way that there exists a bijection ψ between all the $\{\text{DPPN}_\xi\}$ such that

$$(p,q) \in I \Leftrightarrow (\psi(p), \psi(q)) \in I$$

and

$$(q,p) \in O \Leftrightarrow (\psi(q), \psi(p)) \in O \qquad (24)$$

for all of the local decision process Petri nets.

Corollary 4. *Let* HDPPN *be a non blocking (unless p is a global equilibrium point) symmetric hierarchical decision process Petri net and let $\sigma^\triangle$ be an optimum global strategy. Set*

$$L = \min_{i=1,\dots,|\alpha|} \{\alpha_i\}.$$

Then

$$\sigma^\triangle U \leq \sigma U, \qquad \forall \sigma, \sigma^\triangle,$$

where the σ and $\sigma^\triangle$ are represented by a matrix and U is represented by a vector.

Proof.   From the previous corollary, thanks to the symmetric property, we obtain

$$\sigma^{\triangle} U \leq \sigma U, \quad \forall \sigma, \sigma^{\triangle}.$$

■