IDEA GROUP PUBLISHING
701 E. Chocolate Avenue, Hershey PA 17033-1117, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.idea-group.com
ITP4254

Petri Nets with Clocks and Applications To the Model of Processes

Gabriel Vilallonga, Daniel Riesco, Germán Montejano and Roberto Uzal
Universidad Nacional de San Luis, Departamento de Informática, Argentina
Tel: + 54 (0) 2652 – 424027 ext. 251, Fax: + 54 (0) 2652 – 430059, {gvilallo|driesco|gmonte|ruzal}@unsl.edu.ar

L. Felice
Universidad Nacional del Centro de la Provincia de Buenos Aires, INTIA
Departamento de Computación y Sistemas Tandil – Argentina, lfelice@exa.unicen.edu.ar

ABSTRACT

Petri Nets are tools for the analysis and design of concurrent systems. There is a formal theory, which supports Petri Nets. We propose Petri Nets with Clocks (PNwC) which has a high expressive power in the concurrent and asynchronous process modeling and gives the possibility to model real time systems.

The Petri Nets with Clocks are useful to model systems with temporal requirements via specification of clocks, using temporal invariants for the places and temporal conditions in the transitions. In PNwC we force to specifies invariants in places and transitions. Using invariants in places allows the specifications of hard deadlines constrains (upper bound constrains): when a deadline is reached the progress of time is blocked by the invariant and the action becomes urgent. An algorithm for the analysis of a PNwC has been proposed in [1], [2]. This algorithm is oriented to the verification and correction of errors in the modeling of the time variable. The algorithm generates information about temporal unreachable states and process deadlocks with temporal blocks. Also, it corrects places invariants and transitions conditions.

In this paper we give an approach to the use of PNwC for the Business Processes Modeling, allowing to study the models through a qualitative analysis.

INTRODUCTION

Petri Nets (PN) are tools for system studying and modeling. PN theory allows system modeling, obtaining a mathematical representation of the system [3]. The analysis of the corresponding PN gives important information about the system structure and its dynamic behavior. This information can be used to evaluate the system model and for improving and changing the model.

When studying different types of systems we are confronted with some systems in which time plays an important role: communications protocols, real time systems, business processes, etc. An important part of these system's requirements that must be fulfilled are the temporal requirements. The growing complexity and critical nature of these systems have motivated the search for verification methods [4, 6].

The PN models use an efficient method of analysis of networks' behavior. In some cases the networks are analyzed by means of simulations more than by the generation of the state space. This does not guarantee that the states with very low probability happen in long runs. The analysis of these states with low probability can be object of a serious analysis, for example, the lost one of a message in a communication network. The systematic analysis of the state space takes all the events in account, even those improbable ones.

To the power of structure analysis and system behavior that PN have, we have added a structure of an algorithm that allows for temporal analysis of the extended PN [1].

Petri Nets with Clocks (PNwC), proposed here has a high expressive power in the concurrent and asynchronous process modeling and give the possibility to model real time systems. PNwC includes additional temporal elements, clocks, which are not taken into consideration in the literature concerning the extensions of PN with time [6, 16, 17, 18].

We propose the use of PNwC to formalize Business Processes Modeling, allowing to study the models through a quantitative and qualitative analysis.

TIMING IN PNS

The time is introduced in Petri nets to model the interaction among several activities considering their starting and completion time.

Several approaches are possible for the introduction of temporal specification in PN models, time associated with places: Timed Place PN (TPPN), time associated with tokens; time associated with arcs and time associated with transition: Timed Transition PN (TTPN).

Considering TTPN two types of firing policies can be assumed:

1. three-phase firing:
   - tokens are consumed from inputs place when the transition is enabled,
   - the delay elapses, and
   - tokens are generated in outputs places.
2. atomic firing:
   - tokens remain in input place for the transition delay, they are consumed from input place, and
   - generated in output places when the transition fires.

The TTPN can be classified according to the functions of distributions that are applied at the times of fires. On one them are those that possess function of distribution discrete called D-timed nets. The nets that has function of exponential distribution are called M-Timed Nets. The Stochastic Petri Nets are those nets that possess negative exponential distribution .If to the SPN adds instantaneous transitions its are denominated Generalized SPN.

In the case of discrete firing time distribution the resulting class of nets is called D-Timed nets since the firing times are deterministic or constant, there is a nonnegative number assigned to each transition of a net which determines the duration of its firings [21].

Timed nets with exponentially distributed firing times are called M-timed nets. The rates of exponential distribution are assigned to transition of a net. For the memoryless property of the exponential distribution, the history of firing is immaterial, and this simplifies the description of states and states transitions [22].

The Stochastic Petri Nets (SPN) has atomic firing in witch all transitions delays are random variable with negative exponential distributions. The dynamics behavior of a SPN is described through a stochastic process. The SPN techniques provide a performance evaluation approach based on a formal description [26].

The Generalized Stochastic Petri Nets (GSPN) have two classes of transitions, timed transitions, like in SPNs, and immediate transitions, whose delays are deterministically zero. The immediate transitions have been introduced in the model to account for instantaneous actions, to implement specific modeling features and to account for time scale differences [23, 24].

The D-timed nets and M-timed nets use different mechanism of firing that in stochastic nets, consequently have rather different properties [21, 22, 23].

One class of Timed Tokens PN are the timed Coloured Petri (CP) nets. The timed CP-net is an extension of CP-net. In a timed CP-nets, each token carries a colour and a time stamp. The time stamps are specified by expressions and its can depend upon colour values and can be specified by probability distributions. These features allow specify fixes delay, interval delays and exponentials delays.

The PNwC, descripted in this paper, are TPPN, since the time is associated in the places. This time is specified by means of a place invariant using clocks. The invariants represent deadline. When the time reach that deadline some transition, which has this place like an input place must be executed.

A transition is enabled when the input places have the necessary amount of tokens and, also, the values of the system's clocks satisfies the invariants of each input place. The firing of a transition takes place if this is enabled and, considering the system's clocks, has to satisfy the condition associated to this one. This new firing causes the creation of tokens in outputs place and the update of the system's clocks.

PETRI NETS WITH CLOCKS

PN are limited in their modeling and design power when used for systems where time is part of the system's specification. Timed Graphs (TG) [5, 12, 13, 14] on the other hand are a useful tool to specify system time constraints.

The power inherent of TG and PN has motivated us to extend PN theory with temporal requisites using TG. To the power of structure analysis and system behavior that PN have, we have added a structure of an algorithm that allows the temporal analysis of the extended PN [2].

The definition of PNwC is presented here. By extending PN using TG we profit from the advantage in modeling systems with asynchronous and concurrent behavior (PN) and the possibilities of formalizing systems with temporal requisites.

Definition 1. *A Clock* **x**

A Clock **x** is a positive real variable, i.e., $\mathbf{x} \in \mathbf{R}^{+0} = \{ z / z \in \mathbf{R}^{+} \vee z = 0\}$; where $\mathbf{R}^{+}$ are all real positives.

Definition 2. *Set of all Clocks X*

$\mathbf{X} = \{x_1, \ldots, x_k\}$ is the *set of all Clocks*.

Definition 3. *Restricted predicates* $\mathbf{\Omega_X}$

$\mathbf{\Omega_X}$ is a set of restricted predicates on the places defined as a Boolean combination of atoms that take the form **x # c**, where $\mathbf{x} \in \mathbf{X}$, # is a binary relation on the set $\{<, >, =, \geq, \leq\}$ and $\mathbf{c} \in \mathbf{R}^{+0}$.

Definition 4. *Restricted predicates* $\mathbf{\Psi_X}$

$\mathbf{\Psi_X}$ is a set of restricted predicates on the transitions defined as a Boolean combination of atoms that take the form **c # x # c'** or the form **x # c'** where $\mathbf{x} \in \mathbf{X}$, # is a binary relation on the set $\{<, =, \leq\}$, $\mathbf{c}, \mathbf{c'} \in \mathbf{R}^{+0}$. Every clock in $\mathbf{\Psi_x}$ must have an upper limit **c'**.

Definition 5. *Valuation* **VaL**

VaL is a set of all vectors of dimension **k**, where **k** is the cardinality of set **X**, and each element of the vector belong to $\mathbf{R}^{+0}$,

$\mathbf{VaL} = \{v / v = (v_1, \ldots, v_k) \wedge k = | X | \wedge \ \ j, 1 \leq j \leq k \cdot v_j \in \mathbf{R}^{+0}\}$

Definition 6. *A Petri Net with Clocks* **PNwC**

A Petri Net with Clocks is a PN extended based on TG, with a finite set of clocks whose values are incremented uniformly with time. The restrictions associated with the system are expressed by invariants on places and the association of an enabling condition with each transition. A clock can be reset in each transition. Also, the firing of a transition shall be an instantaneous action that does not consume time. Time runs only at places and for no more than what is established by the associated invariant. Formally the structure of a PNwC is a t-uple:

$$\mathbf{PNwC = < P, T, I, O, X, Inv, C, A >} \text{ where:}$$

- $\mathbf{P} = \{ p_1, p_1, \ldots, p_n \}$ is a finite set of *places*, $\mathbf{n \geq 0}$.
- $\mathbf{T} = \{ t_1, t_1, \ldots, t_m \}$ is a finite set of *transitions*, $\mathbf{m \geq 0}$, $\mathbf{P \cap T = \varnothing}$.
- $\mathbf{I: T \rightarrow P^{\infty}}$, is the input function, a mapping from transitions to bags of places.
- $\mathbf{O: T \rightarrow P^{\infty}}$, is the output function, a mapping from transitions to bags of places.
- **X** is in **def 2**,
- $\mathbf{Inv: P \rightarrow \Omega}$, associates to each place $\mathbf{p_i \in P}$, a restricted predicate $\mathbf{\Omega \in \Omega_X}$ (**def 3**) called **place invariant**.
- $\mathbf{C: T \rightarrow \Psi}$, associates to each transition $\mathbf{t \in T}$, a restricted predicate $\mathbf{\Psi \in \Psi_X}$ (**def 4**) called **transition condition.**
- $\mathbf{A: T \rightarrow w}$, set of clocks of the transition that are initialized to zero $\mathbf{w \subseteq X}$.

Definition 7. *Affectation* $\boldsymbol{\alpha}$

An affectation $\boldsymbol{\alpha}$ is a function $\boldsymbol{\alpha}$**: VaL x T → VaL**

$\boldsymbol{\alpha}(\mathbf{v, t}) = \mathbf{v'}$ iff $\ \ x_i \in \mathbf{X} \cdot [x_i \in \mathbf{A(t)} \Rightarrow \mathbf{v'(i) = 0} \wedge x_i \notin \mathbf{A(t)} \Rightarrow \mathbf{v'(i) = v(i)}]$

Definition 8. Marked Timed Petri Net **MPNwC**

A Marked PNwC is defined as **MPNwC = < P, T, I, O, X, Inv, C, A, μ >** where **P, T, I, O, X, Inv, C** and **A** are in **def 6**, and $\boldsymbol{\mu} \in \mathbf{M}$.

The marking $\boldsymbol{\mu}$ is as an n-vector $\boldsymbol{\mu} = (\mu_1, \mu_2 \ldots \mu_n)$, with $\mathbf{n = | P |}$ and $\mu_i \in \mathbf{Nat_0}$ with $\mathbf{1 \leq i \leq n}$.

The set of all marking **M** is the set of all vectors of dimension **n**, $\mathbf{M \subseteq Nat_0^{n.}}$,

$$\mathbf{M} = \{\mu = (\mu_1, \ldots, \mu_n) \wedge n = | P | \wedge \ \ i, 1 \leq i \leq n \cdot \mu_i \in \mathbf{Nat_0}\}$$

Definition 9. *Invariant of the Marking* **InvM(μ)**

Invariant of the Marking **InvM(μ)** is the conjunction of the invariants of the places where the number of tokens is greater than zero.

$$\mathbf{InvM(\mu)} \text{ iff } \mathbf{\Lambda\ Inv(p_i)} \text{ where } \mathbf{p_i \in P \wedge \mu(p_i) > 0}$$

Definition 10. *A Predicate applied to Valuation* $\mathbf{\Phi}$ **[v]**

$\mathbf{\Phi}$ **[v]** iff $((x_i \# c)$ is in $\mathbf{\Phi}) \Rightarrow \mathbf{v(i) \# c}$ holds,

where $x_i \in \mathbf{X}$, # is a binary relation on the set $\{<, >, =, \geq, \leq\}$, $\mathbf{c \in R^{+0}}$.

Definition 11. *A State* **q**

A state of a **PNwC** is a pair $\mathbf{q = (\mu, v)}$, $\boldsymbol{\mu} \in \mathbf{M}$ and $\mathbf{v} \in \mathbf{VaL}$, where the valuation **v** of the clocks satisfy the invariants of the net's places, i.e., **InvM(μ)[v]** holds.

Definition 12. *The set of all states* **Q**

The set of all possible states of a PNwC is represented by $\mathbf{Q \subseteq M\ x\ VaL}$, such that:

$$\mathbf{Q = \{(\mu, v) \mid \mu \in M \wedge InvM(\mu)[v]} \text{ holds } \}$$

EXECUTION OF MPNWC

The execution of the MPNwC is done using the successor marking and the system state changes.

Definition 13. *Enabled Transition in MPNwC* **E(t, q)**

Let $\mathbf{q = (\mu, v)}$ be a possible state of a PNwC, where $\boldsymbol{\mu}$ is a marking and **v** the valuation of the clocks. In **q**, the valuation of the clocks satisfies the associated invariants to each place in the marking by **def 11**.

A transition $\mathbf{t} \in \mathbf{T}$ in a **MPNwC** is enabled $\mathbf{E(t, q)}$ in the state $\mathbf{q} = (\boldsymbol{\mu}, \mathbf{v})$,

$\mathbf{E(t, q)}$ iff $\quad \mathbf{p_i} \in \mathbf{I(t)} \cdot \boldsymbol{\mu}(\mathbf{p_i}) \geq \#(\mathbf{p_i}, \mathbf{I}(\mathbf{p_i})) \wedge \mathbf{C(t)[v]}$ holds.
/* **defs 6, 10** */

Definition 14: *System State Changing* $\rightarrow$

The System State Changing is represented by the following expression:

$$\mathbf{SC} = \langle \mathbf{Q}, \rightarrow \rangle$$

where **Q** is the set of all states (**def 19**).

The changing relation $\rightarrow \subseteq \mathbf{Q} \times (\mathbf{T} \cup \mathbf{R}^{+}) \times \mathbf{Q}$ has two types of changing: temporals and instantaneous. The notation is $\mathbf{q} \rightarrow^{time} \mathbf{q'}$ for temporal changing and $\mathbf{q} \rightarrow^{t} \mathbf{q'}$ for instantaneous changing, where $\mathbf{q}, \mathbf{q'} \in \mathbf{Q}$ and **time**, $\mathbf{t} \in \mathbf{T} \cup \mathbf{R}^{+}$.

Definition 15 : *Temporal State Changing* $\rightarrow^{time}$

Temporal state changing represents the elapse of time by a changing labeled **time** from the state $(\boldsymbol{\mu}, \mathbf{v})$ to $(\boldsymbol{\mu}, \mathbf{v+time})$, $(\boldsymbol{\mu}, \mathbf{v}) \in \mathbf{Q}$, $\mathbf{time} \in \mathbf{R}^{+0}$.

$(\boldsymbol{\mu}, \mathbf{v}) \rightarrow^{time} (\boldsymbol{\mu}, \mathbf{v+time})$ iff $\quad \mathbf{y}, \mathbf{y} \in \mathbf{R}^{+0} \cdot 0 \leq \mathbf{y} \leq \mathbf{time} \Rightarrow \mathbf{InvM}(\boldsymbol{\mu})[\mathbf{v+y}]$ holds /* **def 9** */

Definition 16: *Instantaneous or discrete State Changing (by transition)* $\rightarrow^{t}$

An instantaneous state changing is given by the execution of a transition $\mathbf{t} \in \mathbf{T}$, where the changing is labeled **t**, from the state $(\boldsymbol{\mu}, \mathbf{v})$ to state $(\boldsymbol{\mu}', \mathbf{v}')$ as **def 14.**

$(\boldsymbol{\mu}, \mathbf{v}) \rightarrow^{t} (\boldsymbol{\mu}', \mathbf{v}')$ iff $\mathbf{E}(\mathbf{t}, (\boldsymbol{\mu}, \mathbf{v})) \wedge \boldsymbol{\alpha}(\mathbf{v}, \mathbf{t}) = \mathbf{v}'$ /* **defs 7, 14** */

ANALYSIS METHOD

Analysis of a PNwC is based on the exploration of the symbolic execution of the system being analyzed. This execution is studied using the PN reachability graph. When the reachability graph is constructed each graph node represents a symbolic state. The following definitions are necessary for the formalizing of the process model that employs PNwC. The analysis algorithm allows model checking to detect errors in the structure as well as modeling of the time variable. In contrast with the algorithm described in [20] this algorithm allows the analysis of concurrent and asynchronous process as well as the time variable.

Symbolic Execution of PNwC

A symbolic state of a PNwC is a generalization of the concept of state. The state is a pair $(\boldsymbol{\mu}, \mathbf{v})$ as in **def 11.** The symbolic state is a pair $\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle$ as in **def 17.** The difference between state and symbolic state is that state uses a valuation and the symbolic state uses a predicate $\boldsymbol{\Omega}$.

Definition 17. *Symbolic States* $\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle$

The analysis verification technique that follows are based on symbolic execution for a given set of symbolic states [14].

A symbolic state of a PNwC is a pair $\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle$ where $\boldsymbol{\mu} \in \mathbf{M}$, and $\boldsymbol{\Omega} \in \boldsymbol{\Omega}_{X}$.

Definition 18: *Set of all States* $\|[\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle]\|$

$\|[\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle]\|$ is the set of all states $(\boldsymbol{\mu}, \mathbf{v})$ of the symbolic state $\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle$ such that $\boldsymbol{\Omega}[\mathbf{v}]$ holds. /* **defs 10, 11** */

Definition 19: *Temporal Successors of Symbolic States* $\langle \boldsymbol{\mu}, \mathbf{succ}_{time}{}^{\boldsymbol{\mu}}(\boldsymbol{\Omega}) \rangle$

The temporal successors of symbolic state that are in $\|[\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle]\|$ are characterized by the symbolic state $\langle \boldsymbol{\mu}, \mathbf{succ}_{time}{}^{\boldsymbol{\mu}}(\boldsymbol{\Omega}) \rangle$, where $(\boldsymbol{\mu}, \mathbf{v}) \in \|[\langle \boldsymbol{\mu}, \mathbf{succ}_{time}{}^{\boldsymbol{\mu}}(\boldsymbol{\Omega}) \rangle]\|$ iff $(\boldsymbol{\mu}, \mathbf{v\text{-}time}) \rightarrow^{time} (\boldsymbol{\mu}, \mathbf{v}) \wedge \boldsymbol{\Omega}[\mathbf{v\text{-}time}]$ holds. /* **defs 15, 10** */

Definition 20: *Successors of Symbolic States by a Transition* $\langle \boldsymbol{\mu}', \mathbf{succ}_{t}(\boldsymbol{\Omega}) \rangle$

The successors of symbolic state by a transition **t** from $\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle$ to $\langle \boldsymbol{\mu}', \mathbf{succ}_{t}(\boldsymbol{\Omega}) \rangle$ are:

$(\boldsymbol{\mu}', \mathbf{v}') \in \|[\langle \boldsymbol{\mu}', \mathbf{succ}_{t}(\boldsymbol{\Omega}) \rangle]\|$ iff $\exists\, (\boldsymbol{\mu}, \mathbf{v}) \in \|[\langle \boldsymbol{\mu}, \boldsymbol{\Omega} \rangle]\| \cdot (\boldsymbol{\mu}, \mathbf{v}) \rightarrow^{t} (\boldsymbol{\mu}', \mathbf{v}')$
/* **def 16** */

Definition 21: *Symbolic Sequence* $\langle \boldsymbol{\mu}_0, \boldsymbol{\Omega}_0 \rangle\, \mathbf{t}_0 \langle \boldsymbol{\mu}_1, \boldsymbol{\Omega}_1 \rangle\, \mathbf{t}_1 \langle \boldsymbol{\mu}_2, \boldsymbol{\Omega}_2 \rangle \ldots$

The analysis of PNwC is done exploring sequences of symbolic states. We define the symbolic sequences as follows.

A symbolic sequence with start $\langle \boldsymbol{\mu}_0, \boldsymbol{\Omega}_0 \rangle$ is a succession of symbolic states:

$$\langle \boldsymbol{\mu}_0, \boldsymbol{\Omega}_0 \rangle\, \mathbf{t}_0 \langle \boldsymbol{\mu}_1, \boldsymbol{\Omega}_1 \rangle\, \mathbf{t}_1 \langle \boldsymbol{\mu}_2, \boldsymbol{\Omega}_2 \rangle \ldots$$

where $\boldsymbol{\mu}_0$ is the initial marking, $\boldsymbol{\Omega}_0$ are the all clocks initialized in zero and

$$\langle \boldsymbol{\mu}_{i+1}, \boldsymbol{\Omega}_{i+1} \rangle = \langle \mathbf{succM}(\boldsymbol{\mu}_i, \mathbf{t}), \mathbf{succ}_{ti}(\mathbf{succ}_{time}{}^{\boldsymbol{\mu}\, i}(\boldsymbol{\Omega}_i)) \rangle$$

Definition 22: *Symbolic Reachability Graph*

The Reachability Graph (RG) of a PNwC is composed of sequences which are constructed from the initial state $\langle \boldsymbol{\mu}_0, \boldsymbol{\Omega}_0 \rangle$. Let $\langle \boldsymbol{\mu}_i, \boldsymbol{\Omega}_i \rangle$ be any symbolic state, if $\quad \mathbf{t}_{jr} \in \mathbf{T}, 1 \leq \mathbf{r} \leq \mathbf{k} \cdot \mathbf{E}(\boldsymbol{\mu}_i, \mathbf{t}_{jr})$ then the symbolic RG will be composed of the symbolic sequences with the following characteristics:

$$\langle \boldsymbol{\mu}_i, \boldsymbol{\Omega}_i \rangle\, \mathbf{t}_1 \langle \boldsymbol{\mu}_{j1}, \boldsymbol{\Omega}_{j1} \rangle$$
$$\ldots$$
$$\langle \boldsymbol{\mu}_i, \boldsymbol{\Omega}_i \rangle\, \mathbf{t}_k \langle \boldsymbol{\mu}_{jk}, \boldsymbol{\Omega}_{jk} \rangle$$

where **k** is the number of enabled transitions in $\langle \boldsymbol{\mu}_i, \boldsymbol{\Omega}_i \rangle$, and where **k** new sequences are generated that have the particularity that up to the symbolic state $\langle \boldsymbol{\mu}_i, \boldsymbol{\Omega}_i \rangle$ of the sequences they are all equal, and from then on they might be different.

MODELLED TECHNIQUE OF BUSINESS PROCESSES

Business Process Reengineering has emerged as a modeling and analysis methodology for making the organization of an enterprise more efficient [7, 8, 9].

The Business Process Modeling (BPM) is described as a set of partially ordered steps to reach a business goal [10, 11, 19].

A component of a process is called a process element, representing activities without internal substructures.

An agent is an actor (human or machine) who performs one or more process elements. A coherent set of process elements to be assigned to an agent as a unit of functional responsibility is also called role, and the product created or modified by the execution of a process element is referred to as work. A BPM is an abstract description of a business process constituted by their respective process elements, together with their assignment to agent. The assignment describes the dependencies, coordination and interaction among agents. The most natural way of interaction among agents is the flow of work. Like the assignment of process elements to the responsibility of agent, a BPM must also appropriately describe the input and output relation of work for every process element. The BPM describe work scheduling and management strategy applied by an agent to execute the assigned process element. This is essential especially for agent acting in several roles (involved in more than one process). For a BPM to be adequate with respect to dependencies of processes as determined by their appearance and duration, it must reflect time intervals and duration of process element executions. Consequently, the modeling formalism has to provide sufficient expressiveness to characterize the time dynamics of the system.

The modeling power of PNwC meets those requirements, since they allow formalizing time in the places as well as the transitions.

This work presents an approach to obtain the properties of BPM using PNwC.

BUSINESS PROCESS MODELING WITH PNWC

Our approach tends to the application of the PNwC to the BPM, is by this that a component of a process, represent activities without internal substructures. The set of places, corresponds to "work deposits", stores for pieces of work represented by tokens. In these places, it is where an activity is made, they have associate a time for its concretion, specified by means of a temporary restriction. By this is that our extension adjusts to the business modelled one, fitting itself in the Timed Place PN.

In [15] the timed Petri nets has been vital for the preanalysis of the structural properties in the decomposition of BPM for improve a distributed simulation methods to accelerate the evaluation. Here we present a method for model analysis and verification that goes beyond simulation.

In a correspondence among PNwC and business process concepts, the static structure of the business process topology is expressed in a PNwC graph, whereas the behavioral aspect is encoded in the initial marking and the transition firing rules of the PNwC. The relationship between PNwC and BPM respect of theirs main concepts can more formally be summarized as follows: A BPM is a MPNwC **/* def. 8 */**

BPM = **<S, X, Inv, C, A, μ_0>**, where:

- S is the structure of standard PN, **S = <P, T, I, O>**. P is the set of places, corresponds to "work deposits", stores for pieces of work represented by tokens.

T is the set of transitions T = {t1, t2, …} stands for process elements, i.e. abstract, atomic processes to be conducted upon invocation (enabling). Each transition has associated a condition of firing. Transition t_i don't consume time for their enabling. The transition t_i is also refer as process elements. Work is seen as a product created or modified by the enactment of a process element, i.e. the firing of a transition. A process element may require a resource or a set of resources for their execution; it may or may not consume time.

The functions I and O model the flow or work among process elements.

- **X** is the set of clocks **/* def 2** */ represent all the variables what models the time in the systems. Each element of X can model time restriction in invariants and conditions.
- **Inv: P → Ω**, /* **def 6** */, the work deposit's invariant, that stand how many time may stay a work, represented by tokens, in a work deposit.
- **C: T → Ψ**, /* **def 6** */, the process element's condition, that stands the enabling interval time. Establishes what is the necessary time for a job to pass to another deposit.
- **A: T → w**, **/* def 6 */**, the process element's affectation, stand the new values of clocks, a new initialization of the set of clocks. The affectation reflects the end of a task mensuration and the beginning of another one. Clocks are reset to be able to time a new set of activities.
- **μ_0** is the initial assignment of works to work deposits.

DYNAMIC BEHAVIOR

Analogously, we describe the rules for the definition of the dynamic behavior of BPM with PNwC.

Process element enabling rule: A process element t_i is enabled, **E(t_i, q),** in some state **q = (μ, v)**, iff each of its input deposits contains a "sufficient" amount of work, and the temporal restriction associated to the process holds **/*def 13*/.**

Process element execution rule: When a process element $t_i \in$ T executes in the state **(μ, v)** produce a state changing to **(μ', v')**, **/*def 16*/.** That means changes in the marking removing a certain amount of work from its input deposits and creating a certain amount into its output deposits, **μ'**, and the new valuation **v'**.

A business process, or equivalently a process, finally appears as a set of partially ordered process elements, described as a spatial region of a PNwC graph.

CONCLUSIONS

We have presented here a formalism to model systems with time restrictions. We believe that is very important the verification of the system's integrity regarding its structure and its temporal specifications. Deadlock detection and temporal blockings as well as the consistency of the restrictions have been proposed in previous work.

PNwC has a high expressive power in the concurrent and asynchronous process modeling and have the possibility to model real time systems. PNwC, developed here, includes additional temporal elements, clocks, which are not taken into consideration in the literature concerning the extensions of PN with time.

PN theory has been used for developing the PNwC formalism, including the analysis. On the other hand, TG theory has been used to include temporal aspects of PNwC.

Modeling and analysis of business organization is required for the purpose of redesigning an enterprise's business process to make the organization more effective (Business Process Reengineering), as well as for the purpose of establishing advanced coordination technology in the organization. Here is presented an abstract frame of business process systems and a modeling formalism based on PNwC.

Our work proposes the application of the PNwC for the BPM, covering the deficit of modeling the time variable in the places, TPPN, only contemplated by theoretical aspects. This work allows modeling business with variable time. This formalism allows the qualitative analysis of BPM.

REFERENCES

[1] G. Montejano, D. Riesco, G. Vilallonga, A. Dasso, L. Favre "An Analysis Algorithm for Timed Petri Nets" Software Engineering (SE'98). International Association of Science and Technology for Development. Las Vegas, USA. 1998.

[2] D. Riesco, G. Montejano, G. Vilallonga, A. Dasso, R. Uzal. "Underlying Formalism for a Timed Petri Net Algorithm", IASTED International Conference Software Engineering and Applications, Octubre de 1999, Scottsdale, Arizona, USA.

[3] J. Peterson, Petri Net Theory and the Modelling of Systems. Prentice Hall, Englewood Cliffs, New Jersey, 1981.

[4] Alur, Courcoubetis and Dill. Model checking for real-time systems, in "Proceedings of the Firth Symposium on Logic in Computer Science", p. 414-425, IEEE C. S. P., Los Alamitos, CA.

[5] Sifakis and Yovine. Compositional Specifications of Timed Systems. STACS '96.

[6] Ghezzi, Mandrioli, Morascas and Pezze -1991 "A unified high-level Petri Net model for time critical system". IEEE Transactions on Software Ingeniering 17(2), 160-172.

[7] Hammer, M, "Beyond Reengineering: How the process-centered organization is changing out work and our lives", Harper Collins, 1996.

[8] Hammer, M. and Champy, J. "Reengineering the Corporation: A Manifesto for Business Revolution", Harper Collins Publishing, Inc., 1993

[9] Jacobson, I. "Objectifying Business Process Reengineering", Addison Wesley, 1996

[10] Hollingsworth, D. Workflow Management Coalition. The Workflow Reference Model. Document Number TC00-1003. Issue 1.1. Jan-95. http://www.wfmc.org

[11] Jacobson, I, Booch, G., Rumbaugh, J, "The Unified Software Development Process", Addison-Wesley, 1999.

[12] Olivero. Modélisation et analyse de systémes temporisés et hybrides. Thése, Institut National Polytechnique de Grenoble, France, September 1994.

[13] Yovine, S. Méthodes et outils pour la vérification symbolique de systémes temporises, Phd thesis, Institut National Polytechnique de Grenoble, France, May 1993.

[14] Daws, A. Olivero, S. Tripakis, S. Yovine. The tool Kronos. VERIMAG associated with IMAG. 1996.

[15] A. Ferscha. "Optimistic Distributed Execution of Business Process Models". 31st Hawaii International Conference on System Sciences (HICSS'98), published by the IEEE.

[16] M. Zuberek. Timed Petri nets: Definitions, Properties, and applications. Microelectronics and Reliability, 31(4):627-644, 1991.
[17] Ghezzi, Mandrioli, Morascas and Pezze -1989 "A general way to put time in Petri nets". Proceeding 5th International Workshop on Software Specification and Design, Pittsburgh, May 89, pages 60-67, IEEE Computer Society Press, 1989.
[18] Buy and Sloan. Analysis of Real-Time Programs with Simple Time Petri Nets. ACM 1994.
[19] Aguilar, Marc; Rautert, Tankred; Pater, Alexander; "Business Process Simulation: A Fundamental Step Supporting Process Centered Management", Proceeding of the 1999 IEEE, Winter Simulation Conference.
[20] Henzinger, Nicollin, Sifakis and Yovine. "Symbolic model checking for real-time systems", Information and Computation, 111(2):193-244, 1994
[21] W. M. Zubereck. "M-timed Petri nets, priorities, preemptions, and performance evaluation of systems"; in: **Advances in Petri Nets 1985** (Lecture Notes in Computer Science 222), pp.478-498, Springer Verlag 1986.
[22] W. M. Zubereck. "D-timed Petri nets and modelling of timeouts and protocols"; **Transactions of the Society for Computer Simulation**, vol.4, no.4, pp.331-357, 1987.
[23] M. Ajmone Marsan, G. Balbo, G. Conte. "A class of generalizad stochastic Petri nets for the perfomance evaluation of multiprocessor systems", ACM Trans. on Computer Systems, vol. 2, no. 2, pp. 93-122, 1984
[24] G. Chiola, M. Ajmone Marsan, G. Balbo, G. Conte. "Generalized Stochastic Petri Nets: a Definition at the Net Level and its Implications", IEEE Transactions on Software Engineering, Vol. 19, No. 2, pp. 89-107, February 1993
[25] M. Ajmone Marsan, G. Balbo, G. Chiola, G. Conte, S. Donatelli, and G. Franceschinis. "An Introduction to Generalized Stochastic Petri Nets", *International Journal of Microelectronics and Reliability* vol.31, no.4, pp.699-725, (1991).
[26] M.K Molloy. "Perfomance analysis using stochastic Petri Nets"; IEEE Trans. on Computers, vol.31, no.9, pp.913-917, 1982.

Related Content

Simulation to Improve Management of Perishable and Substitutable Inventory

Duong Nguyen Khanh Linh and Lincoln C. Wood (2015). *Encyclopedia of Information Science and Technology, Third Edition (pp. 915-922).*

www.irma-international.org/chapter/simulation-to-improve-management-of-perishable-and-substitutable-inventory/112484

Computer-Aided Ceramic Design: Its Viability for Building User-Centered Design

Folasayo Enoch Olalere and Ab Aziz Bin Shuaib (2014). *Contemporary Advancements in Information Technology Development in Dynamic Environments (pp. 266-286).*

www.irma-international.org/chapter/computer-aided-ceramic-design/111615

An Artificial Intelligent Centered Object Inspection System Using Crucial Images

Santosh Kumar Sahoo and B. B. Choudhury (2018). *International Journal of Rough Sets and Data Analysis (pp. 44-57).*

www.irma-international.org/article/an-artificial-intelligent-centered-object-inspection-system-using-crucial-images/190890

Up-to-Date Summary of Semantic-Based Visual Information Retrieval

Yu-Jin Zhang (2015). *Encyclopedia of Information Science and Technology, Third Edition (pp. 1294-1303).*

www.irma-international.org/chapter/up-to-date-summary-of-semantic-based-visual-information-retrieval/112527