RESOURCE MANAGEMENT FOR ONE CLASS OF WORKFLOW PROCESSES
A Stochastic Petri Net Based Approach

Oleg Gorbunov and Valery Sokolov
Department of Computer Science, Yaroslavl State University, Yaroslavl, Russia
{*falcon, sokolov*}*@uniyar.ac.ru*

Keywords: Workflow processes, Petri nets, Stochastic Petri nets, Stochastic Resource-Constrained Petri nets, Management of resources.

Abstract: In the paper the new approach for modeling workflow processes have been proposed. This approach is based on the special class of stochastic Petri nets and allows to model resources that is needed for tasks execution. Such models are well suited both for qualitative and quantitative analysis of workflow processes. In the paper the function reflecting the cost of waiting the task execution due to the lack of resources have been introduced. The problem of minimizing of this function have been stated. The decision approach for this problem have been introduced. This approach manages resources by means of priorities. In general, optimal priorities may be found during simulation.

1 INTRODUCTION

One of the most popular and relatively recent methodology of enterprise management is the management of workflow processes. This methodology deals with the models of workflow processes. Many languages and formalisms are proposed for modeling them. One of such formalisms is Petri nets which have been proven to be a successful formalism for this reason (van der Aalst, 1998; van der Aalst and van Hee, 2002).

Many special classes of Petri nets were proposed for modeling workflow processes. In general, workflow processes are modeled by WF-nets (van der Aalst, 1998), i.e. Petri nets with one initial and one final places and every place or transition being on a path from the initial place to the final one. The execution of a case is represented as a firing sequence that starts from the initial marking consisting of a single token on the initial place. The token on the final place with no tokens left on the other places indicates the proper termination of the case execution. A model is called sound iff every reachable marking can terminate properly. Such models reflect the partial ordering of activities in the process and abstract from resources, e.g. machines or personnel, that actually execute tasks and any quantitative measures of its execution.

In (K.M. van Hee, 2005) a notion of $RCWF$-nets and a respective soundness property was introduced. Such models represent WF-nets that take resources into account.

The concept of time was intentionally avoided in the classical Petri net as timing constraints may prevent certain transitions from firing. Many different ways of incorporating time in Petri nets have been proposed. Some timed Petri net models use deterministic delays (Ramchandani, 1973; Sifakis, 1977). The others use interval timing (Merlin, 1974; van der Aalst, 1993) or stochastic delays (G. Florin, 1980; M.A. Marsan, 1984; M.A. Marsan, 1985).

In real systems execution of tasks in workflow processes depends on the various conditions, such as availability of free resources and all needed information and so on. Hence for simulation and quantitative analysis of workflow systems these external conditions are essential.

In (Reijers, 2003) a model that takes into account both resources and timing is proposed. In that book different heuristic rules for allocation of additional resources that minimize mean throughput time of the process are also discussed.

Gorbunov O. and Sokolov V.
RESOURCE MANAGEMENT FOR ONE CLASS OF WORKFLOW PROCESSESA Stochastic Petri Net Based Approach.
DOI: 10.5220/0004459901680174
In *Proceedings of the First International Symposium on Business Modeling and Software Design (BMSD 2011)*, pages 168-174
ISBN: 978-989-8425-68-3

2 STOCHASTIC PETRI NETS WITH PRIORITIES

We propose a class of stochastic Petri nets with priorities SPN^e. These nets combine properties of $GSPN$-nets proposed in (M.A. Marsan, 1985) and Interval Timed Petri nets (van der Aalst, 1993). SPN^e-nets are based on Petri nets with priorities, so begin with these formalism.

2.1 Petri Nets with Priorities

Definition 2.1 (Petri nets with priorities) *Petri net with priorities (PN_{pr}-net) is a tuple (P,T,R,Pr), where (P,T,R) – Petri net; $Pr \in T \to \mathbb{N} \cup \{0\}$ – priority function, that assign for each transition t natural number $Pr(t)$, priority of the transition.*

The definition of marking is the same as for ordinary Petri nets but firing rule differs. The transition that is active in Petri net P,T,R is potentially active in Petri net P,T,R,Pr with priorities.

Denote $M_{pr}()$ a function $M_{pr}() \in 2^T \to 2^T$, that for any set of transitions from T returns the subset of transitions with maximal priority:

$$\forall J \subseteq T \;\; M_{pr}(J) = \{t \in J | \overline{\exists} t' \in J : Pr(t') > Pr(t)\}.$$

Potentially active transition t_j of the PN_{pr}-net $N = (P,T,R,Pr)$ is active in marking m, if there is no another potentially active transition $t_i \in T$: $Pr(t_i) > Pr(t_j)$. So, the existence of priorities restricts the number of active transitions in comparison with the same Petri net without priorities. Denote $At(m)$ the set of active transitions of the PN_{pr}-net N in marking m. From the definition of $At(m)$ follows that two transitions t_i and t_j, $t_i \neq t_j$ are active in marking m only if they have the same priority: $Pr(t_i) = Pr(t_j)$. Active transition may fire. Firing rule is the same as in the ordinary Petri net.

It is well known that the expressive power of Petri nets with priorities is greater than of the ordinary Petri nets. So, in general, if we model workflow processes by means of WF-nets with priorities, soundness property would be undecidable.

Let us consider free-choice Petri nets with priorities. Remind that by constructing WF-nets with task refinement approach using basic structures of choice, sequential and parallel execution, a free-choice WF-net will be obtained (van der Aalst, 2000). Relation SC (structural conflict relation on the set of T) for such nets is reflexive, transitive and symmetric. So, we may conclude that it is an equivalence relation and the set T may be divided into the disjoint subsets $SC_1, SC_2 \dots SC_k : SC_1 \cup SC_2 \cup \dots \cup SC_k = T$.

Obviously, for free-choice PN_{pr}-net N all transitions in any subset SC_i potentially active or not potentially active at the same time. It is easy to prove that for such nets if there exist at least two transitions $t_i, t_j \in T : t_i \neq t_j, t_i SC t_j, Pr(t_i) \neq Pr(t_j)$ then there exist dead transitions that will never be active.

Obviously, for free-choice Petri net $N = (P,T,R,Pr)$, with priorities such that $\forall t_i, t_j \in T : t_i SC t_j \Rightarrow Pr(t_i) = Pr(t_j)$, for any marking m $At(m)$ is empty or consists of subsets $SC_{a1}, SC_{a2}, \dots, SC_{ap}$ with the same priority.

If we constrain the structure of the net by the free-choice property and require certain rules on priorities assignment, the soundness property will be decidable.

The following theorem may be proved (Gorbunov, 2006).

Theorem 2.1 *If a free-choice WF-net $N = (P,T,R)$ is sound, the WF-net N' with priorities: $N' = (P',T',R',Pr')$: $P' = P, T' = T, R' = R, \forall t'_i, t'_j \in T' : t'_i SC t'_j \Rightarrow Pr(t'_i) = Pr(t'_j)$ with the same initial marking $m'_i = m_i$ is sound.*

2.2 Stochastic Petri Nets

Tokens have time stamps that denotes time when token will be available for transition execution. While executing, transitions assign time stamps to the produced tokens.

Definition 2.2 (Stochastic Petri net with priorities) *A stochastic Petri net with priorities $N \in SPN^e$ is a tuple (P,T,R,W,Pr):*

- *(P,T,R,Pr) – free-choice PN_{pr} net;*
- *$W \in T \to \mathbb{R}^+$.*

There exist two types of transitions: timed and immediate. A transition t is a timed transition iff $Pr(t) = 0$ and is an immediate transition otherwise.

Firing of immediate transitions takes no time. Delays of timed transitions are defined by the negative exponential probability function. For timed transitions W defines the rate of executions, i.e. the parameter of negative exponential probability function of the delays: $\forall t \in T, Pr(t) = 0 : P\{\text{delay of execution of t } \leq x\} = 1 - e^{-W(t)x}$.

For immediate transitions W is used for resolving conflicts between transitions.

For SPN^e-net $N = (P,T,R,W,Pr)$ timed state space $S \subseteq P \to (\mathbb{R} \to \mathbb{N})$ is defined. For timed state $s \in S$ for any $p \in P, s(p)$ is multiset on $\mathbb{R}$. Timed state defines for any place p the number of tokens and their time stamps.

If in a timed state s we abstract from token time stamps, we obtain marking s_m as in ordinary Petri nets. Function $U_t()$ makes an appropriate transformation:

Definition 2.3 (Function $U_t()$)

$$\forall p \in P: \ U_t(s,p) = |s(p)|.$$

So, $s_m = U_t(s)$.

For SPN^e-nets some initial timed state $m_0 \in S$ is fixed.

Function $first()$ for timed state $s \in S$ and position $p \in P$ returns the minimal time stamp of tokens in p:

$$first(s(p)) = \min(\{k \in \mathbb{R} | s(p)(k) > 0\}).$$

Definition 2.4 (Function $ttime(s,t)$)

$$\forall s \in S, t \in T \ ttime(s,t) = \begin{cases} \max\limits_{p \in {}^\bullet t}\{first(s,p)\}, \\ \qquad if\, t \in E_t(U_t(s)); \\ \textit{not defined otherwise} \,. \end{cases}$$

Function $ftime()$ for timed state $s \in S$ returns first moment of time, when some transition in $E_t(U_t(s))$ can fire:

$$ftime(s) = \min_{t \in E_t(U_t(s))} ttime(s,t).$$

Here E_t is function that for any marking of Petri net returns the set of active transitions.

Definition 2.5 (Function $fire()$) *Function $fire()$ returns the set of transitions, that can fire in timed state $s \in S$:*

$$\begin{aligned} fire(s) &= M_{pr}(\{t \in E_t(U_t(s)) | \\ &\qquad ttime(s,t) = ftime(s)\}). \end{aligned}$$

If $t \in fire(s)$, it can fire at the timed state s.

If $fire(s)$ consists of some transitions, they have the same priority value. It can be shown, that the set of $fire(s)$ is empty or consists of sets $SC_{a1}, SC_{a2}, \ldots, SC_{ap}$ with the same priority value. Suppose $fire(s) = SC_{a1}, SC_{a2}, \ldots, SC_{ap}$.

The probability of firing the transition $t_b \in fire(s)$ is defined by its relational weight among other transitions from $fire(s)$:

$$P\{t_b \text{ will fire in } s\} = \begin{cases} \dfrac{W(t_b)}{\sum\limits_{u \in fire(s)} (W(u))}, \\ \qquad \text{if } t_b \in fire(s) \\ 0, \text{ otherwise.} \end{cases} \tag{1}$$

When a transition t is firing, tokens with the smallest time stamps are removed from its input places and tokens with time stamps equal to the moment of firing increased by the firing delay d are added to its output places. Firing delay d is sampled from the probability function associated with the delay of transition. The new timed state obtained from timed state $s \in S$ by firing the transition $t \in T$ with delay $d \in \mathbb{R}$, is defined by the function g:

$$g(s,t,d)(p) = \begin{cases} s(p), \text{ if } p \notin {}^\bullet t \text{ and } p \notin t^\bullet \\ s(p) - [first(s(p))], \\ \qquad \text{if } p \in {}^\bullet t \text{ and } p \notin t^\bullet \\ s(p) + [ftime(s) + d], \\ \qquad \text{if } p \notin {}^\bullet t \text{ and } p \in t^\bullet \\ s(p) - [first(s(p))] + \\ \qquad [ftime(s) + d], \\ \qquad \text{if } p \in {}^\bullet t \text{ and } p \in t^\bullet. \end{cases}$$

2.3 Stochastic Workflow Nets (SWF^e-nets)

A SWF^e-net N is a tuple (P,T,R,W,Pr):

- (P,T,R) is a WF-net;
- (P,T,R,W,Pr) is a SPN^e-net.

For the SWF^e-net $N = (P,T,R,W,Pr)$ the initial timed state m_0 is defined as follows:

$$\forall p \in P, m_0(p) = \begin{cases} [0], \text{ if } p = i \\ 0, \text{ otherwise.} \end{cases}$$

At the initial timed state m_0 the net contains one token in the place i with the time stamp equal to 0, other places don't contain tokens.

2.4 Stochastic Resource-Constrained Workflow Nets ($SRCWF^e$-nets)

In this section a stochastic extension for $RCWF$-nets (K.M. van Hee, 2005) is proposed. Stochastic $RCWF$-net ($SRCWF^e$-net) N is a tuple $(P_p \cup P_r, T, R_p \cup R_r, W, Pr)$:

- $(P_p \cup P_r, T, R_p, W, Pr)$ – SWF^e-net;
- $(P_p \cup P_r, T, R_p \cup R_r)$ – $RCWF$-net with places i and f as source and sink places.

Denote $P = P_r \cup P_p$.

Definition 2.6 (Initial timed state) *$SRCWF^e = (P_p \cup P_r, T, R_p \cup R_r, W, Pr)$ with the timed state space S, the initial timed state $m_0 \in S$ is defined as follows:*

$$for\ p \in P, m_0(p) = \begin{cases} [0] \ if\ p = i, \\ l[0] \ if\ p \in P_r, \\ \qquad where\ l \in \mathbb{N}, l > 0 \\ \emptyset \ otherwise. \end{cases}$$

In the initial timed state m_0 there is one token in place i with time stamp 0 and some tokens in places from P_r. Places from P_r (resource places) contain multisets of tokens with time stamp 0. Every place from P_r denotes a resource class. The quantity of tokens in the resource position denotes the quantity of resources of that class.

There are two possible classes of methods for quantitative analysis of $SRCWF^e$-nets: simulation and analytical methods. Note, that analytical methods are applicable only for restricted subclasses of $SRCWF^e$-nets with additional constraints on structure, initial timed state and so on. In general, at present, analytical methods are inapplicable for quantitative analysis (Gorbunov, 2005; Gorbunov, 2006).

3 RESOURCE MANAGEMENT FOR ONE CLASS OF $SRCWF^E$-NETS

3.1 $SRCWF_s^{e1}$-nets

Denote the set of transitions of the $SRCWF^e$-net N which have input places from P_r as T_r: $T_r = \{t \in T | \exists p \in P_r : (p,t) \in R_r\}$. Suppose, that all places from P_r have some names $p_{r1},\dots,p_{rs}$. Denote subsets of $t \in T_r$, such that $(p_{ri},t) \in R_r$, as T_{ri}. Suppose that all transitions from T_{ri} are denoted as $t_{(ri)(1)}, t_{(ri)(2)}, \dots, t_{(ri)(n_i)}$, where $|T_{ri}| = n_i$.

We will use a special class of $SRCWF^e$-nets (denote it $SRCWF_s^{e1}$-nets) with some restrictions and modifications.

The net $N = (P_p \cup P_r \cup p_g, T \cup \{t_i,t_f\}, R_p \cup R_r \cup \{(t_i,i),(t_i,p_g),(p_g,t_i),(f,t_f)\}, W, Pr)$ is a $SRCWF_s^{e1}$-net, iff $(P_p \cup P_r, T, R_p \cup R_r, W, Pr) - SRCWF^e$-net with some restrictions:

1. the net (P_p,T,R_p) is a state machine non-cyclic net;
2. in the net $(P_p,T,R_p) \overline{\exists} t_l,t_m : t_l \in T_r, t_m \in T, t_l \neq t_m, t_l SC t_m$;
3. in the net (P_p,T,R_p,Pr) : $\forall t_i,t_j \in T : t_i SC t_j \Rightarrow Pr(t_i) = Pr(t_j)$;
4. for any $t \in T_r : \exists! p_{ri} \in P_r : (p_{ri},t),(t,p_{ri}) \in R_r$;
5. for any $t \in T_r : (Pr(t) = 0)$;

$Pr(t_i) = 0$, $Pr(t_f) = 1$.

In other words, the transition t_i is a timed transition and t_f is an immediate transition.

The initial marking $m_0 \in S$ is defined as follows:

$$\forall p \in P, m_0(p) = \begin{cases} 1[0], \text{ if } p = i; \\ 1[0], \text{ if } p = p_g; \\ l[0], \text{ if } p \in P_r, \\ \qquad \text{where } l \in \mathbb{N}, l > 0; \\ \emptyset \text{ otherwise} \end{cases}$$

Note, that the transition t_i will generate the poisson stream of tokens with the rate $W(t_i)$. After each firing of t_i place p_g will contain one token with time stamp increased by the sampled delay of t_i.

The special transition t_f consumes tokens from the place f. The value of $W(t_f)$ is of no importance and, for certainty, let be 1.

Due to the constraints, if any resource position is connected by the output (input) arc with some transition, such position must be connected with it by the input (output) arc. At the same time this transition cannot be connected with any other resource positions. In other words, a resource becomes free after fulfilling the task (firing the transition). Obviously, places from P_r are bounded, moreover, at any reachable marking the number of tokens in any place from P_r is the same as in the initial marking.

All resource places are connected with timed transitions only. That is, if some resource is needed for some task, this task must consume time. At the same time, in the model there may be timed transitions which are not connected with resource places. Such transitions model some time delays which don't depend on resources and are defined by external factors.

Definition 3.1 (Function $ttime_p(s,t)$)

$$\forall s \in S, t \in T \; ttime_p(s,t) = \begin{cases} \max\limits_{p \in {}^\bullet t \restriction_{P_p}} \{first(s,p)\}, \\ \qquad \text{if } t \in E_t(U_t(s)); \\ \text{don't defined} \\ \qquad \text{otherwise .} \end{cases}$$

The function $ttime_p(s,t)$ for timed state s and potentially active transition $t \in T, t \in E_t(U_t(s))$ result in the moment of time when the transition t could fire, if we abstract from places in P_r.

Definition 3.2 (Waiting time) *For $SRCWF_s^{e1}$-net $(P_p \cup P_r \cup p_g, T \cup \{t_i,t_f\}, R_p \cup R_r \cup \{(t_i,i),(t_i,p_g),(p_g,t_i),(f,t_f)\}, W, Pr)$, that induces the stochastic process $\pi = \{(X_n,Y_n) | n = 0,1,2,\dots\}$, where X_n is a timed state after n firings, Y_n is a transition that will fire at the state X_n, define the stochastic variable $WT_t(j)$, waiting time of transition $t \in T$ due to the lack of resources:*

$$WT_t(j) = \begin{cases} ttime(X_j,Y_j) - ttime_p(X_j,Y_j), \\ \qquad \text{if } t = Y_j, \\ \text{is not defined otherwise.} \end{cases}$$

$WT_t(j)$ equals the waiting time of firing of t due to the lack of resources, when t fires in the state X_j and is not defined otherwise. Denote by $E(WT_t)$ the mathematical expectation of $WT_t(j)$. Of course, there must be some restrictions on functions W to obtain finite values of $E(WT_t)$.

3.2 Soundness of the Nets Underlying $SRCWF_s^{e1}$-nets

Consider $SRCWF_s^{e1}$-net $N = (P_p \cup P_r \cup p_g, T \cup \{t_i, t_f\}, R_p \cup R_r \cup \{(t_i, i), (t_i, p_g), (p_g, t_i), (f, t_f)\}, W, Pr)$.

WF-net (P_p, T, R_p) is a state machine net. Hence, WF-net (P_p, T, R_p) is sound.

Due to the structural constraints of $SRCWF_s^{e1}$-nets and theorem (2.1), Petri net (P_p, T, R_p, Pr) with priorities is sound.

It is obvious that $RCWF$-net with priorities $P_p \cup P_r, T, R_p \cup R_r, Pr$ is sound due to the structural constraints.

3.3 Problem Statement

Let N be a $SRCWF_s^{e1}$-net. Denote $PF \in T_r \to \mathbb{R}$ a function that assigns for each timed transition from T_r some penalty for waiting per unit of time due to the lack of resources. $PF(t)$ may reflect the cost of waiting or some measure of client dissatisfaction.

The problem is to minimize the function F:

$$F = \sum_{t \in T_r} E(WT_t) PF(t) \to \min. \qquad (2)$$

Another important characteristic of workflow processes is throughput time (Reijers, 2003). Note that it is possible to vary function F without changing throughput time of the process.

3.4 Decision Approach

Let us introduce some transformation rule with $SRCWF_s^{e1}$-net N. Denote γ the maximum value of the function Pr in N: $\gamma = M_{pr}(T)$. For every time transition $t_{(ri)(j)} \in T_{ri}$ add (in the set T) new immediate transition $t'_{(ri)(j)}$: $W(t'_{(ri)(j)}) = W(t_{(ri)(j)})$, $PF(t'_{(ri)(j)}) = PF(t_{(ri)(j)})$, $Pr(t'_{(ri)(j)}) \in \{1+\gamma, 2+\gamma, \ldots, n_i+\gamma\}$ and the new place $p'_{(ri)(j)}$ (in the set P_p), ${}^\bullet t'_{(ri)(j)} = {}^\bullet t_{(ri)(j)}$, $t'^\bullet_{(ri)(j)} = \{p'_{(ri)(j)}\}$, ${}^\bullet t_{(ri)(j)} = \{p'_{(ri)(j)}\}$.

Denote the modified $SRCWF_s^{e1}$-net N as N_f. Note, that N_f is not a $SRCWF_s^{e1}$-net. The set T_r of the net N_f consists of immediate transitions. This transformation rule preserves the soundness of $RCWF$-net with priorities that underlies N_f.

Now we obtain the possibility to change the value of function F by changing the priorities of transitions from sets T_{ri}.

It may be shown that to obtain the same value of F in the net N_f, the priorities of transitions within each set T_{ri} must be the same (for example, $1+\gamma$).

In general, simulation may be used to obtain some optimal result. In brute force approach, $|T_{r1}|^{|T_{r1}|} * |T_{r2}|^{|T_{r2}|} * \ldots$ simulations may be carried out to obtain some optimal result. Moreover, if some transitions from some T_{ri} have the same priority, the value of function W may be changed to obtain the optimal result.

3.5 Example

Let us introduce an example that illustrates the approaches discussed above. In Figure 1 some $SRCWF_s^{e1}$-net N is illustrated.

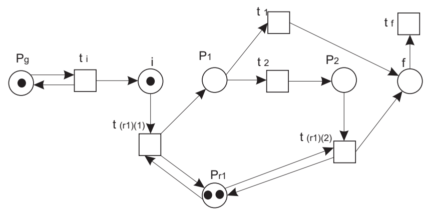

Figure 1: $SRCWF_s^{e1}$-net N.

The characteristics of the net N are specified in Table 1.

Table 1: Characteristics of N.

Transition	W	Pr	PF
t_i	10	0	
t_1	0.2	1	
t_2	0.8	1	
$t_{(r1)(1)}$	20	0	50
$t_{(r1)(2)}$	30	0	100
t_f	1	1	

By applying transformation rules from 3.4 the net N_f illustrated in Figure 2 is produced. Priorities of transitions within set T_{r1} are selected arbitrarily.

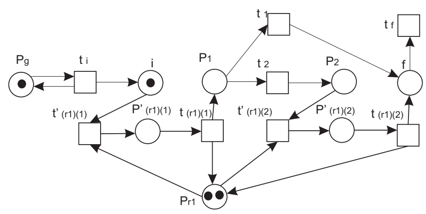

Figure 2: The net N_f obtained from N.

The characteristics of the net N_f are specified in Table 2.

Table 2: Characteristics of N_f.

Transition	W	Pr	PF
t_i	10	0	
t_1	0.2	1	
t_2	0.8	1	
$t_{(r1)(1)}$	20	0	
$t'_{(r1)(1)}$	20	2	50
$t_{(r1)(2)}$	30	0	
$t'_{(r1)(2)}$	30	3	100
t_f	1	1	

4 CONCLUSIONS

This paper opens many ways for further work. One way is to develop heuristic rules for assigning priorities without simulation for some classes of nets and functions F (may be not linear). The other way is to weaken the constraints of $SRCWF_s^{e1}$-net such as the structural constraint of the state machine. Some heuristic rules may also be developed to obtain values of W for some T_{ri} of the net N_f in the case of deriving transitions with the same priorities.

ACKNOWLEDGEMENTS

Authors appreciate prof. E.A. Timofeev for the useful advices and discussion in the paper related topics.

REFERENCES

G. Florin, S. N. (1980). Evaluation based upon stochastic petri nets of the maximum throughput of a full duplex protocol. In C. Girault, W. R., editor, *Applications and Theory of Petri Nets*, pages 280–288. Springer-Verlag.

Gorbunov, O. (2005). Optimisation of workflow processes execution. *Modeling and Analysis of Informational Systems*, 12(1):45–51.

Gorbunov, O. (2006). *Modeling and Optimisation of Workflow Processes*. Phd thesis, Yaroslavl State University, Yaroslavl.

K.M. van Hee, A. Serebrenik, N. S. M. V. (2005). Soundness of resource-constrained workflow nets. In Girardo, G. and Darondeau, P., editors, *Applications and Theory of Petri Nets 2005*, volume 3536, pages 250–267. Lecture Notes in Computer Science, Springer-Verlag.

M.A. Marsan, G. B. e. a. (1985). On petri nets with stochastic timing. In *International Workshop on Timed Petri Nets*, pages 80–87. IEEE Computer Society Press.

M.A. Marsan, G. Balbo, G. C. (1984). A class of generalised stochastic petri nets for the performance evaluation of multiprocessor systems. *ACM Transactions on Computer Systems*, 2(2):93–122.

Merlin, P. (1974). *A Study of the Recoverability of Computer Systems*. Phd thesis, University of California, Irvine.

Ramchandani, C. (1973). *Performance Evaluation of Asynchronous Concurrent Systems by Timed Petri Nets*. Phd thesis, Massachusetts Institute of Technology, Cambridge.

Reijers, H. (2003). *Design and Control of Workflow Processes*. Springer-Verlag.

Sifakis, J. (1977). Use of petri nets for performance evaluation. In H. Beilner, E. G., editor, *Third International Symposium IFIP. Measuring, modeling and evaluating computer systems*, pages 75–93, Amsterdam. Elsevier Science Publishers.

van der Aalst, W. (1993). Interval timed coloured petri nets and their analysis. In Marsan, M., editor, *Application and Theory of Petri Nets*, volume 691, pages 453–472. Lecture Notes in Computer Science, Springer-Verlag.

van der Aalst, W. (1998). The application of petri nets to workflow management. *The Journal of Circuits, Systems and Computers*, 8(1):21–66.

van der Aalst, W. (2000). Workflow verification: Finding control-flow errors using petri net-based techniques. *Business Process Management. Lecture Notes in Computer Science*, 1806:161–183.

van der Aalst, W. and van Hee, K. (2002). *Workflow Management: Models, Methods and Systems*. MIT Press, London.

PETRI NET BASICS

Definition 4.1 (Petri net) *Petri net N is a tuple (P,T,R), where*

- *P – finite set of places;*
- *T – finite set of transitions, $(P \cap T = \emptyset)$;*
- *R – flow relation, $R \subseteq (T \times P) \cup (P \times T)$.*

We use $^\bullet t$ to denote the set of input places of a transition t: $p \in {^\bullet t}$ iff $p \in R(p,t)$. $t^\bullet$ have the similar meaning: it is the set of output places of a transition t: $p \in t^\bullet$ iff $p \in R(t,p)$.

Definition 4.2 (Petri net marking) *The marking (state) m of Petri net N is a mapping m: $P \to \mathbb{N}$. A marking is represented by the vector $(M(p_1)\dots M(p_n))$, where $p_1,\dots,p_n$ is an arbitrary fixed enumeration of P.*

Definition 4.3 (Firing rule) *A marking m of a Petri net (P,T,R) enables a transition $t \in T$ if it marks every place in $^\bullet t$. If t is enabled at m, then it can fire, and* *its firing leads to the successor marking m' (written $m \xrightarrow{t} m'$) which is defined for every place $p \in P$:*

$$m' = \begin{cases} m(p) \text{ if } p \notin t^{\bullet} \text{ and } p \notin {}^{\bullet}t, \text{ or } p \in t^{\bullet} \text{ and } p \in {}^{\bullet}t \\ m(p)+1 \text{ if } p \in t^{\bullet} \text{ and } p \notin {}^{\bullet}t \\ m(p)-1 \text{ if } p \notin t^{\bullet} \text{ and } p \in {}^{\bullet}t \end{cases}$$

Definition 4.4 (Free choice Petri net) *Petri net N is called a free choice Petri net, if for any transitions $t_1, t_2 \in T$: if ${}^{\bullet}t_1 \cap {}^{\bullet}t_2 \neq \emptyset$, then ${}^{\bullet}t_1 = {}^{\bullet}t_2$.*

Definition 4.5 (Structural conflict of transitions) *A structural conflict of transitions is a relation SC on the set of transitions T: $\forall t_i, t_j \in T : t_i SC t_j$ iff ${}^{\bullet}t_i \cap {}^{\bullet}t_j \neq \emptyset$.*