IEEE SMC 2003 Conference Proceedings，USA，Oct. 2003，pp.1175-1180

Approximate Performance Analysis of Workflow Model [*]

JianQiang Li, YuShun Fan
Department of Automation
Tsinghua University
Beijing, China
lijq99@mails.tsinghua.edu.cn, fan@cims.tsinghua.edu.cn

MengChu Zhou
Department of ECE
New Jersey Institute of Technology
Newark, NJ07102-1982 U.S.A.
zhou@njit.edu

Abstract - *Multi-dimension Workflow net (MWF-net) [1], which includes process, organization, and resource perspectives, is introduced. Using the structure analysis of the TWF-net in the process perspective and the perspectives mapping, the routing of transaction instances in the multi-TWF-nets can be projected into the flow of transaction instance between different resource pools in the resource perspective. After the relevant work in [1] is briefly reviewed, the boundedness verification method of a MWF-net is proposed. A MWF-net is bounded implies the corresponding queuing network in the resource perspective has stable solution. Based on the discussion of several operational principles in the context of workflow model, an approximate method for performance analysis of a workflow model is presented.*

Keywords: workflow, Petri nets, performance analysis, operational analysis.

1 Introduction

Workflow modeling and analysis play an important role in the research of workflow techniques and successful implementation of workflow management. Using Petri nets as a base mechanism to represent a workflow model has been extensively studied [2]. Based on the Workflow net (WF-net) [3], Multi-dimension Workflow net (MWF-net) is proposed for workflow performance analysis in [1] to overcome two shortcomings in existing techniques: only one isolated workflow process is considered, and almost no organization and resource information is considered.

Several relevant concepts are reviewed here. A Petri net PN is called a WF-net iff: (1) PN has two special places: ε and θ where ε is a source place: $^{\bullet}\varepsilon = \phi$ and θ is a sink place: $\theta^{\bullet} = \phi$; and (2) If we add a new transition t to PN which connects θ with ε, namely, $^{\bullet}t=\{\theta\}$, $t^{\bullet}=\{\varepsilon\}$, then the resulting extended net $\overline{PN}=(\overline{P},\overline{T},\overline{F})$, where $\overline{P}=P$, $\overline{T}=T\cup\{t\}$, and $\overline{F}=F\cup\{(\theta,t),(t,\varepsilon)\}$, is strongly connected. In a WF-net, building blocks such as AND-split, AND-join, OR-split, and OR-join are used to model Sequential Control Structure (SCS), Concurrent Control Structure (CCS), Alternative Control Structure (ACS), and Iterative Control Structure (ICS). Obviously, a WF-net gives only process control specification of a workflow model. In a TWF-net [1], a firing delay representing the execution durations of activities is associated with each transition in a WF-net. MWF-net is a five tuple $<W, O, R, F_P, F_R>$, where $W=\{w_1,w_2,\cdots,w_u\}$ is a set of TWF-nets; $O=\{o_1,o_2,\cdots o_v\}$ is a set of roles, and o_i is a role defined in the organization perspective; $R=\{r_1,r_2,\cdots,r_q\}$ is a set of resource pools, each r_i denotes a resource pool defined in the resource perspective; $F_P\subseteq\{O\times T_{w_1}, O\times T_{w_2}, \cdots, O\times T_{w_u}\}$ represents the mapping relation between process perspective and organization perspective, where T_{w_i} is the transition set of TWF-net w_i; $F_R\subseteq O\times R$ represents the mapping relation between organization perspective and resource perspective.

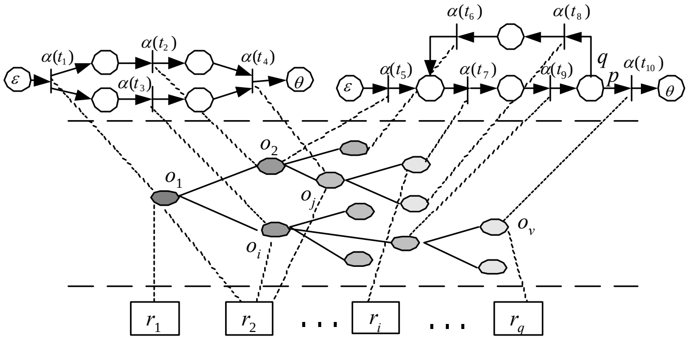

Figure 1. The graphic representation of an MWF-net

A MWF-net shown in Fig. 1 includes three levels. The first level is process perspective, where some free choice TWF-nets are used to specify the process control and timing constraints of workflow models. The second level is the organization perspective, in which each transition is appointed a specific role for its firing (executing the corresponding task). Each role of the organization perspective can be performed by members of several resource pools described in the resource perspective of the third level. If there are Y_i individual resource agents in each resource pool r_i, we have $Y=[Y_1, Y_2, \cdots, Y_q]$ which is called resource state of the MWF-net. Heuristically, the routing of transaction instances in multi-TWF-nets can be projected into the flow of transaction instance between different resource pools (shown in Fig. 2). If each resource pool is viewed as a service station, and an individual resource belonging to this resource pool is a service center, the performance analysis of a workflow model can be transformed as indices calculation of the queuing network.

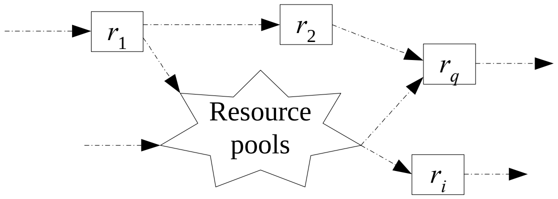

Figure 2. Queuing network in the resource perspective

Assuming a transition has exponentially distributed firing delay and transaction arrival rate for each TWF-net is a Poisson process, [1] shows how to compute the transaction instance arrival rate. It mainly includes three steps: model decomposition, relative iterative coefficient computing, and perspectives mapping.

A decomposition algorithm (see appendix for its detailed description) from a free-choice TWF-net to a set of Free Choice subnets (FC-subnets) is presented. Applying the decomposition algorithm to all TWF-nets in the MWF-net, u FC-subnet sets R^k ($1\le k \le u$), each one of which corresponds to w_k, can be obtained. Each TWF-net in the MWF-net describes the process control and time aspects of a kind of transaction instance which can also be partitioned into several types, and each FC-subnet in R^k of w_k corresponds to a routing path of a specific type transition instance in w_k. Suppose that each w_k includes n_k transitions (then the total number of transitions in the MWF-net is $n=\Sigma n_k$) and is built for the processing of s_k types of transaction instances $[I^k_1,I^k_2,\cdots,I^k_{s_k}]$, there is a vector $\boldsymbol{\beta}^k=[\beta^k_1, \beta^k_2,\cdots,\beta^k_{s_k}]$ satisfying $\beta^k_1+\beta^k_2+\cdots+\beta^k_{s_k}=1$, where $\beta^k_i \ge 0$ is the percentage of I^k_i in the total number of arrival transaction instances of w_k. Assuming the average transaction instance arrival rate for w_k is λ^k, the arrival rate of transaction instance of type I_i (i.e., the transaction instance arrival rate of FC-subnet PN^k_i) is the ith item of $\lambda^k_{\mathrm{I}}=\lambda^k*\boldsymbol{\beta}$.

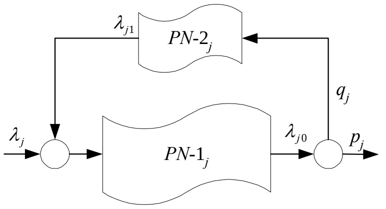

Figure 3. An example of an ICS

For each transition $t\in PN^k_i$, there are a relative iterative coefficient $\delta_j(t)$ to an ICS $ICS\text{-}j\in PN^k_i$ and a relative iterative coefficient $\Omega_i(t)$ to PN_i. If t doesn't belong to *ICS-j*, its relative iterative coefficient to *ICS-j* is 1, i.e. $\delta_j(t)=1$. Otherwise, t's iterative coefficient relative to *ICS-j* is determined by its position in *ICS-j*. [1] show a method to calculate the relative iterative coefficient $\delta_j(t)$ of t to an ICS *ICS-j*, then $\Omega_i(t)$ to PN^k_i. In Figure 3, an ICS *ICS-j* of an FC-subnet PN^k_i includes PN-1_j and PN-2_j two components. In the steady state the departure rates from the two components will be λ_{j0} and λ_{j1}, respectively. Arrivals to the component PN-1_j occur either from the outside environment at the rate λ_j or from the PN-2_j component at the rate λ_{j1}. Therefore, $\lambda_{j0}=\lambda_{j1}+\lambda_j$. Given that a transaction instance just completed processing in PN-1_j, it will next go out of the ICS with probability p_j or request PN-2_j service with probability q_j. Therefore, $\lambda_{j1}=\lambda_{j0}q$. Thus, $\lambda_{j0}= \lambda_j/p_j$ and $\lambda_{j1}=q_j\lambda_j/p_j$. Then, the relative iterative coefficients of all the transitions in PN-1_j to *ICS-j* are $1/p_j$, and the relative iterative coefficients of all the transitions in PN-2_j to *ICS-j* are q_j/p_j. Assuming transition t belong to n ICSs of PN^k_i, and its relative iterative coefficient $\delta_x(t)$ to each *ICS-x* has been obtained, then the relative iterative coefficient to PN^k_i can be calculated as $\Omega_i(t)=\prod_{x=1}^{n}\delta_x(t)$.

Based on the relative iterative coefficient computing for each transition in the corresponding FC-subnet, a transition vs. FC-subnet matrix B_k is constructed for w_k:

$$b_{ij}=\begin{cases}\Omega_i(t_j) & \text{If } PN^k_i \text{ includes transition } t_j\\ 0 & \text{Otherwise}\end{cases}$$

Each item b_{ij} of the matrix B_k can be seen as the average firing times of transition t_j to complete the processing of transaction instance I^k_i. Then, $\lambda_{T_k}=\lambda^k_{\mathrm{I}}*B_k$, where each item of λ_{T_k} represents the average transaction instance arrival rate of the corresponding transition in w_k.

To realize the perspectives mapping analysis, the transition vs. role matrix C_k for all w_k and the role vs. resource matrix D for the MWF-net are constructed from the mapping relations between these three perspectives, i.e., F_{OP} and F_{OR}:

$$d_{ij}=\begin{cases}\gamma_{ij} & \text{If resource pool } r_j \text{ performs } \gamma_{ij} \text{ part of job appointed to role } o_i\\ 0 & \text{Otherwise}\end{cases}$$

$$c_{ij}=\begin{cases}1 & \text{If the firing of transition } t_j \text{ need the support of role } o_i\\ 0 & \text{Otherwise}\end{cases}$$

C_k specify distinctly the firing of a transition in w_k need the support of which kind of role, and d_{ij} represent the probability that the requests of role o_i is allocated to a resource agent of resource pool r_j. For each resource pool in R, the arrival rate of service requests (transaction instance processing) which are generated from w_k, can be acquired, i.e., the kth item of $\lambda^k_R=\lambda_{T_k}*C_k^T*D$, where C_k^T is the transpose of C_k. After λ^k_R is obtained for all the u

TWF-nets, the total service request arrival rate can be computed as $\lambda_R = \Sigma_k \lambda^k{}_R$, where the jth item λ_{r_j} represents r_j's transaction instance (service request) arrival rate.

Through the three steps analysis, the routing of transaction instances in the TWF-net is mapped into the service requests arrival rate of the resource pools.

2 Boundedness verification

Boundedness verification of a workflow model concerns whether congestion or overflow may occur in the environment of multi-instances running concurrently. A MWF-net is bounded if every place in anyone of the u TWF-nets is bounded. Then, workflow's boundedness verification corresponds to verify if there is a place that can have an infinite number of tokens. We know from research that there is two reasons may cause the unboundedness of workflow. One is that some process control structures (such as a trap or other structural conflicts [4]) cause the number of tokens in some places grows indefinitely. Obviously, this kind of boundedness verification belongs to the logical level analysis of workflow model. As mentioned above, we only consider the sound WF-net in this paper, and van der Aalst [3] demonstrates that a WF-net is sound if and only if the extended WF-net with a token in its source place ε is live and bounded, then, this situation (the unboundedness is caused by the error in net structure) has been avoided. The other is that the extended time constraints of a sound WF-net cause the unboundedness. Boundedness verification considered in this paper belongs to the temporal level.

A similar boundedness verification method of TCWF-net has been given in [5]. Due to the fact that the resource and organization information is not considered in that paper, it can only be apply to a workflow model in which there is a dedicated server for an activity's execution. Now, in the framework of the MWF-net containing multiple TWF-nets as well as the organization and resource information, the boundedness verification of a workflow model is discussed.

We know the service requests of each resource pool may be generated by several transitions, and the transaction instance processing requests coming from one transition may be dispatched (through the perspective mapping) to several resource pools. Although the firing delays of all transitions follow exponential distribution, the service time of a resource pool whose requests (of transaction instance processing) are generated from many transitions doesn't follow exponential distributions. But its average value can be calculated.

First, the matrix $C=[C_1 \vdots C_2 \vdots \cdots \vdots C_u]$ is constructed. Then, the matrix $A_{n\times q}=C^T * D$ is defined for the MWF-net, where n is the total number of transition in the MWF-net and $a_{ij}=\sum_{l=1}^{q} c_{li} d_{lj}$ represents the probability that the arrival transaction instances (service request) of transition t_i ($1\le i \le n$) is dispatched (through the perspective mapping) to resource pool r_j. If $\alpha(t_i)$ represents the firing delay of transition t_i, and X_j represents the service time of resource pool r_j, we have $X_j=\alpha(t_1)I_{jt_1}+\alpha(t_2)I_{jt_2}+\cdots+\alpha(t_m)I_{jt_n}$, where I_{jt_i} is the indicator function that takes on the value of 1 if resource pool r_j's service request is from transition t_i and 0 otherwise. As mentioned above, the firing delay of each transition t_i ($1\le k \le n$) in the MWF-net follows exponential distribution, and $E(\alpha(t_i))=1/\mu_{t_i}$. Also we know that $E(I_{jt_i})=a_{ij}\lambda_{t_i}/\lambda_{r_j}$, where λ_{t_i} is the transaction instance arrival rate of transition t_i. Then, the average service time of the resource agent in resource pool r_j can be calculated as below $E(X_j)=\dfrac{1}{\lambda_{r_j}}\sum_{i=1}^{n}\dfrac{a_{ij}\lambda_{t_i}}{\mu_{t_i}}$. After the average request arrival rate λ_{r_j} and service rate $\mu_{r_j}=1/E(X_j)$ of each resource pool r_j are both obtained, a theorem about the boundedness of a workflow model can be derived.

Theorem 1: A sound MWF-net is bounded iff $\forall r_j\in R$, $\lambda_{r_j}<Y_j\mu_{r_j}$

This theorem can be easily proved and thus its proof is omitted here. Obviously, if each resource pool is look as a queuing system, Theorem 1 corresponds to verify if the queuing system has a stable solution. If a queuing system has no stable solution means that the the average number of transaction instances in the waiting queue of corresponding resource pool is infinitity. Then, there is at least one input place of the transition whose firng genereted resquests to this resource pool is unbounded.

3 Approximate performance analysis

One of the most important characteristics of queuing networks determining their popularity was the development of efficient, polynomial complexity numerical solution algorithms, based on their product form solution. Unfortunately, the important property of product-form network in the resource perspective is destroyed by the synchronization primitive and the service discipline of the service station (resource pool).

The traditional aproach to derive the solution depends on a series of assumptions used in the theory of stochastic processes, such as stationary stochastic process, stochastic equilibrium, exponential service time, and ergodic. Although some of these assumptions are difficult to be proved to hold by observing the system in a finite time period, even most can be disproved empirically [7], many analysts have experienced puzzlement at the accuracy of queuing network results. In applying and validating the results of Markovian queuing network theory, a more applicable method, i.e., operational analysis [7], is developed to derive mathematical equations relating observable quantities in queuing systems. Assuming the observation period is infinite, an approximate performance evaluation of a workflow model based on the operational analysis of queuing network models are discussed.

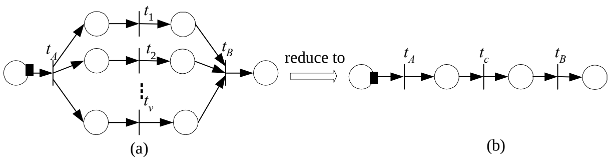

Figure 4. An example of CCS structure

Considering the synchronization primitives in the queuing network of resource perspective, we know they are derived from the synchronization structures, i.e., CCS, in the process control. For the CCS shown in Fig. 4(a), if its transaction instance arrival rate is Poisson process, the arrival rates of transitions t_A, t_1, …, and t_v are Poisson processes and of t_B's arrival rate is not. A transaction instance routing into a CCS through a fork transition with v branches is divided into v independent sub-requests (sub–transaction-instance) that may run in parallel. These v sub-requests must be synchronized at the synchronous transition of the CCS. The synchronization is dependent on the lastly arrived sub-request among v sub-requests at the synchronous transition. Obviously, it can be seem as a process consists of alternative phases-that is, during any single experiment, the process experiences one and only one of the many alternate phases-and these phases have independent exponential distributions. Thus, the firing delay of t_c in Fig. 4(b) is a hyper-exponentially distributed random variable (The hyper-exponential distribution models random variables with more variability than the exponential distribution [8]). t_c's firing delay can be approximated as an exponentially distributed random variable. For that reason, the transaction instance arrival of the synchronization transition t_B can be approximated as a Poisson process. In fact, [6] has shown that the arrival rate of the AND-join activity, which corresponds to the synchronization transition in the TWF-net, can be approximated as a Poisson process. Then, customer arrival rates generated by the firing of t_B to the corresponding resource pools that are appointed for the execution of transition (activity) t_B can be approximated as Poisson processes. Since the transaction instance arrival process of each transition in all TWF-nets at the process level is Poisson process, the transaction instance arrival process of each resource pool is also a Poisson process.

In [7], the weakest known assumption leading to a product form solution is given, i.e., flow balance, one-step behavior, device homogeneity, and routing homogeneity. Flow balance: the number of transaction instance arrivals at a given resource pool must be (almost) the same as the number of departures from that resource pool during the observation period; One step behavior: the only observable state changes of the queuing network in the resource perspective result from single transaction instance either entering the workflow system, or moving between pairs of resource pools in the workflow system, or exiting from the workflow system; Device homogeneity: the service time of a resource pool (server) is defined by the firing delays of corresponding transitions, it must not depend on the queue length of other resource pool; Routing homogeneity: the routing of transaction instance in the queuing network in the resource perspective is decided by the control logic in the process perspective, it must be independent of local queue lengths. Obviously, the queuing network at the third level of MWF-net satisfies all these four conditions.

Thus, each resource pool r_j can be viewed as a $M/G/Y_j$ queuing system. However, the solution to $M/G/Y_j$ cannot be explicitly obtained. We first calculate the variance of the service time of each resource pool, and use Pollaczek-Khintechine formula to obtain all the relevant performance indices for $M/G/1$.

Assuming there is only one resource agent in a resource pool r_j, it can be seen an a $M/G/1$ queuing system. Then, $D(\alpha(t_i))=\frac{1}{\mu_{t_i}^2}$, $E(\alpha^2(t_i))=D(\alpha(t_i))+ E^2(\alpha(t_i))=$

$$\frac{1}{\mu_{t_i}^2}+\frac{1}{\mu_{t_i}^2}=\frac{2}{\mu_{t_i}^2} \quad \mathrm{E}(I_{jt_i})= \mathrm{E}(I^2{}_{jt_i})= \frac{a_{ij}\lambda_{t_i}}{\lambda_{r_j}}$$

$$D(X_j)=E(X_j{}^2)\text{-}E^2(X_j)$$

$$=E((\alpha(t_1)I_{jt_1}+\alpha(t_2)I_{jt_2}+\cdots+\alpha(t_m)I_{jt_n})^2)\text{-}E^2(X_j)$$

$$=E(\alpha^2(t_1)I^2{}_{jt_1}+\alpha^2(t_2)I^2{}_{jt_2}+\cdots+\alpha^2(t_n)I^2{}_{jt_n}+2\alpha(t_1)\alpha(t_2)I_{jt_1}I_{jt_2}$$
$$+2\alpha(t_1)\alpha(t_3)I_{jt_1}I_{jt_3}+\cdots+2\alpha(t_{n\text{-}1})\alpha(t_n)I_{jt_{n\text{-}1}}I_{jt_n})\text{ -}E^2(X_j)$$

$$=E(\alpha^2(t_1))E(I^2{}_{jt_1})+E(\alpha^2(t_2))E(I^2{}_{jt_2})+\cdots+E(\alpha^2(t_n))E(I^2{}_{jt_n})$$
$$\text{-}E^2(X_j)$$

$$=\sum_{i=1}^{n}\frac{2\lambda_{t_i}}{\mu_{t_i}^2\lambda_{r_j}}\text{-}(\sum_{i=1}^{n}\frac{a_{ij}\lambda_{t_i}}{\mu_i\lambda_{r_j}})^2$$

For the $M/G/1$ model of r_j, Pollaczek-Khintechine and Little formulae are used to calculate average customer number in the system L_s, average queue length L_q, average sojourn time in the system W_s, and average waiting time in the queue W_q as: $L_s = \rho + \dfrac{\rho^2 + \lambda^2 D(X_j)}{2(1-\rho)}$, $L_q = L_s - \rho$, $W_s = \dfrac{L_s}{\lambda}$, $W_q = \dfrac{L_q}{\lambda}$.

If there are Y_j resource agent in r_j, it is a $M/G/Y_j$ queuing system, its approximations of the performance measures are given based on the techniques mentioned in [9]. Let $W_{q(M/G/Y_j)}$ be the average waiting time of an arbitrary transaction instance in a $M/G/Y_j$ queue. If a service time x has a distribution with mean $1/\mu$ and variance $D(x)$ then $Y_j x$ has mean Y_j/μ and variance $Y_j^2 D(x)$, that is, the squared coefficient of variation is unchanged through scaling. Then, $\delta = W_{q(M/G/Y_j)}/W_{q(M/G/1)}$ is the ratio by which the mean waiting time changes by going from one server to Y_j servers. Assume that this ratio is fairly insensitive to the inter-arrival and service time distributions, i.e., $W_{q(M/G/Y_j)}/W_{q(M/G/1)} = W_{q(M/M/Y_j)}/W_{q(M/M/1)} = \delta$. Then we can approximate $W_{q(M/G/Y_j)}$ as $\delta W_{q(M/G/1)}$. Then $W_{s(M/G/Y_j)} \cong \delta W_{q(M/G/1)} + \dfrac{1}{\lambda_{r_j}} \sum_{i=1}^{n} \dfrac{a_{ij}\lambda_{t_i}}{\mu_i}$. Hence from Little's formula one obtains $L_{q(M/G/Y_j)} \cong \delta L_{q(M/G/1)}$ and $L_{s(M/G/Y_j)} \cong \delta L_{q(M/G/1)} + \lambda_{r_j}/\mu_{r_j}$.

4 Conclusions

Under the framework of MWF-net, the routing of transaction instances in the multi-TWF-nets are projected into the flow of transaction instance between different resource pools. Then, the performance analysis of a workflow model can be transformed as indices calculation of the queuing network in the resource perspective, where each resource pool is a queuing system. Based on the assumptions of invariance for service times, visit ratios, and routing frequencies, which are the properties of many real system, an approximate method for quantitative analysis of a workflow model are given. In fact, there is an additional Service Time Homogeneity (STH) assumption in the operational analysis of queuing network. STH, which is violated in many real system, asserts that the load dependent mean service times have the same values of mean service time. In practice, STH is the dominating factor for error generation. [7] has shown that most performance indices, such as resource utilizations and system response times, are insensitive to the STH. Thus, the method presented in this paper is applicable.

References

[1] J. Q. Li, Y. S. Fan, and M. C. Zhou, Performance Modeling and Analysis of Workflow, under the review of IEEE Trans. SMC: Part A.

[2] K. Salimifard and M. Wright, Petri net-based modeling of workflow systems: An overview, *European Journal of Operational Research*, 134 (2001) pp. 664-676.

[3] W. M. P. van der Aalst. The Application of Petri Nets to Workflow Management. *The Journal of Circuits, Systems and Computers* 8 (1): 21-66, (1998).

[4] Murata. Petri Nets: Properties, Analysis and Applications. *Proceeding of the IEEE*, Vol. 779(4), Appril 1989.

[5] J. Q. Li, Y. S. Fan, and M. C. Zhou. Timing Constraint Workflow Nets for Workflow Analysis. Accepted by IEEE Trans. On SMC: Part B, Oct. 2002. It can be downloaded from http://web.njit.edu/~zhou/smcb-oct02.pdf.

[6] Jin Hyun Son and Myoung Ho Kim, Improving the Performance of Time-constrained Workflow Processing. *Journal of Systems and Software* 2001:211-219.

[7] Peter J. Denning and Jeffrey P. Buzen, The Operational Analysis of Queuing Network Models, *ACM Computing Surveys* (*CSUR*), vol. 10 (3), pp. 255-261, Sept. 1978.

[8] Trivedi K S. Probability & Statistics with Reliability, Queuing, and Computer Science Applications. Prentice-Hall, Inc, 1982

[9] John A. Buzacott, and J. George Schanthikumar, Stochastic Models of Manufacturing systems Prentice-Hall Inc. 1993.

Appendix

An elementary path in $PN = (P, T, F)$, called a path for short, is $(x_1, x_2, \ldots, x_k)$ such that arc (x_i, x_{i+1}) exists $1 \le i \le k-1$, and $x_i = x_j$ implies $i = j$, $1 \le i, j \le$ k where $x_i \in P \cup T$. It is called an (elementary) circuit if $x_i = x_j$, $1 \le i, j \le$ k implies $i = 1$ and $j = k$. $PN_1 = (P_1, T_1, F_1)$ is a subset of $PN = (P, T, F)$, $path = \{t_0, p_1, t_1 \cdots p_m, t_m\}$ in PN is a transition path of PN_1 iff: (1) It is a path; (2) $t_0, t_m \in T_1$; (3) $p_j \notin P_1$, $1 \le j \le m$, and $t_j \notin T_1$, $1 \le j < m$. Symmetrically, $path = \{p_0, t_1, p_1 \cdots t_m, p_m\}$ in PN is a place path of PN_1 iff: (1) It is a path; (2) $p_0, p_m \in P_1$; (3) $t_j \notin T_1$, $1 \le j \le m$, and $p_j \notin P_1$, $1 \le j < m$. A place p in $PN = (P, T, F)$ is a choice place iff $|p^\bullet| \ge 2$. Suppose $PN_1 = (P_1, T_1, F_1)$ is a path or subnet of $PN = (P, T, F)$ and p' is the first or source place. Given a choice place $p \in P_1$, its choice degree relative to PN_1 is defined as the number of choice places in the path from place p' to p in PN_1. Suppose that R is a subnet (path) set, a choice place p is proper iff $\forall$subnet (path) in R, p is not on it or p's choice degree is the least. Since the model decomposition focuses on a TWF-net's structure, only its structure specification $PN = (P, T, F)$ is used in the decomposition algorithm.

Algorithm 1 (Decomposition)

Step 1: Construct the extended net $\overline{PN}=(\overline{P},\overline{T},\overline{F})$ by adding a new transition t between the intial and end places, i.e., ${}^\bullet t=\{\theta\}$, $t^\bullet=\{\varepsilon\}$.

Step 2: Corresponding to an arbitrary choice place $p^1\in P_C=\{p|\ |p^\bullet|\geq 2\}$, a circuit $PN_1=(P_1, T_1, F_1)$ passing p^1 and t is constructed. In addition, $R_1=\{PN_1\}$, and $P_S=\phi$, where P_S is a choice place set and R_1 is a subnet set.

Step 3: Repeat the following steps until $\forall\ PN_j=(P_j, T_j, F_j)\in R_1$, $\Psi_j=\{p|\ p\in P_j\wedge |p^\bullet|\geq 2\wedge p\notin P_S\}$ is empty. Denote the resulting subnet set as $R=R_1=\{PN_1, PN_2,\cdots, PN_n\}$.

3.1: $R_2=R_1$, $PL=\phi$;

3.2: For every $PN_j=(P_j, T_j, F_j)\in R_2$, if $\Psi_j=\{p|\ p\in P_j\wedge |p^\bullet|\geq 2\wedge p\notin P_S\}$ is nonempty, choose the choice place $p\in\Psi_j$ whose choice degree in PN_j is the least one in Ψ_j and $PL=PL\cup\{p\}$;

3.3: For every place $p_\alpha\in PL$, if p_α generated from Ψ_k of $PN_k\in R_2$ belongs to Ψ_m ($m\neq k$) of $PN_m\in R_2$ and the choice degree of p_α is not the least one in Ψ_m, then $PL=PL\backslash\{p_\alpha\}$;

3.4: $P_S=P_S\cup PL$, and for every place $p_l\in PL$ do

For every $PN_j=(P_j, T_j, F_j)\in R_2$, if $p_l\in P_j$, there must be a nonempty transition set $\eta=\{t|t\in p_l^\bullet\wedge t\notin T_j\}$. Then, $|\eta|$ place paths $path_i=\{p_l, t_i, \cdots, p_{ie}\}$ ($1\leq i\leq|\eta|$), where only p_l and p_{ie} belong to PN_j and $t_i\in\eta$, are constructed. If there are c place paths (among the $|\eta|$ place paths) where p_{ie} located in a path from p_l (not including p_l) to θ, using each of the c place paths to replace the corresponding path from p_l to p_{ie} of the original PN_j respectively, c new subnets PN_w ($1\leq w\leq c$) are obtained and added to R_1. For the rest $|\eta|$-c place paths where p_{ie} is located in a path from ε to p_l (including p_l), merging all of them to PN_j and c new PN_w as PN_j' and c PN_w', then using PN_j' and c PN_w' to replace the corresponding original subnets in R_1.

Step 4: Repeat the following exhaustively:

For every $PN_k=\{P_k, T_k, F_k\}\in R$, if there is $t'\in T_k$ with a nonempty set $\{p|\ p\in t'^\bullet\wedge p\notin P_k\}$, then for every $p_1\in t'^\bullet\backslash P_k$, an arbitrary transition path $path=\{t_0, p_1, t_1\cdots p_m, t_m\}$ is constructed and merged into PN_k, where $t'=t_0$, $t_m\in T_k$. However, if there are choice places in the merged $path$ for some new PN_ks, all these PN_ks are deleted from R and collected into R'_1, and the following steps are repeated until $\forall\ PN_j=(P_j, T_j, F_j)\in R'_1$, the set $\Psi'_j=\{p'|\ p'\in P_j\wedge|\ p'^\bullet|\geq 2\wedge p'\notin P_S\}$ is empty. Then the result R'_1 is merged into R.

4.1: $R'_2=R'_1$, $PL'=\phi$;

4.2: For every $PN_j=(P_j, T_j, F_j)\in R'_2$, if $\Psi'_j=\{p'|\ p'\in P_j\wedge|\ p'^\bullet|\geq 2\wedge p'\notin P_S\}$ is nonempty, each p' must belong to a path from t_0 to t_m. Choose $p'\in\Psi'_j$ whose choice degree relative to the path from t_0 to t_m is the least one in Ψ'_j, and let $PL'=PL'\cup\{p'\}$;

4.3: For every place $p'_\alpha\in PL'$, if p'_α generated from Ψ'_r of $PN_r\in R'_2$ belongs to Ψ'_s ($r\neq s$) of $PN_s\in R'_2$ and the choice degree of p'_α is not the least one in the path from t_0 to t_m (in Ψ'_s), then $PL'=PL'\backslash\{p'_\alpha\}$;

4.4: $P_S=P_S\cup PL'$, and for every place $p'_l\in PL'$ do

For every $PN_j=(P_j, T_j, F_j)\in R'_2$, if $p_l'\in P_j$, a nonempty post transition set $\eta'=\{t'|t'\in p_l'^\bullet(\text{in } PN)\wedge\ t'\notin T_j\}$ exists. Then, $|\eta'|$ place paths $path_i'=\{p_l', t_i', \cdots, p_{ie}'\}$ ($1\leq i\leq|\eta|$), where only p_l' and p_{ie}' belong to the transition path from t_0 to t_m in PN_j and $t_i'\in\eta$, are constructed. If there are c' paths where p_{ie}' does not located in $\{t_0, p_1, t_1\cdots, p_l'\}$, using each of the c' place paths to replace the corresponding part from p_l' to p_{ie}' of the transition path from t_0 to t_m in PN_j respectively, c' new subnets $PN_{w'}$ ($1\leq w'\leq c'$) are obtained and added to R_1. For the rest $|\eta'|$-c' place paths, merging all of them to PN_j and c' new $PN_{w'}$ as PN_j' and c' $PN_{w'}'$ replacing the corresponding original subnets in R'_1.

Step 5: Deleting transition t and its corresponding arcs (θ,t) and (t,ε) from each subnet in R.

In Step 1, the extended MWF-net is constructed. Step 2 is used to construct a circuit PN_1 passing transition t and an arbitrary choice place in PN. In each iteration of Step 3, some choice places of $P_C\backslash P_S$ are investigated. After Step 3.3, all the proper choice places in $P_C\backslash P_S$ of current iteration are selected into PL. Step 3.4 first generates all the new subnets according to the $path_i$ that is a part of ACS, then merges the rest $path_i$ that is a part of ICS to the corresponding old subnets in R_1. After Step 3, all the subnets (generated from PN_1), each one of which can be extended to an extended FC-subnet by the application of Step 4, are found. Step 4 extends every subnet in R from its synchronous transitions (whose output places are more than one in PN), Steps 4.1-4.4 (similar to Steps 3.1-3.4) construct and merge all the new subnets that are derived from the choice places in the new merged transition paths of PN_j. Then the corresponding extended FC-subnets are obtained. Step 5 deletes the transition t for each extended FC-subnet and the resulting $R=\{PN_1, PN_2,\cdots, PN_n\}$ is obtained. Each element PN_i of R, where there are only ICSs, CCSs and SCSs, corresponds to the routing path of a specific type of transaction instance in a TWF-net.