Petri Nets Based Reliability Evaluation Of Services

Jiajun Xu

School of Computer and Engineering
Beihang University
Beijing, China
e-mail: xujiajun@cse.buaa.edu.cn

Shuzhen Yao

School of Computer and Engineering
Beihang University
Beijing, China
e-mail: szyao@buaa.edu.cn

Abstract—**In this paper, a new method is proposed to measure the reliability of service composition. The service composition is related to structure and interaction associated. This paper first studies structure associated service composition description language BPEL; while to the service interaction, it models the interactive features by applying Petri Net. It gives the definition of Composited Service Stochastic Petri Net, and builds the reliability evaluation model and method, containing structure and interaction associated service composition evaluation and simplification method. The approach is validated through a case and further research directions are pointed out.**

Keywords-Service Composition; Petri Net; reliability evaluation

I. INTRODUCTION

Web services are emerging as a major technology for deploying automated interactions between distributed and heterogeneous applications[1-3]. Web service composition is a distributed model to construct new web services on top of existing primitive or other composite web services, through a certain association. The association generally includes two categories: structural association and interaction association. Structural associated composition includes sequence, switch, loop, and parallel logical relationship; interaction is involved in the service composition between requester and provider. Both of them have impacts on the reliability of web service. However, current researches address only one aspect, leading to the problem that the evaluation mechanism is not scientific and comprehensive enough. This paper will focus on the structural associated and interaction associated, construct composited service model, and create an evaluation mechanism based on the Petri Nets.

II. RELATED WORK

In the research field of reliability evaluation, there have proposed a variety of analysis techniques. Among which, Petri Net with the characters of concurrency, asynchronous, and uncertainty is a strong dynamic modeling and analysis tool. It can not only be used to describe the local behavior, but also the combination characteristics, for modeling and analyzing the reliability of the system.

BPEL has become the main description language that is related to the structure. Comparing to the interaction associated, there are several major typical method for modeling of the interaction: process algebra [4], STD (State Transition Diagrams) [5], DFA (Deterministic Finite Automata) [6], π-calculus [7], Petri Net [8], Fuzzy Petri Net[9], CPN(Colored Petri Nets)[10] and some other techniques and methods.

There has been a lot of work focused on the modeling and analysis of BPEL based on Petri Nets. Hinz et al proposes a tool BPEL2PN to transform BPEL to Petri Net. The tool could support a variety of transformation operations (including standard, and exception handling related behavior), and could be used for understanding the BPEL inner semantics[11]. Ouyang et al describes another tool called BPEL2PNML which focus on the control structure of BPEL; the Petri Net model can be used after transformation for testing unreachable and conflict[12]. Malhotra et al proposes a transformation tool called BPEL2oWFN which is mainly used for analysis control flow related properties of BPEL[13]. Another part of the method relies on high-level Petri Nets, such as Zhu et al proposed a CPN based method to verify the Web service composition[10]. The work proposed in these papers provides valuable reference for our Petri Nets based modeling and analyzing methods.

In the following, the paper would focus on service composited Petri Net model from viewpoint of the structural associated and interaction associated, apply on colored Petri Nets and stochastic Petri Net theory to perform reliability evaluation of services, and then come with examples to verify the usability and correctness of the method.

III. CPN-BASED SERVICE COMPOSITION MODEL

A. CPN-based Structure model

BPEL is used to describe the behavior and composition of Web services, and its pattern contains basic activities and structured activities. Each defined activity in its pattern can be abstracted into a Petri Net model [11]. In this paper, the CPN based formal representation of the BPEL process is an expansion of it. In order to clearly express the control flow of the BPEL the color set in the model and to specify the color of the arc extension and limited.

Definition 1. (CPN based formal representation of the BPEL): The formal representation of the BPEL process is a CPN: N=(P,T,F,C,W,M0,λ),

1) P is a finite set of places. P= $P_I \cup P_m$, and $P_I \cap P_m = \varnothing$, P_I is the set of internal places. P_m is the set of message places.(Internal places P_I represent the internal control logic, and message places P_m represent the messages exchanged between services.)

2) T is a finite set of transitions. F = F1 $\cup$ Fm, and F1$\cap$Fm=$\varnothing$,F1 is the normal arc whereF1$\subseteq$(P1$\times$T)$\cup$(T$\times$P1), and Fm is the arc from Pm(message arc) Fm$\subseteq$FmR $\cup$ FmS.

All the arc is associated with a color set C. FmR⊆(Pm×T) means arc of receiving message, and FmS⊆(T×Pm) means arc of sending message. (P and T is disjoint from each other, P∩T=∅).

3) C is a color set, C:P∪T→ΣC, it link each Place P to a color set C(p), and link each Transition to a color set C(t), which means all the token in P and T belongs to a color type and stands for message; each color in C stands for a message.

4) W:F→(ID,CDS) means the function on the arc, ID means the identity of the Arc, CDS means the data structure or object of service interface.

5) M_0 is the initial marking of CPN.

6) λ is the firing rate set on the transitions, $\lambda_i \in \lambda$ is the firing rate of transition t_i.

The BPEL activities can be divided into two categories: basic activities and structured activities. *<receive>*, *<reply>*, *<invoke>*, *<sequence>*, *<if>*, *<flow>*,, *<while>*, and *<repeatUntil>*, are key elements to describe the control logic of BPEL.

Our work tackles the issue of service composition in structure aspect, it does not consider BPEL constructs such as *compensation, fault handler, assign,* and *correlation set, link condition, variables, and so on.*
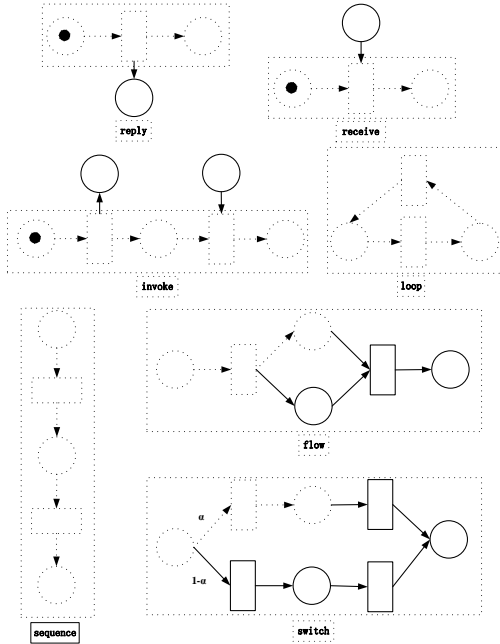

Figure 1.   Transforming BPEL into CPN.

Fig. 1 illustrates how to transform some BPEL elements into Petri Nets. BPEL activities and internal control logic are modeled with internal places and transitions; the messages exchanging are modeled with message places.

In BPEL, a process is built up by plugging language constructs together. Therefore we translate each pattern of BPEL into a Petri Net. Such a net forms a pattern of the respective BPEL construct. Each pattern has an interface for joining it with other patterns as it is done with BPEL

structures. In the CPN definition of BPEL process, each pattern has only one source place, standing for starting and only one sink place for ending.

B.   CPN-based Interaction model

To interaction model, the service could be modeled to as simple input and output, and this paper uses Petri Nets to model loosely coupled web service, and to evaluation the reliability of service interactive models.

The web service interactions are among the available services or user interactions; former will build a number of available services as a whole, the latter is used to satisfy the service requester. From the viewpoint of interaction protocols, the two can be unified as the service requester and service by establishing session. A session is a set of associated meta message m. The session(m1, m2, ..., mn) represents a set of meta message m1, m2, ..., mn exchanging.

To describe the reliable interaction of service-oriented systems, the following definition is introduced.

Definition 2. CSSPN (Composited Service Stochastic Petri Net). Given a CPN of requester model *Requester*=(P_1, T_1, F_1, C_1, W_1, $M_{1,0}$, λ_1) and a CPN of provider model *Provider*=(P_2, T_2, F_2, C_2, W_2, $M_{2,0}$, λ_2). After the commutation of *Requester* and *Provider*, the Petri Net after interaction CSSPN=*<Requester, Provider, Session>*.

CSSPN is based on the concept of CPN and stochastic Petri Nets. In the Places of CSSPN, data type of token is defined, to stand for complex data structures or objects or service message and reduce the complexity of modeling. We define token to be message, and the arc means that the message passing mechanism. CSSPNs interact through the input the arc output interface with the outside world or other CSSPN. Transition means the message processing unit. CSSPN is very suitable to describe the distributed, parallel system, and it not only can describe the static model of service interaction protocols, but also dynamic interaction, to provide support for the reliability analysis.

Different types of service requests reflect the differences in cooperation, thereby forming different kinds of cooperative interaction. The composited service is obtained by combining the new service by a plurality of basic services in accordance with certain logical structural relationship. However, because actual deployed service applications interact through the session. Thus, when the session state changes, it would lead to a greater difference in performance. In this case, if it does not consider the impact of session state in the service composition modeling, the reliability of the services will result in a large deviation.

Fig. 2 represents a single service requester/provider session model, in which places mean the state, and transitions mean processing of the message. In the session of request/response protocol: When the service requester makes a request to invoke a service, the session begins; the service provider receives the message from the requester and makes a reply, after the processing is completed. The transition T_LOST represents that interactive messages are lost in the network environment, and it goes to place P_LOST. The transition T_NORMAL represents that interactive messages are transmitted normally, and the transition T_UNMATCH

represents that the detected message exchange sequence is a departure from a defined standard message exchange sequence. The firing rate of T_LOST, T_UNMATCH, T_NORMAL, and T_TRANSTIME are λ_1, λ_2, λ_3, and λ_4, respectively. Thus, the probability of loss of messages in the network environment $lost_{MSG} = \lambda_1/(\lambda_1+\lambda_3)$. The firing rate of T_UNMATCH is λ_2. The probability of failure of the system interaction $fail_{MSG}=(\lambda_1+\lambda_2)/(\lambda_1+\lambda_2+\lambda_3)$.
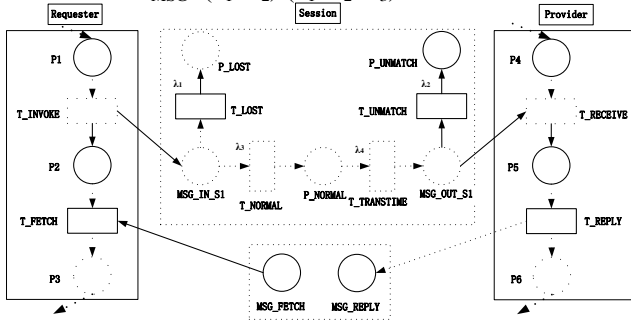

Figure 2.   Single service requester/provider session model.

For reliability analysis of the system, after the service composition and service request/provide interaction models are built, the service node failure model is considered, and finally a system performance model containing various services structures and interactions is established to perform the evaluation.

IV.   RELIABILITY EVALUATION

In service-oriented system, reliability is the probability of the normal execution of the system. To analyze the reliability of the entire system, even if a single service is very reliable, the overall reliability of the system may also be damaged in the service composition. The main reason is that the BPEL coordinates multiple separate services, thus, the operation of many services composition need to maintain integrity and reliability between multiple separate services.

A.   Reliability Analysis of Service Node

This paper gets the structured information of BPEL, and then according to the structural properties, the reliability of composited service is calculated from the reliability of basic services (non-repairable and repairable systems).

The service reliability can be broadly divided into two categories: one for non-repair system; another repairable system. As Shown in Fig. 3, MSG_INVOKE_IN means the received message, and MSG_INVOKE_OUT means the returned message. Inside the box, the model represents the state of the system processing. The received token reaches place UP. Place UP and DOWN means that the service is running in normal or failure state. A λ and a μ represent the failure rate and the recovery rate of a service. For non-repair system, the system recovery rate μ is 0; the node can not be recovered.
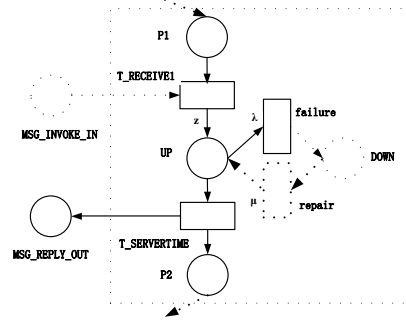

Figure 3.   Service failure/repair model.

1)   Reliability analysis of non-reparable service node

The so-called non-repairable system is the service that once enabled, until failure. Such a system only has two states: one for running, another for failure, and will not be repaired after a failure.

For non-repairable system, according to the definition of system reliability, let the failure rate λ, and time t, the service node reliability R_{SN} is:

$$R_{SN} = \exp\left(-\int_0^t \lambda(t)dt\right) = e^{-\lambda t} . \tag{1}$$

2)   Reliability analysis of reparable service node

For repairable system, error leads to the failure of the service system. Usually the fault-tolerant method is to repair the service from the failure state, and the relationship between the failure and repair can vary widely, and the failure model is a Poisson process[14-15], and it is usually assumed that each failure event along with a repair process.

For repairable system, according to the definition of reliability of the system, let the failure rate λ, repair rate μ, and it has nothing to do with t, the service node the reliability R_{SN} is:

$$R_{SN} = \frac{u}{\lambda(t)+u} + \frac{\lambda(t)}{\lambda(t)+u}\exp\left(-\int_0^t (\lambda(t)+u)dt\right) = \frac{\mu}{\lambda+\mu} . \tag{2}$$

B.   Reliability Analysis Of Service Session

Services interact in the Network between service deliveries. In the process of transmission, because of network problem or unmatched of service interface, it could lead to the problem like loss of information, the fails or errors of session, therefore, the same service may show much difference in reliability.
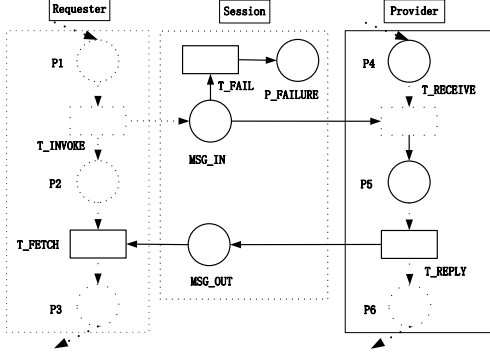
Figure 4. Simplified session model.

In order to reduce the scale of the model, so as to reduce the state of the system, at the same time to make it easy to set model parameters, we propose a simplified session model. As shown in Fig. 4, the places MSG_IN_S1, P_NORMAL, and MSG_OUT_S1, and transitions T_NORMAL and T_TRANSTIME are merged to place MSG_IN. Transitions T_LOST and T_UNMATCH are merged to transition T_FAIL. P_LOST and P_UNMATCH are merged to place P_FAILURE. The failure probability of the session interaction $p_f = (\lambda_1+\lambda_2)/(\lambda_1+\lambda_2+\lambda_3)$; the success probability of success for the $p_S = \lambda_3/(\lambda_1+\lambda_2+\lambda_3)$. Thus, the reliability of service interaction R_{SI} is $\lambda_3/(\lambda_1+\lambda_2+\lambda_3)$.

C. Rules to Compute Reliability

Complex system can be decomposed into simple subsystems. If the failure rate λ and the repair rate μ of a subsystem are known, it can be used to calculate the reliability of complex system, without the need to directly test the normal operation of complex system. Complex system depends on the normal operation of the subsystem.

Definition 3. Service Execution Reliability. Service execution reliability is the production of the reliability of service node and the reliability of service interaction. The service execution reliability of the system is:

$$R_{SE} = R_{SN} \times R_{SI}. \tag{3}$$

For multi-services system model, when the reliabilities of the services are independent, the system reliability could be computed from the independent service reliabilities. In most real systems, especially service-oriented system, there is a significantly structure dependency in the functions of services. These dependencies can be simulated by Petri Nets, to model service synchronization and collaboration. Base on the single service model of the previous section, the computation of reliability of complex system with sequence, switch, loop, and parallel structure are described.

Theorem1. Suppose service system B is composed of n sequence services, the reliability of n services are n independent random variables, and their parameters are as $R_{SE,1}, R_{SE,2}, ..., R_{SE,n}$, the equivalent reliability R_{SE} of the n sequence services is:

$$R_{SE} = \prod_{i=1}^{n} R_{SE,i}. \tag{4}$$

Theorem2. Suppose service system B is composed of n switch services, the reliability of n services are n independent random variables, and their parameters are as $R_{SE,1}, R_{SE,2}, ..., R_{SE,n}$, and the probabilities of the executives are $\rho_1, \rho_2, ..., \rho_n$, $\sum_{i=1}^{n}\rho_i = 1$, the equivalent reliability R_{SE} of the n switch services is:

$$R_{SE} = \sum_{i=1}^{n} \left(\rho_i R_{SE,i}\right). \tag{5}$$

Theorem3. Suppose service system B is composed of one loop service, the reliability of the service is $R_{SE,1}$, and the service is repeated n times, the equivalent reliability R_{SE} is:

$$R_{SE} = R_{SE,1}\left(\alpha R_{SE,1}\right)^n = \alpha^n R_{SE,1}^{n+1}. \tag{6}$$

It is more difficult to compute the reliability of the parallel structure, because this structure contains *AND* parallel and *OR* parallel structure. In *AND* parallel structure, all services should be running, while the *OR* parallel structure only requires a few of these services. In this paper, AND parallel and *OR* parallel would be discussed.

In *AND* parallel structure, it means that only when all the services of the parallel structure are running properly, the entire parallel structure could work normally, which often appears in the case of concurrent actions taken to improve performance, thus the Theorem 1 could be applied.

Theorem4. Suppose service system B is composed of n OR parallel services, the reliability of n services are n independent random variables, and their parameters are as $R_{SE,1}, R_{SE,2}, ..., R_{SE,n}$, the equivalent reliability R_{SE} is:

$$R_{SE} = 1 - \prod_{i=1}^{n}\left(1 - R_{SE,i}\right). \tag{7}$$

V. CASE STUDY AND ANALYSIS

In order to better show the availability of the evaluation method, we use the case of the TSS (Tourism Services System). As shown in Fig. 5 it gives BPEL model of TSS. First, the user sends a request. Secondly, TSS, including airfare, hotel and tour services, receives the request and starts travel planning process. The service provider provides three concurrent tasks, which would interact with hotel center service, ticket center service, and tour center service (three services can be executed in parallel). Then the system receives the results messages of three concurrent services. Three services would be invoked. After the join of three services, it invokes another service provider BankCredit services, standing for the credit card payment process; then

TSS receives the results of the BankCredit information. Finally, after confirmed by the end-user, the process ends.
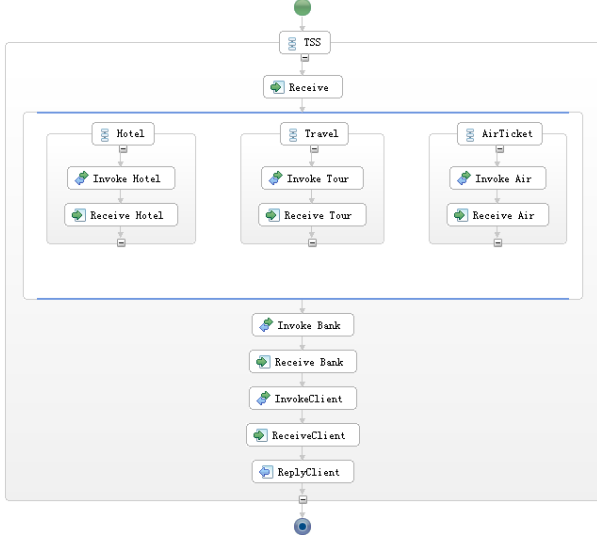

Figure 5.   BPEL model of TSS

The case contains two kinds of typical structural activities (flow and sequence) and three basic activities (invoke, receive and reply); the TSS receives the request from the client, and coordinates the four services (Hotel, AirTicket, Tour and Bank). The process involves two structures and four interactions

The TSS provides a complete functional service, but because services that have the same function make a different in reliabilities, it is difficult to evaluate the overall system reliability. Given BPEL of TSS, we first created CSSPN model including system structure and interaction associated, as shown in Fig.6. It contains the user requests, three concurrent services Hotel, AirTicket and Tour, and BANK. The dotted line box INVOKE_HOTEL, INVOKE_TICKET,INVOKE_TOUR, and NVOKE_BANK represent the invoking of the hotel services, ticketing services, tour services and bank credit card services. A Firing identifies a traveling of message on the arc, and receiving message indicates the start of this behavior and replying the message indicates the completion.

In order to verify the correctness and effectiveness of our reliability evaluation method, we use the example represented in Fig. 6 to calculate the reliability of the system, and the results would be analyzed.

In this example, the structure of the services composition includes three parallel services, one sequence and four service sessions. The reliability of three parallel services Hotel, AirTicket, and Tour could be computed by applying the formula in Theorem 1; three parallel services and Bank Services are in sequence, the reliability can be calculated y applying Theorem 1, too. To get interaction reliability of the four services, the method of the previous section is referenced. In the example, the structure of the service composition are given, we set the failure rates and repair rates of service nodes and the reliabilities of service interactions. The numerical analysis is divided into three kinds of situations, and the results would be analyzed and discussed.
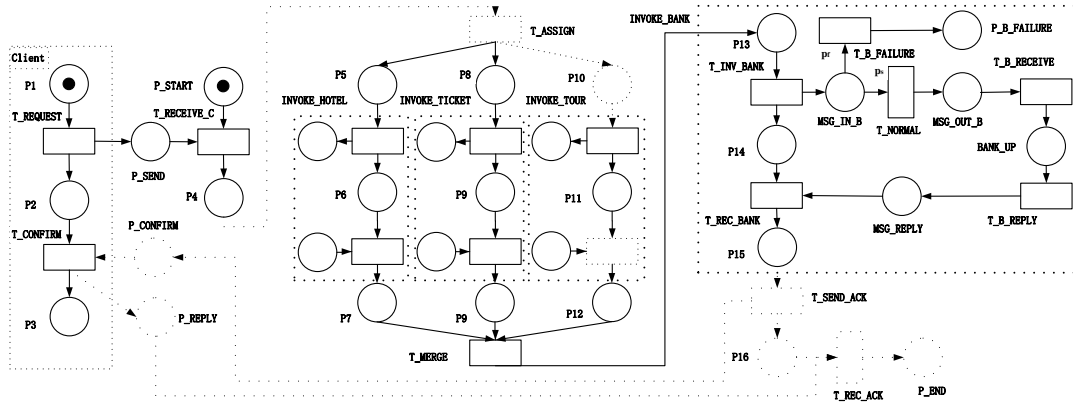

Figure 6.   CSSPN model of TSS

Experiment l: to perform service composition reliability evaluation of the four services, the reliability evaluation results obtained are shown in Table I.

TABLE I.   RELIABILITY EVALUATION VALUE OF EXPERIMENT 1

	HOT	TIC	TOU	BAK	PAR	PAR	TSS
failure	0.1	0.1	0.1	0.1	/	/	/
recover	0.9	0.9	0.9	0.9	/	/	/
R_{SI}	0.98	0.98	0.98	0.98	/	/	/
R_{SE}	0.88	0.88	0.88	0.88	0.73	0.9	0.61

Experiment 2: on the basis of Experiment 1, certain interference is added for reliability evaluation of the service composition system, the system reliability results obtained are shown in Table II.

TABLE II. RELIABILITY EVALUATION VALUE OF EXPERIMENT 2

	HOT	TIC	TOU	BAK	PAR	PAR	TSS
failure	0.2	0.1	0.1	0.1	/	/	/
recover	0.8	0.9	0.9	0.9	/	/	/
R_{SI}	0.98	0.98	0.98	0.98	/	/	/
R_{SE}	0.78	0.88	0.88	0.88	0.65	0.9	0.54

Experiment 3: on the basis of experiment 2, certain interference is added for reliability evaluation of the service composition system; the system reliability results obtained are shown in Table III.

TABLE III. RELIABILITY EVALUATION VALUE OF EXPERIMENT 3

	HOT	TIC	TOU	BAK	PAR	PAR	TSS
failure	0.2	0.1	0.1	0.2	/	/	/
recover	0.8	0.9	0.9	0.8	/	/	/
R_{SI}	0.98	0.98	0.98	0.9	/	/	/
R_{SE}	0.78	0.88	0.88	0.72	0.65	0.8	0.44

According to the above experimental results, similar to the sequential and *AND* parallel structure have the same effect to the system reliability, this is because in the given service structural and interaction model, the services are all executed correctly to make the system work given.

We give a reliability evaluation method for system reliability calculation, and it provides the basis and foundation of the reliability evaluation. The experimental results verify the reasonableness and usability of the method that we have proposed in this paper.

VI. CONCLUSION AND FUTURE WORK

The study of service reliability evaluation is to become a research hotspot. The modeling and analysis of service structures and interactions in the open network environment is a feasible and effective method for reliability evaluation. It is a topic worthy of further study, and might play a more important role in future service oriented computing.

In this paper, a CPN-based services reliability evaluation model and method could easily be used for composited service system. Based on this method, the definition of composited services stochastic Petri Nets is given; the modeling and reliability evaluation method are proposed. All these will be the basis of our follow-up work.

This further work includes considering the reliability evaluation methods when service node has the fault-tolerant backup mechanism; the work needs to be further used in the actual project to verify the effectiveness and feasibility of the method.

REFERENCES

[1] Hull R, Su J W. Tools for composite web services: a short overview. SIGMOD Rec, 2005, 34: 86-95

[2] Fan W, Geerts F, Gelade W, et al. Complexity and composition of synthesized web services. In: Lenzerini M, Lembothe D, eds. Proceedings of 27th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. Vancouver BC: ACM, 2008. 231–240

[3] DENG Ting,HUAI JinPeng, et al. Complexity of synthesis of composite service with correctness guarantee. Scientia Sinica Informationis, 2012, 42(7): 789-802..

[4] Ferrara A．Web services：A process algebra approach[c]//Proc of the 2nd Int Conf on Service Oriented Computing(ICSOC' 04). Piscataway, NJ: IEEE, 2004: 242-251

[5] Ralf König.State-Based Modeling Method for Multiagent Conversation Protocols and Decision Activities.Agent Technologies, Infrastructures, Tools, and Applications for E-Services, volume 2592 of Lecture Notes in Computer Science.Springer, 2003:151-166.

[6] Lihui Lei,Zhenhua Duan. Automating web service composition for collaborative business processes.11th International Conference on Computer Supported Cooperative Work in Design. 2007:894-899.

[7] Roberto Lucchi , Manuel Mazzara. A pi-calculus based semantics for WS-BPEL. The Journal of Logic and Algebraic Programming. 2007, 70(1): 96–118.

[8] Zhang Jia , Chung J en Yao , Chang C. K. , Kim S. . WS2Net : A Petri Net based specification model for web services. In : Pro ceedings of t he 2nd IEEE International Conference on Web Services, San Diego, California , USA , 2004 , 420-427.

[9] Rui Fu; Weiqing Dong; Guang Yang; Yiduo Mei; Xiaoshe Dong, Fuzzy Petri Net-Based Optimized Semantic Web Service Composition, Grid and Cooperative Computing, 2008. GCC '08. Seventh International Conference on , vol., no., pp.496,502, 24-26 Oct. 2008

[10] Zhu Jun,Guo Changguo. et a1. A Runtime Monitoring Web Services Interaction Behaviors Method Based on CPN. Journal of computer research and development. 2011, 48(12): 2277-2289

[11] Hinz S,Schmidt K,Stahl C.Transforming BPEL to Petrinets[c]//Proc of the 3rd Int Conf on Business Process Management(BPM' 05). Piscataway, NJ, IEEE-2005:220-235

[12] Ouyang C,Verhaek E,van der Aalst W,et a1. Formal semantics and analysis of control flow in WS-BPEL. Science of Computer Programming. 2007, 67(2/5):162-198

[13] Lohmann N, Massuthe P, et a1.Analyzing interacting WS-BPEL processes using flexible model generation. Data & Knowledge Engineering, 2008,64(1): 38-54.

[14] Malhotra M. Triveddi KS. Dependability modeling using Petri Nets. IEEE Trans Reliab. 1995, 44(3):428-440.

[15] Simona Bernardi, Andrea Bobbio, Susanna Donatelli. Petri Nets and Dependability.Lectures on Concurrency and Petri Nets/Lecture Notes in Computer Science. 2004: 125-179.

.