A Method for Analyzing and Predicting Reliability of BPEL Process

Liqiong Chen
Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai, China
Email: qiong_1207@126.com

Zhiqing Shao, Guisheng Fan and Hanhua Ma
Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai, China
Email: zshao@ecust.edu.cn, gs_fan@126.com, mhanhua@163.com

***Abstract*—As an important means to compose independent services together to fulfill a function, service composition is widely applied in different applications. However, the process of composition is complex and error-prone, which makes a formal modeling and analysis method highly desirable. A BPEL process based on Petri net (BPEL-Net) model is presented in this paper, which is capable of capturing behavior of the participating services accurately. A set of rules are proposed to convert atomic activity and structural activity of BPEL process into BPEL-Net model, the transactional properties of services and failure processing between services are also characterized by BPEL-Net. Based on the states of constructed BPEL-Net, we advance the concept of transfer matrix to analyze reliability and related properties. What's more, we put forward two simplification schemas for BPEL-Net. Finally, a specific example is given to simulate analytical process with tool Matlab, the results show that the method can be a good solution to analyze the reliability of BPEL processes.**

***Index Terms*—Service composition, BPEL, transactional properties, transfer matrix, Petri net, Matlab**

I. Introduction

Service Oriented Computing (SOC) is an approach to distributed computing that views software resources as dynamically discoverable services available on the internet. Web services are a well known and widely used technology for implementing the SOC, which should facilitate integration of newly built and legacy applications both within and across organizational boundaries [1]. With the development of theories and techniques of Web service, a single service often can't satisfy the functional requirements in practical application. In order to increase service sharing, it is necessary to compose Web service to provide a more powerful service. The result of such composition displays an automated implementation process [2]. Among the results, BPEL is the most popular one which is used to specify business processes for service composition.

The BPEL process contains elements for modeling the structure of business processes. A BPEL process realizes all its communication as Web Service, providing a simple, standards-based approach to describe Web services of real world as well as performing invocations of Web Services [3]. However, errors in BPEL process are easily expanding, and service itself has complex interface. Moreover, black-box nature of service becomes an obvious obstacle to service composition [4]. Therefore, there is a growing interest for analyzing related performance for BPEL process, which can detect possible errors before execution.

A BPEL process based on Petri net (BPEL-Net) model is proposed in this paper. It is used to analyze and predict the reliability of BPEL process. Firstly, we describe the activities of BPEL process which include atomic activity and structural activity. We define an extensible set of crucial transactional properties (such as initial, abort, cancel and fail) for each service and declaratively specify the failure processing strategies between service. Secondly, based on the states of constructed BPEL-Net, we advance the concept of transfer matrix, a technique for analyzing the reliability and related properties of BPEL-Net model. We also put forward two simplification schemas for transfer matrix, and a theorem is given to explain the soundness of simplification.

The remainder of this paper is organized as follows: Section 2 introduces the related concepts of BPEL-Net model. In Section 3, we construct the corresponding BPEL-Net model for BPEL process. Section 4 presents the concept of transfer matrix and analyzes the reliability of BPEL-Net model. In Section 5, we explain the feasibility and practicability of our methods by a specific example. Section 6 presents some related works while Section 7 is conclusion. Your goal is to simulate the usual appearance of papers in a Journal of the Academy Publisher. We are requesting that you follow these guidelines as closely as possible.

This work was partially supported by the NSF of China under grants No.60373075, the Science-Technology Development Foundation of Shanghai of China under Grant No.06dz15004-1.

II. BASIC CONCEPT AND DEFINITION

Petri nets are a well-known modeling technique that has formal semantics, which can be used to model and analyze several types of systems including concurrent, asynchronism and distribution. The basic concepts of it can refer to [5]. First, we will introduce the related concepts, definitions and terms of *BPEL-Net*.

Definition 1: A 5-tuple $PN = (P; T; F; W; M_0)$ is called Petri nets iff:

(1) $P = \{p_1, p_2, ..., p_n\}$ is a finite set of place, n≥ 0, which represents the state of *BPEL process*;

(2) $T = \{t_1, t_2, ... , t_m\}$ is a finite set of transition, $m \geq 0$ and $P \cup T \neq \varnothing$, $P \cap T = \varnothing$, which represents an operation of BPEL process;

(3) $F \subseteq (P \times T) \cup (T \times P)$ is a directed arc set, F is called flow function;

(4) $W : F \rightarrow N^*$, N^* is the non-negative integral set, W is called the weight of arc, whose default value is 1;

(5) $M_0 : P \rightarrow N^*$, M_0 is called initial marking, which represents the initial state of BPEL process.

Definition 2: A 4-tuple $\Sigma = (PN; P_s, P_e, \lambda)$ that meets following conditions is called BPEL-Net:

(1) PN is a Petri Net, known as base net of Σ, which describes the relation between services;

(2) P_s, P_e represent the beginning and termination position of BPEL process respectively;

(3) λ: $T \rightarrow R_0$ is firing probability of transition, which describes the success probability of service, and default value is 1.

In BPEL-Net model, the place represents the position of process, while transition represents the operation of process. At a moment, the distribution of token in each place is called BPEL-Net 's marking (state), the number of token in place p is denoted by M(p); for any x $\in (P \cup T)$, the set ${}^\bullet x = \{y | y \in (P \cup T) \wedge (y, x) \in F\}$ and $x^\bullet = \{y | y \in (P \cup T) \wedge (x,y) \in F\}$ are the corresponding input and output of x respectively. The enabled of transition and firing rules of BPEL-Net model are similar to the definition of Petri nets.

III. CONVERTING BPEL PROCESS INTO BPEL-NET

BPEL process describes service composition based on process, each element in process is called activity (As activity and service is one-to-one match, activity is also called service in the following). We will first construct BPEL-Net model of atomic service and structural service, and describe the failure processing between services for BPEL process in detail.

A. Converting Atomic Service into BPEL-Net

Atomic service can implement simple task, in this paper, we divide it into canceled services and not canceled services according to the transactional properties of services. The canceled service refers that the system can cancel its operation once other services have failed. While not canceled service refers that the system can not cancel its execution, and it must be coordinated to reach termination state and not generate any output. Below we will model these two types of service, the possible states of service WS_i are initial, active, end, abort, cancel and fail, and denoted by places $P_{in,i}$, $P_{ab,i}$, $P_{c,i}$, $P_{e,i}$, $P_{ca,i}$, $P_{fa,i}$ respectively. Let $\lambda_{e,i} = SP_i$, $\lambda_{fa,i}$ = 1- SP_i; SP_i represses the success probability of service WS_i. The dotted places denoted by line in figure refer the possible input or output. The function of place $P_{ac,i}$ is to make service be in a sole state at any moment. While place $P_{tr,i}$ represents the failure processing interface of service WS_i. If place $P_{tr,i}$ has tokens, which means other services have failed and must do the corresponding processing for service WS_i such as abortion and cancellation. The model of canceled service is shown in Fig. 1(a), the dotted frame represents the normal processing, for example, transitions $t_{in,i}$, $t_{e,i}$ represent the beginning and termination operation of service WS_i. Abortion, cancellation and fail operation are denoted by transitions $t_{ab,i}$, $t_{ca,i}$, $t_{fa,i}$ respectively. The model of not canceled service is shown in Fig. 1(b). In this model, if service is in active state and other service has failed, the system will coordinate service to reach the termination state, but transition $t_{ce,i}$ couldn't generate any output.

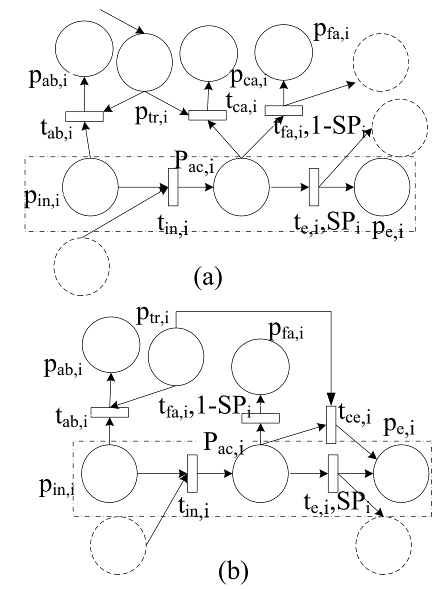

Figure 1. BPEL-Net of Service WS_i

B. Converting Structural Services into BPEL-Net

The structural relations of BPEL process including <*Sequence*>, <*While*>, <*Flow*>, <*Switch*> and <*Pick*>, namely the relations between services are sequence, loop, parallel and choice, the corresponding BPEL-Net model is shown in Fig. 2.

The <*Sequence*> relation refers those two services WS_i and WS_j can execute in sequence. The corresponding BPEL-Net model Σ is shown in Fig. 2(a). Service WS_j can be fired only after executing service WS_i, then we add a place P_{ij} between the termination operation $t_{e,i}$ of service WS_i and accessing resource operation $t_{in,j}$ of service WS_j , which makes ${}^\bullet P_{ij} = t_{e,i}$, $P_{ij}{}^\bullet = t_{in,j}$.

The <*While*> relation refers that service WS_i can execute n times repeatedly (where n is finite). Assuming service WS_i has executed n times repeatedly before the beginning of service WS_j, then the corresponding BPEL-

Net model Σ is shown in Fig. 2(b). We introduce the place $P_{w,i}$, P_{ij} and transition $t_{w,i}$, which make ${}^{\bullet}P_{w,i} = t_{e,i}$, $P_{w,i}^{\bullet} = {}^{\bullet}P_{ij} = t_{w,i}$, $P_{ij}^{\bullet} = t_{in,j}$, $t_{e,i}^{\bullet} = \{P_{in,i}, P_{w,i}\}$, ${}^{\bullet}t_{ij} = P_{w,i}$, $W(P_{w,i}, t_{w,i}) = n$.

The <*Flow*> relation refers that one or more services can be executed in parallel. Let services WS_n, WS_m be the public forward and backward service of services WS_j and WS_k, the corresponding BPEL-Net model Σ is shown in Fig. 2(c). We introduce the place P_{nj}, P_{nk}, P_{jm}, P_{km} which make ${}^{\bullet}P_{nj} = {}^{\bullet}P_{nk} = t_{e,n}$, $P_{nj}^{\bullet} = t_{in,j}$, $P_{nk}^{\bullet} = t_{in,k}$, $P_{jm}^{\bullet} = P_{km}^{\bullet} = t_{in,m}$, ${}^{\bullet}P_{jm} = t_{e,j}$, ${}^{\bullet}P_{km} = t_{e,k}$.

The <*Switch*> relation refers that the system can choose one path from various branches based on special conditions. The execution process is shown in Fig. 2(d). $\lambda_{in,k} = ch_k$, $\lambda_{in,j} = ch_j$, where ch_i, ch_j represent the probability of choosing services WS_j and WS_k respectively. Let service WS_n, WS_m be the public forward and backward service of WS_j and WS_k respectively, the corresponding BPEL-Net model Σ is shown in Fig. 2(d). We introduce place P_{njk} and P_{jkm}, which make ${}^{\bullet}P_{njk} = t_{e,n}$, $P_{njk}^{\bullet} = \{t_{in,j}, t_{in,k}\}$, $\lambda_{in,j} = ch_j$, $\lambda_{in,k} = ch_k$, $P_{jkm}^{\bullet} = t_{in,m}$, ${}^{\bullet}P_{jkm} = \{t_{e,j}, t_{e,k}\}$.

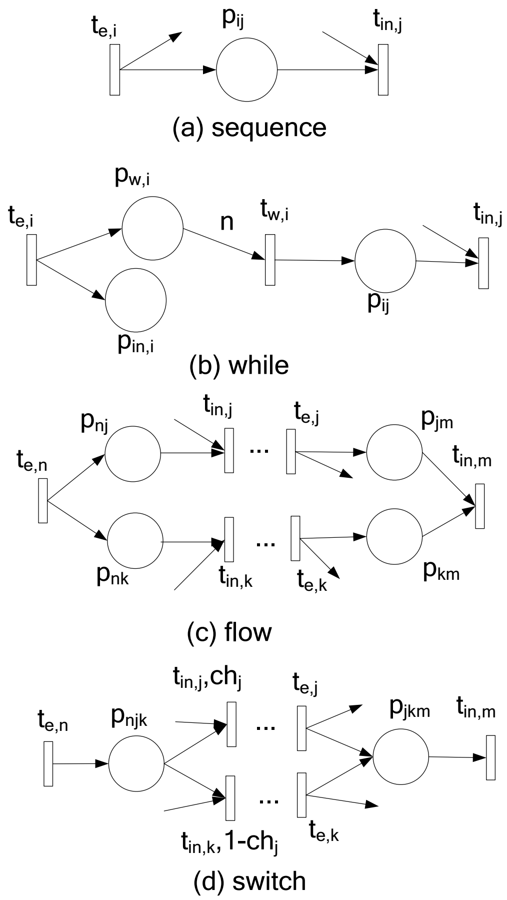

Figure 2. BPEL-Net of Structural Service.

C. Modeling Failure Processing for BPEL-Net

According to the characteristics of service, if the relation between services WS_i and service WS_j is parallel. The process has following failure processing strategies when service WS_i failed: 1) if service WS_j is in active state and can be canceled, then canceling the operation of service WS_j; 2) if service WS_j is in active state and can not be canceled, then coordinating the operation of service WS_j to reach the termination state; 3) if service WS_j is in initial state, then aborting service WS_j.

If the relation between services WS_i and WS_j is sequence, there has following processing strategies when other services failed: 1) if service WS_i has failed, then aborting service WS_j; 2) if service WS_i has been aborted, then aborting service WS_j too.

The failure processing model of BPEL-Net is shown in Fig. 3. Among them, we introduce the place P_c and transition t_c to represent the initial state and processing operation of failure processing respectively, which make:

1) ${}^{\bullet}P_c = \{t_{fa,1}, t_{fa,2}, \ldots t_{fa,n}\}$; $P_c^{\bullet} = t_c$

2) ${}^{\bullet}t_c = P_c$, $t_c^{\bullet} = \{P_{tr,1}, P_{tr,2}, \ldots, P_{tr,n}\}$ Where n is the total number of services that BPEL process called.

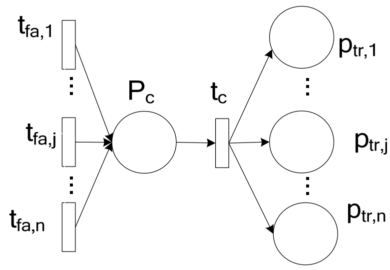

Figure 3. Modeling Failure processing.

D. Forming the Whole Application

According to BPEL-Net model of atomic service and structural service, the process of converting BPEL process into BPEL-Net model is described in the following:

1) Begin sign <process> and end sign <process> are expressed by two places, and assign the initial token to place <process>.

2) Determining the atomic services and structural services that BPEL process included and provides basic information such as input and output parameters.

3) The elements in process are presented by BPEL-Net model of atomic services and structural services. Deleting some redundant transitions and places to simplify the model.

4) Introducing initial place P_s and transition t_s which characterize the beginning operation of the whole application, and ${}^{\bullet}t_s = \{P_s\}$, $t_s^{\bullet} = \{P_{in,i} | i = 1,2, \ldots, n\}$, ${}^{\bullet}P_s = \varnothing$, $P_s^{\bullet} = \{t_s\}$, $M_0(P_s) = 1$. Meanwhile, we introduce the termination place P_e and transition t_e which characterize the termination operation of the whole application, and ${}^{\bullet}t_e = \{P_{e,i} | i = 1, 2, \ldots, n\}$, $t_s^{\bullet} = \{P_e\}$, ${}^{\bullet}P_e = \{t_s\}$, $P_s^{\bullet} = \varnothing$.

Ⅳ. CONVERTING BPEL PROCESS INTO BPEL-NET

The state space of complicated BPEL process will exponential growth as the total number of transition increasing. In order to reduce the complexity of analyzing reliability for BPEL process, we present the concept of transfer matrix in the following, and analyze the relation between transfer matrix and reliability, thus converting the analysis of reliability into computing power of transfer matrix.

A. The Transfer Matrix for BPEL-Net

BPEL-Net model is starting from initial state S_0 and will generate new state through effectively firing enabled transitions, thus establishing a state space (known as state graph). The state graph view state as node, the transition and its firing probability denoted by edge. We can analyze the reliability of BPEL process through the state graph of BPEL-Net model. However, the state graph of BPEL-Net may be complicated; it is difficult to analyze it by directly computing. So it is necessary to further abstract state graph.

First, we will define several special states, let $S=(M, TS)$ be a state of Σ: if $S \in ES(\Sigma)$ and there has service WS_i which makes $M(P_{fa,i}) = 1 \wedge ET(M) = \varnothing$, then S is called failure state of Σ, denoted by $S_{fa\to i}$ if $M(P_e)=1$, then S is called success state of Σ, denoted by S_e.

In this paper, we analyze the reliability of BPEL process by computing the failure probability of participating services. Mapping into BPEL-Net model is the probability from initial state to failure state, denoted by $Pr_{fa\to i}$. Denoting P_{re} as the probability from initial state to success state.

Property 1: In BPEL-Net model, if S_j can reach from S_i, then S_i can reach state S_j in finite steps.

Proof: The proposition is equivalent to prove that model does not have deadlock and endless loop. Because services in BPEL-Net model can be fired only after obtaining all required resources, and will not require additional resources during processing. That is, BPEL-Net model does not meet one of the necessary conditions of deadlock generated: the transition has been obstructed due to the requirement of other resources, and doesn't release its resource. Therefore, BPEL-Net model does not have deadlock. Also we don't consider the infinite service called, so the model does not have endless loop. In summary, if S_j can reach from S_i, then S_i can reach state S_j in finite steps.

Property 1 explains the state space of corresponding BPEL-Net model is finite, thus we can realize to analyze reliability of BPEL process through it.

Let state $S' = S[t_i>$, as each transition t_i has a firing probability λ_i, then the conditional probability from state S to state S' is λ_i, we call λ_i is the transformation probability from S to S'.

Definition 4: The transformation probability a_{ij} from state S_i to state S_j meets following conditions:

$$a_{ij}=\begin{cases}\lambda_{ij} : \exists t_{ij} \in T, S[(t_{ij}, w_{ij}) > S_j \\ 0 : not\ \exists t_{ij} \in T, S[(t_{ij}, w_{ij}) > S_j\end{cases} \quad (1)$$

Let the number of reachable state in BPEL-Net model Σ be L, the L-order square matrix A is called transfer matrix of Σ if it meets following conditions: $A=[a_{ij}]_{L\times L}$, where a_{ij} is the transformation probability from state S_i to state S_j. Denoted $A^{(n)}$ as n-power of matrix A, while $a^{(n)}_{ij}$ is the element in the ith row and jth column of matrix $A^{(n)}$. Vector $R^{(n)}_{i,A}$, $C^{(n)}_{j,A}$ represent the ith row and jth column of $A^{(n)}$ respectively. The ith row of matrix A represents the transformation probability from state S_{i-1} to all reachable states, while the jth column of matrix A represents the transformation probability from all reachable state to state S_{j-1}. $R_{i,A}$ is called transformation vector of state S_{i-1}.

Property 2: the probability from state S_i to state S_j by n steps is equal to the value of $a^{(n)}_{ij}$ in $A^{(n)}$.

Proof: mathematical induction

(1) If n=1, we can draw the conclusions from the definition of a_{ij}.

(2) Assuming the proposition is established when $n\le k$, now we will prove the proposition is established when $n = k + 1$, $a^{(k+1)}_{ij} = R^{(k)}_{i,A} * C_{j,A} = \sum_{r=1}^{L-1} a_{ir}^{(k)} * a_{rj}$. Because the proposition is established when $n\le k$, that is, $a^{(k)}_{ir}$ is equal to the probability from state S_i to state S_r by k steps, while a_{rj} is equal to the probability from state S_r to state S_j by one step. Therefore $a^{(k)}_{ir} * a_{rj}$ is equal to the probability from state S_i to state S_r by k steps and reach S_j from S_r by one step. Because the choice of r is arbitrary, we can get $a^{(k+1)}_{ij}$ is equal to the probability from state S_i to S_j by $k+1$ steps, that is, the proposition is established when $n=k+1$.

In summary, the probability from state S_i to state S_j by n steps is equal to the value of $a^{(n)}_{ij}$ in $A^{(n)}$.

Property 2 explains the probability from state S_i to state S_j by n steps is equal to the value of $a^{(n)}_{ij}$ in $A^{(n)}$. $a^{(n)}_{ij}$ is also called n order probability from state S_i to state S_j. We can convert the analysis of reliability into computing power of transfer matrix through property 2.

Property 3: $\forall i, j < L'$, if state S_i and S_j are reachable, then there exists $K_{ij} \in N$ which makes $\forall E \in N$, $b_{ij}^{K_{ij}+E} = 0 \wedge b_{ij}^{K_{ij}} \neq 0$.

Proof: from property 1, we can know that state S_i can reach state S_j in finite steps. We may assume that there has q firing sequences $\delta_1, \delta_2, \ldots, \delta_q$ from state S_i to state S_j. Set $K_{ij}=\max\{|\delta_1|, |\delta_2|, \ldots, |\delta_q|\}$, then $b_{ij}^{K_{ij}}$ represents the probability from state S_i to state S_j by K_{ij} steps. From the definition of K_{ij}, we can draw $b_{ij}^{K_{ij}} \neq 0$, and because K_{ij} is the maximum steps from state S_i to state S_j, therefore, $\forall E \in N$ there has $b_{ij}^{K_{ij}+E} = 0$.

Property 3 illustrates that we can analyze the reliability of BPEL process by computing stop condition of transfer matrix's power. The probability P_{ij} from state S_i to state S_j is equal to $\sum_{r=1}^{K_{ij}} a_{ij}^{(r)}$, that is, the reliability of BPEL process is got by computing the firing probability of all paths between states.

B. Simplification

Below we will propose two simplification schemas for transfer matrix, which can maintain the probability between states unchanged.

(1) Schema 1: Eliminating the corresponding row and column of termination state S_e.

Because Pre can be computed by $1-\sum_{i=1}^{|WS|}\Pr_{fa\to i}$, and all transitions couldn't be fired in termination state S_e, that is, no state can be reached from state S_e. Mapping into transfer matrix is that transformation vector of state S_e is zero vector. Therefore, eliminating the corresponding row and column of termination state S_e from matrix A will not affect the analysis of reliability for BPEL process.

(2) Schema 2: If the probability from state S_I to state S_J is equal to 1, and only S_I can reach state S_J, mapping into transfer matrix is: $a_{IJ} = 1$ and $\sum_{K=1}^{L-1} a_{Ik} = \sum_{K=1}^{L-1} a_{kJ} = 1$, then we can eliminate the Ith row and Jth column from matrix A.

Assuming matrix B is got by eliminating the Ith row and Jth column from matrix A. Because schema 2 can reduce order for transfer matrix, the corresponding mapping of any state S_k after simplifying is:

(1) If $k < I$ and $k < J$, then state S_k is corresponding to the $(k\text{-}1)$th row and $(k\text{-}1)$th column of matrix B;
(2) If $k < I$ and $k > J$, then state S_k is corresponding to the (k -1)th row and $(k\text{-}2)$th column of matrix B;
(3) If $k > I$ and $k < J$, then state S_k is corresponding to the $(k\text{-}2)$th row and kth column of matrix B;
(4) If $k > I$ and $k > J$, then state S_k is corresponding to the $(k\text{-}2)$th row and $(k\text{-}2)$th column of matrix B.

Below we will give theoretical basis for schema 2:

Theorem 1: In the transfer matrix A, if $a_{IJ} = 1$ and $\sum_{K=1}^{L-1} a_{Ik} = \sum_{K=1}^{L-1} a_{kJ} = 1$, B is the matrix got by eliminating the Ith row and Jth column from matrix A, then for any states s_i, s_j (s_i, s_j are not the states that removed from A), the probability P_{ij} from state S_i to state S_j in A is equal to the probability P_{ij}' from state S_i to state S_j in B.

Proof: according to the actual mapping of a_{ij}, b_{pq}, we can divide the proposition into proving the probability of corresponding path between any states unchanged. Let the firing sequences from state S_i to state S_j be $\delta_1, \delta_2, \ldots, \delta_m$. Now we will prove the corresponding firing probability of any firing sequence δ_q remain unchanged after simplifying. There are two cases for firing sequence δ_q:

(1)S_I is included in firing sequence δ_q , because states S_I and S_J have met above conditions, then $\delta_q=\{S_i, t_1, S_1, \ldots, t_k, S_I, t_{IJ}, S_J, \ldots, t_n, S_j\}$, that is, the firing probability of δ_q is $P_q=\lambda_1 *\lambda_2 \ldots *\lambda_k *\lambda_{IJ} \ldots *\lambda_n=\lambda_1 *\lambda_2 \ldots *\lambda_k * 1 \ldots * \lambda_n= \lambda_1 *\lambda_2 \ldots *\lambda_k \ldots * \lambda_n$. Therefore, eliminating the Ith row and Jth column from matrix A has not effect the computation of P_q;

(2) S_I is not included in firing sequence δ_q, then eliminating the Ith row and Jth column from matrix A has not effect the computation of P_q. Because $P_{ij}=P_1 + P_2 + \ldots + P_m$, the schema 2 can not change the value of P_{ij}.

In summary, for any states s_i, s_j (s_i, s_j are not the states that removed from A), the probability P_{ij} from state S_i to state S_j in A is equal to the probability P'_{ij} from state S_i to state S_j in B.

Theorem 1 provides theoretical support for schema 2, which explain n-order probability between states can be maintained by schema 2. We can continuously use simplification schemas until no states can meet the conditions, thus getting a matrix B, which is called a reliable matrix of Σ.

V. CONVERTING BPEL PROCESS INTO BPEL-NET

In this section, we will display the above analysis process through an example of applying visa. The composition process of system is: When customers applying for visa, first, getting archive service (WS_0) is to obtain entry permits, after getting permits, querying visa service (WS_1) provides the related visa information and fee computing service (WS_2) is responsible for computing the corresponding fee; information audit services (WS_3) is to check whether the applicant's information is complete. The customers can choose Beijing visa service (WS_4) or Shanghai visa service (WS_5) or local visa service (WS_6) according to the specific conditions; finally, saving archive service (WS_7) is to save related material to central database. The expression between services is $WS_0 > (WS_1 \| WS_2) > WS_3 > (WS_4+WS_5+WS_6) > WS_7$, where >, + and || represent the sequence, choice and parallel relation between services respectively. The properties of service is shown in table I. The choice probability of service WS_4, WS_5, WS_6 is 0.2，0.3，0.5 respecively。For the above expression, we can use Zen-Flow tool introduced by reference [18] to generate the corresponding BPEL process. We can also get the corresponding BPEL-Net model through the above modeling process, which is shown in Fig. 4.

TABLE I. THE PROPERTIES OF SERVICE

WS	*SP*	*Cancel*	*WS*	*SP*	*Cancel*
WS_0	0.91	can	WS_1	0.98	can
WS_2	0.96	can	WS_3	0.89	not
WS_4	0.95	can	WS_5	0.97	not
WS_6	0.94	can	WS_7	0.95	can

Fig. 4 is the BPEL-Net model of whole application, the specific parameters of each service have marked in the corresponding places and transitions. Place P_s and P_e represent the beginning and termination operation of BPEL-Net model respectively. Transition t_c will fire the failure processing in the case of other service has failed.

According to the firing rule of BPEL-Net model and definition of transfer matrix, we can use tool Matlab to generate transfer matrix of Fig. 4. The corresponding non-zero elements in transfer matrix is shown in Table II, which has 120 reachable states, so the order of matrix A is 120. Therefore, it is necessary to simplify it for reducing the complexity of computation.

TABLE II. THE NON-ZERO ELEMENTS IN TRANSFER MATRIX

R	*C*	Value	*R*	*C*	Value	*R*	*C*	Value
S_2	S_3	0.91	S_2	$S_{fa\to0}$	0.09	S_3	S_4	1
S_3	S_5	1	S_4	S_6	0.98	S_4	$S_{fa\to1}$	0.02
S_4	S_7	1	S_5	S_8	0.96	S_5	$S_{fa\to2}$	0.04
S_5	S_9	1	S_6	S_{10}	0.96	S_6	$S_{fa\to1}$	0.04
S_7	S_{11}	0.98	S_7	$S_{fa\to1}$	0.02	S_7	S_{12}	0.96
S_7	$S_{fa\to2}$	0.04	S_9	S_{11}	0.98	S_9	$S_{fa\to1}$	0.02
S_9	S_{12}	0.96	S_9	$S_{fa\to2}$	0.04	S_8	S_{13}	0.96
S_8	$S_{fa\to1}$	0.02	S_8	S_{10}	0.98	S_{11}	S_{10}	0.96
S_{12}	S_{10}	0.98	S_{10}	S_{14}	0.89	S_{10}	$S_{fa\to3}$	0.11
S_{14}	S_{15}	0.2	S_{14}	S_{16}	0.3	S_{14}	S_{17}	0.5
S_{15}	S_{18}	0.95	S_{15}	$S_{fa\to4}$	0.05	S_{16}	S_{19}	0.97
S_{16}	$S_{fa\to5}$	0.03	S_{17}	S_{20}	0.94	S_{17}	$S_{fa\to6}$	0.06
S_{18}	S_{20}	0.95	S_{18}	$S_{fa\to7}$	1	S_{19}	S_{21}	0.95
S_{19}	$S_{fa\to7}$	1	S_{20}	S_{21}	0.95	S_{20}	$S_{fa\to7}$	1

Based on the proposed simplification schemas, we can use tool Matlab to reduce order for matrix A, thus getting a reliable matrix B which only has 39 order, as shown in Fig. 5(we only display non-zero elements in the matrix B, C is the column of matrix B and R is the row of matrix B). We can also get the failure probability of each service are 0.09, 0.0357, 0.0721, 0.0039, 0.0076, 0.0069, 0.0229, 0.0399 respectively by using property 3 and tool Matlab, and success probability is 0.7588.

The example show that the number of state in BPEL-Net model will be relative larger when BPEL process is complex, which makes the analysis of state graph more difficult. However, transfer matrix can not only clearly show the structure of BPEL process, but also represent the relation between states. Simplification schemas can reduce the complexity of computation. The results show that: 1) Comparing with the analysis by Petri nets, using transfer matrix can gain significant efficiency, and in the complicated BPEL process, analyzing reliability through tool Matlab has higher efficiency; 2) Transfer matrix can be more intuitive to show the process for analyzing reliability, and reducing the complexity of the analysis for state graph.

Ⅵ. RELATED WORKS

In recent years, the problem of performance analysis in service composition has attracted great interest in the research community. And there have been some related works about formal analysis of BPEL. In this section, we will discuss some related work that is directly or indirectly of interest to our research work.

In the existing attempts, some process algebra approaches that have been used to analyze the behavior of BPEL process, which are given in reference [6, 7, 8]. A CCS-based model which generalizes the behavior of Web services and their composition is proposed in [6], which can capture the semantics of complex service interactions and their respective specificities. while the authors in [7] presented the semantics of WS-CDL in terms of process algebra CSP and verifying concurrent processes. Therefore, all the properties can be verified automatically in the CSP framework correspondingly. Process algebra is also used in [8] to design and verify the BPEL and automatically obtaining the corresponding BPEL code.

Another work about formal analysis of BPEL process is finite state machine [9, 10, 11, 12]. In [9], authors abstractly model dynamic properties of the key language constructs through the construction of a BPEL abstract machine, and provided a precise and well defined semantic framework for establishing the key language attributes. Reference [10, 11, 12] are also the works based on finite state machines and abstract state machines.

The works most close to ours include: A checking tool for translating BPEL specifications into the input language of the Petri nets model called LoLA has been proposed in [13], which demonstrated that the semantics is well suited for computer aided verification purposes. Lisa Wells [14] used Colored Petri Nets and CPN Tools for generating different kinds of performance-related output, and for running multiple simulation replications. Three services scheduling algorithm are presented in [15] to calculate the non-function quality of service composition. An XML-based deployment descriptor is proposed in [16] to specify the non-functional requirements in a declarative way. Reference [17] introduces the concept of a transactional pattern for specifying flexible and reliable composite Web services, which can simply connect together transactional patterns to define a composite Web service.

To some extent, our work has been influenced by the above research results. Below are some of the key differences when comparing the above approaches to the one presented in this paper:

(1) Compared to generic finite state machines and process algebra, Petri nets provide a much broad basis for computer aided verification, so we think it is more natural to model different input and output information of BPEL process by means of Petri nets;

(2) There are also some works described based on Petri nets, but these works only involves a certain part of BPEL process, they don't take into account the crucial transactional properties of services and failure processing between services;

(3) We propose the concept of transfer matrix to represent the relation between states, which makes the analysis of reliability easier. Hence our proposed approach is suitable to analyze reliability of BPEL process at design time, which is especially significant in the context of service composition.

Ⅶ. CONCLUSIONS

Along with the development of Web Service technology, service composition technology has been

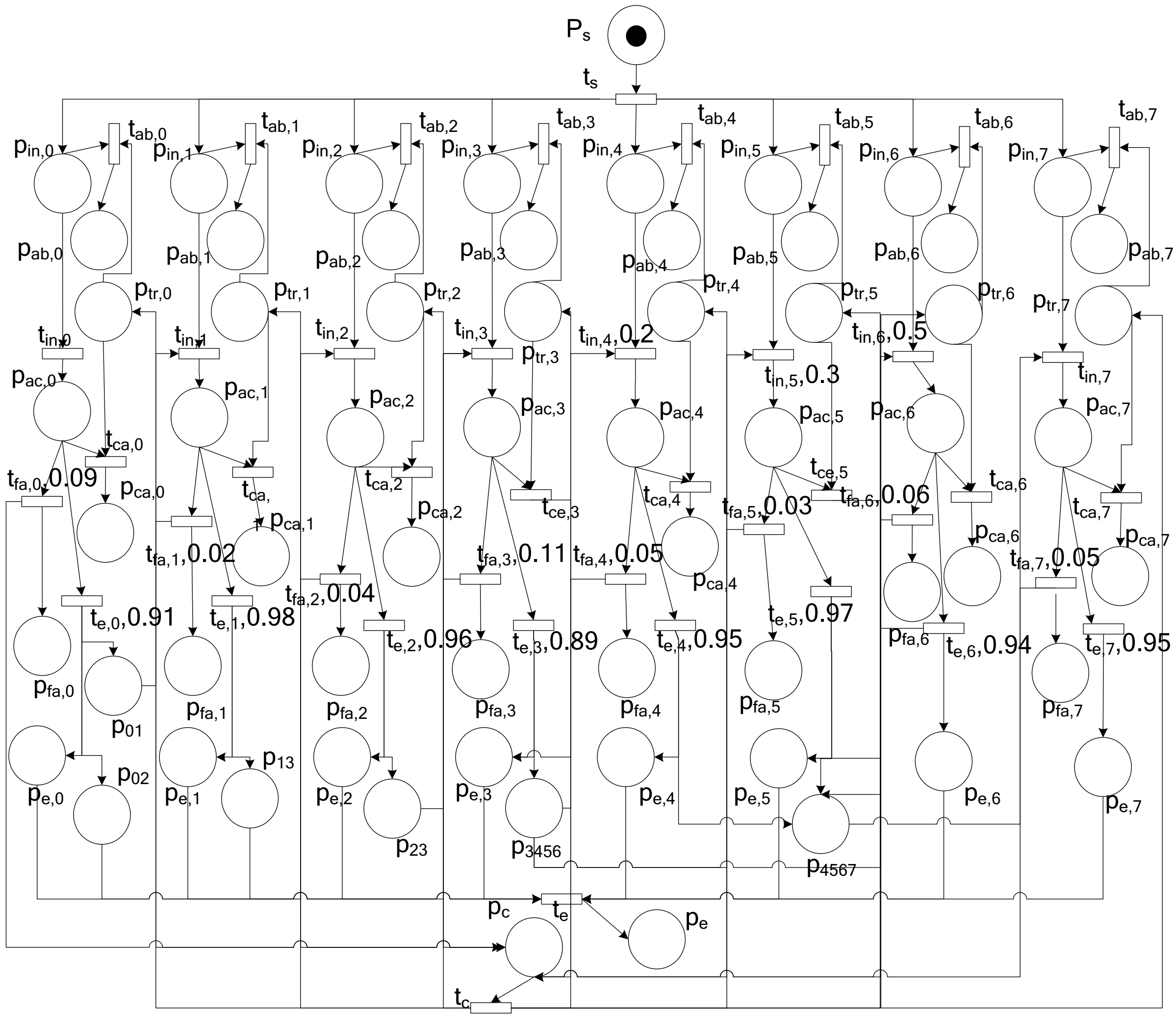

Figure 4. BPEL-Net Model of Whole Application

used to combine various Web services into a newer and more powerful service, and several languages to describe and standardize the process of Web service composition have appeared. BPEL is one of these languages. However, in addition to the necessary functions, the composite process should provide the quality of service (QoS) needed by the users. Reliability is one of the important attributes of QoS. How to analyze the reliability of BPEL process that can be offered to service consumers when building process has been a challenge research.

In this paper, we have proposed an approach for analyzing the reliability of BPEL process, formal definition of atomic services, structural services, failure processing strategies et al related elements, so as to better describe the execution of BPEL process. We express the relations between states through transfer matrix and convert the analysis of reliability into computing power of transfer matrix, thus reducing the complexity of analysis. We proposed two simplification schemas for transfer matrix and given judgment theorem to prove the soundness. Finally, we displayed the feasibility and practically of our method through a special example.

Since service composition are becoming more complex in actual application, it is becoming harder to develop them. Our future work involves considering the integration of offset configuration tools. Besides, the dynamic service binding is an important issue when integrating the available services; we will also further investigate the dynamic service binding issue in our proposed model.

ACKNOWLEDGMENT

The research described here was partially supported by the NSF of China under grants No.60373075, the Science-Technology Development Foundation of Shanghai of China under Grant No.06dz15004-1.

REFERENCES

[1] M. P. Papazoglou and W. J. Heuvel, "Service oriented architectures: approaches, technologies and research issues," International Journal on Very Large Data Bases. 2007, 16(3), pp. 389 - 415.

[2] A. D. Ambrogio and P. Bocciarelli, "A model-driven approach to describe and predict the performance of composite services," In Proceedings of the 6th

international workshop on Software and performance. New York, ACM. pp. 78-89. 2007..
[3] B. Luciano and G. Sam, "A dynamic and reactive approach to the supervision of bpel processes," In Proceedings of the 1st conference on India software engineering conference. New York, ACM. pp. 39–48. 2008
[4] K. Mt, V. Dniel, and G. Lszl, "Formal modeling of bpel workflows including fault and compensation handling," In Proceedings of the 2007 workshop on Engineering fault tolerant systems. New York, ACM. 2007.
[5] T. Murata. Petri nets: Properties, analysis and application. In Proceedings of the IEEE, volume 77, 1989, pp. 541–580.
[6] W. Zhang, Z.Cao and B. Li, "A CCS based model for describing and verifying the behavior of web service," In 2007 IFIP International Conference on Network and Parallel Computing Workshops. IEEE Computer Society. pp. 987–994. 2007.
[7] J. Li, J. He, H. Zhu, and G. Pu, "Modeling and verifying web services choreography using process algebra," In 31st IEEE Software Engineering Workshop. IEEE Computer Society. pp. 256–268. 2007.
[8] F. Andrea, "Web services: a process algebra approach," In Proceedings of the 2nd international conference on Service oriented computing. New York, ACM. pp. 242–252. 2004.
[9] F. Roozbeh, G. Uwe, and V. Mona, "An abstract machine architecture for web service based business process management," International Journal of Business Process Integration and Management. 2006, 1(4). pp. 279–291.
[10] C. E. Gerede, R. Hull, O. H. Ibarra, and J. Su, "Automated composition of e-services: lookaheads," In Proceedings of the 2nd international conference on Service oriented computing. New York, ACM. pp. 252–562. 2004.
[11] A. Deutsch, L. Sui, V. Vianu, and D. Zhou, "Verification of communicating data-driven web services," In Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. New York, ACM. pages 90–99. 2006.
[12] K. E. Fakih, A. Kolomeez, S. Prokopenko, and N. Yevtushenko, "Extended finite state machine based test derivation driven by user defined faults," In 2008 International Conference on Software Testing, Verification, and Validation. IEEE Computer Society. pp. 308–317. 2008.
[13] H. Sebastian, S. Karsten, and S. Christian, "Transforming bpel to petri nets," Business Process Management. Springer Berlin / Heidelberg,.vol. 3649. pp. 220–235. 2005.
[14] W. Lisa, "Performance analysis using CPN tools," In Proceedings of the 1st international conference on Performance evaluation methodolgies and tools. New York, ACM. 2006.
[15] Y. Wang, H. Wang and X. Xu, "Web services scheduling:Binding the cost with the time," In Proceedings of the First International Conference on Semantics, Knowledge,and Grid. Guilin Guanxi, China. pp. 141–141, 2005.
[16] A. Charfi, B. Schmeling, A. Heizenreder, and M. Mezini, "Reliable, secure, and transactedweb service compositions with ao4bpel," In Proceedings of the European Conference on Web Services. New York, ACM. pp. 339–346. 2006..
[17] B. Sami, P. Olivier, and G. Claude, "Extending workflow patterns with transactional dependencies to define reliable composite web services," In Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services. New York, ACM. pp. 339–346. 2006.
[18] M. Alberto, P. M. Marta, J. P. Ricardo, and P. S. Francisco. Zenflow: A visual web service composition tool for bpel4ws. In Proceedings of the 2005 IEEE Symposium on Visual Languages and Human-Centric Computing. IEEE Computer Society. pp. 181–188. 2005.

Liqiong Chen. She was born in 1982, Ph. D. candidate. Her research interests include distributed computing, embedded systems and formal methods.

Zhiqiong Shao. He was born in 1967, professor, Ph. D. supervisor, IEEE senior member, ACM member, China Computer Federation senior member. His research interests include software engineering, information security and formal methods.

Guisheng Fan. He was born in 1980, Ph. D. candidate. His research interests include service oriented computing, distributed computing and formal methods.

Xiuying Wang. She was born in 1970, Ph. D. candidate. His research interests include service oriented computing, distributed computing and formal methods.