ISSN: 1813-4890

Web Service Composition based on SCPN Net

Fucheng Xue[a], Quanqiu Jia, Yawei Zhang
Taishan College of Science and Technology, Taian, Shandong, 271000 China
[a]Punisher117@163.com

Abstract

With the rapid development of the Internet, web service composition has become a hot issue nowadays. There are some problems in existing service compositions such as the single model, the too-high complexity of the model when dealing with complicated problems, the easy appearance of low efficiency or easily getting trapped in local best value when an intelligent algorithm is used solo to solve problems, and so on. To solve all these problems, the author of this paper put forward a service composition model based on Petri Net, SCPN (Service Composition Petri Net), and verified the feasibility of this model via experiments.

Keywords

Web Service Composition; Petri Net; Qos.

1. Introduction

Web service as a new web application mode has developed very fast and has been widely applied in recent years. Web service composition(WSC) realizes seamless integration among services by reusing various existing services spreading over the internet and constructing applications that can satisfy customers' demands. Along with the rapid development and wide application of web service technologies, how to efficiently and dynamically aggregate various existing web services and construct new value-added services to meet different customers' demands has become a new application demand and research subject[1]. Web service composition is a process of dynamically discovering the existing web services according to certain rules and composing them into a service or a system of added value and larger granularity so as to satisfy the customers' complicated demands and raise the software's productivity.

At present, the methods to solve web service composition problems can be divided into two categories. One category is based on the attribute values of the quality of services(QoS). To construct an optimal solution model via intelligent algorithms, realize service composition through QoS attribute perception and obtain a composing service of maximum QoS value via global optimization services is a classic NP-hard problem [2]. The other category is formal methods which might be based on dynamic composition of workflow, based on the method of a semantic network, or based on PI-Calculus or tools like Petri Net. Formal methods based on the algebra process or tools like Petri Net are common methods of web service composition[3]. If Petri is used alone to solve service composition problems, the complexity will be too high when the amount of services is too large; When an intelligent algorithm is used alone to solve the QoS model, mostly the intelligent algorithm will get trapped in the local best value when it is approaching the best value and the logic relations among various service compositions can not be specifically presented. In Reference[4], Huo and his partners prove that compared with NSGA-II and MOPSO, the Eliteguided Multi-Objective Artificial Bee Colony (EMOABC) they propose gets the best optimization in a short time. A modified GA, Coevolutionary Genetic

Algorithm (CGA), is put forward in Reference[5] and this algorithm uses a true coding method to solve QoS-based service selection problems.
In this paper, a web service composition model based on Petri Net was proposed.

2. Service Composition Model based on Petri Net

There are many applications in the cloud environment and through these applications requests of users can be fulfilled highly efficiently. Take one of the services as an example. When a request R is sent from a user, it is first decomposed into multiple sub-tasks $R=(Task_1,Task_2,...,Task_n)$ fulfilled. Thereinto, every sub-task $Task_i,(i=1,2,...,n)$ fulfills a specific function. Task R is fulfilled by its sub-task sets according to certain service logic. In the process of executing every sub-task, it can be discovered through service that every sub-task $Task_i,(i=1,2,...,n)$ can be fulfilled by multiple candidate atomic services. Suppose a Sub-task $Task_i$ has m candidate atomic services, represented as $S_i=(s_{i1},s_{i2},...,s_{im})$, then from these candidate service sets, by comparing the services' non-functional attributes, a web service with the best overall performance $s_{ik}(k=1,2,...,m)$ can be selected to execute the sub-task's task. When all the sub-tasks have selected their corresponding atomic services, service composition can be conducted. Through service selection, the QoS value of every composing service can be computed, and finally, the service composition scheme with the best performance can be selected. The final composition scheme is executed according to a certain logical sequence of the request. To fulfill users' requests, every task's atomic service sets which are of certain related functions have to be integrated through service composition. Since the existence of execution logical relations among different services, to understand the relations among services, the author of this paper adopted Petri Net in modeling and dynamically analyzed and verified the network system.

Definition 3.2 Define an atomic service and it can be represented as a quaternion $CS=\{s\_id,s\_function,s\_type,s\_status,s\_constr\}$. Thereinto s_id represents service identification; $s\_function$ represents the service's execution function; s_type represents service type; s_status represents the status of the service; s_constr represents the constraint relation among services or assessment index value.

In the service composition process, the composition should be conducted in accordance with users' demanding logic. The execution relation among services has a great impact on solving QoS values and different execution relations have different solving modes. Thereinto, the basic execution structures are specifically represented as follows. Meanwhile, the execution structures appearing in the service composition process correspond with those in Petri Net one by one.

(1) Sequential Relation

Services are executed in sequential order. Thereinto, the black mark represents the start and end of the service execution, as is represented in Figure 1.

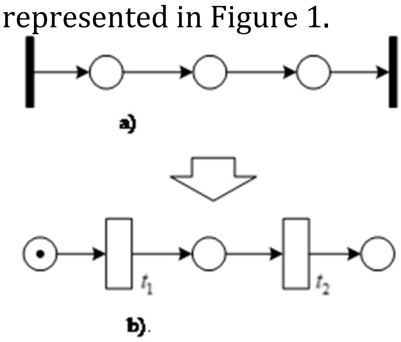

Figure 1. Sequential Structure of Services

ISSN: 1813-4890

(2) Parallel Relation

Parallel relation means multiple sub-tasks can be conducted simultaneously. The executions of multiple atomic services don't interfere with one another and can be conducted simultaneously. When all are fulfilled, the next task will be executed according to execution logic, as is represented in Figure 2.

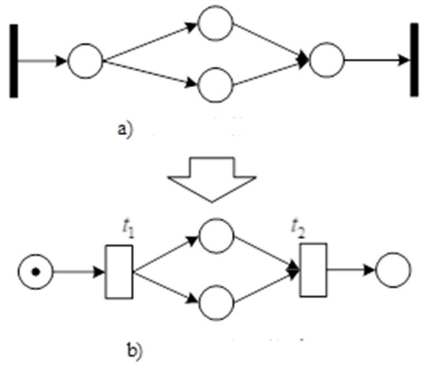

Figure 2. Parallel Relation of Services

(3) Selective Relation

Selective relation means a path for execution should be selected from multiple paths. Before deciding which one is the execution path, QoS values cannot be precisely computed. But to decide which one is the executive branch is selected in systematic operation. Thus, a probability model is used to assess which branch to select. As is represented in Figure 3.

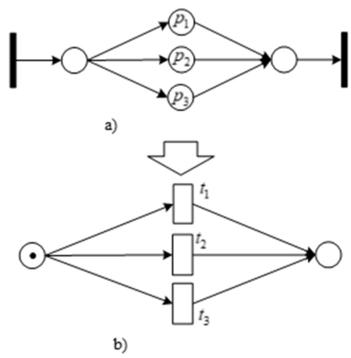

Figure 3. Selective Relation of Services

(4) Circular Relation

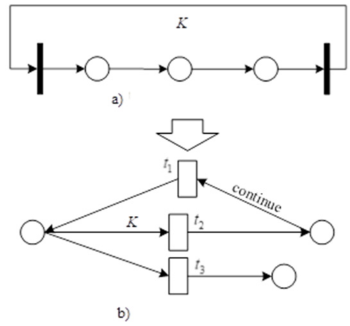

Figure 4. Circular Relation of Services

Circular relation is a special branch relation. Its circulation process can be converted to sequential relations. When the time of circulation is K, it means that there are K loop bodies in sequential execution and QoS values are computed in this model, as is shown in Figure 4.

Definition 3.3 Define a Service Composition Petri Net, (SCPN). It is composed of an eleven-tuple $P=\{P,T,E,L,I,O,F,E_{th},L_{th},S,M_{cs}\}$. Thereinto, if and only if:

(1) $P=\{p_1,p_2,\ldots,p_m\}$ represents a finite set of repositories, thereinto, $p_i,(i=1,2,\ldots,m)$ represents No. i atomic service in candidate atomic service set;

(2) $T=\{t_1,t_2,\ldots,t_n\},(n>0)$ represents a finite set of transitions, thereinto, $P\cap T=\varnothing$, $P\cup T\neq\varnothing$;

(3) $E=\{e_1,e_2,\ldots,e_m\},(m>0)$ represents a finite set of assessed values of atomic services' response time and this shows the execution efficiency of the atomic services in service composition;

(4) $L=\{l_1,l_2,\ldots,l_m\},(m>0)$ represents a finite set of atomic services' failure frequency;

(5) $I:P\times T\to\{0,1\}$ represents the output arc function from a repository to a transition, in terms of repositories;

(6) $O:T\times P\to\{0,1\}$ represents the output arc function from a transition to a repository, in terms of transitions;

(7) $F\subseteq(P\times T)\cup(P\times T)$ represents a finite set of SCPN's directed arcs;

(8) $E_{th}=\{e_{th\_1},e_{th\_2},\ldots,e_{th\_m}\}$ represents a finite set of critical values of services' response time;

(9) $L_{th}=\{l_{th\_1},l_{th\_2},\ldots,l_{th\_m}\}$ represents a finite set of critical values of services' failure frequency

(10) $S:\{E\to E_{th},L\to L_{th}\}$ represents the mapping relation function from the service's two assessed values to its critical values;

(11) $dom(F)\cup cod(F)=P\cup T$;

(12) $M:P\to\{0,1,2,\ldots\varphi\},(\varphi\in N^+)$ represents the SCPN's identification function. Its initial identification is M_0 and it is the Petri Net's initial status; Any repository and transition should be linked by a directed arc and no isolated point is allowed.

Definition 3.4 In SCPN net, suppose Repository $p_m\in I(t_i)$ meets all the conditions of $M(p_i)\geq\varphi$, $e_i\geq e_{th\_i},l_i\geq l_{th\_i},(i=1,2,\ldots,m)$, then Transition t_i has happened right in Identification M and it is represented as $M[t_i>$.

In this optimized service composition model, every repository represents a candidate atomic service. If assessed values of services' response time $e_i\geq e_{th\_i}$ and assessed values of services' failure $l_i\geq l_{th\_i}$, then it means that the services' performances have met the basic requirements of service quality and can be composed. Literature References

3. Experiments and the Analyses of the Results

To verify the feasibility of the Petri Net Model proposed in this paper, we adopted Petri's Reachable Identifying Graph analyzing method[6] to dynamically analyze the performance of the Petri Net TBPN and made a deadlock judgment on the execution logic of system services.

The SCPN's net reachable status was analyzed through the reachable identification graphs. The example's initial identifying status is $M_0=\{1,0,0,0,0,0,0,0,0\}$. Its reachable set is shown in Table 1. There are a total of 9 types of status identification. The model's identification is $P_{tbpn}=(p_1,p_2,p_3,p_4,p_6,p_7,p_8,p_9)$, with every row representing an identification.

Table 1. The Reachable Set of SCPN Net

p_1	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9
1	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**
0	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**
0	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**

The analysis result of the net's reachable status is shown in Figure 5.

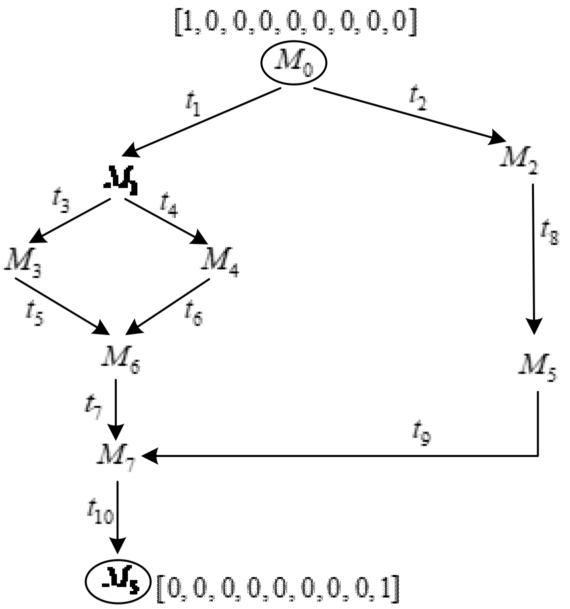

Figure 5. Analysis Result of the Reachable Status

The BTPN's reachable status identification graph was analyzed. The status sequence $(M_0, M_1, M_4, M_6, M_7, M_8)$ was analyzed as an example and the results were as follows. Thereinto $Stat$ represented status and $Tran$ represented transition.

(1) $M_0 = \{1,0,0,0,0,0,0,0,0\}$, $Stat = \{M_0\}$, $Tran = \varnothing$;

(2) $M_1 = \{0,1,0,0,0,0,1,0,0\}$, $Stat = \{M_0, M_1\}$, $Tran = \{t_1\}$;

(3) $M_4 = \{0,0,0,1,0,0,0,0,0\}$, $Stat = \{M_0, M_1, M_4\}$, $Trasition = \{t_1, t_4\}$;

(4) $M_6 = \{0,0,0,0,0,0,1,0,0\}$, $Status = \{M_0, M_1, M_4, M_6\}$, $Trasition = \{t_1, t_4, t_6\}$;

(5) $M_7 = \{0,0,0,0,0,0,0,1,0\}$, $Status = \{M_0, M_1, M_4, M_6, M_7\}$, $Trasition = \{t_1, t_4, t_6, t_7\}$;

(6) $M_8 = \{0,0,0,0,0,0,0,0,1\}$, $Status = \{M_0, M_1, M_4, M_6, M_7, M_8\}$, $Trasition = \{t_1, t_4, t_6, t_7, t_{10}\}$.

The operation results showed that the initial status of the system M_0 could always arrive at the ending status of the system M_8 through transition sequences. There were no deadlocks in this net system and all the statuses included all the repositories and transitions. Therefore, this net

ISSN: 1813-4890

system had liveness and full coverage and the logical structure among service compositions had correctness and validity.

4. Conclusion

For the existing problems of web service composition, such as the single model, the inability to clearly represent logic relations among service compositions, the low efficiency of intelligent algorithms when the scales of problems become complicated, and so on, the author of this paper first put forward a service composition model SCPN based on Petri Net and verified the feasibility of this newly proposed model.

References

[1] Zhang Yanping, Jing Zihui, et al. "Dynamic Web Service Composition Based on Discrete Particle Swarm Optimization." Computer Science, 42(06)(2015):71-75.

[2] Samia Chibani Sadouki, Abdelkamel, Tari, "Multi-Objective and Discrete Elephants Herding Optimization Algorithm for QoS Aware Web Service Composition." RAIRO-Oper. Res,53 (2019): 445–459.

[3] Guo Yubin, Du Yuyue, Xi Jianqing, "A CP-Net Model and Operation Properties for Web Service Composition."Chinese Journal of Computers, 29(07)(2006).

[4] Huo Y, Qiu P, Zhai J, Fan D, Peng H , "Multi-Objective Service Composition Model Based on Cost-Effective Optimization." Appl Intell 48(3)(2018) :651–669.

[5] Li Yuanzhang, Hu Jingjing, Wu Zhuozhuo, Liu Chen1, Peng Feifei, Zhang Yu ,"Research on QoS Service Composition Based on Coevolutionary Genetic Algorithm." Soft Computing, 2018 DOI: 10. 1007/ s00500-018-3510-5.

[6] Yuan CY. The Application of Petri Net. Beijing: The Press of Science, 2013 (in Chinese).