2017 13th International Conference on Computational Intelligence and Security

Conformance Checking and QoS Selection Based on CPN for Web Service Composition

Chen Liping
School of Network Security and Informatization,
Weinan Normal University
Weinan, China
E-mail:wnchlp@126.com

HaWeitao
School of Network Security and Informatization,
Weinan Normal University
Weinan, China
E-mail:wnhwt@126.com

***Abstract*—The development of new services by composition of existing ones has gained considerable momentum as a means of integrating heterogeneous applications and realizing business collaborations. In this paper we use skyline computation to select services for composition efficiently, reducing the number of candidate services to be considered. Then a novel color Petri net model of Web service composition is presented that combines QoS-based optimal service selection and consistence verification. In the model we define aggregation functions, and use a Multiple Attribute Decision Making approach for the utility function to achieve optimal services selection of QoS properties. We also propose a consistence verification approach to identify potential logical inconsistence of the semantic Web service process before the deployment. Proofs are also presented. We evaluate our approach experimentally using both real and synthetically generated datasets.**

Keywords- Service Composition; Web Service Selection; Skyline; QoS Optimization; Web Service Consistence Verification

I. INTRODUCTION)

Web Services are distributed applications that interoperate across heterogeneous networks and provide services that are hosted and executed on remote systems. Service oriented architecture (SOA) is gaining prominence as a key architecture to support BPM (Business Process Management) and integrate applications in diverse and heterogeneous distributed environments. Web services composition is becoming an efficient and cost-effective way to develop modern business applications, and it aggregates some existing Web services which are developed by different organizations and offer diverse functional, transactional, and nonfunctional properties.

II. RELATED WORK

In the last years, although the problem of web service selection and composition has received much attention of many researchers, designing a composite Web service which ensures not only correct and reliable execution but also optimal QoS remains an important challenge. Indeed, these two aspects of selection always are implemented separately.

Web services transactions have received much attention recently. Industrial Web services transaction specifications emerge. WS-AtomicTransaction, WS-BusinessActivity and WS-TXM rely on ATM to define transactional coordination protocols. Like ATM these protocols are unable in most cases to model Business process due to their limited control structure. It also ensures reliability on behalf of process adequacy or the opposite. Indeed, a transactional pattern taken alone as a composition of transactional patterns can be considered as a transactional protocol.

In one hand, WSBPEL and WS-CDL follow a workflow approach to define services compositions and services choreographies. Like workflow systems these two languages meet the business process need in term of control structure. However, they are unable to ensure reliability especially according to the designers' specific needs.

Transaction has achieved a great success in the database community [1], [2]. One of the most important reasons is that the operations in database have clear transactional semantics. However, this is not the case in Web services. To solve this problem, the extension mechanism of WSDL can be exploited to explicitly describe the transactional semantics of Web services operations [3], [4].

There are many works adopt three kinds of transactional properties proposed in [5] to express the different transactional semantics of Web services. Based on this classification, Vidyasankar and Vossen [6] analyze the termination property of a composite service. Bhiri et al. [7] define a set of transactional rules to verify the required failure atomicity specified by ATS [8], given that the skeleton of a composite service and the transactional properties of its component services. Montagut and Molva [9] propose an approach to deduce the required transactional properties of every task based on ATS and then use the result to guide service selection.

For these researches Web services composition based on transactional properties ensures a reliable execution, however, an optimal QoS composite Web service is not guaranteed.

QoS guarantee for Web services is one of the main concerns of the SLA framework. There are projects studying QoS-empowered service selection. In [10], authors present a QoS-aware Web service compositions which is middleware-supporting quality-driven. But the method is based on integer linear programming and best suited for small-size problems as its complexity increases exponentially with the increasing problem size. For the [11], the authors propose an extensible QoS computation model that supports an open and fair

0-7695-6341-4/17/31.00 ©2017 IEEE
DOI 10.1109/CIS.2017.00066

management of QoS data by incorporating user feedback. However, the problem of QoS-based composition is not addressed by this work. The work of Zeng at al. [12], [13] focuses on dynamic and quality-driven selection of services. The authors use global planning to find the best service components for the composition. They use linear programming techniques [14] to find the optimal selection of component services. Linear programming methods are very effective when the size of the problem is small, but suffer from poor scalability due to the exponential time complexity of the applied search algorithms [15]. Despite the significant improvement of these algorithms compared to exact solutions, both algorithms do not scale with respect to the number of candidate web services, and hence are not suitable for real-time service composition. The proposed skyline based algorithm in this paper is complementary to these solutions as it can be used as a pre-processing step to prune non-interesting candidate services and hence reduce the computation time of the applied selection algorithm.

III. CANDIDATE WEB SERVICES SCREENING BASED ON SKYLINE COMPUTATION

A. Basic concept

The basic skyline consists of all non-dominated database objects. That means all database objects for which there is no object in the database that is better or equal in all dimensions, but in at least one aspect strictly better. Assuming every database object to be represented by a point in n-dimensional space with the coordinates for each dimension given by its scores for the respective aspect, we can formulate the problem as:

The Skyline Problem: Given set $O:=\{o1,\cdots,oN\}$ of N database objects, n score functions $s1,\cdots sn$ with $si : O \rightarrow [0,1]$ and n sorted lists $S1,\cdots,Sn$ containing all database objects and their respective score values using one of the score function for each list; all lists are sorted descending by score values starting with the highest scores. Wanted is the subset P of all non-dominated objects in O, i.e. $\{oi \in P | \neg \exists oj \in O:(s1(oi) \leqslant s1(oj) \wedge \cdots \wedge sn(oi) \leqslant sn(oj) \wedge \exists q \in [1,\ldots,n]: sq(oi)<sq(oj))\}$.

B. Screened skyline web services

QoS-based service composition is a constraint optimization problem which aims at selecting individual services that meet QoS constraints and also provide the best value for the utility. For a composite web service with n activities and l candidate services per activity, there are ln possible combinations to be examined. Hence, performing an exhaustive search can be very expensive in terms of computation time and, therefore, inappropriate for run-time service selection in applications with many services and dynamic needs. Skyline computation offers a new solution of finding optimal data from huge data sets, whose computation can be expensive and whose applications require fast response times.

The main idea in our approach is to perform a skyline query on the services in each activity to distinguish between those services that are potential candidates for the composition, and those that can not possibly be part of the final solution. The latter can effectively be pruned to reduce the search space.

IV. A COLOR PETRI NET WITH QOS SELECTION

A. New web service ontology description

Definition 1. New Web Service Ontology. The new Web service ontology is a triple (SBA,SFI,QoS). SBA includes all of service basic attributes, as ID, name, URL of service description file, function, domain, provider information, etc. SFI is a set of interfaces. An interface is expressed as quad (in, pre, out, eff), where ‘in’ is the set of input parameters; ‘pre’ is interface precondition; ‘out’ is the set of output parameters; ‘eff’ is execution result. QoS is expressed as non-functional attributes.

B. A color QoS petri net

Definition 2. Color QoS Petri Net (CQPN). The color QoS Petri net is a 13-tuple (Σ, P, T, A, N, ξ, G, B, α, β, γ, μ, IN), where:

Σ is a finite nonempty set of color $\Sigma=\{SFI\times QoS\}$

P is a finite nonempty set of places

T is a finite nonempty set of transitions $P\cup T=\emptyset$

A is a flow relation $A\subseteq(P\times T)\cup(T\times P)$ for the set of arcs.

N is node function $N:A\rightarrow P\times T\cup T\times P$

C is color function $C:P\rightarrow\Sigma$

G is guard function $G:T\rightarrow Expression$, $\forall t\in T:[Type(G(t))=Bool\wedge Type(Var(G(t)))\subseteq \Sigma$. Expression is a functional expression; Type(G(t)) expresses type of G(t); Var(G(t)) expresses the set of variables; Type(Var(G(t))) expresses types set of all of variables in function G(t)

E is arc function $E: A\rightarrow Expression$, $\forall a\in A: [Type(E(a)) =C(p(a))MS\wedge Type(Var(E(a)))\subseteq\Sigma]$, and C(p(a))MS expresses multiple color set of place p of arc a

$\alpha: T\rightarrow Q+$ is minimum time that is spent by firing transition

$\beta: T\rightarrow(Q+\cup\{\infty\})$, $\alpha\leq\beta$ is maximum time that is spent by firing transition

$\gamma: T\rightarrow Expression$, $\forall t\in T: [Type(\gamma t(y))=Q\wedge y\in[\alpha,\beta]]$, and γ is cost function of transition in the interval $[\alpha,\beta]$of delay time.

$\mu: T\rightarrow SP$, $SP=[0,1]$ is the probability of transition executing.

IN is an initialize function.

Definition 3. A marking M is a vector (M(p1), M(p1), M(p1),…, M(p1)), $\forall p\in P:M(p) \in C(p)MS$.

Definition 4. Enabling Condition.

$\forall\mu\in Var(G(t): G(t)<b(\mu)>=true, b(\mu)\in C(p), p\in\{p\in P | \exists a\in A:N(a)=(p,t)\}$

$\forall p\in P: E(p,t)\leq M(p)$.

When the two conditions are met transition t will enable under marking M, which is indicated M[t>.

Definition 5. Firing Rule. When $\alpha(t) \leq d(t) \leq\beta(t)$ is true, transition t is fired. The fired result is $\forall p\in\{p\in P | \exists a\in$

$A:N(a)=(p,t)\vee N(a)=(t,p)\}:M'=M(p)-E(p,t)+E(t,p)$, which is indicated $M[t> M'$.

V. EXPERIMENTATION

In the section, we use two scenarios to evaluate the effectiveness and the efficiency of our approach. In the first scenario, different services are generated to implement the activities of example workflow. The result shows that the composite services which are composed by the selected component services not only are executed correctly and reliably but have optimal QoS. In the second one, we use the OWL-S service retrieval test collection OWLS-TC v22. The execution time of QoS services selection with skyline computation is compared with that without skyline computation.

For the first scenario, we use the OWL-S service retrieval test collection OWLS-TC v22.

This collection contains services retrieved mainly from public IBM UDDI registries, and semi-automatically transformed from WSDL to OWL-S. We apply skyline to select the best candidates for QoS selection. We compare execution time of QoS selection using skyline computation with the time without using it.

The second scenario is implemented as follow. In order to evaluate the behavior of our service selection approach, we write program whose input is a workflow composed of n activities and the output is a TCWS corresponding to a list of elementary Web services or composite Web services assigned to each activity of the input workflow. Experiments were conducted by implementing the proposed service selection approach with the program on a PC Core i3 with 2GB RAM, Windows 7, and Java 2 Enterprise Edition V1.5.0. The experiments involved composite services varying the number of activities and varying the number of Web services. The example in this paper is based upon a travel scheduling service composition.

We select twelve atomic Web services for composite Web service, and carried out the experiment for ten groups in HA and EA and each group probability of transition executing is generated by random from 0.8 to 1.0. The results of experiment are shown in Table 1. The execution path selected by our approach can meet the requirement of users and Table 1 shows that with the increasing of the atomic Web services, the selection time is relatively stable.

TABLE I. TEN GROUPS IN HA AND EA

Group	EM	HA
1	0.673	0.592
2	0.668	0.591
3	0.704	0.704
4	0.579	0.576
5	0.764	0.754
6	0.668	0.646
7	0.714	0.708
8	0.629	0.617
9	0.651	0.651
10	0.621	0.621
Average	0.667	0.645

VI. CONCLUSION

In this paper, we use that the color Petri net model allows describing not only a static vision of a system, but also its dynamic behavior, and it is expressive enough to capture the semantics of complex Web services combinations and their respective interactions. In this paper we defined a colored Petri net extended with QoS properties and give the semantics for quality colored Petri net in terms of quality timed transition model. We also propose a consistence verification approach based on the color Petri net which can detect the logical inconsistence of the semantic Web service process before the deployment, enhancing the robust of the process and the user's satisfaction. We define aggregation functions, and use a Multiple Attribute Decision Making approach for the utility function to achieve Qos-based optimal service selection. Using the color Petri net achieves reliable execution and optimal QoS of composite Web service.

Web services in SWS as new candidate services which are variables for QoS aggregation function and select Qos-based optimal service that maximizes the overall utility value from SWS.

As shown by experiment results, our approach is quadratic in term of selection and service size, in the majority of cases, which is the best solution in terms of QoS.

ACKNOWLEDGMENT

This work is partially supported by the project of scientific research fund of Education department of shaanxi province, No. 16JK1273.

REFERENCES

[1] A. Elmagarmid, Transaction Models for Advanced Database Applications.Morgan-Kaufmann, 1992.

[2] G. Weikum and G. Vossen, Transactional Information Systems:Theory, Algorithms, and the Practice of Concurrency Control and Recovery. Morgan-Kaufmann, 2002.

[3] T. Mikalsen, T. Tai, and I. Rouvellou, "Transactional Attitudes: Reliable Composition of Autonomous Web Services," Proc. WorkshopnDependable Middleware-Based Systems at the Dependable Systems and Network Conf., 2002.

[4] P.F. Pires, M.R.F. Benevides, and M. Mattoso, "Building Reliable Web Services Compositions," Proc. Web, Web-Services, and Database Systems, NODe Web and Database-Related Workshops, pp. 59-72,2003.

[5] S. Mehrotra, R. Rastogi, A. Silberschatz, and H. Korth, "A Transactional Model for Multidatabase Systems," Proc. Int'l Conf.Distributed Computing Systems (ICDCS '92), pp. 56-63, 1992.

[6] K. Vidyasankar and G. Vossen, "A Multi-Level Model for Web Services Composition," Proc. IEEE Int'l Conf. Web Services (ICWS '04), pp. 462-469, 2004.

[7] S. Bhiri, O. Perrin, and C. Godart, "Ensuring Required Failure Atomicity of Composite Web Services," Proc. Int'l Conf. World Wide Web (WWW '05), pp. 138-147, 2005.

[8] M. Rusinkiewicz and A. Sheth, "Specification and Execution of Transactional Workflows," Modern Database Systems: The Object Model, Interoperability, and Beyond, ACM Press/Addison-Wesley,1995.

[9] F. Montagut and R. Molva, “Augmenting Web Services Composition with Transactional Requirements,” Proc. IEEE Int’l Conf. Web Services (ICWS ’06), pp. 91-98, 2006.

[10] ZENG, L. BENATALLAH, B., NGU, A., DUMAS, M., KALAGNANAM, J., AND CHANG, H. 2004. Quality-aware middleware for Web service composition. IEEE Trans. Softw. Eng. 30, 5, 311–327.

[11] Y. Liu, A. H. H. Ngu, and L. Zeng. Qos computation and policing in dynamic web service selection. In International World Wide Web Conference, pages 66–73, 2004.

[12] L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Z. Sheng. Quality driven web services composition. In International World Wide Web Conference, pages 411–421, 2003.

[13] L. Zeng, B. Benatallah, A. H. H. Ngu, M. Dumas,J. Kalagnanam, and H. Chang. Qos-aware middleware for web services composition. IEEE Trans. on Software Engineering, 30(5):311–327, 2004.

[14] G. L. Nemhauser and L. A. Wolsey. Integer and Combinatorial Optimization. Wiley-Interscience, New York, NY, USA, 1988.

[15] I. Maros. Computational Techniques of the Simplex Method. Springer, 2003.

[16] G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[17] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[18] I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[19] K. Elissa, “Title of paper if known,” unpublished.

[20] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[21] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[22] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[23] Electronic Publication: Digital Object Identifiers (DOIs):

Article in a journal:

[24] D. Kornack and P. Rakic, “Cell Proliferation without Neurogenesis in Adult Primate Neocortex,” Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/science.1065467.

Article in a conference proceedings:

[25] H. Goto, Y. Hasegawa, and M. Tanaka, “Efficient Scheduling Focusing on the Duality of MPL Representatives,” Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07), IEEE Press, Dec. 2007, pp. 57-64, doi:10.1109/SCIS.2007.357670.