http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 8 (2): 181-187, 2009
ISSN 1812-5638

Independent Global Constraints-Aware Web Service Composition Optimization

[1,2,3]Xianwen Fang, [1,2]Changjun Jiang and [1,2]Xiaoqin Fan
[1]Department of Computer Science and Technology, Tongji University, Shanghai, 201804, China
[2]Key Lab of Embedded System and Service Computing, Ministry of Education, Shanghai 201804, China
[3]Department of Information and Computer Science, Anhui University of Science and Technology, Huainan Anhui, 232001, China

Abstract: In semi-automatic service composition, developers should construct the process model according to concrete application requirement and then the instance services are bound automatically for every abstract task and make the composite service with optimal performance. This study presents independent global constrains-aware Web service composition method based on associate Petri net (APN) and genetic algorithm (GA). Firstly, an APN modeling methods which can describe multi-attribute multi-constraint relations and associate relation between component services is proposed. Secondly, combining with the properties of APN, GA is used to search a legal firing sequence with the biggest trust value in the APN model and the composite service corresponding to the legal firing sequence is the optimal solution. Finally, the experimental simulation is given out. Theoretical analysis and experimental results indicate that this method owns both lower computation cost and higher success ratio of service composition.

Key words: Associate petri net, genetic algorithm, web service composite, optimization

INTRODUCTION

At present, the Service composition has been a popular research. It is defined as the integration of a variety of existing services by certain process logic to satisfy the users' requirements more effectively. The composition approach can be classified into manual composition, semi-auto composition and automatic composition (Brahim and Athman, 2004). It is unrealistic to manually analyze and compose the existing services appearing with tremendous growth recently. In semi-automatic service composition, developers should firstly construct the process model and then the instance services are selected automatically for every abstract task.

It is well accepted that semantic-based approaches achieve higher recall than syntactical ones (Ardagna and Pernici, 2007). Semantic relationships considered in the process of service composition mainly include the consistency between the signatures of function or attribute oriented to text and the compatibility of the attribute type oriented to data and the satisfiability between attribute value and constraint condition oriented to numerical and so on. Locating a composite service would require accurate specifications of both service descriptions and user requests. Constraints are used in user requests to accurately describe the services that need to be located. They are of two types: local and global constraints. The former restricts the values of a particular attribute of a single service, whereas the latter simultaneously restricts the values of two or more attributes of multiple constituent services. Global constraints can be classified based on the complexity of solving them (i.e., determining the values that should be assigned to their attributes) as either strictly dependent or independent. A (global) constraint is strictly dependent if the values that should be assigned to all the remaining restricted attributes can be uniquely determined once a value is assigned to one. Services that conform to strictly dependent global constraints can be easily located in polynomial time (Gooneratne *et al*., 2007).

Any global constraint that is not strictly dependent, is independent. For example, available Date. Computer≤ approval Date. Insurance ≤date. Pickup is an independent global constraint. The class of composite services that conform to independent global constraints is probably the most interesting to study, as their location is known to be NP-hard (Zeng *et al*., 2004).

As a consequence most of the existing matching techniques (for locating composite services) do not consider independent global constraints (Gooneratna and Tari, 2008). Nonetheless, there are some that consider them and they use integer programming solutions focusing on local optimizations (Zeng *et al*., 2004) and AI planners (Yu *et al*., 2007) to efficiently locate conforming composite services. However, all the techniques of the latter type are syntactic-based

Corresponding Author: Xianwen Fang, Department of Computer Science and Technology, Tongji University, Shanghai, 201804, China

approaches. Gooneratne and Tari (2008) proposes a semantic-based matching technique that locates services conforming to independent global constraints. The proposed technique that incorporates a greedy algorithm performs better than the existing techniques. Experimental results also show that the proposed approaches achieve a higher recall than syntactic-based approaches. But the proposed greedy algorithm approaches are low efficiencies and the approximating method can not represent some problem accurately.

BASIC CONCEPTION

The concept of associate Petri net (APN) is derived from fuzzy Petri net (FPN). An APN is a directed graph, which contains three types of nodes: places, squares and transitions where circles represent places, squares represent thresholds of association degree and bars represent transitions. Each place may contain a token associated with a truth-value between zero and one. Each transition is associated with a trust value between zero and one. Directed arcs represent the relationships between places. The structures, reasoning and operations of APN are generalized from basic Petri net. Here, we only introduce several conceptions correlating with the paper close, other Petri Nets terms in the literature (Murata, 1989).

Definition 1: A 13-tuple APN = (P, T, S, D, Λ, Γ, I, O, C, α, β, W, Th) is called associate Petri net, where:

- $P = \{P_1, P_2, ..., P_n\}$ is a finite set of places
- $T = \{T_1, T_2, ..., T_i\}$ is a finite set of transitions
- $S = \{S_1, S_2, ..., S_m\}$ is a finite set of supports
- $D = \{D_1, D_2, ..., D_n\}$ is a finite set propositions
- $\Lambda = \{\tau_1, \tau_2, ..., \tau_m\}$ is a finite set of thresholds of the supports
- $\Gamma = \{r_1, r_2, ...r_i\}$ is a finite set of thresholds of the confidence
- $P \cap T \cap D = \Phi$; $|P|=|D|$
- I: $T \rightarrow P^\infty$ is an input function, a mapping from transitions to bags of places
- O: $T \rightarrow P^\infty$ is an output function, a mapping from transitions to bags of places
- C: $T \rightarrow [0, 1]$ is the confidence degree of relationship between zero and one
- α: $P \rightarrow [0, 1]$ is an associated function, a mapping from places to real values between zero and one
- β: $P \rightarrow D$ is an associated function, a bijective mapping from places to propositions
- W: $S \rightarrow [0, 1]$ is an associated function that assigns a real value between zero to one to each support
- Th: $S \rightarrow \Lambda^\infty$ is an associated function that defines a mapping from support to thresholds

Definition 2: In an APN, a transition may be enabled to fire. A transition t_i is enabled (represented as $M[t>$) if for all $p_j \in I(t)$, $\alpha(p_j) \geq \lambda$, $s_m \geq \tau$ and $c_i \geq \gamma_i$, where, λ, τ_m, γ_i are threshold values and λ, τ_m, $\gamma_i \in [0, 1]$. A transition t_i fires by removing the tokens from its input places to pass through all support squares and then depositing one token into each of its output places.

The token value in a place p_i, $p_j \in P$, is denoted by $\alpha(p_j)$, where $\alpha(p_j) \in [0, 1]$. If $\alpha(p_j) = y_i$ and $\beta(p_j) = d_i$, then it indicates that the proposition d_j is y_i.

If the antecedent portion or consequence portion of an associate Petri reasoning (APR) contains and ore connectors, then it is called a composite APR. The composite ARP can be distinguished into the following five basic reasoning rule types. Using this simple mechanism, all the APRs can be mathematically and graphically illustrated. By carefully connecting related place and assigning reasonable trust values to transitions, we can come up with an APN that can make a decision based on the expertise during its construction (Shih *et al*., 2007).

THE INDEPENDENT GLOBAL CONSTRAINT COMPOSITE SERVICES MODELING BASED ON APN

The realization of semi-automatic service composition demands that firstly operators make the universal service composition model in accordance to application demands according to given business background. A request consists of a composite service function template and a constraint model. The former specifies the function of services that need to be aggregated to form a composite service. It is defined as a collection of service function. Constraints are included in a request to provide an accurate description of the required services. The constraint model described here contains an independent global constraint. Such constraints are formed with a collection of binary attribute comparisons (binary constraints).

Definition 3 (Constraint attributes): The attributes that have semantic constraint relation are called as constraint attribute.

Definition 4 (Services): A service can be defined as WS = (basic attributes, functional attributes, constraint attributes, QoS), basic attributes are used to describe the basic description of the services, such as service providers, service names and so on, function attributes express as functional description of services, such as the I/O and so on, constraint attributes are the attributes that have semantic constraint relations among component services, QOS means the quality of service.

Definition 5 (Service request): Service request is the abstract description of the users' requirements, which can be expressed as a triple tuple RWS = (function requirements, constraint requirements, QoS requirements), where the function requirements is the function descriptions of service request, constraint requirements is the constraint description of service request, QoS requirements is the quality of service requirements description of service request.

The composite service acquisition consists of three phases: (1) candidate services acquisition, (2) constraint attributes identification and (3) optimal composite service acquisition. The first phase locates services with different functions description according to abstract task requirements in a composite service template. A candidate service that satisfies the requirement description of an abstract task is a service with a particular function. By locating candidate services, it ensures that the constituent services of a composite service are of appropriate types. The second phase identifies constraint attributes of services conforming to a given independent global constraint and certain model is used to represent the constraint relations. The proposed methods adopt the support nodes of APN to descript the constraint and it is benefit to analyze the multi-attributes and multi-constraints problem. In the final phase, according to the QoS requirement, the optimal candidate services that satisfy the function and constraint requirement are obtained to form the composite services.

The proposed approach models independent global constraints for composite Web services by the APN. The place nodes of the APN model of services composition represent the component services, the support nodes represent constraint relations between component services in adjacent process task and the transition nodes represent trust value between component services in adjacent process task. The modeling methods about the independent global constraints for composite Web services based on the APN are presented as follows:

- According to the domain knowledge, a common composite services process model including abstract tasks and the dependent relation between abstract tasks is built by the business personnel, the process model including abstract tasks and the dependent relation between abstract tasks
- For each abstract task, locate dynamically instance services from candidate services sets having same function attributes, each component services is represented by a place, the component services satisfying the same abstract task are put the same level
- The independent global constraint relations between constraint attributes of component services are represented by support nodes. The threshold value τ of the support node is the constraint attribute value of the former place node and the support value s is the constraint attribute value of the latter place node. If the s and τ satisfy the constraint relation requirement (such as $s \geq \tau$, $s \leq \tau$ or $s \in \tau$ and so on), then the former place is linked with the latter place by directed arc
- The associate relations between the instances services bound to adjacent process task are represented by transition nodes, the threshold value γ of the transition node is set by the composite needs and the trust value c is obtained by the data mining on the basis of the history calling relation. If the c is less than γ, then the path from the former place to the latter place is not taken into account during the services composition
- Taking advantage of the rules 1 to 4, an APN model can be obtained about independent global constraints for composite Web services. The problem of locating instance services is turned into searching the legal firing sequences in the APN model. Furthermore, the optimal composite services are the legal firing sequences having the largest trust value in the APN model

Example 1: Suppose a composite service has three abstract tasks, task 1 has two candidate services whose function attributes are consistent semantically with the abstract task, likewise, task 2 has three candidate services, task 3 has two candidate services. The full APN model is depicted in Fig. 1, when some constraint conditions can not be satisfied; some corresponding paths are excluded.

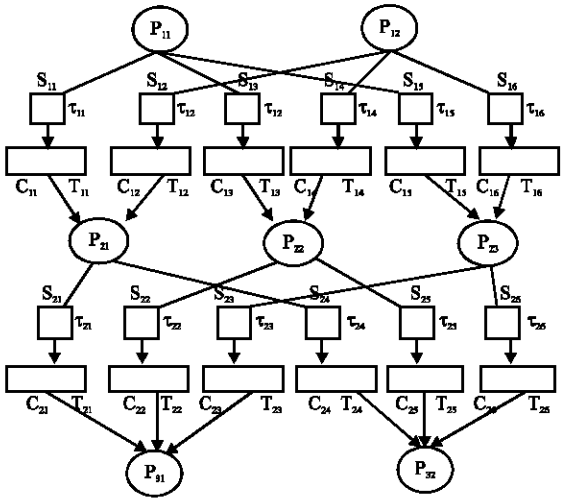

Fig. 1: The APN model of example 1

THE SERVICE COMPOSITION OPTIMIZATION WITH INDEPENDENT GLOBAL CONSTRAINT BASED ON GA

The study makes full use of APN model, when encoding chromosomes, use transition elements of APN model (i.e., operation serial met constraint conditions) rather than elements of problem space (e.g., each service), this type of encoding can reduce locating space effectively on this level of granularity so that efficiency of algorithm execution is improved. Concretely, regarding firing sequences of APN as chromosomes, every legal firing sequence in Petri Net model represent a feasible Web services composition. It can make service composition algorithm closely related to Petri Net models, which turns the problem of composite service optimum which is independent global constraint into the problem of searching for the best firing sequences based on the APN model. So, the problem to be solved can be converted into the corresponding APN model, then the best firing sequence can be found by optimizing legal sequence with GA and in the searching process the problem can be tackled with the full use of the legal sequence algorithm of Petri net. If simply adopting Petri Net to do composite operation, it is very complex when the scale of composition is comparatively large. If GA method is adopted only, searching has comparatively large randomness. It goes against resolving problems effectively. This method takes these two aspects into consideration simultaneously, which reduces the complication and searching randomness effectively.

Definition 6 (Composite service): A service tuple $CS = [ws_1, ws_2, ..., ws_m]$ is a composite service that conforms to independent global constraint in the APN model if:

- $\forall ws_i$, $ws_i \in P_{ik}$ $(1 \le i \le m)$, here, k is the place number of the i-th level, P_{ik} is the place in the APN model
- $\forall ws_j$ $(1 \le j \le m-1)$, ws_j is described using an attribute a'_i where, $a'_i \in (a_1, ..., a_n)$ and ws_{j+1} is described using an attribute a'_{i+1}, where, $a'_{i+1} \in (a_1, ..., a_n)$, $S_{j(j+1)}$ is support relation between the constraint attributes a'_j and a'_{i+1}, which conforms to global constraint
- $\forall ws_j$ $(1 \le j \le m-1)$, the trust value $C_{j(j+1)}$ is obtained by the data mining on the basis of the history calling relation between ws_j and ws_{j+i}, the threshold value $\gamma_{j(j+1)}$ is set by the composition needs, $C_{j(j+1)} \ge \gamma_{j(j+1)}$

The optimization methods of independent global constraint for composite service based GA and APN are as follows:

Chromosome coding: According to the requirement of composite process, we select fixed number (such as K) firing sequences from APN model with constraint as the first population of the GA. For an APN, if it exists a firing sequences $\sigma = t_1 t_2 t_3 ... t_{n-1} t_n$, which makes $M_0[\sigma > M_f$, then σ can be regarded as a chromosome (it is a legal firing sequences satisfying the requirement of service composition), here, M_0 is the first marking, M_f is the end marking, $t_1, t_2, t_3, ..., t_{n-1}, t_n \in T$.

Fitness function: According to the reasoning rules of APN, the trust value of every chromosome σ through reasoning can be computed, meanwhile, it is the trust value of corresponding composite service, we call as the degree of truth TD (σ). Because GA requires that better solutions have larger fitness degree (Prashant Reddy *et al*., 2001) and $0 \le TD(\sigma) \le 1$, so we define the fitness function as follows:

$$f(\sigma) = \begin{cases} TD(\sigma) & 0 \le TD(\sigma) < 1 \\ 0 & \text{others} \end{cases}$$

Selection operator: In the algorithm, we take the trust value ratio as the selection operator, so the number of the present population existing in next generation is:

$$M = \frac{C_{ij}}{\sum C_{ij}} \cdot n$$

here, n is the number of father generation.

Crossover operator: The operation of the crossover operator as follows:

- Putting two chromosomes randomly as a group, suppose two chromosome are σ and σ', respectively
- Swapping the same position transition of two firing sequences corresponding chromosome, generating some new firing sequences
- Checking the new firing sequences generating in step (2) by using the legal firing sequences algorithm of Petri net, reserving legal firing sequences, excluding some illegal firing sequences

Mutation operator: In GA, the aim of introducing the mutation operator is to make GA having local seeking ability. When GA is close to the neighbor domain of the optimal solution by the crossover operator, it can converge to the optimal solution quickly by using the local seeking ability of mutation operator.

Due to adopting the firing sequences as the chromosome in the study, if having a transition of firing

sequences is changed, it is possible to lead to the chromosome is not feasible solution, so we take advantage of the firing sequences algorithm (Changjun, 2002) to check some changed chromosome. The operation of the mutation operator as follows:

- Selecting a locus from chromosome, $\sigma = t_1t_2t_3...t_{n-1}t_n$ randomly, suppose it is $t_pt_p+1...t_q$, putting the marking corresponding to the locus into table of marking (MT)
- Taking a marking M from MT
- Finding all enabled transition under the marking M
- Enabling every enabled transition, producing the child marking M', then computing the trust value of M', here, M is the parent marking
- For each child marking M', if M' is the end marking, we put M' into table of result (RT), else, we put M' into table of child (CT), then goto 3
- Selecting the first N (N is a natural number) marking by the trust value of marking and putting them into MT, excluding other markings, then goto 3
- Taking a marking M_q having the largest trust value from RT, obtaining the transition and parent marking which generate the marking M_q by firing rules, using the methods recursively, a firing sequence can be obtained
- Joining the firing sequences $t_1t_2...t_{p-1}$ before mutation point in the chromosome with the firing sequences obtained in 7, getting a mew chromosome σ' which is a chromosome after mutation
- According to the legal firing sequences algorithm of Petri net, checking the mutation chromosome, reserving legal firing sequences, then computing the optimal firing sequences by APN reasoning rules

EXPERIMENT SIMULATION

The methods of independent global constraint service composition based on GA and APN is proposed, which not only uses APN superiority in the description multi-attribute multi-constraint problems, but also takes fully Petri net's properties when GA locating in the APN model into account. In the theoretical, the method is of great benefit to analyze independent global constraint composite service question, which can avoid high complexity using Petri net methods solely and large randomness using GA only. Next, we use experiment simulation so as to analyze feasibility and validity of the methods. Experiment environment: CPU is Intel dual 1.60 GHz, Memory is 1.00 GB and operation system is Windows XP.

For a given service requirement, using the greedy methods, approximate methods presented in the literature (Gooneratne and Tari, 2008) and the methods based on GA and APN(GA and APN), respectively, the experiment effect (Fig. 2) shows the execution time of GA and APN methods is less than the greedy methods and the approximate methods and when the more of the service number in the service library, the better is the effect. The reason is that we uses fully APN's properties when GA locating in the APN model, which makes locating space lower and some component services which don't satisfy constraint relations or have minor associate relation need not to match each other. So, it can save execution time and improve the time performance of service composition.

In Fig. 3, we locate feasible candidate services from service library (N = 500) and compare the relation graph between the ratio of solution and iteration number. The experiment effect shows the ratio of solution of GA and APN methods is higher than the approximate

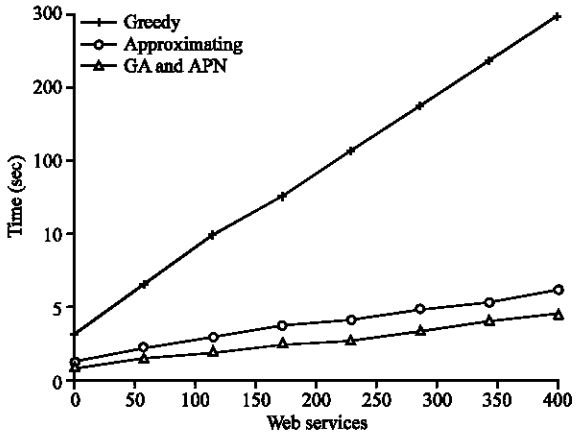

Fig. 2: The execution time of three methods

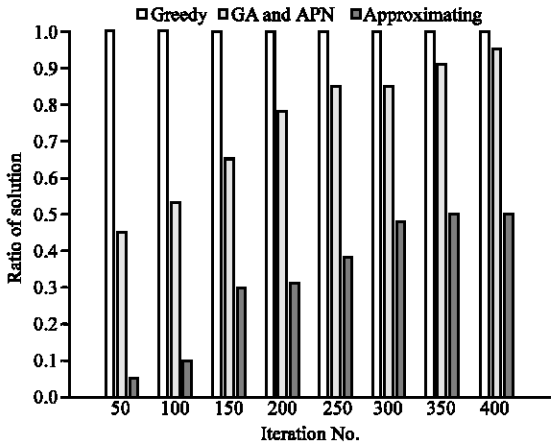

Fig. 3: The relation graph between the ratio of solution and iteration number

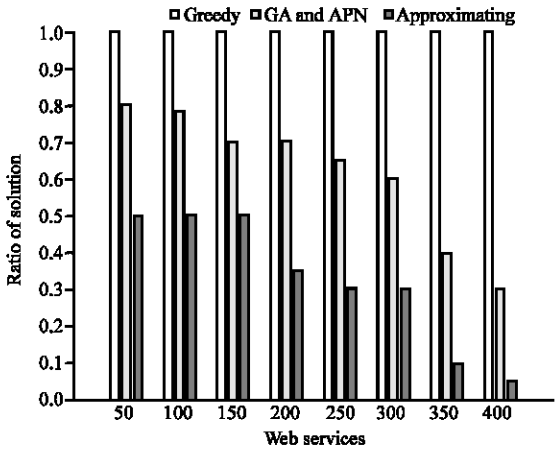

Fig. 4: The relation graph between the ratio of solution and web services

methods, because the greedy methods is an enumerate methods, it can locate all solutions, but it need most time. With the iteration number adding, the ratio of solution of GA and APN and approximating methods increase quickly, but it is flat when iteration number reaches to some extent. In the GA and APN methods, the process of locating the feasible solution is the same as locating the firing sequences in APN model. Because of taking the Petri net legal firing sequences algorithm into account, the methods can guarantee the located solution is feasible. What's more, owing to the selection, crossover and mutation operation of GA, the methods can reduce the loss and miss of feasible solution.

In Fig. 4, we locate feasible candidate services from service library and compare the relation graph between the ratio of solution and web services. the experiment effect shows the ratio of solution of GA and APN methods is higher than the approximate methods, but lower than greedy methods. With the web services number adding, the ratio of solution of GA and APN and approximating methods decrease quickly.

CONCLUSION

For independent global constraint service composition problem, the main research methods are syntactic matching and semantic matching presently. There are index methods and approximate methods to deal with global constraint, which play a good effect in resolving constraint problems, but it is deficient to deal with multi-attribute and multi-constraint problems.

The service composition methods based on GA and APN are presented in the study. Firstly, we model the independent global constraint component services by APN, in APN model, multi-attribute and multi-constraint relations are represented using support set and the associate relation between component services is obtained by the data mining on the basis of the history calling relation. Secondly, using GA to locate legal firing sequences in the APN model, the legal firing sequences are feasible solutions and then according to the reasoning rules of APN, an optimal solution is obtained. Finally, we simulate the execution time and the ratio of solution based greedy methods, approximate methods and GA and APN methods, respectively. Experiment simulation effect shows GA and APN have higher execution efficiency and the success rate of service composition.

The study aims at semi-automatic service composition in the study. In the future, we plan to resolve the independent global constraint for automatic service composition and propose improved GA and APN methods to improve execution efficiency.

ACKNOWLEDGMENTS

We would like to thank the support of the National Natural Science Foundation of China under Grant No. 90718012, No. 60873144 and No. 90818023, the National High-Tech Research and Development Plan of China under Grant No. 2007AA01Z136.

REFERENCES

Ardagna, D. and B. Pernici, 2007. Adaptive service composition in flexible processes. IEEE Trans. Software Eng., 33: 369-384.

Brahim, M. and B. Athman, 2004. Composing web services on the semantic web. VLDB J., 12: 333-351.

Changjun, J., 2002. A polynomial-time algorithm for the legal firing sequences problem of a type of synchronous composition petri net. Sci. China Series E, 22: 116-124.

Gooneratne, N., Z. Tari and J. Harland, 2007. Matching strictly dependent global constraints for composite web services. Proceedings of the European Conference on Web Services, Nov. 26-28, Washington, DC, USA., pp: 139-148.

Gooneratne, N. and Z. Tari, 2008. Matching independent global constraints for composite web services. Proceedings of the 17th International Conference on World Wide Web, Beijing, China, Apr. 21-25, ACM New York, USA., pp: 765-774.

Murata, T., 1989. Petri nets-properties, analysis and applications. Proc. IEEE, 77: 541-580.

Prashant Reddy, J., S. Kumanan and O.V. Krishnaiah Chetty, 2001. Application of petri nets and a genetic algorithm to multi-mode multi-resource constrained project scheduling. Int. J. Adv. Manufact. Technol., 17: 305-314.

Shih, D.H. H.S. Chiang and B. Lin, 2007. A generalized associative petri net for reasoning. IEEE Trans. Knowledge Data Eng., 19: 1241-1251.

Yu, T., Y. Zhang and K.J. Lin, 2007. Efficient algorithms for web services selection with end-to end QoS constraints. ACM Trans. Web, Vol. 1. 10.1145/1232722.1232728

Zeng, L., B. Benatallah, A. Ngu and M. Dumas, 2004. QoS-aware middleware for web services composition. IEEE Trans. Software Eng., 30: 311-327.