

Research Article

A Knowledge-Based Service Composition Algorithm with Better QoS in Semantic Overlay

Huijun Dai,¹ Hua Qu,¹ and Jihong Zhao^{1,2}

¹*School of Electronic Information, Xi'an Jiaotong University, Xi'an, Shaanxi 710049, China*

²*Department of Communication Engineering, Xi'an University of Posts and Telecommunications, Xi'an, Shaanxi 710061, China*

Correspondence should be addressed to Huijun Dai; ddahjun@gmail.com

Received 1 May 2014; Revised 21 August 2014; Accepted 22 August 2014

Academic Editor: Chuandong Li

Copyright © 2015 Huijun Dai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A semantic overlay network (SON) is a visual framework clustered under similar metaknowledge units such as ontologies, algorithms, and rule engines. Knowledge-based service composition (KC) has become a prominent aspect of building new and creative composed service through a combination of semantically similar information at the knowledge level. In this study, a promising approach to construct a standard knowledge model is developed to utilize the progress of KC. To evaluate and optimize the composition, we define the quantity of service (QoS) regarding user requirements in the KC instance, and a KC instance path with better QoS is found in the model using the KC algorithm. Simulation results prove that our approach has a tradeoff between efficiency and equality.

1. Introduction

Knowledge-based applications are organized according to semantic relevance and context in semantic overlay network (SON) with the purpose of retrieving the service in a large-scale intelligent information system [1]. In a SON, many groups of nodes with similar content or documents compose reusable software service into an integrated heterogeneous virtual platform for sharing information [2]. Content-addressable network (CAN) is also essentially a type of SON via content cached at nodes based on content routing for locating content [3, 4]. With the accumulation of massive amounts of service or application from many different domains (e.g., scientific research and personal and corporate communications or interactions [5]) for specific service supplying, the SON aims to thoroughly bind such individual and distributed services into workflow systems that support different kinds of knowledge sharing [6]. The approach of composition must support the evaluation of quality of service (QoS) to optimize the composition effect of reusable software. Therefore, the purpose of knowledge-based composition

(KC) is not different from that of traditional service composition because the former can reuse the more units (e.g., databases, algorithms, and tools from different fields which could also be regarded as individual service [7]) in semantic level. While, knowledge-based service composition makes information sharing more dynamic, flexible, and scalable.

Therefore, KC is intended to share a wider range of information or services scattered across numerous distributed nodes. There are multiple QoS attributes that users pay attention to including price, reputation, reliability, and delay [8]. Although previous studies related to better QoS have contributed significantly to service composition in semantic web, most of them are still limited to traditional semantic web service and do not extend to overlay based on knowledge. This study aims to find a novel way to compose service based on knowledge sharing with better QoS in an SON.

The rest of this paper is organized as follows. The approach is formally modeled and related studies are reviewed in Section 2. Section 3 presents the adopted KC model in a SON. The exhaustive algorithm process is detailed and implemented in Section 4. In Section 5, the algorithm

is examined and evaluated by simulation. The conclusion is provided in Section 6.

2. Related Works

The massive amounts of distributed service or units such as those available from digital libraries and web-accessible text databases motivate researchers to work toward efficient service management in highly distributed environments [9]. By correlating these isolated islands of information, individuals can gain new insights, discover new relationships, and produce more knowledge [10]. A large body of work on knowledge-based service composition in SONs has been generated [6, 9, 11–13]. Certain approaches use algebra in composing ontologies that represent the terminologies from distinct, typically autonomous domains to compose knowledge [14, 15]. Usually, the links of ontology among the domains are called intersections [16].

Knowledge-based service composition is different from knowledge discovery [17] or knowledge creation [18] because that the former enable users to collectively create, share, browse, and query their own individual knowledge [13, 19, 20]. Some recommendations or recommender systems compose a set of services into data processing logic which combine the service according to user preferences or its execution environment [15].

Significant research has focused on optimizing the QoS of service composition [21, 22]. Existing approaches often show a tradeoff in every aspect to gain balanced performance [23, 24]. In fact, KC can be implemented and utilized in e-ScienceNet [25], e-Flow with customized services [26]. Cross-enterprise collaboration has emerged using automatic business composition in the global market [27]. In the scientific research, medical information, and collaborative manufacturing, the creative approach of the KC for a SON is also presented [28] for searching service-oriented knowledge utilities [29]. The process of KC should include information retrieval, knowledge representation, and databases, as well as exploiting pieces of knowledge from the multitude of heterogeneous sources available. In [30], the candidates with the same functionality are organized as a class and ACO4WS is used to test the feasibility and soundness of the approach.

Based on the aforementioned study, such a conclusion can be reached. Firstly, SONs have received growing attention as a scalable and distributed platform for knowledge-sharing applications. Then, KC dynamically combines two or more basic services (called units) into potentially complex value-added knowledge-based workflow to fulfill the functional requirements of the user. Finally, the key to KC is to choose a scheme with better QoS under the QoS constraints.

3. Problem Formulation

In knowledge repository, a unit (subservice) can be expressed as $K = \{I, O, Q\}$, I is the input set, O represents the output set, and Q is the QoS. The QoS of a KC consists of the following parameters: $\{Q_1, Q_2, \dots, Q_n\}$.

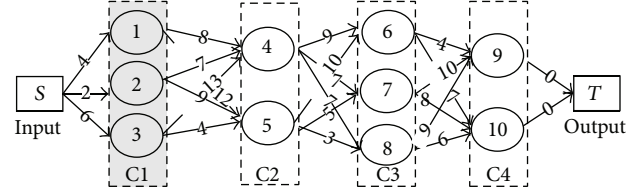


FIGURE 1: Units in SON.

The QoS attributes Q_i are availability, cost, time overhead, and so on. The model can be classified into subservice and compositional service. The subservice is a single unit cached in SON just like the one shown in Figure 1. The model enables the incorporation of various composition approaches by starting from the input of the first unit, traversing through the chosen unit, and ending at the output of the final unit in different knowledge repositories.

N different but semantically related subservice constructs a body of compositional service from M SONs, and KC involves mapping from the composite template to instantiated knowledge. The previous subservice output should be consistent with the next subservice input at the knowledge level. Each subservice in an SON has a few candidates called units, which are those that have the same or similar meaning or function. In Figure 1, points 1, 2, and 3 are candidates in SON1. The points and links between them can form an indirect graph from source to target. A weighted link between points represents the QoS of discovering or retrieving the next node. The QoS labeled in links merges the different QoS attributes of nodes using a cost function. The KC with different QoS indicates different paths in the knowledge graph. So, we can get the following.

The QoS of KC indicates the overall QoS of the instantiated knowledge path, which can be calculated as follows: $Q_{\text{com}} = \sum_{i=1}^n Q_i$, where Q_i represents the merged QoS of nodes and links. For example, the availability and time of the overall QoS of composition can be expressed as

$$\ln \frac{1}{A_{\text{com}}} = \sum_{i=1}^n \ln \frac{1}{A_{\text{node}(i)}} + \sum_{j=1}^{n-1} \ln \frac{1}{A_{\text{link}(j)}}, \quad (1)$$

$$\text{Time}_{\text{com}} = \sum_{i=1}^n \text{RTT}_i + \sum_{j=1}^{n-1} \text{Delay}_j.$$

$A_{\text{node}(i)}$ refers to the availability of node i , $A_{\text{link}(j)}$ stands for the availability of the conjoint link j , and RTT_i and Delay_j represent the round trip time of node and links' delay. Given the semantic similarity of the nodes, their QoS in the same SONs can be considered as equal, and the QoS of the links are not equal because of the different locations. Because the QoS of KC path can translate into the additive metrics, the best QoS of KC also indicates the smallest QoS of links in the knowledge graph.

The KC progress can be formulated based on the knowledge graph model shown in Figure 2 which is converted from Figure 1. In Figure 2, the QoS has been marked in links.

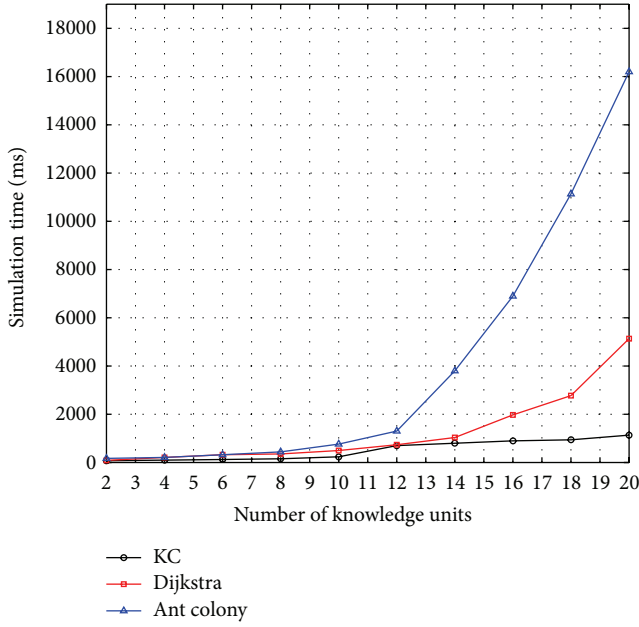


FIGURE 2: Time curves with metaknowledge.

TABLE 1: Adjacent matrix 1.

	S	1	2	3
S	0	4	2	6
1		0	0	0
2			0	0
3				0

4. KC Approach with Better QoS

The target of KC is to find an optimal path in the graph by comparing the total QoS as the cost function. Instead of searching the entire SON, the KC algorithm involves three basic steps: probing and collecting the information; path; backwording.

4.1. An Instance. There is an instance to show how this approach works. In probing, matrices of different nodes in neighbor SONs are constructed filling with the QoS values of the links. Because the knowledge model graph is indirect, the constructing matrix is symmetrical.

During the probing, the constructing matrixes are as in Tables 1, 2, 3, and 4.

The searching results are saved with link-table.

During probing and constructing, we can find an initial path. In this example, from the node of S, the next node of 2 should be chosen. Then from the node 2, the node of 4 should be chosen. Thus, the node sequence is $S \rightarrow 2 \rightarrow 4 \rightarrow 7 \rightarrow 10 \rightarrow T$, in which the QoS of the total path is the sum of each link's QoS indicated as *Cal_sum*. Now, the *Cal_sum* is "24" in this example.

During the path, the minimum is found and chosen from the second matrix of Table 2. The nodes with minimum link are located, then search the shortest link starting from the

TABLE 2: Adjacent matrix 2.

	1	2	3	4	5
1	0	0	0	8	12
2		0	0	7	9
3			0	13	4
4				0	0
5					0

TABLE 3: Adjacent matrix 3.

	4	5	6	7	8
4	0	0	9	7	7
5		0	10	5	3
6			0	0	0
7				0	0
8					0

TABLE 4: Adjacent matrix 4.

	6	7	8	9	10
6	0	0	0	4	7
7		0	0	10	8
8			0	9	6
9				0	0
10					0

nodes. The searching include forward and backward until a path was been chosen.

The backwording progress will traverse all the matrixes and give out different paths with better QoS in their SON. Then, there are M (the number of SON) paths with their nodes and QoS are saved and compared. The final *Cal_sum* of smallest QoS is 19, and the node sequence is $S \rightarrow 3 \rightarrow 5 \rightarrow 8 \rightarrow 10 \rightarrow T$.

In this example, the KC instance is found in which the path with smaller QoS is obtained through a simple linear search algorithm. The time complexity does not depend on the number of SONs (matrixes). This approach may not find a path with best QoS because of the local optimization. But it will get a tradeoff between efficiency and equality. Although the nodes in knowledge model graph continually grow with the number of metaknowledge units, the time complexity of this approach will not soar but increase linearly.

5. Simulation and Analysis

5.1. Simulation Environment. To evaluate the performance of proposed KC algorithms with better QoS, we use them in the specified application. We suppose that the application has certain user requirements as input and the expected results as the output. The number of metaknowledge units spreading across different specific P2P systems is less than 20 (including source and target), and each metaknowledge has a few candidates with the same meaning, input, and output. Then, the workflow graph of this application can be formulated as shown in Figure 2, where N , ($N < 20$)

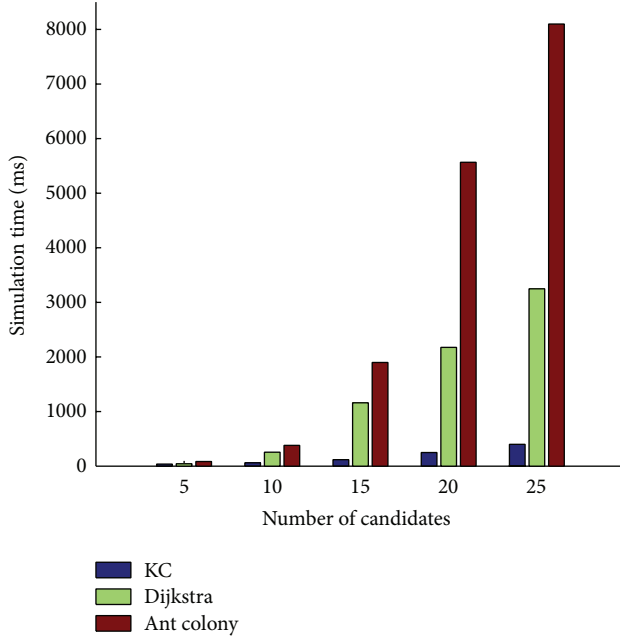


FIGURE 3: Time curves with candidates.

groups of nodes are shown. The underlying physical topology is generated by GT-ITM using n -node transit-stub models. To simulate the performance variation in the real world and the initial availability of each node, we distribute the QoS of each link uniformly in a certain range.

To verify the effectiveness of the proposed approach (KC), we compare it with two types of algorithms: one is the shortest path algorithm with optimal results and the other is the heuristic algorithm (ant colony) with local optimal results in service composition. We have attempted to provide the same conditions (e.g., interface and input data) for three kinds of algorithms.

5.2. Simulation Results. With regard to the optimal algorithm, Dijkstra and ant colony algorithms require more time and calculation to obtain the solution. Therefore, from the perspective of theoretical analysis, the time complexity of the Dijkstra $\Theta(N^2)$ and the time complexity of traditional ant colony algorithms were greater than that of our approach. These two solutions employ the global optimization approach but do not show promising running speed. The time complexity of the KC algorithm increases linearly.

For the sake of a number of candidates in many SONs, which continuously increase along with the growth of the SON, by taking more storage space in computing service (numerous matrices with the QoS of links), the approach obviously has quick convergence speed. However, the KC algorithm is not optimal. In the worst case, the iterations would reach $\Theta(N)$. Three simulation time curves of those algorithms which would vary with the numbers of meta-knowledge units or the candidates are shown in Figures 2 and 3. The number of metaknowledge units influences our approach, whereas the number of the candidates influences the two other approaches.

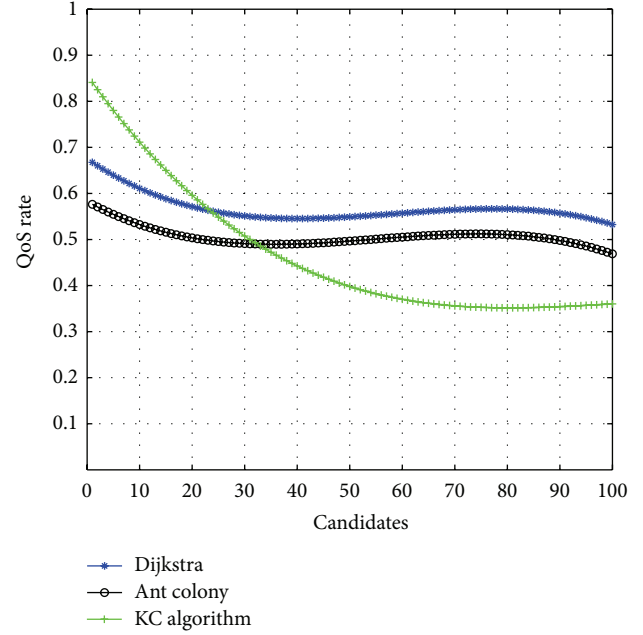


FIGURE 4: QoS rate of three algorithms.

The metric for evaluating the KC approach performance is the QoS rate. This rate is defined as

$$\text{QoS rate} = \frac{\text{numbers of satisfied instances}}{\text{numbers of required}}. \quad (2)$$

For this target, our approach has a lower QoS rate just like what is shown in Figure 4. This figure shows that our QoS rate is less than that of the other algorithms, particularly when a number of candidates exist. Comparing with other algorithms, our approach simplified the composition progress at the cost of QoS rate.

We also implement another two common heuristic algorithms used in service composition for comparison. One is ACO4WS [30], which is based on a general knowledge model, and the other is mixed integer linear programming, an algorithm for semiautomatic merging and alignment ontologies. Both algorithms are applicable to a wide range of knowledge representation and ontology development systems. Three algorithms are based on a different model, but the similarity is that all of them map the function modules into node graphs, and all focus on the matrix of QoS. Three simulation time curves of those algorithms varied with the candidates, as shown in Figure 5.

The experimental results indicate that the QoS rate of each algorithm decreases slightly with the increase of function modules because finding a path in a more complex graph is difficult. However, the QoS rate of KC has a higher value than that of the two other algorithms because KC uses a simple principle to measure QoS (smaller is better), which can fulfill most of the QoS attributes.

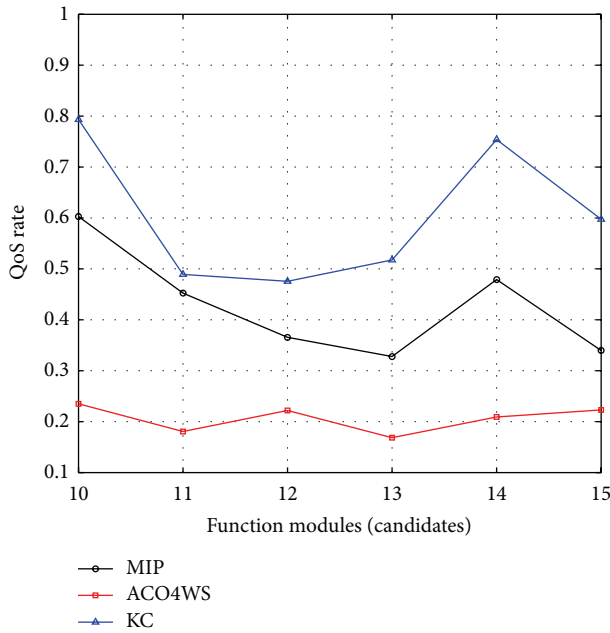


FIGURE 5: QoS rate with three local optimization methods.

6. Conclusions

In this study, we investigated the KC algorithm in the SON based on distributed knowledge systems. Unlike various existing studies, our work converts the problem into two sections for analysis. First, the knowledge-based service composition is transformed into the instance path in the graph with a sequence work flow with interface between the previous output and successor input. Second, the KC algorithm with better QoS is transformed into the shortest path problem, which has a local optimization result related to the KC model. Regarding the performance shown in Section 5, we believe that the proposed approach can open a wide range of new studies intended to improve the performance of knowledge-based information sharing or create new combined service across various disciplines (e.g., digital biomedical and aerospace material industries). We intend to study another algorithm to address such problems in the future. Such an algorithm with better QoS can be solved by adapting other optimization algorithms as part of our future work.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] A. Crespo and H. Garcia-Molina, "Semantic overlay networks for P₂P systems," in *Agents and Peer-to-Peer Computing*, vol. 3601 of *Lecture Notes in Computer Science*, pp. 1–13, Springer, Berlin, Germany, 2005.
- [2] R. Yin, "Study of composing web service based on SOA," in *Proceedings of the 2nd International Conference on Green Communications and Networks 2012 (GCN 2012): Volume 2*, vol. 224 of *Lecture Notes in Electrical Engineering*, pp. 209–214, Springer, Berlin, Germany, 2013.
- [3] Z. Zhengdong, H. Yahong, W. Weiguo, W. Yong, and L. Zengzhi, "CAN-based P2P semantic web services composite architecture," *Journal of Xi'an Jiaotong University*, no. 2, pp. 6–10, 2010.
- [4] A. Fiat and J. Saia, "Censorship resistant peer-to-peer content addressable networks," in *Proceedings of the 13th ACM-SIAM Symposium on Discrete Algorithms (SODA '02)*, pp. 94–103, SIAM, Philadelphia, Pa, USA.
- [5] C. Tang, Z. Xu, and S. Dwarkadas, "Peer-to-peer information retrieval using self-organizing semantic overlay networks," in *Proceedings of the ACM Conference on Computer Communications (SIGCOMM '03)*, pp. 175–186, New York, NY, USA, August 2003.
- [6] A.-L. Lamprecht, T. Margaria, and B. Steffen, "Bio-jETI: a framework for semantics-based service composition," *BMC Bioinformatics*, vol. 10, supplement 10, article S8, 19 pages, 2009.
- [7] A.-L. Lamprecht, T. Margaria, and B. Steffen, "Bio-jETI: a framework for semantics-based service composition," *BMC Bioinformatics*, vol. 10, supplement 10, article S8, 19 pages, 2009.
- [8] A.-L. Lamprecht, T. Margaria, and B. Steffen, "Bio-jETI: a framework for semantics-based service composition," *BMC Bioinformatics*, vol. 10, supplement 10, article S8, 2009.
- [9] B. Aleman-Meza, C. Halaschek, and I. B. Arpinar, *Collective Knowledge, Composition in a Peer-to-Peer Network: A survey for Peer-to-Peer applications*, Idea-Group, 2005.
- [10] G. Deconinck and K. Vanthournout, "Agora: a semantic overlay network," *International Journal of Critical Infrastructures*, vol. 5, no. 1-2, pp. 175–195, 2009.
- [11] Y. Mao, B. T. Loo, Z. Ives, and J. M. Smith, "MOSAIC: declarative platform for dynamic overlay composition," *Computer Networks*, vol. 56, no. 1, pp. 64–84, 2012.
- [12] P. Küngas and M. Matskin, "Semantic web service composition through a P₂P-based multi-agent environment," in *Agents and Peer-to-Peer Computing*, Z. Despotovic, S. Joseph, and C. Sartori, Eds., vol. 4118 of *Lecture Notes in Computer Science*, pp. 106–119, Springer, Berlin, Germany, 2006.
- [13] C. Radeck, A. Lorz, G. Blichmann, and K. Meißner, "Hybrid recommendation of composition knowledge for end user development of mashups," in *Proceedings of the 7th International Conference on Internet and Web Applications and Services (ICIW '12)*, 2012.
- [14] S. R. Chowdhury, F. Daniel, and F. Casati, "Efficient, interactive recommendation of mashup composition knowledge," in *Service-Oriented Computing*, vol. 7084 of *Lecture Notes in Computer Science*, pp. 374–388, Springer, Berlin, Germany, 2011.
- [15] M. Bajaj, "Knowledge composition methodology for effective analysis problem formulation in simulation-based design," ProQuest, 2008.
- [16] J. Gama, *Knowledge Discovery from Data Streams*, hapman & Hall/CRC Data Mining and Knowledge Discovery Series, Chapman & Hall/CRC, Boca Raton, Fla, USA, 2010.
- [17] J. Gama, *Knowledge Discovery from Data Streams*, Chapman & Hall, CRC Press, Boca Raton, Fla, USA, 2010.
- [18] S. Kalasapur, M. Kumar, and B. A. Shirazi, "Dynamic service composition in pervasive computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 7, pp. 907–918, 2007.

- [19] K. Nahrstedt and W. Balke, "A taxonomy for multimedia service composition," in *proceedings of the 12th ACM International Conference on Multimedia*, pp. 88–95, New York, NY, USA, October 2004.
- [20] X. Gu, K. Nahrstedt, R. N. Chang, and C. Ward, "QoS-assured service composition in managed service overlay networks," in *Proceedings of the 23rd International Conference on Distributed Computing Systems*, pp. 194–201, 2003.
- [21] F. Mardukhi, N. NematBakhsh, K. Zamanifar, and A. Barati, "QoS decomposition for service composition using genetic algorithm," *Applied Soft Computing Journal*, vol. 13, no. 7, pp. 3409–3421, 2013.
- [22] J. Liao, Y. Liu, X. Zhu, T. Xu, and J. Wang, "Niching particle swarm optimization algorithm for service composition," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '11)*, pp. 1–6, Houston, Tex, USA, December 2011.
- [23] C. Bussler and A. Haller, *Towards P2P-Based Semantic Web Service Discovery with QoS Support*, Springer, Berlin, Germany, 2006.
- [24] T. Classe, R. Braga, F. Campos, and J. M. N. David, "A semantic peer-to-peer network for service composition in scientific domains," in *Proceedings of the International Conference on Project Management/ (HCIST '13), International Conference on Health and Social Care Information Systems and Technologies, Procedia Technology*, vol. 9, 2013, pp. 215–225.
- [25] F. Casati, S. Ilnicki, L. J. Jin, V. Krishnamoorthy, and M.-C. Shan, "Adaptive and dynamic service composition in eFlow," in *Advanced Information Systems Engineering*, vol. 1789 of *Lecture Notes in Computer Science*, pp. 13–31, 2000.
- [26] F. Skopik, D. Schall, and S. Dustdar, "Managing social overlay networks in semantic open enterprise systems," in *Proceedings of the International Conference on Web Intelligence, Mining and Semantics (WIMS '11)*, May 2011.
- [27] M. Gharzouli and M. Boufaida, "PM4SWS: a P2P model for semantic web services discovery and composition," *Journal of Advances in Information Technology*, vol. 2, no. 1, pp. 15–26, 2011.
- [28] C.-H. Wu, K. Chiang, R.-J. Yu, and S.-D. Wang, "Locality and resource aware peer-to-peer overlay networks," *Journal of the Chinese Institute of Engineers*, vol. 31, no. 7, pp. 1207–1217, 2008.
- [29] P. Świątek, P. Stelmach, A. Prusiewicz, and K. Juszczyszyn, "Service composition in knowledge-based SOA systems," *New Generation Computing*, vol. 30, no. 2-3, pp. 165–188, 2012.
- [30] D. Ardagna and B. Pernici, "Adaptive service composition in flexible processes," *IEEE Transactions on Software Engineering*, vol. 33, no. 6, pp. 369–384, 2007.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

