An Automatic Web Services Composition Model Using Ant Colony System

Ayman I. Srour, Zulaiha Ali Othman, and Abdul Razak Hamdan

***Abstract*—Web and information grow in size and diversity. Likewise, Web services become difficult to be located or discovered regarding the huge number of the Web services exists today. Thus, the end user may face problem to find suitable Web services that meet his request. Usually, a skilled user or developer is required in order to discover and compose Web services. Increasingly, there is a need to automate Web Services aspects such as discovery and composition in such a way that enhance the interaction between services and the potential user. The proposed study aims to investigate the efficiency of using various type of Swarm Intelligent (SI) algorithms with somatic web technology in the Web services composition and also aims to investigate how can improve the overall composition process by applying an effective Web services discovering strategy based on semantic Web technology; on order to reduce the search space of candidate Web services.**

***Index Terms*—Semantic web services composition, user-centric, ant colony optimization.**

I. Introduction

In the recent years, Services integration and interoperability based on Service Oriented Architecture (SOA) have been focus on technical level and Business to Business (B2B) collaboration with totally ignore of Business to Consumer (B2C) [1]. The lack consideration of business to consumer orientation presents a shortcoming for end user to be involved in services development process [2]. This gap between end users and Web services prevents end users to play a productive role in Web service composition process. Therefore, this shortcoming hinders the emergence of user centric Services paradigm [3]. In addition, Web services architecture is mainly based on a complex set of standards that are not user friendly. Such as Web Service Description Language (WSDL) [4], SOAP as access protocol, Universal Description, Discovery and Integration (UDDI) [5] and BPEL as a composition and integration tool. However, all of these sanders are not understandable for end users.

Web services become difficult to be located and discovered due to the huge number of Web services that exists today, end user may face problem to find a suitable Web services that meet his requirements. Usually, a skilled user or programmer is required in order to discover and compose Web services, this make the possibility of involving end user in composition process very difficult task [6].

The current technical aspects of service composition are not of much interest to end users who want to utilize from the benefits offered by SOA. What really matters to end users is how these technologies are presented to them and how they can easily use these technologies to achieve their desired goals [7]. Thus, in Web services composition model, we argue that two main factors may effects and add more flexibilities to end user. First, visualization, apply visual component to identify user requirements. Second, automation, automate the selection and composition of Web services and auto-generate of Web services workflow. Traditionally, user should identify the desire Web service inputs and outputs programmatically using some tools to do so. Though, there is some visual tools such proposed by [8] that enable end user to use graphical user interface such drag and drop to identify Web services workflow manually, it's still lack for automation method which can remove the burden of choosing the best Web services composition plane which fit the user needs. Therefore, we believe that naïve user with little experience unable to realize any application workflow without any automation assistance. End user may able to just identify Web services inputs and outputs with some help from simple graphical tools. Then, they can delegate the rest of Web services composing process to automatic composition.

In spite of some approaches such as intelligent based automatic Web services discovery and composition have been proposed, the problem still consider opens issue [9]. Several optimization algorithms such as Genetic Algorithm (GA), Ant Colony Optimization (ACO) have been applied to solve semantic Web services composition problem [10]-[12]. Unfortunately, these algorithms still face some challenges in term of the efficiency and accuracy. The challenges have identified due to the heterogeneous nature of Web services and also due to the huge number of Web services available today. All of these challenges raise the need of developing a new automatic composition approach based on efficient Web services discovery strategy which can improve the composition process [13]. However, according to literature, semantic Web services composition problem have been solved by using different type of Swarm Intelligent algorithms with different strategies and different semantic technologies which make the evaluation of the best approach become hard task; especially in measuring the success rate, composition length and time elapsed of the composition. Therefore, there is a need to evaluate several types of optimization algorithms in unique semantic Web services environment (such as: text collection and ontology structure) in order to analyze and measure the efficiency of deferent methods regarding proposed model.

However, automating the composition process is valuable especially to the end users. It can hide and reduce the

Manuscript received January 10, 2013; revised May 12, 2013.

The authors are with the National University of Malaysia (e-mail: zao@ftsm.ukm.my).

complexity of composing Web services. As a result, the paper focus on adds flexibility to end user (non-programmer) and adds more efficiency (the efficiency of automatic compotation of Web services) in the Web services composition. Thus, according to the problem, the main objective is to propose a new model for automatic semantic web services composition based to end user perspective. The model basically is relies on enabling end user to be involved in Web services composition process in flexible manner.

II. Related Work

A semantic Web technology has played a major role in Web services discovery and composition. It introduced a framework for semantic description of Web services and related aspects [14] has proposed an automatic Web services composition based on Web service ontology and semantic process ontology and they use Human Assistant (HA) approach as a complement stage of composition process. Another composition approach has proposed by [15], they have used hash table as the data structure to represent the service-Web services input and output relationship in order to enhance the performance. SWSDS is a composition system based on both syntactic and semantic matching for service parameter by switching a so-called search index [16].

[12] defined Web services ontology and their relationships by OWL-S (Web service ontology) and use AC algorithm to compose semantic Web services. Supported by OWL-S, this model calculates the similarity among the Web services ontologies using the degree of correlation between the input and output of Web services, then it construct a graph of Web services. Composition of Web services is transformed into using AC algorithm to find an optimal path in a graph. The authors have proposed four general attributes to build the Web services ontology, namely: Service definition and description, the relationship between services, service levels, services' inputs and outputs. [10] has used an Ant Colony System (ACS) to investigate the search space of semantic web service composition problem. The composition system relies on a set of semantic concepts that describe Web services elements. The authors used Ant algorithm to find the shortest composition path between the requested input and output. The selection criteria were based on the number of concepts related to candidate services.

III. Web Services Composition Model

This work presents a user friendly and efficient automatic Web services composition model. The proposed model relies on four main components namely, Visual Services modeling, User Query Generator, Semantic Composer and Workflow Generator. As shown in Fig. 1, the main propose of proposed automatic composition model is to automate the composition process with considering end user perspective. Furthermore, to improve Web Services composition usability and efficiency. Each component in the proposed model has specific role as the following:

A. Visual Web Services Module

Through this module, end user can build his own composition requirements visually by specifying the desire Web services inputs (input parameters for first Web service) and outputs (the output for the last Web service). The user can identify inputs and outputs parameters types and names through using simple visual components such as toolbars which enable end user to drag and drop the desired entities (textbox, checkbox, widgets, …, etc.). The basic idea visual Web services module is to provide a manageable environment for end user based on simple graphical user interface. Then, user request (graphical plan) will transfer to user query generator component for farther processing.

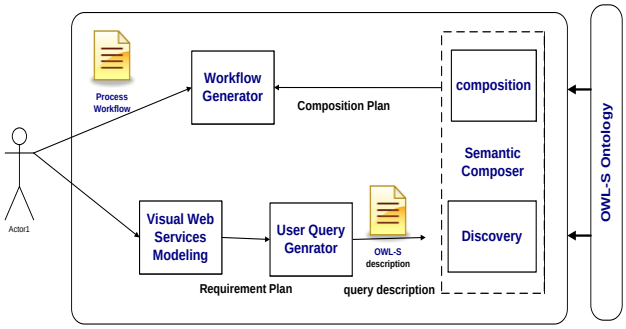

Fig. 1. Automatic web services composition model.

B. User Query Generator

This component responsible for mapping user request (graphical plan) into formal service description which is includes the Inputs, outputs and general service functionality. The generated description language is based on machine readable language such OWL-S. Next, the Query Generator passes an instance of generated description language to Semantic Composer components to manipulate it and start the composition process.

C. Semantic Composer

The proposed component considers as most challenging task in the automatic composition model. The main purpose of the composer is to automatically compose atomic Web services to achieve user goals (query plan). The composition process relies on ontology marching between Web services Input/ Output concepts for each services pair. Then, based on the defined concepts, the composer builds a composition plan.

However, in order to achieve best performance, two phases of composition process have been involved in composition process. First, Web services discovery which uses backward discovery strategy based on set overlapping selection [15]. Second phase is Web services composition which based on Ant Colony System [10]. According to the previous studies, Ant colony algorithm has shown a promising efficiency in composition problem. Therefore, Ant Colony algorithm is proposed to find the shortest path based on the number of matched concepts. The composition criteria between two pair of services rely on matching the output elements of first Web services with the input elements of second services. This process will sequentially repeated until the desired output obtained based on user request. However, the chosen path will forwarded to Workflow Generator to execute the composed service.

D. Workflow Generator

This component responsible for generating workflow process instance for Web service composition based on OWL-S. The insistence ontology will contains process workflow description of composed services. Fig. 1 below shows the automated web services composition which consists of workflow generator to generate plans for making composition and web services modeling visualization to generates requirement plan.

IV. Discussion

This research aims to construct automatic Web services composer that enhance and add more flexibility to Web services composition process. In the future work, The study will focus on how to achieve the research goals and objectives thought implementing the whole research processes As showmen in Fig. 2, the design of proposed research methodology consists of the following three main phases:

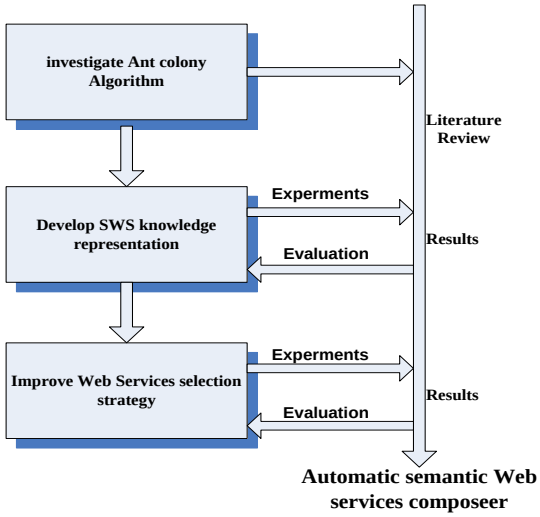

Fig. 2. Development steps.

A. Investigate Ant Colony Algorithm Phase

The purpose of this phase is to achieve the first objective of study by compare different Ant colony Optimization (ACO) and investigate the efficiency of each algorithm regarding Web services composition problem. During this phase, based on the literature, different Ant colony algorithm will be studied and applied to semantic Web services paradigm. This phase will come out with the most suitable algorithm which can fit the semantic Web services composition problem.

B. Develop Semantic Web Services Knowledge Representation Phase

In this phase, suitable Web service representation will be designed based on Web service ontology's language (description logics and rule languages such as OWL-S) in order to produce an efficient representation for the composition problem. The investigation process relies on applying an experiment s and evaluation process.

C. Improve Web Services Selection Strategy Phase

Web services selection strategy will be achieved through this phase by comparing and adopting two different web services selection strategies such as forward and backward Web serves discovery strategy based on set overlapping or union selection which have proposed by [15] in order to obtain the most efficient strategy that can fit the proposed Web services composition model and also to reduce the number of candidates set for composition processes. Figure 2 shows the step on how to improve the web services selection strategy which consists of three phases: investigation the ant colony algorithm, developed SWS knowledge representation and improved the web services selection strategic.

V. Conclusion

The paper introduces a usable and efficient automatic web services composition model based on end user perspective. However, In order to achieve the best composition performance, two issues have been considered in composition process. First, visualization, using visual module to enhance end user interaction with Web services composition. Second, automation, automate Web services composition by applying backward discovery strategy for web services selection and Ant Colony System for Web services composition process. In Future work, the proposed model will be evaluated by developing a prototype in order to test the usability and efficacy of the proposed model.

References

[1] C. M. MacKenzie, K. Laskey, F. McCabe, P. F. Brown, and R. Metz, "Reference model for service oriented architecture 1.0," *OASIS Standard*, vol. 12, 2006.
[2] T. Nestler, *Towards a mashup-driven end-user programming of SOA-based applications*, 2008.
[3] J. Soriano, D. Lizcano, J. J. Hierro, M. Reyes, C. Schroth, and T. Janner, *Enhancing user-service interaction through a global user-centric approach to SOA*, 2008.
[4] A. E. Walsh, "Uddi, soap, and WSDL: the web services specification," *Prentice Hall Professional Technical Reference*, 2002.
[5] L. Clement, A. Hately, C. V. Riegen, and T. Rogers, *Universal description discovery and integration (UDDI) 3.0. 2. 2004*, 2004.
[6] T. Nestler, M. Feldmann, A. Preußner, and A. Schill, *Service Composition at the Presentation Layer using Web Service Annotations*, 2009.
[7] A. Namoun, T. Nestler, and A. De Angeli, *End user requirements for the composable web*, 2010.
[8] J. Janeiro, A. Preußner, T. Springer, A. Schill, and M. Wauer, *Improving the development of service-based applications through service annotations*, 2009.
[9] A. Bansal, M. B. Blake, S. Kona, S. Bleul, T. Weise, and M. C. Jaeger, *WSC-08: Continuing the web services challenge*, 2009.
[10] T. Ghafarian and M. Kahani, *Semantic web service composition based on ant colony optimization method*, 2009.
[11] T. Weise, S. Bleul, D. Comes, and K. Geihs, *Different approaches to semantic web service composition*, 2008.
[12] K. Yan, G. Xue, and S. Yao, *An optimization ant colony algorithm for composition of semantic Web services*, 2009.
[13] S. Bleul, T. Weise, and K. Geihs, "The web service challenge-a review on semantic web service composition," *Service-Oriented Computing (SOC'2009)*, 2009.
[14] I. B. Arpinar, R. Zhang, B. Aleman-Meza, and A. Maduko, "Ontology-driven web services composition platform," *Information Systems and E-Business Management*, vol. 3, pp. 175-199, 2005.
[15] Y. Zhang, K. Raman, M. Panahi, and K. J. Lin, *Heuristic-based service composition for business processes with branching and merging*, 2007.

Ayman I. Srour received his B.Sc. in computer science form Al-azhar University, Gaza - Palestine in 2002. He received his M.Sc. degree in information technology form Utara University, Kedah, Malaysia, in 2009. He is now a PhD candidate at National University of Malaysia. He received a Malaysian Association of Research Scientists - Bronze Medal at Malaysia Technology Expo 2009. His interests include heuristics/ metaheuristic algorithms and optimization techniques.

Zulaiha Ali Othman was born in Malaysia. She graduated her phd in computing from Sheffield Hallam University, England, in 2004. Since that till now she is a lecture and researcher at Data Mining and Optimization Research Group, Faculty of Information Science and Technology, National University of Malaysia. Her research interested is applying and improving data mining techniques in various domain including network traffic, human resource and medical, weather prediction especially focuses on spatio-temporal data mining, etc. The second research interested is improving the metaheuristic optimization is several area such as TSP, VRP and Web Services Selection problem. The more specific research expert is improving the both area using agent. Associate Dr Zulaiha is membership of several computer science committee and has published more than 100 conferences and Journal publications around the world with several research awards.

Abdul Razak Hamdan is Malaysian and awarded his phd from Laborough University, England. Currently he is a Head of Data Mining and Optimisation Research Group, Faculty of Information Science and Technology, National University of Malaysia. He is well known as father of Artificial Intelligence in Malaysia as he has inventing and involving in many technical and applying AI research in various domains with massive good quality publications and awards.