

AN OPTIMAL COMPOSITION PLAN SELECTION USING MULTI OBJECTIVE PARTICLE SWARM OPTIMIZATION

T. Parimalam

Research Scholar, Department of Computer Science,
Erode Arts and Science College, Erode, Tamilnadu, India

Dr. K. Meenakshi Sundaram

Associate Professor, Department of Computer Science,
Erode Arts and Science College, Erode, Tamilnadu, India

ABSTRACT

Domain-ontology based Particle Swarm Optimization (PSO)-inspired Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH) and Improved Bipartite graph is an efficient web service composition approach. It composed a number of web services to achieve high customer satisfaction. It also improves the clustering quality and reduced the processing time of web service composition. The most verification problems are easily verified by using Petri Net in planning, verification and execution phase. In order to provide more efficient automatic web service composition, in this paper multiple Quality of Service (QoS) parameters like cost, accuracy, accessibility, robustness, scalability, modifiability and security are included in Petri net-based algebra approach. While adding a greater number of QoS parameters there will be multi-objective services composition optimization problem. It is solved by introducing a Multi-Objective Genetic Algorithm (MOGA) and Multi-Objective PSO (MOPSO) algorithm. It selects qualitative different services from multiple functionally identical web services which achieve the best reliability model. Thus, the automatic web service composition is performed for discovering the appropriate services by satisfying the customer requirements. Experimental results show that the proposed method performs better than the existing method.

Keywords: Balanced Iterative Reducing and Clustering using Hierarchies clustering, Improved Bipartite graph, Petri net-based algebra, web service, web service composition.

Cite this Article: Parimalam. T And Dr. Meenakshi Sundaram.K, An Optimal Composition Plan Selection using Multi Objective Particle Swarm Optimization, *International Journal of Computer Engineering and Technology*, 10(1)2019 pp.226-235 <http://iaeme.com/Home/issue/IJCET?Volume=10&Issue=1>

1. INTRODUCTION

Web services [1] are considered as self-contained, self-describing, modular applications which can be invoked, located and published across the web. Because of increasing in the number of organizations and companies only implement their core business and outsource other applications over the internet. So, there is a need to effectively and efficiently select and integrate heterogeneous web services. When a single web service cannot satisfy the functionality of the user requirement, a number of web services are composed together to fulfill the user request. This motivates a number of researchers to concentrate on the web service composition.

Automatic web service composition [2] is concerned as the modified description of the general web service composition. The process of developing a composition schema in order to satisfy a series of goals set by a requester is a really complex and multifaceted problem since one has to deal with many different issues at once. First of all, it involves searching in an ever-growing global service repository in order to find matching services that may contribute to the complete satisfaction of the user's requirements. The problems of determining related and composable web services are called as web service composition. A proficient automatic web service composition framework was proposed for automatic web service composition. It consisted of pre-processing phase, service discovery and ranking phase and planning, verification and execution phases. In the pre-processing phase, a domain ontology-based PSO inspired BRICH [3] clustering was used to cluster the web services in a hierarchical manner.

An improved bipartite graph [3] was developed in the discovery phase for efficient matching of the web services in the clusters. Then web services are ranked based on QoS parameters. However, in this framework, the planned services are not properly verified for their correctness which addresses three main concerns are information assurances, net worthiness and interoperability. Thus, the accuracy of web service composition is reduced. In the planning, verification and execution phase, the composition plans are created by getting input from the discovery phase and then the plans are verified for their correctness. Petri net algebra [4] is used to verify the generated plans that models the control flows as a necessary constituent of the reliable web service composition.

The optimal composition plans are generated by considering multiple QoS parameters are cost, accuracy, accessibility (access rate), robustness, scalability, modifiability, and security. However, in the composition plan selection, it is more complex to choose qualitatively different services from multiple functionally identical web services in order to achieve the best reliability. So, in this paper, Multi-Objective Genetic Algorithm (MOGA) and Multi-Objective Particle Swarm Optimization (MOPSO) are used to select the web service compositions based on the optimal QoS parameters. Thus, the automatic web service composition is performed for discovering the appropriate services by satisfying the customer requirements.

2. LITERATURE SURVEY

A general framework was proposed [5] to select the best composite of web services based on web criteria evaluation. It was an extension of web service architecture and it was extended by adding a new Multi-criteria Evaluation Component (MEC) in the registry. The added MEC devoted to multi-criteria condition. A set of evaluation criteria developed a set of recommended composite web services and a set of composite web services are the inputs of the new component. Moreover, solutions were proposed to construct and evaluate the composite web services. The feasibility of the proposed framework was evaluated by developing a prototype based on the open source jUDDI registry.

An Optimal Composition Plan Selection using Multi Objective Particle Swarm Optimization,

An approach was proposed [6] based on the structural theory of Petri nets to verify web services compatibility. It verified process interactions to coordinated web service compositions. The results of the structure theory Petri net utilized to determine the necessary or sufficient structural conditions on web services that guarantee the compatibility of composite web services. The main intent of this work was to provide a structural technique that checks if two or more web services were compatible. Moreover, it provided a better understanding of incompatible sources. However, the substitutability problem is not discussed in this approach.

A simple framework for the selection of reliable web service composition was presented [7]. For the selection of web services with all QoS requirement, the evaluations of reliability on web services were required. The presented framework was considered the design of reliability evaluation for the composition of web services. The availability and accessibility parameters were used for the evaluation of reliability. A simple vector computation was utilized to find out the reliable services.

The problem of automatic dynamic semantic web service composition was to deal and proposed solutions [8] for such problems. The proposed solutions are oriented to express the pre or post conditions of web services. There are different solutions are proposed for the related issues like change in a service environment, Effectiveness, the functional aspect of web services, QoS optimization, composition system and continuing user query arrival. The main intent of the proposed solutions was clearly describing the pre or post conditions and find that these conditions were feasible to compose services during the consideration of pre or postconditions. However, it focused only to a single part of the whole problem.

An approach was presented called Colored Petri net- based method (CPN) [9] for formal modeling and verification of web service composition. CPN was an extended version of Petri net which is a graphical formal method. Moreover, it has some precise automated verification tools and mathematic semantics that are more suitable for concurrency, synchronization systems, and modeling communication. The proposed method detects the bugs at the design stage and ensures the correctness of service composition by dynamic behavioral and structural analysis. It increased the confidence in the correctness of the model.

Particle Swarm optimization was presented [10] to optimize the process of web service composition. In this proposed work processed based on two heuristics, two meta-heuristic methods based on Particle Swarm Optimization (PSO) are introduced to address the optimization problem and several workflow requests are being processed simultaneously. In this make a case for using a meta-heuristic method to this assignment problem, considering also the scalability of the approach. However, this approach has no general convergence theory applicable to practical, multidimensional problems.

A multi-attribute optimization approach was proposed [11] to select the service providers with the best user desired quality. A novel concept named as service skyline and a various set of service skyline computation techniques were developed which returned a set of most interesting service providers. Such providers were non-dominant in all user interested quality attributes. By using skyline providers, the user desired providers were included to select the best service providers.

3. METHODOLOGIES

In this section, the optimal selection of web service composition based on multiple QoS parameters using a Multi-Objective Genetic Algorithm (MOGA) and Multi-Objective Particle Swarm Optimization (MOPSO) is explained in detail. Initially, the web services are clustered and discovered based on domain ontology based PSO inspired BIRCH and improved bipartite graph [3]. The ranked services should be verified for reachability, the existence of deadlocks

and liveness. In the context of web services, verification establishes the upholding of certain properties of a web service. For more efficient automatic web service composition adding few more QoS parameters is considered which optimally select the composition plan. The multiple QoS parameters for the selection of service composition might cause a multi-objective optimization problem. Hence MOGA and MOPSO algorithms are introduced to resolve the multi-objective optimization problem.

3.1. Multi-Objective Genetic Algorithm based Optimal Selection of Web Service Composition

The selection of optimal web service composition plan based on multiple QoS parameters might cause a multi-objective optimization problem which is resolved by using Multi-Objective GA algorithm. The MOGA begins with a selection of composite web services represented by a population of chromosomes. The composite web services from one population are taken and utilized to form a new population. This is processed in such a consideration that the new population will be better than the old population. The composite web services which are then selected to form new composite web services (offspring) according to their fitness value (multiple QoS parameters). This process is repeated until the maximum number of generations is achieved. The fitness value is calculated as,

$$\begin{aligned} \text{Fitness}(x) = & \text{Reliability}(x) + \text{Availability}(x) + \text{Accuracy}(x) + \text{Accessibility}(x) \\ & + \text{robustness}(x) + \text{scalability}(x) + \text{modifiability}(x) \\ & + \text{security}(x) \end{aligned} \quad (1)$$

The MOGA is composed of four operators are selection, crossover, mutation, and replacement. In the selection process, the parent chromosomes are selected from the population. In the crossover process, the new population is created from two existing ones by combining or rearranging the web services of existing chromosomes. The mutation process occurs occasionally for allowing the specific child to obtain the features that are not possessed by either parent. The replacement process is used to decide which data individuals stay or get replaced in a population. Based on these four operators and fitness function, MOGA optimally selects the web service composition plan.

MOGA Algorithm

Input: population of n chromosomes, composite web services

Output: optimal selection of composite web services based on multiple QoS parameters

1. Each chromosome in the population randomly selects the composite web services
2. Evaluate the fitness of each chromosome in the population using equation (1)
3. Based on the fitness value of chromosomes select some chromosomes as a parent from the population
4. Perform crossover and mutation process to create a new population
5. Replace the old population of chromosomes with the new population
6. If the maximum number of generations is reached, then stop, and return the best solution,
7. Else, Go to step 2.

3.2. Multi-Objective Particle Swarm Optimization based Optimal Selection of Web Service Composition

A Multi-Objective PSO (MOPSO) is introduced for optimal selection of composite web services based on QoS parameters. The QoS constraint-based dynamic web service selection algorithm is used. The QoS parameters such as availability, reliability, time, cost, accuracy, accessibility, robustness, scalability, modifiability and security are considered for the optimal selection of services composition. Initially, MOPSO initialized the number of particles and then each particle randomly selects the composite web services. The fitness of each particle in population is calculated based on multiple QoS parameters are based on reliability, availability, accuracy, accessibility, robustness, scalability, modifiability and security. Based on the fitness value each particle updates their position (web service composition) and velocity.

MOPSO algorithm

Input: $gbest=0, pbest = 0, maxiter, c_1, c_2, r_1, r_2, num_p$, composite web services

Output: optimal selection of composite web services based on multiple QoS parameters

1. Each particle randomly selects the composite web services as its position
2. do
3. For $i = 1$ to num_p
4. Evaluate fitness of every particle

$$Fitness_i = \sum_i^{num_p} Reliability_i + Availability_i + Accuracy_i + Accessibility_i + robustness_i + scalability_i + modifiability_i + security_i \quad (2)$$

5. if the current fitness value is better than $pbest$
6. Set current value as the new $pbest$
7. End if
8. End for
9. Choose the particle which has best fitness value as $gbest$
10. For $i = 1$ to num_p
11. Calculate the velocity of particle using

$$v_i^{t+1} = v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (gbest_i^t - x_i^t) \quad (3)$$

12. Update the position (web service composition) of particle using

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (4)$$

13. End for
14. while ($maxiter$)
15. End while

In above algorithm, $gbest$ denotes the global best, $pbest$ denotes the particle best, $maxiter$ denotes the maximum iteration, c_1 and c_2 are the learning factor, r_1 and r_2 are the random number between 0 to 1, num_p is the number of particles in population, v_i^t is the i -th particle velocity at t -th iteration and the position of i -th particle velocity at t -th iteration is denoted as x_i^t . Thus, the optimal web composition plan is selected using the above MOPSO algorithm.

4. RESULT AND DISCUSSION

In this section, the performance of automatic web service composition is evaluated using 10835 indexed OWL-S services in OWL-S Service Retrieval Test Collection version 4.0. It has nine different domains are education, medical care, food, travel, communication, economy, weapon, geography and simulation. The education medical care, food, travel, communication, economy, weapon, geography and simulation domains consist of 286, 73, 34, 197, 59, 395, 40, 60 and 16 services respectively. The efficiency of MOGA and MOPSO based web service composition selection are tested in terms of Time, availability, reliability and cost.

view of the revolutionary changes in software, the process should also scale up to suit the type and size of the project. Different quality and industrial standards notably CMMI, PMBOK Guide, ITIL and PRINCE2 recommend different guidelines and standards to enable achieving the desired outcome from projects. Software organizations use process improvement to achieve their goals. One of the objectives is to improve the quality of the product. This can be achieved by reducing errors, improving the good working environment, adopting best practices and following industrial quality standards.

4.1. Time

Time measures the execution time between the requests sent and results received. It is the amount of time taken for a service to provide a response to various types of requests from composite users.

Table 1 Comparison of Time

No. of web services	Time (secs)	
	MOGA based Composition Selection	MOPSO based Composition Selection
50	6	4
100	11	8
150	13	10
200	15	12
250	17	14

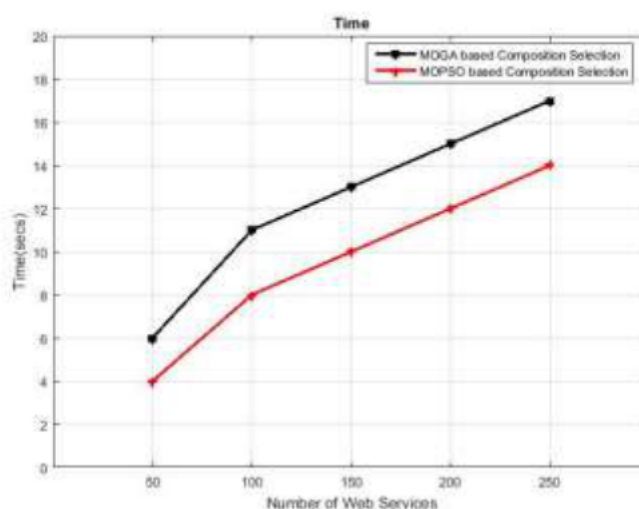


Figure 1 Comparison of Time

An Optimal Composition Plan Selection using Multi Objective Particle Swarm Optimization,

Figure 4.1 shows the comparison of time between MOGA and MOPSO based web service composition. The number of web services is taken in X-axis and the time in seconds is taken in Y-axis. At 250 number of web services, the execution time of MOPSO based web service composition is 17.6% less than MOGA based web service composition. From this analysis, it is proved that the MOPSO based web service composition has better execution time than the MOGA based web service composition.

4.2. Availability

The availability factor Ava_{pq} can be calculated by the services which are responded to the total number of services as,

$$Ava_{pq} = \frac{respond_{pq}}{total\ request_{pq}} \quad (5)$$

Table 2 Comparison of Availability

No. of web services	Availability	
	MOGA based Composition Selection	MOPSO based Composition Selection
50	83	85
100	85	87
150	88	90
200	91	93
250	93	96

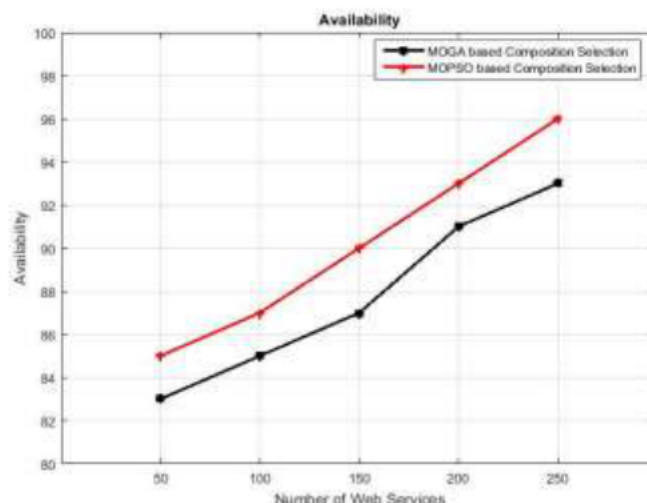


Figure 2 Comparison of Availability

Figure 2 shows the comparison of availability between MOGA and MOPSO based web service composition. The number of web services is taken in X-axis and the availability is taken in Y-axis. If the number of web services is 250, then the availability of MOPSO based web service composition is 3.2% greater than MOGA based web service composition. From this analysis, it is proved that the MOPSO based web service composition has better availability than the MOGA based web service composition.

4.3. Reliability

Reliability is the capability of upholding service and service quality. This measure is used to define the ability of a service which it functions correctly and consistently. It can be calculated as,

$$R_{pq} = e^{-\int_0^{time_{pq}} \gamma(t)dt} * e^{-\gamma * time_{pq}} \tag{6}$$

Table 3 Comparison of Reliability

No. of web services	Reliability	
	MOGA based Composition Selection	MOPSO based Composition Selection
50	0.627	0.64
100	0.634	0.645
150	0.639	0.654
200	0.647	0.662
250	0.652	0.67

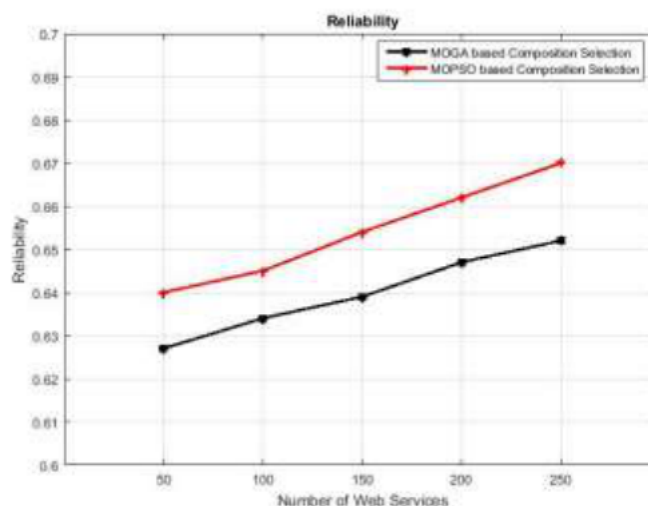


Figure 3 Comparison of Reliability

Figure 3 shows the comparison of reliability between MOGA and MOPSO based web service composition. The number of web services is taken in X-axis and the reliability is taken in Y-axis. When the number of web services is 250, the reliability of MOPSO based web service composition is 2.8% greater than MOGA based web service composition. From this analysis, it is proved that the MOPSO based web service composition has better reliability than the MOGA based web service composition.

4.4. Cost

Cost is the amount that a service requester needs to pay to execute a service using task.

An Optimal Composition Plan Selection using Multi Objective Particle Swarm Optimization,

Table 3 Comparison of Cost

No. of web services	Cost	
	MOGA based Composition Selection	MOPSO based Composition Selection
50	45	40
100	50	43
150	57	49
200	65	56
250	79	65

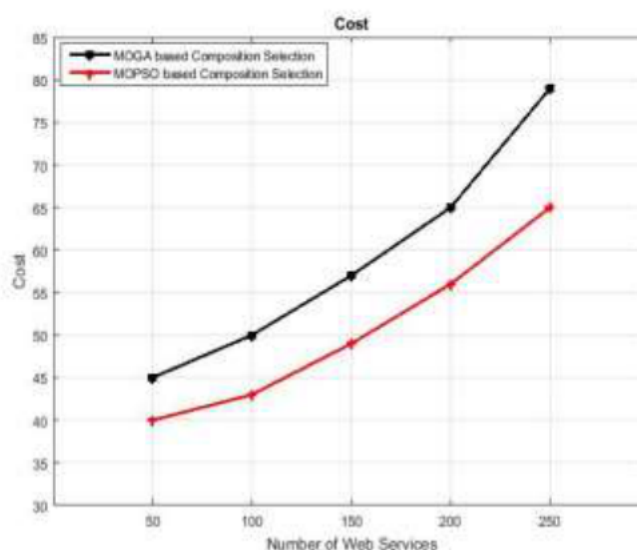


Figure 4 Comparison of Cost

Figure 4 shows the comparison of cost between MOGA and MOPSO based web service composition. The number of web services is taken in X-axis and the cost is taken in Y-axis. When the number of web services is 250, the cost of MOPSO based web service composition is 17.7% less than MOGA based web service composition. From this analysis, it is proved that the proposed MOPSO based web service composition has a better cost than the MOGA based web service composition.

5. CONCLUSION

In this paper, the Petri net with multiple QoS parameters for efficient automatic web service composition is improved by using evolutionary algorithms. In order to select qualitatively different services from multiple functionally identical services and to resolve the multi-objective optimization problem, MOGA and MOPSO algorithm are proposed. The MOGA selects the optimal composition plan based on the crossover and mutation process. In MOPSO algorithm, optimal composition plan is selected by updating the position and velocity of each particle in population. Both the evolutionary algorithms select the more efficient automatic web service composition based on QoS parameters of web services. Experimental results prove that the proposed MOPSO has high accuracy, reliability, availability and execution delay than the MOGA based web service composition method. are key to measure, without measuring, we cannot complete projects successfully and measure the quality of the deliverable.

Metrics are generated by collecting and assimilating related measures over a period of time across similar processes or applications. Software metrics evolved with the changing nature of software.

REFERENCES

- [1] Rodriguez-Mier, P., Pedrinaci, C., Lama, M., and Mucientes, M. An integrated semantic web service discovery and composition framework. *IEEE transactions on services computing*, 9(4), 2016, pp. 537-550.
- [2] Mallayya, D., Ramachandran, B., and Viswanathan, S. An automatic web service composition framework using QoS-based web service ranking algorithm. *The Scientific World Journal*, 2015, 2015.
- [3] Sundaram, K. M., and Parimalam, T. PSO-inspired BIRCH and Improved Bipartite Graph for Automatic Web Service Composition. *International Journal of Applied Engineering Research*, 12(8), 2017, pp. 1765-1771.
- [4] Xia, Y., Luo, X., Li, J., and Zhu, Q. A petri-net-based approach to reliability determination of ontology-based service compositions. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(5), 2013, pp. 1240-1247.
- [5] Haddad, S., Mokdad, L., and Youcef, S. Selection of the Best composite Web Service Based on Quality of Service. *ISSS/BPSC*, 10, 2010, pp. 255-266.
- [6] Barkaoui, K., Eslamichalandar, M., and Kaabachi, M. A structural verification of web services composition compatibility. *CAiSE 2010 Workshop EOMAS'10, Hammamet, Tunisia*, 601, 2010, pp. 30-41.
- [7] Sasikaladevi, N., and Arockiam, L. Reliability evaluation model for composite web services. *International Journal of Web & Semantic Technology*, 1(2), 2010, pp. 16-22.
- [8] Bartalos, P. Effective automatic dynamic semantic web service composition. *Inf. Sci. and Technol. Bulletin ACM Slovakia*, 3(1), 2011, pp. 61-72.
- [9] Tian, B., and Gu, Y. Formal Modeling and Verification for Web Service Composition. *JSW*, 8(11), 2013, pp. 2733-2737.
- [10] Ludwig, S. A. Applying particle swarm optimization to quality-of-service-driven web service composition. *2012 IEEE 26th International Conference on Advanced Information Networking and Applications (AINA)*, 2012, pp. 613-620.
- [11] Yu, Q., and Bouguettaya, A. Multi-attribute optimization in service selection. *World Wide Web*, 15(1), 2012, pp. 1-31.