QoS-based Cloud Manufacturing Service Composition using Ant Colony Optimization Algorithm

Elsoon Neshati
Department of Mechanical Engineering,
Tabriz Branch, Islamic Azad University
Tabriz, Iran

Ali Asghar Pourhaji Kazem
Department of Computer Engineering
Tabriz Branch, Islamic Azad University,
Tabriz, Iran

***Abstract*—Cloud manufacturing (CMfg) is a service-oriented platform that enables engineers to use the manufacturing capacity in the form of cloud-based services that aggregated in service pools on demand. In CMfg, the integration of manufacturing resources across different areas and industries is accomplished using cloud services. In recent years, the interest in cloud manufacturing service composition has grown, due to its importance in different manufacturing applications. When no single service is capable of satisfying the need for a manufacturing service requester, the service combination may be useful in order to fulfill the purpose of the manufacturing service requester. Therefore, the problem of how efficient and effective interconnection of cloud manufactring services has come to fetch many research fields. In this paper, a new algorithm is presented using an ant colony optimization for the problem of cloud manufacturing service composition considering the quality of service.**

Keywords*—Cloud computing; cloud manufacturing; service composition, ant colony optimization

I. Introduction

The 'cloud manufacturing' term was first introduced by Li et al. [1], which targets in creating an integrated and collaborative platform for distributed manufacturing systems based on cloud computing technology. Cloud manufacturing system enables different users to search the qualified manufacturing services from cloud-based resource repositories and dynamically combine them into a virtual manufacturing environment or solution to finish their tasks [1]. Current research efforts are focusing on the development of appropriate descriptions for manufacturing services. Many of theses approaches have chosen for extending web services for the implementation of manufacturing service descriptions [2]. Web services constitute a promising technology perspective for software engineering. The service-oriented system poses several additional challenges in terms of component-based software engineering [3]. Alternatively, several services may be available with the same function (which they call semantic equivalent services). However, they certainly, there are different criteria for service quality. Quality of service features include: cost, response time, accessibility and reliability. In addition, service quality can have other features like precision and frequency. Choosing between different services, including semantic services, provides a function of the quality of service choices. In addition, a user may have certain limitations on the values of some of the features. For example, the cost should not exceed the amount given, which will affect the selection. On the other hand, the service provider can provide a range for the values of the quality of service as part of the contract with the users Potential (hidden, hidden). Also, the quality assurance of the service for this service can be customer-related, so that each of them will be applied to a different instance of that service. For example, a user who buys a service at a price, does not expect a response time to be less than a certain threshold.

The composition of manufacturing cloud services with a knowledge of the quality of service is a key requirement in service-oriented cloud manufacturing system, since it makes it possible to perform complex user activities by meeting the quality of service constraints [3]. Manufacturing services with the same functionality and quality of service are increasing day by day, and providers of these services always have functional requirements along with a set of service quality limits. Therefore, the choice of manufacturing cloud services with the knowledge of the quality of service plays an important role in the composition of manufacturing cloud services. To solve the problem of choosing manufacturing cloud services with the knowledge of the quality of service, some methods with the help of semantic web and some others based on computations service quality traits have been created, but it is clear that the second approach is a more satisfactory solution to meet the global requirements of service quality, because it is a combination of optimization that combines the best composition of services.

Considering the aforementioned aspect of service composition in cloud manufacturing, in this paper, an approach is proposed for QoS-aware manufacturing cloud service composition using ant colony optimization algorithm. The rest of the paper is organized as follow. In the next section, related work will be reviewed. In Section 3, a formal definition of the problem is presented. In Section 4, simulation and experimental results will be discussed and finally Section 5 concludes the paper.

II. Related Works

The problem of choosing a service based on quality of service was first reported by Chang Yu and his colleagues, and was then welcomed by many scholars. In 2007, Hoffman introduced a programmatic formalism to illustrate the composition of Web services, as well as the identification of a particular case of a web service combination called “forward effect” [4]. In 2004, Cardoso, Miller, and Arnold [5], and in 2001, Casati and Sean, as well as in 2004, Greener, discovered dynamic services and service combinations, with the knowledge of service quality, that they were in the service architecture Oriented, the set of constraints is used to describe the functional and non-functional characteristics of the services for search, and the service may be selected according to some desirable criteria of service quality. Michael Jogierer and Grove Mole have used the use of genetic algorithm for optimizing the problem of choosing web services with the knowledge of service quality and implementing this algorithm in their simulation environment in order to compare its efficiency [6]. It has been tested with other methods. In 2009, Zhang and Zhou offered an open cloud computing architecture at an international conference on web services, pointing out that virtualization and service-oriented architecture are two powerful technical key [7].

Cloud manufacturing extends the cloud computing technology with manufacturing infrastructures involved in the entire lifecycle of manufacturing applications. Luo et al. [8] studied the formal description of multidimensional information for manufacturing capability in cloud manufacturing system. Also, Wang et al. [9] discussed standardized data models describing cloud services and relevant features for supporting interoperable cloud manufacturing. Tao et al. [10] presented a modified particle swarm optimization algorithm for manufacturing grid service composition, in which its parameters for particle updating were dynamicaly tuned. In [11], the variant GA and fruit fly optimization was combined to address the QoS-aware cloud computing service composition.

III. Problem Definition

The QoS-aware cloud manufacturing service composition is to find a set of cloud manufacturing candidate services with different functionalities that firstly observe user-defined limits and, secondly, optimize a target function. In this section, the above problem is officially stated. An example of the QoS-aware cloud manufacturing service composition is formally expressed as follows:

- A service composition request in the form of workflow that is modeled using a directed acyclic graph G=(V,E).
- $V=\{T_1,T_2,...,T_n\}$ which n is the number of tasks in workflow.
- E is a set of edges that shows the priority of the tasks.
- Each task T_i $(1 \leq i \leq n)$ has a set of candidate manufacturing cloud services $CS_i=\{cs_i^1, cs_i^2,...,cs_i^{m_i}\}$ in which cs_i^j $(1 \leq i \leq m_i)$ is candidate cloud service.
- m_i is the total number of available candidate manufacturing cloud services for task T_i.
- Each candidate manufacturing cloud service cs_i^j has a set of different quality of service information $QoS_i^j=\{Q_1, Q_2, ..., Q_K\}$ in which Q_i is a quality of service parameter.
- Quality of service related to manufacturing cloud services is stored in the quality of service repository.
- *K:* The number of quality of service parameters for the manufacturing cloud services which are used in the quality of service model.
- *QC:* The set of global restrictions defined by the user $QC=\{C_1, C_2, ..., C_K\}$.

With this in mind, the goal of the QoS-aware manufacturing cloud service composition is to find the near optimal composite manufacturing cloud service where:

$$\forall\, j = 1...K \begin{cases} \sum_{i=1}^{n} S_i.Q_j < C_j & \text{if } Q_j \text{ is additive} \\ \prod_{i=1}^{n} S_i.Q_j > C_j & \text{if } Q_j \text{ is multiplicative} \end{cases}$$

IV. Proposed Approaches

ACO is a heuristic algorithm with efficient local search for combinatorial problems. This paper applies a novel ACO algorithm to QoS-aware manufacturing cloud service composition problem. Different parts of the proposed ACO are presented in the rest of this section.

A. Initialization

The initialization step of the algorithm consists of two processes:

- Creation of initial population
- Initialization of pheromone matrix.

In order to create the initial population, a random solution is created for each Ant_i, $i=1,2,…,k$ where k is the number of ants in the population. A solution in the population is an integer array with size n that the item of index i indicates the candidate manufacturing cloud service executing the task T_i in the workflow. The next step in the initialization phase is the initialization of pheromone matrix. The pheromone matrix is an $m \times n$ matrix that all of its items are set to an initial value τ_0. m is the maximum number of candidate manufacturing cloud services for a task.

$$\tau_{ij} = \tau_0 \qquad 1 \leq i \leq m \text{ and } 1 \leq j \leq n$$

B. Fitness Function

The main objectives of QoS-aware manufacturing cloud service composition problem are satisfying the user’s global constraints while optimizing a fitness function. Therefore, the problem can now be modeled by means of a fitness function and, finally, some constraints. The fitness function should maximize some QoS parameters of manufacturing cloud services such as reliability and availability, while minimizing others such as cost and response time.

Considering the aforementioned aspects of the fitness function, it can be defined as follows:

$$Fitness(sol) = \frac{w_1 * sol.Resp + w_2 * sol.Cost}{w_3 * sol.Avail + w_4 * sol.Reli}$$

Where w_1, w_2, w_3 and w_4 are positive weights which indicate the importance of QoS parameters identified by the user.

C. Pheromone Updating

After initialization step that a solution for each ant is created, the pheromone trails are updated. In fact in the proposed ACO algorithm only global pheromone updating is applied. Pheromone updating is first done by decreasing the pheromone value on all paths by a constant factor. This step of pheromone updating is referred to as pheromone evaporation. After evaporation, all ants increase pheromone values in the pheromone matrix according to their solution's feasibility. In the QoS-aware grid service composition problem, a solution is feasible if it satisfies all of the QoS constraints identified by the user.

Considering the aforementioned aspects of the pheromone updating, it can be implemented by:

$$\tau_{ij} = \left[(1-\rho)\tau_{ij} + \Delta\tau_{ij}^{best}\right]_{\tau_{min}}^{\tau_{max}}$$

$$1 \le i \le m, 1 \le j \le n$$

Where $0<\rho\le 1$ is the pheromone evaporation rate; τ_{max} and τ_{min} are respectively the maximum and minimum bound for the pheromone value and the operator $[x]_b^a$ id defined as follows:

$$[x]_b^a = \begin{cases} a & \text{if } x>a \\ b & \text{if } x<b \\ x & \text{otherwise} \end{cases}$$

Also, $\Delta\tau_{ij}^{best}$ is defined as follows:

$$\Delta\tau_{ij}^{best} = \begin{cases} \frac{1}{F_{best}} & \text{if } cs_j^i \text{ is selected for } T_i \text{ in the best ant(Composite Service)} \\ 0 & \text{Otherwise} \end{cases}$$

Where F_{best} is the fitness value of best ant of the current iterartion. The parameter ρ is used to avoid unlimited accumulation of the pheromone trails.

V. Simulation and Experimental Results

In this section, the results obtained from the simulation of the proposed approach will be presented in comparison with the genetic and particle swarm optimization algorithm. The proposed approach is simulated in MATLAB environment. Different experiments are performed and the results of them stated in the rest of this section. To perform experiments, two different test scenario with 20 and 50 tasks are generated randomly. Also different QoS parameters values are generated by random. One of the main features of heuristic algorithms mainly ACO is the convergence of it. Fig. 1 and 2 show the results of the convergence test of proposed ACO compared to genetic and particle swarm optimization algorithms. Convergence results indicate that the proposed ACO converges to the optimal or near optimal solution as quickly as possible and also proposed ACO generates better composite manufacturing cloud services than genetic and particle swarm optimization algorithms.

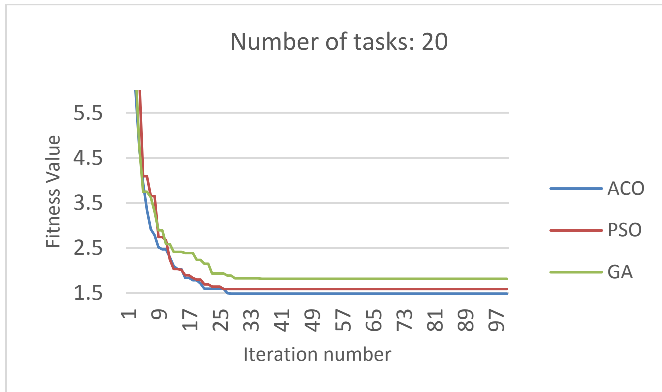

Fig. 1. Convergence test (Number of tasks: 20).

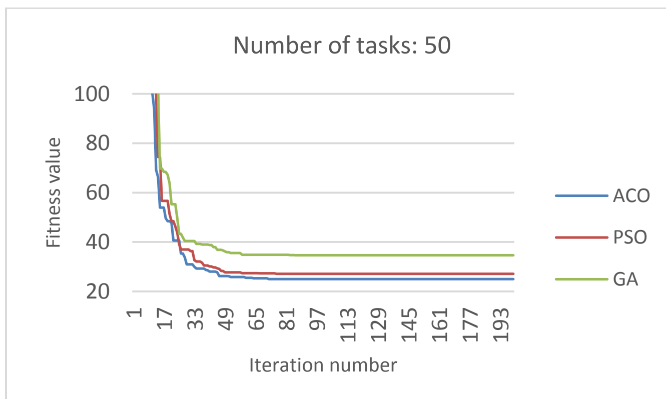

Fig. 2. Convergence test (Number of tasks: 50).

Meta-heuristic algorithms like ACO have an indeterminate and random nature, so it is necessary to examine the stability of these algorithms. The stability of an algorithm is whether the algorithm generates the same results for various executions. To verify the stability of the proposed ACO algorithm for the two scenarios mentioned above, the algorithm is executed 10 times and the fitness value per run is shown in Fig. 3 and 4. Examining the results of the stability test shows good stability of the proposed approach and indicates that the proposed approach converges to optimal solution in every execution of the algorithm.

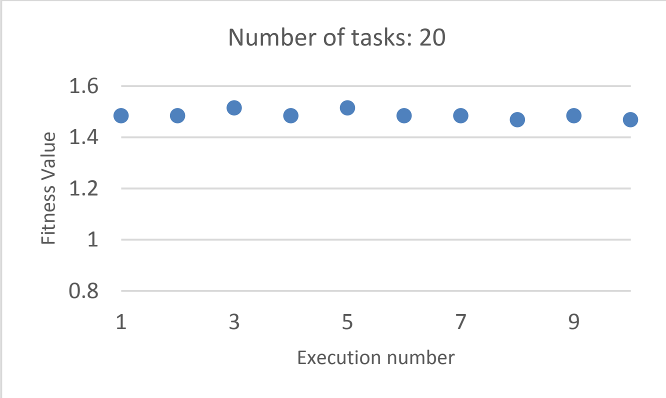

Fig. 3. Stability test (Number of tasks: 20).

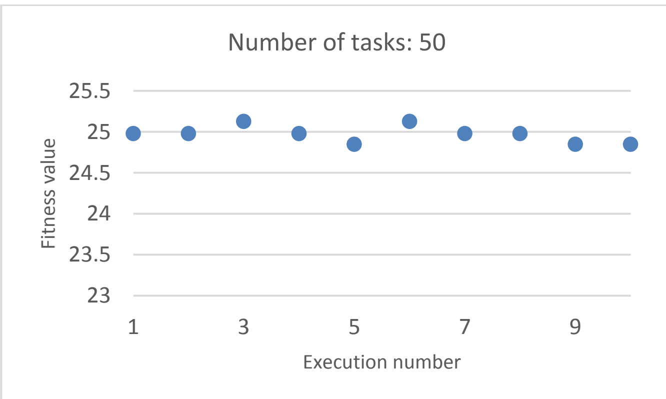

Fig. 4. Stability test (Number of tasks: 50).

VI. Conclusion

The composition of manufacturing cloud services is an important technology for creating value added manufacturing services. The QoS-aware manufacturing cloud service composition with general QoS constraints is a very important problem in manufacturing cloud and service calculations. Designing an excellent algorithm for solving this problem with the capability to find the near optimal solution was the goal of this paper, which has been realized using the ant colony optimization algorithm. Taking into account the uncertainty and dynamic aspect of manufacturing cloud environments, proposed ACO is an efficient approach for the aforementioned problem. Different experiments are performed to evaluate the proposed approach compared to genetic and particle swarm optimization algorithms and the results of them indicate that the it has good convergence speed and stability.

References

[1] B.-H. Li, L. Zhang, S.-L. Wang, F. Tao, J. W. Cao, X. D. Jiang, X. Song, and X. D. Chai, "Cloud manufacturing: a new service-oriented networked manufacturing model," Computer integrated manufacturing systems, vol. 16, no. 1, pp. 1–7, 2010.

[2] F. Tao, L. Zhang, V. C. Venkatesh, Y. Luo, and Y. Cheng, "Cloud manufacturing: a computing and service-oriented manufacturing model," Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, vol. 225, no. 10, pp. 1969–1976, 2011.

[3] V. Diamadopoulou, C. Makris, Y. Panagis, and E. Sakkopoulos, "Techniques to support Web Service selection and consumption with QoS characteristics," Journal of Network and Computer Applications, vol. 31, no. 2, pp. 108–130, 2008.

[4] J. Hoffmann, P. Bertoli, and M. Pistore, "Web Service Composition As Planning, Revisited: In Between Background Theories and Initial State Uncertainty," in Proceedings of the 22Nd National Conference on Artificial Intelligence - Volume 2, 2007, pp. 1013–1018.

[5] J. Cardoso, A. Sheth, J. Miller, J. Arnold, and K. Kochut, "Quality of service for workflows and web service processes," Web Semantics: Science, Services and Agents on the World Wide Web, vol. 1, no. 3, pp. 281–308, 2004.

[6] M. C. Jaeger and G. Muhl, "QoS-based selection of services: The implementation of a genetic algorithm," in Communication in Distributed Systems (KiVS), 2007 ITG-GI Conference, 2007.

[7] L.-J. Zhang and Q. Zhou, "CCOA: Cloud computing open architecture," in Web Services, 2009. ICWS 2009. IEEE International Conference on, 2009, pp. 607–616.

[8] Y. Luo, L. Zhang, F. Tao, L. Ren, Y. Liu, and Z. Zhang, "A modeling and description method of multidimensional information for manufacturing capability in cloud manufacturing system," The International Journal of Advanced Manufacturing Technology, vol. 69, no. 5–8, pp. 961–975, 2013.

[9] X. V. Wang and X. W. Xu, "An interoperable solution for Cloud manufacturing," Robotics and Computer-Integrated Manufacturing, vol. 29, no. 4, pp. 232–247, 2013.

[10] F. Tao, D. Zhao, H. Yefa, and Z. Zhou, "Correlation-aware resource service composition and optimal-selection in manufacturing grid," European Journal of Operational Research, vol. 201, no. 1, pp. 129–143, 2010.

[11] F. Seghir and A. Khababa, "A hybrid approach using genetic and fruit fly optimization algorithms for QoS-aware cloud service composition," Journal of Intelligent Manufacturing, pp. 1–20, 2016.