Research Article

Neeti Kashyap*, A. Charan Kumari, and Rita Chhikara

Service Composition in IoT using Genetic algorithm and Particle swarm optimization

https://doi.org/10.1515/comp-2020-0011 Received Jun 30, 2019; accepted Dec 29, 2019

Abstract: Web service compositions are commendable in structuring innovative applications for different Internetbased business solutions. The existing services can be reused by the other applications via the web. Due to the availability of services that can serve similar functionality, suitable Service Composition (SC) is required. There is a set of candidates for each service in SC from which a suitable candidate service is picked based on certain criteria. Quality of service (QoS) is one of the criteria to select the appropriate service. A standout amongst the most important functionality presented by services in the Internet of Things (IoT) based system is the dynamic composability. In this paper, two of the metaheuristic algorithms namely Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are utilized to tackle QoS based service composition issues. QoS has turned into a critical issue in the management of web services because of the immense number of services that furnish similar functionality yet with various characteristics. Quality of service in service composition comprises of different non-functional factors, for example, service cost, execution time, availability, throughput, and reliability. Choosing appropriate SC for IoT based applications in order to optimize the QoS parameters with the fulfillment of user's necessities has turned into a critical issue that is addressed in this paper. To obtain results via simulation, the PSO algorithm is used to solve the SC problem in IoT. This is further assessed and contrasted with GA. Experimental results demonstrate that GA can enhance the proficiency of solutions for SC problem in IoT. It can also help in identifying the optimal solution and also shows preferable outcomes over PSO.

Keywords: IoT, service composition, particle swarm optimization, genetic algorithm, quality of service

1 Introduction

The term IoT connotes the interconnected devices which allude to an accumulation of different physical gadgets around the globe that are presently associated with the web to gather and share information. The blend of information is retrieved through different sources. IoT is an accumulation of systems and gadgets that can speak with one another. It is perceived as a standout amongst the most imperative territories of future innovation and has got wide consideration from various applications. The peculiarity of the IoT isn't in any new risky development [1, 2]. It is seen as the most important area among the most basic domains of future development and is expanding hugely from a broad assortment of endeavors [3].

Many devices like cameras, television, a refrigerator could be connected to the world via the Internet. The devices become part of the IoT by providing some information which can further be utilized by some application. These devices can be uniquely identified with the help of RFID tags. For example, an IoT based application that is part of the Smart Home can keep track of the devices which are switched on and their power consumption at a particular time. Few IoT devices offer same service with completely different quality parameters like high reliability, low response time and low cost [4]. These days cloud foundations give a section point to revelation, determination, combination and devouring for such appropriate IoT services. Thus, another sort of middleware administration ought to be conceived by a cloud to choose and create the desired services dependent on the end client requirements. In the current literature of optimization in engineering applications, the nature-inspired algorithms have shown promising performance and thus these are popularly and widely used to solve various problems. The contributions of the paper are briefly described as follows:

1. GA and PSO are applied to solve the service composition problem in IoT. These meta-heuristic search al-

Rita Chhikara: Dept of CSE, The NorthCap University, Gurgaon, Haryana, India



^{*}Corresponding Author: Neeti Kashyap: Dept of CSE, The NorthCap University, Gurgaon, Haryana, India; Email: neeti7777@gmail.com

A. Charan Kumari: Dept of Electrical Engineering, Dayalbagh Educational Institute, Agra, Uttar Pradesh, India

gorithms are used to optimize the service composition multi-objective problem into a single objective problem by applying equal weights to all three QoS parameters.

- 2. The implemented algorithms are compared using the fitness value and time. Fitness value plays a key role in comparing their performance.
- The effectiveness of implemented algorithms has been illustrated using a service composition application scenario in IoT environments with the realworld application data set.

The purpose of this paper is to compare the two most popular metaheuristic search algorithms GA and PSO for single-objective optimization to solve multi objective problem of service composition in IoT using QoS parameters such as execution time, service cost and reliability. The rest of the paper is organized as follows: Section 2 presents the concept of Internet of Things along with its architecture and various applications in the literature. Section 3 then outlines the related work. Then in section 4, service composition optimization problem has been described. Section 5 includes the description of implemented algorithms:GA and PSO, Section 6 includes evaluation methodology which comprises of data sets, experimental settings, and solution encoding. Finally, Results and analysis are drawn in Section 7 along with the empirical evaluation of the implemented algorithms on the real-world application data set in Section 8.

2 Internet of Things

The popularity of IoT started from the Auto-ID Center at MIT in 1999. During that time Radio-frequency identification (RFID) was the most important component for IoT. The computer has the ability to manage and maintain records of the things which are tagged using RFID. Besides using RFID, the tagging if things can be done using technologies such as barcodes, QR codes and so on. The term, IoT system in the literature refers to the collection of devices and middleware to manage the interaction of devices and networking. Engineering involves issues related to developing technology to build an IoT system. Further, middleware [5, 6] and programming approaches [7] are being proposed to support the engineering of IoT systems. IoT has potential to build up various applications in diverse areas of automotive industry, medical and healthcare industry, telecommunications, agriculture, retail and supply chain management and so on.

IoT has provided promising opportunities in urbanization with the aim of utilizing various resources like power, water, food, transportation, and healthcare. It can be used to predict the failures in the infrastructure. It has emerging applications related to optimizing infrastructure & mitigate damage to a city's foundations. IoT is a collection of various technologies that have the potential to accelerate various tasks performed by different applications such as Emergency healthcare services. It is a collection of objects like sensors, cameras, etc. that communicate over the internet [8, 9]. Thus, the IoT is largely about interaction [10] between things (by sending messages, signals to each other) and has an interface to human beings. Evidently, the interoperability of things is essential.

Three-layer architecture for the IoT has been proposed in [11]. At the bottom, we have the sensing layer consisting of all the objects comprising the IoT. The next higher level consists of the network layer to form a network of sensing layer objects. An application layer that consists of IoT applications lies on the top [12]. To enable co-operation and integration, ontologies for the comprising of the IoT have been developed. Thus, we have sensor ontologies like SSN (Semantic Sensor Networks) and actuator ontologies like SAN (Semantic Actuator Network). In [13], there are three levels of ontologies, device, domain, and estimation levels. Wang et al. [14] propose a single level ontology for IoT services having seven concepts. Whereas the IoT provided basic communication across things, the issue of how to use IoT to provide services was addressed in the next higher layer. One approach was to use the URI (Uniform Resource Identifier) of things [15], another used web services [16], and a third [17] used REST principles to form the web of things, WoT.

3 Related Work

The IoT is distributed, scalable, highly dynamic, changeable with new types of things being added on, and individual things coming and going from the IoT. Service composition is one of the most popular problems in IoT. Li *et al.* [23] have addressed the service composition problem using Multicriteria based on QoS parameters. Furthermore, a hybrid version of multi-population GA is used to make the algorithm simple. The algorithm has simplicity due to the changing of a population into subpopulations. This algorithm considers some of the QoS parameters to find the optimal solution for the service composition problem. Baker *et al.* [24] defined a method to solve the service composition problem with reduced energy consumption. The key

challenge while designing the energy-efficient algorithm is to handle the interoperability of the heterogeneous devices which are integrated together via cloud-based applications. The proposed algorithm is evaluated in multiple cloud environments and has shown efficient performance.

Fortino *et al.* [25] proposed an agent-based high-level design for IoT based systems. They emphasized the usage of clouds to add high computational ability in IoT systems. It also supported dynamic and decentralized systems. Chen *et al.* [26] explained a trust management protocol that supports scalability and adaptability. It supports SOA-based IoT systems. They have introduced a filtering method to select feedback from IoT node owners having similar interests. The social relationship is categorized as friendship, social contact, and community of interest on the basis of social similarity. Further, an adaptive filter is used to exhibit means of recreation of the predominance in versatile IoT trust convention over EigenTrust and PeerTrust.

Urbieta *et al.* [27] presented a framework that supported dynamic reasoning and adaptability. It is designed on the basis of wEASEL, an abstract service model that uses services and user tasks along with context-aware preconditions, post-conditions, and effects. Evaluation of the framework is performed by combining simple and composite services. Balakrishnan *et al.* [28] have introduced a service composition strategy based on Service Oriented Architecture (SOA). The technique is implemented using a fuzzy-based inference algorithm. The evaluation of the proposed algorithm is performed using real-world web services.

Multiple services on the cloud from different providers can be combined as per the requirements of the user resulting in SC problems. Kurdi *et al.* [29] have explained the problem of service optimization in multi-cloud environments. He has proposed the Multi Cuckoo algorithm using an experimental framework. The proposed algorithm has less running time. The stated problem is complex and comes under the category of NP-hard problems. So, the approach based on heuristics is used to solve the problem. The use of cloud is to provide service to the user irrespective of its geographical location.

4 Service Composition Optimization Problem

The combination of services is referred to as service composition [14]. The solution to this problem can be obtained by utilizing or consolidating accessible services. It encourages proficient application advancement, service reuse,

and the culmination of services with limited use. The objective of SC is to locate an arrangement of services (one for every task) that adds to the composite service and addresses the user's request. There are no specific means of defining the SC that must fulfill the requirements of the user. There are various QoS attributes that are used to characterize the web services. Three main types of QoS parameters such as execution time(t), service cost(c) and reliability(r) are considered [19].

- Execution time (t): It is the time utilized by an IoT service to respond.
- Service cost (c): It is the cost associated with an IoT service.
- Reliability (r): It is defined as the degree of failure of an IoT service.

In light of this, the SC problem is depicted as below: For a given a set of 'n' tasks, the task is defined as the service which is required to fulfill the user requirement,

$$G = \{g_1, g_2, g_3, \dots, g_n\}$$

Each task has 'k' no. of candidate services which provide similar functionality,

$$g_{i=\{s_i^1, s_i^2, ..., s_i^k\}}, \forall i \in [1, 2, ...n]$$

Where 'i' defines the class of a group with similar services in accordance to their task g_i .

The composite service C represents a group of tasks

$$C = \{s_1^a, s_2^b, \dots s_3^k, \forall a, b, c \in [1 \dots k]\}$$

The objective functions for execution time, service cost and reliability are defined as follows:

$$f_1 = \sum_{i=1}^{n} t_i$$
 (1)

$$f_2 = \sum_{i=1}^{n} c_i$$
 (2)

$$f_3 = \prod_{i=1}^n r_i \tag{3}$$

Where t_i is the time of the i^{th} instance of a task. c_i is the cost of an i^{th} instance of a task. r_i is the reliability of the i^{th} instance of a task. By giving equal weight to each QoS property described in equation no. (1), (2) and (3) as three objective functions respectively can be merged into a single objective function as follows in equation no. (4).

$$f = minimize(w_1f_1 + w_2f_2 - w_3f_3)$$
 (4)

such that
$$\sum_{i=1}^{3} w_i = 1$$

5 Algorithms

5.1 Genetic Algorithm

The primary reason for optimization is to utilize the assets in the best way. This implies maximizing or limiting characteristics dependent on a function that in a given circumstance is additionally called an objective function from the scope of accessible qualities. In view of this, we can pick the best solution. The objective might be to save expenses, augment benefits, limit endeavors, etc. [30, 31] There can be two kinds of attributes to be specific namely positive and negative. Positive ought to be boosted and negative ought to be limited. One of the algorithms based on genetics in artificial intelligence is the Genetic Algorithm. The underlying principle of this algorithm is the survival of the fittest. It is preferable to use if the search space is large.

A genetic algorithm provides significant advantages over the typical optimization techniques [32, 33]. The underlying principle of GA is designed on Genetic and revolves around the genetic structures and chromosomes in individual populations. GA stimulates the survival of those individuals which are fittest in consecutive generations to solve the problem. Each generation contains a series of strings homologous to chromosomes.

Each generation represents a solution which is possible corresponding to the value in the search space. After that, individuals in the population are exposed to evolutionary processes. Next-generation solutions are generated with genes that are outstandingly superior to typical generations of previous generations. When the population reaches the stage where children are not created differently from the previous generation, the fitness values are examined in order to identify the most appropriate solution.

A population in GA is defined from the search space, each representing a feasible solution to a problem for which we are designing the solution. Each individual is encoded as a finite length variable of a selected range, typically a binary range perspective. Continuing with the genetic analogy, it is believed that these individuals are chromosomes and, therefore, the variables are similar to the genes. Therefore, a solution which is also called chromosomes consists of many genes. The fitness values are computed according to the exploitation of the objective and assigned to each response that represents the individual solution [34]. Then the best fitness value is identified. By combining chromosome data, the goal of GA is to use the selective "reproduction" of solutions so that the child population is higher than the parent. Once the initial population is generated

randomly, other populations are generated using three operators namely crossover, mutations, and selection. Different steps involved in GA are described as below:

Algorithm 1: Genetic Algorithm [35].

Step 1: Initialize population randomly(P)

Step 2: Compute fitness value corresponding to P using a suitable fitness function

Step 3: Repeat until the best fitness value is found

Step 4: Select parents P1 and P2 from P

Step 5: Crossover is performed on parents resulting in the population (P + 1)

Step 6: Mutation operator is applied to population (P + 1) Step 7: Compute the fitness of the population (P + 1) and then compare it with the best fitness value, update the best value if the fitness value obtained is better than the best value.

5.2 Particle Swarm Optimization (PSO)

It is a stochastic optimization method applied to a population. [36] and is influenced by the social behavior of birds and fish schools. It consists of aggregate particles moving around the search space affected by their best location in the past and the best location of the past of all nearby or a close neighbor. Each repetition of particle speed is updated using:

$$V(t+1) = wV(t) + c_1 \times rand() \times (X_{pbest} - X(t))$$

$$+ c_2 \times rand() \times (X_{gbest} - X(t))$$
(5)

The new location of the particle is calculated as:

$$X(t+1) = X(t) + V(t+1)$$
 (6)

where X(t+1) denotes a new position of the particle. V(t) defines the velocity of the particle. rand is the random number generated between 0 and 1. 'W' inertia weight has been added to control the velocity, ci and c2 are the learning factors to compute X_{pbest} particle's personal best position and X_{gbest} global best position.

The algorithm (below) describes the PSO

Algorithm 2: Particle Swarm Optimization [37]

Step 1: Randomly initialize position $X_i(0)$ and velocity of particles $V_i(0)$

Step 2: Repeat for all number particles

Step 3: Compute the fitness value using equation no. (4)

Step 4: Compute X_{pbest} particle's personal best position and X_{gbest} global best position

Step 5: Update velocity using equation no. (5)

Step 6: Calculate the position using equation no. (6)

Step 7: until the best value is found

6 Evaluation Methodology

This section describes the values of various parameters in the implemented algorithms, data set details and solution encoding scheme.

6.1 Data sets

As a part of the experiment, the number of tasks is ranged from 10 to 30 in steps of 10. For each task 'n', 'm' is the no. of candidates having different QoS parameters. The value of n ranges from 10 to 30. The QoS parameters have different ranges. The range of service cost is from 1 to 100, execution time varies from 0.1 to 0.3 and reliability is between 0.7 and 0.9.

$$1 \le execution \ cost \le 100$$
 (7)

$$0.1 \le execution time \le 0.3$$
 (8)

$$0.7 \le reliability \le 0.9$$
 (9)

6.2 Experimental settings

The three objective functions defined as in equation (1), (2) and (3) have equal weight. For both algorithms, the population size is 100 and the algorithm is executed with an evaluation function of up to 10,000 to guarantee the equivalent basis for its performance. In GA, random mutations have been applied to maintain individual diversity by changing the value of randomly selected genes with a probability of 0.1. The parent selection mechanism is a roulette wheel along with the one-point type of crossover. The crossover probability is taken to be as 0.8 as we get better results on this value. In PSO, the value of inertia weight is taken as 1 acceleration coefficients value of 2.

6.3 Solution encoding

The solution for this problem is to include the possible set of tasks with minimum service cost, minimum execu-

Table 1: Parameter settings

| Population size | 100 | | |
|-----------------------------------|----------------|--|--|
| No. of generations | 100 | | |
| GA | | | |
| Parent selection Method | roulette wheel | | |
| Type of Crossover | one-point | | |
| Value of Crossover probability | 0.8 | | |
| Type of Mutation | Random | | |
| Mutation probability | 0.1 | | |
| PSO | | | |
| W(inertia weight) | 1 | | |
| c1, c2(acceleration coefficients) | 2 | | |
| Maximum no. of iterations | 20 | | |
| | | | |

tion time and maximum reliability of the service. Therefore, each chromosome represents a possible composition of the service. Each solution is coded as a series of integers as an array of integer type. Indices of an array represent the task number and the data indicate candidate number for the task. The coding process is shown schematically below:

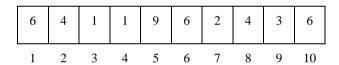


Figure 1: Solution Encoding

The above instance shows a string of solutions with 10 tasks and 10 candidates for executing each task. In this service composition, the sixth candidate for the first task is shortlisted, the fourth candidate for the second task is shortlisted, and the first candidate for the third task is shortlisted. To compose the set of most appropriate services, two of the metaheuristic algorithms namely GA and PSO have been used. The fitness value for each row of the solution set is computed and the minimum value corresponding to the composed services is selected.

7 Results and Analysis

The performance of the algorithms depends on the fitness values. The fitness value is an important component to compare the performance of both algorithms. Since the calculations are stochastic so, to represent their inherent arbitrariness, the algorithms are run for 30 times and the av-

Table 2: Comparison of the fitness values obtained by the algorithms

| No. Tasks | No. of Candidates | GA | | PS0 | | |
|-----------|-------------------|---------|--------|----------|---------|----------|
| | | Mean GA | Std_GA | Mean PSO | Std_PSO | P-value |
| 10 | 10 | 1.3074 | 0.1732 | 2.7942 | 0.157 | 8.74E-14 |
| 10 | 20 | 2.1869 | 0.3085 | 4.378 | 0.2065 | 3.16E-13 |
| 10 | 30 | 2.3831 | 0.2709 | 4.2213 | 0.1611 | 3.89E-13 |
| 10 | 40 | 2.3906 | 0.3897 | 4.2965 | 0.1807 | 3.91E-11 |
| 10 | 50 | 2.4936 | 0.1851 | 4.3747 | 0.107 | 3.04E-16 |
| 20 | 10 | 5.4635 | 0.311 | 8.3558 | 0.2251 | 4.60E-15 |
| 20 | 20 | 5.1382 | 0.4612 | 8.0456 | 0.2719 | 1.31E-12 |
| 20 | 30 | 5.5686 | 0.3737 | 8.2768 | 0.1727 | 4.88E-14 |
| 20 | 40 | 5.3205 | 0.3002 | 8.0194 | 0.147 | 1.37E-15 |
| 20 | 50 | 5.7806 | 0.3359 | 8.4911 | 0.1099 | 3.38E-15 |
| 30 | 10 | 8.3923 | 0.4337 | 12.1446 | 0.1973 | 2.13E-15 |
| 30 | 20 | 8.5143 | 0.4529 | 12.0583 | 0.2434 | 2.17E-14 |
| 30 | 30 | 8.6302 | 0.7546 | 12.5006 | 0.2278 | 7.22E-12 |
| 30 | 40 | 8.6247 | 0.6114 | 12.0243 | 0.2338 | 2.80E-12 |
| 30 | 50 | 8.7545 | 0.3954 | 12.0655 | 0.2634 | 1.80E-14 |

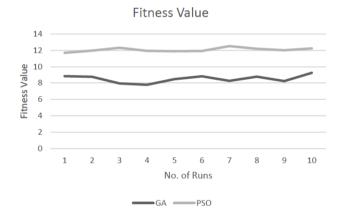


Figure 2: Comparison of Fitness values for 30 tasks and 20 instances of each task

erage value with the standard deviation of fitness values is obtained. The fitness values are analyzed using a t-test with 95% confidence. Figure 2 shows the comparison of fitness values for 30 tasks and 20 instances of 10 runs. It can also be seen that GA converges to a better minimum value than PSO.

As shown in Figure 3, the fitness value obtained using GA is always lesser than that obtained using PSO for any no. of tasks ranging from 10 to 30 with any no. of candidates ranging from 10 to 50 with a step size of 10 for service composition problem in IoT. It indicates that GA can find more suitable composition of services than PSO. Additionally, the mean and standard deviation of the fitness obtained by the algorithms in each test instance along with

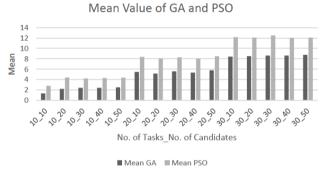


Figure 3: Comparison of Mean value of Fitness function for varying no. of tasks and their instances

the p-value (of t-test) is shown in Table 2. The p-value obtained for each case is less than 0.05, which indicates that there is statistically significant difference between the fitness values obtained using both algorithms.

Figure 3 shows a comparison of the Mean value of Fitness function for varying no. of tasks and their instances. From Table 2, it is clear that the GA fitness value is four times better than the PSO's. As a result, we found that GA can find a better solution for SC problems than PSO in terms of reliability, cost and execution time. The consistency in GA performance can also be seen from the standard deviation value. The result shows that the value of p is much smaller than 0.05. This indicates that the performance of the algorithm in similar values is significantly different from 95% confidence. Figure 4 shows the Box Plots representing GA and PSO fitness values for 10 tasks with 10

62 — N. Kashyap et al. DE GRUYTER

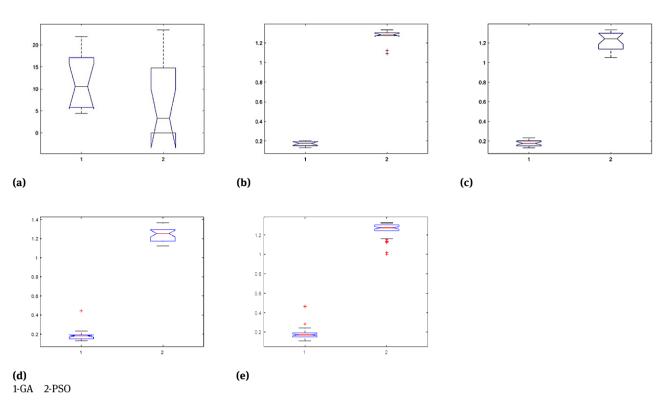


Figure 4: Comparison of the fitness values obtained by GA and PSO using boxplots for 10 tasks with 10 candidates a) 5 runs b)10 runs c) 20 runs d) 30 runs e) 50 runs

candidates in 5 runs, 10 runs, 20 runs, 30 runs, and 50 runs. The graph results also show that the GA approach can find the best SC compared to PSO. In all cases, the notches do not overlap in the box plot, so the result can be considered important. It is concluded that GA works well for service composition problems in IoT. GA is capable to find better solution that is minimum time and cost with maximum reliability even though the candidate services per task increases.

8 Empirical evaluation of the implemented algorithms on the real-world application data set

8.1 Description of the data set

To test the implemented algorithms, a well-known data set is used to facilitate the differences caused by the variance in QoS by invoking different services at different times and by different users under different network environments. This dataset is released by Gary White *et al.* [38], which comprises response time and throughput of various ser-

vices. They used sensors to generate the dataset includes a motion sensor, a gas sensor, a temperature sensor, a pressure sensor, a humidity sensor module, and photoresistor Module. The data from each sensor connected to each Arduino device is made available as a service. Arduino enables the following services: motion service, temperature service, humidity service, gas service, real altitude service, pressure at sea level service, temperatureBMP, altitude service, photosensor service, and pressure service. The response time is measured in seconds and the throughput is measured in Kbps.

8.2 Experimental Results

The implemented algorithms are executed using real-world data sets in order to determine the efficacy of the SC solution in IoT. The empirical analysis on the real data set has been implemented using two QoS parameters namely response time and reliability for 10 tasks and 40 candidates. Both algorithms are run for 30 times and the average value with the standard deviation of fitness values is obtained. The lesser value of the execution time and more value of reliability are the indicators of better performance.

Table 3 shows the comparison of fitness values obtained by using both algorithms. The results show the superiority of GA is more prominent as the candidates are 40 which is large for composition of 10 services. As it can be seen the average value obtained for execution time is less and the value for reliability is more thus experimental results validate that the results obtained using GA are better than the PSO. The results illustrate that the usage of metaheuristic search algorithms to obtain the solution for SC problem in IoT are feasible and effective in providing the solution.

Table 3: Comparison of fitness values obtained by both algorithms

| Parameter: Response Time | | | | | | | |
|--------------------------|----------|----------|--|--|--|--|--|
| Algorithm | Mean | Std. Dev | | | | | |
| GA | 1.24E-02 | 3.49E-03 | | | | | |
| PS0 | 6.15E-03 | 3.74E-01 | | | | | |
| Parameter: Reliability | | | | | | | |
| Algorithm | Mean | Std. Dev | | | | | |
| GA | 1.77 | 3.39E-03 | | | | | |
| PSO | 0.004284 | 4.96E-01 | | | | | |

9 Conclusions and future work

The role of optimization algorithms using empirical evaluation of GA and PSO is shown to solve the problem of service composition in IoT. The problem is to minimize service cost, execution time, reliability and select candidates to execute each task in the service configuration. The experiment is executed for a step size of 10 with the number of tasks ranging from 10 to 30, and the number of candidates for each assignment is also in the range of 10 to 50 with the same step size. The results of the experiments obtained with the algorithms generate 15 test data instances. Also, statistically significant results are obtained with 95% confidence level. The results obtained after the empirical evaluation of the implemented algorithms on the real-world application data set revealed that the solutions obtained using GA are more eloquent than that of PSO. As part of future research, we will try to hybridize the genetic approach with other metaheuristic search methods. It shall be also applied to the datasets from various other applications based on IoT.

References

- [1] Lee, In, and Kyoochun L., The Internet of Things (IoT): Applications, investments, and challenges for enterprises, Business Horizons, 2015, 58(4), 431-440
- [2] Zanella A., Bui N., Castellani A., Vangelista L., Zorzi M., Internet of things for smart cities, IEEE Internet of Things journal, 2014 Feb,1(1),22-32
- [3] Moreno M. V., Zamora M. A., & Skarmeta A. F., 2015, An IoT based framework for user-centric smart building services, International Journal of Web and Grid Services, 11(1), 78-101
- [4] Duan R., Chen X., Xing T., A QoS architecture for IOT, IEEE International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, 717-720
- [5] Liu J., Li X., Chen X., Zhen Y., Zeng L., Applications of internet of things on smart grid in China, 13th IEEE International Conference on Advanced Communication Technology, 2011, 13-17
- [6] Winkler Á., Horváth B., Intelligent decision support technologies in public and individual transport, Intelligent Decision Technologies, 2017, 1-9
- [7] Conzon D., Brizzi P., Kasinathan P., Pastrone C., Pramudianto F., Cultrona P., Industrial application development exploiting IoT vision and model driven programming, 18th IEEE International Conference on Intelligence in Next Generation Networks, 2015, 168-175
- [8] Bao F., Chen R., Trust management for the internet of things and its application to service composition, IEEE international symposium on a world of wireless, mobile and multimedia networks, 2012, 1-6
- [9] Bonetto R., Bui N., Lakkundi V., Olivereau A., Serbanati A., Rossi M., Secure communication for smart IoT objects: Protocol stacks, use cases and practical examples, IEEE international symposium on a world of wireless, mobile and multimedia networks (WoW-MoM), 2012, 1-7
- [10] Li L., Jin Z., Li G., Zheng L., & Wei Q., Modeling and analyzing the reliability and cost of service composition in the IoT: A probabilistic approach, IEEE 19th International Conference on Web Services (ICWS), 2012, 584-591
- [11] Guinard D., Trifa V., Karnouskos S., Spiess P., & Savio D, Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services, IEEE transactions on Services Computing, 2012 3(3), 223-235
- [12] Zeng L.Z., Bouguettaya B., Ngu A.H.H., Jayant K., Henry C, QoSaware middle ware for Web Services composition, IEEE Transactions on Software Engineering, 2004, 30(5), 311-327
- [13] Singh D., Tripathi G., Jara A.J., A survey of Internet-of-Things: Future vision, architecture, challenges and services, IEEE World Forum on Internet of Things (WF-IoT), 2014 Mar 6, 287-292
- [14] Wang W., De S., Toenjes R., Reetz E., Moessner K., A comprehensive ontology for knowledge representation in the internet of things, IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 2012 Jun 25, 1793-1798
- [15] Xu B., Da Xu L., Cai H., Xie C., Hu J., Bu F., Ubiquitous data accessing method in IoT-based information system for emergency medical services, IEEE Transactions on Industrial informatics, 2014 May,10(2),1578-86
- [16] Han S.N., Lee G.M., Crespi N., Heo K., Van Luong N., Brut M., Gatellier P., DPWSim: A simulation toolkit for IoT applications us-

- ing devices profile for web services, IEEE World Forum on Internet of Things (WF-IoT), 2014 Mar 6, 544-547
- [17] Desai P., Sheth A., Anantharam P., Semantic gateway as a service architecture for iot interoperability, IEEE International Conference on Mobile Services, 2015 Jun 27, 313-319
- [18] Li, Q., Dou, R., Chen, F., & Nan, G., A QoS-oriented Web service composition approach based on multi-population genetic algorithm for Internet of things', International Journal of Computational Intelligence Systems, 2014, 7(2), 26-34
- [19] Kashyap N., Kumari A.C., Hyper-heuristic approach for service composition in internet of things, Electronic Government, an International Journal, 2018,14(4),321-39
- [20] Nam W., Cha R., & Kil H., Optimal algorithm for Internet-of-Things service composition based on response time, International Journal of Web and Grid Services, 2016, 12(4), 388-406
- [21] Strunk, A., QoS-aware service composition: A survey, IEEE 8th European Conference on Web Services (ECOWS), 2010, 67-74
- [22] Li L., Li S., Zhao S., QoS-aware scheduling of services-oriented internet of things, IEEE Transactions on Industrial Informatics, 2014 May,10(2), 1497-505
- [23] Liu H., Zhong F., Ouyang B., Wu J., An approach for QoS-aware web service composition based on improved genetic algorithm, International Conference on Web Information Systems and Mining, 2010 Oct 23, 1, 123-128
- [24] Baker T., Asim M., Tawfik H., Aldawsari B., Buyya R., An energyaware service composition algorithm for multiple cloud-based IoT applications, Journal of Network and Computer Applications, 2017, 89, 96-108
- [25] Fortino G., Guerrieri A., Russo W., Savaglio C., Integration of agent-based and cloud computing for the smart objects-oriented IoT, IEEE 18th international conference on computer supported cooperative work in design (CSCWD), 2014 May 21, 493-498
- [26] Chen R., Guo J., Bao F., Trust management for SOA-based IoT and its application to service composition, IEEE Transactions on Services Computing, 2016 May 1, 9(3), 482-95.
- [27] Urbieta A., González-Beltrán A., Mokhtar S.B., Hossain M.A., Capra L., Adaptive and context-aware service composition for IoT-based smart cities, Future Generation Computer Systems. 2017 Nov 1,76,262-74

- [28] Balakrishnan S.M., Sangaiah A.K., Integrated QoUE and QoS approach for optimal service composition selection in internet of services (IoS), Multimedia Tools and Applications. 2017 Nov 1, 76(21), 889-916
- [29] Kurdi H., Ezzat F., Altoaimy L., Ahmed S.H., Youcef-Toumi K., MultiCuckoo: Multi-Cloud Service Composition Using a Cuckoo-Inspired Algorithm for the Internet of Things Applications, IEEE Access, 2018, 6, 567, 37-49
- [30] Goldberg D.E., Holland J.H., Genetic algorithms and machine learning. Machine learning, 1988 Oct 1, 3(2), 95-9
- [31] Modi K.J., Garg S., Dynamic web services composition using optimization approach. International Journal of Computer Science & Communication, 2015 Apr, 6(2), 285-93
- [32] Gen M., Cheng R., Genetic Algorithms & Engineering Design, John Wiley& Sons, Inc., New York, 1997
- [33] Holland J.H., Genetic algorithms, Scientific American, 1992 Jul 1, 267(1), 66-73.
- [34] Mardukhi F., NematBakhsh N., Zamanifar K., Barati A., 2013, QoS decomposition for service composition using genetic algorithm, Applied Soft Computing, 13(7), 3409-3421
- [35] Kim M., Ko I.Y., An efficient resource allocation approach based on a genetic algorithm for composite services in IoT environments, IEEE International Conference on Web Services, 2015 Jun 27, 543-550
- [36] Bai Q., Analysis of particle swarm optimization algorithm. Computer and information science, 2010 Feb 1, 3(1), 180
- [37] Li J. Q., Zhang S.P., Yang L., Fu X.H., Ming Z., Feng G., Accurate RFID localization algorithm with particle swarm optimization based on reference tags, Journal of Intelligent & Fuzzy Systems, 2016 Jan 1, 31(5), 2697-706
- [38] White G., Palade A., Clarke S., QoS Prediction for Reliable Service Composition in IoT. In: Braubach L. et al. (eds) Service-Oriented Computing ICSOC 2017 Workshops. ICSOC 2017. Lecture Notes in Computer Science, vol 10797. Springer