Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.1161 – 1166

RESEARCH ARTICLE

Approaches for Web Service Selection

Vijayalaxmi S Jeure[1], Y.C.Kulkarni[2]

[1]Bharati Vidyapeeth Deemed University, College of Engineering, Pune

[2]Bharati Vidyapeeth Deemed University, College of Engineering, Pune

[1] Vijayalaxmi.jeure@gmail.com; [2] yckulkarni@yahoo.com

Abstract— *Web services is a technology for transmitting data over the Internet and allowing programmatic access to that data using standard Internet protocols. There may be the services providing similar properties, effects, capabilities, and interfaces. Selecting one such similar service is that matches the users requirement is a difficult task. Quality of Service (QoS) attributes provides a differentiation among the competing services, allowing a prospective user to choose the services which best suit to his/her QoS requirements. This paper addresses precisely this component. In this paper we have discussed different approaches for web service selection and proposed particle swarm optimization algorithm for the selection of web services to match consumers with services based on QoS attributes as closely as possible. Particle swarm optimization is the population based stochastic optimization technique. It is the population based search procedure. In this the population of agents called particles is created and uniformly distributed on the region. Particle's position is evaluated according to the function. If the particle's current position is better than it's previous position it is updated. Particle is moved to the new position. Accordingly the evaluation is done to find the best suitable position.*

Keywords— *Web services, Quality of Service, Service Selection, Genetic Algorithm, Memetic Algorithm, Swarm Intelligence*

I. INTRODUCTION

Business processes and consumers use the services which are the implementation of well defined business processes or functionalities. Web services are software systems designed to support interoperable machine-to-machine interaction over a network. The Web Services architecture is based upon the interactions between three roles: Service provider, Service registry, Service requestor. The owner of the service is service provider who hosts access to the services. Service requestor who interacts with the service requires certain functions to be satisfied. Service registry is the searchable registry of service description where service providers publish their service descriptions. This interaction involves different operations as publish, find, and bind operations. With the publish operation is used to publish service description in order for service requestor to find them. With the find operation the service requestor retrieves a service description directly or queries the service registry for the service required. The bind operation is used by the service requestor to invoke or initiate an interaction with the service at

runtime. Selecting a best web service as per the client's requirement is the crucial process. Many web services are increased in the internet with similar functionality from these web services selecting the one which will be as per the client's requirement is the subtle task for web service operators. The architecture can be shown as in the Fig 1.

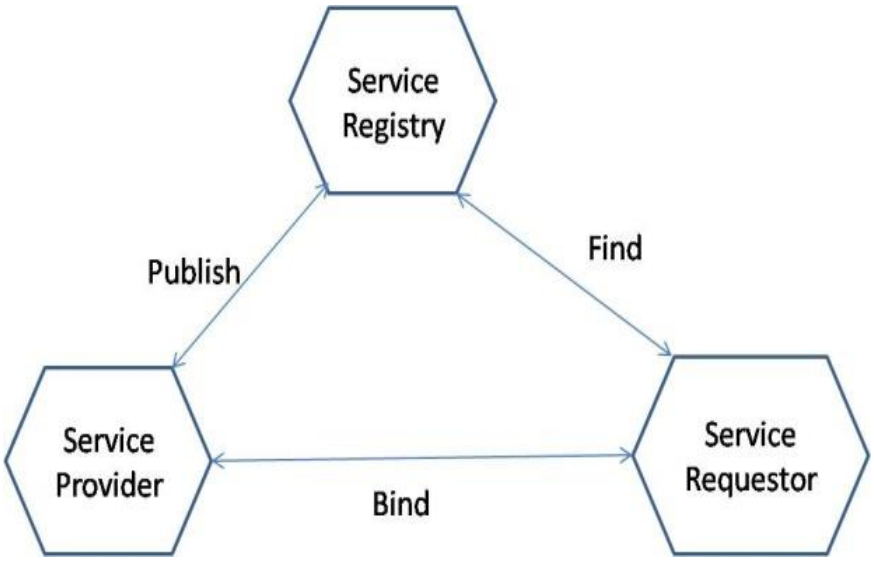

Fig 1.Web Service Architecture

II. QUALITY OF SERVICE

Quality of Service can be defined as "is the capability to respond to the requirements (constraints) of a client and to fulfil these needs with the best criteria (preferences) established by the client. It is calculated based on the non-functional properties of the service".

The functional property of the service can specify what the service can do it. The non functional properties specify how the service can do it. The non-functional property can be defined as "they are all the properties defining the service and the services, including its performances, that characterize how and in which conditions (context) the functionality of the service is executed".

The current service oriented architecture mainly relies on functional properties of web services. Service registries lack mechanisms to handle non functional properties of web services. Non-functional properties can be measured to establish the Quality of Service. The non functional parameters such as waiting time, reliability, availability allows consumer to have confidence in the use of services by aiming to experience good service performances.

In order to select the best service adapted to client's requests, we need some method capable to evaluate and compare different services providing the same functionalities. In this context, Quality of service can be defined as offering service differentiation based on the requirements of clients and applications. This implies the evaluation of the non-functional properties of the service described in a quality model. There is no generic quality model proposed in the literature to evaluate quality of services over web services. Nowadays, most of the quality model proposals are focused in ad hoc applications.

A. Non Functional Properties

Non-functional requirements or quality properties such as performance, scalability, flexibility, and security are crucial for the success of nearly every software project. These non functional requirements are even more important than functional requirements as they can hardly be implemented after making the major design decisions. As these non-functional properties play an important role in discovery, selection and substitution of services, the software architecture should consider these non functional requirements [14]. Based on some non functional properties the services which fulfil a user request and provide the same functionality like accuracy, availability are selected. Some non functional properties can be defined as

TABLE I
NON FUNCTIONAL PARAMETERS

Availability	Quality of a service to be in a temporal and spatial context ready to deliver the demanded service. It can be calculated as (1 − Down Time))/(Measurement Time). The Measurement Time can be adapted to the application.
Accuracy	Error rate that produces a service. It is calculated as the number of errors divided by the total number of executions.
Execution Time	Measures precisely the time the provider takes to execute a request.
Reliability	Probability with which the provider correctly answers a request within a maximum expected time. It is measured as the number of success request divided by the number of request.
Robustness	Establishes the capacity of the system to give an accurate answer in presence of invalid, incomplete or conflicting inputs. These can be estimated at the provider side or tested and making an average from the last experiences.
Security	Comprises other non-functional properties (i.e. data encryption, authentification…). It evaluates the security in the transaction who assures the ACID properties (Atomicity, Consistency, Isolation and Durability). This value can be presented by the provider separately or as an aggregation of the different qualities.
Throughput	Measures the maximum network capacity the client and the provider can get to communicate. It is time dependent, so if a quality reservation is not possible on the network, the throughput would be either the maximum throughput registered or the average.

B. Non Functional Properties of Web Services and Quality of Service

As stated earlier every service has functional and non functional properties. What the service does that is what the service takes as input, what is does and what is the output are defined by functional properties. Non-functional properties define how the service reacts with all these input, processing and output. Behaviour of the service is not changed by the non functional properties.

Quality of Service changes depending on the client and domain. Depending on the importance of non functional properties of the client a service has a quality. For example if a particular service has high availability but no codification system. For the client if the codification is important than availability. Then quality of service for availability is poor as it does not respond to the client demand

III. SERVICE SELECTION ALGORITHM

The problem of service selection on the web consists of having an efficient algorithm that can match multiple service consumers and service providers efficiently, while optimizing multiple objectives (QoS parameters). Multiple clients requesting similar services should be satisfied, and secondly, the assignment process of the service consumers and the service providers should be optimized. Please note that one consumer can only be matched with one provider [1]. Since we have several service consumers and equally numbered service providers, the aim is to match the consumer–provider pairs as closely as possible using a Genetic Algorithm (GA), Memetic Algorithm (MA) and Particle Swarm Optimization (PSO) approach.

The QoS criteria in the context of services are execution price, execution time, reliability, reputation, and availability. The values of these QoS parameters range between 0 to 1. Each consumer provides the QoS values based on its requirement of how the request must be executed, and each service provides the value based on its task execution capability. The service provider has a value for each QoS parameter. The service consumer requests a service provider specifying an upper and lower value for each QoS parameter, whereby for some QoS attributes the lower or upper bound is preferred [1]. In particular, the lower bound is preferred for execution price and execution time, and the upper bound is preferred for reliability, reputation and availability.

A. Genetic Algorithm

GA is a global optimization algorithm that models natural evolution [4]. In GA, individuals form a generation. An individual corresponds to one match. The match is implemented as a vector, which is also referred to as a chromosome. Dimensions in the vector correspond to providers, and values correspond to consumers. Therefore, if the vector has value 2 at its 4th position (dimension), consumer 2 is matched with provider 4. Every number representing a consumer can only be at one position in the vector; otherwise, the vector represents a non-valid match. At the beginning, the first population is randomly initialized. After that, the fitness of the individuals is evaluated using the fitness function. After the fitness is evaluated, individuals have to be selected for paring. The selection method used is tournament selection [1].

Always two individuals are paired, resulting in an offspring of two new individuals. In the pairing phase, a random crossover mask is used, i.e. the positions (dimensions) for which crossover occur are selected randomly. If crossover occurs at certain positions (dimensions), individuals that are mated exchange their values at that position and the resulting individuals are used as offspring. The crossover has to make sure that the offspring presents a valid match. Therefore, if two values are exchanged, other positions in the two match vectors are usually effected as well. The offspring faces mutation with a certain low probability. After mutation, the fitness of the offspring is calculated. Then, either all individuals from the last generation compete against the whole offspring, or the offspring only compete with its corresponding parents. In this implementation, all individuals from the old generation compete with all individuals in the new generation. In order to implement this, all individuals are ordered by their fitness score, using a non-recursive advanced quicksort algorithm, which after sorting truncates the lower half. After the new generation is selected, the GA will start over, and continue with parent selection and crossover until a certain number of iterations are reached [1].

B. Memetic Algorithm

Evolutionary algorithms are not well suited for fine-tuning the search, in particular in complex combinatorial spaces, and therefore, researchers have developed hybridization methods to overcome this problem and to improve the efficiency of the search [5]. The combination of using evolutionary algorithms as well as local search techniques was named Memetic Algorithms (MAs). MAs are basically an extension of evolutionary algorithms that apply a separate process to refine solutions by improving the fitness of the individuals with methods such as hill-climbing or simulated annealing. MAs were inspired by the models of adaptation in natural systems, in particular the combination of the evolutionary adaptation of a population with individual learning. GAs on the one hand and local search on the other hand, are captured within MAs, thus rendering a methodology that balances well between generality and problem specificity. The name of MAs was inspired by Richard Dawkins' concept of a meme, which represents a unit of cultural evolution that can exhibit local refinement [6]. A meme represents a learning or development strategy [7]. Memetic algorithms are also known as Hybrid Evolutionary Algorithms [8], Baldwin an Evolutionary Algorithms [9], Lamarckian Evolutionary Algorithms [10], Cultural Algorithms or Genetic Local Search. All techniques combine local search heuristics with the evolutionary algorithms' operators. Combinations with constructive heuristics or exact methods may also be considered within this class of algorithms. In this research, we apply an exact method for the local search. For some problem domains, MAs have been shown to be both more efficient and more effective than traditional evolutionary algorithms with regards to requiring orders of magnitude fewer evaluations to find optima, and identifying higher quality solutions. In particular, for many combinatorial optimization problems, where large instances have been solved to optimality, and where other meta-heuristics have failed to produce comparable results, such as the quadratic assignment problem and the traveling salesman problem**,** MAs have proven themselves to be very effective[1][7].

IV. SWARM INTELLIGENCE

Swarm intelligence is the collective behaviour of a group (swarm) of animals as single living creatures where collective intelligence emerges via grouping and communication. When the route of a swarm of ants is blocked, it can be observed that they find another new shortest route to their destination; this is robustness. These agents (ants) can be added or removed without compromising the total system due to its distributed nature, this is reliable. Single parts may be break down without impairing the overall system, these complex systems are convenient to work because of simplicity of their individual parts. There are two popular swarm inspired methods: Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO).

A. Particle swarm optimization: Overview

Particle Swarm Optimization (PSO) is a population based stochastic optimization technique developed by Eberhart and Kennedy in 1995 inspired by the social behaviour of flocks of birds and school of fish. In particle swarm intelligent system, bird

flocking behaviour is simulated for optimization problems. The concept is simple, has few parameters, is easy to implement, and has found applications in many areas [13].

Consider the scenario, group of birds are randomly searching for food in the area. Only one particular location of the area being searched contains food. None of the birds knows where the food is. The sole knowledge they have is how far the food is in each iteration. So, the best and most effective strategy to find food is to follow the bird that is nearest to it. Each single solution is a 'bird' in the search space and can be treated as a 'particle'. Each particle has fitness value. The fitness function to be optimized is evaluated for every particle, and has velocities which direct the flying of the particles. By following the currently optimum particles, the particles fly through the problem space. In PSO, each particle is initialized with a group of random particles, which are solutions; optima are searched for by updating subsequent generations. In every iteration, each particle is updated by following two 'best' values. The first best value is the best solution or fitness it has achieved so far. The variable pbest contains the best fitness value. The best value obtained so far by any particle in the population which is tracked by the particle swarm optimizer is global best and called as gbest. These best solutions are obtained from a relation maintained by the current particle's velocity and position [13].

B. Particle swarm optimization: Algorithm and Pseudo Code
As explained in [15]

1) Algorithm

v[] = v[] + c1 * rand() * (pbest[] - present[]) + c2 * rand() * (gbest[] - present[])

present[] = persent[] + v[]

Where v[] is the particle velocity, persent[] is the current particle (solution). pbest[] and gbest[] are defined as stated before. rand () is a random number between (0,1). c1, c2 are learning factors. usually c1 = c2 = 2.

2) Pseudo Code

```
For each particle
   Initialize particle
END
Do
   For each particle
      Calculate fitness value
      If the fitness value is better than the best fitness value (pBest) in history
         set current value as the new pBest
   End
   Choose the particle with the best fitness value of all the particles as the gBest
   For each particle
      Calculate particle velocity according equation
      Update particle position according equation
   End
```

V. CONCLUSION

In this paper we propose approaches like Genetic Algorithm (GA), Memetic Algorithm (MA) and Particle Swarm Optimization (PSO) for web service selection based on the non functional parameter. Accordingly the quality of service is checked. Quality of Service plays major role in the selection of web services. Quality of Service is determined by the execution of non functional parameters like availability, robustness, simplicity, reliability etc. Depending on the client's requirement of the service and the non functional parameters the web service should be selected. There are many approaches for the Web Service Selection as discussed in the paper. We are proposing Particle Swarm Optimization for the selection of web service as this optimization technique is very simple, it has very few parameters, and easy to implement. It has found applications in many areas, Web Service selection is one of them.

REFERENCES

[1] Simone A. Ludwig, "Memetic Algorithm for Web Service Selection"**,** Proceedings of the 3rd workshop on Biologically inspired algorithms for distributed systems international conference on automatic communication and computing, 2011.

[2] Shang-Chia Liu, Sung-Shun Weng, "Applying Genetic Algorithm to Select Web services Based on Workflow Quality of Service", *Journal of Electronic Commerce Research*, VOL 13, NO 2, 2012.

[3] Taher, L., El Khatib, H., "A framework and QoS matchmaking algorithm for dynamic web services selection", *Proceedings of the 2nd International Conference on Innovations in Information Technology* (IIT'05), 2005.

[4] Holland, J.H., "Adaptation in Natural and Artificial Systems", University of Michigan Press, Ann Arbor. 1975

[5] Wolpert, D., Macready, W., "No free lunch theorems for optimization", *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp.67-82, 1997

[6] Dawkins, R,. *The Selfish Gene*, New York: Oxford Univ. Press, 1976

[7] Krasnogor, N., Smith, J., "A Tutorial for Competent Memetic Algorithms: Model, Taxonomy, and Design Issues", *IEEE Trans. On Evolutionary Computation*, vol. 9, no. 5, 2005

[8] Vazquez, M., Whitley, L., "A hybrid genetic algorithm for the quadratic assignment problem", Proceeding *on Genetic Evolution Computing Conference.* pp. 135-142, 2000

[9] Ku, K., Mak, M. "Empirical analysis of the factors that affect the Baldwin effect", *Lecture Notes in Computer Science, Parallel Problem Solving From Nature*, pp. 481-490, 1998

[10] Morris, G.M., Goodsell, D.S., Halliday, R.S., Huey, R., Hart, W.E., Belew, R.K., Olson, A.J., "Automated docking using a lamarkian genetic algorithm and an empirical binding free energy function, Journal Computing". *Chem.*, vol. 14, pp. 1639-1662, 1998

[11]M.C. Frédérique, Claudia Catalina, Evaluation "Framework for Quality of Service in Web Services: implementation in a pervasive environment", Master Thesis Master Research in Informatics Specialized in Technologies of Information & Web

[12] J. O'sullivan, D. Edmond and A.T. Hofstede., "What's in a service? Towards accurate description of non-functional service properties", *Distributed and Parallel Databases Journal*, September 2002.

[13] N.P.Padhy, *Artificial Intelligence and Intelligent Systems*, Oxford University Press, 12th impression 2013.

[14] WSMO Deliverable, D28.4 V0.1, *Non Functional Properties of Web Services* WSMO Working Draft – October 25, 2006.

[15] Mona niyakan Lahiji, *Particle Swarm Optimization (PSO) Algorithm*, CNB scholar journals, Journal of Biology and today's world volume 1, issue 1, 2012

