International Journal of Engineering & Technology, 7 (1.9) (2018) 107-114

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

An efficient scheduling framework for improving QoS in web service composition

T. N. Aruna [1] *, Dr. Vijivinod [2]

[1] *Research Scholar, Department of Computer Applications, Dr. MGR Educational and Research Institute University, Chennai, India*
[2] *Professor and Head of the Department of Computer Applications, Dr. MGR Educational and Research Institute University, Chennai, India*
**Corresponding author E-mail: arunarajesh18@gmail.com*

Abstract

Proficient QoS-based web service determination from the various number of practically substitutable web service to convey complex undertakings are a present call from the business world. QoS-based web service's determination is a multi-target improvement issue. Current methodologies like FCFS, Priority and Multi-queue to explain it. Be that as it may, the execution time of QoS-based web services choice to accomplish the most extreme wellness esteem is as yet a worry for handy circulated applications. This paper proposes a productive method to take care of this issue utilizing the Social Spider Algorithm (SSA).

Keywords: *Web Service Composition; Optimization; Scheduling; Cloud Computing; Social Spider Algorithm.*

1. Introduction

Nowadays, web service pulls in the consideration of the examination group to manage the consistent advancement and arrangement of business forms. Since a solitary web benefit gives basic capacity, it may not fulfill the client's prerequisites. This calls for web service with various capacities to be receptive to the present condition. To this end, web service supplements each other in web service organization to convey complex undertakings. Oriented Architecture (SOA) gives an agreeable method for incorporating and reusing web benefits by receiving standard interfaces (WSDL) and standard conventions, for example, Service Oriented Architecture Protocol (SOAP) to convey complex business forms.
The Web customer programming processing plant helps modelers and engineers make such composite web requisitions. Composite Web requisitions are made out of various discrete, autonomous, yet practically finish pieces. These pieces are coordinated together inside a Web website [1]. Security of web services for any web provision is focused around norms included in OASIS-WSS adaptation 1.0 determination. These guidelines deliver how to give security to messages traded in a web nature's domain. The particular characterizes the central offices for ensuring the trustworthiness and privacy of a message and gives instruments to cohorting security-related cases with the message. Web services security is a message-level standard focused around securing SOAP messages through XML advanced mark, privacy through XML encryption, and accreditation engendering through security tokens. To secure Web services, you must consider an expansive set of security necessities, including confirmation, approval, protection, trust, trustworthiness, secrecy, secure interchanges channels, alliance, designation, and reviewing over a range of provision and business topologies. One of the key prerequisites for the security demonstrate in today's business surroundings is the capability to interoperate with once incongruent securities advances in heterogeneous situations.

These days the vast majority of the business methodology is created under SOA architectures or MVC architectures. To evade administration occupied, seclusion, reusability, expandability and interoperability synthesis of web services is sent in web requisitions. Different area based business, government; training and social insurance are included in different online transactions which are customized utilizing organization. All the services are joined into a solitary workflow for a business handle in SOA structural engineering. All the services, by and large, characterized and utilized generally are for the most part web services. Today, the usefulness, creation of web services is taken mind by coordination. Present different online transactions are customized utilizing coordination as a part of a few domains comprises business, government, training and social insurance.

2. Related works

Building web service creations considering the various numbers of web services which are comparable in their usefulness features the significance of choosing the most appropriate service for that composition. Web service creation comprises of different administration hubs, where every hub compares to an administration from a group of service (i.e., a gathering of practically comparative web service). Subsequently, the number of piece designs increments with the expanding number of web service per group. Web service is chosen in light of their useful properties or non-utilitarian, nature of administration (QoS), properties. QoS determination calculations are intended to meet clients' worldwide limitations [2-3].
QoS web services determination has been demonstrated NP difficult problem [4]. Many components prompt this high multifaceted nature, for example,

1) The clients set distinctive compels for determination and structures,

2) QoS parameters of a web benefit vary than what is asserted by its supplier,
3) The expanding number of accessible practically comparative web services and
4) Inside this colossal pursuit space, the composition might be worked in numerous ways.

The last two variables increment time intricacy of the web services choice. Early research illuminates web benefit determinations as straight programming problems [5-6]. At the show, the determination issue is formalized by the methods for improvement algorithms [7-8]. Hereditary Algorithm (GA) and Particle Swarm Optimization (PSO) calculation are the most utilized methodologies for upgrading web services choice. Contrasted with GA, PSO has a couple of parameters and quick joining speed. The execution time execution of web services choice to accomplish the most extreme wellness esteem regarding totaled QoS is as yet a worry with the keeps expanding the number of accessible web services for an undertaking for reasonable conveyed applications. Besides, displaying QoS parameters of individual and composite web services and the piece procedure are the key focuses in web services determination advancement issue. To this end, this paper advances an enhancement calculation for web services determination inside a piece situation. We embrace Social Spider Algorithm (SSA) which, contrasted with PSO, has speedier merging and less execution time to locate the most enhanced wellness esteem.

A concise study is taken for dissecting the level of security petitioned web administrations. These days web administrations security is one of the significant requests on different associations [1]. [6] Proposed a secure hash message authentication code. A secure hash message authentication code to avoid certificate revocation list checking is proposed for vehicular ad hoc networks (VANETs). The group signature scheme is widely used in VANETs for secure communication, the existing systems based on group signature scheme provides verification delay in certificate revocation list checking. In order to overcome this delay this paper uses a Hash message authentication code (HMAC). It is used to avoid time consuming CRL checking and it also ensures the integrity of messages. The Hash message authentication code and digital signature algorithm are used to make it more secure. In this scheme the group private keys are distributed by the roadside units (RSUs) and it also manages the vehicles in a localized manner. Finally, cooperative message authentication is used among entities, in which each vehicle only needs to verify a small number of messages, thus greatly alleviating the authentication burden.For all instructive foundations, a MLSF is presented for giving multi-level security in the whole level of the structure. A decentralized execution for composite web administration guarantees the security for different requisitions and conjuring strategies [9]. WS-security, WS-strategy were utilized for security procurements to web provisions [10]. Security could be acquired by confirmation inside a particular web administration organization [11].

In the net services paradigm, AOWS may be a net service supported AOCE. A service is enriched with AN aspect-oriented description that supports automatic service discovery. This technique uses AN AO-Connector object that acts as an entranceway to a shopper. The connection receives shopper requests and transmits them to an applicable AOWS. Their connection doesn't outline an advancement structure and mistreatment their connection on an AOWS doesn't turn out a service [12-13]. [14] Discussed about a method, Sensor network consists of low cost battery powered nodes which is limited in power. Hence power efficient methods are needed for data gathering and aggregation in order to achieve prolonged network life. However, there are several energy efficient routing protocols in the literature; quiet of them are centralized approaches, that is low energy conservation. This paper presents a new energy efficient routing scheme for data gathering that combine the property of minimum spanning tree and shortest path tree-based on routing schemes. The efficient routing approach used here is Localized Power-Efficient Data Aggregation Protocols (L-PEDAPs) which is robust and localized. This is based on powerful localized structure, local minimum spanning tree (LMST). The actual routing tree is constructed over this topology. There is also a solution involved for route maintenance procedures that will be executed when a sensor node fails or a new node is added to the network.

In the field of internet technology, internet interact may be as a framework that provides transactional options to service composition. It proposes to compose net services in stratified architectures. Customary net services providing similar useful capabilities area unit bundled mistreatment intermediary pattern to make intermediary services that area unit later composed to make composite services by mistreatment WSTL to specify the execution sequence of specific intermediary service operations. So a composite service during this technique still involves the invocation of specific operations. Also, a composite doesn't exist at each level of composition. So their approach isn't stratified [15].

There is a unit many ways for net service composition. One of them is predicated on element model wherever the elements area unit designed from machine units and that they act like net services. The composite service captures all the operations provided by the subservience and it permits the operations to be invoked in a much-outlined advancement structure [16].

3. Webservice selection and composition

Web services mean getting to web technique effectively from anyplace at whenever through the web. Web services are basically usable under a system or under a system structure or working framework or any correspondence instrument or executing dialect. During the transaction SOAP messages are continuously sent among end-focuses showed with a Uris. To be effective in the e-business the web services ought to be suitable for secure correspondence. There is no SOAP detail having answers for security issues work now. A percentage of the procedures like SSL, IP-sec are giving transport securities.

Services are arranged under the design of SOA, means to give an inexactly coupled combination of services dwelling on various frameworks, composed utilizing diverse programming dialects and with other execution differences. Prevalently considered as the building piece of SOA, Web services (WS) are self-depicting and stage autonomous applications that can be conjured over the Web. Encouraging the get together of services to shape composite services is an essential usefulness in SOA. The issue of naturally collecting WSs to shape arrangements that improve given client inclinations is regularly alluded to as the robotized Web benefit creation issue. It includes two noteworthy difficulties each with its own orderly issues:

1) Automatically develop the control stream of the structure and.
2) Appropriately purpose the information heterogeneity between WSs taking an interest in the piece.

We utilize a various leveled optimal theoretic arranging system for making Web services, called optimal web service selection. Compared with established AI organizers, optimal theoretic arranging can show the vulnerability intrinsic in WSs and give a cost-based process improvement. SSA optimization approach utilizes symbolic arranging schemes that work specifically on First Come First Serve based portrayals of the state space to get the pieces. Thus, it bolsters a computerized elicitation of the relating arranging space from WS depictions and produces a reduced area portrayal in contrast with traditional AI organizers. Also, it handles the adaptability issue by abusing the common order found in forms.

4. QoS model of web service

The same kind of web services is grouped together to create a web service composition where they are similar in functionalities to increase the exposure [8], [9]. These kinds of similar web services are different in terms of their QoS parameters when selecting for composition [10]. In order to increase the efficacy of the web service composition, the web services are selected in terms of QoS values such as time, throughput, availability and probability of success.

The various parameters help to determine the quality of the web services and the total quality of the composite-service. Less execution time, more availability and less distance based web services are selected for composition. To ensure the QoS value the parameter values are classified into two categories such as positive and negative parameters. Hence the parameters are normalized using the following equation (1) and equation (2) for classifying negative and positive parameters respectively such as:

$$p(1 < P < N)$$

Where N represents the parameter number, $QoS_P(S_{ij})$ denotes the p^{th} QoS parameter of the web service S_{ij}, QoS_P^{min} and QoS_P^{max} denote the minimum and maximum values of the p^{th} QoS parameter.

$$QoS_P(S_{ij}) = \begin{cases} \frac{QoS_P^{max} - QoS_P(S_{ij})}{QoS_P^{max} - QoS_P^{min}}, QoS_P^{max} - QoS_P^{min} \neq 1 \\ 1, QoS_P^{max} - QoS_P^{min} = 1 \end{cases} \quad (1)$$

$$QoS_P(S_{ij}) = \begin{cases} \frac{QoS_P(S_{ij}) - QoS_P^{min}}{QoS_P^{max} - QoS_P^{min}}, QoS_P^{max} - QoS_P^{min} \neq 0 \\ 1, QoS_P^{max} - QoS_P^{min} = 1 \end{cases} \quad (2)$$

In accordance to the above equations, the QoS of webservice is demarcated as:

The set of all QoS parameters are defined as a vector having full of webservices S_{ij} is written as

$$QoS(S_{ij}) = QoS_{success}(S_{ij}), QoS_{time}(S_{ij}), QoS_{throughput}(S_{ij})$$

$$, QoS_{distance}(S_{ij}), QoS_{availability}(S_{ij})]$$

Where $QoS_i(S_{ij})$ are the normalization values of probability of success and other parameters of webservice S_{ij}.

Web services are shared into conformations to provide multifaceted responsibilities, which cannot be delivered by a sole web service. The composition is organized as an order, synchronized or circle model. This study deliberates only the order arrangement; as other arrangements could be easily distorted to it.Web service composition emphases essentially on the choice of most suitable fissionable web services from dissimilar web services groups to carry high-quality purposes sustaining user's necessities. As shown by Figure-1, each block signifies web services group comprising functionally related web services and the arrows refer to dissimilar composition paths. Therefore, discover the optimal path from all probable groupings is an optimization problem with the state of maximizing the utility value of the composition Q (Cs).

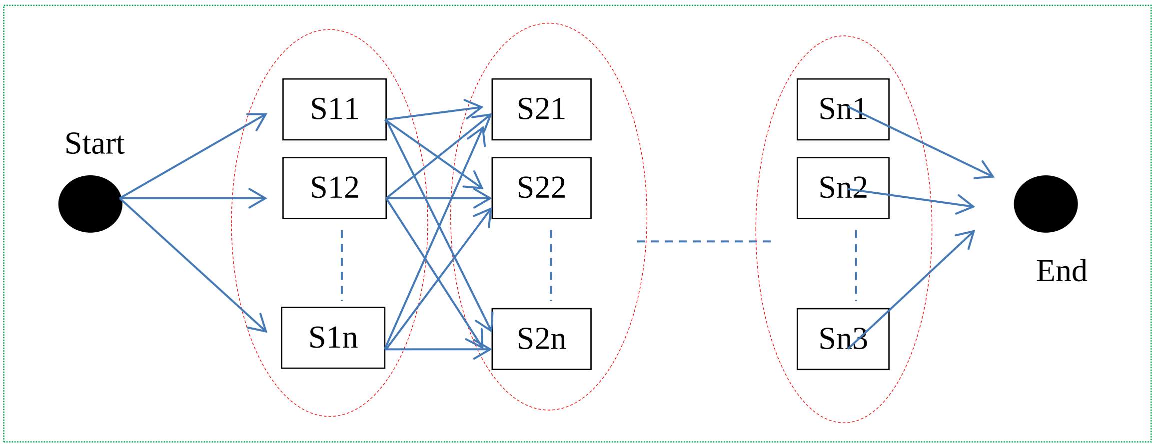

Fig. 1: Functionality of Web Service Selection.

Some of the definitions can help to review the modules of QoS of each web services going to be composite. Also, the set of all constraints used to select the web services are given in Table-1.

Definitions:

Community: $S_i = \{S_{i1}, S_{i2}, \dots, S_{ij}\} \forall j > 1$, *this group has similar functionalities with various QoS values.*

Composition: $Cs = \{S_1, \dots, S_n\}$ *where n represents the web service number from the* i^{th} *community.*

QoS Vector:

$$(Cs) = \begin{bmatrix} QoS_{success}(S_{ij}), QoS_{time}(S_{ij}), QoS_{throughput}(S_{ij}) \\ , QoS_{distance}(S_{ij}), QoS_{availability}(S_{ij}) \end{bmatrix}$$

Are set of QoS parameters

The quality of service function (f) and the parameters are assumed as real numbers to calculate and assign a ranking value for web services. The web services are selected according to the highest rank values of the web services. The web service selection procedure is applied based on certain constraints such as:

$$f(Cs) = \sum_{P=1}^{N} w_k QoS_P(Cs), \text{where } \sum_{P=1}^{N} w_k = 1, 0 \leq w_k \leq 1$$

5. Optimal webservice selection using SSA algorithm

The search space of the optimization problem is defined as a hyper-dimensional spider space. All the positions of the spider are considered as the individual solution where all the solutions must have a position in the search space. The vibrations generated by the spiders are transmitted through the web. Each spider has a position represents the quality of the solution related to the objective function describes the potential of finding the food source at the particular position. Each spider moves in the web towards food and it is assumed that each spider moves freely. The spider cannot move out from the web and it represents an infeasible solution. At each move, the spider generates vibration which is broadcasted over the web. A vibration represents the information of a spider which is transmitted to other spiders. The information is generated and received is measured in accordance with the vibration.

Spider

This SSA algorithm uses the spiders are the agents for the optimization problem. In the initial stage of the algorithm execution, a defined number of spiders are created on the web space. The entire spider has a memory and other information which are given below.

Table 1: Constraints of Webservice Selection for Composition

Probability of Success	Execution Time	Availability	Throughput
$\prod_{s\in Cs} QoS_{success}(S)$	$\sum_{s\in Cs} QoS_{time}(S)$	$\sum_{s\in Cs} QoS_{availability}(S)$	$\min_{s\in Cs}\begin{Bmatrix} QoS_{throughput}(S), \dots, \\ QoS_{throughput}(S) \end{Bmatrix}$

The position of s on the web.

The fitness of the current position of s.

The target vibration of s in the previous iteration.

The number of iterations since s has last changed its target vibration.

The movement that s performed in the previous iteration.

The dimension mask-1 that s employed to guide movement in the previous iteration.

The first two types of information describe the individual situation of s, while all others are involved in directing s to new positions.

In this paper, the SSA algorithm is used for selecting the web services for composition. Using the optimization strategies and the fitness function values over the hyper-dimensional spider space best position based spiders is selected. The web services are considered as the spiders and the fitness function values determine the best one. Each feasible solution denotes the web service to be selected as a best case. The fitness value is calculated using the function f(Cs) based on the target vibration V_{tar}, the total number of iterations. The movement is calculated by changing the position at each time of iteration from the previous position to next position and it is calculated by

$$|Cs(t) - Cs(t-1)|$$

Where t represents the time at spider moves. The dimension mask is represented as M; the length of the binary vector is L, which is the total dimension of the optimization problem. The optimal solution represented by f (Cs) giving maximum QoS values and it is the fitness value written as:

$$f(Cs) = \text{maxmimum}(QoS)$$

$$\text{Intensity} = \frac{1}{QoS^{max} - f(Cs)}$$

The entire SSA algorithm is carried out into three different stages such as initialization, iterative process, and finalization. Initially, the main objectives and the related parameters are initialized. Then the set of all populations are generated randomly and search the optimal spiders over the space within the iteration. Each time the obtained solution is compared with the objective function value and the fitness values to check the best spider. The entire process is terminated whenever the optimal spider is obtained. To do the above process, the following parameters are used in the program.

Pop: Population
Vbest: Best Vibration
Vtar: Target Vibration
R: Total number of iteration

The fitness value is evaluated for all the spiders taken in the process (Pop) and their vibrations are broadcasted in the search space. Once the iteration is over R is set to "0", else it is incremented. The entire procedure of SSA given below in the form of pseudocode, where it can be implemented in any computer programming language and the performance is verified.

```
PseudoCode_SSA ( )
{
// initialization: P= number of populations, pop= { S1, S2,...,SM}, Vtar=0, Cs={ }
for i=1 to P
        Vtar(i) = 0
end for
 repeat {
    for i=1 to M
        fit(Si)  = calculate the fitness value
        Cs(i) =bestfit(Si)
        V (i) = vibration(Cs(Si))
    end for
for i=1 to M
```

$$Intensity\ (i) = \frac{1}{QoS_{(Si)}{}^{max} - f(Cs(Si))}$$

V(Si) = V(Intensity(i))

Vbest (i)= strongest(V(Si))

if (Vbest(i) ≥) Vtar then

Vtar = Vbest

end if

R=R+1

Generate a random number r from [0,1].

if $r > P_c^R$ then

update the dimension mask: $Cs_i^{fo} = \begin{cases} Cs_i^{tar} M_i = 0 \\ Cs_i^{r} M_i = 1 \end{cases}$

End if

Generate new population: $Cs(t+1) = Cs(t) + (Cs(t) - Cs(t-1)) \times r + (Cs^{f0} - Cs) \cdot R$

Address any violated constraint using:

$$Cs_i(t+1) = \begin{cases} (\bar{x}_i - Cs(t)_i) \times r\,if\,Cs_i(t+1) > \bar{x}_i \\ Cs_i(t) - \underline{x_i}) \times r\,if\,Cs_i(t+1) < \underline{x_i} \end{cases}$$

end for

end until

Output the best solution found.

6. Implementation

The necessity of analyzation is especially acute for CS mostly if they are to be created from pre-existing services using automatic algorithms. The main goal is to check the QoS factors for web service composition combined with various QoS parameters in terms of desired behavior. Nowadays the QOS of web service composition is improved by deploying service with SOA framework. The following table shows the analysis report of the web service composition.

Table 1: Time Factor for WSC

Number Of Web Services	Number Of Servers	Execution Time	Response Time
5	1	1.5	4.6
10	2	1.9	5.98
15	3	2.5	8.12
20	4	3.7	11.34
25	5	5.5	15.65

From Table-1, it is clear that the numbers of web services are changed and deployed in single or multiple servers and the execution time, response time is computed. For 5 number of web-services deployed in single server the execution time of the service is 1.5sec and response time is 4.6 sec, for 10 number of web-services deployed in two servers the execution time of the service is 1.9 seconds and response time is 5.98 seconds, for 15 number of webservices deployed in three servers the execution time of the service is 2.5 seconds and response time is 8.12 seconds, for 20 number of webservices deployed in four servers the execution time of the service is 3.7 seconds and response time is 11.34 seconds and for 25 number of webservices deployed in five servers the execution time of the service is 5.5 seconds and response time is 15.65 seconds.

The graphical representation of Table-1 is depicted very clearly in Figure-2. From the experimented results it is concluded that the response time, execution time of the web services depend on the services, deployed in the number of servers and the distance among the REQ place and Service place.

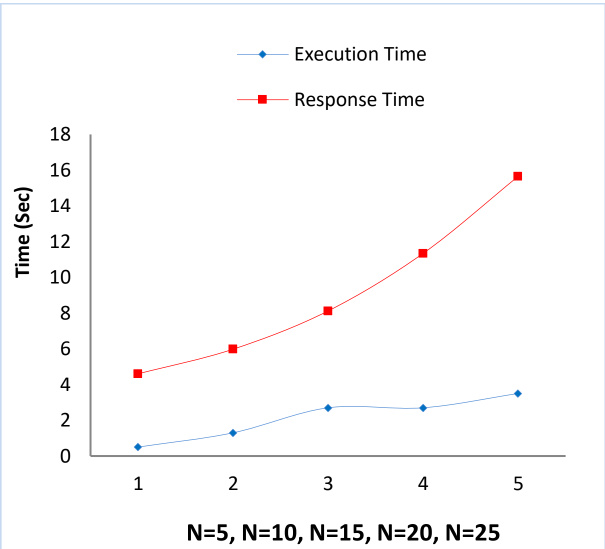

Fig. 2: Execution Time vs. Response Time for Various Web Service Composition.

In order to evaluate the performance of the proposed approach, the execution is carried out before and after deploying the proposed approach in the experimental setup. In order to evaluate, the number of web services are changed as (1, 5, 10, 15 and 20) and executed. The results are verified each time the number of web services is changed. In this paper the response time for various numbers of web services, composition cost and availability of web services during web service composition. Without deploying the proposed approach the experiment is carried out, the above said parameters are calculated and it is treated as the existing approach. Then the proposed approach is deployed and the experiment is carried out.The obtained results in terms of response time, composition cost and availability for a different number of web services.

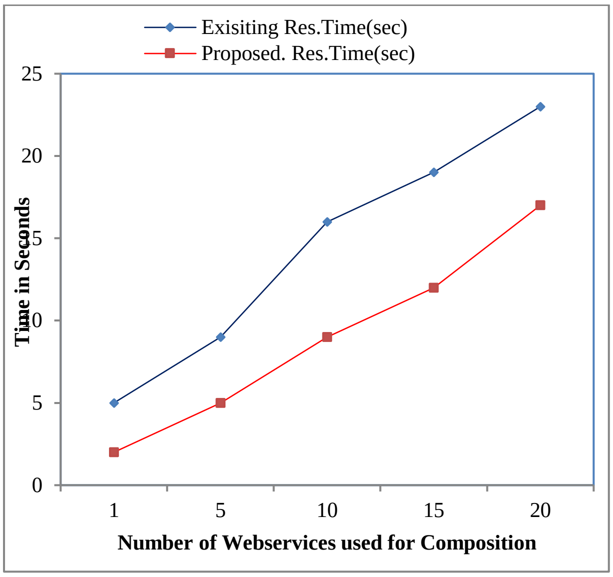

Fig. 3: Number of Web Services vs. Composition Time.

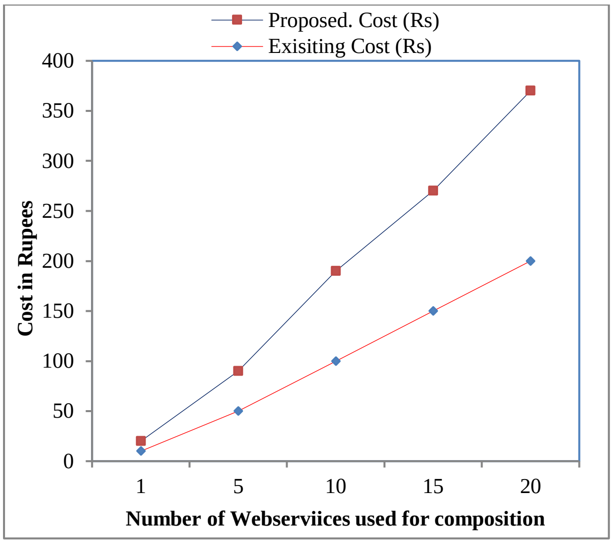

Fig. 4: Number of Web Services vs. Composition Cost.

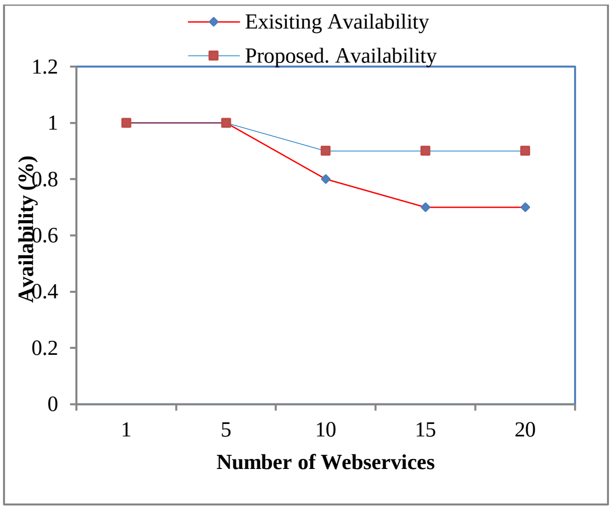

Fig. 5: Number of Web Services vs. Availability.

The obtained results are given in Figure -2 to Figure-5. Figure-2 illustrates the calculated execution time and response time taken for the number of web services deployed in the cloud server. Figure-3 shows the response time taken for composition in accordance with the number of services deployed.Figure-4 shows the computation cost taken for composition. Figure-5 shows the availability of the web services during composition.

In order to evaluate the performance of the proposed approach the response time, the total time taken for complete compilation and execution and throughput obtained by various existing algorithms is compared with the proposed approach. FCFS, RR, Priority Allocation, Multi-Queue are the existing methods compared with the proposed SS Algorithm. The comparison of results is given in the following figure 6.

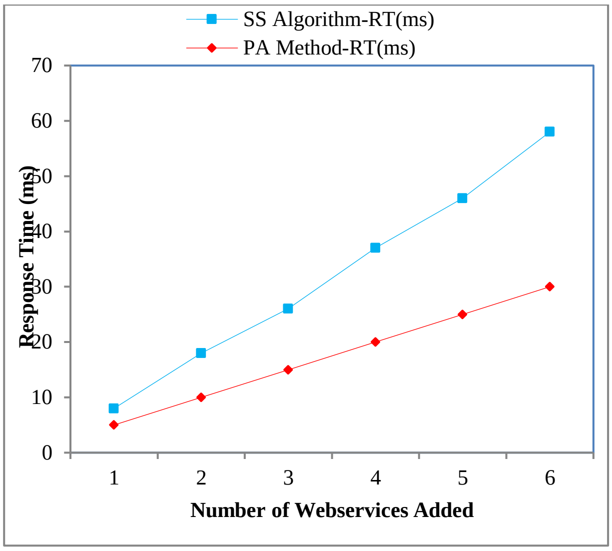

Fig. 6: Number of Web Service vs. Response Time.

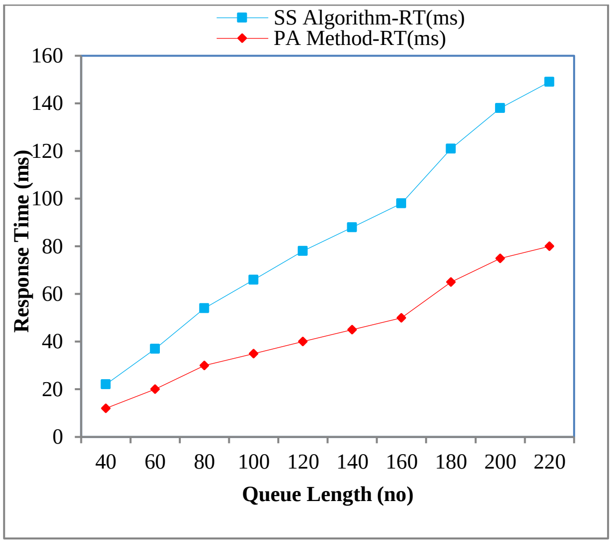

Fig. 7: Queue Length vs. Response Time.

The graphical analysis of Composite Web services (1 to M) being added and sent is shown in the Figure-6. when the queue length exceeding its capacity, the requests of web service-1 and web service-2 that have arrived and awaited for execution is sent is shown in Figure-7.

From the above Figure-9, according to the input taken, earlier services has more burst time than later services in the queue so FCFS scheduling gave lowest waiting time. The percentage of waiting time obtained by the proposed approach is very less comparing with the other existing algorithms is illustrated in Figure-9. Since SSA optimizes the entire web service queue and individual web services in terms of their behavior the waiting time is less.

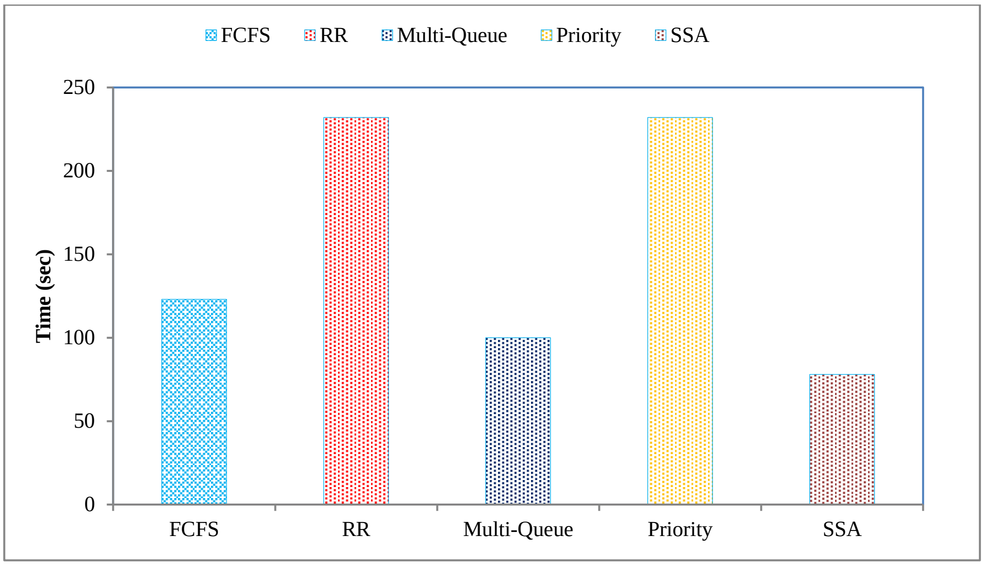

Fig. 8: Comparison of Waiting Time.

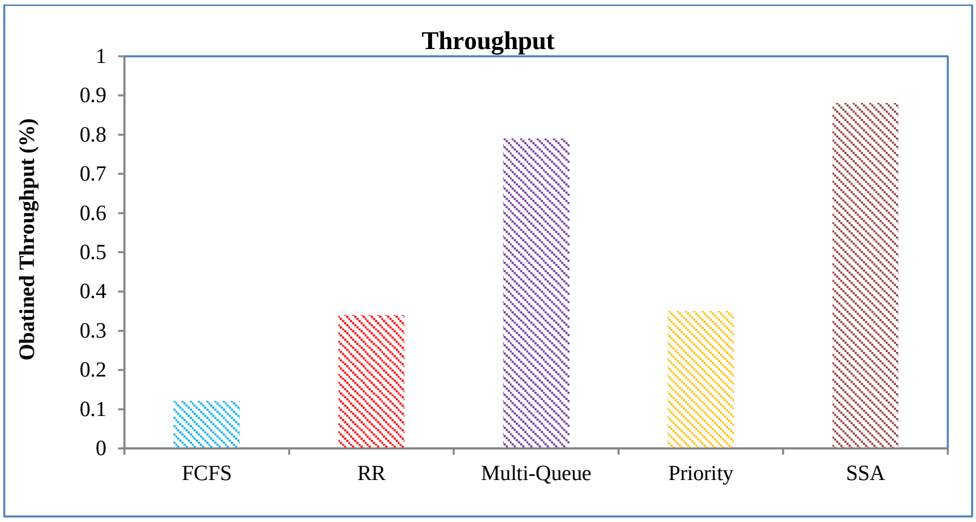

Fig. 9: Comparison of Throughput.

Figure-10 shows the throughput comparison among the five algorithms including the proposed approach. Each individual method obtained an amount of throughput for a fixed number of web services deployed in the cloud and experimented. From the figure, it is noticed that SSA is better than the other existing approaches and it is suitable for web service composition. Execution, time, response time, cost and the success rate are increased gradually the increased number of web services. But the availability of the web services cannot be determined since the web services are busy with other processes due to the cloud is parallel and distributed environment. From the results, it is identified that the proposed approach is better than the existing approach and it is proved.

7. Conclusion

The web service composition with a higher level of quality has become the major demand of the diverse organizations. This paper proposes SSA for SOA environment that provides multiple quality web services such as Request service, payment service, support service, personalized service for the customer, vendor, and management, etc. This framework is designed mainly to provide the QoS increased composite web services and well suited for the E-Commerce to handle the business-related services. This model can be extended to provide the composite web services for several e-businesses as an integrated web services.

8. Future work

In future, the composite web service could be applied in a cloud environment and verify the performance of the SSA.

References

[1] J. G. R. Sathiaseelan, S. Albert Rabara, J. Ronald Martin, "MLSF: A Framework for Multi-Level Secure Composite Web Services" Advances in Information Sciences and Service Sciences Volume 2, Number 4, December 2010.

[2] Hai Y, Yan L, Lu G. Dynamic prediction QoS-Aware web service composition model. JDCTA 2012; 6:490-498. https://doi.org/10.4156/jdcta.vol6.issue12.59.

[3] Zeng L, Benatallah B, Dumas M, Kalagnanam J, Sheng QZ. Quality driven web services composition. WWW 2003:411-421. https://doi.org/10.1145/775152.775211.

[4] Canfora G, Di Penta M, Esposito R, Villani ML. A lightweight approach for QoS-aware service composition. ICSOC 2004.

[5] Yu T, Zhang Y, Lin KJ. Efficient algorithms for Web services selection with end-to-end QoS constraints. TWEB 2007.

[6] Christo Ananth, M.Danya Priyadharshini, "A Secure Hash Message Authentication Code to avoid Certificate Revocation list Checking in Vehicular Adhoc networks", International Journal of Applied Engineering Research (IJAER), Volume 10, Special Issue 2, 2015,(1250-1254).

[7] He J, Chen L, Wang X, Li Y. Web Service Composition Optimization Based on Improved Artificial Bee Colony Algorithm. JNW 2013; 8(9):2143-2149. https://doi.org/10.4304/jnw.8.9.2143-2149.

[8] Zhang T. QoS-aware Web Service Selection based on Particle Swarm Optimization. JNW 2013; 9(3):565-570.

[9] Joachim Biskup1, Barbara Carminati2, Elena Ferrari2, Frank M¨uller1, Sandra Wortmann1, "Towards Secure Execution Orders for Composite Web Services", 2007 IEEE International Conference on Web Services (ICWS 2007) 0-7695-2924-0/07 $25.00 © 2007.

[10] Anis Charfi1 and Mira Mezini "Using Aspects for Security Engineering of Web Service Compositions", Proceedings of the IEEE International Conference on Web Services (ICWS'05) 0-7695-2409-5/05 $20.00 IEEE.

[11] Mohsen Rouached and Claude Godart "Securing Web Service Compositions: Formalizing Authorization policies using Event Calculus" LORIA-INRIA-UMR 7503 BP 239, F-54506 Vandoeuvre-les-Nancy Cedex, France.

[12] J. Grundy, T. Panas, S. Singh, H. Stockle, "An approach to developing web services with aspect-oriented component engineering", In Proceedings of the 2nd Nordic Conference on Web Services, 2003.

[13] J. Hosking S. Singh, J. Grundy, J. Sun, "An architecture for developing aspect-oriented web services", In Proceedings of European Conference on Web Services, Vaxjo, Sweden, 2005.

[14] Christo Ananth, S.Mathu Muhila, N.Priyadharshini, G.Sudha, P.Venkateswari, H.Vishali, "A New Energy Efficient Routing Scheme for Data Gathering ",International Journal Of Advanced Research Trends In Engineering And Technology (IJARTET), Vol. 2, Issue 10, October 2015), pp: 1-4

[15] P. F. Pires, "Web transact: A framework for specifying and coordinating reliable web services compositions", Technical report, Federal University of Rio De Janeiro, 2002.

[16] Kung-Kiu Lau, Cuong Tran, "Composite Web Services", Emerging Web Services Technology, Springer Link, vol. 2, 2008.