ISSN 2278-3091
Volume 9 No.2, March -April 2020
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://www.warse.org/IJATCSE/static/pdf/file/ijatcse24922020.pdf
https://doi.org/10.30534/ijatcse/2020/24922020

Business Policy Violation in Web Service Integration of QoS-Aware Web Service using Hybrid Genetic Algorithms for Security in SOA

G. Amirthayogam[1], C. Anbu Ananth[2]
[1]Research Scholar, Dept. of Computer Science and Engineering, Annamalai University, TamilNadu, India
amir.yogam@gmail.com
[2]Associate professor, Dept. of Computer Science and Engineering, Annamalai University, TamilNadu, India
anbu_ananth2006@yahoo.com

ABSTRACT

The Web services advanced as a pretty good paradigm for publication, service consuming and discovering. The aspects of Service-Oriented Architecture (SOA) are broadly engaged in e-government, e-business, process control, finance, multimedia services, automotive schemes, and many other domains. A Quality-of-Service (QoS) remains as useful for working towards relating to the non-functional principles to the Web services, and then it is employed by way of a significant distinguishing point to the various Web services. Through this incidence, the Web services on the web, the QoS management becomes furthermore essential. Though confident of them are suited a standard, no particular acceptable performance standard maintained by the frameworks of the web services. Additionally, the enforcement of the performance of the existing scheme is generally concerned through service provider besides hardly concerned by the network and the consumer environment. Because of this reason, hybrid Genetic Algorithms (HGAs) used to solve the problems based on QoS service composition. With the help of the proposed method an Adaptive and Secure Service Composition engine (ASSC), proceeds a benefit to an effective heuristic way of the Algorithm to make optimum service compositions in the SOA aspects

Key words: Quality of Service, Change Management Framework, Composite Web Services, Business Policy Violation Web Service, Web Service Integration, Optimization, Genetic Algorithms,

1. INTRODUCTION

The concept of Web Service [1] based on the standards, language- uncertain entity software that accepts mainly structured requests from the other entities software on the remote machines through transport neutral and vendor oriented Communication protocol that making the positive application responses. The standard Web service structure has only two participants. The service consumer and then the service producer. The role of the services provider is to interface and execution of service, whereas the requester practices on the web service [2].

Though the attainments of web service pattern in the existent world are found to be great, it needs to address specific critical problem intricate in the model. One such issue is to manage the Quality of Service in the merged web services; the delivered compound services do not meet the customer's functional requirements, and then also satisfy the needs of QoS [3], which helps to fulfill their hopes and attains their fulfillment (Figure 1).

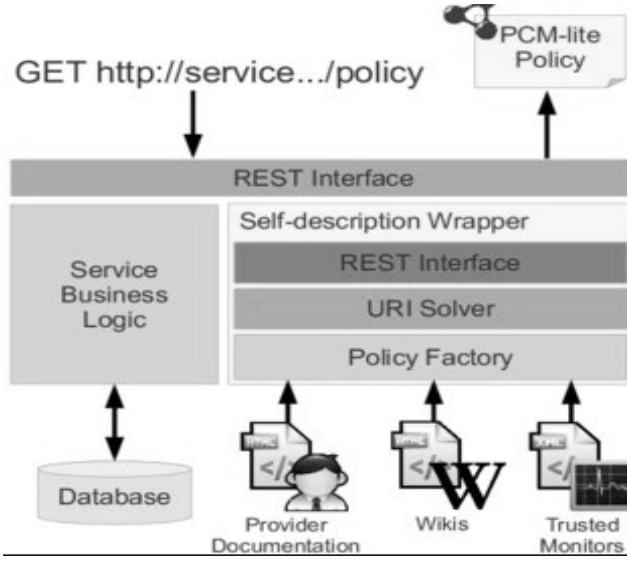

Figure 1: Web -Business Policy Process

Due to the speedy growth of web-based services in enterprise and the rising attractiveness of the micro-services and cloud computing concepts, the SOA becomes an important software pattern for developing the application and distributed services. Allowing for such a speedy growth of such technology (cloud computing), the aspects of SOA [4] has

exposed to require a considerable share in the market concerning the world of enterprise software [5]. Integration between enterprise applications is one of the essential, exciting issues in software engineering discipline for the past decade. The SOA defines a style of architecture, which allows and streamlines the interconnection among the complex applications across or inside the boundaries of the organization.

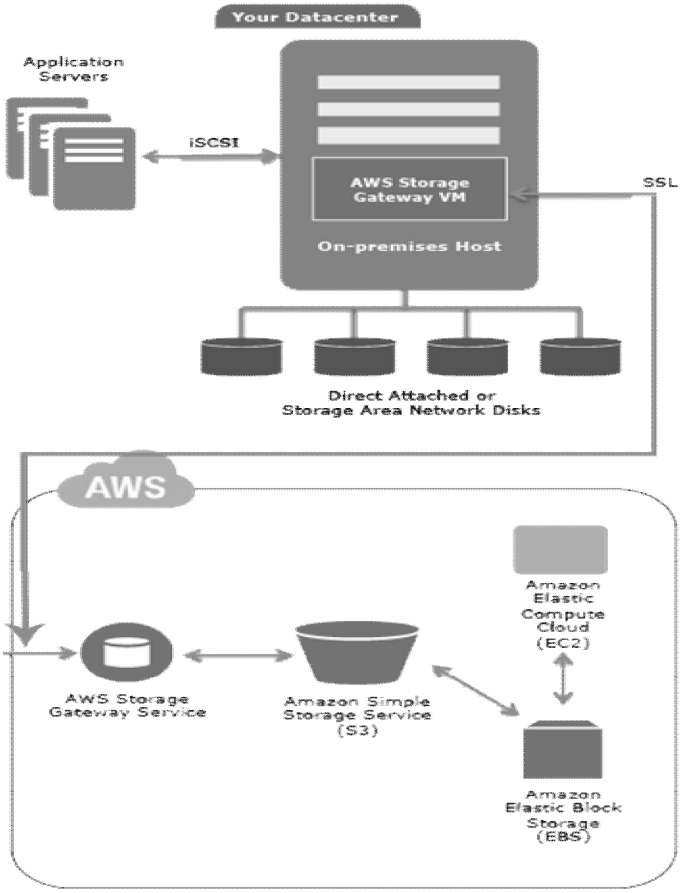

Figure 2: Web Service for Personal Use

The security aspect is the primary concern in SOA (Figure 2). Though a substantial amount of measure prepared to analyze and advance the web services security features, still, some major of the projected results address the experiments carried out insecurity in the client-server or the single services schemes. In detail, they get failed to solve the nature of multi-hop service requests that occurs in the SOA.

The current solutions for security issues allow the service to govern the local resource access or to the communication among the instant service which it's been interacting with. The SOA allows new challenges related to security, which do not exist in the old client-server design. These challenges are unique, and the nature of multi-hop produces it in SOA, which is composed of a sequence of numerous services from autonomous service providers. On the additional side, since the SOA designs are open, the direct accessibility is a challenging task and also the adjustment of the services to third-party in SOA. Therefore, certain services can be down the appeal chain might access a malevolent service deprive of being detected (Figure 3).

Due to the below examination done, to enhance the Web services QoS management; it needs a capable QoS Web service assessment mechanism, precise QoS Web service expectation. Methods and robust QoS-alert fault-tolerance approach for Web services. The first method is besieged at evaluating the Web service by suggesting a Web service distributed in the QoS evaluation structure. Next, this research examines how to practice a Hybrid Genetic Algorithm (HGA) [6] for addressing QoS-based service composition difficulties.

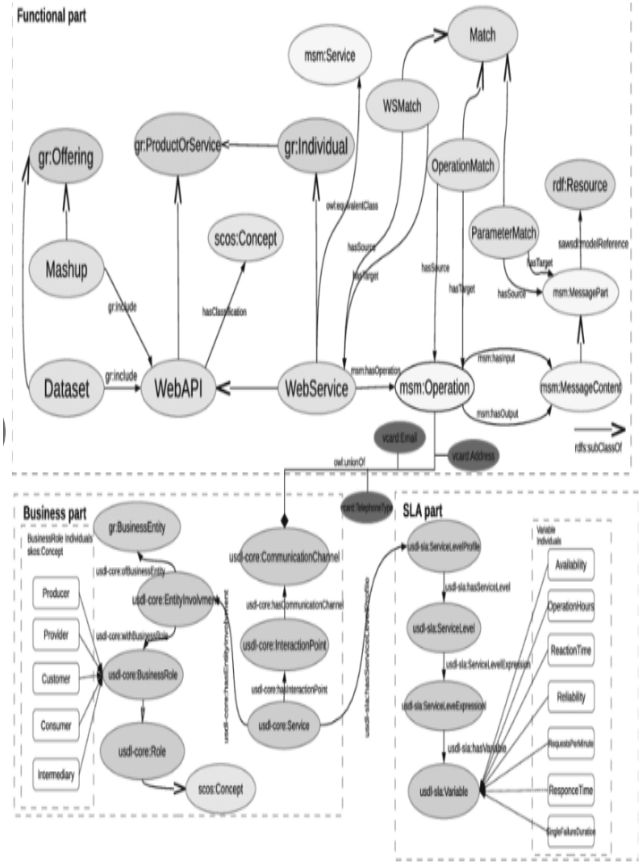

Figure 3: A Network Model for Web Services

2. RELATED WORKS

A web service scheme is empowered by a different type of XML-centered web service principles [7]. The essential standards comprise Web Service Description Language (WSDL), then the Universal Description Discovery and Integration standard (UDDI) [8], and the other one Simple Object Access Protocol (SOAP) [9]. These few principles planned to maintain the actions of discovery, service description, and invocation. Also, in these standards, there

remain few complementary standards, such as WS-Security, WS-Policy, and WS-Trust. Altogether these standards, added with the core standards, encompass the lowest infrastructure requirements for web service pattern, which is known as web services technology stack [10].

Web services can be developed based on various application modules types. For an instant, a web service might be:

- An independent business task, like a withdrawal of funds or depositing a funds service;
- A complete business method that has automated buying of the office supplies; or
- A service-permitted resource, like storage resources, computation-resources, and network-resources.

Muller suggested overall monitoring scheme and reference idea for policy violation to composing service. The QoS constraints typically comprised of the Service Level Contracts of every service. These constraints might also be beneficial for discovering the policy destructions of the comprising services. This might inform the client by the damage and the reasons for the loss in a natural way of understanding.

Li had proposed a forceful plan and charging the control (PCC) structure [11]. This structure helps to switch and optimize the network practice and provides the measure resources and the network information to the subscriber. Moreover, it works conferring to customer's real-time outline and knowledge of use subscription and evading or diminishes the operator's network congestion. It assists in adjusting the existing investment of operates and possible network. The writers established the value of the key to optimizing that is constructed on the convergent capability of the score and charging utilities on facts of service movements and subscriber profile account and life-cycle.

The business procedure employed as a composition of web service, which accomplishes the set of facilities to attain the business goal. It helps to manage the Business Process Management (BPM) [12] approach that contains techniques, methods, and tools for supporting the enactment, design, analysis, and management of the business progressions. BPM's life-cycle comprises of 4 phases: designing, setup, representation, and analysis. The architecture is the first and foremost stage in business process development between these stages as it offers assistance in developing the business process model, organizing tools, finding new opportunities and forecasting threats. Process analysts sketch business procedures using graphical modeling tools, such as ARIS and MID Innovator, during the design phase.

Designing a business process from scratch, however, is always a time-consuming and challenging task. System analysts need resources and frameworks to promote the creation of business processes. To tackle this need, some solutions suggested. They consider utilizing comparative models, comparing business processes to look for similarities, building a query language/finding process models from the event log.

3. A PROPOSED HYBRID GENETIC METHOD FOR WEB SERVICE SELECTION PROBLEM

Genetic Algorithms (GAs) are quest strategies developed by John Holland and focused on the theory of natural selection, a biological process in which stronger individuals in a competitive environment are likely to be the winners. HGA's are usually used to produce acceptable solutions to optimization problems when traditional optimization methods such as gradient descent methods or heuristic-based methods cannot effectively traverse their search space [7]. A GA works on a person community. Every person in the Population denotes a possible outcome for the problem of optimization. Individuals assessed according to their fitness. The fitness level shows how well a person in the community can overcome the optimization issue. A GA begins typically with a population generated at random. Then there is an evolution of the people in the manner of the transition from one society to the next through the application of the genetic operators: mutation, selection, and crossover. The most qualified people will be selected through the selection process to go to the next generation. Crossover exchanges the two individuals' genetic material and creates two new individuals. An individual's genetic content is randomly affected by the mutation. The application of the genetic operators to the population individuals' proceeds until a suitable solution to the problem of optimization discovered [13]. The answer typically found on a predefined stop limit, i.e., a certain number of generations met or the sum of individual variation between different ages decided upon or a predefined health rating. The outcome may not be an optimal solution, but it may be necessary to calibrate the Algorithm so that it often delivers a viable answer that satisfies those requirements.

The pseudo-code of a classical genetic algorithm

a) Arbitrarily create an initial population =Pop Size individuals, Population
b) **Each** Existing Web Service, W of Wx Do
    a. Optimize it using the local optimizer
c) **End**
    a. Terminating **While** the condition is not right to **Do**
    b. **Select-Fit** individuals for reproduction
    c. Apply the knowledge-based crossover operator to generate offspring
    d. Apply the mutation operator to offspring
    e. Optimize each individual using the local optimizer
d) **End**

3.1. Service-based Algorithm

In this Algorithm, we target to find services related to the function currently being used by a user. In the service-user matrix, we appeal the top-N CF algorithm centered on objects. They measure correlations between the systems actually in use and other resources utilizing VSM. Instead, we arrange the funds in a standard descending order. We are gathering the l-top utilities towards suggesting them to the customer. The virtual suggested service encryption based on top-N CF based on items exists defined in Procedure 1. The Algorithm's main stage is to find the resemblance among a service si and additional service sx (line 3 of Procedure 1). We appeal the Vector Space Model (VSM) to calculate that similarity. [14] Implemented VSM in the first position. The calculation of the comparison among two specific documents developed. Now a *k* dimensional space shows papers, where *k* is the sum of altered words. Every article remains illustrated with the elements k as a vector. Every aspect of a vector document refers to the concept that occurs in the text. The weight of the corresponding name is the value of a vector variable. Inverse Document Frequency (IDF) and Term Frequency (TF) define this weightiness. The angle with cosine value generated through two related directions determines the resemblance between the two texts [15].

Algorithm: Reference for Package
Input: S_X: Presently Used Package
Output: A suggested list L

a) S = Set of Packages;
b) **Every** Package in S **Do**
c) Compute S_i, S_x
d) End
e) Sort $S_i \in S$ in descending order
f) Select Top-l services = Recommendation

The weight of a user U_j w.r.t a service S_i, denoted by $W_{i,j}$, i = 1..m, j = 1..n, computed by TF-IDF is given by Equation 1.

$$\begin{aligned} w_{i,j} &= tf_{i,j} \times idf_{j,S} \\ &= \frac{a_{i,j}}{\sum_{k=1}^{n} a_{i,k}} \times \log \frac{m}{|s_t \in S : a_{t,j} > 0|} \end{aligned} \quad (1)$$

3.2. User-based algorithm

Motivated through the idea that users with related attention appear towards pick related substances, we purpose at identifying users with a similar focus in this method, i.e., they used associated packages. We instead choose the programs that were most commonly used by the most critical users and that were not used by the active user to make recommendations. Contrary to the service-based Algorithm, we find each customer to be a text and each service to be a word in this Algorithm. We use the VSM to measure user-likeness. TF-IDF [16] [17] is used for weighting vector components. In concrete terms, Equation 2 measures the weight of a device S_i, which was used by a customer U_j.

$$\begin{aligned} w_{i,j} &= tf_{i,j} \times idf_{i,U} \\ &= \frac{a_{i,j}}{\sum_{k=1}^{m} a_{k,j}} \times \log \frac{n}{|u_t \in U : a} \end{aligned} \quad (2)$$

Algorithm of User Recommendation

Input :UX active-user
Output : A recommended list = L

a) U= Number of Users;
b) **For Each** user U_j in U **Do**
c) Calculate U_j ,U_x
d) **End**
e) Sort $U_j \in U$ in descending order
f) Select Top K Users = The sorted list of $U_j \in U$;
g) **Each** of K Nominated User
h) select the T-most-frequently-used facilities from L = K×T services

3.3. Service-user combination algorithm

In this section, we organize a mixture of the algorithms depends upon service and also on the user. We often make references dependent on the convention data of the specific users, though, rather than choosing the services mostly used by the related users, and we quantify the overlap between services used by those users. Through integrating such procedures, we aim to improve the suggestion efficiency and prevent the user-based Algorithm's possible missed problem.

Algorithm of User-Service Recommendation

Input: Present user U_x, presently used service S_y
Output: A suggested list of L services

a) U = set of users;
b) **For Each** user U_j in U do
c) Compute the similarity between uj and ux;
d) **End**
e) Sort $U_j \in U$ in descending order of similarity values;
f) Select top k users from the sorted list of $U_j \in U$;
g) A[M×K] = usage data of individual users;
h) S = Set of services;
i) **For Each** service S_i in S do
j) Compute the similarity between S_i and S_y based on the new usage matrix A[M×K];
k) **End**
l) Sort $S_i \in S$ in descending order of similarity values;
m) Select Top-l services for a recommendation;

3.4. Performance Evaluation

To check their efficiency, we performed the following two forms of experiments:

1. To thoroughly test the HGA's reliability and scalability, we examined the HGA's on many issues, including different problem sizes with varying levels of inter-service dependency and conflict constraints.

2. A drawback of the tests described above is that the optimality of the solutions sought by the three HGA's not understood by considering only the three HGA's themselves. Therefore, we have contrasted the efficiency of the three HGA's with the Integer Programming (IP)-based process, the solutions of which are often optimal by the system definition.

However, because the IP [18] [19] based can only address the issues with simple workflow patterns, the sequential pattern and therefore the parallel pattern, and since the IP-based method and consequently the HGA's use different QoS aggregation models when involving the parallel mode, which could lead to different QoS results between the IP-based method and therefore the HGA's under the identical solution, we only used the test problems with the sequential pattern, which may always guarantee that the QoS results of the IP-based method and therefore the three HGA's are the identical under the identical solution.

Figures 4 and 5 show how the HGA's average calculation times reconditioned when the Number of abstract web services enhanced from 10 to 100.

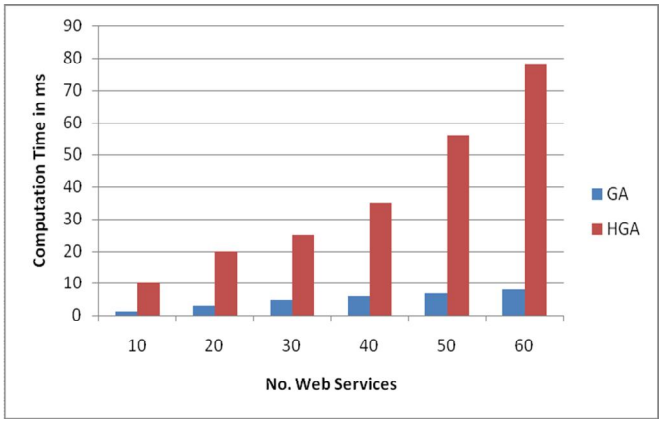

Figure 4: The Impact of the No. Web Services vs. On-Time of Measurement

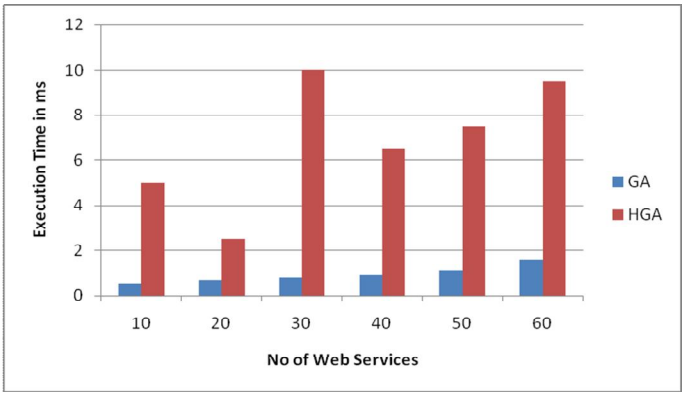

Figure 5: The impact of the No. of web services Vs. Processing period

5. EXPERIMENTAL SETUP

The experimental environment contained the following components:

A. Composite Web Service —Within Microsoft BizTalk, a journey was planning hybrid cloud interface was developed and deployed on a BizTalk server (2 Duo 2.33GHz CPU, a 3.95 GB RAM).

QoS-based web service selection tool - this method created in Microsoft Visual C 2005. The updated HGA's are used by the device to choose Web service according to the QoS desires of a consumer.

B. Set of Concrete Web Services — for each job involved in the travel planning composite web service, we have built 10 candidates web services. Such web services have different time and price of implementation for applicants. We were developed and deployed on a web server (2 Duo 2.33 GHz CPU, a 1.95 GB RAM) in Microsoft Visual C 2005.

C. Web-Based Client Application—The client program built-in Microsoft ASP.NET which is responsible for sending requests to activate the composite web service for travel planning (Figure 6).

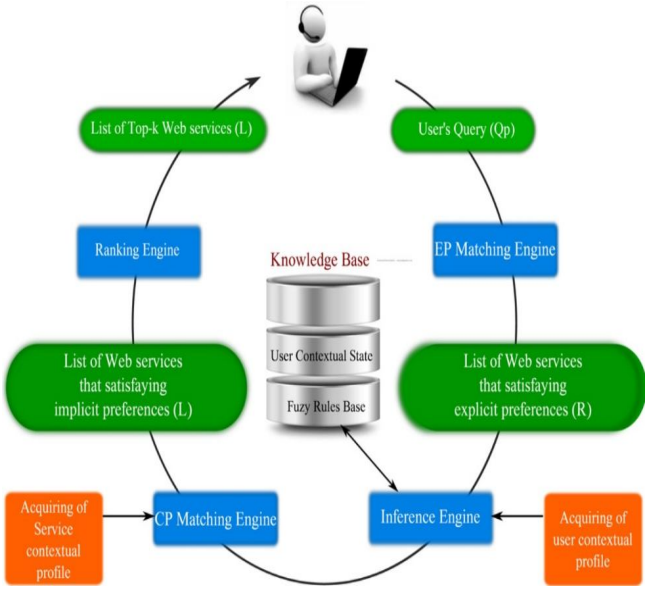

Figure 6: QoS based web service simulation

5.1. Security Evaluation

We have two inputs for information validation. One of the ones is the kind of policy that will be monitored and enforced. The other source of data is a change in a service's trust value, which simulates a service attack that compromises In addition to the performance assessment experiments discussed earlier; we also assessed the security measures on multiple service graphs provided by the PME framework [20] [21]. We approached the safety assessments in two ways. Firstly, we have verified that under the malicious conditions, the SME framework takes the specified security measures correctly. Second, we measured the improvement in trust caused by security measures taken. Security actions come in two categories. Some are preventive, preventing malicious activities. The second sections seek to rectify certain behaviors.

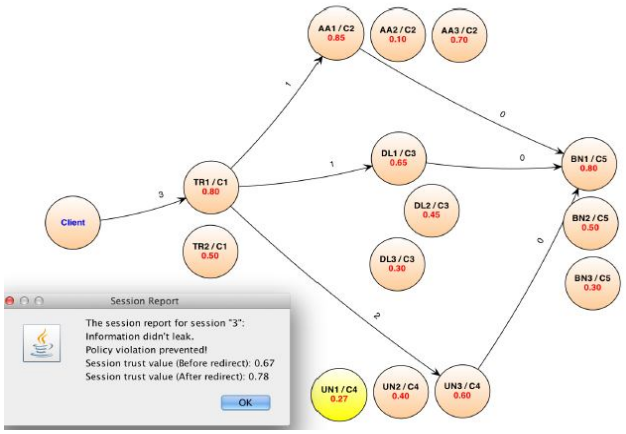

Figure 7: Redirected Policy for Enforcement

Results Figure 7 tested two extreme problems. Maximizing the implementation period for the composite travel planning service and maximizing the expense to the composite travel planning company. It should be remembered that the total execution period includes both the amount of each task's execution times and the initialization time of the composite service case. Nevertheless, the HGA doesn't incorporate the initialization period into the simulation operation. Thus, a defined distance between the actual execution time and the simulation execution time should be created, which is the start-up period. The simulation run time of the composite service under the execution plan observed by the QoS-based web service selection tool was 2:7 Sy for the first question, and the real execution time was 4:8 Sy. The dissimilarity between the simulation rate and the actual value was around 2:1 *ms*. The simulation execution period for the second problem was 18:5 Sy, and the real execution time of the composite operation under the execution plan our HGA's identified was 22 Sy. The difference is about 2 Sy, as well. From the two outcome groups, it concluded that the composite service's actual execution time is consistent with the simulation value of the proposed HGAs.

6. CONCLUSION

We introduced a new integrated web based confidence management system (known as trust manager) in SOA for trust evaluation and maintenance. This part plays a key role in holding end-to-end protection in SOA. We designed and implemented the web session management subsystem and trust maintenance subsystem as part of web service framework. We argued that monitoring should be carried out in these networks in the consumer environment because of the climate of those networks, where network uncertainty is a significant factor in the slow performance of web integrated services. The HGAs are significant contributors to this research to the issue. Experimental outcomes showed good effectiveness and scalability of the HGAs. The HGA is favored because customers add significant value to the consistency of the solution, as the HGA outperforms the other HGA's in terms of the quality of the solution.

REFERENCES

[1] BangYu Wu, Chi-Hung Chi, Zhe Chen, Ming Gu, and JiaGuang Sun, **Workflow-based resource allocation to optimize the overall performance of composite services**. *Future Generation Computer Systems*, 25(3):199 – 212, 2009.

[2] Jia Yu and Rajkumar Buyya. **Scheduling scientific workflow applications with the deadline and budget constraints using genetic algorithms,** *Scientific Programming*, 14(3):217–230, 2006.

[3] Goodubaigari Amrulla, Murlidher Mourya, Rajasekhar Reddy Sanikommu and Abdul Ahad Afroz , **A Survey of : Securing Cloud Data under**

Key Exposure, *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE),*Volume 7,No. 3,June 2018.

[4] Tao Yu and K.-J. Lin. **A broker-based framework for QoS-aware Web service composition,** In EEE '05: *Proceedings of the 2005 IEEE International Conference on e-Technology, e-Commerce and e-Service*, pages 22 – 29, March 2005.

[5]LiangzhaoZeng, Hui Lei, and HenryChang. "**Monitoring the QoS for Web Services. In Service-Oriented Computing",** ICSOC 2007, volume 4749 of Lecture *Notes in Computer Science*, pages 132–144. Springer Berlin / Heidelberg, 2010.

[6]G. Di Modica, V. Regalbuto, O. Tomarchio, and L. Vita. **Enabling re-negotiations of SLA by extending the ws-agreement speci_cation. Services Computing**. SCC 2007, *IEEE International Conference*, pages 248-251, July 2007.

[7] Sudhakar Sengan, Chenthur Pandian S, 2012, **'An Efficient Agent-Based Intrusion Detection System for Detecting Malicious Nodes in MANET Routing**", *International Review on Computers and Software (I.RE.CO.S.),* Vol. 7, No. 6, pp. 3037-304.

[8] D.Gouscos, M. Kalikakis, and P. Georgiadis. **An approach to modeling web service QoS and provision price**, *Web Information Systems Engineering Workshops*, International Conference 121-130, 2003.

[9] Sudhakar S,Chenthur Pandian, S. **"Hybrid cluster-based geographical routing protocol to mitigate malicious nodes in mobile ad hoc network"**, *International Journal of Ad Hoc and Ubiquitous Computing*, Vol.21, No.4, pp.224,2016.https://doi.org/10.1504/IJAHUC.2016.076358

[10] Saiid Tkatek, Otman Abdoun, Jaafar Abouchabaka, Najat Rafalia, **A Hybrid Genetic Algorithms and Sequential Simulated Annealing for a Constrained Personal Reassignment Problem to Preferred Posts**. *International Journal of Advanced Trends in Computer Science and Engineering (IJATCSE),Volume 9,Issue 1,Feb 2020* https://doi.org/10.30534/ijatcse/2020/62912020

[11]Y. Liu, A. H. Ngu, and L. Z. Zeng. **"Qos computation and policing in dynamic web service selection**", In WWW Alt. '04: *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 66{73, New York, NY, USA, 2004. ACM.

[12]Dr.S.Sudhakar, Dr.N.Satheesh, Dr.S.Balu, Amireddy Srinish Reddy, Dr.G.Murugan, "**Optimizing Joins in a Map-Reduce for Data Storage and Retrieval Performance Analysis of Query Processing in HDFS for Big Data**", *International Journal of Advanced Trends in Computer Science and Engineering*, Vol.8, No.5, Oct. 2019. https://doi.org/10.30534/ijatcse/2019/33852019

[13]Web Services Security: **SOAP Message Security 1.1** (WS-Security 2004), 2006.

[14]Z. Zheng and M. R. Lyu. **"Collaborative reliability prediction for service-oriented systems**". In Proc. *IEEE/ACM 32nd Int'l Conf. Software Engineering* (ICSE'10), pages 35–44, 2010.

[15]W. Zhao. Bft-ws **"A byzantine fault tolerance framework for web services**", In EDOCW '07: *Proc. of 7th Int'l IEEE EDOC Conf*. The workshop pages 89–96, 2007.

[16]S.Tiroumalmouroughane, P.Thambidurai**, Identification of Business Policy violation in Service Integration of Long Term Composed Services,** *International Journal of Applied Engineering Research*, Volume 10, 2015.

[17]Marcos Palacios, José García-Fanjul, Javier Tuya, and George Spanoudakis, **Coverage-Based Testing for Service Level Agreements,** *IEEE Transactions on Services Computing*, Volume: 8 Issue: 2, 2015.

[18]Mariano Vargas-Santiago, Saul Pomares Hernndez, Luis, Morales-Rosales, Houda Khlif, HatemHadj-Kacem**,Towards Dependable Web Services in Collaborative Environments Based on Fuzzy Non-functional Dependencies,** 5th *International Conference in Software Engineering Research and Innovation* (CONISOFT), 2017.

[19]Mirsaeid Hosseini Shirvani, **Web Service Composition in the multi-cloud environment: A bi-objective genetic optimization algorithm**, *Innovations in Intelligent Systems and Applications* (INISTA), 2018.

[20]Jagadeesh Gopal, Vellingiri J, Gitanjali J,Arivuselvan K, Sudhakar S, "**An Improved Trusted On-Demand Multicast Routing with QoS for Wireless Networks**", *International Journal of Advanced Trends in Computer Science and Engineering*, Vol.9, No.1,Feb. 2020, https://doi.org/10.30534/ijatcse/2020/39912020

[21]Wandi Aprianto1, Wibowo Kosasih, **"Analysis and Design of Web-Based Knowledge Management System for Real Estate Property**", *International Journal of Advanced Trends in Computer Science and Engineering*, Vol.9, No.1,Feb.2020, https://doi.org/10.30534/ijatcse/2020/01912020