

Computation of QoS While Composing Web Services

Khozema Ali Shabbar
Research Scholar
School of Engineering &
Technology
Career Point University
Kota, India

Dr. Tarun Shrimali
Principal
Faculty of Engineering
Sunrise Group of Institutions
Udaipur, India

Dr. Moheemmed Sha
Assistant Professor
Dept. of Computer Science
Prince Sattam bin Abdulaziz
University
Al Kharj, KSA

Abstract—Composition of web services has emerged as a fast growing field of research since an atomic service in its entirety is not capable to perform a specific task. Composition of web services is a process where a set of web services, heterogeneous in nature, are clubbed together in order to perform a specific task. Individually, Component web services may be performing well as far as Quality of Service (QoS) is concerned but the core issue is that while composing, do they satisfy Users requirements in terms of QoS? Computation of QoS while composing web services appears to be a big challenge. A lot of research work in this regard, has already been undertaken to come out with new, innovative and credible solutions for the same.

This Paper presents a thorough review-study of different frameworks, architectures, methodologies and algorithms suggested by different researchers in their efforts to compute the overall QoS while composing web services. Moreover, Effectiveness of different methods in terms of QoS while composing is also presented.

Keywords—Web Services; Web Services Selection; Web Services Composition; Composite QoS (CQoS); Quality of Service (QoS)

I. INTRODUCTION

Web services are getting more popularity as far as distributed computing and e-business/ commerce are concerned. Since Web services are loosely coupled, allowing developers to construct, produce and compose them even at execution time.

The concept of Web services composition has emerged as an effective method for integration of business related applications where numerous features of web services are combined together in order to satisfy complex requirements that an individual web service could not do. In other words, Composition is the process where prevailing composite web services or atomic web Services are clubbed together to perform a specific business operation. This composition/integration can be done manually or automatically. Since area of scope of this paper is confined to automatic/ dynamic composition of web services, we are presenting here various processes involved in composition, clearly depicting them in the following diagram (Fig 1).

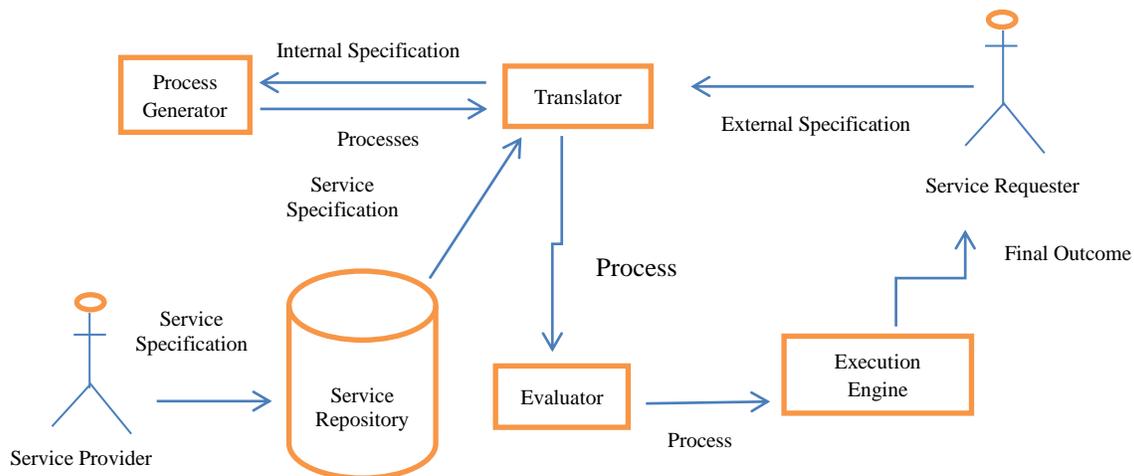


Fig. 1. A Generalized Composition Mechanism

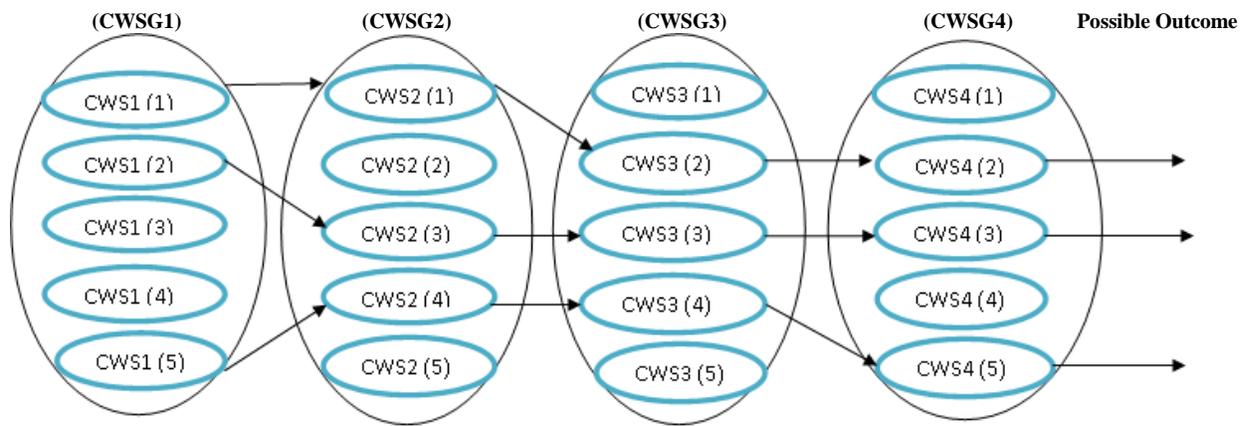


Fig. 2. Three Possible Outcomes of Composite web services

Services published by the service providers lie there in the service repository. Service requester initiates the process by making a service request specifying functional as well as non-functional constraints. This request first goes to the Translator whose job is to translate the request from its external form into a standard form acceptable to the system. Services meeting user's requirements are selected from the repository. Now Process Generator composes these services. As shown in the (Fig 2) above, when there exists more than one possible outcome of Composite Services, all of them meeting user's criteria then it is the job of the Evaluator to evaluate all those composite services and returns the best one.

Here CWSG1, CWSG2.... Stand for Candidate Web Services Group1, Candidate Web Services Group2, and so on. Similarly CWS1, CWS2 ...stand for Candidate Web Service1, Candidate Web Service2, and so on. Candidate services are group based on their functional similarities. Each group is having candidate web services with similar functionality attributes but differing in their QoS. For a particular request let say there are three possible outcomes of composite web services. Then which one is to be selected by the evaluator solely depends upon the QoS of composite web service as a whole and constituent web services individually.

In this paper, we are studying various approaches and techniques adopted to perform composition of Web Services based on QoS, computing the overall QoS of the Composite web services called CQoS. The paper is divided into different sections. Section 2- Related Work. Section 3- QoS based Web Service Composition Methodologies – an overview. Section 4- Comparison of Effectiveness of Composition Methods. Section 5- Conclusion.

II. RELATED WORKS

Composition of web services can be performed manually or automatically. In manual approach, each of the web services gets executed one after another in an ordered way to attain the specified goal, seems to be a complicated method. It requires a lot of time and efforts. With the high number of web services offered over the Internet, automatic composition appears to be more feasible. Automatic composition can be performed using any of the three methods - static, semi-static or dynamic. Static composition is done at design time and

requesters have to form an abstract process model prior to actual composition.

Dynamic composition is considered to be a complicated one as is done at run time upon the user's request. But the main advantage lies in discovering and invoking of web services dynamically on request. Some of the examples of dynamic composition are Negotiation, Semantic Web, intelligence algorithm etc. etc. Various compositions related Research works are discussed here under.

Fatma Siala et al. [1] proposed a Multi-Agents architecture to discover the optimized Composite QoS (CQoS). Composite Web services were selected having negotiation with Multiple Agents. For web service composition, user's preferences were taken into account first then agents were used to negotiate the QoS value and finally, Service Providers, providing different services, were dynamically selected in the composition. The main focus was lying in negotiating with only those Web services providers who are available, resulting in an improved CPU time. Experimental results have even proved it.

Lou Yuan-sheng et al [2] designed a 'QoS and workflow' based framework for Web service selection & dynamic composition and executed a sample system (prototype). It used visual interface to perform customization of Web service composition process. Also, it implemented global optimization algorithms to attain the dynamic selection of the suitable service, meeting composite web service QoS and the existing services QoS.

Dong Rang-sheng et al [3] presented a QoS mechanism to dynamically compose the Web Services addressing the problem of selection & coordination. They also developed a WS_TSC algorithm for the selection of services (considering the constraints like – success rate, composition rate and response time). Lastly, they carried out simulation experiments to measure the performance of dynamic composition of Web services and found better results w.r.t. success rate, composition rate and response time.

Lu Li et al [4] proposed a web service composition selection model built on the concept of Multi-dimension QoS. This concept was introduced to express the QoS attributes of web service composition. Also, it computed the QoS of web service composition depending upon the nature of web service

composition. Finally, Web service composition that optimally satisfies the non-functional requirements (besides functional requirements) was chosen among the web service compositions.

Farhan Hassan Khan et al [5] proposed an innovative technique for auto-dynamic composition of web services combining both of the techniques based on interface and functionality. They focused on issues related to QoS, data distribution and execution problems etc. To resolve the problems of decentralized dataflow, they have come up with a framework which resulted in highest throughput, better response time, and minimal latency. Also, they have presented a solution for the difficulties faced during the composition of web services because of the continuous variations in parametric values (both input/output), independent nature and network related issues of the web services.

Ming-Wei Zhang et al [6] proposed an entirely different technique for the composition of web services on the basis of 'Production QoS rule' and adopted 'black box' technique of analysis to optimize composite services. Execution information pertaining to composite service is first stored and then used as a base for the ensuing statistical analysis and QoS knowledge mining. Web services periodic QoS values are computed and the production QoS rules are mined. (Basically, these rules are applied in order to specify various performances of 'Web service QoS' that take place in diversified environments). Finally, resultant QoS knowledge about Web services is used to discover optimized composite service.

Zhi Zhong Liu et al [7] introduced a reliable "Web Service Composition" method on the basis of the decomposition of global QoS attributes and dynamic prediction of QoS. Proposed method has two critical stages: (1) prior to the composition of web services, decomposition of global QoS attributes into local attributes takes place and the issue related to the dynamic composition of Web services becomes a localised optimization issue. (2) While execution, predicted QoS values are considered to be the base for the selection of the best Web service for the present abstract service.

Rajesh Karunamurthy et al [8] presented an innovative composition mechanism by extending the existing business model of Web service to explicitly carry out composition of Web services. Proposed method supported four characteristics of Web services viz - functional, non-functional, behavioral & semantic enabling identification, selection and clubbing of different Web services being part of composition procedure. The proposed mechanism comprises different components namely – description framework, composition framework and business model.

Sabrina Mehdi et al [9] introduced a model of web service composition built on auto multi-agent based planning with high availability of web services. The said model integrates communities based substitution process, swapping an unsuccessful web service (service agent) with new one. The new one adheres to the community, presenting functionality similar to that of failure one in order to guarantee its availability.

Olfa Hammas et al [10] proposed composition architecture incorporating two aspects: 1) Dynamic Selection i.e. binding of constituent services at runtime and 2) Adaptive Composition i.e. presuming to have updated/ latest knowledge about the status of constituent services at runtime, ensuring global QoS Optimization as well as taking care of failed services by replacing them. An algorithm for the QoS aware service selection based on Ant Colony Optimization was also proposed.

Freddy L'ecu' et al [11] introduced a framework to perform service composition dynamically by performing semantic matchmaking between outputs and inputs service parameters enabling interaction and interconnection. This semantic matchmaking paves the way in discovering semantic compatibilities between service-descriptions defined autonomously. Furthermore, they introduced an algorithm for composition which follows a semantic graph based approach, where a graph denotes service compositions and nodes denote semantic relations among services. Additionally, both non-functional and functional attributes of services are taken into account to compute the most suitable and relevant service composition.

Wang Denghui et al [12] presented a novel recommendation method for the composition of web services calculating QoS credibility of every web service unit. Authors used user's comment/ weight of user's preference to calculate the reliability of different dimensions of QoS.

Pooya Shahrokh et al [13] have proposed a semi-heuristic genetic algorithm (a combination of both a heuristic method and the genetic algorithm). This heuristic method changes chromosomes based on unsatisfied constraints. As per authors, Research outcomes have endorsed the fact that the proposed method satisfies user's requirements more efficiently than other methods.

Namrata Kashyap et al [14] introduced 'QoS based service composition' model incorporating a Membership function. The said function does prioritize functionally similar candidate web services to a further higher level based on response time in order to be included in the composition, enhancing user's satisfaction.

Wei Zhang et al [15] presented a QoS based technique for composition of web services dynamically and used Ant Colony Optimization (ACO) algorithm for optimization. An updated version of ACO algorithm was presented to resolve the multi objective optimization problem aiming to find better settlements between multiple objectives.

Alexandre Sawczuk da Silva et al [16] presented a graph based PSO approach for service composition and selection which ascertains an optimum workflow & near-optimum Web services to be clubbed together in QoS based composition. Authors have addressed successfully various limitations of prevailing PSO-based techniques as the approach adopted here neither requires any selection of an initial configuration nor depends upon the users, having domain expertise. Experimental results of proposed graph based PSO technique were found to be better than greedy based PSO technique.

III. QoS BASED WEB SERVICE COMPOSITION METHODOLOGIES – AN OVERVIEW

There exists many ways to combine web services. Following is a simple diagram (Fig 3) showcasing different methodologies for the composition of Web services.

A. Static Composition

Static composition is performed in design-phase where the design of the software system and the architecture are planned. The components to be used are selected, connected together, compiled and deployed finally. It may work well as far as the web service environment – service components and business

partners change rarely. Examples of static composition engines include Bea WebLogic and Microsoft Biztalk etc. It requires lot of time & efforts.

Static Composition approaches are almost obsolete ones. Rapid growth of Web services and their well-defined interfaces prompted researchers towards auto-dynamic approaches for the Composition of Web service which are found to be more practical & feasible. In the following section it has been discussed in detail.

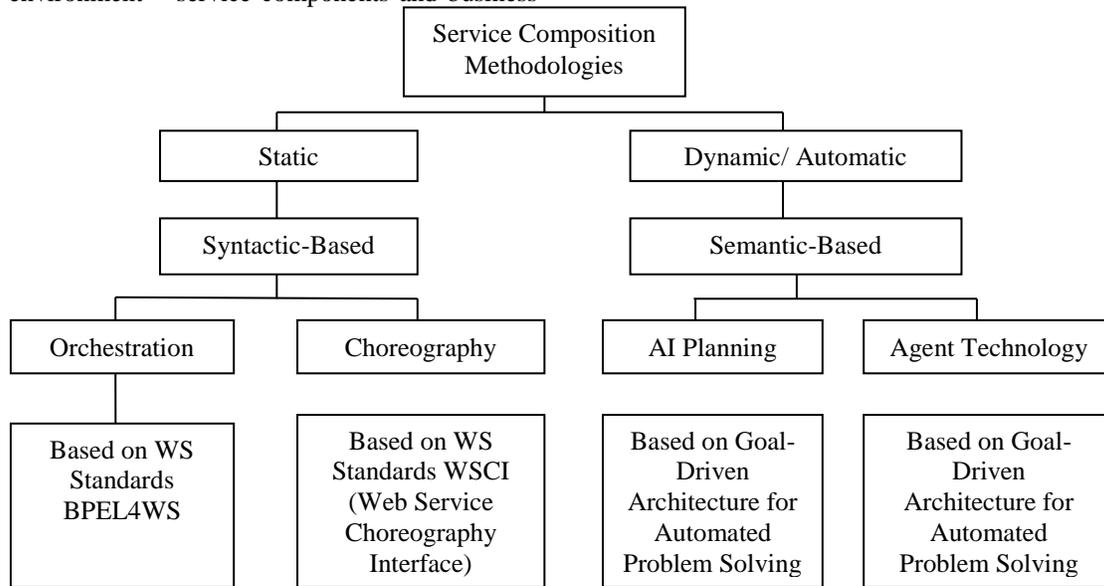


Fig. 3. Service Composition Methodologies

B. Dynamic/ Automated Composition

Area of automatic/ dynamic composition of web services is getting more attention as researchers are seeking innovative ways to perform compositions more proficiently finding better quality of results. The job of performing automatic Composition of web services is complicated one as it searches for candidate web services from a huge web service repository and handles service descriptions, not following a common standard. Most of the auto-dynamic compositions are based on workflow and/or Artificial Intelligence techniques.

1) Based on Multi-agent

In [1] [9] authors are incorporating multi- agents whose role is to talk about the QoS values and make selection of service providers dynamically for different services in the composition to discover the optimal Composite QoS. Both are focusing to ensure high availability of the required web services - [1] by talking to available providers (of Web services) only, resulting in improved CPU time and [9] incorporating a process of substitution, swapping an unsuccessful web service with a new one.

2) Based on Workflow

In [2] authors have proposed a framework for the composition of web services built on work flow incorporating dual features - dynamic composition & QoS scalability. This

‘QoS scalability’ facilitates adding/modifying existing QoS attributes (or say an extension to user defined properties) which can deliver distinct services to satisfy some distinguished QoS constraints. For QoS optimization, a hybrid version of algorithm depending on ant colony algorithm (CA) and genetic algorithm (GA) was also presented.

In [6] authors have proposed an innovative composition mechanism for web services on the basis of ‘Production QoS rules’. The novelty of the mechanism lies in taking into account the relationships of Web service QoS to environments, which is generally neglected. First, execution repository is constructed by recording composite service performance information in a tabular form. Web services QoS point dataset is extracted and timely QoS values are computed. Then the production QoS rules are mined. Finally, estimated TQoS (True QoS) and Web services production QoS rules are utilized to discover optimized composite service.

In [3] authors have proposed a QoS model for dynamically composing Web services addressing problems of service components selection and services composition coordination. Authors have developed a WS _ TSC algorithm for the selection of services based upon the constraints - success rate, composition rate and response time. At the end, they performed simulation experiments in order to measure the performance of dynamic composition of Web services and

found better results w.r.t. success rate, composition rate and response time.

3) *Based on Multi-Dimension QoS*

In [4] authors presented a web service composition selection mechanism built on Multi-dimension (time, spatial, reliable and cost) QoS. Proposed concept of Multi-dimension describes the QoS attributes of web service composition. It then, computed the average QoS of each dimension of the constituent web service to measure the QoS of each dimension of composite web service. At the end, they applied improved Euclidean Distance algorithm to compute the QoS of composite web services in order to find the best composite web service, meeting User's requirements.

In [12] authors presented a reliable QoS-aware recommendation method for the composition of web services wherein at first, all global QoS dimensions of each of service units were added to find the global QoS value of composite service and computed the credibility global QoS dimensions of composite service w.r.t. advertised QoS value and execution result. Next, they selected the lowest QoS value in all of the service units as the local QoS dimension. The credibility local QoS dimensions was computed based on the reliable user's experience. User's preference weight was attached to the credible QoS information in order to compute evaluation result and the service with the highest value was finally chosen.

4) *Based on Improved Genetic Algorithm (IGA)*

In [13] authors have proposed a semi-heuristic genetic algorithm, a combination of both a heuristic method and the genetic algorithm. This heuristic method changes chromosomes based on unsatisfied constraints. Authors have added several heuristic methods to control the randomness of GA in such a way that mutation is carried out based on the quality parameters of the issue so that the algorithm does not move away from the optimal space of the optimal combination without being trapped by the local optimality. The dynamic weighting was adopted too to determine the required quality parameters.

5) *Based on Fuzzy Logic*

In [14] authors proposed Web-service composition System introducing a new formula (member function) that does prioritize the atomic/aspirant web services based on their response times to be included in the composition using fuzzy logic. The main idea here is that a composite service is assumed to be a fuzzy set where best candidate services having highest priority are included and priority is determined based on the response time.

6) *Based on Ant Colony Optimization*

In [15] authors have proposed a technique of performing decomposition of composite services into parallel execution paths, having a general flow structure. The problem of dynamic composition of web services for each execution path was considered to be a multi-objective optimization problem and hence presented MO-ACO (Multi-Objective Ant Colony Optimization) algorithm to handle it. MO-ACO is an improved version of the existing Ant Colony Optimization (ACO) algorithm. Experimental outcomes confirmed that proposed new algorithm was able to discover near-optimum

results for multi-objective problems in a very efficient way as well as was scalable to perform composition of web services, highly complex in nature.

In [10] authors have proposed dynamic composition architecture for web services incorporating mechanism for adaptive composition, ensuring global QoS optimization. The problem of QoS based service selection was mapped to the multi-dimensional multi-choice knapsack problem, to find an atomic service from each service class to be put in the knapsack to build the composite service, ensuring that the aggregated QoS values should satisfy users QoS constraints.

7) *Based on Intelligent Algorithm*

In [7] authors introduced an innovative mechanism for QoS based composition of web services comprising two important steps. At first, global QoS attributes are decomposed into local attributes in order to find the combination of optimum local QoS attributes for each of the service classes and a novel CGA optimization algorithm was constructed combining Genetic Algorithm with that of Culture Algorithm. Secondly, they have designed QoS prediction method on the basis of improved case-based reasoning to predict the candidate services QoS prior to the selection a specific Web service for a particular service.

8) *Based on Graph*

In [11] authors have introduced a model for the composition of web services on the basis of functional attributes wherein services are bound depending on their functional description - Input, Output, Preconditions and Effects (IOPEs). The proposed model used the Causal Link Matrix (CLM) formalism to enable the final service composition computation in the form of semantic graph. Semantic connections between constituent web services are represented by the nodes of the semantic graph. At last, selection of the set of valid service compositions is made depending upon the non-functional attributes of the constituent web services. If a service composition fails to meet the non-functional attributes of the requested service, it is ignored.

In [16] authors have presented a PSO method based on graph for the QoS-aware composition of Web service. Instead of preselecting an abstract workflow for optimization, proposed method relied on the creation of a master graph of candidate services. They presented a greedy-based PSO method that too preselects an abstract workflow. A Set of experiments were conducted on both of the models to compare their efficiency and results of the experiments showed that the graph-based method was proved to be more efficient in producing solutions whose fitness values couldn't be matched by the solutions found from the greedy-based model.

9) *Based on Decentralized Dataflow*

In [5] authors have introduced an algorithm for dynamic composition of web services to resolve composition problems relating to availability, reliability, QoS and data distribution. In order to ensure data availability and make the system more reliable, authors have proposed a mechanism, introducing numerous repositories and Web Service Databases. In other words, data availability and retrieval of up to date information have been guaranteed by means of multiple registries and

aging factor respectively. The proposed system based on Quality of service, is a fault tolerant and reliable one, performing fast data retrieval.

10)Based on Semantic Web

In [8] authors have come up with a novel architecture by extending the existing business model of Web service to explicitly carry out composition of Web services. Proposed architecture supports four characteristics of Web services - non-functional, functional, semantic and behavioral enabling the identification, selection, and clubbing together of various component Web services involved in the composition.

There are two important aspects of the proposed method that distinguishes it from rest of the methods - 1) standard business model of Web service has been extended to carry out composition of existing services in order to create unavailable

services that consumers search for and 2) described and composed Web services integrating all four characteristics altogether in a common semantic domain (other existing description techniques do not allow all of them altogether).

IV. COMPARISON OF EFFECTIVENESS OF COMPOSITION METHODS

In this section, we will be presenting a comprehensive analysis of the various methods employed to perform Web Service Composition (WSC) based on QoS. All the methods discussed in this paper are Semantic based and automatic/dynamic in nature. Below find here are the important factors solely based on the related research works. Each of the related works included here, showcased the effectiveness of the technique/ algorithm employed for the composition of web services.

TABLE I. COMPARISON OF METHODOLOGIES

Related Article	Approach/ Method	Technique/ Algorithm	Parameters	Effectiveness
Fatma Siala et al. [1]	Based on Multi-Agent Negotiation	Multi-Agents Negotiation to find out the best CQoS. Used – ‘Extended FIPA Protocol’ and ‘Contract Net Protocol’	Speed, Accuracy Availability, Price, Reliability, Reputation	Experimental results of the Proposed Multi-Agents Model demonstrated better results in terms of Execution Time (better CPU Time) compared to other existing Models. It can be applied to other distributed Computing Paradigms too
Lou Yuan-sheng et al [2]	Based on Work-flow	A hybrid version of Algo ^m (Ant Colony Optimization + Genetic Algo ^m) for QoS OptimizationFlowchart (established according to User’s QoS requirements) is taken as XML documents to compute Score	Reliability	Proposed Service Composition framework based on QoS supported dynamic Web service selection and the visualization modeling
Dong Rang-sheng et al [3]	Based on Work-flow	Objective function F as Performance Metric Criteria. Developed Algorithm, WS_TSC for the Selection of Services	Rate of Composition, Rate of Success and Response Time	Proposed architecture was found performing better w.r.t. the rate of Composition, rate of the Success and Response time i.e. higher rate of Composition & Success and Less Response time
Lu Li et al [4]	Based on Multi-Dimension QoS	Improved Euclidean Distance Algorithm	Four Dimensions – 1. Time: (Execution & Communication Time) 2. Spatial: (Message length & Storage Capacity) 3. Reliable: (Reliability & Availability) 4. Cost: Service Cost	Proposed Model presented Multi-Dimension QoS which can more exactly describe the QoS Attributes of Composite Web Service as well as help conduct a better study of the QoS of Composite Web Service.

Farhan Hassan Khan et al [5]	Based on Decentralized Dataflow	Auto-Dynamic Composition Mechanism by combining Approaches based on Interface and Functionality.	Reliability, Availability, Latency, Throughput, Response Time	Proposed Mechanism is Reliable, Fault Tolerant, Efficient in Data Retrieval and based on QoS. Proposed Algorithm for Dynamic Web Services Composition resolves the composition issues related to Availability, Reliability, QoS and Data Distribution
Ming-Wei Zhang et al [6]	Based on Workflow	Black-Box technique of Analysis to Optimize Composite Services and Mining Algorithm - QoS based on Production QoS Rule	Cost, Response Time, Robustness, Security	Proposed Service Composition Approach based on the production QoS rule does – <ul style="list-style-type: none"> • Improve the System Efficiency • Increase the System Stability and • Reduce the possibility of service system abnormality in Dynamic Service Execution Environment
Zhi Zhong Liu et al [7]	Based on Intelligent Algorithm	Novel CGA Optimization Algorithm (Integrating GA into the framework of CA) and Improved Cased-Based Reasoning (CBR)	Cost, Response Time, Availability, Reliability	Proposed Approach not only reduces the complexity but also increases the flexibility and the efficiency of Service Composition besides greatly enhancing the credibility of Composite Web service
Rajesh Karunamurthy et al [8]	Based on Semantic Web	Pellet Reasoner and Isabelle Theorem Prover	Non-functional, Functional, Semantic and Behavioral characteristics	Proposed Architecture executes faster Composition of Web Services resulting in higher Response Time and taking into account all of the four characteristics altogether – Non-functional, Functional, Semantic and Behavioral
Sabrina Mehdi et al [9]	Based on Multi-Agent	Multi-Agent Automatic Planning Architecture (using Java/ JADE), Contract Net Protocol	Availability Cost, Response Time, Reliability	Proposed Model ensures higher availability of web services and treats the failure of web service or fault tolerance (Auto- Substitution mechanism based on Communities)
Olfa Hammas et al [10]	Based on Ant Colony Optimization	Ant Knapsack Algorithm	Availability, Reliability, Response Time, Execution Price, Latency, Throughput,	Proposed Composition Model incorporated two Concepts – 1. Dynamic Selection (binding of services at runtime) and 2. Adaptive Composition (assuming to have updated knowledge about constituent web services status)
Freddy L'ecu' et al [11]	Based on Semantic Graph	Causal Link Matrix (CLM) formalism, Algorithms for the Composition of Web Services & Ranking of Composition Results	Cost, Response Time	Proposed general Model for the Composition is quite suitable for Web Services which are described using WSMO (capability model), SA-WSDL, or OWL-S (service profile) specification
Wang Denghui et al [12]	Based on Multi Dimension QoS	QoS Aware Web Service Composition Recommendation Method	Global QoS Dimensions are - Price, Execution Time, Local QoS Dimensions are - Successful Rate, Availability, Reliability, Security	Proposed Approach considered the user's experience to describe all QoS dimensions and calculated the credibility of end user's experience to ensure the objectivity of user commend

Pooya Shahrokh et al [13]	Based on Improved Genetic Algorithm (IGA)	Semi-heuristic Genetic Algorithm (combination of both heuristic method and the genetic algorithm),	Availability, Execution cost, Response Time, Successful Execution Rate, Reputation	Proposed Method can be applied to discover a Composition technique which satisfies user's requirements more efficiently than other methods. Applying heuristic genetic algorithm resulted in improved Execution Time
Namrata Kashyap et al [14]	Based on Fuzzy Logic	Fuzzy logic & Membership functions	Reputation, Availability, Execution Price, Successful Execution rate, Execution Duration	Performance of the new improved Afive formula was found to be better than the existing Atri formula as it successfully prioritized Web Services at more precise and finer level on the basis of their response times Atri and Afive stand for the tri-value and the five-value QoS attribute of A of the Web Services Composition
Wei Zhang et al [15]	Based on Ant Colony Optimization	Multi Objective Optimum Path Selection Technique and MO_ACO Algorithm (an improved Version of existing Ant Colony Optimization Algorithm)	Response Time Cost, Availability, Reliability	Experimental Outcomes demonstrated that Proposed new Algorithm (MO_ACO) was able to discover near-optimum results for Multi Objective Problems in a very efficient way as well as was scalable to perform Composition of web services, very complex in nature
Alexandre Sawczuk da Silva et al [16]	Based on Graph-based Particle Swarm Optimization - PSO	Graph-based Particle Swarm Optimization (PSO) Technique and Service Composition Algorithm for Particle as well as Fitness Function	Availability, Response Time Execution Cost, Reliability	Graph-based technique produced far better solutions surpassing to the solutions achieved from the Greedy-based technique.

After thoroughly reviewing the effectiveness of the above listed QoS based Composition mechanisms, techniques employed by Lu Li et al [4], Farhan Hassan Khan et al [5] and Wei Zhang et al [15] were found to be more practical & effective in terms of quality. Techniques employed by Pooya Shahrokh et al [13] and Rajesh Karunamurthy et al [8] were found to satisfy user's requirements (functional aspects) more efficiently than other methods.

V. CONCLUSIONS

The above mentioned composition methods give us a better insight while composing web services to reach the functionality of a specific task. To accomplish a specific task where an atomic web service is not able to do it lonely, composition of web services appears to be the best solution. Component web services are those best web services selected based on the individual performance in terms of quality (QoS). While composing, computation of overall QoS still remains a big question to be answered to. Composite QoS (CQoS) plays vital role in finding the best Composite Web Service. Though, Researchers have done a great job & have come up with innovative techniques to compute Composite QoS more accurately in order to retrieve the most relevant & desirable composite web service but still, they are more or less depending on either Providers published information or users provided feedback. As discussed earlier, Providers published information may not always be true, credible & up-to-date. Similarly users provided feedback may also be biased sometimes. Hence we reach to the conclusion that an independent & consistent QoS evaluation system is still needed to measure more accurate value of QoS. Researchers

have to focus their research on developing such a novel system which can work independently taking into consideration more number of non-functional attributes. As a future scope, based on the various frameworks reviewed here in this paper, an innovative framework will be proposed incorporating more number of non-functional attributes, so that it can compose web services more efficiently in terms of QoS.

REFERENCES

- [1] Fatma Siala, Khaled Ghedira, "A Multi-Agent Selection of Web Service Providers Driven by Composite QoS", IEEE, pp. 55-60, 2011.
- [2] Lou Yuan-sheng, Tao Zhen-hong, Yue Lu-lu, Xu Hong-tao, Xi Zhi-hong, Wu Zhi-feng, "A QoS-based Web Service Dynamic Composition Framework", IEEE, Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science, pp. 188-192, 2010.
- [3] DONG Rang-sheng, WANG Fei-ming, LUO Xiang-yu, "Dynamic Web Services Composition Based on QoS Model", IEEE, pp. 823-826, 2010.
- [4] Lu Li, Mei Rong, Guangquan Zhang, "A Web Service Composition Selection Approach based on Multi-Dimension QoS", IEEE, The 8th International Conference on Computer Science & Education (ICCSE 2013) Colombo, Sri Lanka, pp. 1463-1468, April, 2013.
- [5] Farhan Hassan Khan, M. Younus Javed, Saba Bashir, Aihab Khan, Malik Sikandar Hayat Khoyal, "QoS Based Dynamic Web Services Composition & Execution", International Journal of Computer Science and Information Security, vol. 7, no. 2, pp. 147-152, February, 2010.
- [6] Ming-Wei Zhang, Bin Zhang, Ying Liu, Jun Na, Zhi-Liang Zhu, "Web Service Composition Based on QoS Rules", JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY, vol. 25, no. 6, pp. 1143-1156, November, 2010.
- [7] Zhi Zhong Liu, Zong Pu Jia, Xiao Xue, Ji Yu An, "Reliable Web service composition based on QoS dynamic prediction", Springer, pp. 1409-1425, July, 2014.

- [8] Rajesh Karunamurthy n, FerhatKhendek,RochH.Glitho, "A novel architecture for Web service composition", *Journal of Network and Computer Applications*, Elsevier, vol. 35, no. 2012, pp. 787–802, November, 2011.
- [9] Sabrina Mehdi, Nacer eddine Zarour, "Composition of web services using multi agent based planning with high availability of web services", *IEEE, 2nd International Conference on Advanced Technologies for Signal and Image Processing - ATSIP'2016 Monastir, Tunisia*, pp. 10-15, March, 2016.
- [10] Olfa Hammas, Saloua Ben Yahia, Samir Ben Ahmed, "Adaptive Web Service Composition Insuring Global QoS Optimization", *IEEE*, 2015.
- [11] Freddy L'ecu'e, Eduardo Silva, and Lu'is Ferreira Pires, "A Framework for Dynamic Web Services Composition", *Emerging Web Services Technology*, vol. 2, 2008.
- [12] Wang Denghui, Huang Hao, Xie Changsheng, "A Novel Web Service Composition Recommendation Approach Based on Reliable QoS", *IEEE, Eighth International Conference on Networking, Architecture and Storage*, pp. 321-325, 2013.
- [13] Pooya Shahrokh, Faramarz Safi-Esfahani, "QoS-based Web Service Composition Applying an Improved Genetic Algorithm (IGA) Method", *International Journal of Enterprise Information Systems*, vol. 12, no. 3, pp. 60-77, July-September 2016
- [14] Namrata Kashyap, Kirti Tyagi, "Dynamic Composition of Web Services Based on Qos Parameters Using Fuzzy Logic", *IEEE, International Conference on Advances in Computer Engineering and Applications (ICACEA), IMS Engineering College, Ghaziabad, India*, pp. 778-782, 2015.
- [15] Wei Zhang, Carl K. Chang, Taiming Feng, Hsin-yi Jiang, "QoS-based Dynamic Web Service Composition with Ant Colony Optimization", *IEEE, 34th Annual Computer Software and Applications Conference*, pp. 493-502, 2010.
- [16] Alexandre Sawczuk da Silva, Hui Ma, Mengjie Zhang, "A Graph-Based Particle Swarm Optimisation Approach to QoS-Aware Web Service Composition and Selection", *IEEE, Congress on Evolutionary Computation (CEC), Beijing, China*, pp. 3127-3134, July, 2014.