International Journal of Web-Based Learning and Teaching Technologies
Volume 17 • Issue 4

An Extensive Review of Web-Based Multi-Granularity Service Composition

Anushree Sah, University of Petroleum and Energy Studies, India
https://orcid.org/0000-0003-3444-5860

Saurabh Rawat, Graphic Era University, Dehradun, India

Tanupriya Choudhury, University of Petroleum and Energy Studies, India
https://orcid.org/0000-0002-9826-2759

Bhupesh Kumar Dewangan, University of Petroleum and Energy Studies, India
https://orcid.org/0000-0001-8116-7563

ABSTRACT

The paper reviews the efforts to compose SOAP, non-SOAP, and non-web services. Traditionally, efforts were made for composite SOAP services; however, these efforts did not include the RESTful and non-web services. A SOAP service uses structured exchange methodology for dealing with web services while a non-SOAP follows a different approach. The research paper reviews the invoking and composing a combination of SOAP, non-SOAP, and non-web services into a composite process to execute complex tasks on various devices. It also shows the systematic integration of the SOAP, non-SOAP, and non-web services describing the composition of heterogeneous services than the ones conventionally used from the perspective of resource consumption. The paper further compares and reviews a different layout model for the discovery of services, selection of services, and composition of services in cloud computing. Recent research trends in service composition are identified, and then research about microservices are evaluated and shown in the form of tables and graphs.

KEYWORDS

Cloud Computing, Microservices, Quality of Service, REST, SOA, SOAP, UDDI, Xml

INTRODUCTION

Working in a fixed humdrum predictable pattern by Organizations in this advance century is being considered outdated. Need is changing as market is turning with consistently high speed. Some renowned technologies like SOA, Cloud Computing, Big Data, Http (Hypertext transfer protocol), SOAP, REST, UDDI, XML etc. are evolving continuously. The need for system and infrastructure over internet emerges. SOA, web services, and Cloud Computing are providing a solution to this problem. These technologies help us in producing interoperable, distributed, parallel computed software systems for both huge enterprise applications or for small businesses. For evolving technologies, like Cloud, the principal constituent is, SOA to deliver IT services. Outer world continuous and rapid changes can be tackled by this. So it is foremost to acknowledge how web services and workflows are scheduled in an IT environment. And also to find out what all are the challenges and issues faced while composing them.

DOI: 10.4018/IJWLTT.285570

With Adoption of SOA, IT directors and users are capable to address today's critical challenges of virtualization and at the same time it will provide a solid basis for enterprises to embrace cloud computing for future world. Cloud computing provides significant advantages over current IT architecture (B. Li et al., 2013).

There are two complementing activities in today's enterprise market: Virtualization and Cloud computing. These technologies are working together and making IT reach new heights. The work will present an overview, style and actuality of service oriented architecture with the objective of laying down challenging issues bothered with miscellaneous traits of assimilating heterogeneous web applications using SOA. A case study will be discussed that will integrate the services using heterogeneous technologies. The proposed work will show web service discovery, selection and composition mechanism.

With the increasing demand, systems are also getting larger and larger than before. Due to this complexity is also increasing with high speed. As we all know that complexity rises exponentially with responsibility and size" (M. Ciavotta et al., 2017) Systems need to be connected with each other. So for the purpose of interconnecting these systems many concepts evolved over the time as per the demands of future. SOA is also a natural evolutionary phase to the procedural, object-oriented (OO), data-centric and component based approaches adopted for the explanatory applications till now (H.-L. Truong and P. Klein, 2020).

There are three types of roles used while developing a web service. It is shown in the Figure 1.

Discovery, Selection and Composition of Heterogeneous Web Services

The Web services composition life cycle is proposed by (Q. Z. Sheng et al, 2017), as shown in Figure 2. My contribution focuses on three main phases of the life cycle: (i) Service Discovery, (ii) Service Selection and (iii) Service Composition for the Heterogeneous Web services.

For composing a service in an efficient and effective way a case study will be made which is the reflection of the work done by me and this case study will also give an insight about the work in real world scenario (Q. Yu et al., 2015).

A service provider publishes the service on a Web service registry, which accommodates distinct service descriptions, i.e., Web Service Description Language (WSDL) file. To retrieve the services queries has been sent to registry by service consumer. A service consumer queries the registry to retrieve services. Service descriptions indicate competence of services and usually accommodate inputs, outputs, functional and non-functional description (J. Liao et al., 2016). Figure 2 shows the various stages involved in the life cycle of service composition, which are described as follows.

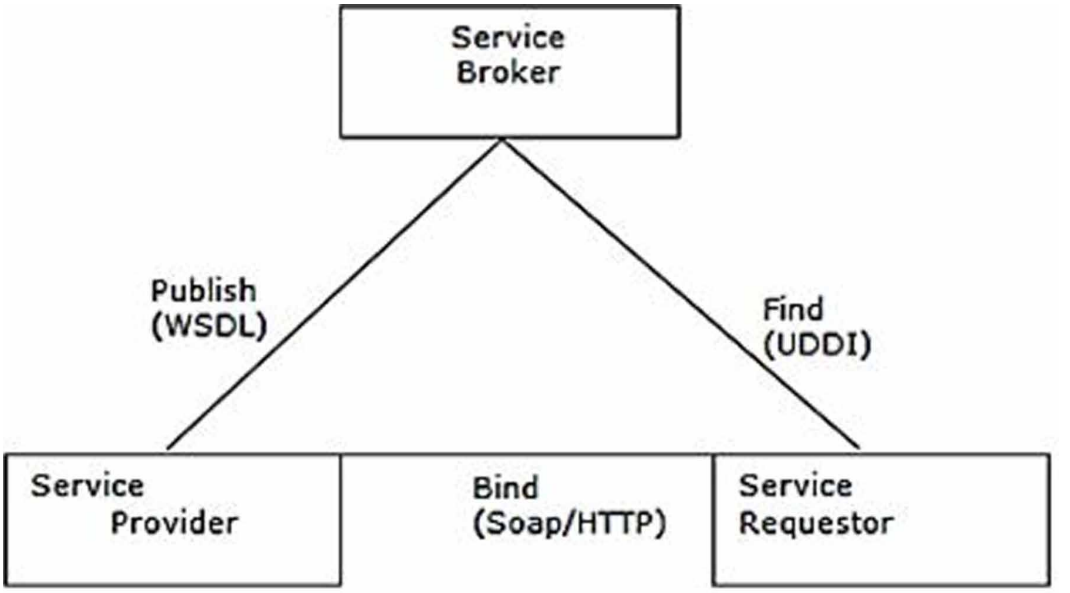
Service Discovery

Service discovery is a process of searching for services based on user requirements. Typically, a service discovery requires performing the matching operation between the capabilities of services published by the services providers in the service repository and a requester queries. An intermediate entity could be the reasoner in this matchmaking process, which will return the matched result (A. Sah et al., 2018).

Service Selection

Service selection is the process of selecting (as well as filtering) an appropriate service from a bank of functionally equivalent services. To accomplish an emended solution, most of the research in service selection is based on Quality of Services (QoS) incorporated in a selection. QoS parameters are the mode by users to express their preferences. From the results of functionally equivalent services, QoS-based techniques are used by user to select a service. Reputation and trust-based mechanisms (C. Xia et al., 2018) are also used to clarify best services from the service repertoire.

Figure 1. General architectural model of web services (Phu, P. H., 2006)



Service Composition

Service composition is defined as a collection of fundamentally and composite services. On top of service description, discovery, and selection activities, the service composition allows a user to create value added services (C. M. Feng et al, 2017). Composite services suggest reusable capabilities to developers (Dewangan, B. K., & Pasricha, A., 2019a). Service composition implements a seamless admittance to a breed of complex services to a user. Control flow information, which outlines the order of execution of services, is maintained in the form of composition plan (also known as an abstract process). In this process, data flow will be performed, which helps to identify reusable data inputs among services (A. Sah et al., 2018).

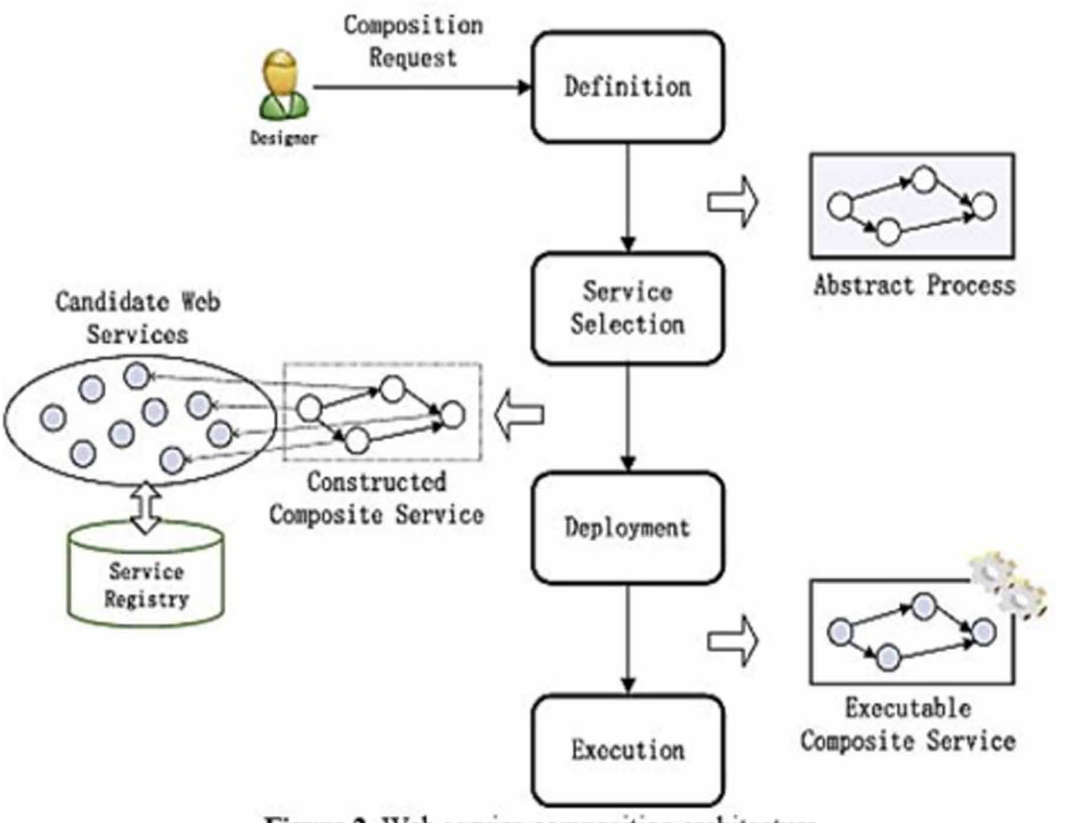
BACKGROUND AND MOTIVATION

IT advancements have changed the way of work to a great extent. Every now and then new techniques are being introduced in the IT industry and almost all other industries are dependent on the IT industry. So while digitizing the market, the major concern is inter communication between different technologies and reusability of already developed services is a major concern. The aim of this research is to discover a scalable, efficient and effective way aimed at interoperating web services using SOA. The following lists the goals our research provides.

The paper (W. Ben Messaoud et al., 2016) is the survey of web service composition and it suggests different composition approaches for web services. In the first section static composition is covered there they explained the concept of web service orchestration and web service choreography. Then they discussed dynamic composition approaches. The dynamin composition is also called on line, post compiled or reactive composition i.e. it means dealing web services on the fly. In the approach BPEL4WS is described and used.

The paper (C. Jatoth et al., 2017) presented a systematic classification and comparisons of all the researches carried out and published during 2005 to 2015. They discuss the different classification of research approaches and show what all types of algorithms are used in QoS aware

Figure 2. Web service composition architecture (Sheng et al., 2014)



web service composition. It is showing the roadmap and direction of the researches going in web service composition using QoS. Parameters like security, throughput, cost, reputation, response time, availability, reliability are the QoS parameters that are widely used in web service composition.

Web service composition frameworks can be made either by graph theory based approaches or by combinatorial approach. In the paper (P. Rodriguez-Mier et al., 2016) an integrate framework for discovery and composition of web services is proposed. First of all a theoretical analysis of graph based service composition is discussed. The framework proposed a method to get best composition graph using a search algorithm for composition thus it minimizes number of services used and provide different graph optimizations to improve the scalability of the system. The proposed framework shows how web service composition system will be developed in the real world scenario.

In the paper (R. Ben Lamine et al., 2017) web services is searched taking into account of user preferences and inputs, then a planning graph is constructed based on semantic context – aware web service discovery. Later a backward search step, semantic and contextual similarities scores are generated for composing web services from the web service list. At the last, a final set of ranked solutions are returned to the user with their specific scores.

The research paper (A. Ouni et al., 2017) provide an algorithm to solve the detection of web services anti-patterns. The anti-patterns lead to the poor designs of the web services and detecting them is a manual and tiresome approach which is also error-prone. This approach uses variety of perspective and expertise to find out potential anti-patterns. There are two algorithm used in this paper. One to detect anti-patterns using genetic programming and second is executed in parallel and employs an EA based on a genetic algorithm. The essential thought of the two algorithms is

to investigate the inquiry space by making a population of applicant arrangements, likewise called people, what's more, advance this population towards an "ideal" arrangement for a particular issue. To assess the arrangements, the wellness work in the two algorithms has two parts. For the to begin with part of the wellness work, GP assesses the location rules in view of the accomplished scope of anti-pattern cases (input), while the GA assesses the identifiers by figuring the aberrance from very much outlined Web administrations (input). At that point an arrangement of best arrangements is chosen from the two algorithms in every emphasis. The algorithms interface with each other utilizing the second segment of the wellness work called the intersection function, where an arrangement of web administrations, not quite the same as the ones used to create the sources of info, is assessed utilizing these arrangements keeping in mind the end goal to boost the intersection between the arrangements of identified anti-patterns by every arrangement.

Previous traditional methods of web services composition usually lack flexibility in selection. To overcome these types of shortcoming the paper (Q. Wu et al., 2016) provides a concept of generalizing component services. A service composition model is formulated which is QoS aware. Paper proposes backtracking based algorithm and extended genetic algorithm to optimize the resulting composite service instance.

The paper (Q. Z. Sheng et al., 2014) has 5 objectives. Paper proposed a Self-adaptive exception handling architecture model. Then runtime collaborative adjustment mechanism has been designed. It is set to make the resource allocation an easier and more effective process using the methodology discussed in the paper. The paper proposes methods that can make traditional SOA better than existing SOA the does not have runtime adaptive regulating mechanism. The paper talks on how we can increment the ability to meet the runtime context changes.

The need for access to social services and public utility is one of the major concerns of the government (Choudhury, T., & Pasricha, A., 2020). What SOA or Service Oriented Architecture can provide is exactly what is needed by the government of any country. The thing about SOA that makes it so suitable for the job is its loosely coupled nature, that means things are not directly dependent on one other (Choudhury, T., & Chandra Satapathy, S., 2020). The problem discussed in the article is that relating to public accessibility of utilities and the functionalities provided by the government. The current measures are lacking behind a lot, especially in terms of scalability and agility. The objective of the article is to find the solution or rather give the solution of the aforementioned problem with the help of SOA and to solve the problems that the government faces in all countries (Y. Lu and X. Xu, 2017).

The design is separated into four major categories which are:

- Government
- Citizen
- Infrastructure
- Human Resource

The article is about the importance of web services description .This description plays an important role for carrying out the smooth communication and services between the service provider and the consumer. There are many services that provide the same functionality and this often creates confusion for the customer as to which service to choose (Dewangan, B. K. et al., 2012). Thus this description helps to choose the most appropriate service (H. Omrana et al., 2013).

The paper did a detailed comparison between prominent frameworks available for describing web services- OWL-S, WSMO, WSDL 2.0, SAWSDL and WS-Policy. OWL-S and WSMO provide new standards with notably rich semantic description but is complex to manage and also require specialized knowledge to handle. WSDL is flexible as well as provide technical details unlike WSMO which deals only with non-functional properties. SAWSDL based grounding provides relationship between OWL-S, WSMO and Web services standards.

The paper (Jain, A., Choudhury, T., 2020) shows an approach to minimizing the challenges in service oriented architecture (SOA). Further in the paper a Automatic Web service Composition (AWSC) architecture is suggested. A prototype that is able to deploy and execute Automated Composite Web services is proposed. Later comparisons of the algorithms for Automatic Web service Composition (AWSC) is shown (B. T. Kuehne et al., 2013).

In the paper (J. Lee et al., 2015) author proposed a framework for composing SOAP, Non-SOAP and Non-Web Services. The paper proposes an approach that uses extended BPEL engine bundles with adapter for direct invocation and composition of SOAP, RESTful and OSGi services. The paper shows how to integrate services and non-service like web contents and mobile applications into a composite service process. In the paper an experimental evaluation is provided to give turnaround time and network traffic of the proposed approach is better than the previous traditional one (Agarwal, A., & Pasricha, A., 2016). The author suggested an approach to integrate or compose heterogeneous web service with various types of contents and protocols.

In most of the previous work all the attention is given to the QoS. No work is done to minimize the number of the web services which is used to full fill the requirement. Most of the approaches available focus on optimization of QoS composition and very less work is done on automatic composition of services which uses minimal number of web services while automatic composition. The paper (P. Rodriguez-Mier et al., 2016) proposes a concept of optimizing the composition of web services using composition graph for a request. The approach uses hybrid global-local search for QoS (Dewangan et al, 2019c). extraction and thus reduces the search space by first identifying the optimal service and then providing multistep optimization (Dewangan et al, 2021a). The comparative analysis of services is mentioned in Table 1.

Motivation

Nowadays, Web services are considered the most demanding mechanism of distributed computing. Moreover, service discovery and selection are the fundamental processes to search and select composable services based on semantic and non-functional description apart from functional and behavioral aspects to allow rapid creation of new Web services from existing services.

Service discovery, selection and composition are in general, complex tasks that require considerable effort especially, when vast amounts of services are available. Web service discovery, selection and composition should be performed in an integrated manner due to the strong dependency among these tasks (P. Rodriguez-Mier et al., 2016). However, researchers have considered the service discovery, selection and composition tasks at individual level.

In the paper (P. Rodriguez-Mier et al., 2016), it is clearly identified that, the discovery and composition of Web services should be based on semantic description and should also be QoS-aware. Web services having the functional, behavioral, semantic, and non-functional properties, should be considered by composition of web services. Composing Web services by considering their agreement from functional, non-functional, behavioral, and semantic prospects in cooperation is a challenging issue. After all, awareness of these traits together increases the performance (Dewangan et al, 2019d) of the solution with increased level of user satisfaction and higher degree of automation.

EVALUATION OF APPROACHES

The paper (W. Ben Messaoud et al., 2016) discusses various approaches for web service composition. Further the paper provides a strong basis for creating reference models for web service composition. In the first part state of art is discussed and in the later part of the paper, framework design of the reference model is discussed.

The main purpose is to study systematic review of the literature done so far till 2015 and to develop a classification on QoS aware web service composition. Paper is a thorough analysis of different databases that focusses on meta- heuristic algorithms for QoS aware web service composition. Paper

Table 1. Comparitive analysis of services

Framework	OWL-S	WSMO	WSDL 2.0	SAWSDL	WS-Policy
W. Ben Messaoud et al., 2016					√
C. Jatoth et al., 2017	√		√		
P. Rodriguez-Mier et al., 2016					
R. Ben Lamine et al., 2017		√			
A. Ouni et al., 2017			√		
Q. Wu et al., 2016					
Q. Z. Sheng et al., 2014		√			
Choudhury, T., & Pasricha, A., 2020				√	
Choudhury, T., & Chandra Satapathy, S., 2020				√	
Y. Lu and X. Xu, 2017	√				
Dewangan, B. K. et al., 2012					√
Jain, A., Choudhury, T., 2020			√		
B. T. Kuehne et al., 2013	√				
J. Lee et al., 2015					√
Agarwal, A., & Pasricha, A., 2016		√			
P. Rodriguez-Mier et al., 2016					√
W. Ben Messaoud et al., 2016				√	

says QoS aware web service composition is NP- Hard and it needs to be solved in acceptable amount of time. After getting data from different research articles it's been observed that most widely used meta-heuristic techniques are Genetic algorithm (38%), PSO (22%) and ACO (10%) (Dewangan et al, 2021b). The paper saves time of various researchers in the field of web service composing by giving a thorough overview of web service composition.

The paper is graph based and it shows an approach to integrate the services or we can say it shows a different manner of composition of web services through discovery and matchmaking (Pasricha, A., & Agarwal, A., 2019).

Paper (R. Ben Lamine et al., 2017) uses semantic context-aware planning graph and proposes a semantic adaptable web services composition method. There are 3 stages used in composition.

- Construction of planning graph based on discovery and selection of atomic web services which takes users context and his preferences. Semantic relation between input and outputs of web service are used to enhance planning graph.
- Backward search is used in which they are extracting a set of best composed web services.
- Based on semantic and context scores, ranking is provided to set of extracted solutions.

R. Ben Lamine discusses about anti-patterns. It says while designing the web services there are certain quality principles which can get violated due to certain factors. Anti-patterns are the programming patterns that have bad design and practices (Singh et al, 2020). The anti-patterns lead

to various types of problems like maintenance, evolution, increased bug rate, inflexible code, fragile design etc.

The major objectives of the research paper are:

- To automate web service anti-pattern detection.
- To fully satisfy the quality principles.
- Reduce the manual effort.
- Reduce the chances of error-prone web-services.
- To provide a better algorithm with different levels of expertise to detect the anti-patterns.

A concept of generalized component service is proposed which expand the candidate space for selection of services for getting better effective solution. To solve optimization problem used backtracking algorithm and extended genetic algorithm. Article shows that the proposed approach outperforms the traditional one.

The major objective of the system proposed is the Self-adaptive exception handling architecture for services resource. Also important is the Runtime collaborative adjustment mechanism that has been designed to deal with requirements and context changes. The system has a lot of advantages when it comes to the allocation of resources and management of user requirements at runtime (Agarwal, A. et al., 2019). Self-adaptive Custom Service Resources Optimization: Custom resource provisioning, improvement is made on selection and search capabilities for the service provisioning acts. An active custom approach that would be personalized according to requirements fragments according to SOA would be made. Self-adaptive customs and service aggregation is implemented and managed. A restart and exception control monitor would hold the abnormal condition for user service. The efficiency of the system would be improved.

The problem discussed in the article is that relating to public accessibility of utilities and the functionalities provided by the government. The current measures are lacking behind a lot, especially in terms of scalability and agility. The objective of the article is to find the solution or rather give the solution of the aforementioned problem with the help of SOA and to solve the problems that the government faces in all countries. The article very beautifully illustrates the use of SOA and how it can be beneficial in public relations. The government should definitely go through this article and consider the possibilities of use of SOA in public utility management.

The problem stated in the article is: while creating the description of web services provided by the provider, we only include the functional and non-functional aspects ignoring the others. Therefore to get over this problem, the better way is to:

- To include functional description
- To include non-functional description
- To include data structure description
- To include technical aspects
- Describe in such a way that the user is able to make appropriate choice of the service.

This research paper (B. T. Kuehne et al., 2013) is based on this very problem. Here we are creating an Automatic Web service Composition architecture. The process of AWSC is to matchmaking among the services and then finding the best service to execute. The matchmaking services don't give the best results if the amount is big so for improvement this paper gives matchmaking frameworks which also can be customized according to user's need (Dewangan, B. K, 2019b). AWSC architecture is made up of the following: First there is AWSC service provider whose work is to get the request from users which will be semantic description based, it performs the matchmaking using repository. Second is Service Provider Repository, its main work is to find the candidate services for the Web service

composition. Third is XML flow writer which will convert the output format of used algorithms in standard xml format (S. Rawat and A. Sah, 2019). Fourth is service dispatcher it take cares of proper execution of web services. Web service Providers used to find the location of web servers. AWSC architecture will be compared based on aspects like Availability, reliability, cost, response time, and reputation etc. The main goal of this paper is to provide Automatic Web service Composition (AWSC) architecture for performance evaluation of Automatic Web service Composition in a real environment. It gives Service dispatcher which will be the perfect way to manage the web services composition.

This paper (J. Lee et al., 2015) presents a framework for composing different types of contents like web contents, RESTful, SOAP, OSGi services and Android Activities. BPEL engine with attached adapters is used to enable the direct invocation and composition of SOAP, RESTful and OSGi services with heterogeneous content types. A case study of smart living room system is shown to evaluate the proposed approach. He proposed approach shows less network traffic and turnaround time as compared to traditional systems available. A systematic integration of SOAP, Non-SOAP and Non-web services using extended BPEL engine is shown. Resource consumption is also less in this approach.

The paper has presented a hybrid algorithm for composing web services automatically (et al., 2017). Proposed approach has four main steps:

- First step is to create the composition graph for the set of the request.
- Second step is to find the optimal composition that have the minimal QoS .
- Multi-step optimizations technique to decrease the search space by finding the equivalent web services.
- Hybrid algorithm uses global-local search to find the optimal QoS with the minimum number of web services.

CONCLUSION AND RESULTS

According to literature survey, still there exist some research gaps that can be worked upon to improve the working of web services. These are:

- There are few web service composition methods that already exist, but there is still the scope of developing a better and more optimized solution for discovery, selection and composition of web services, it may be via mathematical models or by applying different algorithms or a combination of both. The proposed algorithm will improve the success ratio of compositing services. This will also add flexibility to the current scenario of composing services.
- Previously services are composed as fine grained service but nowadays use of coarse grained service is more in demand. So the research work can be done in composing coarse grained web services. Concept of Microservices is a recent and less researched topic. There is scope to work over this area.
- Another major research gap is to make service composition dynamic i.e. composition of services automatically at run time.
- Exception handling concept is not much researched on. Even Testing is major concern as testing can be done on larger and complex collection of web services. Fault tolerant web service composition is also an issue. It is even very important that services are self-healing and they can automatically detect violation of requirement and can also react to them.
- Demand for pervasive composition of services is also an open topic for research. Making services work with different technologies e.g. mobile phones, PDAs, radio frequency identification (RFID) technology, sensor technology etc. is also a concern since services run over these devices are usually resource constrained e.g. Limited battery life, limited memory etc. Extensive research is going on in this field.

Table 2. Year wise publication data of Web Service composition in Elsvier

(a) Composition of services

Year	Articles	Conference Papers	Book Chapters
2020	141	2	18
2019	460	9	31
2018	355	6	27
2017	299	8	28
2016	279	8	37
2015	248	12	44
2014	240	9	43
2013	200	16	45
2012	173	14	35
2011	146	12	26
2010	136	12	24
2009	114	7	22
2008	138	11	33
2007	134	11	36
2006	106	9	23
2005	87	10	30

(b) Microservices

Year	Articles	Conference Papers	Book Chapters
2020	41	0	18
2019	124	0	11
2018	81	0	14
2017	26	0	16
2016	12	1	14
2015	6	0	0
2011	2	0	0

(c) Composition of Microservices

Year	Articles	Conference Papers	Book Chapters
2020	40	0	18
2019	122	0	11
2018	81	0	14
2017	26	0	16
2016	12	1	14
2015	6	0	0

- Nonfunctional properties such as security are also an important concern for composing services. Security issues are arising more when we are dealing with composite web services (Agarwal, A., & Venkatadri, M., 2018).

After thoroughly studying and reviewing the material about services and microservices, the study shows the scenario of the articles, conference paper and book chapters published over Elsvier and the results are further shown in the Figures 3 and 4. The Table 2a shows number of articles published over Composition of service from the year 2005 to 2020. Next Table 2b shows the scenario of publication of articles over Microservices since it came into existence to till now. By visualizing the graph it is seen that very less articles are published in this domain and it the next upcoming research are in the field of SOA (Dewangan, B. K et al., 2018). Further Table 2c shows the number of articles published in Composition of microservices and also shows that very few articles are published in this domain. Further graphs are generated using these tables and are shown in Figure 3.

In next table and graphs the paper shows the scenario of the articles, conference papers and book chapters published/indexed over Web of Science and the results are further shown in Table 3 and Figure 4. The Table 3a shows number of articles published over Composition of service from the year 2005 to 2020. Next Table 3b shows the scenario of publication of articles over Microservices since it came into existence to till now. By visualizing the graph it is seen that very less articles are published in this domain and it the next upcoming research are in the field of SOA. Further Table 3c shows the number of articles published in Composition of microservices and also shows that very few articles are published in this domain. Further in Figure 4 graphs are generated using the data from the Table 2.

CONCLUSION

This research paper discuss the thorough study of the work done in the field of web service composition. This paperanalyze and explore in detail the current enterprise applications and utility of Web services in them. A detailed study of existing web service frameworks is shown year wise. Different evaluation approaches from different research articles are discussed in depth. It is observed that there is still the scope of developing a better and more optimized solution for discovery, selection and composition of web services, it may be via mathematical models or by applying different algorithms or a combination of both. Even the success ratio for compositing services can be enhanced. This will further add flexibility to the current scenario of composing services. Now a day's use of coarse grained service is more in demand. So the research work can be done in composing coarse grained web services. Microservices is a recent and very interesting for research. Other important component is Exception handling. Which should be focused and thought of. Demand for pervasive composition of services is also an open topic for research. Making services work with different technologies e.g. mobile phones, personal digital assistant (PDAs), radio frequency identification (RFID) technology, sensor technology etc. is also a concern. So in our next upcoming papers we will research and discuss these components.

Figure 3a. Research articles trends for Services composability in Elsvier

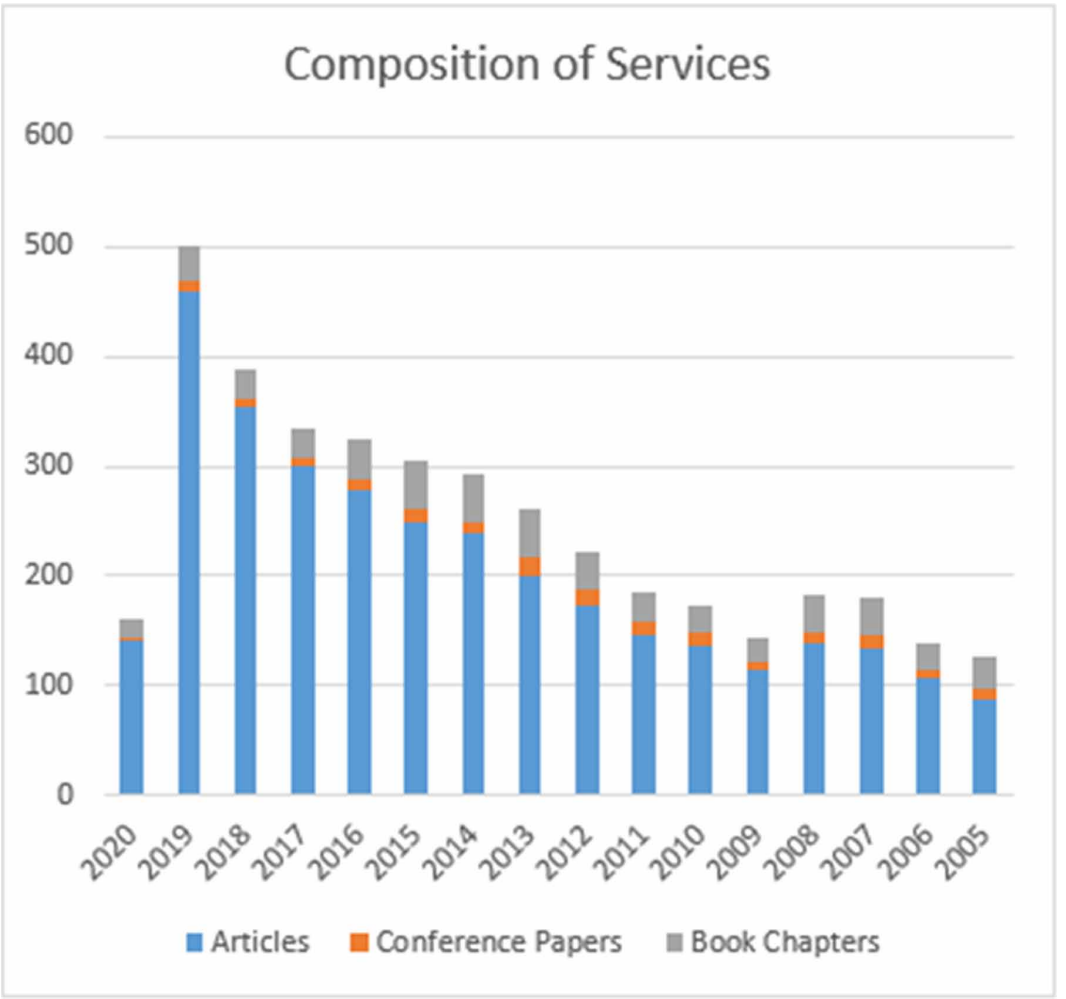


Figure 3b. Research articles trends for Services composability in Elsvier

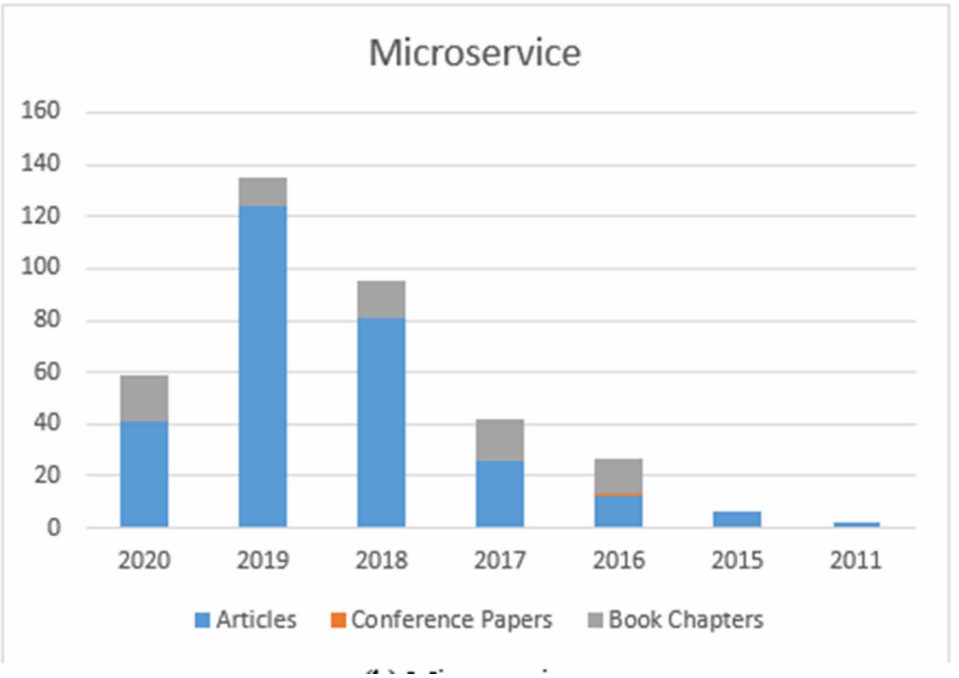


Figure 3c. Research articles trends for Services composability in Elsvier

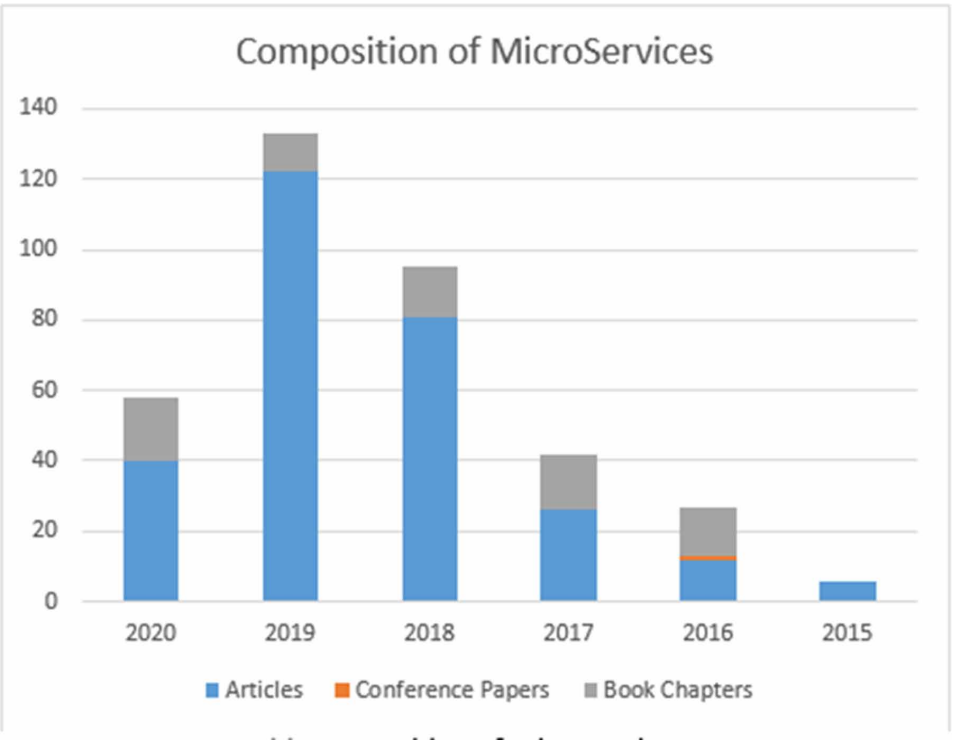


Table 3. Year wise publication data of Web Service composition in Web of Science

(a) Composition of services

Year	Articles	Conference Papers	Book Chapters
2020	5	0	0
2019	133	21	1
2018	114	56	1
2017	109	97	4
2016	144	164	4
2015	144	215	1
2014	167	193	5
2013	150	201	11
2012	157	175	12
2011	158	193	14
2010	110	195	17
2009	112	480	10
2008	101	399	5
2007	83	431	7
2006	160	338	0
2005	148	318	0

(b) Microservices

Year	Articles	Conference Papers	Book Chapters
2020	0	0	0
2019	5	96	1
2018	84	197	0
2017	51	139	1
2016	18	69	1
2015	12	17	0
2014	11	1	0
2013	0	0	0
2012	0	0	0
2011	0	1	0
2010	0	2	0
2009	0	2	0
2008	0	0	0
2007	0	0	0
2006	0	0	0
2005	0	1	0

continued on following page

Table 3. Continued

(c) Composition of Microservices

Year	Articles	Conference Papers	Book Chapters
2020	0	0	0
2019	3	4	0
2018	1	4	0
2017	2	4	0
2016	0	1	0
2015	0	0	0
2014	0	0	0
2013	0	0	0
2012	0	0	0
2011	0	0	0
2010	0	1	0

Figure 4a. Research articles trends for Services composability in Web of Science.

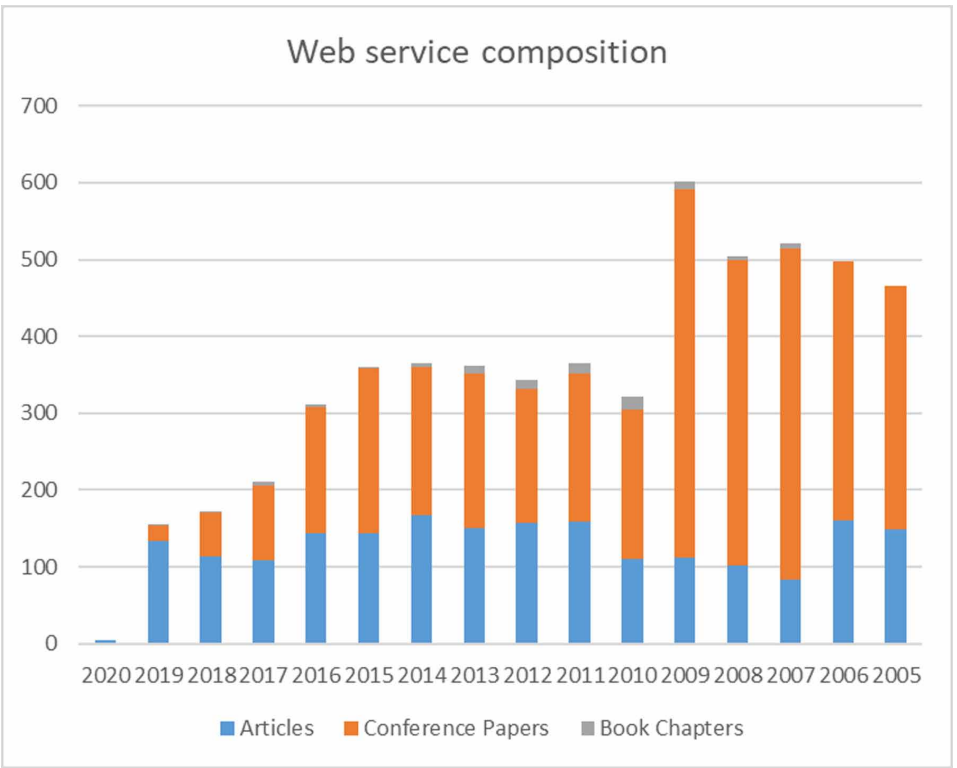


Figure 4b. Research articles trends for Services composability in Web of Science.

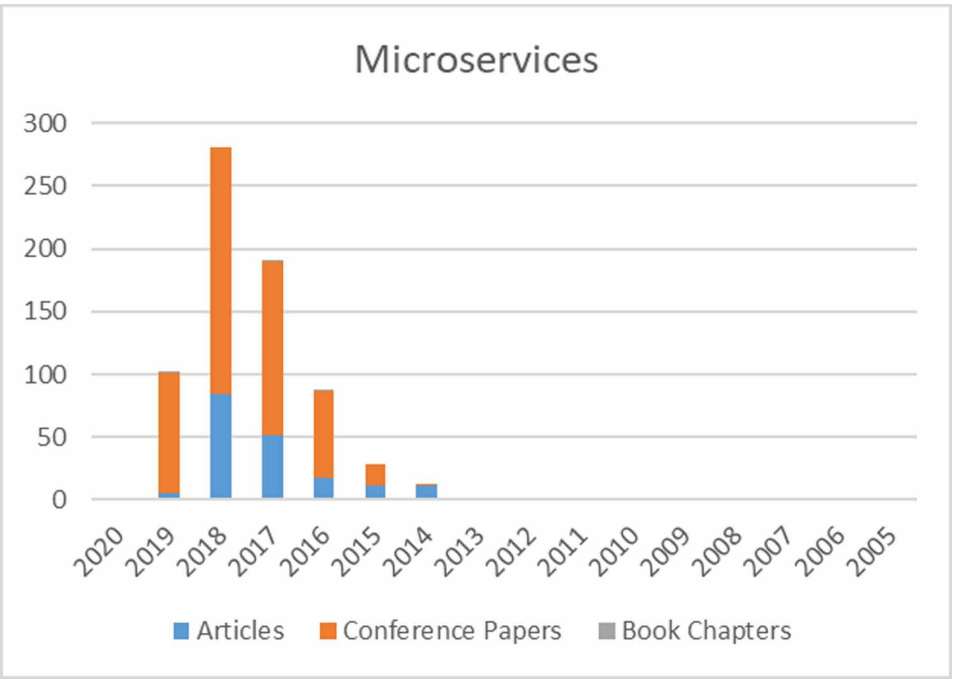
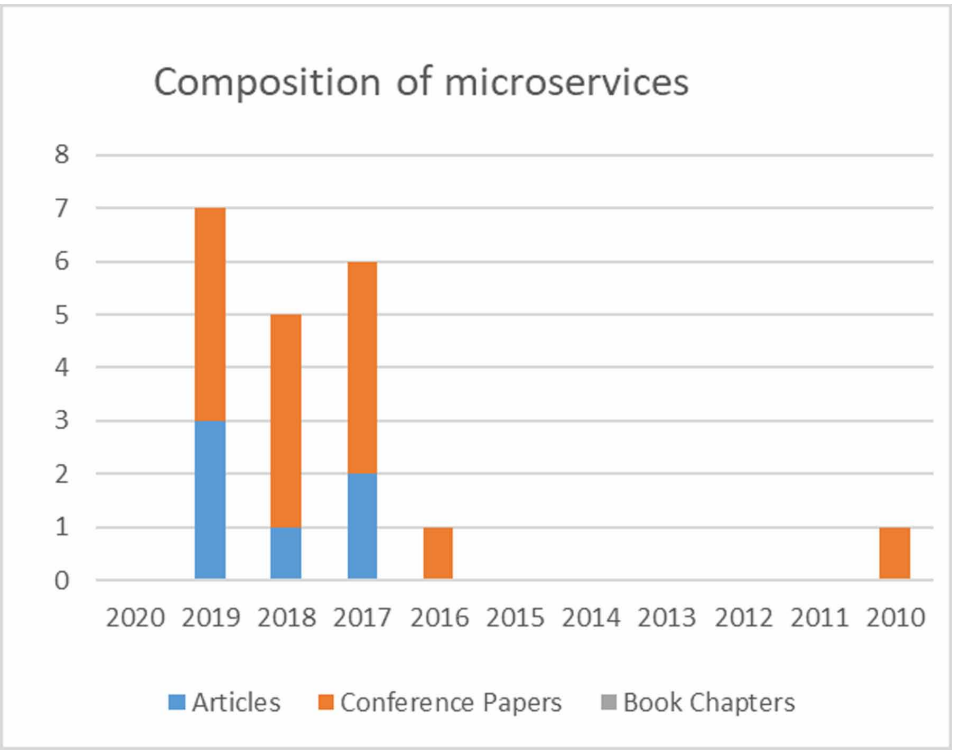


Figure 4c. Research articles trends for Services composability in Web of Science.



REFERENCES

Agarwal, A., & Pasricha, A. (2016). Credential and security issues of cloud service models. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)* (pp. 888-892). IEEE. doi:10.1109/NGCT.2016.7877536

Agarwal, A., & Venkatadri, M. (2018). Autonomic cloud resource management. In *2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC)* (pp. 138-143). IEEE. doi:10.1109/PDGC.2018.8745977

Agarwal, A., Venkatadri, M., & Pasricha, A. (2019). Energy-Aware Autonomic Resource Scheduling Framework for Cloud. International Journal of Mathematical. *Engineering and Management Sciences*, *4*(1), 41–55. doi:10.33889/IJMEMS.2019.4.1-004

Ben Lamine, R., Ben Jemaa, R., & Ben Amor, I. A. (2017). Graph Planning Based Composition for Adaptable Semantic Web Services. *Procedia Computer Science*, *112*, 358–368. doi:10.1016/j.procs.2017.08.016

Ben Messaoud, W., Ghedira, K., Ben Halima, Y., & Ben Ghezala, H. (2016). Survey of web service composition. *2015 5th Int. Conf. Inf. Commun. Technol. Access. ICTA 2015*.

Choudhury, T., & Chandra Satapathy, S. (2020). Extensive review of cloud resource management techniques in industry 4.0: Issue and challenges. *Software, Practice & Experience*, spe.2810. Advance online publication. doi:10.1002/spe.2810

Choudhury, T., & Pasricha, A. (2020). Cloud resource optimization system based on time and cost. International Journal of Mathematical. *Engineering and Management Sciences.*, *5*(4), 758–768. doi:10.33889/IJMEMS.2020.5.4.060

Ciavotta, M., Alge, M., Menato, S., Rovere, D., & Pedrazzoli, P. (2017). A Microservice-based Middleware for the Digital Factory. *Procedia Manufacturing*, *11*, 931–938. doi:10.1016/j.promfg.2017.07.197

Dewangan, B. k., Jain, A., Shukla, R. N., & Choudhury, T. (2021b). *An Ensemble of Bacterial Foraging, Genetic, Ant Colony and Particle Swarm Approach eB-GAP: A Load Balancing Approach in Cloud Computing. Recent Advances in Computer Science and Communications*. Bentham Science. doi:10.2174/2666255813666201218161955

Dewangan, B. K., & Agarwal, A., Venkatadri, & Pasricha, A. (2019b). A Self-Optimization Based Virtual Machine Scheduling to Workloads in Cloud Computing Environment. *International Journal of Engineering and Advanced Technology*, *8*(4), 91–96.

Dewangan, B. K., Agarwal, A., Venkatadri, M., & Pasricha, A. (2018). Resource scheduling in cloud: A comparative study. *International Journal on Computer Science and Engineering*, *6*(8), 168–173.

Dewangan, B. K., Agarwal, A., Venkatadri, M., & Pasricha, A. (2019c). Self-characteristics based energy-efficient resource scheduling for cloud. *Procedia Computer Science*, *152*, 204–211. doi:10.1016/j.procs.2019.05.044

Dewangan, B. K., Agarwal, A., Venkatadri, M., & Pasricha, A. (2019d). Macro: Cost-Oriented Autonomic Resource Management for Cloud Computing. *Vivekananda Journal of Research, 106*.

Dewangan, B. K., & Pasricha, A. (2019a). *Design of Self-Management Aware Autonomic Resource Scheduling Scheme in Cloud*. Academic Press.

Dewangan, B. K., & Shende, M. P. (2012). Survey on user behavior trust evaluation in cloud computing. International Journal of Science. *Engineering and Technology Research*, *1*(5), 113.

Dewangan, B. K., Venkatadri, M., Agarwal, A., Pasricha, A., & Choudhury, T. (2021a). An Automated Self-Healing Cloud Computing Framework for Resource Scheduling. *International Journal of Grid and High Performance Computing*, *13*(1), 47–64. doi:10.4018/IJGHPC.2021010103

Feng, C. M., Gao, Y. L., Liu, J. X., Zheng, C. H., & Yu, J. (2017). PCA Based on Graph Laplacian Regularization and P-Norm for Gene Selection and Clustering. *IEEE Transactions on Nanobioscience*, *16*(4), 257–265. doi:10.1109/TNB.2017.2690365 PMID:28371780

Jain, A., & Choudhury, T. (2020). GAP: Hybrid task scheduling algorithm for cloud. *Revue d'Intelligence Artificielle*, *34*(4), 479–485. doi:10.18280/ria.340413

Jatoth, C., Gangadharan, G. R., & Buyya, R. (2017). Computational Intelligence Based QoS-Aware Web Service Composition: A Systematic Literature Review. *IEEE Transactions on Services Computing*, *10*(3), 475–492. doi:10.1109/TSC.2015.2473840

Kuehne, B. T., Santana, R. H. C., Linnemann, V., & Santana, M. J. (2013). Performance evaluation of an automatic web service composition architecture. *Proc. 2013 Int. Conf. High Perform. Comput. Simulation, HPCS 2013*, 123–130. doi:10.1109/HPCSim.2013.6641402

Lee, J., Lee, S. J., & Wang, P. F. (2016). Lightweight approach for multi-objective web service composition. *IET Software*, *10*(4), 116–124.

Li, B., Ji, S., Qiu, D., Leung, H., & Zhang, G. (2013). Verifying the concurrent properties in BPEL based web service composition process. *IEEE eTransactions on Network and Service Management*, *10*(4), 410–424. doi:10.1109/TNSM.2013.111113.120379

Lu, Y., & Xu, X. (2017). A semantic web-based framework for service composition in a cloud manufacturing environment. *Journal of Manufacturing Systems*, *42*, 69–81. doi:10.1016/j.jmsy.2016.11.004

Omrana, H., El Bitar, I., Belouadha, F. Z., & Roudiès, O. (2013). A comparative evaluation of web services description approaches. *Proc. 2013 10th Int. Conf. Inf. Technol. New Gener. ITNG 2013*, 60–64. doi:10.1109/ITNG.2013.17

Ouni, A., Kessentini, M., Inoue, K., & Cinnéide, M. (2017). Search-Based Web Service Antipatterns Detection. *IEEE Transactions on Services Computing*, *10*(4), 603–617. doi:10.1109/TSC.2015.2502595

Pasricha, A., & Agarwal, A. (2019). Self-characteristics based energy-efficient resource scheduling for cloud. *Procedia Computer Science*, *152*, 204–211. doi:10.1016/j.procs.2019.05.044

Phu, P. H. (2006). *Quality of Service for Interoperable Transactions in Service-Oriented Architecture* (Doctoral dissertation).

Rawat, S., & Sah, A. (2012). An approach to Enhance the software and services of Health care centre. Academic Press.

Rodríguez-Mier, P., Mucientes, M., & Lama, M. (2017). Hybrid Optimization Algorithm for Large-Scale QoS-Aware Service Composition. *IEEE Transactions on Services Computing*, *10*(4), 547–559. doi:10.1109/TSC.2015.2480396

Rodriguez-Mier, P., Pedrinaci, C., Lama, M., & Mucientes, M. (2016). An integrated semantic web service discovery and composition framework. *IEEE Transactions on Services Computing*, *9*(4), 537–550. doi:10.1109/TSC.2015.2402679

Sah, A., Bhadula, S. J., Dumka, A., & Rawat, S. (2018). *A software engineering perspective for development of enterprise applications*. Handb. Res. Contemp. Perspect. Web-Based Syst. doi:10.4018/978-1-5225-5384-7.ch001

Sah, A., Dumka, A., & Rawat, S. (2018). *Web technology systems integration using SOA and web services*. Handb. Res. Contemp. Perspect. Web-Based Syst. doi:10.4018/978-1-5225-5384-7.ch002

Sah, A., Rawat, S., & Pundir, S. (2012). Design, Implementation and Integration of Heterogeneous Applications. *International Journal of Computers and Applications*, *54*(5), 11–16. doi:10.5120/8561-2157

Sheng, Q. Z., Qiao, X., Vasilakos, A. V., Szabo, C., Bourne, S., & Xu, X. (2014). Web services composition: A decade's overview. *Inf. Sci. (Ny)*, *280*, 218–238. doi:10.1016/j.ins.2014.04.054

Sheng, Q. Z., Qiao, X., Vasilakos, A. V., Szabo, C., Bourne, S., & Xu, X. (2014). Web services composition: A decade's overview. *Information Sciences*, *280*, 218–238. doi:10.1016/j.ins.2014.04.054

Singh, I., Sabitha, S., Choudhury, T., Aggarwal, A., & Dewangan, B. K. (2020). Mapping Student Performance With Employment Using Fuzzy C-Means. *International Journal of Information System Modeling and Design*, *11*(4), 36–52. doi:10.4018/IJISMD.2020100103

Truong, H.-L., & Klein, P. (2020, March). DevOps Contract for Assuring Execution of IoT Microservices in the Edge. *Internet of Things*, *9*, 100150. doi:10.1016/j.iot.2019.100150

Wu, Q., Ishikawa, F., Zhu, Q., & Shin, D. H. (2016). QoS-Aware Multigranularity Service Composition: Modeling and Optimization. *IEEE Transactions on Systems, Man, and Cybernetics. Systems*, *46*(11), 1565–1577. doi:10.1109/TSMC.2015.2503384

Xia, C., Zhang, Y., Wang, L., Coleman, S., & Liu, Y. (2018, December). Microservice-based cloud robotics system for intelligent space. *Robotics and Autonomous Systems*, *110*, 139–150. doi:10.1016/j.robot.2018.10.001

Yu, Q., Chen, L., Li, B., & Li, J. (2015). Ant colony optimization applied to web service compositions in cloud computing. *Computers & Electrical Engineering*, *41*(C), 18–27. doi:10.1016/j.compeleceng.2014.12.004

Anushree Sah is a highly experienced IT professional having an experience of more than 13 years in the field of IT industry and education. The author has worked with the renowned companies like Oracle Financial Services & Software Ltd., Western Union, Dencare Ltd., DIT University, UPES etc. The author has completed her bachelor's in Computer Science and Engineering and has Master's degree from University of Greenwich, London, U.K. She holds various academic and administrative responsibilities in her current working place. The author specializes in Programming Languages, Web Technologies, Building Enterprise Application, Service Oriented Computing and Cloud Computing. She has several research papers, conference proceedings, Book Chapters and Project.

Tanupriya Choudhury received his bachelor's degree in CSE from West Bengal University of Technology, Kolkata, India, and master's Degree in CSE from Dr. M.G.R University, Chennai, India.He has received his PhD degree in the year 2016. He has Ten years experience in teaching as well as in Research. Currently he is working as an Associate Professor in dept. of CSE at UPES Dehradun. Recently he has received Global Outreach Education Award for Excellence in best Young researcher Award in GOECA 2018 His areas of interests include Human computing,Soft computing,Cloud computing,Data Mining etc.He has filed 14 patents till date and received 16 copyrights from MHRD for his own softwares.He has been associated with many conferences in India and abroad.He has authored more than 85 research papers till date.He has delivered invited talk and guest lecture in Jamia Millia Islamia University , Maharaja Agersen college of Delhi University, Duy Tan University Vietnam etc. He has been associated with many conferences throughout India as TPC member and session chair etc. He is a lifetime member of IETA, member of IEEE, and member of IET(UK) and other renowned technical societies. He is associated with Corporate and he is Technical Adviser of DeetyaSoft Pvt. Ltd. Noida, IVRGURU, and Mydigital360 etc.He is holding the post of Secretary in IETA (Indian Engineering Teacher's Association-India), He is also holding the Advisor Position in INDO-UK Confederation of Science, Technology and Research Ltd.,London, UK and International Association of Professional and Fellow Engineers-Delaware-USA.

Bhupesh Kumar Dewangan pursued Bachelor of Technology form Pt Ravi Shankar Shukla University (State University), Raipur and Master of Technology from Chhattisgarh Swami Vivekananda Technical University (State Technical University), Bhilai in computer science and engineering. Currently he is pursuing PhD in computer science and engineering and Assistant Professor, Department of Informatics, at University of Petroleum and Energy Studies. He has more than 40 research publications in various international journals and conferences with SCI/ SCOPUS/UGC indexing. His research interests in Autonomic Cloud Computing, Resource Scheduling, Software Engineering, and Testing. He is a member of various organizations like ISTE, IAPFE, etc.