Journal of Network and Computer Applications 81 (2017) 24–36

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Comprehensive and systematic review of the service composition mechanisms in the cloud environments

Asrin Vakili, Nima Jafari Navimipour*

Department of Computer Engineering, Tabriz Branch, Islamic Azad University, Tabriz, Iran

ARTICLE INFO

Keywords:
Cloud computing
Service
Composition
Review
Systematic

ABSTRACT

Typically, cloud computing includes the provisioning of dynamically scalable and virtualized resources as services over the Internet. In the cloud environment, based on the user's needs, various types of services can be delivered that often must be composited to meet the user requests. Therefore, service composition is emerging as a universal technology in order to integrate distributed and heterogeneous services to combine and consolidate the cloud services. This idea focuses on the innovation of a new cloud service including previously existing services for cost and time reducing and efficiency improving. However, to the best of our knowledge, despite the importance of this matter in cloud environments, there doesn’t exist any comprehensive and systematic research and survey in this field. Therefore, the purpose of this paper is to survey the existing techniques and mechanisms which can be addressed in this domain. Briefly, the contributions of this paper are: (1) providing an overview of the existing challenges in a range of problem domains associated with cloud service composition, (2) providing an anatomy of some important techniques throughout scope of cloud service composition and (3) outlining key areas for the improvement of service composition methods in the future research.

1. Introduction

Nowadays, through the Internet and web services (Pooranian et al., 2014) information, data transferring has been facilitated (Navimipour and Zareie, 2015). Cloud computing as the web and Internet-based computing model encompasses the provisioning of dynamically scalable and virtualized resources as services (Bastia et al., 2015; Milani and Navimipour, 2016a, 2016b). Some of the distinguishing characteristics of cloud computing are fast services configuration, elasticity, and scalability (Zhang et al., 2010). It enables an access to remotely configurable computing resources and on-demand hardware and software services providing for minimizing the human efforts needed by the customer as well as improving the service maintaining cost (Ashouraie and Jafari Navimipour, 2015). A cloud service follows pay-as-you-go fashion meaning that customers are merely charged for the time they spend on the service (Chiregi and Navimipour, 2016). Services in cloud computing can be categorized into application and computing services (Armbrust et al., 2009). Regarding the kind of services provided, a cloud might have the form of infrastructure as a service (IaaS), platform as a service (PaaS), software as a service (SaaS) (Milani and Navimipour, 2016a, 2016b; Serrano et al., 2015) and expert as a service (EaaS) (Hazratzadeh and Jafari Navimipour, 2017; Jafari Navimipour et al., 2015; Navimipour, 2015; Navimipour et al., 2015; Navin et al., 2014). It is currently being used to deal with challenging problems in different application domains, including industry, science, and government (Candelaa et al., 2014).

In the cloud environment, many types of services need to be provided depending on the user's needs (Jeong et al., 2015). In many applications, the request can be satisfied by integrating and combining some resources called service composition. Service composition as an NP-hard optimization problem (Canfora et al., 2005; Rao and Su, 2005), refers to a larger service providing by services integrating processes (Armbrust et al., 2010). Service composition is emerging as a universal technology in order to integrate some services over the Internet which are distributed and heterogeneous in order to consolidate business applications throughout organization boundaries (Tout et al., 2015). A cloud architecture permits service (Pooranian et al., 2015) composition to answer the users’ complicated requests in order to improve the accessibility and flexibility of provided services (Xie et al., 2014). This idea focuses on the offering the new cloud services from previously existing services for cost and time reducing and efficiency improving (Kurdi et al., 2015). Several tasks are usually included in a service composition process each of which corresponds to a service class including many candidate services with the same

* Corresponding author.
E-mail address: jafari@iaut.ac.ir (N.J. Navimipour).

http://dx.doi.org/10.1016/j.jnca.2017.01.005
Received 20 May 2016; Received in revised form 1 January 2017; Accepted 4 January 2017
Available online 05 January 2017
1084-8045/

Table 1
Abbreviation table.

Abbreviation	Definition
ACO-WSC	Ant Colony Optimized- Web Service Composition
ATS	Applied Technical Services
BPEL4WS	Business Process Execution Language FOR Web Service
COM2	Combinatorial Optimization algorithm for cloud COMposition
CSCOS	Cloud Service Composition Optimal-Selection
CSSICA	Classified Search Space Imperialist Competitive Algorithm
DCS	Dynamic Cloud Service selection
DAML-S	DARPA agent markup language for services
ICA	Imperialist Competitive Algorithm
MAS	Multi-Agents Self-organizing
OWL-S	Web Ontology Language for Services
PROCLUS	PROjected CLUStering
QoS	Quality of Service
RQs	Research Questions
SR-CNP	Semi-Recursive Contract Net Protocol
STOCCSC	Service Time Optimization in Cloud Computing Service Composition
SLA	Service-Level Agreement
SOAP	Simple Object Access Protocol
WPC	Windows Performance Counters
WCF	Windows Communication Foundation

functions and different QoS constraints when using a composite service. However, the usage of service's resources and the request of composite service have a strong variability.

On the other hand, to the best of our knowledge, despite the importance of service composition mechanism in the cloud environments, there is not any systematic survey and review about the service composition mechanisms in cloud computing that realizes the need for researchers to do more work on service composition. Therefore, the purpose of this paper is to survey and review the existing service composition techniques in cloud computing comprehensively and systematically. Briefly, the contributions to this paper are as follow:

- Providing an overview of the existing challenges in the range of problem domains associated with cloud service composition.
- Providing a systematic study and overview of the existing techniques for service composition, service selection and other actions that need to integrate the cloud services.
- Providing an anatomy of various pivotal techniques within the scope of cloud service composition.
- Outlining key areas for improving the service composition methods in future research.

The structure of this paper is organized as follows. After the introduction, backgrounds of cloud service composition are provided in Section 2. The related work is analyzed and reviewed in Section 3. The research terminologies and prepares selection mechanisms are provided in Section 4. The selected service composition mechanisms in three main categories are described in Section 5. The taxonomy and comparison of the reviewed mechanisms are presented in Section 6. Also, Section 7 maps out same open issues. Finally, Section 8 concludes the paper. Moreover, Table 1 shows the commonly used abbreviation in the paper.

2. Backgrounds

Cloud computing is an emerging technology that is increasingly being adopted by large and small enterprises in order to attain the top-line development and progress by deriving value from data whereas, at the same time, decreasing the cost of running their IT (Austel et al., 2015). The cloud is changing the manner of data processing and information sharing. For instance, data can be kept on local workplaces and file servers for fast access but face the challenge of sharing it with a large number of people. But, in cloud environments, data can be put into one or more cloud storage systems in order to facilitate their sharing with different other users (Nelson and Peterson, 2013). However, because of fast growth of the number of cloud services, a large number of candidate compositions that would use different services might be used to respond the same query (Benouaret et al., 2014). Also, many types of research aimed at automatic service composition have been included in the literature which can be divided into two diverse categories (Lécué et al., 2008) including those focusing on functional aspects (Klusch et al., 2005) and approaches focusing on process service aspects (Berardi et al., 2003). Service composition can be subdivided into three main steps (Henni and Atmani, 2012) including creation of the process model specifying control and data flow among the activities; discovery, selection and binding of concrete services to every activity in the process model; and execution of the composite service by a coordinating entity (Sivasubramanian et al., 2009). Several research efforts have been done which aimed at providing platforms and languages for service composition (Kaklanis et al., 2016). These efforts can be classified into three main categories including workflow-based approaches (Rao and Su, 2004); XML-based approaches, such as BPEL4WS (Andrews et al., 2003); and ontology-based approaches, such as OWL-S (Burstein et al., 2004) and DAML-S (Sycara et al., 2003).

On the other hand, existing research in QoS representation can have the following categorization: single values representation, multiple values representation, and standard statistical distributions (Zheng et al., 2016). In many mechanisms, each QoS metric is considered as a single value able to be the max, min, or mean value of a QoS. Single values are utilized in QoS-driven service selection or composition approaches. QoS which is represented as a constant value cannot reflect the quality variation. The uselessness of single value representation of service QoS has been recognized, and for representing service QoS standard statistical distributions have been adopted. As a result, QoS need to be viewed as stochastic and service composition problem can be regarded as a decision problem under uncertainty. The response time and prices are modeled as independent, and beta-distributed random variables. The most likely, most pessimistic, and most optimistic values of a service are used to determine the beta distribution. In particular, the QoS metrics are modeled as different distribution functions. In real cases, the QoS distribution of a service can be in any shape, and a well-known statistical distribution is not capable of reflecting the irregularly shaped distribution precisely. The QoS can be obtained through QoS monitoring in three strategies for depending on where the measurement is taken places:

- *Client-side monitoring:* The measurement of QoS is performed on the client side. QoS metric that depends on user experience, such as response time, can be measured on the client side. Response time is measured recording the time gap between a client receiving and sending out a Simple Object Access Protocol (SOAP) message.
- *Server-side monitoring:* The measurement of QoS is done on the server side. This technique needs access to the server for the actual service implementation, not always applicable in practice. Windows Performance Counters (WPC) of the Windows Communication Foundation (WCF) can be used to perform the server-side QoS monitoring.
- *Third-party monitoring:* The measurement of QoS is performed by a third party. The outputs of the QoS monitoring of a service will be periodically provided by the third party. User-independent QoS (e.g. price) is usually identical for different users while user-dependent QoS (e.g. failure probability, response time, throughput, etc.) can widely vary for various users as a result of the unpredictable Internet connections and the heterogeneous user environments.

Moreover, service composition in cloud environments tries to choose and interconnect offered services by different service providers on the basis of a specified business process. As mentioned, the business

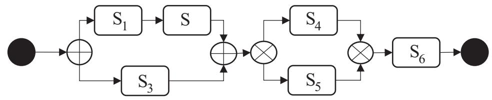

Fig. 1. An example of simplified workflow.

process of the composite service can be fully represented by a workflow language for web services, through which the potential data dependency between tasks is defined (Wu and Zhu, 2013). The most commonly-used composition constructs for service orchestration include Sequence, Flow, Switch, and Loop. For the peer sake of facilitating and clarifying statement later, we denote them by symbols $\rightarrow$, $\oplus$, $\otimes$, and $*$, respectively. Fig. 1 illustrates a simplified workflow representing a service composition example. By using the construct symbols, this workflow can be formally described in an algebraic notation as: $[[S_1 \rightarrow S_2] \oplus S_3] \rightarrow [S_4 \otimes S_5] \rightarrow\ S_6$.

Also, each cloud service has some QoS factors for service evaluating. The mentioned factors will be evaluated in the total possible compounds of cloud services. Five important factors have been defined as follows:

- *Time:* Time is the interval from a user submitting a request to the server and responding the request. Typically it is measured in milliseconds as a time unit.
- *Cost:* The payment of a specified sum of money to order the action that it has needed to do.
- *Scalability:* The capacity to be altered and reformed in various conditions in a cloud environment.
- *Optimization:* The process of finding the best or most effective service combination by applying the appropriate methods.
- *Efficiency:* The ratio of the mechanism to the total cost and time taken.

3. Related work

Some research about service composition in the cloud environment has been carried out. This section will refer to some review papers that discussed the cloud service composition and outlines their main advantageous and disadvantageous.

A review of web service composition has been presented by Sheng et al. (2014) in order to study the service composition in two classes: automated service composition and semi-automated service composition. However, only web service composition was included and the searching methodology and parameters of QoS were not discussed especially in cloud environments.

One of the important studies of the cloud service composition has been carried out by Jula et al. (2014). In their survey, various QoS parameters such as latency, trust, cost, reliability, and availability were analyzed. Also, a detailed classification was included based on different parameters which are depending on the analysis of the existing techniques. Furthermore, they classified articles written up to 2013 into four classes based on their techniques: classic and graph-based methods, combinatorial methods, machine-based methods and frameworks methods. However, there is a gap for discussion in open issue, papers selection mechanism and recently published papers.

Also, in Lemos et al. (2016), the cloud service composition issues including language, knowledge reuse, automation, tool support, execution platform and target users have been evaluated. The issue has been discussed and some articles have been reviewed. However, there exist some gaps for discussing the open issue, service composition modeling, analyzing the QoS parameters and papers selection mechanism.

Finally, reviewing the service composition mechanism in the single and multi-cloud environment have been done by Asghari and Navimipour (2016). But, in this paper, a few mechanisms are investigated and the article searching and selecting methods are not defined. Article categorization is based on usage environment. The single-cloud and real time environments are discarded and only the multi-cloud environments are considered. This paper doesn't contain published papers in 2016 and only explained the brief methods of a few papers. Also, the papers have not been checked based on quality of service parameters. Briefly, the previous review papers suffer from some weakness as follows:

1. The papers don't contain the new proposed mechanisms especially in 2016.
2. The papers don't have the systematic structure, therefore, the article selection method is unclear.
3. Some papers don't investigate the QoS parameters for reviewing the methods.
4. Some papers don't implicate to the service composition method.
5. Many papers do not provide any logical categorization.

The mentioned reasons motivated us to prepare a survey paper that covers all of these deficiencies.

4. Research methodology

In order to have a clear picture of the service composition mechanism in cloud environments, this section provides a systematic literature review (SLR) of service composition mechanism with a specific focus on researches related to cloud environments. An SLR aims at providing a complete and exhaustive summary of the current literature relevant to the research domains (Aznoli and Navimipour, 2016a, 2016b). The first step in conducting a systematic review is to perform a thorough search of the literature for relevant papers (Navimipour and Charband, 2016; Soltani and Navimipour, 2016). The methodology section of a systematic review will list all of the databases and citation indexes that were searched such as Springer, IEEE, and Science Direct and any individual journal that was searched. The titles and abstracts of identified articles were checked for their eligibility and relevance investigated in form of our issue. This set will be related back to the research problem. The research questions are formalized in Section 4.1 and the article selection process is discussed in Section 4.2.

4.1. Question formalization

The present research aims at collecting and investigating all of the credible and effective studies that have examined cloud service composition. More specifically, the extraction of salient features and methods of papers will be considered, and their characteristics will be described. In order to achieve the above-mentioned goals and identify the methods that have been selected by researchers for their studies and result assessment methods, case studies are covered by research questions (RQs) that have been presented in following lines:

1. RQ 1: What is the cloud service composition? And what are the open issues?
   This question was answered in Section 2 and the open issue will be presented in Sections 6 and 7.
2. RQ 2: How is the article searching and selecting to evaluate?
   This question will be answered in Section 4.2.
3. RQ 3: What classification of research methods can be used? And what are examples of it?
   This question will be answered in Section 5.
4. RQ 4: How have researcher conducted the research?
   This question will be answered in Sections 5.1–5.3.
5. RQ 5: What parameters are accounted for?
   This question was answered in Section 2.

4.2. Article selection process

The process of choosing the articles for a systematic literature review is conducted in four stages, including automated search based on keywords; article selection based on the title, abstract, and quality of the publication; and publication and reference analysis (Charband and Navimipour, 2016).

4.2.1. Stage 1: Automated search based on keywords

The primary goal of the search process is to identify journal articles of service composition in the cloud environment with focusing on mechanism and acceptance factors. The search process is conducted via electronic searching on online scientific databases. Therefore, first, we identify electronic databases to find an article for which the following famous databases were used:

- Google scholar (https://scholar.google.com/)
- Springer (http://link.springer.com/)
- IEEE explorer (http://ieeexplore.ieee.org/)
- Science Direct (http://www.sciencedirect.com/)
- Sage (http://online.sagepub.com/)
- Taylor (http://www.tandfonline.com/)
- ACM (http://www.acm.org/)
- Scientific (http://www.scientific.net/)
- Emerald (http://www.emeraldinsight.com/)

By adding synonyms and alternative spellings of the main elements, the following search string was defined:

- "Cloud" AND ("service" OR "resource") AND ("combination" OR "composition" OR "integration")
- ("service" OR "resource") AND ("combination" OR "composition" OR "integration")
- "Cloud" AND ("combination" OR "composition" OR "integration")

4.2.2. Stage 2: Article selection based on the title, abstract, and quality of the publisher

We found 942 articles at searching articles in Stage 1. This stage begins with the selection of certain practical screening criteria to ensure that just high-quality publications and articles are included in the review (Reim et al., 2015). The search string was limited by searching at most for journal articles as they obtain validated empirical results. Therefore, all other types of studies were excluded in the initial search. Other search limitations had not been applied. With all databases, a structurally and semantically uniform search string was used, although in some cases it had to be adapted in order to fulfill the syntax requirements of the given database search engine. During the first search, working paper, revolution editorial note commentaries and book review articles were excluded, the main aim to be a focus on quality publications (Seuring and Müller, 2008).

By means of this strategy, we have found 245 articles which are shown in Fig. 2 where 3% of the articles are related to Sage, 2% are related to Emerald, 5% are related to Taylor, 4% are related to Scientific, 16% are related to Springer, 11% are related to ACM, 17% are related to Science Direct and 36% are related to IEEE conference articles and 7% are related to IEEE journal articles. Also, Fig. 3 illustrates the number of published journal articles between 2003 and 2016.

4.2.3. Stage 3: Publication and relevant analysis

We found 245 articles at searching articles in Stage 2. As it is seen in Fig. 4, 105 conference articles, 14 review articles, and 28 articles which were published before 2012 were removed. At the end, the proposed method in each article has been investigated and 98 articles were selected that their method is directly related to the service composition. Table 2 shows details of the selected articles such as publication year, journal, and authors.

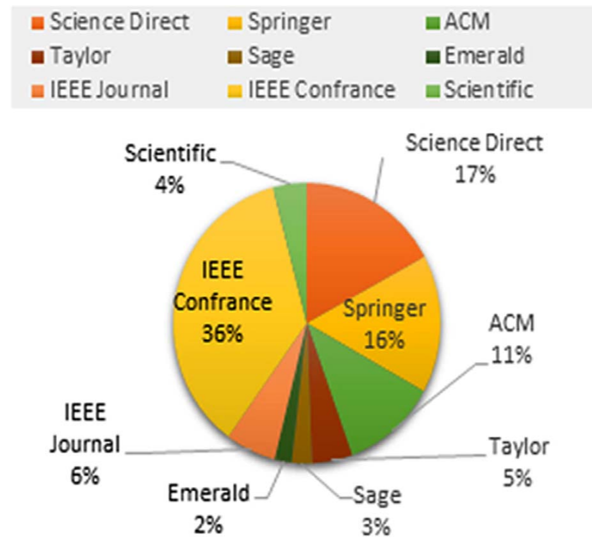

Fig. 2. Percentage of published articles in any publication.

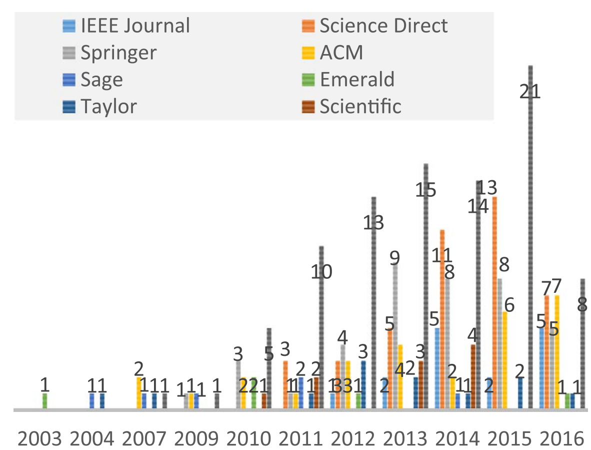

Fig. 3. Number of articles between 2003–2016.

4.2.4. Stage 4: Final evaluation

In this stage, the full body of the selected papers from the previous stage are examined for finding the appropriate papers for review. We select the paper that:

1) Explained proposed method obviously and clearly,
2) determined the research goals and QoS parameters,
3) defined the fitness function clearly in heuristic cases,
4) Provided the comparison with state-of-the-art methods, and
5) Provided and explained the dataset clearly.

The reason for selecting these criteria is that reviewing the well-written articles can help and boost the researchers to do the future works mindfully. This stage results in selection 20 article where are indicated in the last column of Table 2.

5. Review of the selected service composition mechanisms

In this section, 20 selected article according to the mentioned criteria will be reviewed. Also, their techniques basic properties as well as their differences, advantageous and disadvantageous will be discussed and described. The approaches in the literature can be divided into 3 distinct categories including framework-based, agent-based and heuristic-based. Classification of methods and its definition are illustrated in Fig. 5. In Sections 5.1–5.3, these methods and their examples are provided. Also, the pivotal factors to design efficient

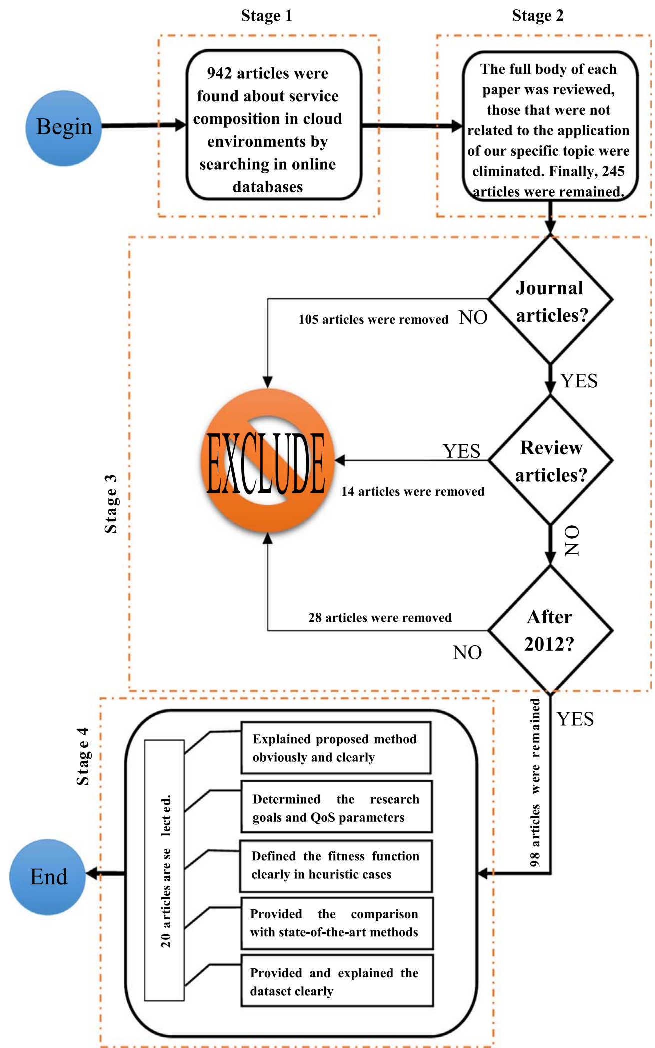

Fig. 4. Filtering method for articles selection.

and applicable cloud service composition techniques and their explanations are illustrated in Section 2.

On the other hand, according to the goal of the mechanism, many QoS parameters were determined. The following description is given some important goal. Exactly, considering the qualitative QoS parameters and using them in decision making is important to transform them into quantitative values using reliable methods. Calculating a QoS value for composite services requires a mathematical model in which all aspects, parameters, user requirements, and tendencies are investigated to improve the past algorithms and specify them to obtain the best solutions or reduce the execution time. In some cases, to achieve this goal, there could be a need to design a new framework. Possessing appropriate and well-specified data structures and databases can play a main role in the design of an efficient algorithm. Using an appropriate indexing method is also helpful in increasing the search speed, especially when the number of cases is very high. One of the main invoices that entice customers and maintains them in utilizing cloud computing is reliability. Since providing a trustworthy and self-adaptable service combination is very important, the most important part of a cloud is in its direct interaction with its users. Encouraging service providers to expose their high-quality services depends on the ability to collect significant profits. If the service composer policy is not to register services based on predefined desires, then it must discover required available simple services in the network. It is critical to have the type of rule that uses optimal discovery methods.

Finally, the investigation of article simulation result and comparing them with mechanism goals to identify which parameters are improved and which one is attenuated. "Low" and "high" are used to shows the parameters (time, cost, efficiency, optimization, scalability) improvement or weakening. These parameters were explained clearly in Section 2.

5.1. Framework-based mechanisms

In this section, first the framework-based cloud service composition mechanism and their basic properties are described in Section 5.1.1. Then 7 framework-based cloud service composition mechanisms are discussed in Section 5.1.2. Finally, their differences, advantageous and disadvantageous are discussed and compared in Section 5.1.3.

5.1.1. Overview of the framework-based mechanism

The framework-based mechanisms are based on a set of assumptions, concepts, values, and practices that constitute and form a way of viewing reality. The framework-based mechanisms are utilized in order to organize and manage for searching, selecting and cloud services composition by completely novel approaches. In Section 5.1.2 some framework-based mechanisms will be shown.

5.1.2. Overview of the selected framework-based mechanisms

Fan et al. (2013) have proposed a method in order to screen the attributes of the basic component of service composition for the description of their interrelationship. The special features of the proposed model include: first, a service composition reliable net is defined as a unified formalism to describe different components of service composition. In the modeling process, the transaction attributes, reliability and failure processing mechanisms are taken into account thus it can be precisely characterized for service composition. Second, a reliable composition strategy and its enforcement algorithm are proposed, which is used to dynamically allocate the available service meeting the required transaction attributes for the task based on the required ATS of the service consumer. The redundancy technique is also used to ensure the reliability of service composition mechanism. Third, the operational semantics and related theories of Petri nets help prove the effectiveness and efficiency of reliable composition strategies. It has been shown that the proposed reliable composition strategies are in line with the application with the abundant service, resource or high success probability, and the reliability of service composition will be affected by the number of invoking services. Therefore, this framework has main advantages such as increasing scalability and efficiency, while it can only be used for SaaS services.

Also, Zhang et al. (2014) have proposed mathematically combined model and the corresponding solution algorithm through analyzing the characteristics of services resource combination in cloud manufacturing. With the purpose of avoiding problems of uncertainty, coarse-grained, diversity and dynamism in the process of services resource combination, a hierarchical model based on the hierarchical manufacturing implementation processes was suggested. Afterward, the QoS has been chosen so as to evaluate the effects of services combination. Finally, an annealing algorithm was developed to solve the proposed model. The obtained results have shown that this approach has high efficiency and optimization, whereas it suffers from low scalability.

The problem of cloud service composition optimal selection in cloud manufacturing has been investigated by Huang et al. (2014a, 2014b). They have established the categories of cloud services and their QoS indexes and based on the perspective of QoS indexes, the relationship among QoS key factors are analyzed and elaborated for different kinds of cloud services, and the corresponding objective functions and constraints of CSCOS are proposed. A new chaos control algorithm is designed to address the CSCOS problem, and it was demonstrated through the simulation results that the algorithm can search better solutions with less time consumption than widely used

Table 2
Details of the selected articles.

Publisher	Year	Journal Name	Author	Selected?
IEEE	2012	Services Computing, IEEE Transactions on	(Paik, Chen, and Huhns)	No
	2013	Parallel and Distributed Systems, IEEE Transactions on	(Dou, Zhang, Liu, and Chen)	No
		Knowledge and Data Engineering, IEEE Transactions on	(Qi Yu and Bouguettaya)	No
	2014	Cloud Computing, IEEE Transactions on	(Dastjerdi and Buyya)	No
		Cloud Computing, IEEE Transactions on	(Tanaka and Murakami)	No
		Services Computing, IEEE Transactions on	(Klein, Ishikawa, and Honiden)	Yes
		Systems Journal, IEEE	(Wu, Dou, Hu, and Chen)	No
		Services Computing, IEEE Transactions on	(Ye, Mistry, Bouguettaya, and Dong)	No
	2015	Services Computing, IEEE Transactions on	(Zhang, Hwang, Khan, and Malluhi)	No
		Services Computing, IEEE Transactions on	(Mostafa and Zhang)	Yes
	2016	Transaction On Services Computing	(Fernandez, Tedeschi, and Priol)	No
		IEEE Transactions on Cloud Computing	(Wang, Zhou, Yang, and Chang)	No
		Journal of Systems Engineering and Electronics	(Zhang, Sun, Tang, and Yuan)	No
		IEEE Transactions on Cloud Computing	(Chen, Huang, Lin, and Shen)	No
Science Direct	2012	Future Generation Computer Systems	(Liu, Wang, Shen, Luo, and Yan)	No
		The Journal of Systems and Software	(Monsieur, Snoeck, and Lemahieu)	No
	2013	Journal of Network and Computer Applications	(Alvaro and Barros)	No
		Future Generation Computer Systems	(Wu and Zhu)	No
		The Journal of Systems and Software	(Fan, Yu, Chen, and Liu)	Yes
	2014	J. Parallel Distrib. Comput.	(Furno and Zimeo)	No
		Information Sciences	(Geebelen et al.)	No
		Computer Networks	(Han, Kim, and Kim)	No
		Fourth International Conference on Selected Topics in Mobile and Wireless Networking	(Mascitti, Conti, Passarella, and Ricci)	No
		Future Generation Computer Systems	(Chen et al., 2016)	No
		International Workshop on Software Defined Networks for a New Generation of Applications and Services	(Stanik, Koerner, and Lymberopoulos)	No
		Expert Systems with Applications	(Ordonez, Alcázar, Corrales, and Falcarin)	No
		Journal of Network and Computer Applications	(Huang, Liu, and Duan)	No
		Expert Systems with Applications	(Huang et al.)	No
	2015	Journal of King Saud University Computer and Information Sciences	(Singh et al., 2015)	Yes
		Future Generation Computer Systems	(Fan, Hussain, Younas, and Hussain)	No
		Expert Systems with Applications	(Jula, Othman, and Sundararajan)	Yes
		2nd International Symposium on Big Data and Cloud Computing	(Bharathi, Vijayakumar, and Pradeep)	No
		Computers and Electrical Engineering	(Tout et al.)	No
		Information Sciences	(Zhao, Shen, Peng, and Zhao)	Yes
		Computers and Electrical Engineering	(Kurdi et al.)	Yes
		Dynamic cloud service selection using an adaptive learning mechanism in multi-cloud computing	(Wang, Cao, and Xiang)	Yes
		Computers and Electrical Engineering	(Qiang Yu, Chen, and Li)	Yes
		Expert Systems with Applications	(Vladimir, Budiselić, and Srbljić)	No
		The Journal of Systems and Software	(Huang and Shen)	No
	2016	Information Sciences	(Wang, Wang, Hu, Zhang, and Gu)	Yes
		Science of Computer Programming	(Walther and Wehrheim)	No
		Future Generation Computer Systems	(Sun, Ma, Zhang, Dong, and Hussain)	No
		Knowledge-Based Systems	(Liu, Chu, Jia, Shen, and Wang)	No
		Optik - International Journal for Light and Electron Optics	(Jian, Zhu, and Xia)	No
Springer	2012	Service-Oriented Computing - ICSOC 2011 Workshops	(Kofler, ul Haq, and Schikuta)	No
		Service-Oriented Computing	(Wittern, Kuhlenkamp, and Menzel)	No
		Communications in Computer and Information Science	(Li, Xinrui, and Xinyu)	No
		Service-Oriented and Cloud Computing	(Demchenko et al.)	No
	2013	Industrial Applications of Holonic and Multi-Agent Systems	(Benmerzoug, Gharzouli, and Zerari)	Yes
		Applied Intelligence	(Gutierrez-Garcia and Sim)	Yes
		Service-Oriented Computing - ICSOC 2012 Workshops	(Kano)	No
		Web Information Systems Engineering – WISE 2013	(Liang, Du, and Li)	No
		Knowledge Discovery, Knowledge Engineering, and Knowledge Management	(Serrano, Shi, Foghlú, and Donnelly)	No
		Quality, Reliability, Security and Robustness in Heterogeneous Networks	(Singh, Vyas, and Varma)	No
		Mobile Networks and Applications	(Wang, Sun, Zou, and Yang)	No
		Central European Journal of Operations Research	(Xiang, Hu, Yu, and Wu)	No
		Service-Oriented Computing	(Ye, Bouguettaya, and Zhou)	No
	2014	Service-Oriented Computing	(Ivanović and Carro)	Yes
		Journal of Grid Computing	(García and Blanquer)	No
		Intelligent Computing Methodologies	(Han, He, Li, and Huang)	No
		Service-Oriented Computing	(Neiat, Bouguettaya, Sellis, and Dong)	No
		Service-Oriented Computing	(Wenge, Schuller, Lampe, Siebenhaar, and Steinmetz)	No
		High-Performance Cloud Auditing and Applications	(Xie et al.)	No
	2015	Intelligent Computing, Communication, and Devices	(Bastia et al.)	Yes
		LISS 2014	(Hu and Zhang)	No
		Applied Intelligence	(Huo, Zhuang, Gu, Ni, and Xue)	Yes
		Telecommunication Systems	(Jeong et al.)	No
		Internet of Things. User-Centric IoT	(Kholidy, Hassan, Sarhan, Erradi, and Abdelwahed)	Yes
		Advances in Service-Oriented and Cloud Computing	(Kiran and Simons)	No
		Technological Innovation for Cloud-Based Engineering Systems	(Pisching, Junqueira, Santos Filho, and Miyagi)	No

(continued on next page)

Table 2 *(continued)*

Publisher	Year	Journal Name	Author	Selected?
	2016	Soft Computing	(Di Martino, Cretella, and Esposito)	No
		Frontiers of Computer Science	(Lin, Hu, and Zhang)	No
		The Journal of Supercomputing	(Karimi, Isazadeh, and Rahmani)	Yes
		The International Journal of Advanced Manufacturing Technology	(Liu and Zhang)	No
		The International Journal of Advanced Manufacturing Technology	(Zhou and Yao)	No
Taylor	2012	International Journal of Computer Integrated Manufacturing	(Guo, Tao, Zhang, Laili, and Liu)	No
		Enterprise Information Systems	(Tao, Guo, Zhang, and Cheng)	No
	2013	International Journal of Computer Integrated Manufacturing	(Liu, Liu, Sun, Li, and Ma)	No
		International Journal of Production Research	(Zhang, Zhang, Chen, and Pan)	No
	2014	Enterprise Information Systems	(Huang, Li, and Tao)	Yes
	2015	International Journal of Production Research	(Lartigau, Xu, Nie, and Zhan)	Yes
		International Journal of Computer Integrated Manufacturing	(Tzafilkou, Protogeros, and Koumpis)	No
Scientific	2013	Instruments, Measurement, Electronics	(Wei, Yang, Zhou, and Zheng)	No
	2014	Applied Decisions in Area of Mechanical	(Liu, Li, Zhang, and Liu)	No
		Material, Mechanical and Manufacturing Engineering	(Zhang and Xu)	No
		Intelligent Materials and Mechatronics	(Zhang, Guo, and Geng)	Yes
		Applied Science, Materials Science, and Information.	(Zhong, Zhu, Huang, and Xin)	No
Sage	2014	Journal of Algorithms and Computational Technology	(Cai and Cui)	No
Emerald	2012	International Journal of Web Information Systems	(Paganelli, Ambra, and Parlanti)	No
	2016	International Journal of Web Information Systems	(Garg, Modi, Chaudhary, Taniar, and Pardede)	No
ACM	2012	Transactions on the Web (TWEB)	(Alrifai, Risse, and Nejdl)	No
	2013	ACM Transactions on Internet Technology (TOIT)	(Li, Madnick, and Zhu)	No
		Transactions on the Web (TWEB)	(Weber, Paik, and Benatallah)	No
		Managing Technical Debt (MTD)	(Alzaghoul and Bahsoon)	No
	2014	Transactions on Internet Technology (TOIT)	(Ye, Bouguettaya, and Zhou)	No
		Transactions on Internet Technology (TOIT)	(Benouaret et al.)	No
	2016	Transactions on Internet Technology (TOIT)	(Amato and Venticinque)	No
		Transactions on the Web (TWEB)	(Zheng et al.)	No

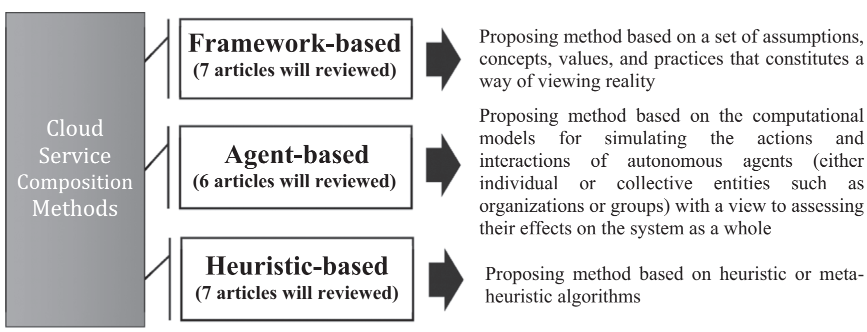

Fig. 5. Classification of the cloud service composition method.

algorithms such as genetic algorithm and typical chaotic genetic algorithm.

Also, a method has been presented by Jula et al. (2015) to apply the PROCLUS algorithm in order to classify cloud service providers which have led to the adoption of a more realistic approach in generating the first generation of solutions of the ICA and creating the CSSICA algorithm. Based on service time values of all provided single services, the classifier has divided service providers into three subcategories called high recommended, recommended and low-recommended providers for selecting a service. The probability values are also calculated based on average service time values of all service providers being assigned to each class. To avoid the hasty decision in determining the weakest empire, 15 iterations are intended as an opportunity time in order to increase the power prior to entering into imperialist competition. Experimental and statistical evaluation of the results revealed that clustering the cloud service pool plays a fundamental role in reaching more appropriate solutions for STOCCSC. Also, CSSICA is capable of reducing the provided composite service time. Furthermore, the CSSICA achieved the most optimal results in solving different-sized problems, and these results indicate that CSSICA is a scalable and efficient algorithm for finding optimal composite services.

Also, Kurdi et al. (2015) have proposed a combinatorial optimization algorithm for cloud service composition that can efficiently utilize multiple clouds. The algorithm ensures that the cloud with the maximum number of services will always be selected before other ones, which increases the possibility of fulfilling service requests with minimal overhead. The COM2 competes successfully with previous multiple cloud service composition algorithms via examining a small number of services which directly relate to execution time without compromising the number of combined clouds. In addition, this mechanism has a number of advantages such as reducing the time and high efficiency, whereas high cost is what suffers it.

Mostafa and Zhang (2015) have proposed two multi-objective approaches to handling QoS-aware service composition with conflicting aims and different limitations on the quality matrices. The first approach is related to the single policy multi-objective composition scenarios while the second one relates to the multiple policy multi-objective composition scenarios. To deal with the uncertainty characteristics inherent in cloud environments, the proposed approaches use reinforcement learning. The simulation results have shown the capabilities of the proposed approaches to efficiently compose services based on multiple QoS criteria in cloud environments. However, it suffers from high cost and complexity.

Finally, Wang et al. (2015) have proposed a cloud service selection model which adopts the cloud service brokers, and a dynamic cloud service selection strategy called DCS. During the process of selecting

Table 3
Side-by-side comparison of the cloud service composition in framework-based mechanisms.

Technique	Approach	Advantage	Weakness
(Huang et al.)	Chaos controls optimal algorithm	1. Low time 2. High optimization	1. High complexity
(Jula et al.)	Classified search space imperialist competitive algorithm	1. Low time 2. High optimization 3. High scalability	1. High complexity
(Kurdi et al.)	Combinatorial optimization algorithm for cloud composition	1. Low time 2. High efficiency	1. High cost
(Wang et al.)	Dynamic cloud service selection	1. Low cost 2. High efficiency	1. High time
(Fan et al.)	SaaS personalization framework for service composition	1. High scalability 2. High efficiency	1. High time
(Mostafa and Zhang)	Multi-objective approach	1. High optimization 2. High scalability	1. High cost 2. High complexity
(Zhang et al.)	Mathematical combined model	1. High efficiency 2. High optimization	1. Low scalability

services, each cloud service broker manages some clustered cloud services and performs the DCS strategy with an adaptive learning mechanism as its core including the incentive, forgetting and degenerate functions. The mechanism is devised to dynamically optimize the cloud service selection and return the most optimal service result to the user. This strategy has better overall efficiency in acquiring high-quality service solutions at a lower computing cost, but, it suffers from the engagement between service functionaries and the time.

5.1.3. Summary of framework-based mechanisms

The framework-based mechanisms for service composition are utilized in order to organize and manage of searching, selecting and composing the cloud services via applicable approaches. A side-by-side comparison of the opted techniques as well as their main advantage and weakness are shown in Table 3.

5.2. Agent-based mechanism

In this section, we first describe the agent-based mechanisms for service composition and their basic properties in Section 5.2.1. Second, we discuss the 6 selected agent-based mechanisms in Section 5.2.2. Finally, their differences, advantages, and disadvantages are discussed and compared in Section 5.2.3.

5.2.1. Overview of agent-based mechanism

The agent-based mechanism which is based on the computational models for simulating the actions and interactions of autonomous agents (either individual or collective entities such as organizations or groups) has a view to assessing their impacts on the system as a whole. In Section 5.2.2, the selected agent-based mechanisms and their anatomy are discussed.

5.2.2. Review of the selected agent-based mechanism

Gutierrez-Garcia and Sim (2013) have presented an agent-based approach in order to compose services in multi-cloud environments for different classes of cloud services. Agents are endowed with a semi-recursive contract net protocol and service capability tables (information catalogs about cloud participants) in order to compose services which are based on consumer requirements. Its findings showed that successfully compose services to satisfy service requirements, autonomously select services based on dynamic fees, effectively cope with constantly changing consumers' service needs that trigger updates, and compose services in multiple clouds even with incomplete information about cloud participants. The advantage of the proposed mechanism is high efficiency and scalability, whereas the disadvantage of this approach is a high time of service composition.

Also, Benmerzoug et al. (2013) have proposed a basis for a theoretical approach for aggregation protocols in order to create a new desired business application. In this approach, the elements are used to specify exchanges of messages between various business partners. Afterward, agents use this specification to enact the integration of business processes at run time. Agents have the capability of forming social structures dynamically through which the commitments to the common goal are shared. Through their coordinated interactions the individual agents achieve globally coherent behavior and act as a collective entity known as a multi-agent system. The proposed mechanism has high scalability and efficiency, but, it suffers from high run time.

Ivanović and Carro (2014) have presented an approach to ensure the scalability of service compositions (which focuses on orchestrations with centralized control flow) – having rich control structure (involving branches, loops and parallel flows), state and data operations – by translating them in a network of actor behaviors behaving correctly with respect to the semantics of the composition specification. Such a network can be instantiated and automatically scaled up/out by the underlying actor platform, the remitting and clustering capabilities of which facilitate deployment in the cloud. The experimental results revealed that the composition can be easily scaled to match the elasticity of the external services and to yield significant efficiency improvements. The same mechanism of monitoring can be utilized to test the composition of a fine-grained level against pre- and post-conditions on the composition as a whole and an individual construct it is built from, and to compute code and path coverage of a test suite. Additionally, the scheme can easily be adapted for simulation of service behavior against different load scenarios. This mechanism suffers from the high time of the service composition.

An agent based automated service composition algorithm has also been proposed by Singh et al. (2015) comprising of request processing and automated service composition phases which is not only responsible for searching comprehensive services but also considers reducing the cost of virtual machines only consumed by on-demand services. In this study, an intelligent and automated assignment strategy has been presented for assigning resources in cloud computing environment. In this mechanism, several intelligent agents have been deployed for the reduction of system complexity by modularization. Broker agent facilitates the search for the optimal data center for the requirements of each user and service composition on user behalf so that a contract is established between two entities. Thus, the proposed mechanism contributes to eliminating user challenge of finding an optimal service provider in any condition and ensures efficient service allocation at the data centers. However, it suffers from high run time.

Bastia et al. (2015) have proposed a multi-agent-based approach in multi-cloud environments for different types of cloud services. In this study, cloud participants and resources are implemented and instantiated by agents. Previously, they presented self-organizing agents who make use of services capability table and the semi-recursive contract net protocol (SR-CNP) so that cloud service compositions be evolved and adapted. They have altered some of the agents' behaviors. As a result, we can reduce the number of passing message by half in order to increase the overall efficiency. Also, they have planned a 2-layered (3 levels of multi-agents) self-organizing (MAS) so that a cloud service composition is established. The obtained results have revealed that this mechanism has high efficiency and scalability, but it suffers from high overhead for agents.

Finally, Wang et al. (2016a, 2016b) have proposed a method that describes a multi-agent reinforcement learning model for the dynamic optimization of web service composition. In this model, the agent can utilize reinforcement learning algorithms so as to have interaction with the environment in real time in order to compute optimal composition strategy dynamically, and multi-agents mechanism can keep higher effectiveness in contrast to single-agent reinforcement learning. They have proposed a distributed Q-learning algorithm, decomposing the task into many sub-tasks and making every agent focus on its own sub-task, so that the convergence rate is accelerated. In addition, they also introduced an experience sharing strategy to improve the efficiency. As a result, based on these methods composite service is allowed to dynamically adjust itself to fit a varying environment, where the properties of the component services continue to change. This mechanism has high efficiency and scalability, but it suffers from high complexity.

5.2.3. Summary of the reviewed agent-based mechanism

The agent-based mechanisms are discussed and reviewed in the previous subsection. As reviewed, Ivanović and Carro (2014), Gutierrez-Garcia and Sim (2013), Benmerzoug et al. (2013) and Singh et al. (2015) suffer from high time of service composition and Bastia et al. (2015) suffers from high messages sending between agents. The important improved using these approaches is scalability. A side-by-side comparison of the reviewed agent-based mechanisms besides their main advantages and weaknesses are shown in Table 4.

5.3. Heuristic-based mechanism

In this section, we will first describe the heuristic-based mechanisms and their basic properties in Section 5.3.1. Second, we will discuss 7 selected heuristic-based mechanisms in Section 5.3.2. Finally, their differences, advantages, and disadvantages are discussed and compared in Section 5.3.3.

5.3.1. Overview of heuristic-based mechanism

The heuristic-based mechanisms are methods based on heuristic or meta-heuristic mechanisms such as ant colony, genetic algorithms, bee colony, gray wolf and etc. The selected heuristic-based mechanisms and their anatomy are discussed in next section.

5.3.2. Review of the selected heuristic-based mechanism

Klein et al. (2014) have described the self-adaptive network-aware approach to service composition. They proposed an approach which applies a realistic network model, accurately estimates the network QoS, and employs a self-adaptive genetic algorithm which is very efficient at optimizing latency, but able to adapt to optimize any kind of other QoS, as well. They build a network model in order to estimate the network latency between arbitrary services and potential users. Their selection algorithm then leverages this model to find a suitable composition with a low latency for a given execution policy. They have employed a self-adaptive genetic algorithm which balances the optimization of latency and other QoS as needed and improves the convergence speed. The proposed mechanism has two main advantages such as low time of service composition and high optimization but it suffers from low scalability.

Yu et al. (2015) have defined and studied the web service composition problem in a multi-cloud based environment. They have proposed two diverse cloud combination algorithms: Greedy-WSC and ACO-WSC. The greedy algorithm repeatedly chooses the cloud that can provide the most requested services until the selected clouds cover all the requirements. In the ACO-WSC algorithm, artificial ants travel on a logical digraph in order to construct cloud combinations. In the logical graph, a cloud base is represented by each node and edges connect each pair of nodes. Each ant chooses its path based on pheromone and heuristic information on the edges of the path. The ant colony iteratively obtains the optimal solution. Their proposed method can efficiently and effectively find high-quality service composition plans with a minimum number of clouds. The proposed method achieves a superior tradeoff between time and quality and it is a practical solution for using multi-cloud service provision environments. However, these approaches suffer from high complexity.

A systematic approach has been presented by Zhao et al. (2015) based on a fuzzy preference model and evolutionary algorithms. Specifically, they have modeled the problem based on the weighted distance rather than a linear utility function, then they presented a fuzzy preference model for preference representation and weight assignment. In the model, a set of fuzzy linguistic preference terms and their properties are introduced for establishing consistent preference order of multiple QoS dimensions, and a weighting procedure is suggested in order to transform the preference into numeric weights. Finally, two evolutionary algorithms, i.e, single evolutionary algorithm and hybrid evolutionary algorithm are introduced, which implement different optimization objectives being able to be used in different SLA management scenarios for service composition. The proposed mechanism has three main advantages i.e. lower time of service composition, high efficiency, and optimization, but suffers low scalability.

Kholidy et al. (2015) have discussed the cloud service composition problem as a multi-objective optimization so that the requirements of user's QoS is satisfied, and have presented an approach for solving the multi-objective problem by modifying it to a single objective problem. This approach used the genetic algorithm to solve the composition problem. In addition, QoS attributes of each individual are considered in the composition pattern and corresponding aggregate functions. Then, the composition system makes decisions on which SaaS and IaaS providers should be selected for the end user. The final objective finds a cloud service composition to minimize the cost and time and improve the throughput. In addition, a comparison is made between the

Table 4
Side-by-side comparison of the agent-based mechanisms in cloud service composition.

Technique	Approach	Advantage	weakness
(Singh et al.)	Agent-based	1. High scalability 2. Low cost	1. High time
(Benmerzoug et al.)	Agent-based	1. High scalability 2. High optimization	1. High time
(Ivanović and Carro)	Focusing on orchestrations with centralized control flow	1. High scalability 2. High optimization	1. High time
(Bastia et al.)	Multi-agent-based	1. High scalability 2. High efficiency	1. High overhead
(Gutierrez-Garcia and Sim)	Agent-based	1. High scalability 2. High efficiency	1. High time
(Wang et al.)	Multi-agent reinforcement learning model	1. High efficiency 2. High scalability	1. High complexity

Table 5
Side-by-side comparison of the cloud service composition using heuristic mechanisms.

Technique	Approach	Advantage	weakness
(Yu et al.)	Greedy ant colony optimized	1. Low time 2. High optimization	1. High complexity
(Zhao et al.)	Fuzzy preference model and evolutionary algorithms	1. Low time 2. High optimization	1. Low scalability
(Kholidy et al.)	Genetic Algorithm	1. High optimization 2. High efficiency 3. Low cost	1. High time
(Klein et al.)	Genetic algorithm	1. Low time 2. High optimization	1. Low scalability
(Huo et al.)	Bee colony	1. High scalability 2. Hi optimization	1. High cost 2. High time
(Lartigau et al.)	Bee colony	1. High efficiency 2. High scalability 3. High optimization	1. High time
(Karimi et al.)	Data mining techniques and genetic algorithm	1. High scalability 2. High efficiency 3. Low time	1. High overhead

proposed approach and other existing algorithms such as Integer Linear Programming. The obtained results have clarified that the proposed approach provides high efficiency and optimization but it suffers from low fitness value for tournament data and high time.

Huo et al. (2015) have proposed a mechanism based on bee colony algorithm. Time attenuation function is added into the service composition model in order to increase the weights of the recent scores, thus, the comprehensive evaluation value of services can describe the variation of the service quality in time. Additionally, they have applied the artificial bee colony algorithm for the service composition problem and uses the exploration of bees for food to simulate the search of the optimal service composition solution. In addition, they have improved the food source encoding, the generating strategy of candidate solutions and the local search strategy, and proposes the discrete best-guided artificial bee colony algorithm to solve the nonlinear integer programming problem. The discrete best-guided artificial bee colony algorithm has advantages in terms of the quality of solution and efficiency, especially for the large-scale data. But is suffers from high cost.

In addition, Lartigau et al. (2015) have proposed a method based on QoS evaluation along with the geo-perspective correlation from one cloud service to another for transportation impact analysis. Since the composition is an exhaustive process in terms of computational time consumption, the proposed method is optimized through an adapted Artificial Bee Colony algorithm based on initialization enhancement. This study takes common aspects of cloud services into consideration such as quality of service parameters but extends the scope to the physical location of the manufacturing resources. Unlike the classic service composition, manufacturing brings additional constraints. The mechanism has high time as a problem but has high efficiency, scalability, and optimization.

Finally, Karimi et al. (2016) have identified the requirements of service composition based on SLA contract in cloud computing environments. In as much as services have been remarkably enhanced in a cloud environment and customers are concerned with the speed of requested service delivery as well as service quality, researchers in this study used data mining techniques such as clustering, association rules in service composition and genetic algorithm for reducing the search space of a problem. Based on the findings of the study, it can be concluded that these techniques can result in the reduction of service composition time and the enhancement of the optimality of compound services. Furthermore, the results of the study indicated that the proposed method is scalable; hence, it is highly appropriate for dynamic cloud environments. But it suffers from high overhead.

5.3.3. Summary of heuristic-based mechanisms of service composition

The selected heuristic-based mechanisms are discussed in the previous section. The important factor that has increased with all of the heuristic-based mechanisms is efficiency. Side-by-side comparison, advantages, and disadvantages of the discussed methods are showed in Table 5.

6. Results and comparison

In the previous sections, we described most popular cloud service composition techniques in three main categories: framework-based, agent-based, and heuristic-based. The framework-based mechanism has high flexibility, so it is suitable for any cloud environment as a single-cloud, multi-cloud and real-time cloud. It can involve more advantages because it uses two or more algorithms in the combination. The agent-based mechanism is also suitable for the multi-cloud environment because it has high quality for task management. Also, the agent-based mechanism has high scalability compared to other mechanisms. The heuristic-based mechanism is suitable for a single cloud environment and it focuses on composition optimization and efficiency. Also, a heuristic-based and an agent-based mechanism suffer from high composition time but the framework-based mechanism has low composition time. Furthermore, the framework-based mechanism has high flexibility to composite the cloud services because it can be used for the exploitation of two or more algorithm advantage.

Selected articles that were shown in Table 2 have checked about QoS parameters. The results are presented in Fig. 6. As shown in Fig. 6, researchers were focused on QoS parameters as efficiency is 30%, optimization is 28%, time is 24%, scalability is 13% and the cost is 5%. These results show that efficiency, optimization and time are in the center of attention. Scalability as an important parameter must be focused more than now in the future. One of the most important goals of service composition is reducing the cost. For this purpose, sometimes directly focus on reducing cost not necessary. But, researchers have to look out for cost parameter when another parameter will be impressed.

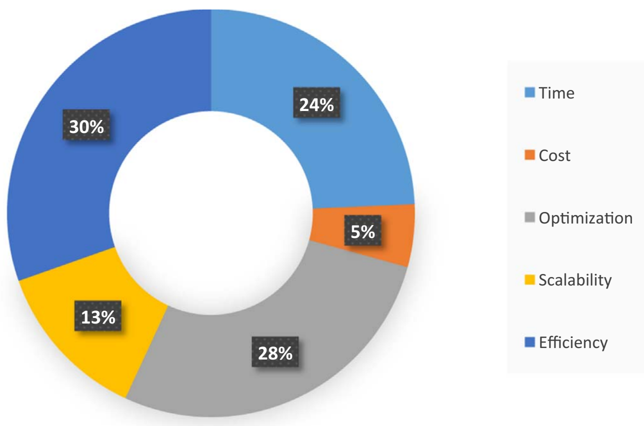

Fig. 6. Considered QoS parameters in the selected articles.

7. Open issues

This section offers various important issues and challenges that are necessary for future work.

1. Important issues and challenges in cloud service composition implementation:

It can be clearly seen that researchers have mainly considered simulator based tools for evaluations. Therefore, implementing the discussed approaches in the real-world experimentation is very interesting in the future. In addition, we believe that the reviewed papers, with its theoretical framework and practical implementation, are also interesting in the industry.

In services composition, it is likely that many changes arise after deployment. For instance, partner services may go down or get updated, and even new policies might be added to govern the composition. To deal with these changes, today's cloud environments are challenged by the need for continuous adaptation of processes. Yet, like other existing service process composition approaches, automated support is not provided for this end. In this context, many existing approaches leverage aspect-oriented programming to support service process execution language with the needed adaptability.

2. Important issues and challenges in QoS parameters

It has been observed that there is not a single mechanism to address all QoS parameters for composite services. For instance, some mechanisms consider scalability, optimization and combination time while another parameter such as cost, efficiency and etc. was ignored. Also, the study of how decompose the global QoS constraints adaptively and predict web service QoS are very interesting. Also, proposing a QoS adaptive prediction model that can schedule the most appropriate QoS prediction method according to the real-time situation is another research direction for future.

3. Important issues and challenges in normalization

It interested to future work to verify and prevent conflicts between aspects at the specification clouds level, a priori to their integration. Also, another extension is to promote cloud computing services towards global standardization. Common standards can be investigated for applications. Integration includes the mash up services in the web, grid, and clouds, which appear as a grid of clouds or cloud of clouds. In addition, it's very important to consider more optimized and flexible solutions, develop a personalization standard with multiple-level solutions for a personalization framework, adapt this framework with IaaS, investigate the security issues of the framework, and study the impact of the relationship among multiple quality objectives.

4. Service composition mechanisms can be used in other environments

Finally, in the future, the reviewed service composition mechanisms can be used in some similar environments as software defined network (Meirosu and John, 2013), social networks (Alamir et al., 2016; Mohammad Aghdam and Jafari Navimipour, 2016; Sharif et al., 2013), vehicular cloud services (Cordeschi et al., 2015), wireless sensor network (Abdollahzadeh and Navimipour, 2016; Aznoli and Navimipour, 2016b; Naranjo et al., 2016), mobile cloud computing (Fernando et al., 2013), vehicular networking (Shojafar et al., 2016), Peer-to-Peer networks (Asghari and Navimipour, 2017; Navimipour and Milani, 2015), and Big data stream mobile computing (Baccarelli et al., 2016; Khezr and Navimipour, 2015).

5. Another issues and challenges

According to the heuristic-based mechanism, other interesting lines for future research can be applying new optimization methods (such as lion optimization algorithm (Yazdani and Jolai, 2016), gray wolf optimization algorithm (Mirjalili et al., 2014), bat optimization algorithm (Yang and Hossein Gandomi, 2012) and etc.) which can also be efficient to service composition. Also, the main interesting line for future work is considering semantic information in web service composition, especially in a distributed and dynamic environment. Also, analyzing the user's satisfaction and store this information as a pattern for future service composition is very interesting.

Another imperative research goal for service composition is to suggest new techniques for generating the first generation of countries in such a way that all parts of wide search space of the problem can be covered by the algorithm. Designing novel operators to enhance its ability in looking more efficiently for most proper solutions in the very large search spaces facilitates its escape from traps of local optimum solutions. Eventually, considering the significant growth of mobile cloud computing and serious QoS differences among mobile devices are very interesting in the future.

8. Conclusion

In this paper, we have systematically surveyed the past and the state of the art mechanisms in service composition in cloud environments. First, we overviewed the cloud computing and then the service composition and the problem of cloud service composition were discussed. Then, we explained research methodology and investigated cloud service composition techniques in three main categories including framework-based, agent-based and the heuristic mechanisms. For each of which, we reviewed and compared several past and the state-of-the-art techniques. We also discussed the advantages and disadvantages of the important methods of each category. The challenges of these methods are addressed so that more efficient service composition techniques can be developed in future. The framework-based mechanisms are used for organizing and managing to search, select and composite the cloud services by novel approaches. The agent-based mechanisms are methods based on a component of approach that has been managing duties such as clustering managing, managing workflow or user requests. The important factor that has increased using agent-based mechanisms is scalability. The heuristic-based mechanisms are methods based on heuristic or meta-heuristic mechanisms such as ant colony, genetic mechanism, and bee colony. The overall data collected in this study help to acquaint the researchers with the state-of-the-art in the cloud service composition area. Exclusively, the answers to the research questions summarized service composition's primary purpose, current challenges, research terminologies, approaches and mechanisms in cloud environments. We sincerely hope that the outcomes of this work could lead researchers to develop more effective service combining method in cloud environments.

References

Abdollahzadeh, S., Navimipour, N.J., 2016. Deployment strategies in the wireless sensor network: a comprehensive review. Comput. Commun. 91, 1–16.
Alamir, P., Jafari Navimipour, N., Ramage, M., Ramage, M., Ramage, M., 2016. Trust evaluation between the users of social networks using the quality of service requirements and call log histories. Kybernetes 45 (10).
Andrews, T., Curbera, F., Dholakia, H., Goland, Y., Klein, J., Leymann, F., Thatte, S., 2003. Business process execution language for web services: version
Armbrust, M., Fox, O., Griffith, R., Joseph, A. D., Katz, Y., Konwinski, A., Stoica, I., 2009. M.: Above the clouds: a Berkeley view of cloud computing
Armbrust, M., Fox, A., Griffith, R., Joseph, A.D., Katz, R., Konwinski, A., Stoica, I., 2010. A view of cloud computing. Commun. ACM 53 (4), 50–58.
Asghari, S., Navimipour, J.N., 2017. Resource discovery in peer to peer networks using an inverted ant colony optimization algorithm. Peer-to-Peer Netw. Appl..
Asghari, S., Navimipour, N. J., 2016. Service Composition Mechanisms in the Multi-Cloud Environments: A Survey
Ashouraie, M., Jafari Navimipour, N., 2015. Priority-based task scheduling on heterogeneous resources in the Expert Cloud. Kybernetes 44 (10), 1455–1471.
Austel, P., Chen, H., Dube, P., Mikalsen, T., Rouvellou, I., Sharma, U., Wang, Y., 2015. A PaaS for composite analytics solutions. Paper presented at Proceedings of the 12th ACM International Conference on Computing Frontiers.
Aznoli, F., Navimipour, N.J., 2016a. Cloud services recommendation: Reviewing the recent advances and suggesting the future research directions. J. Netw. Comput. Appl..
Aznoli, F., Navimipour, N.J., 2016b. Deployment strategies in the wireless sensor networks: systematic literature review, classification, and current trends. Wirel. Pers. Commun., 1–28.
Baccarelli, E., Cordeschi, N., Mei, A., Panella, M., Shojafar, M., Stefa, J., 2016. Energy-efficient dynamic traffic offloading and reconfiguration of networked data centers for big data stream mobile computing: review, challenges, and a case study. IEEE Netw. 30 (2), 54–61.
Bastia, A., Parhi, M., Pattanayak, B., Patra, M., 2015. Service Composition Using Efficient Multi-agents in Cloud Computing Environment. Intelligent Computing, Communication and Devices, Springer. pp. 357–370.
Benmerzoug, D., Gharzouli, M., Zerari, M., 2013. Agent interaction protocols in support of cloud services composition. Industrial Applications of Holonic and Multi-Agent Systems, Springer. pp. 293–304.
Benouaret, K., Benslimane, D., Hadjali, A., Barhamgi, M., Maamar, Z., Sheng, Q.Z., 2014. Web service compositions with fuzzy preferences: a graded dominance relationship-based approach. ACM Trans. Internet Technol. (TOIT) 13 (4), 12.
Berardi, D., Calvanese, D., De Giacomo, G., Lenzerini, M., Mecella, M., 2003. Automatic composition of e-services that export their behavior. Paper presented at the International Conference on Service-Oriented Computing.
Burstein, M., Hobbs, J., Lassila, O., Mcdermott, D., Mcilraith, S., Narayanan, S., Sirin, E., 2004. OWL-S: semantic markup for web services. W3C Memb. Submiss..
Candelaa, L., Castellia, D., Manzib, A., Paganoa, P., 2014. PoS (ISGC2014) 022 Realising Virtual Research Environments by Hybrid Data Infrastructures: the D4Science Experience. Paper presented at the International Symposium on Grids and Clouds (ISGC).
Canfora, G., Di Penta, M., Esposito, R., Villani, M.L., 2005. An approach for QoS-aware service composition based on genetic algorithms. Paper presented at Proceedings of the 7th annual conference on Genetic and evolutionary computation.
Charband, Y., Navimipour, N.J., 2016c. Online knowledge sharing mechanisms: a systematic review of the state of the art literature and recommendations for future research. Inf. Syst. Front., 1–21.
Chen, Y, Huang, J., Lin, C., Shen, X. Multi-Objective Service Composition with QoS Dependencies. IEEE Transactions on Cloud Computing, 2016.
Chiregi, M., Navimipour, N.J., 2016d. A new method for trust and reputation evaluation in the cloud environments using the recommendations of opinion leaders' entities and removing the effect of troll entities. Comput. Hum. Behav. 60, 280–292.
Cordeschi, N., Amendola, D., Shojafar, M., Baccarelli, E., 2015. Distributed and adaptive resource management in Cloud-assisted Cognitive Radio Vehicular Networks with hard reliability guarantees. Veh. Commun. 2 (1), 1–12.
Fan, G., Yu, H., Chen, L., Liu, D., 2013. Petri net based techniques for constructing reliable service composition. J. Syst. Softw. 86 (4), 1089–1106.
Fernando, N., Loke, S.W., Rahayu, W., 2013. Mobile cloud computing: a survey. Future Gener. Comput. Syst. 29 (1), 84–106.
Gutierrez-Garcia, J.O., Sim, K.M., 2013. Agent-based cloud service composition. Appl. Intell. 38 (3), 436–464.
Hazratzadeh, S., Jafari Navimipour, N., 2017. Colleague recommender system in the Expert cloud using the features matrix. Kybernetes.
Henni, F., Atmani, B., 2012. Dynamic web service composition. use of case based reasoning and AI planning. Pap. Presente. ICWIT.
Huang, B., Li, C., Tao, F., 2014a. A chaos control optimal algorithm for QoS-based service composition selection in cloud manufacturing system. Enterp. Inf. Syst. 8 (4), 445–463.
Huang, J., Liu, G., Duan, Q., 2014b. On modeling and optimization for composite network–cloud service provisioning. J. Netw. Comput. Appl. 45, 35–43.
Huo, Y., Zhuang, Y., Gu, J., Ni, S., Xue, Y., 2015. Discrete gbest-guided artificial bee colony algorithm for cloud service composition. Appl. Intell. 42 (4), 661–678.
Ivanović, D., Carro, M., 2014. Transforming Service Compositions into Cloud-Friendly Actor Networks Service-Oriented Computing. Springer, 291–305.
Jafari Navimipour, N., Rahmani, A.M., Habibizad Navin, A., Hosseinzadeh, M., 2015. Expert Cloud: a Cloud-based framework to share the knowledge and skills of human resources. Comput. Hum. Behav. 46 (C), 57–74.
Jeong, H.-Y., Yi, G., Park, J.H., 2015. A service composition model based on user experience in Ubi-cloud comp. Telecommun. Syst., 1–11.
Jula, A., Sundararajan, E., Othman, Z., 2014. Cloud computing service composition: a systematic literature review. Expert Syst. Appl. 41 (8), 3809–3824.
Jula, A., Othman, Z., Sundararajan, E., 2015. Imperialist competitive algorithm with PROCLUS classifier for service time optimization in cloud computing service composition. Expert Syst. Appl. 42 (1), 135–145.
Kaklanis, N., Konstadinidou, A., Votis, K., Tzovaras, D., 2016. Automatic semantic service composition aiming at increasing end-users' accessibility. Paper presented at Proceedings of the 31st Annual ACM Symposium on Applied Computing.
Karimi, M.B., Isazadeh, A., Rahmani, A.M., 2016. QoS-aware service composition in cloud computing using data mining techniques and genetic algorithm. J. Supercomput., 1–29.
Khezr, S.N., Navimipour, N.J., 2015. MapReduce and its application in optimization algorithms: a comprehensive study. Majlesi J. Multimed. Process. 4 (3).
Kholidy, H. A., Hassan, H., Sarhan, A. M., Erradi, A., Abdelwahed, S., 2015. QoS Optimization for Cloud Service Composition Based on Economic Model. Internet of Things. User-Centric IoT. Springer. pp. 355-366.
Klein, A., Ishikawa, F., Honiden, S., 2014. SanGA: a self-adaptive network-aware approach to service compositionServices Computing, IEEE Transactions on 7, 452–464.
Klusch, M., Fries, B., Khalid, M., Sycara, K., 2005. Owls-mx: Hybrid owl-s service matchmaking. Paper presented at the Proceedings of 1st intl. AAAI fall symposium on agents and the Semantic Web.
Kurdi, H., Al-Anazi, A., Campbell, C., Al Faries, A., 2015. A combinatorial optimization algorithm for multiple cloud service composition. Comput. Electr. Eng. 42, 107–113.
Lartigau, J., Xu, X., Nie, L., Zhan, D., 2015. Cloud manufacturing service composition based on QoS with geo-perspective transportation using an improved Artificial Bee Colony optimisation algorithm. Int. J. Prod. Res.(Ahead–Print.), 1–25.
Lécué, F., Silva, E., Pires, L. F., 2008. A framework for dynamic web services composition Emerging Web Services Technology, Volume II, Springer. pp. 59–75.
Lemos, A.L., Daniel, F., Benatallah, B., 2016. Web service composition: a survey of techniques and tools. ACM Comput. Surv. (CSUR) 48 (3), 33.
Lin, L., Hu, J., Zhang, J. Packet: a privacy-aware access control policy composition method for services composition in cloud environments. Frontiers of Computer Science, 1–16, 2016.
Meirosu, C., John, W., 2013. Energy efficiency in software defined networks: Google Patents
Milani, A.S., Navimipour, N.J., 2016e. Load balancing mechanisms and techniques in the cloud environments: systematic literature review and future trends. J. Netw. Comput. Appl..
Milani, B.A., Navimipour, N.J., 2016f. A comprehensive review of the data replication techniques in the cloud environments: major trends and future directions. J. Netw. Comput. Appl. 64, 229–238.
Mirjalili, S., Mirjalili, S.M., Lewis, A., 2014. Grey wolf optimizer. Adv. Eng. Softw. 69, 46–61.
Mohammad Aghdam, S., Jafari Navimipour, N., 2016. Opinion leaders selection in the social networks based on trust relationships propagation. Karbala Int. J. Mod. Sci..
Mostafa, A., Zhang, M., 2015. Multi-Objective Service Composition in Uncertain Environments.
Naranjo, P.G.V., Shojafar, M., Mostafaei, H., Pooranian, Z., Baccarelli, E., 2016. P-SEP: a prolong stable election routing algorithm for energy-limited heterogeneous fog-supported wireless sensor networks. J. Supercomput., 1–23.
Navimipour, J.N., Charband, Y., 2016. Knowledge sharing mechanisms and techniques in project teams: literature review, classification, and current trends. Comput. Hum. Behav..
Navimipour, N.J., 2015. A formal approach for the specification and verification of a Trustworthy human resource discovery mechanism in the Expert cloud. Expert Syst. Appl. 42 (15), 6112–6131.
Navimipour, N.J., Zareie, B., 2015. A model for assessing the impact of e-learning systems on employees' satisfaction. Comput. Hum. Behav. 53, 475–485.
Navimipour, N.J., Milani, F.S., 2015. A comprehensive study of the resource discovery techniques in peer-to-peer networks. Peer-to-Peer Netw. Appl. 8 (3), 474–492.
Navimipour, N.J., Rahmani, A.M., Navin, A.H., Hosseinzadeh, M., 2015. Expert Cloud: a Cloud-based framework to share the knowledge and skills of human resources. Comput. Hum. Behav. 46, 57–74.
Navin, A.H., Navimipour, N.J., Rahmani, A.M., Hosseinzadeh, M., 2014. Expert grid: new type of grid to manage the human resources and study the effectiveness of its task scheduler. Arab. J. Sci. Eng. 39 (8), 6175–6188.
Nelson, J., Peterson, L., 2013. Syndicate: Democratizing cloud storage and caching through service composition. Paper presented at Proceedings of the 4th annual Symposium on Cloud Computing.
Pooranian, Z., Shojafar, M., Javadi, B., Abraham, A., 2014. Using imperialist competition algorithm for independent task scheduling in grid computing. J. Intell. Fuzzy Syst. 27 (1), 187–199.
Pooranian, Z., Shojafar, M., Abawajy, J.H., Abraham, A., 2015. An efficient meta-heuristic algorithm for grid computing. J. Comb. Optim. 30 (3), 413–434.
Rao, J., Su, X., 2004. A survey of automated web service composition methods. Paper presented at the International Workshop on Semantic Web Services and Web Process Composition
Rao, J., Su, X., 2005. A survey of automated web service composition methods. Semantic

Web Services and Web Process Composition, Springer. pp. 43–54.
Reim, W., Parida, V., Örtqvist, D., 2015. Product–Service Systems (PSS) business models and tactics–a systematic literature review. J. Clean. Prod. 97, 61–75.
Serrano, D., Bouchenak, S., Kouki, Y., de Oliveira, F.A., Jr, Ledoux, T., Lejeune, J., Sens, P., 2015. SLA guarantees for cloud services. Future Gener. Comput. Syst..
Seuring, S., Müller, M., 2008. From a literature review to a conceptual framework for sustainable supply chain management. J. Clean. Prod. 16 (15), 1699–1710.
Sharif, S.H., Mahmazi, S., Navimipour, N.J., Aghdam, B.F., 2013. A review on search and discovery mechanisms in social networks. Int. J. Inf. Eng. Electron. Bus. 5 (6), 64.
Sheng, Q.Z., Qiao, X., Vasilakos, A.V., Szabo, C., Bourne, S., Xu, X., 2014. Web services composition: a decade's overview. Inf. Sci. 280, 218–238.
Shojafar, M., Cordeschi, N., Baccarelli, E., 2016. Energy-efficient adaptive resource management for real-time vehicular cloud services
Singh, A., Juneja, D., Malhotra, M., 2015. A novel agent based autonomous and service composition framework for cost optimization of resource provisioning in cloud computing. J. King Saud. Univ. - Comput. Inf. Sci.. http://dx.doi.org/10.1016/j.jksuci.2015.09.001.
Sivasubramanian, S.P., Ilavarasan, E., Vadivelou, G., 2009. Dynamic web service composition: Challenges and techniques. Paper presented at the Intelligent Agent & Multi-Agent Systems, 2009. IAMA 2009. International Conference on.
Soltani, Z., Navimipour, N.J., 2016g. Customer relationship management mechanisms: a systematic review of the state of the art literature and recommendations for future research. Comput. Hum. Behav. 61, 667–688.
Sycara, K., Paolucci, M., Ankolekar, A., Srinivasan, N. 2003. Automated discovery, interaction and composition of semantic web services. Web Semantics: Science, Services and Agents on the World Wide Web, 1(1), 27–46
Tout, H., Mourad, A., Talhi, C., Otrok, H., 2015. AOMD approach for context-adaptable and conflict-free web services composition. Comput. Electr. Eng..
Wang, H., Wang, X., Hu, X., Zhang, X., Gu, M., 2016a. A multi-agent reinforcement learning approach to dynamic service composition. Inf. Sci. 363, 96–119.
Wang, S., Zhou, A., Yang, F., Chang, R.N., 2016b. Towards network-aware service composition in the cloud. IEEE Trans. Cloud Comput..
Wang, X., Cao, J., Xiang, Y., 2015. Dynamic cloud service selection using an adaptive learning mechanism in multi-cloud computing. J. Syst. Softw. 100, 195–210.
Wu, Q., Zhu, Q., 2013. Transactional and QoS-aware dynamic service composition based on ant colony optimization. Future Gener. Comput. Syst. 29 (5), 1112–1119.
Xie, R., Gamble, R., Ahmed, N., 2014. Diagnosing Vulnerability Patterns in Cloud Audit Logs. High Performance Cloud Auditing and Applications, Springer. pp. 119–146.
Yang, X.-S., Hossein Gandomi, A., 2012. Bat algorithm: a novel approach for global engineering optimization. Eng. Comput. 29 (5), 464–483.
Yazdani, M., Jolai, F., 2016. Lion Optimization Algorithm (LOA): a nature-inspired metaheuristic algorithm. J. Comput. Des. Eng. 3 (1), 24–36. http://dx.doi.org/10.1016/j.jcde.2015.06.003.
Yu, Q., Chen, L., Li, B., 2015. Ant colony optimization applied to web service compositions in cloud computing. Comput. Electr. Eng. 41, 18–27.
Zhang, F., Hwang, K., Khan, S., Malluhi, Q. Skyline Discovery and Composition of Inter-Cloud Mashup Services, 2016.
Zhang, H., Guo, R.F., Geng, C., 2014. Hierarchical optimization model of cloud manufacturing services combination. Pap. Presente. Appl. Mech. Mater..
Zhang, Q., Cheng, L., Boutaba, R., 2010. Cloud computing: state-of-the-art and research challenges. J. Internet Serv. Appl. 1 (1), 7–18.
Zhao, X., Shen, L., Peng, X., Zhao, W., 2015. Toward SLA-constrained service composition: an approach based on a fuzzy linguistic preference model and an evolutionary algorithm. Inf. Sci. 316, 370–396.
Zheng, H., Yang, J., Zhao, W., 2016. Probabilistic QoS aggregations for service composition. ACM Trans. Web (TWEB) 10 (2), 12.