The Journal of Systems & Software 188 (2022) 111290

Contents lists available at ScienceDirect

The Journal of Systems & Software

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

New Trends and Ideas

Service composition in dynamic environments: A systematic review and future directions[☆]

Mohammadreza Razian[a,b], Mohammad Fathian[b,*], Rami Bahsoon[c], Adel N. Toosi[d], Rajkumar Buyya[a]

[a] Cloud Computing and Distributed Systems (CLOUDS) Laboratory, School of Computing and Information Systems, The University of Melbourne, VIC 3000, Australia
[b] School of Industrial Engineering, Iran University of Science and Technology, Tehran, Iran
[c] School of Computer Science, The University of Birmingham, Birmingham, 3800, UK
[d] Faculty of Information Technology, Monash University, Melbourne, VIC, 3800, Australia

ARTICLE INFO

Article history:
Received 26 May 2020
Received in revised form 18 February 2022
Accepted 24 February 2022
Available online 9 March 2022

Keywords:
Service composition
Uncertainty
Cloud computing
Distributed services

ABSTRACT

Distributed computing paradigms such as cloud, mobile, Internet of Things, and Fog have enabled new modalities for building enterprise architectures through service composition. The fundamental premise is that the application can benefit from functionally equivalent services that can be traded in the cloud or services repositories. These services can vary in their Quality of Services (QoS) and cost provision. Accordingly, the problem of service composition is the process of choosing a configuration of candidate services from a pool of available ones, considering QoS attribute, cost, and users' preference. Due to the inherent dynamism in service computing environments and communication networks, the advertised QoS values might fluctuate; therefore, service composition under uncertainty is inevitable and challenges satisfying Services Level Agreement (SLA). In this paper, we present a systematic literature review to investigate and classify the existing studies in service composition under uncertainty. We identified 100 relevant studies published between the year 2007 and 2020. To the best of our knowledge, this work is the first to explicate a focused systematic review, classification, taxonomy of approaches, and trends along with their assumptions and applications; and to discuss future research directions in service composition under uncertainty.

Contents

[☆] Editor: Eduardo Almeida.

[*] Corresponding author.

E-mail addresses: razian.mr@gmail.com (M. Razian), fathian@iust.ac.ir (M. Fathian), r.bahsoon@cs.bham.ac.uk (R. Bahsoon), adel.n.toosi@monash.edu (A.N. Toosi), rbuyya@unimelb.edu.au (R. Buyya).

https://doi.org/10.1016/j.jss.2022.111290
0164-1212/© 2022 Elsevier Inc. All rights reserved.

1. Introduction

Computing paradigm has shifted from traditional centralized service providing to the distributed computing paradigms (Buyya et al., 2018; de Almeida et al., 2019). Distributed computing paradigms such as cloud computing, mobile computing, Internet of Things (IoT), and Fog (Edge) computing have enabled new modalities for building enterprise architectures through **service composition** (**SC**) and recomposition.

While a single service (also known as atomic service Rodriguez-Mier et al., 2015) is often developed by a simple (fine-grained) functionality, with the purpose of simplifying the application logic and enhancing its re-usability, business complex requirements seek value-added services through the arrangement of multiple existing ones into workflows, which is known as service composition (Wang et al., 2015a). Therefore, a software application can benefit from services with the same functionality but different Quality of Service (QoS) values (response time, reputation, security, availability, etc.) that can be traded in the cloud and/or can be provided through IoT's *intelligent things*. From the IoT perspective, each intelligent thing (called a node), either located in a smart city (Liu et al., 2019) or an Industry 4.0-based manufacturing system (Xu et al., 2018c), can be considered as a potential source of service. Practically, IoT nodes expose their functionalities such as Sensing-as-a-Service (SaaS) or Video-Surveillance-as-a-Service (VSaaS) though the Web APIs (Application Programming Interface). While IoT nodes generally have a limitation in providing computation and storage resources, cloud computing platforms serve virtually unlimited, pay-as-you-go, and flexible resources. Currently, some organizations have started to present their cloud-based software products containerized (for example, available in Docker Santos et al., 2018 hub), and orchestrated with technologies like Kubernetes (Xu et al., 2018b). Hence, Cloud and IoT play a complementary role and potentially offer a tremendously large number of services distributed through ubiquitous communication networks (Morabito et al., 2018). In this situation, QoS-aware service composition is the process of choosing proficient candidate services according to users' objectives/preferences and constraints (on QoS attributes) to construct a more value-added **composite service**. However, since there exist lots of services that perform the same function albeit with different QoS, service composition becomes a crucial problem to find an optimal set of services to automate a workflow (a set of requirements originated from business logic functions such as authentication, payment, search/recommend a movie/hotel).

In the today's software systems, the need for uncertainty analysis, *"as a first class concern"*, is becoming increasingly important (Garlan, 2010). Due to the variability of QoS values in real-world dynamic environments, which is known as **QoS uncertainty**, the QoS estimation/prediction in service composition has become more challenging. QoS uncertainty refers to the situations involving incomplete or unknown QoS information that arises in partially observable and/or stochastic environments. Usually, the distance between the reality and what we know (advertised QoS values) is called uncertainty (Esfahani and Malek, 2013; Guidara et al., 2016; Sun et al., 2019; Wang et al., 2020; Pham et al., 2020). In the literature, different approaches like Fuzzy set theory, Probabilistic (stochastic) approaches, and Machine Learning techniques have been widely used to address this challenge. In the following, we describe the main components of the service composition ecosystem along with the background and motivation of this survey.

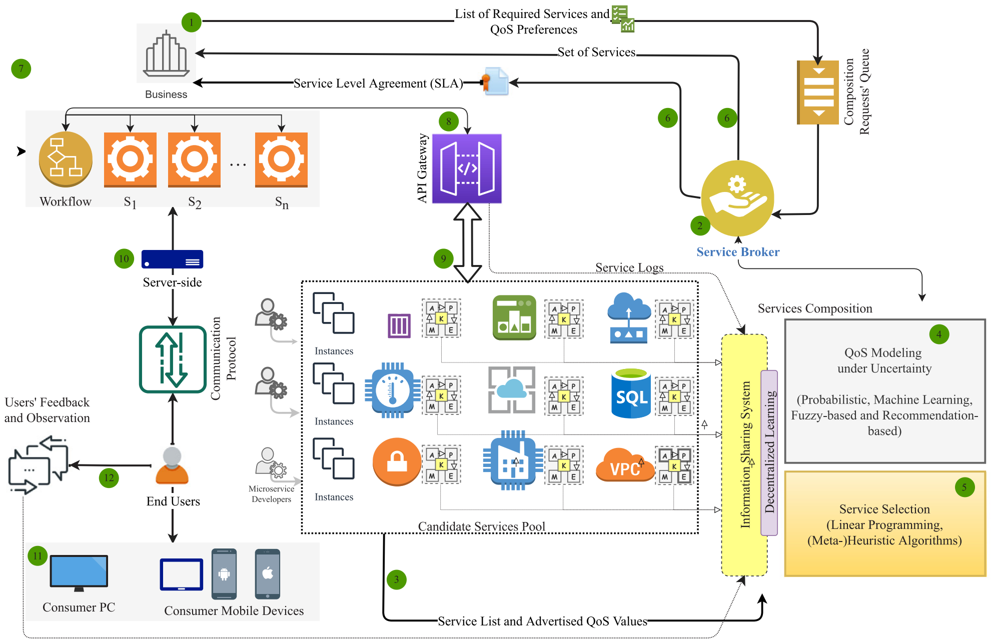


Fig. 1. Main components and relations in QoS-aware service composition problem under uncertainty.

1.1. Background and motivation

Unlike traditional monolithic architecture for building an application, businesses are attracted to distributed applications constructed upon microservices architecture (Roca et al., 2019; Zhao et al., 2019), in which software is constructed by a set of loosely coupled and granular services. Typically, a micro-service exposes its functionality through Web APIs. This type of architecture, as an enabler of DevOps (Bass et al., 2015) (Development and Operations), helps to improve the modularity, flexibility, agility, scalability, and resiliency (Zhang et al., 2018) of software which is a serious need for businesses with dynamic and competitive environments. A service, as a fundamental component of microservices architecture, is identified by its functional and non-functional requirements. Functional requirement determines the responsibility of service while non-functional requirements (known as **QoS attributes**) define the quality aspects of a given service like response time, security, and reputation.

Many research works on service composition have been focused on functional requirements (also known as service matchmaking Rodriguez-Mier et al., 2015). However, in this paper, we focus only on QoS-aware service selection. As shown in Fig. 1, a business (1) submits a composition request to the service broker (Anisetti et al., 2019). This composition request includes the business's required set of services (which is usually stated as a *workflow*) and QoS preferences. The service broker (2), using available candidate services and their advertised QoS values (3), suggests a composite service with guaranteed QoS values (steps 4 and 5) through a service level agreement (6) either in centralized architecture or decentralized architecture like MAPE-K (Monitor-Analyze-Plan-Execute over a shared Knowledge) loop (Weyns, 2020). Finally, the business utilizes the composite service to create its application. The business uses this composite service (7) to form its software application employing API(s) token (8) of each service (9 and 10). Finally, the user interacts with the application (11) and sends his/her feedback and observation (12) to the monitoring sub-system (also, the component *API Gateway* potentially can send service logs to the monitoring subsystem).

However, for a given workflow with *n* tasks (required services) and *m* candidate services for addressing each task, the problem of finding an optimal composition based on user's constraints on QoS values is an *NP*-problem. Many works have been devoted to addressing the service composition problem *(SCP)* with the assumption that the advertised QoS values are deterministic. The fundamental assumption of these approaches is that the advertised QoS values for service providers do not change over the time (Zhou and Yao, 2017; Jatoth et al., 2018). However, this is in stark contrast to reality, where QoS does fluctuate. This fluctuation is attributed to the inherent uncertainty of services computing environment and communication networks, which makes satisfying QoS requirements and achieving Service Level Agreement (SLA) guarantees challenging. To prevent or mitigate the penalties applied (due to SLA violation), the service broker needs to model QoS attributes concerning uncertainty. Among the interesting observation to note, Zheng et al. (2016) explored the impact of uncertainty of response time attribute for YouTube service. They noted that the response time values could dramatically change and, do not fit well-known probability distributions. The adverse effects of an inaccurate QoS model can be critical, if not significant, for a workflow — consider, for example, safety-critical systems. In recent years, a significant portion of research has been devoted to addressing the QoS-ware SC under uncertainty. The influx of research in SC under uncertainty can be attributed to the increasing reliance on computing environments that are characterized by their provision of a pool of shared resources, elastic and unbounded scale, dynamism in their operations, multi-tenancy, and communication networks; this can practically translate into uncertainty in service composition. By using a QoS-aware service composition under uncertainty, businesses not only are able to respond to continuous changes of customer's requirements in a competitive market but also do not require to spend time/cost for service development from scratch. The current research trend aims to investigate the estimation and prediction of QoS fluctuation and their likely consequences on SLA violations and mitigation strategies.

1.2. Goals of systematic literature review

In this paper, we conduct a Systematic Literature Review (SLR) to survey, classify, and report the existing studies in service composition under uncertainty. We identified 100 relevant studies published between the year 2007 and 2020. To the best of our knowledge, our study is the first of its kind to explicate the area of service composition under uncertainty and despite the growing body of research and applications that relate to the subject. Our main goal is to answer the following questions: *how existing research in the area of service composition captures and models uncertainty? What are their strengths, limitations, and suitability of application? How do QoS parameters, dimensions, and metrics differ with the environments? What are the requirements/assumptions in different approaches when dealing with uncertainty?* A comparative framework is developed to compare the approaches against aspects such as the source of uncertainty, methods of QoS modeling, QoS parameters, datasets, and objective function (single or multi-objective model), single or multi-source services, scalability, etc. A technical taxonomy of the existing approach is proposed. We identify gaps and limitations in existing work, and we discuss possible solutions to address these limitations.

The rest of the paper is organized as follows: In Section 2, we define our research methodology along with research questions, inclusion and exclusion criteria. Section 3 presents a technical taxonomy and comparison of existing studies from inception to the current state. Section 4 provides SLR results, technical discussion and comparison on similar works within a category. Section 5 presents research implications, trends and future directions. Threats to the validity of proposed SLR is discussed in Section 6. Finally, in Section 7, we conclude the paper and propose future work.

2. Research methodology

Systematic Literature Review (SLR) starts by defining a review protocol (Brereton et al., 2007). Our research methodology includes three main processes: *Planning Review* is the first step of this methodology, which includes developing research questions and a comprehensive review protocol. The second process is *Conducting Review*, which itself includes developing search queries, finding relevant studies, and providing inclusion and exclusion criteria. The third process, *Document Review*, includes documenting the review and then concluding the findings.

2.1. Planning review

In the *Planning Review* phase, we design research questions, develop the review protocol, and validate review protocol. Before discussing each step, we identify the need for this SLR.

2.1.1. Identifying the need

There are many reasons for performing SLR, including (1) summarizing the existing studies, tools, methods, frameworks, and techniques; (2) identifying research gaps and presenting areas for further exploration and investigation; (3) assisting researchers either in extending the current hypothesis or generation of a new theory. To the best of our knowledge, this is the first SLR in the scope of service composition under uncertainty.

2.1.2. Specifying the research questions

We designed the following Research Questions (**RQ**):

- RQ1: What are the main reasons for the uncertainty according to various service composition environments, including cloud, IoT, Mobile, etc.?
- RQ2: What approaches have been applied to deal with uncertainty?
- RQ3: How do QoS parameters, dimensions, and metrics differ with the approaches?
- RQ4: How the consideration of the uncertainty has evolved as we transit from one environment to another?
- RQ5: What are the requirements/assumptions in different approaches to deal with uncertainty?
- RQ6: Which datasets are applied to evaluate the performance of proposed methods?

2.1.3. Developing and validating the review protocol

The SLR research questions and protocol were developed through a number of brainstorming sessions, discussions, and a preliminary search of the literature. All authors were involved in the process. The process was iterative, where the research questions and search strings were undergone several refinements before they were confirmed for executing the SLR. Measures to ensure consistency of the protocol and search were considered during the iterative process, where authors had taken a *"best-effort"* approach to make sure that the search strings reflect on and consistent with the questions; the data extraction process is relevant to the search; and the data analysis procedure is appropriate for answering the questions. The search protocol was primarily executed by the first author and was checked and discussed by all authors, who have experience in conducting SLRs. We adhered to guidelines in Kitchenham (2004) for evaluating and confirming the protocol.

2.2. Conducting review

The second phase of our research methodology is conducting the review. In this phase, we developed a search string to identify relevant researches. Then, we collected all related studies according to the search string. After that, we selected **primary studies (PSs)** using inclusion and exclusion criteria. Finally, we extracted the desired data and synthesized them.

2.2.1. Studies selection

To provide an extensive search, we explored title, abstract, and keywords of peer-review articles using the following search string along with the term uncertainty in the whole paper (anywhere):

```
(Mobile "OR" Cloud "OR" IoT "OR" Web "OR" Edge "OR" Fog) AND
          Service Composition AND Uncertainty
```

It is worth mentioning that the way a typical search string is applied in different databases may differ due to the difference in syntax, semantics, operator precedence, and default behavior. The first part of the query enforces the search engine to find only web, mobile, cloud, IoT, and fog/edge environments for SC. The second part of the query determines the SC studies. The last part of the query limits the searched items to only the studies addressed the problem of uncertainty in SC. Besides, we selected those studies published between 2007 to 2020. This is because researchers have paid much more attention to web services with the popularity of cloud computing in its modern context from 2007 (Buyya et al., 2010). We obtained 1543 papers from the searched databases listed in Table 1. Because some search engines do not provide a flexible search query in the title, abstract, and keywords parts simultaneously, we applied search string manually to 1543 papers. Finally, we extract 189 papers matched with the above search query for applying the inclusion and exclusion criteria described in the next section.

2.2.2. Inclusion and exclusion

SLR requires the explicit inclusion and exclusion criteria to evaluate the research papers to be investigated (Kitchenham, 2004). We included all the peer-reviewed papers published between 2007 and 2020 as follows:

Table 1
Explored databases and scholar search engines used in studies discovery.

No.	Publishers and databases	URL address
1	ACM Digital Library	https://dl.acm.org/
2	IEEE Xplore Digital Library	https://ieeexplore.ieee.org/
3	Science Direct	https://www.sciencedirect.com/
4	Springer Link	https://link.springer.com/
5	Scopus	https://www.scopus.com/
6	Web of Science	https://clarivate.com/
7	Wiley Online Library	https://onlinelibrary.wiley.com/

- Those studies which the QoS-aware SCP was the main purpose of the article whether or not the authors referred to their study as a QoS-aware.
- SC methods involving uncertainty around QoS-values, i.e., the papers with concentrated on solving the SCP under uncertain QoS values whether or not the authors referred to their study as an uncertainty-aware solution.
- QoS-aware SC papers where uncertainty is attributed to user's inadequate knowledge of the domain, their preference for QoS requirements, etc.
- Service QoS prediction/estimation for service selection based on the dynamicity of environment and incomplete information, which can be used in SC.

However, the articles on the following topics were excluded:

- Papers that neither discuss nor consider uncertainty as part of their formulation and/or QoS modeling.
- Papers that discuss uncertainty in ontology matching, business process, or workflow structures.
- Papers that discuss new ideas or provide preliminary results without implementation
- Papers that discuss service selection without considering SC applications.
- Non-peer-reviewed publications, white papers, and papers written in non-English languages.

To apply the inclusion and exclusion criteria, we manually reviewed the abstract, introduction, conclusion, and other parts of each paper.

2.2.3. Data extraction and synthesis

Furthermore, we checked for "outliers", i.e., the papers that our search query did not include, but they are relevant and worth reviewing. To this aim, we adopted the **backward/forward snowballing** technique (Wohlin, 2014) for extracted papers by using Google Scholar to find the related articles. This helps us to ensure that we covered related studies proficiently. After this stage, we chose 100 most relevant papers as **primary studies** for undertaking reviews. Fig. 2 depicts the whole process of study selection in our SLR. In addition, Figs. 3 and 4 indicate the frequency of publishers and publications year of primary studies (**PSs**), respectively.

2.3. Document review

In the Document Review phase, we concluded the findings, validated them and proposed the results and guidelines to the community. The reporting structure includes general information about the research, the goals of review, the importance of research questions, and the significance of the work of review. Furthermore, the review method (steps taken to conduct the review, results and technical discussion), implications and future directions of review for research and practice, and threats to validity are other parts of the SLR report.

3. Taxonomy and approaches in QoS-aware SC under uncertainty

In the literature, there exist some proposed approaches with different presumptions to deal with uncertainty. Broadly, there are four categories of uncertainty-aware SC approaches according to how they have modeled and managed uncertain QoS values. Enumerating those methods covered in the SLR, our classification includes Machine Learning-based systems, Probabilistic methods, Fuzzy SC, and Service Recommendation as main classes in the proposed taxonomy. Fig. 5 depicts the proposed taxonomy, concluded from the extracted studies. A review studies in each category is presented below:

3.1. Machine learning-based systems

In recent years, researchers have tried to deal with the dynamicity of the SC environment using Machine Learning (ML) algorithms so as learn the changes without assumptions on the shape of QoS values distribution. We categorized these approaches as follows: Reinforcement Learning, Clustering, Classification, and Regression.

3.1.1. Reinforcement learning

The Reinforcement Learning (RL) method is a kind of ML algorithms (Lei et al., 2015a), which is frequently used in modeling QoS values in SC. Wang et al. (2010) propose an RL-based SC to obtain near-optimal execution policies for composite services without prior knowledge about QoS parameters. The reward function is constructed by aggregating QoS values using the simple weight additive (SAW) method. Moustafa and Zhang (2012) use RL for learning the last n service activities to defeat the changes in a run-time environment. Furthermore, Yu et al. (2013b) model SCP by using the Markov Decision Process (MDP) and generate the optimal policy using Q-learning. MDP is useful for studying optimization problems solved by reinforcement learning. In Wei et al. (2017), authors focus on SC in which the rationality of the user's preferences is considered based on the 3σ principle. They propose a constraint-satisfied SCP as MDP to handle the user's tasks and QoS constraints.

To consider the incomplete information, Lei et al. (2014) employ a Partially Observable MDP (POMDP). Additionally, they use reinforcement learning for SC with a Time-based Learning method (Lei et al., 2015b) and maximum-expected total-discount-benefits criterion to compare policies. Wang et al. (2016) use SARSA(λ) RL algorithm for their POMDP problem. To model the conflicting QoS parameters, instead of applying SAW (which has some limitations), a multi-objective POMDP is studied in Mostafa and Zhang (2015). To predict the distribution of large-scale SC, Wang et al. (2015c) integrate the Gaussian process with RL and use Kernel function approximation. They perform an extensive evaluation of a real large-scale dataset. Moustafa and Ito (2018) combine deep learning into RL to find a composite service in the large scale environments. It is worth mentioning that in a large scale problem, in terms of dimensional state and action spaces, deep learning empowers RL to scale the intractable problems (Moustafa and Ito, 2018). Recently, Mahfoudh et al. (2018) proposed a service composition framework based on Coordination model with reinforcement learning. They combine multi-agent RL, nature-inspired coordination model (chemical-based coordination rules), and self-composing services in their framework. They utilize Q-Learning as an RL algorithm and SAPERE (Zambonelli et al., 2011) for the coordination model. To tackle the lack of global knowledge in centralized approaches, D'Angelo et al. (2020), introduce a decentralized learning for self-adaptive QoS-aware service composition. This approach uses a reinforcement learning to make services able to dynamically learn from past experience using the information sharing pattern for architecting the decentralized MAPE-K loop.

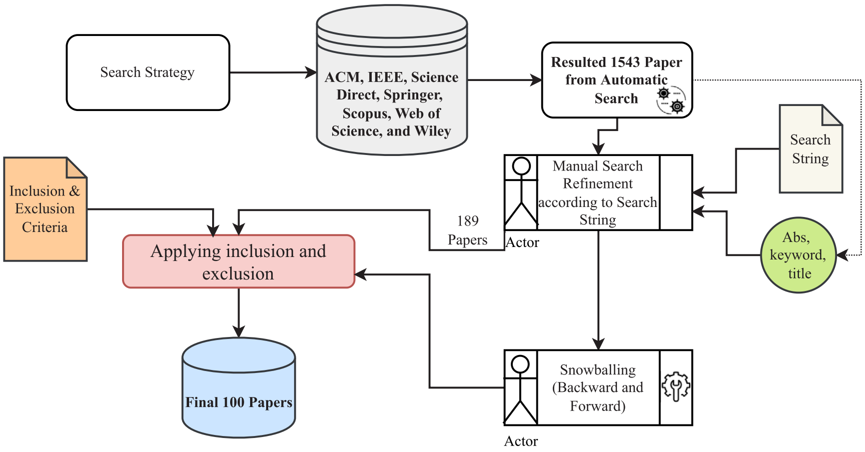

Search Strategy
ACM, IEEE, Science Direct, Springer, Scopus, Web of Science, and Wiley
Resulted 1543 Paper from Automatic Search
Search String
Manual Search Refinement according to Search String
Actor
Abs, keyword, title
Inclusion & Exclusion Criteria
189 Papers
Applying inclusion and exclusion
Final 100 Papers
Snowballing (Backward and Forward)
Actor

Fig. 2. Process of studies (PSs) selection.

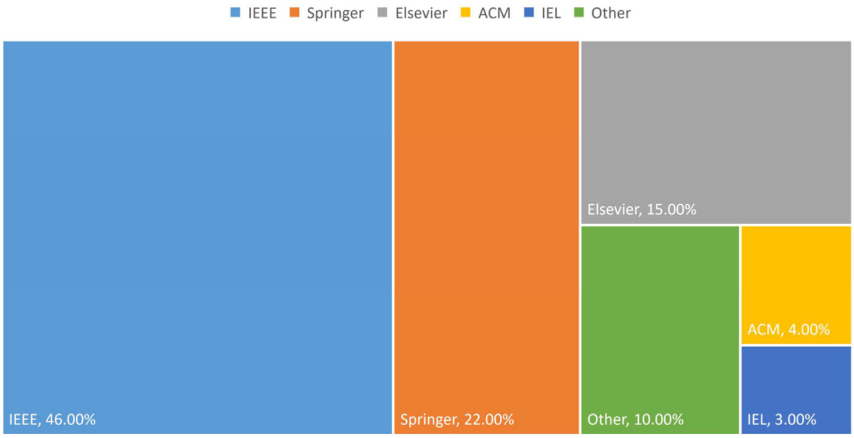


Fig. 3. Publishers name and frequency.

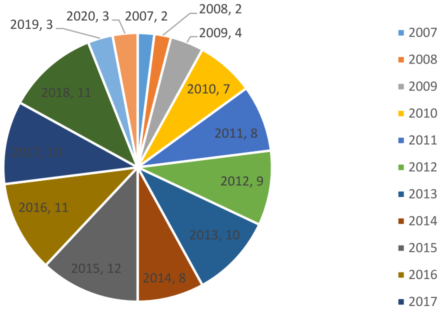


Fig. 4. Publications year and frequency.

3.1.2. Clustering

The clustering algorithms (Fathian et al., 2007; Amiri et al., 2009) attempt to categorize services according to similar functionalities or QoS attributes. Xia et al. (2011) use a clustering density-based method named OPTIC to find the near-optimum composition. Recently, Khanouche et al. (2019) introduce a clustering-based service pruning method using K-means to group and remove candidate services according to their QoS level. They also propose a lexicographic optimization to determine the services satisfying the global QoS constraints along with a search tree to obtain the near-optimum composite service.

3.1.3. Classification

In the literature, a few classification techniques are used to find the values of QoS attributes. Zhang (2010) proposes the Radial Basis Function neural networks (NN) with an modified K-means algorithm to predict QoS of web services. Yu (2012) combines the Matrix Factorization (MF) with a decision tree-based learning method to handle new clients. It is notable that new client, with no previous interaction information, are posed the cold start problem. Efstathiou et al. (2014) consider an SC scenario in a service-based Mobile Ad-hoc Networks (MANET) where the nodes in the formed MANET offer concrete services. They adopt a low-cost statistical model with a surrogate model for the prediction of QoS using a multi-objective evolutionary algorithm, NSGA-II. Surrogate models try to computationally estimate the fitness functions using techniques like Random Forest (RF) and Classification and Regression Trees (CART). Recently, Isolation Forest (Liu et al., 2008), an unsupervised machine learning algorithm, was applied to service composition problems (Razian et al., 2020a) to detect anomalies in historical QoS values before QoS estimation.

3.1.4. Regression

The regression algorithms attempt to approximate the mapping function from input QoS values to numerical or continuous QoS values. Ye et al. (2016) propose a prediction model using multivariate time series based on end-users long-term QoS-aware constraints and monitored QoS data. Sun et al. (2018) introduce

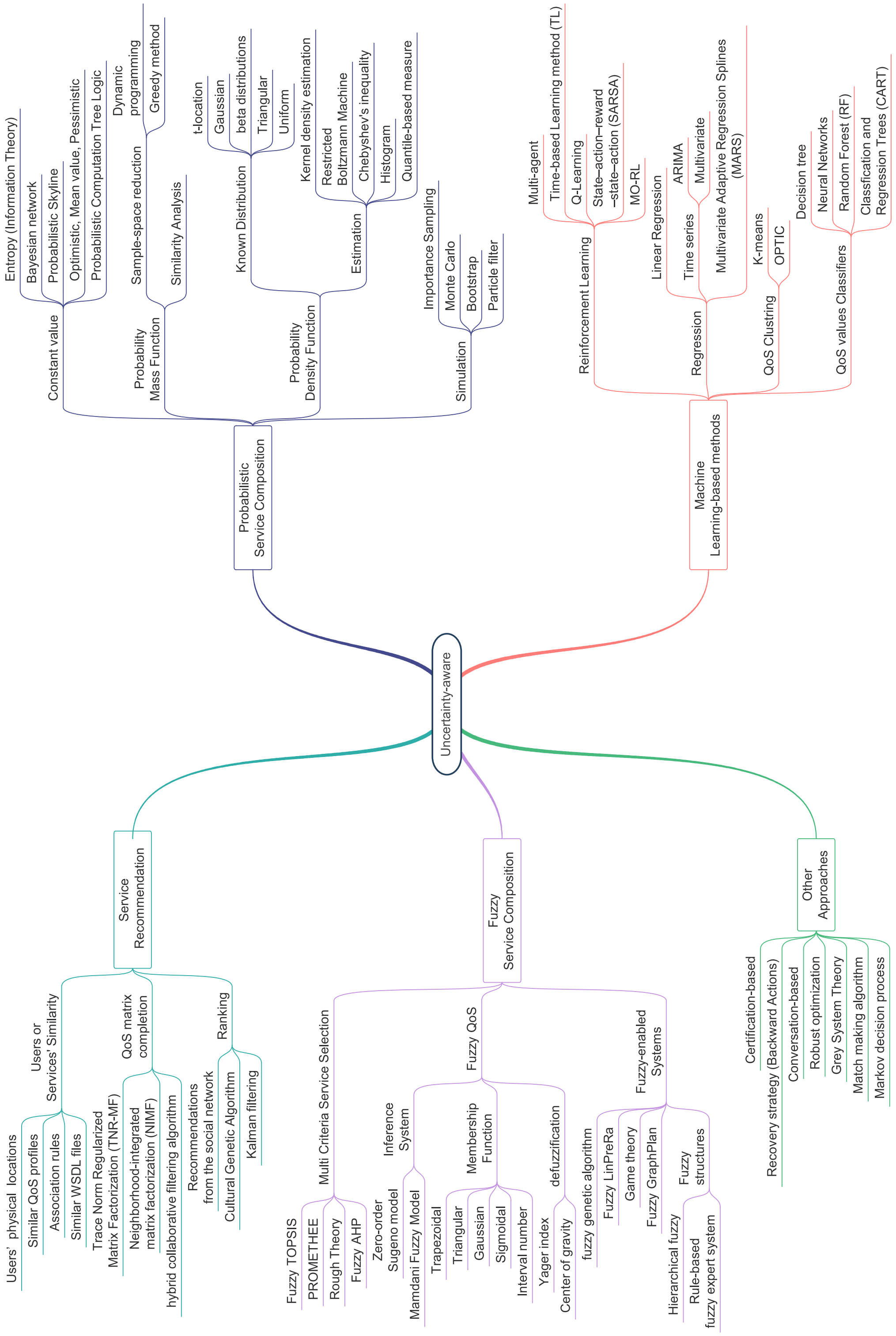


Fig. 5. Taxonomy and approaches in QoS-aware service composition under uncertainty.

a time-series-based method to estimate the QoS values using run time captured data. Guo et al. (2017) study a QoS forecasting time series using the ARIMA model (AutoRegressive Integrated Moving Average). To decrease the search space, they use Skyline service selection (also called Pareto optimality) to prune redundant candidate service. *"The Skyline is defined as those points which are not dominated by any other points"* (Borzsony et al., 2001). According to this definition, the dominance means: *"A point dominates another point if it is as good or better in all dimensions and better in at least one dimension"* (Borzsony et al., 2001). Recently, a time estimation model has been proposed (Zhang et al., 2018) by using a regression model for video applications. In this model, the features of a video like resolution has been investigated. These features are transformed into a log2-scale form which is motivated by Barnes et al. (2008), to obtain a proficient linear fitting.

3.2. Probabilistic

As shown in Fig. 5, in the class of probabilistic, the common methods applied for modeling QoS values are enumerated as Constant Value, Probability Mass Function, Probability Density Function, and Simulation-based.

3.2.1. Constant value

In the category of *Constant Value*, researchers represent QoS values as a single/multiple value(s) (Yasmina et al., 2018). In the following, we present the main approaches in this category.

Optimistic/mean/pessimistic QoS. Wiesemann et al. (2008) propose a multi-objective SC to minimize two conflicting QoS attributes time and price. They use the average value-at-risk (AVaR) measure to quantify the risks related to uncertain parameters. Li et al. (2012) model the SC in IoT as a finite state machine. Because the service providers in IoT are *intelligent devices*, they focus on the reliability and specify the properties of SC using Probabilistic Computation Tree Logic. They apply a tool named PRISM (Kwiatkowska et al., 2011) as a probabilistic model checking for verifying quantitative properties. In Falas and Stelmach (2013), the availability of a service in a typical time frame is studied concerning the number of requests for that service. Furthermore, the impact of context changes on service availability is investigated in Njima et al. (2016). The location and bandwidth are taken into account to calculate the availability of a set of services in an uncertain mobile context. The QoS delivered by services needs to be managed in an adaptive and predictable way. In this respect, a QoS management and optimization framework is introduced in Calinescu et al. (2010) for adaptive service-based systems. They use probabilistic temporal logic to translate user's QoS requirements to identify and enforce optimal system configurations. Cardellini et al. (2011) point that service-oriented paradigm and self-adaptation features are the major trends in software engineering. They propose MOSES methodology to support QoS-driven adaptation of a service-oriented system.

For QoS estimation, Chen et al. (2016b) adopt two approaches: pessimistic estimation to present the worst value of QoS and probabilistic estimation to present an expected value. In Kil et al. (2016), authors find minimum, average, and maximum of QoS values using the past service executions. Therefore, decision-makers are able to opt among optimistic, pessimistic, or average composition. A robust multi-criteria algorithm is proposed in Ramacher and Mönch (2014) using the NSGA-II. For response time, an ex-ante value has been obtained from historical information, and a Pareto frontier has been adopted for selection among alternative services in a reasonable amount of time. Wang et al. (2018a) propose an SC in the field of cyber–physical social systems (Wang et al., 2018a) employing Hofstede's cultural dimension theory (Hofstede, 2011). This theory includes six dimensions to measure the degree of users' preference for the services.

Entropy (Information Theory). Entropy is the average rate at which information is produced by a stochastic source of data. Usually, Entropy and Hyper-Entropy are used to denote the uncertainty (Wang et al., 2011) of QoS values. Malik and Medjahed (Malik and Medjahed, 2010a) consider Information Theory and propose a reputation propagation model to manage trust in SCs. The key criteria of service selection is the service provider' reputation. Additionally, they evaluate the service providers' reputation regarding the credibility values of service raters (consumer's views) (Malik and Medjahed, 2010b). The service raters are considered as honest and dishonest raters, and also service providers are classified into five different behaviors with malicious activities. To find the uncertainty, Gong et al. (2014) consider a two-phase architecture on the basis of the cloud model (Li et al., 1998) by transforming quantitative QoS values from historical QoS values to qualitative QoS concept (uncertainty level). In the second phase, they look for substitute services that satisfy the user's constraints. Recently, a reliable services selection (Wang et al., 2017) method is proposed to filter those candidate services with higher uncertainty. The uncertain candidate services are those services with higher QoS entropy and variance.

Skyline. Skyline concepts that were proposed by the database community were adopted in QoS-aware SC in Alrifai et al. (2010). Yu and Bouguettaya (2010) encode their model founded on p-R-tree and calculate the p-dominant skyline. They assume there are enough historical monitored data collected using some QoS monitoring methods like (Barbon et al., 2006; Jurca et al., 2007). In Wang et al. (2012), the authors incorporate the concept of shared skyline computation with Genetic Algorithm (GA) for re-composition. Sun et al. (2013) use the Particle Swarm Optimization (PSO) algorithm based on Skyline to select the candidate services.

BayesIan network. Bayesian Network (BN) is a probabilistic graphical model that represents a set of variables and their conditional dependencies using a directed acyclic graph. Chen et al. (2013a) propose a web service model with the ability of exception handling based on BN. The model deals with the uncertainty that existed in the execution of a composite service by using failure probability and historical operation data. Furthermore, Ye et al. (2014) propose an economic model using a Bayesian network based on extended Shenoy–Shafer for cloud service composition in the long-term.

3.2.2. Probability mass function (PMF)

Although the representation of QoS values as a single or multiple constant value(s) is easy to model and straightforward to calculate, it does not reflect the QoS values of real Internet-based services (Ivanović et al., 2014; Zheng et al., 2016). Hwang et al. (2007) consider the PMF to present the fluctuating QoS. They also calculate the PMF of different workflow structures like parallel and loop. To calculate the PMF, they compare Greedy and Dynamic Programming methods in terms of computational time. Hwang et al. (2015) extend their previous work by considering local constraint and adjustment module. The former breaks down user's workflow-level constraint (on a given QoS attribute) to the task-level constraints, while the latter tries to conform the locally (task-level) optimum service selection to workflow-level QoS constraint. It is notable that they represent PMF of QoS attributes using users similarity.

3.2.3. Probability density function (PDF)

Some researchers tried to model QoS attributes using known or unknown probability density functions. In the following, we discuss these approaches.

Known distribution. Wu et al. (2009) model and anticipate QoS values based on the stochastic timed colored Petri net. The interval rate of users' requests is considered as a Normal or Poisson distribution. Also, the arrival interval and running time of a service request is considered as an Exponential distribution. To handle the mobility of distributed services in mobile environments, Wang (2011) predicts the availability of the service providers by considering Normal and Uniform distribution. Schuller et al. (2012) remove the candidate services with higher variance. They improve the solution with ILP (Integer Linear Programming) gradually and remove the fluctuated QoS attributes until the termination condition is satisfied. They extend their work in Schuller et al. (2014) using a Genetic Adaptation algorithm to reduce computation time. Deng et al. (2016) investigate a risk-aware selection problem for Mobile SC using probability distribution function. They assume that the probability of staying a mobile service provider in the required distance to the service requester is predictable. They solve the resulted model by a simulated annealing algorithm. The authors in Ye and Li (2018) also assume that the QoS values like availability and reliability follow a Normal distribution. To solve the proposed mathematical optimization model, they use CPLEX and function lsqnonneg in MATLAB.

Unknown distribution. In probability, density estimation is a method of constructing an estimate based on the observed data. Zheng et al. (2010a) estimate PDF of QoS attributes using Gaussian Kernel Density technique by exploiting historical QoS records. This technique creates a smooth curve for a given set of data points. Mezni and Sellami (2018) applied the same technique but using swarm intelligence to find the optimal composition. To increase the speed of the computation of convolution (the QoS of a sequential workflow is the convolution of the PDFs of the component QoS), Zheng et al. (2010b) apply a Fast Fourier Transform (FFT) and develop a tool called QoS DIstribution eStimation Tool (QoSDIST) (Zheng et al., 2010b) for service composition. In Zheng et al. (2011), a calculation method for different workflow structures (like repetitive and concurrent tasks) has been studied. As an example, the aggregation of the time of two concrete services in a sequential structure can be considered as the problem of finding the probability density function of adding two independent variables, which is the convolution of every PDF. In Zheng et al. (2016), they extend previous work by utilizing a depth-first search (DFS) method to calculate the PDF for a composite service under the assumption that the distribution of response time (with continuous values) is achievable from client-side, server-side or third-party monitoring system. Furthermore, quantile-based measure (Ramacher and Mönch, 2013), Restricted Boltzmann Machine (Peng et al., 2017), and Chebyshev's inequality (Elhabbash et al., 2017) have been utilized to make and estimate of the PDF of services' non-functional requirements.

3.2.4. Simulation

For QoS attributes that have been represented by standard distribution, simulation approaches are applied to generate a QoS model. Rosario et al. (2008) propose a soft contract rather than a hard contract by using a distribution of the considered QoS parameters. As a hard contract, they present the phrase *"the response time is required to be less than a fixed value"*(Rosario et al., 2008) that does not fit for real-world scenarios. While they state that a statement like *"a response time is less than T milliseconds for* 95% *of the cases"* (Rosario et al., 2008) is an example of a soften contract which is more possible in real-world scenarios. They developed a tool, namely TOrQuE, which is based on Monte-Carlo dimensioning, to obtain a global probabilistic contract. Furthermore, Yao and Sheng (2011) predict the availability in a given time-slot through a particle filter-based method. Wang et al. (2015b) employ the Importance Sampling technique to examine the QoS probability of composite service using stochastic Project Evaluation and Review Technique (PERT) (Wang et al., 2015b).

3.3. Fuzzy Service composition

A fuzzy model can be employed in situations where a QoS model should reflect experts' opinion due to the lack of complete and reliable data for probabilistic QoS model construction (Jang et al., 1997; Dastjerdi and Buyya, 2014; Zhang et al., 2019c). We categorize Fuzzy-based approaches into three classes: Fuzzy QoS, Multi-Criteria Service Selection, and Fuzzy-enabled Systems.

3.3.1. Fuzzy QoS (FQoS)

QoS attributes can be modeled and assessed as Fuzzy numbers (de Gyvés Avila and Djemame, 2013). (Şora and Todinca, 2015) design an architecture using fuzzy QoS properties containing domain ontology service, functionality finding module, QoS properties directory, and fuzzy ranker. Xu et al. (2017) describe QoS attributes using a triangular fuzzy-valued for fuzzification, and Yager index for defuzzification. Like previous work, Veeresh et al. (2017) consider triangular membership to calculate the rating of the service, max–min to combine crisp input values (response time, energy, throughput and hop count), and Center of gravity for the defuzzification process. By using the rule-based fuzzy reasoning, authors in Tripathy and Tripathy (2018) proposes a dynamic QoS-aware SC, which is enriched with a run-time monitoring module to re-plan when an adaption signal is triggered. For constant monitoring of service, Monitor Specification Language (Tripathy and Patra, 2011) has been employed. Recently, Niu et al. (2019) present the uncertain QoS values as an interval number and solve the obtained SCP by using a non-deterministic multi-objective evolutionary algorithm and uncertain interval Pareto comparison.

3.3.2. Multi-criteria service selection (MCSS)

The approaches in this class consider service selection as a Multi-criteria Decision Analysis (MCDA) problem. Zhang et al. (2011) propose a hybrid QoS model (i.e., different type of like Intuitionistic and triangular numbers) by using TOPSIS (Zhang et al., 2011) and AHP (Zhang et al., 2012). Mu et al. (2014) estimate users' preferences represented by subjective and objective weights. The subjective weights are directly set by users using fuzzy weights, while objective weights are obtained from the user's preference history information of the same service request using Rough Set. An interval-based fuzzy ranking (Jian et al., 2016) approach is proposed by using the dominance concept; hence, instead of simple additive weighting, the authors use PROMETHEE (Behzadian et al., 2010) ranking method.

3.3.3. Fuzzy-enabled systems (FES)

Some approaches incorporate fuzzy theory into well-known techniques like Game Theory which we called *FES*. In addition, a composition technique using Fuzzy theory in mobile ad-hoc networks (Prochart et al., 2007) was developed. Also, a resource management middleware is used to assess the capability of a device for providing a service based on criteria like network signal strength and battery level. They also use the Sugeno method (Sugeno, 1985) for the fuzzy inference. In complex scenarios with a large set of variables, defining the fuzzy system with a flat set of rules results in growing the number of rules exponentially which is known as rule explosion (Wang, 1998; Torra, 2002). To address the problem of rule explosion, the hierarchical fuzzy system has been proposed in Pernici and Siadat (2011). Zhao et al. (2015) develop a multi-objective SLA-constrained SC on the basis of a fuzzy linguistic preference model (Zhao et al., 2015) and weighted Tchebycheff distance. Fuzzy Game Theory (Johannes et al., 2015), fuzzy neural networks (Luo et al., 2015), and Fuzzy SC with modified GraphPlan (Zhu et al., 2018) are also have been utilized in the literature. Recently, Xu et al. (2018a) propose a multi-objective QoS model, including crisp and fuzzy numbers, by using the Genetic algorithm and Pareto dominance.

3.4. Service recommendation

Recommendation systems have been widely used for product recommenders in Netflix, YouTube, and Spotify, Amazon, or content recommenders in social media platforms like Instagram, Facebook, and Twitter (Jiang et al., 2011). Similar methods have been employed in SC for finding the user's desired service in terms of QoS values. In such a situation, service recommender systems try to find the incomplete QoS values by using other service users' experiences (Kazem et al., 2015). We categorized the existing approaches into three classes: Users or services similarity, QoS matrix completion methods, and Ranking.

3.4.1. Users or services similarity (USS)

Rong et al. (2009) utilize the collaborative filtering (CF) method for finding services using users' similarities, association rules, and historical transactions. Collaborative filtering technique finds similar users or services to calculate QoS values based on ratings of similar users/services. The basic idea of Chen et al. (2013b) is that the near users (geographically) can experience a similar quality of service than far users. This idea can be justified because the users in almost a same location can receive network traffic in nearly a same quality. For identifying the similarity between regions, Pearson Correlation Coefficient (PCC) (Chen et al., 2013b) is employed. Also, in Karim et al. (2015), similarly between services using WSDL files is discussed, and the Jaccard similarity measure is applied.

3.4.2. QoS matrix completion

An important problem in CF is handling new users with no previous interaction information (Yu, 2012). To address this problem, Yu et al. (2013a) retrieve a large QoS matrix from a small portion of existing QoS records using the Trace Norm Regularized Matrix Factorization algorithm. Zheng et al. (2009) integrate the item-based approach with the user-based. They conduct a large-scale real-world experiment with 21,197 public web services and use an improved PCC for finding similarity. To predict the missing values, a neighborhood-integrated matrix factorization (Zheng et al., 2012) method for QoS value prediction is proposed by considering the users' previous observation on the quality of services. To achieve a higher prediction accuracy, they combine neighborhood-based and the model-based CF. Additionally, Chen et al. (2017a) propose a personalized QoS model to solve the cold-start problem by using services/users geographical locations.

3.4.3. Ranking

Kuter and Golbeck (2009) propose an SC method, based on users' rating for a given service. For trust calculation, they use synthetic data adopted from FilmTrust (Golbeck, 2006). Li and Wang (2015) use Kalman filtering, also known as linear quadratic estimation (LQE), to generate predicted values. For sake of scalability, they propose a similarity calculation approach based on Euclidean distance for the large-scale dataset. Liu et al. (2016) propose a Cultural Genetic algorithm for service composition. Moreover, they decrease the size of service pool through discovering the top *K* composite service using a technique named Case-Based Reasoning (CBR) (Kolodner, 2014). Also, researchers in Hashmi et al. (2016) propose a social network-enabled negotiation based on recommendations from the social network. They propose multi-objective, multi-agent service negotiation with the presence of fake ratings.

3.5. Other approaches

To deal with incomplete information and inaccurate QoS data, Guoping et al. (2012) use Grey system theory (a method for modeling and forecasting small sample time series). Ramacher and Mönch (2012) explore an MDP model to deal with the uncertainty of response time and solve the obtained model with mixed-integer programming, ILOG OPL 6.3 and CPLEX solver. Tan et al. (2014) propose a GA-based approach, called rGA, with a dynamic-length chromosome to support the on-the-fly partial exploration of state-space; hence, after changing QoS values, they can suggest alternative composite service in a timely manner. Chen et al. (2016a) propose a robust optimization to deal with QoS uncertainty based on Bertsimas and (Bertsimas and Sim, 2004) optimization method. To this aim, they consider an interval for QoS variation and find the optimal composite service according to the number of uncertain parameters and a conservation-degree (Γ) parameter. An adaptive service composition framework based on wEASEL (contExt Aware web Service dEscription Language) (Urbieta et al., 2017) for representing user's tasks is developed. Chen et al. (2018b) propose dynamic service composition along with a mobile application named GoCoMo, to self-organize the process of SC in bluetooth-based mobile ad hoc networks.

One of the important aspects of service composition in distributed environments is to guarantee a given level of security assurance for the services. To handle this requirement, Anisetti et al. (2013) proposed the concept of virtual certificate (it does not involve any real testing activity on the composition) for security certification-based service composition. They extend the BPEL (Business Process Execution Language) process specification with the security parameter. BPEL is a defacto standard for representing web service composition. Notably, this approach needs a certification authority to verify the security properties of services by persistent QoS attributes certification. In Pino et al. (2017) a pattern-driven verification approach is presented for designing, adapting and verifying the security attributes of isolated services. The approach considers both service-level security and workflow-level security using the concept of secure service composition (SCO) patterns. Service-level security verification is based on digital security certificates, which are assigned to services. A certificate includes the service, the API, the service level security property, and the evidence which proves the service is secure. Another example of using certificate approach has been presented in Stephanow and Khajehmoogahi (2017) where continuous security certification of software-as-a-service (SaaS) tries to repeatedly and automatically validate security requirements. Certification-based assurance guarantees the quality of composition by QoS evaluation based on a continuous collection of evidence on the behavior of the system. Recently, Anisetti et al. (2020) proposed a cost-effective deployment of certified cloud composite services. They not only propose a cost-effective composition model, but they also consider the security as an uncertain QoS parameter.

4. Analysis of systematic literature review (SLR) results and technical discussion

In this section, we provide SLR results, technical discussion and comparison on similar works within a category. It should be noted that, we report the results by answering the research questions listed in Section 2.1.2.

4.1. Sources of uncertainty

RQ1: What are the main reasons for the uncertainty according to various service composition environments, including cloud, IoT, Mobile, etc.?

Sources of uncertainty can be categorized in four groups: system, goals (like way of requirement extraction and future goal changes), context (like noise in sensing and different information sources), and humans (such as human behavior and multiple ownership) (Weyns, 2020; Mahdavi-Hezavehi et al., 2017). It is worth mentioning that simplifying assumptions, model drift, incompleteness, future values of parameters, adaptation functions,

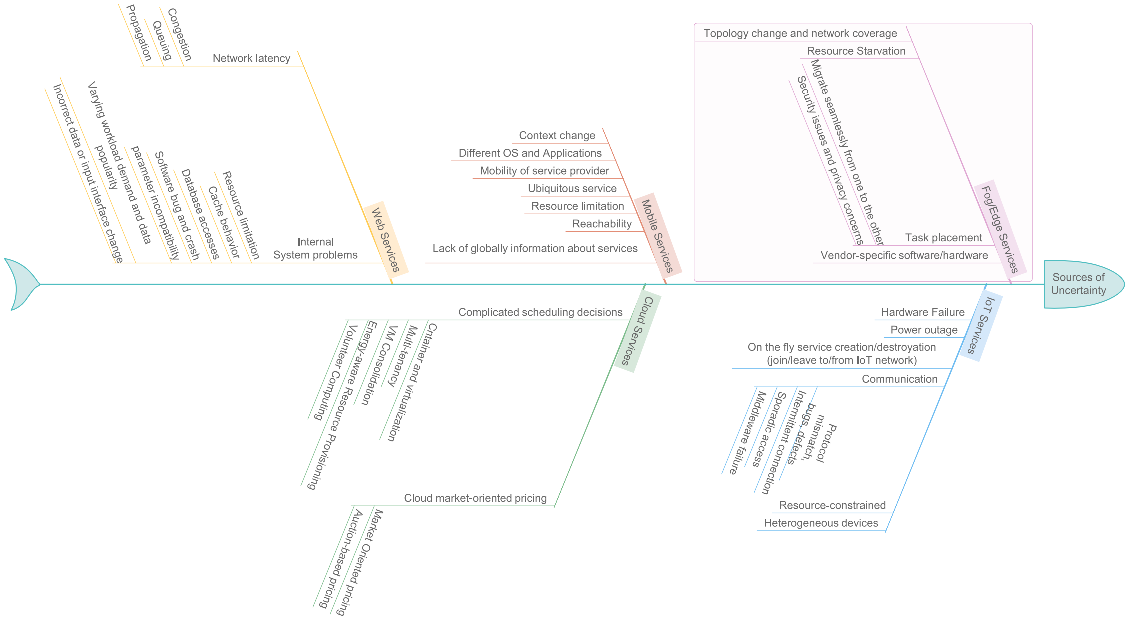


Fig. 6. Reasons of uncertainty in Web, Cloud, Mobile and IoT services.

decentralization, and automatic learning are system-level sources of uncertainty (Weyns, 2020). Fig. 6 indicates the main reasons for the uncertainty of QoS values in web, cloud, mobile, and IoT environments. In the traditional web service environments, a service was hosted in a distant network that would provide services with fixed resource capacity. The factors like network delay and the internal crash were the main reasons for QoS fluctuation. With the introduction of cloud computing architectures, flexible service provisioning with virtually unlimited resources was replaced with previous simple web service architecture. Although, the concepts like Volunteer computing (Elhabbash et al., 2017), Federated Cloud (Toosi et al., 2011), Cloud market (Tafsiri and Yousefi, 2018), VM Consolidation (Sharma et al., 2019), Multi-tenancy (Kumar et al., 2018), and energy-aware resource provisioning (Zhang et al., 2019b) help flexible service delivery, they caused potentially uncertainty in QoS values. Meanwhile, mobile services in an ad-hoc network grew using smartphones and mobile vehicular systems (Gai et al., 2018). In a mobile scenario, the composer placed in a device with mobility, identifies the existing services (Urbieta et al., 2017). In this environment, the uncertainty of mobile services mostly relates to movements (Tabassum et al., 2019) of service requesters or service providers (Deng et al., 2016). Additionally, the lack of stable and global information about available services can lead to uncertainty.

Compared with cloud and mobile, the majority of the services suppliers in IoT are intelligent objects located in varying network infrastructure (Li et al., 2012; White et al., 2018). Due to hardware failure, sporadic access, and intermittent network connection, IoT services are usually more uncertain than cloud. In other words, because of increasingly ubiquitous wireless connectivity, IoT nodes may be occasionally disconnected. Other reasons for the uncertainty of IoT services can be investigated in their hardware; the devices equipped with different operating systems; applications come from various vendors (Huang et al., 2017). Also, in Fog/Edge-based environments, the factors like changing task placement strategy, topology change and network coverage, and resource starvation are the main reasons of uncertainty in delivered QoS values.

4.2. Adopted uncertainty-aware approaches

RQ2: What approaches have been applied to deal with uncertainty?

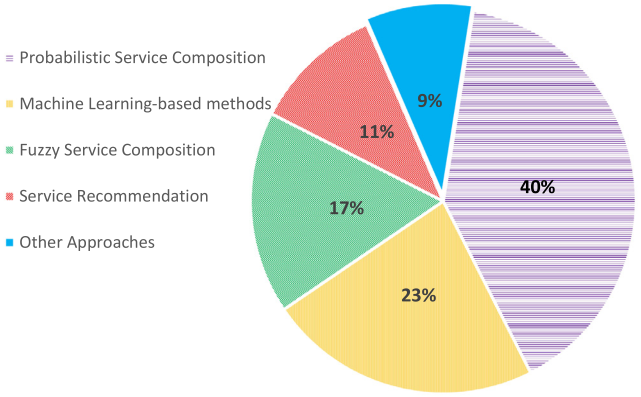


Fig. 7. Percentage of adopted uncertainty-aware approaches.

Typically, an uncertainty-aware service composition includes two various phases: *QoS Uncertainty Model Construction* and *Service Selection*. More precisely, the QoS uncertainty modeling phase determines how uncertain QoS attributes can be estimated or predicted, while the service selection phase identifies which candidate services provide the best composition according to utility function and users' constraints.

4.2.1. QoS Uncertainty model construction phase

According to primary studies, we can see that the uncertainty is arisen from either the variability of the observed QoS values (in an open and dynamic environment) or lack of knowledge about QoS of service (e.g., a new service) (Weyns, 2020). While the former is handled through probability theory, the latter is addressed by the possible theory which focuses on set-valued representations (Baudrit et al., 2006). From Fig. 7, we can see that probabilistic and machine learning are the dominant approaches obtained by authors to construct the QoS uncertainty model.

In the probabilistic approach, *single/multi-value representation* and *standard statistical distributions* have been frequently used to model the QoS attributes. Although these methods are straightforward and easy for QoS estimation, they do not reflect the real-world behaviors of QoS attributes. Furthermore, considering

QoS attributes under the assumption that they follow a known distribution is not always possible in real environments where QoS statistical distribution can take any shape. Some studies, with no assumption of known distribution, try to estimate the QoS values. However, QoS prediction using probabilistic methods needs to create a clear mathematical expression directly, which results in a nonlinear problem (Zhang, 2010). Another method in this approach is forming the probability mass function (PMF) for each QoS attributes. The goal of this method is to count the frequency of occurrence of a typical QoS value in historical data. Using this, the value with the highest frequency is considered as the value of that QoS attribute. Although this method, like previous ones, provides an easy-to-model QoS estimation, forming bins (especially for QoS attributes with continuous values) is not always straightforward and may lead to inefficiency.

The intensive need for **sufficient** and **reliable** data in this approach guided researchers to use fuzzy-based systems. Fuzzy logic will be applied in situations where a model should reflect an expert's opinion, while cannot collect sufficiently large statistical data to apply a probability theory-based approach (Ciszkowski et al., 2012). In the literature, many researchers tried to consider the QoS attributes as fuzzy numbers. They have adopted different types of fuzzy numbers representation, membership functions, and defuzzification methods to model uncertainty in QoS attributes. However, in practice, fuzzy-based SC needs to be set up by the experts for additional analysis and interpretation. Fuzzy multi-criteria service selection has been vastly applied to SC. Fuzzy AHP and Fuzzy TOPSIS are examples of fuzzy multi-criteria decision-making methods. Furthermore, we discovered that the Fuzzy set theory is occasionally used with other well-known approaches to form QoS models that we called *Fuzzy-enabled Systems*. Game theory, genetic algorithm, and neural networks are some examples of adopted approaches in conjunction with fuzzy set theory. Also, hierarchical fuzzy systems have been applied to overcome the problem of scalability of fuzzy systems.

The growing complexity of service computing-based environments, as well as the increasing tendency of automation through learning, has attracted researchers to construct the QoS model based on Machine Learning algorithms. From Fig. 7, Machine Learning-based methods have devoted the second most prevalent approaches in the literature. We discovered that researchers had exploited Machine Learning techniques for two purposes: **prediction** of QoS values whether missing or future value, and **uncertain services pruning**. For instance, Guo et al. (2017) uses autoregressive integrated moving average for QoS forecasting, whereas (Khanouche et al., 2019) use k-means for filtering the unfavorable services.

Additionally, we found that some existing studies attempt to inspire recommendation systems for QoS model construction. They have relied on users or services similarity utilizing some similarity measures, matrix completion methods, and ranking approaches. The likeness of clients is calculated by using the similarity of their QoS experiences. Also, the similarity between the two services is measured based on the similarity of their WSDL files. From the literature, service recommender approaches often suffer from cold start problems (a typical problem in collaboration filtering technique) where a new service/user has no composition history. In the literature, to overcome the incomplete ratings in a user-item matrix, the methods like neighborhood-integrated matrix factorization (Zheng et al., 2012) and Trace Norm Regularized Matrix Factorization (Yu et al., 2013a) are employed. The main idea behind the QoS prediction in service recommender is when a service operates similarly to another service or a user's request is similar to another users' request, the QoS of services can be similar. In addition to the cold start problem, recommender systems have been faced the following challenges, especially for IoT service: monitoring subsystem to collect the user-service rating information imposes excess costs and consumes resources of service providers. Also, users of a service are not limited to humans, while in the IoT environment, most of the service users are intelligent devices. Therefore, scoring for QoS attributes like reputation needs more interpretation according to the environment.

4.2.2. Service selection phase

From Table 2, in the class of *probabilistic* and *fuzzy*, the majority of studies used mathematical optimization methods or (meta-)heuristic algorithms (Xu et al., 2017) to find the (near-)optimal composition. The heuristic and meta-heuristic approaches (Wang, 2011) find a composite service in a timely manner even in large-scale problems (i.e., problems with plenty of tasks in a workflow or a large number of candidate services). However, these approaches do not guarantee finding best composition and usually end with a near-optimum solution (Ghazanfari et al., 2007). Broadly speaking, heuristic algorithms may have two limitations: falling into local optimum and lacking memory-efficiency. To mitigate these drawbacks, meta-heuristic algorithms using some high-level strategies guide search process according to the feedback from the objective function and prior performance (usually the terms exploration and exploitation are used as two important mechanisms for obtaining a proficient search). Simulated Annealing (SA) (Hwang et al., 2015; Deng et al., 2016), single-objective and multi-objective evolutionary algorithm (Zhao et al., 2015; Peng et al., 2017), Particle Swarm Optimization (PSO) (Sun et al., 2013; Mezni and Sellami, 2018), Genetic Algorithm (GA) (Tan et al., 2014; Schuller et al., 2014; Hashmi et al., 2016; Liu et al., 2016; Xu et al., 2017; Sun et al., 2018; Xu et al., 2018a; Niu et al., 2019), and NSGA-II (Efstathiou et al., 2014) are examples of these approaches. Unlike (meta-)heuristic approaches, the mathematical optimization methods like Mixed Integer Programming (MIP) (Razian et al., 2020a) or Integer Programming (IP) (Schuller et al., 2014; Wang et al., 2017, 2018a), result in optimum composition and are best-suited for small-scale scenarios. It is notable that these time-consuming approaches would not fit in scenarios that the user needs a composite service in a highly timely manner.

Furthermore, we observed that in the ML category, many researchers used MDP resolution techniques (such as reinforcement learning) where MDP parameters are not completely available. In the literature, authors normally employed Q-learning as a model-free learning algorithm which requires no knowledge of the system dynamics. From the Service Recommender approaches, PCC-based similarity analysis is the most adopted method for finding similarity. The similarity between users may obtain from similarity measures like the PCC and cosine similarity. Table 2 shows the studies included in each approach and describes the solvating method applied in each study.

4.3. Metrics and dimensions

RQ3: How do QoS parameters, dimensions, and metrics differ with the approaches?

We summarize the metrics and dimensions addressed in each primary study in Table 3. In the following, we investigate these metrics in detail.

QoS Attributes. Table 4 shows the QoS attributes used in PSs. From Fig. 8, we can see that the majority of studies (61%) considered the response time as an uncertain QoS attribute. Furthermore, we observed that availability (28%), reliability (22%), throughput (20%), price (19%), and reputation (15%) had been modeled under uncertainty. However, 21% of studies did not point out the type of QoS attributes explicitly. Despite the high importance of energy consumption and security/safety, these attributes have not received much attention (only 2% and 6%, respectively).

Table 2
Primary studies approaches and solving methods.

Fuzzy-based SC	FQoS	Fuzzy min–max composition (Tripathy and Tripathy, 2018), Fuzzy GA (Xu et al., 2017), Interval number using multi-objective GA (Niu et al., 2019), Self-optimization (de Gyvés Avila and Djemame, 2013), Fuzzy inference (Şora and Todinca, 2015), AODV protocol (Veeresh et al., 2017)
	FES	Game theory (Johannes et al., 2015), Triangular fuzzy GA (Xu et al., 2018a) Fuzzy GraphPlan (Zhu et al., 2018), Fuzzy LinPreRa evolutionary algorithm (Zhao et al., 2015), Fuzzy Neural Networks (Luo et al., 2015), Hierarchical fuzzy (Pernici and Siadat, 2011), Rule based fuzzy expert Sugeno model (Prochart et al., 2007)
	MCSS	Fuzzy AHP - interval-based (Zhang et al., 2012), Fuzzy TOPSIS (Zhang et al., 2011), Fuzzy and Rough Set (Mu et al., 2014), Interval-based PROMETHEE and GA (Jian et al., 2016)
Machine Learning-based methods	Clu.	Kmeans with search tree (Khanouche et al., 2019), OPTIC with heuristic algorithm (Xia et al., 2011)
	Reinforce	Partially Observable MDP (Lei et al., 2015b), Single and multiple policy multi-objective composition scenarios (Mostafa and Zhang, 2015), Multi-Agent RL with Qlearning, ϵ-greedy exploration algorithms (Mahfoudh et al., 2018), Q-Learner (Wang et al., 2010), Moustafa and Zhang (2012), Wei et al. (2017), Yu et al. (2013b) Time-based Learning method (Lei et al., 2015a), deep RL (Moustafa and Ito, 2018), Two-layer Hierarchical Reinforcement Learning (D'Angelo et al., 2020) Q-learning based on gaussian process (Wang et al., 2015c), SARSA(λ) (Wang et al., 2016), time-based (Q-learning) (Lei et al., 2014)
	Class.	Decision Tree and matrix factorization (Yu, 2012), RBF neural networks + improved K-means (Zhang, 2010), Classification and Regression Trees (CART), Random Forest (Efstathiou et al., 2014), Anomaly detection using Isolation Forest Algorithm (Razian et al., 2020a)
	Regression	PSPAS with shortest path algorithm (Zhang et al., 2018), Linear Regression, MARS, NSGA-II (Efstathiou et al., 2014), ARIMA-BASED Time Series and GA (Sun et al., 2018), ARIMA and Skyline using 0-1 MIP (Guo et al., 2017), Multivariate ARIMA and Holt-Winters using R (Ye et al., 2016)
Probabilistic Service Composition	Constant Value for QoS	Bayesian network (BN), extended Shenoy–Shafer to solve Hybrid influence diagram (Ye et al., 2014), Improved BN, SMILE engine (Chen et al., 2013a), Anytime algorithm using DFS (Kil et al., 2016), Average value at risk MILP (Wiesemann et al., 2008), MDP with FSM, temporal logic PCTL, probabilistic model checking with PRISM (Calinescu et al., 2010; Li et al., 2012), Probability of context change, Njima et al. (2016), Hofstedes cultural dimension MIP (Wang et al., 2018a), LPSolve and NSGA-II (Ramacher and Mönch, 2014), Skyline, Int. Prog., heuristic algorithm (Wang et al., 2012), Expected/Pessimistic value Estimation (Chen et al., 2016b), Bubnicki model (Falas and Stelmach, 2013), Back tracking search (Yasmina et al., 2018), Entropy and variance IP (Wang et al., 2017), Expected value, Entropy, and Hyper-Entropy: (Malik and Medjahed, 2010a), Finding providers' reputation (Malik and Medjahed, 2010b), Finding uncertain services (Gong et al., 2014), Lp-Solve (Wang et al., 2011), p-R-forest (p-dominant Service Skyline with R-tree data structure) (Yu and Bouguettaya, 2010), PSO + skyline (Sun et al., 2013) Linear Programming (LP) (Cardellini et al., 2011)
	Known Distr.	Stochastic timed colored Petri net (Wu et al., 2009), known PDF (Ye and Li, 2018), Normal distribution and Greedy adaption heuristic (Schuller et al., 2012), Triangular, Uniform Distribution and GA (Schuller et al., 2014), Uniform distribution, heuristic and metaheuristic algorithm (Wang, 2011), Normal distribution, simulated annealing (Deng et al., 2016)
	Unknown Distr.	Quantile-based measure, MIP and iterative approach (Ramacher and Mönch, 2013), Gaussian Kernel Density estimation and fast Fourier transform (Zheng et al., 2010a), Kernel density estimation, PSO (Mezni and Sellami, 2018), Restricted Boltzmann Machine, evolutionary algorithm (Peng et al., 2017), PDF-Calculation with DFS algorithm (Zheng et al., 2016), histograms (Zheng et al., 2011), Dynamic histograms, Chebyshev's inequality (Elhabbash et al., 2017)
	PMF	Dynamic prog. and Greedy method (Hwang et al., 2007), Prolog (Ivanović et al., 2014), Similarity Analysis, simulated annealing (Hwang et al., 2015)
	Sim.	Particle Filter based Algorithm (Yao and Sheng, 2011), Bootstrap-Based Simulations, T Location-Scale Sampling-Based Simulations (Rosario et al., 2008), Importance Sampling technique (Wang et al., 2015b)
Recommendation	Matrix	Collaborative filtering (CF), neighbor PCC (Zheng et al., 2009), Matrix factorization, Clustering, geographical neighbors improved PCC (Chen et al., 2017a), Neighborhood-integrated matrix factorization (Zheng et al., 2012), Trace Norm Regularized Matrix Factorization (Yu et al., 2013a)

(*continued on next page*)

Table 2 (*continued*).

	Ranking	Case-Based Reasoning — Manhattan distance GA based approach (Liu et al., 2016), Ranking Kalman filtering — Euclidean distance (Li and Wang, 2015), Modified HTN and branch-and-bound, SHOP2 planning system (Kuter and Golbeck, 2009), Ranking within a category, recommendations from the social network, GA (Hashmi et al., 2016)
	USS	Similar WSDL files by Jaccard similarity (Karim et al., 2015), Users' physical PCC for region similarity (Chen et al., 2013b), Collaborative filtering and association rules (Rong et al., 2009)
Others	–	Bertsimas and Sim Robust Optimization (Chen et al., 2016a), MDP and Mixed Integer Programming (Ramacher and Mönch, 2012), Backward-chaining, NS3 nimulator (Chen et al., 2018b), Conversation-based SC IOPE Hybrid-Cosine method, (Urbieta et al., 2017), Grey sequence prediction model (Guoping et al., 2012), Deterministic finite automata/recovery strategy, GA (Tan et al., 2014) Virtual certificate and Symbolic Transition System (STS) (Anisetti et al., 2013), Certification-based assurance (Anisetti et al., 2020) Certification-based assurance and rule-based reasoning process based on the Rete algorithm Drools (Pino et al., 2017)

Table 3
Metrics and dimension used in primary studies.

Metrics and dimension	Studies
Multi-objective	Wiesemann et al. (2008), Jian et al. (2016), Calinescu et al. (2010), Yu and Bouguettaya (2010), Efstathiou et al. (2014), Xu et al. (2018a), Zhao et al. (2015), Sun et al. (2013), Mu et al. (2014), Mostafa and Zhang (2015), Ramacher and Mönch (2014), Guo et al. (2017), Hashmi et al. (2016) and Niu et al. (2019)
Context-aware	Njima et al. (2016), Urbieta et al. (2017), Mahfoudh et al. (2018), Şora and Todinca (2015), Tripathy and Tripathy (2018), Chen et al. (2018b) and Luo et al. (2015)
Adaptive	Calinescu et al. (2010), Cardellini et al. (2011), Moustafa and Ito (2018), Wei et al. (2017), Urbieta et al. (2017), Wang et al. (2010), Yu et al. (2013b), Wang et al. (2012), Tan et al. (2014), Mahfoudh et al. (2018), Lei et al. (2015b), Peng et al. (2017), de Gyvés Avila and Djemame (2013), Tripathy and Tripathy (2018), Chen et al. (2018b), Ye et al. (2016), Pernici and Siadat (2011), Moustafa and Zhang (2012), Li and Wang (2015), Veeresh et al. (2017), Wang et al. (2015c, 2016), Lei et al. (2014), Zhang et al. (2018), Yao and Sheng (2011), Mostafa and Zhang (2015), Elhabbash et al. (2017), Chen et al. (2016b), Xia et al. (2011), Anisetti et al. (2020), Razian et al. (2020a) and D'Angelo et al. (2020)
Scalability	Moustafa and Ito (2018), Mezni and Sellami (2018), Tan et al. (2014), Yu and Bouguettaya (2010), Peng et al. (2017), Wang (2011), Tripathy and Tripathy (2018), Xu et al. (2018a), Pernici and Siadat (2011), Zhao et al. (2015), Chen et al. (2017a), Sun et al. (2013), Wang et al. (2015c), Ramacher and Mönch (2014), Elhabbash et al. (2017), Chen et al. (2016b), Guo et al. (2017), Hashmi et al. (2016), Xu et al. (2017), Liu et al. (2016), Zheng et al. (2016), Luo et al. (2015), Yu et al. (2013a), Zheng et al. (2009), Xia et al. (2011), Chen et al. (2013b), Zheng et al. (2012), Pino et al. (2017), Razian et al. (2020a) and D'Angelo et al. (2020)
Multi-provider	Yu and Bouguettaya (2010), Mahfoudh et al. (2018), Ye et al. (2014), Efstathiou et al. (2014), Wang (2011), Chen et al. (2018b), Ye et al. (2016), Mu et al. (2014), Veeresh et al. (2017), Elhabbash et al. (2017), Hashmi et al. (2016), Deng et al. (2016), Karim et al. (2015), Razian et al. (2020a), D'Angelo et al. (2020) and Anisetti et al. (2020)
Motivation example	Njima et al. (2016), Wu et al. (2009), Urbieta et al. (2017), Yu et al. (2013b), Wang et al. (2012), Tan et al. (2014), Wang et al. (2011), Yu and Bouguettaya (2010), Schuller et al. (2012), Wang et al. (2018a), Şora and Todinca (2015), Lei et al. (2015b), Ye et al. (2014), Efstathiou et al. (2014), Chen et al. (2013a), Wang (2011), Tripathy and Tripathy (2018), Prochart et al. (2007), Chen et al. (2018b), Ye et al. (2016), Zheng et al. (2010a), Xu et al. (2018a), Zhao et al. (2015), Schuller et al. (2014), Chen et al. (2017a), Mu et al. (2014), Veeresh et al. (2017), Lei et al. (2014), Malik and Medjahed (2010a), Zhang et al. (2018), Li et al. (2012), Mostafa and Zhang (2015), Johannes et al. (2015), Yasmina et al. (2018), Elhabbash et al. (2017), Kuter and Golbeck (2009), Chen et al. (2016b), Hashmi et al. (2016), Deng et al. (2016), Xu et al. (2017), Niu et al. (2019), Liu et al. (2016), Zhang et al. (2012), Chen et al. (2013b), Kil et al. (2016), Karim et al. (2015), Pino et al. (2017), Anisetti et al. (2013), Calinescu et al. (2010), Cardellini et al. (2011), Anisetti et al. (2020), Razian et al. (2020a) and D'Angelo et al. (2020)

Scalability. We can observe from Table 3 that 30% of PSs explicitly discussed scalability. The intensive increment in the number of service APIs in cloud and IoT-enabled smart environment need algorithms to work effectively and efficiently. Besides, factors like the number of features in service selection and the amount of historical data play an important role in converting SCP to a large-scale problem. Fig. 9 shows the tendency of researchers to consider scalability as a core feature in their solutions (only two studies between 2007 to 2010 versus 13 studies between 2017 to 2020). More precisely, 32.14% of studies in the category of Probabilistic explicitly considered scalability. Furthermore 25% of studies in the class of Service Recommender, 21.43% of studies in the class of Machine Learning, and 21.43% of studies in the Fuzzy service composition class considered scalability in their methods.

The results show that all approaches potentially can present scalable composition. However, the crucial aspect is how to solve the resulted uncertainty-aware model. It is worth to be mentioned that 40% of scalable approaches use (meta-)heuristic algorithm as a solving method.

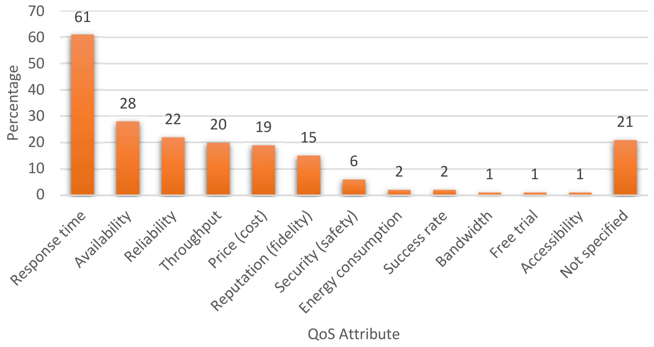


Fig. 8. Percentage of each QoS treated as an uncertain attribute (the summation is greater than 100%, because more than one QoS attributes are considered in some papers).

Table 4
QoS attributes used in primary studies.

QoS attribute	Studies
Availability	Moustafa and Ito (2018), Zhang et al. (2011), Njima et al. (2016), Zhang (2010), Urbieta et al. (2017), Zhu et al. (2018), Jian et al. (2016), Schuller et al. (2012), Peng et al. (2017), Chen et al. (2013a), Prochart et al. (2007), Xu et al. (2018a), Pernici and Siadat (2011), Schuller et al. (2014), Moustafa and Zhang (2012), Wang et al. (2015c), Li et al. (2012), Yao and Sheng (2011), Mostafa and Zhang (2015), Hashmi et al. (2016), Deng et al. (2016), Xu et al. (2017), Niu et al. (2019), Liu et al. (2016), Zhang et al. (2012), Falas and Stelmach (2013), Ye and Li (2018) and Khanouche et al. (2019)
Reliability	Moustafa and Ito (2018), Zhang et al. (2011), Hwang et al. (2007), Zhang (2010), Wu et al. (2009), Wang et al. (2012), Efstathiou et al. (2014), Tripathy and Tripathy (2018), Xu et al. (2018a), Hwang et al. (2015), Moustafa and Zhang (2012), Mu et al. (2014), Wang et al. (2015c), Hashmi et al. (2016), Guoping et al. (2012), Xu et al. (2017), Liu et al. (2016), Cardellini et al. (2011), Zhang et al. (2012), Ye and Li (2018) and Khanouche et al. (2019)
Response time	Moustafa and Ito (2018), Zhang et al. (2011), Sun et al. (2018), Wang et al. (2015b), Hwang et al. (2007), Wei et al. (2017), Chen et al. (2016a), Zhang (2010), Wu et al. (2009), Wiesemann et al. (2008), Wang et al. (2010), Yu et al. (2013b), Zhu et al. (2018), Jian et al. (2016), Wang et al. (2012), Schuller et al. (2012), Ramacher and Mönch (2012), Wang et al. (2017), Efstathiou et al. (2014), Peng et al. (2017), Wang (2011), de Gyvés Avila and Djemame (2013), Tripathy and Tripathy (2018), Chen et al. (2018b), Ye et al. (2016), Rosario et al. (2008), Zheng et al. (2010a), Xu et al. (2018a), Pernici and Siadat (2011), Hwang et al. (2015), Schuller et al. (2014), Li and Wang (2015), Chen et al. (2017a), Mu et al. (2014), Veeresh et al. (2017), Wang et al. (2015c), Zhang et al. (2018), Mostafa and Zhang (2015), Ramacher and Mönch (2013, 2014), Elhabbash et al. (2017), Gong et al. (2014), Guoping et al. (2012), Xu et al. (2017), Ivanović et al. (2014), Niu et al. (2019), Liu et al. (2016), Zhang et al. (2012), Zheng et al. (2011, 2016), Ye and Li (2018), Luo et al. (2015), Yu et al. (2013a), Zheng et al. (2009), Khanouche et al. (2019), Yu (2012), Chen et al. (2013b), Zheng et al. (2012), Karim et al. (2015), Cardellini et al. (2011) and Razian et al. (2020a)
Price (cost)	Zhang et al. (2011), Hwang et al. (2007), Zhang (2010), Wu et al. (2009), Wiesemann et al. (2008), Wang et al. (2010), Cardellini et al. (2011), Schuller et al. (2012), Ye et al. (2014), de Gyvés Avila and Djemame (2013), Tripathy and Tripathy (2018), Ye et al. (2016), Xu et al. (2018a), Mu et al. (2014), Mostafa and Zhang (2015), Johannes et al. (2015), Guoping et al. (2012), Ye and Li (2018) and Khanouche et al. (2019)
Reputation (fidelity)	Zhang et al. (2011), Hwang et al. (2007), Ye et al. (2014), Xu et al. (2018a), Pernici and Siadat (2011), Hwang et al. (2015), Mu et al. (2014), Malik and Medjahed (2010a), Elhabbash et al. (2017), Kuter and Golbeck (2009), Hashmi et al. (2016), Guoping et al. (2012), Niu et al. (2019), Zhang et al. (2012) and Ye and Li (2018)
Throughput	Sun et al. (2018), Chen et al. (2016a), Yu et al. (2013b), Zhu et al. (2018), Jian et al. (2016), Wang et al. (2012), Schuller et al. (2012), Ye et al. (2014), Wang et al. (2017), Peng et al. (2017), Ye et al. (2016), Xu et al. (2018a), Schuller et al. (2014), Li and Wang (2015), Veeresh et al. (2017), Wang et al. (2015c), Hashmi et al. (2016), Xu et al. (2017), Khanouche et al. (2019) and Zheng et al. (2012)
Bandwidth	Zhang (2010)
Energy	Efstathiou et al. (2014) and de Gyvés Avila and Djemame (2013)
Security (safety)	Guoping et al. (2012), Anisetti et al. (2013), Mu et al. (2014), Pino et al. (2017) and Anisetti et al. (2020)
Free trial	Mu et al. (2014)
Accessibility	Hashmi et al. (2016)
Success rate	Xu et al. (2018a, 2017)
Not specified	Mezni and Sellami (2018), Yu et al. (2013b), Tan et al. (2014), Wang et al. (2011), Yu and Bouguettaya (2010), Wang et al. (2018a), Şora and Todinca (2015), Zhao et al. (2015), Sun et al. (2013), Wang et al. (2015c, 2016), Lei et al. (2014), Li et al. (2012), Yasmina et al. (2018), Chen et al. (2016b), Guo et al. (2017), Lei et al. (2015a), Xia et al. (2011), Kil et al. (2016), Calinescu et al. (2010) and D'Angelo et al. (2020)

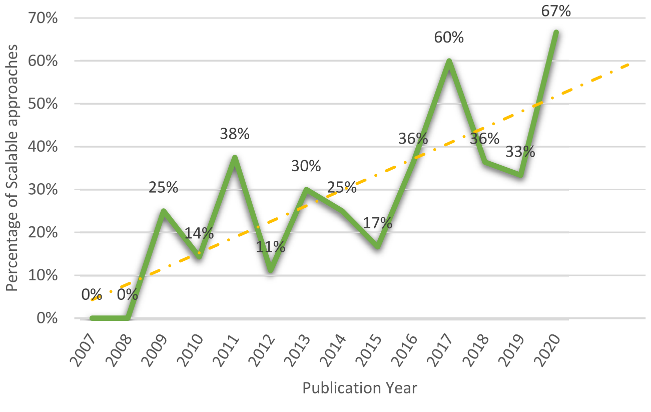


Fig. 9. Frequency of scalable approaches by year.

Objective function. From the literature, the majority of PSs consider a simple additive weighted (SAW) method for QoS aggregation instead of using multi-objective approaches (like finding the Pareto optimum or multi-criteria decision making). Unlike SAW, Pareto optimality can explicitly manage multi-objective models for composition without the need to put weights on the objectives. We found that only a few percentages (14%) of existing studies attempted to model QoS as a multi-criteria (Mu et al., 2014) or multi-objective problem (Sun et al., 2013). Unlike single objective models, exposing the set of possible alternative compositions enables decision-makers to choose through possible composite services with a trade-off between the conflicting objectives.

Motivation scenario/example. To demonstrate the role of service composition in service-oriented architecture (Şora and Todinca, 2015), many researchers (49%) consider a motivation scenario or example (See Table 3). As shown in Fig. 10, Trip planning/travel booking (Tan et al., 2014; Wang et al., 2011; Yu and Bouguettaya, 2010; Ye et al., 2014; Xu et al., 2018a; Zhao et al., 2015; Malik and Medjahed, 2010a; Chen et al., 2016b; Hashmi et al., 2016; Urbieta et al., 2017) and online shopping (Wu et al., 2009; Wang et al., 2012; Chen et al., 2013a, 2018b; Ye et al., 2016; Yasmina et al., 2018; Xu et al., 2017; Liu et al., 2016; Kil et al., 2016) are the most used scenario examples. Furthermore, brokerage and service oriented architecture (Cardellini et al., 2011; Njima et al., 2016; Schuller et al., 2012; Zheng et al., 2010a; Schuller et al., 2014; Niu et al., 2019), cloud market services/volunteer computing (Johannes et al., 2015; Karim et al., 2015; Mostafa and Zhang, 2015; Elhabbash et al., 2017), E-Health system (Kuter and Golbeck, 2009; Anisetti et al., 2020; Razian et al., 2020a), video

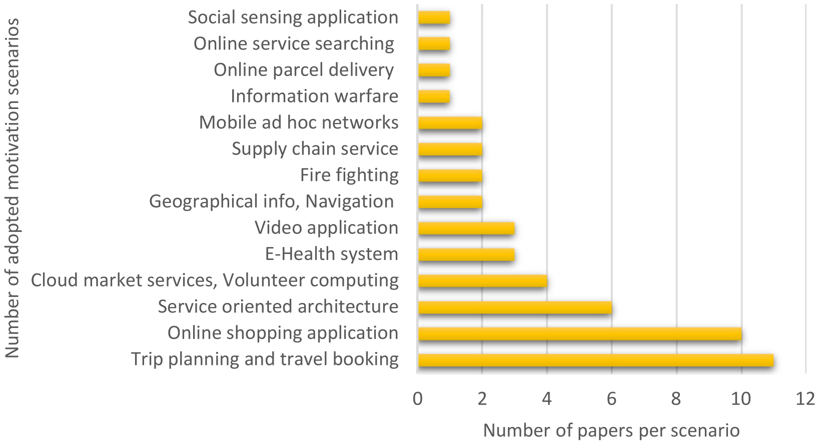


Fig. 10. Frequency of motivation scenarios.

application (Prochart et al., 2007; Zhang et al., 2018; Deng et al., 2016), geographical information and mobile navigation (Zhang et al., 2012; Chen et al., 2017a), fire fighting (Efstathiou et al., 2014; Li et al., 2012), supply chain service (Yu et al., 2013b; Lei et al., 2014), mobile ad hoc networks (Wang, 2011; Veeresh et al., 2017), information warfare (Lei et al., 2015b), online parcel delivery (Tripathy and Tripathy, 2018), online service searching (Chen et al., 2013b) and Social sensing application (D'Angelo et al., 2020) are other scenarios used in the literature.

Multi-source. In multi-source service composition, the broker composes services that have come (provided) from distributed sources/locations. From the literature, there exist three paradigms of multi-source service composition: multi-cloud (i.e., services provided by distributed data-centers or content delivery networks), mobile ad-hoc networks (i.e., services provided by nearby mobile devices), and IoT devices (services provided by IoT smart cities and Industry 4.0). According to Table 3, only a few percentages of primary studies (16%) have pointed to multi-sources service composition explicitly.

Context-aware. Context-aware models focus on various application's domain information in modeling QoS attributes. Consider a delay-sensitive application; the penalty cost of QoS violence would be different from a typical application. Also, a context-aware approach may consider the environmental parameters of service requester/provider in QoS modeling. For example, the location information of a mobile service user can be taken into account in assessing the availability or response time of service. From Table 3, we noticed that only the minority of the aforementioned studies explicitly considers *context* in SC under uncertainty (7%).

Adaptation. Adaptation helps software systems to tackle the uncertainty in highly changing and evolving environments (Menasce et al., 2011). From Shevtsov et al. (2019), self-adaptation provides a principled way to deal with software systems' uncertainty during operation. A promising approach to handle such dynamics is self-adaptation that can be realized by a MAPE-K feedback loop (Monitor-Analyze-Plan-Execute plus Knowledge) (Iglesia and Weyns, 2015).

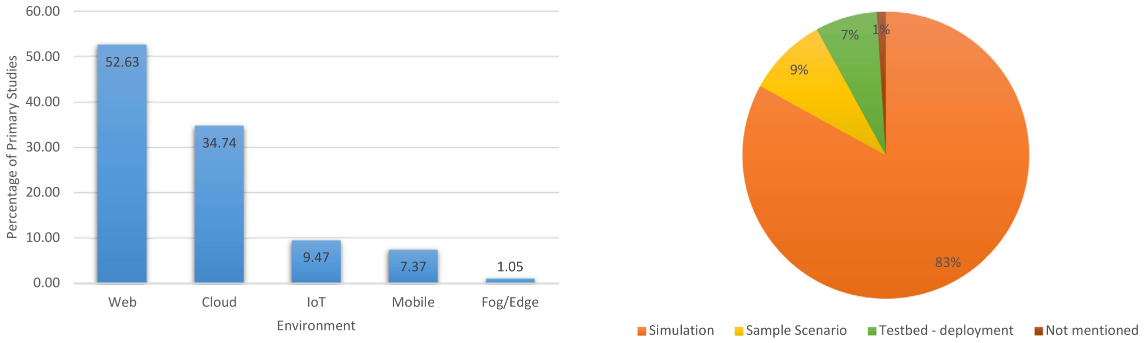
Dynamic adaptation of system parameters and runtime architectural reconfiguration are the general approaches implement adaptation (Shevtsov et al., 2019; Hezavehi et al., 2021). To recover the non-functional aspects of an undertaken composite service, 32% of previous studies claimed that they are working adaptively. The majority of these studies re-plan and compose services when the QoS deviation occurs. Although re-planning is necessary whenever a service is unavailable or unreliable (Wiesemann et al., 2008), the adaptation time can damage the functionality of delay-sensitive applications. Therefore, two directions can be imagined: first, time-sensitive reconfiguration, i.e., reacting to changes at earlier stages, which allows minimizing the interruption time of the execution and expedites the process of finding a feasible recovery (Guidara et al., 2016). Second, proactive adaptability through learning methods can adjust related parameters continuously depending on the QoS changes (Mahfoudh et al., 2018). Therefore, a QoS prediction model that develops an adaptive SC, ensures the completion of composite service in runtime without failure (Liu et al., 2016) and considers a minimum required time is still an open research challenge.

4.4. Environment

RQ4: How the consideration of the uncertainty has evolved as we transit from one environment to another?

In this SLR, we have investigated traditional Web, Cloud, IoT, Mobile, and Fog/Edge environments. The majority of previous studies (52.63%) have been proposed the composition for the traditional web environment. With the advancement of Cloud computing, an unexpected opportunity was provided for deploying services in a more flexible and market-oriented manner; therefore, the development of cloud SC became more interesting for researchers (34.74%). From Fig. 11a, 7.37% of researchers proposed Mobile SC. Variety and movement of mobile devices turn service composition to become extremely sensitive to changes in a communication infrastructure (Efstathiou et al., 2014). If the intelligent device comes along with mobility, its provided service may disappear; thus, automatic and fast service composition is required to overcome these challenges (Liu et al., 2012) in mobile computing environments.

As shown in Fig. 11a, 9.47% of studies have focused on IoT. The functionalities offered by smart objects are usually abstracted as software services (Khanouche et al., 2019). In the IoT paradigm, real-life objects (intelligent objects) can be considered as a source of service (Falas and Stelmach, 2013; D'Angelo et al., 2020). From Fig. 6, services provided by intelligent devices like sensors are influenced by the changes of location, failure in hardware/middleware, poor communication networks, and power limitation (Liu et al., 2012); hence, a composer needs to be able composing services flexibly and assigns services in operation to mitigate composition collapse (Chen et al., 2018b). Finally, because Fog/Edge services have been recently introduced in the literature, the researchers have started proposing service composition in these environments (Velasquez et al., 2017; Wen et al., 2017; Chen et al., 2017b, 2018c). However, currently, there exists only one study on uncertainty-aware Fog/Edge SC (D'Angelo et al., 2020) in selected PSs.



(a) Percentage of adopted environments   (b) Percentage of applied evaluation methods

Fig. 11. Percentage of adopted environments and evaluation methods in primary studies.

Evaluation. From Fig. 11b, the majority of studies used simulation to evaluate their proposed approaches. Furthermore, a few percentages of PSs launched a testbed and deployed a real test environment. Raspberry Pi 3 Model B (Urbieta et al., 2017), Google App Engine (Ivanović et al., 2014), PlanetLab (Zheng et al., 2012), Amazon EC2 and Weka (Karim et al., 2015) are some examples of exploited infrastructures. Furthermore, Mahfoudh et al. (2018) provide a testbed using the Raspberry pi 3, SAPERE middleware equipped with reinforcement learning, Z-wave smart LED light bulb, Multi-sensor Gen 6, and Natural Language Understanding (NLU) system. de Gyvés Avila and Djemame (2013) prepare an experimental environment including 3 computing and server nodes connected by a LAN. Also, a few amounts of studies consider a sample (i.e., small and fixed scenario) to solve and evaluate their proposed method.

4.5. Requirement or assumption

RQ5: What are the requirements/assumptions in different approaches to deal with uncertainty?

We found out that the majority of approaches have constructed their uncertainty-aware QoS model with the assumption that there exists enough QoS historical data. More precisely, they used historical data to calculate the mean and variance of QoS values (Wei et al., 2017), train algorithms (Peng et al., 2017) like BP networks (Zhang, 2010), create PMF or infer probability distributions (Hwang et al., 2007; Wang et al., 2015b; Schuller et al., 2012), find probability of failure (Wang et al., 2012), construct economic models (Ye et al., 2014), and extract fuzzy rules (Luo et al., 2015). From the literature, it is assumed that historical data usually can be originated (Zheng et al., 2016) from service execution logs (Moustafa and Ito, 2018), QoS monitoring mechanisms (Yu and Bouguettaya, 2010; Ramacher and Mönch, 2013), asynchronous monitoring (Chen et al., 2016b), online monitoring subsystems (Calinescu et al., 2010; Sun et al., 2018; Wang et al., 2010; Tan et al., 2014; Tripathy and Tripathy, 2018; D'Angelo et al., 2020; Anisetti et al., 2020), and social network (Hashmi et al., 2016). In some studies, the model has been developed based on expert (decision-maker) opinion (Tripathy and Tripathy, 2018) for the perturbation level (Chen et al., 2016a; Razian et al., 2020a) or confidence index (Falas and Stelmach, 2013). Also, Chen et al. (2018b) presumed that there exists a global semantic matchmaker and Urbieta et al. (2017) considered each resource as an autonomous component.

Furthermore, we discovered the following assumption/requirement: provider's adaptation policy is accessible (Efstathiou et al., 2014; Mezni and Sellami, 2018) and negotiable (Johannes et al., 2015), real-time context information are provided (Njima et al., 2016), service availability values are provided by the supplier (Njima et al., 2016), parameters of the algorithm (Xia et al., 2011) like the number of clusters (Khanouche et al., 2019), exploration and exploitation (Mahfoudh et al., 2018; Khanouche et al., 2019), state transition (Mostafa and Zhang, 2015) internal features of services (Karim et al., 2015) are achievable/tunable/available. Authors in Pino et al. (2017) assume that there exist comprehensive sets of SCO (secure service composition) patterns. Also, the certificate-based studies like (Anisetti et al., 2013) require a certificate authority for validation. Some researchers assumed that statistical description around QoS attributes are available in advance. For example, *"QoS are represented as histograms with the same start point and intervals width"* (Zheng et al., 2011), threshold values of Entropy and Hyper-Entropy (Wang et al., 2011) are determined, availability of mobile device in a time slot is identified (Wang, 2011), probability of staying in a required distance to the service requester (Deng et al., 2016), and Geographical information (Chen et al., 2017a) are achievable. Furthermore, they supposed that the distributions of QoS attributes are known like beta (Wiesemann et al., 2008), normal (Wu et al., 2009; Schuller et al., 2012; Ramacher and Mönch, 2012; Yu et al., 2013b; Lei et al., 2015b), Poisson and exponential (Wu et al., 2009). In addition, decomposition of global constraints (Hwang et al., 2015) to local constraints is presumed. Although, this process speeds up the selection phase, it may lead to an inaccurate QoS modeling. In recommendation systems, Yu et al. (2013a) and Karim et al. (2015) developed their model under the assumption that similar users or services may experience same QoS. Yu et al. (2013a) assumed that *"QoS matrix has a low-rank or approximately low rank structure"*. Many researchers in this category assumed that the user ratings are available (Kuter and Golbeck, 2009; Zheng et al., 2009; Malik and Medjahed, 2010b,a; Yu, 2012; Chen et al., 2013b; Mu et al., 2014; Wang et al., 2018a). However, this assumption may not be imminent in all scenarios.

4.6. Dataset

RQ6: Which datasets are applied to evaluate the performance of proposed methods?

From Fig. 12, we can see that 36 of studies have evaluated their proposed method by using randomly generated QoS values. Some researchers generated random datasets on the basis of the behavior of services on the Internet (Schuller et al., 2014). In addition, random dataset following normal distribution (Wang et al., 2010; Schuller et al., 2012; Ye et al., 2014; Wang et al., 2016; Mostafa and Zhang, 2015), exponential (Wu et al., 2009), uniform (Chen et al., 2013a; Deng et al., 2016), and beta (Wiesemann et al., 2008) distributions have been generated. Table 5 provides the datasets have been used in the literature.

Table 5
The popular datasets used in the literature.

Ref.	Name	Year	Data	Accessibility	Num.
Al-Masri and Mahmoud (2007a,b, 2008)	QWS dataset	2008	Real	Email to author	365
Zheng et al. (2014, 2010c)	WS-DREAM dataset1	2016	Real	Download from website	5825
Zheng et al. (2014)	WS-DREAM dataset2	2016	Real	Download from website	4500
OWLS-TC (2010)	OWLS-TC4	2010	NA	Download from website	1083
Jiang et al. (2012)	I-QoS	2012	Real	NA	825,132

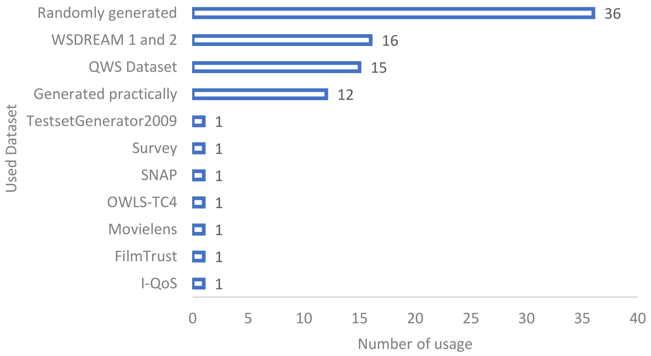


Fig. 12. Percentage of applied datasets.

From our investigations, the WS-DREAM dataset is significantly used in the literature (16 of PSs applied this dataset to assess their method). WS-DREAM datasets maintain two QoS datasets gathered from real Web services. The datasets are publicly released. The first dataset (WS-DREAM dataset1) contains QoS observation of 339 users on 5825 services (Zheng et al., 2014). The second dataset (WS-DREAM dataset2) contains QoS observation of 142 users on 4500 services through 64 consecutive time slices (Zheng et al., 2014). Another popular dataset used in the literature is QWS dataset (15 of PSs) which has been collected by Al-Masri and Mahmoud (2007a, 2008). Researcher also used datasets from other domain such as FilmTrust (Kuter and Golbeck, 2009), Movielens (Rong et al., 2009), and SNAP (Hashmi et al., 2016) in SC. Furthermore, 12 of PSs have generated datasets practically (Zheng et al., 2016). Chen et al. (2018b) used the NS3 simulator to generate their required QoS data. Also, 20 of PSs have not mentioned their used dataset.

5. Research implications and future directions

Based on the results in the previous section, there exist many future research directions that need to be investigated. In this section, as summarized in Fig. 13, we report research challenges that have not been addressed by the research community or still need more investigation.

Emerging environment and infrastructure. In recent years, computing is being transferred to a distributed service delivery model (Buyya et al., 2018). Although cloud providers like Google and AWS tried to decrease service delivery time by distributing their resources all over the world, real-time or delay-sensitive applications like Virtual Reality require less communication delay. Hence, another type of computing was introduced, which is known as Fog/Edge computing (Gogouvitis et al., 2018), to host computational resources in the vicinity of end-users (Varshney and Simmhan, 2019). Additionally, while the solutions like software-defined networking (SDN) and network functions virtualization (NFV) (Bu et al., 2019) make networking architectures more flexible and efficient (Aydeger et al., 2019; Bonfim et al., 2019), they call researchers to investigate the uncertainty factors on quality aspects of services. From Fig. 11a, we can see that the majority of previous studies proposed service composition for traditional web services. However, researchers are expected to focus on emerged service computing paradigms such as Fog (Edge) computing (De Sanctis et al., 2020), where distributed intelligent devices act as both service consumer and provider.

Architecture design. In the traditional service brokerage, a centralized decision maker manages and composes service. However, the lack of global knowledge makes composition error-prune. On the other hand, decentralized service management helps service brokerage to achieve global information and coordination. As an example, GoPrime (Caporuscio et al., 2015) is based on the use of a gossip protocol to collect decentralized information sharing and decision making. We can see that the majority of previous studies proposed service composition in a centralized architecture. Therefore, researchers are expected to focus on decentralized architectures such as Caporuscio et al. (2015), D'Angelo et al. (2020), where distributed nodes can monitor, analyze, plan and execute as well as share knowledge.

Multi-source service composition. From the literature, the majority of PSs (about 84%) assumed that services are provided from a single source. However, using a single provider model poses multiple drawbacks such as the SPF (Single Point of Failure), vendor lock-in, and communication delay. On the other hand, multi-source architecture offers redundancy (mitigate failures), content delivery networks (geographically distributed), more diversity in services, and a more competitive economic model. Although with deploying a multi-cloud scenario, the *five nines* availability could be achievable (Moreno-Vozmediano et al., 2018), other QoS attributes like security would be more affected (i.e., become more uncertain). Also, multi-source architecture potentially brings several challenges: the orchestration of services, global load balancing, cross-cloud private networking (Moreno-Vozmediano et al., 2018), interoperability between the existing cloud and complex maintenance (Ferry et al., 2018). Furthermore, unlike the traditional SC, the majority of the service sources in IoT are scattered devices, which itself causes several factors for uncertainty. The factors like the huge number of candidate services, more volatile and dynamic services and interaction between sources convert SCP as a challenging problem calling for novel and effective approaches. We argue that future service computing environment needs to take multi-source architecture modeling into account, as what has already been deployed in IoT smart cities.

Multi-objective. We observed that only 14% of studies modeled SC as a multi-objective problem. On the other hand, the majority of PSs either ignored multi-objectivity or applied the SAW method and do not care about complex trade-offs between multiple QoS attributes. However, multi-objectivity is required for scenarios involving multiple QoS attributes to be optimized simultaneously where optimal decisions need to be made in the presence of trade-off between conflicting QoS attributes (e.g., maximization of services' reputation while minimizing services' price). Furthermore, multi-objectivity helps to create a more flexible model and possibly better trade-off quality without the need to weights definition (Chen et al., 2018a). Considering the Pareto approach for SC permits the decision-maker to select the desired composition with respect to preferences on the conflicting

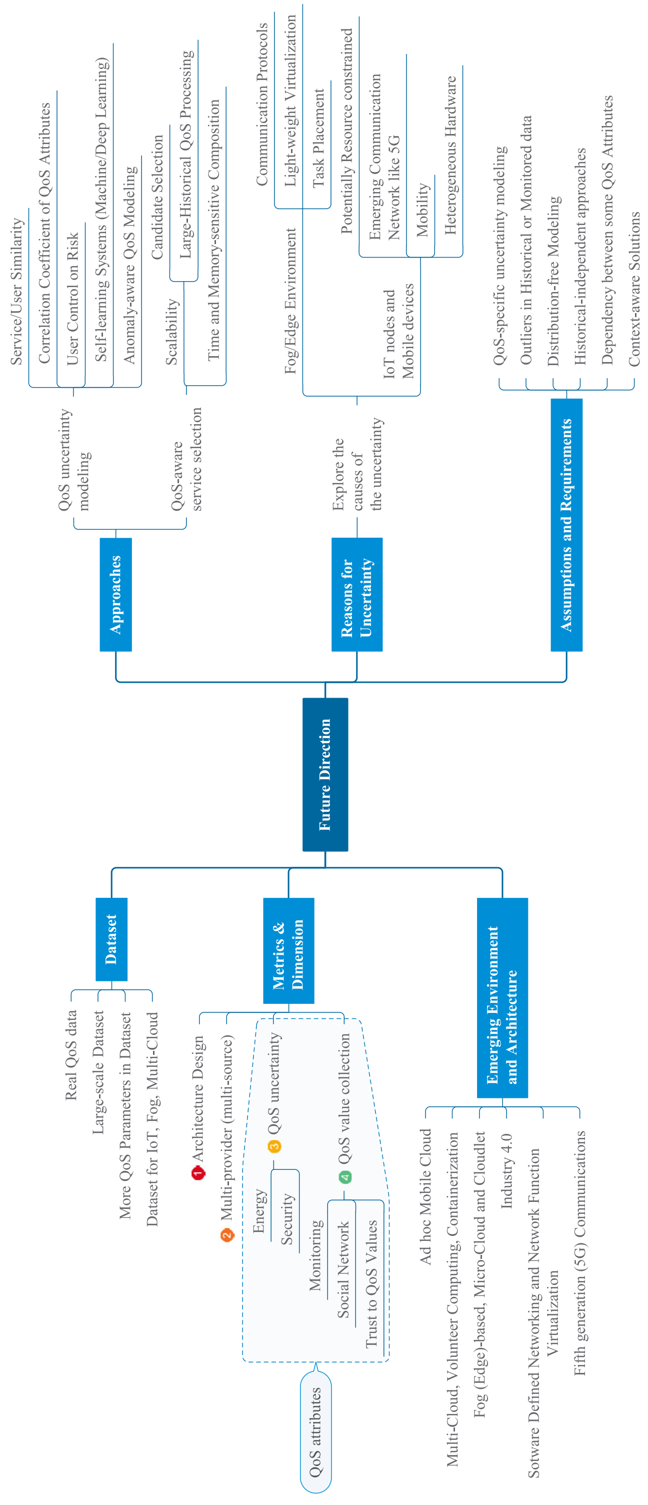

Future Direction
Approaches
QoS uncertainty modeling
Service/User Similarity
Correlation Coefficient of QoS Attributes
User Control on Risk
Self-learning Systems (Machine/Deep Learning)
Anomaly-aware QoS Modeling
QoS-aware service selection
Scalability
Candidate Selection
Large-Historical QoS Processing
Time and Memory-sensitive Composition
Reasons for Uncertainty
Explore the causes of the uncertainty
Fog/Edge Environment
Communication Protocols
Light-weight Virtualization
Task Placement
IoT nodes and Mobile devices
Potentially Resource constrained
Emerging Communication Network like 5G
Mobility
Heterogeneous Hardware
Assumptions and Requirements
QoS-specific uncertainty modeling
Outliers in Historical or Monitored data
Distribution-free Modeling
Historical-independent approaches
Dependency between some QoS Attributes
Context-aware Solutions
Dataset
Real QoS data
Large-scale Dataset
More QoS Parameters in Dataset
Dataset for IoT, Fog, Multi-Cloud
Metrics & Dimension
1 Architecture Design
2 Multi-provider (multi-source)
3 QoS uncertainty
Energy
Security
4 QoS value collection
Monitoring
Social Network
Trust to QoS Values
QoS attributes
Emerging Environment and Architecture
Ad hoc Mobile Cloud
Multi-Cloud, Volunteer Computing, Containerization
Fog (Edge)-based, Micro-Cloud and Cloudlet
Industry 4.0
Sotware Defined Networking and Network Function Virtualization
Fifth generation (5G) Communications

Fig. 13. Research implications and future directions.

QoS attributes. Besides, evolutionary algorithms like NSGA-II (Ramacher and Mönch, 2014; Jatoth et al., 2019) have been widely adopted for solving multi-objective optimization. Researchers are encouraged to consider the trade-off between conflicting objectives using Evolutionary algorithms according to the various QoS attributes and decision spaces (Chugh et al., 2019).

Assumptions and requirements. The assumption of fixed value or well-known probabilistic distribution function (such as normal distribution) has been made in a considerable amount of studies. However, these assumptions do not reflect an accurate estimation of QoS values. Researchers are recommended to relax this assumption (distribution-free approaches) which leads to reliable composition in various environments. Furthermore, the advantage of *historical data*-driven methods is that the models are usually simple to develop. However, they are not always reliable, because they consider more precondition around data such as accessibility, veracity, and consistency of data. Researchers are invited to use a hybrid approach like probabilistic and data-driven approaches, especially for situations where there is not sufficient data. Furthermore, we recommend the design and development of the attribute-specific uncertainty-aware QoS model. This is because the essence of each QoS attributes is different from others. For instance, the response time changes refer to computation and network aspects, while reputation changes are completely dependent on human belief. Some QoS attributes are easier to state as a linguistics variable, which requires specific consideration when applying statistical/mathematical methods. Approaches in Service Recommendation often assume that user-service interaction information is accessible for the composer system. Also, they suppose that similar users or similar services may experience the same QoS (Yu et al., 2013a; Karim et al., 2015). However, these assumptions can happen in specific situations. Therefore, defining more granular parameters for finding similarity still needs more consideration.

Modeling/estimation/calculation. To defeat the weaknesses of constant value representation of QoS attributes, some researchers instead considered a probability distribution for QoS attributes. However, the evaluation of the response time of real services like YouTube (Zheng et al., 2016) shows that it cannot be fitted to standard probabilistic distributions. Although simulation approaches like (Hwang et al., 2015) are independent of the shape of the distribution, simulation methods are time-consuming. As a result, a probabilistic approach with the ability to compose services offered by IoT and mobile devices is still an open challenge. To this aim, the composer system needs to estimate QoS values as well as the accessibility of advertised services. In a mobile environment, in order to make a more accurate QoS estimation, researchers need to model both the provider node mobility and software availability, which would be more challenging (Wang, 2011). From Fig. 14, moving from 2007 to 2020, researchers have started to employ approaches like fuzzy and specifically machine learning. This is because, in the cloud, mobile, and IoT environments, new services are introduced while old ones become obsolete repeatedly along with continuously changing network. In such a situation, a machine learning-based QoS calculation model can adapt to changes. While the number of clouds, mobile and IoT services is continuously increasing, combining learning and optimization for dynamic composition scenarios (Mostafa and Zhang, 2015; Khanouche et al., 2019) is a promising direction for future researches.

Data-driven approaches. Applying data-driven techniques is a promising future direction (Zhang et al., 2019a) for emerging service computing environments like IoT and Fog (Edge), where the number of services is growing, and services are more distributed. However, the method models data from the historical/monitored dataset is not always straight-forward. This is because, in practice, the stored QoS values face the following challenges:

- Missing values: monitoring devices, especially in large scale environments, may fail to collect all service logs. Also, in Service Recommendation class, lack of rating, especially for new services, leads to a sparse user-service matrix.

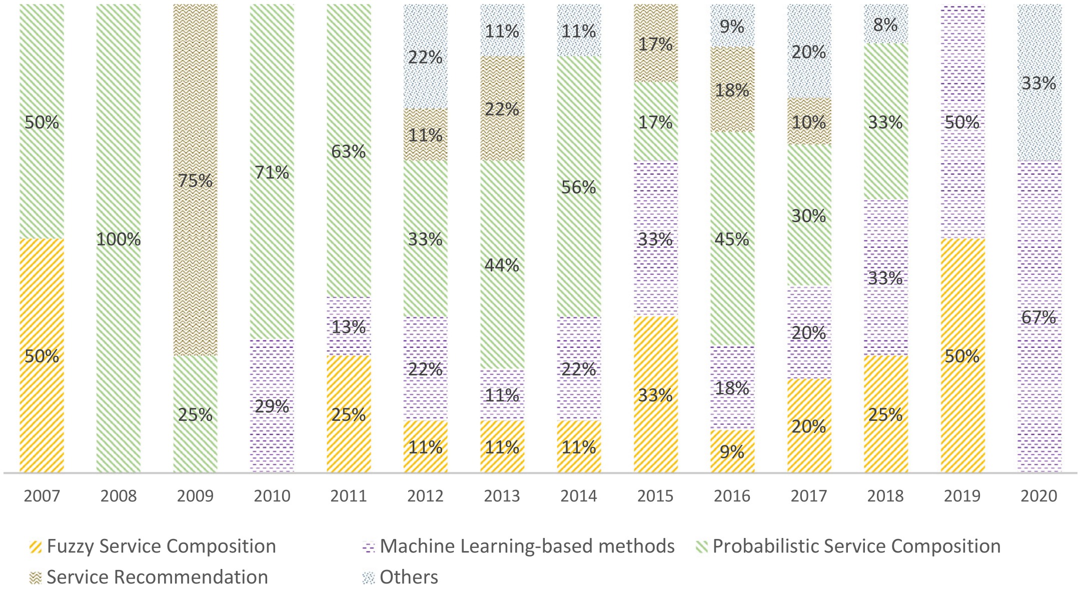


Fig. 14. Proposed approaches by year.

- Data accessibility: in distributed systems, it is not always possible to collect historical QoS values because of export rules and privacy statements.
- Integrity: Some collected data are out of date (Xu et al., 2017) when services are deprecated or upgraded (Chen et al., 2018a)
- Scale: increasing in velocity and volume of meta-data generation in distributed service computing environments like IoT (Huang et al., 2017) seeks scalable methods.
- Diversity: monitored QoS values come from sources with various fields, meaning, structure, and context. Hence, data preparation operations (cleaning and data fusion) seem to be crucial.

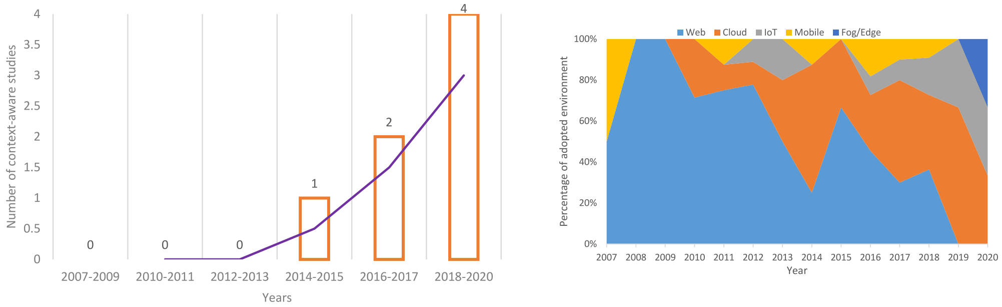
That is to say, researchers need to incorporate these issues into the data-driven QoS-aware service composition under uncertainty in the future work.

Anomaly detection. In Moghaddam et al. (2019), the authors show that there exist anomalies in QoS values of cloud services. Basically, gathering data under different operational situations like high load, system internal errors or crashes, and crowded host or network (Xu et al., 2017) lead to an anomaly in QoS historical values. Therefore, service logs are not suitable for mining directly (without pre-processing) (Kardani-Moghaddam et al., 2019; Razian et al., 2020b). As an example, we can point the network congestion occurred in Beijing, China (Jiang et al., 2012) led to anomalies in logged data. Considering this, one of the important phases of data-driven approaches is removing the anomalies from collected data. We hope that this paper highlights the need for the usage of anomaly detection in QoS modeling.

Service/user similarity. Compare to traditional web and cloud, new/obsolete IoT services may constantly join/leave to/from the network; thus, IoT services similarity calculation would be more complex. Although authors in Chen et al. (2013b) and Karim et al. (2015) proposed user/services similarity, no researches are found in PSs to take IoT or other new service paradigms like Fog into account for similarity calculation. Another guideline that can improve the accuracy of service recommendation approaches is incorporating the trust (Chen et al., 2014; Guo, 2018) into the user/service rating matrix. Furthermore, there are only a few studies that apply service similarity in QoS estimation. In a promising way, service similarity can be used in conjunction with users' similarity. Albeit, feature extraction for finding similarity between services is a challenging task. Additionally, context-aware similarity measures (Wu et al., 2019) can be investigated along with the improvement of well-known similarity measures like PCC (Xue et al., 2019; Lian et al., 2018), Cosine-based, Euclidean distance, or Jaccard similarity (Karim et al., 2015).

Dataset. In Section 4.6, we discussed datasets used in the literature. These datasets only include the QoS values of the traditional web services. While recently, the usage of Cloud services has increased drastically, there is no public and comprehensive QoS dataset for cloud, multi-cloud, mobile, and IoT environments. From Fig. 12, many researchers have used QWS and WSDREAM datasets. However, there exist some shortcomings in these datasets: First, the number of services included in the QWS dataset cannot meet the requirements of large-scale scenarios (Wang et al., 2015c). Second, only two attributes throughput and response time have been considered in the WSDREAM dataset. Third, the datasets do not include the performance of state-of-the-art application software and the execution environment. Therefore, a dataset with more QoS parameters like energy consumption, security, reputation is still a gap in the literature.

QoS Attributes. While the security is a first-class requirement (Stephanow and Khajehmoogahi, 2017), only a few studies have concentrated on the security aspect of service. From Fig. 8, except response time (61 studies from 100), the majority of other QoS attributes have not received much attention. For example, energy consumption, as an important aspect of real-world services (Wang et al., 2018b), has not received much attention in QoS modeling under uncertainty (only two works). This limitation also exist in other parameters like reliability and reputation. Hence, modeling these parameters under uncertainty invokes more effort. In the literature, about 21% of studies have not specified the type of QoS attributes that have been modeled under uncertainty. We argue that applying a common QoS model for all QoS attributes is not possible. For example, the uncertainty-aware method for the reputation of service can be completely different from the uncertainty of response time. Domain and type of variables for declaring attributes (quantitative, linguistic, etc.), data acquisition, and statistical behavior might differ from



(a) Context-aware approaches by year

(b) Discussed environments by year

Fig. 15. Number of context-aware approaches. Percentage of studies in each environment.

one QoS attribute to another. Therefore, another future research theme is attribute-specific QoS modeling under uncertainty for composition. Above all, dynamic QoS modeling is a fundamental aspect of the QoS prediction model, which needs further research to improve the reliability of SC.

Scenario and evaluation. Fig. 10 shows that the majority of motivation scenarios in primary studies have been devoted to the traditional web environment than the cloud and IoT. Therefore, to provide an impressive and inspiring motivation for SC under uncertainty, it could be more advantageous to submit the application of emerging service computing environments like IoT-enabled smart cities (Javadzadeh and Rahmani, 2020) (including Smart Lighting, Smart Building, Smart Energy, Smart Healthcare), Industry 4.0 (Xu et al., 2018c), and upcoming 5G systems (de Almeida et al., 2019). This is because the real deployment of SC relies on the fundamental scenarios, e.g., the number of the service providers, environment conditions and assumptions, source of services, the applications, context, and actual service providers (cloud, IoT). Furthermore, we discovered the lack of real-world implementation and experiments with large-scale and practical applications in many studies. Validating proposed approaches through a real deployment of the multi-cloud, mobile computing, and IoT along as well as conducting more experiments to study the performance of the proposed approaches seek further attempts.

Correlation coefficient of QoS attributes. The ability to incorporate correlation deduced by some of QoS attributes (Hwang et al., 2007) is still an unresolved challenge. In particular, we found no study with an explicit aim for involving the correlation of QoS values in modeling uncertainty. The correlation of QoS attributes defines the relationship between the relative movements of QoS attributes. To achieve this, the composer models uncertainty based on the correlation between QoS attributes to reach a more accurate composition.

User interaction in determining risk. User interaction in determining risk means that users can assign the amount of uncertainty, which is acceptable (Şora and Todinca, 2015) for their application. Considering this, the decision-maker is able to select between an ideal optimal solution and a conservative (near-optimal) solution. It is notable that, when the composer is configured to provide a more robust solution, it should consider a worst-case scenario, which might result in a non-optimal composition. Most of the studies ignored user interaction for determining their acceptable risk in the decision model. Researchers are called to help users further to choose the best composition on the basis of their acceptable uncertainty around the QoS values.

Context-awareness. From Fig. 15a, unlike traditional QoS modeling, researchers have started to incorporate context information of end-users (such as network connection, geographical location, etc.) into their models. Without considering context information, we cannot model QoS attributes accurately. One may say the server can store the context information for each end-user. However, for a distributed service computing environment, storing the context information demands an extra-large amount of memory. Considering this, we suggest a design approach that stores context information in the end device. Whenever the user initiates a request, the state information can be transferred to the service provider. As a result, how to model various contexts while composing services for the associate environment is a considerable challenge which calls the researcher.

Sources of uncertainty. From Fig. 15b, we can see that SC under uncertainty has shifted from simple web services to the cloud and heterogeneous IoT services (Asghari et al., 2018; Mahmud et al., 2018; Rahmani et al., 2018). This poses significant networking and computing uncertainty factors that will affect the QoS attributes. Moreover, uncertainty in Fog/Edge services may happen for a wide variety of reasons. As shown in Fig. 6 (the purple box), communication protocol defect, communication infrastructure disruptions, faults in the operations of the middle-ware, and failure in nodes hosting services are extra sources of uncertainty in Fog/Edge rather than other paradigms. Therefore, it is still an open area to investigate the sources of uncertainty and their impact on each QoS parameter.

Scalability. While the count of the cloud and IoT services is steadily increasing, the design and development of a scalable service composition method is still an open challenge. We observed that 30% of PSs proposed a scalable solution, and others do not consider SC as a large-scale problem. Mathematical approaches can achieve an optimal composition. However, they take more time in a large-scale scenario where the number of tasks within a workflow and/or candidate services grow(s) rapidly. In essence, because SCP is an NP problem, finding an optimal solution with mathematical approaches for large scale problem is computationally not possible. Meanwhile, meta-heuristic algorithms are able to find near-optimal composition in a timely manner (Razian et al., 2020b). Furthermore, some machine learning techniques like Deep Reinforcement Learning are capable of solving large-scale complex optimization models (Moustafa and Ito, 2018). An approach can be called scalable if it targets at least one of the following aspects for QoS modeling or service selection: (1) An extensive number of tasks in a workflow and/or candidate services; (2) An extensive number of QoS attributes for QoS modeling (it has not already been considered in PSs); (3) Big data processing/mining for QoS modeling (it has not already been considered in PSs).

6. Threats to validity

In this section, we discuss threats to the validity of our proposed SLR. The main threats to the validity of this SLR are as follows: threats to studies selection, the threat to data sources, and threats to data extraction and analysis.

6.1. Threats to studies selection

To avoid study and publication bias, we used an automatic search using our developed search string. The search string contains the most probable keywords used in related articles. Because of the lack of flexible search tools in some databases, we had to refine results manually. Although it took a lot of effort and time, it improved the quality of study selection. Due to the fact that there may exist some studies behind the provided search, in addition to common data collection methods used in SLR, additionally, we applied the Snowballing technique to ensure the completeness of study selection. This helped us to discover related studies that are not included in an automatic search.

6.2. Threat to data sources

We have conducted the SLR using several automated searches from the most relevant academic databases to address the research questions. We used seven search databases, including ACM Digital Library, Science Direct, Springer Link, IEEE Xplore Digital Library, Web of Science, Scopus, and Wiley Online Library. We have extracted relevant studies using the proposed search string. After that, the obtained studies were selected according to the inclusion and discarded according to exclusion criteria. An extensive number of sources have been discovered in this SLR, which helps to mitigate the threat to data sources.

6.3. Threats to data extraction and analysis

We analyzed the PSs concerning our research questions, which are primarily about QoS uncertainty in SC. We tried to answer each question in Section 4 and provide corresponding research implications and future directions in Section 5. It may be worthwhile to investigate uncertainty in other phases of the service composition's life cycle, such as business process and workflow structure, to achieve a proficient uncertainty-aware service-oriented architecture not only for uncertainty in QoS values but also uncertainty in entire phases of SC.

7. Summary and conclusions

In this paper, we have reported a Systematic Literature Review (SLR) on service composition under uncertainty. We identified 100 most relevant studies (called as primary studies or PSs) published between the year 2007 and 2020. This SLR provides a taxonomy, comparisons, and analysis of the state-of-art in service composition under uncertainty, covering various distributed paradigms, including cloud, mobile, edge/fog. We identify gaps in current research in order to offer areas for further investigation. Unsurprisingly, the SLR has identified that the most commonly used services for composition were under the classical area of web services research (52.63%). Additionally, 34.74% of studies focused on cloud services, and only a small portion of primary studies considered IoT (9.47%) and Mobile (7.37%) services. We observed that emerging service environments like fog/edge have not yet been highly used for modeling the uncertainty of QoS attributes. This is justified by the fact that IoT- and fog/edge-based services are new technologies that are becoming popular. Concerning adopted approaches, we found that the most widely used approach for solving service composition under uncertainty was probabilistic (40%). Additionally, 23% of studies employed Machine Learning-based methods, 17% Fuzzy Service Composition, and 11% of studies focused on Service Recommendation approaches. We identified that the most commonly considered QoS attributes were the response time (61 of PSs), availability (28 of PSs), reliability (22 of PSs) and throughput (20 of PSs), price (19 of PSs), and reputation (15 of PSs). However, attributes like energy consumption and security are generally under-represented.

We observed that the majority of scalable approaches used (meta-)heuristic algorithm (rather than the mathematical solving methods like Integer Programming). This is justified by the fact that the problem of service composition is NP-hard, and therefore, it requires to be solved in a timely manner. We also observed that there is a lack of real-world/test-bed evaluation and public datasets supporting cloud/IoT based QoS-aware service composition under uncertainty. This research highlights the need for more research in cloud/multi-cloud, mobile/IoT, and emerging fog/edge service composition under uncertainty. More precisely, adaptive, context-aware, and QoS-specific modeling of dynamic and/or heterogeneous distribute services using scalable learning-based and (meta-)heuristic algorithms calls researchers for more investigation. By utilizing this SLR, researchers and practitioners quickly achieve the most related studies that deal with uncertainty in service composition. As future work, we hope the study to inspire and inform researchers into services-aware composition with a new perspective like uncertainty and fuzziness in user preferences, service descriptions, or business workflow.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Al-Masri, E., Mahmoud, Q.H., 2007a. Discovering the best web service. In: International Conference on World Wide Web. ACM, pp. 1257–1258.

Al-Masri, E., Mahmoud, Q.H., 2007b. QoS-based discovery and ranking of web services. In: 2007 16th International Conference on Computer Communications and Networks. IEEE, pp. 529–534.

Al-Masri, E., Mahmoud, Q.H., 2008. Investigating web services on the world wide web. In: International Conference on World Wide Web. ACM, pp. 795–804.

Alrifai, M., Skoutas, D., Risse, T., 2010. Selecting skyline services for QoS-based web service composition. In: Proceedings of the 19th International Conference on World Wide Web. ACM, pp. 11–20.

Amiri, B., Fathian, M., Maroosi, A., 2009. Application of shuffled frog-leaping algorithm on clustering. Int. J. Adv. Manuf. Technol. 45 (1–2), 199–209.

Anisetti, M., Ardagna, C.A., Damiani, E., 2013. Security certification of composite services: A test-based approach. In: 2013 IEEE 20th International Conference on Web Services. IEEE, pp. 475–482.

Anisetti, M., Ardagna, C.A., Damiani, E., Gaudenzi, F., Jeon, G., 2020. Cost-effective deployment of certified cloud composite services. J. Parallel Distrib. Comput. 135, 203–218.

Anisetti, M., Ardagna, C., Damiani, E., Polegri, G., 2019. Test-based security certification of composite services. ACM Trans. Web (TWEB) 13 (1), 3.

Asghari, P., Rahmani, A.M., Javadi, H.H.S., 2018. Service composition approaches in IoT: A systematic review. J. Netw. Comput. Appl. 120, 61–77.

Aydeger, A., Saputro, N., Akkaya, K., 2019. A moving target defense and network forensics framework for ISP networks using SDN and NFV. Future Gener. Comput. Syst. 94, 496–509.

Barbon, F., Traverso, P., Pistore, M., Trainotti, M., 2006. Run-time monitoring of instances and classes of Web service compositions. In: International Conference on Web Services (ICWS'06). IEEE, pp. 63–71.

Barnes, B.J., Rountree, B., Lowenthal, D.K., Reeves, J., De Supinski, B., Schulz, M., 2008. A regression-based approach to scalability prediction. In: Annual International Conference on Supercomputing. ACM, pp. 368–377.

Bass, L., Weber, I., Zhu, L., 2015. DevOps: A Software Architect's Perspective. Addison-Wesley Professional.

Baudrit, C., Dubois, D., Guyonnet, D., 2006. Joint propagation and exploitation of probabilistic and possibilistic information in risk assessment. IEEE Trans. Fuzzy Syst. 14 (5), 593–608.

Behzadian, M., Kazemzadeh, R.B., Albadvi, A., Aghdasi, M., 2010. PROMETHEE: A comprehensive literature review on methodologies and applications. European J. Oper. Res. 200 (1), 198–215.

Bertsimas, D., Sim, M., 2004. The price of robustness. Oper. Res. 52 (1), 35–53.

Bonfim, M.S., Dias, K.L., Fernandes, S.F., 2019. Integrated NFV/SDN architectures: A systematic literature review. ACM Comput. Surv. 51 (6), 114.

Borzsony, S., Kossmann, D., Stocker, K., 2001. The skyline operator. In: Proceedings 17th International Conference on Data Engineering. IEEE, pp. 421–430.

Brereton, P., Kitchenham, B.A., Budgen, D., Turner, M., Khalil, M., 2007. Lessons from applying the systematic literature review process within the software engineering domain. J. Syst. Softw. 80 (4), 571–583.

Bu, C., Wang, X., Cheng, H., Huang, M., Li, K., 2019. Routing as a service (RaaS): An open framework for customizing routing services. J. Netw. Comput. Appl. 125, 130–145.

Buyya, R., Broberg, J., Goscinski, A.M., 2010. Cloud Computing: Principles and Paradigms, Vol. 87. John Wiley & Sons.

Buyya, R., Srirama, S.N., Casale, G., Calheiros, R., Simmhan, Y., Varghese, B., Gelenbe, E., Javadi, B., Vaquero, L.M., Netto, M.A., et al., 2018. A manifesto for future generation cloud computing: Research directions for the next decade. ACM Comput. Surv. 51 (5), 105.

Calinescu, R., Grunske, L., Kwiatkowska, M., Mirandola, R., Tamburrelli, G., 2010. Dynamic QoS management and optimization in service-based systems. IEEE Trans. Softw. Eng. 37 (3), 387–409.

Caporuscio, M., Grassi, V., Marzolla, M., Mirandola, R., 2015. GoPrime: A fully decentralized middleware for utility-aware service assembly. IEEE Trans. Softw. Eng. 42 (2), 136–152.

Cardellini, V., Casalicchio, E., Grassi, V., Iannucci, S., Presti, F.L., Mirandola, R., 2011. Moses: A framework for QoS driven runtime adaptation of service-oriented systems. IEEE Trans. Softw. Eng. 38 (5), 1138–1159.

Chen, T., Bahsoon, R., Yao, X., 2018a. A survey and taxonomy of self-aware and self-adaptive cloud autoscaling systems. ACM Comput. Surv. 51 (3), 61.

Chen, N., Cardozo, N., Clarke, S., 2018b. Goal-driven service composition in mobile and pervasive computing. IEEE Trans. Serv. Comput. 11 (1), 49–62.

Chen, R., Guo, J., Bao, F., 2014. Trust management for service composition in SoA-based IoT systems. In: IEEE Wireless Communications and Networking Conference. IEEE, pp. 3444–3449.

Chen, Y., Jiang, L., Zhang, J., Dong, X., 2016a. A robust service selection method based on uncertain QoS. Math. Probl. Eng. 2016.

Chen, Z., Shen, L., Li, F., You, D., 2017a. Your neighbors alleviate cold-start: On geographical neighborhood influence to collaborative web service QoS prediction. Knowl.-Based Syst. 138, 188–201.

Chen, M., Tan, T.H., Sun, J., Wang, J., Liu, Y., Sun, J., Dong, J.S., 2016b. Service adaptation with probabilistic partial models. In: International Conference on Formal Engineering Methods. Springer, pp. 122–140.

Chen, N., Yang, Y., Li, J., Zhang, T., 2017b. A fog-based service enablement architecture for cross-domain IoT applications. In: 2017 IEEE Fog World Congress (FWC). IEEE, pp. 1–6.

Chen, N., Yang, Y., Zhang, T., Zhou, M.-T., Luo, X., Zao, J.K., 2018c. Fog as a service technology. IEEE Commun. Mag. 56 (11), 95–101.

Chen, Y., Ying, S., Zhang, L., Wu, J., 2013a. Exception detection for web service composition using improved Bayesian network. J. Digit. Inf. Manage. 11 (2), 109.

Chen, X., Zheng, Z., Liu, X., Huang, Z., Sun, H., 2013b. Personalized QoS-aware web service recommendation and visualization. IEEE Trans. Serv. Comput. 6 (1), 35–47.

Chugh, T., Sindhya, K., Hakanen, J., Miettinen, K., 2019. A survey on handling computationally expensive multiobjective optimization problems with evolutionary algorithms. Soft Comput. 23 (9), 3137–3166.

Ciszkowski, T., Mazurczyk, W., Kotulski, Z., Hossfeld, T., Fiedler, M., Collange, D., 2012. Towards quality of experience-based reputation models for future web service provisioning. Telecommun. Syst. 51 (4), 283–295.

D'Angelo, M., Caporuscio, M., Grassi, V., Mirandola, R., 2020. Decentralized learning for self-adaptive QoS-aware service assembly. Future Gener. Comput. Syst. 108, 210–227.

Dastjerdi, A.V., Buyya, R., 2014. Compatibility-aware cloud service composition under fuzzy preferences of users. IEEE Trans. Cloud Comput. 2 (1), 1–13.

de Almeida, I.B., Mendes, L.L., Rodrigues, J.J., da Cruz, M.A., 2019. 5G waveforms for IoT applications. IEEE Commun. Surv. Tutor..

de Gyvés Avila, S., Djemame, K., 2013. Fuzzy logic based QoS optimization mechanism for service composition. In: International Symposium on Service-Oriented System Engineering. IEEE, pp. 182–191.

De Sanctis, M., Muccini, H., Vaidhyanathan, K., 2020. Data-driven adaptation in microservice-based IoT architectures. In: 2020 IEEE International Conference on Software Architecture Companion (ICSA-C). IEEE, pp. 59–62.

Deng, S., Huang, L., Li, Y., Zhou, H., Wu, Z., Cao, X., Kataev, M.Y., Li, L., 2016. Toward risk reduction for mobile service composition. IEEE Trans. Cybern. 46 (8), 1807–1816.

Efstathiou, D., McBurney, P., Zschaler, S., Bourcier, J., 2014. Efficient multi-objective optimisation of service compositions in mobile ad hoc networks using lightweight surrogate models. J. UCS 20 (8), 1089–1108.

Elhabbash, A., Bahsoon, R., Tino, P., 2017. Self-awareness for dynamic knowledge management in self-adaptive volunteer services. In: International Conference on Web Services. IEEE, pp. 180–187.

Esfahani, N., Malek, S., 2013. Uncertainty in self-adaptive software systems. In: Software Engineering for Self-Adaptive Systems II. Springer, pp. 214–238.

Falas, Ł., Stelmach, P., 2013. Web service composition with uncertain non-functional parameters. In: Doctoral Conference on Computing, Electrical and Industrial Systems. Springer, pp. 45–52.

Fathian, M., Amiri, B., Maroosi, A., 2007. Application of honey-bee mating optimization algorithm on clustering. Appl. Math. Comput. 190 (2), 1502–1513.

Ferry, N., Chauvel, F., Song, H., Rossini, A., Lushpenko, M., Solberg, A., 2018. Cloudmf: Model-driven management of multi-cloud applications. ACM Trans. Internet Technol. (TOIT) 18 (2), 16.

Gai, K., Qiu, M., Zhao, H., 2018. Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing. J. Parallel Distrib. Comput. 111, 126–135.

Garlan, D., 2010. Software engineering in an uncertain world. In: Proceedings of the FSE/SDP Workshop on Future of Software Engineering Research. pp. 125–128.

Ghazanfari, M., Alizadeh, S., Fathian, M., Koulouriotis, D.E., 2007. Comparing simulated annealing and genetic algorithm in learning FCM. Appl. Math. Comput. 192 (1), 56–68.

Gogouvitis, S.V., Mueller, H., Premnadh, S., Seitz, A., Bruegge, B., 2018. Seamless computing in industrial systems using container orchestration. Future Gener. Comput. Syst..

Golbeck, J., 2006. Generating predictive movie recommendations from trust in social networks. In: International Conference on Trust Management. Springer, pp. 93–104.

Gong, Y., Huang, L., Han, K., 2014. Service dynamic substitution approach based on cloud model. In: International Conference on Advanced Data and Information Engineering (DaEng-2013). Springer, pp. 563–570.

Guidara, I., Al Jaouhari, I., Guermouche, N., 2016. Dynamic selection for service composition based on temporal and QoS constraints. In: 2016 IEEE International Conference on Services Computing (SCC). IEEE, pp. 267–274.

Guo, J., 2018. Trust-based Service Management of Internet of Things Systems and Its Applications (Ph.D. thesis). Virginia Tech.

Guo, Y., Wang, S., Wong, K.-S., Kim, M.H., 2017. Skyline service selection approach based on QoS prediction. Int. J. Web Grid Serv. 13 (4), 425–447.

Guoping, Z., Longlong, Q., Ningbo, W., 2012. Technology of QoS evaluation based grey system theory. In: International Conference on Computer Science and Network Technology. IEEE, pp. 1934–1937.

Hashmi, K., Malik, Z., Najmi, E., Rezgui, A., 2016. SNRNeg: A social network enabled negotiation service. Inform. Sci. 349, 248–262.

Hezavehi, S.M., Weyns, D., Avgeriou, P., Calinescu, R., Mirandola, R., Perez-Palacin, D., 2021. Uncertainty in self-adaptive systems: A research community perspective. arXiv preprint arXiv:2103.02717.

Hofstede, G., 2011. Dimensionalizing cultures: The Hofstede model in context. Online Read. Psychol. Cult. 2 (1), 8.

Huang, C., Cai, H., Li, Y., Du, J., Bu, F., Jiang, L., 2017. A process mining based service composition approach for mobile information systems. Mob. Inf. Syst. 2017.

Hwang, S.-Y., Hsu, C.-C., Lee, C.-H., 2015. Service selection for web services with probabilistic QoS. IEEE Trans. Serv. Comput. 8 (3), 467–480.

Hwang, S.-Y., Wang, H., Tang, J., Srivastava, J., 2007. A probabilistic approach to modeling and estimating the QoS of web-services-based workflows. Inform. Sci. 177 (23), 5484–5503.

Iglesia, D.G.D.L., Weyns, D., 2015. MAPE-K formal templates to rigorously design behaviors for self-adaptive systems. ACM Trans. Auton. Adapt. Syst. (TAAS) 10 (3), 1–31.

Ivanović, D., Carro, M., Kaowichakorn, P., 2014. Towards QoS prediction based on composition structure analysis and probabilistic models. In: International Conference on Service-Oriented Computing. Springer, pp. 394–402.

Jang, J.-S.R., Sun, C.-T., Mizutani, E., 1997. Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [Book Review]. IEEE Trans. Automat. Control 42 (10), 1482–1484.

Jatoth, C., Gangadharan, G., Buyya, R., 2019. Optimal fitness aware cloud service composition using an adaptive genotypes evolution based genetic algorithm. Future Gener. Comput. Syst. 94, 185–198.

Jatoth, C., Gangadharan, G., Fiore, U., Buyya, R., 2018. QoS-aware Big service composition using MapReduce based evolutionary algorithm with guided mutation. Future Gener. Comput. Syst. 86, 1008–1018.

Javadzadeh, G., Rahmani, A.M., 2020. Fog computing applications in smart cities: A systematic survey. Wirel. Netw. 26 (2), 1433–1457.

Jian, X., Zhu, Q., Xia, Y., 2016. An interval-based fuzzy ranking approach for QoS uncertainty-aware service composition. Optik 127 (4), 2102–2110.

Jiang, W., Lee, D., Hu, S., 2012. Large-scale longitudinal analysis of SOAO-based and RESTful web services. In: International Conference on Web Services. IEEE, pp. 218–225.

Jiang, Y., Liu, J., Tang, M., Liu, X., 2011. An effective web service recommendation method based on personalized collaborative filtering. In: International Conference on Web Services. IEEE, pp. 211–218.

Johannes, A., Nanda, P., He, X., 2015. Resource utilization based dynamic pricing approach on cloud computing application. In: International Conference on Algorithms and Architectures for Parallel Processing. Springer, pp. 669–677.

Jurca, R., Faltings, B., Binder, W., 2007. Reliable QoS monitoring based on client feedback. In: International Conference on World Wide Web. ACM, pp. 1003–1012.

Kardani-Moghaddam, S., Buyya, R., Ramamohanarao, K., 2019. Performance anomaly detection using isolation-trees in heterogeneous workloads of web applications in computing clouds. Concurr. Comput.: Pract. Exper. e5306.

Karim, R., Ding, C., Miri, A., 2015. End-to-end QoS prediction of vertical service composition in the cloud. In: International Conference on Cloud Computing. IEEE, pp. 229–236.

Kazem, A.A.P., Pedram, H., Abolhassani, H., 2015. BNQM: a Bayesian network based QoS model for grid service composition. Expert Syst. Appl. 42 (20), 6828–6843.

Khanouche, M.E., Attal, F., Amirat, Y., Chibani, A., Kerkar, M., 2019. Clustering-based and QoS-aware services composition algorithm for ambient intelligence. Inform. Sci. 482, 419–439.

Kil, H., Cha, R., Nam, W., 2016. Transaction history-based web service composition for uncertain QoS. Int. J. Web Grid Serv. 12 (1), 42–62.

Kitchenham, B., 2004. Procedures for Performing Systematic Reviews, Vol. 33. Keele University, Keele, UK, pp. 1–26, (2004).

Kolodner, J., 2014. Case-Based Reasoning. Morgan Kaufmann.

Kumar, S., Bahsoon, R., Chen, T., Li, K., Buyya, R., 2018. Multi-tenant cloud service composition using evolutionary optimization.

Kuter, U., Golbeck, J., 2009. Semantic web service composition in social environments. In: International Semantic Web Conference. Springer, pp. 344–358.

Kwiatkowska, M., Norman, G., Parker, D., 2011. PRISM 4.0: Verification of probabilistic real-time systems. In: International Conference on Computer-Aided Verification. Springer, pp. 585–591.

Lei, Y., Jiantao, Z., Fengqi, W., Yongqiang, G., Bo, Y., 2015a. Web service composition based on reinforcement learning. In: International Conference on Web Services. IEEE, pp. 731–734.

Lei, Y., Jiantao, Z., Yongqiang, G., Jing, L., Xuebin, M., 2015b. Dynamic web service composition based on state space searching. In: International Conference on Parallel and Distributed Systems (ICPADS). IEEE, pp. 821–826.

Lei, Y., Zhili, W., Luoming, M., Xuesong, Q., Jiantao, Z., 2014. Learning-based web service composition in uncertain environments. J. Web Eng. 13 (5&6), 450–468.

Li, D., Cheung, D., Shi, X., Ng, V., 1998. Uncertainty reasoning based on cloud models in controllers. Comput. Math. Appl. 35 (3), 99–123.

Li, L., Jin, Z., Li, G., Zheng, L., Wei, Q., 2012. Modeling and analyzing the reliability and cost of service composition in the IoT: A probabilistic approach. In: International Conference on Web Services. IEEE, pp. 584–591.

Li, G.-S., Wang, N., 2015. Web service QoS prediction with adaptive calibration. In: International Conference on Computer Science and Applications (CSA). IEEE, pp. 351–356.

Lian, D., Zheng, K., Ge, Y., Cao, L., Chen, E., Xie, X., 2018. GeoMF++: Scalable location recommendation via joint geographical modeling and matrix factorization. ACM Trans. Inf. Syst. (TOIS) 36 (3), 33.

Liu, Z.-Z., Chu, D.-H., Jia, Z.-P., Shen, J.-Q., Wang, L., 2016. Two-stage approach for reliable dynamic Web service composition. Knowl.-Based Syst. 97, 123–143.

Liu, L., Liu, X., Li, X., 2012. Cloud-based service composition architecture for internet of things. In: Internet of Things. Springer, pp. 559–564.

Liu, F.T., Ting, K.M., Zhou, Z.-H., 2008. Isolation forest. In: 2008 Eighth IEEE International Conference on Data Mining. IEEE, pp. 413–422.

Liu, Y., Yang, C., Jiang, L., Xie, S., Zhang, Y., 2019. Intelligent edge computing for IoT-based energy management in smart cities. IEEE Netw. 33 (2), 111–117.

Luo, X., Lv, Y., Li, R., Chen, Y., 2015. Web service QoS prediction based on adaptive dynamic programming using fuzzy neural networks for cloud services. IEEE Access 3, 2260–2269.

Mahdavi-Hezavehi, S., Avgeriou, P., Weyns, D., 2017. A classification framework of uncertainty in architecture-based self-adaptive systems with multiple quality requirements. In: Managing Trade-Offs in Adaptable Software Architectures. Elsevier, pp. 45–77.

Mahfoudh, H.B., Serugendo, G.D.M., Boulmier, A., Abdennadher, N., 2018. Coordination model with reinforcement learning for ensuring reliable on-demand services in collective adaptive systems. In: International Symposium on Leveraging Applications of Formal Methods. Springer, pp. 257–273.

Mahmud, R., Kotagiri, R., Buyya, R., 2018. Fog computing: A taxonomy, survey and future directions. In: Internet of Everything. Springer, pp. 103–130.

Malik, Z., Medjahed, B., 2010a. Maintaining trustworthiness of service compositions. In: International Conference on Frontiers of Information Technology. ACM, p. 23.

Malik, Z., Medjahed, B., 2010b. Trust assessment for web services under uncertainty. In: International Conference on Service-Oriented Computing. Springer, pp. 471–485.

Menasce, D., Gomaa, H., Sousa, J., et al., 2011. Sassy: A framework for self-architecting service-oriented systems. IEEE Softw. 28 (6), 78–85.

Mezni, H., Sellami, M., 2018. A negotiation-based service selection approach using swarm intelligence and kernel density estimation. Softw. - Pract. Exp. 48 (6), 1285–1311.

Moghaddam, S.K., Buyya, R., Ramamohanarao, K., 2019. ACAS: An anomaly-based cause aware auto-scaling framework for clouds. J. Parallel Distrib. Comput. 126, 107–120.

Morabito, R., Cozzolino, V., Ding, A.Y., Beijar, N., Ott, J., 2018. Consolidate IoT edge computing with lightweight virtualization. IEEE Netw. 32 (1), 102–111.

Moreno-Vozmediano, R., Montero, R.S., Huedo, E., Llorente, I.M., 2018. Orchestrating the deployment of high availability services on multi-zone and multi-cloud scenarios. J. Grid Comput. 16 (1), 39–53.

Mostafa, A., Zhang, M., 2015. Multi-objective service composition in uncertain environments. IEEE Trans. Serv. Comput..

Moustafa, A., Ito, T., 2018. A deep reinforcement learning approach for large-scale service composition. In: International Conference on Principles and Practice of Multi-Agent Systems. Springer, pp. 296–311.

Moustafa, A., Zhang, M., 2012. Towards proactive web service adaptation. In: International Conference on Advanced Information Systems Engineering. Springer, pp. 473–485.

Mu, B., Li, S., Yuan, S., 2014. QoS-aware cloud service selection based on uncertain user preference. In: International Conference on Rough Sets and Knowledge Technology. Springer, pp. 589–600.

Niu, S., Zou, G., Gan, Y., Xiang, Y., Zhang, B., 2019. Towards the optimality of QoS-aware web service composition with uncertainty. Int. J. Web Grid Serv. 15 (1), 1–28.

Njima, C.B., Gamha, Y., Romdhane, L.B., 2016. A probabilistic model for web service composition in uncertain mobile contexts. In: International Conference of Computer Systems and Applications (AICCSA). IEEE, pp. 1–7.

OWLS-TC, 2010. OWLS-TC.

Peng, S., Wang, H., Yu, Q., 2017. Estimation of distribution with restricted Boltzmann machine for adaptive service composition. In: 2017 IEEE International Conference on Web Services. IEEE, pp. 114–121.

Pernici, B., Siadat, S.H., 2011. Selection of service adaptation strategies based on fuzzy logic. In: IEEE World Congress on Services. IEEE, pp. 99–106.

Pham, T.-M., Fdida, S., Chu, H.-N., et al., 2020. Modeling and analysis of robust service composition for network functions virtualization. Comput. Netw. 166, 106989.

Pino, L., Spanoudakis, G., Krotsiani, M., Mahbub, K., 2017. Pattern based design and verification of secure service compositions. IEEE Trans. Serv. Comput..

Prochart, G., Weiss, R., Schmid, R., Kaefer, G., 2007. Fuzzy-based support for service composition in mobile ad-hoc networks. In: International Conference on Pervasive Services. IEEE, pp. 379–384.

Rahmani, A.M., Gia, T.N., Negash, B., Anzanpour, A., Azimi, I., Jiang, M., Liljeberg, P., 2018. Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: A fog computing approach. Future Gener. Comput. Syst. 78, 641–658.

Ramacher, R., Mönch, L., 2012. Dynamic service selection with end-to-end constrained uncertain QoS attributes. In: International Conference on Service-Oriented Computing. Springer, pp. 237–251.

Ramacher, R., Mönch, L., 2013. Reliable service reconfiguration for time-critical service compositions. In: International Conference on Services Computing. IEEE, pp. 184–191.

Ramacher, R., Mönch, L., 2014. Robust multi-criteria service composition in information systems. Bus. Inf. Syst. Eng. 6 (3), 141–151.

Razian, M., Fathian, M., Buyya, R., 2020a. ARC: Anomaly-aware Robust Cloud-integrated IoT service composition based on uncertainty in advertised quality of service values. J. Syst. Softw. 164, 110557.

Razian, M., Fathian, M., Wu, H., Akbari, A., Buyya, R., 2020b. SAIoT: Scalable anomaly-aware services composition in CloudIoT environments. IEEE Internet Things J..

Roca, S., Sancho, J., García, J., Alesanco, A., 2019. Microservice chatbot architecture for chronic patient support. J. Biomed. Inform. 103305.

Rodriguez-Mier, P., Pedrinaci, C., Lama, M., Mucientes, M., 2015. An integrated semantic web service discovery and composition framework. IEEE Trans. Serv. Comput. 9 (4), 537–550.

Rong, W., Liu, K., Liang, L., 2009. Personalized web service ranking via user group combining association rule. In: International Conference on Web Services. IEEE, pp. 445–452.

Rosario, S., Benveniste, A., Haar, S., Jard, C., 2008. Probabilistic QoS and soft contracts for transaction-based web services orchestrations. IEEE Trans. Serv. Comput. 1 (4), 187–200.

Santos, E.A., McLean, C., Solinas, C., Hindle, A., 2018. How does Docker affect energy consumption? Evaluating workloads in and out of Docker containers. J. Syst. Softw. 146, 14–25.

Schuller, D., Lampe, U., Eckert, J., Steinmetz, R., Schulte, S., 2012. Cost-driven optimization of complex service-based workflows for stochastic QoS parameters. In: International Conference on Web Services. IEEE, pp. 66–73.

Schuller, D., Siebenhaar, M., Hans, R., Wenge, O., Steinmetz, R., Schulte, S., 2014. Towards heuristic optimization of complex service-based workflows for stochastic QoS attributes. In: International Conference on Web Services. IEEE, pp. 361–368.

Sharma, Y., Si, W., Sun, D., Javadi, B., 2019. Failure-aware energy-efficient VM consolidation in cloud computing systems. Future Gener. Comput. Syst. 94, 620–633.

Shevtsov, S., Weyns, D., Maggio, M., 2019. SimCA* A control-theoretic approach to handle uncertainty in self-adaptive systems with guarantees. ACM Trans. Auton. Adapt. Syst. (TAAS) 13 (4), 1–34.

Şora, I., Todinca, D., 2015. Dealing with fuzzy QoS properties in service composition. In: International Symposium on Applied Computational Intelligence and Informatics. IEEE, pp. 197–202.

Stephanow, P., Khajehmoogahi, K., 2017. Towards continuous security certification of software-as-a-service applications using web application testing techniques. In: 2017 IEEE 31st International Conference on Advanced Information Networking and Applications (AINA). IEEE, pp. 931–938.

Sugeno, M., 1985. Industrial Applications of Fuzzy Control. Elsevier Science Inc..

Sun, X., Chen, J., Xia, Y., He, Q., Wang, Y., Luo, X., Zhang, R., Han, W., Wu, Q., 2018. A fluctuation-aware approach for predictive web service composition. In: International Conference on Services Computing (SCC). IEEE, pp. 121–128.

Sun, Q., Wang, S., Zou, H., Yang, F., 2013. Fast web service selection for reliable service composition application system. Information 16 (3), 2001.

Sun, M., Zhou, Z., Wang, J., Du, C., Gaaloul, W., 2019. Energy-efficient IoT service composition for concurrent timed applications. Future Gener. Comput. Syst. 100, 1017–1030.

Tabassum, H., Salehi, M., Hossain, E., 2019. Fundamentals of mobility-aware performance characterization of cellular networks: A tutorial. IEEE Commun. Surv. Tutor..

Tafsiri, S.A., Yousefi, S., 2018. Combinatorial double auction-based resource allocation mechanism in cloud computing market. J. Syst. Softw. 137, 322–334.

Tan, T.H., Chen, M., André, E., Sun, J., Liu, Y., Dong, J.S., 2014. Automated runtime recovery for QoS-based service composition. In: International Conference on World Wide Web. ACM, pp. 563–574.

Toosi, A.N., Calheiros, R.N., Thulasiram, R.K., Buyya, R., 2011. Resource provisioning policies to increase iaas provider's profit in a federated cloud environment. In: 2011 IEEE International Conference on High Performance Computing and Communications. IEEE, pp. 279–287.

Torra, V., 2002. A review of the construction of hierarchical fuzzy systems. Int. J. Intell. Syst. 17 (5), 531–543.

Tripathy, A.K., Patra, M.R., 2011. Service based system monitoring framework. Int. J. Comput. Inf. Syst. Ind. Manage. Appl.: IJCISIM 3, 924–931.

Tripathy, A.K., Tripathy, P.K., 2018. Fuzzy QoS requirement-aware dynamic service discovery and adaptation. Appl. Soft Comput. 68, 136–146.

Urbieta, A., González-Beltrán, A., Mokhtar, S.B., Hossain, M.A., Capra, L., 2017. Adaptive and context-aware service composition for IoT-based smart cities. Future Gener. Comput. Syst. 76, 262–274.

Varshney, P., Simmhan, Y., 2019. Characterizing application scheduling on edge, fog, and cloud computing resources. Softw. - Pract. Exp..

Veeresh, P., Sam, R.P., Bindu, C.S., 2017. Fuzzy based optimal QoS constraint services composition in mobile ad hoc networks. Int. J. Commun. Netw. Inf. Secur. (IJCNIS) 9 (3), 491–499.

Velasquez, K., Abreu, D.P., Gonçalves, D., Bittencourt, L., Curado, M., Monteiro, E., Madeira, E., 2017. Service orchestration in fog environments. In: 2017 IEEE 5th International Conference on Future Internet of Things and Cloud (FiCloud). IEEE, pp. 329–336.

Wang, L.-X., 1998. Universal approximation by hierarchical fuzzy systems. Fuzzy Sets and Systems 93 (2), 223–230.

Wang, J., 2011. Exploiting mobility prediction for dependable service composition in wireless mobile ad hoc networks. IEEE Trans. Serv. Comput. 4 (1), 44–55.

Wang, P., Ding, Z., Jiang, C., Zhou, M., Zheng, Y., 2015a. Automatic web service composition based on uncertainty execution effects. IEEE Trans. Serv. Comput. 9 (4), 551–565.

Wang, X., Fu, X., Liu, L., Huang, Q., Yue, K., 2015b. A probabilistic approach to analyzing the stochastic QoS of web service composition. In: Web Information System and Application Conference (WISA). IEEE, pp. 147–150.

Wang, S., Guo, Y., Li, Y., Hsu, C.-H., 2018a. Cultural distance for service composition in cyber–physical–social systems. Future Gener. Comput. Syst..

Wang, S., Huang, L., Sun, L., Hsu, C.-H., Yang, F., 2017. Efficient and reliable service selection for heterogeneous distributed software systems. Future Gener. Comput. Syst. 74, 158–167.

Wang, C., Ma, H., Chen, G., Hartmann, S., Branke, J., 2020. Robustness estimation and optimisation for semantic web service composition with stochastic service failures. IEEE Trans. Emerg. Top. Comput. Intell..

Wang, X., Wang, Z., Xu, X., 2012. Analytic profit optimization of service-based systems. In: International Conference on Web Services. IEEE, pp. 359–367.

Wang, H., Wu, Q., Chen, X., Yu, Q., 2015c. Integrating gaussian process with reinforcement learning for adaptive service composition. In: International Conference on Service-Oriented Computing. Springer, pp. 203–217.

Wang, H., Zhang, X., Yu, Q., 2016. Integrating POMDP and SARSA λ for service composition with incomplete information. In: International Conference on Service-Oriented Computing. Springer, pp. 677–684.

Wang, S., Zheng, Z., Sun, Q., Zou, H., Yang, F., 2011. Cloud model for service selection. In: IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, pp. 666–671.

Wang, S., Zhou, A., Bao, R., Chou, W., Yau, S.S., 2018b. Towards green service composition approach in the cloud. IEEE Trans. Serv. Comput..

Wang, H., Zhou, X., Zhou, X., Liu, W., Li, W., Bouguettaya, A., 2010. Adaptive service composition based on reinforcement learning. In: International Conference on Service-Oriented Computing. Springer, pp. 92–107.

Wei, Y., Kudenko, D., Liu, S., Pan, L., Wu, L., Meng, X., 2017. A reinforcement learning based workflow application scheduling approach in dynamic cloud environment. In: International Conference on Collaborative Computing: Networking, Applications and Worksharing. Springer, pp. 120–131.

Wen, Z., Yang, R., Garraghan, P., Lin, T., Xu, J., Rovatsos, M., 2017. Fog orchestration for IoT services: issues, challenges and directions. IEEE Internet Comput. 21 (2), 16–24.

Weyns, D., 2020. An Introduction to Self-Adaptive Systems: A Contemporary Software Engineering Perspective. John Wiley & Sons.

White, G., Palade, A., Cabrera, C., Clarke, S., 2018. IoTPredict: collaborative QoS prediction in IoT. In: IEEE International Conference on Pervasive Computing and Communications (PerCom). IEEE, pp. 1–10.

Wiesemann, W., Hochreiter, R., Kuhn, D., 2008. A stochastic programming approach for QoS-aware service composition. In: International Symposium on Cluster Computing and the Grid (CCGRID). IEEE, pp. 226–233.

Wohlin, C., 2014. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th International Conference on Evaluation and Assessment in Software Engineering. Citeseer, p. 38.

Wu, L., Quan, C., Li, C., Wang, Q., Zheng, B., Luo, X., 2019. A context-aware user-item representation learning for item recommendation. ACM Trans. Inf. Syst. (TOIS) 37 (2), 22.

Wu, Z., Xiong, N., Park, J.H., Kim, T.-H., Yuan, L., 2009. A simulation model supporting time and non-time metrics for web service composition. Comput. J. 53 (2), 219–233.

Xia, Y., Chen, P., Bao, L., Wang, M., Yang, J., 2011. A QoS-aware web service selection algorithm based on clustering. In: International Conference on Web Services. IEEE, pp. 428–435.

Xu, J., Guo, L., Zhang, R., Hu, H., Wang, F., Pei, Z., 2018a. QoS-aware service composition using fuzzy set theory and genetic algorithm. Wirel. Pers. Commun. 102 (2), 1009–1028.

Xu, J., Guo, L., Zhang, R., Zhang, Y., Hu, H., Wang, F., Pei, Z., 2017. Towards fuzzy QoS driven service selection with user requirements. In: International Conference on Progress in Informatics and Computing (PIC). IEEE, pp. 230–234.

Xu, C., Rajamani, K., Felter, W., 2018b. NBWGuard: Realizing network QoS for kubernetes. In: Proceedings of the 19th International Middleware Conference Industry. ACM, pp. 32–38.

Xu, L.D., Xu, E.L., Li, L., 2018c. Industry 4.0: state of the art and future trends. Int. J. Prod. Res. 56 (8), 2941–2962.

Xue, F., He, X., Wang, X., Xu, J., Liu, K., Hong, R., 2019. Deep item-based collaborative filtering for top-N recommendation. ACM Trans. Inf. Syst. (TOIS) 37 (3), 33.

Yao, L., Sheng, Q.Z., 2011. Particle filtering based availability prediction for web services. In: International Conference on Service-Oriented Computing. Springer, pp. 566–573.

Yasmina, R.Z., Fethallah, H., Fedoua, D., 2018. Selecting web service compositions under uncertain QoS. In: International Conference on Computational Intelligence and Its Applications. Springer, pp. 622–634.

Ye, Z., Bouguettaya, A., Zhou, X., 2014. Economic model-driven cloud service composition. ACM Trans. Internet Technol. (TOIT) 14 (2–3), 20.

Ye, H., Li, T., 2018. Web service composition with uncertain QoS: An IQCP model. In: CCF Conference on Computer Supported Cooperative Work and Social Computing. Springer, pp. 146–162.

Ye, Z., Mistry, S., Bouguettaya, A., Dong, H., 2016. Long-term QoS-aware cloud service composition using multivariate time series analysis. IEEE Trans. Serv. Comput. 9 (3), 382–393.

Yu, Q., 2012. Decision tree learning from incomplete QoS to bootstrap service recommendation. In: International Conference on Web Services. IEEE, pp. 194–201.

Yu, Q., Bouguettaya, A., 2010. Computing service skyline from uncertain QoSs. IEEE Trans. Serv. Comput. 3 (1), 16–29.

Yu, Q., Zheng, Z., Wang, H., 2013a. Trace norm regularized matrix factorization for service recommendation. In: International Conference on Web Services. IEEE, pp. 34–41.

Yu, L., Zhili, W., Lingli, M., Jiang, W., Meng, L., Xue-song, Q., 2013b. Adaptive web services composition using q-learning in cloud. In: World Congress on Services. IEEE, pp. 393–396.

Zambonelli, F., Castelli, G., Ferrari, L., Mamei, M., Rosi, A., Di Marzo, G., Risoldi, M., Tchao, A.-E., Dobson, S., Stevenson, G., et al., 2011. Self-aware pervasive service ecosystems. Procedia Comput. Sci. 7, 197–199.

Zhang, J.-h., 2010. A short-term prediction for QoS of web service based on RBF neural networks including an improved k-means algorithm. In: International Conference on Computer Application and System Modeling (ICCASM 2010), Vol. 5. IEEE, pp. V5–633.

Zhang, C., Patras, P., Haddadi, H., 2019a. Deep learning in mobile and wireless networking: A survey. IEEE Commun. Surv. Tutor..

Zhang, X., Wu, T., Chen, M., Wei, T., Zhou, J., Hu, S., Buyya, R., 2019b. Energy-aware virtual machine allocation for cloud with resource reservation. J. Syst. Softw. 147, 147–161.

Zhang, H., Yang, N., Xu, Z., Tang, B., Ma, H., 2018. Microservice based video cloud platform with performance-aware service path selection. In: International Conference on Web Services. IEEE, pp. 306–309.

Zhang, S., Yang, W., Zhang, W., Chen, M., 2019c. A collaborative service group-based fuzzy QoS-aware manufacturing service composition using an extended flower pollination algorithm. Nonlinear Dynam. 95 (4), 3091–3114.

Zhang, L., Zhang, T., Zhang, C., 2012. Web service composition algorithm based on hybrid-QoS and pair-wise comparison matrix. J. Inf. Comput. Sci. 9 (1), 135–142.

Zhang, L., Zou, H., Yang, F., 2011. A dynamic web service composition algorithm based on TOPSIS. J. Netw. 6 (9), 1296.

Zhao, L., Loucopoulos, P., Kavakli, E., Letsholo, K.J., 2019. User studies on end-user service composition: a literature review and a design framework. ACM Trans. Web (TWEB) 13 (3), 15.

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. Inform. Sci. 316, 370–396.

Zheng, Z., Ma, H., Lyu, M.R., King, I., 2009. Wsrec: A collaborative filtering based web service recommender system. In: International Conference on Web Services. IEEE, pp. 437–444.

Zheng, Z., Ma, H., Lyu, M.R., King, I., 2012. Collaborative web service QoS prediction via neighborhood integrated matrix factorization. IEEE Trans. Serv. Comput. 6 (3), 289–299.

Zheng, H., Yang, J., Zhao, W., 2010a. QoS probability distribution estimation for web services and service compositions. In: International Conference on Service-Oriented Computing and Applications (SOCA). IEEE, pp. 1–8.

Zheng, H., Yang, J., Zhao, W., 2010b. QoSDIST: A QoS probability distribution estimation tool for web service compositions. In: Asia-Pacific Services Computing Conference. IEEE, pp. 131–138.

Zheng, H., Yang, J., Zhao, W., 2016. Probabilistic QoS aggregations for service composition. ACM Trans. Web (TWEB) 10 (2), 12.

Zheng, H., Yang, J., Zhao, W., Bouguettaya, A., 2011. QoS analysis for web service compositions based on probabilistic QoS. In: International Conference on Service-Oriented Computing. Springer, pp. 47–61.

Zheng, Z., Zhang, Y., Lyu, M.R., 2010c. Distributed QoS evaluation for real-world web services. In: International Conference on Web Services. IEEE, pp. 83–90.

Zheng, Z., Zhang, Y., Lyu, M.R., 2014. Investigating QoS of real-world web services. IEEE Trans. Serv. Comput. 7 (1), 32–39.

Zhou, J., Yao, X., 2017. A hybrid artificial bee colony algorithm for optimal selection of QoS-based cloud manufacturing service composition. Int. J. Adv. Manuf. Technol. 88 (9–12), 3371–3387.

Zhu, M., Fan, G., Li, J., Kuang, H., 2018. An approach for QoS-aware service composition with GraphPlan and fuzzy logic. Procedia Comput. Sci. 141, 56–63.