Services Composition under Uncertainty: A Systematic Review and Future Directions

Mohammadreza Razian[a,b], Mohammad Fathian[b,*], Rami Bahsoon[c], Adel Nadjaran 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*

Abstract

Distributed computing paradigms such as cloud, mobile, Internet of Things, and Fog have enabled new modalities for building enterprise architectures through services 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 93 relevant studies published between the year 2007 and to-date. 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 services composition under uncertainty.

Keywords: Services Composition, Uncertainty, Cloud Computing, IoT, Mobile Computing, Fog (Edge) Computing

1. Introduction

Computing paradigm has shifted from traditional centralized service providing to the distributed computing paradigms [1, 2]. 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 **services composition** (**SC**) and recomposition. In service composition, the fundamental premise is that 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 node), either located in a smart city [3] or an Industry 4.0-based manufacturing system [4], can be considered as a potential source of service. Practically, IoT nodes expose their

[*] Corresponding author e-mail addresses: fathian@iust.ac.ir

Email addresses: `razian.mr@gmail.com` (Mohammadreza Razian), `fathian@iust.ac.ir` (Mohammad Fathian), `r.bahsoon@cs.bham.ac.uk` (Rami Bahsoon), `adel.n.toosi@monash.edu` (Adel Nadjaran Toosi), `rbuyya@unimelb.edu.au` (Rajkumar Buyya)

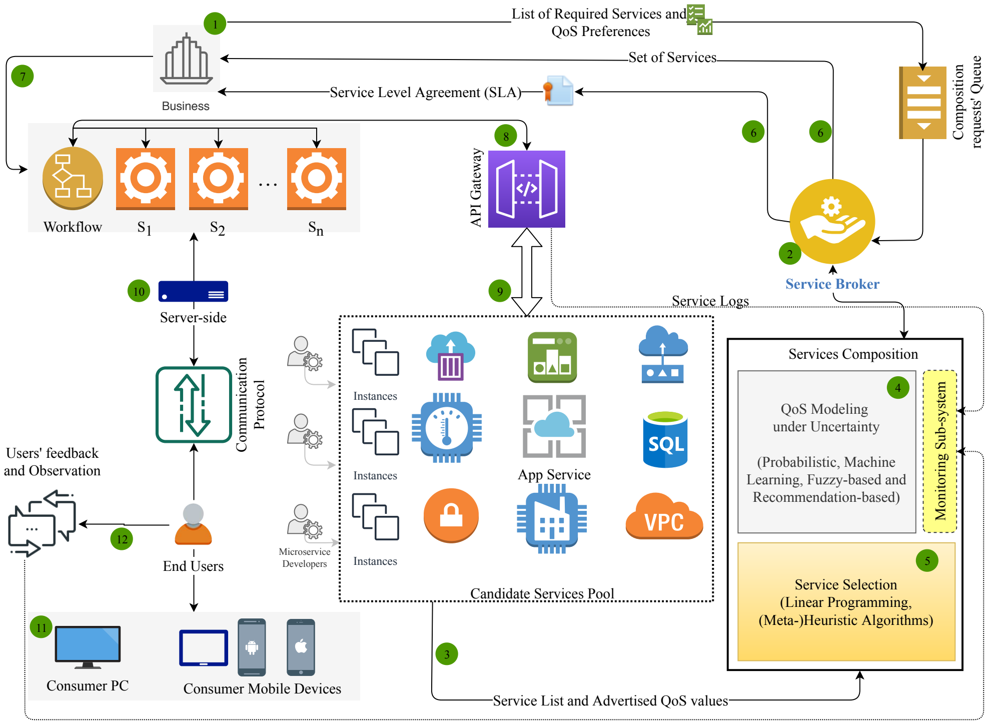

1
List of Required Services and QoS Preferences
Set of Services
Service Level Agreement (SLA)
Business
7
Composition requests' Queue
6
6
8
API Gateway
Workflow
S1
S2
...
Sn
2
Service Broker
10
Server-side
9
Service Logs
Communication Protocol
Instances
Instances
Instances
App Service
SQL
VPC
Microservice Developers
Candidate Services Pool
Services Composition
4
QoS Modeling under Uncertainty
(Probabilistic, Machine Learning, Fuzzy-based and Recommendation-based)
Monitoring Sub-system
5
Service Selection
(Linear Programming, (Meta-)Heuristic Algorithms)
Users' feedback and Observation
12
End Users
11
Consumer PC
Consumer Mobile Devices
3
Service List and Advertised QoS values

Figure 1: Main components and relations in QoS-aware service composition problem under uncertainty

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 [5] hub), and orchestrated with technologies like Kubernetes [6]. Hence, Cloud and IoT play a complementary role and potentially offer a tremendously large number of services distributed through ubiquitous communication networks [7]. In this situation, QoS-aware services 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 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). Besides, due to the variability of QoS values in real-world scenarios, which is called as **QoS uncertainty**, the QoS estimation/prediction in service composition has become more challenging. In literature, different approaches like Fuzzy set theory, Probabilistic (stochastic), and Machine learning have been widely used to address this challenge. In the following, we describe the main components of the services composition ecosystem along with the background and motivation of this survey.

1.1. Background and Motivation

Unlike traditional monolithic architecture for building an application, businesses have attracted much attention to distributed applications constructed upon microservices architecture [8, 9], 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 [10] (Development and Operations), helps to improve the modularity, flexibility, agility, scalability, and resiliency [11] 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 (knows as **QoS attributes**) define the quality aspects of a given service like response time, security, and reputation. As shown in Figure 1, a business (1) submits a composition request to the service broker [12]. 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). 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 services providers do not change over the time [13, 14]. 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 interest observations to note, Zheng et al. [15] explored the impact of uncertainty of response time attribute for *YouTube* service. They noted that the response time values could dramatically change and, therefore, 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 services composition. By using a QoS-aware service composition under uncertainty, businesses not only are able to respond to continuous changes of customers' requirements in a competitive market but also do not require to spend time/cost for service development from scratch. The current research trend has been on estimating and predicting QoS fluctuation and their likely consequences on SLA violations and mitigation strategies.

1.2. Goals of SLR

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 93 relevant studies published between the year 2007 and t-date. To the best of our knowledge, our study is the first of its kind to explicate the area of services 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 services 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 structured 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 current state. Section 4 provides SLR results, technical discussion and comparison on similar works within a category. Section 5 presents research implication, trend and future direction. 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 [16]. 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 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 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 preliminary search of the literature. All authors were involved in the process. The process was iterative, where the research questions and search strings had 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 [17] 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 synthesis them.

Table 1: Explored database 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/

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 Unccertainty
```

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 studies addressed the problem of uncertainty in SC. Besides, we selected those studies published between 2007 to to-date. This is because researchers have paid much more attention to web services with the popularity of cloud computing in its modern context from 2007 [18]. 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 [17]. We include all the peer-reviewed paper published between 2007 and to-date as follows:

- 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 are excluded:

- Papers that neither discuss nor consider uncertainty as part of its 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 which 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 considered the abstract, introduction, conclusion, and other parts of each paper.

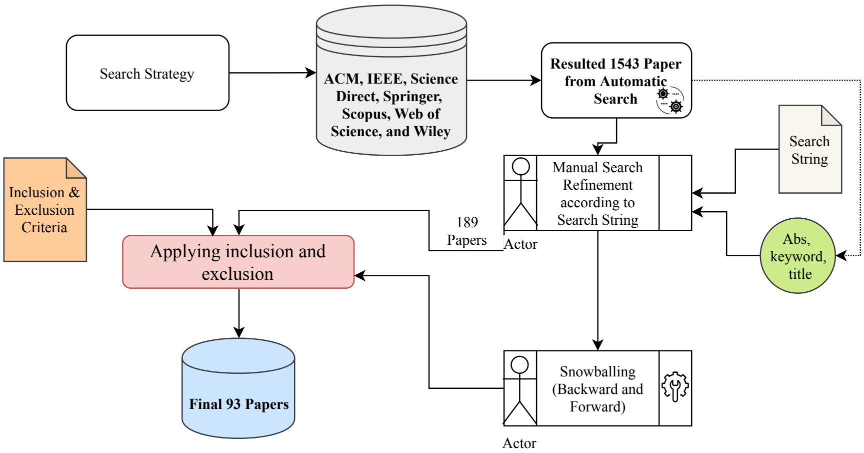

Search Strategy
ACM, IEEE, Science Direct, Springer, Scopus, Web of Science, and Wiley
Resulted 1543 Paper from Automatic Search
Search String
Manual Search Refinement according to Search String
Actor
Abs, keyword, title
Inclusion & Exclusion Criteria
189 Papers
Applying inclusion and exclusion
Final 93 Papers
Snowballing (Backward and Forward)
Actor

Figure 2: Process of studies (PSs) selection

2.2.3. Data extraction and synthesis

Furthermore, we check 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 **backward/forward snowballing** technique [19] 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 **93** most relevant papers as **primary studies** for undertaking reviews. Figure 2 depicts the whole process of study selection in our SLR. In addition, Figures 3 and 4 indicate the publication names and years of final selected primary studies (**PSs**), respectively.

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 Probabilistic methods, Machine Learning-based systems, Fuzzy SC, and Service Recommendation as main classes in proposed taxonomy. Figure 5 depicts the proposed taxonomy, concluded from the extracted studies. In the following, we review studies in each category.

3.1. Machine Learning

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 [20], which is frequently used in modeling QoS values in SC. Wang et al. [21] 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 [22] use RL for learning the last n service activities to defeat the changes in a run-time environment. Furthermore, Yu et al. [23] model SCP using Markov Decision Process (MDP) and generate the optimal policy using Q-learning. MDP is useful for studying optimization problems solved by reinforcement learning. In [24], 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.

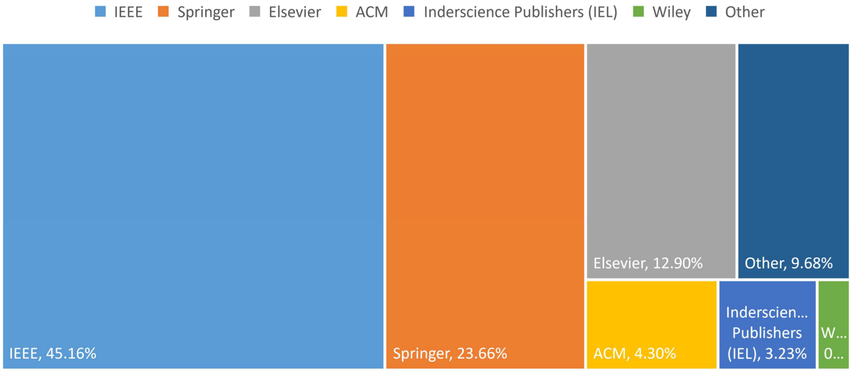


Figure 3: Publications name and frequency

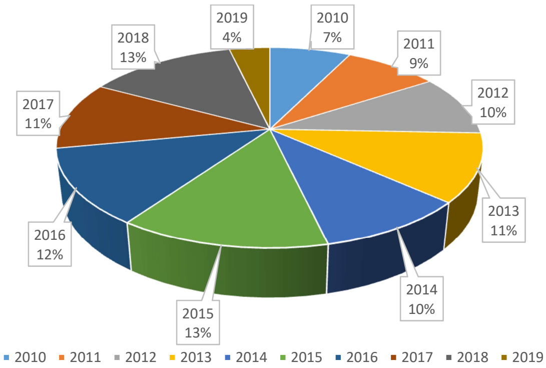


Figure 4: Publications year and frequency

To consider the incomplete information, Lei et al. [25] employ a Partially Observable MDP (POMDP). Additionally, they use reinforcement learning for SC with a Time-based Learning method [26] and maximum-expected total-discount-benefits criterion to compare policies. Wang et al. [27] 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 [28]. To predict the distribution of large-scale SC, Wang et al. [29] integrate the Gaussian process with RL and use Kernel function approximation. They perform an extensive evaluation of a real large-scale dataset. Moustafa et al. [30] 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 [30]. Recently, Mahfoudh et al. [31] proposed a framework for SC. 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* [32] for coordination model.

3.1.2. Clustering

The clustering algorithms [33, 34] attempt to categorize services according to similar functionalities or QoS attributes. Xia et al. [35] use a clustering density-based method named *OPTIC* to find the near-optimum composition. Recently, Khanouche et al. [36] 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 [37] proposes the Radial Basis Function neural networks (NN) with an modified *K-means* algorithm to predict QoS of web services. Yu [38] combines Matrix Factorization (MF) including a learning method based on decision tree to extract data from new clients. It is notable that new client, with no previous interaction information, are posed the cold start problem. Efstathiou et al. [39] 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 prediction of QoS using a multi-objective evolutionary algorithms NSGA-II. Surrogate models try to computationally estimate the fitness functions using techniques like Random Forest (RF) and Classification and Regression Trees (CART).

3.1.4. Regression

The regression algorithms attempt to approximate the mapping function from input QoS values to numerical or continues QoS values. Ye et al. [40] propose a prediction model using multivariate time series based on end-users long-term QoS-aware constraints and monitored QoS data. Sun et al. [41] introduce a time-series-based method to estimate the QoS values using run time captured data. Guo et al. [42] 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*" [43]. 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*" [43]. Recently, a time estimation model has been proposed [11] 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 motivated by [44], to obtain a proficient linear fitting.

3.2. Probabilistic

As shown in Figure 5, in the class of probabilistic, the common methods modeling QoS are 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) [45]. In the following, we present the main approaches in this category.

Optimistic/mean/pessimistic QoS. Wiesemann et al. [46] 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. [47] model the SC in IoT as a finite state machine. Because the service providers in IoT are *devices*, they focus on the reliability and specify the properties of SC using Probabilistic Computation Tree Logic and apply *PRISM* as a probabilistic model checking for verifying quantitative properties. In [48], 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 [49]. The location and bandwidth are taken into account to calculate the availability of a set of services in an uncertain mobile context.

For QoS estimation, Chen et al. [50] adopt two approaches: pessimistic estimation to present the worst value of QoS and probabilistic estimation to present an *expected value*. In [51], 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 [52] 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

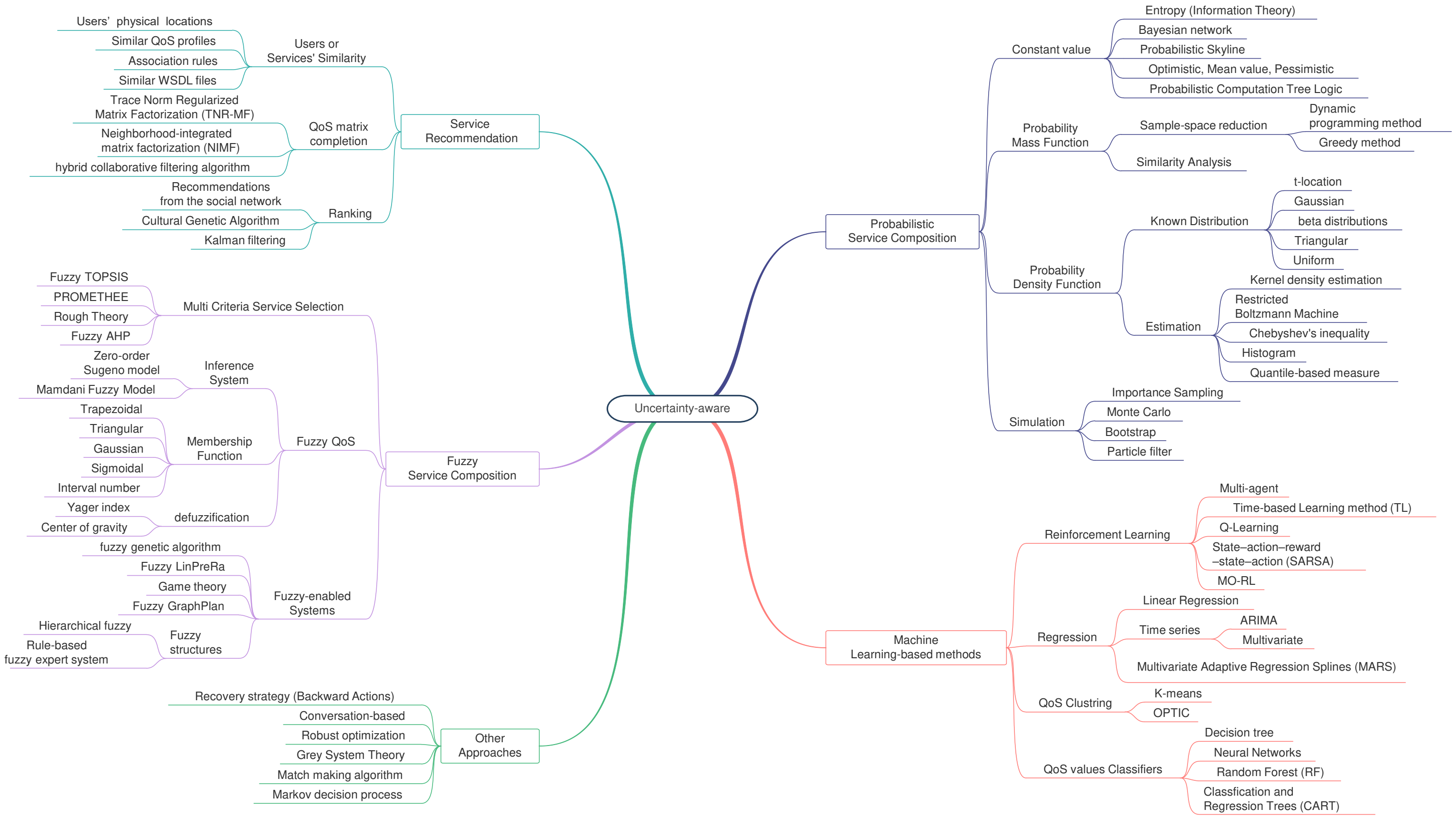


Figure 5: Taxonomy and approaches in QoS-aware service composition under uncertainty

time. Wang et al. [53] propose an SC in the field of "*cyber-physical social systems*" [53] employing Hofstede's cultural dimension theory [54]. 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 [55] of QoS values. Malik and Medjahed [56] consider Information Theory and propose a reputation propagation model to manage trust in SCs. The key criteria of service selection is service provider' reputation. Additionally, they evaluate the service providers' reputation regarding the credibility values of service raters (consumer's views) [57]. 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. [58] consider a two-phase architecture on the basis of the "*cloud model*" [59] 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*" [60] method is proposed to filtering 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 [61]. Yu and Bouguettaya [62] 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 [63, 64]. In [65], the authors incorporate the concept of share skyline computation with Genetic Algorithm (GA) for recomposition. Sun et al. [66] use the Particle Swarm Optimization (PSO) algorithm based on Skyline to select the candidate services.

Bayesian Network. Bayesian Network (BN) is the way of representing probability distribution. Chen et al. [67] 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. [68] 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 [15, 69]. Hwang et al. [70] 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. [71] 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 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 similarity between users.

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. [72] 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, 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 [73] predicts the availability of the service providers by considering Normal and Uniform distribution. Schuller et al. [74] remove the candidate services with higher variance. They improve the solution with ILP gradually and remove the fluctuated QoS attributes until the termination condition is satisfied. They extend their work in [75] using a

Genetic Adaptation algorithm to reduce computation time. Deng et al. [76] 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 [77] also assume 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 the construction of an estimate, based on the observed data. Zheng et al. [78] 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 [79] applied the same technique but using swarm intelligence to find the optimal composition. To increase the speed of the computation of convolution for the QoS, Zheng et al. [80] apply a fast Fourier transform and develop a tools called "*QoS DIstribution eStimation Tool (QoSDIST)*" [80] for service composition. In [81], a calculation method for different workflow structures (like repetitive and concurrent tasks) has been studied. For 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 [15], 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 continues values) is achievable from client-side, server-side or third-party monitoring system. Furthermore, quantile-based measure [82], Restricted Boltzmann Machine [83], and Chebyshev's inequality [84] 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. [85] 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 times is required to be less than a fixed value*"[85] that does not fit for real-world scenarios. While they state that a statement like "*a response time in less than T milliseconds for* 95% *of the cases*"[85] 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 [86] predict the availability in a given time-slot through a particle filter-based method. Wang et al. [87] employ the Importance Sampling technique to examine the QoS probability of composite service using "*stochastic Project Evaluation and Review Technique (PERT)*" [87].

3.3. Fuzzy Service Composition

A fuzzy model can be employed in situations where a QoS model should reflect experts' opinion due to lack of complete and reliable data for probabilistic QoS model construction. We categorize Fuzzy-based approaches in three classes [88]: 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 [89]. Şora and Todinca [90] design an architecture using fuzzy QoS properties containing domain ontology service, functionality finding module, QoS properties directory, and fuzzy ranker. Xu et al. [91] describe QoS attributes using a triangular fuzzy-valued for fuzzification, and *Yager* index for defuzzification. Like previous work, Veeresh et al. [92] 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 defuzzification process. Using rule-based fuzzy reasoning, [93] propose 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*" [94] has been employed. Recently, Niu et al. [95] 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. [96], a hybrid QoS model (i.e. different type of numbers intuitionistic and like triangular) is incorporated into "*TOPSIS*" [96] and "*AHP*" [97]. Mu et al. [98] 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*" [99] approach is proposed by using the dominance concept; hence, instead of simple additive weighting, the authors use "*PROMETHEE*" [100] ranking method.

3.3.3. Fuzzy-enabled Systems (FES)

Some approaches incorporate fuzzy theory into well-known techniques like Game Theory which we called them as *FES*. A composition technique using Fuzzy theory in a "*mobile ad-hoc networks*" [101] was developed. Also, a resource management middleware is used to asses 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*" [102] for the fuzzy inference. To address the problem of rule explosion, hierarchical fuzzy has been proposed in [103]. Zhao et al. [104] develop a multi-objective SLA-constrained SC on the basis of a "*fuzzy linguistic preference model*" [104] and weighted Tchebycheff distance. Fuzzy Game Theory [105], fuzzy neural networks [106], and Fuzzy SC with modified GraphPlan [107] are also have been utilized in the literature. Recently, Xu et al. [108] 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 [109]. Same 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 [110]. We categorized the existing approaches in three classes: Users or services similarity, QoS matrix completion methods, and Ranking.

3.4.1. Users or Services Similarity (USS)

[111] utilize the collaborative filtering (CF) method for finding services using users' similarities, association rules, and historical transactions. The basic idea of [112] is that the near users (geographically) can experience similar quality of service than far users. This idea can be justified because the users in almost same location can receive network traffic in nearly same quality. For identifying the similarity between regions, "*Pearson Correlation Coefficient (PCC)*"[112] is employed. Also, in [113], similarly between services using WSDL files is discussed, and the Jaccard similarity measure is applied.

3.4.2. QoS matrix completion

An important problem faces CF is handling "*new users with no previous interaction information*" [38]. To address this problem, Yu et al. [114] retrieve a large QoS matrix from a small portion of existing QoS records using Trace Norm Regularized Matrix Factorization algorithm. Zheng et al. [115] integrate item-based approach with 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* " [116] method for QoS value prediction is proposed by considering the users' previous observation on quality of services. To achieve a higher prediction accuracy, they combine neighborhood-based and the model-based CF. Additionally, Chen et al. [117] propose a personalized QoS model to solve the cold-start problem by using services/users geographical locations.

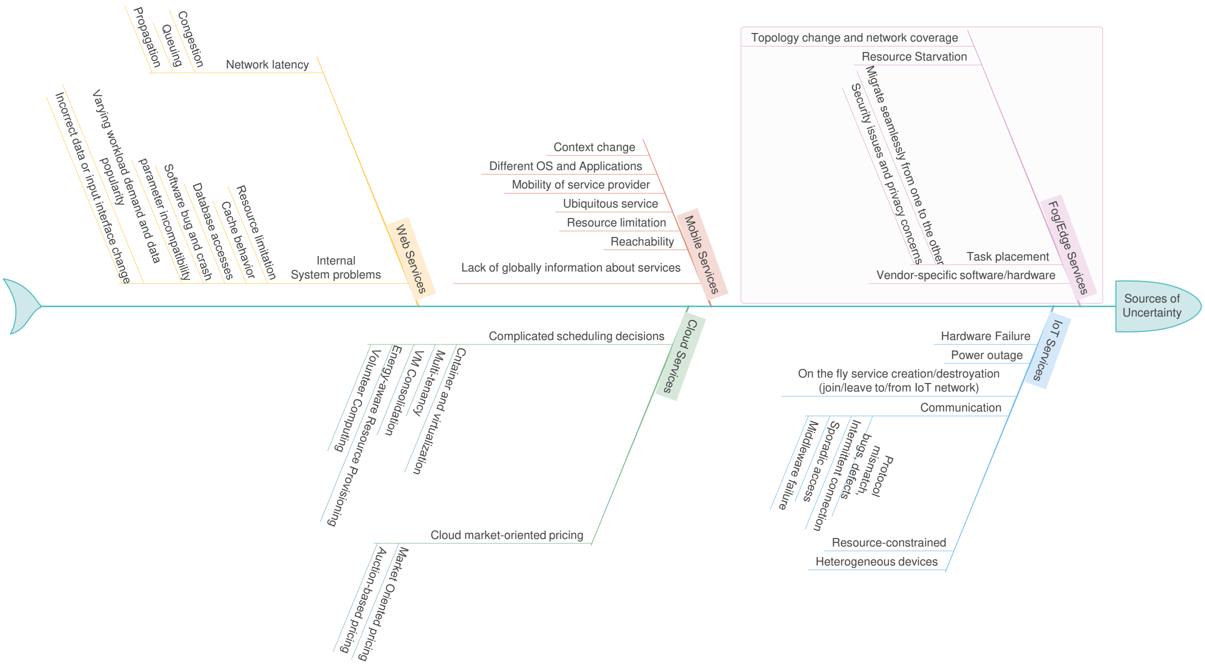

Congestion
Queuing
Propagation
Network latency
Resource limitation
Cache behavior
Database accesses
Software bug and crash
parameter incompatibility
Varying workload demand and data popularity
Incorrect data or input interface change
Internal System problems
Web Services
Context change
Different OS and Applications
Mobility of service provider
Ubiquitous service
Resource limitation
Reachability
Lack of globally information about services
Mobile Services
Topology change and network coverage
Resource Starvation
Migrate seamlessly from one to the other
Security issues and privacy concerns
Task placement
Vendor-specific software/hardware
Fog/Edge Services
Sources of Uncertainty
Complicated scheduling decisions
Cntainer and virtualization
Multi-tenancy
VM Consolidation
Energy-aware Resource Provisioning
Volunteer Computing
Cloud market-oriented pricing
Market Oriented pricing
Auction-based pricing
Cloud Services
Hardware Failure
Power outage
On the fly service creation/destroyation (join/leave to/from IoT network)
Communication
Protocol mismatch, bugs, defects
Intermittent connection
Sporadic access
Middleware failure
Resource-constrained
Heterogeneous devices
IoT Services

Figure 6: Reasons of uncertainty in Web, Cloud, Mobile and IoT services

3.4.3. Ranking

Kuter and Golbeck [118] propose an SC method, based on users' rating for a given service. For trust calculation, they use synthetic data adopted from "*FilmTrust*" [119]. Li and Wang [120] 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. [121] 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) [122]. Also, [123] 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. [124] use Grey system theory (a method for modeling and forecasting small sample time series). Ramacher and Mönch [125] 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. [126] 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. [127] propose a robust optimization to deal with QoS uncertainty based on "*Bertsimas and Sim robust*" [128] 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)*" [129] for representing user's tasks is developed. Chen et al. [130] propose dynamic service composition along with an mobile app named "*GoCoMo*", to self-organize the process of SC in a bluetooth-based mobile ad hoc networks.

4. Analysis of SLR results and Technical Discussion

In this section, we provide SLR results, technical discussion and comparison on similar works within a category. It is notable 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 uncertainty according to various service composition environments, including cloud, IoT, Mobile, etc.?

Figure 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 introducing cloud computing architecture, flexible service provisioning with virtually unlimited resources was replaced with previous simple web service architecture. Although, the concepts like Volunteer computing [84], Federated Cloud [131], Cloud market [132], VM Consolidation [133], Multi-tenancy [134], and energy-aware resource provisioning [135] 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 [136]. In a Mobile scenario, the composer placed in a device with mobility, identifies the existing services [129]. In this environment, the uncertainty of mobile services mostly relates to movements [137] of service requesters or service providers [76]. Additionally, the lack of stable and globally 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 [47, 138]. 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 as they are mostly hardware dependent; the devices equipped with different operating systems; applications come from various vendors [139].

4.2. Adopted Uncertainty-aware Approaches

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

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 uncertainty is because of 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). While the former is handled through probability theory, the latter is addressed by the possible theory which focuses on set-valued representations [140]. From Figure 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, without assuming a known distribution, try to estimate the QoS values distribution. However, QoS prediction using probabilistic methods needs to create a clear mathematical expression directly, which results in a nonlinear problem [37]. Another method in this approach is forming probability mass function (PMF) for each QoS attributes. The goal of this method is counting 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.

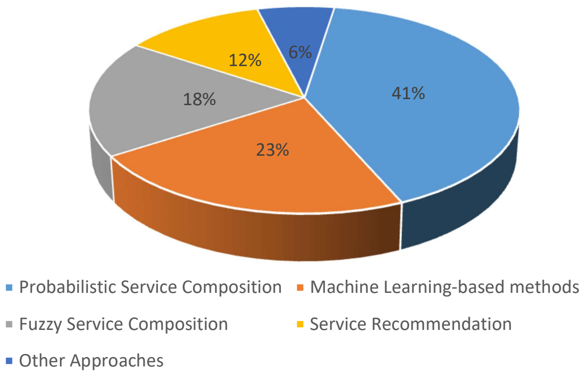


Figure 7: Percentage of adopted uncertainty-aware approaches

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 [141]. In 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 as *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 researches to construct the QoS model based on Machine learning algorithms. From Figure 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 weather missing or future value, and **uncertain services pruning**. For instance, [42] uses autoregressive integrated moving average for QoS forecasting, whereas [36] 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 literature, to overcome the incomplete ratings in a user-item matrix, the methods like "*neighborhood-integrated matrix factorization*" [116] and "*Trace Norm Regularized Matrix Factorization*" [114] 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. As a conclusion, it is demonstrative that researchers were more interested in using Machine learning based algorithm in recent years. We found that between 2015 to 2019, 31% of proposed approaches are based on Machine Learning, 28% Fuzzy Service Composition, 26% probabilistic, 9% Service Recommendation, and 6% for remaining approaches.

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 [91] to find the (near-)optimal composition. The heuristic and meta-heuristic approaches [73] 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 warranty to result in best composition and usually end with a *near-optimum* solution [142]. 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) [71, 76], single-objective and multi-objective evolutionary algorithm [83, 104], particle swarm optimization (PSO) [66, 79], Genetic Algorithm (GA) [41, 75, 91, 95, 108, 121, 123, 126], and NSGA-II [39] are examples of these approaches. Unlike (meta-)heuristic approaches, the mathematical optimization methods like Mixed Integer Programming (MIP) or Integer Programming (IP) [53, 60, 75], result in optimum composition and are best-suited for small-scale scenarios. It notable that time-consuming approaches would not fit in scenarios in which the user needs a composite service on time.

Furthermore, we observed that in the ML category, many researchers used MDP as a stochastic control process. The goal in and MDP model is finding the optimal policy using Q-learning and dynamic programming (Value or Policy Iteration). However, in the literature, authors normally employed Q-learning to learn an optimal policy. Q-learning is able to work in a stochastic environment based on rewards for each action. 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 Pearson Correlation Coefficient (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 Figure 8, we can see that the majority of studies (63.44%) considered the response time as an uncertain QoS attribute. Furthermore, we observed that availability (30.11%), reliability and throughput (similarly 21.51%), price (19.35%), and reputation (16.13%) had been usually modeled under uncertainty. However, 20.43% of studies did not point out the type of QoS attributes explicitly. Despite the high importance of energy consumption and security/safety, it has not received much attention (only 2.15% and 3.23%, respectively).

Scalability. We can observe from Table 3 that, 29.03% of PSs explicitly discussed scalability. The intensive increment in the number of services APIs in cloud and IoT-enabled smart environment need algorithms to work effective and efficient. Besides, factors like the number of features in service selection and amount of historical data play an important role in converting SCP to a large-scale problem. Figure 9 shows the tendency of researchers to consider scalability as a crucial feature in their solutions (only two studies between 2007 to 2010 versus 15 studies between 2015 to 2019 considered scalability). Furthermore, we explored the portion of scalable approaches, which is Machine learning 14.81%, Fuzzy service composition 22.22%, Recommender 25.93%, and Probabilistic 33.33%. The result shows that all approaches potentially

Table 2: Primary studies approaches and solving methods

Fuzzy-based SC	FQoS	Fuzzy min-max composition MATLAB [93], Fuzzy GA [91], Interval number using multi-objective GA [95], Self-optimization [89], Fuzzy inference [90], AODV protocol [92]
	FES	Game theory [105], Triangular fuzzy GA [108] Fuzzy GraphPlan [107], Fuzzy LinPreRa evolutionary algorithm [104], Fuzzy Neural Networks [106], Hierarchical fuzzy [103], Rule based fuzzy expert Sugeno model MATLAB [101]
	MCSS	Fuzzy AHP - interval-based [97], Fuzzy TOPSIS [96], Fuzzy and Rough Set [98], Interval-based PROMETHEE and GA [99]
Machine Learning-based methods	Clu.	Kmeans with search tree [36], OPTIC with heuristic algorithm [35]
	Reinforce	Partially Observable MDP [26], Single and multiple policy multi-objective composition scenarios [28], Multi-Agent RL with Qlearning,ϵ-greedy exploration algorithms [31], Q-Learner [21], [22], [24], [23] Time-based Learning method [20], deep RL [30], Q-learning based on gaussian process [29], SARSA(λ) [27], time-based (Q-learning) [25]
	Class.	Decision Tree and matrix factorization [38], RBF neural networks + improved K-means [37], Classification and Regression Trees (CART), Random Forest [39]
	Regression	PSPAS with shortest path algorithm [11], Linear Regression, MARS, NSGA-II [39], ARIMA-BASED Time Series and GA [41], ARIMA and Skyline using 0-1 MIP [42], Multivariate ARIMA and Holt-Winters using R [40]
Probabilistic Service Composition	Constant Value for QoS	Bayesian network (BN), extended Shenoy-Shafer to solve Hybrid influence diagram [68], Improved BN, SMILE engine [67], Anytime algorithm using DFS [51], Average value at risk MILP, CPLEX [46], MDP with FSM, temporal logic PCTL, probabilistic model checking with PRISM [47], Probability of context change, [49], Hofstedes cultural dimension MIP [53], LPSolve and NSGA-II [52], Skyline, Int. Prog., heuristic algorithm [65], Expected/Pessimistic value Estimation [50], Bubnicki model [48], Back tracking search [45], Entropy and variance IP [60], Expected value, Entropy, and Hyper-Entropy: [56], Finding providers' reputation [57], Finding uncertain services [58], Lp-Solve [55], *p-R-forest* [62], PSO + skyline [66] p-dominant Service Skyline with R-tree data structure
	Known Distr.	Stochastic timed colored Petri net [72], known PDF, CPLEX [77], Normal distribution, CPLEX and Greedy adaption heuristic [74], Triangular, Uniform Distribution, GA and CPLEX [75], Uniform distribution, heuristic and metaheuristic algorithm [73], Normal distribution, simulated annealing [76]
	Unknown Distr.	Quantile-based measure, MIP and iterative approach [82], Gaussian Kernel Density estimation and fast Fourier transform [78], Kernel density estimation, PSO [79], Restricted Boltzmann Machine, evolutionary algorithm [83], PDF-Calculation with DFS algorithm [15], histograms [81], Dynamic histograms, Chebyshev's inequality [84]
	PMF	Dynamic prog. and Greedy method [70], Prolog [69], Similarity Analysis, simulated annealing [71]
	Sim.	Particle Filter based Algorithm [86], Bootstrap-Based Simulations, T Location-Scale Sampling-Based Simulations [85], Importance Sampling technique [87]
Recommendation	Matrix	Collaborative filtering (CF), neighbor PCC [115], Matrix factorization, Clustering, geographical neighbors improved PCC [117], Neighborhood-integrated matrix factorization [116], Trace Norm Regularized Matrix Factorization [114]
	Ranking	Case-Based Reasoning - Manhattan distance GA based approach [121], Ranking Kalman filtering - Euclidean distance in MATLAB [120], Modified HTN and branch-and-bound, SHOP2 planning system [118], Ranking within a category, recommendations from the social network, GA [123]
	USS	Similar WSDL files by Jaccard similarity [113], Users' physical PCC for region similarity [112], Collaborative filtering and association rules [111]
Others	-	Bertsimas and Sim Robust Optimization [127], MDP and Mixed Integer Programming [125], Backward-chaining, NS3 nimulator [130], Conversation-based SC IOPE Hybrid-Cosine method, [129], Grey sequence prediction model [124], Deterministic finite automata/recovery strategy, GA [126]

can present scalable composition. However, the crucial aspect is how to solve the resulted uncertainty-aware model. It is worth mentioning that 44.44% of scalable approaches use (meta-)heuristic algorithm as a solving

Table 3: Metrics and Dimension used in PSs

Metrics and Dimension	**Studies**
Multi-objective	[28, 39, 42, 46, 52, 62, 66, 95, 98, 99, 104, 108, 123]
Context-aware	[31, 49, 90, 93, 106, 129, 130]
Adaptive	[11, 21–31, 35, 40, 50, 65, 83, 84, 86, 89, 92, 93, 103, 120, 126, 129, 130]
Scalability	[15, 29, 30, 35, 42, 50, 52, 62, 66, 73, 79, 83, 84, 91, 93, 103, 104, 106, 108, 112, 114–117, 121, 123, 126]
Multi-provider	[31, 39, 40, 62, 68, 73, 76, 84, 92, 98, 113, 123, 130]
Motivation Example	[11, 23, 25, 26, 28, 39, 40, 45, 47, 49–51, 53, 55, 56, 62, 65, 67, 68, 72–76, 78, 84, 90–93, 95, 97, 98, 101, 104, 105, 108, 112, 113, 117, 118, 121, 123, 126, 129, 130]

Table 4: Studies and used QoS attributes

QoS Attribute	**Studies**
Availability	[22, 28–30, 36, 37, 47–49, 67, 74–77, 83, 86, 91, 95–97, 99, 101, 103, 107, 108, 121, 123, 129]
Reliability	[22, 29, 30, 36, 37, 39, 65, 70–72, 77, 91, 93, 96–98, 108, 121, 123, 124]
Response time	[11, 15, 21, 23, 24, 28–30, 36–41, 46, 52, 58, 60, 65, 69–75, 77, 78, 81–85, 87, 89, 91–93, 95–99, 103, 106–108, 112–117, 120, 121, 124, 125, 127, 130]
Price (cost)	[21, 28, 36, 37, 40, 46, 68, 70, 72, 74, 77, 89, 93, 96, 98, 105, 108, 124]
Reputation (fidelity)	[56, 68, 70, 71, 77, 84, 95–98, 103, 108, 118, 123, 124]
Throughput	[23, 29, 36, 40, 41, 60, 65, 68, 74, 75, 83, 91, 92, 99, 107, 108, 116, 120, 123, 127]
Bandwidth	[37]
Energy	[39, 89]
Security (safety)	[98, 124]
Free trial	[98]
Accessibility	[123]
Success rate	[91, 108]
Not specified	[20, 23, 25, 27, 29, 35, 42, 45, 47, 50, 51, 53, 55, 62, 66, 79, 90, 104, 126]

method.

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 percentage (15.05%) of existing studies attempted to model QoS as a multi-criteria [98] or multi-objective problem [66]. 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 clarify the problem of SC in service computing architecture [90], many researchers (46.24%) present motivation scenario or example. As shown in Figure 10, Trip planning/travel booking [50, 55, 56, 62, 68, 104, 108, 123, 126, 129] and Online shopping [40, 45, 51, 65, 67, 72, 91, 121, 130] are the most used scenario examples. Furthermore, brokerage and service oriented architecture [49, 74, 75, 78, 95], cloud market services/Volunteer computing [28, 84, 105, 113], video application [11, 76, 101], geographical information and mobile navigation [97, 117], fire fighting [39, 47], supply chain service [23, 25], mobile ad hoc networks [73, 92], Information warfare [26], online parcel delivery [93], online service searching [112] and E-Health system [118] are other scenarios used in the literature.

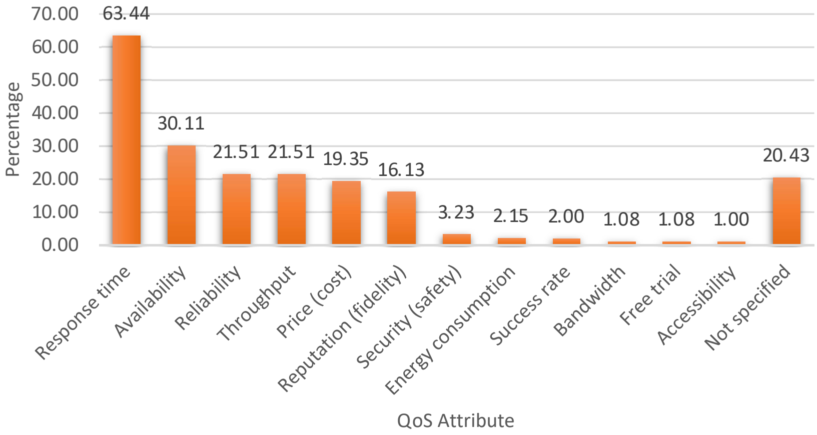


Figure 8: Percentage of each QoS treated as an uncertain attribute

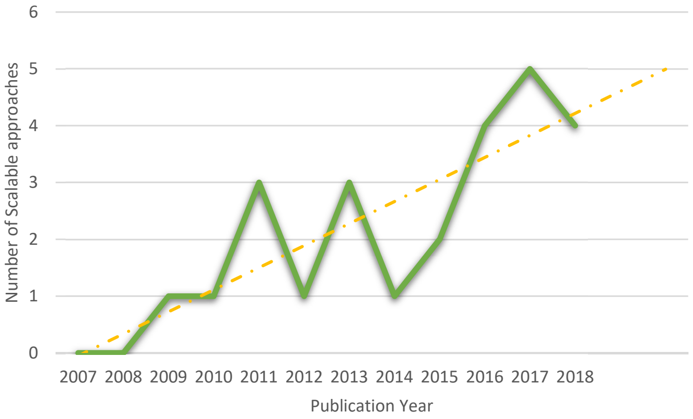


Figure 9: Frequency of scalable approaches by year

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 have pointed to multi-sources service composition explicitly (13.98%).

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, 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 (7.53%).

Adaption. We observed that 29.03% of previous studies explicitly pointed out that they work adaptively. Because of the changes in service computing and the dynamicity of the communication networks environ-

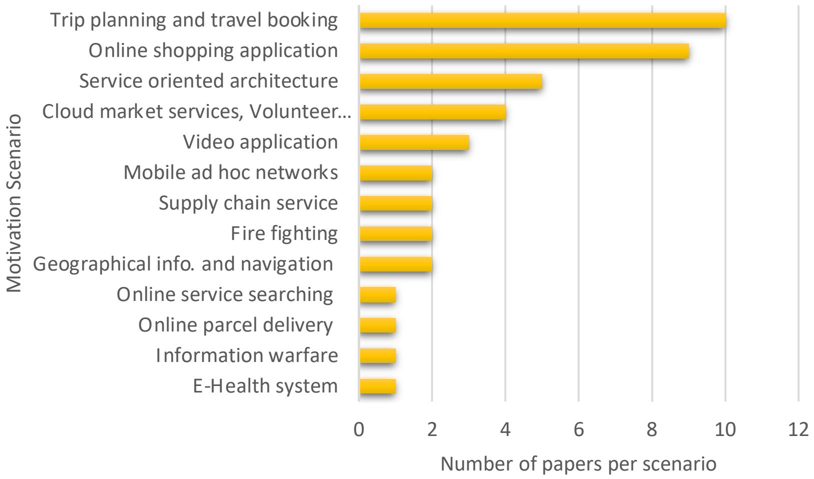
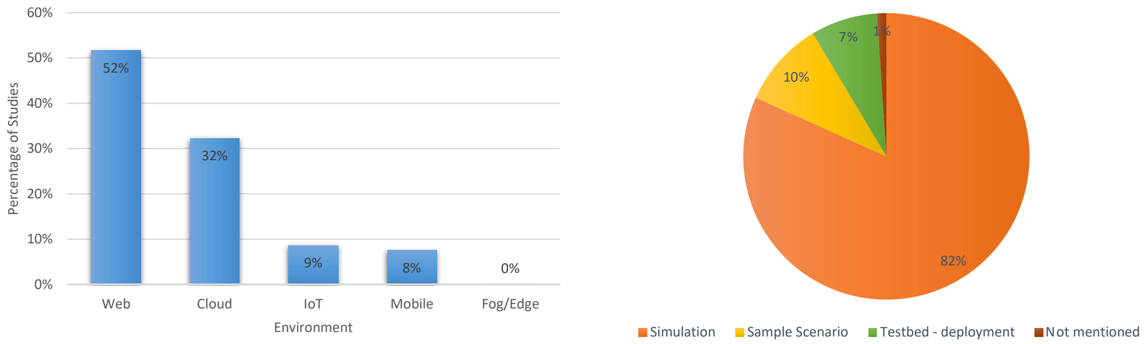


Figure 10: Frequency of motivation scenarios



(a) Percentage of adopted environments

(b) Percentage of applied evaluation methods

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

ments, SC needs to be steadily adapted to operate continuously and effectively [15] which leads to mitigate the SLA violence penalty [89], especially in a dynamic environment where service failures and QoS degradation take place commonly [52]. For example, in [35], they re-select concrete services if the QoS values of a currently selected services change. However, if replacing alternative services takes more cost, or there is no alternate service (according to users' preferences), this approach will fail [50]. To eliminate this limitation, researchers are recommended to model environments through agent-based learning systems [28] and adapt to changes pro-actively.

4.4. Environment

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

The challenging problem is not only how to make an efficient SC over the wide variety of heterogeneous services, but also how their uncertain QoS values can be managed and aggregated according to **distributed service environment**. In this SLR, we have investigated *traditional Web*, *Cloud*, *IoT*, *Mobile*, and *Fog/Edge*. The majority of previous studies (51.61%) have been proposed 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 fashion; therefore, the development of

Table 5: The popular datasets used in the literature

Ref.	Name	Year	data	Accessibility	Num.
[148–150]	QWS dataset	2008	real	email to author	365
[151, 152]	WS-DREAM Dataset1	2016	real	download from website	5,825
[151]	WS-DREAM Dataset2	2016	real	download from website	4,500
[153]	OWLS-TC4	2010	NA	download from website	1.083
[154]	I-QoS	2012	real	NA	825,132

cloud SC became more interesting for researchers (32.26%). From Figure 11a, 7.53% of researchers proposed Mobile SC. Variety and movement of mobile devices turn service composition to become extremely sensitive to changes in a communication infrastructure [39]. 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 [143].

As shown in Figure 11a, 8.6% of studies have focused on IoT. The functionalities offered by smart objects are usually abstracted as software services [36]. In the IoT paradigm, real-life objects (intelligent objects) can be considered as a source of service [48]. From Figure 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 [143]; hence, a composer needs to be able composing services flexibly and assigns services in operation to mitigate composition collapse [130]. Finally, because Fog/Edge services have been recently introduced in the literature, the researchers have started proposing service composition in these environments [144–147]. However, currently, there does not exist uncertainty-aware Fog/Edge SC in selected PSs.

Evaluation. From Figure 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 [129], Google App Engine [69], PlanetLab [116], Amazon EC2 and Weka [113] are some examples of exploited infrastructures. Furthermore, Mahfoudh et al. [31] 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. Avila and Djemame[89] 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 exist enough QoS historical data. More precisely, they used historical data to calculate the mean and variance of QoS values [24], train algorithms [83] like BP networks [37], create PMF or infer probability distributions [70, 74, 87], find probability of failure [65], construct economic models [68], and extract fuzzy rules [106]. From the literature, it is assumed that historical data usually can be originated [15] from service execution logs [30], QoS monitoring mechanisms [62, 82], asynchronous monitoring [50], online monitoring subsystems [21, 41, 93, 126], and social network [123]. In some studies, the model has been developed based on expert (decision-maker) opinion [93] for perturbation level [127] or confidence index [48]. Also, [130] presumed that there exist a global semantic matchmaker and [129] considered each resource as an autonomous component.

Furthermore, we discovered following assumption/requirement: provider's adaptation policy is accessible [39, 79] and negotiable [105], real-time context information are provided [49], service availability values are provided by the supplier [49], parameters of algorithm [35] like number of clusters [36], exploration and exploitation [31] [36], state transition [28] internal features of services [113] are achievable/tunable/available. Some researchers assumed that statistical description around QoS attributes are available in advance. For

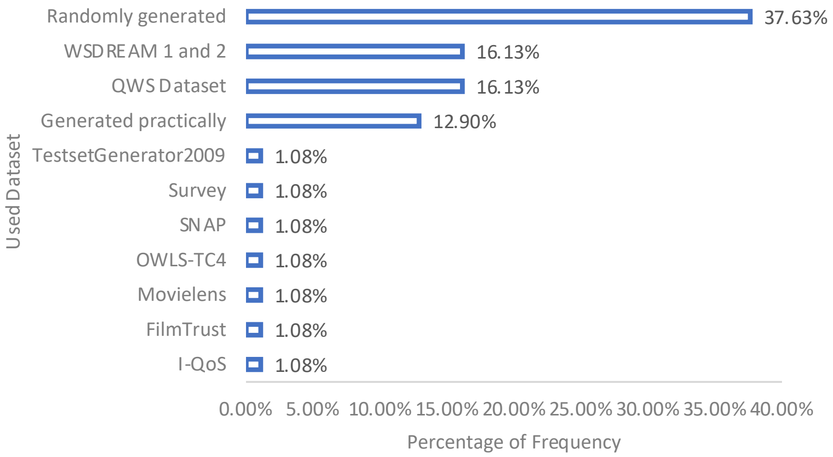


Figure 12: Percentage of dataset used for evaluation

example, "*QoS are represented as histograms with the same start point and intervals width*" [81], threshold values of Entropy and Hyper-Entropy [55] are determined, availability of mobile device in a time slot is identified [73], probability of staying in a required distance to the service requester [76], and Geographical information [117] are achievable. Furthermore, they supposed that the distributions of QoS attributes are known like beta [46], normal [23, 26, 72, 74, 125], Poisson and exponential [72]. In addition, decomposition of global constraints [71] to local constraints is presumed. Although, this process speeds up the selection phase, it may lead to an inaccurate QoS modeling. In recommendation systems, [113, 114] developed their model under the assumption that similar users or services may experience same QoS. Yu et al. [114] assumed that "*QoS matrix has a low-rank or approximately low rank structure*". Many researches in this category assumed that the user ratings are available [38, 53, 56, 57, 98, 112, 115, 118]. 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 Figure 12, we can see that 33.33% of studies have evaluated their proposed method by using randomly generated QoS values. Some researchers generated random datasets on the basis of the behaviour of services on the Internet [75]. In addition, random dataset following normal distribution [21, 27, 28, 68, 74], exponential [72], uniform [67, 76], and beta [46] distribution have been generated. Table 5 provides the datasets have been used in the literature. The QWS dataset, which is significantly used in literature (16.13%), has been collected by Al-Masri et al. [148, 150]. Another popular dataset used in the literature is WS-DREAM (16.13%). 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 5,825 services [151]. The second dataset (WS-DREAM dataset2) contains QoS observation of 142 users on 4,500 services through 64 consecutive time slices [151].

Researcher also used datasets from other domain such as FilmTrust [118], Movielens [111], and SNAP [123] in SC. Furthermore, 12.9% of researchers have generated datasets practically [15]. Chen et al. [130] used the NS3 simulator to generate their required QoS data. Also, 20.43% of studies have not mentioned about their used dataset.

5. Research Implication 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 Figure 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 [1]. Although cloud providers like Google and Cloudflare 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 [155], to host computational resources in the vicinity of end-users [156]. Additionally, while the solutions like software-defined networking (SDN) and network functions virtualization (NFV) [157] make networking architectures more flexible () and efficient [158, 159], they call researchers to investigate the uncertainty factors on quality aspects of service. From Figure 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, where distributed intelligent devices act as both services consumers and providers.

Adaption. To recover the non-functional aspects of an undertaken composite service, 29.03% of previous studies explicitly 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 [46], the time wastage can damage the functionality of a delay-sensitive application. 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 [160]. Second, *proactive adaptability* through learning methods can adjust related parameters continuously depending on the QoS changes [31]. Therefore, a QoS prediction model that develops an adaptive SC, ensures the completion of composite service in runtime without failure [121] and considers a minimum required time is still an open research challenge.

Multi-Source Services Composition. From the literature, the majority of PSs (about 86%) assumed that services are provided from a single source. However, using a single provider model poses multiple drawbacks such as single point of failure 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 [161], 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 [161], interoperability between the existing cloud and complex maintenance [162]. 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 15.05% 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 [163]. Considering the Pareto approach for SC permits the decision-maker to select the desired composition with respect to preferences on the conflicting QoS attributes. Besides, evolutionary algorithms like NSGA-II [52, 164]

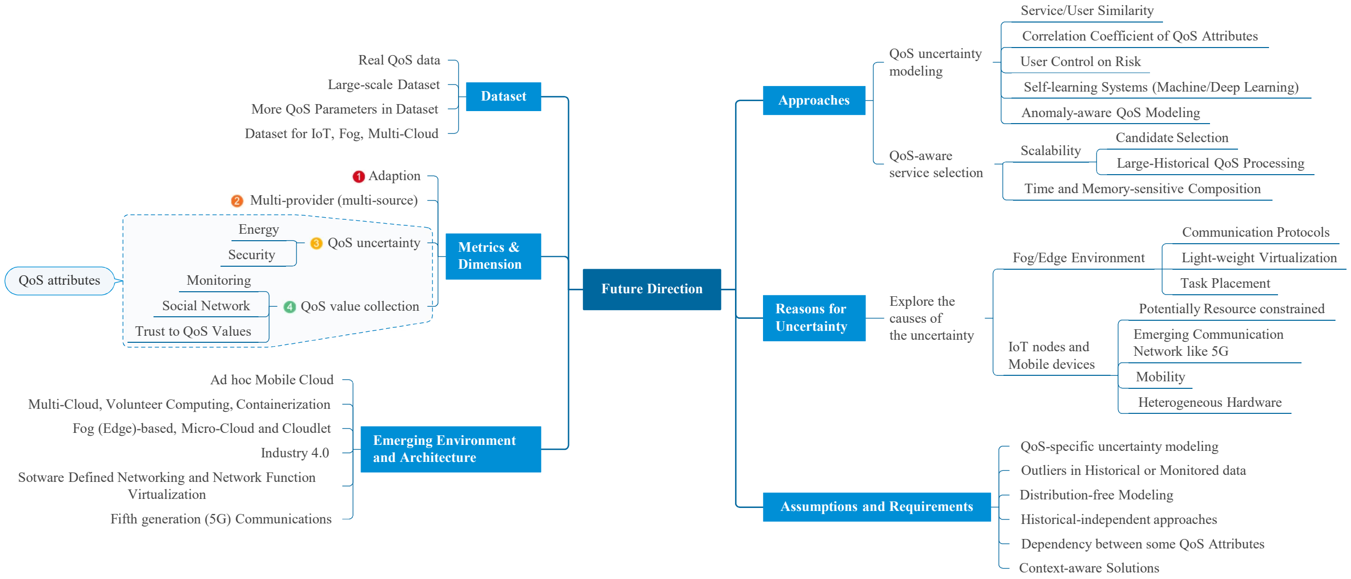

Future Direction
Dataset
Real QoS data
Large-scale Dataset
More QoS Parameters in Dataset
Dataset for IoT, Fog, Multi-Cloud
Metrics & Dimension
1 Adaption
2 Multi-provider (multi-source)
3 QoS uncertainty
Energy
Security
4 QoS value collection
Monitoring
Social Network
Trust to QoS Values
QoS attributes
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
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

Figure 13: Research implication and future directions

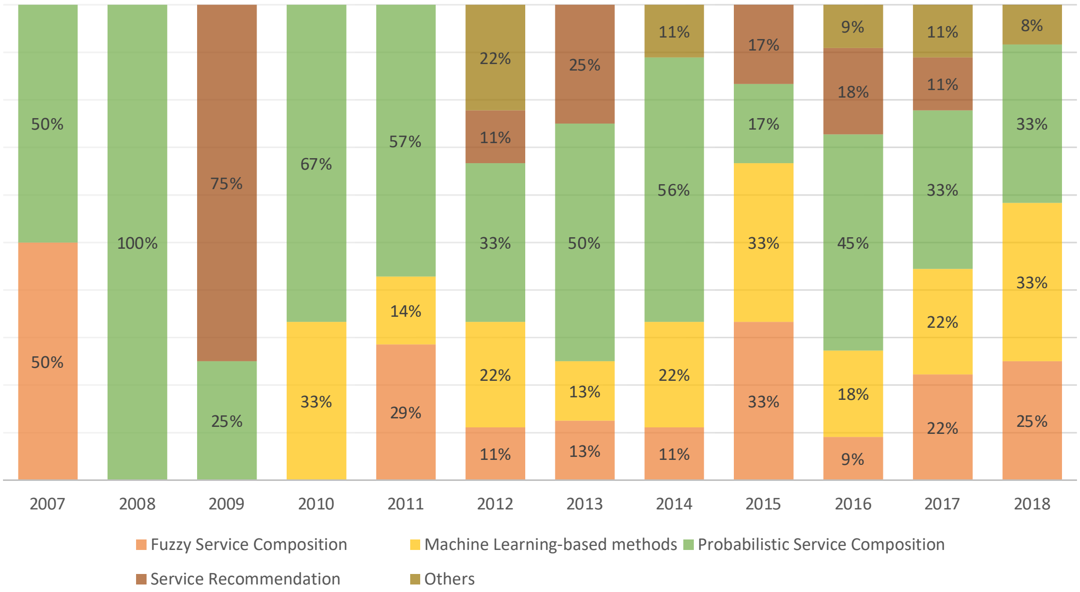


Figure 14: Proposed approaches by year

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 and decision spaces [165].

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 a reliable composition suits for 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 [113, 114]. 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 [15] shows that it cannot be fitted to standard probabilistic distributions. Although simulation approaches like [71] 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 [73]. From Figure 14, moving from 2007 to 2019, 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 [28, 36] is a promising direction for future researches.

Data-driven Approaches. Applying data-driven techniques is a promising future direction [166] 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:

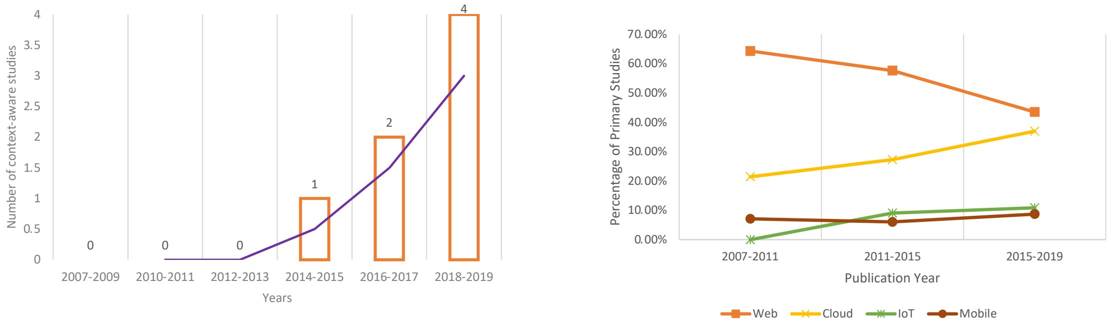
- Missing values: monitoring devices, especially in large scale environments, may fail to collect all service logs. Also, in Service Recommendation class, lack of rating, especially for new services, leads to a sparse user-service matrix.
- Data accessibility: in distributed systems, it is not always possible to collect historical QoS values because of export rules and privacy statements.
- Integrity: Some collected data are out of date [91] when services are deprecated or upgraded [163]
- Scale: increasing in velocity and volume of meta-data generation in distributed service computing environments like IoT [139] 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) seems to be crucial.

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

Anomaly Detection. In [167], 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 [91] lead to an anomaly in QoS historical values. Therefore, service logs are not suitable for mining directly (without pre-processing) [168, 169]. As an example, we can point network congestion occurred in Beijing, China [154] 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 use 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 [112, 113] 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 [170, 171] into the user/service rating matrix. Furthermore, there are only a few studies that apply *services* 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 [172] can be investigated along with the improvement of well-known similarity measures like PCC [173, 174], Cosine-based, Euclidean distance, or Jaccard similarity [113].

Dataset. In Section 4.6, we discussed on datasets used in literature. These datasets only include the QoS values of the tradition 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 Figure 12, many researchers have used QWS and WSDREAM datasets. However, there exist some shortcoming in these datasets: First, the number of services included in the QWS dataset cannot meet



(a) The trend in moving towards context-aware approaches

(b) Discussed Environment between 2007 to 2019

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

the requirements of large-scale scenarios [29]. 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. Furthermore, a dataset with more QoS parameters like energy consumption, security, reputation is still a gap in the literature.

QoS Attributes. From Figure 8, except response time (59 studies from 93), the majority of other QoS attributes have not received much attention. For example, energy consumption, as an important aspect of real-world services, has not received much attention in QoS modeling under uncertainty (solely two works targeted for). This limitation also relies on other parameters like security (only three works were found). Hence, modeling these parameters under uncertainty invokes more effort. In the literature, about 20% of studies have not specified the type of QoS attributes that have been modeled under uncertainty. We argue that we cannot apply a common QoS model for all QoS attributes. 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 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. Figure 10 shows that the majority of motivation scenarios in primary studies have been devoted to the traditional web environment than 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 [175] (including Smart Lighting, Smart Building, Smart Energy, Smart Healthcare), Industry 4.0 [4], and upcoming 5G systems [2]. This is because the real deployment of SC relies on the fundamental scenarios, e.g., the number of the service provider, environment conditions and assumptions, source of services, the applications, context, and an actual service providers (cloud, IoT). Furthermore, we discovered the lack of that 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 [70] 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 [90] 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 Figure 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 context while composing services for the associate environment is a considerable challenge which calls the researcher.

Sources of Uncertainty. From Figure 15b, we can see that SC under uncertainty has shifted from simple web services to the cloud and heterogeneous IoT services [176–178]. 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 Figure 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 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 29.03% 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. Furthermore, some machine learning techniques like Deep Reinforcement Learning are capable of solving large-scale complex optimization models [30]. 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 a number of 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 has 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 is primarily about QoS uncertainty in SC. We tried to answer each question in Section 4 and provide corresponding research implication and future directions in Section 5. It may be worthwhile to investigate uncertainty in other phases of the services 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 93 most relevant studies published between the year 2007 and to-date. This SLR provides a taxonomy, comparisons, and analysis of the state-of-art in services 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 (51.61%). Additionally, 32.25% of studies focused on cloud services, and only a small portion of primary studies considered IoT (8.6%) and Mobile (7.52%) services. We observed that emerging service environments like fog/edge have not yet been used for modeling the uncertainty of QoS attributes. This is justified by the fact that IoT- and fog/edge-based services are new technologies. Concerning adopted approaches, we found that the most widely used approach for solving service composition under uncertainty was probabilistic (41%). Additionally, 23% of studies employed Machine Learning-based methods, 18% Fuzzy Service Composition, and 12% of studies focused on Service Recommendation approaches. We identified that the most commonly considered QoS attributes were the response time (63.44%), availability (30.11%), reliability and throughput (21.51%), price (19.35%), and reputation (16.13%). 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 an 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 services 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 research into services-aware composition with a new perspective like uncertainty and fuzziness in user preferences, service descriptions, or business workflow.

References

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

[2] I. B. de Almeida, L. L. Mendes, J. J. Rodrigues, M. A. da Cruz, 5g waveforms for iot applications, IEEE Communications Surveys & Tutorials.

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

[4] L. D. Xu, E. L. Xu, L. Li, Industry 4.0: state of the art and future trends, International Journal of Production Research 56 (8) (2018) 2941–2962.

[5] E. A. Santos, C. McLean, C. Solinas, A. Hindle, How does docker affect energy consumption? evaluating workloads in and out of docker containers, Journal of Systems and Software 146 (2018) 14–25.

[6] C. Xu, K. Rajamani, W. Felter, Nbwguard: Realizing network qos for kubernetes, in: Proceedings of the 19th International Middleware Conference Industry, ACM, 2018, pp. 32–38.

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

[8] S. Roca, J. Sancho, J. García, Á. Alesanco, Microservice chatbot architecture for chronic patient support, Journal of Biomedical Informatics (2019) 103305.

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

[10] L. Bass, I. Weber, L. Zhu, DevOps: A software architect's perspective, Addison-Wesley Professional, 2015.

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

[12] M. Anisetti, C. Ardagna, E. Damiani, G. Polegri, Test-based security certification of composite services, ACM Transactions on the Web (TWEB) 13 (1) (2019) 3.

[13] J. Zhou, X. Yao, A hybrid artificial bee colony algorithm for optimal selection of qos-based cloud manufacturing service composition, The International Journal of Advanced Manufacturing Technology 88 (9-12) (2017) 3371–3387.

[14] C. Jatoth, G. Gangadharan, U. Fiore, R. Buyya, Qos-aware big service composition using mapreduce based evolutionary algorithm with guided mutation, Future Generation Computer Systems 86 (2018) 1008–1018.

[15] H. Zheng, J. Yang, W. Zhao, Probabilistic qos aggregations for service composition, ACM Transactions on the Web (TWEB) 10 (2) (2016) 12.

[16] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain, Journal of systems and software 80 (4) (2007) 571–583.

[17] B. Kitchenham, Procedures for performing systematic reviews, Keele, UK, Keele University 33 (2004) (2004) 1–26.

[18] R. Buyya, J. Broberg, A. M. Goscinski, Cloud computing: Principles and paradigms, Vol. 87, John Wiley & Sons, 2010.

[19] C. Wohlin, 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, 2014, p. 38.

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

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

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

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

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

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

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

[27] H. Wang, X. Zhang, Q. Yu, Integrating pomdp and sarsa λ for service composition with incomplete information, in: International Conference on Service-Oriented Computing, Springer, 2016, pp. 677–684.

[28] A. Mostafa, M. Zhang, Multi-objective service composition in uncertain environments, IEEE Transactions on Services Computing.

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

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

[31] H. B. Mahfoudh, G. D. M. Serugendo, A. Boulmier, N. Abdennadher, 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, 2018, pp. 257–273.

[32] F. Zambonelli, G. Castelli, L. Ferrari, M. Mamei, A. Rosi, G. Di Marzo, M. Risoldi, A.-E. Tchao, S. Dobson, G. Stevenson, et al., Self-aware pervasive service ecosystems, Procedia Computer Science 7 (2011) 197–199.

[33] M. Fathian, B. Amiri, A. Maroosi, Application of honey-bee mating optimization algorithm on clustering, Applied Mathematics and Computation 190 (2) (2007) 1502–1513.

[34] B. Amiri, M. Fathian, A. Maroosi, Application of shuffled frog-leaping algorithm on clustering, The International Journal of Advanced Manufacturing Technology 45 (1-2) (2009) 199–209.

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

[36] M. E. Khanouche, F. Attal, Y. Amirat, A. Chibani, M. Kerkar, Clustering-based and qos-aware services composition algorithm for ambient intelligence, Information Sciences 482 (2019) 419–439.

[37] J.-h. Zhang, 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, 2010, pp. V5–633.

[38] Q. Yu, Decision tree learning from incomplete qos to bootstrap service recommendation, in: International Conference on Web Services, IEEE, 2012, pp. 194–201.

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

[40] Z. Ye, S. Mistry, A. Bouguettaya, H. Dong, Long-term qos-aware cloud service composition using multivariate time series analysis, IEEE Transactions on Services Computing 9 (3) (2016) 382–393.

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

[42] Y. Guo, S. Wang, K.-S. Wong, M. H. Kim, Skyline service selection approach based on qos prediction, International Journal of Web and Grid Services 13 (4) (2017) 425–447.

[43] S. Borzsony, D. Kossmann, K. Stocker, The skyline operator, in: Proceedings 17th international conference on data engineering, IEEE, 2001, pp. 421–430.

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

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

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

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

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

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

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

[51] H. Kil, R. Cha, W. Nam, Transaction history-based web service composition for uncertain qos, International Journal of Web and Grid Services 12 (1) (2016) 42–62.

[52] R. Ramacher, L. Mönch, Robust multi-criteria service composition in information systems, Business & Information Systems Engineering 6 (3) (2014) 141–151.

[53] S. Wang, Y. Guo, Y. Li, C.-H. Hsu, Cultural distance for service composition in cyber–physical–social systems, Future Generation Computer Systems.

[54] G. Hofstede, Dimensionalizing cultures: The hofstede model in context, Online Readings in Psychology and Culture 2 (1) (2011) 8.

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

[56] Z. Malik, B. Medjahed, Maintaining trustworthiness of service compositions, in: International Conference on Frontiers of Information Technology, ACM, 2010, p. 23.

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

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

[59] D. Li, D. Cheung, X. Shi, V. Ng, Uncertainty reasoning based on cloud models in controllers, Computers & Mathematics with Applications 35 (3) (1998) 99–123.

[60] S. Wang, L. Huang, L. Sun, C.-H. Hsu, F. Yang, Efficient and reliable service selection for heterogeneous distributed software systems, Future Generation Computer Systems 74 (2017) 158–167.

[61] M. Alrifai, D. Skoutas, T. Risse, Selecting skyline services for qos-based web service composition, in: Proceedings of the 19th international conference on World wide web, ACM, 2010, pp. 11–20.

[62] Q. Yu, A. Bouguettaya, Computing service skyline from uncertain qows, IEEE Transactions on Services Computing 3 (1) (2010) 16–29.

[63] F. Barbon, P. Traverso, M. Pistore, M. Trainotti, Run-time monitoring of instances and classes of web service composi-

tions, in: International Conference on Web Services (ICWS'06), IEEE, 2006, pp. 63–71.

[64] R. Jurca, B. Faltings, W. Binder, Reliable qos monitoring based on client feedback, in: International Conference on World Wide Web, ACM, 2007, pp. 1003–1012.

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

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

[67] Y. Chen, S. Ying, L. Zhang, J. Wu, Exception detection for web service composition using improved bayesian network, Journal of Digital Information Management 11 (2) (2013) 109.

[68] Z. Ye, A. Bouguettaya, X. Zhou, Economic model-driven cloud service composition, ACM Transactions on Internet Technology (TOIT) 14 (2-3) (2014) 20.

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

[70] S.-Y. Hwang, H. Wang, J. Tang, J. Srivastava, A probabilistic approach to modeling and estimating the qos of web-services-based workflows, Information Sciences 177 (23) (2007) 5484–5503.

[71] S.-Y. Hwang, C.-C. Hsu, C.-H. Lee, Service selection for web services with probabilistic qos, IEEE Transactions on Services Computing 8 (3) (2015) 467–480.

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

[73] J. Wang, Exploiting mobility prediction for dependable service composition in wireless mobile ad hoc networks, IEEE Transactions on Services Computing 4 (1) (2011) 44–55.

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

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

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

[77] H. Ye, T. Li, Web service composition with uncertain qos: An iqcp model, in: CCF Conference on Computer Supported Cooperative Work and Social Computing, Springer, 2018, pp. 146–162.

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

[79] H. Mezni, M. Sellami, A negotiation-based service selection approach using swarm intelligence and kernel density estimation, Software: Practice and Experience 48 (6) (2018) 1285–1311.

[80] H. Zheng, J. Yang, W. Zhao, Qosdist: A qos probability distribution estimation tool for web service compositions, in: Asia-Pacific Services Computing Conference, IEEE, 2010, pp. 131–138.

[81] H. Zheng, J. Yang, W. Zhao, A. Bouguettaya, Qos analysis for web service compositions based on probabilistic qos, in: International Conference on Service-Oriented Computing, Springer, 2011, pp. 47–61.

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

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

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

[85] S. Rosario, A. Benveniste, S. Haar, C. Jard, Probabilistic qos and soft contracts for transaction-based web services orchestrations, IEEE Transactions on Services Computing 1 (4) (2008) 187–200.

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

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

[88] J.-S. R. Jang, C.-T. Sun, E. Mizutani, Neuro-fuzzy and soft computing-a computational approach to learning and machine intelligence [book review], IEEE Transactions on Automatic Control 42 (10) (1997) 1482–1484.

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

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

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

[92] P. Veeresh, R. P. Sam, C. S. Bindu, Fuzzy based optimal qos constraint services composition in mobile ad hoc networks, International Journal of Communication Networks and Information Security (IJCNIS) 9 (3) (2017) 491–499.

[93] A. K. Tripathy, P. K. Tripathy, Fuzzy qos requirement-aware dynamic service discovery and adaptation, Applied Soft Computing 68 (2018) 136–146.

[94] A. K. Tripathy, M. R. Patra, Service based system monitoring framework, International Journal of Computer Information Systems and Industrial Management Applications: IJCISIM 3 (2011) 924–931.

[95] S. Niu, G. Zou, Y. Gan, Y. Xiang, B. Zhang, Towards the optimality of qos-aware web service composition with uncer-

tainty, International Journal of Web and Grid Services 15 (1) (2019) 1–28.

[96] L. Zhang, H. Zou, F. Yang, A dynamic web service composition algorithm based on topsis, Journal of networks 6 (9) (2011) 1296.

[97] L. Zhang, T. Zhang, C. Zhang, Web service composition algorithm based on hybrid-qos and pair-wise comparison matrix, Journal of Information and Computational Science 9 (1) (2012) 135–142.

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

[99] X. Jian, Q. Zhu, Y. Xia, An interval-based fuzzy ranking approach for qos uncertainty-aware service composition, Optik-International Journal for Light and Electron Optics 127 (4) (2016) 2102–2110.

[100] M. Behzadian, R. B. Kazemzadeh, A. Albadvi, M. Aghdasi, Promethee: A comprehensive literature review on methodologies and applications, European journal of Operational research 200 (1) (2010) 198–215.

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

[102] M. Sugeno, Industrial applications of fuzzy control, Elsevier Science Inc., 1985.

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

[104] X. Zhao, L. Shen, X. Peng, W. Zhao, Toward sla-constrained service composition: An approach based on a fuzzy linguistic preference model and an evolutionary algorithm, Information Sciences 316 (2015) 370–396.

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

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

[107] M. Zhu, G. Fan, J. Li, H. Kuang, An approach for qos-aware service composition with graphplan and fuzzy logic, Procedia Computer Science 141 (2018) 56–63.

[108] J. Xu, L. Guo, R. Zhang, H. Hu, F. Wang, Z. Pei, Qos-aware service composition using fuzzy set theory and genetic algorithm, Wireless Personal Communications 102 (2) (2018) 1009–1028.

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

[110] A. A. P. Kazem, H. Pedram, H. Abolhassani, Bnqm: a bayesian network based qos model for grid service composition, Expert Systems with Applications 42 (20) (2015) 6828–6843.

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

[112] X. Chen, Z. Zheng, X. Liu, Z. Huang, H. Sun, Personalized qos-aware web service recommendation and visualization, IEEE Transactions on Services Computing 6 (1) (2013) 35–47.

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

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

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

[116] Z. Zheng, H. Ma, M. R. Lyu, I. King, Collaborative web service qos prediction via neighborhood integrated matrix factorization, IEEE Transactions on Services Computing 6 (3) (2012) 289–299.

[117] Z. Chen, L. Shen, F. Li, D. You, Your neighbors alleviate cold-start: On geographical neighborhood influence to collaborative web service qos prediction, Knowledge-Based Systems 138 (2017) 188–201.

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

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

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

[121] Z.-Z. Liu, D.-H. Chu, Z.-P. Jia, J.-Q. Shen, L. Wang, Two-stage approach for reliable dynamic web service composition, Knowledge-Based Systems 97 (2016) 123–143.

[122] J. Kolodner, Case-based reasoning, Morgan Kaufmann, 2014.

[123] K. Hashmi, Z. Malik, E. Najmi, A. Rezgui, Snrneg: A social network enabled negotiation service, Information Sciences 349 (2016) 248–262.

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

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

[126] T. H. Tan, M. Chen, É. André, J. Sun, Y. Liu, J. S. Dong, Automated runtime recovery for qos-based service composition, in: International Conference on World Wide Web, ACM, 2014, pp. 563–574.

[127] Y. Chen, L. Jiang, J. Zhang, X. Dong, A robust service selection method based on uncertain qos, Mathematical Problems in Engineering 2016.

[128] D. Bertsimas, M. Sim, The price of robustness, Operations research 52 (1) (2004) 35–53.

[129] A. Urbieta, A. González-Beltrán, S. B. Mokhtar, M. A. Hossain, L. Capra, Adaptive and context-aware service compo-

sition for iot-based smart cities, Future Generation Computer Systems 76 (2017) 262–274.

[130] N. Chen, N. Cardozo, S. Clarke, Goal-driven service composition in mobile and pervasive computing, IEEE Transactions on Services Computing 11 (1) (2018) 49–62.

[131] A. N. Toosi, R. N. Calheiros, R. K. Thulasiram, R. Buyya, 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, 2011, pp. 279–287.

[132] S. A. Tafsiri, S. Yousefi, Combinatorial double auction-based resource allocation mechanism in cloud computing market, Journal of Systems and Software 137 (2018) 322–334.

[133] Y. Sharma, W. Si, D. Sun, B. Javadi, Failure-aware energy-efficient vm consolidation in cloud computing systems, Future Generation Computer Systems 94 (2019) 620–633.

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

[135] X. Zhang, T. Wu, M. Chen, T. Wei, J. Zhou, S. Hu, R. Buyya, Energy-aware virtual machine allocation for cloud with resource reservation, Journal of Systems and Software 147 (2019) 147–161.

[136] K. Gai, M. Qiu, H. Zhao, Energy-aware task assignment for mobile cyber-enabled applications in heterogeneous cloud computing, Journal of Parallel and Distributed Computing 111 (2018) 126–135.

[137] H. Tabassum, M. Salehi, E. Hossain, Fundamentals of mobility-aware performance characterization of cellular networks: A tutorial, IEEE Communications Surveys & Tutorials.

[138] G. White, A. Palade, C. Cabrera, S. Clarke, Iotpredict: collaborative qos prediction in iot, in: IEEE International Conference on Pervasive Computing and Communications (PerCom), IEEE, 2018, pp. 1–10.

[139] C. Huang, H. Cai, Y. Li, J. Du, F. Bu, L. Jiang, A process mining based service composition approach for mobile information systems, Mobile Information Systems 2017.

[140] C. Baudrit, D. Dubois, D. Guyonnet, Joint propagation and exploitation of probabilistic and possibilistic information in risk assessment, IEEE Transactions on Fuzzy Systems 14 (5) (2006) 593–608.

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

[142] M. Ghazanfari, S. Alizadeh, M. Fathian, D. E. Koulouriotis, Comparing simulated annealing and genetic algorithm in learning fcm, Applied Mathematics and Computation 192 (1) (2007) 56–68.

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

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

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

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

[147] N. Chen, Y. Yang, T. Zhang, M.-T. Zhou, X. Luo, J. K. Zao, Fog as a service technology, IEEE Communications Magazine 56 (11) (2018) 95–101.

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

[149] E. Al-Masri, Q. H. Mahmoud, Qos-based discovery and ranking of web services, in: 2007 16th International Conference on Computer Communications and Networks, IEEE, 2007, pp. 529–534.

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

[151] Z. Zheng, Y. Zhang, M. R. Lyu, Investigating qos of real-world web services, IEEE Transactions on Services Computing 7 (1) (2014) 32–39.

[152] Z. Zheng, Y. Zhang, M. R. Lyu, Distributed qos evaluation for real-world web services, in: International Conference on Web Services, IEEE, 2010, pp. 83–90.

[153] OWLS-TC4, Owls-tc version 4.0, `http://projects.semwebcentral.org/projects/owls-tc/`, accessed: 2019-05-16.

[154] W. Jiang, D. Lee, S. Hu, Large-scale longitudinal analysis of soap-based and restful web services, in: International Conference on Web Services, IEEE, 2012, pp. 218–225.

[155] S. V. Gogouvitis, H. Mueller, S. Premnadh, A. Seitz, B. Bruegge, Seamless computing in industrial systems using container orchestration, Future Generation Computer Systems.

[156] P. Varshney, Y. Simmhan, Characterizing application scheduling on edge, fog, and cloud computing resources, Software: Practice and Experience.

[157] C. Bu, X. Wang, H. Cheng, M. Huang, K. Li, Routing as a service (raas): An open framework for customizing routing services, Journal of Network and Computer Applications 125 (2019) 130–145.

[158] A. Aydeger, N. Saputro, K. Akkaya, A moving target defense and network forensics framework for isp networks using sdn and nfv, Future Generation Computer Systems 94 (2019) 496–509.

[159] M. S. Bonfim, K. L. Dias, S. F. Fernandes, Integrated nfv/sdn architectures: A systematic literature review, ACM Computing Surveys (CSUR) 51 (6) (2019) 114.

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

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

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

[163] T. Chen, R. Bahsoon, X. Yao, A survey and taxonomy of self-aware and self-adaptive cloud autoscaling systems, ACM Computing Surveys (CSUR) 51 (3) (2018) 61.

[164] C. Jatoth, G. Gangadharan, R. Buyya, Optimal fitness aware cloud service composition using an adaptive genotypes evolution based genetic algorithm, Future Generation Computer Systems 94 (2019) 185–198.

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

[166] C. Zhang, P. Patras, H. Haddadi, Deep learning in mobile and wireless networking: A survey, IEEE Communications Surveys & Tutorials.

[167] S. K. Moghaddam, R. Buyya, K. Ramamohanarao, Acas: An anomaly-based cause aware auto-scaling framework for clouds, Journal of Parallel and Distributed Computing 126 (2019) 107–120.

[168] S. Kardani-Moghaddam, R. Buyya, K. Ramamohanarao, Performance anomaly detection using isolation-trees in heterogeneous workloads of web applications in computing clouds, Concurrency and Computation: Practice and Experience (2019) e5306.

[169] M. Razian, M. Fathian, R. Buyya, Arc: Anomaly-aware robust cloud-integrated iot service composition based on uncertainty in advertised quality of service values, Journal of Systems and Software 164 (2020) 110557.

[170] R. Chen, J. Guo, F. Bao, Trust management for service composition in soa-based iot systems, in: IEEE Wireless Communications and Networking Conference, IEEE, 2014, pp. 3444–3449.

[171] J. Guo, Trust-based service management of internet of things systems and its applications, Ph.D. thesis, Virginia Tech (2018).

[172] L. Wu, C. Quan, C. Li, Q. Wang, B. Zheng, X. Luo, A context-aware user-item representation learning for item recommendation, ACM Transactions on Information Systems (TOIS) 37 (2) (2019) 22.

[173] F. Xue, X. He, X. Wang, J. Xu, K. Liu, R. Hong, Deep item-based collaborative filtering for top-n recommendation, ACM Transactions on Information Systems (TOIS) 37 (3) (2019) 33.

[174] D. Lian, K. Zheng, Y. Ge, L. Cao, E. Chen, X. Xie, Geomf++: Scalable location recommendation via joint geographical modeling and matrix factorization, ACM Transactions on Information Systems (TOIS) 36 (3) (2018) 33.

[175] G. Javadzadeh, A. M. Rahmani, Fog computing applications in smart cities: A systematic survey, Wireless Networks 26 (2) (2020) 1433–1457.

[176] P. Asghari, A. M. Rahmani, H. H. S. Javadi, Service composition approaches in iot: A systematic review, Journal of Network and Computer Applications 120 (2018) 61–77.

[177] R. Mahmud, R. Kotagiri, R. Buyya, Fog computing: A taxonomy, survey and future directions, in: Internet of everything, Springer, 2018, pp. 103–130.

[178] A. M. Rahmani, T. N. Gia, B. Negash, A. Anzanpour, I. Azimi, M. Jiang, P. Liljeberg, Exploiting smart e-health gateways at the edge of healthcare internet-of-things: A fog computing approach, Future Generation Computer Systems 78 (2018) 641–658.