Hybrid Metaheuristics for QoS-Aware Service Composition: A Systematic Mapping Study

HADI NAGHAVIPOUR1, (Member, IEEE), TEY KOK SOON1, (Member, IEEE), MOHD YAMANI IDNA BIN IDRIS1, (Member, IEEE), MORTEZA NAMVAR2, ROSLI BIN SALLEH1, (Member, IEEE), AND ABDULLAH GANI3, (Senior Member, IEEE)

1Department of Computer System and Technology, Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur 50603, Malaysia
2Business School, The University of Queensland, Brisbane, QLD 4000, Australia
3Faculty of Computing and Informatics, University Malaysia Sabah, Labuan 88400, Malaysia

Corresponding authors: Abdullah Gani (abdullahgani@ums.edu.my), Hadi Naghavipour (hadi.napu@gmail.com), and Mohd Yamani Idna Bin Idris (yamani@um.edu.my)

This work was supported in part by the Malaysia Ministry of Higher Education, Fundamental Research Grant Scheme (FRGS) under Grant FP005-2021.

ABSTRACT With the advent of Service-Oriented Architecture (SOA), services can be registered, invoked, and combined with their identical Quality of Services (QoS) attributes to create a new value-added application that fulfils user requirements. Efficient QoS-aware service composition has been a challenging task in cloud computing. This challenge becomes more formidable in emerging resource-constrained computing paradigms, such as the Internet of Things and Fog. Service composition has regarded as a multi-objective combinatorial optimization problem that falls in the category of NP-hard. Historically, the proliferation of services added to problem complexity and navigated solutions from exact (none-heuristics) approaches to near-optimal heuristics and metaheuristics. Although metaheuristics have fulfilled some expectations, the quest for finding a high-quality, near-optimal solution has led researchers to devise hybrid methods. As a result, research on service composition shifts towards the hybridization of metaheuristics. Hybrid metaheuristics have been promising efforts to transcend the boundaries of metaheuristics by leveraging the strength of complementary methods to overcome base algorithm shortcomings. Despite the significance and frontier position of hybrid metaheuristics, to the best of our knowledge, there is no systematic research and survey in this field with a particular focus on strategies to hybridize traditional metaheuristics. This study’s core contribution is to infer a framework for hybridization strategies by conducting a mapping study that analyses 71 papers between 2008 and 2020. Moreover, it provides a panoramic view of hybrid methods and their experiment setting in respect to the problem domain as the main outcome of this mapping study. Finally, research trends, directions and challenges are discussed to benefit future endeavours.

INDEX TERMS Service computing, cloud computing, quality of service, service composition, metaheuristics, hybrid metaheuristics, mapping study.

I. INTRODUCTION Service computing represents a broad computational framework that aims to provide architectures, tools and techniques to support services. Services have resulted from the historical evolution of computing from the early age of representing information to reasoning and, finally, knowledge creation. Services were arising from the notion of adding action to knowledge [1]. Service computing has been a driving force for an exuberant growth in Service Oriented Computing (SOC) [2] as a technology framework for defining, registering and calling the services [3]. These emerging trends give rise to cloud computing as a revolutionary delivery model for IT resources. Cloud computing reshaped the IT industry by pioneering a delivery model in which elastic services, computing resources, and data have distributed over computer networks. Today, governmental institutions, academia, and business enterprises can benefit excessively from cloud technologies advantages in accessibility, reliability, and scalability, all with an affordable price tag. Cloud computing’s advent sparked a paradigm shift that, in some accounts [4], [5], is regarded as the fifth utility after water, electricity, telephone, and gas.

The associate editor coordinating the review of this manuscript and approving it for publication was Barbara Masini.
The Internet of Things (IoT) also has become an avenue for the application of service composition. With billions of objects connected daily, the IoT is truly revolutionary technology that enabling unprecedented human connections in both the virtual and real worlds [6]. The IoT system requires to compute complex sensor and actuator composition where hardware dimensions will add to the complexity of the problem. The hardware aspects of the problem [7] which is selecting proper hardware configuration from diverse pool of devices such as sensors and actuator form a highly complex design problem.

The practice of servitization is gaining momentum. Servitization is the transformation of a product or system into a service-oriented model. The IoT service will lead to a more widespread servitization. IoT services can turn existing devices into value-added services. In this context, the notions of service selection, discovery, and composition become critical for IoT environment in order to create new value-added services [8]. As a result, IoT devices can publish their features, and QoS attributes to be searched or discovered by existing devices. In context of smart transportation, QoS-aware service composition has used to avoid collision by making real-time decisions in Unmanned Aerial Vehicle (UAV) swarms that are highly latency sensitive [9].

One of many possible applications of efficient composition methods is selections and identification platforms that can point out the optimal solution based on quality attributes of the data repository and consumer requirements. As it has illustrated in Fig. 1, service composition is an NP-hard multi-objective combinatorial optimization problem [10]. In this context, service composition is represented by aggregation of QoS value in a composition [11]. Hence, service selection is critical for composing multiple atomic services with the same functionality but different quality in a fully automated or semi-automated composition strategy that can acquire optimal QoS values. The primary goal of QoS-aware service composition is to find an optimal aggregation of QoS value; however, there are challenges associated with composition execution. The fundamental issue is to satisfy user requirements with a wide range of functional or non-functional quality attributes to gain user satisfaction regardless of the bulkiness of the service repository. Even in the simplest definition, QoS-aware service composition is a multi-objective combinatorial optimization problem that is considered an NP-hard problem [12].

The problem complexity is attributed to the need for optimizing conflicting QoS attributes simultaneously. Some earlier studies attempt to solve the problem using a group of exact methods (non-heuristics) in a relatively limited service repository.

A. From None-heuristic to Metaheuristic

Multidimensional decision criteria [13], [14] and Multi-dimension Multi-choice 0-1 Knapsack Problem (MMKP) [15], [16] broadly used to model this problem. Moreover, the researcher attempted to apply exact methods such as integer linear programming techniques [17], [18], mixed integer programming [13], [19] and dynamic programming [20]. In addition, techniques like greedy search [21], approaches such as divide-and-conquer [19], a system called RuGQoS [22] and an execution planned (BB4EPS) [23] used to outline the exact optimal solution for a relatively constrained service repository.

Although exact methods have been effective, they become insufficient when the service repository expands due to an unprecedented increase in time complexity. Researchers also propose heuristic approaches to perform service composition in a reasonable time budget by sacrificing exact optimality to attain a near-optimal solution. Despite increasing interest in enforcement learning [24]–[28], heuristic search techniques [29]–[34], A∗ algorithm [35], [36], hill-climbing [37], BF algorithm [38] and Dijkstra [39], [40] were concurrent methods for heuristic-based service composition in current literature. In a graph, [41], or tree representation [42], heuristic search techniques are also applied to find the optimal path. Moreover, skyline discovery is also used in several works [43]–[45].

Since heuristics methods are problem-dependent, researchers tried to adopt a problem-independent strategy that applies to wide-ranging problems. These efforts led to applying metaheuristic algorithms as a universal high-level search strategy. Our survey acknowledges the excessive use of nature metaheuristic optimization methods. Applications of trajectory-based metaheuristics such as tabu search, [46], and simulated annealing [47], [48] encountered in existing literature. However, population-based metaheuristics formed the majority of proposed solutions.

Canfora et al. [49] has been pioneer to adopt genetic algorithm to tackle the problem. Researchers followed suit by proposing variants of genetic algorithms [50]–[57], and genetic programming [58]–[63].

In chronological order of concurrence in existing literature, swarm intelligent techniques including, Particle Swarm Optimization (PSO) [64]–[68], Ant Colony Optimization (ACO) [69]–[71], Artificial Bee Colony (ABC) [72]–[74], Artificial Immune System (AIS) [72]–[74], Cuckoo Search (CS) [75], [76], Harmony Search (HS) [76] and multi-objective optimization techniques [77] applied to resolve the problem.

B. Hybrid Metaheuristic

Our survey could identify 202 methods for this problem between 2003 and 2020. In this study, the more we delved into the existing metaheuristics, the more it became evident that many, if not most of them, do not entirely adhere to one traditional metaheuristic rather than a combination of algorithms [78]. Our survey indicates that heuristic approach made 14 %, of solutions, metaheuristic 40 %, and hybrid metaheuristic 36 % of solutions. In the present study, we observed that hybrid metaheuristic approaches gained ground to address this optimization problem, as it is shown in Fig. 2.
Recent application of hybrid metaheuristic has shown promising results [79]–[81]. One of the most sought-after strategies for enhancing metaheuristic efficiency has been hybridization with other techniques to attain a better search mechanism with minimal time complexity.

It is a quite challenging task to select a hybrid metaheuristic from expanding number of intelligent computational methods. To clarify, as theoretical scenario, assuming that there are $n$ algorithms, if one chooses $2 \leq k \leq n$, the possible hybrid algorithm denoted with $C$ as follows:

$$C_k^n = \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

A plethora of empirical studies indicates that hybrid metaheuristics outperform traditional ones by minimizing shortcomings and maximizing the strengths. With that in mind, the challenge here is to find optimal solutions that deliver results. Moreover, this imperative goal is achievable by conducting a holistic empirical study for an intended hybrid model. Since the implementation of any potential hybrid method has been fraught with technical difficulty, a prior knowledge of existing practice will be highly beneficial for future efforts.

C. SCOPE AND CONTRIBUTIONS

We believe it would be unreliable to complete a service composition survey that accurately represents all viewpoints due to extensive research on this domain. Despite the rise of hybrid metaheuristics that has become almost equal to the adoption of solo metaheuristics, the hybrid methods represent the frontier approach for service composition. This development is attributed to the proliferation of services that may fail the most efficient metaheuristics. In addition, what marks an algorithm design fulfilling is not merely achieving superiority on identical solo performance metrics. In contrast, it is a trade-off between algorithm performance variables. Therefore, it has become a daunting task for practitioners to find the most appropriate methods without prior structured knowledge. It is also difficult for researchers to assess research gaps and future research trends. To the best of our knowledge, an exclusive investigation of hybrid metaheuristics for service composition does not exist in the current literature. Therefore, this study contributes on the following fronts:

- This study infers current hybrid strategies concerning service composition problem formulation and experiment setup.
- This paper also propose a taxonomical analysis of hybrid metaheuristics for service composition.
- Finally, this mapping study discusses the research trends, issues, challenges and future directions.
The rest of the paper is organized as follows. A brief background on the problem in various forms of problem definition is brought in Section II. A step-by-step explanation of the research methodology is discussed in Section III. In follow, classifications of hybrid metaheuristics are described in Section IV. The mapping study results are analyzed in Section V. Finally, Research challenges and future direction are addressed in chapter VI followed by concluding remarks in Section VII.

II. BACKGROUND

Service-oriented computing gives rise to the need for cloud interoperation. Cloud interoperation is an essential operation in which services communicate or combine to create a new value-added application that fulfills user requirements. Interoperability and portability are manifested by Buyya et al. [5] as one of cloud computing’s research challenges. The challenge of cloud interoperability and portability is twofold. Firstly, it is an issue of integration. Secondly, it is a problem of service discovery and selection. As a result, various service integration, open middleware standards such as WS-BPEL [82] WSDL, and SOAP [83] were devised to facilitate service communication.

QoS-aware service composition process is attributed to QoS properties, composition strategy, and fitness approach. The problem modeling will lead to the formation of a combinatorial multi-objective optimization that falls under the umbrella of NP-hard problems. Service composition is the process of invoking services from the service repository (Si) and producing composition (Do) where any generated composition (Do') with service instances (Si') which, cover the following subset of Do' ⊆ Do and Si' ⊆ Si. Geyik et al. [84] justified service composition from a computational complexity perspective and classified the problem under NP-hard by referring to a set cover problem, a well-known NP-complete.

Service instances usually are advertised by multiple QoS values, which has known as QoS property. QoS properties are a tangible representation of services that can match user expectations. A recent investigation [85] revealed that response time (%58), cost (%34) and price (%28) were the most commonly used negative properties, while availability (%44), reliability (%42), throughput (%20), and reputation (%16) reported favorable properties that are expected to be maximized. Furthermore, authors in [86] classified QoS properties into two binary categories of user-independent and dependent. Wang et al. [87] draw a line of distinction between QoS properties by categorizing them into quantitative such as time or qualitative such as reputation. Although each of these works has its own merits, the widely accepted approach has been classifying properties into functional or non-functional categories [88]. While functional properties describe system behavior, non-functional properties demonstrate system operation.

A. COMPOSITION MODELS

The composition process is akin to the workflow management systems [89]. A planner provides an abstract workflow description in a process management environment to identify, select, and integrate services according to the abstract workflow. The result of the workflow execution considered the service composition. An alternative to this approach is discovering service connections, transiting from the initial state to the desired one.

The Fig. 3 shows composition modes of sequential, loop, conditional, and parallel [90]. However, many articles investigated service combinations in sequential mode using strategies that convert others to sequential mode. Due to QoS properties’ conflicting nature, service composition is deemed a multi-objective optimization problem. However, it can also be treated as a single-objective objective when an aggregate function is employed for several services with multiple quality attributes. QoS aggregation functions aimed to calculate the quality for different composition modes by a computational scheme such as a Simple Additive Weight.
fitness function (SAW). The key idea here is to aggregate QoS attribute values into a single score used for optimization computation. These formulas are only suitable for single-value QoS. However, multi-objective optimization methods employ an independent fitness function to optimize conflicting attributes in a Pareto front of solutions. Herein, probabilities of the different construct choice denoted by $p_1, \ldots, p_d$ where
\[
\sum_{i=1}^{d} p_i = 1
\]
holds. Moreover, $\mathcal{I}$ represents the average number of iterations in the iterative construct. Table 1 list four popular QoS attributes, Reliability (R), Availability (A), Cost (C), Time (T), and for the n service instances. The aggregation function is defined according to composition modes that orchestrate a QoS-aware service composition workflow.

### B. PROBLEM FORMALIZATION
QoS-aware service composition in straightforward expression is the problem of selecting an optimal composition from the pool of candidate services denoted by $S_{i=(1,2,\ldots,n)}$. It is expected to execute a workflow consists of the given task set of $T = \{T_1, T_2, \ldots, T_n\}$, which satisfy user requirements.

The absolute majority of researchers adopted vector and graph representation. However, in a fully automated approach, the tree-based presentation was employed. When the composition order was imperative, a permutation-based definition has proposed in problem formalization.

### C. GRAPH-BASED REPRESENTATION
There are several ways in which service composition can define. A generic approach has been representing the problem as multi-stage graph planning. The first stage is all about labelling the graph. Next, the labelled graph transfers to a weighted graph and finally, service composition will be interpreted to finding the shortest path with optimized cost. This definition has inspired by the directed acyclic graph, denoted by $G = (S_i, P_i)$, Whereas S stands for services and P for service attributes. Moreover, $P_i$ is compromised of service parameters, and $S_i$ contains services whose inputs rendered the service from the previous layer.

### D. QoS VECTOR-BASED FORMULATION
A more popular alternative to this approach is to represent services (S) and user requirements with vectors. Service composition (SC) in vector representation has been defined by selecting a set of service vectors that fits user quality requirements. In this definition, given a request denoted by R, the objective of service composition is to find a functionally correct composite service that meets the user requirements $R_S$, by optimizing the quality attributes as follows:
\[
SC = \{ R_{S,1}, \ldots, R_{S,n} \}
\]

On this basis, Eq. 1 shows [91] how the QoS value normalized into a value between 0 to 1 where positive attribute differs from negative values. QoS(S) =
\[
\begin{cases}
\frac{QoS(S) - Q_{min}}{Q_{max} - Q_{min}} & \text{if } Q \text{ is positive and } Q_{max} - Q_{min} \neq 0, \\
\frac{Q_{max} - QoS(S)}{Q_{max} - Q_{min}} & \text{if } Q \text{ is negative and } Q_{max} - Q_{min} \neq 0, \\
1 & \text{otherwise.}
\end{cases}
\]

Following Eq. 2 is a optimisation of QoS-aware service composition in which $j$ is the number of quality attributes and $n$ represent quality attributes counts where $\sum_{n=1}^{n} W_n = 1$ holds.

Optimise $QoS(S) = \sum_{n=1}^{j} W_n \cdot \hat{Q}_n(S)$

### E. RELATED WORKS
QoS-aware service composition has been researched excessively over the last two decades. As a result, various review works were presented. Here, we have analyzed related survey published since 2014. Jula et al. [103] have made a systematic review of cloud service composition based on 34 articles published from 2009 to 2013. This review has classified selected methods into the following categories: graph-based algorithms, combinatorial algorithms, machine-based, structures, and frameworks-based. This review pinpoint the importance of real-time service composition. Abdelmaboud et al. [102] also provide a classification of techniques for service composition with a heavy emphasis on the type of QoS attributes and the cloud layer solution that solution was implemented. This study scopes four years and did not provide a taxonomical analysis of solutions. Chandrashekar et al. [101] investigate a large body of work between 2005 and 2015 to analyze 84 selected computational intelligence-based for QoS-aware service compositions. Moreover, the authors introduced a classification that divided the proposed method into three categories: none heuristic, heuristics, and metaheuristics. This classification is influenced profoundly by a direct association between time complexity and service repository size. The concluding remarks in this work encourage further study on hybrid solutions combining data mining techniques and optimization methods. A detailed mapping study conducted by She et al. [96] places composition methods under the same classification [101]. Authors in [99] and [98] investigate the service composition for the Internet of things and cloud environment through the context of nature-inspired metaheuristics. These early survey focus has been on the application environment or taxonomical analysis of approaches for the problem.

Furthermore, Vakili and Navimipour [100] investigated this problem thoroughly in cloud environments and described the most popular cloud service composition techniques in three main categories: heuristic-based, framework-based, and agent-based. This research highlights the importance of the
TABLE 2. Analysis of existing surveys in current literature in respect to their proposed taxonomy and domain focus.

<table>
<thead>
<tr>
<th>Reference</th>
<th>Year</th>
<th>Method</th>
<th>Period</th>
<th>Publisher</th>
<th>Taxonomy or Main Focus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Masdari et al. [92]</td>
<td>2021</td>
<td>SLR</td>
<td>2010 to 2019</td>
<td>Springer</td>
<td>Classification of the bio-inspired methods for service composition</td>
</tr>
<tr>
<td>Alinani et al. [93]</td>
<td>2020</td>
<td>SLR</td>
<td>2013 to 2020</td>
<td>IEEE</td>
<td>Classification of QoS attributes and approaches focusing on cloud manufacturing</td>
</tr>
<tr>
<td>da Silva et al. [94]</td>
<td>2020</td>
<td>Survey</td>
<td>2005 to 2019</td>
<td>IEEE</td>
<td>Taxonomy of evolutionary computational methods from a technical perspective</td>
</tr>
<tr>
<td>Hayyolalam et al. [95]</td>
<td>2019</td>
<td>SLR</td>
<td>2013 to 2018</td>
<td>Springer</td>
<td>Binary taxonomy of heuristic-based and non-heuristic</td>
</tr>
<tr>
<td>Zie et al. [96]</td>
<td>2019</td>
<td>SMS</td>
<td>2009 to 2018</td>
<td>Elsevier</td>
<td>Taxonomy: Exacts (none-heuristics), heuristics and metaheuristics</td>
</tr>
<tr>
<td>Asghari et al. [99]</td>
<td>2018</td>
<td>SLR</td>
<td>2012 to 2017</td>
<td>Elsevier</td>
<td>Classification of service composition approaches for Internet Of Things</td>
</tr>
<tr>
<td>Abdellazizouad et al. [102]</td>
<td>2015</td>
<td>SMS</td>
<td>2008 to 2012</td>
<td>Elsevier</td>
<td>Taxonomy of service composition approaches in cloud environments</td>
</tr>
<tr>
<td>Jula et al. [103]</td>
<td>2014</td>
<td>SLR</td>
<td>2009 to 2013</td>
<td>Elsevier</td>
<td>Classification of cloud service composition approaches</td>
</tr>
</tbody>
</table>

real-world implementation of proposed methods. In this study, 105 articles published from 2009 to 2018 were investigated to create an insight into various aspects of service composition in cloud environments. The results of this analysis indicate that 66% of studies focus on the SaaS layer on clouds, 30% support multi-objective optimization, and response time has been the most used QoS attribute. Cloud manufacturing (CMfg) is a recent paradigm that emerges by efficiently integrating distributed manufacturing resources. On this notion, service composition and optimal selection play a central role in the design process of modern cloud manufacturing. Hayyolalam et al. [95] propose a binary taxonomy of service composition techniques for this domain where solutions are classified into heuristic and none-heuristics approaches. Moreover, Hayyolalam and Kazem [97] conducted a systematic literature review between 2013 to 2019 concerning the service composition domain and investigated computational intelligence techniques for cloud manufacturing. The overwhelming application of evolutionary computing techniques, particularly genetic algorithms, is asserted in most surveys. In this context, da Silva et al. [94] surveyed computation techniques evolutionarily from an in-depth technical lense. Finally, Masdari et al. [92] tried to investigate bio-inspired service composition schemes and mechanisms. However, the authors did not provide holistic, inclusive taxonomy for the surveyed techniques as detailed in Fig. 4 and Table 2 in respect to the advent of hybrid metaheuristics in current literature.

F. MOTIVATIONS OF THIS STUDY

Despite existing surveys merits, their taxonomical analysis does not represent the historical evolution of solutions for service composition concerning the latest development in the fields. Therefore, the imperfect classification of solutions has been one of the motivating factors in conducting this study. In spite of early-stage reviews that are quite outdated, most surveys over the last five years place hybrid metaheuristics under the category of metaheuristics. Our position is not merely subscribing to a specific taxonomy but highlighting frontier solutions ignored in current literature. An in-depth study of the hybrid approach provides insight into metaheuristics’ shortcomings as base algorithms. Moreover, it explains the motivation and ramifications of proposed solutions to overcome the base algorithm pitfalls concerning various composition scenarios. Nonetheless, the existing surveys have failed to investigate the hybrid metaheuristics architecture on various applications and environments.
III. RESEARCH METHODOLOGY

The number of mapping studies is rising due to growing interest in the methodology. Systematic literature reviews (SLR) are mainly employed to answer the specific research question, while the mapping study (MS) deem to be a suitable strategy to review the broader topic in order to structure the research domain [104]. Despite differences, both methodologies emphasize a need for a systematic approach rather than conducting intuitive research. Petersen et al. [105] suggest that mapping study act as a complementary study to systematic literature reviews. Moreover, Kitchenham et al. [106], [107] argue that a mapping study can mark an initial point for a series of future research endeavors.

Finding research gaps [108] is one of the prominent goals of the mapping study. The promising results delivered by hybrid metaheuristics necessitate a detailed investigation of existing hybridized metaheuristics. In order to achieve this goal, this study has adopts a fusion of research methodologies according to the flowchart illustrated in Fig. 5. The overall framework was designed according to the guideline developed by Petersen et al. [104].

First of all, we have selected 71 articles for further investigation. Then, we conducted a systematic review [109] for selected hybrid methods to extract data.

The extracted data used to structure the research domain according to mapping study guideline [104]. The bibliometric analysis of selected studies is presented to provide a detailed insight into existing works.

1) RESEARCH QUESTIONS

According to PICO criteria [110], which stand for review through the population, intervention, comparison, and outcome, research questions was designed as follows:

- RQ1: What are the current hybrid metaheuristic techniques for QoS service composition in cloud environments?
- RQ2: What has been the researcher’s motivations to propose a specific hybrid method?
- RQ3: What are the composition approach, hybridization strategy, and experimental setup?
- RQ4: What are the noticeable trends and research future direction?

A. SEARCH STRATEGY AND DATA SOURCE

Search strategy refers to the selection process involving identifying search strings, search scope, and data sources that heavily influence the results. This paper applied a five-stage study selection process over the following data sources to extract data.

- IEEE explorer - ieeexplore.ieee.org
- Google scholar - scholar.google.com
- SpringerLink - link.springer.com
- ACM Digital Library - dl.acm.org
- Science Direct – www.sciencedirect.com

1) APPLYING SEARCH STRINGS

The search strings selected according to PICO criteria (population, intervention, comparison, outcomes) [110] aimed to result in a comprehensive search as follows: : (("quality of service" OR "QoS" OR "QoS-aware") AND ("Service composition" OR "Web service composition" OR "Service combination") AND ("metaheuristics" OR "Survey" OR "Hybrid" OR "Algorithm" OR "Review")). We purposely added review and survey as a strategy to aggregate a pool of related studies by synthesizing references in the previous survey. Our search resulted in over 1660 articles on the primary collection. Next, we limited the search space for the paper published between 2008 and 2020.

2) INCLUSION AND EXCURSION CRITERIA

After limiting the studies to a period, the obtained studies were filtered using the inclusion and exclusion criteria defined earlier. The inclusion criteria are described in following:

- Studies that are available in English.
- Studies presented at international conferences or publications in peer-reviewed journals.
- Studies devoted to resolving the problem of QoS-aware cloud service composition.
- Studies that address service composition in clouds or other environments such as manufacturing.
The exclusion criteria are as follows:

- Studies that are not available in full-text version.
- Studies that did not consider as a peer reviews article.
- Studies that did not consider the quality of services in composition.
- Studies that do not explain a hybrid metaheuristic for solving the problem.

This process resulted in a primary pool of 765 articles, including the studies extracted from similar reviews and surveys on QoS-aware service composition. A second assessment was conducted according to inclusion and exclusion criteria to narrow down the selection pool, which led to a selection of 620 articles by removing the repeated articles and irrelevant.

3) QUICK SCREENING AND FULL-TEXT READING
A quick screening process, including the skim and scamming technique employed to expedite the search process to find relevant articles by looking at keywords, titles, and abstracts. As a result, 202 method articles covering exacts, heuristic, and metaheuristics (Including hybrid-metaheuristics) methods were selected for full-text reading.

4) SNOWBALLING AND QUALITY ASSESSMENT
After full-text reading, an initial set of 66 hybrid methods out of 620 filtered studies has become candidates for further investigation. Moreover, backward and forward [111] done to ensure no imperative study missed. Finally, inspired by the quality evaluation checklist proposed in [112], quality assessment was performed to ensure that the selected articles carry a minimum quality threshold of at least three factors out of four criteria according to the following criteria:

1) Relevancy to research objectives and questions.
2) Presence of adequate information for data extraction.
3) Data source validation and journal impact factor.
4) Article contribution toward research fundamental objective.

As a result of the above circumstances, 71 final articles were selected to serve this study’s purpose.

5) DATA EXTRACTION
In order to extract data, a systematic review of the final selection was conducted to outline the researcher’s motivations to propose a specific hybridization strategy. Having reviewed related work, a detailed investigation of hybrid metaheuristics presented as the main body of the review.

The following data were extracted from the 71 selected articles to address the research question after reviewing primary and secondary studies as follows: (1) Title (2) Authors names (3) Publication years (4) Publication venue and their quality index (6) Base algorithm and hybrid strategy (7) Composition strategy (8) Fitness strategy and approach (9) and metrics such as optimization mode, datasets type, and experiment settings. The data extraction performed by the first author were reviewed and agreed upon by co-authors to ensure acceptable data accuracy.

![Bibliometric Analysis](image_url)
![“Search Space”](image_url)
![“Hybrid Approach”](image_url)

**FIGURE 6.** Bibliometric mapping and terminology co-occurrence.
B. BIBLIOMETRIC ANALYSIS

In this survey, visual text mining is used to back up the arguments by using VOSviewer [113], a research tool devised for bibliometric mapping [114]. As a result, terms co-occurrence for titles and abstracts of selected articles demonstrate in a network association map demonstrated in above Fig. 6. In addition, the following terms, “hybrid approach” and “search space,” were highlighted due to their high relevancy to this review.

In-text analysis of titles and abstract, the threshold has been set to 6 for minimum co-occurrences of a term in 1429 words, in which 56 terms meet the threshold requirements. Table 3 shows six highly cited authors as a result of a setting where co-occurrence thresholds adjusted to five among 196 authors.

TABLE 3. Bibliometric analysis for co-authorships.

<table>
<thead>
<tr>
<th>Author</th>
<th>Documents</th>
<th>Total Link Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>ma, hui</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>chifu, viorica rozina</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>pop, cristina bianca</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>salomie, ian</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>zhang, mengie</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>jatoth, Chandrashekar</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>

C. THREATS TO VALIDITY

The most severe threats to the validity of this study are selection bias and data extraction inaccuracies. The term “selection bias” refers to an article that was incorrectly included or excluded. To address this issue, we used EndNote X9 to organize and eliminate duplicates from the literature. Additionally, some risks stem from the selection process, as each researcher’s interpretation of inclusion and exclusion criteria varies. The first author’s data extractions were reviewed and agreed upon by co-authors to mitigate this risk.

D. LIMITATION

One limitation in literature review works reported by Kitchenham and Brereton [108] as assessment bias for extracted data based on author perspective. Although avoiding personal bias is not possible ideally, the selected article only agreed to be included upon the approval of all co-authors to enhance the reliability of the study selection. Another limitation is attributed to search string constraints that may cause some articles to be ignored within the search scope and excluded in the search process. In order to mitigate the risks and address these limitations, we used the guideline proposed by Blake and Nowlan et al. [115] in which provides a strategy to evaluate search strings in the planning phase.

IV. CLASSIFICATION OF HYBRID METAHEURISTIC

This section presents a comprehensive classification of hybrid metaheuristics to reveal insight into the problem domain, including parameters such as hybridization strategy, optimization mode, composition strategy, and experimental setting, as summarized in Fig. 7 and Fig. 8 respectively.

Canfora et al. [49] marked the early application of metaheuristic by proposing a genetic algorithm for tackling service composition. Researchers pursued this work in a continuous quest to achieve an optimal or near-optimal solution.

Hybrid Genetic algorithm-HGA has been the most popular hybrid approach. Ye and Mounla [116] developed a combination of case-based reasoning (CBR), integer programming, and genetic algorithm to reduce time complexity by reusing the execution path. Ma and Dong [117] applied a genetic algorithm with an aggregate preference function obtained by linear programming techniques. Ai and Tang [118] developed a repaired genetic algorithm that incorporated a guided heuristic search technique called minimal-conflict hill-climbing repair. Moreover, Ma and Zhang [119] introduced a fitness function with penalty character while the relation matrix coding scheme used to overcome the slow convergence problem. A combination of genetic algorithms with trajectory-based metaheuristic such as tabu search [120] and simulated annealing [121] are proposed to leveraged the memory in the exploration phase to enhance the performance. Bao et al. [122] adopt an orthogonal design in population generation and crossover operation. Seghir and Khababa [123] proposed a local search strategy with a fly fruit optimization to generate a high-quality initial population. Composition accuracy is compromised or neglected in many similar works. Nevertheless, in a premutation based service composition in cloud’s Infrastructure as a Service (IaaS), Mistry et al. [79] proposed a hybrid adaptive genetic algorithm with novel QoS modelling. Furthermore, Faruk et al. [124] presented a hybrid GA-PSO as an improvement strategy to acquire convergence intensity.

A large body of evidence suggests that operator modification in metaheuristics has been the most used approach to improving algorithm performance. In this context, Feng and LEI [125] used quantum-bit to code chromosomes to enhance the quantum genetic algorithm’s evolution process. Liang and Huang [126] and Chen et al. [127] incorporated rough set into the genetic algorithm for reducing population domain and constraining crossover operators. Tang and Ai [128] implemented a knowledge-based crossover operator. Moreover, Ait Wakrime et al. [129] developed a combination
of exact methods (SAT encoding) with a genetic algorithm. Que et al. [130] implemented a hybrid solution with information enthalpy inherited from an artificial immune system to address the deterministic probability of genetic algorithm operator, which leads to premature converge problem. Sadeghiram et al. [131] developed a cluster guided genetic algorithm by implementing k-means in the process of generating the initial population.

A retrospective analysis of the selected article shows that incorporating novel QoS modeling has also been a viable strategy. Feng et al. [132] introduced a multi-objective fuzzy QoS attribute to be optimized by a genetic algorithm. Moreover, Jatoth et al. [133] presented an Optimal Fitness Aware Cloud Service Composition (OFASC) where enables connectivity between QoS attributes. Data clustering has been one research line to enhance the genetic algorithm’s performance by reducing search space by discovering service repository patterns. Data clustering techniques called Quality Constraints Decomposition (QCD) [134], and k-means [135] incorporated with genetic algorithms led to a more efficient solution. Jatoth et al. [136] developed an evolutionary algorithm with a skyline operator to prune the service repository. Since generating the initial population is a critical element in genetic algorithms efficiency, the learning process in machine learning techniques such as Q-learning [137] used to improve the quality of generated population to accelerate the convergence.

**Hybrid Genetic Programming-HGP** is a particular application of genetic algorithm which, widely used in fully automated composition strategy with a graph-based representation. In fully automated composition, abstract workflows construct the output-input connections between services where the algorithm operator is used in an evolutionary process to select composition service in an unconstrained way. Furthermore, another hybrid model with skyline operator [81] was introduced to filter the service repository. Furthermore,
Yu et al. [138] exploit the greedy search to generate the initial population. [139]

**Hybrid Multi-Objective Genetic Algorithm-HNSGAII** is a multi-objective approach based on the genetic algorithm. In most of the research works, the problem of service composition has converted to mono-objective optimization. However, Da Silva et al. [140] implement a hybridize NSGAII with a multi-objective evolutionary algorithm (MOEA/D), in which a simple local search is implemented. Liu et al. [141] introduced a differential evolution (DE) to play the role of algorithm operator in traditional NSGA-II. Furthermore, Liu and Zhang [142] proposed algorithms that apply the combination of multi-objective evolutionary approaches and decision-making methods.

**Hybrid Particle Swarm Optimization-HPSO** is one of the most prominent swarm intelligence algorithms, which reported in existing literature as the second most co-occurred hybrid metaheuristics. Algorithm operators inspired by ergodic chaos theory [143], [144] and quantum mechanics [145], [146] employed to address easy being in traditional PSO. Yin et al. [147] implement genetic operators into PSO to improve swarm diversity. Moreover, Wang et al. [148] proposed an improved PSO efficiency by using skyline techniques to prune redundancies in the service repository. In the same context, Hossain et al. [149] developed K-means clustering for PSO that runs on Hadoop to reduce the time complexity. In addition, Chifu et al. [150] adopted a data clustering method to identify the dense area in search space to create a robust search capability for finding global optima and avoiding local entrapments. Furthermore, Gharbi and Mezni [151] proposed a method called relational concept analysis (RCA) to reduce search space. Heuristics techniques such as clonal selection algorithm [152] and harmony search [153] incorporated with skyline operator and a predatory search strategy [144] to enhance search capacity of traditional PSO. Da Silva et al. [154] implemented a fully automatic service composition using a planning algorithm. Hybrid PSO with metaheuristics including artificial immune system algorithm [155] and harmony search [153] proposed to guarantees a delicate balance between exploration and exploitation. Haytamy and Omara [156] developed a two-stage hybrid model where the output of recurrent neural network methods (long short term memory-LSTM) fed to PSO in order to transfer QoS data to sequential information. Moreover, Hosseinizadeh et al. [157] developed a hybrid PSO with an artificial neural network to enhance QoS parameters.

**Hybrid Artificial Bee Colony-HABC** is another swarm intelligence and nature-inspired metaheuristic that has been frequently used to tackle the problem Huo et al. [158] developed a guided ABC inspired by the search mechanism in PSO and implement an analytic hierarchy process (AHP) into QoS modeling. This hybrid approach employed roulette wheel selection in the genetic algorithm for the onlooker forging operator in ABC. Copula theory, a state-of-the-art statistics frontier proposed by Zhou and Yao [159] for selecting an initial elite population in ABC to improve search efficiency. Furthermore, in another paper [160], the authors proposed differential evolution operators incorporated in a multi-objective ABC to improve the updating equation and information exchange between forging bees. At the same time, the scalar optimization subproblems of MOEA/D used to generate a better initial population.

Novel QoS modeling, including interval number multi-objective [161], and fuzzy ranking method with fuzzy numbers introduced [162], and ameliorated with artificial bee colony to achieve better results. Zhou and Yao [163] proposed a cuckoo search with a levy flight operator to improve global exploration capability. Authors in [164] also used levy flight in the onlooker bee phase to address slow algorithm convergence and poor exploitation. Finally, It is noticeable that most hybrid artificial bee colony algorithms for service composition proposed for the clouds manufacturing environment.

**Hybrid Ant Colony Optimization-HACO** also has been effective to solve the service composition, which is primarily associated with graph representation of problem whereby each pool of concrete services is organized in a series of connected layers to perform composition according to user requirements. In this case, service composition will be an exercise to find the optimal path. Researchers have shown interest in the hybridization of an ant colony with other metaheuristics.

Yang et al. [165] developed a hybrid-ACO whereby a genetic algorithm was employed to select the critical parameter for ACO. Genetic algorithm celebrated for its agile, robust search mechanism while suffering from lack of solid search capability in finding global optima in some cases. On the other side, ACO has a reputation as a global search algorithm while suffering from slow converge, particularly large-scale problems. Hence, Yang et al. [166] proposed a hybridization where the ant-colony algorithm served as the seed of genetic operation. Liu et al. [167] integrated an ant system to culture algorithm. Poor stagnation has been reported as ACO pitfalls in existing literature. Alayed et al. [168] improved the ACO algorithm to enhance diversity and avoid stagnation by embodying a swapping process.

**Hybrid Artificial Immune System-HAIS** inspired by the immune system and mapped to service composition problem using the clonal selection process. Service composition is represented by antibodies in immune system inspiration, while fitness function act as an antigen. Salomie et al. [169] introduced a hybrid genetic operator in the clonal selection process to avoid the local entrapment pitfall. Moreover, Gao et al. [170] introduced a new artificial immune algorithm based on the immune memory clone and clone selection algorithm by incorporation the fuzzy triangular numbers in QoS modeling.

**Hybrid Grey Wolf Optimization -HGWO** is one of the latest metaheuristics that showed a robust search capability in a wide range of problems. Bouzary and Chen [171] proposed a hybrid GWO embedded with genetic operators to avoid the stagnation of algorithm in the hunting process for cloud manufacturing applications. Furthermore,
Bhaskar et al. [172] developed a hybrid method by combining levy flight operators in hunting mechanisms in GWO, which run in the MapReduce environment. The follow-up effort to improve the algorithm’s performance enhanced the initial population quality by a backward learning strategy. Yang et al. [173] proposed a hybrid multi-objective grey wolf optimizer considering an amalgam of service quality attributes and energy consumptions.

Other Hybrid Methods also appeared frequently in the existing literature by given dominance to the newest metaheuristics. Pop et al. [174] combined cuckoo search with reinforcement learning and Tabu search. In this arrangement, enforcement learning keeps track of service replacement while tabu search, a trajectory metaheuristic set to accelerate the slow convergence. In a graph representation, authors in [175] developed a hybrid solution by inheriting the evolutionary operator to overcome optimum stagnation in the Firefly algorithm. Similarly, Sadouki and Tari [176] developed an Elephants Herding Optimization (EHO) algorithm based on crossover operator. Modification of algorithm operators remained popular to deal with the newest metaheuristic. In this context, Gavvala et al. [10] embedded an eagle strategy inspired by eagle search in the whale optimization algorithm’s exploratory phase to achieve a proper balance between exploration and exploitation. Podili et al. [177] developed a hybrid BAT with a differential optimization technique. In addition, a series of work suggested the application of traditional metaheuristic, including PSO in a Mutant Beetle Swarm [178], the discrete immune algorithm in a fruit fly optimization [179], and gravitational attraction search embodiment in traditional imperialist competitive algorithm [180] in order to exploit advantages of both methods in a unified hybrid solution. Furthermore, Chifu et al. [181] proposed a honey-bees mating optimization algorithm with a fusion of components from the genetics algorithm, tabu search, and reinforcement learning. In the same context, A beetle antennae search algorithm integrated with PSO [178] in order to generate a better initial population and consequently achieve faster convergence. Several scholars have recently developed more sophisticated methods.

Moreover, Li et al. [182] proposed a harris hawks optimization (HHO) algorithm based on a chaotic sequence to avoid falling in the local optima trap. The authors also employed K-mean for clustering and sorting of service sets. The size of the search plays a critical role in algorithm performance. Peng et al. [183] proposed a multi-clusters adaptive brain storm optimization incorporated with a twin support vector machine (TWSVM) to reduce search space.

The selected approaches (Table 4) published on well-known venues, including Springer, IEEE, and Elsevier, with a relatively equal share of 25% as shown in Fig. 9. Less than 20% of articles also distributed to other journals, mainly on a solo basis.

### V. RESULTS AND DISCUSSION

In this section, a detailed investigation of selected studies is presented to address research questions. First of all (RQ1, RQ2), a systematic literature review conducted to elaborate on current hybridization practice and researcher motivation in which results are shown in Table 5 and Fig. 11. Furthermore, meticulous data extraction performed to structure the research domain and highlights research trends and directions (RQ3, RQ4). The results were counted separately by dividing the number of occurrences of each parameter with the sum of all

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No</td>
<td>Other Methods</td>
</tr>
<tr>
<td>1</td>
<td>BAT</td>
</tr>
<tr>
<td>1</td>
<td>BSO</td>
</tr>
<tr>
<td>1</td>
<td>CS</td>
</tr>
<tr>
<td>1</td>
<td>EHO</td>
</tr>
<tr>
<td>1</td>
<td>ES</td>
</tr>
<tr>
<td>1</td>
<td>Firefly</td>
</tr>
<tr>
<td>1</td>
<td>Fruit Fly</td>
</tr>
<tr>
<td>1</td>
<td>Cuckoo Search</td>
</tr>
<tr>
<td>1</td>
<td>HBM</td>
</tr>
<tr>
<td>1</td>
<td>HHO</td>
</tr>
<tr>
<td>1</td>
<td>ICA</td>
</tr>
</tbody>
</table>

$S (1, 2, 3, 4, 5) = Strategy (1, 2, 3, 4, 5)$
parameter occurrences using Eq.3.

\[
\text{percentage}(i) = \frac{\text{occurr}_{\text{no}}(i)}{\sum_{j=1}^{n} \text{occurr}_{\text{no}}(j)} \quad (3)
\]

A. COMPOSITION APPROACH

There is not a unique approach to perform service composition. The researchers have adopted the twofold strategy, one semi-automated [87], and the other fully automated [140], [184]. Most current articles (%94) proposed a semi-automated approach where an abstract workflow fulfills user requirements. In contrast, fully automated service composition [185] get away with predefined workflow by setting an appropriate workflow to transit from the current state to the desired one. The fully automated composition is merely suggested in the evolutionary algorithms. The vast majority (72%) of articles propose a hybrid method in vector representation, while only (23%) articles used graphs in their problem modeling. Vector representation has been common in population-based algorithms given dominance to GA and PSO, while graph representation was popular with ACO, ABC, and AIS. It is worth remarking that only %6 of selected articles adopted multi-objective optimization with independent fitness function.

B. CHOICE OF ALGORITHM

Hybrid metaheuristics have been trending methods in service composition. Our investigation shows nature-inspired metaheuristics cemented itself as an ideal strategy and primary choice for web service composition. The evolutionary algorithms [186] such as genetic algorithm, genetic programming, and NSGAI made up over 40% of hybrid solutions due to their scalability and robust search capability. Next, the particle swarm optimization algorithm has been the second most used choice (21%). A hybrid variant of artificial bee colony made up 10% of the solutions. The sole hybrid instances of the latest metaheuristics formed 11% of methods while the rest of the nature-inspired algorithm only made less than 4%.

The core objective of these studies has been to obtain an optimal or near-optimal solution for this NP-hard problem in a minimal time budget. In order to achieve this, a broad spectrum of solutions evolved from none heuristics (exact methods) to heuristic and, finally, metaheuristics. Furthermore, the continuous quest to find optimal or near-optimal solutions led researchers to investigate hybrid metaheuristics that overcome traditional algorithms’ inadequacy. Each of these endeavors, examined with an empirical experiment to prove the superiority of proposed hybrid solutions. The need for hybrid metaheuristics arises from search inadequacy in traditional metaheuristics such as slow or premature convergence, stochastic behaviour and insufficiency in big service composition. Hence, It is high in the research agenda to find an optimal algorithm design to resolve the problem in a reasonable time budget, even for big service composition.

C. HYBRIDIZATION STRATEGY

The hybrid strategies for metaheuristics are shown graphically in Fig.10. Traditional metaheuristics inadequacies influence the idea behind proposing a hybrid algorithm in many cases. The objectives of the hybridized model can be summarized in the following points:

- To improve solution qualities.
- To accelerate convergence.
- To avoid local entrapment.
- To reduce search space.
- To improve exploration or exploitation.

Above all, S-1 over half of the studies (54%) proposed a hybrid method by replacing or modifying one or a few operators that inherited search mechanisms from other methods. According to Dokeroglu et al. [189] metaheuristics introduced before 2000, which are an overwhelming portion, named classical. Classical metaheuristics such as GA, PSO, ABC, and ACO predominantly suffer from easy being and randomness in search space, particularly in service composition. Therefore, much research is dedicated to proposing a hybrid alternative with dynamic operators. As we were delving into empirical evidence of the selected study, it can be noticed that premature convergence and slow convergence are critical setbacks in metaheuristics that trigger the idea of hybridization in general. The observations over empirical studies show evolutionary algorithms, particularly genetic algorithms and swarm intelligence techniques, especially particle swarm optimization, are the current paradigm for the base of the algorithm. Moreover, empirical evidence suggests that genetic algorithm suffers from slow convergence while PSO-based algorithms diverge at local optima in some circumstances. Operator modification is shown to be a potent remedy to overcome algorithm weakness by integrating another algorithm operator. For instance [171], the evolutionary operators from the evolutionary algorithm have been inherited and implemented into the swarm intelligence method. By this practice, the hybrid approach benefits from the search capability of evolutionary algorithms and the fast convergence of swarm intelligence.
### TABLE 5. Selected hybrid metaheuristics methods: S stand for strategy.

<table>
<thead>
<tr>
<th>Reference</th>
<th>Algorithm (Base)</th>
<th>Year</th>
<th>Multi Objective</th>
<th>Hybridization Strategy</th>
<th>Experimental Setup</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ye and Mouloua [116]</td>
<td>GA</td>
<td>2008</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Ma and Zhang [119]</td>
<td>GA</td>
<td>2009</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Parejo et al. [120]</td>
<td>GA</td>
<td>2010</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Gao et al. [121]</td>
<td>GA</td>
<td>2013</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Liang and Huang [126]</td>
<td>GA</td>
<td>2014</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Yang et al. [165]</td>
<td>ACO</td>
<td>2015</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Liu et al. [167]</td>
<td>ACO</td>
<td>2016</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Liu et al. [141]</td>
<td>GA</td>
<td>2017</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Tang and Ai [128]</td>
<td>GA</td>
<td>2018</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Wang and He [143]</td>
<td>PSO</td>
<td>2019</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Salomie et al. [169]</td>
<td>AIS</td>
<td>2020</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Pop et al. [174]</td>
<td>Cuckoo Search</td>
<td>2021</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Pop et al. [175]</td>
<td>FireFly</td>
<td>2022</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Liu et al. [146]</td>
<td>PSO</td>
<td>2023</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Zhao et al. [152]</td>
<td>PSO</td>
<td>2024</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Zhao et al. [155]</td>
<td>PSO</td>
<td>2025</td>
<td>✓</td>
<td>✓</td>
<td>WSDream</td>
</tr>
<tr>
<td>Liu et al. [186]</td>
<td>GA</td>
<td>2026</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Mardukhi et al. [134]</td>
<td>GA</td>
<td>2027</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Wang et al. [139]</td>
<td>GP</td>
<td>2028</td>
<td>✓</td>
<td>✓</td>
<td>WSC2008</td>
</tr>
<tr>
<td>Jula et al. [177]</td>
<td>ICA</td>
<td>2029</td>
<td>✓</td>
<td>✓</td>
<td>C#</td>
</tr>
<tr>
<td>Wang et al. [187]</td>
<td>PSO</td>
<td>2030</td>
<td>✓</td>
<td>✓</td>
<td>QWS,SG</td>
</tr>
<tr>
<td>Ma et al. [139]</td>
<td>GP</td>
<td>2031</td>
<td>✓</td>
<td>✓</td>
<td>WSC2008</td>
</tr>
<tr>
<td>Yu et al. [138]</td>
<td>GP</td>
<td>2032</td>
<td>✓</td>
<td>✓</td>
<td>WSC2008</td>
</tr>
<tr>
<td>Yin et al. [147]</td>
<td>PSO</td>
<td>2033</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Chiu et al. [150]</td>
<td>PSO</td>
<td>2034</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Hto et al. [158]</td>
<td>ABC</td>
<td>2035</td>
<td>✓</td>
<td>✓</td>
<td>QWS,SG</td>
</tr>
<tr>
<td>Peng et al. [132]</td>
<td>GA</td>
<td>2036</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Chen et al. [127]</td>
<td>GA</td>
<td>2037</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Liu and Zhang [142]</td>
<td>NSGA-II</td>
<td>2038</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Jatoh and Gangadhara [145]</td>
<td>PSO</td>
<td>2039</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Faruk et al. [124]</td>
<td>GA</td>
<td>2040</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Bao et al. [120]</td>
<td>GA</td>
<td>2041</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Seghir and Khabbaba [123]</td>
<td>GA</td>
<td>2042</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>da Silva et al. [154]</td>
<td>PSO</td>
<td>2043</td>
<td>✓</td>
<td>✓</td>
<td>WSC-2009</td>
</tr>
<tr>
<td>Hossain et al. [149]</td>
<td>PSO</td>
<td>2044</td>
<td>✓</td>
<td>✓</td>
<td>n-m</td>
</tr>
<tr>
<td>Zhou and Yao [139]</td>
<td>ABC</td>
<td>2045</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Zhou and Yao [164]</td>
<td>ABC</td>
<td>2046</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Zhou and Yao [163]</td>
<td>ABC</td>
<td>2047</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Podili et al. [177]</td>
<td>BAT</td>
<td>2048</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Savarala and Cheila [179]</td>
<td>FruitFly</td>
<td>2049</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Kartimi et al. [135]</td>
<td>GA</td>
<td>2050</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Elsayed et al. [137]</td>
<td>GA</td>
<td>2051</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Chiu et al. [180]</td>
<td>HBM</td>
<td>2052</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Zhou et al. [160]</td>
<td>ABC</td>
<td>2053</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Que et al. [130]</td>
<td>GA</td>
<td>2054</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Sadeghizam et al. [131]</td>
<td>GA</td>
<td>2055</td>
<td>✓</td>
<td>✓</td>
<td>WSC-2008, WSDream</td>
</tr>
<tr>
<td>Munty et al. [79]</td>
<td>GA</td>
<td>2056</td>
<td>✓</td>
<td>✓</td>
<td>Public Stats</td>
</tr>
<tr>
<td>Jatoh et al. [136]</td>
<td>GA</td>
<td>2057</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Liu et al. [141]</td>
<td>NSGA-II</td>
<td>2058</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Da Silva et al. [140]</td>
<td>NSGA-II</td>
<td>2059</td>
<td>✓</td>
<td>✓</td>
<td>WSC-2008, QWS</td>
</tr>
<tr>
<td>Xu et al. [144]</td>
<td>PSO</td>
<td>2060</td>
<td>✓</td>
<td>✓</td>
<td>n-m</td>
</tr>
<tr>
<td>Seghir et al. [161]</td>
<td>ABC</td>
<td>2061</td>
<td>✓</td>
<td>✓</td>
<td>WSDream</td>
</tr>
<tr>
<td>Alayed et al. [168]</td>
<td>ACO</td>
<td>2062</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Sadouni and Tari [176]</td>
<td>EHO</td>
<td>2063</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Gavvala et al. [10]</td>
<td>ES</td>
<td>2064</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Yang et al. [166]</td>
<td>GA</td>
<td>2065</td>
<td>✓</td>
<td>✓</td>
<td>n-m</td>
</tr>
<tr>
<td>Jatoh et al. [133]</td>
<td>GA</td>
<td>2066</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Bouzayr and Chen [171]</td>
<td>GWO</td>
<td>2067</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Fekih et al. [153]</td>
<td>PSO</td>
<td>2068</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Seghir et al. [162]</td>
<td>ABC</td>
<td>2069</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Gao et al. [170]</td>
<td>AIS</td>
<td>2070</td>
<td>✓</td>
<td>✓</td>
<td>RG</td>
</tr>
<tr>
<td>Yang et al. [178]</td>
<td>BHA</td>
<td>2071</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Peng et al. [182]</td>
<td>BSO</td>
<td>2072</td>
<td>✓</td>
<td>✓</td>
<td>n-m</td>
</tr>
<tr>
<td>Ati Wakhrie et al. [129]</td>
<td>GA</td>
<td>2073</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Bhaskar et al. [172]</td>
<td>GWO</td>
<td>2074</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Yang et al. [173]</td>
<td>GWO</td>
<td>2075</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Li et al. [181]</td>
<td>HHO</td>
<td>2076</td>
<td>✓</td>
<td>✓</td>
<td>QWS,SG</td>
</tr>
<tr>
<td>Hosseinzadeh et al. [157]</td>
<td>PSO</td>
<td>2077</td>
<td>✓</td>
<td>✓</td>
<td>QWS</td>
</tr>
<tr>
<td>Hayatamy and Omara [156]</td>
<td>PSO</td>
<td>2078</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
<tr>
<td>Gharbi and Mezni [151]</td>
<td>PSO</td>
<td>2079</td>
<td>✓</td>
<td>✓</td>
<td>SG</td>
</tr>
</tbody>
</table>
Randomicity that governs the stochastic optimization process of traditional metaheuristics has been the root of inefficiency. Therefore, one major effort has been to modify the search operator by leveraging chaos theory, quantum
mechanics, probability, and statistical analysis to generate a quality initial population with adaptive search mechanisms. These practices accelerate the convergence, improve solution quality and provide immunity against local entrapment. In this way, studies enhance the base algorithm by implementing an operator like levy flight [172] and local optimizer technique [128] to balance exploration and exploitation.

S-2 Around 20 % of research efforts have improved a metaheuristic algorithm by combining it with another metaheuristic to minimize the weakness and maximize their strength. These are the typical examples of this approach. For instance, Yang et al. [166] proposed a hybrid genetic using a robust global search mechanism of ant colony optimization to prevent the hybrid method from falling into the trap of local optima. Our survey shows that a population-based metaheuristic frequently employed as a base algorithm due to its hybridization capacity. The major disadvantage of combining multiple metaheuristics is sophistication, which leads to higher time complexity. Therefore, this strategy has been less popular in comparison to the practice of operator modifications.

S-3 Only 5 % of articles incorporate a hybridized fitness function with a combination of a few techniques to improve the algorithm performance. Since the incorporation of novel QoS modeling improved the performance of the base algorithm, it has been regarded as a hybrid strategy. One objective of this practice has been to enhance population diversity [119]. Fuzzy representations of QoS attribute [190] for the uncertain environment also shown to be a viable strategy.

An absolute majority (75%) of research works adopted the Simple Additive Weighting (SAW) fitness function when service composition problems are defined as single objective combinatorial optimization in a vector representation. The key idea here is computing aggregated QoS normalized values to represent the total quality index for composition. In the graph representation, 10% articles used structural information for the fitness function that counts the number of nodes in the overall structure to find the optimal path. An independent fitness function has been proposed in 10% of selected articles for multi-objective optimization in which conflicting QoS attributes can optimize independently.

S-4 There is a close association between search space dimensions and search method efficacy. Therefore, some researchers used service clustering techniques [191] to reduce the search space in order to improve algorithm performance. Service clustering has been a proven hybrid strategy to accelerate the search process.

The typical approach is to specify QoS attributes per atomic service instances; however, to employ clustering techniques, quality attributes can globally specify per composite service in a specific cluster. In the service composition domain, methods including but not limited to skyline [43], [192] [188], K-means [149], SVM [183] frequently has encountered in current literature.

S-5 In recent years, Machine Learning (ML) has gradually become a viable hybrid strategy to incorporate into metaheuristics. The machine learning techniques including Q-learning [137], deep learning [156], reinforcement learning [174] has been employed to improve the search mechanism on different grounds. While machine learning incorporated methods generating a high-quality population, they are not highly competitive from the time of execution perspective attributed to the sophistication of ML enabled solutions.

D. EXPERIMENT SETTINGS

According to a survey conducted in 2008, 5,077 WSDL-described web services extracted and are available on the Internet [193]. Cloud computing’s current popularity has fueled the rapid growth of cloud-based services, with a 2013 survey revealing 6,686 cloud services [194]. The most commonly used data sets in the entire experimental domain have been QWS [195] (20%) followed by WS-DREAM [196] (9%) and web service challenge (6%) [197]. Although public datasets were popular with researchers, over half of the research works used simulated (36%) and randomly generated (20%) datasets. Matlab (27%), Java (21%), and C programming (10%) made up the majority of implementation tools.

Multiple metrics used to evaluate the performance of metaheuristics. First of all, the total quality index or maximum achieved fitness value has been used to compare algorithm search capability. Moreover, time of execution and convergence behaviour are frequently used in empirical studies. Chen et al. [127] proposed a metric called hit rate, which is a percentage of the global optimum obtained from the exhaustive enumerations to represent algorithm convergence. However, metrics such as hit rate did not popularize in comparison with the existing system of measurement.

E. RESEARCH TRENDS

Analyses of the trends as it demonstrates in Fig. 12 show both numbers of publication and their impact (h-index) has gradually increased over the selected time frame. Researcher interest in hybrid metaheuristics can signify hybrid methods success in resolving service composition. It also indicates that solo metaheuristics are not sufficient to deal with the complexity of the problem. The application of genetic algorithms and particle swarm optimization has been steady and continuous. On the other hand, hybrid approaches based on artificial bee colony and grey wolf optimization are getting ground recently. The use of simple additive weight fitness function in the vector representation has also been a steady and continuous trend over time, and rising recently.

In recent years, researchers showed interest in implementing the multi-objective service composition with an independent fitness function. A noticeable growing trend in choosing Matlab and QWS data is apparent. Further, inferred trends show how algorithm modification inspired by other metaheuristics or algorithm operators has always been a viable
VI. ISSUES, CHALLENGES, AND FUTURE DIRECTIONS

Choice of an algorithm for a discrete and highly constrained problem of service composition remains an open issue. The challenge here is to achieve optimization algorithms that are potentially preventing subpar performance in different composition scenarios. This study indicates that vector representation in semi-automated service composition has been extensively researched. However, fully automated composition, with tree-based and graph-based representations, has not been thoroughly investigated and remains an open question on many fronts.

A. ALGORITHM DESIGN

The main objective of algorithm design for service composition is to devise fast convergence metaheuristics that can determine optimal solutions for large and dynamic service repositories in a minimal time budget. Algorithm efficiency even becomes more critical in resource-constrained emerging computational paradigms such as fog/edge, where the computational resources are not abundant as it is in cloud environments. On a general note, challenges arise when algorithm performance metrics conflict toward achieving excellence. Therefore, here, efficiency spells out a specific trade-off of performance metrics in respect to deployment conditions. For instance, execution time to perform service composition in the edge of networks with constrained computational resources may be more critical than quality optimally. In contrast, those algorithms designed to deploy in a high-performance computational environment are expected to produce the highest optimal solution. Another perspective is the stochastic behaviour of the majority of nature-inspired metaheuristics. The fluctuation of results in every algorithm run may lead to an unstable system, mainly when the optimal quality index should be mapped to the service repository to return precise service instances. On this basis, the current study did not address the stochastic behaviour of metaheuristics and the trade-off between algorithm performance metrics.

The use of multi-objective techniques is on the rise, which allows for a better understanding of how QoS attributes interact. These studies also contribute to the understanding of behavioural differences between multi-objective composition and single-objective. Earlier comparisons presumed that the multiple QoS attributes are always in conflict, even though the exact nature of the relationships between QoS attributes is unknown. Hence, adopting a multi-objective optimization problem formulation where QoS attributes were not considered independent parameters need further research. The remaining issue is that the existing studies did not discuss the merits of algorithm design based on these approaches with comparative analysis in terms of algorithm performance and validity of solutions.

B. SEARCH SPACE AND BIG SERVICE COMPOSITION

While exact methods (none-heuristics) can efficiently solve limited composition problems, big service composition may fail even the most efficient metaheuristics. In the case of fully automated composition, the proliferation of services leads to high computation costs. Thus, one remaining challenge is to design a hybrid approach to manage search space to maintain the efficiency of metaheuristics. Proliferation results in the vastness of search space and poses a significant threat to algorithm efficiency. This development directed research in a multifold direction. In order to deal with the bulkiness of the service repository, one research direction has been to devise preprocessing stage [136], [182] that structure the services. However, the stochastics arrival of services and dynamism governing the services in the real world makes this approach less feasible. Limiting search space to clusters and regions has been a popular strategy to reduce search space. However, the threat of pruning global optimal always has been associated...
with such an approach. Finally, service mining is deemed to be a challenging issue and less likely explored.

C. SOLUTION ACCURACY ANDENCODING STRATEGY
One of the most widespread assumptions of the existing method is the correctness of the best solution. However, finding a representation that successfully supports the creation of functionally correct solutions always has been fraught with difficulty. The complexity is attributed to interpretation results produced by metaheuristics. Even though the structure of the solutions is often straightforward to comprehend, the fitness results are not readily apparent in most cases. The empirical study in real-world scenarios eventually results to devise verification process. Although some attempts have been made to address this issue, it is still a challenging area. In order to address this challenge, metrics within the verification process should be devised to represent the accuracy and feasibility of the selected composite service. There is no clear understanding of how solutions are functionally correct in the context of fully automated, which has been dominated by genetic programming. In most existing works, the encoding strategy in metaheuristics for service composition where solutions are discrete or binary has not been stated clearly.

The absolute majority of solutions used continuous versions of metaheuristics without any adoption, while the discrete [162] or binary version [198] of the algorithm has been employed occasionally. A fundamental issue in a continuous version is mapping the final solution to discrete or binary values, which can become highly sophisticated when services proliferate. In order to avoid this, modification on the continuous version of metaheuristics requires to suit discrete or binary encoding. One outcome in binary encoding is an expansion of solution length when abstract workflow becomes more comprehensive. We believe that implementing service composition in the real-world scenario must be given further attention, emphasizing the correctness of solutions.

D. SECURE SERVICE COMPOSITION
Despite the wealth of literature available in the field, there is a lack of analysis on the security aspect of service composition. Existing research merely investigated the quality attributes of services. Although security is represented as one of the quality attributes named risk in few instances, security-aware [199] service composition did not give the weight that severe security challenges of today demand. The security threats have become more sophisticated by the advent of the multi-cloud environments where elastic services from multiple clouds give rise to security breaches. One promising research direction is to devise security-aware service composition with an eye on quality attributes. Moreover, exploring an efficient algorithm to tackle this problem formulation should be placed in the research agenda.

E. DYNAMIC AND UNCERTAIN SERVICE REPOSITORY
The emerging computational paradigm and unstable mobile environments that promote flexible service delivery caused uncertainty in QoS values. Nonetheless, the widespread assumption among existing works is QoS attributes are certain and static. In addition, the inter-dependency of QoS attributes has also been ignored without considering the correlation between attributes that affect overall QoS values. [200]. Although it is not feasible to propose a model covering all QoS attributes, ignoring uncertainty and inter-dependency lead to solutions that are not applicable in real-world scenarios. In the context of a dynamic and uncertain environment, the challenge arises in determining the optimal composition due to the dynamic availability or failure of services. One important research direction is the management of uncertainty [201] by investigating online optimization techniques that periodically improve the quality of services.

F. MACHINE LEARNING INCORPORATION
Machine Learning (ML) incorporation in metaheuristics is an underdeveloping area of research. Metaheuristics are scalable and problem independent search mechanisms with huge inspiration from natural phenomena. The randomosity in stochastic metaheuristics gives ground to machine learning embodiment by harnessing their learning capability to enhance metaheuristics performance. However, this hybridized model results in high computational complexity while excelling in terms of solution quality. Hence, studying hybrid models that are relatively efficient in terms of time complexity should be placed in the future research agenda.

G. TOWARDS EMERGING COMPUTATIONAL PARADIGM
Service composition in the cloud is an established practice while has not been fully explored in emerging computational paradigms. Constrains of resources at the edge of networks demand techniques with a delicate balance between efficiency and optimally. The advance in virtualization technology gives rise to microservices technology that serves end-users by clustering traditional monolithic architecture into a group of services in this uncertain and inter-related context. Microservice composition is the challenge of determining optimal solutions while providing the highest user experience possible [202]. Moreover, mobility and uncertainty governing the edge environment create new issues that have not been anticipated previously. These issues are attributed to the fact that the current experiment setting considers offline deployment and ignores real-world scenarios where service should deploy online. Hence, service composition in the context of the emerging computational paradigm requires further research.

VII. THE CONCLUDING REMARKS
The fundamental objective of the service composition study has been to achieve fast convergence and stable techniques that can locate high-quality solutions when
services proliferate. The empirical evidences show this objective may not realize within the realm of traditional metaheuristics. Fast convergence, avoiding local entrapment, dealing with large search space, and achieving high-quality solution has been motivating factor to devise hybrid metaheuristics that can achieve a delicate balance between exploration and exploitation. Therefore, hybrid approaches were employed to transcend the boundary of metaheuristics by leveraging the strength of multiple methods.

This study investigates the hybrid strategies proposed between 2008 and 2020 for QoS-aware service composition to provide taxonomical analysis focusing on hybrid metaheuristics as frontier solutions. The core contribution of this article is to provide a taxonomical analysis and of hybrid strategies for QoS-aware service composition with respect to the problem domain. Our survey indicated that hybrid approaches were extensively (54%) about modifying algorithm operators favouring more robust search mechanisms or suggesting hybridized methods with other metaheuristics (20%). As the proliferation of services added to the complexity of the problem over time, the researcher incorporated data clustering techniques (8%) to reduce search space. One contribution has been to redefine quality attributes with a more innovative approach or embody unique QoS modelling (13%) parameters into service composition. Machine learning (5%) incorporation with metaheuristics has been the latest trend in the field.

An overview of results also indicates that two-thirds of the hybrid approach tried to reform the base algorithm within the context of metaheuristics (inhering other heuristics operators or algorithms) while the rest subscribed to out of box techniques such as data clustering and machine learning. We also observed that an amalgam of data clustering, metaheuristic, and parallel computing techniques eventually results in an efficient method. However, a comprehensive empirical study that evaluates the efficiency of proposed methods in holistic terms is missing. Above all, the recent hybridization of machine learning, genetic programming, and the adoption of fully automated composition signalling a paradigm shift to hyper-heuristics.

ACKNOWLEDGMENT
The authors acknowledge Prof. Rajkumar Buyya input on Section VI of this manuscript.

REFERENCES


HADI NAGHAVIPOUR (Member, IEEE) received the B.Eng. degree in telecommunication engineering and the master’s degree in software engineering from Staffordshire University, U.K. He is currently pursuing the Ph.D. degree in computer science with the University of Malaya. As a Data Evangelist, he founded Geek Diamond (geekdiamond.com), in 2015, for commercializing innovative service designs. His research interests include but are not limited to service design, service computing, and selection.

TEY KOK SOON (Member, IEEE) received the B.Eng. and Ph.D. degrees in electrical engineering from the University of Malaya, Malaysia, in 2011 and 2014, respectively. Since 2011, he has been a Senior Lecturer with the Department of Computer System and Information Technology, Faculty of Computer Science and Information Technology (FCSIT). His research interests include the Internet of Things, renewable energy, control systems, energy management, and grid management.

MOHD YAMANI IDNA BIN IDRIS (Member, IEEE) received the M.Sc. degree in computer science and the Ph.D. degree in electrical engineering from Universiti Malaya. He has vast experience in research and has published many high-quality journals. He is currently an Associate Professor with the Faculty of Computer Science and Information Technology, Universiti Malaya, Kuala Lumpur, Malaysia. His research interests include the Internet of Things, renewable energy, control systems, energy management, and the IoT security, image processing, and computer vision.

ROSLI BIN SALLEH (Member, IEEE) received the B.Sc. degree in computer science from the University of Malaya, in 1994, and the master’s and Ph.D. degrees in data communication from the University of Salford, U.K., in 1997 and 2001, respectively. After finished his Ph.D. studies, he started as a Junior Lecturer and was appointed to the current position as an Associate Professor, in 2013. Until 2019, he has supervised to completion of nine Ph.D. students and more than 30 master students. He is currently an Associate Professor with the Department of Computer Systems and Technology, Faculty of Computer Science and Information Technology, University of Malaya. He has published more than 30 ISI articles in various ISI journals. His main research interests include software defined networking, mobile IP, and network security. He has received more than hundreds of thousands USD grants, many awards, and recognitions, and was appointed as a Panel in many national and international institutions.

ABDULLAH GANI (Senior Member, IEEE) received the Diploma degree in computer science from ITM, the B.Phil. and M.Sc. degrees in information management from the University of Hull, U.K., and the Ph.D. degree in computer science from the University of Sheffield, U.K. He is currently a Professor and the Dean Faculty of Computing and Informatics, University Malaysia Sabah. He is also an Honorary Professor with the Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia. He obtained tertiary academic qualifications from the University of Hull. He has vast teaching experience due to having worked in a number of educational institutions locally and abroad—schools, Malay Women Teaching College, Melaka, Ministry of Education, the Rotherham College of Technology and Art, Rotherham, U.K., The University of Sheffield, U.K. His interest in research kicked off, in 1983, when he was chosen to attend the three-month Scientific Research Course at RECSAM by the Ministry of Education, Malaysia. Since then, more than 150 academic articles have been published in proceedings and respectable journals internationally within top 10% ranking. He received a very good number of citations in Web of Science and Scopus databases. He actively supervises numerous students at all level of study, such as bachelor’s, master’s, and Ph.D. degrees. His research interests include self-organized systems, machine learning, reinforcement learning, and wireless related networks. He is also serves as a reviewer to several high-quality journals, and the Chief-in-Editor of the Malaysian Journal of Computer Science and ISI-indexed journal.

MORTEZA NAMVAR received the B.Sc. degree in computer engineering, the M.Sc. degree in IT engineering, and the Ph.D. degree in information systems. He is currently an Assistant Professor in business information systems and analytics with The University of Queensland (UQ), Brisbane, Australia. His research interests include the areas of social media analytics, deep learning, topic modeling, and text feature engineering. His articles have been published in several journals (CAIS and ITP) and peer-reviewed conference proceedings (ICIS and HCISS). He has served as an Associate Editor for ACIS and PACIS. He served as the Track Chair for ACIS and a reviewer for various top tier journals, including I&M and Internet Research and conferences (ICIS, ACIS, and HCISS).

MORETEZA NAMVAR received the B.Sc. degree in computer engineering, the M.Sc. degree in IT engineering, and the Ph.D. degree in information systems. He is currently an Assistant Professor in business information systems and analytics with The University of Queensland (UQ), Brisbane, Australia. His research interests include the areas of social media analytics, deep learning, topic modeling, and text feature engineering. His articles have been published in several journals (CAIS and ITP) and peer-reviewed conference proceedings (ICIS and HCISS). He has served as an Associate Editor for ACIS and PACIS. He served as the Track Chair for ACIS and a reviewer for various top tier journals, including I&M and Internet Research and conferences (ICIS, ACIS, and HCISS).