A META SYNTHESIS ON CLOUD TASK SCHEDULING ALGORITHMS: COVID-19 AND ONWARDS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS OF
THE MIDDLE EAST TECHNICAL UNIVERSITY
BY

ATA HÜSEYİN AKSÖZ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

MAY 2024

Approval of the thesis:

A META SYNTHESIS ON CLOUD TASK SCHEDULING ALGORITHMS: COVID-19 AND ONWARDS

Submitted by **ATA HÜSEYİN AKSÖZ** in partial fulfillment of the requirements for the degree of **Master of Science in Information Systems Department, Middle East Technical University** by,

Prof. Dr. Banu Günel Kılıç
Dean, **Graduate School of Informatics** ____________________

Prof. Dr. Altan Koçyiğit
Head of Department, **Information Systems** ____________________

Prof. Dr. Sevgi Özkan Yıldırım
Supervisor, **Information Systems Dept., METU** ____________________

Examining Committee Members:

Prof. Dr. Pekin Erhan Eren
Information Systems Dept., METU ____________________

Prof. Dr. Sevgi Özkan Yıldırım
Information Systems Dept., METU ____________________

Asst. Prof. Dr. Banu Yüksel Özkaya
Industrial Engineering Dept., Hacettepe University ____________________

Date: 27.05.2024

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Ata Hüseyin AKSÖZ

Signature : ____________________

ABSTRACT

A META SYNTHESIS ON CLOUD TASK SCHEDULING ALGORITHMS: COVID-19 AND ONWARDS

Aksöz, Ata Hüseyin
MSc., Department of Information Systems
Supervisor: Prof. Dr. Sevgi ÖZKAN YILDIRIM

May 2024, 102 pages

This thesis seeks to analyze different task scheduling algorithms proposed for Cloud Computing field and categorize such algorithms for different use cases. It is hypothesized that COVID-19 pandemic had huge impact on Cloud Computing field. The pandemic has shown that current Cloud Computing infrastructure is inferior as most business processes transferred to the Cloud and the network traffic increased. The research focuses on meta-synthesis of cloud task scheduling algorithms proposed during COVID-19 pandemic. How the pandemic has affected the design concerns is investigated. Narrative synthesis and thematic analysis of these algorithms in terms of their capabilities, performance, advantages and disadvantages are also done. It is known that efficient task scheduling is an issue. This issue has gained attention of lots of researchers who have proposed new algorithms or at least a variant of traditional scheduling algorithms. When those works are reviewed, it is seen that each algorithm has its pros and cons. Currently there is no such algorithm that can become the standard for task scheduling in Cloud Computing. This current situation in the field proves that there is a need for a meta synthesis of proposed algorithm. Hopefully this study will enlighten the Cloud Computing field for upcoming researchers. The aim of this thesis is to review state of the art algorithms and investigate how COVID-19 pandemic has affected design concerns of such algorithms. The final aim is to end up with a work that can be used as a starting point for many new researchers.

Keywords: Cloud computing, task scheduling, algorithm, meta-synthesis, systematic review

ÖZ

BULUT BİLİŞİM GÖREV ZAMANLAMA ALGORİTMALARI ÜZERİNE BİR META SENTEZ: COVID-19 VE SONRASI

Aksöz, Ata Hüseyin
Yüksek Lisans, Bilişim Sistemleri Bölümü
Tez Yöneticisi: Prof. Dr. Sevgi ÖZKAN YILDIRIM

Mayıs 2024, 102 sayfa

Bu tez, Bulut Bilişim alanı için önerilen görev zamanlama algoritmalarını analiz edip farklı kullanım senaryolarına göre kategorize etmeyi hedeflemektedir. COVID-19 pandemisinin, Bulut Bilişim alanına büyük etki ettiği hipotezi üzerinde durulmaktadır. Pandemi, mevcut Bulut Bilişim altyapısının iş süreçlerinin büyük bir kısmının Bulut'a taşındığı ve ağ trafiğinin arttığı koşullarda yetersiz olduğunu göstermiştir. Çalışma, COVID-19 pandemisi sırasında önerilen bulut görev zamanlama algoritmalarının meta-sentezi üzerine odaklanmaktadır. Pandeminin tasarım endişelerini nasıl etkilediği araştırılmaktadır. Bu algoritmaların yetenekleri, performansları, avantajları ve dezavantajları açısından anlatısal sentez ve tematik analiz de gerçekleştirilmektedir. Bulut Bilişim'de verimli görev zamanlamanın bir sorun olduğu bilinmektedir. Bu sorun, birçok araştırmacının dikkatini çekmiş ve yeni algoritmalar veya en azından geleneksel zamanlama algoritmalarının bir varyantını önermişlerdir. İncelenen çalışmalar, her bir algoritmanın kendi artıları ve eksileri olduğunu göstermektedir. Şu anda Bulut Bilişim'de görev zamanlama için standart haline ulaşmayı başarmış bir algoritma bulunmamaktadır. Bu alanın mevcut durumu, önerilen algoritmaların meta sentezine ihtiyaç olduğunu kanıtlamaktadır. Bu çalışma, gelecekteki araştırmacılar için Bulut Bilişim alanını aydınlatmayı amaçlamaktadır. Tezin amacı, son teknoloji algoritmaları gözden geçirmek ve COVID-19 pandemisinin bu tür algoritmaların tasarım endişelerini nasıl etkilediğini araştırmaktır. Ana hedef, birçok yeni araştırmacı için bir başlangıç noktası olarak kullanılabilecek bir çalışma oluşturmaktır.

Anahtar Sözcükler: Bulut bilişim, görev zamanlama, algoritma, meta-sentez, sistematik tarama

To my family and beloved ones…

ACKNOWLEDGMENTS

First of all, I would like to express my gratitude to my respectful supervisor Prof. Dr. Sevgi ÖZKAN YILDIRIM for her dedication, leadership and guidance. Her guidance has motivated me during my studies and led to a high-quality dissertation.

Next, I would like to thank my wife Nil for her support and motivation. She supported my studies and long study hours that I have passed. Her motivation and emotional support guided me through my studies.

I thank my mother Elvan, father Hakan and brother Bora for their guidance through my journey in life. Family is everything.

I would also like to thank my colleagues and managers at ASELSAN Inc. for their support and understanding during my studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CPU	Central Processing Unit
VM	Virtual Machine
SLR	Systematic Literature Review
SM	Systematic Mapping
ACO	Ant Colony Optimization
PSO	Particle Swarm Optimization
IWD	Intelligent Water Drops
GA	Genetic Algorithm
LB	Load Balancing
DVFS	Dynamic Voltage Frequency Scaling
DNS	Dynamic Shutdown
PICOC	Population, Intervention, Comparison, Outcome, Context
ML	Machine Learning
IaaS	Infrastructure as a Service
WaaS	Workplace as a Service
PaaS	Platform as a Service
SaaS	Software as a Service
DaaS	Data as a Service
WOA	Whale Optimization Algorithm
ABC	Artificial Bee Colony
DE	Differential Evolution
SOS	Symbiotic Organisms Search
CSA	Crow Search Algorithm
PeSOA	Penguin Search Optimization Algorithm
WCA	Water Cycle Algorithm
BBMO	Bumble Bee Mating Algorithm
SFO	Sunflower Optimization
SSA	Squirrel Search Algorithm
SSA	Salp Swarm Algorithm
RR	Round Robin
RVEA	Reference Vector Guided Evolutionary Algorithm
CRO	Chemical Reaction Optimization

GWO	Grey Wolf Optimization
ICA	Imperialist Competitive Algorithm
FA	Firefly Algorithm
HA	Hungarian Algorithm
HBB-LB	Honey Bee Behavior Based Load Balancing
HGSO	Henry Gas Solubility
GSA	Gravitational Search Algorithm
TSA	Thermodynamic Simulated Annealing Algorithm
MVE	Majority Voting Ensemble
MILP	A Mixed Integer Linear Formulation
FSDA	Fuzzy Self-Defense Algorithm
FPA	Flower Pollination Algorithm
DVFS	Dynamic Voltage Frequency Scaling
LLS	Last Level Cache
ALO	Ant Lion Optimization Algorithm
PeSOA	Penguin Search Optimization Algorithm
BWM	Best Worst Method
TOPSIS	Technique for Order Preference by Similarity to Ideal Solutions
BATS	Bandwidth Aware Divisible Task Scheduling
BAR	BAR Optimization Algorithm
BSA	Balanced Scheduling Algorithm
AEO	Artificial Ecosystem Based Optimization
DoI	Degree of Imbalance
WHO	World Health Organization

CHAPTER 1

1 INTRODUCTION

1.1 Problem Definition

Computing power and worldwide internet infrastructure improved enormously in last few years. The limit of computer processing power is no more measured in local domain. Huge computer networks distributed across the globe offer massive processing power as well as reliability. This situation gave the opportunity of conducting business tasks with new approaches. Due to the efficiency and reliability of distributed systems, businesses began adapting their services by offering them through such networks. This has resulted in the introduction of cloud networks and a new research field called “cloud computing“.

The massive improvement in computer processing power gained through the introduction of cloud systems has a downside. There is a huge demand for cloud-based services from both businesses and users. Yet cloud computing is so new topic and little research have been done in this field. Current cloud computing approaches are not efficient enough to expose the full potential of the processing power of cloud computing.

As the complexity of a cloud system increases, the number of resources that are needed to be managed also increase. Good resource management is the key point of maintaining efficiency in computing either in the cloud or local domain. Resource management is related with the scheduling of process tasks and allocation of resources. Since the introduction of single core CPUs, efficient multitasking has been a concern thus many different task scheduling algorithms have been offered. Now this problem has made its way to the cloud domain with much higher complexity due to increased number of resources, users and increase in parallel computation.

1.2 Motivation

Studies on task scheduling and resource allocation in Cloud Computing try to solve the problem of efficiently allocating cloud Virtual Machine (VM) resources to real-time tasks and running these tasks via a scheduling algorithm. Resource allocation is allocating VM resources to user tasks at platform level whereas task scheduling is determining how long each user task runs on the allocated resource and how tasks alternate. The context of this study is the issue of task scheduling in cloud computing.

There are several important aspects that happen during scheduling process which are performance, cost, efficiency and reliability. Task scheduling issue is one of the most important problems in cloud computing field. In one of the earliest publications on Cloud Computing, Armbrust et al. [1] stated that one major obstacle of Cloud Computing concerns scheduling of virtual machines. Since then, many publications were made and many algorithms/methods have been offered. However, efficient scheduling and resource allocation is still a problem. In a more recent publication, Singh et al. [2] stated that researchers still face problems while choosing a suitable scheduling algorithm that is efficient.

Many algorithms can be found in literature. Researchers tried to come out with efficient algorithms for years that fits the demands of the researchers. As Cloud Computing research area becomes more mature, number of proposed algorithms and surveys also increase. In such case it becomes inevitable to conduct a systematic review of such studies to support the future of the research area. It is critical to analyze the state-of-the-art using a strictly systematic methodology.

COVID-19 pandemic was a massive hit on global businesses and companies. It has changed work ethics, norms and daily human life for sure. Up until global pandemic happened, Cloud Computing was viewed not as a necessity for business processes but more of a luxury or choice for accomplishing certain tasks in a different way. However, people started working from home, schools shut down and e-learning trend begun. People created a huge demand on digital platforms such as Zoom, Netflix, Amazon etc. For this reason, COVID-19 pandemic was definitely a milestone for the development of Cloud Computing field. During these times, the capabilities of cloud platforms in any sector is pushed to their limits as the demand was massive compared to previous years. It can be inferred that the COVID-19 pandemic was a real-world benchmark for testing capabilities, performance and weakness of Cloud Computing services. Because of that, it can be inferred that the pandemic has affected the development of Cloud Computing. The main focus of this paper is to review cloud task scheduling algorithms proposed after the COVID-19 pandemic begun. The motivation for choosing this timeframe is filtering out the outdated ideas which are practically underperforming and staying up-to-date within the field.

This study covers the planning, conducting and reporting phases of a meta synthesis on task scheduling algorithms proposed for Cloud Computing. Review, survey and Systematic Mapping (SM) studies have been done in the field of resource allocation and task scheduling before. However, to the best of our knowledge, there is not any review study that is conducted with a systematic methodology. In other words, this study is the first meta-synthesis study on task scheduling algorithms proposed for Cloud Computing during COVID-19 pandemic.

1.3 Outline of the Thesis

During the Meta-Synthesis process, formal SLR methodology proposed by Kitchenham and Charters [3] for conducting SLR in Software Engineering field is used as the literature review part. This thesis study is divided into chapters whose contexts are as follows;

- Chapter 2 covers the background of the topic and related work found in literature. In this chapter, a brief introduction to cloud computing and task scheduling is done. The fundamentals of task scheduling and various approaches are explained briefly. Next, other secondary studies related with task scheduling algorithms are reviewed. These secondary studies are summarized in order to understand the findings and challenges of other reviewers.

- Chapter 3 states the systematic research, review and meta synthesis methodology that is followed during this study. The methodology and the steps of research and meta synthesis processes are explained.

- Chapter 4 shares the results. Firstly, initial analysis and statistics of the materials found in the literature are mentioned. Next, research questions are addressed and analysis of reviewed materials are shared.

- Chapter 5 discusses the findings and threats.

- Chapter 6 makes the conclusion of the study.

CHAPTER 2

2 BACKGROUND AND RELATED WORK

2.1 Preliminary Knowledge

Technological developments have always led to new demands from society as capabilities of systems increase. Storage costs and processing (computing) power have been the leading limiting factors of computing society for years. Scientists and engineers have been trying to develop new approaches for using storage space efficiently. Storage media capacities have increased enormously due to improvements in nanotechnology and production methodologies. This has led to a decrease in data storage costs [4].

Another significant development in computing field has happened in processing power. Microprocessor technology and parallel programming have taken huge steps which in turn made it easier to process tasks with heavy workload. Effortless computation means decrease in costs of computation.

Internet is one of the key developments of 20th century. In the beginning of 80s, internet was only accessible to a handful of people and was beginning to be accepted by the society in early 90s[4]. However, during early 2000s, huge improvements in internet infrastructure are done which led to new researches on how internet can change the way we do business, communicate and process data. A study published in 1997 questions how internet can revolutionize business sector. It is also stated that the potential use cases of internet in business processes should be based on services rather than products [5]. These are the first signs of cloud computing idea and a shift from product-oriented business approach to service and people-oriented approach.

The questioning of potential business uses of internet combined with the developments in digital storage and processing power lead to sharing resources across computers. The basis of this approach is distributed systems in which various tasks and resources are processed and used collaboratively by the members of a common network. The use cases of distributed systems can be categorized under 3 groups which are utility, grid and cloud computing [6]. Grid and utility computing are traditional use cases which are the basis of cloud computing idea.

The term “cloud” corresponds to a pool of computer related resources [7]. Difference of cloud computing from other kind of distributed systems is that cloud computing is versatile and capable of offering various services. One strength of cloud computing is the capability of resource management. Resource management involves load balancing and virtualization. These aspects make cloud computing environments much more complex than traditional distributed computing environments.

Due to the versatility aspect, there are several cloud computing service structures available. Different business sectors and business processes benefit from different kind of service structures. Over the years, businesses have evolved in order to implement cloud computing environments to their processes and adapted the service models accordingly. This has resulted in the following widely used and accepted service models today [4]:

- Infrastructure as a Service (IaaS)
- Software as a Service (SaaS)
- Workplace as a Service (WaaS)
- Platform as a Service (PaaS)
- Data as a Service (DaaS)

The definitions and details of these service models are out of context of this study. What we need to know is that all service models offer computing resources but with different use cases.

As stated before, cloud computing is a well-structured resource pool that provides service to many users. Due to this reason, management of cloud environment is critical. Resource pool management is done via “resource allocation” or “task scheduling”. The terms are used interchangeably in literature. The aim of task scheduling is to delegate user tasks to various resources for load balancing purposes. Resource allocation is the other way around in which resources are allocated to tasks for certain amount of time in order to serve all users’ demands. The aim of both are the same: managing resource pool in an efficient way and maximizing utility.

Task scheduling is not only a concern for cloud computing environments. It is also studied in operating systems in order to utilize parallel and concurrent computing. Several algorithms for task scheduling have been developed and are being used at kernel level of operating systems. However, such algorithms are not enough to manage highly versatile environment of cloud. For this reason, many new algorithms have been developed which are mostly nature inspired. None of these algorithms became the standard of cloud computing domain and developments still continue.

2.2 Related Work

It is critical to locate and analyze similar studies that are available in literature. Studies are categorized as primary, secondary and tertiary by Kitchenham and Charters [3]. Primary studies are defined as empirical studies that investigates a specific question. Secondary studies are studies that locate, review and analyze primary studies that are related with research questions. Finally, tertiary studies are defined as studies that consider secondary studies during review process in order to make a summation on same research topic.

SLR is considered as a secondary study by Kitchenham and Charters [3]. As stated in Chapter 1, this thesis consists of the phases of a meta-synthesis. Due to this reason, we searched for other secondary sources. Secondary sources consist of SLR, Systemmatic Mapping (SM), surveys and other forms of literature review studies (non-SLR). Each researcher identifies their secondary study by a different name but the key point for a secondary study is that it reviews primary studies. Thus we have looked for other studies that analyze primary studies in the field of Task Scheduling algorithms for Cloud Computing.

There aren’t much secondary studies that reviews task scheduling algorithms for Cloud Computing. For this reason, studies related with the topic of “task allocation” are also considered as these topics are similar to task scheduling. We have found 22 secondary studies which consist of systematic reviews, surveys and taxonomies from various WEB databases. These studies are then analyzed and categorized.

2.2.1 Analysis of Secondary Studies

Arunarani et al. [8] have done a literature survey in which they reviewed 65 papers from years 2005-2018. These papers consist of studies that propose an algorithm for scheduling problem in Cloud Computing. The authors have divided scheduling techniques to sub categories and classified the reviewed algorithms according to used technique. They have also classified the algorithms according to their intended application area (use-case). Another categorization the authors have done is made according to the evaluation metric of the study. It is stated that algorithms are mostly evaluated according to task completion time thus timing is an important constraint in scheduling. Authors have stated that future research in scheduling topic should focus on the ways of combining task scheduling and VM consolidation strategies in an effective way.

Motlagh el al. [9] have conducted a SLR in which they reviewed 67 papers that proposes an algorithm for Cloud Computing task scheduling issue. This work is similar to the work of Aunarani et al. [8] as Motlagh et al. have also reviewed the algorithms, categorized them according to the technique and commented on algorithm evaluation methods. However, authors have done these steps in a more systematic way as they have followed the SLR methodology. The authors have derived the advantages and disadvantages of each algorithm and commented on all of them. The algorithms proposed in that 67 papers are categorized under 3 main categories according to the cloud environment the algorithm is intended to be used at. Those are single cloud, multi-cloud and mobile cloud environments. In each category, authors divided the algorithms into 4 groups according to the evaluation metrics considered during the design phase of these algorithms. Those groups are energy aware, QoS aware, cost aware and multi objective algorithms. The study concludes that most popular environment the algorithms are designed for is single cloud environments. The most popular evaluation metric group

Keivani et al. [10] have done a literature review in which they have reviewed several task scheduling algorithms. Authors have categorized the algorithms depending on used techniques. They have derived the advantages and disadvantages of algorithms. However, they have commented on pros and cons of algorithms from the used technique perspective. In other words, they didn’t comment on each individual algorithm’s pros and cons. Authors have concluded their work by stating that no perfect algorithm exists in cloud computing task scheduling domain.

Singh et al. [11] have done a literature review on meta-heuristics based cloud computing task scheduling algorithms. Their research topic is more specific and isolated than previously mentioned studies as they focus on a specific category of task scheduling algorithms. They have gathered various papers from years 2008-2016. Authors categorized meta-heuristics approach under 2 types which are bio-inspired and swarm intelligence-based methods. Reviewed algorithms are classified according to these 2 categories and compared with other algorithms. Authors concluded that makespan (task completion time) is the most emphasized evaluation metric for scheduling algorithms. This conclusion is similar to the conclusion of Arunarani et al. [8].

Hosseinzadeh et al. [12] have done a comprehensive review on multi-objective scheduling algorithms. According to the reviewed papers, authors have categorized scheduling schemes under 2 categories which are “meta-heuristic” and “heuristic” scheduling schemes. They have further divided meta-heuristic category as “single objective” and “multi objective” optimization algorithms. Authors stated that heuristic algorithms focus on a specific scheduling problem whereas meta-heuristic algorithms focus on the optimal solution by applying heuristic algorithms. Authors’ main focus in their study is multi-objective meta-heuristic algorithms. In this aspect, this study can be considered as the continuation of the work of Singh et al. [11]. Authors also emphasize the power consumption and energy management issues in cloud computing scheduling schemes. During their review process, they have used “energy efficiency” as one of the main evaluation criteria. Authors have divided scheduling problem to 2 separate problems which are “task scheduling” and “workflow scheduling”. Scheduling in cloud environment is defined as allocating VMs to tasks. It is stated that workflow scheduling shares the same concerns with task scheduling but also considers constraints such as dependencies between tasks, data costs etc. During the review process, authors have divided multi-objective schemes under several categories according to the methodology used in the algorithms and later classified each reviewed algorithm as task scheduling or workflow scheduling algorithm. This paper's review process is one of the most detailed one among the other related works.

Ullah et al. [13] have done a review study in which they reviewed papers that propose a task allocation algorithm from the years of 2015-2021. They reviewed 106 papers however no categorization is done for reviewed algorithms. Algorithms are reviewed according to their advantages, weaknesses and addressed problems. Authors have focused on selecting ML based algorithms for review. This study provides detailed information about cloud infrastructure and load balancing issue. Also, the number of addressed algorithms and papers are relatively high compared with other related secondary studies. Due to these reasons, this study may be beneficial for new researchers.

Housssein et al. [14] have conducted a SLR study in which they deeply investigated the task scheduling problem. Authors have compared other secondary studies with their study and evaluated them according to the inclusion of following aspects:

- Taxonomy
- State-of-the-art
- Open issues,
- Future trends
- Year-wise comparison
- QoS-based comparative analysis
- Comparison of simulation tools
- Graphical representations

Authors claim that their work includes all these aspects. The main focus of their work is meta-heuristics based algorithms. However, authors have reviewed traditional and heuristic algorithms as well in order to let researchers differentiate meta-heuristic algorithms from the other ones. It is stated that research on scheduling in cloud computing started in 2005 but only after mid 2008 real development has occurred. Due to this reason, authors reviewed 71 papers from years 2011-2020. Taxonomy of scheduling strategies is provided and it is mentioned that scheduling algorithms are divided as "static" and "dynamic" algorithms. It is also stated that most popular meta-heuristic approach is "Particle Swarm Optimization (PSO)".

Jawade et al. [15] have done an analytical survey and reviewed 42 papers. These papers consist of task scheduling algorithms and approaches. Authors have classified the algorithms with 2 categorization methods. Algorithms use either static or dynamic model for scheduling and can be either multi objective based or single objective based. We have seen this kind of grouping (static vs. dynamic) in the work of Housssein et al. [14]. Authors have commented on each of the reviewed algorithms briefly. The performance evaluation methods are extracted from each paper and shared with a tabular format. Final comments state that future research in the domain of Cloud Computing task scheduling can focus on security and privacy concerns.

Pol et al. [16] have conducted a survey in which they briefly explained Cloud Computing fundamentals and stated the task scheduling issue. Authors have grouped scheduling algorithms under 2 main categories which are static and dynamic task scheduling algorithms. We have seen this categorization also in [14] and [15]. It is stated in this study that the main goal in designing a new task scheduling algorithm is to minimize makespan. Authors mention that "Pair Based Task Scheduling Algorithm" minimizes layover time but lacks the ability of fault tolerance. It is emphasized in this paper that Pair Based Task Scheduling Algorithm should be improved and combined with Ant Colony Optimization (ACO) algorithm.

Keivani et al. [17] have done a review study in which they have categorized task scheduling algorithms and reviewed them. Authors have stated the advantages and disadvantages of these algorithms as well. They have categorized the algorithms in literature as heuristic, evolutionary and economic scheduling algorithms. According to the paper, heuristic algorithms are easily implementable and provide quick solutions. In other words, this kind of algorithms are approximate. However, heuristic algorithms don't provide the best available solution so they are not accurate. For this reason, evolutionary and economic algorithms are focused more in this study. It is stated that no single algorithm can be mentioned as the best one. Further research should be focused on reliability, error handling and availability of task scheduling algorithms according to this study. Another future study recommendation is obtaining multi-objective algorithms with combining heuristic and meta-heuristic algorithms for faster scheduling approaches.

Sharma et al. [18] have done a survey study in which they made several different categorization of scheduling algorithms. Static, dynamic, user-level, heuristic, real time and workflow scheduling algorithms are explained and their definitions are given. In this paper, multi-objective scheduling algorithms are reviewed thoroughly. Final conclusions state that current solutions don't provide availability and reliability to the fullest. Kievani et al. [17] has also stated this problem in their work's final comments.

Bulchandani et al. [19] have done a survey on several different scheduling algorithms and categorized these algorithms. In this paper, tasks are categorized first which is not seen in previous studies. According to the paper, tasks can either be independent or dependent. In similar, scheduling algorithms can either be static or dynamic. Both dependent and independent tasks can be scheduled using static and dynamic methods. However, the specific algorithm used for scheduling is different for each. For a better understanding, Figure 1 shows the algorithm types for dependent tasks and independent tasks.

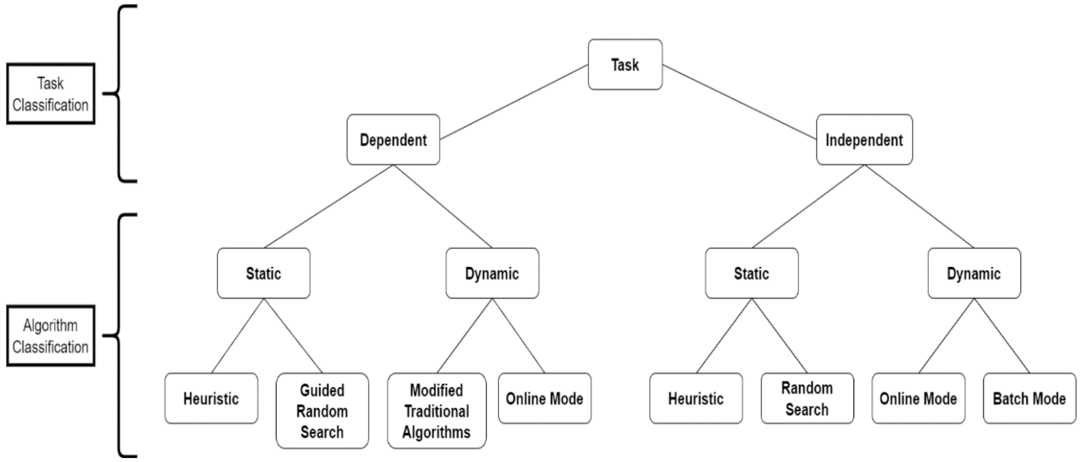

Figure 1 - Task and Algorithm Categorization Done by Bulchandani et al.

Ahari et al. [20] have categorized scheduling algorithms as static and dynamic algorithms. It is also noted in the paper that all heuristic algorithms are classified under dynamic algorithms which is a different approach from the view of Bulchandani et al. [19]. It is enforced in this study that Intelligent Water Drops (IWD) algorithms are quicker, provide higher quality and work better in dynamic cases. Authors propose a scheduling system based on Particle Swarm Optimization (PSO) and IWD algorithms. The study is finalized by stating that IWD algorithm should be improved and there is a huge need for further making research on scheduling algorithms.

Bharot et al. [21] state that Genetic Algorithm (GA) is a well-proven algorithm which is also widely used in Artificial Intelligence (AI) field. It is proposed in this study that task scheduling problem in cloud environment can be approached by focusing on Load Balancing (LB) techniques. Authors state that an improved version of the traditional GA can increase efficiency of task scheduling. Thus the fundamentals of GA is explained and the proposed approach is stated in this study.

Vijayalakshmi et al. [22] focus more on the efficiency of scheduling algorithms and reviews the literature according to 3 perspectives; energy consumption, load balancing and negotiation. For each concern category, authors reviewed algorithms and commented on them. Different kind of GA, ACO and PSO based algorithms are reviewed. The effects of COVID-19 on cloud computing field is also examined in this study.

Huang et al. [23] have done a review on GA, ACO and PSO algorithms used for scheduling. Their work summarizes these 3 algorithms and explains how each algorithm works. It is stated that GA, ACO and PSO algorithms tend to cause better results during scheduling process compared with other type of algorithms. It is stated in this study that each algorithm tends to fail or produce undesired results at certain cases. Authors finalize their study by stating that future research should focus on the weaknesses of ACO algorithm under the scenario of high number of resources with the requirement of longer task runtime.

Daniel et al. [24] have done a survey on load balancing issues. It is stated in this paper that task assignment is the most important process in Cloud Computing field. The importance of dynamic load balancing is emphasized more compared with other types of load balancing approaches. Authors stated that load balancing algorithms are classified as either static or dynamic. Static algorithms suit more to stable and homogeneous environments. Their downside is not being flexible to the changes in the environment. However, dynamic algorithms easily adapt to changing cases. They are also able to adjust scheduling decisions based on runtime parameters. Their downside is they need more processing power and carry larger overhead. Authors mention the challenges in Cloud Computing such as storage/replication, algorithm complexity, point of failure and distribution of nodes in cloud. The study continues by categorizing load balancing algorithms as static or dynamic. Authors then review various load balancing algorithms in the literature. Another concern of the study is privacy protection in cloud environment. Various algorithms intended to be used for protecting privacy are reviewed.

Tom et al. [25] have done a review study covering various task scheduling algorithms. It is stated that makespan is the mostly used metric for evaluating the algorithms by their designers. This finding supports the statements of Singh et al. [11] and Pol et al. [16]. According to the study, load balancing is a key factor of task scheduling algorithms in terms of affecting energy usage and environment. It is also stated that computing community is currently concerned with energy consumption. Due to this reason, future work should focus on developing energy efficient task scheduling algorithms.

Zubair et al. [26] have done a review on task scheduling issues in Cloud Computing. It is stated in this study that resource management is one of the most critical issues for decades. However, it is also stated that the findings of task scheduling issue related research and their algorithms are not currently enough and beneficial for computing community. One of the aims of this paper is hybridizing heuristic techniques with meta-heuristic techniques to achieve a more beneficial method. By this aim, it shows that authors have followed the future research recommendation of Keivani et al. [17]. This study specifically focuses on convergence issue of meta-heuristic algorithms. Convergence is a desired situation that occurs in genetics which is also a case in GA that leads to the optimum solution. However, premature convergence is not desired as the solution may not be the global optimum but a local optimum solution as stated by Andre et al. [27] in an early study.

Atiewi et al. [28] have done a review study on energy efficiency of task scheduling algorithms on cloud. Authors have reviewed various task scheduling algorithms that focus on achieving better power consumption and higher energy efficiency. It is stated in this study that the combination of Dynamic Voltage Frequency Scaling (DVFS) and Dynamic Shutdown (DNS) schemes in task scheduling algorithms result in the better energy efficiency compared with other approaches.

Alkayal et al. [29] have done a survey on PSO algorithms used in cloud environment. It is stated in this study that meta-heuristic approaches are faster and among the meta-heuristic approaches, PSO algorithms perform faster and are easy to implement. It is stated that the downside of PSO algorithms are slow convergence and high chance of premature convergence, which is converging to the local optima. Authors explain the characteristics of PSO algorithms in general and categorize them according to objective number and goal. It is concluded that PSO algorithms perform better in cloud environment compared with other methods. Future research should focus on storage cost and transfer time concerns of task scheduling algorithms.

A general summary of the reviewed secondary studies can be seen in Table 1.

Table 1 - Secondary Study Review Summary

Reference Paper	Study Type	Journal	Publication Year	Key Elements
[8]	Survey/Review	The International Journal of Escience	2019	-Huge number of papers reviewed. -Categorization of algorithms -Analysis of algorithms -Future research recommendation

Table 1 cont.

[9]	SLR	International Journal of Communication Systems	2020	-Systematic review methodology - Huge number of papers reviewed. - Categorization of algorithms -Analysis of algorithms -Advantages and disadvantages of algorithms
[10]	Survey/Review	19TH IEEE Mediterranean Electrotechnical Conference	2018	-Categorization of Algorithms -Advantages and disadvantages of categories
[11]	Survey/Review	Knowledge and Information Systems	2017	-Focus on meta-heuristic approach -Categorization of Algorithms
[12]	Survey/Review	Journal of Grid Computing	2020	-Focus on multi-objective meta-heuristic algorithms -Continuation of study [7] -Comments on energy efficiency

Table 1 cont.

[13]	Survey/Review	Artificial Intelligence Review	2022	-Highest number of review papers -No categorization is done -Pros and cons of algorithms are stated. -Good for new researchers.
[14]	SLR	Swarm and Evolutionary Computation	2021	-Systematic review methodology -Focus on meta-heuristics based algorithms
[15]	Survey/Review	International Journal of Engineering Trends and Technology	2021	-Categorization of algorithms -Future research recommendation
[16]	Survey/Review	ICSCCC 2021 - International Conference on Secure Cyber Computing and Communications	2021	-Fundamentals of Cloud Computing is mentioned
[17]	Survey/Review	2nd International Conference on Advances in Big Data, Computing and Data Communication Systems	2019	-Categorization of algorithms -Future research recommendation

Table 1 cont.

[18]	Survey/Review	International Journal of Scientific and Technology Research	2020	-Different categorization approaches -Focused on multi-objective algorithms
[19]	Survey/Review	International Journal of Scientific and Technology Research	2020	-Categorization of tasks -Categorization of algorithms based on task types
[20]	Survey/Review	Proceedings of the International Conference on Trends in Electronics and Informatics	2019	-Categorization of algorithms -Focus on IWD algorithms
[21]	Survey/Review	International Conference on Computing and Information Technology	2020	-Focus on GA -Focus on load balancing
[22]	Survey/Review	Journal of Green Engineering	2020	-Focus on efficiency of algorithms -Focus on effects of COVID-19 on Cloud Computing
[23]	Survey/Review	Advanced Materials Research Vols. 926-930 (2014)	2014	-Focus on GA, ACO and PSO algorithms -Future research recommendation
[24]	Survey/Review	Journal of Advanced Research in	2019	-Focus on load balancing issues

Table 1 cont.

		Dynamical and Control Systems		-Focus on privacy concerns of Cloud Computing algorithms
[25]	Survey/Review	International Conference on Inventive Computation Technologies	2020	-Focus on load balancing and energy consumption -Future research recommendation
[26]	Survey/Review	International Conference of Reliable Information and Communication Technology	2020	-Focus on hybridization of heuristic and meta-heuristic algorithms.
[27]	Survey/Review	Advances in Engineering Software	2001	-Focus on Genetic Algorithm -Focus on premature convergence
[28]	Survey/Review	IEEE Long Island Systems, Applications and Technology Conference	2016	-Focus on energy efficiency of algorithms. -Focus on DVFS and DNS
[29]	Survey/Review	International Conference on Electrical and Computing Technologies and Applications	2017	-Focus on PSO algorithms -Categorization of PSO algorithms.

It is seen from these studies that task scheduling issue in Cloud Computing is credited by many researches. Most of the researchers in this field agree and even clearly state that there is certainly a need for conducting review studies to group and summarize various task scheduling algorithms in literature. Each researcher focused on different concerns of task scheduling in their studies. Researchers insist on studying further on task scheduling issues and make future study recommendations to direct new researchers in this field.

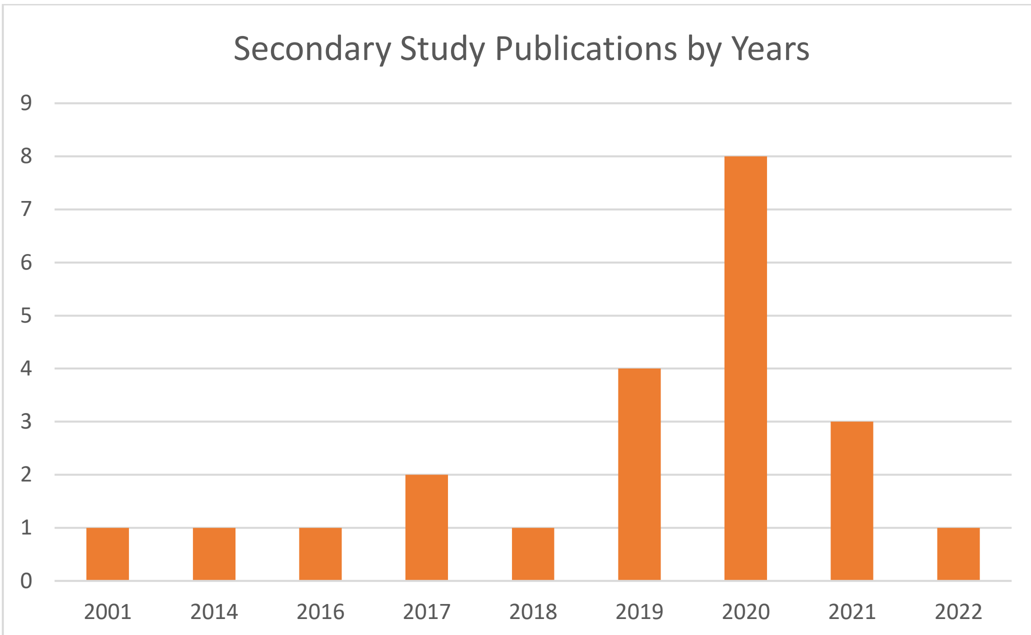

Figure 2 - Publication of Secondary Studies Related with Task Scheduling by Years

From these 22 secondary studies that are reviewed by us, it can be stated that there is a shortcoming of reviews conducted with a systematic methodology. The massive peak of study count in 2020 can also be correlated with the effects of COVID-19 pandemic which led researchers to focus more on Task Scheduling issues. This is the first sign of the impact of COVID-19 pandemic on Cloud Computing Task Scheduling issue as the topic gained attention during the first year of pandemic.

CHAPTER 3

3 RESEARCH METHODOLOGY

In this chapter, the methodology of our research is explained. As this thesis is a meta synthesis study, it must follow a strict methodology to locate, evaluate and analyze primary studies in literature. What makes a meta synthesis study different from other synthesis studies is the focus on qualitative data and thematic analysis. In meta synthesis, researchers should focus on the qualitative information extracted from the reviewed materials and make thematic analysis using that information. Thematic analysis is the process of combining the findings in reviewed studies to locate common patterns and themes.

In order to extract qualitative data from literature correctly, SLR is chosen as the review strategy. SLR follows a well-defined strategy at all parts of the study which is proposed by Kitchenham and Charters [3]. This is critical to extract necessary qualitative data. Meta synthesis process is made of three stages which are;

- planning,
- conducting,
- reporting

the review. These stages can be summarized as given in Table 2.

Table 2 - Stages of Meta Synthesis

1. Planning
Identification of the need for a research
Commissioning review[1]

[1] Optional step

Table 2 cont.

	Specifying research questions
	Developing review protocol
	Evaluating review protocol
2.	**Conducting**
	Identification of research
	Selection of primary studies
	Study quality assessment
	Data extraction and monitoring
	Data synthesis
3.	**Reporting**
	Specifying dissemination mechanisms
	Formatting the main report
	Evaluating the report

The Meta Synthesis stages given in Table 2 are explained in detail in the following section. Figure 3 shows the phases of a Meta-Synthesis study.

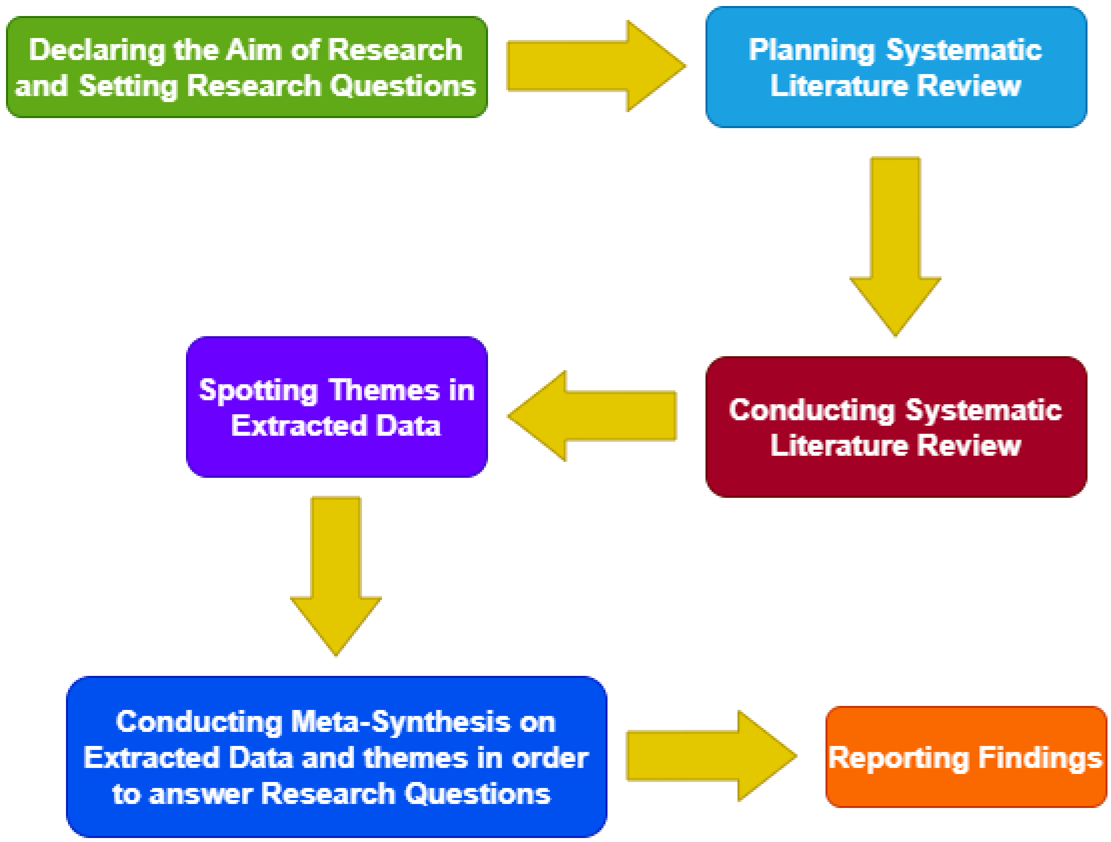

Figure 3 - Meta-Synthesis Steps Followed

Figure 3 shows the steps that are followed during this Meta-Synthesis study. SLR studies also have a similar approach until the data synthesis steps. These stages are combined together in our study. In Figure 4 the SLR methodology proposed by Kitchenham and Charters is given.

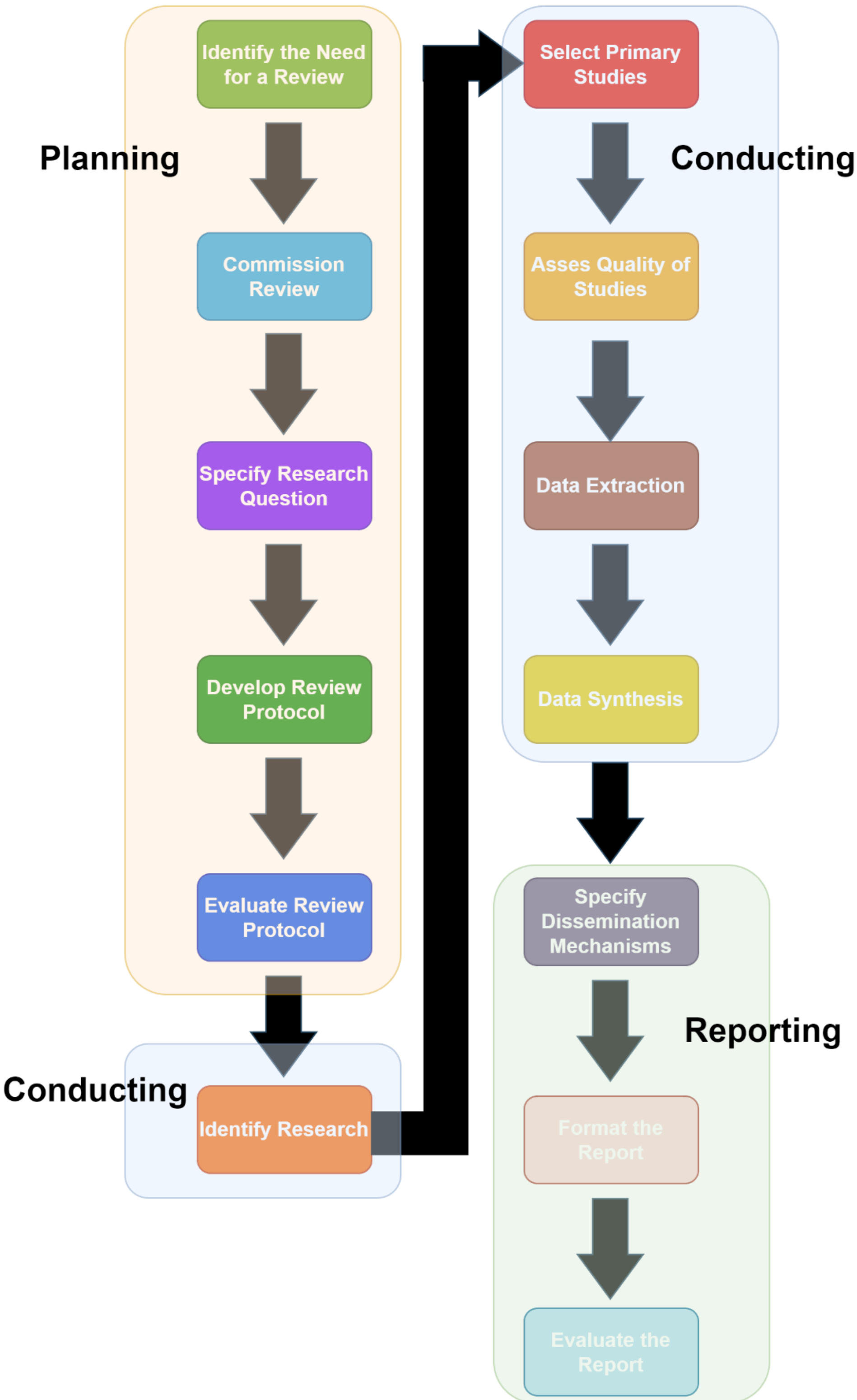

Figure 4 - SLR Steps

3.1 Planning

3.1.1 Identification of the Need for a Review

Each Meta Synthesis study should state the reason for conducting the research. The reason for conducting the research must be valid in terms of offering something new to the research field and beneficial for researchers. The statement of the reason why this Meta Synthesis is done is explained thoroughly in 1.2. Kitchenham and Charters [3] suggest reviewing other secondary studies, specifically SLR studies, at this step in order to get knowledge about how an SLR is conducted in the research field under consideration. For this reason, we reviewed various secondary studies that review task scheduling algorithms and summarized them at 2.2.1. This step was important in terms of understanding the perspective of other researchers during data synthesis and extraction steps. For example, some researchers have done meta-analysis on the reviewed paper while some have done thematic analysis. Some researchers have categorized the algorithms with similar approaches while the others held totally different opinions. It is also noted that some researchers followed the future research recommendations of other researchers.

3.1.2 Comissioning Review

This Meta Synthesis study is conducted solely by myself so no commissioning is done. This step is also stated as optional by Kitchenham and Charters [3].

3.1.3 Research Questions

Research questions are the basis of data synthesis so they must be defined according to the problem definition. To define the research questions, PICOC criteria which is suggested by Kitchenham and Charters [3] is used. The derived PICOC criteria is given in Table 3.

Table 3 - PICOC Criteria

Population	Cloud Computing
Intervention	Task Scheduling and Resource Scheduling Algorithms
Comparison	*Not applicable*
Outcome	Summary of state-of-the-art, categorization of current scheduling algorithms, summary of current scheduling algorithms. Future research recommendations.
Context	Scholarly (peer reviewed) articles

PICOC criteria is then used for determining research questions. Determining the research questions accurately is one of the most important steps of a Meta-Synthesis as data extraction and synthesis stages depend on the goals of research questions. For this reason, we have chosen 3 research questions. We further divided these questions to sub questions to achieve the goals of our study. Chosen research questions are as follows:

- **RQ 1 –** How are task scheduling algorithms for cloud computing environment categorized?

  Motivation:

  Categorization of algorithms depending on their similarities would be helpful to new researchers. A new researcher in this field may not know the general aspects of task scheduling algorithms. A fine categorization would lead researchers to study general categories of algorithms which would in turn help them understand the fundamentals of task scheduling algorithms and differentiate the algorithms proposed in literature.

  - **RQ 1.1 –** What is the percentage of algorithms in literature that fall in each category?
  - **RQ 1.2** – What are the advantages and disadvantages of each algorithm category?
  - **RQ 1.3** – In which scenarios are individual algorithms beneficial?

- **RQ 2 –** What is the mostly used evaluation metric for task scheduling algorithms in cloud computing.

  Motivation:

  Evaluation of the proposed algorithms in terms of performance metrics and business concerns are just as important as developing the algorithm. Evaluation results provide both qualitative and quantitative information which can be used for making empirical meta-analysis of algorithms. They also state the trustworthiness of the algorithm itself and the study under consideration.

  - **RQ 2.1 –** Which evaluation/simulation environment are used?
  - **RQ 2.2 –** How reliable are evaluation methods and are they repeatable under different cases?

- **RQ 3 –** What are the future research recommendations?

  Motivation:

  Cloud Computing is a new trend in computing field and task scheduling issue is just one of the research areas of Cloud Computing. It is for sure that task scheduling issue isn't solved. For this reason, it is important for new researchers to get informed about current status of research and avoid repetition. Recommendations given by experienced researchers is critical for achieving progression in research field.

  - **RQ 3.1 –** Which topics lack enough research currently in task scheduling field and need further attention?
  - **RQ 3.2 –** Is there any common agreement among researchers in further research direction?

3.2 Conducting - Review Protocol

Research questions determined and explained in 3.1.3 should be answered by following a systematically developed review protocol. This part explains how the literature material is searched and filtered according to a selection criterion, what literature databases and sources are used, how the data is extracted and synthesized to answer research questions.

3.2.1 Source Selection and Search Protocol

Accessing relevant study materials is a critical element of Meta Synthesis studies. For this reason, search protocol must be designed in a way that leads the researcher to relevant studies. It is also important for a Meta Synthesis study to be replicable and repeatable so source selection and search protocol must be well documented.

This section explains the search protocol thoroughly. Source selection and search protocol are divided to several steps. In the following part of this section, those steps are explained and motivation behind certain decisions during protocol development is stated.

- **STEP 1 - Search String Forming and Automatic Search**

Main sources used in this study are online databases and article search engines. Source selection is critical for being able to access all related materials in literature. For this reason, well respected databases are chosen. For automatic search, following databases are used:

- Web of Science[2]
- Scopus[3]
- IEEE Xplore[4]
- ACM Digital Library[5]
- Science Direct[6]

The search process is divided to several steps. Initially, search keyword is formed using the PICOC criterion derived in 3.1.3. Using such criterion helps developing search keyword which is more accurate in terms of retrieving source materials which are related more with the research questions. It is impossible to use all the source materials found during initial automatic search without filtering them. However, forming high quality search strings increase the chance of finding materials that are more relevant with the context of Meta Synthesis. This lessens the workload during selection of found studies and will lead to less material thrown away during application of inclusion/exclusion criterion.

---

[2] https://www.webofscience.com
[3] https://www.scopus.com
[4] https://ieeexplore.ieee.org
[5] https://dl.acm.org
[6] https://www.sciencedirect.com

Final search string is as follows:

Table 4 - Final Search String

(“cloud computing” OR “cloud-computing”) AND (“task scheduling” OR “task-scheduling” OR “task allocation” OR “task-allocation” OR “resource allocation” OR “resource-allocation”) AND “algorithm*”

The final search string is refined for each database due to syntax and search tool differences. Specific search strings used at each database can be seen in Table 5.

Table 5 - Search Strings List

Database	**Search String**
Web of Science	("cloud computing") AND ("task scheduling" OR "task allocation" OR "resource allocation") AND (algorithm OR technique OR strategy OR model)
Scopus	"cloud computing" AND "task scheduling" OR "task allocation" OR "resource allocation" AND algorithm OR technique OR strategy OR model
IEEE Xplore	("Document Title":cloud computing) AND ("Document Title":task scheduling OR "Document Title":task allocation OR "Document Title":resource allocation) AND ("Document Title":algorithm OR "Document Title":strategy OR "Document Title":technique OR "Document Title":model)
ACM Digital Library	("cloud computing") AND ("task scheduling" OR "task allocation" OR "resource allocation") AND (algorithm OR technique OR strategy OR model)
Science Direct	(cloud computing) AND (task scheduling OR task allocation OR resource allocation) AND (algorithm OR technique OR strategy OR model)

The search results are stored in Mendeley reference management tool.

- **STEP 2 – Elimination of Duplicates**

It is highly possible for multiple databases to store the same papers. For this reason, search results are inspected at Mendeley and duplicate ones are removed. The decision for a duplicate paper is done if there exists another paper in resources with same authors and same title.

- **STEP 3 – Initial Selection of Studies**

This is the first step of examining the search pool of a Meta Synthesis study. After removing duplicate studies, remaining ones are inspected very briefly. During inspection, title and abstract of the papers are reviewed. It should be clearly stated in the abstract that the study proposes a new task scheduling approach in cloud computing field or the algorithm should at least be a variant of available algorithms. In case the abstract didn't specifically mention that the study proposes a new algorithm or reviews an algorithm, then the introduction and conclusion sections of the study is reviewed with same approach. If it is decided that the study is relevant, it is then further included in study selection process.

- **STEP 4 – Inclusion and Exclusion Criteria**

After the final pool of studies are established, each paper is reviewed more thoroughly. During this inspection, papers are reviewed according to predetermined inclusion and exclusion criteria. For a paper in the pool to be eligible for data extraction and synthesis, it must pass all the inclusion criteria. If a paper fails to pass at least 1 inclusion criteria or passes at least 1 exclusion criteria, the paper is removed from the final pool of eligible papers.

Inclusion and exclusion criteria are given in Table 6.

Table 6 - Inclusion/Exclusion Criteria for Study Selection

Inclusion Criteria	Exclusion Criteria
IC1: Study must consider at least 1 task scheduling algorithm intended to be used in Cloud Computing	**EC1:** No task scheduling algorithm is reviewed in the paper
IC2: Study must be published between the years 2020-2021 included	**EC2:** Study is published before 2020 or after 2021
IC3: Language of the selected study must be English	**EC3:** Language of the study is not English
IC4: The study must be peer reviewed and must be published in either a conference or a journal	**EC4:** Study is not peer reviewed such as grey literature.
IC5: The study must be published in a well-reputed journal or conference	**EC5:** Study is a secondary study such as SLR and SM.

3.2.2 Study Quality Assesment

Once the final pool of candidate studies is formed by applying inclusion and exclusion criteria given in Table 6, it is critical to evaluate the quality and trustworthiness of these studies. This is the final evaluation step of search protocol and marks the beginning of review phase. Kitchenham and Charters[3] state that quality assessment can be used in two different cases. It can be used either for further eliminating candidate papers or for supporting data analysis.

Step 4 is applied in order to evaluate the relevancy of the studies. Papers that are unrelated with our research questions, research context or off-topic are filtered at this step. Study quality assessment is applied for deciding whether relevant studies are of good quality in terms of the clear statements given in paper such as problem definition, future study recommendations etc.

To evaluate the quality of the retrieved studies, an assessment criterion is formed. The questions given in this criterion is asked for each primary study and answers are noted down. Study quality assessment criteria is given in Table 7.

Table 7 - Quality Assessment Criteria

Quality Assessment Questions		
Q1	Is the problem definition stated clearly?	
	A1	Problem definition doesn’t exist.
	A2	Problem definition exists but not clear.
	A3	Problem definition exists and is stated clearly.
Q2	Do all research questions addressed?	
	A1	None of the research questions are addressed.
	A2	Research questions are addressed partially.
	A3	Research questions are addressed fully.
Q3	Do the authors give future study recommendation?	
	A1	No
	A2	Yes
Q4	Is the research methodology explained in detail?	
	A1	No explanation
	A2	Partial Explanation
	A3	Full explanation

It is important to note that quality assessment in this study is intended only to support data extraction and analysis protocols. Thus, quality assessment is done at review step after the final pool of studies is set. In other words, no extraction of studies is done via quality assessment.

3.2.3 Data Extraction

Data extraction is the step which each paper in the pool is reviewed separately to answer research questions of the Meta Synthesis we are conducting. This process must be done in a planned and systematic way in order to use the retrieved information at data synthesis step effectively. For this reason, data extraction forms are designed as suggested by Kitchenham and Charters[3].

Data extraction forms include data items that stores information about reviewed papers. Some of those data items are based on standard information about papers such as authors, publish date and journal/conference information.

Second type of data items are questions that are asked by reviewer while reviewing the paper in order to get answers for research questions.

Third type of data items are based on quality assessment criteria to review the quality aspect of the studies. As stated in 3.2.2, quality assessment is done as a means of data analysis so items related with quality assessment are also stored in the same form with data extraction items. Data extraction form used in this study is given in Table 8.

Table 8 - Data Extraction Form

#	Data Item	Description	Addressed RQ/QA
1	Study ID Tag		-
2	Author(s)		-
3	Study Title		-
4	Publication Year		-
5	Country of Origin		-
6	Publication Type		-
7	Publication Source		-
8	Which category does the algorithm belong to?		RQ 1 RQ 1.1

Table 8 cont.

9	What are the pros and cons of the algorithm according to the authors?		RQ 1.2
10	What is the intended use of the algorithm?		RQ 1.3
11	How is the algorithm's performance evaluated?		RQ 2 RQ 2.1
12	What are the comments of authors on validity and trustworthiness of the evaluation method?		RQ 2.2
13	Is it possible to reevaluate the algorithm with proposed evaluation method and end up with same results?		RQ 2.2
14	What are the future research recommendations of authors?		RQ3 RQ 3.2
15	Which topics need attention according to authors?		RQ 3 RQ 3.1
16	Problem definition stated by authors		QA 1
17	Do all research questions addressed?		QA 2
18	Do the authors give future research recommendation?		QA 3
19	Is the research methodology explained in detail		QA 4

3.2.4 Data Synthesis

Data that is extracted from the reviewed studies don’t have much meaning without synthesis. Once the data extraction forms are filled and all study materials are reviewed, the gathered information is combined and processed in a methodological way to answer research questions and share results of the review.

This study is based on qualitative research methodologies. The aim of the study is to share qualitative findings of other researchers within the context of task scheduling algorithms in cloud computing. We focus more on the comments, findings and interpretations of the primary study authors so qualitative research approach fits better to our study’s data synthesis methodology.

There are several approaches for synthesizing data in a qualitative research study. In this study, data is synthesized with narrative and meta-synthesis approaches. Data extracted from reviewed studies is combined together and categorized qualitatively according to their occurrence frequencies. Narrative synthesis is used mostly to describe the findings of reviewed studies and meta-synthesis is used for deriving general results and new insights from statistics. The meta-synthesis process is finalized by making general outcomes to answer research questions.

The findings are synthesized in a transparent way to summarize the contents of various primary studies. Cruzes et al. [31] states that narrative synthesis approach is one of the most used approaches in review studies of Software Engineering domain. During this study’s synthesis phase, the recommendations of Cruzes et al. [31] for narrative synthesis method are followed.

3.2.4.1 *Meta Synthesis*

Meta Synthesis is a method used for examining qualitative research data. It is different from Meta-Analysis as the aim of Meta-Analysis is to end up with cause and effect relations. However, the main aim of Meta-Synthesis is examining data thoroughly by spotting common themes in a pool of studies in order to understand and explain a topic. [32]

Meta-Synthesis is also different from Systematic Literature Review as the main focus of SLR is to statistically analyze quantitative data within a pool of studies. However, qualitative Meta-Synthesis aims to answer questions of “how” and “why”. Outcome of Meta-Synthesis studies is critical for policy makers as Meta-Synthesis studies are more informative and explorative compared with SLR and Meta-Analysis studies. [33].

One of the most critical aspect of Meta-Synthesis is that it is more than summarization of a pool of studies. Meta-Synthesis studies aims to interpret the findings of basic SLR and propose new views and perspectives on a topic. [34]

The aim of this study is to provide an insight for new researchers on Task Scheduling issue in Cloud Computing field which is best supported by Meta-Synthesis. Meta-Synthesis is transparent in nature, which means findings are extracted and reported from the pool of studies as is and the vision is created from clean data. [35] This is what we exactly want as researchers as we want to provide an insight on current issues and trends without altering the reality.

Qualitative Meta-Synthesis began being used in Medical researches then found its way into Education field. Following this trend, it has gained attention of Information Systems researchers. It is supported that IS domain can benefit hugely from the use of Meta-Synthesis in studies. [36] In many articles supporting the use of Meta-Synthesis in IS domain, it is mentioned that Meta-Synthesis is more than standard literature review.

CHAPTER 4

4 RESULTS

In this chapter, the gatherings of search protocol and review process is discussed. Initial database search through 5 databases has resulted in 1262 papers. The search was done through ACM, IEEE Xplore, ScienceDirect, Scopus and WebOfScience digital databases. The search results are exported and saved to Mendeley Reference Manager tool. Retrieved papers are inspected for duplicates by using Mendeley Desktop "Check for Duplicates" function. 616 duplicate papers were found in total and removed from the library. The initial pool of papers contained 646 papers at this step. These were the papers eligible for application of inclusion/exclusion criteria on. Once inclusion/exclusion criteria are applied, 529 more papers are excluded and number of papers in the pool is decreased to 88. After this point, we began reviewing the papers more detailly and extracted necessary data. During this process, 17 more papers were found irrelevant in terms of context and excluded from the final pool of studies. All this search and review process ended up with 71 final papers which made our final pool of reviewed studies. Figure 5 shows the study filtration process.

Detailed investigation on reviewed papers showed that 5 papers which are published in 2020 are actually papers whose manuscripts are sent to the corresponding journals in 2019. These papers are revised in 2020 but may hold information that doesn't consider the effects of the pandemic as WHO declared COVID-19 as Public Health Emergency of International Concern on 30 January 2020. This is a minor deviation and won't affect the general outcomes.

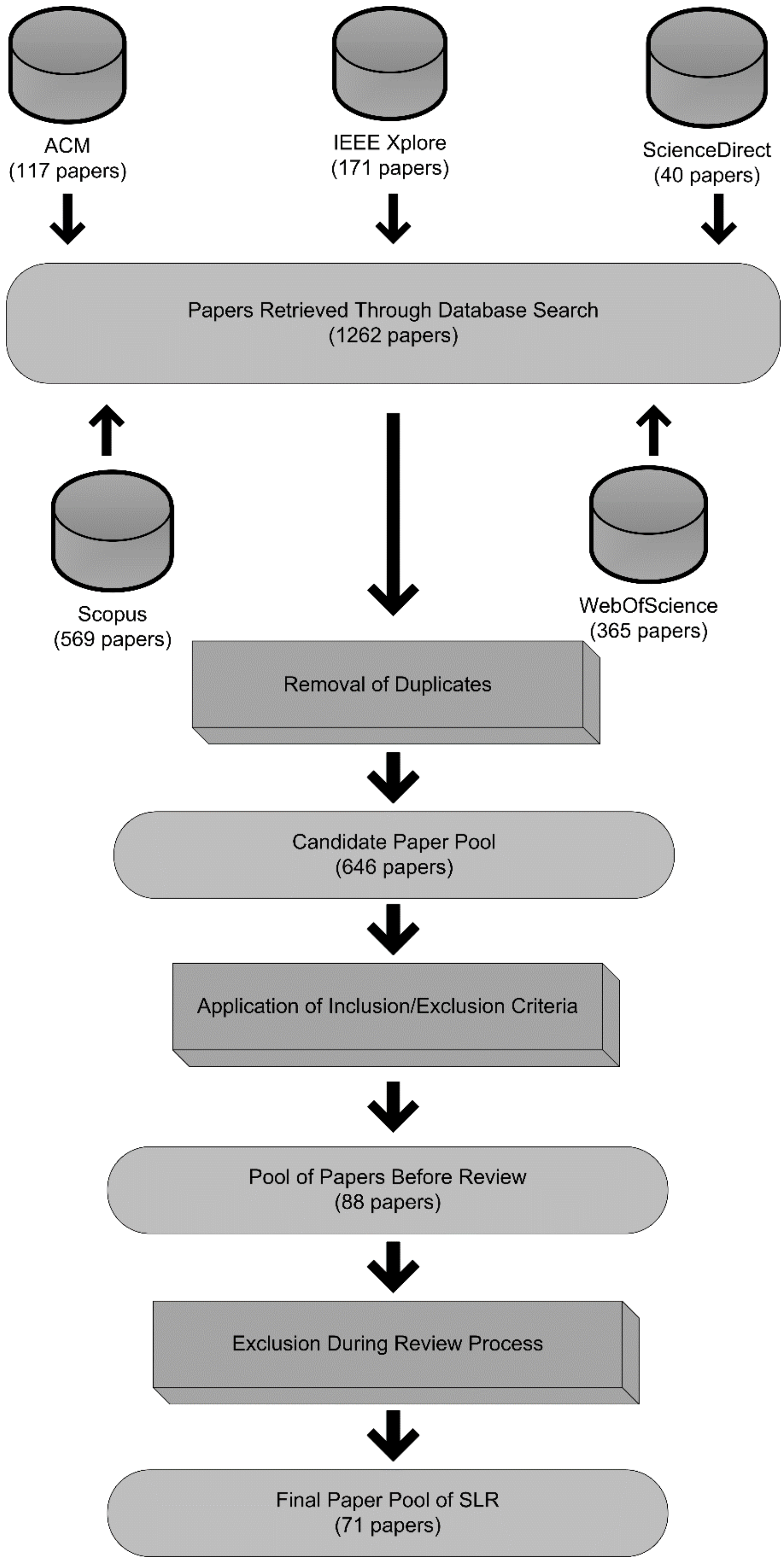

Figure 5 - Study Retrieval Process

4.1 Bibliometric Analysis

This section consists of bibliometric analysis done for reviewed 71 papers. Year wise distribution shows that more studies were published in 2020 than in 2021 as given in Figure 6.

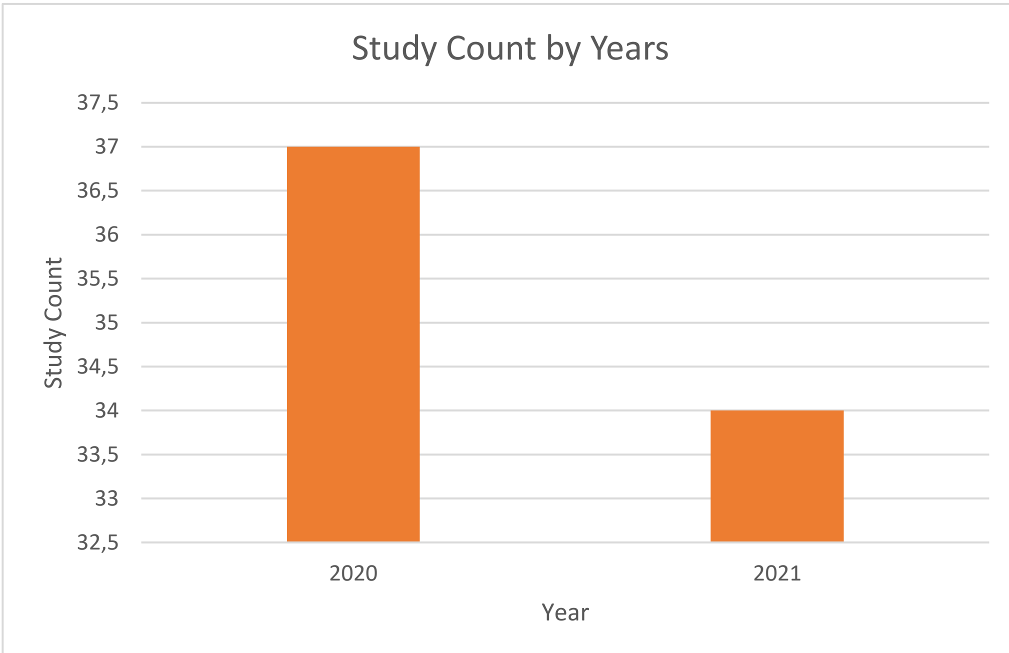

Figure 6 - Year-wise Distribution of Publications

Impact of countries on the literature are derived by noting down the origins of the publications. Origins of the authors' universities are chosen as the origin of studies. If a publication was done by multiple authors, then each one's origins are counted as one and recorded. However, if origins of multiple authors of a study were the same country, it was counted as one. Top 3 most influential countries are China, India and Iran. China has the most impact in literature by influencing 35,21% of the reviewed studies. It is followed by India with a percentage of 29,58% and by Iran with 12,68%. The least influential countries are Algeria, Syria, Australia, South Africa, Sri Lanka, UAE, Tunisia, Taiwan, Poland and Pakistan with a percentage of 1,41%. Figure 7 shows the distribution of studies by origin countries in terms of percentage and Figure 8 shows the same distribution in terms of study count.

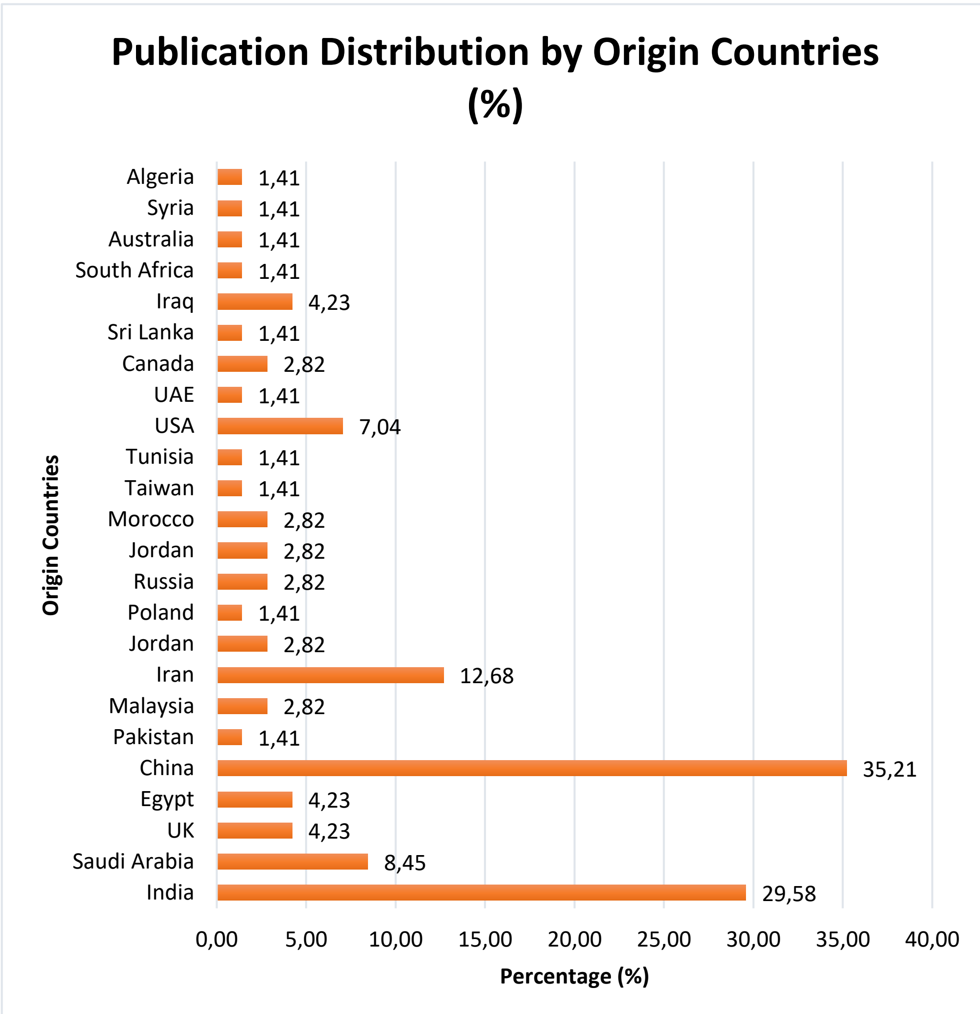

Figure 7 - Impact of Countries on Studies by Percentage

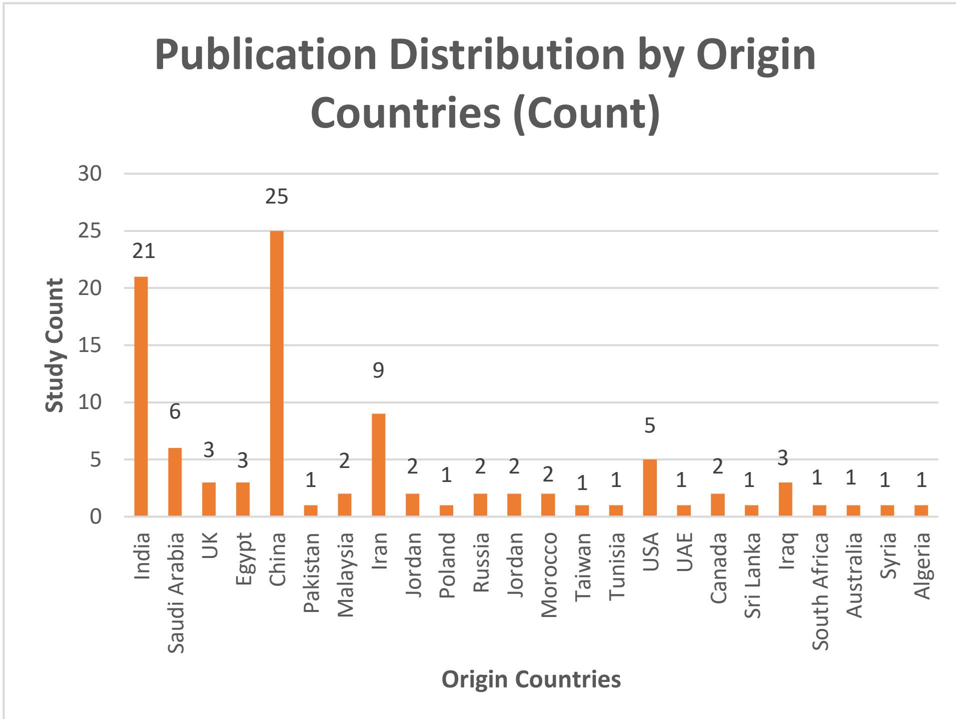

Figure 8 - Impact of Countries on Studies by Count

Top performing journals were Cluster Computing, J. Supercomputing, Journal of Ambient Intelligence and Humanized Computing, International Journal of Communication Systems, Journal of King Saud University - Computer and Information Sciences, Soft Computing and Future Generation Computer Systems. Only 1 study was included from other journals so they are not mentioned separately.

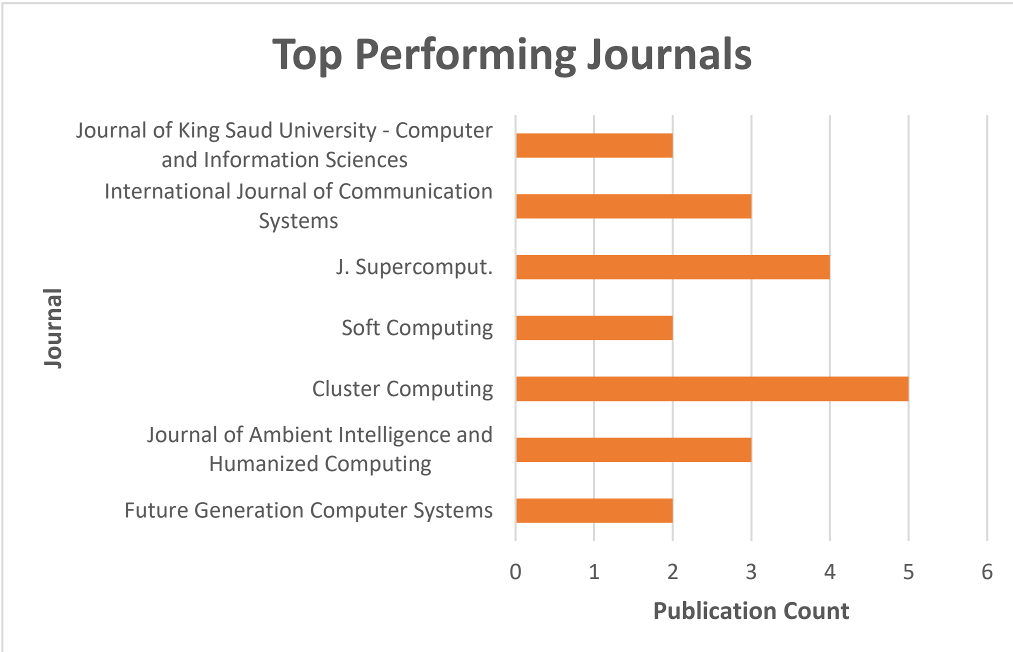

Figure 9 - Top Performing Journals

48 of the reviewed publications are Journal papers and 23 are Conference Proceedings. In percentage, this distribution corresponds to 67,61% Journal papers and 32,89% Conference Proceedings.

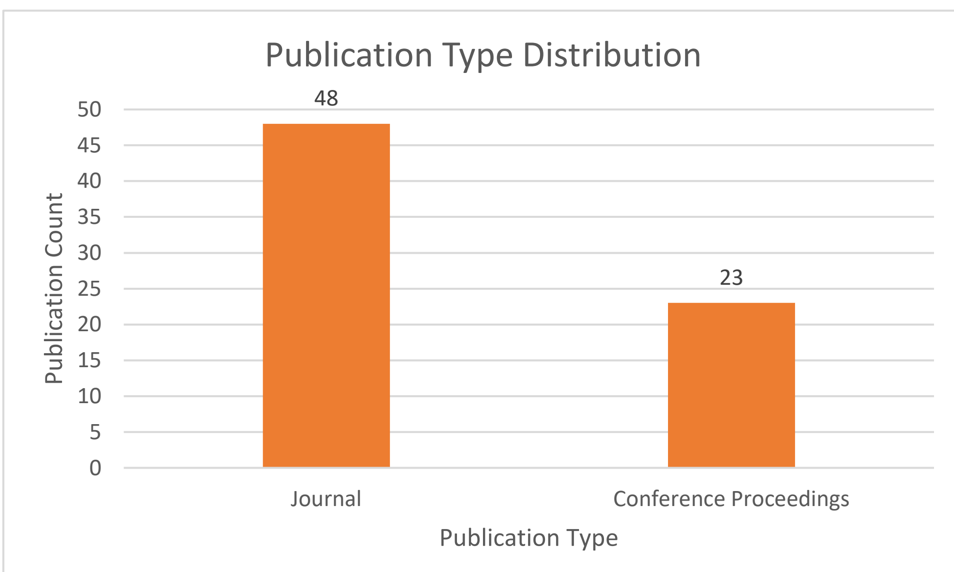

Figure 10 - Publication Type Distribution

4.2 Study Quality Assessment

Quality assessment results for reviewed studies are given in the charts below. Charts show the percentages of the answers given for assessment questions.

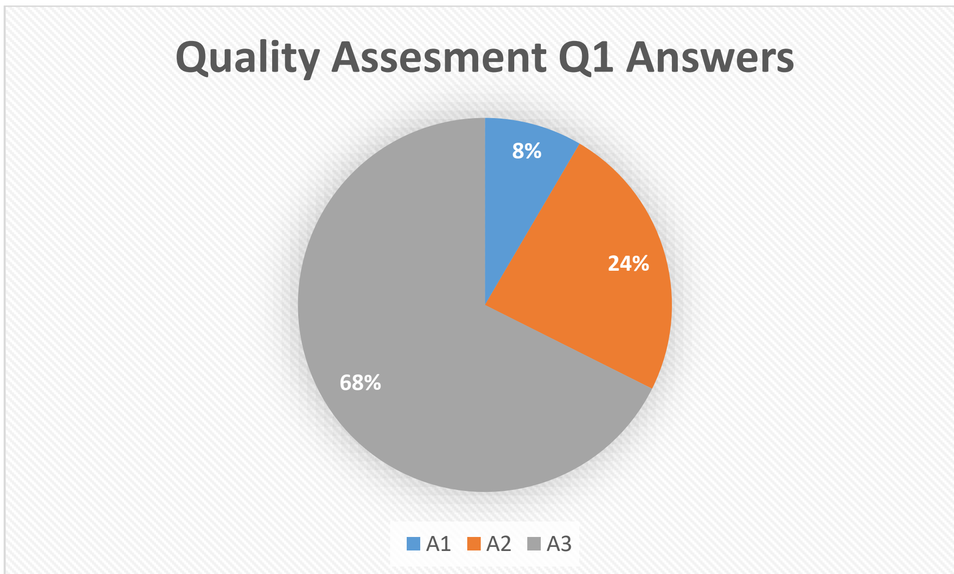

Figure 11 - Quality Assessment Q1 Answer Percentages

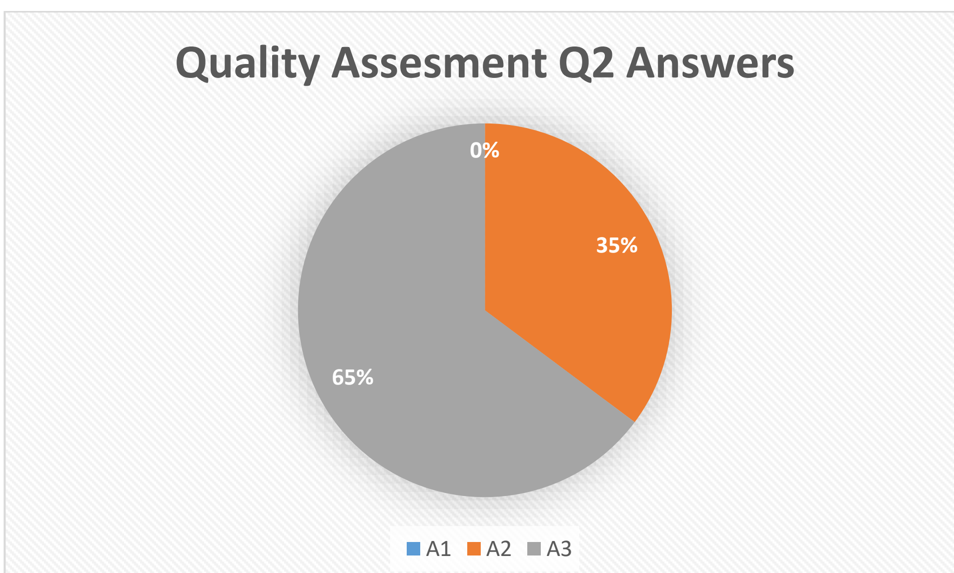

Figure 12 - Quality Assessment Q2 Answer Percentages

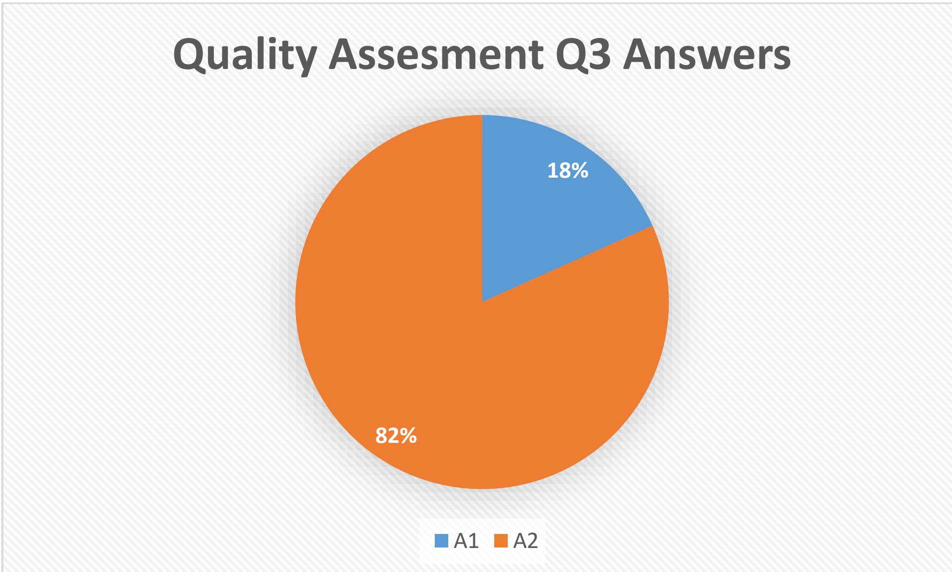

Figure 13 - Quality Assessment Q3 Answer Percentages

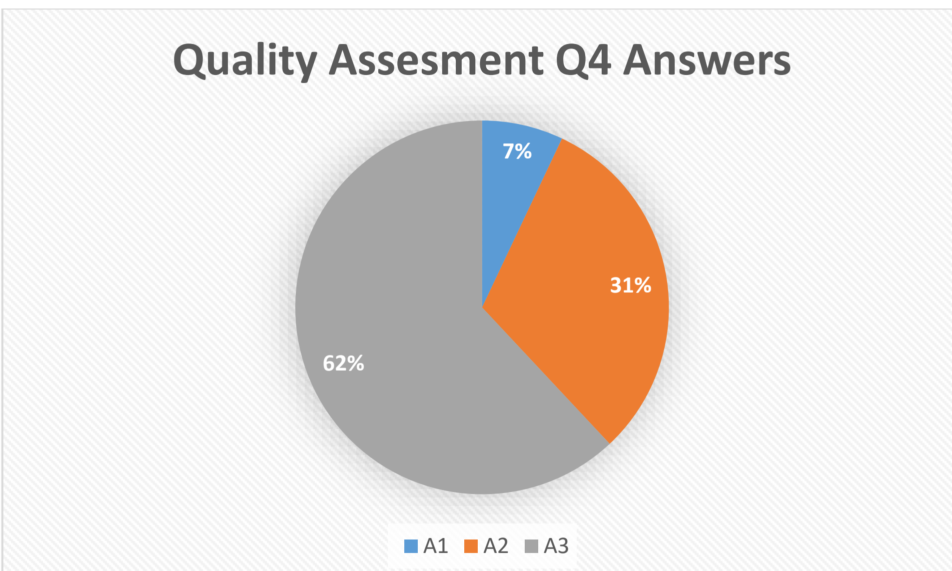

Figure 14 - Quality Assessment Q4 Answer Percentages

4.3 Findings

In this section of our study, answers to research questions are given by synthesizing the extracted data using qualitative methods. By using narrative and thematic analysis approaches, findings of the Meta-Synthesis study are stated. Findings are categorized under research questions. Each subsection gives answers to a specific research question.

4.3.1 RQ 1 – How are task scheduling algorithms for cloud computing environment categorized?

4.3.1.1 *Review of studies*

There is no common agreement among the authors of reviewed studies in terms of algorithm categorization. In some studies, it is clearly stated how scheduling algorithms in Cloud Computing are categorized. Some of them don't state it clearly but the categorization can be inferred from the mentioning. However, some studies neither make any categorization nor state the category of the proposed algorithm. 32,39% (n=23) studies don't state any categorization scheme for task scheduling algorithms in Cloud Computing whereas 67,61% (n=48) studies state a scheme for categorization of task scheduling algorithms.

Availability of general categorization schemes across reviewed studies is given in Table 9.

Table 9 - General Algorithm Categorization Availability Across Studies

No Categorization Available	Categorization Available
[S11], [S12], [S14], [S16], [S17], [S18], [S19], [S20], [S21], [S26], [S27], [S32], [S36], [S43], [S44], [S45], [S50], [S51], [S55], [S59], [S66], [S67], [S71]	[S1], [S2], [S3] [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S13], [S15], [S22], [S23], [S24], [S25], [S28], [S29], [S30], [S31], [S33], [S34], [S35], [S37], [S38], [S39], [S40], [S41], [S42], [S46], [S47], [S48], [S49], [S52], [S53], [S54], [S56], [S57], [S58], [S60], [S61], [S62], [S63], [S64], [S65], [S68], [S69], [S70]

It is noticed that some authors state the type of their proposed algorithm even though they don't mention any general categorization scheme. In 22,54% (n=16) of the reviewed studies, the type of the proposed algorithm isn't stated. Whereas in 77,46% (n=56) of the studies, a type or classification for the proposed algorithm is stated clearly.

Availability of specific categorization across reviewed studies is given in Table 10.

Table 10 - Specific Algorithm Categorization Availability Across Studies

No Categorization Available	**Categorization Available**
[S11], [S12], [S14], [S16], [S19], [S32], [S34], [S36], [S43], [S44], [S45], [S51], [S53], [S66], [S67], [S71]	[S1], [S2], [S3], [S4], [S5], [S6], [S7], [S8], [S9], [S10], [S13], [S15], [S17], [S18], [S20], [S21] [S22], [S23], [S24], [S25], [S26], [S27], [S28], [S29], [S30], [S31], [S33], [S35], [S37], [S38], [S39], [S40], [S41], [S42], [S46], [S47], [S48], [S49], [S50], [S52], [S54], [S55], [S56], [S57], [S58], [S59], [S60], [S61], [S62], [S63], [S64], [S65], [S68], [S69], [S70]

4.3.1.2 *Meta Synthesis and Thematic Analysis*

At this part, we have done meta synthesis of the reviewed studies to propose a categorization scheme for task scheduling algorithms in cloud computing. In the previous part, we have grouped studies in terms of whether they include a categorization scheme or not. Here we have synthesized the studies to locate common themes for categorization of cloud task scheduling algorithms.

It is safe to state that there is no single type of categorization for task scheduling algorithms exists however, there are common patterns in the chosen categorization types of authors. One popular type of categorization is grouping algorithms as either "**heuristic**" or "**meta-heuristic**" algorithms. In 47,89% (n=34) of the reviewed studies, it is stated that one kind of categorization scheme is heuristic/meta-heuristic categorization. The algorithms that fall into these categories stated by the authors are consistent across studies. However, it is noticed in some studies that some authors have chosen to group meta-heuristic and heuristic algorithms under a single category. By doing so, they have called this group of algorithms simply as "heuristic" algorithms. 8,45% (n=6) of studies didn't make this diversification and simply called meta-heuristic algorithms as heuristics.

Another popular categorization scheme is grouping algorithms as "**traditional**" algorithms. The studies which incorporate traditional grouping in their categorization schemes tend to use it in conjunction with heuristic, including meta-heuristic, algorithms. 7,04% (n=5) of studies consider traditional algorithms as a category. 1,41% (n=1) of studies considered the category "**classical**" algorithms which covers the same algorithms as traditional algorithms. This is actually an interchangeable term for traditional algorithms.

"**Evolutionary**" algorithms is another major category mentioned in studies. This kind of algorithms are structured under heuristic, specifically meta-heuristic if the authors used this term, algorithms. This group of algorithms represent algorithms based on Genetic Algorithm (GA), Differential Evolution Algorithm (DE). 7,04% (n=5) of the studies considered evolutionary category.

1,41% (n=1) of studies mention "**swarm intelligence based**" algorithms which is actually positioned as a subcategory of meta-heuristic algorithms. This category includes algorithms such as Particle Swarm Optimization and Genetic Algorithm according to Shi et. al. [S61]

Another category which is not mentioned in most studies is "**machine learning based**" algorithms. 1,41% (n=1) of studies consider this category in task scheduling categorization scheme. 1,41% (n=1) of studies consider "**game theory based**" algorithms as a separate category.

1,41% (n=1) of studies mention another categorization scheme as "**nature-inspired**" and "**conventional**" algorithms. Kumar et. al. [S68] mentioned this scheme in their study. According to them, nature inspired algorithms is a subcategory of meta-heuristic algorithms and conventional algorithms cover all other type of algorithms.

"**Hybrid**" algorithms is another category mentioned by 9,86% (n=7) of the studies. This categorization is used in conjunction with heuristic/meta-heuristic categorization scheme and represents algorithms that demonstrate characteristics of, or based on, both heuristic and meta-heuristic algorithms.

4.3.1.3 *Proposed Categroization Schemes*

Depending on the problem-solving approach of the algorithms, a categorization scheme can be established with 5 main groups; Traditional(classical), Heuristics based, Game Theory based and Machine Learning based algorithms. The subcategories of this scheme can be seen in Figure 15.

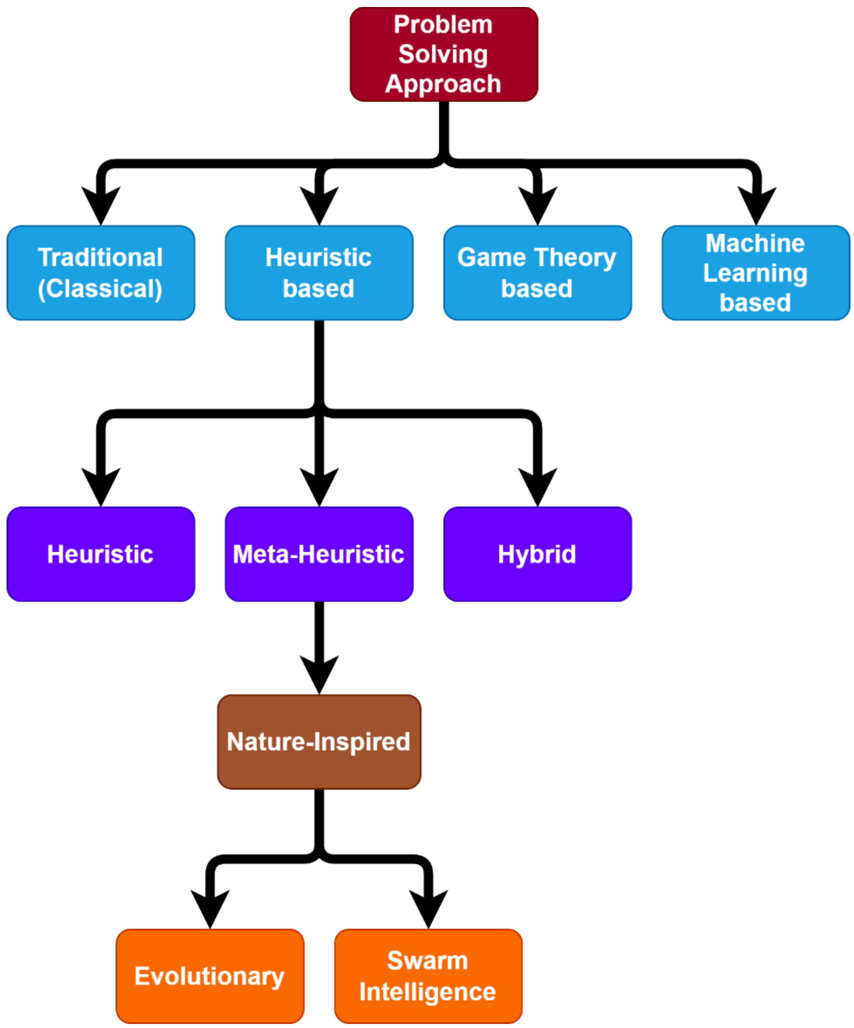

Figure 15 - Algorithm Categorization based on Problem-solving Approach

Another common pattern found in the studies for categorizing algorithms is grouping them depending on their resource mapping strategy. Most studies have considered grouping task scheduling algorithms under 2 categories, which are called "**static**" or "**dynamic**" algorithms. 11,27% (n=8) of the reviewed studies mentioned static/dynamic categorization as a way to categorize task scheduling algorithms.

According to Fellir et. al. [S62], the main difference between static and dynamic scheduling algorithms is the environmental considerations they make during scheduling decisions. Static algorithms consider only the initial environmental conditions such as VM cluster state and VM workloads once a task arrives at the scheduler. The changing conditions of the cluster isn't considered after a task is allocated. However, dynamic algorithms actively check the conditions of VM cluster and make adjustments on the allocation scheme during the execution of tasks. It is inferred from these studies that being static or dynamic is correlated with how the algorithm maps the resources of the cloud environment. From these findings, we decided to call this categorization scheme as "categorizing depending on the resource mapping strategy". According to this scheme, algorithms can either be static or dynamic. This categorization scheme is visualized in Figure 16.

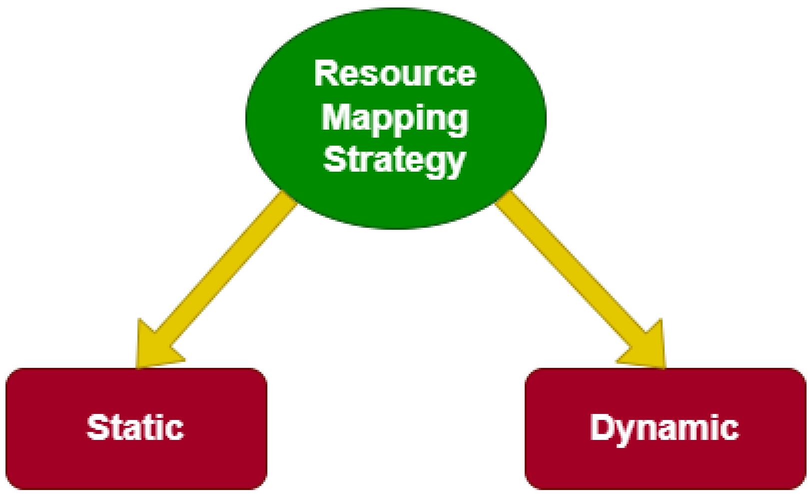

Figure 16 - Proposed Categorization Scheme Based on Resource Maping Strategy

Sanaj et. al. [S34] mentioned categorizing algorithms depending on how they execute the tasks. Algorithms can be either "**centralized**" or "**decentralized**" depending on where the algorithm is executed. 1,41% (n=1) of studies considered centralized/decentralized categorization scheme. This categorization scheme is visualized in Figure 17.

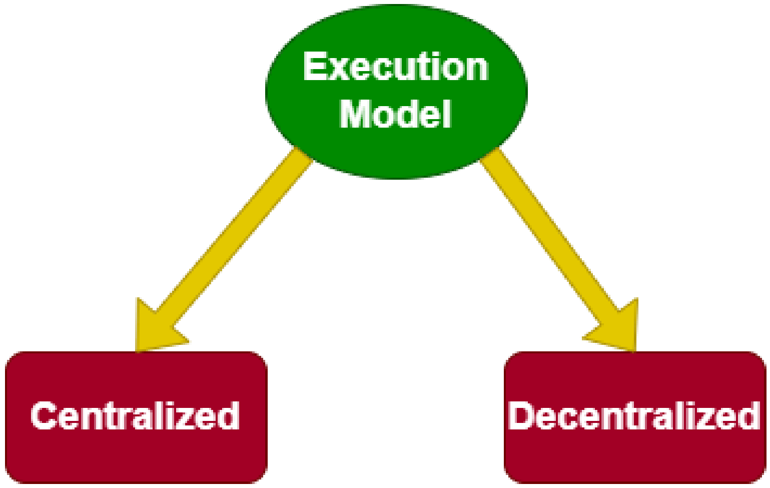

Figure 17 - Proposed Categorization Scheme Based on Execution Model

2,82% (n=2) of the studies considered classifying algorithms depending on the scheduling goals such as QoS considerations. According to this scheme, algorithms can be categorized either as "**multi-objective**" or "**single-objective**". Guo et. al. [S6] states that single-objective algorithms consider only a single scheduling goal during execution whereas multi-objective ones consider different goals during execution. These goals can be reducing computational costs, reducing energy consumption, reducing makespan, increasing resource utilization etc. This categorization scheme is visualized in Figure 18.

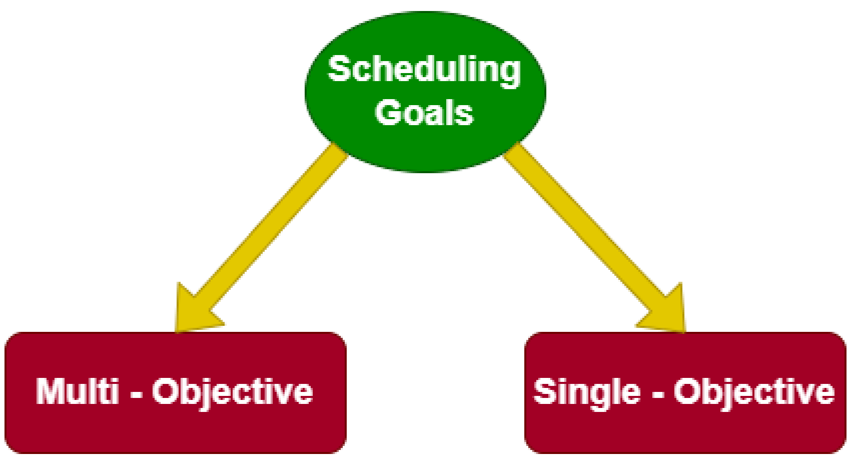

Figure 18 - Proposed Categorization Scheme Based on Scheduling Goals

One different categorization scheme is classifying algorithms depending on when they make allocation decisions for tasks. In this scheme, algorithms can be either "**batch**" or "**instantaneous**". Instantaneous algorithms can also be referred to as "**online**" or "**immediate**". Wilczyński et. al. [S35] states that batch mode algorithms group tasks once they arrive at scheduler and make scheduling decisions for a group of tasks. However, instantaneous mode algorithms schedule tasks as soon as they arrive at scheduler. 2,82% (n=2) of the studies consider batch/instantaneous categorization scheme. Figure 19.

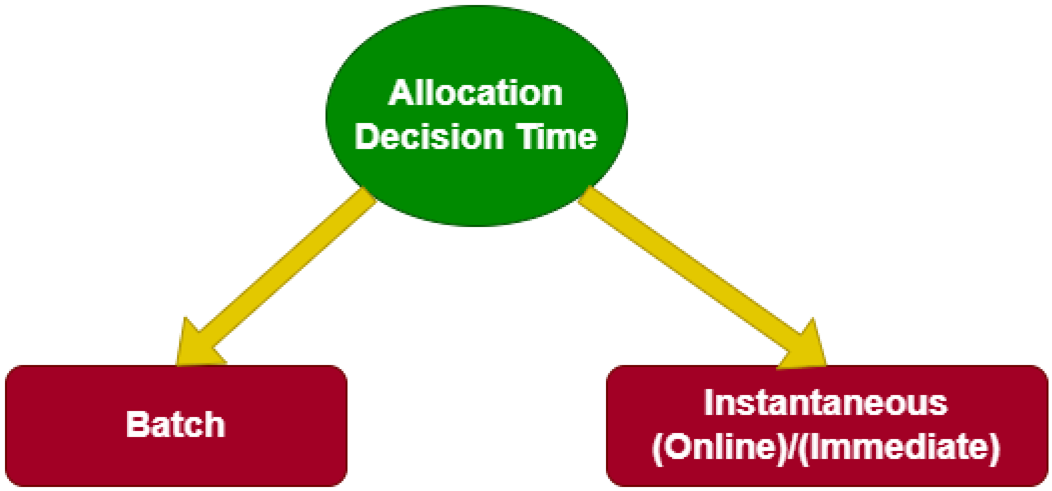

Figure 19 - Proposed Categorization Scheme Based on Allocation Decision Time

By this synthesis, we located common themes and patterns used in categorization of algorithms and generalized the categorization scheme of cloud task scheduling algorithms. We propose the following generalized categorization scheme to classify cloud task scheduling algorithms as given in Table 11. We have also classified the algorithms using our scheme and grouped them. This information can be used as a reference for checking how each category of algorithm behave. General categorization scheme proposed by us is given in Figure 20.

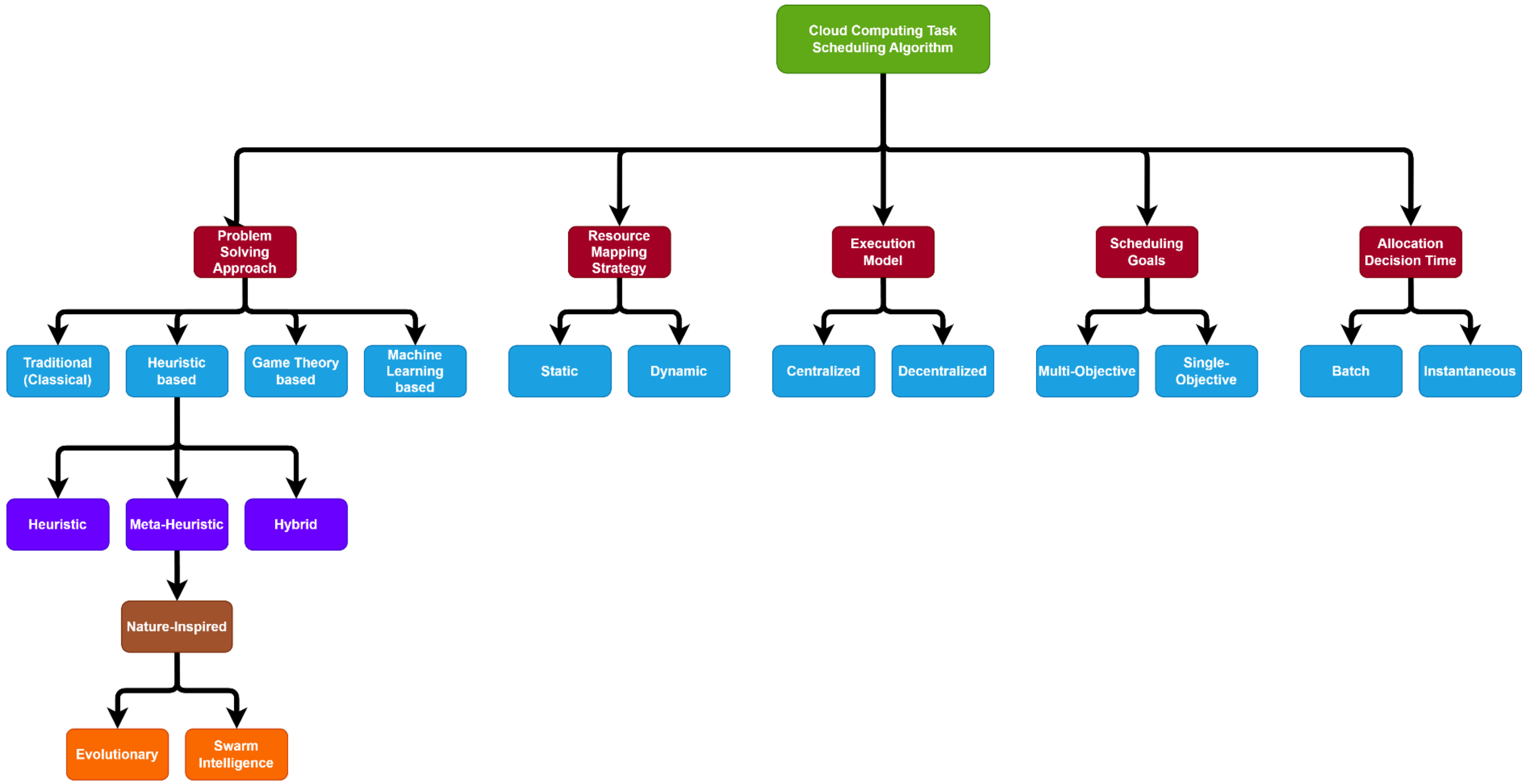

Figure 20 - Proposed Categorization Scheme

Categorization scheme distribution across whole reviewed studies is given in Figure 21.

Table 11 - Algorithm Categorization Schemes of Reviewed Studies

Categorization Scheme	Categories	Studies
Depending on problem solving approach	Traditional Heuristics Game Theory based Machine Learning based	[S1], [S2], [S3], [S4], [S7], [S8], [S9], [S10], [S15], [S22], [S23], [S24], [S28], [S29], [S30], [S31], [S33], [S34], [S37], [S38], [S39], [S40], [S46], [S47], [S48], [S49], [S52], [S53], [S54], [S56], [S57], [S58], [S60], [S61], [S63], [S65], [S68], [S69], [S70]
Depending on resource mapping strategy	Static Dynamic	[S5], [S13], [S25], [S41], [S42], [S62], [S64], [S70]
Depending on execution model	Centralized Decentralized	[S34]
Depending on scheduling goals	Multi-objective Single-objective	[S6], [S47]
Depending on allocation decision time	Batch mode Instantaneous mode	[S34], [S35]
NA	NA	[S11], [S12], [S14], [S16], [S17], [S18], [S19], [S20], [S21], [S26], [S27], [S32], [S36], [S43], [S44], [S45], [S50], [S51], [S55], [S59], [S66], [S67], [S71]

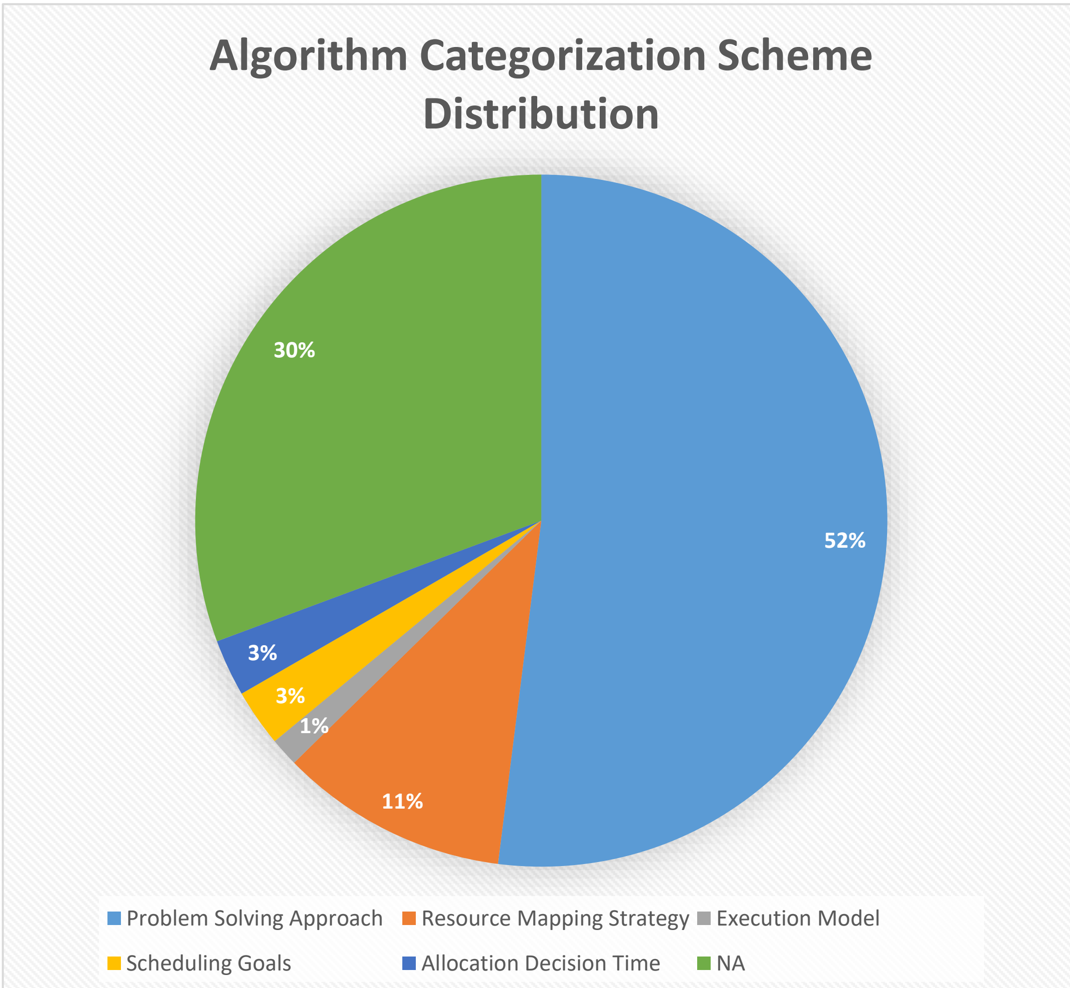

Figure 21 - Algorithm Categorization Scheme Distribution Across Reviewed Studies

4.3.2 RQ 1.1 – What is the percentage of algorithms in literature that fall in each category?

Among the reviewed algorithms, 9,86% (n=7) of them was categorized as heuristic by their developers. However, 5 out of 7 are actually based on meta-heuristic algorithms. The authors of studies [S50], [S54], [S55], [S57] and [S65] chose to categorize meta-heuristic and non meta-heuristic algorithms in a single category as heuristic algorithms. This case was explained in 4.3.1. 35,21% (n=28) of the algorithms are categorized as meta-heuristic. 9,86% (n=7) of them was categorized as hybrid. However, among the algorithms categorized as hybrid, 1 of them was hybridization of meta-heuristic and machine learning methods [S28] and 1 of them was hybridization of meta-heuristic and traditional algorithms [S49].

7,04% (n=5) of the proposed algorithms are categorized as evolutionary algorithms. 1,41% (n=1) of the algorithms are Game Theory based, 1,41% (n=1) of the algorithms are Machine Learning based and again 1,41% (n=1) of them are Nature inspired.

7,04% (n=5) algorithms are classified as dynamic. 1,41% (n=1) algorithms are classified as semi-dynamic and 1,41% (n=1) of the algorithms are classified as semi-static. This type of algorithms can be basically called hybrid(dynamic/static).

1,41% (n=1) of the algorithms are tagged as batch mode algorithms.

1,41% (n=1) of the algorithms are tagged as multi-objective algorithms.

Finally (n=16) of the studies didn't specifically mention a category for their proposed algorithms. It is highly possible that the reviewed algorithms might offer characteristics of other algorithm categories. In other words, an algorithm mentioned as meta-heuristic might also be a dynamic algorithm in terms of allocation decision time categorization. However, it might introduce errors in our study if we try to guess other aspects of algorithms. For this reason, categories of algorithms are shared in an objective manner.

Table 12 - Categorization Schemes Adopted in Studies

Categorization Scheme	Categories	Studies
Depending on problem solving approach	Nature Inspired	[S68]
	Heuristic	[S1], [S40], [S50], [S54], [S55], [S57], [S65]
	Meta-Heuristic	[S2], [S3], [S4], [S5], [S10], [S15], [S17], [S18], [S20], [S21], [S22], [S23], [S24], [S26], [S29], [S30], [S31], [S33], [S38], [S39], [S47], [S52], [S58], [S63], [S69]
	Game Theory based	[S59]
	Machine Learning based	[S27]

Table 12 cont.

	Evolutionary	[S46], [S48], [S56], [S60], [S61]
	Hybrid	[S7], [S8], [S9], [S28], [S37], [S49], [S70]
Depending on resource mapping strategy	Dynamic	[S5], [S13], [S25], [S62], [S64]
	Hybrid	[S41], [S42]
Depending on scheduling goals	Multi-objective	[S6]
Depending on allocation decision time	Batch mode	[S35]
NA	NA	[S11], [S12], [S14], [S16], [S19], [S32], [S34], [S36], [S43], [S44], [S45], [S51], [S53], [S66], [S67], [S71]

4.3.2.1 *Meta Synthesis of Common Base Algorithms*

It is noticed in reviewed studies that nearly all of the proposed algorithm are based on other well-known algorithms. However, 9,86% (n=7) of the studies didn't mention the base algorithms. From the available information, we analyzed the most frequently used algorithms; 15,49% (n=11) algorithms are based on "Genetic Algorithm (GA)", 11,27% (n=8) algorithms are based on "Ant Colony Optimization (ACO)", 11,27% (n=8) algorithms are based on "Particle Swarm Optimization (PSO)", ", 8,45% (n=6) algorithms are based on "Whale Optimization Algorithm (WOA)", 4,23% (n=3) algorithms are based on "Artificial Bee Colony Algorithm (ABC)", 4,23% (n=3) algorithms are based on "Differential Evolution Algorithm (DE)", 2,82% (n=2) algorithms are based on "Crow Search Optimization (CSA)", 2,82% (n=2) algorithms are based on "Bat Algorithm, 2,82% (n=2) algorithms are based on "Symbiotic Organisms Search Algorithm (SOS)" and 2,82% (n=2) algorithms are based on "Game Theory". GA is the most influential algorithm in reviewed literature.

Top performing algorithms' distribution and number of studies that used these algorithms can be seen in Figure 22. Full list of used algorithms versus respectful studies is given in Table 13

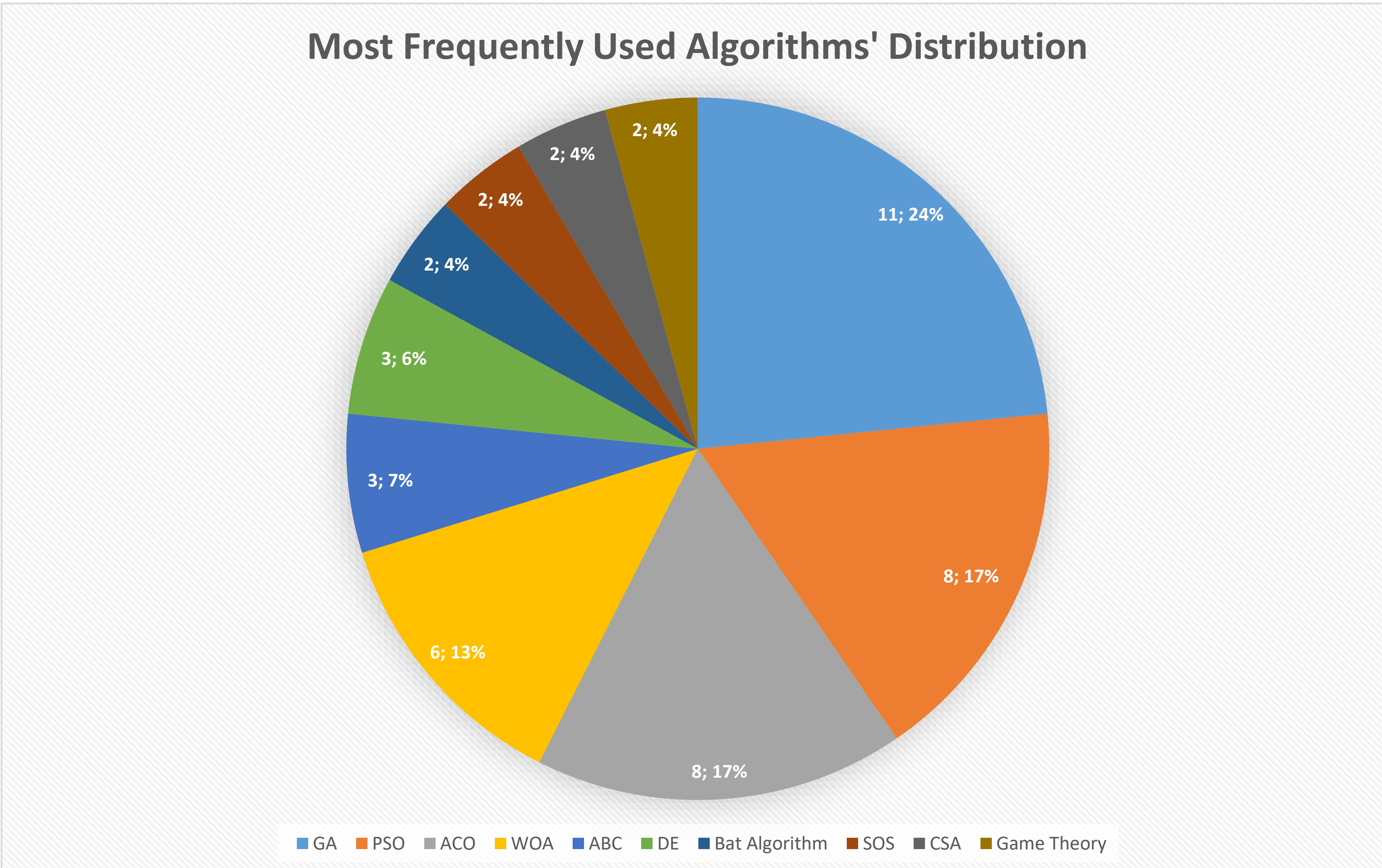

Figure 22 - Most Frequently Used Algorithms

Table 13 - Algorithm Bases of Studies

Base Algorithm	Abbreviation of Algorithm in Literature	Relevant Studies
Genetic Algorithm	GA	[S2], [S5], [S10], [S12], [S22], [S28], [S29], [S33], [S34], [S48], [S57]
Particle Swarm Optimization Algorithm	PSO	[S4], [S7], [S15], [S33], [S46], [S49], [S60], [S70]
Ant Colony Optimization Algorithm	ACO	[S7], [S8], [S11], [S14], [S30], [S54], [S57], [S65]
Whale Optimization Algorithm	WOA	[S17], [S18], [S21], [S24], [S32], [S34]
Artificial Bee Colony Algorithm	ABC	[S9], [S47], [S55]
Differential Evolution Algorithm	DE	[S50], [S56], [S61]
Bat Algorithm	-	[S8], [S9]
Symbiotic Organisms Search Algorithm	SOS	[S63], [S69]
Crow Search Algorithm	CSA	[S38], [S68]
Game Theory	-	[S41], [S42]
Penguin Search Optimization Algorithm	PeSOA	[S38]
Water Cycle Algorithm	WCA	[S19]
Bumble Bee Mating Algorithm	BBMO	[S52]
Sunflower Optimization Algorithm	SFO	[S18]
Squirrel Search Algorithm	SSA	[S39]

Table 13 cont.

Salp Swarm Algorithm	SSA	[S3]
Round Robin Algorithm	RR	[S1]
Reference Vector Guided Evolutionary Algorithm	RVEA	[S16]
Chemical Reaction Optimization Algorithm	CRO	[S4]
Grey Wolf Optimization Algorithm	GWO	[S15]
Markov Decision Process	-	[S66]
Imperialist Competitive Algorithm	ICA	[S58]
Firefly Algorithm	FA	[S58]
Hungarian Algorithm	HA	[S36]
Honey Bee Behavior Based Load Balancing Algorithm	HBB-LB	[S64]
Henry Gas Solubility Algorithm	HGSO	[S21]
Gravitational Search Algorithm	GSA	[S43]
Thermodynamic Simulated Annealing Algorithm	TSA	[S22]
Majority Voting Ensemble	MVE	[S28]
Electro Search Algorithm	-	[S5]
A Mixed Integer Linear Formulation	MILP	[S37]
Fuzzy Self-Defense Algorithm	FSDA	[S6]
Flower Pollination Algorithm	FPA	[S2]

Table 13 cont.

Fair-based Reinforcement Learning Model	-	[S45]
Dynamic Voltage Frequency Scaling Algorithm	DVFS	[S71]
Last Level Cache	LLS	[S71]
Ant Lion Optimization Algorithm	ALO	[S26]
Deep Reinforcement Learning Algorithm	-	[S27]
Deep Q Learning Algorithm	-	[S53]
Decision Tree Model	-	[S51]
Penguin Search Optimization Algorithm	PeSOA	[S38]
Best Worst Method	BWM	[S67]
Technique for Order Preference by Similarity to Ideal Solutions	TOPSIS	[S67]
Bandwidth Aware Divisible Task Scheduling Optimization Algorithm	BATS	[S31]
BAR Optimization Algorithm	BAR	[S31]
Balanced Scheduling Algorithm	BSA	[S59]
Artificial Ecosystem Based Optimization Algorithm	AEO	[S3]

4.3.3 RQ 1.2 – What are the advantages and disadvantages of each algorithm category?

4.3.3.1 *Meta Synthesis of Advantages and Disadvantages of Derived Categories*

In most studies, proposed algorithms are evaluated by making comparisons with other algorithms in same simulation environment. In such cases, mostly the benefits and advantages of the proposed algorithms are shared but not their disadvantages. In 97,18% (n=69) studies, the advantages of the algorithms are clearly stated whereas only in 32,39% (n=23) studies the disadvantages are clearly stated. However, it is noticed that authors tend to comment on the advantages and disadvantages of general algorithm types.

In study [S1], the main focus is the disadvantages of traditional Round Robin algorithm. Authors mention that the traditional RR algorithm is underperforming because of miscalculations of Quantum Time (QT). [S1] proposes an enhanced version of RR algorithm to overcome this issue. Evaluation results show that traditional algorithms have better average waiting time but worse average response time compared with the proposed algorithm.

It is mentioned in [S2] that traditional algorithms are not capable of handling task scheduling problems in cloud environment due to the distributed and virtualization-based nature of cloud. Traditional algorithms are old and designed for solving localized task scheduling problems. Usage of Meta-Heuristic algorithms is enforced and stated that Meta-Heuristic algorithms offer better and optimal solutions in cloud environment.

Tasks can be categorized as independent and dependent. Study [S4] focuses on scheduling of independent tasks and states that Heuristic algorithms are not capable of scheduling independent tasks in cloud. Authors clearly stated that Meta-Heuristic based solutions are the best approach for scheduling independent tasks. One reason is that Meta-Heuristic algorithms don’t need pre-information of resources and tasks. According to them, other benefits of Meta-Heuristic approaches over Heuristic ones are low cost and high energy efficiency. It is also stated in [S4] that PSO algorithm is so popular in cloud task scheduling field and is easy to implement compared with other Meta-Heuristic algorithms. However, the downsides of Meta-Heuristic algorithms are also mentioned in this study. It is stated that Meta-Heuristic approaches tend to converge to local minima and experience premature convergence problem, which results in non-optimal solutions. Study [S5] also mentions this downside of Meta-Heuristic algorithms and states that Genetic Algorithm offers the best local optima solution but underperforms while providing the global optima solution.

Study [S15] states the exact same opinion as [S4] that Meta-Heuristics are better and PSO is the most popular choice. [S46] is another study that promotes PSO and further supports that PSO is the simplest evolutionary algorithm which is easily implementable.

When Multi-Objective algorithms are considered, most algorithms suffer from high maximum completition time, high deadline violation rate and low resource utilization according to study [S6].

[S7] and [S10] are other studies which clearly state that Meta-Heuristic algorithms perform better in cloud environment. In [S7] it is mentioned that Heuristic methods offer the best possible optimal solution but it is not possible to be sure that the solution is optimal. Meta-Heuristic solutions are preferred more over Heuristic solutions because of their speed, accuracy, ergodicity, flexibility and simplicity. Meta-Heuristics are fast converging. [S10] states that Meta-Heuristic methods explore a larger solution space and combined with Heuristic approaches, more powerful algorithms can be achieved. [S26] commented on the behavior of exploring larger solution space and stated that this results in a tendency to fall to local optima. However, it is still supported that Meta-Heuristics have improved performance of task scheduling problem in cloud.

Study [S17] reviews WOA algorithm, which is a Meta-Heuristic approach, and presents that WOA is slow converging and has high tendency to fall into local optima solution. Due to this reason, WOA has low convergence precision. [S43] is another study which comments on pros and cons of a specific algorithm which is GSA. Authors mentioned that GSA tend to converge to local optimum solutions and they try to overcome that.

In [S19], it is again stated that Meta-Heuristic methods don't guarantee that the offered solution is optimal. What makes Meta-Heuristic approaches more popular is their capability to offer at least near-optimal solutions in a short time frame. Study [S21] further supports this idea and states that Meta-Heuristic approaches are effective in solving real-world problem. In other words, these approaches may not provide the exact optimal solutions but are enough for getting the job done.

[S22] commented on advantages and disadvantages of specific Meta-Heuristic algorithms which are GA and TSA. It is stated in this study that GA is good at finding global optimal solutions and TSA is good at finding local optimal solutions. It is also mentioned that Meta-Heuristic algorithms are way too sensitive to their initialization parameters.

[S28] compared Heuristic and Meta-Heuristic approaches and presented that Heuristics provide problem specific solutions but Meta-Heuristics provide more common solutions.

[S35] presented a different solution to task scheduling problem and based their proposed algorithm on Blockchain technology. It is stated in this study that most Blockchain based algorithms' consensus mechanism is based on proof-of-work scheme. The authors decided to adopt a new scheme called proof-of-schedule which is faster than the more common one.

Most of the proposed algorithms provide software-based solutions to task scheduling problem in cloud according to study [S51]. This study argues that it is a waste of hardware resources thus hardware-based solutions are better at cutting execution times of tasks.

[S54] states that traditional scheduling algorithms such as Random Allocation Algorithm and Greedy Algorithm don't focus on load balancing, QoS and resource utilization. However, also comments that Meta-Heuristic approaches such as ACO have unstable convergence problem. Even though ACO is powerful in solving multi-objective problems according to the authors.

As previously mentioned [S4], [S15] and [S46] comments on PSO; [S5] and [S22] comments on GA and [S54] comments on ACO. Study [S57] makes a comprehensive comparison of these 3 algorithms and states that GA is good at global search but is slow converging and has low evolution rate, ACO's path search is way too random and PSO is fast and efficient but is prone to providing local optima solution.

[S60] argued that Game Theory based approaches coordinate task and energy distribution in a better way.

Study [S64] focuses on the differences of static and dynamic algorithms. It is stated in this study that static algorithms are less complex than dynamic ones but not suitable for highly volatile cloud environments.

- **RQ 1.3 – In which scenarios are individual algorithms beneficial?**

Reviewed studies have stated the intended use cases and cloud environment conditions for most proposed algorithms. 1,41% (n=1) of studies [S4] stated that the proposed algorithm is intended to be used in environments with independent tasks and when deadline constraint is important.

1,41% (n=1) of algorithms [S71] are intended to be used when data intensive workload is encountered in the cloud environment.

2,82% (n=2) of algorithms [S10], [S58] are beneficial in dynamic environment where efficient load balancing is necessary.

1,41% (n=1) of algorithms [S55] are intended to be used in cloud environment that offer E-Learning services. These services are highly resource dependent.

8,45% (n=6) of algorithms [S2], [S44], [S45], [S57], [S59], [S67] focus on energy efficiency goals. Study [S57] also stated that the proposed algorithm focuses on green cloud computing.

1,41% (n=1) of algorithms [S54] are intended to be used in environments where both cost and execution time constraints important. This means that the proposed algorithm focuses on both service provider's and customer's constraints.

5,63% (n=4) studies [S3], [S29], [S62], [S66] focus on fog cloud environment and cloud services intended for IoT devices.

1,41% (n=1) of algorithms [S51] is intended to be used in RISC based architecture especially in ARM CPUs.

4,23% (n=3) of algorithms [S5], [S14], [S35] are intended to be used in multi-cloud environment.

7,04% (n=5) of algorithms [S6], [S16], [S26], [S60], [S69], are intended to be used in multi-objective environment.

2,82% (n=2) of algorithms [S12], [S57] are intended to be used in environments where QoS constraints are important and prioritized.

1,41% (n=1) of algorithms [S18] are intended to be used in environments where on-demand resource allocation is being done.

1,41% (n=1) of algorithms [S20] are intended to be used in environments where priority-based scheduling is needed.

1,41% (n=1) of algorithms [S42] are intended to be used in environments where reliability is an important concern.

56,34% (n=40) studies didn't specify an intended use case for the proposed algorithms. These algorithms are treated as being effective at various use cases and can be applied in general task scheduling problems.

We have grouped intended use cases of algorithms in 17 categories according to the statements of reviewed studies. The influence of use cases on proposed algorithms can be seen in Figure 23.

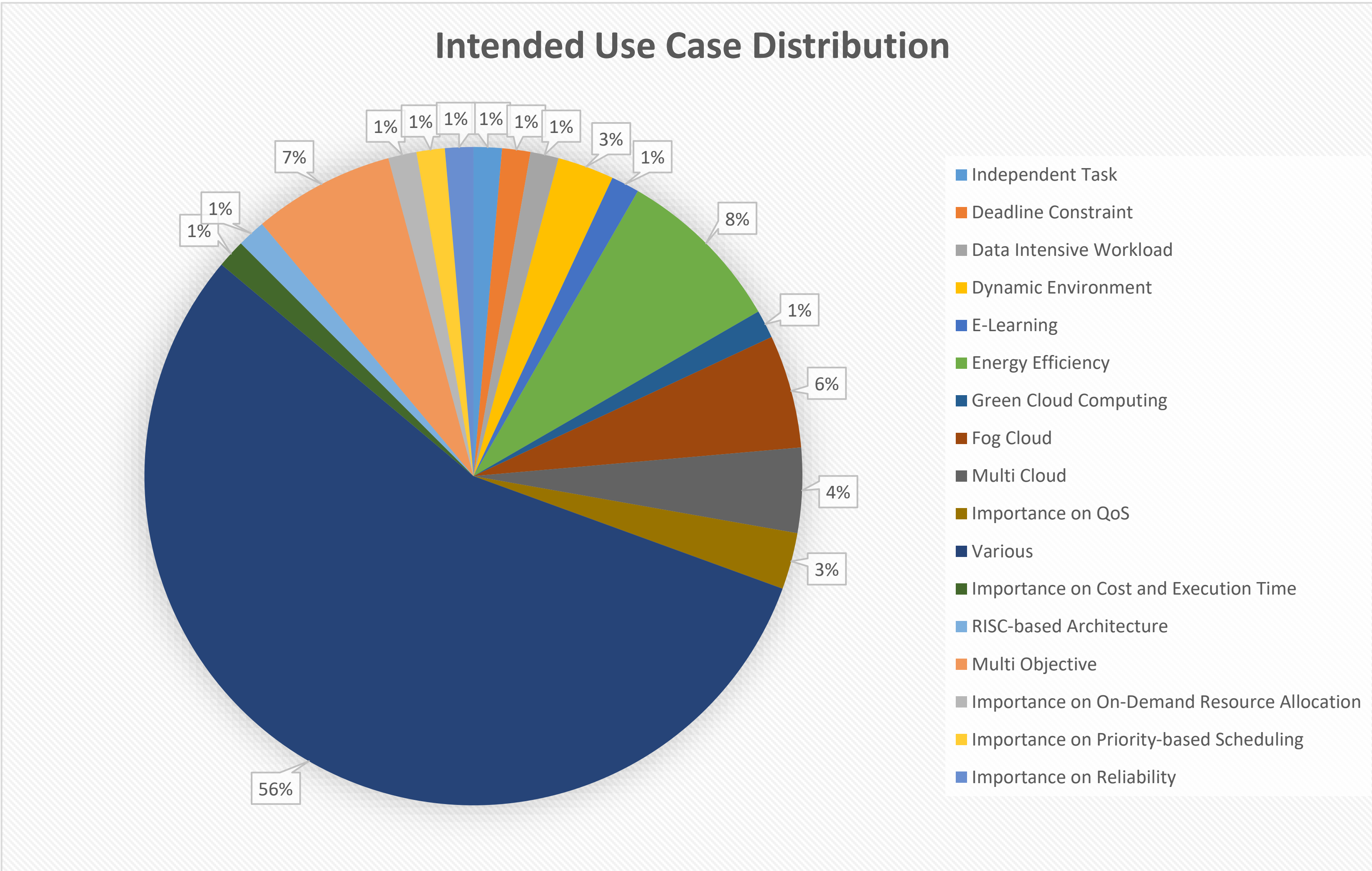

Figure 23 - Intended Use Case Distribution

4.3.4 RQ 2 – What is the mostly used evaluation metric for task scheduling algorithms in cloud computing?

Top 5 evaluation metrics used in reviewed studies are makespan, resource utilization, energy consumption, cost of computation and completition time respectively. Makespan is by far the most considered evaluation metric which influenced 54,93% (n=39) of the studies. The upcoming metrics resource utilization and energy consumption are used equally in 22,54% (n=16) of the studies. Cost of computation is considered in 19,72% (n=14) and completition time in 18,31% (n=13) studies.

The full list of used evaluation metrics and corresponding studies can be seen in Table 14.

Table 14 - Evaluation Metrics Used in Studies

Evaluation Metric	Corresponding Studies
Makespan	[S3], [S4], [S5], [S7], [S8], [S9], [S10], [S13], [S15], [S16], [S19], [S20], [S21], [S22], [23], [S25], [S26], [S28], [S29], [S30], [S32], [S35], [S37], [S38], [S39], [S40], [S46], [S47], [S52], [S53], [S58], [S59], [S61], [S63], [S64], [S67], [S68], [S69], [S70]
Cost of Computation	[S2], [S4], [S5], [S7], [S11], [S12], [S16], [S34], [S39], [S41] [S56], [S60], [S62], [S69]
Completition Time	[S2], [S6], [S11], [S12], [S14], [S17], [S19], [S33], [S34], [S37], [S44], [S49], [S65]
Response Time	[S1], [S5], [S9], [S13], [S23], [S26], [S31], [S45], [S64]
Convergence Accuracy (Fitness)	[S9], [S15], [S28], [S33], [S43], [S49]
Load Balance	[S11], [S23], [S30], [S44], [S58], [S61], [S65]
Resource Utilization	[S2], [S6], [S7], [S10], [S13], [S15], [S18], [S20], [S23], [S30], [S31], [S38], [S39], [S41], [S65], [S67]

Table 14 cont.

Speedup	[S12], [S22]
Efficiency	[S12], [S15], [S22], [S45], [S58]
Data Migration Time	[S14]
User Satisfaction	[S14], [S16], [S38]
Deadline Violation/Success Rate	[S6], [S29], [S65]
Energy Consumption	[S2], [S4], [S15], [S19], [S23], [S27], [S29], [S32], [S39], [S45], [S57], [S62], [S66], [S67], [S70], [S71]
Convergence Speed	[S45], [S50], [S53], [S65]
VM Load	[S16], [S24], [S38], [S53]
Execution Time	[S4], [S15], [S28], [S30], [S54], [S56], [S57], [S60], [S70], [S71]
Fairness	[S41], [S42]
Degree of Imbalance	[S13], [S20], [S26], [S40], [S63], [S64], [S70]
Performance Improvement Rate	[S3], [S20], [S21], [S26]
Economic Cost	[S17], [S24]
Memory Load	[S17]
Computation Time?	[S29]
SLA Violation	[S39]
Number of Offloaded Tasks	[S66]
SINR	[S66]
Waiting Time	[S1], [S15], [S34], [S45]
Bandwidth	[S56]

Table 14 cont.

Steady State Availability	[S42]
Layover Time	[S36]
Task Completition Rate	[S16]
CPU Time	[S34], [S47], [S58]
Throughput	[S3], [S34], [S40]
Number of Migrated Tasks	[S13]
System Scale	[S59]
Schedule Length Ratio	[S22]
Skewness	[S18]
Resource Elimination Rate	[S49]
Memory Utilization	[S18]
Runtime	[S19], [S27], [S48], [S51]
Dropped Task Rate	[S19], [S27]
Turnaround Time	[S1], [S31], [S34]
Context Switch	[S1]

4.3.5 RQ 2.1 – Which evaluation/simulation environment are used?

High percentage of reviewed algorithms are evaluated by making simulations. However, 18,31% (n=13) of studies didn't share evaluation results. Some studies also share the simulation environment, simulation parameters and evaluation methodology in detail. Algorithms are first implemented in various environments and then simulation in simulation tools. Among the simulation tools, mostly used simulation environment is "CloudSim". The distribution of simulation environments can be seen in Figure 24.

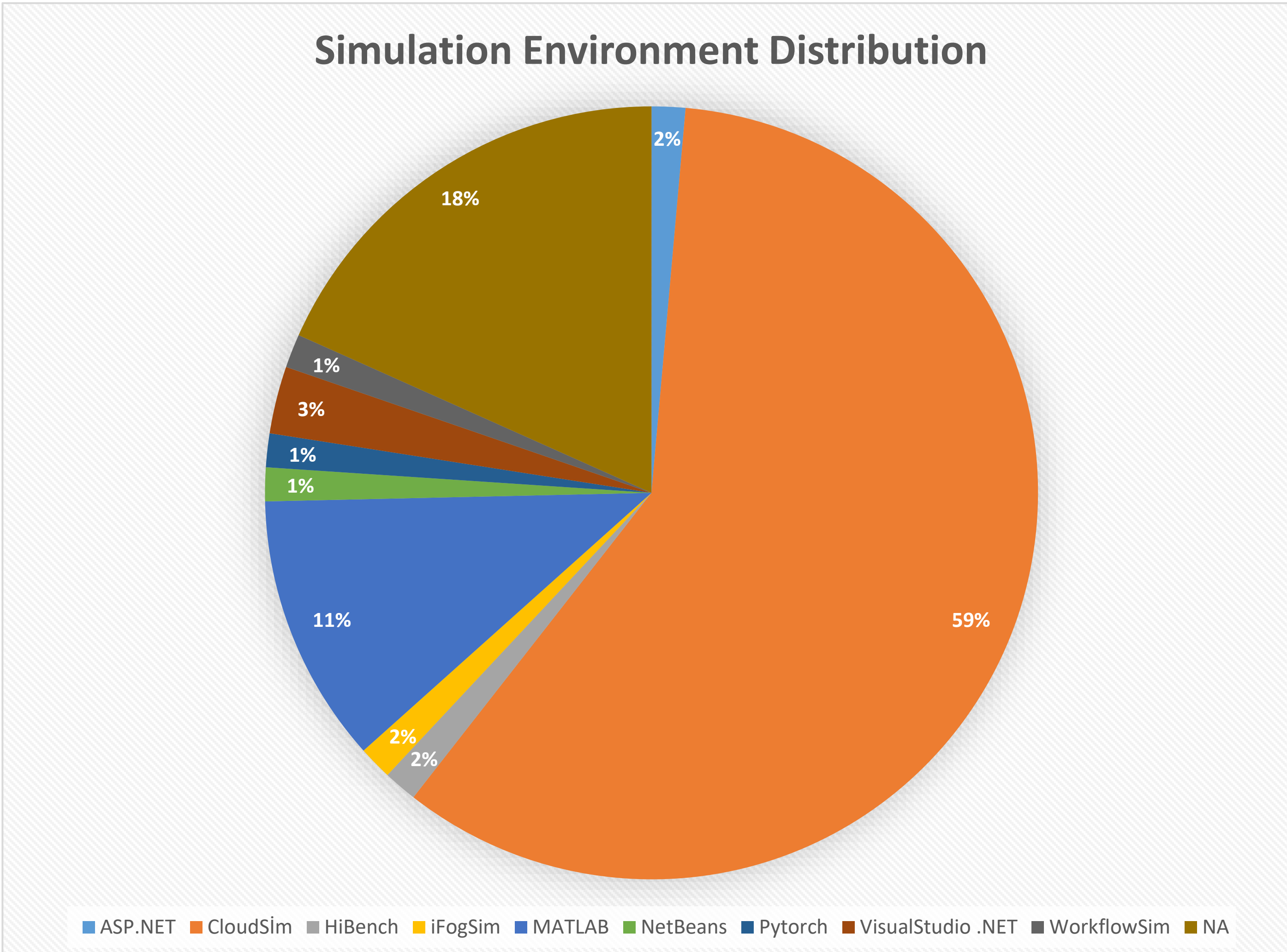

Figure 24 - Simulation Environment Distribution

Algorithms and their corresponding evaluation/simulation environment is given in Table 15.

Table 15 - Simulation Environment Used in Studies

Evaluation Metric	Corresponding Studies
CloudSim	[S1], [S4], [S5], [S7], [S8], [S9], [S10], [S11], [S13], [S14], [S16], [S17], [S20], [S21], [S23], [S25], [S26], [S28], [S30], [S31], [S32], [S33], [S34], [S35], [S38], [S39], [S40], [S44], [S45], [S48], [S50], [S52], [S54], [S56], [S57], [S61], [S63], [S64], [S65], [S67], [S68], [S69], [S71]
NA	[S6], [S8], [S12], [S18], [S22], [S37], [S41], [S42], [S43], [S47], [S49], [S55], [S59]
MATLAB	[S3], [S24], [S29], [S36], [S46], [S60], [S66], [S70]
Visual Studio .NET	[S19], [S58]
ASP.NET	[S2]
iFogSim	[S61]
HiBench	[S51]
NetBeans	[S15]
Pytorch	[S27]
WorkflowSim	[S53]

4.3.6 RQ 2.2 – How reliable are evaluation methods and are they repeatable under different cases?

The most popular approach used in evaluating the algorithms is simulating them in a simulation environment and comparing the metric results with other similar algorithms. To do so, similar algorithms are also simulated and evaluated using the same metrics. This increases the trustworthiness of the proposed algorithm as the performance f the algorithm is compared with well-known algorithms. A researcher can evaluate their algorithms' performance and share the quantitative results of evaluations. However, this information is not enough for other researchers while comparing several algorithms. Each researcher's simulation environment and parameters can be different which make the comparison harder. For this reason, researchers tend to compare their algorithms with other ones in same evaluation environment. The candidate algorithms to be compared are chosen depending on the base algorithm of proposed algorithm.

However, not all algorithms are compared with other ones. In %2,82 (n=2) studies, Comparison is done but the compared algorithm is generic. No reference for the evaluated algorithm is given in [S41] and [S51]. In %2,82 (n=2) studies, no information about simulation environment is given. Only the evaluation and comparison results are shared in [S6], [S8]. No comparison is done in %2,82 (n=2) studies ([S12], [S37]). In %1,41 (n=1) studies, no evaluation is done.

Besides the comparison quality of the studies, explanation of evaluation methodology and simulation environment is important. We decided that %30,99 (n=22) studies didn't properly explained the evaluation methodology and simulation environment.

4.3.7 RQ 3 – What are the future research recommendations?

The most encountered future recommendation is improving the proposed algorithm. Some studies recommend including more parameters during design while some recommend evaluating the proposed algorithm with different metrics. 71,83% (n=51) studies recommend improving the proposed algorithm in some way or other. 5,63% (n=4) studies recommended comparing the proposed algorithm with other different algorithms. 21,83% (n=15) studies didn't give any future research recommendations.

[S1] proposed an algorithm based on traditional RR algorithm. Researchers recommend improving Quantum Time calculation method of RR. [S2] recommends including fairness feature in proposed algorithm and redo evaluation in real cloud environment. [S3] recommends extending proposed algorithm to make it multi-objective. [S4] recommends focusing more on scheduling of dependent tasks and including more QoS parameters in the proposed algorithm.

[S5] considers applying proposed model to different application areas and evaluating the algorithm using DoI and energy consumption metrics. [S7] considers evaluating proposed algorithm with other metrics and improving time complexity of algorithm. [S8] recommends enhancing QoS and resource allocation aspects of the algorithm. [S9] recommends enhancing load prediction feature and evaluating with different metrics. [S10] recommends incorporating SLAs in proposed algorithm. [S11] considers focusing on task dependency. [S13] recommends improving multi-objective features of the proposed algorithm. [S14] recommends studying the correlation among tasks. [S15] considers studying VM placement for improving QoS features.

As seen in given examples, there are various future research recommendations across studies. The full list of future study recommendations can be seen in Table 16.

Table 16 - Future Research Recommendations List

Study	Future Research Recommendation
[S1]	*Improving RR algorithm in terms of QT calculation method. *Applying new techniques for QT calculation Integrating RR with meta-heuristic algorithms. *Evaluation can be done with large datasets of tasks.
[S2]	*Including fairness feature in proposed algorithm. *VM and task count can be increased for simulation. *Evaluation can be done in real cloud environment.
[S3]	*Proposed algorithm can be extended to become multi-objective.
[S4]	*Focus on scheduling of dependent tasks. *Include new QoS parameters such as energy and power consumption, task rejection ratio, load balancing, turnaround time.
[S5]	*Applying proposed model to other application models. *Improve the proposed algorithm. *Consider degree of imbalance and energy efficiency during evaluation.
[S7]	*Evaluating proposed algorithm with other fitness functions. *Evaluating proposed algorithm with other QoS metrics. *Improving time complexity of proposed algorithm.
[S8]	*Multi optimization can be focused. *QoS and resource allocation aspects can be enhanced.
[S9]	*Enhance proposed algorithm with load prediction and test with different service broker policy.

Table 16 cont.

[S10]	*Proposed algorithm can be extended to incorporate Service Level Agreements (SLA).
[S11]	*Task dependency will be considered. *Number of tasks in evaluation scenario will be increased.
[S13]	*Generalizing proposed algorithm in geographical clouds with dispersed datacenters. *Presenting proposed method as an autonomic multi-objective scheduling.
[S14]	*Correlation among tasks should be investigated.
[S15]	*Virtual machine placement for QoS improvement.
[S16]	*Security issues in scheduling model will be investigated. *Dynamic characteristics such as VM failure will be investigated. *Fault tolerant scheduling model will be established.
[S17]	*Developing efficient scheduling system suitable for various task workloads.
[S18]	*Increasing the performance of proposed algorithm using advanced optimization algorithms.
[S19]	*Improving proposed algorithm in terms of load balancing, number of dropped tasks, makespan. *Developing parallel version of proposed algorithm.
[S20]	*Incorporating energy consumption, VM and task migration to the proposed algorithm.
[S21]	*Improving performance of proposed algorithm. *Extending HGSWC for multi-objective optimization. *Improve HGSWC for IoT, fog computing, image segmentation, data clustering and classification purposes.
[S22]	*Presenting cloud reliability prediction model for evaluating cloud providers. *Improving proposed algorithm to make its final scheduling insertion based.
[S23]	*Developing efficient scheduling algorithm by considering priority.
[S24]	*Reducing operating costs of proposed algorithm. *Studying multi-objective meta-heuristic algorithms more.

Table 16 cont.

[S25]	*Improve the proposed algorithm to achieve more objectives.
[S26]	*Comparing proposed algorithm with more algorithms. *Applying proposed algorithms to other optimization problems. *Improving proposed algorithm to consider other parameters.
[S28]	*Testing proposed algorithm in real cloud environment. *Improving proposed algorithm to consider power consumption, task waiting time etc. *Considering real time cloud tasks.
[S29]	*Improving proposed algorithm by considering dependency among tasks and network topology between fog nodes.
[S30]	*Improving proposed algorithm by considering security issues, energy consumption, cost, throughput and time complexity.
[S31]	*Employing proposed algorithm with real time workflow to prove its efficiency.
[S32]	*Evaluating proposed algorithm with Grey Wolf Optimization (GWO) algorithm.
[S35]	*Improving proposed algorithm by considering DAG model. *Focusing more on multi-cloud technologies.
[S36]	*Improving proposed algorithm in terms of efficiency by focusing on task restriction problem.
[S37]	*Making proposed algorithm work in different environments such as Fog Cloud.
[S40]	*Comparing proposed algorithm with workflow scheduling based approaches. *Evaluating proposed algorithm with other metrics.
[S42]	*Improving proposed algorithm by considering other reliability parameters.
[S43]	*Improving proposed algorithm by focusing on number of tasks to be completed per unit time in terms of cost.
[S44]	*Improving proposed algorithm by considering differences in data centers and their physical environments. *Optimizing packing algorithm.

Table 16 cont.

[S46]	*Improving proposed algorithm by considering load balancing and energy consumption.
[S47]	*Improving the local search ability of proposed algorithm. (enhance balance between exploitation and exploration). *Increase global search ability of proposed algorithm. *Enhance proposed algorithm in order to solve other problems. (dynamic task scheduling, resource assignment). *Improve timing metrics of proposed algorithm (time between tasks, I/O times for different size tasks).
[S50]	*Cost and energy factors should be considered while designing the algorithm.
[S52]	*Proposed algorithm will be optimized to reduce the complexity.
[S53]	*Proposing a new model for energy consumption in order to reduce costs when deadline is loose.
[S54]	*Stability and efficiency of scheduling algorithms need to be examined. *Proposed algorithm must be enhanced by considering throughput rate and migration time etc.
[S55]	*Applying other techniques based on cats and camels instead of bees and comparing with proposed algorithm.
[S56]	*Combining Shapley value with other evolutionary algorithms. *Applying the proposed approach to other environments such as Fog/Edge Computing.
[S58]	*Combination of other evolutionary algorithms can be proposed as new algorithms.
[S59]	*Improving proposed algorithm by adding more influence factors.
[S60]	*Improving proposed algorithm by adding more factors in the target function. (ex. Current load on the VM and expense of migration process)
[S61]	*Studying the proposed algorithm more. *Applying the proposed algorithm to other problems such as VM migration. Cloud security etc.
[S62]	*Comparing proposed algorithm with more algorithms. *Improving proposed algorithm by considering response time.

Table 16 cont.

[S64]	*Improving proposed algorithm by applying it to workflows. *Improving proposed algorithm by considering other factors such as cost reduction.
[S65]	*Improving the proposed algorithm.
[S67]	*Improving proposed algorithm by considering large-scale data centers. It can be done by using SCORE simulator as an extension to Google Omega.
[S68]	*Improving proposed algorithm.
[S69]	*Proposing new methods by considering more constraints during design.
[S70]	*Improving proposed approach by using different heuristic approaches as starting point of PSO algorithm.
[S71]	*Evaluating the proposed algorithm in terms of cost efficiency. Improving proposed algorithm by considering minimizing cost and QoS.
[S6], [S12], [S27], [S33], [S34], [S38], [S39], [S41], [S45], [S48], [S49], [S51], [S57], [S63], [S66]	NA

4.3.8 RQ 3.1 – Which topics lack enough research currently in task scheduling field and need further attention?

Reviewed studies show that there is no general agreement across researchers in terms of attention needing topics. The full list of attention needing topics can be seen in Table 17.

Table 17 - Attention Needing Topics List

Study	Attention Needing Topics
[S1]	*QT (Quantum Time) of Round Robin algorithm must be tuned efficiently.
[S2]	*Fairness of algorithm in heterogeneous environment.
[S3]	*Job shop scheduling and vehicle routing.
[S4]	*Premature convergence issue *Multiple QoS parameters should be focused on.
[S5]	*Algorithms proposed for multi-cloud environment are not efficient. *Algorithms doesn't provide global optima and local optima.
[S6]	*Maximum completition time, deadline violation rate and utilization of resource metrics of multi-objective algorithms.
[S7]	*Energy consumption, user data security, resource utilization rate.
[S10]	*Multi-objective approach
[S11]	*Load balancing
[S13]	*Communication overheads.
[S14]	*Focusing on both time and cost simultaneously.
[S15]	*Reducing waiting time.
[S16]	*Convergence and diversity of algorithm at all stages should be studied.
[S17]	*Long scheduling time, high cost consumption, and large virtual machine load.
[S18]	*Elasticity of algorithms. Handling resource requirements dynamically. *Energy usage.
[S19]	*Multi-objective optimization.
[S20]	*Task scheduling based on priority.

Table 17 cont.

[S21]	*Being able to generate new solutions in order to update worse solutions found by HGSO algorithm.
[S22]	*Meta-heuristic algorithms are good global optimizers but bad local optimizers. A new solution must be found to find balance.
[S23]	*Solving the problem of trapping in local optima solutions in SSA.
[S24]	*Optimization of convergence speed and accuracy.
[S25]	*Makespan of algorithms must be decreased.
[S26]	*Optimization based meta-heuristic algorithms must be developed.
[S27]	*ML based algorithms must be investigated for task scheduling issue solution.
[S29]	*Focusing on deadline of tasks, energy consumption and computation time.
[S30]	*Load balancing and makespan must be considered together.
[S32]	*Multi objective scheduling.
[S33]	*Minimizing maximum time span of tasks.
[S34]	*Time needed for sending the tasks must be studied.
[S35]	*Energy consumption and security of scheduling algorithms.
[S36]	*Focusing on task pairing and reducing layover time.
[S37]	*Dependent tasks should also be considered while developing algorithm.
[S38]	*Deciding to focus on QoS or resource utilization *Meeting QoS requirements.
[S39]	*Performance and task planning effectiveness of algorithms.
[S40]	*Total execution time of the tasks must be considered while designing algorithms.
[S41]	*Minimizing cost is more important than execution time in computational grid.

Table 17 cont.

[S42]	*Reliability of scheduling algorithm.
[S44]	*Energy consumption of scheduling algorithms.
[S47]	*Flexible task scheduling with multiple objectives.
[S48]	*Reducing population size of evolutionary algorithms.
[S49]	*Efficiency of algorithms is important.
[S51]	*Most algorithms treat all cores in SoC equally. To increase efficiency, cores should be focused individually.
[S53]	*Algorithms should be dynamic in nature.
[S54]	*Achieving multi objective comprehensive performance optimization.
[S55]	*Research should be done on smart resource allocation in e-learning.
[S56]	*Multiple QoS parameters should be considered while developing algorithm. *A common QoS model should be developed.
[S57]	*Energy consumption of cloud computing has become an obstacle. It must be solved.
[S58]	*Load balancing
[S59]	*Task execution time and energy efficiency.
[S60]	*More performance metrics and QoS parameters should be considered while developing a scheduling algorithm. Such as reliability, availability, quickness, and scalability.
[S62]	*Algorithms should consider changes in user request during scheduling process.
[S63]	*Allocation of large-scale tasks to small resources and small-scale tasks to large resources must be solved.
[S64]	*Load balancing
[S65]	*Falling into local optimization problem must be solved. *Unbalanced task scheduling.

Table 17 cont.

[S66]	*Task scheduling should be based on global perspective and task dependency should be considered to cut cost.
[S67]	*Energy efficient load balancing
[S69]	*QoS
[S70]	*Initialization of population of metaheuristic algorithms must be improved.
[S71]	*Balancing power and energy consumption.
[S8], [S9], [S12], [S31], [S43], [S45], [S46], [S50], [S52], [S61], [S68]	NA

4.3.9 RQ 3.2 – Is there any common agreement among researchers in further research direction?

4.3.9.1 *Meta Synthesis on Further Research Direction*

The most frequently encountered future research recommendations are either enhancing the proposed algorithms or making evaluations of proposed algorithms by comparing them with different algorithms. Enhancement recommendations on the proposed algorithms vary across studies. Most researchers recommend enhancements by focusing on different metrics during implementation. However, there is no common agreement among researchers in terms of enhancement needing metrics.

Most researchers compared their algorithms with other similar algorithms. Similarities are decided according to the algorithm category, intended use and algorithm bases. For example, a researcher whose algorithm is based on ACO tend to compare it with original ACO algorithm and other meta-heuristic algorithms. However, there is a large pool of algorithms to be compared as seen in Table 13. Due to this reason, most researchers recommend evaluating their algorithms by other algorithms available in literature in order to establish a stronger evaluation base.

The common pattern found in future research directions of other researchers show that their proposed algorithms should be improved. This is a strong proof for our hypotheses of current cloud task scheduling algorithms are incapable of achieving business processes. Even the designers of algorithms proposed during COVID-19 pandemic recommend improving their algorithms. This shows us that even the algorithms designed with the consideration of the impacts of COVID-19 are inferior.

CHAPTER 5

5 DISCUSSIONS

This chapter summarizes the findings of the study and comments on the validity threats of our Meta Synthesis methodology. Findings that are shared in Chapter 4 are summarized according to the research questions and validity threats are investigated.

The following section is a recap of findings which points out the common themes found across studies. This part consists of the thematic analysis of the reviewed studies.

5.1 Discussion on Findings

Findings of RQ1 show that there are different approaches for categorizing cloud task scheduling algorithms. These approaches are categorizing algorithms depending on their problem-solving approach, resource mapping strategy, execution model, scheduling goals and allocation decision time. It is found in reviewed literature that there are 5 different categorization schemes. Most common approach is categorizing algorithms depending on their problem-solving approach.

Findings of RQ1.1 show that Most common problem-solving approach is Meta-Heuristics. Most common resource mapping strategy is dynamic scheduling. Most common scheduling goal is multi-objective. Most common allocation decision time is batch mode.

Findings of RQ1.2 show that Meta-Heuristic algorithms are the most powerful type of algorithms in cloud environment. Meta-Heuristics tend to fit in most scenarios and provide faster solutions. However, their major problem is converging to local optima and not providing the most optimal solution. Heuristic algorithms are easy to implement and not as complex as Meta-Heuristics. However, they underperform in most situations. Researchers tend to hybridize heuristics and create meta/meta and meta/non-meta hybrid heuristic algorithms. This is used to combine advantages of different algorithms.

Findings of RQ1.3 show that most researchers stated their algorithm's intended use cases. These use cases are environments with independent tasks, deadline constraints, data intensive workload, dynamic behavior, e-learning, green cloud computing, fog cloud, multi-cloud, importance on QoS, importance on cost and execution time, RISC-based architectures, multi objective goals, on-demand resource allocation, priority-based scheduling, importance on reliability and varying cases. Most algorithms are developed for addressing energy efficiency issues.

Findings of RQ2 show that there are numerous evaluation metrics which are used for evaluating performance of cloud task scheduling algorithms. The most common metric is makespan. Makespan is defined as the total time taken from the beginning to the ending of a task.

Findings of RQ2.1 show that there are 9 different simulation environments in literature. The most common environment is CloudSim with a major difference. 43 studies are conducted on CloudSim. The second most common environment is MATLAB with 8 studies.

Findings of RQ2.2 show that researchers tend to compare their algorithms with other similar algorithms during evaluation phase. They comment on the differences and validate the performance of their algorithms by displaying how they performed in a challenging environment compared with other well-reputed algorithms. The second mostly used approach for increasing the validity of evaluation is explaining the evaluation methodology. Not all researchers explain the methodology of their evaluation strategy. This situation makes it harder for other researchers to validate the evaluation results as the methodology is not explained well.

Findings of RQ3 show that there is no specific future research recommendation that is consistent across reviewed studies. However, most researchers recommend either enhancing the proposed algorithms or comparing the proposed algorithm with different algorithms that are not commented in the corresponding study.

Findings of RQ3.1 show that there is a huge variety of opinions across researchers in terms of attention needing topics in cloud computing field. Most noticeable topics are energy efficiency issues, fog computing environment, multi-objective goals and QoS. According to the researchers, their proposed algorithms provide some sort of solution to the problems mentioned in corresponding topics.

Finally, findings of RQ3.2 show that there is no common agreement among researchers in terms of future research recommendation. As mentioned before, there is only common patterns across studies, which are either enhancing the algorithms or focusing on the comparison.

5.2 Task Scheduling Trends as of 2023

When the literature is searched, we still encounter articles about new task scheduling algorithms. This is a sign that efficient scheduling is still a problem after COVID-19 pandemic. The perfect approach isn't decided yet. Researchers still recommend further developing existing and traditional algorithms. [37] One of the main issues in Task Scheduling in 2023 is energy efficiency. It is supported that Container technology is newly emerging and can provide a new insight on energy efficient Task Scheduling. Researchers are recommended to study the benefits of using Container based system designs for energy efficient Task Scheduling.

It is often mentioned in recent studies that changes in business trends have showed us current computing infrastructure is inferior and needs to be developed. [37] It is spotted that Meta-Heuristic algorithms, especially Nature Inspired ones, still tend to perform better than other algorithm categories. GA and PSO algorithms are the most referenced ones during algorithm design periods. [38] In one of the most cited studies, it is mentioned that Meta-Heuristic algorithms perform better than heuristic ones thus heuristic algorithms should be hybridized with Meta-Heuristic ones. [39] Sing et al. suggested that optimization of current Task Scheduling algorithms is still an issue in this domain. [40]

Workflow scheduling is another concern in 2023. Businesses are transforming to Industry 4.0 standards which means workflow scheduling should be further researched. [41] IoT applications still also need a stable and optimized task scheduling method. Resource allocation and clustering in IoT field continues to be a concern for developers. [42]

These all show us that task scheduling still continues to be an issue at Cloud Computing in 2023. Same concerns that affected researchers during COVID-19 pandemic still continues to affect them. This proves that our findings from the studies published during the pandemic are still valid and trustable.

5.3 Threats on Validity

Meta-Synthesis studies are prone to validity threats due to researcher-based biases and other environmental factors. In this chapter, we have discussed the possible threats on the validity of this study.

5.3.1 Threat on Construct Validity

Construct validity considers if the information that researcher wanted to extract from reviewed studies are actually extracted. It can be explained also as extracting the related data and not irrelevant information. During data extraction phase, we have followed a transparent strategy and didn't make any comment or interpretation on the data provided by respectful researchers. In other words, if the researchers didn't clearly state some sort of information, we didn't include in our data extraction form. For example, some researchers did clearly mention that there exists "N" number of algorithm categories and stated what these algorithm types are. Some researchers didn't clearly state the categories but it can be implied from the general sections of the reviewed study. In this case, we considered this case as there is no categorization provided by study. The most possible construct validity threat in our study is if the extracted information is noted down the forms correctly. Data extraction was done solely by one researcher and no cross check is done.

5.3.2 Threats on Internal Validity

Internal validity considers if all relevant studies in review domain is accessed and reviewed. This threat on internal validity is correlated with the approach followed during 3.2.1 Source Selection and Search Protocol. The used search terms, selected databases and inclusion/exclusion criteria should be examined to comment on internal validity. First, we have derived the main search term according to PICOC criteria which is a widely accepted method in literature. Then we have altered the main search term for specific databases considering their query mechanisms. During review process of 2.2 Related Work phase, we have noticed that term "resource scheduling" is used interchangeably with "task scheduling" term by some researchers. We have also included this keyword in search terms to avoid missing possible studies. We have used multiple databases. During application of inclusion/exclusion criteria, we have followed a rational approach and applied the criteria depending on facts. For example, "well reputed" sources are decided depending on their Q-Rankings and H-Index provided by SJR[7]. Only Q2 and Q1 sources with H-Index above 20 is chosen.

[7] https://www.scimagojr.com/

5.3.3 Threat on External Validity

External validity considers the generalizability of the study. As we have followed a systematic methodology during our research, the outcomes will be consistent across other studies. Software Engineering literature is way too heterogeneous in terms of results and this is also the case for Cloud Computing. Our pool of reviewed studies is large enough to include different perspectives from various studies. For this reason, we are sure that another study that will be done with same principles will result in similar outcomes.

5.3.4 Threat on Conclusion Validity

Conclusion validity considers how well the data is extracted and a conclusion that is consistent to the data is derived. It is also important for the same conclusion to be inferred by other researchers if the study is repeated. In 4 RESULTS section, we have shared the analysis results of the extracted data from reviewed studies. We have also pointed out which studies hold the corresponding information in tabular format. These make the provided data traceable. The corresponding studies can be rereviewed and the same conclusion would be derived.

CHAPTER 6

6 CONCLUSION & FUTURE WORK

6.1 Conclusion

Development of Cloud Computing was inevitable due to the changing world and changing needs. In order to adapt the changing needs, new computing infrastructures are developed and incorporated to the distributed computing environment. These innovations lead to the development of Cloud Computing which gained respect from specialists of the field. It is proven that Cloud Computing has huge potential in solving daily needs of people.

Over the years, Cloud Computing brought some problems with it after large corporations began offering services based on Cloud Computing. Once a newly developed technology begins to be used in the field, its drawbacks attract attention. One such problem is the task scheduling mechanism of Cloud Computing services. During the literature review and background work phases, we have we came across numerous studies which prove that task scheduling is a massive problem and needs solution.

Many researchers have proposed algorithms to address task scheduling issue of Cloud Computing. However, we have noted that not much collective review of such algorithms was done before. We wanted to address this problem and provide information about the state of the art. Cloud Computing has not seen many challenges over the years beginning from its first services. COVID-19 pandemic has increased the demand on Cloud Computing massively and pushed the services to their limits in terms of performance and efficiency. We thought that this may have affected the direction task scheduling algorithm development is going and proved it during literature review phase. In order to analyze the state of the art, we decided to conduct a Meta-Synthesis study covering the years of 2020 and 2021 which is the period after the pandemic began.

In this Meta-Synthesis study, we have reviewed 71 studies that propose a new task scheduling algorithm. Our initial database search resulted in 1262 papers and we have deducted this number by applying inclusion/exclusion criteria. We have carefully extracted data from these studies and shared the contents using narrative synthesis method, analyzed the contents to derive statistical outcomes and did meta-synthesis on these statistical findings to answer research questions.

The main purpose of this study was to discover the aspects of proposed algorithms and the considerations done during the development of such algorithms. We have made a general categorization of these algorithms consistent with the approaches taken by researchers. The advantages and disadvantages of specific categories are analyzed and the question of which types of algorithms are beneficial for certain applications is answered. The evaluation environments used in these studies end the frequency of them is shared.

One important aspect is the reliability of evaluation results. Reliability of evaluation results are commented on in this study.

The synthesis of the extracted data show that there is a trend in dynamic meta-heuristics algorithms which focus on improving makespan metric. The fast convergence speed and general-applicability of meta-heuristic algorithms are frequently stated in reviewed studies. However, meta-heuristic algorithms are not perfect and tend to provide non-optimal solutions. It is noted that the differences of algorithms come out when there is high number of tasks in the environment.

Future research recommendations of researchers are mostly either enhancing the proposed algorithms or including other metrics during evaluation.

All these outcomes show that task scheduling algorithms still need improvement and no single perfect algorithm exists that can be used in any environment. There are issues within Cloud Computing field that are agreed on across researchers such as low energy efficiency and slow operations.

6.2 Future Work

It is found that heuristic algorithms didn't gain much attention as meta-heuristic algorithms. New researchers that are interested in technical aspects of task scheduling issue of Cloud Computing and want to improve existing algorithms or provide new ones can focus on heuristic algorithms. Hybridization of algorithms is done mostly across meta-heuristic ones and more meta/non-meta hybrid algorithms can be provided and evaluated.

According to the findings, algorithms' performance difference occurs more when high number of tasks exist in the environment. New studies can try to improve performance differences when there is low number of tasks in the environment.

The goal of this study is to provide information about state of the art for Cloud Computing. This study has thought of COVID-19 pandemic as a milestone for Cloud Computing field and reviewed studies published after the pandemic has begun. New Meta Synthesis and Meta-Analysis studies can be done which covers time period before the pandemic to demonstrate the differences in state of the art. More comparative studies can be done in terms of time frame.

REFERENCES

General References

[1] M. Armbrust *et al.*, “A View of Cloud Computing,” *Commun. ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010, doi: 10.1145/1721654.1721672.

[2] S. Singh and I. Chana, “A Survey on Resource Scheduling in Cloud Computing: Issues and Challenges,” *J Grid Comput*, vol. 14, no. 2, pp. 217–264, Jun. 2016, doi: 10.1007/s10723-015-9359-2.

[3] B. Kitchenham and S. Charters, “Guidelines for performing Systematic Literature Reviews in Software Engineering,” vol. 2, Jul. 2007.

[4] V. v Arutyunov, “Cloud Computing: Its History of Development, Modern State, and Future Considerations,” *SCIENTIFIC AND TECHNICAL INFORMATION PROCESSING*, vol. 39, no. 3, pp. 173–178, Jul. 2012, doi: 10.3103/S0147688212030082.

[5] M. C. Angelides, “Implementing the Internet for business: A global marketing opportunity,” *Int J Inf Manage*, vol. 17, no. 6, pp. 405–419, 1997, doi: https://doi.org/10.1016/S0268-4012(97)00024-8.

[6] J. Surbiryala and C. Rong, “Cloud Computing: History and Overview,” in *2019 IEEE Cloud Summit*, Aug. 2019, pp. 1–7. doi: 10.1109/CloudSummit47114.2019.00007.

[7] G. Boss, P. Malladi, D. Quan, L. Legregni, and H. Hall, “Cloud computing,” *IBM white paper*, vol. 321, pp. 224–231, 2007.

[8] A. R. Arunarani, D. Manjula, and V. Sugumaran, “Task scheduling techniques in cloud computing: A literature survey,” *FUTURE GENERATION COMPUTER SYSTEMS-THE INTERNATIONAL JOURNAL OF ESCIENCE*, vol. 91, pp. 407–415, Feb. 2019, doi: 10.1016/j.future.2018.09.014.

[9] A. A. Motlagh, A. Movaghar, and A. M. Rahmani, “Task scheduling mechanisms in cloud computing: A systematic review,” *INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS*, vol. 33, no. 6, Apr. 2020, doi: 10.1002/dac.4302.

[10] A. Keivani, F. Ghayoor, and J.-R. Tapamo, “A Review of Recent Methods of Task Scheduling in Cloud Computing,” in *2018 19TH IEEE MEDITERRANEAN ELECTROTECHNICAL CONFERENCE (IEEE MELECON’18)*, 2018, pp. 104–109.

[11] P. Singh, M. Dutta, and N. Aggarwal, "A review of task scheduling based on meta-heuristics approach in cloud computing," *Knowl Inf Syst*, vol. 52, no. 1, pp. 1–51, Jul. 2017, doi: 10.1007/s10115-017-1044-2.

[12] M. Hosseinzadeh, M. Y. Ghafour, B. Hama Hawkar Kamaran and Vo, and A. Khoshnevis, "Multi-Objective Task and Workflow Scheduling Approaches in Cloud Computing: a Comprehensive Review," *J Grid Comput*, vol. 18, no. 3, pp. 327–356, Sep. 2020, doi: 10.1007/s10723-020-09533-z.

[13] A. Ullah, N. M. Nawi, and S. Ouhame, "Recent advancement in VM task allocation system for cloud computing: review from 2015 to2021," *Artif Intell Rev*, vol. 55, no. 3, pp. 2529–2573, Mar. 2022, doi: 10.1007/s10462-021-10071-7.

[14] E. H. Houssein, A. G. Gad, Y. M. Wazery, and P. N. Suganthan, "Task Scheduling in Cloud Computing based on Meta-heuristics: Review, Taxonomy, Open Challenges, and Future Trends," *Swarm Evol Comput*, vol. 62, Apr. 2021, doi: 10.1016/j.swevo.2021.100841.

[15] P. B. Jawade, D. Sai Kumar, and S. Ramachandram, "A compact analytical survey on task scheduling in cloud computing environment," *International Journal of Engineering Trends and Technology*, vol. 69, no. 2, pp. 178–187, 2021, doi: 10.14445/22315381/IJETT-V69I2P225.

[16] S. S. Pol and A. Singh, "Task Scheduling Algorithms in Cloud Computing: A Survey," in *ICSCCC 2021 - International Conference on Secure Cyber Computing and Communications*, 2021, pp. 244–249. doi: 10.1109/ICSCCC51823.2021.9478160.

[17] A. Keivani and J.-R. Tapamo, "Task scheduling in cloud computing: A review," in *icABCD 2019 - 2nd International Conference on Advances in Big Data, Computing and Data Communication Systems*, 2019. doi: 10.1109/ICABCD.2019.8851045.

[18] P. Sharma, S. Shilakari, U. Chourasia, P. Dixit, and A. Pandey, "A survey on various types of task scheduling algorithm in cloud computing environment," *International Journal of Scientific and Technology Research*, vol. 9, no. 1, pp. 1513–1521, 2020.

[19] N. Bulchandani, U. Chourasia, S. Agrawal, P. Dixit, and A. Pandey, "A survey on task scheduling algorithms in cloud computing," *International Journal of Scientific and Technology Research*, vol. 9, no. 1, pp. 460–464, 2020.

[20] V. Ahari, R. Venkatesan, and D. P. P. Latha, "A survey on task scheduling using intelligent water drops algorithm in cloud computing," in *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, 2019, pp. 39–45. doi: 10.1109/ICOEI.2019.8862777.

[21] N. Bharot and S. Shukla, "A Review on Task Scheduling in Cloud Computing using parallel Genetic Algorithm," in *2020 International Conference on Computing and Information Technology, ICCIT 2020*, 2020. doi: 10.1109/ICCIT-144147971.2020.9213822.

[22] R. Vijayalakshmi and S. K. V. Jayakumar, "Green cloud computing: An extensive survey in selecting multi-objectives for task scheduling in sustaining energy efficiency," *Journal of Green Engineering*, vol. 10, no. 11, pp. 11569–11593, 2020.

[23] M. G. Huang and Z. Q. Ou, *Review of task scheduling algorithm research in cloud computing*, vol. 926–930. 2014. doi: 10.4028/www.scientific.net/AMR.926-930.3236.

[24] A. Daniel, P. Rajakumar, N. V. Kousik, N. Yuvaraj, and S. Jayasri, "A survey on various load balancing algorithm to improve the task scheduling in cloud computing environment," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, no. 8 Special, pp. 2397–2406, 2019.

[25] L. Tom and V. R. Bindu, *Task Scheduling Algorithms in Cloud Computing: A Survey*, vol. 98. 2020. doi: 10.1007/978-3-030-33846-6_39.

[26] A. A. Zubair, S. B. A. Razak, M. A. B. Ngadi, A. Ahmed, and S. H. H. Madni, *Convergence-based task scheduling techniques in cloud computing: A review*, vol. 1073. 2020. doi: 10.1007/978-3-030-33582-3_22.

[27] J. Andre, P. Siarry, and T. Dognon, "An improvement of the standard genetic algorithm fighting premature convergence in continuous optimization," *Advances in Engineering Software*, vol. 32, no. 1, pp. 49–60, Jan. 2001, doi: 10.1016/S0965-9978(00)00070-3.

[28] S. Atiewi, S. Yussof, M. Ezanee, and M. Almiani, "A review energy-efficient task scheduling algorithms in cloud computing," in *2016 IEEE Long Island Systems, Applications and Technology Conference, LISAT 2016*, 2016. doi: 10.1109/LISAT.2016.7494108.

[29] E. S. Alkayal, N. R. Jennings, and M. F. Abulkhair, "Survey of task scheduling in cloud computing based on particle swarm optimization," in *2017 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2017*, 2017, vol. 2018-Janua, pp. 1–6. doi: 10.1109/ICECTA.2017.8251985.

[30] 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*, 2014. doi: 10.1145/2601248.2601268.

[31] D. S. Cruzes and T. Dybå, “Synthesizing evidence in software engineering research,” in *Proceedings of the 2010 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, 2010, pp. 1–10.

[32] D. Walsh, and S. Downe, “Meta-synthesis method for qualitative research: a literature review” in *Journal of Advanced Nursing*, 2005, pp. 204-211.

[33] M. A. Mohammed, R. J. Moles, and T. F. Chen, “Meta-synthesis of qualitative research: the challenges and opportunities” in *International Journal of Clinical Pharmacy*, 2016, pp. 695-704

[34] R. Campbell, P. Pound, C. Pope, N. Britten, R. Pill, M. Morgan et al. “Evaluating meta-ethnography: a synthesis of qualitative research on lay experiences of diabetes and diabetes care.” in *Social Sci Med*., 2003, pp. 671–684.

[35] S. Child, V. Goodwin, R. Garside, T. Jones-Hughes, K. Boddy, and K. Stein, “Factors influencing the implementation of fall-prevention programmes: a systematic review and synthesis of qualitative studies” in *Implement Sci.*, 2012, pp. 1-14.

[36] R. J. Skinner, R. R. Nelson, and W. Chin, "Synthesizing Qualitative Evidence: A Roadmap for Information Systems Research," in *Journal of the Association for Information Systems*, 2022, pp. 639-677.

[37] M. Khenwar, A. Sisodia, S. Vishnoi, and R. Kumar, “Exploration: Cloud Computing Scheduling Techniques” in *Scandinavian Journal of Information Systems*, 2023, pp. 673-679.

[38] F. S. Prity, M. H. Gazi, K. M. Uddin, “A review of task scheduling in cloud computing based on nature inspired optimization algorithm” in *Cluster Computing*, 2023, pp. 3037-3067.

[39] Z. Sun, B. Zhang, C. Gu, R. Xie, B. Qian and H. Huang, "ET2FA: A Hybrid Heuristic Algorithm for Deadline-Constrained Workflow Scheduling in Cloud," in *IEEE Transactions on Services Computing*, 2023, pp. 1807-1821.

[40] R. M. Singh, L. K. Awasthi, G. Sikka, “Towards Metaheuristic Scheduling Techniques in Cloud and Fog: An Extensive Taxonomic Review”, in *ACM Computing Surveys*, 2023 pp. 1-43.

[41] Y. Xie, F. -X. Gui, W. -J. Wang and C. -F. Chien, "A Two-stage Multi-population Genetic Algorithm with Heuristics for Workflow Scheduling in Heterogeneous Distributed Computing Environments," in IEEE Transactions on Cloud Computing, 2023, pp. 1446-1460.

[42] G. Saravanan, S. Neelakandan, P. Ezhumalai et al., "Improved wild horse optimization with levy flight algorithm for effective task scheduling in cloud computing" in *Journal of Cloud Comp*uting, 2023, pp. 12-24

Reviewed Studies

[S1] F. Alhaidari and T. Z. Balharith, "Enhanced round-robin algorithm in the cloud computing environment for optimal task scheduling," *Computers*, vol. 10, no. 5, 2021, doi: 10.3390/computers10050063.

[S2] N. K. Walia *et al.*, "An Energy-Efficient Hybrid Scheduling Algorithm for Task Scheduling in the Cloud Computing Environments," *IEEE Access*, vol. 9, pp. 117325–117337, 2021, doi: 10.1109/ACCESS.2021.3105727.

[S3] M. Abd Elaziz, L. Abualigah, and I. Attiya, "Advanced optimization technique for scheduling IoT tasks in cloud-fog computing environments," *Future Generation Computer Systems*, vol. 124, pp. 142–154, Nov. 2021, doi: 10.1016/j.future.2021.05.026.

[S4] K. Dubey and S. C. Sharma, "A novel multi-objective CR-PSO task scheduling algorithm with deadline constraint in cloud computing," *Sustainable Computing: Informatics and Systems*, vol. 32, p. 100605, Dec. 2021, doi: 10.1016/j.suscom.2021.100605.

[S5] S. Velliangiri, P. Karthikeyan, V. M. Arul Xavier, and D. Baswaraj, "Hybrid electro search with genetic algorithm for task scheduling in cloud computing," *Ain Shams Engineering Journal*, vol. 12, no. 1, pp. 631–639, Mar. 2021, doi: 10.1016/j.asej.2020.07.003.

[S6] X. Guo, "Multi-objective task scheduling optimization in cloud computing based on fuzzy self-defense algorithm," *Alexandria Engineering Journal*, vol. 60, no. 6, pp. 5603–5609, 2021, doi: https://doi.org/10.1016/j.aej.2021.04.051.

[S7] K. Dubey and S. C. Sharma, "A hybrid multi-faceted task scheduling algorithm for cloud computing environment," *International Journal of System Assurance Engineering and Management*, Mar. 2021, doi: 10.1007/s13198-021-01084-0.

[S8] M. Y. Uddin, H. A. Abdeljaber, and T. A. Ahanger, "Development of a Hybrid Algorithm for efficient Task Scheduling in Cloud Computing environment using Artificial Intelligence," *International Journal of Computers, Communications and Control*, vol. 16, no. 5, pp. 1–12, 2021, doi: 10.15837/ijccc.2021.5.4087.

[S9] A. Ullah and N. M. Nawi, “An improved in tasks allocation system for virtual machines in cloud computing using HBAC algorithm,” *J Ambient Intell Humaniz Comput*, 2021, doi: 10.1007/s12652-021-03496-z.

[S10] R. Gulbaz, A. B. Siddiqui, N. Anjum, A. A. Alotaibi, T. Althobaiti, and N. Ramzan, “Balancer genetic algorithm-a novel task scheduling optimization approach in cloud computing,” *Applied Sciences (Switzerland)*, vol. 11, no. 14, 2021, doi: 10.3390/app11146244.

[S11] J. Ge, D. Yu, and Y. Fang, “Multi-dimensional QoS Cloud Computing Task Scheduling Strategy Based on Improved Ant Colony Algorithm,” in *Journal of Physics: Conference Series*, 2021, vol. 1848, no. 1. doi: 10.1088/1742-6596/1848/1/012031.

[S12] A. Y. Hamed and M. H. Alkinani, “Task Scheduling Optimization in Cloud Computing Based on Genetic Algorithms,” *Computers, Materials & Continua*, vol. 69, no. 3, pp. 3289–3301, 2021, doi: 10.32604/cmc.2021.018658.

[S13] F. Ebadifard and S. M. Babamir, “Autonomic task scheduling algorithm for dynamic workloads through a load balancing technique for the cloud-computing environment,” *Cluster Comput*, vol. 24, no. 2, pp. 1075–1101, 2021, doi: 10.1007/s10586-020-03177-0.

[S14] Y. Su, Z. Bai, and D. Xie, “The optimizing resource allocation and task scheduling based on cloud computing and Ant Colony Optimization Algorithm,” *J Ambient Intell Humaniz Comput*, 2021, doi: 10.1007/s12652-021-03445-w.

[S15] M. S. A. Khan and R. Santhosh, “Task scheduling in cloud computing using hybrid optimization algorithm,” *Soft comput*, 2021, doi: 10.1007/s00500-021-06488-5.

[S16] J. Xu, Z. Zhang, Z. Hu, L. Du, and X. Cai, “A Many-Objective Optimized Task Allocation Scheduling Model in Cloud Computing,” *Applied Intelligence*, vol. 51, no. 6, pp. 3293–3310, Jun. 2021, doi: 10.1007/s10489-020-01887-x.

[S17] L. Jia, K. Li, X. Shi, and O. Kaiwartya, “Cloud Computing Task Scheduling Model Based on Improved Whale Optimization Algorithm,” *Wirel. Commun. Mob. Comput.*, vol. 2021, Jan. 2021, doi: 10.1155/2021/4888154.

[S18] L. S. Subhash and R. Udayakumar, “Sunflower Whale Optimization Algorithm for Resource Allocation Strategy in Cloud Computing Platform,” *Wirel. Pers. Commun.*, vol. 116, no. 4, pp. 3061–3080, Feb. 2021, doi: 10.1007/s11277-020-07835-9.

[S19] B. Saemi, M. Sadeghilalimi, A. A. Rahmani Hosseinabadi, M. Mouhoub, and S. Sadaoui, “A New Optimization Approach for Task Scheduling Problem Using Water Cycle Algorithm in Mobile Cloud Computing,” in *2021 IEEE Congress on*

Evolutionary Computation (CEC), 2021, pp. 530–539. doi: 10.1109/CEC45853.2021.9504780.

[S20] H. ben Alla, S. ben Alla, A. Ezzati, and A. Touhafi, "A Novel Multiclass Priority Algorithm for Task Scheduling in Cloud Computing," *J. Supercomput.*, vol. 77, no. 10, pp. 11514–11555, Oct. 2021, doi: 10.1007/s11227-021-03741-4.

[S21] M. Abd Elaziz and I. Attiya, "An Improved Henry Gas Solubility Optimization Algorithm for Task Scheduling in Cloud Computing," *Artif. Intell. Rev.*, vol. 54, no. 5, pp. 3599–3637, Jun. 2021, doi: 10.1007/s10462-020-09933-3.

[S22] M. Tanha, M. Hosseini Shirvani, and A. M. Rahmani, "A Hybrid Meta-Heuristic Task Scheduling Algorithm Based on Genetic and Thermodynamic Simulated Annealing Algorithms in Cloud Computing Environments," *Neural Comput. Appl.*, vol. 33, no. 24, pp. 16951–16984, Dec. 2021, doi: 10.1007/s00521-021-06289-9.

[S23] K. Mishra, R. Pradhan, and S. K. Majhi, "Quantum-Inspired Binary Chaotic Salp Swarm Algorithm (QBCSSA)-Based Dynamic Task Scheduling for Multiprocessor Cloud Computing Systems," *J. Supercomput.*, vol. 77, no. 9, pp. 10377–10423, Sep. 2021, doi: 10.1007/s11227-021-03695-7.

[S24] L. Ni, X. Sun, X. Li, J. Zhang, and C. de Maio, "GCWOAS2: Multiobjective Task Scheduling Strategy Based on Gaussian Cloud-Whale Optimization in Cloud Computing," *Intell. Neuroscience*, vol. 2021, Jan. 2021, doi: 10.1155/2021/5546758.

[S25] P. Singh, "Scheduling tasks based on branch and bound algorithm in cloud computing environment," in *2021 8th International Conference on Signal Processing and Integrated Networks (SPIN)*, Aug. 2021, pp. 41–46. doi: 10.1109/SPIN52536.2021.9565972.

[S26] L. Abualigah and A. Diabat, "A Novel Hybrid Antlion Optimization Algorithm for Multi-Objective Task Scheduling Problems in Cloud Computing Environments," *Cluster Comput*, vol. 24, no. 1, pp. 205–223, Mar. 2021, doi: 10.1007/s10586-020-03075-5.

[S27] T. Oudaa, H. Gharsellaoui, and S. ben Ahmed, "An Agent-Based Model for Resource Provisioning and Task Scheduling in Cloud Computing Using DRL," *Procedia Comput. Sci.*, vol. 192, no. C, pp. 3795–3804, Jan. 2021, doi: 10.1016/j.procs.2021.09.154.

[S28] A. Gupta, K. M. Soni, and S. Singhal, "A hybrid metaheuristic and machine learning algorithm for optimal task scheduling in cloud computing," in *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, Jul. 2021, pp. 1–7. doi: 10.1109/ICCCNT51525.2021.9579688.

[S29] F. Hoseiny, S. Azizi, M. Shojafar, F. Ahmadiazar, and R. Tafazolli, "PGA: A Priority-aware Genetic Algorithm for Task Scheduling in Heterogeneous Fog-Cloud Computing," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, May 2021, pp. 1–6. doi: 10.1109/INFOCOMWKSHPS51825.2021.9484436.

[S30] Z. Shafahi and A. Yari, "An efficient task scheduling in cloud computing based on ACO algorithm," in *2021 12th International Conference on Information and Knowledge Technology (IKT),* Dec. 2021, pp. 72–77. doi: 10.1109/IKT54664.2021.9685674.

[S31] G. Sreenivasulu and I. Paramasivam, "Hybrid optimization algorithm for task scheduling and virtual machine allocation in cloud computing," *Evol Intell*, vol. 14, no. 2, pp. 1015–1022, 2021, doi: 10.1007/s12065-020-00517-2.

[S32] S. Mangalampalli, S. K. Swain, and V. K. Mangalampalli, "Prioritized Energy Efficient Task Scheduling Algorithm in Cloud Computing Using Whale Optimization Algorithm," *Wirel Pers Commun*, 2021, doi: 10.1007/s11277-021-09018-6.

[S33] X. Fu, Y. Sun, H. Wang, and H. Li, "Task scheduling of cloud computing based on hybrid particle swarm algorithm and genetic algorithm," *Cluster Comput*, May 2021, doi: 10.1007/s10586-020-03221-z.

[S34] M. S. Sanaj and P. M. Joe Prathap, "An efficient approach to the map-reduce framework and genetic algorithm based whale optimization algorithm for task scheduling in cloud computing environment," *Mater Today Proc*, vol. 37, no. Part 2, pp. 3199–3208, 2021, doi: 10.1016/j.matpr.2020.09.064.

[S35] A. Wilczyński and J. Kołodziej, "Modelling and simulation of security-aware task scheduling in cloud computing based on Blockchain technology," *Simul Model Pract Theory*, vol. 99, p. 102038, 2020, doi: https://doi.org/10.1016/j.simpat.2019.102038.

[S36] S. K. Panda, S. S. Nanda, and S. K. Bhoi, "A Pair-Based Task Scheduling Algorithm for Cloud Computing Environment," *J. King Saud Univ. Comput. Inf. Sci.*, vol. 34, no. 1, pp. 1434–1445, Jan. 2022, doi: 10.1016/j.jksuci.2018.10.001.

[S37] A. K. Bhardwaj, Y. Gajpal, C. Surti, and S. S. Gill, "HEART: Unrelated parallel machines problem with precedence constraints for task scheduling in cloud computing using heuristic and meta-heuristic algorithms," *Softw Pract Exp*, vol. 50, no. 12, pp. 2231–2251, 2020, doi: 10.1002/spe.2890.

[S38] H. Singh, S. Tyagi, and P. Kumar, "Crow–penguin optimizer for multiobjective task scheduling strategy in cloud computing," *International Journal of Communication Systems*, vol. 33, no. 14, 2020, doi: 10.1002/dac.4467.

[S39] M. S. Sanaj and P. M. Joe Prathap, “Nature inspired chaotic squirrel search algorithm (CSSA) for multi objective task scheduling in an IAAS cloud computing atmosphere,” *Engineering Science and Technology, an International Journal*, vol. 23, no. 4, pp. 891–902, 2020, doi: https://doi.org/10.1016/j.jestch.2019.11.002.

[S40] K. M. S. U. Bandaranayake, K. P. N. Jayasena, and B. T. G. S. Kumara, “An Efficient Task Scheduling Algorithm using Total Resource Execution Time Aware Algorithm in Cloud Computing,” in *2020 IEEE International Conference on Smart Cloud (SmartCloud),* Nov. 2020, pp. 29–34. doi: 10.1109/SmartCloud49737.2020.00015.

[S41] Z. Gao, Y. Wang, Y. Gao, and X. Ren, “Multiobjective noncooperative game model for cost-based task scheduling in cloud computing,” *Concurr Comput*, vol. 32, no. 7, 2020, doi: 10.1002/cpe.5570.

[S42] K. Li, Y. Wang, and M. Liu, “A non-cooperative game model for reliability-based task scheduling in cloud computing,” *International Journal of Communication Systems*, vol. 33, no. 15, 2020, doi: 10.1002/dac.4512.

[S43] R. G. Shooli and M. M. Javidi, “Using gravitational search algorithm enhanced by fuzzy for resource allocation in cloud computing environments,” *SN Appl Sci*, vol. 2, no. 2, 2020, doi: 10.1007/s42452-020-2014-y.

[S44] B. Liang, X. Dong, Y. Wang, and X. Zhang, “A Low-Power Task Scheduling Algorithm for Heterogeneous Cloud Computing,” *J. Supercomput.*, vol. 76, no. 9, pp. 7290–7314, Sep. 2020, doi: 10.1007/s11227-020-03163-8.

[S45] K. Karthiban and J. S. Raj, “An Efficient Green Computing Fair Resource Allocation in Cloud Computing Using Modified Deep Reinforcement Learning Algorithm,” *Soft Comput.*, vol. 24, no. 19, pp. 14933–14942, Oct. 2020, doi: 10.1007/s00500-020-04846-3.

[S46] X. Huang, C. Li, H. Chen, and D. An, “Task Scheduling in Cloud Computing Using Particle Swarm Optimization with Time Varying Inertia Weight Strategies,” *Cluster Comput*, vol. 23, no. 2, pp. 1137–1147, Jun. 2020, doi: 10.1007/s10586-019-02983-5.

[S47] J. Li and Y. Han, “A Hybrid Multi-Objective Artificial Bee Colony Algorithm for Flexible Task Scheduling Problems in Cloud Computing System,” *Cluster Comput*, vol. 23, no. 4, pp. 2483–2499, Dec. 2020, doi: 10.1007/s10586-019-03022-z.

[S48] Q. Yang and X. Xie, “Research on Cloud Computing Task Scheduling Based on Improved Evolutionary Algorithm,” in *Proceedings of the 2020 3rd International Conference on E-Business, Information Management and Computer Science*, 2020, pp. 566–572. doi: 10.1145/3453187.3453396.

[S49] Y. Liang, Q. Cui, L. Gan, Z. Xie, and S. Zhai, “A Cloud Computing Task Scheduling Strategy Based on Improved Particle Swarm Optimization,” in *Proceedings of the 2020 2nd International Conference on Big Data and Artificial Intelligence*, 2020, pp. 543–549. doi: 10.1145/3436286.3436500.

[S50] X. Fu, Y. Hu, and Y. Sun, “Cloud Computing Task Scheduling Based on Improved Differential Evolution Algorithm,” in *Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture*, 2020, pp. 118–124. doi: 10.1145/3421766.3421785.

[S51] C. Chen, H. Shi, Z. Wang, and Z. Yu, “A Task Scheduling Algorithm Based on Big.LITTLE Architecture in Cloud Computing,” in *2020 6th International Conference on Big Data and Information Analytics (BigDIA)*, Dec. 2020, pp. 94–99. doi: 10.1109/BigDIA51454.2020.00023.

[S52] M. T. Alotaibi, M. S. Almalag, and K. Werntz, “Task Scheduling in Cloud Computing Environment Using Bumble Bee Mating Algorithm,” in *2020 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT)*, Dec. 2020, pp. 01–06. doi: 10.1109/GCAIoT51063.2020.9345824.

[S53] A. K. Maurya, “Resource and Task Clustering based Scheduling Algorithm for Workflow Applications in Cloud Computing Environment,” in *2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC)*, Nov. 2020, pp. 566–570. doi: 10.1109/PDGC50313.2020.9315806.

[S54] Z. He, J. Dong, Z. Li, and W. Guo, “Research on Task Scheduling Strategy Optimization Based onACO in Cloud Computing Environment,” in *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, Jun. 2020, pp. 1615–1619. doi: 10.1109/ITOEC49072.2020.9141743.

[S55] N. M. Alhakkak, “Modeling Smart Cloud Computing Resource Allocation in E-Learning,” in *2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL)*, Jul. 2020, pp. 274–277. doi: 10.1109/CVIDL51233.2020.00-87.

[S56] K. Ma, A. Bagula, O. Ajayi, and C. Nyirenda, “Aiming at QoS: A Modified DE Algorithm for Task Allocation in Cloud Computing,” in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, Jun. 2020, pp. 1–7. doi: 10.1109/ICC40277.2020.9148980.

[S57] Z. Zong, “An Improvement of Task Scheduling Algorithms for Green Cloud Computing,” in *2020 15th International Conference on Computer Science & Education (ICCSE)*, Aug. 2020, pp. 654–657. doi: 10.1109/ICCSE49874.2020.9201785.

[S58] S. M. G. Kashikolaei, A. A. R. Hosseinabadi, B. Saemi, M. B. Shareh, A. K. Sangaiah, and G.-B. Bian, "An Enhancement of Task Scheduling in Cloud Computing Based on Imperialist Competitive Algorithm and Firefly Algorithm," *J. Supercomput.*, vol. 76, no. 8, pp. 6302–6329, Aug. 2020, doi: 10.1007/s11227-019-02816-7.

[S59] J. Yang, B. Jiang, Z. Lv, and K.-K. R. Choo, "A Task Scheduling Algorithm Considering Game Theory Designed for Energy Management in Cloud Computing," *Future Gener. Comput. Syst.*, vol. 105, no. C, pp. 985–992, Apr. 2020, doi: 10.1016/j.future.2017.03.024.

[S60] A. Khodar, L. v. Chernenkaya, I. Alkhayat, H. A. Fadhil Al-Afare, and E. N. Desyatirikova, "Design Model to Improve Task Scheduling in Cloud Computing Based on Particle Swarm Optimization," in *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, Jan. 2020, pp. 345–350. doi: 10.1109/EIConRus49466.2020.9039501.

[S61] X. Shi, X. Zhang, and M. Xu, "A Self-Adaptive Preferred Learning Differential Evolution Algorithm for Task Scheduling in Cloud Computing," in *2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA)*, Aug. 2020, pp. 145–148. doi: 10.1109/AEECA49918.2020.9213606.

[S62] F. Fellir, A. el Attar, K. Nafil, and L. Chung, "A multi-Agent based model for task scheduling in cloud-fog computing platform," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, Feb. 2020, pp. 377–382. doi: 10.1109/ICIoT48696.2020.9089625.

[S63] W. Li, Z. Tang, and F. Qi, "A Hybrid Task Scheduling Algorithm Combining Symbiotic Organisms Search with Fuzzy Logic in Cloud Computing," in *2020 IEEE 23rd International Conference on Computational Science and Engineering (CSE)*, Dec. 2020, pp. 16–23. doi: 10.1109/CSE50738.2020.00010.

[S64] F. Ebadifard, S. M. Babamir, and S. Barani, "A Dynamic Task Scheduling Algorithm Improved by Load Balancing in Cloud Computing," in *2020 6th International Conference on Web Research (ICWR)*, Apr. 2020, pp. 177–183. doi: 10.1109/ICWR49608.2020.9122287.

[S65] X. Wei, "Task scheduling optimization strategy using improved ant colony optimization algorithm in cloud computing," *J Ambient Intell Humaniz Comput*, 2020, doi: 10.1007/s12652-020-02614-7.

[S66] Z. Gao, W. Hao, R. Zhang, and S. Yang, "Markov decision process-based computation offloading algorithm and resource allocation in time constraint for mobile cloud computing," *IET Communications*, vol. 14, no. 13, pp. 2068–2078, 2020, doi: 10.1049/iet-com.2020.0062.

[S67] R. Khorsand and M. Ramezanpour, “An energy-efficient task-scheduling algorithm based on a multi-criteria decision-making method in cloud computing,” *International Journal of Communication Systems*, vol. 33, no. 9, 2020, doi: 10.1002/dac.4379.

[S68] K. R. P. Kumar, K. Kousalya, K. R. Prasanna Kumar, and K. Kousalya, “Amelioration of task scheduling in cloud computing using crow search algorithm,” *Neural Comput Appl*, vol. 32, no. 10, SI, pp. 5901–5907, May 2020, doi: 10.1007/s00521-019-04067-2.

[S69] A. Belgacem, K. Beghdad-Bey, and H. Nacer, “Dynamic resource allocation method based on Symbiotic Organism Search algorithm in cloud computing,” *IEEE Transactions on Cloud Computing*, vol. 10, no. 3, pp. 1714–1725, 2020, doi: 10.1109/TCC.2020.3002205.

[S70] S. A. Alsaidy, A. D. Abbood, and M. A. Sahib, “Heuristic initialization of PSO task scheduling algorithm in cloud computing,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, Part A, pp. 2370–2382, 2020, doi: https://doi.org/10.1016/j.jksuci.2020.11.002.

[S71] S. N. Prasad, S. Kulkarni, P. Venkatareddy, S. Nagendra Prasad, S. Kulkarni, and P. Venkatareddy, “Cache Aware Task Scheduling Algorithm for Heterogeneous Cloud Computing Environment,” in *Proceedings - 2020 5th International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN 2020*, 2020, pp. 154–158. doi: 10.1109/ICRCICN50933.2020.9296177.