OPTIMIZED TASK SCHEDULING BASED ON HYBRID SYMBIOTIC ORGANISMS SEARCH ALGORITHMS FOR CLOUD COMPUTING ENVIRONMENT

ABDULLAHI, MOHAMMED

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy (Computer Science)

Faculty of Computing
Universiti Teknologi Malaysia

JULY 2017

To my beloved parents, Audu Saba and Maryam Salihu

ACKNOWLEDGEMENT

First and Foremost, all praises be to ALLAH, the Almighty, on whom we ultimately depend for sustenance and guidance.

My gratitude goes to my supervisor, Assoc. Prof. Dr. MD Asri bin Ngadi for his guidance, patience and diligence without which this feat of ensuring the success of this research work cannot be achieved. Many thanks also, to my examiners Prof. Dr. Abdul Samad bin Ismail and Prof. Dr. Saadiah binti Yahya for providing feedbacks that helped immensely in improving this thesis. I Also, I acknowledge the support received from lecturers, faculty members and staff of faculty of computing, Universiti Teknologi Malaysia for the series of programmes which aided this work.

Many thanks to my friends and colleagues especially Dr. Shaffii Muhammad Abdulhamid, Dr. Abubakar Muhammad Umaru, Dr. Gaddafi Abdulsalam, Dr. Barroon Ismaeel Ahmad, Dr. Mustapha Aminu Bagiwa, Dr. Ibrahim Fulatan and Dr. Ibrahim Abdullahi, Salihu Idi Dishing, Usman Joda, Sahabi Yusuf Ali, Aliyu Mohammed Abali, Ahmed Aliyu, Jibrin Ndejiko and Abdullahi Mohammed Lapai for all their inputs during the numerous discussions. Also, Barrister Jamila Abali, Barrister Abdullahi Adamu Mohammed, Musa A. Musa, and Ashafa Ibrahim who have been a source of moral support through their fervent prayers. I am lucky and glad to have them in my life.

I am extremely grateful to my family for their moral and financial support as well encouragements. Thank you for always being my source of strength while pursuing my goals in life. Firstly, I would like to express my deepest gratitude to my parents Audu Saba and Maryam Salihu, for everything they have done for me in life. None of my achievements would have been possible without their steadfast encouragement and support. A special thanks goes to my wife, Amina Isa, and children, Asiyah, Abdulrahaman, and Saleemah for their endless love, support and prayers throughout my PhD. Although we have lived apart for years now, they've always been there when I needed them the most. I must also thank my brothers and sisters especially Ibrahim Abdullahi (Ndasebe), for his priceless companionship.

Finally, my gratitude goes to Tertiary Education Trust FUND (TETFUND) Nigeria, for providing the funding through Ahmadu Bello University, Zaria-Nigeria, without which this work couldn't have been possible. Special thanks to Prof. Sahalu Balarebe Junaidu, Prof. Kabiru Bala, Prof. A.K. Adamu, and Prof. Ibrahim Sule.

ABSTRACT

In Cloud Computing model, users are charged according to the usage of resources and desired Quality of Service (QoS). Task scheduling algorithms are responsible for specifying adequate set of resources to execute user applications in the form of tasks, and schedule decisions of task scheduling algorithms are based on QoS requirements defined by the user. Task scheduling problem is an NP-Complete problem, due to the NP-Complete nature of task scheduling problems and huge search space presented by large scale problem instances, many of the existing solution algorithms incur high computational complexity and cannot effectively obtain global optimum solutions. Recently, Symbiotic Organisms Search (SOS) has been applied to various optimization problems and results obtained were found to be competitive with state-of-the-art metaheuristic algorithms. However, similar to the case other metaheuristic optimization algorithms, the efficiency of SOS algorithm deteriorates as the size of the search space increases. Moreover, SOS suffers from local optima entrapment and its static control parameters cannot maintain a balance between local and global search. In this study, Cooperative Coevolutionary Constrained Multi-objective Symbiotic Organisms Search (CC-CMSOS), Cooperative Coevolutionary Constrained Multi-objective Memetic Symbiotic Organisms Search (CC-CMMSOS), and Cooperative Coevolutionary Constrained Multi-objective Adaptive Benefit Factor Symbiotic Organisms Search (CC-CMABFSOS) algorithms are proposed to solve constrained multi-objective large scale task scheduling optimization problem on IaaS cloud computing environment. To address the issue of scalability, the concept of Cooperative Coevolutionary for enhancing SOS named CC-CMSOS make SOS more efficient for solving large scale task scheduling problems. CC-CMMSOS algorithm further improves the performance of SOS algorithm by hybridizing with Simulated Annealing (SA) to avoid entrapment in local optima for global convergence. Finally, CC-CMABFSOS algorithm adaptively turn SOS control parameters to balance the local and global search procedure for faster convergence speed. The performance of the proposed CC-CMSOS, CC-CMMSOS, and CC-CMABFSOS algorithms are evaluated on CloudSim simulator, using both standard workload traces and synthesized workloads for larger problem instances of up to 5000. Moreover, CC-CMSOS, CC-CMMSOS, and CC-CMABFSOS algorithms are compared with multi-objective optimization algorithms, namely, EMS-C, ECMSMOO, and BOGA. The CC-CMSOS, CC-CMMSOS, and CC-CMABFSOS algorithms obtained significant improved optimal trade-offs between execution time (makespan) and financial cost (cost) while meeting deadline constraints with no computational overhead. The performance improvements obtained by the proposed algorithms in terms of hypervolume ranges from 8.72% to 37.95% across the workloads. Therefore, the proposed algorithms have potentials to improve the performance of QoS delivery.

ABSTRAK

Dalam model Pengkomputeran Awan, pengguna dikenakan caj mengikut penggunaan sumber dan Kualiti Perkhidmatan (QoS) yang dikehendaki. Algoritma Penjadualan Tugas bertanggungjawab menentukan set sumber yang mencukupi untuk melaksanakan aplikasi pengguna dalam bentuk tugas, dan keputusan jadual algoritma penjadualan tugas adalah berdasarkan keperluan QoS yang ditakrif oleh pengguna. Masalah penjadualan tugas merupakan masalah *NP-Complete* yang disebabkan oleh sifat masalah penjadualan tugas *NP-Complete* dan ruang carian besar yang ditunjukkan melalui masalah berskala besar, kebanyakan daripada penyelesaian algoritma sedia ada mendatangkan kerumitan pengiraan tinggi dan tidak boleh mendapatkan penyelesaian optimum global secara berkesan. Baru-baru ini, Carian Organisme Simbiotik (SOS) telah diguna untuk pelbagai masalah pengoptimuman dan hasil yang diperolehi didapati bersaing dengan algoritma metaheuristik yang canggih. Namun begitu, serupa dengan algoritma pengoptimuman metaheuristik yang lain, kecekapan algoritma SOS merosot apabila saiz ruang carian meningkat. Selain itu, SOS mengalami kerugian disebabkan perangkap optima tempatan dan parameter kawalan statiknya tidak dapat mengekalkan keseimbangan antara carian tempatan dengan global. Dalam kajian ini, algoritma-algoritma Carian Organisme Simbiotik Pelbagai-Objektif Kekangan Evolusi Sama Koperatif (CC-CMSOS), Carian Organisme Simbiotik Mimetik Pelbagai-Objektif Kekangan Evolusi Sama Koperatif (CC-CMMSOS), dan Carian Organisme Simbiotik Faktor Suai Faedah Pelbagai-Objektif Kekangan Evolusi Sama Koperatif (CC-CMABFSOS) dicadangkan untuk menyelesaikan masalah kekangan pengoptimuman penjadualan tugas berskala besar pelbagai objektif pada persekitaran pengkomputeran awan IaaS. Untuk menangani isu pengskalaan, konsep Evolusi Sama Koperatif bagi meningkatkan SOS, iaitu CC-CMSOS menjadikan SOS lebih cekap untuk menyelesaikan masalah penjadualan tugas berskala besar. Algoritma CC-CMMSOS juga meningkatkan prestasi algoritma SOS dengan menghibridkannya dengan Simulasi Penyepulih-Indapan (SA) untuk mengelakkan perangkap dalam optima tempatan untuk penumpuan global. Akhir sekali, algoritma CC-CMABFSOS disesuaikan dengan parameter kawalan SOS untuk mengimbangi prosedur carian tempatan dan global bagi kelajuan penumpuan yang lebih pantas. Prestasi CC-CMSOS, CC-CMMSOS, dan algoritma CC-CMABFSOS yang dicadangkan dinilai pada simulator SimAwam, menggunakan kedua-dua kesan beban kerja standard dan beban kerja yang disintesis untuk contoh-contoh masalah yang lebih besar sehingga 5000. Selain itu, CC-CMSOS, CC-CMMSOS, dan algoritma CC-CMABFSOS dibandingkan dengan algoritma pengoptimuman pelbagai objektif, iaitu EMS-C, ECMSMOO dan BOGA. Algoritma-algoritma CC-CMSOS, CC-CMMSOS dan CC-CMABFSOS telah mencapai penigkatan keseimbangan optima yang ketara antara masa pelaksanaan dengan kos kewangan yang menepati tarikh akhir tanpa overhed pengiraan. Peningkatan prestasi algoritma yang dicadangkan dari segi julat hiperisipadu adalah daripada 8.72% kepada 37.95% merentasi beban kerja. Oleh itu, algoritma yang dicadangkan mempunyai potensi untuk meningkatkan prestasi penghantaran QoS.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACO	-	Ant Colony Optimization
BA	-	Bee Algorithm
BoT	-	Bag of Tasks
C-PSO	-	Catfish Particle Swarm Optimization
CC-CMSOS	-	Cooperative Coevolutionary Constrained Multi-objective Symbiotic Organisms Search
CC-CMSOS	-	Cooperative Coevolutionary Constrained Multi-objective Memetic Symbiotic Organisms Search
CC-CMSOS	-	Cooperative Coevolutionary Constrained Multi-objective Adaptive Benefit Factor Symbiotic Organisms Search
CCGA	-	Cooperative Co-evolutionary Genetic Algorithm
CGA	-	Co-evolutionary Genetic Algorithm
CLS	-	Chaotic Local Search
CPU	-	Central Processing Unit
CRO	-	Chemical Reaction Optimization
CS	-	Cokoo Search
CSO	-	Cat Swarm Optimization
DAG	-	Direct Acyclic Graph
DE	-	Differential Evolution
EFT	-	Earliest Finish Time
ETC	-	Execution Time to Compute
GA	-	Genetic Algorithm
HEFT	-	Heterogeneous Earliest Finish Time
HTTP	-	Hyper Text Transfer Protocol
IaaS	-	Infrastructure as a Service
LCA	-	League Championship Algorithm
LLCF	-	Least Loaded Cloud First
MAGA	-	Multi-Agent Genetic Algorithm
MIPS	-	Million Instructions Per Second
NSGA II	-	Non-dominated Sorting Genetic Algorithm

PaaS	-	Platform as a Service
PEFT	-	Predict Earliest Finish Time
PBA	-	Particle Bee Algorithm
PSO	-	Particle Swarm Optimization
QoS	-	Quality of Service
SA	-	Simulated Annealing
SaaS	-	Software as a Service
SFLA	-	Shuffled Frog Leaping Algorithm
SI	-	Swarm Intelligence
SLA	-	Service Level Agreement
SOS	-	Symbiotic Organisms Search
TS	-	Tabu Search
VM	-	Virtual Machine
VNS	-	Variable Neighbourhood Search

LIST OF SYMBOLS

ω	-	Inertia weight
τ_{ij}	-	Pheromone deposit
α	-	Pheromone deposit control parameter
η_{ij}	-	Heuristic information
β	-	Heuristic information control parameter
δ^i	-	Temperature descending rate
β_1	-	Benefit factor 1
β_2	-	Benefit factor 2
γ_f	-	Adaptive weight parameter
P_τ	-	Probability of success

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Overview

Cloud Computing provides scalable and elastic access to resources on pay-per-use model, this eliminates the need for organizations and companies to invest in owning hardware and software resources. Large scale scientific and industrial applications like astronomy, physics, bioinformatics, data mining and business-informatics demand high computational power for their execution in reasonable amount of time (Deelman *et al.*, 2009; Juve *et al.*, 2013). To meet up with the increasing computational demand of large scale applications, Cloud Computing is witnessing high rate deployment of large scale applications in recent times, because Cloud provides elastic and flexible compute resources which can be leased on pay-per-use model (Foster *et al.*, 2008). Large scale applications consist of huge number of tasks which are executed on Infrastructure-as-a-Service clouds. Cloud Computing services are offered in form of Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). SaaS service model delivers applications to end users via Internet and these applications are accessed using client applications like web browsers. SaaS is usually used for service applications like web-mail, and document editing applications. PaaS provides application developers with environment for development, testing and hosting of their applications.

Moreover, IaaS provides access to flexible and scalable computing resources for large scale application deployment. With IaaS model, virtualized compute resources called virtual machines (VMs) with pre-configured CPU, storage, memory, and bandwidth are leased to users by paying for what they use only. Various VM instances are available to the users at different prices to serve their various application needs, this gives users the freedom to control compute resource at their disposal. IaaS provides three inherent benefits to users. First, users lease resource on demand, and charged based on pay-per-usage similar to basic utilities like electricity, gas, and water. This enables users to shrink or expand their resource subscription base on the needs of their application. Second, IaaS Cloud provides direct resource provisioning which improve the performance of user applications. Third, users can demand for leased resources any time and any where according to the desired level of service. However, determining the adequate number of resources to execute a set of large scale task on IaaS Cloud is still an open problem (Wu *et al.*, 2015).

Furthermore, there are two parties in Cloud Computing environment, the Cloud service providers and Cloud service consumers. Providers own high computing resources housed in data centers, and the resources are leased to consumers on pay-per-use model. Whereas, the Cloud service consumers lease resources from providers to execute user's applications. On one hand, the target of provider is to maximize return on investment as much as possible. To that effect, providers wants to schedule as many user applications as possible on each resource to maximize the utilization of resources. On the other hand, consumers wish to have their requests served at minimal cost. To satisfy the consumers requests, Cloud services must be provisioned according to the required quality of service (QoS) and service level agreement (SLA). QoS is the capacity of guaranteeing a certain level of performance based on some criteria defined by consumer (Rimal *et al.*, 2011) whereas SLA is a legal written document describing QoS requirements (Rimal *et al.*, 2009). However, scheduling large scale tasks based on user's QoS is still a challenging issue. Therefore, the focus of this thesis is on QoS task scheduling in IaaS Clouds on large scale perspective.

1.2 Problem Background

Task scheduling algorithms play critical role in harnessing the benefits of Cloud Computing for efficient execution of large scale applications on IaaS Cloud. Scheduling algorithms are responsible for specifying adequate set of compute resources to execute user applications in the form of tasks, and schedule decisions of the algorithms are based on QoS requirements defined by the user. Meeting desired QoS requirements depends on effective use of underlying compute resources, therefore, a task scheduler has to be aware of respective challenges introduced by essential features of the Cloud Computing environment. Unlike other distributed computing platforms such as clusters and grids, Cloud users control the types of compute resources to be used for executing their applications which introduced a number of challenges to task scheduling algorithms. Firstly, flexibility prompts the need for task schedulers to be able to determine the adequate number of resources for executing given set tasks without violating user QoS requirements, and to avoid resource underprovisioning or overprovisioning. Secondly, scheduling algorithms must be able to efficiently cope with large scale problem. Lastly, task scheduling techniques must be able to find optimal trade-offs between QoS objectives like execution time (makespan), financial cost (cost), among others without violating imposed constraints like deadline and budget to prevent the users from paying unnecessary prices which is a multi-objective optimization problem. The conflicting nature of tasks scheduling objectives makes multi-objective task scheduling optimization problem more challenging especially for large scale task scheduling problems. In the rest of the thesis, execution time and financial cost will be referred as makespan and cost respectively.

Generally, task scheduling problems in Cloud have been proven to be a hard Nondeterministic Polynomial time (NP-hard) optimization problem (Ullman, 1975; Liu *et al.*, 2013), that is there no deterministic algorithm that can find optimum solution to task scheduling problem within an acceptable period of time. Furthermore, users have various QoS objectives, mostly involving makespan and cost. The QoS objectives are mostly conflicting that is no single optimal solution that can satisfy the requirements of all the objectives, therefore trade-off solutions have to be sought. Additionally, various consumers specify constraints for their objectives, like budgets, and deadline that task scheduling algorithms must satisfy. Furthermore, the underlying cloud infrastructure have diverse economic and system characteristics like pay-per-use model, heterogeneity, dynamism, elasticity and performance variations which further complicates task scheduling problem. One of the big challenges in cloud computing service provisioning is to offer the requested services in accordance to constrained QoS objectives of users such as makespan and cost using task scheduling techniques, especially for large scale task scheduling problems. Moreover, several task scheduling techniques have been suggested for distributed systems like cluster and grids (Daoud and Kharma, 2008). These techniques try to minimize makespan as a primary objective in static environment like clusters and grids, whilst this is acceptable for such computing platforms, users stand the risks of being charged prohibitive and unnecessary costs under pay-per-usage model of Cloud infrastructure provision. For instance, the number and type of VMs used, and period of time for VMs usage have effect on the total cost of application execution on Cloud. Normally, users pay less for slower resources as compared to faster resources, hence the scheduler is faced with a multi-objective optimization problem of finding time-cost trade-offs in selecting suitable services considering pay-per-use model.

Due to the practical applications and challenges of executing large scale applications, task scheduling of applications on the large scale have become an emerging research in cloud computing and have attracted significant attention of researchers in recent times. Various heuristics have been applied to solve task scheduling problems which generate optimal solutions for small size problems (Chen *et al.*, 2013; Ming and Li, 2012; Mao *et al.*, 2014; Patel *et al.*, 2015). However, the quality of solutions produced by these techniques degrades woefully as the problem size and number of variables to be optimized increases. Also, these heuristic methods do not have provisions and support for meeting various QoS requirements. In contrast, many cloud users requires certain QoS satisfaction especially for scientific and business domain applications. In recent times, attempts have been made to address task scheduling problems using metaheuristic algorithms to address this problem (Hameed *et al.*, 2014; Wu *et al.*, 2015; Singh and Chana, 2016b). Utilizing metaheuristic algorithms for solving task scheduling problems in Cloud have shown promising improvements in achieving efficiency, by reducing the solution search space. However, metaheuristic algorithms incur high computational time and in some cases return local optimum solution especially when dealing with large solution space, also, these techniques may suffer from premature convergence and imbalance between local and global search (Tsai and Rodrigues, 2014; Guzek *et al.*, 2015; Kalra and Singh, 2015; Zhan *et al.*, 2015; Xue *et al.*, 2016; Meena *et al.*, 2016). These limitations result to sub-optimal task schedule solutions which affects the performance of service provision in terms of meeting the desired QoS objectives.

Task scheduling optimization approaches either focused on single objective or multi-objective. The single objective task scheduling optimization approaches, only try to optimize either makespan or cost with some constraints, especially deadline or budget (Zuo *et al.*, 2014; Rodriguez and Buyya, 2014; Netjinda *et al.*, 2014a; Tawfeek *et al.*, 2015; Li *et al.*, 2015c, 2016; Nirmala and Bhanu, 2016; Zhong *et al.*, 2016; Meena *et al.*, 2016; Liu *et al.*, 2016). The constrained QoS aware algorithms attempt to optimize trade-offs between some QoS objectives without violating user imposed constraints (Lu *et al.*, 2014). However, because of the rapid development of Cloud, several QoS objectives and constraints needs to be considered which makes task scheduling a multi-objective optimization problem. The complexity of the multi-objective task optimization formulation arise from the fact that users and providers have different optimization goals. Users are mainly concerned with minimizing makespan and cost while meeting certain imposed constraints, whereas providers want to maximize resource utilization and energy consumption while meeting user QoS requirements. In this situation, task scheduling have to be solved as a multi-objective optimization problem trying to optimize many and yet conflicting objectives, where it is not possible to obtain optimal solution with regards to all objectives. Therefore, a good trade-offs between the objectives need to obtained.

Multi-objective task scheduling optimization challenge is an important consideration because of its direct effect on both cloud service providers and consumers (Zhan *et al.*, 2015). In cloud computing platform, task scheduling algorithms must optimize financial cost of leasing compute resources in addition to execution time (makespan) and other QoS metrics. Generally, cloud providers offer heterogeneous set of resources (VM instances) at various prices with varied performance. In this way, task scheduling problem needs to be formulated as a multi-objective optimization problem that intend to optimize conflicting objectives such as maksepan and financial cost of task execution. With multi-objective formulation, there is no single solution which is optimal with respect to all objectives, but a set of trade-off solutions called Pareto front (Tao *et al.*, 2014). Multi-objective task scheduling optimization problems are usually solved using aggregation, hierarchical, Pareto, and coevolutionary multi-swarm approaches. The aggregation (weighted) approach is the common method for solving multi-objective task scheduling problems. The approach assign weights to multiple objectives and sum up the objectives to form single objective function. For instance, Delavar and Aryan (2014) proposed GA based task scheduling algorithm to optimize makespan, reliability, and load balancing of applications by putting into consideration the heterogeneous characteristics of compute resources. Also, Shen *et al.* (2016) developed GA algorithm for adaptive scheduling of tasks considering energy consumption and makespan performance. Casas *et al.* (2016) proposed GA based task scheduling technique for optimizing makespan and cost. Zuo *et al.* (2015) proposed ACO based task scheduling algorithm to optimize budget and deadline constrained task scheduling problems, the proposed approach simultaneously makespan and cost within a given budget and deadline. However, the results of different objectives is dependent on the values of the assigned weights which may not adequately represent the decision of the user. Moreover, the approach produce only solution which is not adequate for multi-objective decision problems.

The hierarchical approaches optimize task scheduling objectives in a sequential order, the optimization ordering of the objectives are determined based on their importance and solution to the objectives are alternately sought based on their ordering. For instance, the approach proposed by Teng *et al.* (2007) used sorting strategy, the objective functions are optimized in sequential order. The optimization of an objective is continuously carried until no further improvement is possible, then next objective is optimized while meeting the constraints of the previous optimized objectives. Similar approach was used by Zhang *et al.* (2014) to optimize makespan and cost. However, these approaches are time consuming especially when there are several objectives with constraints, since it requires several iteration of optimization process. Moreover, the importance of the objectives is dependent on the problem, and performance of the approach may be significantly affected by the ranking of the objectives.

To overcome the challenges of fitness assignment problem, new efforts have been reported to use techniques for solving multi-objective task scheduling problems efficiently. These techniques are based on using multiple populations for multiple objectives for solving multi-objective problems where each population optimize one objective (Zhan *et al.*, 2013). Each population is optimized using existing optimization algorithm. Yao *et al.* (2016) proposed endocrine-based co-evolutionary multi-swarm multi-objective algorithm to find optimal trade-offs solutions between energy consumption, makespan, and cost. The proposed strategy adopted multi-swarm optimization strategy where each swarm corresponds to one objective and PSO is used to optimize each objective. A novel competition and cooperation strategy is designed to avoid swarms getting trapped in local optima. Similarly, Li *et al.* (2015a) presents co-evolutionary multi-swarm PSO algorithm to obtain optimal trade-off solutions between makespan and cost. Learning between the particles is enhanced using renumber strategy (Li *et al.*, 2015b). However, the proposed techniques can not scale well since the efficiency of PSO algorithms is challenged by local optima entrapment and imbalance between local and global search. Moreover, efficiently exchanging information between swarms and avoidances of local Pareto Fronts are still challenging issues with co-evolutionary multi-swarm multi-objective task scheduling approaches.

To overcome the drawbacks of both aggregation and hierarchical approaches, Pareto-based optimization approaches have been put forth for addressing multi-objective task scheduling problems (Tao *et al.*, 2014; Durillo *et al.*, 2014). Pareto approaches finds several optimal trade-off solutions for the objectives for the optimization problem. The concept of Pareto dominance is applied to assign fitness to individuals. The Pareto approach does not require transforming multiple objectives into single objective formulation, and generate several trade-off solutions in a single run. Tao *et al.* (2014) presents a hybrid GA algorithms to obtain Pareto optimal solutions for makespan and energy consumption. Pareto optimal trade-offs between makespan, cost, and energy consumption was solved using list scheduling heuristics and hybrid PSO respectively (Fard *et al.*, 2014; Yassa *et al.*, 2013). Similarly, Verma and Kaushal (2017) presents PSO based multi-objective task scheduling algorithm to obtain optimal trade-offs between makespan, cost, and energy consumption while meeting deadline and budget constraints respectively. Xu *et al.* (2014) put forth multi-objective GA for workflow task scheduling problem to simultaneously minimize makespan and cost while considering the priorities of the tasks. Moreover, Zhang *et al.* (2017) proposed multi-objective GA algorithm to obtain Pareto optimal trade-offs between energy consumption, and reliability for deadline constrained task scheduling problems. However, with Pareto task scheduling approaches, it is difficult to select appropriate individual for the next generation since Pareto dominance is a partial order (Zhan *et al.*, 2013). Therefore, the solutions obtained may not cover the entire Pareto Front (PF) if the selection operator fails to keep adequate diversity. Thus, developing multi-objective task scheduling that effectively assign fitness to individuals while keeping solution to efficiently estimate the entire PF remains challenging research.

Many task scheduling optimization problems often introduce constraints which could be loose, moderate, or tight,the imposed constraints makes the solution seeking process more difficult since some regions of search space could be infeasible. By convention, metaheuristic algorithms are characterized by solving unconstrained optimization problems, therefore constrained optimization problems needs to be transformed unconstrained form and appropriate penalty factors are applied in the case of constraint violation. Static penalty function is one of the common constraint method handling strategies, static penalty function is usually applied to penalize infeasible solutions by decreasing their fitness values according to their degree of constraint violation. However, finding a suitable value for penalty function is difficult (Chen *et al.*, 2015b; Liu *et al.*, 2016). For instance, Rodriguez and Buyya (2014) presents PSO algorithm for solving deadline constrained cost optimization problem for workflow scheduling on cloud and used static penalty function to identify the particles violate the constraints are inferior to the feasible ones. However, this may result to premature convergence of search procedure which is a common issue with PSO. Another common approach for constraint handling is eliminating infeasible solutions as the iterative process proceeds. However, some infeasible solutions hold vital information that are essential in guiding search direction, thus they may be useful in next generations of individuals in finding optimal solutions (Kianpisheh *et al.*, 2016; Meena *et al.*, 2016; Ambursa *et al.*, 2016). Furthermore, Huang (2014) presented improved GA for constrained workflow scheduling problem, in their encoding approach task execution queue on VM is indicated in addition to task to VM assignment. Individuals are first evolved using the objective function and evolved population is changed when there is constrain violation. With this method there is no need to define penalty function for constraint violation. However, the approach needs to evolve for many generations which result to high computation time. To avoid the difficulty of defining problem specific factor for penalty functions, Liu *et al.* (2016) put fort a self-adaptive penalty function handle deadline constraint violation in solving cost optimization based task workflow scheduling problem using co-evolutionary GA. The proposed approach is able to accelerate the convergence speed of GA while preventing premature convergence. However, the performance of GA is challenged when traversing large search space. Thus, addressing constrained task scheduling optimization problems is still an active research area.

1.3 Problem Statement

Execution of large scale applications in cloud computing environment is only beneficial, if execution of tasks of the application can be scheduled across compute resources in a manner to achieve a reasonable execution time. To harness the benefits of cloud, task scheduling algorithms play a critical role, task scheduling algorithms assign tasks to compute to meet certain optimization objectives. The common objectives of task scheduling formulations are minimization of financial cost (cost), total execution time (makespan), reliability, security, energy consumption, resource utilization among others. Also, in many task scheduling formulations, user impose certain constraints like budget and deadline. Various QoS optimization techniques have been proposed for distributed systems like clusters and grids. However, these techniques cannot be adapted to cloud environment being an utility based computing platform which is characterized by heterogeneity, dynamism and elasticity. Also, the few works for cloud environment do not either consider the essential characteristics of cloud or the performance of these techniques degrades as the problem size increases. Moreover, cloud based solutions approaches to multi-objective QoS problems are mostly based on weighted sum technique which converts multi-objective formulation to a single objective. However, the assigned weights to each objective may not represent the actual desire of the user and these approaches cannot provide various trade-offs from which the user can choose most suitable option.

In cloud computing environments, task scheduling techniques play a crucial role in meeting various user QoS requirements with diverse QoS objectives and optimization constraints. Users requirements are not only numerous and conflicting, but do include constraints which could be tight, moderate or loose. To solve a constrained optimization problem, the problem needs to be transformed into unconstrained optimization problem. To solve the transformed problem, static penalty function is usually applied to penalize infeasible solutions by decreasing their fitness values according to their degree of constraint violation. However, finding a suitable value for penalty function is difficult. Another common approach is eliminating infeasible solutions as the iterative process proceeds. However, with this approach some infeasible solutions hold vital information that are essential in guiding search direction, thus they cannot be eliminated. However, most of the proposed solutions approaches fail to meet user constraints and provide inadequate QoS optimization results. Therefore, there is need for efficient task scheduling techniques to properly model QoS requirements of applications and handle user constraints effectively. Also, the new techniques should provide solutions that meet QoS requirements without violating the specified constraints.

There are some grid based heuristic task scheduling algorithms that have been adapted for cloud environment, however, these algorithms have made little success in cloud. These heuristic based algorithms produce optimal results for small size problems, however, their performance degrades with large size problems. Further, heuristic techniques do not have provision handling multiple QoS requirements (Ming and Li, 2012; Mao *et al.*, 2014). Based on the constraints imposed by large scale problems, task scheduling problem is identified as NP-hard. Recently, many works have been influenced by nature inspired techniques to provide solutions to increasing complexity and scale of cloud computing system. The NP-hard problems are in most cases being tackled by metaheuristic algorithms like genetic algorithms (GA), Particle Swarm optimization (PSO), and Ant Colony optimization (ACO). These techniques have shown promising performance over heuristic techniques particularly for large scale scheduling problems, the algorithms can find optimal global solution in some cases. However, the computational complexity of these algorithms increases exponentially as the size the problem increases. Moreover, local optimum solutions are return in other cases, and these techniques still suffers from issues like entrapment of search procedure in local optima, premature convergence, and imbalance between global search and local search, resulting to sub-optimal results. Local optima are defined as the relative best solutions within a neighbor solution set which is not necessarily an optimal, therefore, local optima entrapment could result to slow convergence and non-optimal task schedules. Global search is the ability of the algorithm to search for new new solution far from the current solution in the search space. Local search is to search the surrounding search area nearby the current solution, something like local search. Finding an algorithm that could can balance local and global search is challenging. Furthermore, most of the existing works fail to capture the essential features of cloud computing like heterogeneity, elasticity, dynamism, and uncertainty of computing resources there by fail to fulfill user QoS needs. There is need for metaheuristic based optimization algorithms that can efficiently cope with large search space when scheduling large scale applications. Hence, there is scope for further development of task scheduling solutions for further improved solutions. Therefore, this thesis presents Symbiotic Organisms Search (SOS) based task scheduling algorithms for large scale task scheduling optimization on IaaS cloud.

Symbiotic Organisms Search algorithm is a recently introduced metaheuristic algorithm in Cheng and Prayogo (2014) and has gathered considerable interest of researchers from natural computing. SOS was originally proposed to handle continuous benchmark and engineering problems, which was shown to have a robust performance and has faster convergence speed when compared with GA (Deb *et al.*, 2002), PSO (Kennedy, 2011), Differential Evolution (DE) (Qin *et al.*, 2009), Bees Algorithm (BA) (Pham *et al.*, 2011), and Particle Bee Algorithm (PBA) (Cheng and Lien, 2012) which are the traditional metaheuristic algorithms. SOS have proven to be efficient for optimizing complex multidimensional search space while handling multi-objective and constrained optimization problems. Active researches on SOS since its introduction includes hybridization, discrete optimization problems, constrained and multi-objective optimization. Hybridization intends to combine the strengths of SOS like global search ability and rapid optimization, with other related techniques to address some of the issues with SOS performance, like entrapment in local optima.

Some efforts have been made in adapting and modifying SOS algorithms to handle multi-objective and constrained optimization, which are important aspect of facilitating design and optimization of various problems in engineering and computer science. While significant success have been achieved in this areas in recent times, optimization of problems in these areas still remain active research issue. As a result, SOS algorithm have been applied to solve optimization problems in a variety of domains like economic dispatch, power optimization, construction project scheduling, design optimization of engineering structures, transportation, energy optimization, wireless communication, and machine learning. With trend of application of SOS to optimization problems, SOS have shown to provide all-purpose principles that can easily be adapted to solve wide range of optimization problems in various domains. SOS algorithm have been to be very effective and easily adaptable to various application requirements, with potentials for hybridization and modifications. However, SOS still face challenges like local optima entrapment, imbalance between local search and global search, constraint handling, large scale optimization, and multi-objective optimization, and these are still important research focus as evident from the literature. Further understanding and refinements of SOS algorithm and challenges of using it to solve large scale optimization problems are needed.

This research focused on addressing three different issues in SOS algorithm: scalability, slow convergence, and imbalance between local and global search for efficient optimization of large scale task scheduling problems on IaaS Cloud Computing environment.

1.4 Research Hypothesis

The following research hypothesis have been formulated to address the stated problems.

i. The performance of a metaheuristic algorithms for large scale task scheduling degrades with increase in dimensionality of solution search space.

ii. The convergence speed of a metaheuristic algorithm for task scheduling optimization is slow down by local optima entrapment of its search procedure, thus, leading to high computation time and non-optimal solutions.

iii. The optimality of solutions obtained by a metaheuristic algorithm for task scheduling optimization is dependent on its ability to make a proper balance between local and global search.

1.5 Research Questions

The following research questions will be answered to address the stated problems based on the above stated hypothesis.

i. How to enhance the performance of a metaheuristic algorithm to cope huge solution search space for large scale task scheduling optimization?

ii. How to increase the convergence speed of a metaheuristic algorithm by avoiding possible local optima to reduce computation time and enhance optimality of solutions for large scale task scheduling optimization?

iii. How to enhance the global convergence of metaheuristic algorithm by balancing between local and global search to increase the optimality of solutions for large scale task scheduling optimization?

1.6 Research Aim

The aim of this research is to develop an efficient constrained multi-objective QoS task scheduling optimization technique for IaaS Cloud Computing Environment based on the state-of-the-art metaheuristic Symbiotic Organism Search (SOS) algorithm that is scalable and able of avoiding local optima entrapment to ensure faster convergence, with adaptive control parameters to adequately balance local and global search for global convergence.

1.7 Research Objectives

The following objectives are set to achieve the aim of this research:

i. To design and develop a large scale task scheduling technique for Infrastructure as a Service (IaaS) Cloud Computing environment using state-of-the algorithm that can optimize execution time and cost while meeting the users deadline constraints.

ii. To further improve the above Cloud Computing task scheduling technique in order to avoid entrapment in local optima for faster convergence.

iii. To ultimately design and develop a Cloud Computing task scheduling technique that could adaptive control parameters to balance the local and global search procedure for global convergence.

1.8 Scope of the Research

This research is conducted within the following scope:

i. The study of the set out objectives are carried through extensive simulation using CloudSim 3.0.3 simulation framework.

ii. This research considers only task scheduling on IaaS cloud environment.

iii. The precedence constraints between the tasks to be scheduled are out of the scope this research.

iv. Only the VM instances suitable for compute intensive tasks and pricing model offered by Amazon EC2 are considered in this research.

1.9 Research Contributions

The main contributions of this research are as follows:

i. Cooperative Coevolutionary Constrained Multi-objective Symbiotic Organisms Search (CC-CMSOS) for large scale task scheduling optimization to find optimal trade-offs between makespan and cost while meeting deadline constraint thereby reducing the computational time and improving global convergence.

ii. Cooperative Coevolutionary Constrained Multi-objective Memetic Symbiotic Organisms Search (CC-CMMSOS) for large scale task scheduling optimization to find optimal trade-offs between makespan and cost while meeting deadline constraint thereby ensuring faster convergence.

iii. Cooperative Coevolutionary Constrained Multi-objective Adaptive Benefit Factor Symbiotic Organisms Search (CC-CMABFSOS) for large scale task scheduling optimization to find optimal trade-offs between makespan and cost while meeting deadline constraint thereby ensuring global convergence.

1.10 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 examines the related literatures and analyzes the current issues encountered in task scheduling optimization on cloud computing environment which forms the bases for realization of the research's methods. The review of different task scheduling techniques were carried out. The focus of the review was primarily on metaheuristic algorithms for their efficiency to support large scale task scheduling optimization on IaaS Cloud. In addition, developments and applications of SOS algorithms are discussed, as a potential solution to large scale task scheduling optimization.

Chapter 3 describes the methodology of this study, it covers the general framework that list the approaches of achieving the research objectives in a systematic manner. The steps required for design and development of the proposed algorithms are outlined. Furthermore, the experimental test bed which comprises of the simulation tools, and workloads for evaluating the efficacy of the proposed algorithms are established as well as metrics of evaluation and comparison with other algorithms for benchmark.

Chapter 4 presents the design and development of constrained multi-objective SOS algorithms for large scale task scheduling optimization problems, with detailed discussion on design and development of the algorithms. The designed algorithms respectively address the issues of scalability, slow convergence, and imbalance between local and global search of standard SOS algorithms for efficient optimization of QoS objective for large scale task scheduling problems.

Chapter 5 presents the results analysis and discussion of multi-objective Symbiotic Organisms Search (SOS) algorithms for large scale task scheduling optimization problems on IaaS cloud environment.

Chapter 6 concludes the thesis with a summary of contributions and possible future directions of this research.

REFERENCES

Abdollahzade, M., Miranian, A., Hassani, H. and Iranmanesh, H. (2015). A new hybrid enhanced local linear neuro-fuzzy model based on the optimized singular spectrum analysis and its application for nonlinear and chaotic time series forecasting. *Information Sciences*. 295, 107–125.

Abrishami, S., Naghibzadeh, M. and Epema, D. H. (2012). Cost-driven scheduling of grid workflows using partial critical paths. *IEEE Transactions on Parallel and Distributed Systems*. 23(8), 1400–1414.

Abrishami, S., Naghibzadeh, M. and Epema, D. H. (2013). Deadline-constrained workflow scheduling algorithms for Infrastructure as a Service Clouds. *Future Generation Computer Systems*. 29(1), 158–169.

Adarsh, B., Raghunathan, T., Jayabarathi, T. and Yang, X.-S. (2016). Economic dispatch using chaotic bat algorithm. *Energy*. 96, 666–675.

Ai, L., Tang, M. and Fidge, C. J. (2010). QoS-oriented sesource allocation and scheduling of multiple composite web services in a hybrid cloud using a random-key genetic algorithm. In *17th International Conference on Neural Information Processing*. 22-25 November. Sydney, N.S.W.: Springer.

Alatas, B. (2010). Chaotic bee colony algorithms for global numerical optimization. *Expert Systems with Applications*. 37(8), 5682–5687.

Ali, M., Siarry, P. and Pant, M. (2012). An efficient differential evolution based algorithm for solving multi-objective optimization problems. *European journal of operational research*. 217(2), 404–416.

Alkhanak, E. N., Lee, S. P. and Khan, S. U. R. (2015). Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities. *Future Generation Computer Systems*. 50, 3–21.

Alla, H. B., Alla, S. B., Ezzati, A. and Mouhsen, A. (2017). A novel architecture with dynamic queues based on fuzzy logic and particle swarm optimization algorithm for task scheduling in cloud computing. In *Advances in Ubiquitous Networking 2*. (pp. 205–217). Springer.

Ambursa, F. U., Latip, R., Abdullah, A. and Subramaniam, S. (2016). A particle swarm optimization and min–max-based workflow scheduling algorithm with QoS satisfaction for service-oriented grids. *The Journal of Supercomputing*, 1–34. doi: 10.1007/s11227-016-1901-x.

Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. *et al.* (2010). A view of cloud computing. *Communications of the ACM*. 53(4), 50–58.

Balachennaiah, P. and Suryakalavathi, M. (2015). Real Power Loss minimization using symbiotic organisms search algorithm. In *2015 Annual IEEE India Conference (INDICON)*. 17-20 December. New Delhi, India: IEEE, 1–6.

Banerjee, S. and Chattopadhyay, S. (2017). Power optimization of three dimensional turbo code using a novel modified symbiotic organism search (MSOS) algorithm. *Wireless Personal Communications*. 92(3), 941–968.

Beegom, A. A. and Rajasree, M. (2014). A particle swarm optimization based pareto optimal task scheduling in cloud computing. In *International Conference in Swarm Intelligence*. 17-20 October. Hefei, China: Springer, 79–86.

Blum, C., Puchinger, J., Raidl, G. R. and Roli, A. (2011). Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*. 11(6), 4135–4151.

BoussaïD, I., Lepagnot, J. and Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*. 237, 82–117.

Buyya, R. and Murshed, M. (2002). Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. *Concurrency and computation: practice and experience*. 14(13-15), 1175–1220.

Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J. and Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation computer systems*. 25(6), 599–616.

Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. F. and Buyya, R. (2011). CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*. 41(1), 23–50.

Casanova, H. (2001). Simgrid: A toolkit for the simulation of application scheduling. In *Cluster computing and the grid, 2001. proceedings. first ieee/acm international symposium on*. 15-18 May. Brisbane, Australia: IEEE, 430–437.

Casas, I., Taheri, J., Ranjan, R., Wang, L. and Zomaya, A. Y. (2016). GA-ETI: An enhanced genetic algorithm for the scheduling of scientific workflows in cloud environments. *Journal of Computational Science*. doi:http://doi.org/10.1016/j.jocs.2016.08.007.

Chen, H., Wang, F., Helian, N. and Akanmu, G. (2013). User-priority guided Min-Min scheduling algorithm for load balancing in cloud computing. In *Parallel Computing Technologies (PARCOMPTECH), 2013 National Conference on*. 21-23 February. Bangalore,India: IEEE, 1–8.

Chen, H., Zhu, X., Guo, H., Zhu, J., Qin, X. and Wu, J. (2015a). Towards energy-efficient scheduling for real-time tasks under uncertain cloud computing environment. *Journal of Systems and Software*. 99, 20–35.

Chen, W.-N. and Zhang, J. (2012). A set-based discrete PSO for cloud workflow scheduling with user-defined QoS constraints. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 14-17 October. COEX, Seoul, Korea: IEEE, 773–778.

Chen, Y.-H. and Huang, H.-C. (2015). Coevolutionary genetic watermarking for owner identification. *Neural Computing and Applications*. 26(2), 291–298.

Chen, Z.-G., Du, K.-J., Zhan, Z.-H. and Zhang, J. (2015b). Deadline constrained cloud computing resources scheduling for cost optimization based on dynamic objective genetic algorithm. In *2015 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, 708–714.

Cheng, M.-Y. and Lien, L.-C. (2012). Hybrid Artificial Intelligence–Based PBA for Benchmark Functions and Facility Layout Design Optimization. *Journal of Computing in Civil Engineering*. 26(5), 612–624.

Cheng, M.-Y. and Prayogo, D. (2014). Symbiotic organisms search: a new metaheuristic optimization algorithm. *Computers & Structures*. 139, 98–112.

Cheng, M.-Y., Prayogo, D. and Tran, D.-H. (2015). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search. *Journal of Computing in Civil Engineering*. 30(3), 04015036.

Coello, C. C. (2006). Evolutionary multi-objective optimization: a historical view of the field. *IEEE computational intelligence magazine*. 1(1), 28–36.

Daoud, M. I. and Kharma, N. (2008). A high performance algorithm for static task scheduling in heterogeneous distributed computing systems. *Journal of Parallel and distributed computing*. 68(4), 399–409.

Das, S. and Bhattacharya, A. (2016). Symbiotic organisms search algorithm for short-term hydrothermal scheduling. *Ain Shams Engineering Journal*.

de Oliveira, F. B., Enayatifar, R., Sadaei, H. J., Guimarães, F. G. and Potvin, J.-Y. (2016). A cooperative coevolutionary algorithm for the Multi-Depot Vehicle Routing Problem. *Expert Systems with Applications*. 43, 117–130.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA II. *IEEE transactions on evolutionary computation*. 6(2), 182–197.

Deelman, E., Gannon, D., Shields, M. and Taylor, I. (2009). Workflows and e-Science: An overview of workflow system features and capabilities. *Future Generation Computer Systems*. 25(5), 528–540.

Deelman, E., Vahi, K., Juve, G., Rynge, M., Callaghan, S., Maechling, P. J., Mayani, R., Chen, W., da Silva, R. F., Livny, M. *et al.* (2015). Pegasus, a workflow management system for science automation. *Future Generation Computer Systems*. 46, 17–35.

Dekkers, A. and Aarts, E. (1991). Global optimization and simulated annealing. *Mathematical programming*. 50(1), 367–393.

Delavar, A. G. and Aryan, Y. (2011). A Synthetic Heuristic Algorithm for Independent Task Scheduling in Cloud System. *IJCSI*.

Delavar, A. G. and Aryan, Y. (2014). HSGA: a hybrid heuristic algorithm for workflow scheduling in cloud systems. *Cluster computing*. 17(1), 129–137.

Dib, N. (2016a). Synthesis of antenna arrays using symbiotic organisms search (SOS) algorithm. In *Antennas and Propagation (APSURSI), 2016 IEEE International Symposium on*. 26 June-1 July. El Conquistador Resort, Fajardo, Puerto Rico: IEEE, 581–582.

Dib, N. I. (2016b). Design of Linear Antenna Arrays with Low Side Lobes Level Using Symbiotic Organisms Search. *Progress In Electromagnetics Research B*. 68, 55–71.

Dikaiakos, M. D., Katsaros, D., Mehra, P., Pallis, G. and Vakali, A. (2009). Cloud computing: Distributed internet computing for IT and scientific research. *IEEE Internet computing*. 13(5), 10–13.

Doering, J., Juan, A. A., Kizys, R., Fito, A. and Calvet, L. (2016). Solving Realistic Portfolio Optimization Problems via Metaheuristics: A Survey and an Example. In *Modeling and Simulation in Engineering, Economics and Management*. (pp. 22–30). Springer.

Doerner, K., Hartl, R. F. and Reimann, M. (2001). Cooperative ant colonies for optimizing resource allocation in transportation. In *Workshops on Applications of Evolutionary Computation*. Springer, 70–79.

Dosoglu, M. K., Guvenc, U., Duman, S., Sonmez, Y. and Kahraman, H. T. (2016).

Symbiotic organisms search optimization algorithm for economic/emission dispatch problem in power systems. *Neural Computing and Applications*, 1–17.

Duman, S. (2016). Symbiotic organisms search algorithm for optimal power flow problem based on valve-point effect and prohibited zones. *Neural Computing and Applications*, 1–15.

Durillo, J. J., Nae, V. and Prodan, R. (2014). Multi-objective energy-efficient workflow scheduling using list-based heuristics. *Future Generation Computer Systems*. 36, 221–236.

Dutta, D. and Joshi, R. (2011). A genetic: algorithm approach to cost-based multi-QoS job scheduling in cloud computing environment. In *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*. 25-26 February. Mumbai, Maharashtra, India: ACM, 422–427.

Ebner, M. (2006). Coevolution and the red queen effect shape virtual plants. *Genetic Programming and Evolvable Machines*. 7(1), 103–123.

Eki, R., Vincent, F. Y., Budi, S. and Redi, A. P. (2015). Symbiotic Organism Search (SOS) for Solving the Capacitated Vehicle Routing Problem. *World Academy of Science, Engineering and Technology, International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*. 9(5), 850–854.

Engelbrecht, A. P. (2007). *Computational intelligence: an introduction*. John Wiley & Sons.

Fard, H. M., Prodan, R. and Fahringer, T. (2014). Multi-objective list scheduling of workflow applications in distributed computing infrastructures. *Journal of Parallel and Distributed Computing*. 74(3), 2152–2165.

Feitelson, D. G., Tsafrir, D. and Krakov, D. (2014). Experience with using the parallel workloads archive. *Journal of Parallel and Distributed Computing*. 74(10), 2967–2982.

Foster, I., Zhao, Y., Raicu, I. and Lu, S. (2008). Cloud computing and grid computing 360-degree compared. In *2008 Grid Computing Environments Workshop*. 12-16 November. Austin, Texas: Ieee, 1–10.

Gabi, D., Ismail, A. S., Zainal, A., Zakaria, Z. and Abraham, A. (2016a). Orthogonal Taguchi-based cat algorithm for solving task scheduling problem in cloud computing. *Neural Computing and Applications*, 1–19.

Gabi, D., Ismail, A. S., Zainal, A., Zakaria, Z. and Abraham, A. (2016b). Orthogonal Taguchi-based cat algorithm for solving task scheduling problem in cloud computing. *Neural Computing and Applications*, 1–19.

Gandomi, A. H. and Yang, X.-S. (2014). Chaotic bat algorithm. *Journal of Computational Science*. 5(2), 224–232.

Garg, S. K. and Buyya, R. (2011). Networkcloudsim: Modelling parallel applications in cloud simulations. In *Utility and Cloud Computing (UCC), 2011 Fourth IEEE International Conference on*. 5-8 December. Melbourne, Victoria, Australia: IEEE, 105–113.

Gogos, C., Valouxis, C., Alefragis, P., Goulas, G., Voros, N. and Housos, E. (2016). Scheduling independent tasks on heterogeneous processors using heuristics and Column Pricing. *Future Generation Computer Systems*. 60, 48–66.

Goh, C.-K. and Tan, K. C. (2009). A competitive-cooperative coevolutionary paradigm for dynamic multiobjective optimization. *IEEE Transactions on Evolutionary Computation*. 13(1), 103–127.

Gu, J., Hu, J., Zhao, T. and Sun, G. (2012). A new resource scheduling strategy based on genetic algorithm in cloud computing environment. *Journal of Computers*. 7(1), 42–52.

Guha, D., Roy, P. and Banerjee, S. (2016). Quasi-oppositional symbiotic organism search algorithm applied to load frequency control. *Swarm and Evolutionary Computation*. 33, 46–67.

Guo, L., Zhao, S., Shen, S. and Jiang, C. (2012a). A particle swarm optimization for data placement strategy in cloud computing. In *Information Engineering and Applications*. (pp. 946–953). vol. 154. Springer.

Guo, L., Zhao, S., Shen, S. and Jiang, C. (2012b). Task scheduling optimization in cloud computing based on heuristic algorithm. *Journal of Networks*. 7(3), 547–553.

Guvenc, U., Duman, S., Dosoglu, M. K., Kahraman, H. T., Sonmez, Y. and Yılmaz, C. (2016). Application of Symbiotic Organisms Search Algorithm to solve various economic load dispatch problems. In *INnovations in Intelligent SysTems and Applications (INISTA), 2016 International Symposium on*. 2-5 August. Sinaia, Romania: IEEE, 1–7.

Guzek, M., Bouvry, P. and Talbi, E.-G. (2015). A Survey of Evolutionary Computation for Resource Management of Processing in Cloud Computing [Review Article]. *IEEE Computational Intelligence Magazine*. 10(2), 53–67.

Guzek, M., Varrette, S., Plugaru, V., Pecero, J. E. and Bouvry, P. (2014). A holistic model of the performance and the energy efficiency of hypervisors in a high-performance computing environment. *Concurrency and Computation: Practice and Experience*.

26(15), 2569–2590.

Hameed, A., Khoshkbarforoushha, A., Ranjan, R., Jayaraman, P. P., Kolodziej, J., Balaji, P., Zeadally, S., Malluhi, Q. M., Tziritas, N., Vishnu, A. *et al.* (2014). A survey and taxonomy on energy efficient resource allocation techniques for cloud computing systems. *Computing*, 1–24.

Huang, J. (2014). The workflow task scheduling algorithm based on the GA model in the cloud computing environment. *Journal of Software*. 9(4), 873–880.

Ishibuchi, H., Hitotsuyanagi, Y., Tsukamoto, N. and Nojima, Y. (2010). Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space. In *International Conference on Parallel Problem Solving from Nature*. 11-15 September. Krakw, Poland: Springer, 91–100.

Jiang, Y., Shao, Z., Guo, Y., Zhang, H. and Niu, K. (2015). DRSCRO: A Metaheuristic Algorithm for Task Scheduling on Heterogeneous Systems. *Mathematical Problems in Engineering*. 2015.

Jiao, L., Luo, J., Shang, R. and Liu, F. (2014). A modified objective function method with feasible-guiding strategy to solve constrained multi-objective optimization problems. *Applied Soft Computing*. 14, 363–380.

Juve, G., Chervenak, A., Deelman, E., Bharathi, S., Mehta, G. and Vahi, K. (2013). Characterizing and profiling scientific workflows. *Future Generation Computer Systems*. 29(3), 682–692.

Kahraman, H. T., Dosoglu, M. K., Guvenc, U., Duman, S. and Sonmez, Y. (2016). Optimal scheduling of short-term hydrothermal generation using symbiotic organisms search algorithm. In *Smart Grid Congress and Fair (ICSG), 2016 4th International Istanbul*. 20-21 April. Istanbul, Turkey: IEEE, 1–5.

Kalra, M. and Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. *Egyptian Informatics Journal*. 16(3), 275–295.

Kanimozhi, G., Rajathy, R. and Kumar, H. (2016). Minimizing Energy of Point Charges on a Sphere using Symbiotic Organisms Search Algorithm. *International Journal on Electrical Engineering and Informatics*. 8(1), 29.

Kant, A., Sharma, A., Agarwal, S. and Chandra, S. (2010). An ACO approach to job scheduling in grid environment. In *International Conference on Swarm, Evolutionary, and Memetic Computing*. 16-18 December. Chennai, India: Springer, 286–295.

Kasahara, Y. and Yonezawa, Y. (1996). The properties of complex evolution in chaos generation process. In *Evolutionary Computation, 1996., Proceedings of IEEE*

International Conference on. 20-22 May. Nagoya University, JAPAN: IEEE, 874–879.

Kaur, P. and Mehta, S. (2017). Resource provisioning and work flow scheduling in clouds using augmented Shuffled Frog Leaping Algorithm. *Journal of Parallel and Distributed Computing*. 101, 41–50.

Kaur, T. and Chana, I. (2015). Energy efficiency techniques in cloud computing: A survey and taxonomy. *ACM Computing Surveys (CSUR)*. 48(2), 22.

Kennedy, J. (2011). Particle swarm optimization. In *Encyclopedia of machine learning*. (pp. 760–766). Springer.

Kessaci, Y., Melab, N. and Talbi, E.-G. (2013). A Pareto-based metaheuristic for scheduling HPC applications on a geographically distributed cloud federation. *Cluster Computing*. 16(3), 451–468.

Kianpisheh, S., Charkari, N. M. and Kargahi, M. (2016). Reliability-driven scheduling of time/cost-constrained grid workflows. *Future Generation Computer Systems*. 55, 1–16.

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. *et al.* (1983). Optimization by simmulated annealing. *science*. 220(4598), 671–680.

Kliazovich, D., Bouvry, P., Audzevich, Y. and Khan, S. U. (2010). GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. 6-10 December. Miami, Florida, USA: IEEE, 1–5.

Kousalya, K. and Balasubramanie, P. (2009). To Improve Ant Algorithm's Grid Scheduling Using Local Search. *International Journal of Intelligent Information Technology Application*. 2(2).

Kumar, P. and Verma, A. (2012). Independent task scheduling in cloud computing by improved genetic algorithm. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2(5).

Latiff, M. S. A., Madni, S. H. H., Abdullahi, M. *et al.* (2016). Fault tolerance aware scheduling technique for cloud computing environment using dynamic clustering algorithm. *Neural Computing and Applications*, 1–15.

LaTorre, A., Muelas, S. and Peña, J.-M. (2015). A comprehensive comparison of large scale global optimizers. *Information Sciences*. 316, 517–549.

Le Hoang, S. (2014). Optimizing municipal solid waste collection using chaotic particle swarm optimization in GIS based environments: a case study at Danang City, Vietnam. *Expert systems with applications*. 41(18), 8062–8074.

Li, H.-H., Chen, Z.-G., Zhan, Z.-H., Du, K.-J. and Zhang, J. (2015a). Renumber coevolutionary multiswarm particle swarm optimization for multi-objective workflow scheduling on cloud computing environment. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*. 25-28 May. Sendai, Japan: ACM, 1419–1420.

Li, H.-H., Fu, Y.-W., Zhan, Z.-H. and Li, J.-J. (2015b). Renumber strategy enhanced particle swarm optimization for cloud computing resource scheduling. In *2015 IEEE Congress on Evolutionary Computation (CEC)*. 11 - 15 July. Madrid, Spain: IEEE, 870–876.

Li, K., Xu, G., Zhao, G., Dong, Y. and Wang, D. (2011). Cloud task scheduling based on load balancing ant colony optimization. In *2011 Sixth Annual ChinaGrid Conference*. 22-23 August. Dalian, Liaoning, China: IEEE, 3–9.

Li, Q., Wang, Z.-y., Li, W.-h., Li, J., Wang, C. and Du, R.-y. (2013). Applications integration in a hybrid cloud computing environment: Modelling and platform. *Enterprise Information Systems*. 7(3), 237–271.

Li, X., Xu, J. and Yang, Y. (2015c). A chaotic particle swarm optimization-based heuristic for market-oriented task-level scheduling in cloud workflow systems. *Computational intelligence and neuroscience*. 2015, 81.

Li, X. and Yao, X. (2012). Cooperatively coevolving particle swarms for large scale optimization. *IEEE Transactions on Evolutionary Computation*. 16(2), 210–224.

Li, Z., Ge, J., Yang, H., Huang, L., Hu, H., Hu, H. and Luo, B. (2016). A security and cost aware scheduling algorithm for heterogeneous tasks of scientific workflow in clouds. *Future Generation Computer Systems*. 65, 140–152.

Liu, H., Abraham, A. and Hassanien, A. E. (2010a). Scheduling jobs on computational grids using a fuzzy particle swarm optimization algorithm. *Future Generation Computer Systems*. 26(8), 1336–1343.

Liu, H., Jin, H., Xu, C.-Z. and Liao, X. (2013). Performance and energy modeling for live migration of virtual machines. *Cluster computing*. 16(2), 249–264.

Liu, H., Xu, D. and Miao, H. K. (2011). Ant colony optimization based service flow scheduling with various QoS requirements in cloud computing. In *Software and Network Engineering (SSNE), 2011 First ACIS International Symposium on*. 19-20 December. Wuhan, China: IEEE, 53–58.

Liu, K., Jin, H., Chen, J., Liu, X., Yuan, D. and Yang, Y. (2010b). A compromised-time-cost scheduling algorithm in SwinDeW-C for instance-intensive cost-constrained workflows on cloud computing platform. *International Journal of High Performance Computing Applications*. 24(4).

Liu, L., Zhang, M., Buyya, R. and Fan, Q. (2016). Deadline-constrained coevolutionary genetic algorithm for scientific workflow scheduling in cloud computing. *Concurrency and Computation: Practice and Experience*. 29(5).

Lu, G., Tan, W., Sun, Y., Zhang, Z., Tang, A. *et al.* (2014). QoS constraint based workflow scheduling for cloud computing services. *Journal of Software*. 9(4), 926–930.

Lu, X. and Gu, Z. (2011). A load-adapative cloud resource scheduling model based on ant colony algorithm. In *2011 IEEE International Conference on Cloud Computing and Intelligence Systems*. 15-17 September. Beijing, China: IEEE, 296–300.

Lyden, S. and Haque, M. E. (2016). A simulated annealing global maximum power point tracking approach for PV modules under partial shading conditions. *IEEE Transactions on Power Electronics*. 31(6), 4171–4181.

Mahdavi, S., Shiri, M. E. and Rahnamayan, S. (2015). Metaheuristics in large-scale global continues optimization: a survey. *Information Sciences*. 295, 407–428.

Malawski, M., Juve, G., Deelman, E. and Nabrzyski, J. (2015). Algorithms for cost-and deadline-constrained provisioning for scientific workflow ensembles in iaas clouds. *Future Generation Computer Systems*. 48, 1–18.

Mao, Y., Chen, X. and Li, X. (2014). Max–min task scheduling algorithm for load balance in cloud computing. In *Proceedings of International Conference on Computer Science and Information Technology*. 2123 September. Kunming, China: Springer, 457–465.

Masdari, M., ValiKardan, S., Shahi, Z. and Azar, S. I. (2016). Towards workflow scheduling in cloud computing: a comprehensive analysis. *Journal of Network and Computer Applications*. 66, 64–82.

Mastelic, T., Oleksiak, A., Claussen, H., Brandic, I., Pierson, J.-M. and Vasilakos, A. V. (2015). Cloud computing: Survey on energy efficiency. *ACM Computing Surveys (CSUR)*. 47(2), 33.

Mathiyalagan, P., Suriya, S. and Sivanandam, S. (2010). Modified ant colony algorithm for grid scheduling. *IJCSE) International Journal on Computer Science and Engineering*. 2(02), 132–139.

Meena, J., Kumar, M. and Vardhan, M. (2016). Cost Effective Genetic Algorithm for Workflow Scheduling in Cloud Under Deadline Constraint. *IEEE Access*. 4, 5065–5082.

Mezmaz, M., Melab, N., Kessaci, Y., Lee, Y. C., Talbi, E.-G., Zomaya, A. Y. and Tuyttens, D. (2011). A parallel bi-objective hybrid metaheuristic for energy-aware scheduling for cloud computing systems. *Journal of Parallel and Distributed*

Computing. 71(11), 1497–1508.

Ming, G. and Li, H. (2012). An improved algorithm based on max-min for cloud task scheduling. In *Recent Advances in Computer Science and Information Engineering*. (pp. 217–223). Springer.

Mocanu, E. M., Florea, M., Andreica, M. I. and Ţăpuş, N. (2012). Cloud computing-task scheduling based on genetic algorithms. In *Systems Conference (SysCon), 2012 IEEE International*. 19-22 March. British Columbia, Canada: IEEE, 1–6.

Morshedlou, H. and Meybodi, M. R. (2014). Decreasing impact of SLA violations: A proactive resource allocation approachfor cloud computing environments. *IEEE Transactions on Cloud Computing*. 2(2), 156–167.

Moschakis, I. A. and Karatza, H. D. (2015a). A meta-heuristic optimization approach to the scheduling of Bag-of-Tasks applications on heterogeneous Clouds with multi-level arrivals and critical jobs. *Simulation Modelling Practice and Theory*. 57, 1–25.

Moschakis, I. A. and Karatza, H. D. (2015b). Multi-criteria scheduling of Bag-of-Tasks applications on heterogeneous interlinked clouds with simulated annealing. *Journal of Systems and Software*. 101, 1–14.

Nama, S., Saha, A. and Ghosh, S. (2016a). Improved symbiotic organisms search algorithm for solving unconstrained function optimization. *Decision Science Letters*. 5(3), 361–380.

Nama, S., Saha, A. K. and Ghosh, S. (2016b). A Hybrid Symbiosis Organisms Search algorithm and its application to real world problems. *Memetic Computing*, 1–20.

Netjinda, N., Sirinaovakul, B. and Achalakul, T. (2014a). Cost optimal scheduling in IaaS for dependent workload with particle swarm optimization. *The Journal of Supercomputing*. 68(3), 1579–1603.

Netjinda, N., Sirinaovakul, B. and Achalakul, T. (2014b). Cost optimal scheduling in IaaS for dependent workload with particle swarm optimization. *The Journal of Supercomputing*. 68(3), 1579–1603.

Nirmala, S. J. and Bhanu, S. M. S. (2016). Catfish-PSO based scheduling of scientific workflows in IaaS cloud. *Computing*. 98(11), 1091–1109.

Núñez, A., Merayo, M. G., Hierons, R. M. and Núñez, M. (2013). Using genetic algorithms to generate test sequences for complex timed systems. *Soft Computing*. 17(2), 301–315.

Nuñez, A., Vázquez-Poletti, J. L., Caminero, A. C., Carretero, J. and Llorente, I. M. (2011). Design of a new cloud computing simulation platform. In *International Conference on Computational Science and Its Applications*. 20-23 June. Santander, Spain: Springer, 582–593.

Osman, I. H. and Kelly, J. P. (2012). *Meta-heuristics: theory and applications*. Springer Science & Business Media.

Ostermann, S., Iosup, A., Yigitbasi, N., Prodan, R., Fahringer, T. and Epema, D. (2009). A performance analysis of EC2 cloud computing services for scientific computing. In *International Conference on Cloud Computing*. 19-21 October. Munich, Germany: Springer, 115–131.

Ostermann, S., Plankensteiner, K., Prodan, R. and Fahringer, T. (2010). GroudSim: an event-based simulation framework for computational grids and clouds. In *European Conference on Parallel Processing*. 31 August - 3 September. Ischia, Italy: Springer, 305–313.

Oxley, M. A., Pasricha, S., Maciejewski, A. A., Siegel, H. J., Apodaca, J., Young, D., Briceno, L., Smith, J., Bahirat, S., Khemka, B. *et al.* (2015). Makespan and energy robust stochastic static resource allocation of a bag-of-tasks to a heterogeneous computing system. *IEEE Transactions on Parallel and Distributed Systems*. 26(10), 2791–2805.

Panda, A. and Pani, S. (2016). A Symbiotic Organisms Search algorithm with adaptive penalty function to solve multi-objective constrained optimization problems. *Applied Soft Computing*. 46, 344–360.

Pandey, S., Wu, L., Guru, S. M. and Buyya, R. (2010). A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments. In *2010 24th IEEE international conference on advanced information networking and applications*. 20-23 April. Perth, Australia: IEEE, 400–407.

Parsopoulos, K. E. (2012). Parallel cooperative micro-particle swarm optimization: A master–slave model. *Applied Soft Computing*. 12(11), 3552–3579.

Patel, G., Mehta, R. and Bhoi, U. (2015). Enhanced Load Balanced Min-min Algorithm for Static Meta Task Scheduling in Cloud Computing. *Procedia Computer Science*. 57, 545–553.

Pham, D., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S. and Zaidi, M. (2011). The bees algorithm–a novel tool for complex optimisation. In *Intelligent Production Machines and Systems-2nd I* PROMS Virtual International Conference (3-14 July 2006)*.

Potter, M. A. and De Jong, K. A. (2000). Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary computation*. 8(1), 1–29.

Prasad, D. and Mukherjee, V. (2016). A novel symbiotic organisms search algorithm for optimal power flow of power system with FACTS devices. *Engineering Science and Technology, an International Journal*. 19(1), 79–89.

Puchinger, J. and Raidl, G. R. (2005). Combining metaheuristics and exact algorithms in combinatorial optimization: A survey and classification. In *International Work-Conference on the Interplay Between Natural and Artificial Computation*. Springer, 41–53.

Qin, A. K., Huang, V. L. and Suganthan, P. N. (2009). Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on Evolutionary Computation*. 13(2), 398–417.

Rajagopalan, A., Sengoden, V. and Govindasamy, R. (2015). Solving economic load dispatch problems using chaotic self-adaptive differential harmony search algorithm. *International Transactions on Electrical Energy Systems*. 25(5), 845–858.

Rajathy, R., Taraswinee, B. and Suganya, S. (2015). A novel method of using symbiotic organism search algorithm in solving security-constrained economic dispatch. In *Circuit, Power and Computing Technologies (ICCPCT), 2015 International Conference on*. 19-20 March. Nagercoil, India: IEEE, 1–8.

Ramezani, F., Lu, J., Taheri, J. and Hussain, F. K. (2015). Evolutionary algorithm-based multi-objective task scheduling optimization model in cloud environments. *World Wide Web*. 18(6), 1737–1757.

Ranaldo, N. and Zimeo, E. (2009). Time and cost-driven scheduling of data parallel tasks in grid workflows. *IEEE Systems Journal*. 3(1), 104–120.

Ren, Y. and Wu, Y. (2013). An efficient algorithm for high-dimensional function optimization. *Soft Computing*. 17(6), 995–1004.

Rimal, B. P., Choi, E. and Lumb, I. (2009). A taxonomy and survey of cloud computing systems. *INC, IMS and IDC*, 44–51.

Rimal, B. P., Jukan, A., Katsaros, D. and Goeleven, Y. (2011). Architectural requirements for cloud computing systems: an enterprise cloud approach. *Journal of Grid Computing*. 9(1), 3–26.

Rius, J., Cores, F. and Solsona, F. (2013). Cooperative scheduling mechanism for large-scale peer-to-peer computing systems. *Journal of Network and Computer Applications*. 36(6), 1620–1631.

Rodriguez, M. A. and Buyya, R. (2014). Deadline based resource provisioningand scheduling algorithm for scientific workflows on clouds. *IEEE Transactions on*

Cloud Computing. 2(2), 222–235.

Sabar, N., Abawajy, J. and Yearwood, J. (2016). Heterogeneous cooperative co-evolution memetic differential evolution algorithms for big data optimisation problems. *IEEE Transactions on Evolutionary Computation*. 21(2), 315–327.

Saha, D., Datta, A. and Das, P. (2016). Optimal coordination of directional overcurrent relays in power systems using Symbiotic Organism Search (SOS) optimization technique. *IET Generation, Transmission & Distribution*.

Sawant, S. (2011). A genetic algorithm scheduling approach for virtual machine resources in a cloud computing environment.

Secui, D. C. (2016). A modified Symbiotic Organisms Search algorithm for large scale economic dispatch problem with valve-point effects. *Energy*. 113, 366–384.

Sharma, Y., Javadi, B., Si, W. and Sun, D. (2016). Reliability and Energy Efficiency in Cloud Computing Systems: Survey and Taxonomy. *Journal of Network and Computer Applications*.

Shayeghi, H. and Ghasemi, A. (2014). A modified artificial bee colony based on chaos theory for solving non-convex emission/economic dispatch. *Energy Conversion and Management*. 79, 344–354.

Shen, G. and Zhang, Y.-Q. (2011). A shadow price guided genetic algorithm for energy aware task scheduling on cloud computers. In *International Conference in Swarm Intelligence*. 12-15 June. Chongqing, China: Springer, 522–529.

Shen, Y., Bao, Z., Qin, X. and Shen, J. (2016). Adaptive task scheduling strategy in cloud: when energy consumption meets performance guarantee. *World Wide Web*, 1–19.

Shojafar, M., Javanmardi, S., Abolfazli, S. and Cordeschi, N. (2015). FUGE: A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method. *Cluster Computing*. 18(2), 829–844.

Singh, S. and Chana, I. (2016a). QoS-aware autonomic resource management in cloud computing: a systematic review. *ACM Computing Surveys (CSUR)*. 48(3), 42.

Singh, S. and Chana, I. (2016b). A survey on resource scheduling in cloud computing: Issues and challenges. *Journal of Grid Computing*. 14(2), 217–264.

Somasundaram, T. S. and Govindarajan, K. (2014). CLOUDRB: A framework for scheduling and managing High-Performance Computing (HPC) applications in science cloud. *Future Generation Computer Systems*. 34, 47–65.

Sonmez, Y., Kahraman, H. T., Dosoglu, M. K., Guvenc, U. and Duman, S. (2016).

Symbiotic organisms search algorithm for dynamic economic dispatch with valve-point effects. *Journal of Experimental & Theoretical Artificial Intelligence*, 1–21.

Sridhar, M. and Babu, G. R. M. (2015). Hybrid particle swarm optimization scheduling for cloud computing. In *Advance Computing Conference (IACC), 2015 IEEE International*. IEEE, 1196–1200.

Srinivasan, A., Quadir, M. A. and Vijayakumar, V. (2015). Era of cloud computing: a new insight to hybrid cloud. *Procedia Computer Science*. 50, 42–51.

Strobl, M. A. and Barker, D. (2016). On simulated annealing phase transitions in phylogeny reconstruction. *Molecular phylogenetics and evolution*. 101, 46–55.

Suresh, S. and Lal, S. (2017). Multilevel Thresholding based on Chaotic Darwinian Particle Swarm Optimization for Segmentation of Satellite Images. *Applied Soft Computing*. 55, 503–522.

Szabo, C., Sheng, Q. Z., Kroeger, T., Zhang, Y. and Yu, J. (2014). Science in the cloud: allocation and execution of data-intensive scientific workflows. *Journal of Grid Computing*. 12(2), 245–264.

Talatahari, S. (2016). SYMBIOTIC ORGANISMS SEARCH FOR OPTIMUM DESIGN OF FRAME AND GRILLAGE SYSTEMS. *ASIAN JOURNAL OF CIVIL ENGINEERING (BHRC)*. 17(3), 299–313.

Tao, F., Feng, Y., Zhang, L. and Liao, T. (2014). CLPS-GA: A case library and Pareto solution-based hybrid genetic algorithm for energy-aware cloud service scheduling. *Applied Soft Computing*. 19, 264–279.

Tawfeek, M. A., El-Sisi, A., Keshk, A. and Torkey, F. A. (2015). Cloud task scheduling based on ant colony optimization. *Int. Arab J. Inf. Technol.* 12(2), 129–137.

Tayal, S. (2011). Tasks scheduling optimization for the cloud computing systems. *IJAEST-INTERNATIONAL JOURNAL OF ADVANCED ENGINEERING SCIENCES AND TECHNOLOGIES*. 1(5), 111–115.

Tejani, G. G., Savsani, V. J. and Patel, V. K. (2016). Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. *Journal of Computational Design and Engineering*. 3(3), 226–249.

Teng, S., Lee, L. H. and Chew, E. P. (2007). Multi-objective ordinal optimization for simulation optimization problems. *Automatica*. 43(11), 1884–1895.

Tilak, S. and Patil, D. (2012). A survey of various scheduling algorithms in cloud environment. *International Journal of Engineering Inventions*. 1(2), 36–39.

Tiwari, A. and Pandit, M. (2016). Bid based economic load dispatch using symbiotic organisms search algorithm. In *Engineering and Technology (ICETECH), 2016*

IEEE International Conference on. 17-18 March. Tamil Nadu, India: IEEE, 1073–1078.

Tran, D.-H., Cheng, M.-Y. and Prayogo, D. (2016). A novel Multiple Objective Symbiotic Organisms Search (MOSOS) for time–cost–labor utilization tradeoff problem. *Knowledge-Based Systems*. 94, 132–145.

Trunfio, G. A. (2014). Enhancing the firefly algorithm through a cooperative coevolutionary approach: an empirical study on benchmark optimisation problems. *International Journal of Bio-Inspired Computation*. 6(2), 108–125.

Tsai, C.-W. and Rodrigues, J. J. (2014). Metaheuristic scheduling for cloud: A survey. *IEEE Systems Journal*. 8(1), 279–291.

Ullman, J. D. (1975). NP-complete scheduling problems. *Journal of Computer and System sciences*. 10(3), 384–393.

Varalakshmi, P., Ramaswamy, A., Balasubramanian, A. and Vijaykumar, P. (2011). An optimal workflow based scheduling and resource allocation in cloud. In *International Conference on Advances in Computing and Communications*. Springer, 411–420.

Verma, A. and Kaushal, S. (2017). A hybrid multi-objective Particle Swarm Optimization for scientific workflow scheduling. *Parallel Computing*. 62, 1–19.

Verma, S., Saha, S. and Mukherjee, V. (2015). A novel symbiotic organisms search algorithm for congestion management in deregulated environment. *Journal of Experimental & Theoretical Artificial Intelligence*, 1–21.

Vincent, F. Y., Redi, A. P., Yang, C.-L., Ruskartina, E. and Santosa, B. (2017). Symbiotic organism search and two solution representations for solving the capacitated vehicle routing problem. *Applied Soft Computing*. 52, 657–672.

Wang, B., Song, Y., Sun, Y. and Liu, J. (2016a). Managing Deadline-constrained Bag-of-Tasks Jobs on Hybrid Clouds with Closest Deadline First Scheduling. *KSII Transactions on Internet & Information Systems*. 10(7).

Wang, W.-J., Chang, Y.-S., Lo, W.-T. and Lee, Y.-K. (2013). Adaptive scheduling for parallel tasks with QoS satisfaction for hybrid cloud environments. *The Journal of Supercomputing*. 66(2), 783–811.

Wang, X., Wang, Y. and Cui, Y. (2014). A new multi-objective bi-level programming model for energy and locality aware multi-job scheduling in cloud computing. *Future Generation Computer Systems*. 36, 91–101.

Wang, X., Wang, Y. and Cui, Y. (2016b). An energy-aware bi-level optimization model for multi-job scheduling problems under cloud computing. *Soft Computing*. 20(1), 303–317.

Wang, Y., Li, Y., Chen, Z. and Xue, Y. (2016c). Cooperative particle swarm optimization using MapReduce. *Soft Computing*, 1–11.

Wang, Y.-N., Wu, L.-H. and Yuan, X.-F. (2010). Multi-objective self-adaptive differential evolution with elitist archive and crowding entropy-based diversity measure. *Soft Computing*. 14(3), 193–209.

Wari, E. and Zhu, W. (2016). A survey on metaheuristics for optimization in food manufacturing industry. *Applied Soft Computing*. 46, 328–343.

Wickremasinghe, B. (2010). *Cloud Analyst: A Cloud-Sim-Based Tool For Modeling And Analysis Of Large Scale Cloud Computing Environments. MEDC Project*.

Woldesenbet, Y. G., Yen, G. G. and Tessema, B. G. (2009). Constraint handling in multiobjective evolutionary optimization. *IEEE Transactions on Evolutionary Computation*. 13(3), 514–525.

Wu, F., Wu, Q. and Tan, Y. (2015). Workflow scheduling in cloud: a survey. *The Journal of Supercomputing*. 71(9), 3373–3418.

Wu, Q., Law, R., Wu, E. and Lin, J. (2013a). A hybrid-forecasting model reducing Gaussian noise based on the Gaussian support vector regression machine and chaotic particle swarm optimization. *Information Sciences*. 238, 96–110.

Wu, Q., Yun, D., Lin, X., Gu, Y., Lin, W. and Liu, Y. (2012). On workflow scheduling for end-to-end performance optimization in distributed network environments. In *Workshop on Job Scheduling Strategies for Parallel Processing*. 25 May. Shanghai, China: Springer, 76–95.

Wu, Z., Liu, X., Ni, Z., Yuan, D. and Yang, Y. (2013b). A market-oriented hierarchical scheduling strategy in cloud workflow systems. *The Journal of Supercomputing*. 63(1), 256–293.

Wu, Z., Ni, Z., Gu, L. and Liu, X. (2010). A revised discrete particle swarm optimization for cloud workflow scheduling. In *Computational Intelligence and Security (CIS), 2010 International Conference on*. 11-14 December. Guangxi Zhuang Autonomous Region, China: IEEE, 184–188.

Xu, Y., Li, K., Hu, J. and Li, K. (2014). A genetic algorithm for task scheduling on heterogeneous computing systems using multiple priority queues. *Information Sciences*. 270, 255–287.

Xue, B., Zhang, M., Browne, W. and Yao, X. (2016). A survey on evolutionary computation approaches to feature selection. 20(4), 606 – 626.

Xue, S.-J. and Wu, W. (2012). Scheduling workflow in cloud computing based on hybrid particle swarm algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*. 10(7), 1560–1566.

Yang, X.-S. (2011). Metaheuristic optimization: algorithm analysis and open problems. *Experimental algorithms*, 21–32.

Yang, Z., Tang, K. and Yao, X. (2008). Large scale evolutionary optimization using cooperative coevolution. *Information Sciences*. 178(15), 2985–2999.

Yao, G., Ding, Y., Jin, Y. and Hao, K. (2016). Endocrine-based coevolutionary multi-swarm for multi-objective workflow scheduling in a cloud system. *Soft Computing*, 1–14.

Yassa, S., Chelouah, R., Kadima, H. and Granado, B. (2013). Multi-objective approach for energy-aware workflow scheduling in cloud computing environments. *The Scientific World Journal*. 2013.

Yu, J. and Buyya, R. (2005). A taxonomy of workflow management systems for grid computing. *Journal of Grid Computing*. 3(3-4), 171–200.

Yu, Y. and Xinjie, Y. (2007). Cooperative coevolutionary genetic algorithm for digital IIR filter design. *IEEE Transactions on Industrial Electronics*. 54(3), 1311–1318.

Yuan, H., Bi, J., Tan, W., Zhou, M., Li, B. H. and Li, J. (2016). TTSA: an effective scheduling approach for delay bounded tasks in hybrid clouds. *IEEE transactions on cybernetics*.

Zhan, Z.-H., Li, J., Cao, J., Zhang, J., Chung, H. S.-H. and Shi, Y.-H. (2013). Multiple populations for multiple objectives: A coevolutionary technique for solving multiobjective optimization problems. *IEEE Transactions on Cybernetics*. 43(2), 445–463.

Zhan, Z.-H., Liu, X.-F., Gong, Y.-J., Zhang, J., Chung, H. S.-H. and Li, Y. (2015). Cloud computing resource scheduling and a survey of its evolutionary approaches. *ACM Computing Surveys (CSUR)*. 47(4), 63.

Zhang, B., Sun, L., Yuan, H., Lv, J. and Ma, Z. (2016). An improved regularized extreme learning machine based on symbiotic organisms search. In *Industrial Electronics and Applications (ICIEA), 2016 IEEE 11th Conference on*. 5-7 June. Hefei, China: IEEE, 1645–1648.

Zhang, F., Cao, J., Li, K., Khan, S. U. and Hwang, K. (2014). Multi-objective scheduling of many tasks in cloud platforms. *Future Generation Computer Systems*. 37, 309–320.

Zhang, H., Jiang, G., Yoshihira, K., Chen, H. and Saxena, A. (2009). Intelligent workload factoring for a hybrid cloud computing model. In *2009 Congress on Services-I*. 6-10

July. Los Angeles, CA: IEEE, 701–708.

Zhang, L., Li, K., Li, C. and Li, K. (2017). Bi-objective workflow scheduling of the energy consumption and reliability in heterogeneous computing systems. *Information Sciences*. 379, 241–256.

Zhang, Q., Cheng, L. and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*. 1(1), 7–18.

Zhao, C., Zhang, S., Liu, Q., Xie, J. and Hu, J. (2009). Independent tasks scheduling based on genetic algorithm in cloud computing. In *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*. 24-26 September. Beijing, China: IEEE, 1–4.

Zhao, Q., Xiong, C., Yu, C., Zhang, C. and Zhao, X. (2016). A new energy-aware task scheduling method for data-intensive applications in the cloud. *Journal of Network and Computer Applications*. 59, 14–27.

Zheng, W. and Sakellariou, R. (2013). Budget-deadline constrained workflow planning for admission control. *Journal of grid computing*. 11(4), 633–651.

Zhong, Z., Chen, K., Zhai, X. and Zhou, S. (2016). Virtual machine-based task scheduling algorithm in a cloud computing environment. *Tsinghua Science and Technology*. 21(6), 660–667.

Zhu, Z., Zhang, G., Li, M. and Liu, X. (2016). Evolutionary multi-objective workflow scheduling in cloud. *IEEE Transactions on Parallel and Distributed Systems*. 27(5), 1344–1357.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M. and Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation*. 7(2), 117–132.

Zuo, L., Shu, L., Dong, S., Zhu, C. and Hara, T. (2015). A Multi-Objective Optimization Scheduling Method Based on the Ant Colony Algorithm in Cloud Computing. *IEEE Access*. 3, 2687–2699.

Zuo, X., Zhang, G. and Tan, W. (2014). Self-adaptive learning PSO-based deadline constrained task scheduling for hybrid IaaS cloud. *IEEE Transactions on Automation Science and Engineering*. 11(2), 564–573.