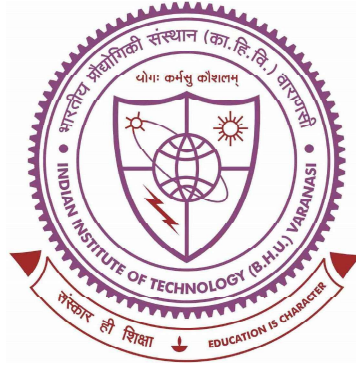


Some AI-Enhanced Multi-Objective Techniques for Task Scheduling in Cloud



Thesis submitted in partial fulfilment

for the Award of

DOCTOR OF PHILOSOPHY

in

COMPUTER SCIENCE AND ENGINEERING

by

SHWETA KUSHWAHA

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY

(BANARAS HINDU UNIVERSITY)

VARANASI – 221005


Roll Number
21071004

Year of Submission
2025

CERTIFICATE

It is certified that the work contained in the thesis titled **Some AI-Enhanced Multi-Objective Techniques for Task Scheduling in Cloud** by **Shweta Kushwaha** has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

It is further certified that the student has fulfilled all the requirements of Comprehensive Examination, Candidacy, and SOTA for the award of a PhD Degree.



Supervisor

Prof. Ravi Shankar Singh

Department of Computer Science and Engineering

Indian Institute Of Technology

(Banaras Hindu University)

Varanasi – 221005

पर्यवेक्षक/Supervisor

संगणक विज्ञान एवं अनियंत्रितकी विभाग

Department of Computer Sc. & Engg

भारतीय प्रौद्योगिकी संस्थान

Indian Institute of Technology

(काशी हिन्दू विश्वविद्यालय)


(Banaras Hindu University)

वाराणसी/Varanasi-221005

DECLARATION BY THE CANDIDATE

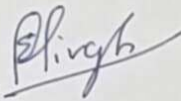
I, **Shweta Kushwaha**, certify that the work embodied in this thesis is my own bona fide work and carried out by me under the supervision of **Prof. Ravi Shankar Singh** from **July 2021 to July 2025**, at the **Department of Computer Science and Engineering**, Indian Institute of Technology (Banaras Hindu University), Varanasi. The matter embodied in this thesis has not been submitted for the award of any other degree/diploma. I declare that I have faithfully acknowledged and given credits to the research workers wherever their works have been cited in my work in this thesis. I further declare that I have not willfully copied any other's work, paragraphs, text, data, results, *etc.*, reported in journals, books, magazines, reports, dissertations, theses, *etc.*, or available at websites and have not included them in this thesis and have not cited as my own work.

Date: 15/07/25
Place: Varanasi, India

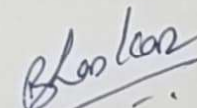
Signature of the Student 

CERTIFICATE BY THE SUPERVISOR

It is certified that the above statement made by the student is correct to the best of my/our knowledge.


Supervisor
Prof. Ravi Shankar Singh
Department of Computer Science and Engineering
Indian Institute Of Technology
(Banaras Hindu University)

Varanasi-221005
संगणक विज्ञान एवं अभियांत्रिकी विभाग
Department of Computer Sc. & Engg.
भारतीय प्रौद्योगिकी संस्थान
Indian Institute of Technology
(काशी हिन्दू विश्वविद्यालय)
(Banaras Hindu University)
वाराणसी/Varanasi-221005


आचार्य व विभागाध्यक्ष
Professor & Head
Signature of Head of Department
Department of Computer Sc. & Engg.
भारतीय प्रौद्योगिकी संस्थान
Indian Institute of Technology
(वाराणसी/Varanasi-221005)

COPYRIGHT TRANSFER CERTIFICATE

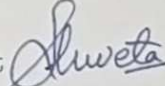
Title of the Thesis: Some AI-Enhanced Multi-Objective Techniques for Task Scheduling in Cloud

Name of the Student: Shweta Kushwaha

COPYRIGHT TRANSFER

The undersigned hereby assigns to the Indian Institute of Technology (Banaras Hindu University), Varanasi all rights under copyright that may exist in and for the above thesis submitted for the award of the Doctor of Philosophy.

Date: July 15, 2025
Place: Varanasi, India

Signature of the Student 

Note: However, the author may reproduce or authorize others to reproduce material extracted verbatim from the thesis or derivative of the thesis for author's personal use provided that the source and the Institute's copyright notice are indicated.

In loving memory of my father,
Late. Shri. Nagendra Kushwaha

ACKNOWLEDGEMENT

I sincerely thank my supervisor, Prof. Ravi Shankar Singh, for his guidance, support, and continuous motivation throughout my Ph.D. journey. His encouragement and insightful feedback were key to shaping this thesis and bringing it to completion. I also express my gratitude to the RPEC members, Prof. Sanjay Kumar Singh, Prof. Ravi Shankar Singh and Dr. Rajeev, for their constructive feedback. Special thanks to Prof. Sanjay Kumar Singh for refining my research skills and quality of my work. I am grateful to the High-Performance Computing (HPC) Lab at IIT(BHU) for fostering a supportive and collaborative environment.

I especially acknowledge my labmates, Dr. Shri Prakash Dwivedi, Dr. Rambabu Medara, Dr. Naveen Mani Upadhyay, Dr. Swati Gupta and Dr. Yaman Dua, for their support. I am deeply grateful to my close friends, Mr. Vinod Kumar, Mr. Prasanth Kumar Bevara, Mr. VVD Prasad Chelluri and Mrs. Garima Singh, for their unwavering belief in me during the most challenging times. I would also like to express my profound appreciation to Dr. Rambabu Medara and Mr. Vinod Kumar for their unwavering support, insightful guidance, and constant motivation throughout my Ph.D. journey.

I owe a special debt of gratitude to my uncles, Dr. Dayashankar Maurya and Mr. Shailesh Maurya, who have been a constant source of support, inspiration and wisdom throughout my journey. I am also deeply thankful to my mother, grandmother, brothers and whole family for their unconditional love, support, numerous teachings, and sacrifices. Lastly, I dedicate my work to my father, whose selfless care, enduring strength, belief, and boundless love have made everything possible.

(Shweta Kushwaha)

Contents

List of Figures	xix
List of Tables	xxi
List of Abbreviations	xxv
Abstract	xxvii
1 Introduction	1
1.1 Background of Cloud Computing	1
1.1.1 Applications of Cloud Computing	2
1.1.2 Key Factors and Motivations :	3
1.2 Scheduling in Cloud Computing	4
1.2.1 Importance of Efficient Scheduling in Cloud Environments	4
1.2.2 Challenges in Scheduling	5
1.2.3 Independent Task Scheduling	6
1.2.4 Workflow Scheduling in Cloud	6
1.2.5 Scheduling Constraints	7
1.2.6 Scheduling Objectives	8
1.3 System Model	9
1.3.1 Task Model	10
1.3.2 Cloud Resource Model	10
1.3.3 Task Scheduling Model	11
1.3.4 Workflow Model	12
1.3.5 Workflow Scheduling Model	13

1.3.6	Cost Model	14
1.3.7	Energy Consumption Model	15
1.3.8	Resource Utilization Model	16
1.4	Framework and Tools	16
1.4.1	CloudSim	16
1.4.2	WorkflowSim	17
1.4.3	Google Colaboratory	17
1.5	Benchmark Workflows	17
1.5.1	Montage	18
1.5.2	SIPHT	18
1.5.3	CyberShake	19
1.5.4	Inspiral LIGO	19
1.5.5	Epigenomics	20
1.6	Metrics for Performance Analysis	21
1.6.1	Evaluation Metrics	21
1.6.2	Pareto Optimality Metrics	21
1.6.3	Statistical Analysis	22
1.6.4	Computational Complexity	23
1.7	Contribution of the Thesis	23
1.8	Thesis Organization	25
2	Literature Review	27
2.1	Research Focus Areas in Cloud Computing	27
2.2	Systematic Literature Review of Tasks Scheduling in Cloud Computing	29
2.2.1	Formation of Query for Searching Articles	29
2.2.2	Inclusion and Exclusion Criteria for Articles	30
2.3	Review Procedure of Articles for Task Scheduling in Cloud Computing	31
2.4	Categorization of Strategies for Task Scheduling in Cloud Environment	32
2.5	Heuristic Strategies for Task Scheduling in Cloud	33
2.5.1	Transition: Heuristic to Metaheuristic Strategies	37
2.6	Metaheuristic Strategies for Tasks Scheduling in Cloud	37
2.6.1	Transition: Metaheuristic to Hybrid Strategies	40

2.7	Hybrid Strategies for Tasks Scheduling in Cloud	43
2.7.1	Transition: Hybrid to Learning-based Strategies	43
2.8	Learning-based Strategies for Task Scheduling in Cloud	44
2.9	State Of The Art	50
2.9.1	Heuristic Strategies	50
2.9.2	Metaheuristic Strategies	52
2.9.3	Hybrid Strategies	54
2.9.4	Learning-based Strategies	56
2.10	Research Gaps	56
2.11	Research Objectives	58
3	Deadline and Budget Constrained Scheduling	59
3.1	Introduction	59
3.2	Deadline and Budget Constrained AOA	60
3.2.1	Motivation	60
3.2.2	Problem Formulation	61
3.2.3	Proposed Methodology	62
3.2.3.1	Deadline and Budget Interval Calculation Strategy	62
3.2.3.2	Deadline and Budget Constrained AOA	64
3.2.4	Experimental Setup	72
3.2.5	Performance Analysis	73
3.2.6	Performance Observations	79
3.3	Multi Layer Perceptron Optimized Archimedes Algorithm	81
3.3.1	Motivation	82
3.3.2	Problem Formulation	82
3.3.3	Proposed Methodology	83
3.3.3.1	Sensitivity Analysis for Various Scenarios	83
3.3.3.2	Data Preparation and Feature Scaling	85
3.3.3.3	Training and Optimization	86
3.3.3.4	Multi Layer Perceptron Optimized AOA	87
3.3.4	Experimental Setup	90
3.3.5	Performance Analysis	92

3.3.6	Performance Observations	97
3.4	Summary	98
4	Local Optima Avoidance-based Scheduling	99
4.1	Introduction	99
4.2	Modified Local Escaping Archimedes Optimization Algorithm	100
4.2.1	Motivation	100
4.2.2	Problem Formulation	101
4.2.3	Proposed Methodology	102
4.2.3.1	Local Escaping Operation	103
4.2.3.2	Modified Local Escaping Archimedes Optimization	104
4.2.4	Experimental Setup	107
4.2.5	Performance Analysis	110
4.2.6	Performance Observations	113
4.3	Opposition-based Learning Enhanced Manta Ray Foraging	114
4.3.1	Motivation	115
4.3.2	Problem Formulation	116
4.3.3	Proposed Methodology	117
4.3.3.1	Deadline and Budget Interval Calculation	118
4.3.3.2	Manta Ray Foraging Optimization	118
4.3.3.3	Opposition-Based Learning	121
4.3.3.4	Local Escaping Operation	122
4.3.3.5	OBL-Enhanced Manta Ray Foraging	124
4.3.4	Experimental Setup	125
4.3.5	Performance Analysis	126
4.3.6	Performance Observations	131
4.4	Summary	132
5	Reinforcement Learning-based Scheduling	133
5.1	Introduction	133
5.1.1	Motivation	133
5.1.2	Problem Formulation	134

5.2	Proposed Methodology	135
5.2.1	Reinforcement Learning Framework	135
5.2.2	Actor and Critic Networks	136
5.2.3	Advantage Function and Policy Update	137
5.2.4	Temporal Difference Learning and Regularization	138
5.2.5	The Advantage Actor Critic Strategy	140
5.3	Experimental Setup	141
5.4	Performance Analysis	142
5.5	Performance Observations	146
5.6	Summary	146
6	Conclusions and Future Research Directions	147
	References	151
	List of Publications	167

List of Figures

1.1 Applications of Cloud Computing	2
1.2 Workflow Scheduling Framework	7
1.3 Structure of Scientific Workflows	18
1.4 Structure of the thesis and outline of key chapters for task scheduling in cloud	25
3.1 Deadline and Budget Constrained AOA	63
3.2 Illustration of Archimedes' Principle	64
3.3 Structure of the Encoded Object	66
3.4 Comparison of ADB with state of the art for Makespan	73
3.5 Comparison of ADB with state of the art for Cost	73
3.6 Comparison of ADB with state of the art for Energy Consumption	74
3.7 Comparison of ADB with state of the art for Resource Utilization	74
3.8 Percentage of Deadline and Budget Adherence (Montage)	75
3.9 Percentage of Deadline and Budget Adherence (SIPHT)	75
3.10 Percentage of Deadline and Budget Adherence (CyberShake)	75
3.11 Percentage of Deadline and Budget Adherence (Inspiral)	76
3.12 Percentage of Deadline and Budget Adherence (Epigenomics)	76
3.13 Scheduling using MLP optimized Archimedes Algorithm	84
3.14 Comparison of MLPOA with state of the art for Makespan	92
3.15 Comparison of MLPOA with state of the art for Cost	92
3.16 Comparison of MLPOA with state of the art for Energy Consumption	93
3.17 Comparison of MLPOA with state of the art for Resource Utilization	93
3.18 Percentage of Deadline Adherence	94
3.19 Percentage of Budget Adherence	94

4.1	Modified Local Escaping Archimedes Approach	102
4.2	Convergence analysis of state of the art for various workflows	108
4.3	Comparison of MLEAO with state of the art for Makespan	110
4.4	Comparison of MLEAO with state of the art for Cost	110
4.5	Comparison of MLEAO with state of the art for Energy Consumption	111
4.6	Comparison of MLEAO with state of the art for Resource Utilization	111
4.7	Opposition-Based Learning Enhanced Manta Ray Foraging Approach	117
4.8	Comparison of MRFOBL with state of the art for Makespan	126
4.9	Comparison of MRFOBL with state of the art for Cost	126
4.10	Comparison of MRFOBL with state of the art for Energy Consumption	127
4.11	Comparison of MRFOBL with state of the art for Resource Utilization	127
4.12	Percentage of Deadline Adherence	128
4.13	Percentage of Budget Adherence	128
5.1	Advantage Actor-Critic Strategy	135
5.2	Makespan and Scheduling time for 50 episodes	142
5.3	Makespan and Scheduling time for 100 episodes	143
5.4	Makespan and Scheduling time for 150 episodes	143
5.5	Makespan and Scheduling time for 200 episodes	143
5.6	Makespan and Scheduling time for 250 episodes	144
5.7	Makespan and Scheduling time for 300 episodes	144
5.8	Makespan and Scheduling time for 350 episodes	144
5.9	Makespan and Scheduling time for 400 episodes	144
5.10	Makespan and Scheduling time for 450 episodes	145
5.11	Makespan and Scheduling time for 500 episodes	145

List of Tables

2.1	Task scheduling using Heuristic Approaches	34
2.1	Task scheduling using Heuristic Approaches (Continued)	35
2.1	Task scheduling using Heuristic Approaches (Continued)	36
2.2	Task scheduling using Metaheuristic Approaches	38
2.2	Task scheduling using Metaheuristic Approaches (Continued)	39
2.3	Task scheduling using Hybrid Approaches	41
2.3	Task scheduling using Hybrid Approaches (Continued)	42
2.4	Task scheduling using ML-based Approaches	45
2.4	Task scheduling using ML-based Approaches (Continued)	46
2.4	Task scheduling using ML-based Approaches (Continued)	47
2.4	Task scheduling using ML-based Approaches (Continued)	48
2.4	Task scheduling using ML-based Approaches (Continued)	49
2.5	Comparative analysis of Heuristic Approaches	51
2.6	Comparative analysis of Metaheuristic & Hybrid Approaches	53
2.7	Comparative analysis of ML-based Task Scheduling Approaches	55
3.1	VM Instances	72
3.2	Comparison of ADB with state of the art for S-Metric	76
3.3	Comparison of ADB with state of the art for Hypervolume	77
3.4	Comparison of ADB with state of the art for Dominance	77
3.5	Comparison of ADB with state of the art for T-Test	78
3.6	Comparison of ADB with state of the art for ANOVA (Montage)	78
3.7	Comparison of ADB with state of the art for ANOVA (SIPHT)	78
3.8	Comparison of ADB with state of the art for ANOVA (CyberShake)	79

3.9	Comparison of ADB with state of the art for ANOVA (Inspiral)	79
3.10	Comparison of ADB with state of the art for ANOVA (Epigenomics)	79
3.11	Sensitivity analysis of different parameters for various workflows	84
3.12	VM Instances	91
3.13	Comparison of MLPOA with state of the art for S-Metric	95
3.14	Comparison of MLPOA with state of the art for Hypervolume	95
3.15	Comparison of MLPOA with state of the art for Dominance	95
3.16	Comparison of MLPOA with state of the art for T-Test	96
3.17	Comparison of MLPOA with state of the art for ANOVA (Montage)	96
3.18	Comparison of MLPOA with state of the art for ANOVA (SIPHT)	96
3.19	Comparison of MLPOA with state of the art for ANOVA (CyberShake)	97
3.20	Comparison of MLPOA with state of the art for ANOVA (Inspiral)	97
3.21	Comparison of MLPOA with state of the art for ANOVA (Epigenomics)	97
4.1	VM Instances	107
4.2	Sensitivity analysis of different parameters for various workflows	109
4.3	Comparison of MLEAO with state of the art for S-Metric	112
4.4	Comparison of MLEAO with state of the art for Hypervolume	112
4.5	Comparison of MLEAO with state of the art for Dominance	113
4.6	Summary of Notation	120
4.7	VM Instances	125
4.8	Comparison of MRFOBL with state of the art for S-Metric	129
4.9	Comparison of MRFOBL with state of the art for Hypervolume	129
4.10	Comparison of MRFOBL with state of the art for Dominance	129
4.11	Comparison of MRFOBL with state of the art for T-Test	130
4.12	Comparison of MRFOBL with state of the art for ANOVA (Montage)	130
4.13	Comparison of MRFOBL with state of the art for ANOVA (SIPHT)	130
4.14	Comparison of MRFOBL with state of the art for ANOVA (CyberShake)	131
4.15	Comparison of MRFOBL with state of the art for ANOVA (Inspiral)	131
4.16	Comparison of MRFOBL with state of the art for ANOVA (Epigenomics)	131
5.1	VM Instances	141

5.2 Comparison of A2CS with state of the art for ANOVA (Makespan) 145

5.3 Comparison of A2CS with state of the art for ANOVA (Scheduling Time) 145

List of Abbreviations

Abbreviation	Description
A2CS	Advantage Actor Critic Strategy
ACO	Ant Colony Optimization
ADB	Deadline and Budget Constrained Archimedes Optimization
AI	Artificial Intelligence
ANOVA	Analysis of Variance
AOA	Archimedes Optimization Algorithm
CPOP	Critical Path on a Processor
CPU	Central Processing Unit
DAG	Directed Acyclic Graph
DDPG	Deep Deterministic Policy Gradient
DQN	Deep Q Network
EARES	Energy Aware Reliability Enhancement Scheduling
FCFS	First Come First Serve
FPA	Flower Pollination Algorithm
GA	Genetic Algorithm
GWO	Grey Wolf Optimization
HEFT	Heterogenous Earliest Finish Time
LEO	Local Escaping Operation
MDP	Markov Decision Process
MFGA	Modified Fuzzy Genetic Algorithm
MI	Million Instructions
MIPS	Million Instructions Per Second

LIST OF ABBREVIATIONS

Abbreviation	Description
ML	Machine Learning
MLEAO	Modified Local Escaping Archimedes Optimization
MLP	Multi Layer Perceptron
MLPOA	Multi Layer Perceptron Optimized Archimedes Approach
MRFO	Manta Ray Foraging Optimization
MRFOBL	Opposition-Based Learning Enhanced Manta Ray Foraging
MVO	Multi Verse Optimizer
NNRL	Neural Network-based Reinforcement Learning
OBL	Opposition Based Learning
PE	Processing Element
PSO	Particle Swarm Optimization
QoS	Quality of Service
ReLU	Rectified Linear Unit
RL	Reinforcement Learning
SIPHT	Three Dimensional
SJF	Shortest Job First
SLA	Service Level Agreement
SLR	Systematic Literature Review
SOS	Symbiosis Organisms Search
SOTA	State-of-the-Art
VM	Virtual Machine

Abstract

Cloud computing has emerged as the backbone of scalable and cost-effective computational infrastructures, particularly for scientific and data-intensive applications. Within this paradigm, task scheduling plays a critical role in mapping tasks onto virtualized resources while optimizing multiple objectives such as execution time, monetary cost, energy consumption, and resource utilization. The challenge, however, lies in the intrinsic complexity of the task scheduling problem, which is NP-hard due to the combinatorial nature of resource-task allocations, availability of cloud resources, and the necessity to meet Quality of Service constraints like deadlines and budgets. This thesis addresses this multifaceted problem by developing five novel AI-enhanced approaches, each targeting a different dimension of the task scheduling problem. These methods are evaluated using well-known scientific workflows, such as Montage, CyberShake, SIPHT, Inspiral, and Epigenomics, and randomly generated tasks with varying characteristics under different resource configurations. The evaluation is multi-objective in nature and grounded in rigorous performance metrics and statistical validation.

The first chapter of the thesis introduces the core motivations, challenges, and research objectives. It elaborates on the complications of dealing with large-scale workloads in heterogeneous and elastic cloud environments. It models the system architecture for the task and workflow scheduling problem and provides the mathematical formulations for task model, workflow model, resource model, task scheduling model that incorporates makespan and scheduling time and workflow scheduling model that incorporates makespan, cost, energy consumption and resource utilization models. Further, it discusses the frameworks such as CloudSim, WorkflowSim and Google Colaboratory, as the simulation tools utilized in this work. The chapter discusses the experimental methodology employed to validate the proposed approaches. It discusses the benchmark workflows on which the scheduling approaches were tested, followed by the metrics used for

performance analysis of the approaches, including the Pareto-optimality metrics and statistical analysis. It outlines the objectives of the research, which include developing scalable, deadline and budget-constrained scheduling, local optima avoidance-based scheduling and reinforcement learning-based methods. The introduction also presents a high-level overview of the proposed approaches.

Chapter 2 presents an extensive literature survey that categorizes existing workflow scheduling algorithms into several major paradigms: rule-based heuristics, metaheuristic algorithms, hybrid approaches and machine learning-based techniques. This chapter critiques the limitations of mono-objective and static scheduling algorithms, including their inability to handle task dependencies, fluctuating resource availability, and competing optimization objectives. It also explores hybrid models and emerging reinforcement learning techniques and lists their shortcomings in terms of convergence speed, generalizability, and constraint handling. The survey concludes by identifying significant research gaps: the need for hybrid, adaptive, multi-objective and scalable algorithms that can provide feasible, near-optimal schedules for tasks with varying characteristics while simultaneously satisfying deadline and budget constraints.

Chapter 3 addresses this gap with the first two proposed algorithms. The Deadline and Budget-Constrained Archimedes Optimization Algorithm extends the physics-inspired Archimedes Optimization Algorithm to the workflow scheduling domain by embedding budget and deadline constraints into the optimization process. ADB dynamically adjusts the search process based on the feasibility and fitness of candidate schedules, offering improved convergence behavior, incorporating exploration and exploitation and better constraint compliance than other approaches. The chapter then introduces the Multi-Layer Perceptron Optimized Archimedes approach, which hybridizes the AOA metaheuristic with a Multi-Layer Perceptron. The MLP serves as a surrogate evaluator that learns from historical schedule data to predict optimal control parameter values. By using the fitness based on previous scheduling decisions, it enables more effective tuning of the AOA's parameters, thereby guiding the search process more efficiently. This hybrid approach accelerates convergence while maintaining high schedule quality and constraint satisfaction. Experimental results show that both ADB and MLPOA outperform baseline methods on various performance indicators.

Chapter 4 continues the exploration of metaheuristic enhancements with two additional con-

tributions. The Modified Local Escaping Archimedes Optimization Algorithm is proposed to overcome local optima entrapment and enhance exploration in high-dimensional scheduling problems. MLEAO incorporates a Local Escaping Operator that is selectively activated to introduce perturbation into the population. The second model in this chapter, Manta Ray Foraging Optimization with Opposition-Based Learning, draws inspiration from manta ray foraging behavior and augments it with various Opposition-Based Learning strategies. These opposition techniques are designed to widen the search space coverage and accelerate convergence towards global optima while maintaining population diversity. The model retains the LEO mechanism from MLEAO to ensure perturbation in the stages of stagnation. The combined strategies make MRFOBL highly effective in solving complex, multi-objective scheduling problems. Comparative experiments demonstrate that these approaches produce superior Pareto fronts and outperform other multi-objective optimization algorithms while maintaining computational efficiency.

Chapter 5 introduces a paradigm shift from metaheuristic-based optimization to adaptive decision-making through reinforcement learning. The Advantage Actor-Critic Strategy models the scheduling problem as a Markov Decision Process and learns a policy to assign tasks to virtual machines in a way that maximizes cumulative reward. The actor network proposes actions, while the critic network computes the advantage and evaluates those actions. The reward function is carefully designed to balance multiple objectives: minimizing makespan and scheduling time. Unlike static optimization approaches, A2CS continuously learns and adapts to environmental changes, including fluctuating resource availability and dynamic task arrivals. This makes it particularly suitable for real-time scheduling scenarios. The experimental evaluation shows that A2CS generalizes well across different task characteristics and maintains high-quality performance. It represents an important contribution to the development of autonomous, self-learning scheduling systems in cloud environments.

Each of the five proposed approaches is rigorously evaluated using scientific workflows and tasks of varying sizes and characteristics. The thesis employs a rich set of performance metrics, makespan, execution cost, energy consumption, resource utilization, as well as Pareto-optimality indicators like hypervolume, S-metric and dominance rate. Statistical significance is validated using t-tests and ANOVA to ensure the robustness of performance comparisons. Collectively, the models demonstrate clear advantages over existing scheduling algorithms in both constrained and unconstrained scenarios. The ADB and MLPOA models provide robust performance in

constrained environments, while MLEAO and MRFOBL enhance solution diversity and convergence quality. A2CS, on the other hand, introduces adaptability and self-learning capabilities, positioning itself as a scalable solution for real-time, cloud-native environments.

The work addresses key challenges such as multi-objective trade-offs, deadline and budget constraint handling, scalability, search diversity, local optima avoidance and adaptability. The research findings open new avenues for future exploration, including online adaptive scheduling, federated reinforcement learning, explainable AI in scheduling, and integration with emerging paradigms like edge computing and fog computing. This thesis thus lays a solid foundation for the development of next-generation intelligent task schedulers capable of meeting the diverse and evolving demands of cloud computing environments.

Keywords: Task Scheduling, Workflow Scheduling, Cloud Computing, Multi-Objective Optimization, Deadline Constraint, Budget Constraint, Metaheuristic and Hybrid Algorithms, Archimedes Optimization Algorithm, Manta Ray Foraging Optimization, Opposition Based Learning, Local Escaping Operator, Multi-Layer Perceptron, Reinforcement Learning, Advantage Actor-Critic Learning, Scientific Workflows, Makespan Optimization, Cost Minimization, Energy Efficiency, Resource Utilization, AI-enhanced Scheduling, Pareto Optimality.