

Abstract

In today's era, when each one of us want every component of technology to be available anytime and anywhere, Cloud Computing has provided us with this ease of doing things according to user demand. Huge amount of resources are available but all these resources could not be utilised efficiently as even the distributed computing could not use it as efficiently as it should be used. Cloud Computing is a term used to name the on demand computing facility provided by many online vendors such as Amazon, Google, Microsoft. Cloud Computing has emerged as one of the fastest growing technologies in the computer industry. It offers on demand, pay per use access to infrastructure, platforms and applications. Due to increase in its popularity, huge number of requests need to be handled in an efficient manner. Task scheduling helps the requests to assign a particular resource so as to perform effectively. The numerous requests a server receive may or may not be dependent. Workflow scheduling is used where there is dependency between tasks. Many algorithms have been developed for scheduling of these tasks.

This thesis addresses the critical issue of efficient task scheduling in cloud computing environments. The exploration begins with an introduction to cloud computing, emphasizing the significance of on-demand computing resources and the challenges posed by task scheduling. A comprehensive survey of existing literature and methodologies is presented to establish the context and highlight gaps in current research.

Our first focus is on workflow scheduling, where the Jaya Optimization Algorithm is introduced and compared with various nature-inspired algorithms, including Particle Swarm Optimization, Genetic Algorithm, Ant Colony Optimization, Honey Bee, and Cat Swarm Optimization. The evaluation revolves around execution cost and makespan, utilizing benchmark functions such as Montage, CyberShake, Inspiral, and Sipht. We have also used the same algorithm aiming to minimize both makespan and execution cost, offering solutions based on user-assigned weights. The comparative analysis against other algorithms highlights the superior performance of the Jaya algorithm concerning both cost and time.

The thesis also explores multi objective task scheduling algorithms in cloud. Firstly, we describe Grey Wolf Optimization (GWO) for workflow scheduling in a cloud environment, focusing on multi-objective optimization to minimize operational cost, time, and achieve load balancing. The study demonstrates the effectiveness of the GWO algorithm in reducing overall costs and implementation time, contributing to improved system reliability and operational efficiency. We also introduce the Multi-Objective Whale Optimization-

Based Scheduler (WOA-Scheduler) for cloud task scheduling. This approach employs the Whale Optimization Algorithm to simultaneously optimize cost, time, and load balancing. Comparative analyses highlight the scheduler's effectiveness in diverse cloud computing environments.

Also, we have introduced some Hybrid Task Scheduling Optimization algorithms. Starting with Hybrid Jaya-Particle Swarm Optimization algorithm, the hybrid approach is meticulously crafted to tackle the complexities of task scheduling within a cloud environment. The integration of Jaya and PSO exhibits superior results, outperforming established algorithms like Genetic Algorithm, Honey Bee, Cat Swarm Optimization, Ant Colony Optimization, and Jaya itself, particularly in execution cost and time efficiency. Our work introduces an Improved Jaya Harmony Search (IJHS) algorithm for task scheduling. This innovative approach harnesses the strengths of Jaya for parameter optimization within the Harmony Search framework. The overarching goal of achieving load balancing, cost minimization, and timely task completion guides the IJHS algorithm. Through comprehensive simulations and comparative analyses, the IJHS algorithm demonstrates its efficacy and superiority against various algorithms, including Harmony Search, Genetic Algorithm, Particle Swarm Optimization, and First Come First Serve. Also, we have introduced an algorithm named mergedCSO. This algorithm demonstrates the integration of CSO and PSO algorithms along with their implementation for the application to task scheduling. It shows better results than individual CSO and PSO in terms of Execution Cost and Running Time of algorithms.

Collectively, this thesis contributes to the understanding and advancement of cloud-based task scheduling, presenting innovative algorithms and approaches to address the dynamic challenges posed by evolving cloud services.