# Project Report: Task Scheduling Optimization in Fog-Cloud Environments

## 1. Introduction

This project explores various scheduling algorithms for optimizing task allocation in a Fog-Cloud computing environment. Fog-Cloud environments combine the low latency of fog computing with the high computational power of cloud computing, making them ideal for processing a wide range of tasks. However, optimizing the allocation of these tasks to the available resources is a significant challenge, especially when considering multiple conflicting objectives like minimizing makespan, energy consumption, and processing costs.

### 1.1 Objective

The primary goal of this project is to implement and evaluate different scheduling algorithms to optimize the allocation of tasks across fog and cloud nodes. The algorithms are evaluated based on their ability to minimize the following key performance metrics:

- **Makespan:** The total time required to complete all tasks.
- **Energy Consumption:** The total energy consumed by the fog and cloud nodes during task execution.
- **Processing Cost:** The monetary cost associated with processing the tasks.

## 2. Project Overview

### 2.1 Base Article Reference

The foundation of this project is the article "A Multi-objective Model for Task Scheduling Optimization in Fog-Cloud Computing Environments" by Sadoon Azizi. The article presents a multi-objective optimization model that addresses the challenges of task scheduling in Fog-Cloud environments by simultaneously optimizing the makespan, energy consumption, and processing costs.

### 2.2 Implemented Algorithms

Several scheduling algorithms were implemented, including:

- **Random Scheduling:** A baseline algorithm that randomly assigns tasks to nodes.
- **Power of Two Choices (P2C) Scheduling:** An algorithm that selects the best node out of two random choices.
- **Genetic Algorithm (GA):** An evolutionary algorithm that evolves a population of solutions over iterations to find an optimal task-node mapping.
- **Simulated Annealing:** An optimization technique that gradually refines the solution by mimicking the annealing process in metallurgy.
- **Particle Swarm Optimization (PSO):** An optimization algorithm that simulates a group of particles (potential solutions) moving through the solution space.
- **Proposed Scheduling Algorithm:** A heuristic algorithm that combines various strategies to enhance the scheduling performance.
- **Enhanced Hybrid Genetic Algorithm (E-HGA):** A hybrid approach that integrates the proposed algorithm with genetic algorithms to achieve a more balanced optimization.

## 3. Performance Metrics

### 3.1 Results of the Proposed Algorithm

- **Makespan:** 18.7440

- **Energy Consumption:** 38,295.51
- **Processing Cost:** 114.59
- **Fit Value:** 0.6363

### 3.2 Results of the Enhanced Hybrid Genetic Algorithm (E-HGA)

- **Makespan:** 20.3846
- **Energy Consumption:** 39,516.23
- **Processing Cost:** 108.32
- **Fit Value:** 0.6505

## 4. Analysis and Comparison with the Base Article

### 4.1 Comparison with Base Article

The base article reported significant improvements in optimizing the three key metrics using the proposed heuristic algorithm. The article claims approximately 98%, 43%, and 32% improvements over random, genetic, and power of two choices algorithms, respectively.

### 4.2 Performance Analysis

- **Makespan:** The Proposed Algorithm yields a lower makespan compared to E-HGA, indicating better efficiency in completing all tasks. This result is particularly important for real-time applications where minimizing completion time is critical.
- **Energy Consumption:** The Proposed Algorithm is more energy-efficient than E-HGA, which is crucial for resource-constrained environments where energy conservation is a priority.

- **Processing Cost:** E-HGA achieves a lower processing cost, making it more suitable for cost-sensitive environments. This improvement highlights the advantage of integrating the proposed algorithm with genetic algorithms.
- **Fit Value:** E-HGA has a higher fit value than the Proposed Algorithm, indicating a better overall balance between the three metrics. The higher fit value suggests that E-HGA provides a more comprehensive optimization, making it preferable in scenarios where all three metrics are equally important.

### 4.3 Conclusion

The results demonstrate that while the Proposed Algorithm excels in minimizing makespan and energy consumption, the Enhanced Hybrid Genetic Algorithm (E-HGA) offers a more balanced optimization, particularly in reducing processing costs. The higher fit value achieved by E-HGA indicates that it is more effective in meeting the multi-objective optimization goals outlined in the base article.

### 4.4 Recommendations

- **For environments where completion time is critical:** The Proposed Algorithm is recommended due to its lower makespan.
- **For cost-sensitive environments:** E-HGA is preferable due to its lower processing cost and higher overall fit value.

## 5. Conclusion and Future Work

### 5.1 Summary

This project successfully implemented and evaluated various scheduling algorithms in a Fog-Cloud computing environment. The analysis of the results shows that while the Proposed Algorithm performs well in specific metrics, the Enhanced Hybrid Genetic Algorithm (E-HGA) provides a more balanced optimization across multiple objectives.

### 5.2 Future Work

Future work could involve:

- **Integration with Real-world Data:** Testing the algorithms with real-world datasets to assess their performance in practical scenarios.
- **Network Considerations:** Extending the model to account for network-related constraints such as latency and bandwidth, which are crucial in distributed computing environments.
- **Scalability:** Evaluating the scalability of the algorithms in larger, more complex Fog-Cloud environments.

# Contact Information


- **Developer:** Kiarash Karimi
- **Location:** Shiraz, Iran
- **GitHub:** [kiarashkarimi]
- **Email:** akiarash10@gmail.com


Friday - August 30 - 2024

2:45 am ( Tehran UTC+3:30 )