Cloud-based Monte Carlo Simulations: An Embarrassingly Parallel Approach

Brikelda Sema, Daniel González Arango, Nicolle García, Santiago Tejada Orozco

LULEÅ UNIVERSITY OF TECHNOLOGY

INTRODUCTION

Embarrassingly parallel (EP) algorithms allow independent processing of data segments with minimal communication, optimizing performance in distributed settings. Parallel and distributed computing are essential for high-performance applications like scientific simulations, data analysis, and artificial intelligence [1, 2]

USE CASES

- **Bag-of-tasks (BoT) applications** [3]**:**
  - Monte Carlo simulations.
  - Massive searches.
  - Video Rendering.
  - Data mining algorithms.
- **Some classes of online services:**
  - Index processing in web search.

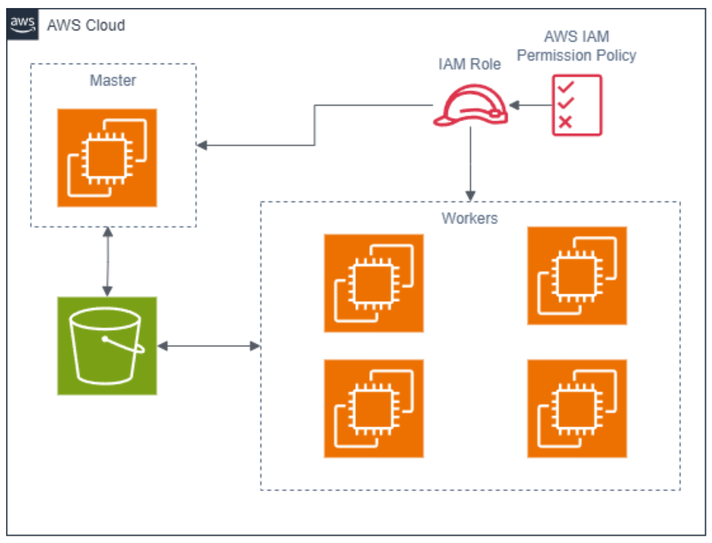
EXPERIMENTAL SETUP

Three scenarios were made to demonstrate the advantages of embarrassingly parallelism. Each scenario was executed 30 times for further analysis.

Scenario 1
The Montecarlo method was used in an EC2 instance to approximate the area under the curve of a multidimensional function. The function considers 1,000,000 samples and 10 dimensions.

Scenario 2
Using the same instance of previous scenario, multithreading was added, employing four threads.

Scenario 3
EP was implemented by creating a master instance that splits the tasks for the four worker instances. The workers pick their tasks from the bucket and return the results. Finally, the master computes the average integral and the elapsed time.



DEMO OVERVIEW & OBJECTIVES

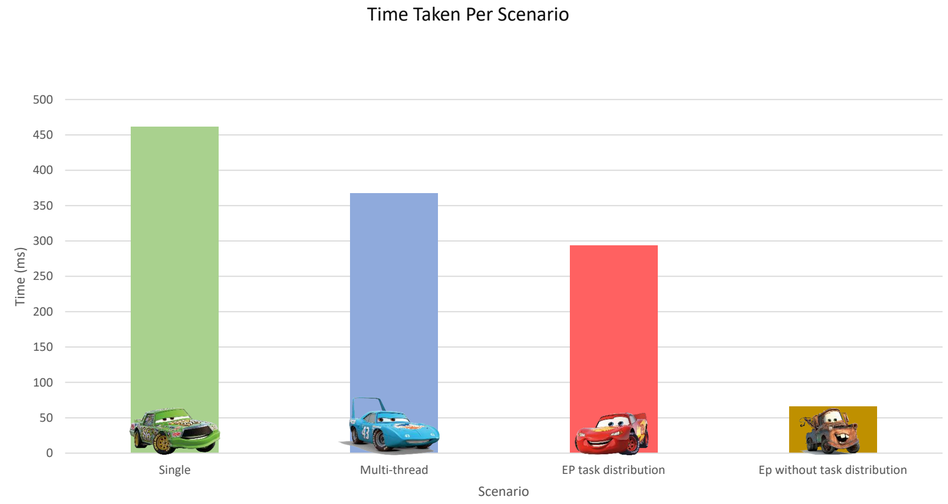
Estimate a high-dimensional integral using a Monte Carlo simulation, with the workload distributed across several AWS EC2 worker instances. Each worker processes a portion of the overall workload, saving intermediate results to AWS S3. A master instance then gathers these results from S3 and combines them to compute the final outcome.

The objectives defined are:

- Create a practical example to demonstrate embarrassingly parallel computing by showing how the Monte Carlo simulation can be split into independent tasks that run on multiple EC2 instances.
- Compare the execution time for the different simulation scenarios.
- Evaluate the impact of the overhead introduced by having a master instance that coordinates the distribution of tasks across worker instances.

RESULTS

- One-way ANOVA was used to evaluate the function results, and it was shown there is not significant difference between the scenarios for the integral computation.
- The following bar chart shows the average computation time per scenario. As can be seen, the calculation using EP yielded the best results, being significantly faster.



CONCLUSION

It was demonstrated that using cloud resources for an embarrassingly parallel approach can effectively reduce the execution time while maintaining the accuracy of the calculations.

REF.