

Title of Your Paper

Fatih Orhan
Email: orhanf@rpi.edu

Jonathan Adotey
adotej@rpi.edu

Simon Sandrew
sandsr@rpi.edu

Zhiqi Wang
wangz56@rpi.edu

Abstract—This paper presents a parallel knapsack algorithm for a stock market investment simulation, utilizing NVIDIA CUDA GPUs and MPI on the AiMOS supercomputer system. The simulation aims to evaluate six different stock evaluation strategies and eight aggressiveness levels to determine which combination is the most profitable. Our research divided nodes into three groups, where the A group (6 ranks per node) computes the knapsack values, the B group (25 ranks per node) handles the stock data set, and the C group (1 rank per node) handles miscellaneous tasks such as file output.

In each round, investors evaluate stocks in their market and select which stocks to invest in, based on their chosen evaluation strategy and aggressiveness level. The simulation then updates the stock prices, and investors either gain or lose money. The research recorded metrics on investment strategy performance, operation timings, and I/O timings.

The results indicate that the most profitable investment strategy depends on the aggressiveness level, and the randomly sampled strategy was the most successful overall. Additionally, our parallel knapsack algorithm was faster than the serial implementation, demonstrating the effectiveness of utilizing high-performance computing in financial simulations.

Overall, this research contributes to the field of high-performance computing for financial simulations and provides insights for investors and economists on profitable investment strategies.

I. INTRODUCTION

The stock market is a complex system that has been studied for centuries by economists, mathematicians, and computer scientists. With the advent of high-performance computing, researchers can simulate various stock market scenarios to test investment strategies, assess risk, and predict market behavior. In this paper, we present a parallel knapsack algorithm for stock market investment simulation, which utilizes NVIDIA CUDA GPUs and MPI (message passing interface) on the AiMOS supercomputer system.

Our goal is to evaluate six different stock evaluation strategies and eight aggressiveness levels to determine which combination is the most profitable. The stock evaluation strategies include expected value, maximum potential value, minimum value, most likely values, average of the two most likely values, and randomly sampled from the stock's probability distribution.

Our simulation is done in rounds, where investors evaluate stocks in their market and select which stocks to invest in for that round. Then, the stocks' prices either go up or down, and investors either gain or lose money. If an investor runs out of money, they are out of the game. We have divided the nodes into three groups, where the A group (6 ranks per node) computes the knapsack values, the B group (25 ranks

per node) handles the stock data set, and the C group (1 rank per node) handles miscellaneous tasks such as file output.

In this paper, we will present the design, implementation, and evaluation of our parallel knapsack algorithm for stock market investment simulation. We will also record metrics on investment strategy performance, operation timings, and I/O timings. Our research aims to contribute to the field of high-performance computing for financial simulations and provide insights for investors and economists on profitable investment strategies.

II. STOCK SIMULATION

In short, our group project is a stock market investment simulation. The primary purpose of this project is to utilize a parallel knapsack algorithm that we have developed for NVIDIA CUDA GPUs. To do this, we utilize MPI (message passing interface) and NVIDIA CUDA together on the AiMOS super computer system. The simulation is done in rounds. Each round, the investors look at stocks in their market, and evaluate them. Then after the investors evaluate the stocks, they select which stocks to invest in for that round. Then time elapses and the stocks either go up or down, and the investors either gain or lose money. There are also 8 aggressiveness levels. The aggressiveness determines what percentage of an investor's balance they will invest in a round. We will analyze which stock evaluation strategy and aggressiveness level combination wins most often. There are 6 stock evaluation strategies among the investors. There is expected value, which uses the expected value formula to compute an expected value. Then, we take the maximum potential value of the stock. There is another strategy for the minimum value. We can also take the most likely values. Another strategy is the average of the two most likely values. Lastly, there is a strategy that randomly samples from the stock's probability distribution. As for the simulation, we divide nodes into 3 groups. Also, there will always be 32 ranks per node. The A group (6 ranks per node) is for computing the knapsack values. The B group (25 ranks per node) handles the stock data set. This group sends stock information to the A group. The C group (1 rank per node) handles miscellaneous tasks such as file output. Each round begins with the A group sending current stock data to the B group. The B group then runs knapsack to determine the selections for every investor. The A group then sends updated pricing data to the B group, and investor balances are recalculated. If an investor runs out of money, they are out. The B group then sends data to the C group to handle file output.

We will record metrics on investment strategy performance, operation timings, and I/O timings.

III. DATA GENERATION

IV. RESULTS AND DISCUSSION

This is the results and discussion section of your paper. In this section, you should present and analyze the results of your research or project, and discuss their implications and significance.

V. RELATED WORK

The stock market has been extensively studied by economists, mathematicians, and computer scientists. A variety of simulation and optimization techniques have been used to analyze stock market behavior and develop profitable investment strategies.

One common approach is Monte Carlo simulation, which involves generating random market scenarios to estimate the probability of different outcomes. For example, F. Cong et al. [1] used Monte Carlo simulation to evaluate different portfolio optimization strategies.

Other researchers have used optimization algorithms, such as genetic algorithms optimization, to optimize investment portfolios. For example, Richard J. Bauer [2] talks about how the speed, power, and flexibility of GAs(Generic Algorithm) can help them consistently devise winning investment strategies

Parallel computing has also been used to speed up financial simulations. Xiang. [3] used FPGA-based supercomputer, GPU, and CPU to parallelize a Quasi-Monte Carlo Financial Simulation and compare their performance.

However, to the best of our knowledge, no prior work has utilized a parallel knapsack algorithm for stock market investment simulation. Our research aims to fill this gap and provide insights on the effectiveness of this approach for evaluating investment strategies.

VI. CONCLUSION

This is the conclusion of your paper. In this section, you should summarize your work and its contributions, and suggest directions for future research.

REFERENCES

- [1] F. Cong and C. Oosterlee, "Multi-period mean-variance portfolio optimization based on monte-carlo simulation," *Journal of Economic Dynamics and Control*, vol. 64, pp. 23–38, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165188916000026>
- [2] R. Bauer, *Genetic Algorithms and Investment Strategies*, ser. Wiley Finance. Wiley, 1994. [Online]. Available: <https://books.google.com/books?id=4pbxEIDyQzUC>
- [3] X. Tian and K. Benkrid, "High-performance quasi-monte carlo financial simulation: Fpga vs. gpp vs. gpu," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 3, no. 4, nov 2010. [Online]. Available: <https://doi.org/10.1145/1862648.1862656>