

GPU Accelerated Tabu Search for Solving Mathematical Optimization Problems



Kwamaine Taylor and Antonio Segura
Applied Research Experience
in Electrical and Computer Engineering
AREECE 2018

School of Engineering and Computer Science, Oakland University
Faculty Mentor: Dr. Shadi Alawneh



Introduction

Tabu search is a metaheuristic search method that uses adaptive memory and heuristic search methods to efficiently solve incomplete and imperfect optimization problems. This poster explores the feasibility of using tabu search with Cuda-Enabled GPUs (Graphics Processing Unit) to improve optimization results. The work presented in this poster investigates the improvements that tabu search can provide in parallel computing. It will show that tabu search improved GPU has higher speeds of optimization compared to both single-core and multi-core CPUs (Central Processing Unit). Conclusions will be drawn regarding the feasibility and performance of using tabu search on Cuda-Enabled GPUs.

Linear Optimization Problem

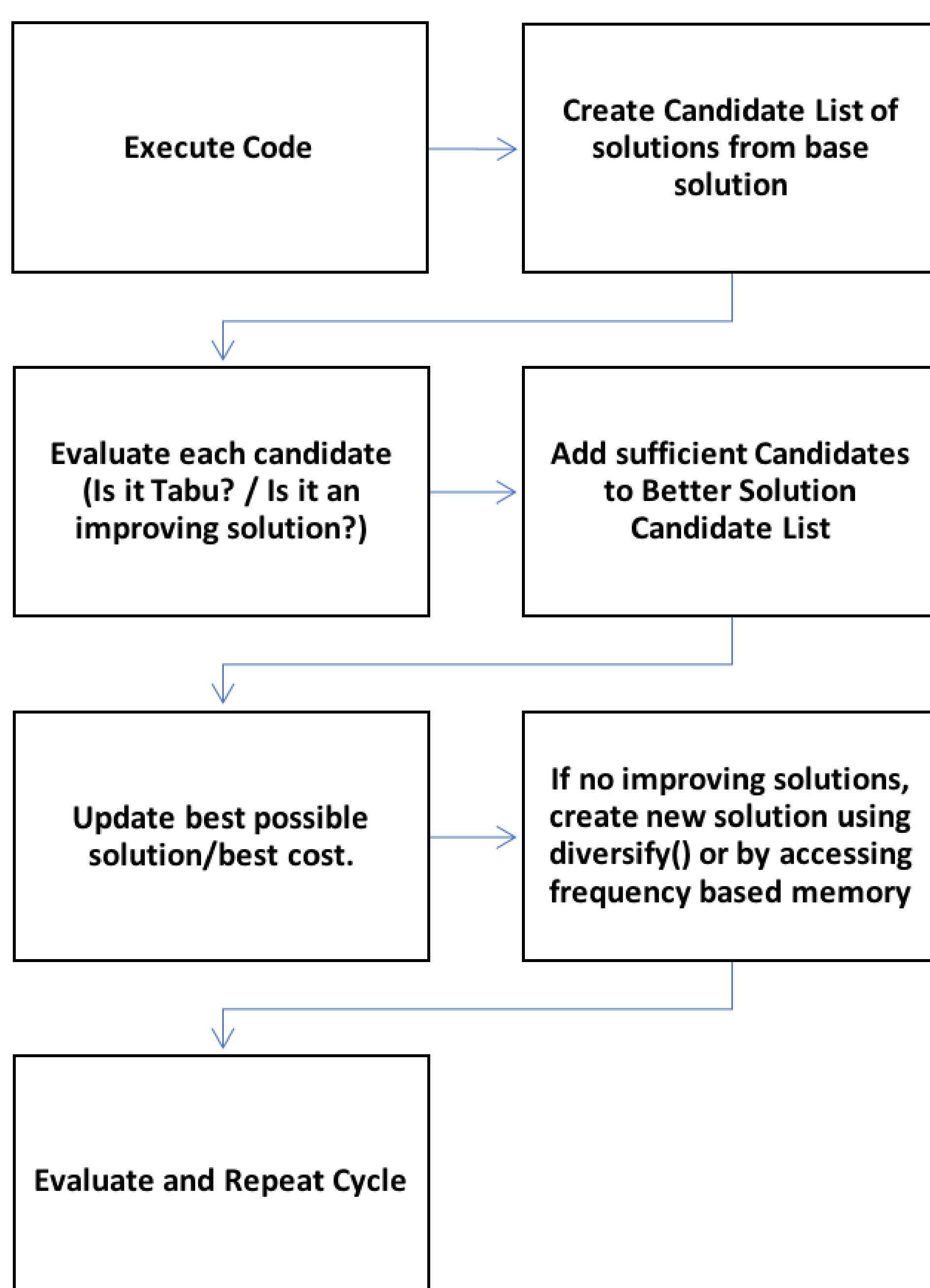
The problem we sought out to solve is a more limited linear optimization problem, solely based upon finding the lowest possible value from an objective function as quickly as possible- including inequality constraints that bounded the range that each variable could occupy. The goal is to benchmark our program in regards to how quickly it can solve an optimization problem with as little as

General Format, Minimized Linear Optimization Problem

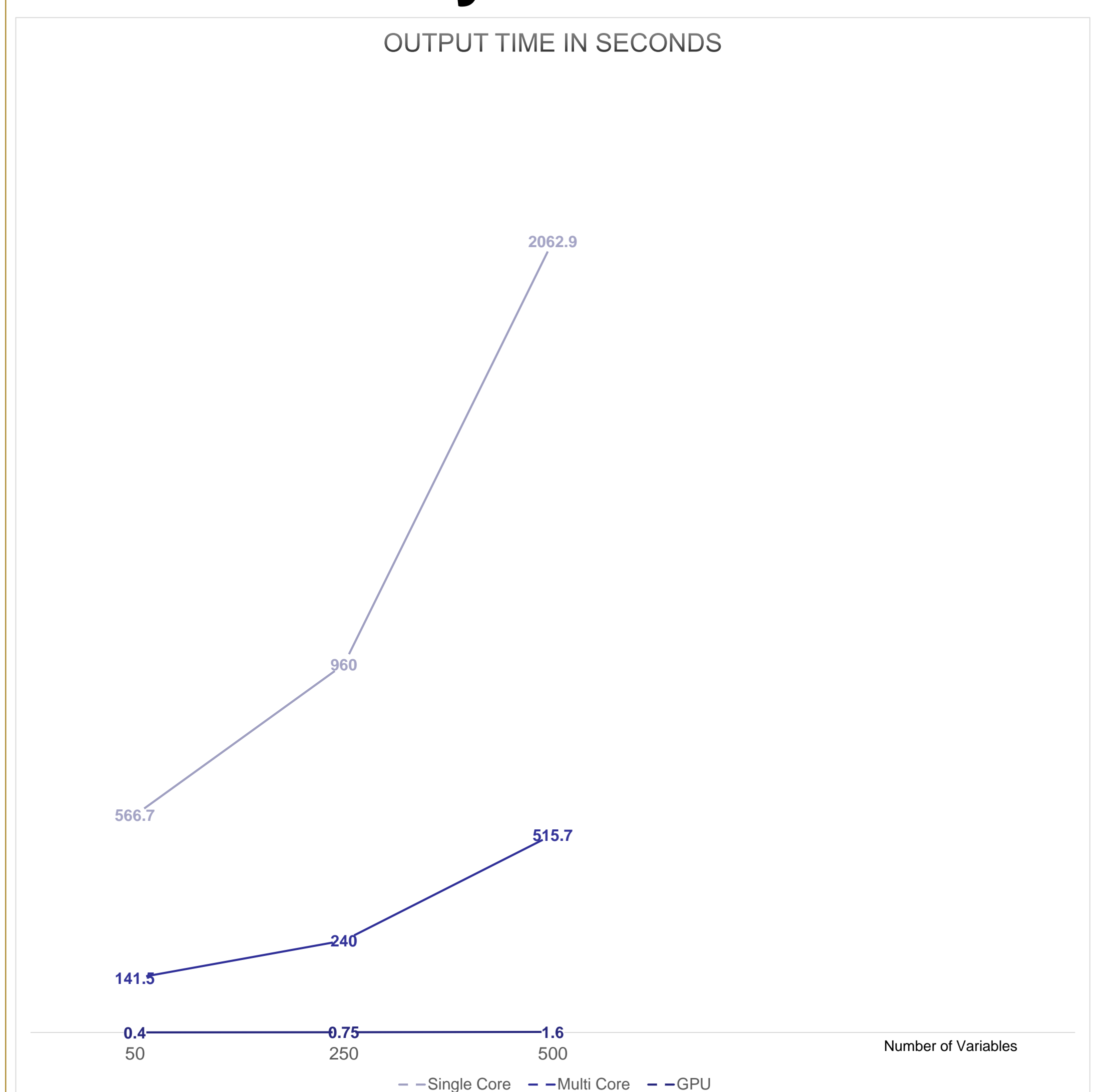
$$C_1X_1 + C_2X_2 \dots + C_nX_n$$
$$a_1 < X_1 < b_1 \quad a_2 < X_2 < b_2 \quad a_n < X_n < b_n$$

Design

The main focus of design is to maximize performance with a linear equation that has an unlimited amount of variables. Tabu Search allows various ways of approaching this idea. We mainly focus on the use of hill climbing and candidate list through vectors. Within Visual Studio, we used C++ and Open MP to create a single and multi core CPU enabled workplace. Our GPU workspace is produced in CUDA C. This project was tested on an Intel i5-6500 quad core at 3.20 GHz [CPU] and a NVIDIA GeForce GTX 1060 6 GB with 1280 CUDA cores [GPU].



Preliminary Results



Conclusion and Future Work

In conclusion, we can gather that GPU produces better results because of its core availability. With the use of GPU, we can reduce our output time, even as the number of variables interest drastically. Further research can be done to reduce limitations. As a result, could also result in better results.