GPU-based Massively Parallel Implementation of Metaheuristic Algorithms

Robert Nowotniak, Jacek Kucharski

Computer Engineering Department, Technical University of Lodz

{rnowotniak,jkuchars}@kis.p.lodz.pl

Abstract

In this research, implementation of a state-of-the-art evolutionary algorithm, Quantum-Inspired Genetic Algorithm[1,2,3] (QIGA), in massively parallel environment (Graphics Processing Units) has been presented. Contrary to many recent papers concerning parallel implementation of evolutionary algorithms, in this paper a novel approach has been taken. QIGA algorithm has been implemented entirely as a computational kernel. Parallelization of the algorithm has been performed on two levels: In a block of threads, each thread transforms a separate individual or different gene; In each block, separate populations with same or different parameters are evolved. ***Finally, the computations have been distributed to eight GPU devices, and over 400x speedup has been gained in comparison to sequential implementation of the algorithm in ANSI C on one Intel Core i7 2.93GHz CPU core****. Correctness of the results has been verified in statistical analysis. The presented approach can be applied to experimentation with a broad class of metaheuristics.*

1. CUDA™ Massively Parallel Architecture

The architecture of a modern graphics card is usually organized as follows. The card can be equipped with several GPU devices. Each GPU device, has many **streaming multi-processors** (SMs) with own flow control and on-chip shared memory units. Each multi-processor has many **streaming processors** (SPs), consisting of independent arithmetic login units (ALUs).

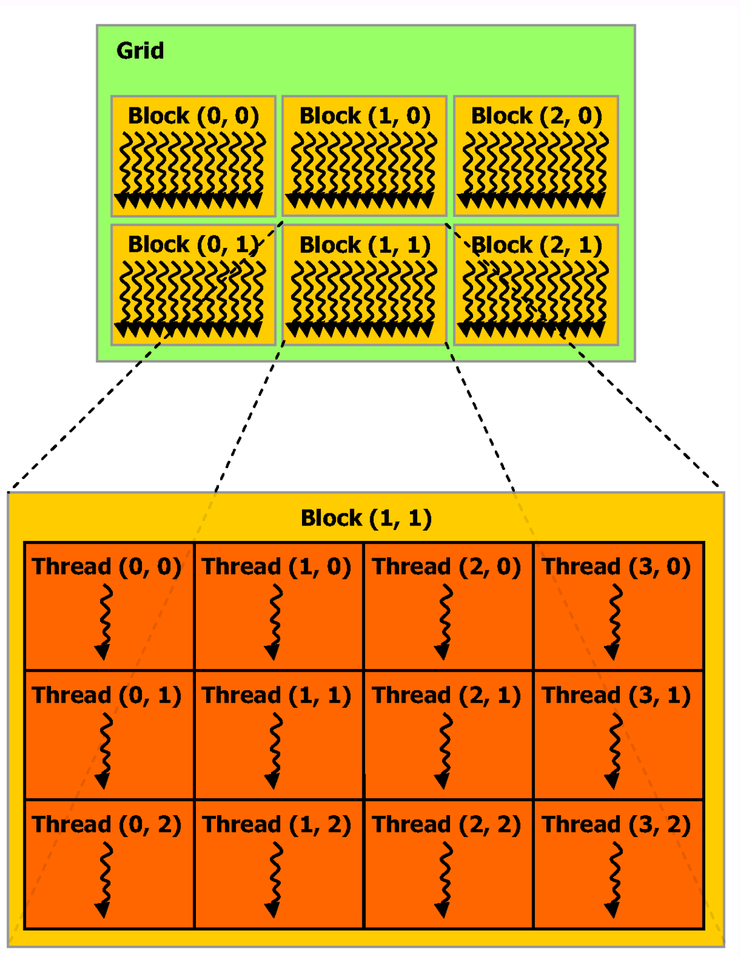


Figure 1: *Grid of Thread Blocks*

In CUDA, threads are grouped in **blocks**, and blocks constitutes a two-dimensional **grid**, which has been presented in Figure 1. The block size and number of threads per block affect multiprocessor occupancy, and their actual values are the programmer decision. **Warp** is a task scheduling unit, and it consists of 32 threads, each on a separate **lane**.

2. Implementation of Metaheuristics on CUDA

In our approach, parallelization has been performed on two levels: **In a block of threads, each thread transforms a separate individual or different gene; In each block, a separate experiment with different population is conducted**. It has been illustrated in Figure 2. If evaluation of the fitness function does not involve processing large amounts of data, essential data structures can be often stored entirely in the very fast shared memory (on-chip memory in Streaming Multiprocessors).

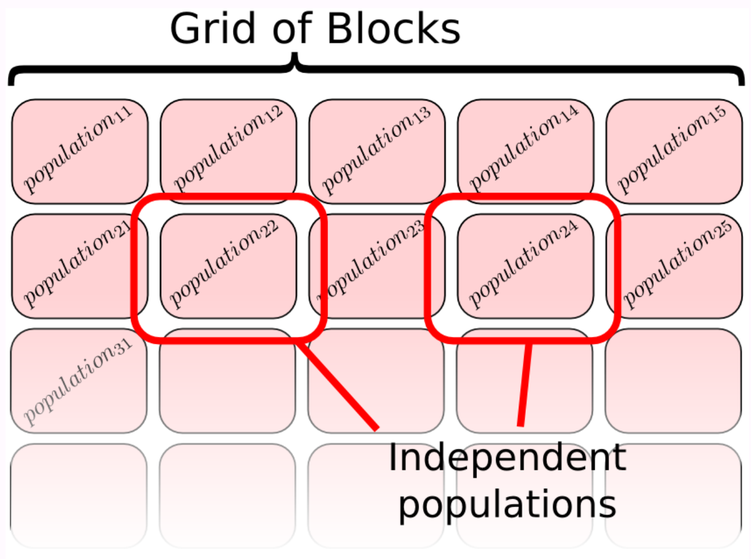
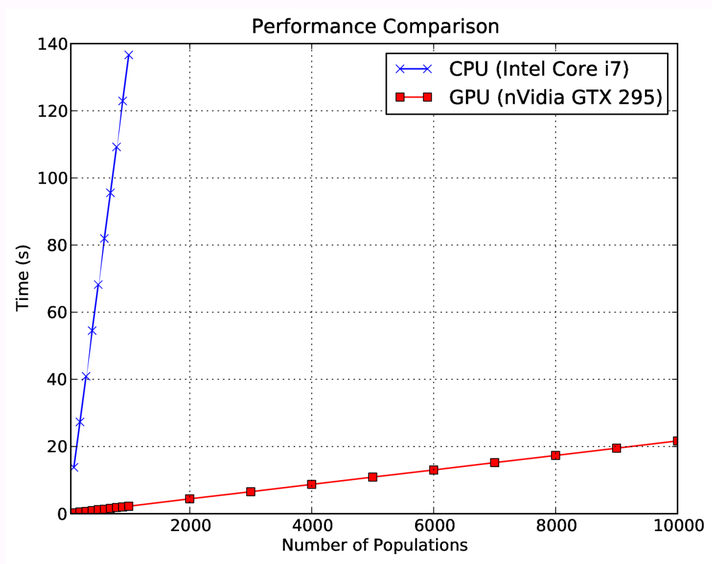


Figure 2: *Proposed approach to parallelization of the experimentation procedure*

3. Experimental Results



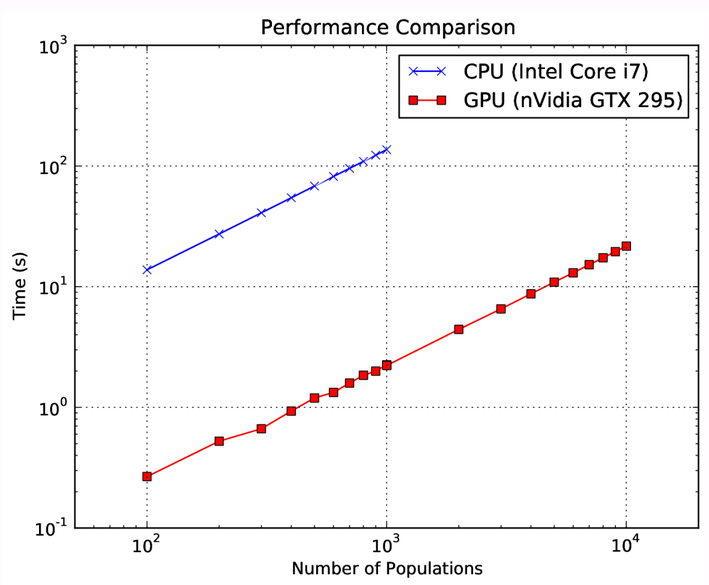


Figure 3: *Performance Comparison (Intel Core i7 2.93GHz CPU vs NVidia dual-GPU GTX 295)*

Firstly, Quantum-Inspired Genetic Algorithm has been implemented as a typical sequential program in ANSI C running on CPU for comparison. Secondly, implementation on NVidia GTX 295 in CUDA C has been created. Performance comparison has been presented in Figure 3. **The speedup gained on GTX-295 is about 120x in comparison to the sequential implementation**. The outcomes correctness has been verified in statistical analysis, which has been presented in Figure 4 and 5.

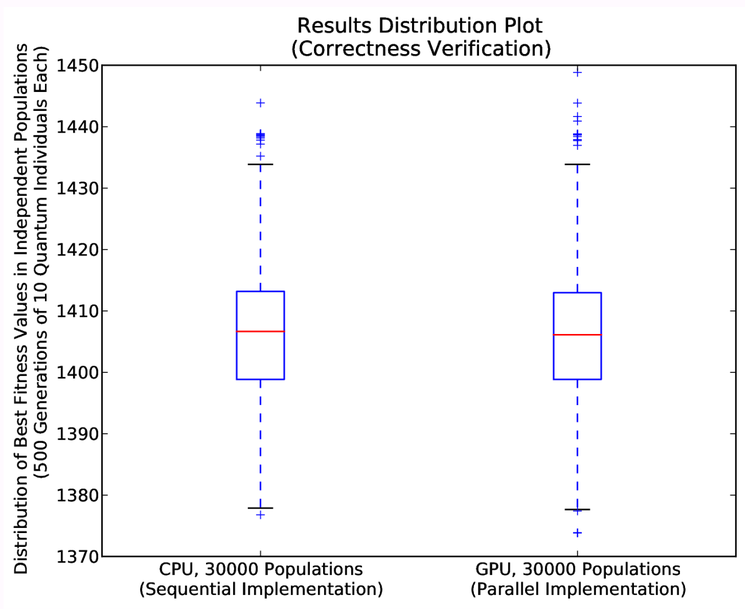


Figure 4: *Correctness verification.*

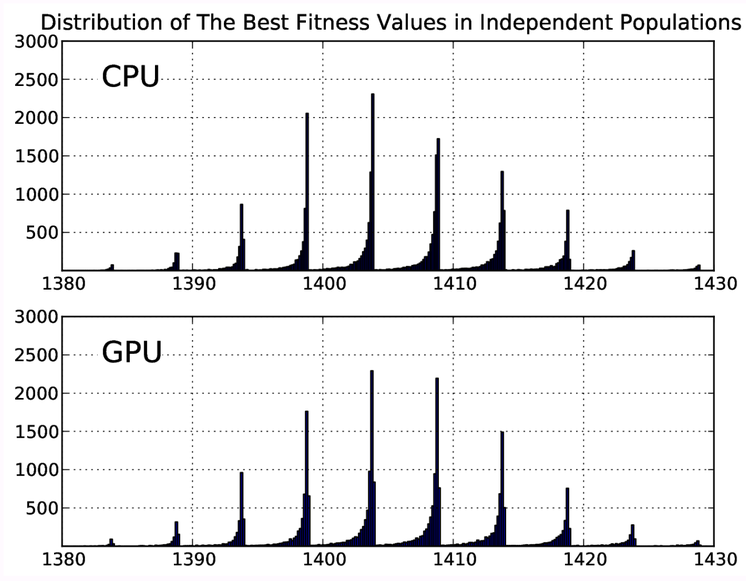


Figure 5: *Correctness verification (histogram of 30000 evolutions)*

Finally, an experiment has been conducted with distributed calculations on eight GPU devices (4 x Tesla T10 GPU, GTX 285, dual-GPU GTX 295 and Tesla C2070 GPU). **On this configuration, the speedup gained was over 400x. The total number of available streaming multiprocessors is very important in this approach.**

4. References

1. Han K.H., Kim J.H., Quantum-inspired evolutionary algorithm for a class of combinatorial optimization, Evolutionary Computation, IEEE Transactions on, No. 6 (2002), 580-593
2. **Nowotniak R.**, **Kucharski J.**, Building Blocks Propagation in Quantum-Inspired Genetic Algorithm, Scientific Bulletin of Academy of Science and Technology, Automatics, 2010, ISSN 1429-3447
3. Zhang G., Quantum-inspired evolutionary algorithms: a survey and empirical study, Journal of Heuristics, Springer, (2010), 1-49