Performance analyze to tune a parallel hybrid ant system

Omar Abdelkafi, Julien Lepagnot, and Lhassane Idoumghar

Université de Haute-Alsace (UHA)
LMIA (E.A. 3993)
4 rue des frères lumière, 68093 Mulhouse, France
{omar.abdelkafi,julien.lepagnot,lhassane.idoumghar}@uha.fr

Mots-Clefs. Parallel hybrid metaheuristics, TSP, GPU.

1 Introduction

The Graphics Processing Unit (GPU) became one of the main platforms to design massively parallel metaheuristics. Thanks to the advent of CUDA (Compute Unified Device Architecture), the use of GPU for non-graphic applications has become easier and many metaheuristics have taken advantage of this evolution. In CUDA programming, the execution on GPU is conducted by the kernel. It is a code called from the CPU (the host) and duplicated on the GPU (the device) to run in a parallel way. The kernel is executed in a grid, which is a set of blocks where every block is a set of threads.

Hybrid metaheuristics [1][2][3] are one of the most efficient classes of optimization algorithms. The idea is to combine metaheuristics [4] and other techniques for optimization. With the combination of different techniques, these methods can require a longer computation time than others. This is one of the reasons that lead the community to propose parallel hybrid metaheuristics [5] using platforms like GPU.

In this work, we present a performance analysis to enhance a parallel hybrid ant system (PH-AS) [6]. This algorithm is hybridized with a massively parallel local search (PLS) to found the best solution to the travelling salesman problem (TSP) by solving some instances problem from the TSPLIB.

2 Experimental analysis

The first test, in this work, was to analyze the impact of different initial populations. Two initial populations was tested: the parallel random initialization with the *curand* library and the parallel initialization with the *nearest neighborhood heuristic*. The parallel random initialization generates a random solution for each thread and the curand library provides different random values for different threads in parallel. The parallel initialization with the nearest neighborhood heuristic uses the index of the thread as the second city to visit and build the rest of the solution with the heuristic. It leads to different initial solutions. Our experimental results show that the parallel random generation can lead to better results with the curand library.

The second test is a performance comparisons between different neighborhood structure. The switch structure, the 2-opt structure and the combination of switch with 2-opt structure was tested in the PLS. Our experimental results show that the combination between the two structures (switch and 2-opt) leads to the best results.

For the experimental analysis of our hybrid, we use a NVIDIA GeForce GTX680 GPU. The experiment focus was on the execution time and the solution quality.

References

1. C. Blum, J. Puchinger, G.R. Raidl, A. Roli, Hybrid metaheuristics in combinatorial optimization: A survey, Applied Soft Computing, v.11 n.6, pp.4135-4151, (2011).
2. J. Lepagnot, L. Idoumghar, D. Fodorean: Hybrid Imperialist Competitive Algorithm with Simplex approach: Application to Electric Motor Design, 2013 IEEE International Conference on Systems Man and Cybernetics (SMC), Manchester UK, pp. 2454-2459, October (2013).
3. M.I Aouad, L. Idoumghar, R. Schott, O. Zendra, Sequential and Distributed Hybrid GA-SA Algorithms for Energy Optimization in Embedded Systems, the IADIS International Conference Applied Computing 2010, pp. 167-174, (2010).
4. C. Blum, A. Roli, Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison, ACM Computing Surveys, Vol.35, pp.268-308, (2003).
5. C. Cotta, E.G. Talbi, E. Alba, Parallel Hybrid Metaheuristics, in Parallel Metaheuristics: A New Class of Algorithms, John Wiley and Sons, (2005).
6. O. Abdelkafi, J. Lepagnot, L. Idoumghar, Multi-level parallelization for hybrid ACO, International Conference on Swarm Intelligence Based Optimization (ICSIBO), pp 77-84, (2014).