DISTRIBUTION AND PARALLELIZATION OF A GENETIC ALGORITHM

Sebastian Kurella, Justin MacPherson, Hannes Pfammatter, Valentin Anklin

Department of Computer Science
ETH Zürich
Zürich, Switzerland

1. PROBLEM

We have chosen to parallelize and distribute a genetic algorithm (GA). On first glance, genetic algorithms might appear embarrassingly parallel, as one could just independently evolve populations of the same problem and merge the result by taking the best individual among all the parallel instances. Although this does speed up the computation compared to a sequential implementation, it does not increase the diversity of the gene pool compared to the scenario where these subpopulations are allowed to mix.

As an exemplary genetic algorithm we plan to use the Travelling Salesman Problem (TSP) [1], however, our implementation of a scalable genetic algorithm should in principle be applicable to any genetic algorithm. As a benchmark for our implementation we plan to use the speedup compared to a sequential algorithm as well as an embarrassingly parallel implementation on the graph data set found at [2].

2. DISTRIBUTION APPROACH

As a model for the distributed genetic algorithm we choose an island model. We have two opportunities for parallelization: Within islands the individuals are separate and might be executed concurrently, however, for cross-over and selection operations these individuals need to be coordinated. Between islands the exchange of fit individuals, called migration, needs to be coordinated. Even though it is simpler to design and implement this process synchronously, we aim to use an asynchronous protocol for migration to maximize efficiency. We think the migration can even be done using Remote Direct Memory Access (RDMA), which should further improve efficiency.

3. RELATED WORK

The idea to parallelize GAs has been around for a long time [3] and [4], but more recent developments in the field such as MPI and RDMA have not been considered as far as we know. There are several newer approaches that use Hadoop to distribute a GA such as [5] and [6], however, these approaches use the file system to coordinate instead of asynchronous message passing which might slow down the implementation.

4. REFERENCES

[1] Michael Held and Richard M Karp, "The traveling-salesman problem and minimum spanning trees," *Operations Research*, vol. 18, no. 6, pp. 1138–1162, 1970.

[2] "Tsplib," Website, June 2018, http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/.

[3] Chrisila B. Pettey, Michael R. Leuze, and John J. Grefenstette, "A parallel genetic algorithm," in *Proceedings of the Second International Conference on Genetic Algorithms on Genetic Algorithms and Their Application*, Hillsdale, NJ, USA, 1987, pp. 155–161, L. Erlbaum Associates Inc.

[4] Shyh-Chang Lin, William F. Punch, and Erik D. Goodman, "Coarse-grain parallel genetic algorithms: categorization and new approach," *Proceedings of 1994 6th IEEE Symposium on Parallel and Distributed Processing*, pp. 28–37, 1994.

[5] Harun Rasit Er and Nadia Erdogan, "Parallel genetic algorithm to solve traveling salesman problem on mapreduce framework using hadoop cluster," *arXiv preprint arXiv:1401.6267*, 2014.

[6] Filomena Ferrucci, Tahar Kechadi, Pasquale Salza, and Federica Sarro, "A framework for genetic algorithms based on hadoop," *Computing Research Repository (CoRR)*, 11 2013.