

GPUs for Genetic and Evolutionary Computation

2010 Genetic and Evolutionary Computation Conference
Wednesday – Sunday July 7 –11, 2010 Portland, Oregon, USA



We are pleased to announce the official start of the GPU competition of GECCO-2010 with the publication of the competition rules and the scoring system.

The Goal

This competition focuses on the applications of genetic and evolutionary computation that can maximally exploit the parallelism provided by low-cost consumer graphical cards. The competition will award the best applications both in terms of degree of parallelism obtained, in terms of overall speed-up, and in terms of programming style.

Rules and Regulations

Entrants must submit (1) the application sources with the instructions to compile it and (2) a two page description of the application. Submissions will be reviewed by a committee of researchers from the evolutionary computation community and from industry. Each reviewer will score the submission according to 12 criteria concerning the submitted algorithm, the speed-up it achieves, and its impact on the evolutionary computation community. The total score will be obtained as the weighted sum of the 12 separate scores.

Submissions should be mailed to gecco2010@gpjpgpu.com no later than June 23rd, 2010. The final scores will be announced during GECCO.

Important Dates

Submission deadline: June 23rd 2010
Conference: July 7th-11th 2010

Organizers

Simon Harding, Memorial University of Newfoundland, Canada
David Luebke, NVIDIA
Pier Luca Lanzi, Politecnico di Milano
Edmondo Orlotti, NVIDIA
Antonino Tumeo, Pacific Northwest National Laboratory, USA



Sponsor of the GECCO-2010 competitions.



Scoring

Submissions will be reviewed by a panel of researchers from the evolutionary computation community and from industry who will score each submission according to the following criteria.

Algorithm (50% of the total score)

Novelty	10%	Does the algorithm exploit the GPU in a novel way? (e.g., not just for fitness evaluation?)
Efficiency	10%	Does the algorithm efficiently use the GPU?
GPU-side	10%	How much of the algorithm is implemented GPU side?
Elegance	5%	Is the algorithm simple, easy to understand?
Portability	5%	Is the code parameterized for different GPU architectures and/or across vendors?
Suitability	10%	Does it use features of the GPU architecture logically and to the advantage of the program?

Speed (20% of the total score)

Speedup	10%	How much is the speed up compared to a well coded CPU version?
Resources	5%	What is the resource utilization? (Ideally a program should use the 100% of the GPU).
Scalability	5%	Will it scale? E.g. to new hardware, multiple GPUs, GPUs with fewer/more processors?

Evolutionary Computation (30% of the total score)

Utility	10%	Do the results benefit the EC/GA/GP community?
Practicality	10%	Were the results practically obtainable without GPU acceleration?
Science	10%	Is the system used to generate better quality science? For example, increasing statistical significance, increasing coverage of test cases or demonstrating greater generalization.