Accelerating Scientific Model Optimization with a Pipelined FPGA-Based Differential Evolution Engine

Manuel de Castro[1], Roberto R. Osorio[2], Yuri Torres[3], and Diego R. Llanos[4]

[1] Departamento de Informática, Universidad de Valladolid, Valladolid, Spain, ORCID: 0000-0003-3080-5136

[2] CITIC, Computer Architecture Group, Universidade da Coruña, A Coruña, Spain, ORCID: 0000-0001-8768-2240

[3] Departamento de Informática, Universidad de Valladolid, Valladolid, Spain, ORCID: 0000-0002-3037-3567

[4] Departamento de Informática, Universidad de Valladolid, Valladolid, Spain, ORCID: 0000-0001-6240-9109

Differential Evolution (DE) [5] with Numerical Integration (NI) is an ideal target for Custom Computing Machines on FPGAs, since it produces deep pipelines, requires minimal external memory bandwidth, and benefits from large memory bandwidth. DE is a genetic algorithm used for scientific model optimization. For each candidate solution in a generation, it involves the following steps: (1) *Mutation*, where the solution produces a randomly-altered mutant vector; (2) *Crossover*, where some of the solution's values are exchanged for those of the mutant; (3) *Limiting*, where all the values are corrected to be in a valid range; and (4) *Selection*, where the new solution is tested against the original, with the best one surviving.

We propose a generic FPGA-based DE architecture, parameterized to accommodate to different scientific models. It supports both non-adaptive and adaptive NI methods. The core DE engine is programmed in VHDL for high adaptability and performance, whereas the scientific models and their NI are programmed in C++ for flexibility and easiness of development. It comprises the following modules: (1) *Control*, which initializes the state, manages the *memArrays*, and reads the output results; (2) *memArrays*, which store the model's parameter's values for each candidate solution, and the solution's computed cost; (3) *Crossover*, which hybridizes candidate solutions to produce a new one; (4) *Limits*, which manages out-of-range values from the previous model; (5) *Iteration Mux*, which manages the iterations and inputs of *Numerical Integration*; (6) *Numerical Integration*, which obtains model's results from the candidate solutions; and (7) *Error Evaluation*, which evaluates the obtained results and decides which candidate solutions survives. Our architecture achieves high performance by leveraging "Islands of Solutions" [1], [4].

We test our architecture using two particular applications: The Hodgking-Huxley model [2] with the RK4 NI, and the Circadian model [3] with the adaptive RKF NI. Both consist of multiple differential equations, with tens of parameters to solve. They comprise real-world applications, of higher complexity than previous works in the literature. Our architecture is able to implement them in a fully pipelined manner.

The compiler reports resource usages orders of magnitude higher than previous works, with the high-level-coded NI modules consuming the most part.

We perform an experimental evaluation, comparing our architecture with a sequential CPU reference, and a high-performance GPU implementation, running in an NVIDIA A100 GPU. When targeting a Virtex Ultrascale+ XCVU13P FPGA, our architecture achieves superior performance than CPU and GPU for both of the tested applications, with CPU-relative speedups of up to $\times 52.6$ for Hodgking-Huxley and $\times 41.67$ for Circadian (with the GPU achieving up to $\times 24.03$ and $\times 11.76$, respectively). Moreover, our proposal is, on average, $\times 9.0$ (Hodgkin-Huxley) and $\times 4.11$ (Circadian) more energy efficient than the GPU. Additionally, we observe that GPU and FPGA performance scale with the amount of solutions computed each generation, with FPGA scaling being more pronounced.

The poster associated to this extended abstract, presented at FCCM 2025, can be found on Researchgate.

Acknowledgments

This work was supported in part by Grant PID2022-142292NB-I00 (NATASHA Project); by grant TED2021–130367B–I00, funded by MCIN/AEI/10.13039/501100011033; and by MCIN/AEI/10.13039/501100011033 through the EU Grant PID2022-136435NB-I00. Manuel de Castro has been supported by a FPU 2022 grant. This research was supported by grants from NVIDIA and utilized NVIDIA A100. The authors would like to thank Sergio Alonso Pascual for his help developing the GPU software used.

References

[1] Alba, E., Luque, G., and Nesmachnow, S. Parallel metaheuristics: Recent advances and new trends. *International Transactions in Operational Research 20* (08 2012), 1–48.

[2] Hodgkin, A. L., and Huxley, A. F. A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of Physiology 117*, 4 (1952), 500–544.

[3] Locke, J., Millar, A., and Turner, M. Modelling genetic networks with noisy and varied experimental data: the circadian clock in Arabidopsis thaliana. *Journal of Theoretical Biology 234*, 3 (2005), 383–393.

[4] Penas, D., Banga, J., González, P., and Doallo, R. Enhanced parallel differential evolution algorithm for problems in computational systems biology. *Applied Soft Computing 33* (2015), 86–99.

[5] Storn, R., and Price, K. Differential evolution. A simple and efficient heuristic for global optimization over continuous spaces. *J. of Global Optimization 11*, 4 (1997), 341–359.