High-Performance Computing Group

Dr. Tobias Kenter

Computer Science Department and

Paderborn Center for Parallel Computing

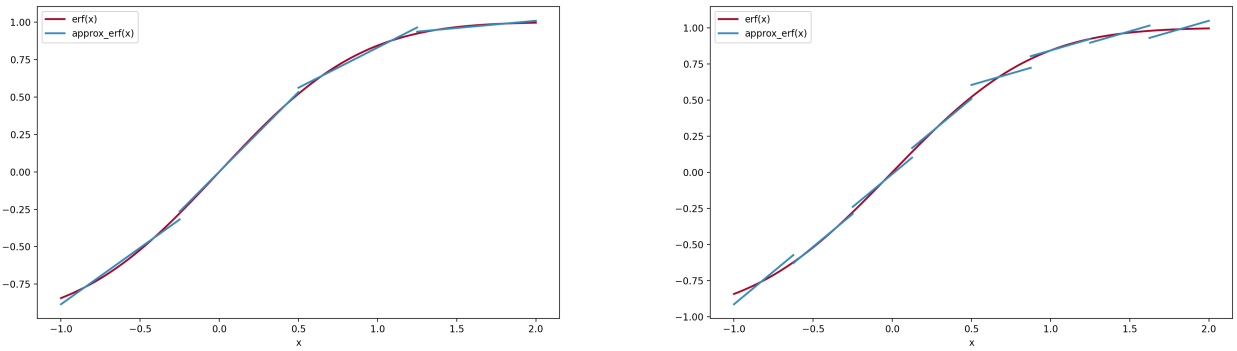
Systematic Function Approximation for FPGAs

Bachelor or Master Thesis

At a glance

- Research examples and best practices for function approximation
- Implement script or tool for automatic coefficient search and error metrics
- Use HLS tools to define cost model for building blocks
- Implement found approximation of selected functions like erf(x) on FPGAs

Scientific simulations often require the evaluation of trigonometric functions like sine and cosine, or of transcendental functions like exponential, logarithm or the error function. Current high-level synthesis tools for FPGAs generally support all commonly used functions, but their implementations can be either not accurate enough, or consume excessive amounts of resources. For example the built-in erf() function in the FPGA backend of Intel oneAPI was not accurate enough [1] for the Metalwalls software used for supercapacitor simulations.



Coarse approximation of the erf() function with (left) 4 segments with individual gradients or (right) 8 segments sharing 2 gradients

The goal of this thesis is to perform a systematic analysis of function approximation approaches and parameters that are suitable to approximate target functions with given accuracy requirements as efficient as possible. To this end, you will first look into previous research on this topic, some of which comes from the early years of FPGA computing [2]. After identifying the most promising approaches, you will build a tool that automatically calculates the fitting parameters for a given approximation scheme and evaluates the accuracy results. Then, after quantifying the cost of different components like coefficient storage, adders and multipliers with current tools on current FPGAs, you perform a design space exploration for optimal implementations. Finally, you can prototypically validate some implementations in hardware and compare them to the existing library versions.

Further reading:

[1] C. Prouveur, M. Haefele, T. Kenter, and N. Voss. **FPGA acceleration for HPC supercapacitor simulations**. In Proc. Platform for Advanced Scientific Computing Conf. (PASC), 2023. https://doi.org/10.1145/3592979.3593419

[2] J. Pineiro, J. Bruguera, and J. Muller. Faithful powering computation using table look-up and a fused accumulation tree. In Proc. IEEE Symp. on Computer Arithmetic (ARITH), pages 40–47, 2001. https://doi.org/10.1109/ARITH.2001.930102

Contact:

Tobias Kenter, Phone: 05251/60-4340 E-Mail: kenter@uni-paderborn.de

For information on theses offered by the HPC group or PC2 please visit:
https://en.cs.uni-paderborn.de/hpc/teaching/open-theses