Accelerating the MMD Algorithm

Michael Schlösser and Kenneth B. Kent
University of New Brunswick
Faculty of Computer Science
m09j8@unb.ca ken@unb.ca

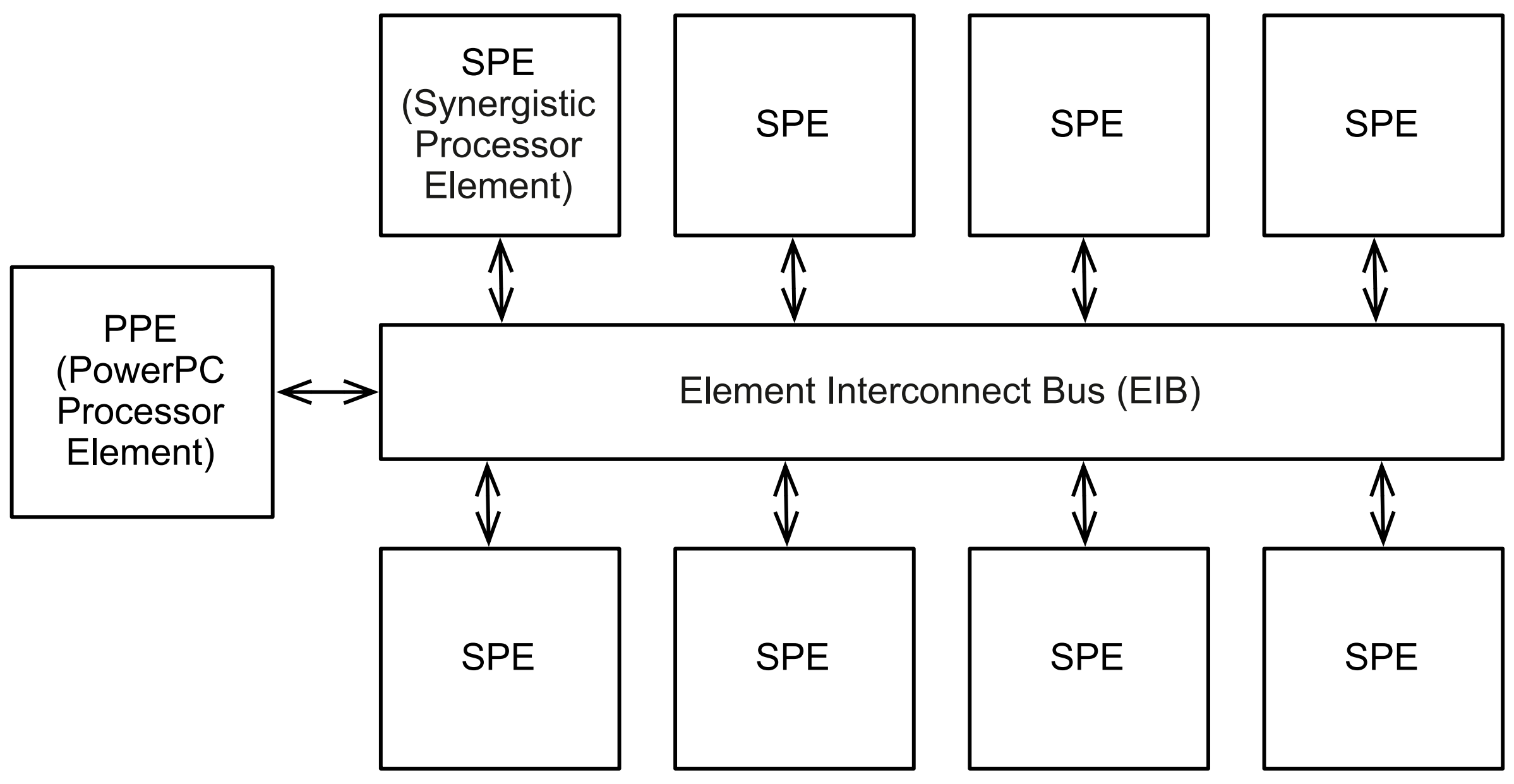
Motivation

The MMD algorithm is used to synthesize and optimize reversible logic networks. The runtime of the algorithm is strongly dependent on the size of the input network and can result in a runtime of several hours up to several days. Due to the time complexity, the algorithm uses heuristics in order to compute near optimal solutions.
The goal of this project is the acceleration of the algorithm. An accelerated version could be used to compute bigger inputs more efficiently or for more exact results. To achieve this goal the algorithm will be parallelized using both the Cell Broadband Engine (Cell BE) and an OpenCL based GPU implementation.

Materials

The Cell Broadband Engine (Cell BE)

The Cell Broadband Engine (Cell BE) is an implementation of the Cell Broadband Engine Architecture (CBEA), defined by IBM. It is based upon the 64-Bit PowerPC Architecture and extends it in order to provide a processor specification for parallel computing. For this project a Cell BE processor inside a Playstation 3 (PS3) is used.


SPE (Synergistic Processor Element)
SPE
SPE
SPE
PPE (PowerPC Processor Element)
Element Interconnect Bus (EIB)
SPE
SPE
SPE
SPE

The processor consists of one PowerPC Processor Element (PPE) which serves as program logic controller and several Synergistic Processor Elements (SPE) optimized for data-rich operations.

Graphics Processing Unit (GPU)

GPUs are highly parallel many-core processors commonly used in order to calculate real-time computer graphics. In the last couple of years they have been newly-discovered and utilized for computationally intensive general purpose tasks. Due to their massively parallel architecture they have the potential to outperform any common CPU currently available on the market.

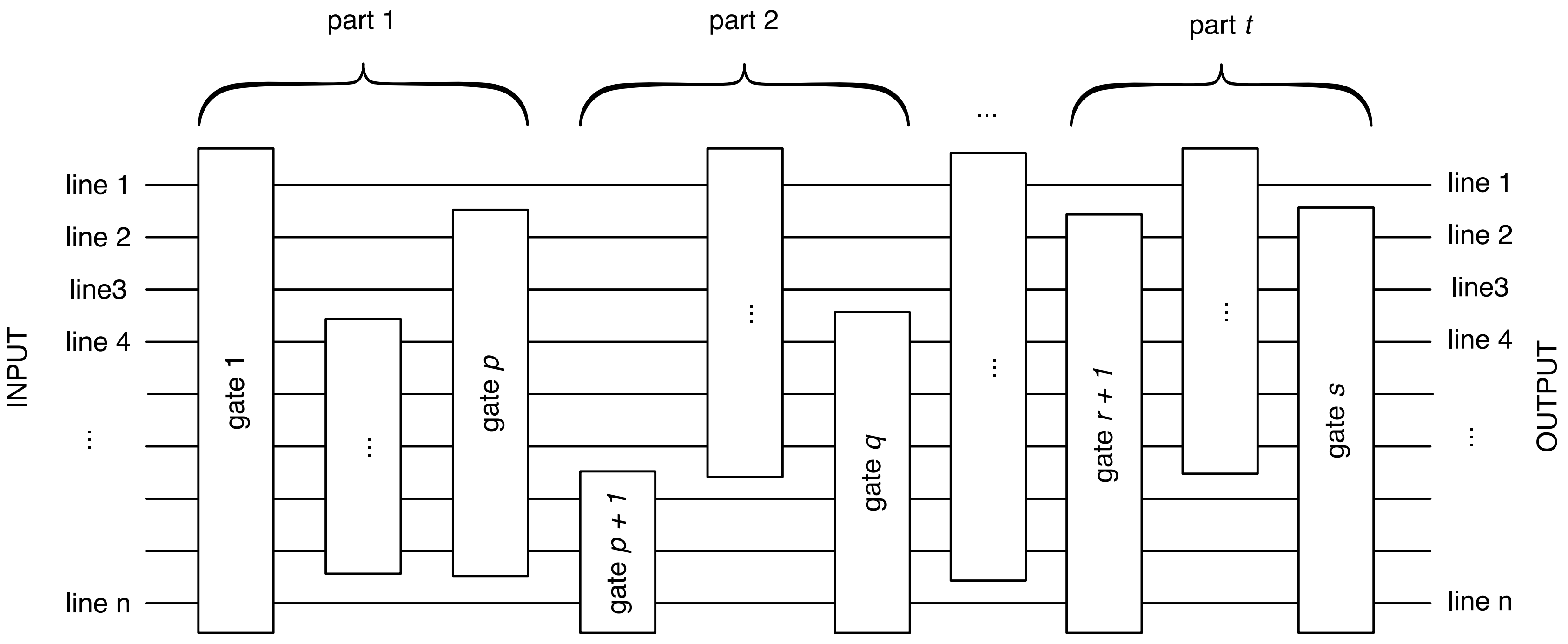
OpenCL

The Open Computing Language (OpenCL) is an open standard for hardware device agnostic parallelization of software applications. The standard is maintained by the Khronos Group which also is responsible for standards like OpenGL.
Any hardware architecture that is able to provide an implementation of the OpenCL specification is able to run OpenCL programs. The concept of writing a parallel program once and run it on heterogeneous devices like CPUs, GPUs or the Cell BE make OpenCL interesting for parallelization tasks.

Method

Both the Cell BE and GPUs are highly parallel hardware architectures. In order to be able to accelerate the MMD algorithm, goal of this research work is to exploit the underlying hardware as good as possible. This involves taking several hardware specific characteristics into account. These include e.g. number of parallel processing units or memory bandwidth and size, just to name a few.

The contemplated approach focuses on the aforementioned optimization step, the so called template matching. That means an already synthesized input network will be loaded and divided into *n* independent parts. The MMD algorithm will then be executed on every single part in parallel. In the end all *n* results will be collected and merged into a new and optimized reversible network.



Anticipated Results

With respect to the project goal, an acceleration of the algorithm can be expected. Especially for big input networks the parallel approach has several advantages that should lead to a shorter runtime of the algorithm. The current sequential approach only takes a small part of the whole circuit into account. That means only a small sequence of gates get optimized although the algorithm could work on several parts simultaneously. The parallel nature of this approach makes it possible to target the idling parts of the circuit and therefore deliver faster result.

References

D.M. Miller, D. Maslov, and G.W. Dueck. *A transformation based algorithm for reversible logic synthesis*. In Design Automation Conference, 2003. Proceedings, pages 318–323, June 2003.

IBM. *Cell Broadband Engine Architecture*. https://www-01.ibm.com/chips/techlib/techlib.nsf/ techdocs/1AEEE1270EA2776387257060006E61BA/$file/CBEA_v1.02_11Oct2007_pub.pdf (2009-18-12), October 2007.

Khronos OpenCL Working Group. *The OpenCL Specification.* http://www.khronos.org/registry/ cl/specs/opencl-1.0.48.pdf, June 2009.

REVLIB - *The Online Resource for Reversible Functions and Circuits.* Website: http://revlib.org/.

University of New Brunswick Faculty of Computer Science
Reconfigurable Computing Research Group