

# Theoretical and experimental results on Kilo/Mega Core Architectures

## A Summary of the results

Many core computers are the current approach to the solution of computationally intensive problems. At the present day, many core may mean 1000 but soon few 100000 or even more processing elements will be available on a desktop machine. The granularity of the processor cores can be different. For example on an FPGA “processors” can be logic processors (Configurable logic blocks), arithmetic processors (DSP blocks) and general purpose processor (e.g. Microblaze or ARM ) are in the same chip, computing together, forming heterogeneous many core system. In the mainstream products, FPGAs and GPUs, multi cellular architectures are present combining locally coupled processor and or memory arrays.

New ideas and new methods are required in algorithm development for these types of architectures. Using the cellular architecture of processor and memories new kind of parallel algorithms have to be developed For the same Physical multi-cellular machine architecture different Virtual Machines can be defined, the latter is serving for algorithmic and software development. However, all virtual machines are to be uniquely mapped to a specific Physical Machine (a chip or chip set, etc.). Hence all computing element has a geometrical position on the Physical Machine and a geometric topographic address on the Virtual Machine. In this new scenario of computer science and computer engineering computational complexity, computing power is a multi-parameter vector, any algorithm solving a problem will have a speed-power-area-bandwidth- accuracy metric. These parameters and this metric should be handled simultaneously and optimized.

During the last few years we considered two specific tasks we have developed new principles how to solve topographic and non-topographic problems in general on a kilo- or mega-core chip. Simulation of different physical phenomenon usually requires high performance computing facilities. Acceleration of the algorithms by specialized architectures is a key factor to increase the detail of the simulations. Fluid flow simulation was examined which requires the solution of a Partial Differential Equation. After discretization in time and space the structure of the computation on a grid is topographic, while generation and optimization of the arithmetic circuits is a non-topographic problem.

On the other hand, size and power limited tasks for UAV navigation are also examined which can be broken to topographic and non-topographic problems. Namely, detection of a specific target requires topographic processing capabilities and can be efficiently solved on a cellular architecture. While the tracking problem for collision avoidance of an UAV related to a bigger plane can be solved by a non-topographic algorithm. For example computation of an unscented nonlinear Kalman filter which is widely used for navigation and control of UAVs.

The new principles are summarized below.

First of all, we do not have efficient compilers though many examples are solved daily. In some of our problems we have used the CNN Universal Machine as an intermediate platform for compiling algorithms. Namely, we have developed an efficient CNN Universal Machine simulator on the FPGA or GPU and if an algorithm is described by as a cellular wave computing algorithm defined on a CNN Universal Machine, the compiling problem is solved.

As to the general principles, a few comments and results are important.

We emphasize that due to the topographic nature of the Physical machine, an optimal *partitioning and placement is also needed* on the virtual machine, or on the intermediate “compiler prototype” (like the CNN Universal Machine)

As to specific techniques and results the followings were found.

- (i) A Field Programmable Gate Array (FPGA) based framework has been described to accelerate simulation of a tough topographic problem, a nonlinear wave problem, that is a complex physical spatio-temporal phenomena, such as fluid dynamics (the hard test-problem since the invention of the digital computer). We have studied this problem since a while to develop a prototype virtual machine platform for them.
- (ii) In the course of the design of an arithmetic unit which consist of locally controlled groups of floating point units both *partitioning and placement* aspects have to be considered. To solve this problem a framework has been elaborated in which the partitioning is based on an initial floorplan of the vertices of the layered data-flow graph of the algorithm solving a given PDE or another problem. We have formulated this problem as a method for *implementing a data-flow graphs on FPGAs*. The algorithm can minimize the number of cut nets and guarantee deadlock-free partitions.
- (iii) In case of topographic problems, *complicated geometries require unstructured spatial discretization which results in irregular memory access patterns* severely limiting computing performance. *Data locality* is improved by mesh node renumbering technique which results in predictable memory access pattern. Additionally, *storing a small window of node data in the on-chip memory* of the FPGA can increase data reuse and *decrease memory bandwidth requirements*.
- (iv) As one example, in numerical solution of fluid flows the generation of the floating-point data path and control structure of the arithmetic unit containing dozens of operators is a very challenging task when the goal is high operating frequency. Efficiency and use of the framework is described by a case study solving the Euler equations on an unstructured mesh using finite volume technique. On the currently available largest FPGA the generated architecture contains three processing elements working in parallel providing two orders on magnitude (90 times) speedup compared to a high performance microprocessor.
- (v) Numerical algorithms can be implemented by using fixed –point or Floating point arithmetic with different precision. The selection of the *optimal data precision* to the different steps of an algorithm is *NP complete*. As an example, the solution of the advection equation was analyzed by using first and second order discretization methods. For this example if the resources are on an FPGA, and power dissipation or processing speed is prescribed, we have developed and algorithm for determining an optimal bit width.
- (vi) A Fermi NVIDIA GPU based implementation of a CNN Universal Machine was also developed, and was analyzed and compared to a multicore, multithreaded CPU. The basic goal of this project is to get a clear picture about the capabilities and limits of GPUs . In analysis of globally interacting structures GPGPU-s and FPGA-s were compared and the Kepler GPGPU-s showed some speed advantage over the largest Virtex .7 FPGA.
- (vii) Key algorithms are the different Kalman filter implementations in trajectory estimation of the intruder airplane. To increase the speed of computation of unscented nonlinear Kalman filter was implemented on Xilinx Spartan 6 LX45

FPGA. The speed of this implementation is approximately 1000 iteration /s. It would be important if we want to follow multiple targets.

We have developed new principles and techniques for implementing topographic and non-topographic problems on kilo- and mega-processor chips.

## References

Cs. Nemes, Z. Nagy and P.Szolgay, Automatic generation of locally controlled arithmetic unit via floorplan based partitioning, Proc. of CNNA 2012, 2012. Torino

A Kiss, Z. Nagy, Á. Csík and P.Szolgay, Examining the accuracy and the precision of PDEs on FPGA, Proc. of CNNA 2012, 2012. Torino

A.Hiba, Z. Nagy and M.Ruszinkó, „Memory access optimization for computations on unstructured meshes”, Proc. of IEEE CNNA 2012, 2012. Torino

E. László, P.Szolgay, Z. Nagy, „Analysis of a GPU based CNN Implementation”, Proc. of IEEE CNNA2012, 2012. Torino

Z. Nagy, Cs. Nemes, A. Hiba, A. Kiss, Á. Csik and P. Szolgay, „Accelerating unstructured finite volume solution of 2-D Euler equations on FPGAs”, Proc. of 15th International Conference on Fluid Flow Technologies, Budapest, 2012.

Z. Nagy, Cs. Nemes, A. Hiba, A. Kiss, Á. Csík, P. Szolgay, "FPGA Based acceleration of Computational Fluid Flow Simulation on Unstructured Mesh Geometry" Proceedings of FPL, Oslo, 2012.

Z.Nagy, A.Kiss, Á. Zarándy, T. Zsedrovits, B. Vanek, T. Péni, J. Bokor, T. Roska, ' Volume and power optimized high-performance system for UAV collision avoidance, Proc. of IEEE ISCAS 2012, Seoul, 2012.

Cs. Nemes, G Barcza, Z. Nagy, Ö. Legeza and P.Szolgay, Implementation trade-offs of the density matrix renormalization group algorithm on kilo-processor architectures, Proc. of ECCTD, 2013. Dresden

A Kiss, Z. Nagy, P.Szolgay, T. Roska, Gy. Csaba, X. S. Hu, W. Porod, FPGA-implementation of a Holographic Pattern-Matching Algorithm, Proc. of ECCTD, 2013. Dresden

Z. Nagy, C.Nemes, A.Hiba, Á Csík, A.Kiss, M.Ruszinkó and P. Szolgay, “Accelerating unstructured finite volume computations on fieldprogrammable gate arrays”, Concurrency and Computation:Practice and Experience, Vol. 26, no 3, pp 615-643, 2013.  
DOI:10.1002/cpe.3022

C.Nemes, G.Barcza, Z.Nagy, Ö. Legeza and P.Szolgay, "The density matrix renormalization group algorithm on kilo-processor architectures: implementation and trade-offs", Computer Physics Communications, 2014, DOI :10.1015/j.cpc.2014.02.021

A. Kiss, Z. Nagy P. Szolgay, T.Roska, Gy. Csaba, S. Hu, W. Porod, „FPGA-based simulation of 3D light Propagation”, Proc CNNA2014, Notre Dame, 2014.

A. Kiss, Z. Nagy P. Szolgay, T.Roska, Gy. Csaba, S. Hu, W. Porod, „Emulating optically inspired massively parallel on-Boolean operators on FPGA”, Proc CNNA2014, Notre Dame, 2014.

L.Tepelea V.Tiponut, P.Szolgay, A.Gacsádi” Multicore portable system for assisting visually impaired people”, Proc CNNA2014, Notre Dame, 2014.

B.J. Borbély, Z.Kincses, Zs.Vörösházi, Z. Nagy, P. Szolgay ” A Modular tsrt platform for real-time measurement and analysis of EMG signals for improved prosthesis control”, Proc CNNA2014, Notre Dame, 2014.

A. Hiba, Z. Nagy, M. Ruzinkó and P.Szolgay, "Data locality-based mesh partitioning methods for dataflow machines", Proc CNNA2014, Notre Dame, 2014.

E. László, M.B.Giles, J.Appleyard, P.Szolgay, "Methods to utilize SIMT and SIMD instruction level parallelism in triangular solvers", Proc CNNA2014, Notre Dame, 2014.