Session:

Proceedings of the 4th conference of the
Hellenic Society for Computational Biology & Bioinformatics - HSCBB09
NHRF, Athens, 18-20/12/2009

XX/12/2009

# Future Reconfigurable Architectures for Phylogenetic Inference

**Nikolaos Alachiotis and Alexandros Stamatakis**

The Exelixis Lab,Dept. of Computer Science, Technische Universität München, Germany

*Correspondence to: {alachiot,stamatak}@in.tum.de

## ABSTRACT

**Motivation:** FPGAs (Field Programmable Gate Arrays) have become more powerful in the last few years because of the availability of DSPs and block RAMs that allow for floating-point intensive scientific codes to be efficiently executed. The field of phylogenetic inference entails applications that are suitable for FPGA-based acceleration due to long execution times, like, for instance, the Phylogenetic Likelihood Function (PLF, Felsenstein (1981)) which is being used in Maximum Likelihood (Stamatakis, 2006) and Bayesian Inference programs. The 1st and 2nd generation of a dedicated computer architecture for the PLF have been presented in Alachiotis *et al.* (2009a) and Alachiotis *et al.* (2009b). Here, we provide an overview of the 3rd—significantly more flexible—generation of this architecture that is currently under development.

**Methods:** The 1st generation PLF architecture was only able to calculate plain likelihood scores and not the widely used log likelihood scores. In addition, only fully balanced binary trees were supported and the input was restricted to DNA data. Furthermore, all branch lengths had to be fixed and the alignment size was limited to a maximum of 512 taxa and 1,000 columns.

The 2nd generation (see Figure 1) of the PLF architecture provided support for arbitrary (unbalanced) tree topologies. The input sequences were stored in external memory and the PLF architecture was able to determine which sequences/ancestral probability vectors to retrieve from memory in order to perform likelihood computations according to the tree topology. Unfortunately, all other aforementioned restrictions still hold.

The 3rd generation of the PLF architecture, that is currently under development, will deploy the techniques developed for the 2nd generation to compute the PLF on arbitrary tree topologies, extended by support for so-called partial tree traversals. Different types of input data (morphological, DNA, protein) will also be supported. Moreover, a dedicated FPGA logarithm unit to calculate log-likelihood scores and a dedicated exponential unit that will allow for computing the PLF on trees with different branch lengths will be integrated. An advanced scaling technique as implemented in RAxML will be used to ensure numerical stability on tress with more than 512 taxa.

**Results:** The 1st and 2nd generation of our architecture yielded speedups between 3 and 14 compared to general purpose CPUs. Via a complete pipeline redesign, we expect to achieve significantly improved speedups with the more versatile 3rd generation.

**Discussion:** Advances in the field of reconfigurable computing allow for usage of FPGAs to speed up floating-point intensive Bioinformatics kernels such as the PLF by developing dedicated computer architectures.

## REFERENCES

Alachiotis, N., Sotiriades, E., Dollas, A., and Stamatakis, A. (2009b). Exploring FPGAs for Accelerating the Phylogenetic Likelihood Function. In *Proc. of HICOMB2009*.

Alachiotis, N., Stamatakis, A., Sotiriades, E., and Dollas, A. (2009a). A Reconfigurable Architecture for the Phylogenetic Likelihood Function. In *Proc. of FPL2009*.

Felsenstein, J. (1981). Evolutionary trees from DNA sequences: a maximum likelihood approach. *J. Mol. Evol.*, **17**, 368–376.

Stamatakis, A. (2006). RAxML-VI-HPC: maximum likelihood-based phylogenetic analyses with thousands of taxa and mixed models. *Bioinformatics*, **22**(21), 2688–2690.

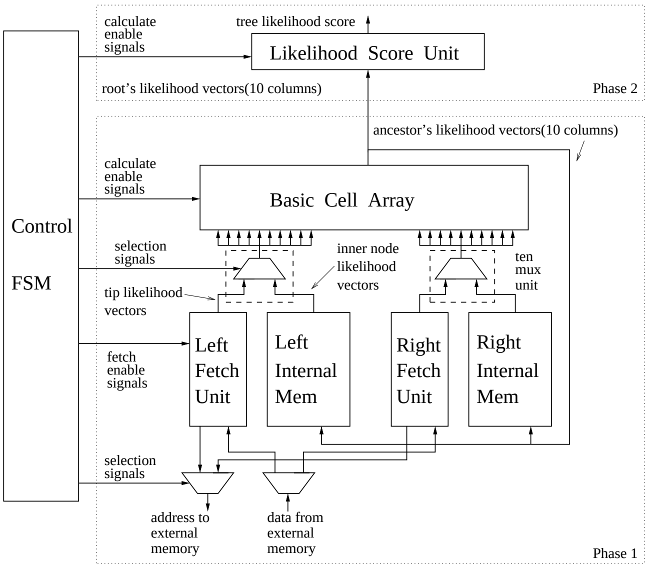

**Fig. 1.** Overview of the basic vector-like computational kernel of the 2nd generation PLF-architecture. The Basic Cell Array (BCA), that carries out the main bulk of the likelihood computations, performs up to 150 mathematical operations in parallel per clock cycle. The group of modules to the left and below the Basic Cell Array (BCA) are used to handle and traverse arbitrary (unbalanced) tree topologies. The units below the BCA are also used to access nucleotide sequences and ancestral probability vectors at internal or external memory addresses.