SYNTHESIS OF MASSIVELY PARALLEL ALGORITHMS & THEIR MAPPING ONTO RECONFIGURABLE SYSTEMS

Issam W. Damaj, and Ali E. Abdallah
School of Computing, Information Systems and Mathematics,
South Bank University, London

Key words to describe the work: Functional Specifications, Communicating Sequential Processes, Reconfigurable Computing, Parallel Algorithms, Wireless Communications Security, DNA Matching, Molecular Modelling.

Key Results: Development of systematic methods for massive parallelisation of functional algorithms and synthesis of efficient implementations on reconfigurable platforms, and its application to fast hardware implementations of computationally intensive problems.

How does the work advance the state-of-the-art? : First, the development of a generic method: applicable to a wide range of applications; Second, effective exploitation of the potential power of recently available reconfigurable circuits; Third, the correctness is ensured by construction.

Motivation (problems addressed): Emergence of relatively cheap massively parallel reconfigurable Hardware (e.g. Celoxica RC 1000). The pressing need for fast execution of computationally intensive applications. Advances in systematic methods for massive pipelining and data parallelisation of algorithms.

Introduction

Reconfigurable computers (RCs) offer the potential to greatly accelerate the execution of a wide variety of applications. Its key feature is the ability to perform computations in hardware to increase performance, while retaining much of the flexibility of a software solution. Field Programmable Gate Arrays (FPGAs), an instance of RCs, has recently enabled RC chips with 100 Million gates (Celoxica, Xilinix). Thus, affording more scalability and cost effectiveness due to hardware reuse. In addition, FPGAs offer much flexibility for the design of integrated circuits (ICs) chips.

With the emergence of such reconfigurable hardware chips, the presence of a rapid development environment for these scalable hardware circuits is very useful. Moreover, it would constitute the cornerstone solution for the ever-increasing need for more: efficiency, scalability and flexibility in realizing massively parallel algorithms for a wide area of applications. Namely, Wireless Communications Security, DNA Matching, Molecular Modelling, Digital Image Processing (DIP), Digital Signal Processing (DSP), Computer Graphics, and other applications.

Rapid Development Model

The rapid development model adopts the transformational programming approach for deriving massively parallel algorithms from functional specifications (See Figure 1). The functional notation is used for specifying algorithms and for the reasoning about them. This is usually done by carefully combining small number of high order functions (like *map*, *zip* and *fold*) to serve as the basic building blocks for writing high-level programs. The systematic methods for massive parallelisation of algorithms work by carefully composing "off the shelf" massively parallel implementation of each of the building blocks involved in the algorithm. The underlying parallelisation techniques are based on both pipelining and data parallelism. The emphasis in this method is on correctness, scalability and reusability.

To describe parallelism we use Hoare's CSP that allows issues of immense practical importance (such as data distribution, network topology, and locality of communications) to be carefully reasoned about. Relating the Functional Programming and CSP fields gives the ability to exploit a well-established FP programming paradigms and transformation techniques in order to develop efficient CSP processes.

The reconfigurable hardware realization step is done using Handel-C an automated compilation development model. Handel-C uses much of the syntax of conventional C with the addition of inherent parallelism. Handel-C relies on the parallel constructs in CSP to model concurrent hardware resources. Accordingly, any algorithm described with CSP could be implemented with Handle-C. For the desired hardware realization, Handel-C enables the integration with VHDL and EDIF along with various synthesis and place-and-route tools.

A part of the work already done in that field involves: Decomposition Strategise for Pipelined Parallelism, Combining Divide-and-Conquer and Function Decomposition, and other applications.

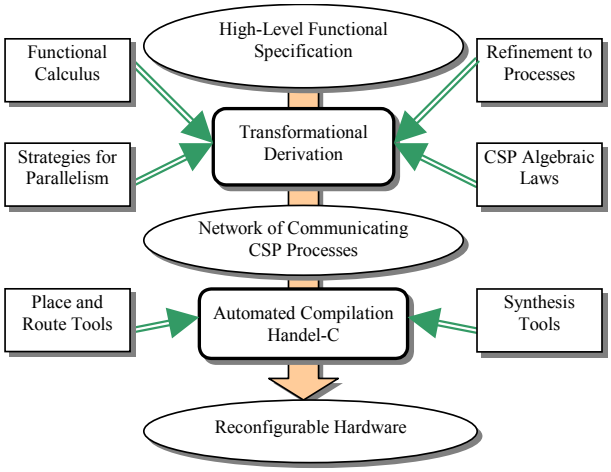

Fig 1. An overview of the transformational derivation, and the hardware realization processes.

Implementing Computationally Intensive Applications

The purpose of the current research is to systematically apply the proposed development method for a wide range of computationally intensive applications. These include Wireless Communications Security, DNA Matching, Molecular Modelling, and Dynamic XML Database Queries.

Wireless communications security is an area of crucial importance to telecommunications industry, where correctness and speed are the major concern. FPGAs seem to provide the ideal platform for implementing various complex security algorithms in hardware. Indeed, the design of integrated circuits (ICs) for the implementation of wireless security algorithms is the trend that seems to be driving wireless communications forward to an explosive growth making communications capabilities truly powerful.

In recent years, biologists are increasingly using state-of-the-art parallel supercomputers. This is particularly true in the field of biological research known as gene-sequence analysis, which involves matching DNA and RNA sequences obtained from sample biological tissue against sequences known to represent certain genes. Therefore, the introduction of reconfigurable computing to this area offers much needed speed for the realization of the solution to the problem.

A third computationally intensive application is interactive molecular modelling through parallelism. With the trials to solve many challenging problems in, for example, the design of drugs, chemicals and polymers and materials, and to understand the molecular problems relevant to combustion, atmospheric chemistry, chemical engineering of processes and environmental restoration, treatment of waste etc.

Conclusion

The ultimate goal of the current research is the development of a systematic method for rapid development and synthesis of massively parallel algorithms. Where, the targeted hardware is reconfigurable giving the advantages found in each phase of the development model. The addressed new areas of application are mainly Wireless Communications Security, DNA matching, and molecular modelling. The key issue in the proposed model is the production of engineering efficient, reusable, and correct solutions by construct as opposed to trial and testing.