Physical Design Automation for the Modern, Hybrid and Heterogeneous FPGA Architectures

Contents

1 Problem Statement and Significance

Field programmable Gate Arrays(FPGA) enable faster implementation and emulation of circuit designs on hardware for a wide variety of applications. The flexibility provided by FPGAs through the presence of reconfigurable elements has increased their popularity in comparison with the conventional ASIC designs. The impact of EDA tools is significant in the following areas: Low power VLSI Design, Technology Mapping, FPGA Placement and Routing and Verification and Testing. There is a need for improved procedures and algorithms in physical design automation using new techniques in the chip manufacturing process mainly in **Technology mapping** and **Placement and Routing**.

1.1 Technology Mapping Problem

Modern day FPGAs include heterogeneous resources such as Look-up Tables and/or reasonably big configurable Embedded Memory Blocks (EMB) to cater to the on-chip memory requirements of systems or applications mapped on them. While mapping applications on to such FPGAs, some of the EMBs may be left unused. We present a methodology to utilize such unused EMBs as large look-up tables to map multi-output combinational sub-circuits of the application, with area and delay minimization as the main objectives. Hybrid Field Programmable Architectures (HFPAs) are realized using a combination of the following programmable logic devices, namely, Lookup Tables (LUTs) and Programmable Logic Arrays (PLAs). The HFPAs provide the designers with the advantages of both LUT-based Field Programmable Gate Arrays and PLAs. Specifically, use of PLAs can result in reduction of area required for mapping a circuit. We present MEMMAP [1], which maps the circuits into heterogeneous FPGAs and SHAPER [2], which maps the circuits onto HFPAs, which focus on *area reduction* and *delay minimization* based on reconvergence in the circuits.

1.2 FPGA Placement and Routing Problem

Studies have shown that in the near future, there possibly exists a saturation of Moore's law. The only possible solution to this could be a paradigm shift in the fabrication process and major changes at the architectural level. With the advent of 3D-VLSI, Moore's law would still continue to hold good. In 3D-FPGAs, the average Manhattan distance between any two elements in 3D-FPGA is less when compared to that in 2D-FPGA, due to the vertical stacking of the silicon layers. FPGA Placement and Routing Problem is defined as follows: Given a synthesized net-list of a circuit which consists of the logical elements and the interconnections between them, compute,

1. a mapping of the different logical elements in the net-list on to the configurable logic blocks of the FPGA (Placement); and,
2. a mapping of the interconnects in the net-list on to the routing resources of the FPGA such that, the connectivity between various logical elements as specified in the net-list is maintained (Routing).

such that, the total interconnect length and the channel width (maximum number of interconnects incident on every side/face of a switch block in the FPGA) are minimized. We present a placement and routing tool [3], for 3D-FPGA using **Reinforcement Learning** and **Support Vector Machine**(SVM) techniques.

2 Related Work and Limitations

2.1 Technology Mapping for Hybrid and Heterogeneous FPGAs

The focus of different technology mapping algorithms proposed is based on various optimization criteria: *performance, area, delay and routability*. Architectures involving multi-output LUTs and heterogeneous LUTs have been studied and technology mapping algorithms for such architectures have also been proposed. Most such algorithms are targeted to lookup-tables with two-outputs. The mapping techniques used in these algorithms are not suitable for arrays with eight or more outputs. However, in the existing algorithms for heterogeneous FPGAs [4, 5, 6], the focus was either to achieve area minimization or delay optimization but not both. Algorithms proposed for mapping circuits into LUT based homogeneous FPGAs are not immediately extensible to heterogeneous and hybrid FPGAs because of the presence of multiple logic resources. We have proposed mapping algorithms which aims for both area and delay reduction using reconvergence analysis [1]. The issue of mapping logic into PLAs has also been studied before. But, the algorithms used for mapping circuits to CPLDs which contain PLAs alone, cannot be used to map to HFPAs. In the previously existing algorithms [7, 8, 9], the whole circuit is searched exhaustively for identifying regions that are amenable for mapping to PLAs. However, our proposed methodology [2] is based on the intuition that the reconvergence in a circuit creates regions with large number of logic nodes, less inputs and outputs. We believe that these regions are ideal for mapping into PLAs.

2.2 3D-FPGA Placement and Routing

Two-dimensional FPGA architectures are well studied and reported in the literature and there are many algorithms available for Placement and Routing for 2D-FPGA. On the other hand, though roots of 3D-FPGA started way back in 1996, not much has been achieved in practical aspects. Only few architectural designs and Placement and Routing algorithms have been proposed for 3D-FPGAs [10, 11, 12, 13]. We have used RL and SVM for solving the 3D-FPGA problem [3]. Though Reinforcement Learning concepts are applied in various problem solving techniques [14, 15], to the best of our knowledge, [14] is the only instance reported in the literature, where, RL is used to solve a problem in the area of VLSI. Our proposed methodology is one of the first applications to use SVM in conjunction with a batch RL algorithm to solve the FPGA placement and Routing problem.

3 Technical Approach and Results Achieved

3.1 Technology Mapping for Heterogeneous and Hybrid FPGAs for Area Reduction using Reconvergence Analysis

The primary focus of this work is to check the effectiveness of using reconvergence in the circuit as the basis for mapping circuits to FPGAs containing multiple logic resources. We consider two architectures, the heterogeneous one with LUTs and embedded memory blocks and the hybrid one with LUTs and PLAs. This work presents two automated technology mapping tools, MEMMAP and SHAPER, that automatically map circuits to heterogeneous and hybrid FPGAs respectively so that an area and delay optimized solution

is achieved. There are four phases in the algorithm: *Preprocessing, Reconvergence Analysis, Memory or PLA Mapping and LUT Mapping*. The overall strategy of both algorithms is to first extract the reconvergent sub-circuits that can be mapped into the non-LUT resources. Each sub-circuit thus obtained is expanded using appropriate heuristic until there exists scope for expansion. The residual circuit is mapped onto the LUTs using DAG-Map algorithm. To illustrate the effectiveness of our approach, the mapping tools have been used to map a set of circuits from various benchmark suits including MCNC, ISCAS85 and ISCAS89 and the results have been compared with other mapping tools reported in literature. MEMMAP which is the heterogeneous FPGA mapper deleted 51% LUTs when compared to SMAP, an algorithm previously reported for the problem of heterogeneous technology mapping. The Hybrid Technology Mapping Algorithm, SHAPER when compared with other methodologies resulted in further 18% decrease in number of LUTs. For a detailed description of the algorithm and results please refer our papers which are already published [1, 2].

3.2 Placement and Routing for 3D-FPGAs using Reinforcement Learning and Support Vector Machines

The main idea is to employ RL to improve the performance of local search algorithms. Local search algorithms start with some initial random solution to the problem and progressively improve on this solution by searching in the neighborhood of the current solution for a better solution. The quality of the final solution found depends on the initial solution. Typically the local search algorithm is run multiple times and the best among the different solutions produced is retained. We improve upon the performance of a given local search algorithm by employing RL to learn a good starting point for the local search process, one that leads to a good final solution. Hence, finding a state with high value function would lead to a good final solution. In this work, we choose to employ regression with *support vector machines* (SVMs). SVMs are a popular tool for solving non-linear classification and regression problems. They work on the principle that if a non-linear classification/regression problem is transformed to a high dimensional space, then it is more likely to be linear in that space. In [3], we use a Gaussian kernel based SVM for the training. Using RL and SVM 28% reduction in wirelength and 7% reduction in channel width is acheived when compared to the Mondrian approach [12].

4 Contributions of the Research

4.1 Technology Mapping

Our proposed technology mapping algorithms are used to map LUT based FPGA architectures containing embedded memory blocks(MEMMAP) and programmable logic arrays(SHAPER). For the first time, the concept of reconvergence is used in the field of FPGA mapping and is shown to be effective. Some salient features of the proposed approach are:

1. Since most of the reconvergent regions are mapped onto non-LUT resources, the circuit to be mapped into LUTs contains less reconvergent regions. The DAG-Map algorithm used for mapping into LUTs

is more efficient when reconvergence in circuit is less. DAG-Map algorithm is optimal when the initial network is a tree or a set of trees [6].

2. All potential regions that can be mapped into non-LUT resources are obtained in a single phase. Hence the same algorithm can be used for mapping to both single and multiple resources without adding extra complexity.

3. The second phase of the algorithm exploits the flexibility for reconfiguration provided by embedded memory blocks. This is done by expanding a given overlapping reconvergent region to meet constraints of the nearest configuration of the memory block or PLA.

4.2 3D-FPGA Placement and Routing using RL and SVM

1. The proposed solution is one of the very few instances, wherein, RL is used for solving a problem in the area of VLSI.

2. The learning process models the common features among the different instances of the placement and routing problem on FPGA and hence is instance independent. In other words, learning done on one set of representative benchmark circuits is used to place and route the remaining benchmark circuits effectively. The approach is made instance independent by virtue of the chosen set of features.

3. This is one of the first applications to use SVM in conjunction with a batch RL algorithm.

5 Further Work

1. **Technology Mapping:** Performing power estimation to see how much of a power reduction could be achieved by mapping circuits into embedded memory blocks or PLA blocks is an interesting open problem.

2. **3D-FPGA Placement and Routing:** Further effort can be directed towards the design of better feature set leading to greater improvement in performance. Another interesting direction of research is to do power analysis and incorporate thermal and fabrication issues in our cost function leading to power aware placement and routing.

References

[1] B. Jayaram, A. Manojkumar, **R. Manimegalai**, V. Kamakoti, "MemMap-pd: Performance Driven Technology Mapping Algorithm for FPGAs with Embedded Memory Arrays",*Proceedings of Reconfigurable Architectures Workshop RAW 2004*, in New Mexico, USA. pp: 152

[2] **R. Manimegalai**, B. Jayaram, A. Manojkumkar, V. Kamakoti, "SHAPER:Synthesis for Hybrid FPGAs containing PLAs using Reconvergence Analysis", *to appear in Proceedings of International Conference on Field Programmbale Technology -2004*, in Australia.

[3] **R. Manimegalai**, E. Siva Soumya, V. Muralidharan , B. Ravindran, D.Bhatia ,V. Kamakoti, "Placement and Routing for 3D-FPGAs using Reinforcement Learning and Support Vector Machines", *to appear in International Conference on VLSI Design 2005*, Kolkotta, India.

[4] R.Murgai, Y.Nishizaki, N.Shenoy, R.K.Brayton and A.Sangiovanni-Vincentelli, "Logic synthesis algorithms for table look up programmable gate arrays," In *Proceedings ACM/IEEE Design Automation Conference* pp. 620-625, 1990.

[5] S. J. E. Wilton, "SMAP: Heterogeneous technology mapping for FPGA's with embedded memory arrays," In *Proceedings ACM/SIGDA International Symposium Field-Programmable Gate arrays*, pp. 171-178, February 1998.

[6] K.C.Chen, J.Cong, Y.Ding, A.B.Kahng and P.Trajmar, "DAG-Map: Graph-based FPGA technology mapping for delay optimization," In *IEEE transactions on Design and Test of Computers*, 9(3):7-20, September 1992.

[7] A.Kaviani and S.Brown, "Technology Mapping Issue for an FPGA with Lookup Tables and PLA-like Blocks," In *ACM/SIGDA 8th International Symposium on Field-Programmable Gate Arrays*, pp. 60-66, February 2000.

[8] K.Yan, "Practical Logic Synthesis for CPLDs and FPGAs with PLA-style Logic Blocks," In *Proceedings of the conference on Asia South Pacific Design Automation Conference*, pp-231-234, January 2001.

[9] S.Krishnamoorthy and R.Tessier, "Technology Mapping Algorithms for Hybrid FPGAs containing Lookup-Tables and PLAs," In *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 22(5): 545-559, May 2003.

[10] S. M. Alam, D. E. Troxel and C. V. Thompson, "A Comprehensive Layout Methodology and Layout Specific Circuit Analyses for Three-Dimensional Integrated Circuits", *ISQED 2002*, pp.246-251.

[11] W. M. Meleis, M. Leeser, P. Zavracky, M. M. Vai, "Rothko:Architectural Design of a Three Dimensional FPGA", *IEEE Proceedings of 17th Conference on Advanced Research in VLSI (ARVLSI '97)*, September, 1997, Ann Arbor, MI, pp. 16-23.

[12] M. Alexander, J.P. Cohoon, J. Karro, E.L. Peters, and G. Robins, "Placement and Routing for Three-Dimensional FPGAs",*In Fourth Canadian Workshop on Field-Programmable Devices*, Toronto, Canada, May 1996, pp. 11-18.

[13] J. Karro and J. Cohoon, "A Spiffy Tool for the Simultaneous Placement and Global Routing for Three-Dimensional Field Programmable Gate Arrays," *In Ninth Great Lakes Symposium on VLSI, Ann Arbor, Michigan*, March 1999, pp. 226-227.

[14] J. A. Boyan, and A. W. Moore , "Learning Evaluation Functions for Global Optimization and Boolean Satisfiability", *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, 1998, pp. 3-10.

[15] R. Moll, A. G. Barto, T. J. Perkins, and R. S. Sutton, "Learning Instance-Independent Value Functions to Enhance Local Search", *Advances in Neural Information Processing Systems Conference*, Denver, Colorado, The MIT Press 1999, pp.1017-1023.