

Multiple Circuit Partitioning Implementation Using Primal Dual Approach

Ravinder Singh, Sandeep Singh, Ramandeep Singh

Abstract—The study presents a number of extensions and applications of hyper graph-partitioning algorithms for solving some of the emerging, as well as the existing complex problems in the physical-design-automation phase of electronic circuit fabrication. It is used to encounter the key challenges like (i) devising scalable algorithms to handle exponentially increasing design sizes (ii) devising algorithms to solve new problems that arise because of the evolving architectures of pre-fabricated chips, such as FPGAs. A multi-commodity flow treatment is proposed for partitioning without a size constraint, and an iterative improvement algorithm is proposed for partitioning with a size constraint.

Index Terms— Hyper graph partitioning algorithm, FPGA, Multi-pin net model, Multi hyper graph partitioning algorithm.

I. INTRODUCTION

Modern Electronic Design Automation (EDA) algorithms are increasingly becoming more complex for a number of reasons; (i) exponential increase in the size of designs implemented (dictated by Moore's law), (ii) complex deep sub-micron effects that emerge because of shrinking process geometries that require active consideration for accurate operation of the circuit, and (iii) heterogeneity of circuit elements used in modern chips. These changing requirements force us to devise new algorithmic solutions to effectively solve newly-emerging complex problems. The EDA is all about automating the tasks involved in fabricating electronic circuits on silicon.

Modern circuit designers specify their circuits in Hardware Description Languages (HDL) such as VHDL or Verilog. Once the specifications are complete, they are synthesized into gate-level netlist (circuit), which contains gates (cells) such as AND gates, OR gates and so on, and the wires (nets) that interconnect gates. Chen and Cheng (2000), in the paper entitled "Tutorial on VLSI Partitioning", introduced the partitioning with applications to VLSI circuit Designs. Zhao and Zhao (2002), in the paper entitled "An Effective Algorithm for Multiway Hypergraph Partitioning", proposed an effective multiway partitioning algorithm. ADYA et al. (2003), in the paper entitled "Benchmarking for

Large-Scale Placement and Beyond", reviewed motivations for benchmarking, especially for commercial EDA, analyzed available benchmarks, and pointed out major pitfalls in benchmarking. Alpert et al. (2010), in the lecture on topic "Primal Dual Algorithms", presents the use of Primal-Dual (PD) method to obtain a good approximation algorithm. Karypis and Kumar (2011), in the paper entitled "Recent Directions in Netlist Partitioning: A Survey", described research directions in netlist partitioning during the past two decades, in terms of both problem formulations and solution approaches.

The graph/hypergraph partitioning algorithms are fundamental tools in EDA that enable a divide-and-conquer strategy to cope with ever-increasing design sizes. The partitioning algorithms are most commonly used during synthesis and placement. Hypergraph partitioning is an important problem with extensive applications to many areas, including VLSI design, efficient storage of large databases on disks, and information retrieval. Hauck and Borriello (2013), in the paper entitled "Design and Implementation of Move-Based Heuristics for VLSI Hypergraph Partitioning", summarized the techniques of implementing move-based hypergraph partitioning heuristics and evaluated their performance in the context of VLSI design applications.

The problem is to partition the vertices of a hypergraph into k equal-size sub-domains, such that the number of the hyperedges connecting vertices in different sub-domains (called the *cut*) is minimized. Note that the hypergraph-partitioning algorithms, as opposed to graph partitioning algorithms, are predominantly used in EDA, because of their ability to natively and accurately represent the underlying circuit. The importance of the problem has attracted a considerable amount of research interest, and over the last thirty years a variety of heuristic algorithms have been developed that offer different cost-quality trade-offs.

II. METHODS

- A. Study of input netlist files, benchmark files and choosing the adequate benchmark files that will be used as input to the partitioning algorithms.
- B. Study of various partitioning techniques & various clustering methods etc.
- C. Study & Implementation of FM algorithm. Verifying the FM Algorithm using Benchmark files.
- D. Study & implementation of multiway partitioning algorithm using the concepts of iterative and primal algorithms. Verifying the results using benchmark files.
- E. Compare the output files created by the two methods.

Manuscript received October 23, 2014.

Ravinder Singh, Department of Electronics and Communication Engineering, Guru Nanak Dev Engineering College, Ludhiana, 141002, India

Sandeep Singh, Department of Electronics and Communication Engineering, Guru Nanak Dev Engineering College, Ludhiana, 141002, India

Ramandeep Singh, Department of Electronics and Communication Engineering, Guru Nanak Dev Engineering College, Ludhiana, 141002, India.

A. PROBLEM FORMULATION

The Literature Survey reveals that a lot of work has been done in the field of partitioning algorithms. As Partitioning plays an important role in solving large, complicated design problems. Given a system, partitioning divides the whole circuit from the system level to the board level, from the board level to the chip level, and from the chip level to the macro-cell level. At each level the circuits are further divided into smaller sub circuits.

B. OBJECTIVE

The objective of the work is to implement first implement the FM algorithm in C/C++, verify it using a benchmark netlist file or a netlist file generated from a VHDL/Verilog code.

- Implementation of **Fiduccia-Mattheyses (FM) Heuristic** for VLSI Netlist partitioning.
- Reduce the nets with higher cardinality).
- Compare their benefits and drawbacks.
- Implementation of multiway partitioning technique like Primal dual technique, iterative FM etc. (Multiple objectives: 1. Reduce the net cut, 2. Reduce the span of the nets, 3. Reduce the nets with higher cardinality).

The motivation of this thesis is to use C/C++, **GCC compiler** and **benchmark standards / VHDL netlists** to realize and analyze FM and multiway partitioning algorithms. The following steps are taken to realize the objective

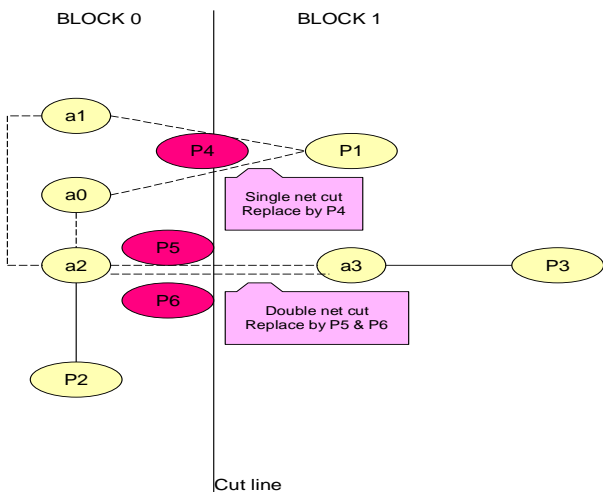


Fig. 1: Partitioning Representation

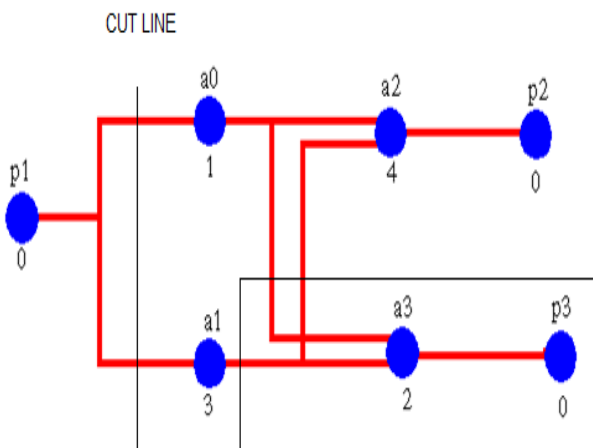


Fig. 2: Netlist Graph Representation with Cutline

C. UNIFORM MULTI-PIN NET MODEL

The multi-pin net model has always been a major concern of the KL based algorithms. A good multi-pin net model can correctly reflect the immediate gain of a move as well as the potential gain of a move. The probability of choosing the best move is thus increased.

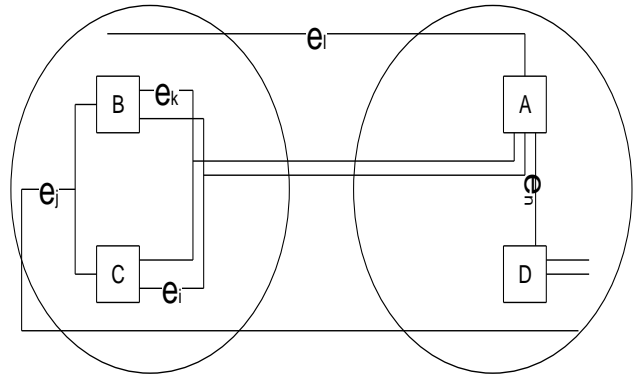


Fig. 3: Multi-Pin Net Model

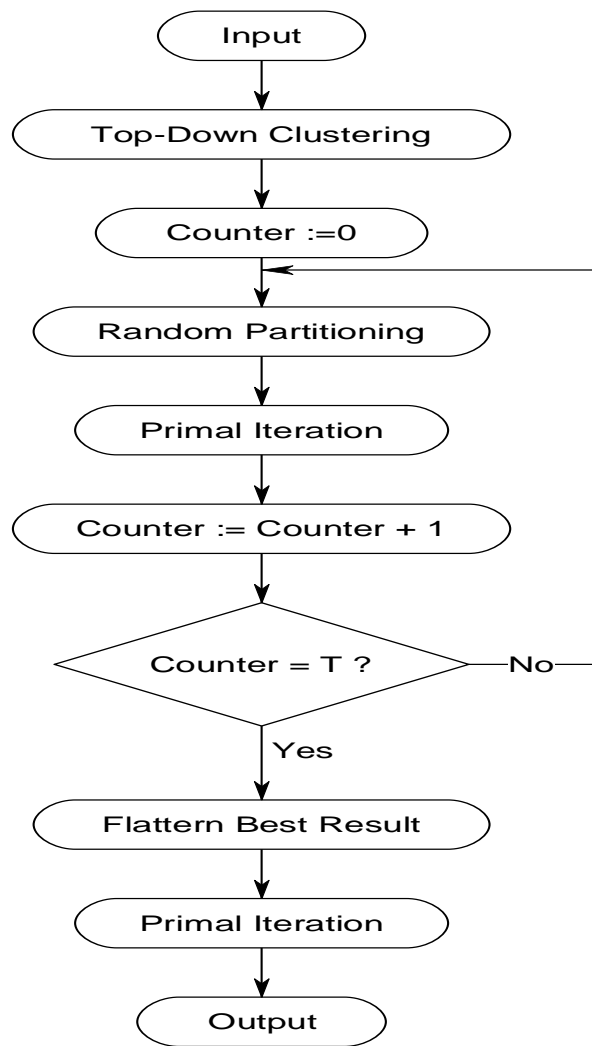


Fig.4: Primal Process-Flow

D. COMPARISON

FM algorithm is primarily a bi-partitioning technique; it is capable of dividing a single netlist file into two equally sized partitions. With the advancements in the technology we are coming up with more & more complex IC's day by day, this creates bigger circuits which in hand requires the circuit to be partitioned into more than two partitions; this is not possible with the core FM technique, but as FM algorithm itself is very stable so it can be a good stable platform for other multiway partitioning algorithms.

III. RESULTS AND DISCUSSION

A number of benchmark circuits like IC67.net, IC116.net, IC157.net, IBM0.net, IBM1.net and one randomly generated circuit form a VHDL file Test_kp.net were used to compare the various algorithms. The algorithms were implemented in c/c++ and run on a dual core Turion machine. The algorithms were run for atleast 40 times for each netlist file and there averages were tabulated (the objective considered was: minimize the connections between the partitions/blocks).

Table.1: Comparison of Traditional (FM) and Primal-Dual algorithm

Level 0 Partitioning			
Netlist File	Traditional Algorithm	Primal Dual Approach	Improvement %
IC 67	38	38	0
IBM0	574	573	.174
Sample	3	3	0
Level 1 Partitioning			
Netlist File	Traditional Algorithm	Primal Dual Approach	Improvement %
IC 67	48	42	12.5
IBM0	646	530	17.9
Sample	4	2	50
Level 2 Partitioning			
Netlist File	Traditional Algorithm	Primal Dual Approach	Improvement %
IC 67	44	31	29.54
IBM0	704	524	25.56
Sample	3	2	33.33

The data in the Table 1 show much greater improvement for primal approach then for the traditional FM algorithm. The algorithms were run for 2 partitions, 4 partitions, 8 partitions (shown as 0 level, 1 level, 2 level partitions). The average improvements are .058%, 26.8%, 29.43% respectively. For zero level partition all the algorithms show identical results, as all are based on same traditional algorithm i.e FM. Except for 0-level partitioning, where both algorithms reach the same value for most of the cases, almost all cases experience a noticeable improvement.

IV. CONCLUSIONS

FM algorithm is primarily a bi-partitioning technique; it is capable of dividing a single netlist file into two equally sized partitions. With the advancements in the technology we are coming up with more & more complex IC's day by day, this creates bigger circuits which in hand requires the circuit to be partitioned into more than two partitions; this is not possible with the core FM technique, but as FM algorithm itself is very stable so it can be a good stable platform for other multiway partitioning algorithms. Iteration / repetition based algorithms remove the two drawbacks of FM algorithm that are: partition size constraint and the no. of partitions constraint. A multiple-way network partitioning algorithm unlike FM algorithm can handle & cover more than one objective function. FM algorithm can only act as aid but is insufficient to meet all the constraints of partitioning. A multiple-way network partitioning algorithm was implemented which covered three different objective functions (top down clustering, primal approach & Multi-pin net).

REFERENCES

- [1]Sao-Jie Chen and Chung-Kuan Cheng, "Tutorial on VLSI Partitioning", pp. 175-218, Vol. 11, No. 3, VLSI Design, 2000.
- [2]Zhizi Zhao, Lixin Tao, and Yongchang Zhao, "An Effective Algorithm for Multiway Hypergraph partitioning", IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications, VOL. 49, no. 8, August 2002.
- [3]Saurabh N. Adya, Mehmet C. Yildiz, Igor L. Markov, Paul G. Villarrubia, Phiroze N. Parakh and Patrick H. Madden, "Benchmarking For Large-Scale Placement and Beyond", paper in proc. of ISPD, April 2003.
- [4]C. J. Alpert, J. H. Huang, and A. B. Kahng, "Multilevel circuit partitioning", In Proc. of the 34th ACM/IEEE Design Automation Conference, 2010.
- [5]G. Karypis and V. Kumar, "Multilevel k-way hypergraph partitioning", In Proceedings of the Design and Automation Conference, 2011.
- [6]George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekhar, "Multilevel hypergraph partitioning: Application in vlsi domain", IEEE Transactions on VLSI Systems. A short version appears in the proceedings of DAC 2007.
- [7] Scott Hauck and Gaetano Borriello, "An Evaluation of Bipartitioning Techniques", IEEE Transactions on CAD, Volume 16, No. 8, pp. 849-866, August 2013.