FastPlace: An Analytical Placer for Mixed-Mode Designs

Natarajan Viswanathan
nataraj@iastate.edu

Min Pan
panmin@iastate.edu

Chris Chong-Nuen Chu
cnchu@iastate.edu

Department of Electrical and Computer Engineering
Iowa State University, Ames, IA 50011-3060

ABSTRACT

Modern designs often contain a combination of a large number of standard cells and macro blocks. Traditionally large macro blocks are handled at the floorplanning level, after which their positions are fixed. The standard cells are then handled during the placement level. Current designs can have hundreds of large and medium sized macro blocks and a large number of standard cells. As a result, traditional floorplanning techniques cannot scale to this problem, both in terms of runtime and solution quality. Hence a technique is required to simultaneously handle this combination of placeable objects.

In this paper, we present a combined placement and floorplanning approach for mixed-mode placement. We extend the efficient analytical placement algorithm *FastPlace* by integrating a simulated annealing based floorplanner to solve the global placement problem for mixed-mode designs. We also present an efficient and effective detailed placement algorithm to improve the wirelength of the global placement solution based on a greedy swapping heuristic.

Categories and Subject Descriptors

B.7.2 [**Hardware, Integrated Circuits, Design Aides**]: Placement and Routing

General Terms

Algorithms, Design

Keywords

Analytical Placement, Floorplanning, Mixed-mode Placement

1. INTRODUCTION

Current designs often contain over a million placeable components, and it is predicted that circuit sizes will continue to double every three years [1]. Such a design complexity imposes enormous challenges on placement algorithms. Placement tools have to produce a good-quality result satisfying various design objectives, such as timing, power, congestion etc. They should also be efficient enough to deliver the solutions in a reasonable amount of run-time. Hence, it is essential to have efficient placement algorithms that can handle a variety of design objectives.

Modern designs often contain a combination of a large number of standard cells and macro blocks. This design style, known as mixed-mode design, complicates the placement step and imposes a lot of difficulty on placement tools due to the varied sizes of the placeable components. Traditionally, large macro blocks were handled during the floorplanning stage followed by standard cell placement, during which the macro blocks were treated as fixed. Current designs can have hundreds of large and medium sized macro blocks and thousands or millions of standard cells. As a result, traditional floorplanning techniques cannot scale to this problem, both in terms of runtime and solution quality. With an increasing trend toward mixed-mode design, and an increase in the number of macro blocks, it is necessary to have techniques that can simultaneously handle this combination of placeable objects in mixed-mode design.

Over the last few years, the mixed-mode placement problem has generated a lot of interest. Previous approaches in this area include [2, 3, 4] that used a combination of min-cut placement and floorplanning to solve the placement problem. [9] proposed a recursive bisection algorithm based on the fractional cut approach of [5] for mixed-mode placement. [11] used a quadratic placement algorithm along with a bottom-up two-level clustering strategy and slicing partitions to remove overlaps. In [6], a simulated annealing based multi-level optimization tool was employed for handling mixed-mode designs.

Analytical placement is quite favourable in solving large scale mixed-mode placement problems. Analytical placement techniques are quite efficient and can also seamlessly handle large blocks during placement. In this paper, we present a combined placement and floorplanning approach for mixed-mode placement. We extend the efficient analytical placement algorithm *FastPlace* [10] by integrating a simulated annealing based floorplanner to solve the global placement problem for mixed-mode designs. We also present an efficient detailed placement algorithm to improve the wirelength of the global placement solution by swapping standard cells based on greedy heuristic.

The rest of the paper is organized as follows: Section 2 provides an overview of the algorithm. Section 3 describes the individual steps of the global placement flow. Section 4 describes the legalization technique, followed by Section 5 that describes the detailed placement algorithm.

ISPD'05, April 3–6, 2005, San Francisco, California, USA.
ACM 1-59593-021-3/05/0004.

2. OVERVIEW OF THE ALGORITHM

Our flow is composed of three stages: Global placement, Legalization and Detailed Placement. The overall flow of the placement algorithm is summarized in Figure 1 and the individual components of the flow are discussed in more detail in Sections 3–5.

Algorithm *Mixed-mode Placement*

Stage 1: Global Placement

1. Perform Initial Global Placement on entire netlist.
2. Perform Physical Clustering of standard cells into soft blocks and perform fixed-die Floorplanning on soft blocks and macro blocks.
3. Perform Final Global Placement on entire netlist.

Stage 2: Legalization

1. Legalize macro blocks.
2. Determine placeable segments in each row and move standard cells among rows to satisfy row capacities.
3. Move standard cells among placeable segments to satisfy segment capacities.
4. Legalize standard cells within segments.

Stage 3: Detailed Placement

For standard cells,

1. Perform Global Swapping.
2. Perform Local Swapping.
3. Perform Local Shifting.

Figure 1: The Mixed-mode Placement Algorithm.

3. GLOBAL PLACEMENT

In this section we describe the individual components of the global placement stage. The global placement stage comprises of three steps: *Initial Global Placement*, *Physical Clustering and Floorplanning* followed by *Final Global Placement*.

3.1 Initial Global Placement

The aim of the Initial Global Placement step is to spread the cells over the placement region to obtain a coarse global placement that serves as a good initial solution for the subsequent physical clustering. During this step we consider the entire netlist (comprising of both standard cells and macro blocks) for placement. As described in [10], we use an iterative procedure in which we alternate between Global Optimization and Cell Shifting to spread the standard cells and macro blocks over the placement region. We handle the macro blocks in the same manner as the standard cells during Cell Shifting. To accelerate this step for circuits that have a size above a fixed threshold, we employ a hyperedge clustering scheme similar to the one described in [8]. For all other circuits, a flat placement is performed over the entire netlist.

3.2 Physical Clustering and Floorplanning

The aim of the second step is to remove the overlap among the macro blocks and also fix the relative positions of the macro blocks. During this step, the standard cells are initially clustered based on their locations obtained after the initial placement step. The entire placement region is divided into bins and and a physical clustering algorithm is used to cluster the standard cells into soft blocks (i.e., the aspect ratio of these blocks can vary during the subsequent floorplanning). The macro blocks in the circuit are not clustered with any of the cells and are treated as hard blocks (i.e., the aspect ratio of these blocks are fixed during floorplanning). This clustering creates a floorplanning instance and also serves as an intial solution for the floorplanner. After clustering, a simulated annealing based fixed-die floorplanner is used to remove the overlap among the macro blocks and also simultaneously reduce the wirelength of the placement. Finally, all the standard cells belonging to soft blocks are declustered and placed at the center of their corresponding floorplanning blocks.

3.3 Final Global Placement

The aim of this step is to further reduce the overlap among the cells and also reduce the wirelength of the placement. During this step, we interleave an Iterative Local Refinement technique along with Global Optimization and Cell Shifting to reduce the wirelength of the placement. We move both standard cells and macro blocks around the placement region. While moving the macro blocks, we maintain their relative ordering obtained after the floorplanning step. We also introduce additional constraints on the macro blocks to avoid creating any overlaps amongst them. This stage yields a well distributed placement solution with a very good value for the total wirelength.

4. LEGALIZATION

After global placement, we assign the macro blocks to the nearest legal positions such that there is no overlap among them. We then fix the positions of the macro blocks for all subsequent steps and treat them as placement blockages. We then divide each row in the placement region into placeable segments based on the overlap of the blockages with the row. We now use a greedy heuristic to bring every row in the placement region to within its capacity by moving the standard cells. This heuristic also tries to simultaneously reduce the wirelength of the placement. Once the rows have been brought under capacity, we move the cells among the placeable segments to satisfy their respective capacities. The cells are then assigned to legal positions within each segment.

5. DETAILED PLACEMENT

In this section, we give an overview of our detailed placement algorithm. After all the standard cells have been assigned to legal positions within each placeable segment, we employ an efficient and powerful detailed placer to further reduce the wirelength. It comprises of a *Global Swapping* step, a *Local Swapping* step followed by a *Local Shifting* of cells.

In the *Global Swapping* step, we try to find better positions for cells in terms of wirelength over the entire placement region. For every standard cell, we determine its "optimal" region. The "optimal" region for a cell is one in which the total wirelength of the placement is minimized when all the other cells are fixed. The "optimal" region for a cell is determined based on the median idea of [7]. Consider cell i.

Let $N_i = \{n_1, n_2, ..., n_k\}$ denote the set of nets that contain i. We traverse all the nets in N_i and find their respective bounding boxes. Here, cell i is excluded from the nets when computing their bounding boxes. For each net $n_j \in N_i$, we find its bounding box $\{(x_l[n_j], y_l[n_j]), (x_r[n_j], y_u[n_j])\}$ - the lower left and upper right coordinates. The optimal position for cell i is given by (x_{opt}, y_{opt}), where x_{opt} and y_{opt} are the medians of the x series ($x_l[1]$, $x_r[1]$, $x_l[2]$, $x_r[2]$, ...) and y series ($y_l[1]$, $y_u[1]$, $y_l[2]$, $y_u[2]$, ...) of bounding boxes. In general, the optimal position is a region rather than a point because the total number of elements in the x and y series are even. In some cases, the optimal region can degrade to a point or a line when the two medians of the x and/or the y series carry the same value. After the "optimal" region is found, we determine the intersection of the "optimal" region with the placeable segments and swap the current cell with a standard cell present in this intersection so as to reduce the total wirelength.

During the *Local Swapping* step, we consider only the cells within each segment. We look at any two adjacent cells within a segment and swap them if the swapping can reduce the wirelength.

Finally, we apply a *Local Shifting* technique to all the cells that have space around them. This technique preserves the ordering of the cells and moves them locally within the segments to find its best position in terms of the wirelength while all the other cells are fixed.

All three steps of the detailed placement are very efficient and result in a significant reduction in the total wirelength from the initial legalized placement.

6. REFERENCES

[1] *The International Technology Roadmap for Semiconductors*. Semiconductor Industry Association, 2004.

[2] S. N. Adya, S. Chaturvedi, J. A. Roy, D. Papa, and I. L. Markov. Unification of partitioning, floorplanning and placement. In *Proc. IEEE/ACM Intl. Conf. of Computer-Aided Design*, pages 550–557, 2004.

[3] S. N. Adya and I. L. Markov. Consistent placement of macro-blocks using floorplanning and standard-cell placement. In *Proc. Intl. Symp. on Physical Design*, pages 12–17, 2002.

[4] S. N. Adya and I. L. Markov. Combinatorial techniques for mixed-size placement. *ACM Trans. on Design Automation of Electronic Systems*, to appear, 2005.

[5] A. Agnihotri, M. C. Yildiz, A. Khatkhate, A. Mathur, S. Ono, and P. H. Madden. Fractional Cut: Improved recursive bisection placement. In *Proc. IEEE/ACM Intl. Conf. of Computer-Aided Design*, pages 307–310, 2003.

[6] C. C. Chang, J. Cong, and X. Yuan. Multi-level placement for large-scale mixed-size IC designs. In *Proc. Asia South Pacific Design Automation Conf.*, pages 325–330, 2003.

[7] S. Goto. An efficient algorithm for the two-dimensional placement problem in electrical circuit layout. *IEEE Trans. on Circuits and Systems*, Vol. CAS-28(1):12–18, 1981.

[8] G. Karypis. *Multilevel Optimization in VLSICAD*. Kluwer Academic Publishers, 2003.

[9] A. Khatkhate, C. Li, A. R. Agnihotri, M. C. Yildiz, S. Ono, C.-K. Koh, and P. H. Madden. Recursive bisection based mixed block placement. In *Proc. Intl. Symp. on Physical Design*, pages 84–89, 2004.

[10] N. Viswanathan and C. C.-N. Chu. Fastplace: Efficient analytical placement using cell shifting, iterative local refinement and a hybrid net model. *IEEE Trans. Computer-Aided Design*, to appear, 2005.

[11] H. Yu, X. Hong, and Y. Cai. MMP: A novel placement algorithm for combined macro block and standard cell layout design. In *Proc. Asia South Pacific Design Automation Conf.*, pages 271–276, 2000.