Planning Buffer Locations by Network Flows *

Xiaoping Tang and D.F. Wong
Department of Computer Sciences
University of Texas at Austin
Austin, TX 78712
(tang, wong)@cs.utexas.edu

Abstract

The problem of planning the locations of large number of buffers is of utmost importance in deep submicron VLSI design. Recently, Cong et al in [1] proposed an algorithm to directly address this problem. Given a placement of circuit blocks, a key step in [1] is to use the free space between the circuit blocks for inserting as many buffers as possible. This step is very important because if all buffers can be inserted into existing spaces, no expansion of chip area would be needed. An effective greedy heuristic was used in [1] for this step. In this paper, we give a polynomial-time optimal algorithm for solving the problem of inserting maximum number of buffers into the free space between the circuit blocks. In the case where the "costs" of placing a buffer at different locations are different, we can guarantee to insert maximum number of buffers with minimum total cost. Our algorithm is based on efficient min-cost network-flow computations.

1. Introduction

Rapid advances in integrated circuit technology have led to a dramatic increase in the complexity of VLSI circuits. According to the National Technology Roadmap for Semiconductors [2], we will soon have designs in less than 0.1 micron technology with over 100 million transistors. With the evolution of VLSI fabrication technology, the communication between different components, i.e., interconnect on a chip, is heavily increased. Interconnect delay, especially global interconnect delay, has become the dominant factor in deep submicron design in determining the overall circuit performance and complexity. Many techniques are employed to reduce interconnect delay. Among them, buffer insertion has shown to be an effective approach [3]. For example, optimal buffer insertion can achieve 7X reduction over the delay of a 2cm global interconnect as shown in [3]. As transistor count and chip dimension get larger and larger, more and more buffers are expected to be needed for high-performance. For example, it was estimated in [4] that close to 800,000 buffers are needed for 50nm technology. Since buffers are implemented by transistors, they can not be placed over the existing circuit blocks. Placing a large number of buffers between circuit blocks could significantly impact the chip floorplan. Therefore, it is necessary to start buffer planning as early as possible.

Many buffer insertion algorithms have been proposed in the past few years [5, 6, 7, 8, 9, 10] but all of them were designed for post-layout interconnect optimization of a single net. [11] combined routing and buffer insertion to minimize interconnect delay, but it is also only applicable for a single net. Recently, Cong et al in [1] proposed an algorithm to directly address the problem of planning buffer locations for all nets. Given a placement of circuit blocks, a key step in [1] is to use the free space between the circuit blocks for inserting as many buffers as possible. This step is very important because if all buffers can be inserted into existing spaces, no expansion of chip area would be needed. An effective greedy heuristic was used in [1] for this step. In this paper, we give a polynomial-time optimal algorithm for solving the problem of inserting maximum number of buffers into the free space between the circuit blocks. In the case where the "costs" of placing a buffer at different locations are different, we can guarantee to insert maximum number of buffers with minimum total cost. Our algorithm is based on efficient min-cost network-flow computations.

The rest of the paper is organized as follows. Section 2 introduces the concepts of feasible region and buffer zone for buffer insertion. Section 3 formulates the buffer planning problem as a network flow problem which can then be solved by classical min-cost network-flow algorithms. The most difficult part in the network construction is the computation of buffer rooms which are subdivisions of the feasible regions. Section 4 gives efficient algorithms to compute all buffer rooms and their intersections with buffer zones. Finally, we conclude the paper with some remarks in Section 5.

*This work was partially supported by the Texas Advanced Research Program under Grant No. 003658288 and by a grant from the Intel Corporation.

ISPD 2000, San Diego, CA.