# Congestion Reduction during Placement with Provably Good Approximation Bound

X. YANG
Synplicity, Inc.
M. WANG
Cadence Design Systems, Inc.
R. KASTNER
University of California, Santa Barbara
and
S. GHIASI and M. SARRAFZADEH
University of California, Los Angeles

This paper presents a novel method to reduce routing congestion during placement stage. The proposed approach is used as a post-processing step in placement. Congestion reduction is based on local improvement on the existing layout. However, the approach has a global view of the congestion over the entire design. It uses integer linear programming (ILP) to formulate the problem of conflicts between multiple congested regions, and performs local improvement according to the solution of the ILP problem. The approximation algorithm of the formulated ILP problem is studied and good approximation bounds are given and proved. Experiments show that the proposed approach can effectively alleviate the congestion of global routing results. The low computational complexity of the proposed approach indicates its scalability on large designs.

Categories and Subject Descriptors: B.7.2 [**Integrated Circuits**]: Design Aids

General Terms: Algorithms, Experimentation

Additional Key Words and Phrases: Physical design, placement, routability, congestion

This work was supported in part by the National Science Foundation (NSF) grant CCR-0090203. A preliminary version of this paper appeared in the Proceedings of the 2001 International Conference on Computer Aided Design (ICCAD).
Authors' addresses: X. Yang, Synplicity Inc., 600 California Ave. Sunnyvale, CA 94086; email: xjyang@synplicity.com; Maogang Wang, Cadence Design Systems, Inc., 555 River Oaks Parkway, Building 4, MS 4A1, San Jose, CA 95134; email: mgwang@cadence.com; Ryan Kastner, Engineering I, Room 4123, Department of ECE, University of California, Santa Barbara, Santa Barbara, CA 93106; email: kastner@ece.ucsb.edu; S. Ghiasi and M. Sarrafzadeh, Computer Science Department, University of California, Los Angeles, 405 Hilgard Ave., Los Angeles, CA 90095; email: {soheil; majid}@cs.ucla.edu.

## 1. INTRODUCTION

As VLSI system complexity continues to increase, physical design is getting more and more difficult. Traditional placement tools focus on minimizing total wirelength to obtain better routability and smaller layout area [Dunlop and Kernighan 1985; Sun and Sechen 1995; Kleinhans et al. 1991]. Despite the pervasive use of a half-perimeter wirelength objective, there is a mismatch between wirelength and congestion objectives in placement [Caldwell et al. 2000]; a placement with less total wirelength does not necessarily correspond to a better layout after routing. Congestion—an important objective indicating routability—has not drawn enough research attention in placement related studies. Although dealing with congestion is widely addressed in routing algorithms, in most cases, a portion of routing violation cannot be removed given fixed cell locations. It is of value to consider routability in the placement stage where the effort on congestion reduction would be more effective [Kahng et al. 2000]. Two recently proposed congestion estimation models, a Rent's rule based model [Yang et al. 2001] and a probabilistic model [Lou et al. 2001], showed the trend of addressing the congestion problem at early design stages.

In Cheng [1994], a routability model was proposed and incorporated in the annealing-based placement. The use of the model effectively reduces congestion. However, the proposed approach discards the extensive research work on wirelength minimization, and it significantly degrades placement speed. A multi-partitioning technique using pre-determined Steiner trees was introduced in Mayrhofer and Lauther [1990]. The restriction on the number of partitions confines the performance of the approach. A congestion driven placement approach was proposed in Parakh et al. [1998]. It uses an area router to evaluate local congestion during placement. Several other approaches [Tsay and Chang 1992; Wang et al. 2000a, 2000b], also incorporate routing within placement. In practice, combining global router and placer is an effective way to improve routability, yet researchers study more efficient approaches to handle the increasing design size.

A recent study [Wang et al. 2000b] shows that a post-processing technique is effective to minimize congestion because congestion correlates with wirelength in a global view. However, reducing congestion after a wirelength-driven placement is a non-trivial problem. Traditionally, people perturb existing placement within a window around the congested area [Tsay and Chang 1992; Wang et al. 2000a]. Local improvement within small windows has limited effect, whereas expanding search windows will cause interactions between congested areas, resulting in unpredictable results.

This paper presents a novel technique based on integer linear programming (ILP) to alleviate congestion in placement. The proposed approach is used as a post-processing step during the detailed placement stage. We study the difference between placement congestion and routing congestion, propose the congestion expansion technique to reduce congestion, and transform the expansion problem of multiple congested areas into an ILP problem. We also discuss the approximation algorithms for the proposed ILP problem and provide a good alternative approach to solve the ILP problem efficiently. Our goal is to achieve a

less congested layout after global routing by modifying the existing placement. To demonstrate the effectiveness of the proposed method, we use the overflow after global routing as a measurement of the placement quality. The proposed approach effectively reduces the overflow of global routing, as well as the total routed wirelength.

The rest of this paper is organized as follows. Section 2 gives preliminaries. Section 3 describes the routing estimation and congestion measurement used in this work. In section 4, we introduce an ILP based algorithm, which alleviates congestion during the detailed placement. Additionally, an approximation algorithm to solve the ILP problem is proposed and analyzed. Section 5 presents experimental results to show the effectiveness of the approach. We conclude in section 6.

## 2. PRELIMINARIES

A *circuit* can be modeled with a hypergraph $G(C, N)$, where $C$ is a set of cells and $N$ is a set of nets. A *net* $e \in N$ is a subset of $C$ that contains two or more cells. A *placement* is a set of locations for all cells within a rectangular chip area.

During detailed placement and global routing, the core area is divided into $m \times n$ rectangular *global bins*. For standard-cell designs, we set $n$ to the number of standard-cell rows; $m$ is set so that the average number of cells per global bin is less than 5.

The boundaries of the global bins are *global bin edges*. For each global bin edge $e$, the *routing demand* $d(e)$ is the number of wires that cross this boundary; the *routing supply* $s(e)$ is the number of wires that are allowed to cross the boundary. The *overflow* of a boundary, *overflow*$(b)$, is $\max(d(b) - s(b), 0)$. The *total overflow* of a design is the sum of the overflows over all the global bin edges of the design. A lower overflow usually indicates better routability of the circuit.

The *bounding box* of a net is the minimum rectangle that contains all the cells belonging to this net. The *total bounding box wirelength* of a design is the summation of the half-perimeter of the bounding box over all the nets. If we assume that the width and the height of global bins are unit length, the *normalized total bounding box wirelength* of a design has similar definition with total bounding box wirelength, but measured by global bin grid units. The *total routed wirelength* is the sum of actual wirelength over all the nets (including each wire segment), measured by global bin grid units.

## 3. CONGESTION IN PLACEMENT

### 3.1 Routing Estimation

To evaluate the congestion during placement, fast and accurate routing estimation is required [Cheng 1994; Lou et al. 2001]. Selecting a routing estimation model highly depends on the internal mechanism of the global router. For a general maze router, three conventional routing estimation models are widely

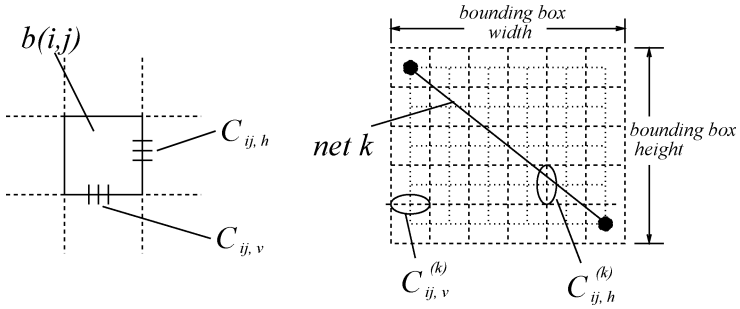



Fig. 1. Bounding box routing estimation model.

used. They are bounding box model, star model and minimum spanning tree (MST) model. The MST model is accurate but it is also computationally expensive. The bounding box model requires the least computation for updating. It also generates reasonable estimation. In this work we adopt the bounding box model of Cheng [1994], as illustrated in Figure 1.

For each global bin $b(i, j)$ at column $i$ and row $j$, let $C^{(k)}_{ij,h}$ denote the number of horizontal wire crossings on the right edge of global bin $b(i, j)$ induced by net $k$. Similarly, let $C^{(k)}_{ij,v}$ denote the number of vertical wire crossings on the bottom edge of global bin $b(i, j)$ induced by net $k$. If we use $x\ \min(k)$, $x\ \max(k)$, $y\ \min(k)$ and $y\ \max(k)$ to describe the bounding box of net $k$, we have,

$$C^{(k)}_{ij,h} = \begin{cases} \dfrac{q(k)}{y\ \max(k) - y\ \min(k) + 1} & x\ \min(k) \le i < x\ \max(k) \\ & y\ \min(k) \le j \le y\ \max(k) \\ 0 & \textit{otherwise} \end{cases}$$

$$C^{(k)}_{ij,v} = \begin{cases} \dfrac{q(k)}{x\ \max(k) - x\ \min(k) + 1} & x\ \min(k) \le i \le x\ \max(k) \\ & y\ \min(k) \le j < y\ \max(k) \\ 0 & \textit{otherwise} \end{cases}$$

where $q(k)$ is a compensation factor defined in [Cheng 1994]. The bounding box wirelength under-estimates the actual wiring for nets with more than three terminals. Therefore $q(k)$ has been introduced in order to model multi-terminal nets. $q(k)$ depends on the number of terminals of net $k$. $q(k)$ is 1 for 2-terminal or 3-terminal nets, and slowly increases to 2.79 for nets with 50 terminals.

With the routing estimation for each net, we can calculate the total estimated number of crossings for global bin edges. For each global bin $b(i, j)$, the routing demand of its right and bottom edge are:

$$C_{ij,h} = \sum_{k=1}^{N} C^{(k)}_{ij,h}$$

$$C_{ij,v} = \sum_{k=1}^{N} C^{(k)}_{ij,v}$$

### 3.2 Congestion Cost

We study the standard cell placement problem without consideration of macro cells. Therefore we assume uniformly distributed routing tracks for the entire core area. Let $Cap_h$ and $Cap_v$ be the number of tracks for vertical and horizontal global bin edges, respectively. For bin $b(i, j)$, the overflow of the right edge $OF_{ij,h}$ is $\max(C_{ij,h} - Cap_h, 0)$, and the overflow of the bottom edge $OF_{ij,v}$ is $\max(C_{ij,v} - Cap_v, 0)$.

The congestion cost function of the design can be modeled using overflow only. A more reasonable cost function would be a combination of wirelength and overflow (for a study of different congestion cost functions, see Cong and Madden [1998]). In this work, we employ a combination of wirelength and a quadratic function of overflow. The horizontal congestion of the design $COST_h$ is,

$$COST_h = \sum_{i=1}^{m-1} \sum_{j=1}^{n} \left(C_{ij,h} + OF_{ij,h}^2\right)$$

The vertical congestion $Cost_v$ is:

$$COST_v = \sum_{i=1}^{m} \sum_{j=1}^{n-1} \left(C_{ij,v} + OF_{ij,v}^2\right)$$

The total congestion cost $COST$ is the sum of $COST_h$ and $COST_v$.

Additionally, the total overflow $OF$ of the layout is the sum of overflow over all the global bin edges:

$$OF = \sum_{i=1}^{m-1} \sum_{j=1}^{n} OF_{ij,h} + \sum_{i=1}^{m} \sum_{j=1}^{n-1} OF_{ij,v}$$

### 3.3 Bin Congestion Degree

To identify the congested area, we define a *congestion degree* $C_{ij}$ for each global bin $b(i, j)$ as the average relative congestion of its four edges.

$$C_{ij} = \frac{1}{4}\left(\frac{C_{ij,h}}{C_{avg,h}} + \frac{C_{(i-1)j,h}}{C_{avg,h}} + \frac{C_{ij,v}}{C_{avg,v}} + \frac{C_{i(j-1),v}}{C_{avg,v}}\right)$$

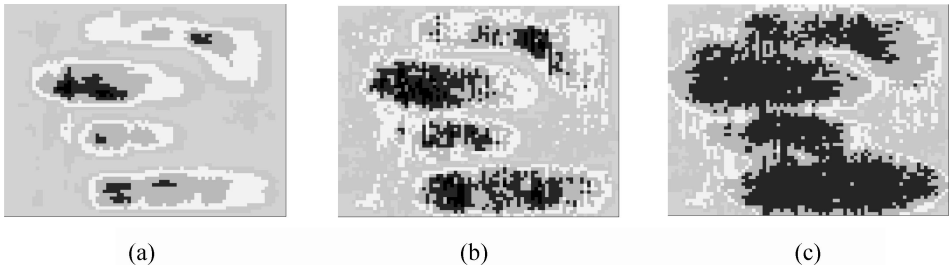
where $C_{avg,h}$ and $C_{avg,v}$ are the average numbers of horizontal and vertical crossings for all of the bins, respectively. They are obtained by,

$$C_{avg,h} = \frac{1}{(m-1)n} \sum_{i=1}^{m-1} \sum_{j=1}^{n} C_{ij,h}$$

$$C_{avg,v} = \frac{1}{m(n-1)} \sum_{i=1}^{m} \sum_{j=1}^{n-1} C_{ij,v}$$

## 4. CONGESTION REDUCTION IN DETAILED PLACEMENT

We focus on the integration of the congestion reduction technique into the detailed placement stage for the following reasons. First, the total routing



(a) (b) (c)

Fig. 2. Congested areas expand from placement to routing. (a) Estimated routing congestion in placement. (b) Congestion after maze-routing using large capacity without rip-up and re-route. (c) Congestion after maze-routing using tight capacity with rip-up and re-route. Note that the congestion threshold in placement is higher than that in routing, resulting in the expansion of the congested areas.

demand of the design globally correlates to the total wirelength. Minimizing total wirelength indirectly reduces congestion. Traditionally people use the half perimeter of the bounding box to estimate routed wirelength of a net. Extensive research on minimizing bounding box wirelength can be utilized in global congestion reduction.[2] Second congestion reduction in detailed placement costs less in terms of computation time. In most cases, combining the routing estimation into global placement will dramatically degrade the placement speed. Considering that a main portion of placement time should be allocated to timing issues, the excessive computational time for congestion reduction is discouraged. Finally, in the detailed placement stage, more accurate layout information has been revealed. Therefore the routing estimation is more accurate, and the reduction of the estimated congestion is more likely to be transformed to the reduction of routing congestion.

### 4.1 Congested Region Expansion

Based on routing estimation, there are two ways to identify a congested area in placement. We can define the congested bin as a global bin with a congestion degree (described in Section 3.3) greater than a certain threshold value. Or we can define the bin as congested if at least one of its edges is congested, that is, the overflow of at least one edge is greater than zero. Congested regions are unevenly distributed throughout the chip area.

As expected, a congested region in placement will shrink after global routing, since routers "intelligently" handle congestion. However, if we change the point of view, by setting a higher threshold value in placement than in routing, the congested region is actually enlarged due to the detours in the congested spot. The tighter the routing resources, the larger the congested area. Figure 2 gives an example of this phenomenon. It is the expansion that makes the placement congestion problem hard: the effort on reducing the congestion in placement may be unnecessary to routing, or may cause new congested areas.

[2]The authors also believe that a congestion-driven global placement (different from minimizing wirelength) would be more effective, and it should draw research attention as well.

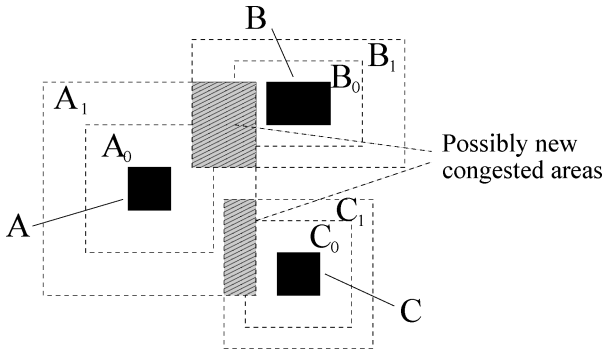



Fig. 3. Overlaps between expansion areas for multiple congested regions.

### 4.2 Multiple Congested Area Expansion Based on Integer Programming

The expansion of a congested region suggests: (a) the congestion reduction should be performed within a larger region than the congested region; (b) certain techniques are required to handle the conflicts between the expansions of multiple congested regions. We name the congestion optimization region the *expansion area*. For a single congested region, it is desired to use a larger expansion area so the congestion can be effectively reduced. However, we should bound the expansion area since a larger expansion area requires longer running time. Additionally, the expansion area of one region may overlap with that of another region if these two congested regions are close. This may cause unexpected congested regions. A larger expansion area increases the likelihood of the existence of the overlapped regions. An arbitrative mechanism is needed to determine the expansion range for each congested region. We transform this arbitrating problem into an integer programming problem.

Assume we have $K$ congested regions. Two rectangular expansion areas are assigned to each congested region, a smaller one and larger one. These expansion areas are overlapped as shown in Figure 3. If we try to reduce congestion for one congested region without consideration of the other congested region's expansion, two expansion areas may overlap and a new congested region may be created.

The problem is to find a combination of expansion schemes for all of the congested regions, such that the maximum congestion over the core area is minimized. First, we use a simplified model to describe the expansion. As shown in Figure 4, a congested region has two expansion areas: $E_0$ and $E_1$. For $E_0$, the expected average congestion degrees (or average density) $d_0$ is:

$$d_0 = \frac{1}{A_0} \sum_{(i,j)\in E_0} C_{ij}$$

where $A_0$ is the area of $E_0$.

Similarly, the expected average congestion degree $d_1$ for area $E_1$ is:

$$d_1 = \frac{1}{A_1} \sum_{(i,j)\in E_1} C_{ij}$$

where $A_1$ is the area of $E_1$.

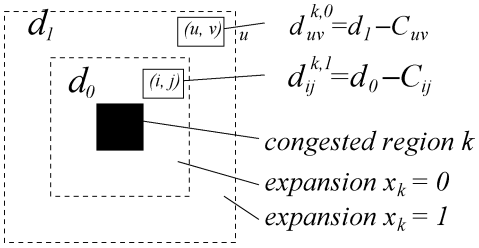



Fig. 4. Two expansion areas for a congested region.

For any global bin $b(i, j)$ in the expansion area $E_0$, its congestion degree before and after expansion scheme 0 are $C_{ij}$ and $d_0$, respectively. Thus, for congested region $k$, we define the *incremental degree* for any global bin $b(i, j)$ at expansion scheme 0:

$$d_{ij}^{k,0} = \begin{cases} d_0 - C_{ij} & \text{if} \quad b(i,j) \in E_0 \\ 0 & \text{otherwise} \end{cases}$$

The incremental degree for any global bin $b(i, j)$ at expansion scheme 1:

$$d_{ij}^{k,1} = \begin{cases} d_1 - C_{ij} & \text{if} \quad b(i,j) \in E_1 \\ 0 & \text{otherwise} \end{cases}$$

For each congested region $k$, there is a corresponding binary variable $x_k$. $x_k$ is 0 if the expansion $E_0$ is chosen, or 1 if $E_1$ is chosen.

The expansion scheme problem can be transformed into a 0–1 integer linear program (ILP) problem:

$$\text{minimize} \qquad C_{\max} \tag{1}$$

$$\text{s.t.} \qquad C_{ij} + \sum_{k=1}^{K} d_{ij}^{k,0}(1 - x_k) + d_{ij}^{k,1} x_k \le C_{\max} \tag{2}$$

$$i = 1, \ldots, m$$
$$j = 1, \ldots, n$$

$$x_k \in \{0, 1\} \qquad k = 1, \ldots, K \tag{3}$$

where $C_{\max}$ is the maximum congestion degree over all the global bins. For each global bin there is one constraint. If a global bin $b(i, j)$ is located in the expansion area $E_0$ of congested region $k$, a term $d_{ij}^{k,0}(1 - x_k) + d_{ij}^{k,1} x_k$ will be added to its constraint. If bin $b(i, j)$ is located in the expansion area $E_1$ but not $E_0$, an item $d_{ij}^{k,1} x_k$ will be generated. If bin $b(i, j)$ is located in neither of these two areas, no constraint is created from the congested region $k$. If a bin is not covered by any of the congested regions, it has a simple constraint: $C_{ij} \le C_{\max}$.

The transformed ILP problem can be optimally solved if the number of congested regions is limited. The problem solution determines the expansion scheme for each congested region. Local congestion reduction will be performed within the pre-determined expansion areas.

### 4.3 Approximation Algorithm for ILP

In this section, we will discuss the approximation approach of the formulated ILP problem. When the number of congested regions is large (e.g. $> 100$), the ILP problem cannot be solved efficiently. Even if it is a moderate number (e.g. $\sim 50 - 60$), a certain amount of running time still prevents the approach from being used in the inner loop of the optimization flow. We need to solve the transformed ILP problem as quickly as possible. On the other hand, we do not require the optimality of the solution because the solution only provides an estimated target for the congestion reduction approach. Therefore we look into approximations of the original problem.

One approach to solve this problem is using linear programming relaxation and *threshold rounding*. First we relax the integrality constraint $x_k \in \{0, 1\}$ for all $k$ and obtain the relaxation linear programming (LP) problem in which $x_k \in [0, 1]$ for all $k$. This LP problem can be solved efficiently. We solve this relaxed LP problem to obtain the fractional solution $\widehat{x}_k$. Then we use threshold rounding to obtain a solution of the original ILP problem: for all $k$, if $\widehat{x}_k \geq 0.5$, set $x_k$ to 1, otherwise set it to 0. Let $C^*$ denote the optimal value of the objective function for the original ILP problem, and let $C_{app}$ denote the objective value obtained by the approximation approach (relaxation and rounding), We have:

THEOREM 4.1. *For the ILP problem described by* (1)*,* (2) *and* (3)*, the relaxation and threshold rounding approach finds a solution with an objective value $C_{app}$ such that $C_{app} \leq 2C^*$, where $C^*$ is the optimal solution for the ILP problem.*

PROOF. Consider the left side of the constraint inequality in the ILP problem. For each global bin $b(i, j)$, each congested region $k$ corresponds to a portion of the constraint: $d_{ij}^{k,0}(1 - x_k) + d_{ij}^{k,1}x_k$. Let $\widehat{D}_{ij}^k$ denote the value of this portion before rounding and $D_{ij}^k$ denote the value after rounding. Assume for the optimal solution of the relaxed LP problem, $\widehat{x}_k < 0.5$ and $x_k$ is set to 0, we have:

$$\begin{aligned}\frac{D_{ij}^k}{\widehat{D}_{ij}^k} = \frac{d_{ij}^{k,0}(1 - x_k) + d_{ij}^{k,1}x_k}{d_{ij}^{k,0}(1 - \widehat{x}_k) + d_{ij}^{k,1}\widehat{x}_k} &\leq \frac{d_{ij}^{k,0}}{d_{ij}^{k,0}(1 - \widehat{x}_k) + d_{ij}^{k,1}\widehat{x}_k} \\ &\leq \frac{d_{ij}^{k,0}}{d_{ij}^{k,0}(1 - \widehat{x}_k)} \\ &\leq 2\end{aligned}$$

Similarly, for $\widehat{x}_k \geq 0.5$ and $x_k = 1$, the ratio between $D_{ij}^k$ and $\widehat{D}_{ij}^k$ is also less than or equal to 2. Therefore,

$$\frac{C_{ij} + \sum_{k=1}^{K} D_{ij}^k}{C_{ij} + \sum_{k=1}^{K} \widehat{D}_{ij}^k} \leq 2$$

$$i = 1, \ldots, m \quad j = 1, \ldots, n$$

Since the objective function $C_{\max}$ is determined by the maximum left side of inequality constraint for all $i$ and $j$, the value of the objective function after

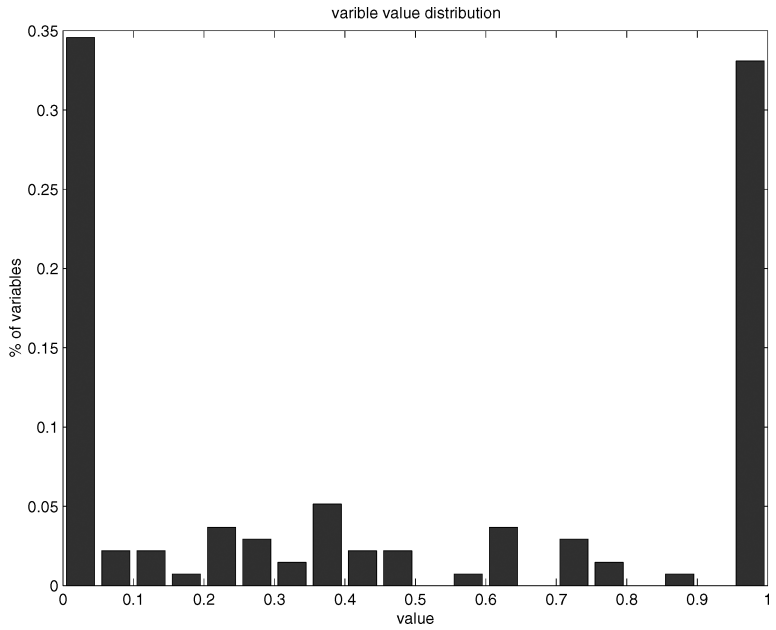



Fig. 5. Relaxed ILP solution distribution. Each bar shows the percentage of variables that have values in the corresponding range. The left-most bar shows that there are 35% of variables with values between 0 and 0.05; the second left bar shows that there are 2% of variables with values between 0.05 and 0.10 and so forth.

rounding $C_{app} \leq \widehat{C}$. Here $\widehat{C}$ is the optimal objective of the relaxed LP problem. Since the solution of the relaxed LP problem is the dual bound of the original ILP problem, that is, $\widehat{C} \leq C^*$, we have $C_{app} \leq 2C^*$ □

Theorem 4.1 gives a 2-approximation algorithm for the proposed ILP problem because the algorithm has a performance ratio[3] at most 2. In practice, there exists a better approximation feature for the algorithm. This is because the solution of the relaxed LP problem is not uniformly distributed over $[0, 1]$. Figure 5 shows the solution distribution of $\hat{x}_k$ for a series of problems. The figure is obtained in the following way. For a given circuit, we assign two expansion plans for each congested region, the ILP problem is then formulated, relaxed and solved. We repeat this process by setting different expansion plans. Finally, we count the total number of variables for all of the problems, and the number of variables within a given range.

According to our experimental observations, we may use a distribution model to describe the $x_k$ distribution. Here we adopt a square distribution $f(x)$ as illustrated in Figure 6. Although this is not an accurate model to describe the solution feature, we will show later that it is useful if it bounds the real distribution.

[3]Performance ratio of an algorithm is the ratio between the solution delivered by this algorithm and the optimal solution.

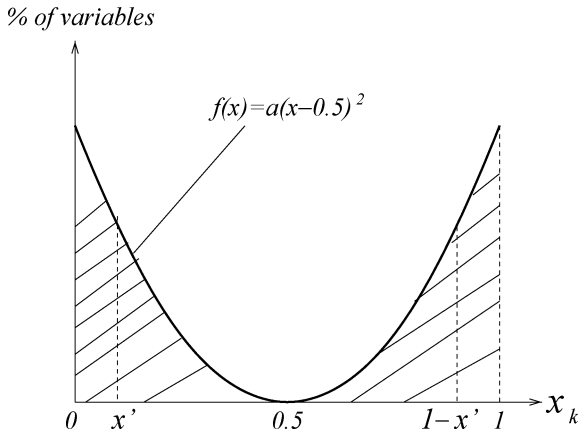



Fig. 6. Assumed distribution of the solution for the relaxed ILP.

Assume the probability distribution function is $f(x) = c(x - 1/2)^2$. $c$ is a constant such that

$$\int_0^1 c\left(x - \frac{1}{2}\right)^2 dx = 1 \tag{4}$$

From (4) we obtain $c = 12$. Thus, $f(x) = 12(x - 1/2)^2$. For any given $x'$, $0 < x' < 1/2$, the probability $p_1$ that $x_k \leq x'$ or $x_k \geq 1 - x'$ can be computed by

$$p_1 = 2\int_0^{x'} f(x)\,dx \tag{5}$$

We thus have

THEOREM 4.2. *For the ILP problem described by* (1)*,* (2) *and* (3)*, if the solution distribution* $f(x) = 12(x - 1/2)^2$*, the relaxation and threshold rounding approach finds an* $\alpha$*-approximate solution, with probability at least* $p$*. The relationship between* $\alpha$ *and* $p$ *is*

$$p = \left(8\left(\frac{\alpha - 1}{\alpha}\right)^3 - 12\left(\frac{\alpha - 1}{\alpha}\right)^2 + 6\left(\frac{\alpha - 1}{\alpha}\right)\right)^K$$

*where* $K$ *is the number of variables in the ILP problem.*

PROOF. According to (5) and $f(x)$, for a given variable $x_k$ and a given value $x'$ $(0 < x' < 1/2)$, the probability $p'$ with $x_k \leq x'$ or $x_k \geq 1 - x'$ is

$$p' = 8x'^3 - 12x'^2 + 6x' \tag{6}$$

When $x_k \leq x'$, in the worst case, rounding $x_k$ to 0 will increase the left side of a constraint from $d_0(1 - x_k) + d_1x_k$ to $d_0$. So, the ratio between the constraint value with or without rounding is:

$$\frac{d_0}{d_0(1 - x_k) + d_1x_k} \leq \frac{1}{1 - x_k} \leq \frac{1}{1 - x'}$$

The similar case occurs when $x_k \geq x'$. The ratio is:

$$\frac{d_1}{d_1(1-x_k)+d_1x_k} \leq \frac{1}{x_k} \leq \frac{1}{1-x'}$$

So the threshold rounding approach is an $\alpha$-approximation algorithm and the approximation factor

$$\alpha = \frac{1}{1-x'}$$

Thus, we have

$$x' = \frac{\alpha-1}{\alpha}$$

Plug it into (6), we have

$$p' = 8\left(\frac{\alpha-1}{\alpha}\right)^3 - 12\left(\frac{\alpha-1}{\alpha}\right)^2 + 6\left(\frac{\alpha-1}{\alpha}\right)$$

This is the probability for one variable. The probability $p$ that all $K$ variables are less than $x'$ or greater than $1-x'$ is $p'^K$. □

Theorem 4.2 gives the probability of producing a bounded solution. For instance, the probability of producing a 1.5-approximate solution for 40 variables is 0.22.

Noting that the actual distribution does not necessarily obey the square distribution, we have

COROLLARY 4.3. *Theorem 4.2 holds for any distribution $g(x)$ if*

$$\int_0^{x'} g(x)\,dx \geq \int_0^{x'} f(x)\,dx, \quad \text{and}$$
$$\int_{1-x'}^{1} g(x)\,dx \geq \int_{1-x'}^{1} f(x)\,dx \quad x' = \frac{\alpha-1}{\alpha}$$

The analysis on the approximation algorithm for the ILP problem is not directly used in our congestion reduction approach. However, it reveals the problem features and suggests a fast approximation method to solve the ILP problem. In some specific cases where the ILP problem contains too many variables, and cannot be solved efficiently, the approximation algorithm can be used to obtain a reasonable solution within a short amount of time.

### 4.4 Congestion Reduction in Detailed Placement

Based on the congestion analysis, we propose an approach to alleviate the routing congestion in the detailed placement stage. The entire flow of the approach is described by Algorithm 1.

**Algorithm 1.** Congestion Reduction Algorithm for Multiple Congested Area
Input: Circuit $G$ and a detailed placement $P_0$,
Output: Placement $P_1$ with alleviated congestion
Snap cells into global bins according to their current position;
**for all** net $n \in N$ do
  Do routing estimation for net $n$;
**end for**
Calculate average horizontal/vertical routing demand over all of the edges: $C_{avg,h}$, $C_{avg,v}$
**for all** global bin $b(i, j), i = 1, \ldots, m, j = 1, \ldots, n$
  Assign an estimated congestion degree $C_{ij}$ to this bin;
**end for**
Identify congested regions;
Assign two expansion ranges for each congested region;
Formulate the ILP problem and solve it;
Determine expansion range for each congested region by ILP;
**for all** congested region **do**
  Expand this region according to pre-determined expansion range;
  Do local improvement within the range to reduce congestion;
**end for**
Create detailed placement by spreading cells in each global bin.

The congestion reduction approach starts from an existing placement after wirelength minimization. The core area is divided into uniform global bins. Cells are assigned into global bins according to their current position. Based on the routing estimation model and bin congestion degree described in Section 3, we have a congestion distribution map for the current placement.

The next step is to identify the congested regions. This is accomplished by picking a congested global bin as the seed, checking the neighborhood bins and including the congested bins in this group of congested bins. Then we use the minimum rectangle that contains these connected congested bins as one congested region. A new seed is then picked to form the next congested region. For some designs, many congested bins are connected. A large congested region will be found based on the above approach. However, a large congested region may degrade the effect of the congestion reduction within its range. In this case, we set a maximum area for the congested regions to prevent the formation congested regions that are too large.

Once we have all of the congested regions, we assign two expansions to each. The size of the expansion area is proportional to the original congested region. We use $x\%$ to denote the expansion scale of a congested region, that is, the width and height of the expanded region will be $x\%$ longer than that of the original region. For each congested region, we have two expansion plans: a larger one and a smaller one. The selection between these two plans is made by formulating and solving the ILP problem described in Section 4.2.

The expansion area determines the range of local improvement for a congested region. The local improvement is based on cell swapping. A pair of cells are randomly chosen and swapped. Routing estimations of the nets that connect these two cells are re-evaluated. The swap will be accepted if the total congestion cost in the expansion area is lower after swapping, and will be rejected otherwise. In order to speed up the performance, cell swapping is performed

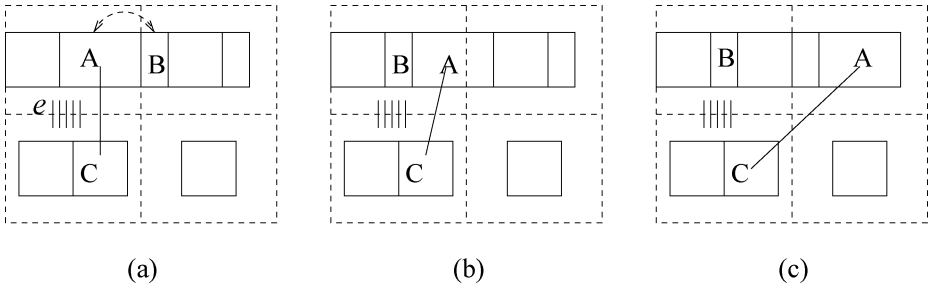



Fig. 7. Cell reordering in the congestion reduction approach: (a) Cells A and B are swapped to reduce congestion on edge $e$; (b) Swapping in the global bin stage will be nullified in detailed placement; (c) Re-order cell A after swapping to ensure the effect of swapping.

based on global bin structure—cells are located at the center of global bins. This may cause cost function mismatch between the current placement and the final placement after resolving the overlaps. To limit the mismatch, row balance and bin balance are maintained during congestion reduction.[4]

A *cell re-ordering* technique is used in this approach to enhance the congestion-driven swapping. This technique is described in Figure 7. In Figure 7(a) cell A has an interconnect to cell C. Swapping cells A and B will reduce the congestion cost on edge $e$. However, A and B may not be actually swapped. In Figure 7(b), since cell A is wider than cell B, after swapping and resolving overlap, cell A is still in its original global bin. If we use the reordering technique to put cell A at the right side in the global bin when it enters the bin, the probability of a valid swap will be much higher.

Local improvement is performed for each congested region. After a given number of improvement iterations, the algorithm terminates by spreading cells in each global bin. A final detailed placement is generated and the global routing will be executed on this detailed placement.

## 5. EXPERIMENTAL RESULTS

The proposed approach has been implemented in C. All the experiments were done in the Linux environment on a PC with a 733MHz CPU. The experimental circuits are chosen from IBM-PLACE benchmarks [ERLAB(a)]. Table I shows the circuit statistics. Number of grids and vertical/horizontal capacities in global routing are also shown.

Table II shows the effect of congestion reduction as a post-processing. We compare two placement flows in the experiments. First, we place each circuit using a wirelength-driven placer, Dragon [Wang et al. 2000c], then route it using a global router Labyrinth [ERLAB(b)] which is based on maze routing and rip-up and re-route. In the second run, we apply the proposed congestion reduction approach on the same placement, followed by global routing using the same parameters. We compare the global routing results (overflow and routed wirelength) for placements with or without congestion reduction. For

[4]We use row balance factor 0.01 and bin balance factor 0.50 in this work.

Table I. Tested Circuit Statistics, Including Number of Cells, Number of Nets, Number of Global Bins, Average Number of Cells per Bin and Vertical/Horizontal Capacity in Global Routing

| *ckt* | *#cells* | *#nets* | Grids | #c/bin | V/H Cap |
|---|---|---|---|---|---|
| ibm01 | 12,036 | 13,056 | $64 \times 64$ | 2.9 | 12/14 |
| ibm02 | 19,062 | 19,291 | $80 \times 64$ | 3.7 | 22/34 |
| ibm03 | 21,924 | 26,104 | $80 \times 64$ | 4.3 | 20/30 |
| ibm04 | 26,346 | 31,328 | $96 \times 64$ | 4.3 | 20/23 |
| ibm05 | 28,146 | 29,647 | $128 \times 64$ | 3.4 | 42/63 |
| ibm06 | 32,185 | 34,935 | $128 \times 64$ | 3.9 | 19/34 |
| ibm07 | 45,135 | 46,885 | $192 \times 64$ | 3.6 | 21/36 |
| ibm08 | 50,977 | 49,228 | $192 \times 64$ | 4.1 | 21/32 |
| ibm09 | 57,746 | 59,454 | $256 \times 64$ | 3.1 | 14/28 |
| ibm10 | 67,692 | 72,760 | $256 \times 64$ | 4.1 | 27/40 |

Table II. Global Routing Results Comparison Between Placement Without Congestion Reduction (NCR) and With CR. The Total Overflow, Bounding Box Wirelength, Normalized Bounding Box Wirelength and Normalized Routed Wirelength are Compared. Lower Overflows are Shown in Boldface. Wirelength With CR is Relative to that Without CR. Runtime is in CPU Seconds

| | Overflow | | BB-WL | | Grid-BB-WL | | Grid-routed-WL | | Runtime |
|---|---|---|---|---|---|---|---|---|---|
| *ckt* | NCR | CR | NCR | CR | NCR | CR | NCR | CR | (secs) |
| ibm01 | 398 | 309 | 4.79 | 0 | 52742 | 0 | 76517 | −0.9% | 11 |
| ibm02 | 492 | 292 | 13.82 | −0.7% | 142240 | −0.1% | 204734 | −2.4% | 33 |
| ibm03 | 209 | 181 | 13.02 | 0 | 129654 | 0 | 185194 | −1.8% | 25 |
| ibm04 | 882 | 778 | 16.31 | +0.1% | 147943 | 0 | 196920 | −0.7% | 39 |
| ibm05 | 251 | 0 | 37.14 | 0 | 472110 | 0 | 689671 | −9.8% | 134 |
| ibm06 | 834 | 540 | 20.67 | −0.1% | 241014 | −0.1% | 346137 | −0.9% | 169 |
| ibm07 | 697 | 686 | 30.76 | +0.1% | 327253 | −0.1% | 449213 | +0.4% | 164 |
| ibm08 | 665 | 654 | 33.56 | −0.1% | 339442 | −0.1% | 469666 | −0.2% | 181 |
| ibm09 | 505 | 268 | 30.38 | −0.1% | 367378 | −0.1% | 481176 | −1.2% | 232 |
| ibm10 | 588 | 383 | 59.81 | +1.1% | 513215 | +1.2% | 679606 | −0.3% | 389 |

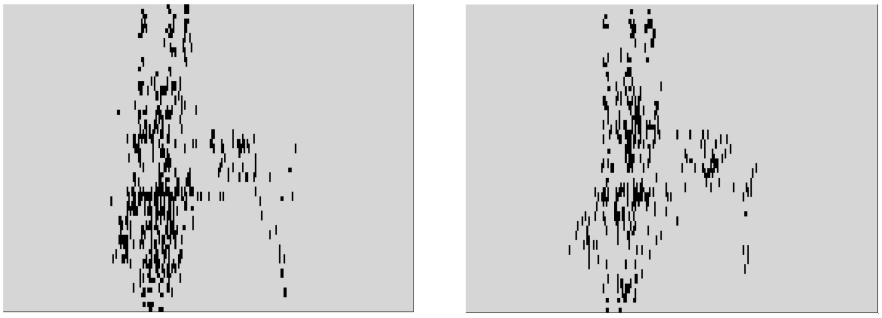
each circuit, we intentionally adjust the vertical/horizontal capacity in global routing to obtain a reasonable amount of overflow.

As can be seen, the congestion reduction approach considerably reduces the total overflow after global routing. For the best case among all the results, circuit *ibm05*, the total overflow turns to zero by congestion reduction. Almost all the circuits have a shorter Total routed wirelength after congestion reduction. This indicates that wirelength is not sacrificed due to the reduction of overflow. We also noted that the total bounding box wirelength (both actual or normalized grid wirelength) with or without congestion reduction do not differ significantlly. This reveals: (a) the bounding box wirelength is no longer a good metric of routability in detailed placement, and (b) our proposed approach creates little perturbation on the existing placement. Finally, the short amount of running time shows that the method can scale well on large circuits.

Table III shows the effect of the ILP based expansion technique in congestion reduction. For a given circuit ibm02, which contains 9 congested regions, we use a single expansion technique with different expansion areas from 10% to 100%. The double expansion technique is also applied with different combinations of

Table III. Comparison Between Single Expansion and ILP Based Expansion for Circuit ibm02 (9 congested regions). Overflow is Reported for each Expansion Area $x\%$ in The Single Expansion Case, and for each Combination of Two Expansion Areas $x_1\%/x_2\%$ in The Double Expansion Case. Both ILP and Relaxed LP are Used in Double Expansion Case

| Single Expansion | | | | Double Expansion | | | |
|---|---|---|---|---|---|---|---|
| | | | | ILP | | LP | |
| exp % | OF | exp % | OF | exp % | OF | exp % | OF |
| 10 | 447 | 60 | 433 | 10/60 | 433 | 10/60 | 412 |
| 20 | 457 | 70 | 439 | 20/70 | 396 | 20/70 | 381 |
| 30 | 473 | 80 | 355 | 30/80 | 436 | 30/80 | 458 |
| 40 | 427 | 90 | 457 | 40/90 | 388 | 40/90 | 421 |
| 50 | 430 | 100 | 424 | 50/100 | 380 | 50/100 | 425 |



(a) (b)

Fig. 8. Comparison of different layouts with and without congestion reduction for circuit ibm10. Overflowed edges are shown in black. (a) layout without congestion reduction, overflow = 588, (b) layout with congestion reduction, overflow = 383.

expansion areas. To solve the formulated ILP problem, we use an ILP solver[5] or relax the problem and solve the corresponding LP problem. All of these methods are tested by reporting the total overflow after global routing. As can be seen, in general the double expansion scheme produces better placement compared to the single expansion—20%/70% combination is better than either 20% or 70% expansion.

Figure 8 shows the effect of our congestion reduction approach on circuit *ibm*10. Figure 8(a) is the overflowed edge map after global routing without congestion reduction. Figure 8(b) is the map with congestion reduction, containing fewer overflowed edges.

## 6. CONCLUSION AND DISCUSSION

We have developed an algorithm to alleviate congestion during placement. In the congestion reduction process, a routing estimation model is used to evaluate edge and bin congestion. Congested spots on the design are relieved using local improvement within a search window. Such a window size is determined by

[5]CPLEX 7.0

formulating and solving an ILP problem when dealing with multiple congested regions. We also study the approximation algorithm for the proposed ILP problem. Experimental results on overflow of routed designs show that the proposed method is an effective way to improve design routability.

In this paper, we focus on a post-processing congestion reduction problem. However, circuit performance is becoming more and more critical for modern designs. Meeting timing constraints and producing routable designs are two indispensable objectives in the placement stage. It should be noted that congestion and timing are two highly related objectives in placement. In many cases, addressing the congestion issue would greatly help timing optimization. An obvious example is the following. If some regions of a placed circuit are highly congested, it is more likely to have many detoured wires in the final layout. Excessive detoured wires cause bad timing prediction in placement and nullify any timing-driven placement method. Another example is that reducing congestion is helpful for buffering, since the newly added buffers have little impact on routability if the current congestion is under control. In some other cases, however, reducing congestion may have a negative effect on timing improvement. For better routability, some cells are moved out of high density regions. This may cause longer interconnections on critical paths. An appropriate trade-off between congestion and timing optimization is desired in this case.

## ACKNOWLEDGMENTS

The authors would like to thank the reviewers for their detailed feedback. The authors are also grateful to Prof. Jason Cong, Prof. Miodrag Potkonjak, Prof. Lieven Vandenberghe for their valuable comments on the manuscript.

## REFERENCES

CHENG, C. E. 1994. "RISA: Accurate and Efficient Placement Routability Modeling". In *International Conference on Computer-Aided Design*, pages 690–695.

CALDWELL, A. E., KAHNG, A. B., AND MARKOV, I. L. 2000. "Can Recursive Bisection Alone Produce Routable Placements?" In *Design Automation Conference*, pages 477–482.

CONG, J. AND MADDEN, P. 1998. "Performance Driven Multi-Layer General Area Routing for PCB/MCM Designs." In *Design Automation Conference*, pages 356–361.

DUNLOP, A. E. AND KERNIGHAN, B. W. 1985. "A Procedure for Placement of Standard Cell VLSI Circuits." *IEEE Trans. Comput. Aided Design*, *4*, 1, 92–98.

ERLAB(a). "IBM-PLACE benchmark." http://er.cs.ucla.edu/benchmarks/ibm-place/.

ERLAB(b). "Labyrinth." http://www.cs.ucla.edu/~kastner/labyrinth/.

KAHNG, A. B., MANTIK, S., AND STROOBANDT, D. 2000. "Requirements for Models of Achievable Routing." In *International Symposium on Physical Des.* pages 4–11.

KLEINHANS, J. M., SIGL, G., JOHANNES, F. M., AND ANTREICH, K. J. "GORDIAN: VLSI Placement by Quadratic Programming and Slicing Optimization." *IEEE Trans. Comput. Aided Des. 10*, 3, 365–365.

LOU, J., KRISHNAMOORTHY, S., AND SHENG, H. S. 2001. "Estimating Routing Congestion using Probabilistic Analysis." In *International Symposium on Physical Design*, pages 112–117.

MAYRHOFER, S. AND LAUTHER, U. 1990. "Congestion-Driven Placement Using a New Multi-partitioning Heuristic." In *International Conference on Computer-Aided Design*, pages 332–335.

PARAKH, P. N., BROWN, R. B., AND SAKALLAH, K. A. 1998. "Congestion Driven Quadratic Placement." In *Design Automation Conference*, pages 275–278.

SUN, W. J. AND SECHEN, C. 1995. "Efficient and Effective Placement for Very Large Circuits." *IEEE Trans. Comput. Aided Des. 14*, 3, 349–359.

TSAY, R. S. AND CHANG, S. C. 1992. "Early Wirability Checking and 2-D Congestion-Driven Circuit Placement." In *International Conference on ASIC*. pages 50–53.

WANG, M., YANG, X., EGURO, K., AND SARRAFZADEH, M. 2000a. "Multi-Center Congestion Estimation and Minimization During Placement." In *International Symposium on Physical Design*, pages 147–152.

WANG, M., YANG, X., AND SARRAFZADEH, M. 2000b. "Congestion Minimization During Top-Down Placement." *IEEE Trans. Comput. Aided Des. 19*, 10, 1140–1148.

WANG, M., YANG, X., AND SARRAFZADEH, M. 2000c. "Dragon2000: Fast Standard-cell Placement for Large Circuits." In *International Conference on Computer-Aided Design*, pages 260–263.

YANG, X., KASTNER, R., AND SARRAFZADEH, M. 2001. "Congestion Estimation During Top-down Placement." In *International Symposium on Physical Design*, pages 164–169.

Received December 2001; revised December 2002; accepted Feburary 2003