Probabilistic Congestion Prediction with Partial Blockages

Zhuo Li[1], Charles J. Alpert[1], Stephen T. Quay[1], Sachin Sapatnekar[3], and Weiping Shi[4]

[1] IBM Austin, 11501 Burnet Road, Austin, TX 77858.
[3]Dept. of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455.
[4]Dept. of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843.

***Abstract*— Fast and accurate routing congestion estimation is essential for optimizations such as floorplanning, placement, buffering, and physical synthesis that need to avoid routing congestion. Using a probabilistic technique instead of a global router has the advantage of speed and easy updating. Previously proposed probabilistic models [1] [2] do not account for wiring that may already be fixed in the design, e.g., due to macro blocks or power rails. These "partial wiring blockages" certainly influence the global router, so they should also influence a probabilistic routing prediction algorithm. This work proposes a probabilistic congestion prediction metric that extends the work of [2] to model partial wiring blockages. We also show a new fast algorithm to efficiently generate the congestion map and demonstrate the effectiveness of our methods on real routing problems.**

I. Introduction

Routability is a key factor when performing floorplanning or trying to close on timing via physical synthesis. A designer could expend considerable effort trying to get the design into a good state in terms of timing and signal integrity, only to subsequently find that it is unroutable. Ideally, the designer should be able to invoke snapshot routability analysis that allows him or her to understand the routability issues involved from making floorplanning or optimization decisions.

There are three ways to obtain this kind of analysis: (i) empirical models [3] [4], (ii) global routers [5] [6], or (iii) probabilistic analysis [1] [2]. Among them, only probabilistic routing congestion analysis can efficiently provide since it avoids actually performing global routing. Instead, given a placement, it examines the set of nets in the design and uses probability theory to compute the expected congestion for each routing tile.

Lou *et al.* [1] propose an algorithm that consider all possible pin-to-pin routes within the bounding box of the pins; each route is then assigned an equal probability. Westra *et al.* [2] recognize that this approach invariably produces biases congestion towards the middle of the bounding box instead of the periphery. It turns out that since routers also try to minimize the insertion of vias, the periphery of the bounding box actually has more congestion than the interior. Westra *et al.* [2] propose to use only L- and Z-routes for the non-obvious routes. Each possible L shape and each possible Z shape share equal probability in their model. Similar models are also used in [6] [7]. Our approach also adopts this philosophy.

Lou *et al.* and Westra *et al.* both briefly mention techniques to handle complete wiring blockage. In practice, blockages with absolutely no available tracks are rarely seen. Before global routing, several resources may stake claim to wiring resources which then become fixed for global routing. They include: 1) local wiring on the bottom layers for the internal pin connections for a gate, 2) power grids on multiple layers, 3) pre-routed clock wires, planned buses, or datapaths, 4) and hierarchical logic, memory, or IP blocks. These may already have been completely routed. Even if not, their routes may be hidden from the top-level routing congestion map. These blocks are unlikely to block 100% of the routing resources since generally there should be some routing resources allocated on the top layers. Previous approaches fail to model that essentially every tile of a routing congestion map is neither completely empty or completely full. There is almost always some amount of wiring blockage that the global router must take into account. Thus, it is certainly reasonable for a probabilistic router to do the same. A global router is more likely to route a net in a lower congestion region than in a higher one; our proposed probabilistic router achieves the same behavior. We also propose a new method to improve the complexity of generating the probabilistic congestion map proposed in [2].

The remainder of the paper is as follows. Section II overviews previous work in probabilistic routing congestion. Section III presents definitions and notation. Section IV explains how to extend previous work to handle partial wiring blockages; several toy examples are presented to show how the probability computation leads to a more intuitive result. A new fast algorithm for congestion map updating is proposed. Finally, we show our experimental results in Section V.

II. Overview of Previous Work

The work of Westra *et al.* [2] adopts a scheme similar to that of Lou *et al.* [1]. Probabilistic routing analysis is done exclusively for two-pin nets. Multi-pin nets are broken up into sets of two-pin nets by constructing either a minimum spanning tree (MST) or rectilinear steiner tree (RST) over the pins. Also both approaches do not model detours since they account for a small fraction of the total global routing wire length. Lou *et al.* [1] consider all possible detour free paths between two pins, while Westra *et al.* only consider L-shaped and Z-shaped for non-obvious routes. For their testcases, only 1.2% of the two-pin nets have more than two bends, meaning only that percentage of nets did not have an L- or Z- shape. Westra *et al.*'s model suggests high probabilistic routing usage along the boundary of a net's bounding box.

Westra *et al.* classifies nets into four different categories: short nets, flat nets, L-shaped nets, and Z-shaped nets. A short net is a net that locates in one tile. A flat net spans at most one tile in either the vertical or horizontal direction. An L-shaped net and a Z-shaped net spans more than one tile in either direction and have one and two bends, respectively. See Figure 1(a) for examples of each. For each net, a number of likely paths is considered and the probabilistic usages are assigned to each bucket in the bounding box of the net. The net can be seen as spreading over its possible paths where each path has the same probability. Routing congestion is defined as the ratio of usage to capacity. If there are partial blockages in some buckets, the usage of these buckets are not changed at all. Instead, Westra *et al.* apply a simple model in which the number of blocked tracks are subtracted from the capacity of a bucket.

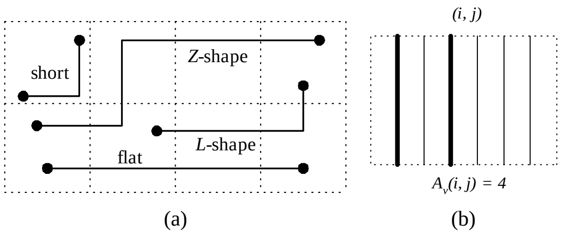


Fig. 1. (a) Different nets considered in this paper. (b) The bucket $(i,\ j)$ with total vertical capacity 6 and partial blockages 2 (darken lines). The available vertical capacity is 4.

III. Definitions

We divide the layout into rows and columns of *buckets* (or tiles or bins). Each bucket contains a set of wiring tracks; the bucket size can be varied to trade off speed for solution quality. Following previous works, the height and width of all buckets are assumed to be same, and defined as H and W. Let the distance between the left, right, bottom, and top border of a bucket and a pin p be denoted by l_p, r_p, b_p, and t_p, respectively. The pin's coordinates are given by (x_p, y_p). A 2-pin net f may span a number of buckets. The number of horizontal and vertical buckets net f spanned are defined as the width and height of the bounding box containing f, respectively.

Let $U_h^f(i,j)$ and $U_v^f(i,j)$ be the horizontal and vertical usages due to a net f in the bucket with coordinates $(i,\ j)$, with i being the coordinates in vertical direction. Define the available horizontal and vertical capacity of the bucket $(i,\ j)$ as $A_h(i,j)$ and $A_v(i,j)$, where $A_h(i,j)$ and $A_v(i,j)$ are the number of available horizontal and vertical wiring tracks inside bucket (i,j). Fig 1(b) shows one bucket with $A_v(i,j) = 4$. The (probabilistic) horizontal and vertical congestion of a bucket is defined to be the ratio of its total (probabilistic) usages contributed by all nets to its available capacity. We use the term congestion or usage to denote the average of horizontal and vertical congestion or usage.

Following the convention of [1] and [2], assume that for analysis purposes, two pins of a net lie in the lower-left and upper-right corners of their bounding boxes.

IV. Handling Partial Wiring Blockage

We now present the probabilistic computation for the four different classes of nets: short, flat, L-shaped, and Z-shaped.

A. Short Nets

A short net is a net that two pins are located in one tile. We handle short nets in the same way as in [2]. Figure 2(a) illustrates the usage computation.

B. Flat Nets

A flat net spans at most one tile in either the vertical or horizontal direction, and is defined as a horizontal or vertical flat net.

Consider a vertical flat net that spans from $(i,\,j)$ to $(i{+}k,\,j)$, as shown in Figure 2(b). The vertical usage of the each bucket is as same as in [2], $U_v(i,j) = t_a/H$, $U_v(i+k,j) = b_b/H$, $U_v(i+m,j) = 1$, $0 < m < k$. Note that the wire traveling through the bucket will occupy a full vertical track.

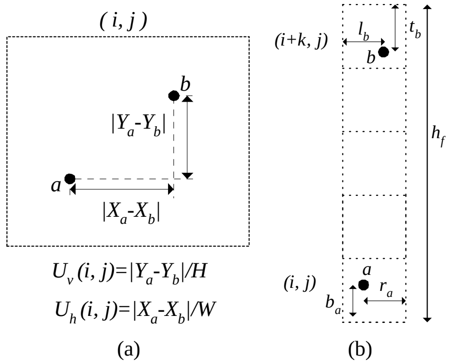


Fig. 2. (a) A short net. (b) A vertical flat net.

The short horizontal route between a and b could occur in any of the buckets. We assume that it is more likely to occur in a bucket with less partial routing blockage. We choose to make it proportional to available routing capacity,

$$U_h(i+m,j) \quad = \quad \frac{|x_a - x_b| \cdot A_h(i+m,j)}{W \cdot S}, 0 \leq m \leq k$$

where $S = \sum_{m=0}^{k} A_h(i+m,j)$. Note that if one assumes equal bucket capacities, the formula becomes $U_h(i+m,j) = |x_a - x_b|/(W \cdot h_f)$, which was given in [2].

C. L-Shaped Nets

Nets with $w_f > 1$ and $h_f > 1$ need to have at least one bend. An L-shaped route has two possible configurations as shown in Figure 3. Let α be the probability of using route A and $1-\alpha$ be the probability of using Route B. In [2], α was implicitly assumed to be 0.5. Clearly, if one of the two routes has more wiring blockage than the other, this needs to take an alternative value.

Let S_{hA} and S_{vA} be the number of available tracks at the horizontal and vertical directions for the L-shaped route A shown in Fig. 3, where $S_{hA} = \min_{0 \leq n \leq l} A_h(i, j+n)$, $S_{vA} = \min_{0 \leq m \leq k} A_v(i+m, j+l)$. Let S_{hB} and S_{vB} be the number of available tracks at the horizontal and vertical directions for another L-shaped route B derived similarly, then $\alpha = \min(S_{hA}, S_{vA})/(\min(S_{hA}, S_{vA}) + \min(S_{hB}, S_{vB}))$.

The horizontal and vertical usage of each bucket on A route in Fig. 3 are as same as in [2], where $U_h(i,j) = r_a/W$, $U_h(i,j+l) = l_b/W$, $U_v(i,j+l) = t_a/H$, $U_v(i+k,j+l) = b_b/H$, $U_h(i,j+n) = 1$, $0 < n < l$, and $U_v(i+m,j+l) = 1$, $0 < m < k$.

As mentioned in [2], L-shapes with $w_f = h_f = 2$ are different from other L-shapes since no Z-shape is possible.

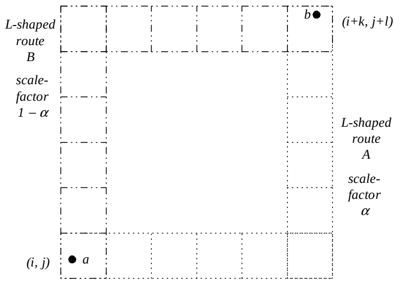


Fig. 3. An L-shaped net

D. Z-Shaped Nets

Z-shapes are the most complicated case. If both w_f and h_f are greater than 2, we can have two orientations: horizontal and vertical, named after their orientation of the center piece of the Z-shapes.

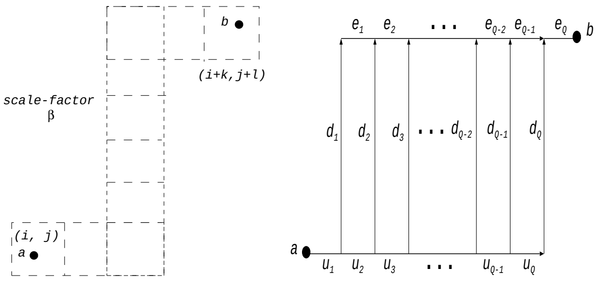
Now let consider the usage of vertical Z-shapes as shown in Fig. 4(a). The routing graph which shows all possible choices of the vertical Z-shape in Fig. 4(a) is also shown in Fig. 4(b), which is a directed graph. In the figure, $Q = w_f - 2$, each edge u_n corresponds a part of the net entering from the bucket (i, $j+n-1$) to the bucket (i, $j+n$), each edge d_n corresponds a part of the net entering from the bucket (i, $j+n$) to the bucket ($i+k$, $j+n$), and each edge e_n corresponds a part of the net entering from the bucket ($i+k$, $j+n$) to the bucket ($i+k$, $j+n+1$), $n = 1, \ldots, Q$. There is a capacity F associated with each edge, where

$$\begin{aligned} F(u_n) &= \min(A_h(i,j+n-1), A_h(i,j+n)), \\ F(d_n) &= \min_{0 \le m \le l} A_v(i+m,j+n), \\ F(e_n) &= \min(A_h(i+k,j+n), A_h(i+k,j+n+1)). \end{aligned}$$

Given Q possible routes $R(n) = \{u_1, \ldots, u_n, d_n, e_n, \ldots, e_Q\}$, $n = 1, \ldots, Q$, a vertical Z-shaped net routed by global router must be one of the shape. Define $P(n)$ as the probability of a vertical Z-shaped net routed with the shape $R(n)$. Then $P(n)$ must satisfy the following properties: 1) $0 \le P(n) \le 1$, $n = 1, \ldots, Q$; 2) $\sum_{n=1}^{Q} P(n) = 1$. Suppose all $P(n)$s are already known, then the usage of each bucket can be derived as follows.

For the leftmost bucket in the bottom row, the horizontal and the vertical cost are $U_h(i,j) = r_a/W$, $U_v(i,j) = 0$. For the other buckets in the bottom row, the horizontal usage consists of two terms: the first for the case the vertical segment $d(n)$ on route $R(n)$ will start in this bucket. The horizontal usage in that case would be 0.5 since the bend would on average be in the middle of the bucket. The other term is for the case $d(n)$ is at the *right* of the bucket and the usage would be 1. Only if the bend occurs in a bucket, it gets vertical usage. The total vertical usage of t_a/H is spread over the candidate buckets. Therefore, we have $U_h(i,j+n) = 1/2P(n) + P_R(n)$, $U_v(i,j+n) = t_a \cdot P(n)/H$, $1 \le n < l$, where $P_R(n) = \sum_{m=n+1}^{Q} P(m)$.

For the top row, the reasoning is similar. Horizontal bucket usage consists of two terms. One for the case that the bend occurs in that bucket, and one for the case the bend occurs to its left. Vertical usage is spread over the buckets. The horizontal and vertical cost for these buckets are $U_h(i+k,j+l) = l_b/W$, $U_v(i+k,j+l) = 0$, $U_h(i+k,j+n) = 1/2P(n) + P_L(n)$, $1 \le n < l$, and $U_v(i+k,j+n) = b_b \cdot P(n)/H$, $1 \le n < l$. where $P_L(n) = \sum_{m=1}^{n-1} P(m)$.


b
(i+k, j+l)
scale-factor β
(i, j)
a
e_1 e_2 ... e_{Q-2} e_{Q-1} e_Q
b
d_1 d_2 d_3 ... d_{Q-2} d_{Q-1} d_Q
a
u_1 u_2 u_3 ... u_{Q-1} u_Q

(a) (b)

Fig. 4. (a) A vertical Z-shaped net. (b) The routing graph of Z-shaped net.

The horizontal and vertical usage for bucket ($i+m$, $j+n$) in the center of the netbox are

$$\begin{aligned} U_v(i+m,j+n) &= P(n), \\ U_h(i+m,j+n) &= 0, \end{aligned} \tag{1}$$

where $0 < m < k$, $0 < n < l$.

Define S_v and S_h as the total capacities of vertically and horizontally oriented Z-shapes. In order to find the total Z-*usage*, we therefore scale the horizontal and vertical Z-usages with $\beta = S_h/(S_h + S_v)$ and $1 - \beta = S_v/(S_h + S_v)$, respectively, as we did with a factor α for the L-shapes, and sum them all up.

In previous work [2], $P(n) = 1/Q$ for every route, and $\beta = (w_f - 2)/(w_f + h_f - 4)$. This assumption is only valid when there is no partial blockage. Look at the example in Fig. 5(a), based on the model of [2], each route has same probability $1/3$. However, it is more intuitive for global router to route a net in the area with less wiring blockages.

We propose a new metric to define $P(n)$ taken into consideration of the partial blockage information. Since there are many different assumptions that one could make that lead to different probabilities assigned to routes, we define $P(n)$ based on the minimum capacities of each route. For each vertical Z-shaped route $R(n)$, its probability $P(n)$ is $P(n) = F(n)/S_v$, where $F(n)$ is the minimum capacity of route $R(n)$ and $S_v = \sum_{n=1}^{Q} F(n)$ is the total minimum capacities of the all vertical Z-shaped routes. It is obvious that $F(n)$ is related to the capacities of every edge on route $R(n)$. However, for every edge u_m, its capacity is shared by $Q-m-1$ routes, and for every edge e_m, its capacity is shared by m routes. Therefore, for one specific route $R(n)$, it is hard to know the exact capacities of edges u_m and e_m contributed to $R(n)$. If the capacities of all u and e edges are infinity, $F(n)$ is equal to the capacity of unique edge d_n of each route. In other cases, for each route $R(n)$, $n = 1, \ldots, Q$, we will redistribute the capacity of every edge u_m, $m = 1, \ldots, n$, on route $R(n)$ according to the ratio between the capacity of unique edge d_n of $R(n)$ and the total capacities of all unique edges of routes sharing edge u_m. The capacity of every edge e_m, $m = n, \ldots, Q$ is redistributed in a similar way. After we get new capacity of every edge on route $R(n)$, the minimum capacity $F(n)$ can be easily computed.

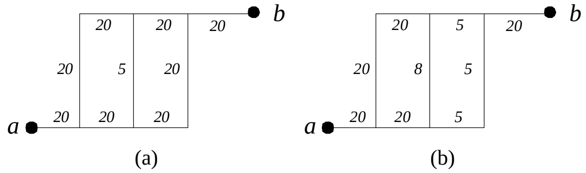


Fig. 5. Two examples of Z-shaped nets. The capacity of each edge is shown on the graph.

For $n = 1, \ldots, Q$, define $F_L(d_n)$ and $F_R(d_n)$ as the total vertical capacities of all edges d_m to the left and right of edge d_n including edge d_n itself, $F_L(d_n) = \sum_{m=1}^{n} F(d_m)$, $F_R(d_n) = \sum_{m=n}^{Q} F(d_m)$. The minimum capacity $F(n)$ of each route $R(n)$ can be computed as

$$\begin{aligned} F(n) &= \min\{F(u_1)\frac{F(d_n)}{F_R(d_1)}, \ldots, F(u_n)\frac{F(d_n)}{F_R(d_n)}, F(d_n), \\ &\quad F(e_n)\frac{F(d_n)}{F_L(d_n)}, \ldots, F(e_Q)\frac{F(d_n)}{F_L(d_Q)}\}. \end{aligned}$$

We can also rewrite as $F(n) = F(d_n) \cdot \min\{K_u(n), 1, K_e(n)\}$. where

$$\begin{aligned} K_u(n) &= \min\{\frac{F(u_1)}{F_R(d_1)}, \ldots, \frac{F(u_n)}{F_R(d_n)}\}, \\ K_e(n) &= \min\{\frac{F(e_n)}{F_L(d_n)}, \ldots, \frac{F(e_Q)}{F_L(d_Q)}\}. \end{aligned}$$

Similar analysis can be done for horizontal Z-shaped routes, and S_h and β can be derived accordingly. It is easy to prove when there is no partial blockages and the capacities of each bucket are same, our model will give exactly the same $P(n)$ as in [2].

The example in Fig. 5(a) is used to illustrate our algorithm. It is easy to get $F_L(d_1) = F_R(d_3) = 20$, $F_L(d_2) = F_R(d_2) = 25$, $F_L(d_3) = F_R(d_1) = 45$.

$$\begin{aligned} F(1) &= 20 \cdot \min\left(\frac{20}{45}, 1, 1, \frac{20}{25}, \frac{20}{45}\right) = \frac{400}{45}, \\ F(2) &= 5 \cdot \min\left(\frac{20}{45}, \frac{20}{25}, 1, \frac{20}{25}, \frac{20}{45}\right) = \frac{100}{45}, \\ F(3) &= \frac{400}{45}. \end{aligned}$$

Therefore $P(1) = 4/9$, $P(2) = 1/9$ and $P(3) = 4/9$. Compared to the method in [2], the probability of path $R(2)$ is reduced by 2/9 and are evenly distributed to $R(1)$ and $R(3)$. Suppose we try to route 20 two-pin nets where all lower left pins are in the same bucket, and all upper right pins are in the same buckets. The vertical probabilistic congestion of d_1, d_2 and d_3 are $4/9 \cdot 20/20 = 4/9$, $1/9 \cdot 20/5 = 4/9$, and $4/9 \cdot 20/20 = 4/9$, respectively. The congestion of d_1, d_2 and d_3 derived from previous method [2] are $1/3$, $4/3$ and $1/3$, respectively. Our model predicts that the congestion of all three routes are equal. This is more likely to happen since this solution leaves evenly space for each route and predicts the router will take less probability to route this Z-shaped net with the second route. We expect a clever enough global router behaves more like our predicted model than the one proposed in [2].

For another example in Fig. 5(b), with similar analysis, we can get $P(1) = 0.4447$, $P(2) = 0.1779$ and $P(3) = 0.3774$. Suppose we try to route 10 two-pin nets where all lower left pins are in the same bucket, and all upper right pins are in the same buckets. The horizontal probabilistic congestion of e_2 (the edge on the top row with capacity 5) is 0.996 compared to 1.067 derived from [2]. Our metric will put more routes on the third route than the second since the first route occupies most of tracks of e_2 and therefore the probability of the second route is decreased.

E. Algorithm and Fast Map Updating

Following the work of [2], it is necessary to decide what the relative probabilities of L-shapes versus Z-shapes should be. Let $\gamma = \#nets_L/(\#nets_L + \#nets_Z)$ be the probability of taking an L-route over a Z-route. The value γ can be chosen by previous design experience, i.e., how many routes are optimally routed or fixed by the designer. The combination probabilistic usages are $U_{LZ} = \gamma U_L + (1 - \gamma)U_Z$.

The algorithm **CPPB** (Congestion Prediction with Partial Blockage) to predict the congestion map of given nets is illustrated as follows.

```
Algorithm CPPB
Input Nets Set $N$
Output The Congestion Map for $N$
Begin
Create maps with loaded partial wiring blockage
    For each net $f \in N$
        pin-pairs = MST(n) or pin-pairs = RST(n)
    For each pin-pair $(a, b)$
        If $(a, b)$ is a short net
            update short-congestion($(a, b)$)
        Else if $(a, b)$ is a flat net
            update flat-congestion($(a, b)$)
        Else
            update $\gamma$ × $L$-shape-congestion($(a, b)$)
            update $(1 - \gamma)$ × $Z$-shape-congestion($(a, b)$)
```

It is easy to see that for short nets, flat nets and L-shape nets the algorithm takes linear time with respect to the maximum between horizontal buckets and vertical buckets the net spanned, $\max(w_f, h_f)$. For Z-shaped nets, with dynamic programming it takes $O(\max(w_f, h_f))$ in the bounding box to compute all $F_L(d_n)$, $F_R(d_n)$, $K_u(n)$, $K_e(n)$, $F(n)$, $P(n)$,

$P_L(n)$ and $P_R(n)$s for both vertical and horizontal Z-shaped nets, i.e., $F_L(d_n) = F_L(d_n - 1) + F(d_1)$. Also, it takes $O(\max(w_f, h_f))$ to update the usage value of the top and bottom rows for the vertical Z-shaped net, and the left and right columns for the horizontal Z-shaped net. However, it still takes $O(w_f \cdot h_f)$ time to update the usage value for all other buckets. Thus the total complexity is $O(\#nets \cdot \#buckets)$, which is the same as the one proposed in [2].

We propose a new algorithm to improve this complexity and the one in [2]. Using vertical Z-shaped net as example, we know that the horizontal usage of all center buckets are zero. The vertical usage for all buckets on edge d_n are the same as show in Eq. 1, which is $P(n)$. Instead of updating them explicitely, for all buckets in one column, we can create a temporary usage map with initial value zero in every buckets, and store a positive value, e.g., $P(n)$ before bucket ($i+k+1$, $j+n$), and a negative value, $-P(n)$ after bucket (i, $j+n$). After the temporary map is derived, the usage of each bucket in the final map can be derived by scanning from the top bucket in the temporary map and summing up all usage values in the temporary map before this bucket. Note that the tempory map has one more row and one more column than original map since all values stored in the tempory map exist in the interval of every two buckets in the final usage map and for n buckets, there are $n+1$ intervals. The usage value for buckets in short nets, flat nets, L-shaped nets and the buckets in the peripheral of Z-shaped nets can be directly stored in the final usage map. Since the usage updating process for all center buckets in Z-shaped nets can be done in $O(\max(w_f, h_f))$ with this tempory map method, the total algorithm complexity is $O(\#nets \cdot \max(h_f, w_f) + MapSize)$, where $MapSize$ is the total number of buckets in the whole map. Our new algorithm dramaticaly speeds up the congestion map construction when there are many global nets across the chip.

It should be pointed out that for a high congested design, detour can not be avoided, and the assumption of Westra *et al.* may be not valid. Global router also need several iterations to reroute the detour net. Note that probabilistic approach can be extended to handle detour net by multiple iterations. The high congested area is identified. In next iteration, all nets going over that area are identified. For each net, the bounding box are expanded (like creating two qseudo pins in the corner of expanding box) and the congestion of each bucket in this expanding box are recomputed.

V. Experimental Results

The probabilistic congestion router was implemented in C++ in an AIX environment on an IBM RX6000 machine (1.1 GHz) with 2G memory. For our experiments we consider two designs, Chip A, a random logic macro with 60801 routable nets and Chip B, an industrial ASIC with 225100 routable nets. The size of buckets for Chip A and B are 22500 and 172638, respectively. Before performing global routing or probabilistic routing estimation, we first load the wiring blockage map, which includes wiring for internal pin connections, power grid, IP blocks, and I/O cells. Figures 7(a) and 8(a) show the wiring blockages before routing. Brighter color means more blockages (part with yellow color has more blockage than green). Note how clearly partial wiring blockages caused by macro blocks can be seen for Chip A. For Chip B the wiring blockage is dispersed across blockages in the middle of the chip and C4 pads.

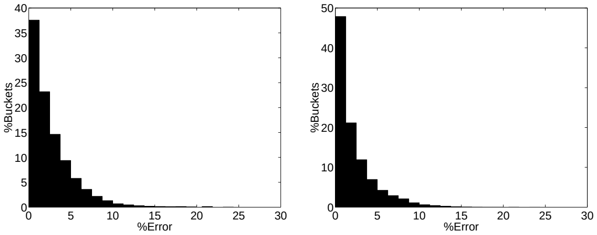
To construct the predicted probabilistic map, we use a γ value of 0.6, which is also the average ratio of L routes to Z routes observed by [2]. Figures 7(b) and 8(b) show the results from the probabilistic congestion map predicted by our method while Figures 7(c) and 8(c) show the usage constructed by the industrial global router. We use MST to break multi-pin nets to 2-pin nets. The comparison between these maps shows that our predicted congestion and the real congestion seen by global router match quite well. For each bucket, we compute the relative error of the probabilistic congestion map relative to the global router map. The error histograms for these sets of buckets are shown in Figure 6. We observe that in both cases 85% of the buckets have relative error less than 5% error and that less than 1% of the buckets have more than 15% error.

The running time of Chip B for different bucket sizes is shown in Table I. We can see it is linear with respect to the number of buckets.

TABLE I

The CPU time of Chip B for different bucket sizes. The bucket size is repsented by #columns × #rows

Grid size	Running time (second)
97 × 96	31
278 × 276	45
417 × 414	68
556 × 552	100
834 × 828	189
1112 × 1105	311



(a)

(b)

Fig. 6. Error Distribution for (a) Chip A and (b) Chip B

References

[1] J. Lou, S. Krishnamoorthy, and H. S. Sheng, "Estimating routing congestion using probabilistic analysis," in *Proceedings of ISPD*, pp. 112–117, 2001.

[2] J. Westra, C. Bartels, and P. Groeneveld, "Probabilistic congestion prediction," in *Proceedings of ISPD*, pp. 204–209, 2004.

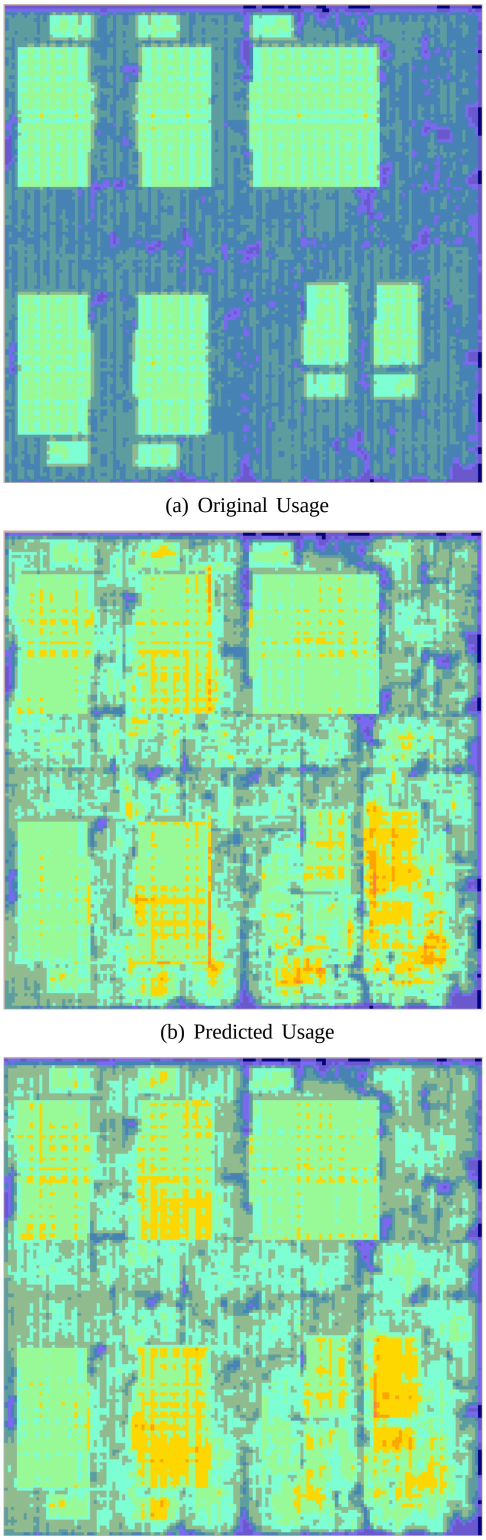
[3] C.-L. E. Cheng, "Risa: accurate and efficient placement routability modeling," in *Proceedings of ICCAD*, pp. 690–695, 1994.

[4] M. Wang, X. Yang, K. Eguro, and M. Sarrafzadeh, "Multi-center congestion estimation and minimization during placement," in *Proceedings of ISPD*, pp. 147–152, 2000.

[5] J. Cong and P. H. Madden, "Performance driven multi-layer general area routing for pcb/mcm designs," in *Proceedings of DAC*, pp. 356–361, 1998.

[6] H.-M. Chen, H. Zhou, F. Y. YOung, D. F. Wong, H. H. Yang, and N. Sherwani, "Integrated floorplanning and interconnect planning," in *Proceedings of ICCAD*, pp. 354–357, 1999.

[7] Y. Ma, X. Hong, S. Dong, S. Chen, Y. Cai, C. K. Cheng, and J. Gu, "Dynamic global buffer planning optimization based on detail block locating and congestion analysis," in *Proceedings of DAC*, pp. 806–811, 2003.

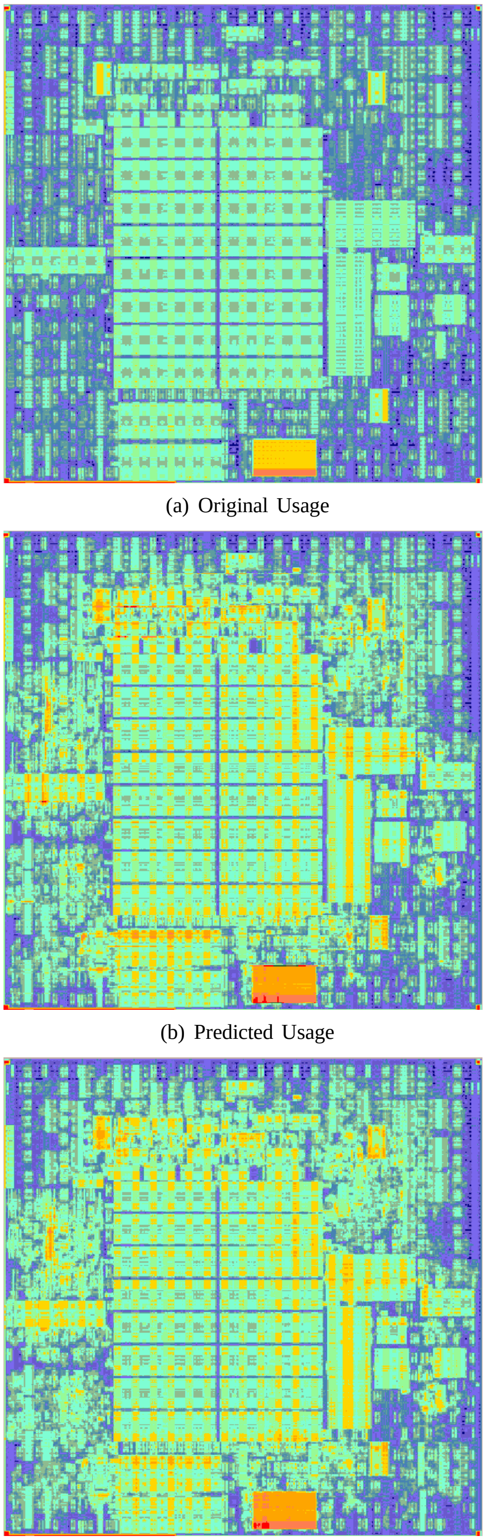


(a) Original Usage

(b) Predicted Usage

(c) Usage according to global router

Fig. 7. Usage map for Chip A



(a) Original Usage

(b) Predicted Usage

(c) Usage according to global router

Fig. 8. Usage map for Chip B