Optimization Letters (2007) 1:281–297
DOI 10.1007/s11590-006-0027-0

ORIGINAL PAPER

An ILP based hierarchical global routing approach for VLSI ASIC design

Zhen Yang · Anthony Vannelli · Shawki Areibi

Accepted: 15 August 2006 /
Published online: 27 October 2006

Abstract The use of integrated circuits in high-performance computing, telecommunications, and consumer electronics has been growing at a very fast pace. The level of integration as measured by the number of logic gates in a chip has been steadily rising due to the rapid progress in processing and interconnect technology. The interconnect delay in VLSI circuits has become a critical determiner of circuit performance. As a result, circuit layout is starting to play a more important role in today's chip designs. Global routing is one of the key sub-problems of circuit layout which involves finding an approximate path for the wires connecting the elements of the circuit without violating resource constraints. In this paper, several integer programming (ILP) based global routing models are fully investigated and explored. The resulting ILP problem is relaxed and solved as a linear programming (LP) problem followed by a rounding heuristic to obtain an integer solution. Experimental results obtained show that the proposed combined WVEM (wirelength, via, edge capacity) model can optimize several global routing objectives simultaneously and effectively. In addition, several hierarchical methods are combined with the proposed flat ILP based global router to reduce the CPU time by about 66% on average for edge capacity model (ECM).

Z. Yang (✉) · A. Vannelli
Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada N2L 3G1
e-mail: z6yang@uwaterloo.ca

A. Vannelli
e-mail: vannelli@uwaterloo.ca

S. Areibi
Engineering System and Computing, University of Guelph, Guelph, ON, Canada N1G 2W1
e-mail: sareibi@uoguelph.ca

Keywords VLSI physical design · Standard cell global routing · Integer Linear Programming

1 Introduction

A VLSI chip can today contain hundreds of millions of transistors and is expected to contain more than 1 billion transistors in the next decade (7). In order to handle this rapid growth in integration technology, the design procedure is therefore divided into a sequence of design steps. VLSI circuit layout is the design step in which a physical realization of a circuit is obtained from its functional description. Due to the inherent complexity of circuit layout it is often decomposed into several distinct sub-problems, such as partitioning, placement, and routing. The routing problem is further divided into two phases: global routing and detailed routing phase. In the global routing phase, a "rough" route for each net is generated without specifying the actual geometric layout of wires, whereas in the detailed routing, the actual geometric layout of each net within the assigned routing regions is established.

Since all versions of the global routing problem are NP-hard (11), a variety of heuristic algorithms have been developed for it. Basically, these algorithms can be classified into two classes: *Sequential global routing* and *Concurrent global routing*. The most common approach to global routing is sequential routing.[1] In such an approach, nets are first ordered according to their importance, then based on the ordering nets are routed sequentially. The quality of a sequential global router largely depends on the ordering of nets. Due to the sequential nature of these techniques, they fail to give adequate results. Besides, the sequential heuristic techniques cannot provide a key answer as to whether or not a feasible solution exists. In other words, if they fail to find a feasible solution, it is not clear whether this is attributable to the non-existence of a feasible solution or due to shortcomings of the heuristic. Moreover, when a heuristic does find a feasible solution, it is not known whether or not this solution is optimal, or how far it is from optimality. To avoid the net ordering problem and make the solution more predictable, concurrent based global routing algorithms in the form of integer programming models have been developed to route all the nets simultaneously. In the mathematical programming based approach, global routing is formulated as a 0/1 integer programming (IP) problem. Given a set of Steiner trees (13) for each net and a routing graph, the objective of the IP technique is to select a Steiner tree for each net from its set of Steiner trees without violating the channel capacities while minimizing the total wirelength. This approach tends to result in a more global solution and no initial ordering of the nets is required. In this paper, several mathematical programming based global routing models are investigated. Also, different hierarchical methods are combined with the mathematical programming based router to improve the computation time. One of the main contributions of this paper is the

[1] Most industrial tools utilize Maze Routers as a solver.

integration of an ILP based approach within a top-down and bottom-up approach which differs from traditional methods that seek to use Maze routers within the flow (8).

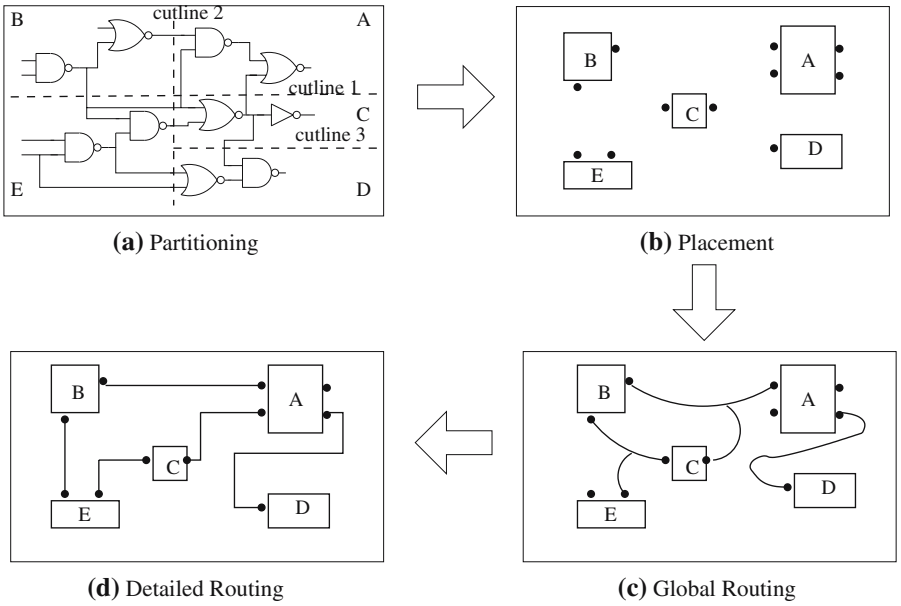
The remainder of the paper is organized as follows. Section 2 presents the global routing problem and different global routing objectives. In Sect. 3 an ILP based global routing method with different formulations is introduced. Section 4 introduces two hierarchical global routing methods and gives the experimental results. Finally, Sect. 5 summarizes the paper and proposes possible future work.

2 Background

VLSI physical design automation is a process of determining the location of devices and their connectivity inside the boundary of a VLSI chip. It is one of many interrelated complex tasks in VLSI circuit design. Not surprisingly, this complex task is handled by dividing the original task into more tractable sub-tasks such that a physical design can be realized in reasonable amount of time. The different stages of physical design cycle are shown in Fig. 1. Initially, a large circuit is divided into a collection of smaller modules according to some criteria through a partitioning phase. Following that, the placement step constructs a layout indicating the positions of the cells so that all the nets can be routed and the total layout area is minimized. In this stage, the exact locations of cells are determined. Following the placement stage, interconnections between components are physically assigned to allowable routing regions by global and detailed routers. To reduce the complexity of physical design, restricted design styles are used. Standard-cell layout style is a topology that provides a good design time and production size due to its pre-designed standard cell library and regular structure. In this paper, we mainly focus on the global routing problem for standard-cell layout style.

2.1 Global routing problems

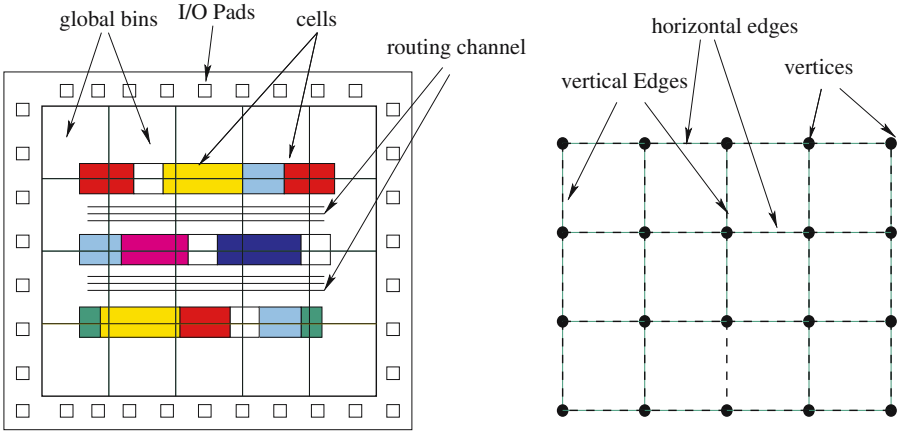
The goal of global routing is to decide the connection pattern for each net that satisfies different objectives. The input to the global routing problem consists of a net-list that indicates the interconnections between terminals and placement information including the terminal positions and the location of routing channels in between them. The global routing problem is typically presented as a graph problem, where the routing regions and the module connections are modeled using a grid graph. Initially, a given circuit is partitioned into a set of rectangular regions, called global bins. The cells are placed inside these bins and each cell is assumed to be placed in the center of the global bin, as shown in Fig. 2a. It is easy to see that the global bins and edges can be transformed into a grid graph (Fig. 2b). The vertices of the graph represent possible positions of net terminals, and the horizontal and vertical edges (called grid edge) that



(a) Partitioning **(b)** Placement

(c) Global Routing **(d)** Detailed Routing

Fig. 1 Physical design cycle



(a) Global Bin Graph **(b)** The corresponding Grid Graph

Fig. 2 Grid graph for standard-cell layout style

lie between two adjacent vertices represent channels along or regions through which wiring can be routed. A net is an unordered set of points on the grid graph. A route (or tree) of a net is a set of grid edges used to connect all the terminals of the net. As there is finite routing resources, each grid edge has a capacity. With such a graph representation, one can solve the graph version of the global routing problem instead of the original problem.

2.2 Global routing objectives

Every global routing method depends on the evaluation metric employed to measure the goodness of the technique. There are two traditional primary objectives in the global routing problem: minimizing chip area and achieving routable designs.

For the standard-cell layout style, since the total chip area is approximately equal to the area of the modules plus the area occupied by the interconnect, minimizing the wire-length is approximately equivalent to minimizing the chip area. Usually, a Rectilinear Steiner tree or spanning tree based wire-length is used in global routing. As the routing layers tend to increase to more than two, the number of vias or bends plays a vital role in the routability and timing of the nets. Minimizing the number of vias tends to reduce the number of metal contacts in the routing, therefore, improving the performance of the circuit. To minimize the number of vias in a layout, admissible routes in the global routing are limited to those that have lower number of bends. The total number of bends of a tree is equivalent to the sum of the number of bends between any two terminals, which equals to the route direction change from horizontal to vertical or vice versa. Congestion minimization is another important objective of global routing. To examine the effectiveness of the congestion minimization, different measures are used to evaluate the results. The most commonly used congestion metrics are congestion map, the maximum routing density of all the grid edges, and the total overflow (2,6,9). A congestion map visually plots the congestion in the design by assigning different colours to different congestion costs. A lighter colour usually means a higher congestion cost. For a given routing graph $G = (V, E)$, the *routing demand* d_e of edge e, is the number of wires crossing e, the *routing supply* s_e is the number of wires that are allowed to cross edge e. The difference between routing demand and supply is formally described as:

$$overflow_e = \begin{cases} d_e - s_e & \text{if } d_e > s_e; \\ 0 & \text{if } d_e \leq s_e. \end{cases}$$

The total overflow of a layout is defined as the summation of the overflow for all global edges. The amount of total overflow reflects the amount of total shortage of routing resources in the layout. The maximum routing density $\mathrm{Max}_{\mathrm{rd}}$ can be described as: $\mathrm{Max}_{\mathrm{rd}} = \max_{e \in E}\{d_e\}$.

3 ILP based global routing models

In (1), the research work focuses on enhancing the solution of the global routing problem by using an ILP approach. Figure 3 shows the main steps of the approach proposed in (1). The first important step in the ILP based global routing approach attempts to produce a set of admissible routes for each net.

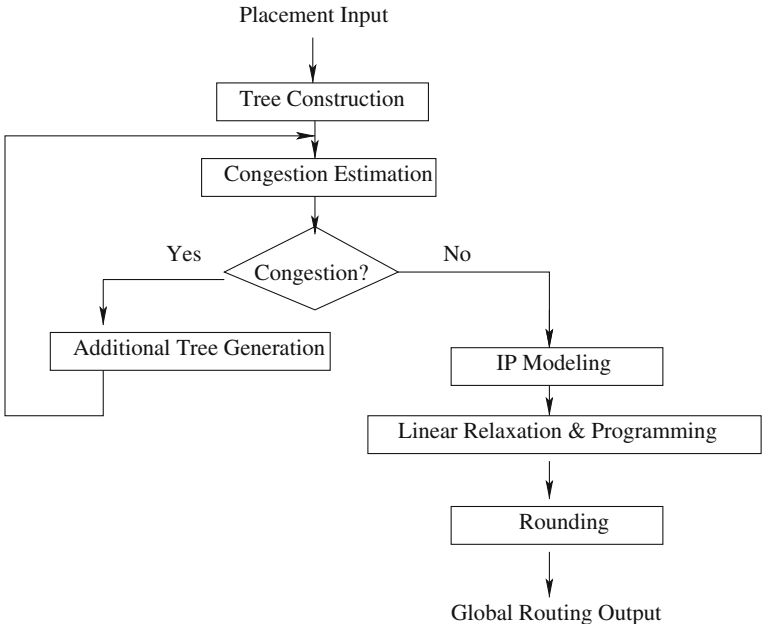


Fig. 3 Flow chart of the global routing methodology

In the practical circuit, the terminals in a net are connected by the horizontal and vertical wires. Therefore, only the Rectilinear Spanning Trees or Rectilinear Steiner Trees are considered in the tree construction process. These trees become the unknown variables of the ILP problem. Obviously, the number of trees for each net should not be too large, since the size of the ILP problem is a function of the number of trees. On the other hand, a number of trees should be built for each net to guarantee the feasibility of the problem. To remedy this problem, an additional tree generation step is proposed in (1) to reduce the number of trees created for each net, while ensuring that the constructed trees will likely result in a promising (feasible and optimal) solution. Initially, the potentially congested areas in the routing graph are predicted by a heuristic technique (1) in the congestion estimation stage. This priori congestion information is then used in the additional tree generation stage to eliminate the congested areas by adding trees to the nets passing through these areas iteratively. The congestion of the circuit is re-estimated after each additional tree generation step. The time complexity of the global routing methodology is $O(n)$, where n is the total number of routed nets. Due to the large size of today's VLSI problems, and the fact that they are known to be NP-hard, solving the ILP problem using traditional method, e.g., branch and bound, can be impractical. Thus, the ILP is typically relaxed and solved as a linear programming (LP) problem followed by rounding heuristics to obtain an integer solution.

3.1 Wire-length model (WLM)

For the Wire-length minimization (WLM model) (12), the global routing problem is formulated as a 0/1 BIP problem by associating a variable x_i with each tree which connects a net. The variable x_i equals "1" if that particular tree is selected and "0" otherwise. All the possible ways to connect nets in the two routing layers are represented by a (0,1) matrix $[a_{ij}]$, with the ith row corresponding to the ith edge in the grid graph and each column corresponding to different ways of connecting a net. The element in a_{ij} is expressed as:

$$a_{ij} = \begin{cases} 1 & \text{if tree } j \text{ passes through edge } i; \\ 0 & \text{otherwise.} \end{cases}$$

The capacity of each edge is represented by the following constraint:

$$\sum_{j=1}^{t} a_{ij} x_j \leq c_i, \quad i \in \{1, \ldots, p\}$$

where p is the number of edges on the grid graph, t is the total number of trees produced for all the nets, and c_i is the edge capacity of the ith edge. In this paper, the edge capacity c_i is represented by a single variable "$z_{\max}$" for all the edges. The global routing problem with the objective of minimizing the total wire-length is then formulated as:

$$\text{Minimize} \sum_{j=1}^{t} w_{lj} x_j \tag{1}$$

Subject to

$$\sum_{x_j \in N_k} x_j = 1, \quad k \in \{1, \ldots, n\}; \textit{ selection of one tree for each net} \tag{2}$$

$$\sum_{j=1}^{t} a_{ij} x_j \leq z_{\max}, \quad i \in \{1, \ldots, p\} \tag{3}$$

$$x_j \in \{0, 1\} \quad j \in \{1, \ldots, t\} \tag{4}$$

$$z_{\max} \in \{0, C\} \tag{5}$$

where w_{lj} represents the weight associated with the length of tree j. This weight is calculated as the normalized length of tree j with respect to the rest of the trees in net k:

$$w_{lj} = \frac{\text{length of tree } j}{\text{max length of trees constructed for net } k} \tag{6}$$

3.2 ECM and VMM models

The constraints of edge capacity model (ECM) and Via minimization model (VMM) are similar to those used in the WLM model. The only difference is the objective function. For the ECM model, the objective function is

$$\text{Minimize} \quad Z_{\max}$$

where $z_{\max}$ represents the maximum capacity of the edges. For the VMM model the objective function is

$$\text{Minimize} \sum_{j=1}^{t} w_{bj} x_j$$

where w_{bj} is the weighting factor associated with the number of vias (bends) in the path of tree j and is calculated as follows:

$$w_{bj} = \frac{\text{number of bends in tree } j}{\text{max number of bends in trees produced for net } k} \tag{7}$$

3.3 Combined model (WVEM)

To optimize the total wire-length, the number of via and maximum routing density simultaneously, a combined model (WVEM) is proposed in this paper. The objective function of the WVEM model is given by

$$\text{Minimize} \sum_{j=1}^{t} (\beta_l w_{lj} + \beta_b w_{bj}) x_j + \beta_z z_{\max} \tag{8}$$

where w_{lj} and w_{bj} are the weighting factors associated with wire-length and number of bends, respectively. The scalars β_l, β_b and β_z are used to change the emphasis of the respective weighting factors. Our work in (14) summarizes results for the WVEM model with different sets of weighting. Due to the importance of congestion in routing a circuit the scalars have been set as follows: $\beta_l = 1$, $\beta_b = 1$ and $\beta_z = 20$. The results in (14) show that as the weight of maximum channel capacity β_z increases, the maximum routing density is reduced effectively. The WVEM model has the same constraints as the WLM model [shown in Eqs. (2)–(5)].

3.4 Preliminary results

The proposed techniques were implemented in the 'C++' programming language on a 900 MHz Sun Blade 2000 workstation with 1 GIGA Byte memory.

The iLog CPLEX 9.0 package was used to solve the ILP problems. The test circuits used to evaluate the ILP models in this paper are based on the MCNC'91 benchmarks (10). This test set consists of ten circuits ranging in size from 125 to over 25,000 cells. The circuits have been grouped into three categories according to size: small, medium and large.

Table 1 shows a comparison between the WLM, VMM and edge capacity minimization model (ECM) for all test circuits. The columns "Wire", "Bends" and "M_{rd}" list the total wire-length, total number of bends and maximum routing density respectively. The "Imp" entry row in Table 1 shows the deterioration of different parameters in the WLM, VMM and ECM models relative to each other. For example, in the WLM model the "*Bends*" entry shows the deterioration with respect to the VMM model and the "M_{rd}" entry with respect to the ECM model respectively. From this table, it can be seen that each model can improve the corresponding objective effectively. To solve the global routing problem with several competing objectives, the combined model (WVEM) incorporates the design factors into a single objective function as penalty terms. For the remainder of the paper we will only show the experimental results of large size circuits due to interest in scalability issues.

Table 2 compares ECM with the combined model (WVEM) for large size circuits. Row "Imp" shows the average percentage improvement. As shown in this table, the WVEM model produces better results in terms of total number of bends and CPU time. The total wire-length and maximum routing density are also improved by 6 and 4%, respectively.

Table 3 compares the maze router with the proposed ILP based global router for large size circuits. As shown in this table, the ILP based router produces better results than those obtained via the maze router on the total wire-length and total number of bends, especially for large circuits such as "avq.small" and "avq.large". The maximum routing density is only deteriorated by about 7% on average.

4 Hierarchical global routing

The flat ILP based global routing approach described earlier tends to encounter a scaling problem as circuits increase in size. Consequently, a hierarchical global routing becomes essential for the design flow as shown in Fig. 4. Inspired by the successful application of multilevel methods in circuit partitioning and placement a top-down hierarchical routing algorithm would attempt to use a hierarchy on the routing graph to decompose the large routing problem into smaller and more manageable pieces. There are several approaches to hierarchical global routing (14): top-down hierarchical routing, bottom-up hierarchical routing and multilevel global routing (3–5). All of these methods attempt to solve the scaling problem of large designs by building multilevel hierarchical representations of the routing regions.

Table 1 Results comparison of WLM, VMM and ECM models

LP relaxation + rounding results for WLM, VMM and ECM models														
Circuit	Total	Total	WLM model				VMM model				ECM model			
	Cells	Nets	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time
Fract	125	147	27,517	99	9	2	28,143	85	10	2	28,943	120	5	2
Prim1	833	876	644,423	609	17	15	651,485	540	15	15	669,328	692	8	15
Struct	1,952	1,920	320,663	495	10	5	321,915	425	12	4	337,900	633	7	5
Imp	–	–	–	−13%	−44%	−4%	−1%	–	−46%	–	−4%	−27%	–	−4%
Ind1	3,085	2,478	986,911	1,098	25	30	989,845	1,015	26	31	1,020,282	1,314	17	30
Prim2	3,121	3,136	3,187,597	2,493	24	78	3,197,111	2,266	23	74	3,253,505	2,737	15	75
Bio	6,514	5,742	1,032,444	2,093	12	91	1,039,595	1,865	12	88	1,046,962	2,160	9	92
Imp	–	–	–	−9%	−33%	−3%	0%	–	−33%	–	−2%	−17%	–	−2%
Ind2	12,637	13,419	12,091,689	9,490	21	1,512	12,114,767	8,272	25	1,510	12,832,210	12,137	15	3,756
Ind3	15,433	21,938	47,205,901	16,457	33	1,493	47,395,661	14,023	33	1,497	50,311,413	21,164	32	3,531
avq.s	21,918	22,124	9,096,280	12,146	21	2,801	9,122,104	10,076	20	2,793	9,884,393	16,848	13	7,574
avq.l	25,178	25,384	10,411,364	12,994	23	3,080	10,446,320	10,831	18	3,040	11,121,285	17,198	13	12,921
Imp	–	–	–	−16%	−28%	0%	0%	–	−27%	–	−7%	−35%	–	−68%

Table 2 Results comparison of ECM model and WVEM model

LP relaxation + rounding results for ECM model and WVEM model										
Circuit	Total	Total	ECM model				WVEM model			
	Cells	Nets	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time
Ind2	12,637	13,419	12,832,210	12,137	15	3,756	12,100,759	8,299	13	1,523
Ind3	15,433	21,938	50,311,413	21,164	32	3,531	47,308,679	14,037	32	1,527
Imp	–	–	–	–	–	–	6%	33%	4%	58%
avq.small	21,918	22,124	9,884,393	16,848	13	7,574	9,112,536	10,109	13	2,840
avq.large	25,178	25,384	11,121,285	17,198	13	12,921	10,432,967	10,876	12	3,085
Imp	–	–	–	–	–	–	7%	38%	4%	71%
avg-Imp	–	–	–	–	–	–	6%	35%	4%	68%

Table 3 Comparison of maze router and ILP based router

Results comparison of maze routing and ILP based routing (WVEM model)								
Circuit	Maze routing + rip-up rerouting				Flat level ILP routing (WVEM)			
	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time
Ind2	12,966,960	17,267	12	91	12,100,759	8,299	13	1,523
Ind3	50,599,000	27,504	30	98	47,308,679	14,037	32	1,527
Imp	–	–	–	–	7%	50%	–7%	–
avq.small	10,253,285	24,515	11	218	9,112,536	10,109	13	2,840
avq.large	11,886,555	30,306	12	325	10,432,976	10,876	12	3,085
Imp	–	–	–	–	12%	62%	–8%	–
avg-Imp	–	–	–	–	8%	57%	–7%	–

4.1 Top-down global routing

Figure 5 shows the proposed top-down routing algorithm. The first step in this top-down global routing is to build up the hierarchical levels of routing region representations. At the flat level (i.e. level 1), the routing region is partitioned into an array of fine grid bins. During the coarsening process, the flat level grid bins are merged to build the coarser level representations. At each coarsening level, a maximum of four adjacent grid bins from the previous level are merged to build a new grid cell in the current level. Based on the newly merged grid, the coordinates of each terminal cell are updated and a new net-list is generated such that all the terminal cells of each net are located in different grid bins. During this process, some local nets are absorbed since all the terminal cells of the nets are relocated to a single grid cell. The time complexity of building the hierarchical levels is $O(b)$, where b is the number of global bins. The ILP based router is invoked at each coarsening level to route all the current level nets based on the flat level grid graph. Figure 4 illustrates the framework of the top-down hierarchical routing.

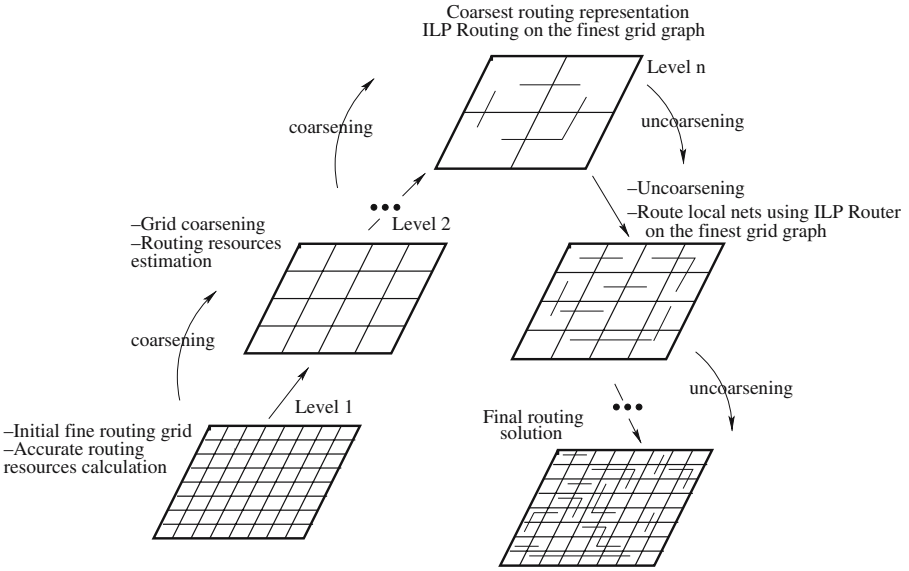


Fig. 4 Hierarchical global routing

Top-down Routing

1. Set i=1;
2. **FOR** level i and $i <$ top_level
   Generate grid graph for level i+1 by merging bins of level i;
   Update grid cell coordinates for level i+1;
   Regenerate global nets for level i+1;
   i= i + 1;
   END FOR
3. Set i = top_level;
4. **FOR** level i and $i >$ flat_level
   Perform ILP routing for current level global nets;
   Update routing demand of each grid edge;
   Perform grid uncoasrening;
   i=i -1 ;
   END FOR
5. Perform ILP routing for flat level unrouted nets.
6. Stop.

Fig. 5 Top-down hierarchical global routing

Table 4 shows the experimental results of 4, 5, 6-level top-down routing for large size circuits. Column "M_{rd}" shows the maximum routing density after final routing and row "Imp" shows the average percentage improvement produced by the top-down routing. From the table, we can see that the 5-level routing scheme can reduce the computation time of large size circuits largely by about 37% with the increasing total wire-length, total number of bends and maximum routing density. The number of levels to be used (i.e., currently set to five) is

guided by experimental work and therefore no analytical method is available to justify the appropriate level. The above experimental results indicate that for ECM model, the top-down routing can improve the computation time of flat level routing, however at the expense of solution quality.

4.2 Bottom-up global routing

In (8), the authors suggests that a shorter net has a higher priority than a longer net since the former enjoys less freedom when searching for a path to route it. However, in the top-down routing, the longer nets enjoy higher priority than the shorter nets. Therefore, in this section, we attempt to route local nets (i.e. shorter nets) first at each level of coarsening. The algorithm is shown in Fig. 6.

The algorithm starts by coarsening the finest grid bins of level 1. At level 2, the ILP based global router finds routing paths for the local nets (i.e. those nets that are entirely located inside a grid bin). The routing demand of each grid edge is then updated according to the routing results of local nets and considered in the next level routing. Coarsening continues until the ideal coarse level is reached. At the coarsest level, the local nets are first routed followed by the routing of the remaining unrouted global nets. As illustrated by Fig. 4, the bottom-up global routing attempts to solve the ILP formulation at level 1 and terminates at the coarsest level (i.e. level l). The time complexity of bottom-up routing is $O(l \times n)$, where l is the total number of hierarchical levels and n is the number of nets.

Figure 7 shows the effects of different hierarchical levels on solution quality using bottom-up routing. The quality of solution is measured by the total wire-length, total number of bends, maximum routing density and computation time. The results clearly show that 5-level bottom-up routing gives the best results in terms of computation time, while 6-level routing gives best results in terms of total number of bends and maximum routing density. For the total wire-length, the bottom-up routing with levels greater than three produce good results. According to this figure, hierarchical levels that are greater than five will tend to deteriorate the solution quality. Therefore, in this research, the highest coarsening level is set to five. In Table 5, the 5-level top-down routing is compared with the 5-level bottom-up routing. The results of flat level routing is also listed for the purpose of comparison. The "Imp" row highlights the average percentage improvement achieved by the top-down and bottom-up routing compared with the flat level routing. The results clearly indicate that the bottom-up routing achieves better results than that of top-down routing with noticeable improvement in computation time. From the comparison, we can also see that the bottom-up routing not only improves the computation time but also the total wire-length and total number of bends. The maximum routing density is only slightly increased by about 6%. Table 5 also indicates that the maximum routing density produced by bottom-up routing is less than that of

Table 4 ECM model: flat level routing versus top-down routing (4,5,6 levels)

Flat level routing and top-down routing without refinement (ECM model)																
Bench	Flat level				Top-down (4 levels)				Top-down (5 levels)				Top-down (6 levels)			
Marks	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time
Ind2	12,832,210	12,137	15	3,756	13,717,463	14,596	15	3,009	14,179,394	15,608	17	2,895	14,274,916	15,802	17	2,971
Ind3	50,311,413	21,164	32	3,531	51,819,487	22,204	38	2,623	52,678,398	22,996	39	2,796	53,002,808	23,416	40	2,970
avq.s	9,884,393	16,848	13	7,574	10,570,610	19,533	17	7,168	10,453,827	18,882	16	5,300	10,387,073	18,686	15	5,824
avq.l	11,121,285	17,198	13	12,921	11,612,625	19,803	15	8,063	11,647,939	19,930	16	6,446	11,749,173	20,529	16	6,955
Imp	–	–	–	–	−4%	−12%	−14%	25%	−5%	−13%	−17%	37%	−6%	−14%	−17%	33%

Bottom-up Routing

1. Set i=1;
2. **FOR** level i and $i <$ top_level
   Generate grid graph for level i+1 by merging bins of level i;
   Update grid cell coordinates for level i+1;
   Regenerate local nets for level i+1;
   Perform ILP routing for local nets of level i+1;
   Update routing demand of each grid edge;
   $i = i + 1$;
   END FOR
3. Perform ILP routing for top level unrouted nets.
4. Stop.

Fig. 6 Bottom-up hierarchical global routing

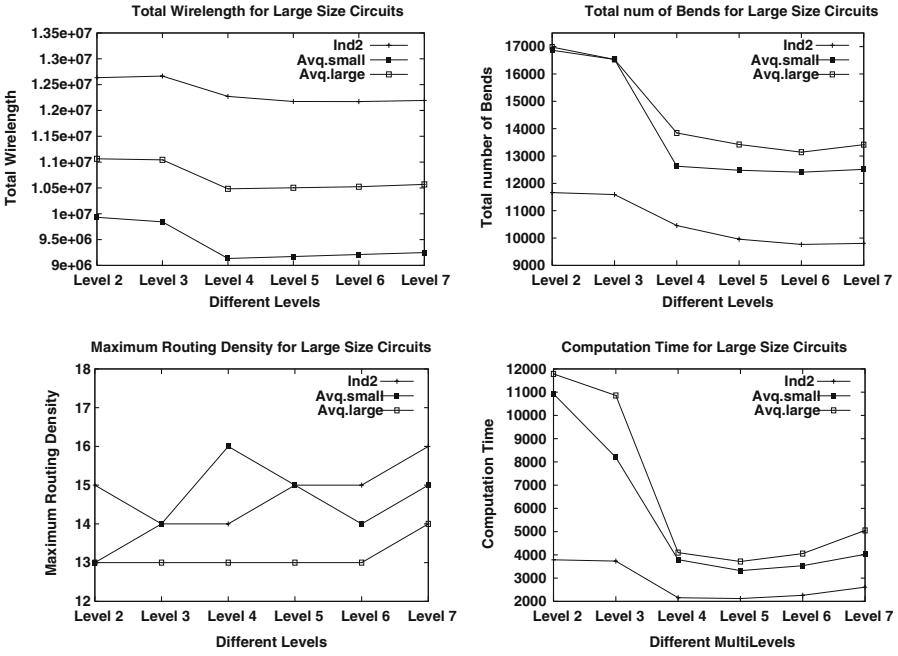


Fig. 7 Bottom-up routing of different levels for ECM model

the top-down routing. In the bottom-up routing, shorter nets that have less freedom are routed first. As the coarsening continues, the routing resource is consumed and the grid edges become congested. However, at the coarser level, most of the nets are longer nets, which have more freedom and thus, more admissible routes can be built for them to relieve the congestion.

Table 5 Comparison of flat level, top-down and bottom-up routing

Flat level routing verses top-down routing vs bottom-up routing (ECM model)												
Circuit	Flat level				Top-down 5-level				Bottom-up 5-level			
	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time	Wire	Bends	M_{rd}	Time
Ind2	12,832,210	12,137	15	3,756	14,179,394	15,608	17	2,895	12,172,848	9,959	15	1,757
Ind3	50,311,413	21,164	32	3,531	52,678,398	22,996	39	2,796	47,554,903	16,985	35	1,547
Imp	–	–	–	–	−5%	−14%	−16%	22%	5%	19%	−6%	55%
avq.s	9,884,393	16,848	13	7,574	10,453,827	18,882	16	5,300	9,172,372	12,481	15	2,777
avq.l	11,121,285	17,198	13	12,921	11,647,939	19,930	16	6,446	10,503,129	13,420	13	3,221
Imp	–	–	–	–	–5%	−12%	−18%	43%	7%	24%	−7%	70%
avg-Imp	–	–	–	–	–5%	−13%	−17%	37%	6%	22%	−6%	66%

5 Conclusion and future work

In this paper, an ILP based global routing algorithm is introduced and different mathematical formulations are then investigated and explored. Experimental results obtained indicate that the proposed combined model (WVEM) can optimize several global routing objectives simultaneously and effectively. To speedup the computation time of the proposed ILP based global router, a top-down hierarchical method is combined with the flat ILP based global routing approach. Experimental results show that the 5-level top-down routing can reduce the computation time by almost 37% for the ECM model, however, at the expense of the quality of solution. A bottom-up global routing method is then proposed in this paper. By comparing the 5-level bottom-up global routing with flat ILP based routing for the ECM model, the total wire-length and total number of bends are improved by 6 and 22% respectively. The computation time is also improved by 66% with a slight increase of maximum routing density. Our Future work involves applying different multi-objective optimization methods to the hierarchical global routing method and in particular considering power dissipation and performance. An analytical based method for determining the number of levels of hierarchical routing is also currently being investigated.

References

1. Behjat, L.: New modeling and optimization techniques for the global routing problem. PhD thesis, University of Waterloo, ON, Canada (2002)
2. Cheng, L., Song, X.Y., Yang, G.W., Tang, Z.W.: A fast congestion estimator for routing with bounded detours. In: Proceedings of the 2004 conference on Asia South Pacific design automation, pp. 666–670 (2004)
3. Chen, T., Chang, Y.: Multilevel full-chip gridless routing considering optical proximity correction. In: Proceedings of the 2005 conference on Asia South Pacific design automation, pp. 1160–1163 (2005)
4. Chen, T., Chang, Y., Lin, S.: A novel framework for multilevel full-chip gridless routing. In: Proceedings of the 2006 conference on Asia South Pacific design automation, pp. 636–641 (2006)
5. Cong, J., Fang, J., Zhang, Y.: MARS-A multilevel full-chip gridless routing system. IEEE Trans. Comput. Aided Des. **24**, 382–394 (2005)

6. Hadsell, R.T., Madden, P.H.: Improved global routing through congestion estimation. In: Proceedings of the 40th DAC, pp. 28–34, IEEE/ACM, Anaheim, CA (2003)
7. Kang, S.-M., Leblebici, Y.: CMOS Digital Integrated Circuits. McGraw-Hill, New York (2003)
8. S. L., Chang, Y.: A novel framework for multilevel routing considering routability and performance. In: Proceedings of the 2002 International Conference on Computer Aided Design, pp. 44–50 (2002)
9. Lou, J., Krishnamoorthy, S., Sheng, H.S.: Estimating routing congestion using probabilistic analysis. In: International Symposium on Physical Design, pp. 112–117 (2001)
10. MCNC: www.cbl.ncsu.edu/benchmarks/layoutsynth92/ (1991)
11. Sherwani, N.: Algorithms for VLSI Physical Design Automation. Kluwer, Boston (1999)
12. Vannelli, A.: An adaptation of the interior point method for solving the global routing problem. IEEE Trans. Comput. Aided Des. **10**, 193–203 (1991)
13. Warme, D.M.: A new exact algorithm for rectilinear steiner trees. In: International Symposium on Mathematical Programming (1997)
14. Yang, Z., Areibi, S., Vannelli, A.: An ILP based hierarchical global routing approach for VLSI ASIC design. University of Waterloo, Technical Report, University of Waterloo, Waterloo, ON, (2006)