Four-Bend Top-Down Global Routing

J. D. Cho, *Senior Member, IEEE*, and M. Sarrafzadeh, *Fellow, IEEE*

***Abstract*—We propose a new global net distribution approach for high-performance $m \times m$ two-dimensional arrays of very large scale integration and multichip-modules. The objective is to route n nets with minimum density of global cells, using a "small" number of bends. There are a number of applications where it is necessary to limit the number of bends on each wire. For example, it is desirable to limit the number of bends on each microstrip (transmission) line, for mismatches of line impedance can cause reflections from the junction points such as bends and vias. Furthermore, for high-performance routing, intersections of wires cause the use of more vias, which in turn require the use of more routing resources (because of the larger via pitch). This is the first paper that addresses a graph-theoretic framework to solve the *bend-constrained global routing* problem in two-dimensional arrays. In this paper, at each level of an underlying quad-tree, we present a novel four-bend routing algorithm by decomposing the original problem at level i into two subproblems that can be solved exactly based on a two-stage approach of smaller-sized linear program followed by min-cost flow networks. The overall (i.e., entire level of the four-way partition hierarchy) constraint and variable size for the first stage is $O(md_o)$, while the overall run time for the second stage is $O(n^3 \log n^2)$. The time complexity of such a hierarchical approach is one order of magnitude less than one of constructing a global routing using the min-cost-flow-based flat design approach. Last, we present an extension that permits a limited degree of control over the number of bends. The proposed algorithm can also be used for estimating the wireability in the early design planning stage for high-level synthesis. Experimental results showed the effectiveness of the proposed algorithm.**

***Index Terms*—Bend minimization, deep submicrometer, global routing, linear programming, minimum cost flow, very-large-scale-integration (VLSI) layout.**

I. Introduction

UNLIKE conventional approaches, the primary goal of routing in high-performance packages (such as multichip modules (MCM's) and wafer-scaled integrated circuits [2]) is to meet the high performance requirements rather than minimizing the layout area minimization only. There are several new and interesting design automation problems associated with this new packaging environment [9]–[10], [13], [29].

Manuscript received December 19, 1996; revised April 22, 1998. This work was supported by the academic research fund of the Ministry of Education, Republic of Korea, through the Inter-University Research Center (ISRC 97-E—2031) at Seoul National University. This paper was recommended by Associate Editor R. H. J. M. Otten.

J. D. Cho is with the School of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 440-746 Korea (e-mail: jdcho@yurim.skku.ac.kr).

M. Sarrafzadeh is with the Department of Electrical and Computer Engineering, Northwestern University, Evanston, IL 60208-3118 USA (e-mail: majid@ece.nwu.edu).

Publisher Item Identifier S 0278-0070(98)06762-1.

With signals in the gigahertz range, the electrical characteristics of the packages require the signal lines to be treated as transmission lines. On most conventional interconnecting substrates, transmission-line delay is linearly proportional to the distance traveled; but with multilayer thin films, delay becomes proportional to the distance squared. The reason is the high series resistance of thin conductive lines and the high capacitance of these lines to ground due to thin dielectrics. In practice, transmission lines are not perfectly uniform. That is, in the package level, significant reflections can be generated from capacitive and inductive discontinuities along the transmission lines. In terms of distortion, branches (stubs) are an order of magnitude more important than bends. In terms of delay, however, due to the additive inductance and capacitance in vias (i.e., bends), removing an excessive number of bends is more important than removing branches. Moreover, in a multilayer ceramic substrate of MCM, wires at different levels do not have exactly the same impedance. Such mismatches of line impedance can cause reflections from the junction points such as vias and bends. Furthermore, for high-performance routing, intersections of wires cause the use of more vias, which in turn require the use of more routing resources (because of the larger via pitch). If we minimize the number of segments per net (i.e., number of bends), the number of vias will be reduced proportionally. Therefore, propagation delays associated with discontinuities (e.g., see [11], [15], and [16]) could be minimized through careful design.

Since high-performance circuits are usually designed aggressively (i.e., most nets are considered as critical nets), we need to force them to have a small number of bends. Thus, the scheme also has an important application in estimating necessary wiring space and difficulty of routing in the early high-level synthesis step.

For the routing problem in two-dimensional arrays, there have been various approaches based on hierarchical wiring [4], [23], sequential methods [6], [21], [22], [26], simulated annealing [20], [33], linear programming [25], [27], multicommodity flow [5], [31], and flat approaches [17], [30]. Reference [19] presented a four-via MCM router that routes nets, using no more than four bends per net, assigned to one x–y plane pair at a time to minimize the number of layers.

Global routing is known to be NP-complete even in the case of one-bend routing of two-terminal nets [17]. From the result of global net distribution with uniform density, detailed routing assigns nets to layers so as to minimize the number of layers, preserving planarity in each layer. The global analysis of the routing region leads to "uniform" density distribution

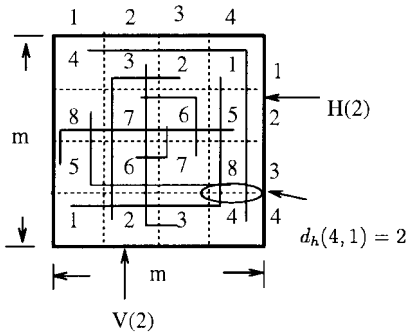

Fig. 1. An instance of one-bend global routing.

and results in a minimum number of layers in the detailed layer assignment stage.[1]

In this paper, we focus on finding a "good" bend-constrained global routing, taking performance issues into consideration (such as wire length and net criticality, as well as area and the number of bends). The proposed algorithm employs a novel four-way top-down recursion, allowing at most four bends per net. Each level of the recursion, a small-sized linear programming (LP) followed by min-cost flow networks[2] is constructed to find an exact solution in that level. Experimental results show that the routing quality of our algorithm is comparable to that of the previous algorithms that generate a large number of bends.

This paper is organized as follows. In Section II, we define and formulate the main problem. In Section III, we introduce a global routing strategy based on top-down recursion using four-way partitioning. An effective algorithm is presented by employing the two-stage min-cost filmed networks. Furthermore, to provide more flexibility on the number of bends, a *depth-constrained terminal propagation* technique is proposed in Section IV. In Section V, experimental results will complement the proposed theories. The conclusion of this paper is given in Section VI.

II. Problem Formulation

We adopt the multilayer global routing environment involving n two-terminal nets in two-dimensional arrays (refer to Fig. 1). We assume that a path goes from cell to cell rather than from grid point to grid point. Each plane consists of a two-dimensional $m \times m$ grid, being a square tessellation of the plane, with 1×1 being the basic cell-grid size, and each cell contains at most one pin. Nets are allowed to cross the borders of the basic cell. The problem of bend-constrained global routing in two-dimensional arrays with which we are concerned in this paper is: given a netlist and the maximum number of allowable bends, find a global routing with a minimum global density while minimizing the wire length on critical nets.

The global density is defined as follows.

[1]For the layer assignment problem that arises in MCM, see [13].

[2]Minimum-cost network flows are solved by a variation of the simplex algorithm and can be solved more than 100× faster than equivalently sized linear programs.

Definition 1 (global density d_R) (refer to Fig. 1): Let $cell(i, j)$ denote a cell at the ith row and the jth column of the routing region. In general, more than one net may cross the border of a cell. Let $d_h(i, j)$ denote the number of nets crossing the border of cells (i, j) and $(i, j+1)$, $1 \le i \le m$ and $1 \le j \le m-1$. Similarly, let $d_v(i, j)$ denote the number of nets crossing the border of cells (i, j) and $(i+1, j)$, $1 \le i \le m-1$ and $1 \le j \le m$. $d_{h\,\max} = \max_{i,j} d_h(i, j)$ is the *horizontal density* of the problem and $d_{v\,\max} = \max_{i,j} d_v(i, j)$ is the *vertical density* of the problem; $d_R = \max(d_{v\,\max}, d_{h\,\max})$ is the *global density* of the problem. That is, the global density is the maximum number of wires crossing a border between two cells in a routing solution R. A *vertical channel*, denoted by $V(j)$, consists of cells (i, j), $i = 1, \cdots, m$, and a *horizontal channel* $H(i)$ consists of cells (i, j), $j = 1, \cdots, m$.

The minimum d_R is referred to as an *optimal density*, denoted by d_{opt}. Fig. 1 illustrates an instance of one-bend routing with $d_{\text{opt}} = d_R = 2$.

The top-down recursive algorithm has a great attraction such that each level of hierarchy has a manageable-size problem that can be solved exactly. In this section, we employ such a technique to solve the four-bend routing problem in two-dimensional arrays.

Two top-down partitioning paradigms have been introduced. One is to partition a routing region into four square subregions successively (i.e., structured as a *quad-tree*) [4]. The other is to bipartition subregions on the basis of binary cut trees, dividing the routing region vertically or horizontally in a single partition step [24]. The quad-tree approach is more precise than the approach of binary cut trees. The former yields a truly two-dimensional routing paradigm, while the latter results in a one-dimensional partitioning procedure to solve the two-dimensional routing problem. There are two ways of forming the partitioning size: area-based partitioning (slice or rectangular) and point-based partitioning. For the application of high-performance packages, the former with square bucketing is sufficient since their routing substrates are very dense and pins are distributed evenly over the plane.

Therefore, we process the top-down recursion by first partitioning the top level representing the whole routing region into four square subregions. That is, a quadrisection having four quadrants is considered at each node of the *quad-tree*. For an $m \times m$ grid (without loss of generality, we assume that m is a power of two), there are $\log_2 m$ (i.e., $0, 1, \cdots, \log_2 m - 1$) levels of two-dimensional arrays. That is, level zero is the topmost level consisting of a 2 by 2 array, while level $\log_2 m - 1$ is the bottommost level consisting of an $m \times m$ array.

The following preliminary definition provides a property of level i of the top-down hierarchy.

Definition 2 [$QM(i)$] (refer to Fig. 2): The quadrisection map $QM(i)$ consists of four quadrants Q_k^i, $k \in (1, 2, 3, 4)$ (labeled counterclockwise from the upper right corner). U_k^i denotes the number of nets with one terminal inside and one terminal outside Q_k^i. A common boundary of two adjacent quadrants is said to be a *cut line*. Hence, there are two vertical

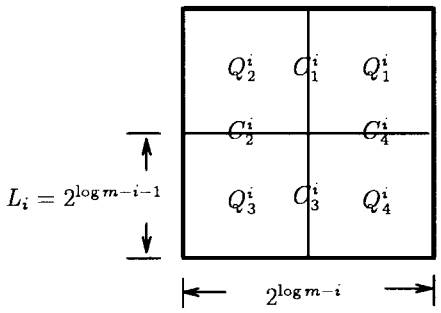

Fig. 2. QM(i).

and two horizontal *cut lines* in $QM(i)$, denoted by C_k^i. The length of each cut line is denoted by $L_i = 2^{\log m - i - 1}$.

Definition 3 (lower bound density d_0): When we recursively partition the square region into 2 by 2 square subregions, there are square subregions of sides $L_i = 2^{\log m - i - 1}$, where $0 \leq i \leq \log m - 1$ in an $m \times m$ two-dimensional array. Consider all square subregions Q_k^i with side L_i, where $0 \leq i \leq \log m - 1$ and $1 \leq k \leq 4$. Let us denote $\max_i \max_k |U_k^i|$ as $|U_i|$. Consider the worst case where all unconnected nets must leave one side of the square containing $|U_i|$ nets. Then, we define the loosely estimated density as $d_0 = \max_i(\lceil |U_i|/L_i \rceil)$, where $0 \leq i \leq \log m - 1$.

In our top-down hierarchical routing, the problem is decomposed into smaller square subregions $QM(i)$ that are solved exactly at level i. Let us consider routing the set of unconnected nets simultaneously at level i in four quadrants with each side $2^{\log m - i - 1}$. There are at most $U_i = 2^{\log m - i - 1} d_0$ *unconnected* terminals [since $U_i/(2^{\log m - i - 1}) < d_0$ by Definition 3] leaving one side of the square Q_k^i, $1 \leq k \leq 4$. We refer to the subproblems at each level as the *quadrisection map routing problem* (QMRP).

III. Top-Down Four-Bend Global Routing

In this section, we develop a near optimal algorithm called 4-BR based on a small-sized linear programming followed by a min-cost flow on a transformed network that is solved exactly at level i of the top-down hierarchy. We also present an alternative approach that provides a limited degree of control over the number of bends in the later section. We describe a two-stage approach in each level of the hierarchy based on the linear programming followed by a min-cost flow on a transformed network. We shall call the algorithm *four-bend quadrisection routing (4-BQR)* that contains the following new features.

For uniform density distribution, we keep track of the density contribution on the cut lines at each level of hierarchy. Thus, the minimization process for level i incorporates the existing density on the cut lines that has been accumulated by the previous levels $(0 \sim i-1)$. The process of *linear programming* yields an exact solution but may require excessive computing time since it is an implicit enumeration algorithm in that all feasible assignment candidates are implicitly, but exhaustively, examined. Therefore, to reduce computation time, we split the number of variables and constraints into two independent stages. Furthermore, since we want to restrict ourselves to few admissible routes such as four-bend routes, the size of the linear programming at each level of hierarchy can be significantly reduced.

In this paper, for brevity, we mainly describe an algorithm for the instance with two-terminal nets such that a net $i \in N$ consists of two terminals. An extension to multiterminal nets will be described at the end of this section.

Definition 4 (upper bound capacities): Let u_k denote the capacity that is upper bound on the number of nets that cross the cut line $C = \{C_1, C_2, C_3, C4\}$. Let F_{jk} $(|F_{jk}| = f_{jk})$ denote a set of nets such that one terminal is in Q_j and the other terminal is in Q_k. Let F_k $(|F_k| = f_k)$ denote the set of nets crossing a cut line C_k. Let u_{kj} $(= \lceil f_k/L \rceil)$ denote the capacity assigned to channel j crossing C_k, where L is the length of cut line.

Let us state the objective of each stage of the proposed two-stage algorithm. Stage 1, given the capacity constraints on cut lines, attempts to find a feasible solution minimizing *global density* of "four cut lines." Stage 2, given the capacity of cut lines, attempts to find a feasible solution minimizing *global density* of "channels on four cut lines."

The problem is defined as a *surjection*, $G : N \rightarrow C$, where C denotes the set of cut lines for the first stage (the set of channels for the second stage) subject to a homogeneous density distribution under the capacity constraint.

For net i, the cost $w_{i,j,k}$ is the cost of segments of net i passing through channels j and k, defined as

$$w_{i,j,k} = r_i \times \ell_{i,j,k} + b_{i,j,k}.$$

Net criticality on net i is denoted by r_i. If the information about net criticality is given, then the net assigned by a higher criticality will be chosen with a shorter net. Here, $\ell_{i,j,k}$ is the length of net i with two terminals, denoted i and i'. In Stage 1, $\ell_{i,j,k}$ corresponds to the shortest path distance between two terminals i and i'; in Stage 2, $\ell_{i,j,k} = \ell_{i,j} + \ell_{j,k} + \ell_{k,i'}$, where $\ell_{i,j}$ is the shortest distance between a terminal i and channel j, $\ell_{j,k}$ is the shortest distance between channel j and channel k, and $\ell_{k,i'}$ is the shortest distance between a terminal i' and channel k. The value of $b_{i,j,k}$ is the number of bends generated by net i. If some nets have equal length, then we prefer one with smaller bends. Note that the value of $b_{i,j,k}$ is at most four at Stage 2, so that the maximum number of bends for each net is at most four (refer to Fig. 3).

Now, let us establish the following *linear programming* formulation for Stage 1 of the 4-BQR.

Minimize

$$\begin{aligned}
&\sum (w_{i,14}x_{i,14} + w_{i,23}x_{i,23}, \forall i \in F_{24}) \\
&\quad + \sum (w_{i,12}x_{i,12} + w_{i,34}x_{i,34}, \forall i \in F_{13}) \\
&\quad + \sum (w_{i,1}x_{i,1} + w_{i,234}x_{i,234} : \forall i \in F_{12}) \\
&\quad + \sum (w_{i,2}x_{i,2} + w_{i,341}x_{i,341} : \forall i \in F_{23}) \\
&\quad + \sum (w_{i,3}x_{i,3} + w_{i,412}x_{i,412} : \forall i \in F_{34}) \\
&\quad + \sum (w_{i,4}x_{i,4} + w_{i,123}x_{i,123} : \forall i \in F_{14})
\end{aligned}$$

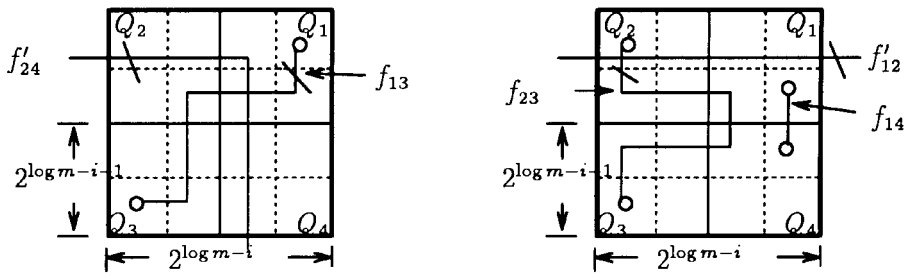

Fig. 3. An illustration of four-bend routing. f'_{ij} is the number of nets already routed in levels $0 \sim (i-1)$, and f_{ij} is the number of nets being routed in level i.

subject to capacity constraint

$$\sum(x_{i,1}+x_{i,12}+x_{i,14}+x_{i,341}+x_{i,412}+x_{i,123}) \le u_1$$
$$\sum(x_{i,2}+x_{i,12}+x_{i,23}+x_{i,234}+x_{i,412}+x_{i,123}) \le u_2$$
$$\sum(x_{i,3}+x_{i,34}+x_{i,23}+x_{i,234}+x_{i,341}+x_{i,123}) \le u_3$$
$$\sum(x_{i,4}+x_{i,34}+x_{i,14}+x_{i,234}+x_{i,341}+x_{i,412}) \le u_4, \quad \forall i \in N$$

and assignment constraint

$$x_{i,12}+x_{i,34}=1, \forall i \in F_{13}$$
$$x_{i,23}+x_{i,14}=1, \forall i \in F_{24}$$
$$x_{i,1}+x_{i,234}=1, \forall i \in F_{12}$$
$$x_{i,2}+x_{i,341}=1, \forall i \in F_{23}$$
$$x_{i,3}+x_{i,412}=1, \forall i \in F_{34}$$
$$x_{i,4}+x_{i,123}=1, \forall i \in F_{14}.$$

Here, the 0-1 variable $x_{p,q}$ is one if and only if net p is assigned to the cheapest channel (or channels) q. The capacity constraints indicate that we want to have a uniform density distribution on cut lines. The assignment constraints ensure that each net is assigned to a distinct channel and each net in $(F_{13} \cup F_{24})$ uses at most two channels to be connected; each net in $(F_{12} \cup F_{23} \cup F_{34} \cup F_{14})$ uses either one channel or three channels to be connected.

The upper bound capacity at each level can be derived as follows. There are two types of nets in our four-bend routing (refer to Fig. 3): Type 1) one terminal is in Q_k and the other terminal is in $Q_{(k+1)mod4}$ and Type 2) one terminal is in Q_k and the other terminal is in $Q_{(k+2)mod4}$. Let us denote by f_{ij} the number of nets whose one terminal is in Q_i and other terminal in Q_j. Let us denote by $f_{\max_1}$ the maximum f_{ij} of $\{f_{12}, f_{23}, f_{34}, f_{14}\}$ and by $f_{\max_2}$ the second maximum f_{ij} of $\{f_{12}, f_{23}, f_{34}, f_{14}\}$.

To distribute evenly over cut lines, for Type 1 nets, we need at most $\lceil (f_{\max_1}+f_{\max_2})/2 \rceil$ tracks to be routed by crossing a single cut line. Among the Type 1 nets, we need at most $\lceil (f_{\max_1}-f_{\max_2})/2 \rceil$ nets to be routed by crossing the three cut lines if $f_{\max_1}-f_{\max_2} \ge 2$.

For Type 2 nets, we need at most $\lceil (f_{13}+f_{24})/2 \rceil$ tracks when the nets are distributed evenly over two cut lines. In all, we need at most[3]

$$\left\lceil \frac{f_{13}+f_{24}+f_{\max_1}+f_{\max_2}}{2} \right\rceil.$$

Let us define *density history* on a cut line k, denoted d'_k, of levels $1 \sim (I-1)$. We incorporate the density history of levels $1 \sim (I-1)$ on each channel into the upper bound capacity u_k at level I as shown in (1) at the bottom of the page. Here, $f'_{13} = \max(d'_1, d'_2) + \max(d'_3, d'_4)$, $f'_{24} = \max(d'_1, d'_4) + \max(d'_2, d'_3)$, $f'_{\max_1} = \max(d'_1, d'_2, d'_3, d'_4)$, and $f'_{\max_2}$ is the second maximum of $\{d'_1, d'_2, d'_3, d'_4\}$. An example of Stage 1 is shown as follows (refer to Fig. 4).

A. Example for Stage 1

Let $w_i = \ell_i$ and the length of each cut line C_1, C_2, C_3, C_4 be one

$$u_1 = u_2 = u_3 = u_4 = \left\lceil \frac{f_{13}+f_{24}+f_{\max_1}+f_{\max_2}}{2} \right\rceil = \left\lceil \frac{1+2+1+0}{2} \right\rceil = 2.$$

Minimize

$$W = 4x_{1,14}+4x_{1,23}+4x_{2,14}+4x_{2,23}+4x_{3,4}+6x_{3,123}+3x_{4,12}+3x_{4,34}$$

subject to capacity constraints

$$x_{1,14}+x_{2,14}+x_{3,123}+x_{4,12} \le u_1$$
$$x_{1,23}+x_{2,23}+x_{3,123}+x_{4,12} \le u_2$$
$$x_{1,23}+x_{2,23}+x_{3,123}+x_{4,34} \le u_3$$
$$x_{1,14}+x_{2,14}+x_{3,4}+x_{4,34} \le u_4$$

[3]Even distribution of unconnected nets in an earlier level of hierarchy based on the presented upper bound may be lower than global (i.e., optimal) density. The reason is that a density in a lower level of hierarchy may be greater than one of a higher level of hierarchy. This tight upper bound causes Type 1 nets to detour crossing three cut lines. We can avoid detouring of the Type 1 nets by relaxing the upper bound. One effective way is to find $\max(d_0+\alpha, \lceil (f_{13}+f_{24}+f_{\max_1}+f_{\max_2})/2 \rceil)$, where $d_0+\alpha$ is an estimated upper bound on global density. We can measure the bound by once running the proposed algorithm, and then use the bound in the next iteration.

$$\left\lceil \frac{f_{13}+f_{24}+f_{\max_1}+f_{\max_2}+f'_{13}+f'_{24}+f'_{\max_1}+f'_{\max_2}}{2} \right\rceil \quad (1)$$

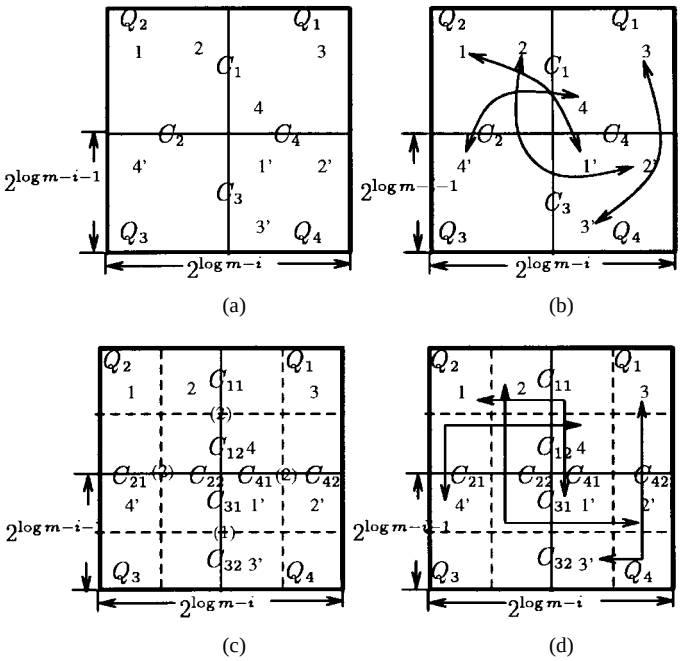

Fig. 4. An instance of 4-BQR.

and assignment constraints

$$x_{1,14}+x_{1,23}=1$$
$$x_{2,14}+x_{2,23}=1$$
$$x_{3,4}+x_{3,123}=1$$
$$x_{4,12}+x_{4,34}=1.$$

A solution to the first stage is $x_{1,14}=x_{2,23}=x_{3,4}=x_{4,12}=1$, $f_1=f_2=f_4=2$, $f_3=1$, with $(w_{1,14}+w_{2,23}+w_{3,4}+w_{4,12}=15)$.

In Stage 2, similar to Stage 1, the upper bound capacity u_{kj} (see Definition 4) is computed as follows. We incorporate the density history of levels $1\sim(I-1)$ on each channel into the upper bound capacity u_{kj} at each level as

$$u_{1j}=\left\lceil\frac{f_1+d'_1}{L}\right\rceil,\ \forall j\in C_1$$
$$u_{2j}=\left\lceil\frac{f_2+d'_2}{L}\right\rceil,\ \forall j\in C_2$$
$$u_{3j}=\left\lceil\frac{f_3+d'_3}{L}\right\rceil,\ \forall j\in C_3$$
$$u_{4j}=\left\lceil\frac{f_4+d'_4}{L}\right\rceil,\ \forall j\in C_4$$

where d'_k is the density history on a cut line k, as defined just before Stage 1.

B. Example for Stage 2

Minimize

$$\begin{aligned}F=&4x_{1,11,41}+4x_{1,12,41}+6x_{1,11,42}+6x_{1,12,42}\\&+6x_{2,21,31}+4x_{2,22,31}+8x_{2,21,32}+6x_{2,22,32}\\&+4x_{3,41}+4x_{3,42}+5x_{4,11,21}+3x_{4,12,21}\\&+5x_{4,11,22}+3x_{4,12,22}\end{aligned}$$

subject to assignment constraints

$$x_{1,11,41}+x_{1,12,41}+x_{1,11,42}+x_{1,12,42}=1,\ \text{for net}\,1$$
$$x_{2,21,31}+x_{2,22,31}+x_{2,21,32}+x_{2,22,32}=1,\ \text{for net}\,2$$
$$x_{3,41}+x_{3,42}=1,\ \text{for net}\,3$$
$$x_{4,11,21}+x_{4,12,21}+x_{4,11,22}+x_{4,12,22}=1,\ \text{for net}\,4.$$

Let us assume that $d'_1=d'_2=d'_3=d'_4=0$. Then

$$u_{11}=u_{12}=\lceil f_1/L\rceil=\lceil 2/2\rceil=1$$
$$u_{21}=u_{22}=\lceil f_2/L\rceil=\lceil 2/2\rceil=1$$
$$u_{31}=u_{32}=\lceil f_3/L\rceil=\lceil 1/2\rceil=1$$
$$u_{41}=u_{42}=\lceil f_4/L\rceil=\lceil 2/2\rceil=1$$

and channel capacity constraints

$$x_{1,11,41}+x_{1,11,42}+x_{4,11,21}+x_{4,11,22}\le u_{11}=1$$
$$x_{1,12,41}+x_{1,12,42}+x_{4,12,21}+x_{4,12,22}\le u_{12}=1$$
$$x_{2,21,31}+x_{2,21,32}+x_{4,11,21}+x_{4,12,21}\le u_{21}=1$$
$$x_{2,22,31}+x_{2,22,32}+x_{4,11,22}+x_{4,12,22}\le u_{22}=1$$
$$x_{2,21,31}+x_{2,22,31}\le u_{31}=1$$
$$x_{2,21,32}+x_{2,22,32}\le u_{32}=1$$
$$x_{1,11,41}+x_{1,12,41}+x_{3,41}\le u_{41}=1$$
$$x_{1,11,42}+x_{1,12,42}+x_{3,42}\le u_{42}=1.$$

A solution to the second stage is $x_{1,11,41}=x_{2,22,31}=x_{3,42}=x_{4,12,21}=1$ with $d_R=1$ and $w_{1,11,41}+w_{2,22,31}+w_{3,42}+w_{4,12,21}=15$.

Now, the second stage of the 4-BQR can be formulated as a network flow that can be optimally solved by a min-cost flow. Refer to Fig. 5. Let us denote i and i' as two terminals of net $i\in N$. Let M be the set of two terminals such as $\{i,\ i'|\forall i\in N\}$. Let us denote by C the set of channels and C' the set of possible channel assignments of nets. $G'=(V'(=s\cup M\cup C'\cup C\cup t), A=(=A_{s,M}\cup A_{M,C'}\cup A_{C',C}\cup A_{C,t}))$.

Arc capacity upper bound $c_{i,j}$ and weight $w_{i,j}$ for each edge (i,j) in A are assigned as

$$\begin{cases}\text{if} & (i,j)\in A_{C,t},\ \text{then}\ c_{i,j}=u_{i,p,q};\\ \text{else} & c_{i,j}=1\end{cases}$$

where $u_{p,q}$ is an upper bound capacity $u_{p,q}$ assigned to channel q on cut line P with respect to node i. Arc weight $w_{i,j}$

$$\begin{cases}\text{if} & (i,j)\in A_{M,C'},\ \text{then}\ w_{i,j}=w_{p,q};\\ \text{else} & w_{i,j}=0\end{cases}$$

where the cost $w_{p,q}$ is the cost of a wire path for terminal p assigned to channel q. The cost $w_{p,g}$ is computed using, for example, the following linear equations:

$$w_{1,11}+w_{1,41}=w_{1,11,41}=4$$
$$w_{1,12}+w_{1,41}=w_{1,12,41}=4$$
$$w_{1,11}+w_{1,42}=w_{1,11,42}=6$$
$$w_{1,12}+w_{1,42}=w_{1,12,42}=6.$$

For each net crossing the cut lines, we need to determine the path of running the net across channels on the cut lines.

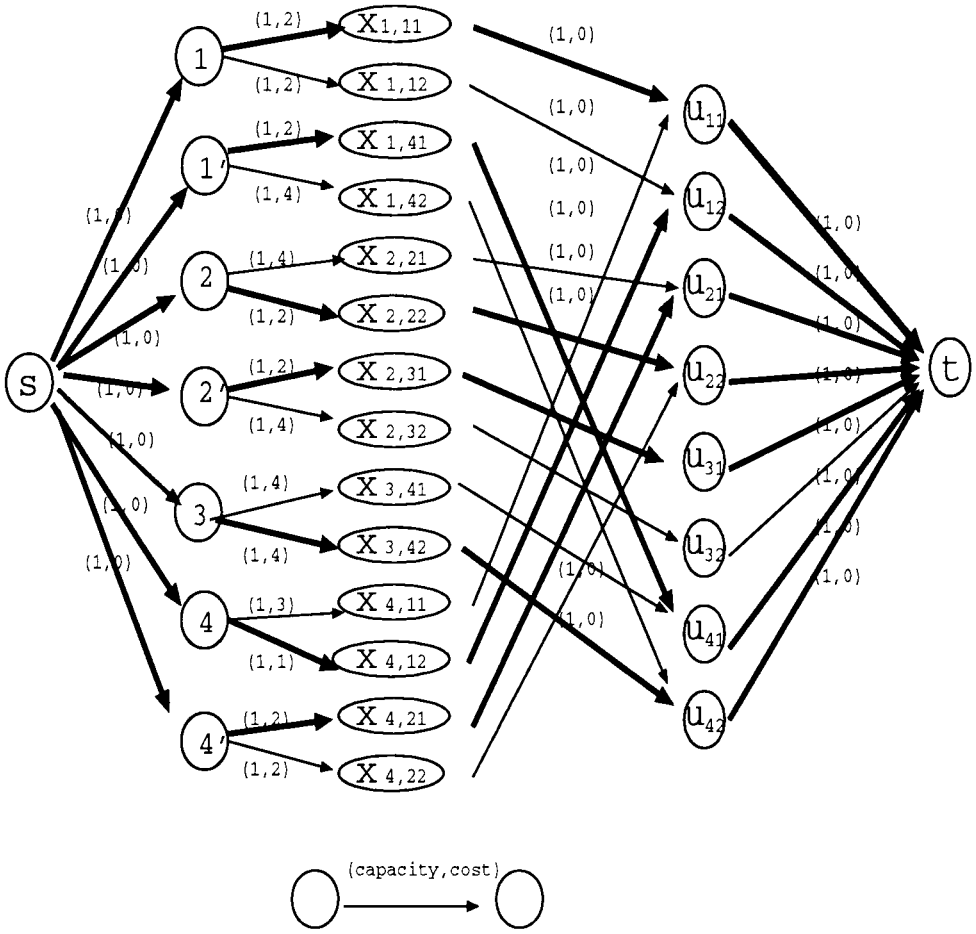

Fig. 5. Min-cost flow network for Stage 2 of 4-BQR and its min-cost flow solution (bold lines) that corresponds to Fig. 4(d).

Then the problem is solved by a min-cost flow [1] on G', for example, as shown in Fig. 5.

Note that the LP formulation in Stage 1 has a different structure of one in Stage 2. Similar to Stage 2, the cost $w_{p,g}$ can be computed using, for example, the following linear equations:

$$\begin{aligned} w_{1,1} + w_{1,4} &= w_{1,14} = 4 \\ w_{1,2} + w_{1,3} &= w_{1,23} = 4 \\ w_{1,1} + w_{1,3} &= w_{1,13} = \infty \\ w_{1,2} + w_{1,4} &= w_{1,24} = \infty. \end{aligned}$$

Note that the third and fourth are infeasible solutions and thus are assigned by infinite cost. In this case, we do not have a solution for $w_{1,1}$, $w_{1,2}$, $w_{1,3}$ and $w_{1,4}$. Therefore, the problem of Stage 1 cannot be solved by a min-cost flow.

Note that every net in $QM(I)$ completes routing at level I, providing a solution to the 4-BQR such that channel density on cut lines is minimized.

Theorem 1 (4-BQR): The 4-BQR is an exact algorithm achieving the minimum density of global cells to the QMRP so as to produce at most four bends per net.

To complete routing for nets in all levels, this routing process of 4-BQR continues proceeding down the top-down hierarchy until all unconnected nets in $QM(I)$ of levels have been laid out.

The complexity of the algorithm for Stage 1 is analyzed in terms of the number of variables and the number of constraint equations. The number of variables at level i is bounded by $4U_i$, where U_i is the number of unconnected nets at level i. The overall size of variables for $\log m$ levels of top-down hierarchy is given as follows:

$$\begin{aligned} \sum_{i=0}^{\log m-1} 4U_i &= \sum_{i=0}^{\log m-1} \left(2^{\log m-i+1} d_0\right) \\ &= 2 \cdot m \cdot d_0 = O(md_0). \end{aligned}$$

The number of constraints is $3U_i$. Similarly, the overall size of constraints is $1.5 \cdot m \cdot d_0 = O(md_0)$.

When applying min-cost flow at Stage 2 as in Fig. 5, the number of edges and nodes in G' at level i is $O(U_i L_i^2)$ and $O(U_i + L_i^2)$, respectively. Thus, the overall time complexity for the entire level for Stage 2 is

$$\begin{aligned} &\sum_{i=0}^{\log m-1} O\left(U_i L_i^2\right)^2 \log U_i + L_i^2 \\ &\cdot \sum_{i=0}^{\log m-1} O\Big(\Big(2^{(\log m-i-1^3} d_0\Big)^2 \\ &\qquad \cdot \log\left(2^{\log m-i-1} d_0 + 2^{(\log m-i-1)^2}\right)\Big) \\ &= O(m^6 \cdot (d_0)^2 \cdot (\log m d_0)^2). \end{aligned}$$

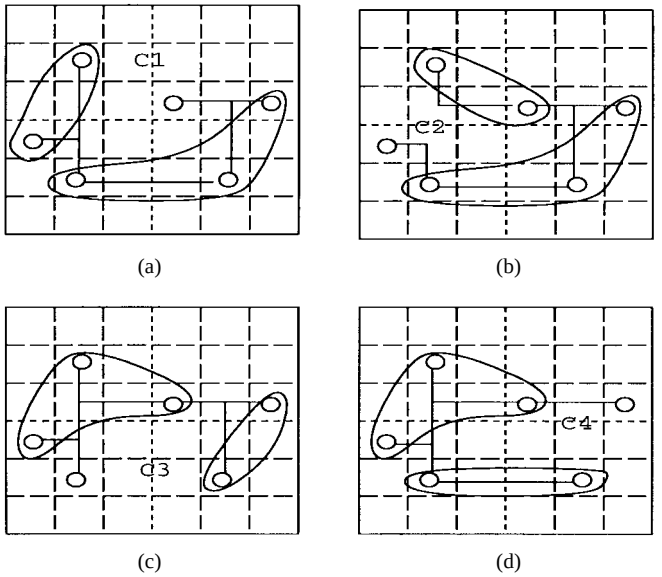

Fig. 6. Identifying a set of subnets for a multiterminal net while generating four best net topologies (with different set of cut-line assignments) at each level of top-down global routing.

In terms of n with the practical instances described before, the algorithm takes $O(n^3 \log n^2)$ computational time. The time complexity is one order of magnitude less than one of constructing a global routing graph using $(e = n^2)$ edges.

Multiterminal nets have many possible routing topologies such as daisy chain, Steiner tree, star, and A-tree [34]. It is desirable that the multiterminal nets are routed with their favorite topologies as listed above. For example, one restricts to a specific routing pattern for a multiterminal net with a goal-driven min-cost Steiner tree[4] having minimum wire length, and/or minimum bends, and/or minimum stubs. A stub or branch in a tree introduces extra delay and/or ringing in the received signal waveform [32]. However, it is impractical to consider all configurations of a large fan-out net because the number of net topologies as a function of the number of a large fan-out receivers increases rapidly.

Thus, our strategy is to generate four best net topologies for each multiterminal net, as shown in Fig. 6, each topology crossing a different set of cut lines. We can generate four such favorite topologies based on the various net topology construction described above. Then we select unconnected terminals of the net at each level of hierarchy, one terminal from each quadrant for each subnet in which the subnet has a terminal, as illustrated in Fig. 7. The illustration is self-contained; thus, we omit the algorithm details for brevity in this paper.

A nice feature of the two-stage approach described above is that the first stage splits multiterminal unconnected nets at level i into a set of two terminals. Thus, the complexity of the second stage is significantly reduced by considering only split two-terminal nets. However, the LP formulation in Stage 1 for multiterminal nets is different from one of two-terminal nets. For multiterminal nets with three or more terminals spread over four quadrants, there are four possible net topologies crossing a different set of cut lines. Let us assume that one decomposes the multiterminal nets into two-terminal nets *a priori*, then more than four topologies are required for all split two-terminal nets. Therefore, we know that the complexity of our algorithm on the instances of multiterminal nets is not more than one of two-terminal nets.

[4]For nets with large terminals, a min-cost Steiner heuristic is used.

IV. Depth-Constrained Terminal Propagation

In this section, we finally propose an alternative approach that provides a limited degree of control over the number of bends k.

Based on a binary cut tree, a custom chip routing algorithm using linear assignment together with hierarchical net decomposition is proposed in [24]. The algorithm is based on a top-down hierarchical scheme. At each level of hierarchy, the current routing problem is partitioned into two subproblems by assigning pins to channels on a cut line (or bisector). To find an optimal pin assignment for a cost function, a linear assignment is used to minimize the total summation of the costs subject to the capacity constraints of the channels (or holes). Then, to determine the path of nets inside each subregion, interface (or pseudo) pins are created on the cut lines, and nets are broken into several parts that can be processed independently. The process of cutting and linear assignment continues until all the nets are connected or all the boundaries are included in the cuts. Note that this approach produces many bends. We will refer to the process of assigning pseudoterminals to cut lines in each level of hierarchy as *terminal propagation*. *Terminal propagation depth* (denoted as δ_i) for a net i denotes the number of levels of top-down hierarchy through which a net's pseudoterminals are allowed to propagate.

Let us assume that we use the top-down terminal propagation technique based on the four-way partition, and that the number of bends produced by net configuration in depth i is limited to three. Then, as shown in Fig. 7, the number of maximum possible bends produced in successive levels of terminal propagation[5] is $3 \times 4^{\delta}$ (for the case of binary cut tree, $2^{\delta+1}$), where δ is the depth of terminal propagation. Therefore, the depth of terminal propagation should be restricted to a small number.

Here, we introduce a notion of *depth-constrained terminal propagation* (DCTP).[6] The underlying idea of DCTP is as follows. Channel assignment is performed using a two-bend routing [17] (we shall refer to it as 2-BQR) or 4-BQR at each level of the top-down recursion assigning nets to cut lines. Then each net is decomposed into at most four subnets by introducing a so-called *pseudoterminal*. For each net leaving a channel on a *cut line*, a *pseudoterminal* is placed on the channel through which a net crossing a cut line should pass. The concept of net decomposition and terminal propagation

[5]Since there are four quadrants at each level, the propagated pseudoterminals in each quadrant introduce at most three bends, as shown in Fig. 8.

[6]The depth of terminal propagation for a net is determined by the net's criticality. For the most critical nets, only the shortest routes with minimum bends are chosen and they are excluded from the route selection process in the later level of top-down hierarchy. The deeper terminal propagation in the successive level is exclusively allowed for less critical nets.

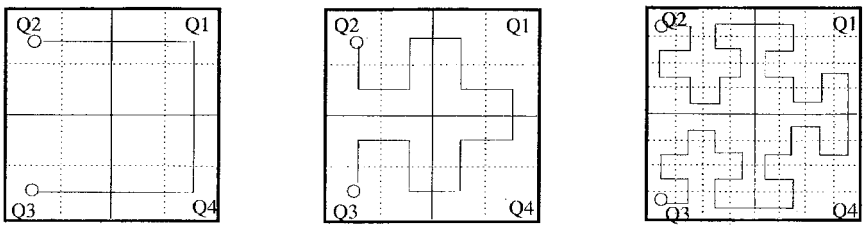

Fig. 7. An instance of 4-BQR.

at each level of hierarchy enables us to solve the smaller problems exactly.

To complete routing for nets at all levels, this routing process continues proceeding down the top-down hierarchy until all unconnected nets in $QM(I)$ of levels $(1 \sim \log m)$ have been laid out. We shall refer to the proposed algorithm as $k - BR$

Algorithm k-BR
Input: A set of nets N. δ_i for such net i
Output: A k-bend wire routing
For $i = 0$ to $\log m - 1$.
begin
Compute U_k; /* a set of unconnected nets in QM(k) at level k*/
For $i = 1$ to U_k.
begin
For each net i s.t. $\delta_i = k$,
apply inter- and intraquadrant channel assignment using 2- or 4-BQR
For each net i s.t. $\delta_i > k$,
apply interquadrant channel assignment and terminal propogation
end
end.

Here, *interquadrant channel assignment* is to assign nets to channels C_{ij} on cut lines C_i, whereas *intraquadrant channel assignment* is to assign nets to channels inside each quadrant.

Since the depth of terminal propagation controls the number of bends, we readily have the following.

Theorem 2: There is a polynomial time algorithm to solve the k-BR problem, producing $i \times j^{\delta} + k$ bends. Here, δ is the depth of terminal propagation. The value i $(\in (1, 2, 3, 4))$ is the number of quadrants that we allow a net to propagate its terminals inside the corresponding quadrants. The value j $(\in (2, 3, 4))$ is the number of bends allowed for each net at each level of the top-down hierarchy, and the value k is $j - i$ when $j \geq i$; otherwise 0.

Using 2-BQR on $QM(I)$, the number of bends generated is at most four $\times$ 2^{δ} when $\delta > 0$ (at most two when $\delta = 0$). Using 4-BR while restricting the number of bends at each level of top-down hierarchy to three or four, it is at most $4 \times 4^{\delta}$. Therefore, given the user-specified depth δ, the upper bound on the number of bends can be restricted to some degree for example, two (with $\delta = 0$) or eight (with $\delta = 1$) or 16 (with $\delta = 2$) using 2-BQR, and four (with $\delta = 0$) or 12 (with $\delta = 1$) or 48 (with $\delta = 2$) using 4-BR. In this manner, the upper bound on the number of bends can be constrained to an almost arbitrary number of bends.

TABLE I
A COMPARISON BETWEEN $KLR^{+}87$ AND OURS

KLR^{+} 87 (one-bend routing)	
#vars/#cons	bound
$O(m)/O(m^2)$	$d_R \leq \lceil m/2 \rceil + 1$
4-BR (Section 3, 4-bend routing)	
#vars/#cons	bound
$O(md_0^{\dagger})/O(md_0) + O(nd_0 \log(nd_0))$	$d_R \leq \lceil 2d_{opt} \rceil \log(m/\lfloor\lfloor 2d_0 \rfloor\rfloor)$

vars: variables, cons: constraints, $\dagger : d_0 \leq m/2$

V. EXPERIMENTAL RESULTS

It will be insightful to compare our approach with previous LP-based global routing approaches in terms of the problem formulation, number of variables, and number of constraints (refer to Table I). One existing linear programming approach is proposed in [17], in which Karp *et al.* used one-bend routing by rounding to obtain integral approximations to solutions of linear equations. In this paper, a four-bend routing based on a two-stage approach of LP followed by min-cost flow formulation at each level of a top-down quad-tree is used to improve the global density. Thus, we know that the bound on d_R, after applying the two-stage approach, is certainly less than the bound achieved by the approximation algorithm.

Mathematical programming methods are computationally inefficient for problems with a large number of variables. However, the time complexity of our hierarchical routing scheme compares favorably with the usual greedy approaches based on shortest paths, which routes an $m \times m$ array in time $O(m^4)$. Note that usually, clustering followed by placement prior to global routing produces short spans of nets. Thus, the number of unconnected nets at higher levels of the top-down hierarchy is usually not greater than that of the lower level of the hierarchy. Based on this observation, the number of variables are distributed over all levels of recursion hierarchy. Thus, in practice, the linear programming method is feasible for realistic problem sizes.

We include some experimental results to complement the theoretical results. We compared our approach with the two-bend Eulerian-tour-based global router (referred to as 2-BR) [17] and two other existing global routers: a flat net-by-net approach based on min-max Steiner trees, proposed by Chiang *et al.* [6], and a multicommodity flow algorithm proposed by Carden and Cheng [5]. We tested our algorithm (4-BR) using the examples provided in [5] and [6] (Diff-4, Diff-8, and Diff-16). Since the proposed method uses the hierarchical method,

TABLE II
A COMPARISON AMONG CSW90, CC91, 2-BR, AND 4-R

	CSW90[7]			*CC91[9]*			*2-BR[18]*			*4-BR‡*		
Ex (nets/⋆/∗)	m-b	d_R	w-l	m-b	d_R	w-l	m-b	d_R	w-l	m-b	d_R	w-l
Ex1(4/8/1)	2	2	11	1	1	16	1	1	16	1	1	16
Diff-4(8/32/2)	2	2	35	2	2	32	1	2	32	2	2	32
Diff-8(32/256/4)	◇	4	296	◇	4	256	2	5	260	2	4	256
Diff-16(128/2048/8)	◇	9	2214	◇	9	2050	2	9	2068	3	8	2050

‡: 4-BR in Section 3.
⋆: lower bound wirelength = bounding-box wirelength for all nets (a net's bounding-box length: half perimeter of the smallest bounding box enclosing two terminals of the net)
∗: lower-bound density = maximum number of nets that must cross the border of a cell on cutlines
m-b : maximum bends; w-l : total wire length.
Ex1 : Instance shown in Figure 2.
Diff-4 : Instance shown in Figure 1.
Diff-8,Diff-16 :Instances shown in [6].
◇: N/A, but can be increased upto $O(m)$.

TABLE III
AN EXPERIMENTAL RESULT ON LARGE-SIZED EXAMPLES

Ex (nets/⋆/∗/†/‡)	$m-b$(#4 − *bendnets*)	d_R	d_R^{avg}	$w-l$	time (sec.)
R1 ((2+1+1)/15/2/14/14)	3(0)	2	2	15	0.02
R2 ((20+30+100)/570/75/400/254)	4(25)	75	63	620	0.08
R3 ((100+50+350)/1950/250/1300/804)	4(100)	250	213	2150	0.63
R4 ((300+200+500)/3800/500/3000/2004)	3(0)	500	375	3800	3.56
R5 ((500+400+300)/4400/600/4200/3004)	3(0)	600	525	4400	6.86
R6 ((800+200+800)/7000/900/5600/3804)	3(0)	900	700	7000	13.11
R7 ((750+500+1250)/9500/1250/7500/5004)	3(0)	1250	938	9500	25.1
R8 ((700+800+2000)/13200/1750/10000/6504)	4(250)	1750	1375	13700	34.1

nets = $(f_{24} + f_{13} + f_{max})$; †: number of variables in LP; ‡: number of constraints in LP; d_R^{avg}: average density over four cutlines. time is measured by tms_utime which is the CPU time used while executing instructions in the user space of the calling process. The remaining are defined as in Table 2.

it will be insightful to compare with the algorithms presented in [21] and [23].

From Table II, we observe that by using smaller bends, 4-BR usually leads to a routing with less wire length when compared with [6] (with comparable global densities) and comparable wire length when compared with [5]. In the case of Diff-16, 4-BR produced a routing with the optimum density, and the wire length was two units greater than the lower bound wire length.

To test the linear program for Stage 1 of 4-BR with the large-sized instances, we implemented an LP generator in the C language running on a SUN SPARC classic and tested a set of random examples using Simplex-based lp-solver [3] (which has a time complexity similar to that of the Simplex-based min-cost flow algorithm). The computing time for the largest size problem with 3500 nets was 34.2 s, as shown in Table III. We do not perform extensive experiments on the min-cost flow formulation of Stage 2, but the correctedness of the formulation was verified by running the min-cost flow algorithm (NETFLO) [18].

The experimental results showed that 4-BR yields a near optimal solution in terms of global density and wire length.

We also simulated the situation of how much the bends affect signal distortion compared with the branches using HSPICE after extracting circuits from layout by EPC (Avante tool). In terms of distortion, branches are an order of magnitude more important than bends, even though our timing model with EPC may not successfully simulate the transmission effects. However, in terms of delay, due to the additive inductance and capacitance in vias (i.e., bends), removing excessive numbers of bends is more important than removing branches.

VI. CONCLUSION

In this paper, we presented effective approaches to the global routing problem arising in high-performance layouts. We subject to minimize the total length and number of layers (i.e., global density), constraint to minimum number of bends. Simultaneous treatment of nets by our method generated a "good" solution using only four bends per net. We developed an enhanced four-bend routing strategy with performance-driven constraints (i.e., congestion, net criticality, length, and bend), allowing at most four bends per net. The complexity of the problem is significantly alleviated by employing a two-step strategy of smaller sized LP followed by min-cost flow networks in each level of the top-down hierarchy. To provide flexibility to control the number of bends, the *depth-constrained terminal propagation* approach has also been introduced. The proposed approaches can be directly applicable to the high-performance systems where limiting

the number of bends is critical. A reviewer commented that branchings (stubs) are an order of magnitude more important than bends with respect to signal distortion. Therefore, we are currently investigating an extension to minimize the number of long (more than a certain threshold) stubs and the number of bends simultaneously.

ACKNOWLEDGMENT

The authors wish to acknowledge the suggestions of anonymous reviewers.

REFERENCES

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows.* Englewood Cliffs, NJ: Prentice-Hall, 1993.
[2] H. B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI.* Reading, MA: Addison-Wesley, 1990, pp. 81–112.
[3] M. Berkelaar. (1996). *lp-solve 2.0.* [Online]. Available FTP: ftp.es.ele.tue.nlat/pub/lp_solve.
[4] M. Burstein and R. Pelavin, "Hierarchical wire routing," *IEEE Trans. Computer-Aided Design,* vol. CAD-2, pp. 223–234, Oct. 1983.
[5] R. C. Carden IV and C. K. Cheng, "A global router using an efficient approximate multicommodity multiterminal flow algorithm," in *Proc. IEEE/ACM Design Automation Conf.,* , 1991, pp. 316–321.
[6] C. Chiang, M. Sarrafzadeh, and C. K. Wong, "Global routing based on Steiner min–max trees," *IEEE Trans. Computer-Aided Design,* vol. 9, no. 12, pp. 1315–1325, 1990.
[7] J. D. Cho and M. Sarrafzadeh, "The pin redistribution problem in multichip modules," *Math. Programming B,* vol. 63, pp. 297–330, Feb. 1994.
[8] J. D. Cho, S. Raje, K. F. Liao, and M. Sarrafzadeh, "M2R: A new multilayer routing system for high-performance MCM's," *IEEE Trans. Circuits Syst.,* vol. 41, pp. 253–255, Apr. 1994.
[9] W. W.-M. Dai, "Performance driven layout of thin-film substrates for multichip modules," in *Proc. IEEE Multichip Module Workshop,* 1990, pp. 114–121.
[10] ——, "Topological routing in SURF: Generating a rubber-band sketch," in *Proc. IEEE Design Automation Conf.,* 1991, pp. 39–48.
[11] J. M. Ho, M. Sarrafzadeh, G. Vijayan, and C. K. Wong, "Layer assignment for multi-chip modules," *IEEE Trans. Computer-Aided Design,* vol. CAD-9, pp. 1272–1277, Dec. 1990.
[12] L.-T. Hwang and I. Turlik, "Calculation of voltage drops in the vias of a multichip package," in *MCNC Tech. Rep.,* Tech. Rep. Series TR90-41, 1990.
[13] ——, "The skin effect in thin-film interconnections for ULSI/VLSI packages," in *MCNC Tech. Rep.,* Tech. Rep. Series TR91-13, 1991.
[14] R. M. Karp, F. T. Leighton, R. L. Rivest, C. D. Thompson, U. V. Vazirani, and V. V. Vazirani, "Global wire routing in two-dimensional arrays," *Algorithmica,* vol. 2, no. 1, pp. 113–129, 1987.
[15] J. L. Kennington, *Algorithms for Network Programming.* New York: Wiley, 1980, pp. 244–256.
[16] K. Y. Khoo and J. Cong, "An efficient multilayer MCM router based on four-via routing," in *Proc. IEEE/ACM Design Automation Conf.,* 1993, pp. 590–595.
[17] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science,* vol. 220, pp. 671–680, 1983.
[18] J. T. Li and M. Marek-Sadowska, "Global routing for gate arrays," *IEEE Trans. Computer-Aided Design,* vol. CAD-3, pp. 298–307, Oct. 1984.
[19] K. F. Liao, M. Sarrafzadeh, and C. K. Wong, "Single-layer global routing," in *Proc. 4th Ann. IEEE Int. ASIC Conf. Exhibit,* Sept. 1991, pp. 14-4.1–14-4.4.
[20] W. K. Luk, P. Sipala, M. Tamminen, D. Tang, L. S. Woo, and C. K. Wong, "A hierarchical global wiring algorithm for custom chip design," *IEEE Trans. Computer-Aided Design,* vol. CAD-6, pp. 518–533, July 1987.
[21] M. Marek-Sadowska, "Route planner for custom chip design," in *Proc. IEEE Int. Conf. Computer-Aided Design,* Nov. 1986, pp. 246–249.
[22] G. Meixner and U. Lauther, "A new global router based on a flow model and linear assignment," in *Proc. IEEE Int. Conf. Computer-Aided Design,* IEEE, Nov. 1990, pp. 44–47.
[23] R. Nair, "A simple yet effective technique for global wiring," *IEEE Trans. Computer-Aided Design,* vol. CAD-6, no. 2, pp. 165–172, Mar. 1987.
[24] Y. Nishizaki, M. Igusa, and A. Sangiovanni-Vincentelli, "Mercury: A new approach to macro-cell global routing," in *Proc. VLSI,* Germany, 1989, pp. 411–419.
[25] J. B. Orlin, "A faster strongly polynomial minimum cost flow algorithm," in *Proc. 20th ACM Symp. Theory of Computing,* 1988, pp. 377–387.
[26] B. Preas, M. Pedram, and D. Curry, "Automatic layout of silicon-on-silicon hybrid packages," in *Proc. Design Automation Conf.,* 1989, pp. 394–399.
[27] M. Sarrafzadeh and D. Zhou, "Global routing of short nets in two-dimensional arrays," *Int. J. Computer Aided VLSI Design,* vol. 2, no. 2, pp. 197–211, 1990.
[28] E. Shargowitz and J. Keel, "A global router based on multicommodity flow model," *Integration: VLSI J.,* vol. 5, pp. 3–16, 1987.
[29] M. Sriram and S. M. Kang, *Physical Design for Multichip Modules.* Norwell, MA: Kluwer, 1994
[30] M. P. Vecchi and S. Kirkpatrick, "Global wiring by simulated annealing," *IEEE Trans. Computer-Aided Design,* vol. CAD-2, no. 4, pp. 215–222, 1983.
[31] A. Vittal and M. Marek-Sadowska, "Minimal delay interconnect design using alphabetic trees," in *Proc. Design Automation Conf.,* 1994, pp. 392–396.

J.-D. Cho (M'90–SM'96) received the B.S. degree in electronics from Sung Kyun Kwan University, Seoul, Korea, in 1980. He received the M.S. degree from Polytechnic University, Brooklyn, NY, in 1989, and the Ph.D. degree from Northwestern University, Evanston, IL, in 1993, both in computer science.

He was a Senior CAD Engineer with Samsung Electronics, Co., Ltd., Buchun, Korea. Since 1995, he has been Assistant Professor of Electronic Engineering, Sungkyunkwan University, Suwon, Korea. His research interests include very-large-scale-integration (VLSI) algorithmic design automation. He is as a Guest Editor of *VLSI Design* for the special issue "Design Automation for VLSI Interconnects." He is the author of *High-Performance Design Automation of MCM's and Packages* (World Scientific, 1996) and coauthor of an invited chapter in the *Encyclopedia of Electrical and Electronics Engineering* in the area of VLSI circuit layout (Wiley, forthcoming).

Dr. Cho received the Best Paper Award at the 1993 Design Automation Conference in the area of physical design.

Majid Sarrafzadeh (M'87–SM'92–F'96) received the B.S., M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign in 1982, 1984, and 1987, respectively.

In 1987, he joined Northwestern University, Evanston, IL, as an Assistant Professor. Since 1997, he has been a Professor of electrical engineering and computer science there. His research interests lie in the area of very-large-scale-integration (VLSI) computer-aided design (CAD), design and analysis of algorithms, and VLSI architecture. He has served on the technical program committees of numerous conferences in the area of VLSI design and CAD, including ICCAD, EDAC, and ISCAS. He has served as Committee Chair of a number of these conferences, including the International Conference on CAD and International Symposium on Physical Design. He will be the General Chair of the 1998 International Symposium on Physical Design. He has published approximately 150 papers. He is a coeditor of *Algorithmic Aspects of VLSI Layout* (World Scientific, 1994), coauthor of *An Introduction to VLSI Physical Design* (McGraw-Hill, 1996), and author of an invited chapter in the *Encyclopedia of Electrical and Electronics Engineering* in the area of VLSI circuit layout (Wiley, forthcoming). He is on the editorial board of *VLSI Design Journal* and is Coeditor-In-Chief of the *International Journal of High-Speed Electronics*. He has collaborated with many industries in the past ten years, including IBM and Motorola.

Dr. Sarrafzadeh received a National Science Foundation Engineering Initiation Award, two distinguished paper awards in ICCAD, and the Best Paper Award for physical design in the Design Automation Conference for his work in the area of high-speed VLSI clock design. He is an Associate Editor of IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS.