

Reliability-Driven Power/Ground Routing for Analog ICs

JING-WEI LIN and TSUNG-YI HO, National Cheng Kung University
IRIS HUI-RU JIANG, National Chiao Tung University

Electromigration and voltage drop (IR-drop) are two major reliability issues in modern IC design. Electromigration gradually creates permanently open or short circuits due to excessive current densities; IR-drop causes insufficient power supply, thus degrading performance or even inducing functional errors because of nonzero wire resistance. Both types of failure can be triggered by insufficient wire widths. Although expanding the wire width alleviates electromigration and IR-drop, unlimited expansion not only increases the routing cost, but may also be infeasible due to the limited routing resource. In addition, electromigration and IR-drop manifest mainly in the power/ground (P/G) network. Therefore, taking wire widths into consideration is desirable to prevent electromigration and IR-drop at P/G routing. Unlike mature digital IC designs, P/G routing in analog ICs has not yet been well studied. In a conventional design, analog designers manually route P/G networks by implementing greedy strategies. However, the growing scale of analog ICs renders manual routing inefficient, and the greedy strategies may be ineffective when electromigration and IR-drop are considered. This study distances itself from conventional manual design and proposes an automatic analog P/G router that considers electromigration and IR-drops. First, employing transportation formulation, this article constructs an electromigration-aware rectilinear Steiner tree with the minimum routing cost. Second, without changing the solution quality, wires are bundled to release routing space for enhancing routability and relaxing congestion. A wire width extension method is subsequently adopted to reduce wire resistance for IR-drop safety. Compared with high-tech designs, the proposed approach achieves equally optimal solutions for electromigration avoidance, with superior efficiencies. Furthermore, via industrial design, experimental results also show the effectiveness and efficiency of the proposed algorithm for electromigration prevention and IR-drop reduction.

Categories and Subject Descriptors: J.6 [**Computer-Aided Engineering**]: *Computer-aided design*

General Terms: Algorithms, Design, Performance

Additional Key Words and Phrases: Analog, electromigration, IR-drop, power/ground network, routing, Steiner tree

ACM Reference Format:

Lin, J.-W., Ho, T.-Y., and Jiang, I. H.-R. 2012. Reliability-driven power/ground routing for analog ICs. *ACM Trans. Des. Autom. Electron. Syst.* 17, 1, Article 6 (January 2012), 26 pages.
DOI = 10.1145/2071356.2071362 <http://doi.acm.org/10.1145/2071356.2071362>

Authors' addresses: J.-W. Lin, Department of Computer Science and Information Engineering, National Cheng Kung University, No.1, University Road, Tainan City 701, Taiwan; email: dattle@eda.csie.ncku.edu.tw; T.-Y. Ho (contact author), Department of Computer Science and Information Engineering, National Cheng Kung University, No.1, University Road, Tainan City 701, Taiwan; email: tyho@csie.ncku.edu.tw; I. H.-R. Jiang, Department of Electronics Engineering and Institute of Electronics, National Chiao Tung University, No.1001, Ta-Hsueh Road, Hsinchu City 300, Taiwan; email: hui.ru.jiang@gmail.com.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from the Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701, USA, fax +1 (212) 869-0481, or permissions@acm.org.

© 2012 ACM 1084-4309/2012/01-ART6 \$10.00

DOI 10.1145/2071356.2071362 <http://doi.acm.org/10.1145/2071356.2071362>

1. INTRODUCTION

As the world adapts nanometer knowledge, the rapid advance of technology provides the capability to improve the speed and packing density of ICs. Device miniaturization requires scaling down the supply voltage and power consumption. In addition, as the feature size continues to shrink, allocating equal or more space for power/ground (P/G) networks on smaller-sized chips is still necessary to maintain reliability. Hence, controlling and distributing the P/G networks has already become an essential reliability issue in recent years. In digital IC designs, P/G routing can typically be generated and analyzed using power grids [Dutta and Marek-Sadowska 1989; Silva et al. 2008], a subject that has been studied extensively. Nevertheless, because of the specific design rules, analog layout automation is relatively arduous. For analog ICs, studies on the P/G routing problem are scant; routing is usually accomplished manually. Analog designers apply their skills and adopt greedy strategies to route P/G networks. However, the growing scale of analog ICs renders manual design inefficient, and greedy strategies ineffective. For P/G routing in analog ICs, two major design reliability concerns arise from electromigration and voltage drop (IR-drop) [Todri et al. 2007]. Electromigration can open or shorten a wire, thus leading to a permanent circuit failure; IR-drop may also result in insufficient power, hence degrading the performance or even inducing functional errors. Both types of failure are triggered by insufficient wire widths. Although expanding the wire width alleviates electromigration and IR-drop, unlimited expansion not only increases the routing cost, but may also be unfeasible due to the limited routing resource. In addition, electromigration and IR-drop manifest mainly in the power/ground (P/G) network. Therefore, routing the P/G network with elaborate considerations of wire widths is desirable to prevent electromigration and IR-drop.

1.1 Background

According to a previous study [Lienig 2006], electromigration may trigger the migration of metal atoms due to excessive current densities in interconnects. When the current density exceeds the tolerance of a metal, the atoms inside the metal suffer from two unbalanced forces: the electrostatic field, F_{field} , and the electron wind, F_{wind} [Lienig 2006]. F_{field} is caused by the electric field strength in an interconnect, while F_{wind} is generated by the momentum transfer between electrons and metal ions (atoms). Generally, with an over-stressed current density, F_{wind} is more powerful than F_{field} , as shown in Figure 1(a). The unbalanced forces cause the atoms in the metal to migrate in the direction of the flow of electrons. Over a time period, hillocks (short circuit) and voids (open circuit) are formed, as shown in Figure 1(b) and Figure 1(c), eventually resulting in a permanent circuit failure. The rapid downscaling of circuits has recently led to ever-shrinking interconnects, signifying that the wires suffer from even higher current densities. The shrinking metal also increases the seriousness of the electromigration issue. Electromigration often occurs in interconnects that bear large current flows, such as the P/G network in analog ICs. Black proposed an empirical model of electromigration to estimate the mean time to failure ($MTTF$) of a metal [Black 1969]:

$$MTTF = \frac{A}{J^n} \cdot \exp\left(\frac{E_a}{k \cdot T}\right), \quad (1)$$

where A is the material constant, J is the current density, E_a is the activation energy (material-dependent), k is the Boltzmann constant, T is the working temperature, and n is a scaling factor (usually set to 2 [Black 1969]). Equation (1) indicates that the current density (J) and the working temperature (T) greatly influence $MTTF$. The working temperature might sustain stability and lead $MTTF$ to depend only on

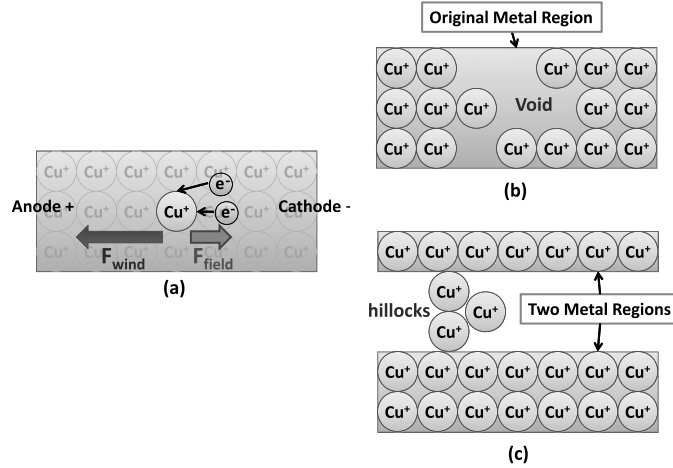


Fig. 1. Electromigration: (a) effect of F_{field} and F_{wind} for a Cu atom; (b) voids (open circuit) caused by electromigration; (c) hillocks (short circuit) caused by electromigration.

the current density. Hence, during the physical design stage, electromigration-aware routing should be applied; given the maximum tolerable current density of a metal, the router should ensure that the current flow within the metal does not exceed the limit. This study focuses on the current density as the major consideration to prevent electromigration. The current density J is defined as.

$$J = \frac{I}{A_{cross}} \leq J_{max}, \quad (2)$$

where I is the current flow, A_{cross} is the cross-sectional area, and J_{max} is the maximum current density. In modern IC designs, the wire thickness is almost a constant; hence A_{cross} is directly proportional to the wire width W_{wire} . Recent studies used the total wire area to estimate the routing cost [Jiang et al. 2010; Lienig and Jerke 2003; Yan and Chen 2008]. The wire area of each wire segment is defined as the multiplication of its wire length L_{wire} and its wire width W_{wire} :

$$A_{wire} = L_{wire} \times W_{wire}. \quad (3)$$

Although expanding the wire width helps alleviate electromigration, unlimited expanding wire widths increase the routing cost and might be infeasible for a congested design. According to Eq. (1), to meet the target $MTTF$, the maximum J can be obtained. Therefore, to prevent electromigration with the least routing cost, the wire width W_{wire} should be proportional to its current flow. This study defines the electromigration constraint as the wire width constraint and current density constraint—that is, if a wire is wider to load the current and verify that the current density is under the maximum current density, J_{max} and the width are under the maximum wire width w_{max} , and the electromigration safety is addressed.

In addition, IR-drop (voltage drop) is caused by excessive wire resistance. Copper has the second lowest electrical resistance of metal: $1.68 \times 10^{-8} \Omega \cdot m$. Although it is extremely small, when the current is flowing in the copper foils, the *small* resistance is still harmful to the supply voltage. This phenomenon, referred to as IR-drop, results in a voltage drop available at load devices, which is especially severe for low supply voltages for advanced processes. The IR-drop V_{drop} is defined as

$$V_{drop} = I \times R = I \times \left(\rho \times \frac{L_{wire}}{A_{cross}} \right), \quad (4)$$



Fig. 2. IR-drop: considering a simple power supply system with a source supply current with the voltage V_{supply} , the target receives the voltage with only $V_{load} = V_{supply} - V_{drop}$.

where ρ is the electrical resistivity of the material (measured in ohm-metres). Since the current flow at any terminal is a constant, the wider and shorter the wires, the lower the IR-drop. If a long wire forms an unfeasible resistance, the wire width can be adequately expanded to reduce the wire resistance and improve IR-drop. As shown in Figure 2, the voltage available at the load device V_{load} is V_{supply} minus V_{drop} .

1.2 Previous Work

Existing studies chiefly focus on electromigration avoidance [Jiang et al. 2010; Lienig and Jerke 2003; Yan and Chen 2008]. Lienig et al. developed a current-driven wire planning method based on the Delaunay triangulation mesh [Lienig and Jerke 2003]. Given the current source and sink terminals, the method begins with a mesh graph construction. The edges connecting the nearest pairs of terminals are subsequently greedily selected before the wire widths are determined accordingly. However, this method is unsuitable for all circuits. Because edges are only provided from the triangulation mesh, if certain nontriangulation edges are required, this method may not generate feasible solutions. Yan et al. proposed another greedy method that sorts the pairwise area gains of terminals and selects them in a nondecreasing order. This heuristic method may lead to suboptimal solutions [Yan and Chen 2008]. Jiang et al. first proved that this problem belongs to Class P instead of Class NP-hard [Jiang et al. 2010]. This means that electromigration-aware routing can be solved in polynomial time. Based on the greedy-choice property, they modeled the problem on a multisource multisink flow network and solved it optimally. Since their algorithm is strongly polynomial, they adopted a fast greedy method to obtain the initial flow. However, the results of this study show that the inferior initial solution costs more iterations of refinement to converge the flow into an optimal solution.

Figure 3 shows the routing results of the aforementioned works on an example with three current sources (terminals with positive current values) and four current sinks (terminals with negative current values) [Jiang et al. 2010]. In Figure 3(a), for example, a route is present from a source at (4, 6) to a sink at (5, 1). The wire area of this route is 24 (6 (wirelength) \times 4 (wire width) = 24). The minimum wirelength Steiner tree, shown in Figure 3(a), has the minimum wirelength (33), although without the minimum wire area (182). The wire area can be reduced to 154 and 144 by using the current-density-aware methods proposed by Lienig and Jerke [2003] and Yan and Chen [2008], respectively, as shown in Figure 3(b) and Figure 3(c). Figure 3(d) shows the optimal results from Jiang et al. [2010] of a minimum wire area 142. Compared with Jiang et al. [2010], this study proposes a more empirically efficient optimal algorithm for electromigration avoidance.

1.3 Our Contribution

As mentioned above, prior studies have chiefly focused on electromigration-aware wiring topology only. Since both electromigration and IR-drop are essential for P/G routing in analog ICs, this study constructed a P/G routing framework to overcome

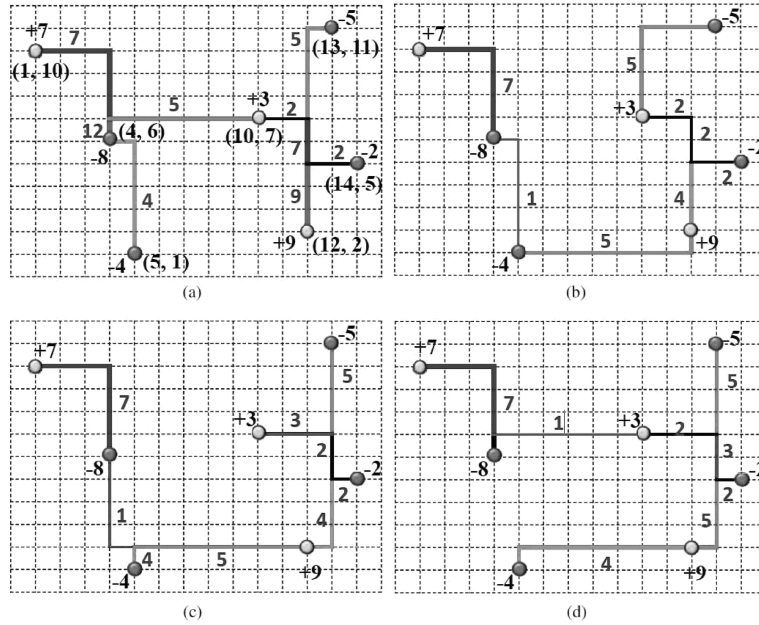


Fig. 3. Wire area comparison between previous algorithms and the proposed algorithm: (a) minimum wire-length = 33 and wire area = 182; (b) Lienig et al.: Wire area = 154; (c) Yan et al.: Wire area = 144; (d) the proposed algorithm and that of Jiang et al.: Wire area = 142 (optimal).

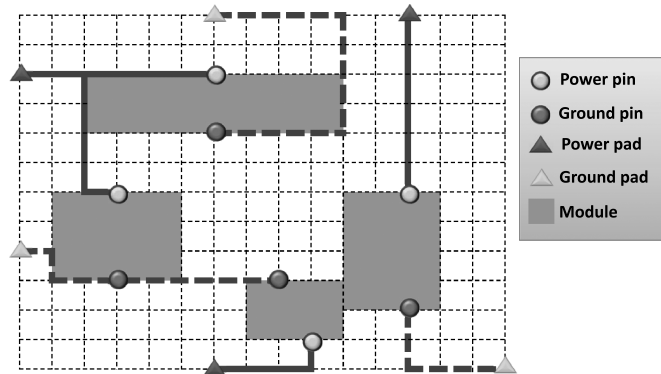


Fig. 4. Analog Power/Ground routing example.

electromigration and IR-drop: taking into consideration the maximum tolerable current density and IR-drop, sources and sinks with current values, and the feasible wire width range, the P/G network is routed to meet the current-density and IR-drop bounds. Figure 4 illustrates the analog P/G routing model. Each module contains two pins, one of which is the power pin, and the other is the ground pin. Around the chip, at least one pair of power and ground pads is present. This study provides each module with sufficient supplies while satisfying the electromigration and IR-drop constraints.

First, using the transportation formulation [Russell 1969; Winston 2004] for global P/G routing, this study constructed an electromigration-aware rectilinear Steiner tree

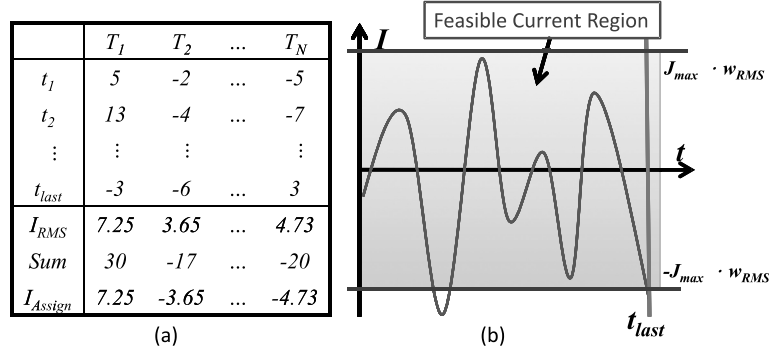


Fig. 5. (a) Current vector example. Terminal has value I_{Assign} ; (b) simulated current variation in the time period ($t \in [0, t_{last}]$). Wire is assigned to the terminal with wire width w_{RMS} . The feasible current region can be calculated using $J_{max} \cdot w_{RMS}$.

with the minimum routing cost. Second, for detailed P/G routing, without changing the solution quality, wires were bundled to release increased routing space to enhance routability and alleviating congestions. A wire width extension method was subsequently adopted to reduce wire resistance for IR-drop safety. Compared with high-tech methods, the proposed approach achieves equally optimal solutions for electromigration avoidance, with superior efficiencies. Furthermore, the experimental results show the effectiveness and efficiency of the proposed algorithm not only on electromigration, but also on IR-drop.

The major contributions of this study are as follows:

- This study proposes an automatic analog P/G router considering electromigration and IR-drop.
- This study adopted the transportation formulation to obtain an optimal solution with a superior efficiency to tackle electromigration.
- This study presents effective and efficient wire bundling and wire sizing methods to manage IR-drop.

The remainder of this article is organized as follows: Section 2 provides the problem formulation; Section 3 presents an overview of the proposed framework; Section 4 details the proposed algorithm for electromigration avoidance; Section 5 introduces the algorithm for IR-drop safety; Section 6 shows the experimental results; and last, Section 7 offers conclusions.

2. PROBLEM FORMULATION

This study proposes an automatic analog P/G router considering electromigration and IR-drop. The current is the chief concern for managing the electromigration and IR-drop. In practice, the current varies from each of the different terminals. Thus, the current model of Lienig and Jerke [2003] is adopted for analog ICs. The current value at each time interval is obtained via simulation and saves as a vector. Every terminal is considered with its root mean square (RMS) current value (I_{RMS}), which is calculated by using the vector. The total current in the vector is also calculated. The sum indicates whether the terminal tends to be a source ($I_{Assign} \geq 0$) or a sink ($I_{Assign} < 0$). An example is shown in Figure 5.

Based on the current flow I_{Assign} , the related wire width w_{RMS} is assigned to the terminal. According to Eq. (2), $I = J \cdot A_{cross} \propto J \cdot w$. The feasible current region (or interval) can be estimated as $[-J_{max} \cdot w_{RMS}, J_{max} \cdot w_{RMS}]$. If all current values in the vector are bounded by this interval, the electromigration effect can be avoided, and the IR-drop can also be reduced.

The analog P/G routing, considering the electromigration and IR-drop problem, can be formulated as follows:

Input

- A set of n modules is randomly distributed on the routing layer.
- Each module contains a power pin and a ground pin with their equivalent RMS current values.
- Each module contains the tolerant voltage drop V_{drop} .
- A set of m pairs of power pads and ground pads are present with their current values.
- The maximum current density J_{max} and the corresponding maximum wire width w_{max} are present.

Objective

- Automatically construct an analog P/G routing solution.
- Minimize the total wiring area.
- Reduce IR-drop.
- The power net and ground net should not overlap.

Output

- P/G routing topology for analog ICs

In each time period, the total current flow between terminals follows Kirchhoff's current law. If the RMS model is used, the law may not be satisfied. The following section first assumes that the total sum of the current follows the law. The final section shows the inequality between total supply current and total demand current.

Without a loss of generality, in the following section, this study regards the power pads as sources $\{S_1, S_2, \dots, S_m\}$, and the power pins as targets $\{T_1, T_2, \dots, T_n\}$ to develop the proposed algorithm.

3. OVERVIEW OF THE PROPOSED FRAMEWORK

Figure 6 shows an overview of the proposed algorithm. The routing procedure contains two stages: the global routing stage and the detailed routing stage. The global routing stage handles electromigration, while the detailed routing stage manages IR-drop.

The global routing stage involves consideration of the current flow and constructing an electromigration-aware rectilinear Steiner tree. This study defines the current-driven electromigration-avoidance rectilinear Steiner tree (CDEARST) problem. CDEARST is solved optimally via the transportation formulation. First, the wirelength between terminals is calculated while considering the presence of modules. Because wires cannot cross over modules, modules are regarded as obstacles before performing obstacle-avoiding routing. Second, according to the wirelength computation, the transportation problem is formulated. The basic feasible (BF) solution is obtained by using Russell's method [Russell 1969], and is verified with an optimality

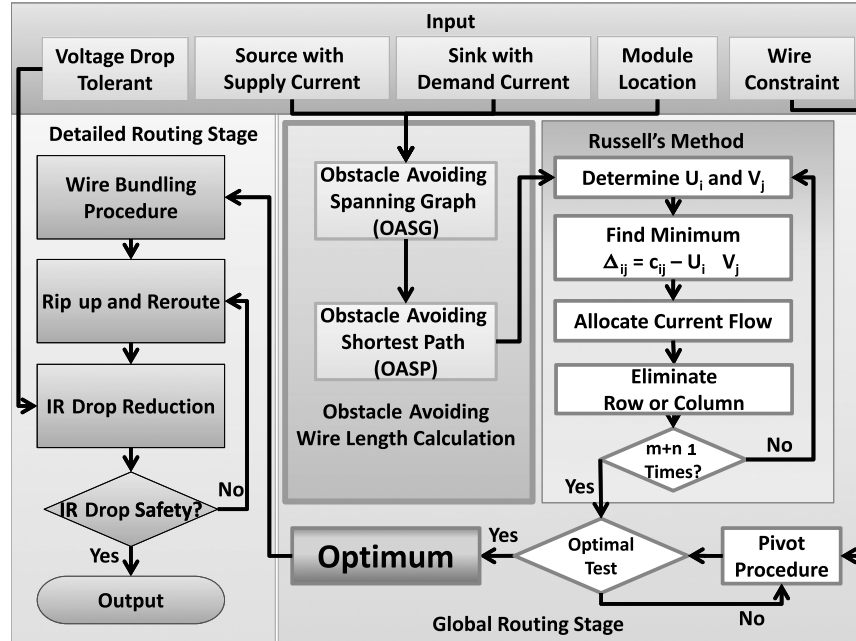


Fig. 6. Overview of our algorithm.

test. If the BF solution is not optimal, it is refined by the pivot procedure. Third, this study constructs power nets and ground nets by using the transportation formulation individually. In the global routing stage, this study did not handle the overlap between the power nets and ground nets; they will be removed in the detailed routing.

In the detailed routing stage, without altering the optimality, this study performs the wire-bundling procedure, which entails bundling the power (ground) net to release routing space for enhancing the routability and relaxing congestions. Upon performing the wire-bundling procedure, the congestion is analyzed. The negotiation-based routing technique is applied, which was introduced in PathFinder [McMurchie and Ebeling 1995] for rip-up and reroute. Finally, after obtaining a routing solution, this study analyzes the IR-drop for each route from the power pad to any power sink by using Eq. (4). The IR-drop value is compared with the tolerant voltage drop V_{drop} . If the IR-drop value exceeds V_{drop} , the wire width is expanded to increase A_{wire} , and the total IR-drop is reduced to prevent IC failures.

Figure 7 shows the P/G routing results of the benchmark RPG2. RPG2 has ten modules that are randomly distributed in the chip; each module contains two pins: one is the power pin (red), and the other is the ground pin (blue). Around the chip, four power pads and five ground pads are present. Figure 7(a) shows the optimal results of CDEARST after using the transportation approach. Figure 7(b) shows the results of wire bundling from the CDEARST. Figure 7(c) shows the example of our final routing result. Before we perform the realistic detailed routing and deal with the line overlapping, we must estimate the route and congestion. Figure 7(a) shows the global routing result, we allocate the current flow and obtain the initial routing congestion. Then, we further relax the congestion by wire bundling. The example is shown in Figure 7(b). In the two phases, we just estimate the routing resource, so we do not deal with the lines overlapping. That is the reason why vertical lines overlapping

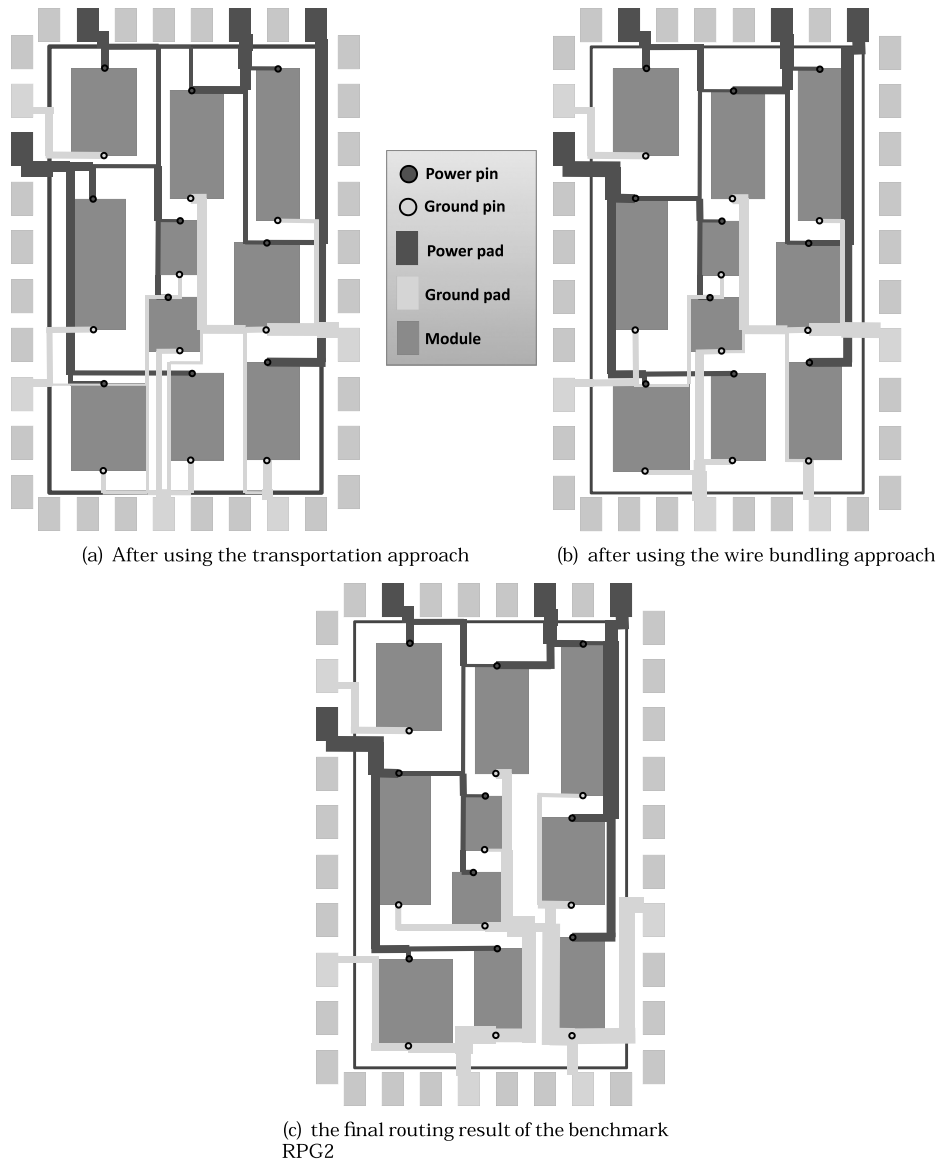


Fig. 7. P/G routing of RPG2.

each other in Figure 7(a) and Figure 7(b). We deal with overlapping problem by both negotiation-based routing method and rip-up and reroute in detailed routing.

4. ELECTROMIGRATION AVOIDANCE

With a set of modules that are randomly distributed on the routing layer, each module contains one power pin and one ground pin with their equivalent RMS current values, a set of power pads around the chip with their supply currents, and a set of ground pads around the chip with their tolerant currents. A CDEARST is constructed to connect

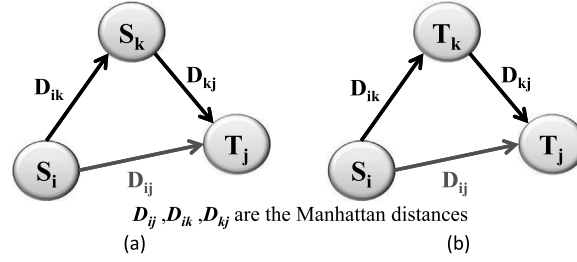


Fig. 8. (a) $S_i \rightarrow S_k \rightarrow T_j$ or $S_i \rightarrow T_j$; (b) $S_i \rightarrow T_k \rightarrow T_j$ or $S_i \rightarrow T_j$.

all power (ground) pads and power (ground) pins, enabling current sufficiency for each wire segment and minimizing the total wire area.

To tackle the CDEARST problem, obstacle-avoiding routing is first performed to compute the exact wirelength. Thereafter, this study formulates the CDEARST problem as the transportation problem, and adopts Russell's method and pivot procedure to obtain the solution. By using Russell's method, the basic feasible (BF) solution can be achieved before the solution is verified by using the optimality test. If the BF solution is not optimal, the pivot procedure is adopted to fix it. Prior to discussing the detailed algorithms, this study provides certain properties of CDEARST and provides alternate proof to show that the CDEARST problem belongs to Class P.

4.1 Property

The solution space of the CDEARST problem can be reduced by using the following properties:

Property 1. Only the current transportation that *directly* connects a source to a sink requires consideration.

PROOF. A current flow f_{ij} is assumed to exist from a source S_i to a sink T_j . Figure 8 shows that only two situations must be considered. Only vertical and horizontal lines are allowed for rectilinear routing. Each possible practical route between arbitrary two sources/sinks can be abstracted by the slant wire segment of the length equal to their Manhattan distance. The wire area is obtained by multiplying f_{ij} by the Manhattan distance D_{ij} . Since f_{ij} is fixed, only the Manhattan distance of the wire requires consideration. From the triangle inequality, $D_{ik} + D_{kj} \geq D_{ij}$. Hence, the optimal solution is the direct connection between a source and a sink. \square

Before this problem is proven to belong to Class P, the transportation problem is introduced as follows:

A set of m *supply points* from which goods are shipped— the supply point i can supply most S_i units. A set of n *demand points* to which the goods are shipped—the demand point j must receive a minimum of D_j units. Each unit produced at the supply point i and shipped to the demand point j incurs a *cost* of C_{ij} . Manage the determination of a minimum-cost plan for transporting goods from a number of sources to a number of destinations.

Definition 1. A *balanced transportation* means that the total supply equals the total demand.

Table I. Isomorphism of Two Problems

CDEARST	Transportation Problem
Source terminals	Supply points
Sink terminals	Demand points
Source RMS current	Supply S_i units good
Sink RMS current	Demand D_j units good
Distance between terminals c_{ij}	Transportation cost C_{ij}
Wire width w_{ij}	Shipped x_{ij} good
Minimum wire area $\sum c_{ij} \cdot w_{ij}$	Minimum cost $Z = \sum C_{ij} \cdot x_{ij}$

A balanced transportation problem can be solved optimally by using linear programming in polynomial time. x_{ij} is the number of units of goods being transported from the i th source to the j th demand. The formulation follows:

$$\min Z = \sum_{i=0}^m \sum_{j=0}^n C_{ij} \cdot x_{ij} \quad (5)$$

$$\text{s.t.} \quad \sum_{j=1}^n x_{ij} = S_i \quad (i = 1, 2, \dots, m) \quad (6)$$

$$\sum_{i=1}^m x_{ij} = D_i \quad (j = 1, 2, \dots, n) \quad (7)$$

$$x_{ij} \geq 0 \quad (i = 1, 2, \dots, m; j = 1, 2, \dots, n) \quad (8)$$

Property 2. The Current-Driven Electromigration-Avoidance Rectilinear Steiner Tree (CDEARST) construction belongs to P-class instead of NP-class.

PROOF. Because the transportation problem belongs to P-class, a transformation method in polynomial time for both inputs are constructed before the two problems are isomorphic (or equivalent). The transformation is shown in Table I. \square

The CDEARST problem thus belongs to P-class, signifying that it can be solved optimally in polynomial time.

Even when only a single wire width is available, this problem is still different from the original Steiner tree routing problem. Figure 9 shows the difference. The problem seems identical when the wire width equals 1. However, differences are present. In Figure 9(a), the route is superposed to reduce the wirelength, which helps decrease the total wirelength without affecting the total wire area. Figure 9(a) and Figure 9(b) are both optimal wire area solutions. If the wire width is constrained to 1, Figure 9(b) is the appropriate solution. Hence, the CDEARST can be solved in polynomial time, although this does *not* mean that the original Steiner tree routing problem can be solved in polynomial time.

Instead of using the traditional method, (e.g., simplex method) to solve the transportation linear programming, an alternative approach is proposed to obtain an optimal solution efficiently.

4.2 Obstacle-Avoiding Wirelength Calculation

Before solving the current distribution problem, the distance from each source (supply) to each sink (destination) must be known. Lin et al. [2008] presented an efficient method to construct the obstacle-avoiding spanning graph, which can be used

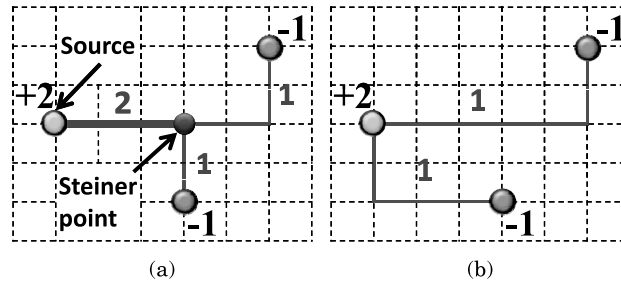


Fig. 9. Difference between the proposed routing and the original Steiner tree routing: (a) wirelength = 9 and wire area = 12; (b) wirelength = 12 and wire area = 12.

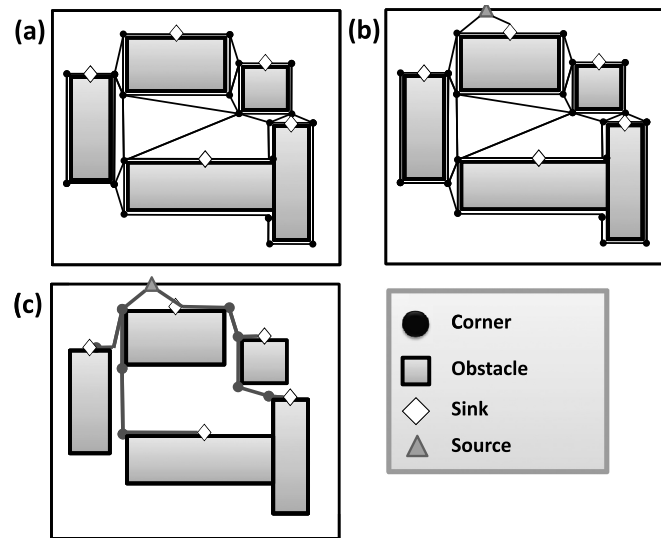


Fig. 10. Obstacle-avoiding wirelength calculation: (a) obstacle-avoiding spanning graph (OASG); (b) adding a source to the OASG; (c) obstacle-avoiding shortest path (OASP).

to calculate the wirelength. However, this study proved that only edges with direct connections between a source and a sink require consideration. Therefore, this study adopted the method in the study by Lin et al., with certain modifications. The modified obstacle-avoidance method comprises (1) an obstacle-avoiding spanning graph (OASG); and (2) an obstacle-avoiding shortest path (OASP). When the route must perform obstacle-avoiding routing, the OASG step is used. Thereafter, the OASP step is applied to obtain the distance between the single source to all sinks.

4.2.1 Obstacle-Avoiding Spanning Graph (OASG). An OASG is a connected graph that connects all vertices without any edges intersecting obstacles. For speeding up the shortest-path algorithm, this study prevents the algorithm from calculating the distance between any two sources or any two sinks. This step involves constructing the OASG with all sinks and obstacle corners. Thereafter, the source is added to the OASG to obtain the distance between the source to all sinks. Upon reading the input data, the distribution of all obstacles and sinks are analyzed, and the corresponding OASG is constructed, as shown in Figure 10(a). V is a set of vertices of all obstacle corners and U is a set of all sinks. The plane of each vertex v in $V \cup U$ is divided into four

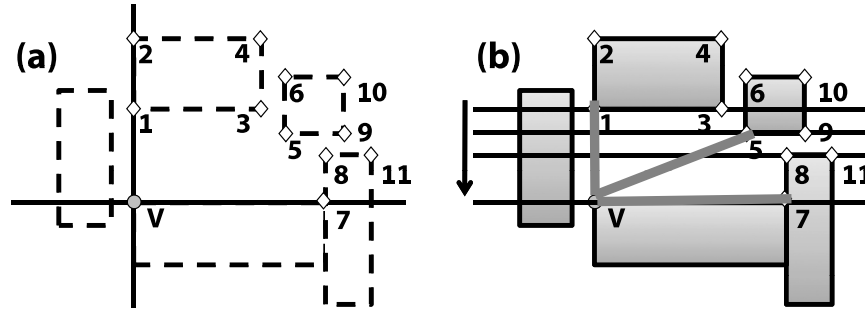


Fig. 11. (a) Example of the upper-right corner of V ; (b) using the horizontal sweeping line approach, with points 1, 5, and 7 connected to the reference point V .

regions, as shown in Figure 11(a). As defined in the study by Lin et al. [2008], a vertex f is a neighbor of v if no other vertex in $V \cup U$ or obstacle is inside or on the bounding box of v and f . The OASG is constructed by using a sweep line algorithm. Their X and Y coordinates are sorted, as shown in Figure 11(a). A horizontal sweep line technique is subsequently implemented to find all possible connections of neighbor corners, as illustrated in Figure 11(b). The details of the sweep line algorithm can be found in the study by Lin et al. [2008], which proved that the OASG implies the shortest path of any two vertices in $V \cup U$. This property can help find the shortest obstacle-avoiding path quickly.

4.2.2 Obstacle-Avoiding Shortest Path (OASP). Based on the graph topology, the Manhattan distance between the source and sinks is obtained. After constructing the OASG, a source is added to the graph and the sweep line algorithm is applied to connect the source and the original OASG. An example of the result of this step is shown in Figure 10(b). Finally, the shortest path from the source to the sink is obtained, which can be implemented using Dijkstra's single source shortest path algorithm [Cormen et al. 2009]. The same procedure is performed for all the sources until all the distances between the sources and sinks are obtained. The results are shown in Figure 10(c).

4.3 Transportation Problem

Upon calculating the distance between each source and sink, the current can be distributed. The linear programming solver (e.g., simplex method) is used to obtain the optimal solution. However, this problem tends to have a large number of constraints and variables. Using the traditional simplex method may require extensive computation. Fortunately, certain efficient methods have been proposed [Winston 2004]. This study presents the transportation simplex method in the following sections, that is, a special streamlined algorithm for solving the problem. First, a discussion of how to find a basic feasible (BF) solution is provided. Second, the method for pivoting the BF solution into an optimal solution is shown. Finally, this study describes how to manage the wire width constraint using a separation method.

4.3.1 Basic Feasible Solutions

Definition 2. Basic variables are obtained by the simultaneous solution of the system of functional constraints. The other variables are nonbasic variables equal to zero.

For a linear programming system, the number of basic variables generally equals the number of functional constraints. For a transportation formulation with m sources and n destinations (sinks), the number of functional constraints is $m + n$. However, if a set

```

Algorithm : Russell's approximation method
Input : Supply =  $\{S_1, \dots, S_m\}$ , Demand =  $\{D_1, \dots, D_n\}$ ,
        Cost =  $\{C_{11}, C_{12}, \dots, C_{mn}\}$ 
Output : Initial BF solution matrix,  $Flow_{m \times n}$ 
1  Row =  $\{1, 2, \dots, m\}$ ; Col =  $\{1, 2, \dots, n\}$ ;
2  For  $k = 1$  to  $m + n - 1$  do
3    For  $i = 1$  to  $m$  do
4      For  $j = 1$  to  $n$  do
5        cost =  $C_{ij}$ ;
6        if (cost >  $U_i$ )
7           $U_i =$  cost;
8        if (cost >  $V_j$ )
9           $V_j =$  cost;
10       end
11     end
12     For  $i = 1$  to  $m$  do
13       For  $j = 1$  to  $n$  do
14          $\Delta_{ij} = C_{ij} - U_i - V_j$ ;
15       end
16     end
17     Find minimum  $\Delta_{ij}$  &&  $i \in Row, j \in Col$ ;
18     if ( $S_i \geq D_j$ )
19        $S_i = S_i - D_j$ ;  $D_j = 0$ ;
20        $Flow_{ij} = D_j$ ;  $Col \setminus j$ ;
21     else
22        $D_j = D_j - S_i$ ;  $S_i = 0$ ;
23        $Flow_{ij} = S_i$ ;  $Row \setminus i$ ;
24     end
25   return Flow

```

Fig. 12. Russell's approximation method.

of values satisfies all but one of the constraints of a balanced transportation problem, the values automatically satisfy the other constraint [Winston 2004]. In other words, only $m + n - 1$ basic variables that yield a basic solution for transportation problems are required. This study illustrates Russell's approximation method [Russell 1969] to find the $m + n - 1$ basic variables for a BF solution.

Figure 12 details Russell's method. In each iteration, the unit cost C_{ij} must be considered to obtain the optimal solution. In the proposed approach, the unit cost C_{ij} is identified as the Manhattan distance between sources and sinks. Figure 13 shows an example of constructing the table. In Figure 13(a), the source with coordinate (12, 2) is used as the example, with the Manhattan distance (unit cost) calculated from (12, 2) to every sink. The result in the table are subsequently shown in Figure 13(b). For each source row i (destination column j), the largest unit cost C_{ij} that still remains in the row (column) is obtained and set as U_i (V_j). Using Figure 14(a) as the example, the column (13, 11) displays three unit costs (13, 7, and 10). Because the supply is exhausted (i.e., equal to 0), the unit cost (13) is not considered. Hence, the largest unit cost still remaining in the column (13, 11) is 10. $V_4 = 10$ is set, as shown in Figure 14(a). Thereafter, $\Delta_{ij} = C_{ij} - U_i - V_j$ is calculated. This value indicates the critical entry, which is chosen as a basic variable. When Δ_{ij} is small, the cost C_{ij} is also small, and the largest unit costs for the row (U_i) and column (V_j) are large. If the minimum Δ_{ij} is chosen as a basic variable entry, distributing the current with the small cost is identical. Furthermore, this also prevents distributing the current with the high cost in the row (U_i) and column (V_j). Upon selecting the minimum Δ_{ij} , the allocation is sufficiently enlarged to use up exactly the remaining supply in its row or the remaining demand in its column. Rows or columns that have smaller remaining supplies or demands are removed from consideration. If rows and columns have the same remaining supplies and demands, one is arbitrarily selected. Figure 14(a) shows the third iteration. The values of $\Delta_{ij} = C_{ij} - U_i - V_j$ are listed below the unit cost. In this

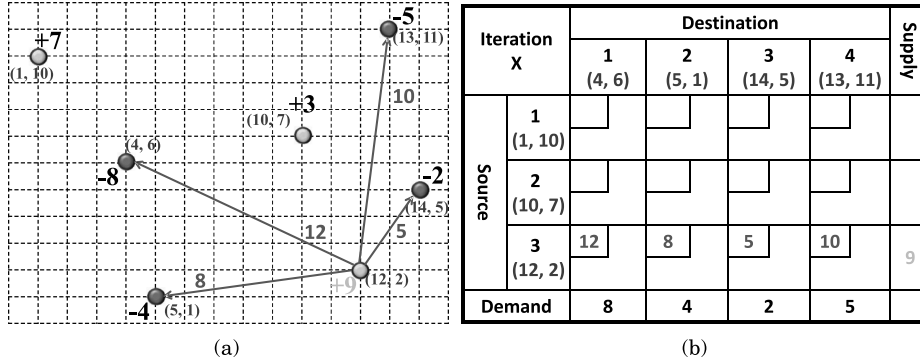


Fig. 13. An example of how to construct the table.

Iteration 3		Destination				Supply	U_i
		1 (4, 6)	2 (5, 1)	3 (14, 5)	4 (13, 11)		
Source	1 (1, 10)	7 7	13	18	13	0	-
	2 (10, 7)	7 -16	11 -11	6	7 -14	3	11
	3 (12, 2)	12 -12	8 -15	5 2	10 -12	7	12
Demand		1	4	0	5		
	V_j	12	11	-	10		

(a) Iteration 3: Each row (column) contains their remained supply (demand) and related U_i (V_j). In each entry, their related cost is shown. The values of $\Delta_{ij} = C_{ij} - U_i - V_j$ are below the cost. The minimum $\Delta_{21} = 16$ is chosen and allocated $x_{21} = 1$

No.	U_1	U_2	U_3	V_1	V_2	V_3	V_4	Min Δ_{ij}	Allocation
1	18	11	12	12	13	18	13	$\Delta_{33}=-25$	$x_{33}=2$
2	13	11	12	12	13	-	13	$\Delta_{11}=-18$	$x_{11}=7$
3	-	11	12	12	11	-	10	$\Delta_{21}=-16$	$x_{21}=1$
4	-	11	10	-	11	-	10	$\Delta_{24}=-14$	$x_{24}=2$
5	-	-	10	-	8	-	10	$\Delta_{32}=-10$	$x_{32}=4$
6	-	-	10	-	-	-	10	$\Delta_{34}=-10$	$x_{34}=3$

(b) example of each iteration procedure, their related Δ_{ij} , and allocation

Fig. 14. Example of Russell's approximation method.

iteration, the minimum $\Delta_{21} = 16$ is chosen and $x_{21} = 1$ is allocated before proceeding to the next iteration. Figure 14(b) shows the entire iteration procedure.

The complexity of Russell's method is $O(m * n)$ for each iteration. Although another recognized method can be used, Vogel's method [Winston 2004], has a complexity

$O(m+n)$ for each iteration. In most cases, especially when $m+n$ is large, Russell's method yields a superior BF solution. This helps reduce testing time whether the BF solution is optimal, and also requires fewer pivots to achieve the optimum solution, thus used to find the BF solution.

4.3.2 *Optimality Test.* The LP problem can be written as follows:

$$Z = \mathbf{C}_{BV} \cdot x_{BV} + \mathbf{C}_{NBV} \cdot x_{NBV} \quad (9)$$

$$s.t. \quad B \cdot x_{BV} + N \cdot x_{NBV} = b \quad (10)$$

$$x_{BV}, x_{NBV} \geq 0, \quad (11)$$

where \mathbf{C}_{BV} (\mathbf{C}_{NBV}) is the row vector of the objective function coefficients for basic (nonbasic) variables. x_{BV} (x_{NBV}) represents the basic (nonbasic) variables, and B (N) is the matrix, with its j th column being the constraint column for basic (nonbasic) variables. b is the right-hand side of the constraint. The constraint is multiplied (10) by $\mathbf{C}_{BV}B^{-1}$ and added to (9). The optimal row is obtained as follows:

$$Z + (\mathbf{C}_{BV}B^{-1}N - \mathbf{C}_{NBV}) \cdot x_{NBV} = \mathbf{C}_{BV}B^{-1}b. \quad (12)$$

Hence, the coefficient of x_{ij} , called \bar{c}_{ij} , is obtained in the optimal row with the optimal value $\mathbf{C}_{BV}B^{-1}b$.

$$\bar{c}_{ij} = \mathbf{C}_{BV}B^{-1}a_{ij} - c_{ij} \quad (13)$$

where a_{ij} is the column of constraints, and c_{ij} is the objective function coefficient for x_{ij} . If x_{ij} is increased by 1, the cost directly increases by c_{ij} . In addition, increasing x_{ij} by 1 is equivalent to reducing the right-hand sides of the i th supply constraint and the j th demand constraint by 1. This increases the Z value $-u_i - v_j$. Therefore, if x_{ij} is increased by 1, Z increases by a total of $c_{ij} - u_i - v_j$. If a nonbasic variable x_{ij} has $\bar{c}_{ij} = u_i + v_j - c_{ij} > 0$, Z can be decreased by \bar{c}_{ij} per unit. Because a minimization problem is solved, the BF solution is optimal if all of the \bar{c}_{ij} values are negative. Otherwise, the biggest positive entry must be pivoted into a basic variable. The \bar{c}_{ij} value can be determined after $\mathbf{C}_{BV}B^{-1}$. Since the $m+n$ equations have one redundant equation that can be deleted without altering the feasible region, a constraint was dropped to render $\mathbf{C}_{BV}B^{-1}$ with only $m+n-1$ elements. The first supply constraint is dropped, and the following assumption is made:

$$\mathbf{C}_{BV}B^{-1} = [u_2 \ u_3 \ \cdots \ u_m \ v_1 \ v_2 \ \cdots \ v_n] \quad (14)$$

where u_2, \dots, u_m correspond to $m-1$ supply constraints and v_1, v_2, \dots, v_n correspond to n demand constraints. In the optimal row, the coefficient \bar{c}_{ij} of the basic variable x_{ij} must equal zero ($\bar{c}_{ij} = 0$). Thus,

$$\mathbf{C}_{BV}B^{-1}a_{ij} - c_{ij} = 0 \quad (15)$$

can be assigned to determine $m+n-1$ variables in $\mathbf{C}_{BV}B^{-1}$. Due to the special constraint of the transportation problem, the number of calculations can be reduced. For each basic variable x_{ij} except $i=1$, the equation $u_i + v_i = c_{ij}$ can be obtained. If setting $u_1 = 0$, the total equations are followed for any i and j . Therefore, to solve $\mathbf{C}_{BV}B^{-1}$, only the following simultaneous equations require solving:

$$\begin{cases} u_1 & = 0 \\ u_i + v_j & = c_{ij} \\ 1 \leq i \leq m, 1 \leq j \leq n, x_{ij} \in x_{BV} \end{cases} \quad (16)$$

	$v_1=6$	$v_2=9$	$v_3=8$	$v_4=12$		$v_1=6$	$v_2=9$	$v_3=8$	$v_4=12$
$u_1=0$	6	9	11	12		6	9	11	12
	5	$+0$	-3	-0		5	7	-3	-1
$u_2=-3$	8	6	5	8		8	6	5	8
	-5	-0	1	$+0$		-5	6	1	7
$u_3=-2$	11	8	6	10		11	8	6	10
	-7	-1	10	$+0$		-7	-1	10	-1

(a)
(b)

Fig. 15. (a) $\bar{c}_{ij} = u_i + v_j - c_{ij}$ with a signed operation on the table. The most positive entry is $\bar{c}_{24} = +1$, which is chosen as an enter variable to form a loop. $\theta = \min(7, 13) = 7$; (b) when all the $\bar{c}_{ij} \leq 0$, the optimal solution is obtained.

After obtaining each u_i and v_j , $\bar{c}_{ij} = u_i + v_j - c_{ij}$ can be determined for each nonbasic variable. If $\bar{c}_{ij} \leq 0$ for all nonbasic variables, the BF solution is optimal. Otherwise, the most positive \bar{c}_{ij} is chosen as an enter variable and the pivoting procedure is performed. Figure 15 shows an example of the procedure and the unsigned value in the entry is the original flow value. Figure 15(a) lists the values u_i and v_j after solving the simultaneous equations in Eq. (16). $\bar{c}_{ij} = u_i + v_j - c_{ij}$ is subsequently determined. The entry at the cross of column 1 and row 2 is used as an example. $c_{ij} = c_{12}$ is the unit cost, and equal to 8. Hence, $\bar{c}_{ij} = \bar{c}_{12} = u_1 + v_2 - c_{12} = (-3) + (6) - (8) = -5$ is obtained. \bar{c}_{ij} is listed with a signed value under the unit cost in Figure 15(a).

Definition 3. A loop is an ordered sequence that contains at least four different entries, and satisfies the following:

- (1) Any two continuous entries lie in either the same row or column.
- (2) Fewer than three entries lie in the same row or column.
- (3) The last entry lies in the same row or column as the first entry.

This study starts at the enter variable x_{ij} before moving along the row or column. When a basic variable is reached, another direction is taken. This procedure is repeated until x_{ij} is reached to form a loop. Thereafter, the smallest value θ is found among all odd entries in the loop. The entries with θ are removed from the basic variables. To perform the pivot, θ is decreased for all odd entries, and increased for all even entries. The superior BF solution is obtained; Figure 15 shows the procedure.

Before adding the wire-size constraint, an approach to solve the unbalance transportation is proposed. In this problem, the sum of supplies is not equal to the sum of demands. If the total supply exceeds total demand, a dummy demand terminal can be added to balance the transportation problem. Let the demand of the dummy terminal be equal to the excessive supply, which can also be accomplished when the total demand exceeds the total supply. Because the shipment from or to the dummy terminal is not a real shipment, a cost of zero is assigned. With the addition of the dummy terminal, this problem can be solved using the aforementioned method. Finally, the dummy terminal can be removed. Although some sources may remain and a certain current or sinks may receive insufficient current, a high-quality electromigration-avoiding routing tree can be obtained.

4.3.3 Wire-Size Constraint Model. In modern IC design, only certain wire sizes can be used. If the wire size is not constrained, an unfeasible solution may be obtained. This study presents a method for preventing the current flow from exceeding the maximum

		Destination					
		T ₁	T ₂	T ₃	T ₄		
Source	S ₁ ¹	7	5	13	18	13	
	S ₁ ²	M	2	13	18	13	
	S ₂	7	1	11	6	7	2
	S ₃	12	8	4	5	2	10

(a)

		Destination					
		T ₁	T ₂	T ₃	T ₄		
Source	S ₁ ¹	7	5	13	18	13	
	S ₁ ²	M		13	18	13	2
	S ₂	7	3	11	6	7	
	S ₃	12	8	4	5	2	10

(b)

Fig. 16. (a) $w_{max} = 5$ and the source S_1 is separated into S_1^1 and S_1^2 . The cost from S_1^2 to T_1 is set to M ; (b) optimal solution with wire constraint. Wire area = 154.

wire size w_{max} . Upon obtaining the optimal solution, the feasibility of the current flow is checked. If a source contains a current of more than w_{max} , the excess current is diverted to sinks.

To prevent the source S_i from supplying current that exceeds P_i , S_i is separated into $\{S_i^1, S_i^2, \dots, S_i^n\}$, with supply $P_i^k = w_{max}$, $1 \leq k \leq n-1$, where $n = \lceil P_i/w_{max} \rceil$ and $\sum_{k=0}^{n-1} P_i^k + P_i^n = P_i$. Thereafter, a new table is constructed for the transportation problem. Based on the optimal solution, if S_i supplies the sink T_j with current flow, $F_{ij} > w_{max}$. In the new extended table, S_i^m is randomly chosen to supply the flow w_{max} with the original cost. The other $S_i^k, k \neq m$, are set to T_j with a large cost $M \rightarrow \infty$, and the remaining flow is distributed to those entries evenly. A large cost prevents S_i from supplying currents to the sink T_j with a large cost. Upon construction of the new table, the pivot procedure is used to obtain the optimal solution under the wire constraint. Figure 16 shows the results with $w_{max} = 5$.

Upon obtaining the optimal solution, separate sources that flow to the same sink are combined into one flow value, since they actually start at S_i and reach the same target. Separating sources into several parts is the same as adding new constraints to the problem. The number of basic variables exceeds $m + n - 1$. Although a nontree topology may be obtained, electromigration is avoided if the solution is feasible. In certain cases, even with sufficient M , a flow under the huge cost still exists. This situation implies that the solution is unfeasible, and is caused by an improper wire-size constraint. In other words, with the given wire-size constraint, avoiding electromigration is impossible, and thus must be resolved. Figure 7(a) shows the result after using the transportation approach.

5. IR-DROP REDUCTION

Although the optimal CDEARST of the power net and ground net are constructed separately, the wires may overlap. To handle this problem, this study performed detailed routing and IR-drop reduction, as shown in the following sections.

Without changing the solution quality, wires are bundled to free routing space to enhance the routability and relax congestion. After performing the wire-bundling procedure, this article adopts the negotiation-based routing technique for detailed routing. Finally, the wire-width extension method is adopted to decrease the wire resistance for IR-drop reduction. The detailed algorithm is discussed in the following section.

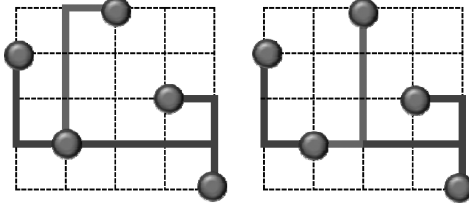


Fig. 17. An example of L-shape bundling.

Algorithm : MinCost-L-RST ($\Phi_x(v)$)
Input : Tree T
Output : Minimum cost spanning tree

- 1 If v is a leaf in T_r , return $\Phi_x(v)$ as the x L-shape of e_v .
- 2 If v is not a leaf and not the root, then fix the layout of e_v to be the x L-shape.
- 3 Recursively computes $\Phi_u(w_i)$ and $\Phi_l(w_i)$ for the children w_i , $i = 1, 2, \dots, d$ of v .
- 4 If there is no enough routing space for wire width, the wire bundle failed, and set the high cost for $\Phi_u(w_i)$ and $\Phi_l(w_i)$.
- 5 For each of the 2^d different ways of selecting either the L-RST $\Phi_u(w_i)$ or $\Phi_l(w_i)$, computes the total amount of overlap in the resulting L-RST of T_v^+ , as the sum of the overlaps in the star T_v^S and the amount of overlaps in the selected L-RST's of the children. Return $\Phi_u(v)$ to be the L-RST that maximizes the total amount of overlap.

Fig. 18. MinCost-L-RST ($\Phi_x(v)$).

5.1 Wire Bundling

After constructing the power nets and ground nets, the nets of the same type are bundled when possible. Figure 17 shows an example of the wire-bundling procedure. This study maximizes the overlaps via a dynamic programming approach, which is proposed in the study by Ho et al. [1990]. Bundling segments do not affect the total wire area, though they can alleviate the congestion. Furthermore, the bundling approach is beneficial for reducing IR-drop, since it can free additional space to extend the wire width. Figure 7(b) shows the results after using the wire-bundling approach.

The dynamic programming method used by Ho et al. [1990] is shown below. The input tree is rooted at any point r , resulting in the rooted tree T_r . For a point set S , and a point $v \in S$, the subtree is denoted (i.e., star), which is induced by v and its neighbors as T_v^S . The edge incident is also denoted from its parent on v as e_v and $T_v^+ = T_v^S \cup e_v$. The procedure of MinCost-L-RST computes the minimum cost L-RST $\Phi_x(v)$, where x is either the upper L-shape u or the lower L-shape l . If there is no enough routing space for wire width, the wire bundle failed, and set the high cost for $\Phi_u(w_i)$ and $\Phi_l(w_i)$. d is the number of children of the point v in the rooted tree T_r . The procedure is shown in Figure 18.

5.2 Detailed Routing

If no overlapping is present between any power net and ground net, the feasible solution is obtained; otherwise, the congestion is analyzed. This study adopted the negotiation-based routing technique, which was introduced in PathFinder [McMurchie and Ebeling 1995]. The history-based cost function for estimating the cost of an edge r is defined as follows:

$$cost_r = b_r + h_r \times p_r, \quad (17)$$

Table II. Complexity of the Proposed Algorithm and the Three High-Tech Algorithms

	Method	Complexity
[Lienig and Jerke 2003]	Delaunay triangulation based and greedy choose edges	$O((m+n) \lg(m+n))$
[Yan and Chen 2008]	Greedy width assignment for source-sink pair	$O(m^2n)$
[Jiang et al. 2010]	Greedy method for initial solution	$O((mn) \lg(mn))$
	Negative-cycle detection in flow network for optimal solution	$O(kmn)$
Proposed	Russell's approximation method for initial solution	$O((mn)(m+n))$
	Pivot procedure for optimal solution	$O(kmn)$

m: # of source n: # of sink k: # of refinement

where b_r is the base cost of using the edge r and is set to its current flow (i.i., one unit of wire area), and $h_r \times p_r$ is the congestion cost of an edge r . The historical term h_r is updated in the following manner during iterations:

$$h_r^{i+1} = \begin{cases} h_r^i + 1 & \text{if } r \text{ has overflow} \\ h_r^i & \text{otherwise} \end{cases} \quad (18)$$

where i is the iteration count and $h_g^1 = 1$. The penalty term p_r is defined as follows:

$$p_r = \left(\frac{d(r) + 1}{s(r)} \right)^k, \quad (19)$$

where $s(r)$ is the supply of an edge r and represents the number of available routing tracks; $d(r)$ is the demand on the edge and the number of wires that utilize an edge r ; and k is a constant that controls the rising rate of p_r .

In the highly congested region, since the detoured net with small current flow may slightly increase the wire area, rip-up and reroute segments with low current flow are conducted. The ripped-up net is first rerouted by adopting the monotonic routing approach [Pan and Chu 2006]. This study performs the reroute procedure within the bounding box without any detour. This method would not increase the total wire area. If the monotonic routing method fails to find an overflow-free path, the detour constraint is released to compose the reroute procedure for detouring the segment within the bonding box. However, if the procedure still fails, the segment reroute is composed without a bounding box constraint. This study continues using the same congested region procedure to each of the remaining nets, and rips-up and reroutes the marked nets.

5.3 Wire Sizing

This step involves analyzing the IR-drop for each route from the power pad to any power sink by using Eq. (4). The current I is distributed by performing the transportation approach, and the cross-sectional area A_{wire} is proportional to the current. The electrical resistivity of the material ρ is provided in the input. Upon obtaining the routing solution, the total wire length L_{wire} can be estimated with ease. By using Eq. (4), the IR-drop for each net is obtained. This study compares the IR-drop value with the tolerant voltage drop V_{drop} of each module. If the IR-drop value exceeds V_{drop} ,

Table III. Comparison of the Proposed Algorithm and Three High-Tech Algorithms

Bench- mark	m	[Lienig and Jerke 2003]	[Yan and Chen 2008]	[Jiang et al. 2010]	Ours	Improvement of Wire-Area			
		Wire-Area (A)	Wire-Area (B)	Wire-Area (C)	Wire-Area (D)	A	B	C	D
INP1	7	154	144	142	142	1.08	1.01	1	1
INP2	16	122	98	90	90	1.36	1.09	1	1
INP3	10	210	128	116	116	1.81	1.10	1	1
INP4	16	32	40	32	32	1	1.25	1	1
IND1	10	6661	5319	5301	5301	1.26	1.01	1	1
IND2	10	79400	79000	74200	74200	1.07	1.06	1	1
IND3	10	Fail	6181	5513	5513	Fail	1.12	1	1
IND4	25	23112	16000	13728	13728	1.68	1.17	1	1
IND5	33	31466	20814	17044	17044	1.85	1.22	1	1
RT01	75	Fail	189437	144071	144071	Fail	1.31	1	1
RT02	180	Fail	10638884	7151286	7151286	Fail	1.49	1	1
RT03	303	Fail	3360716	2325806	2325806	Fail	1.44	1	1
RT04	475	Fail	6475941	4205757	4205757	Fail	1.54	1	1
RT05	850	Fail	59207240	37318054	37318054	Fail	1.59	1	1

m: # of terminals

the wire width can be expanded to increase A_{wire} , to reduce the total IR-drop value. The wire width is extended until the IR-drop value is smaller than the V_{drop} . When we use the wire-sizing method for reducing the IR-drop, we check whether there is enough routing resource for the wire-width expansion or not. If there are some obstacles blocking the wire sizing, we first try to separate the wire from the source into two or more nets as we mentioned in Section 4.3.3 and Figure 16. Then, we route all the separated nets on the region and avoid the blocked obstacles. However, if there is not enough routing space for the separated nets, for the IR-drop safety, we rip-up and reroute the original net (i.e., the net without separation). Then, the negotiation-based routing would obtain another routing solution. In certain cases, even when we adopt rip-up and reroute the method, we still cannot obtain an IR-drop safety routing solution. That means, with the given infeasible IR-drop threshold, avoiding IR-drop is impossible, and the chip must be redesigned. Figure 7(c) shows the routing results.

5.4 Complexity

Table II shows the comparison between the proposed algorithm and three high-tech algorithms. The algorithm proposed by Lienig and Jerke [2003] is with the lowest complexity. However, the algorithm may fail to route or obtain a huge wire-area solution. The algorithm by Yan and Chen [2008] can route all benchmarks, although the complexity is the greatest, and the runtime is extremely large. The algorithm by Jiang et al. [2010] can solve the wiring topology optimally. Thus, this study compares the proposed algorithm with that of Jiang et al. [2010]. Although the proposed algorithm derives the superior initial solution with a slightly higher complexity than that

Table IV. Comparison of the Proposed Method and that of Jiang et al. [2010]

Bench- mark	m	[Jiang et al. 2010]		Ours		Improvement of Runtime	
		#Iteration	Runtime (A)	#Iteration	Runtime (B)	A	B
INP1	7	0	<0.001	0	<0.001	N/A	
INP2	16	1	<0.001	0	<0.001		
INP3	10	0	<0.001	0	<0.001		
INP4	16	0	<0.001	2	<0.001		
IND1	10	0	<0.001	1	<0.001		
IND2	10	3	<0.001	2	<0.001		
IND3	10	2	<0.001	1	<0.001		
IND4	25	17	0.001	17	<0.001		
IND5	33	37	0.003	15	<0.001		
RT01	75	78	0.011	47	0.001	11.00	1
RT02	180	415	0.284	105	0.005	56.80	1
RT03	303	949	1.689	229	0.026	64.96	1
RT04	475	1197	5.720	628	0.150	38.13	1
RT05	850	3487	57.665	924	0.691	83.45	1

m: # of terminals Runtime: seconds

of Jiang et al. [2010], the algorithm can reduce the iterations significantly in the following step for finding the optimal solution. The experimental results show that the proposed algorithm is more efficient than that of Jiang et al. [2010], and obtains the same optimal solution.

6. EXPERIMENTAL RESULTS

The proposed algorithm was implemented in the C++ programming language. For a fair comparison, this study compares the results in two different platforms. Tables III, IV, and V are compared on a PC with an Intel®Core™ i7 CPU 920 of 2.67GHz frequency and 4GB memory under Windows 7 Enterprise 32 bit. Table VI runs on a 2.0 GHz Intel Xeon workstation with 16GB memory. The operating system used was Linux, and the compiler was gcc (Version 4.3.3). The algorithms by Lienig and Jerke [2003], Yan and Chen [2008], and Jiang et al. [2010] were implemented on the same platform for a fair comparison.

Table III compares the proposed algorithm of CDEARST with three high-tech algorithms. Fourteen benchmarks are present from the study by Jiang et al. [2010]. The first benchmark (INP1) is shown in Figure 3. Russell's approximation method and the pivot procedure were implemented. Lienig and Jerke [2003] may generate unfeasible solutions, especially for large benchmarks. Because the optimal edges are chosen greedily in the triangular mesh graph and the terminals that have been considered are deleted, isolated terminals may be produced. The method used by Yan and Chen [2008] does not consider the competition of all sources, and hence the solution quality is limited. Jiang et al. [2010] proposed a method to obtain optimal solutions. However, they adopted a greedy heuristic to obtain the initial solution. Therefore, it may consume a large number of iterations in the step for finding the optimal solution.

Table V. Results of the proposed Obstacle-Avoiding Routing Algorithm

Bench- mark	m/k	Ours with Obstacle-Avoiding			
		Wire-Area	OAWLC Runtime	#Pivot	Trans. Runtime
IND1	10/32	3358	0.005	1	0.000
IND2	10/43	58700	0.008	2	0.000
IND3	10/50	5179	0.000	0	0.000
IND4	25/79	4441	0.032	14	0.000
IND5	33/71	8878	0.038	8	0.000
RT01	10/500	8493	0.851	1	0.000
RT02	50/500	257804	1.239	23	0.000
RT03	100/500	57073	2.551	74	0.002
RT04	100/1000	63608	6.958	67	0.001
RT05	200/2000	459359	42.444	219	0.012
RC01	10/10	83900	0.001	0	0.000
RC02	30/10	253210	0.004	6	0.000
RC03	50/10	407600	0.009	25	0.000
RC04	70/10	352000	0.019	51	0.001
RC05	100/10	617710	0.056	72	0.001
RC06	100/500	486216	2.281	50	0.001
RC07	200/800	796156	6.938	246	0.013
RC08	200/800	710029	11.423	207	0.011
RC09	200/1000	760628	14.834	168	0.01
RC10	500/100	2114680	15.069	756	0.193
RC11	1000/100	1568429	80.104	1374	1.338
RC12	1000/10000	7357445	729.102	1696	1.612

m: # of terminals k: # of obstacles Runtime: seconds

OAWLC: Obstacle-Avoiding Wirelength Calculation

Trans.: Transportation

Table IV compares the proposed algorithm of CDEARST with that of Jiang et al. [2010]. The number of iterations (pivots) is usually proportional to the number of terminals (sources and sinks). Russell's algorithm is the method for finding the initial solution that approximates to the optimal solution. The algorithm should obtain a more accurate initial solution that is close to the optimal solution, while fewer terminals are processed. In other words, in certain circuits with fewer terminals, the optimal solution can be obtained when only adapting Russell's method. Furthermore, refining the initial solution that is iteratively obtained via Russell's method is unnecessary to obtain an optimal solution, which is the reason why 0 pivots exist in certain circuits, as shown in Table IV and Table V. Hence, in small benchmarks, Russell's method can obtain the optimal solution without any pivots. For the large benchmarks, the method provides good-quality BF solutions before the solution is pivoted into an optimal solution. The results show that Russell's method obtains superior initial solutions to reduce the iteration (pivot) times. Furthermore, all the procedures were calculated in a two-dimensional array, which is efficient and effective. The results show that the

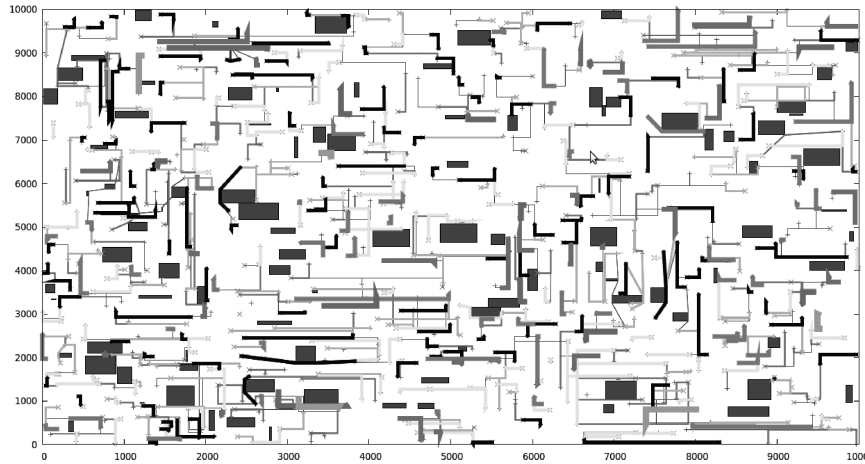


Fig. 19. RC11 output. Obstacles are shown in blue. Wire area = 1568429; runtime = 80.104 sec; number of pivot = 1374; total transportation runtime = 1.338 sec.

proposed algorithm can also obtain optimal solution, although with fewer than 11 CPU times.

Table V shows the obstacle-avoiding routing results. The 23 benchmarks with $m + n$ terminals that range from 7 to 1000 were used. Five industrial benchmark (IND1 to IND5) from Synopsys; 12 benchmarks (RC1 to RC12); and 5 random test cases (RT01 to RT05) were present [Lin et al. 2008]. Since these benchmarks do not have current information, this study randomly assigned either a positive or negative current value to each terminal. The total sum of the current values must follow Kirchhoff's current conservation law. Even under the RMS model, when the total current sum is not equal to zero, the proposed algorithm is still effective. In both situations, the proposed algorithm guarantees an optimal solution.

Figure 19 shows the large benchmark result (RC11), which contains 1000 terminals and 100 obstacles. The total wire area is 1568429.

Table VI shows the P/G routing results in analog ICs. Five benchmarks with randomly distributed modules are present. The power pads and ground pads are situated on the boundary of the chip. The tolerant voltage drop and the required current of each module are fixed, and two layers are used for routing. In practical analog IC designs, area routing is often adopted, which signifies that each routing layer can assign horizontal routing and vertical routing wires. The proposed router adopts area routing and renders the routing result in a single layer as much as possible. Maximum current density quantifies the result after addressing EM. The lower current density identifies the better prevention of EM. Without addressing the EM, due to the high current density, it leads to the influence of *MTTF*, and creates the permanent failure of the chip by EM. The initial wire area after using the transportation approach is shown in the row labelled, CDEARST. The final total wire area using the proposed router is shown in the row labeled, Routing. The increased wire area is caused by net rerouting and IR-drop reduction. Net rerouting prevents power nets and ground nets from overlapping; IR-drop reduction maintains sufficient voltage for each module. The ratio of free routing space is used to estimate the routing congestion. The wire-bundling procedure obtains additional free routing space, and is conducive to detail routing and IR-drop reduction. Finally, Eq. (4) is used to calculate the total voltage drop. Assigning the A_{wire} in direct proportion to I equalizes the initial voltage drop to the total wire length (L_{wire}). After detailed routing, the voltage drop of each module is estimated before ex-

Table VI. Results of the Proposed Analog P/G Routing Algorithm

Benchmark		RPG1	RPG2	RPG3	RPG4	RPG5
#Module		5	10	10	10	10
#Power pad		5	4	13	5	10
#Ground pad		5	5	15	5	10
#Max wire width		10	15	50	75	100
Maximum Current Density	Without EM	18	15	583	801	613
	EM	1.8	1	11.66	10.68	6.136
Wire-Area	CDEARST	2715	1447	487385	802150	513790
	Routing	5837	2193	916283	1042795	1238612
Free Routing Space	Initial	35.6%	28.0%	24.3%	35.7%	13.4%
	Wire Bundling Procedure	38.3%	31.9%	27.6%	40.1%	21.7%
Voltage Drop	Initial	531 ρ	360 ρ	72356 ρ	130349 ρ	205516 ρ
	Wire Width Expanding	507.3 ρ	323.7 ρ	69950.7 ρ	125078.6 ρ	193498.4 ρ

ρ : electrical resistivity of the material

CDEARST: current-driven electromigration-aware rectilinear Steiner tree

panding the wire width for unfeasible module routes. Figure 7(c) shows the routing results of the benchmark RPG2.

7. CONCLUSION

This study proposed an automatic analog P/G router that manages the electromigration and IR-drop. By using the transportation approach, this article solved the current-driven electromigration-aware rectilinear Steiner tree (CDEARST) problem efficiently and optimally. To handle IR-drop, this study first bundled the wire to free more routable areas and alleviate congestion. Thereafter, a wire width extension method was adopted to further reduce IR-drop. The experimental results show the effectiveness and efficiency of the proposed algorithm.

REFERENCES

- BLACK, J. 1969. Electromigration - A brief survey and some recent results. *IEEE Trans. Electron Devices* 16, 4, 338–347.
- CORMEN, T. H., LEISERSON, C. E., RIVEST, R. L., AND STEIN, C. 2009. *Introduction to Algorithms* 3rd Ed., MIT Press.
- DUTTA, R. AND MAREK-SADOWSKA, M. 1989. Automatic sizing of power/ground (p/g) networks in VLSI. In *Proceedings of the IEEE/ACM Design Automation Conference (DAC'89)*. ACM, New York, 783–786.
- HO, J.-M., VIJAYAN, G., AND WONG, C. K. 1990. New algorithms for the rectilinear steiner tree problem. *IEEE Trans. Comput.-Aid. Des. Integrat. Circuits Syst.* 9, 2, 185–193.
- JIANG, I. H.-R., CHANG, H.-Y., AND CHANG, C.-L. 2010. Optimal wiring topology for electromigration-avoidance considering multiple layers and obstacles. In *Proceedings of the International Symposium on Physical Design (ISPD'10)*. ACM, New York, 177–184.
- LIENIG, J. 2006. Introduction to electromigration-aware physical design. In *Proceedings of the International Symposium on Physical Design (ISPD'06)*. ACM, New York, 39–46.
- LIENIG, J. AND JERKE, G. 2003. Current-driven wire planning for electromigration avoidance in analog circuits. In *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC'03)*. ACM, New York, 783–788.

- LIN, C.-W., CHEN, S.-Y., LI, C.-F., CHANG, Y.-W., AND YANG, C.-L. 2008. Obstacle-avoiding rectilinear Steiner tree construction based on spanning graphs. *IEEE Trans. Comput.-Aid. Des. Integrat. Circuits Syst.* 27, 4, 643–653.
- MCMURCHIE, L. AND EBELING, C. 1995. Pathfinder: A negotiation-based performance-driven router for FPGAs. In *Proceedings of the ACM Symposium on Field-Programmable Gate Arrays (FPGA'95)*. ACM, New York, 111–117.
- PAN, M. AND CHU, C. 2006. A step to integrate global routing into placement. In *Proceedings of the IEEE International Conference on Computer-Aided Design (ICCAD'06)*. IEEE, Los Alamitos, CA, 464–471.
- RUSSELL, E. J. 1969. Extension of Dantzig's algorithm to finding an initial near-optimal basis for the transportation problem. *Oper. Res.* 17, 1, 187–191.
- SILVA, J. M. S., PHILLIPS, J. R., AND SILVEIRA, L. M. 2008. Efficient representation and analysis of power grids. In *Proceedings of the Conference on Design, Automation, and Test in Europe (DATE'08)*. ACM, New York, 420–425.
- TODRI, A., MAREK-SADOWSKA, M., AND CHANG, S.-C. 2007. Analysis and optimization of power-gated ICS with multiple power gating configurations. In *Proceedings of the IEEE International Conference on Computer-Aided Design (ICCAD'07)*. IEEE, Los Alamitos, CA, 783–790.
- WINSTON, W. L. 2004. *Operations Research Applications and Algorithms* 4th Ed., Thomson Brooks/Cole.
- YAN, J.-T. AND CHEN, Z.-W. 2008. Electromigration-aware rectilinear Steiner tree construction for analog circuits. In *Proceedings of IEEE Asia Pacific Conference on Circuits and Systems (APCCAS'08)*. IEEE, Los Alamitos, CA, 1692–1695.

Received January 2011; revised July 2011; accepted August 2011