Post Global Routing RLC Crosstalk Budgeting *

Jinjun Xiong Jun Chen James Ma Lei He
ECE Department, University of Wisconsin
Madison, WI 53706

Abstract

Existing layout optimization methods often assume a set of interconnects with given RLC crosstalk bounds in a routing region. RLC crosstalk bound partitioning is critical for effectively applying these methods at the full-chip level. In this paper, we develop an optimal crosstalk budgeting scheme based on linear programming (LP) formulation, and apply it to shield insertion and net ordering at the full-chip level. Experiment results show that compared to the best alternative approach, the LP based method reduces the total routing area by up to 7.61% and also uses less runtime. To the best of our knowledge, this is the first in-depth work that studies the RLC crosstalk budgeting problem.

1. INTRODUCTION

As VLSI technology advances, crosstalk has gained a growing importance. Interconnect optimization techniques such as net ordering [1] and shielding consider net segments within individual routing regions, and assume that appropriate noise bounds are given for these net segments. Therefore, the noise budgeting problem should be solved in order to distribute overall crosstalk bounds specified for the whole nets into crosstalk bounds for net segments in different routing regions. Such a crosstalk budgeting problem has been studied for net ordering and shielding under capacitive crosstalk constraints in [2]. The solution is based on iterations between the following two procedures: crosstalk risk estimation and crosstalk bound partitioning. Crosstalk risk estimation computes the number of shields needed to meet the partitioned crosstalk bounds for a given region with consideration of net ordering. It is formulated and solved approximately as an NP-hard graph optimization problem. Crosstalk bound partitioning is a two-pass integer linear programming (ILP) optimization, minimizing the number of shields for the current global routing solution. Rip-up and re-route can be carried out to adjust the global routing for further reduction of shields.

It is assumed that coupling exists only between adjacent wires in [2]. However, this assumption no longer holds for inductive coupling, which exists between both adjacent and non-adjacent wires and becomes increasingly significant in high performance designs. Recent interconnect optimization for RLC crosstalk reduction includes simultaneous shield insertion and net ordering (SINO) [3], twisted bundle layout structure [4], and differential signaling [5]. These methods assume that crosstalk bounds are given for net segments in individual routing regions. Because the noise budgeting in [2] is no longer applicable to RLC crosstalk constraints, new RLC noise budgeting techniques are needed to leverage the interconnect optimization techniques in [3, 4, 5].

In this paper, we study the *i*SINO problem that considers RLC crosstalk budgeting for the full-chip level SINO solution. The rest of the paper is organized as follows: Section 2 presents the background knowledge. Section 3 formulates the *i*SINO problem and presents a three-phase *i*SINO algorithm framework. Different RLC crosstalk budgeting schemes as well as their mathematical formulations are discussed for this framework. Section 4 reports experiment results using MCNC benchmarks. Section 5 concludes the paper with discussions of future work. The reader is strongly recommended to refer to the technical report [6] for details of this work.

*This research is partially supported by NSF CAREER Award CCR-0093273 and a grant from Intel. Lei He is now a professor at UCLA Electrical Engineering. Address comments to xiongl@cae.wisc.edu and lhe@ee.ucla.edu.

2. BACKGROUND

2.1 Preliminaries

For simplicity of presentation, we assume two routing layers, one for horizontal wires and the other for vertical wires. The routing layers are divided by pre-routed power/ground (P/G) networks into routing *regions*. A route for a net contains a sequence of net *segments* in different routing regions. A *shield* is a wire directly connected (without through devices) to P/G networks. We also assume that all signal and shield wires (except for P/G wires which are often wider) have the same width and spacing. We summarize the notations frequently used in this paper in Table 1.

According to [3], two signal nets N_i and N_j are *logically sensitive* (or in short, *sensitive*) to each other if, through logic synthesis or timing analysis [7], a switching event on N_j causes N_i to malfunction (due to extraordinary crosstalk or delay variation). In this case we call N_j an aggressor for N_i and N_i a victim of N_j. The *logic sensitivity rate* (or in short, *sensitivity rate*) of N_i is defined as the ratio of the number of aggressors for N_i to the total number of signal nets. During the global routing stage, however, two logically sensitive net segments are considered to be *physically sensitive* to each other only if they are routed within the same region, because we assume no crosstalk (coupling) between different regions separated by P/G wires. Therefore, the *physical sensitivity rate* is defined as the ratio of the number of aggressors for the net segment N_{it} of N_i to the total number of net segments in the region R_t and is expected to be lower than the overall

R_t	routing regions in a chip
$\mathcal{R}$	set of all routing regions
$\lvert\mathcal{R}\rvert$	total number of routing regions
CLM	set of routing regions in a column for horizontal wires
$\mathcal{CLM}$	set of all CLM's
ROW	set of routing regions in a row for vertical wires
$\mathcal{ROW}$	set of all ROW's
l_t	length of region R_t
C_t	total number of tracks in region R_t
O_t	total number of tracks occupied by obstacles in region R_t
G_t	set of net segments in region R_t
$\lvert G_t\rvert$	total number of net segments in region R_t
$\mathcal{G}$	set of net segments in all regions
$\lvert\mathcal{G}\rvert$	total number of net segments in all regions
S_t	set of shield segments in region R_t
$\lvert S_t\rvert$	total number of shield segments in region R_t
N_i	signal net
$\mathcal{N}$	set of all signal nets
$\lvert\mathcal{N}\rvert$	total number of signal nets
p_{i0}	source pin of net N_i
p_{ij}	j^{th} sink of net N_i
P_i	set of all sinks for net N_i
$\lvert P_i\rvert$	total number of sinks for net N_i
N_{it}	net segment of net N_i in region R_t
n_i	total number of net segments in the route for net N_i
r_{it}	physical sensitivity rate of N_{it} in region R_t
H_{ij}	set of regions containing the route for sink p_{ij} of net N_i
K_{it}	total inductive coupling for net segment N_{it}
$\overline{K_{it}}$	bound of K_{it}
LSK_{ij}	LSK value of sink p_{ij} of net N_i
$\overline{LSK_{ij}}$	bound of LSK at sink p_{ij} of net N_i

Table 1: Notations that are frequently used in this paper. They will be explained in detail when they are first used.

(logic) sensitivity rate.

2.2 LSK Model for RLC Crosstalk

For any two segments N_{it} and N_{jt} in Region R_t, the inductive coupling coefficient between them is

$$k_{it,jt} = \frac{L_{it,jt}}{\sqrt{L_{it} \cdot L_{jt}}} \quad (1)$$

where $L_{it,jt}$ is the mutual inductance between N_{it} and N_{jt}, and L_{it} and L_{jt} are the self inductance for N_{it} and N_{jt}, respectively. A formula-based K_{eff} model has been developed in [3] to calculate the coupling coefficients $k_{it,jt}$. Furthermore, the total amount of inductive coupling induced on N_{it} can be represented by the sum of the inductive coupling coefficients

$$K_{it} = \sum_{j \neq i} k_{it,jt} \quad (2)$$

for all net segments N_{jt}'s that are sensitive to N_{it}.

To consider the effect of interconnect length and the general case where the total coupling is not uniform in all routing regions, a length-scaled K_{eff} (LSK) model was proposed in [8], where the LSK value is defined as

$$LSK = \sum_t l_t \cdot K_{it} \quad (3)$$

where l_t is the length of R_t and K_{it} is the total coupling for N_{it}. A similar length scaled approach was also adopted in the early work by [9, 2] to model the capacitive crosstalk, which was computed by the product of coupled length and the coupling capacitance that exists only between adjacent wires.

2.3 SINO Problem and Shield Estimation

The SINO problem [3] finds a min-area shield insertion and net ordering (SINO) solution within a given routing region such that all net segments in the region are capacitive crosstalk free (i.e., no sensitive net segments are adjacent to each other) and have inductive crosstalk less than the given bounds ($\overline{K_{it}}$'s). Furthermore, a formula to estimate the number of shields in a min-area SINO solution within region R_t has been developed in [10]. However, in this paper, we use an estimation formula different from that in [10]. The new formula (but not the one in [10]) guarantees that the coefficients of any $\overline{K_{it}}$ are always negative as required by the definitions of these coefficients.

The new formula has the following format:

$$\lvert S_t\rvert = a_1 \cdot \sum_{N_{it} \in G_t} \overline{K_{it}} \cdot r_{it} + a_2 \cdot \sum_{N_{it} \in G_t} r_{it} \quad (4)$$

where notations from Table 1 are used, and a_1 and a_2 are constant coefficients. Note that the number of net segments $\lvert G_t\rvert$ and physical sensitivity rates r_{it}'s are fixed in a region R_t for the given global routing solution, hence (4) can be further simplified as a linear combination of the given coupling bounds:

$$\lvert S_t\rvert = \sum_{N_{it} \in G_t} \alpha_{it} \cdot \overline{K_{it}} + \beta_t \quad (5)$$

In order to obtain the coefficients in (4), for a given routing region, we generate 10,000 routing solutions with different combinations of the number of net segments, sensitivity rate r_{it} (ranging from 20% to 80%), and $\overline{K_{it}}$ value. After running SINO for all cases, the number of shield in each SINO solution is collected. The data are then evenly divided into five groups. A multi-variable curve-fitting process that minimizes the least square error is employed to obtain the coefficients under different groups. As shown in Table 2 the coefficients obtained from different groups are fairly consistent, and the values of the coefficient of determination (R^2) show a well acceptable goodness-of-fit. This proves that (4) is a good template to estimate the number of shields needed for a min-area SINO solution. Further experiments show that even for different SINO implementations, the template is still valid. Detailed experiments and results can be found in the technical report [6]. The coefficients finally used in this paper are obtained by putting data from all groups together and re-doing the curve-fitting process. The final values are $a_1 = -0.10491$, $a_2 = 0.49392$ with $R^2 = 0.87$. However, we should point out that the region based SINO solution always assumes that the crosstalk bound for each net segment

	I	II	III	IV	V
a_1	-0.10956	-0.10408	-0.09781	-0.10605	-0.10795
a_2	0.50339	0.47515	0.47025	0.49420	0.51500
R^2	0.8186	0.8071	0.8419	0.8711	0.8972

Table 2: Coefficients for shield estimation equation.

is given a priori, which is not true in real VLSI design as the crosstalk bound is given at each sink instead of at each net segment. Therefore, we must solve a crosstalk budgeting problem to obtain information in this form. Our algorithm is presented in the next section.

3. PROBLEM FORMULATION AND ALGORITHM

3.1 *i*SINO Problem

FORMULATION 1. **(*i*SINO problem)** *Given a global routing solution and the RLC crosstalk bound for each sink, the iSINO problem determines the coupling bound for each net segment and finds a min-area SINO solution within each region such that the RLC crosstalk bound is satisfied at each sink and the total routing area is minimized.*

***i*SINO algorithm overview**
Given global routing solution and RLC crosstalk bound for each sink
Phase I: Crosstalk budgeting at the global routing level.
Phase II: SINO within each region.
Phase III: Local refinement.

Figure 1: Overview of the three-phase algorithm for *i*SINO problem.

The *i*SINO problem has a high complexity, as its subproblem to find a SINO solution within a region is already NP-hard [3]. Therefore, we develop the following three-phase heuristic algorithm (see Figure 1): In Phase I we find the crosstalk bound for each net segment in all regions. In Phase II we perform SINO in each region by using the algorithm developed in [3]. In Phase III, we carry out a local refinement procedure to completely eliminate the remaining (but very limited) RLC crosstalk violations and further reduce the routing congestion. The algorithm and implementation of Phase III can be found in the technical report [6]. Below we discuss Phase I in details.

3.2 RLC Crosstalk Budgeting

We propose two crosstalk budgeting schemes. One is *uniform budgeting* scheme; the other is an *optimal budgeting* scheme based on *linear programming* (LP). Note that the input of a budgeting scheme is the crosstalk bound (in LSK values) for each sink and the output is the coupling bound for each net segment.

3.2.1 Uniform Crosstalk Budgeting

The simplest scheme to perform crosstalk budgeting is to distribute the crosstalk bound uniformly along the route and is denoted as CB/UD. Let $\overline{LSK_{ij}}$ be the crosstalk bound at sink p_{ij} for net N_i, *len* be the total routing length from the source p_{i0} to sink p_{ij}, then each routing region R_t on the path is assigned a *uniform* crosstalk budget:

$$\overline{K_{it}} = \frac{\overline{LSK_{ij}}}{len} \quad (6)$$

If segment N_{it} is shared by multiple paths starting from the same source to different sinks, we use the minimum value computed for these paths according to (6). One can easily see that this budgeting scheme is not able to consider the non-uniform routing congestion distribution among different regions.

3.2.2 Crosstalk Budgeting Constraints

Before presenting the LP based optimal budgeting schemes, we discuss three common design constraints that must be satisfied: (i) the LSK value should be less than the given crosstalk bound $\overline{LSK}$ at each sink. (ii) the number of estimated shield should be positive. And (iii) for net segment N_{it} in region R_t, the budgeted bound $\overline{K_{it}}$ should not exceed a maximum value $\overline{K_{it}^{max}}$, which can be obtained as follows: assuming there is no shield in the region, and all N_{it}'s aggressors are placed as close to it as possible, the K_{it} value in this case is $\overline{K_{it}^{max}}$. The three constraints can be expressed formally as follows:

$$\sum_{R_t \in H_{ij}} l_t \cdot \overline{K_{it}} \leq \overline{LSK_{ij}} \qquad \forall p_{ij} \in P_i \text{ and } \forall N_i \in \mathcal{N} \quad (7)$$

$$\sum_{N_{it} \in G_t} \alpha_{it} \cdot \overline{K_{it}} + \beta_t \geq 0 \qquad \forall R_t \in \mathcal{R} \quad (8)$$

$$\overline{K_{it}} \leq \overline{K_{it}^{max}} \qquad \forall N_{it} \in R_t \text{ and } \forall R_t \in \mathcal{R} \quad (9)$$

Because all the above constraints should be considered for any LP-based budgeting scheme to be presented, we will not repeat them explicitly later on.

3.2.3 One-Dimension Optimal Budgeting

Without loss of generality, we call the global routing within a row of regions that allow only horizontal wires as one-dimension routing. We also call the number of tracks occupied by net segments, shields and/or obstacles in region R_t as its height h_t, and the maximum height h_{max} among all regions as the height of the routing solution. We then formulate the following CB/1D problem:

FORMULATION 2. **(One-dimension crosstalk budgeting (CB/1D) problem)** *For a given one-dimension routing solution, the CB/1D problem partitions crosstalk bounds among regions such that the maximum height h_{max} is minimized.*

The CB/1D problem can be mathematically stated as:
minimize h_{max} **s.t.**

$$\sum_{N_{it} \in G_t} \alpha_t \cdot \overline{K_{it}} + \beta_t + |G_t| + O_t \leq h_{max} \quad \forall R_t \in \mathcal{R} \quad (10)$$

where the left hand side of new constraint (10) computes the estimated height of region R_t. Further, $\overline{K_{it}}$ is the unknown we need to solve for the CB/1D problem and for the CB/2D-*p* problem to be presented.

3.2.4 Pseudo Two-Dimension Optimal Budgeting

For a two-dimensional global routing consisting of an array of routing regions, let CLM (ROW) be the set of routing regions in a column (row) for horizontal (vertical) wires, and $\mathcal{CLM}$ ($\mathcal{ROW}$) be the set of all CLM's (ROW's). Then, the height h for CLM is defined as the total number of tracks occupied by net segments, shields and obstacles in CLM, and the height h_{max} of the total routing area is defined as the maximum h among all $CLM \in \mathcal{CLM}$. The width w for ROW and w_{max} for the total routing area can be defined similarly. The pseudo two-dimension optimal budgeting (CB/2D-*p*) problem is defined as follows:

FORMULATION 3. **(Pseudo two-dimension crosstalk budgeting (CB/2D-p) problem)** *For a given global routing solution, the CB/2D-p problem partitions crosstalk bounds among all routing regions such that the weighted sum $\gamma \cdot h_{max} + \theta \cdot w_{max}$ is minimized, where γ and θ are two positive constants.*

The CB/2D-p problem can be mathematically stated as:
minimize $\gamma \cdot h_{max} + \theta \cdot w_{max}$ **s.t.**

$$\sum_{R_t \in CLM} \Big(\sum_{N_{it} \in G_t} \alpha_t \cdot \overline{K_{it}} + \beta_t + |G_t| + O_t \Big) \le h_{max} \qquad \forall CLM \in \mathcal{CLM} \quad (11)$$

$$\sum_{R_t \in ROW} \Big(\sum_{N_{it} \in G_t} \alpha_t \cdot \overline{K_{it}} + \beta_t + |G_t| + O_t \Big) \le w_{max} \qquad \forall ROW \in \mathcal{ROW} \quad (12)$$

where the left hand sides of constraints (11) and (12) compute the height and width of an entire column and row, respectively. We approximate the objective of minimizing the total routing area ($h_{max} \cdot w_{max}$) by minimizing the weighted sum of h_{max} and w_{max}. Because h_{max} and w_{max} often have similar values in practice, minimizing their weighted sum provides a good solution for minimizing their product but with a much reduced complexity[1].

3.2.5 Heuristic Constraints

Note that shielding estimation (4) is not "detailed" enough as illustrated by the following example. Let us assume in a given routing region, all net segments have the same sensitivity rate and the sum of $\overline{K_{it}}$ over all net segments is fixed as $\overline{K_t^{sum}}$. In this case, evenly distributing $\overline{K_t^{sum}}$ among all net segments or giving $\overline{K_t^{max}}$ to only one net segment does not make difference in terms of our LP formulation, but may make difference in reality. For example, if a net segment has a high coupling bound and the rest segments have low bounds, the SINO solution may have a large number of shields in order to meet these low coupling bounds. In contrast, the SINO solution under a more balanced coupling bounds for all net segments may have fewer shields.

We have observed the above unbalanced budgeting in our experiments. To avoid this, we propose the following heuristic constraints: (i) According to (4), if all net segments in one region can obtain the same bound $\overline{K_{it}}$, then positive shield constraints (8) would imply

$$\overline{K_{it}} \le -a_2/a_1 \qquad \forall N_{it} \in R_t \quad (13)$$

where a_1 and a_2 are constants given in Section 2.3. Experiments show that the above (13) often provides a tighter upper bound for $\overline{K_{it}}$ than (9) does. And (ii) for a given routing region, we prefer the budgeting scheme that will give a higher budget bound $\overline{K_{it}}$ to a net with higher sensitivity rate r_{it} in order to reduce shields in this region. I.e.,

$$\overline{K_{it}} \le \overline{K_{jt}} \qquad \forall r_{it} \le r_{jt} \quad (14)$$

Theoretically, the above constraint is valid only if we ignore the congestion difference between different routing regions. But it leads to nice results in our experiments with the presence of nonuniform congestion distribution.

[1]Minimizing the product is a quadric programming problem, but minimizing the sum is a linear programming problem.

3.2.6 Main Property

It is easy to verify that the CB/1D and CB/2D-p problems with various constraints are all linear programming (LP) problems. For the rest of the paper, we use CB/LP to represent either CB/1D or CB/2D-p whenever there is no ambiguity. We call the CB/LP without considering constraints (13) and (14) as CB/LP(1), the CB/LP with (13) as CB/LP(2), and the CB/LP with (13) and (14) as CB/LP(3).

4. EXPERIMENT RESULTS

We have implemented the *i*SINO algorithm in C/C++ on UNIX/Linux platforms. A simplex based LP engine, *lp-solver* ([11]), is integrated into the framework to solve the optimization budgeting problems. Below, we present experiments using two-pin bus structures and MCNC benchmark circuits *Primary*1.1 and *Primary*1.2. The two benchmark circuits are placed by DRAGON [12], and routed by our own router using the iterative deletion algorithm.

Test circuit	Number of nets	Number of pins	Regions (row×col)	Obstacle segments
64-bus.1	64	128	1×10	16
64-bus.2	64	128	15×10	32
*Primary*1.1	1266	2303	8×16	460
*Primary*1.2	1266	2303	16×16	643

Table 3: Test circuit characteristics.

In our experiments, we assume that buffers are inserted so that no wires are longer than 1000 μm, and the average logic sensitivity rate over the chip is 50% and 70%, respectively. We also assume that all sinks have the same bound $\overline{LSK}$ = 1000, but our algorithm and implementation can handle non-uniform bounds. We define the gap between the LSK value and the bound $\overline{LSK}$ at a sink as its *LSK slack*. Further, we randomly generate obstacles in each region. We summarize the test circuit characteristics in Table 3, and present experiment results in Tables 4 and 5.

4.1 Validation of Algorithms

According to the maximum/average LSK values among all sinks in columns 9 of Tables 4 and 5, all LSK values meet the given bound $\overline{LSK}$, i.e., our *i*SINO algorithm completely eliminates the crosstalk violations. For each and every budgeting scheme, when the sensitivity rate and obstacles are increased, the routing area and number of shields are increased. We report the area in Phase I based on our shielding estimation. As shown in columns 4 of Tables 4 and 5, all LP schemes achieve smaller area compared to the UD scheme, and the area reduction can be up to 15.49%. All the above observations are the same as expected, and indicate the correctness of our problem formulation and program implementation.

4.2 Comparison between budgeting Schemes

Further, we compare various budgeting schemes based on Phase II results. From Table 4, we observe that among the three LP-based budgeting schemes, LP(3) is always the best one in terms of the routing area reduction. LP(2) can achieve very similar quality results as LP(3), while LP(1) is always the worst. Because of this, we only report results from LP(2) and LP(3) for the MCNC benchmark circuits in Table 5. The budgeting results from MCNC benchmark show that LP(3) is always better than LP(2). Under all cases, UD is worse than LP(3). For bus-structures, up to 7.61% routing area reduction is seen for LP(3) over UD; while for MCNC benchmark, as high as 5.26% reduction is achieved for LP(3) over UD.

For CB/UD, CB/LP(2) and CB/LP(3) budgeting schemes, the estimated routing areas from Phase I are well consistent with the true routing areas obtained from Phase II. An interesting observation is that even though LP-based budgeting schemes may consume far more shields than CB/UD, the routing areas from LP may not be larger. Indeed, the routing area is even smaller for LP(2) and LP(3). Comparing the estimated shield number from Phase I and the real shield number from Phase II, we notice that their values are not as close as that of routing area.

4.3 Impact of local refinement

As shown in Tables 4 and 5, local refinement in Phase III can reduce the number of shields by a large margin without violating LSK bound. For example, as high as 57.4% shield reduction is achieved for $Primary1.2$ circuits under an average logic sensitivity 50% from LP(2). On the other hand, the reduction of routing area in Phase III is effective for UD, but not that effective for LP. Once again, routing areas from LP(2) and LP(3) are smaller than that from UD even after Phase III. Similar routing area is obtained for both LP(2) and LP(3) after Phase III.

4.4 Running time

We report the running time for different RLC crosstalk budgeting schemes as well as the total running time including Phase II and Phase III in Table 6. Only LP(3)'s running time is reported as LP(3) is slowest among all LP schemes. The running time is based on the two MCNC benchmark circuits. As shown in Table 6, even though the LP based budgeting scheme costs more time than the simpler uniform budgeting scheme, the total running time for LP is much less than that of UD. Therefore, we pay little more computation time for a good budgeting scheme initially, but gain the reduction of total routing area and save the total running time at the end.

Sensitivity Rate	Circuit	CB/UD		CB/LP(3)	
		Budget	Total	Budget	Total
50%	P1.1	0.23	3002.26	10.54	2378.53
50%	P1.2	0.40	2709.08	5.75	2638.90
70%	P1.1	0.22	3010.56	31.36	2700.38
70%	P1.2	0.40	2935.31	12.04	2703.23

Table 6: Running time in *seconds* for CB/UD and CB/LP(3) budgeting schemes, as well as the total running time for *i*SINO algorithms.

4.5 Experiment Discussion

Shielding estimation (4) is critical to our LP formulation. It is worthwhile to point out that the estimation is not *exact* in the following senses: (1) The formula based estimation results in a continuous value, but the number of shields is an integer in reality; (2) Even though we can round the estimated shield number to an integer, it may still be different from the number of shields obtained by the detailed SINO algorithm. We argue that the absolutely exact estimation is not needed in our problem context, and our experiments have shown that the approximate estimation achieves good noise budgeting for detailed SINO procedures (in Phases II and III). Theoretically, we can formulate the noise budgeting problem as an ILP problem by rounding up the estimated number of shields to an integer. However, such an ILP problem will be much less efficient compared to our LP problem formulation.

From Table 4, we can see that LP(1) without good heuristic constraints is not necessarily optimal in Phases II and III. As illustrated in section 3.2, the shield estimation equation (4) only reflects the total effects of all net segments in a given region, therefore it can not clearly differentiate each individual net segment. On the other hand, our LP formulation treats each net segment as an individual optimization variable as in the reality. Because of this discrepancy between our estimation and LP formulation, LP(1) may be worse than UD in Phases II and III (even though LP(1) is always better than UD according to the area, which is the LP objective function in Phase I).

To improve LP(1), we can either develop a more sophisticated shield estimation considering each net segment independently, or impose more sensible constraints while maintaining the simpler format of estimation equation. We argue that the latter approach used in this paper is better. Because the first approach may end up with a problem formulation intractable. On the contrary, with a simpler but less accurate equation, we can formulate a very elegant linear programming problem and hence solve it efficiently. The heuristic constraints in (13) and (14) are two of many possible ways to provide such a guidance. We choose them because their simplicity and linearity enable us to solve it efficiently via linear programming method. The experiment results support the effectiveness of these heuristics.

5. CONCLUSIONS AND FUTURE WORK

Existing layout optimization methods often assume a set of interconnects with RLC crosstalk bounds in a routing region. RLC crosstalk bound budgeting is critical for effectively applying these methods at the full-chip level. In this paper, we have developed an optimal crosstalk bound budgeting scheme based on linear programming (LP) formulation, and have applied it to shield insertion and net ordering (SINO) at the full-chip level. Experiments have shown that compared to the best alternative approach, the LP based approach reduces the routing area by up to 7.61% and uses less runtime.

Shields are naturally a part of the power/ground network. Our future work will focus on power/ground design (including shielding) with consideration of both power and signal integrity, as well as routing area/congestion reduction.

6. REFERENCES

[1] T. Gao and C. L. Liu, "Minimum crosstalk channel routing," in *ICCAD*, pp. 692–696, Nov. 1993.

[2] T. Xue, E. S. Kuh, and D. Wang, "Post global routing crosstalk risk estimation and reduction," in *ICCAD*, pp. 302–309, Nov. 1996.

[3] L. He and K. M. Lepak, "Simultaneous shielding insertion and net ordering for capacitive and inductive coupling minimization," in *ISPD*, 2000.

1	2	3	4	5	6	7	8	9
Sensitive Rate	Budget Scheme	Phase I		Phase II		Phase III		$\overline{LSK}$
		Shield	$hmax$	Shield	$hmax$	Shield	$hmax$	1000
64-bus.2 with 16 obstacle segments								
50%	UD	114.1	92.7 (0.00)	73	88 (0.00)	45	85 (0.00)	950.0/553.9
	LP(1)	128.0	80.0 (-13.70)	108	88 (0.00)	72	86 (1.18)	994.4/520.6
	LP(2)	128.0	80.0 (-13.70)	98	83 (-5.68)	62	82 (-3.53)	998.5/624.0
	LP(3)	128.0	80.0 (-13.70)	81	83 (-5.68)	58	82 (-3.53)	980.2/623.8
70%	UD	158.8	97.6 (0.00)	103	92 (0.00)	76	88 (0.00)	981.8/701.5
	LP(1)	158.8	83.4 (-14.55)	220	99 (7.61)	187	94 (6.82)	992.5/321.2
	LP(2)	158.8	83.4 (-14.55)	195	89 (-3.26)	176	87 (-1.14)	977.7/355.3
	LP(3)	158.8	83.4 (-14.55)	123	85 (-7.61)	96	83 (-5.68)	980.1/705.0
64-bus.3 with 32 obstacle segments								
50%	UD	114.1	108.7 (0.00)	72	104 (0.00)	48	101 (0.00)	958.1/556.2
	LP(1)	132.7	96.0 (-11.68)	103	102 (-1.92)	70	100 (-0.99)	981.6/451.8
	LP(2)	132.7	96.0 (-11.68)	82	99 (-4.81)	61	98 (-2.97)	982.7/579.5
	LP(3)	132.7	96.0 (-11.68)	80	99 (-4.81)	58	98 (-2.97)	970.0/547.2
70%	UD	158.8	113.6 (0.00)	104	108 (0.00)	79	105 (0.00)	964.2/695.4
	LP(1)	184.7	96.0 (-15.49)	178	104 (-3.70)	141	100 (-4.76)	995.0/652.6
	LP(2)	184.7	96.0 (-15.49)	120	100 (-7.41)	104	99 (-5.71)	999.9/732.1
	LP(3)	184.7	96.0 (-15.49)	119	100 (-7.41)	99	99 (-5.71)	980.1/751.0

Table 4: Comparison of the total numbers of shields, routing areas in $hmax$, and the maximum/average LSK values under different budgeting schemes after iSINO algorithm for 64-bit bus structures. The values in parenthesis are the percentage of reduction over CB/UD's results.

1	2	3	4	5	6	7	8	9
Sensitive Rate	Budget Scheme	Phase I		Phase II		Phase III		$\overline{LSK}$
		Shield	$hmax + wmax$	Shield	$hmax + wmax$	Shield	$hmax + wmax$	1000
Primary1.1 (8 x 16)								
50%	UD	108.6	230.4 + 147.5 (0.00)	168	232 + 151 (0.00)	98	220 + 149 (0.00)	995.6/283.4
	LP(2)	423.6	226.5 + 145.5 (-1.56)	283	228 + 145 (-2.61)	157	218 + 143 (-2.17)	996.2/262.1
	LP(3)	422.1	226.9 + 145.7 (-1.40)	274	222 + 145 (-4.18)	156	218 + 143 (-2.17)	998.5/259.8
70%	UD	159.0	236.8 + 149.6 (0.00)	244	237 + 154 (0.00)	161	228 + 150 (0.00)	999.4/311.0
	LP(2)	509.2	225.8 + 143.8 (-4.35)	442	234 + 150 (-1.79)	315	225 + 148 (-1.32)	996.1/289.2
	LP(3)	548.7	231.9 + 146.8 (-1.99)	428	229 + 149 (-3.32)	304	222 + 148 (-2.12)	998.9/308.6
Primary1.2 (16 x 16)								
50%	UD	261.0	193.6 + 194.8 (0.00)	258	194 + 194 (0.00)	129	188 + 192 (0.00)	995.5/269.9
	LP(2)	682.4	190.7 + 194.8 (-0.75)	392	182 + 192 (-3.61)	167	182 + 185 (-3.42)	997.0/268.9
	LP(3)	688.9	190.7 + 194.8 (-0.75)	378	182 + 191 (-3.87)	170	182 + 185 (-3.42)	997.5/266.5
70%	UD	368.5	199.2 + 199.1 (0.00)	395	200 + 199 (0.00)	228	191 + 193 (0.00)	998.4/310.8
	LP(2)	878.7	190.1 + 193.2 (-3.77)	709	187 + 198 (-3.51)	397	183 + 185 (-4.17)	997.3/270.8
	LP(3)	879.0	190.6 + 193.3 (-3.62)	692	182 + 196 (-5.26)	406	183 + 187 (-3.65)	999.0/272.5

Table 5: Comparison of the total numbers of shields, routing areas in ($hmax + wmax$), and the maximum/average LSK values under different budgeting schemes after iSINO algorithm for MCNC benchmark circuits. The values in parenthesis are the percentage of reduction over CB/UD's results.

[4] G. Zhong, H. Wang, C.-K. Koh, and K. Roy, "A twisted bundle layout structure for minimizing inductive coupling noise," in *ICCAD*, 2000.

[5] Y. Massoud, J. Kawa, D. MacMillen, and J. White, "Modeling and analysis of diferential signaling for minimizing inductive cross-talk," in *DAC*, 2001.

[6] J. Xiong, J. Chen, and L. He, "Post global routing optimization with RLC crosstalk constraints," in *University of Wisconsin, Technical Report, ECE-02-07*, 2002.

[7] P. Chen, Y. Kukimoto, C.-C. Teng, and K. Keutzer, "On convergence of switching windows computation in presence of crosstalk noise," in *ISPD*, 2002.

[8] J. D. Ma and L. He, "Towards global routing with RLC crosstalk constraints," in *DAC*, 2002.

[9] H. Zhou and D. F. Wong, "Global routing with crosstalk constraints," *TCAD*, November 1999.

[10] J. D. Ma, A. Parihar, and L. He, "Pre-routing estimation of shielding for RLC signal integrity," in *ICCD*, 2001.

[11] M. Berkelaar, "lp-solver 3.2: a public domain (MI)LP solver," in *ftp://ftp.ics.ele.tue.nl/pub/lp_solve/*.

[12] M. Wang, X. Yang, and M. Sarrafzadeh, "DRAGON2000: Standard-cell placement tool for large industry circuits," in *ICCAD*, 2000.