Gate-Size Optimization Under Timing Constraints for Coupling-Noise Reduction

Debjit Sinha and Hai Zhou, *Senior Member, IEEE*

***Abstract*—This paper presents a gate-sizing algorithm for coupling-noise reduction, which optimizes the area or power consumption (represented as a weighted sum of gate sizes) of a circuit while ensuring that its timing constraints are met. A problem for gate-size optimization under coupling-noise and timing constraints is formulated, and is broken down into two subproblems of gate-size optimization under noise and timing constraints, respectively. The subproblem of gate-size optimization under noise constraints is solved as a fixpoint computation problem on a complete lattice. The proposed algorithm to solve this problem is guaranteed to yield the optimal solution, provided it exists. The subproblem for circuit optimization under timing constraints is considered as a geometrical programming problem. The solutions to the two problems are finally combined to solve the original problem in a Lagrangian relaxation (LR) framework. Experimental results demonstrating the effectiveness of the algorithms are reported for the International Symposium on Circuits and Systems (ISCAS) benchmarks and larger circuits. The obtained results are compared to the approach where successive iterations of gate sizing are performed for timing and for noise reduction independently. This alternative design approach is driven by the algorithms used to solving the mentioned subproblems, respectively.**

***Index Terms*—Coupling noise, fixpoint, gate sizing, Lagrangian relaxation (LR), optimization, signal integrity.**

I. Introduction

With the progress of deep-submicrometer technologies, shrinking geometries have led to a reduction in the self-capacitance of wires. Meanwhile, coupling capacitances have increased as wires have a larger aspect ratio and are brought closer together. For present-day processes, the coupling capacitance of a net can be as high as the sum of its area capacitance and fringing capacitance. Trends indicate that the role of coupling capacitances will be even more dominant in the future as feature sizes shrink [1].

Coupling effects can be classified into two types, namely, functional noise and delay variation. Functional noise refers to a spurious signal that is induced due to coupling on a net being held quiet by its driver. This noise causes a glitch that may propagate to a dynamic node or a latch, changing the circuit state and causing a functional failure [2], [3]. Simultaneous switchings on multiple coupled nets in a circuit affect delays on the nets, and thereby cause delay variations [4] in the circuit. These effects are critical in modern high-performance designs and their significance is manifested with the usage of more aggressive and less noise-immune circuit structures like dynamic logic. This paper focuses on functional noise, but the proposed ideas may be extended to handle delay variations as well. We use the term noise and coupling-noise interchangeably to denote functional noise in the rest of this paper.

Approaches to noise reduction include routing and wire perturbations [5], [6], buffer insertion [7], and driver sizing. In the postrouting stage, coupling noise on some nets may be critical and need to be fixed. In this scenario, gate sizing is an attractive approach to noise reduction since it may not require rerouting, unlike the other mentioned approaches. Flexibility through scalable libraries and existing fill space aid incremental gate sizing, without affecting routing.

Noise induced on a net is dependent on the size of its driving gate and the size of the driving gate of each coupled net. This noise can be reduced if the size of the net's driving gate is increased, or if driving gate sizes of the coupled nets are decreased. However, when the driving gate of a net is sized up, it may increase the noise it induces on other nets as an aggressor. On the other hand, when the driving gate size of a net is reduced, noise induced on itself may increase. Since coupling is symmetric on the coupled nets, it is artificial to classify them as aggressors or victims. A net could be an aggressor and a victim at the same time. Though it is plausible to classify a net either as an aggressor or a victim based on the strength of the noise on itself and other coupled nets, with changing driving gate sizes, the role of a net may change. Consequently, we do not classify nets as aggressors and (or) victims in our approach.

Gate sizing for noise reduction has been shown to be effective by researchers. Xiao and Marek-Sadowska [8] propose a transistor-sizing algorithm for noise reduction. A gate-sizing algorithm for noise optimization is proposed by Hashimoto *et al.* in [9], but is limited to sizing driving gates of aggressor nets only. The postroute gate-sizing algorithm for noise reduction proposed by Becer *et al.* [10] does not guarantee an optimal solution. These gate-sizing approaches to noise reduction handle the timing constraints of the circuit as constraints on gate sizes during optimization. This may require timing budgeting before the size-bound generations and can therefore overconstrain the problem.

In this paper, we propose iterative algorithms for circuit optimization under constraints on noise, timing, and gate sizes. A weighted sum of gate sizes is used as the metric for circuit optimization. The formulated optimization problem is broken into subproblems of circuit optimization under noise and timing constraints, respectively. The former is a gate-size-optimization problem under noise constraints without considering timing

Manuscript received October 25, 2004; revised January 19, 2005. This work was supported by the National Science Foundation under Grant CCR-0238484. This paper was recommended by Associate Editor L. Scheffer.

The authors are with the Department of Electrical Engineering and Computer Science, Northwestern University, Evanston, IL 60208 USA (e-mail: haizhou@ece.northwestern.edu).

Digital Object Identifier 10.1109/TCAD.2005.855932

requirements explicitly, while the latter is a gate-size-optimization problem under given timing constraints only.

The subproblem of gate-size optimization under noise constraints is solved as a fixpoint computation [11]–[13] problem on a complete lattice. The proposed algorithm to solve this problem is guaranteed to converge to the optimal solution, provided it exists. If timing constraints of the circuit are translated to gate-size constraints, as in previous approaches, the proposed approach to solving this subproblem yields the optimal solution to the gate-size-optimization problem under noise and timing constraints. The subproblem for circuit optimization under timing constraints is considered as a geometrical programming problem. An optimal algorithm for simultaneous gate and wire-size optimization under timing constraints is presented by Chen *et al.* in [14]. We adopt their idea to solving our latter subproblem. The solutions to the two problems are finally combined to solve the original problem in a Lagrangian relaxation (LR) [15]–[19] framework. Experimental results demonstrating the effectiveness of the algorithms are reported for the International Symposium on Circuits and Systems (ISCAS) benchmarks [20] and larger circuits. We compare our results to the approach where successive iterations of gate sizing are performed for timing and for noise reduction independently. This alternative design approach is driven by the algorithms used to solve the mentioned subproblems, respectively.

The rest of this paper is organized as follows. We present our problem formulation in Section II. The gate-size-optimization subproblem under noise constraints is solved as a fixpoint computation problem in Section III. We briefly consider the gate-size-optimization subproblem under timing constraints in Section IV. An iterative algorithm based on LR to solve the original problem is proposed in Sections V and VI. Experimental results are presented in Section VII. Finally, we draw conclusions in Section VIII.

II. Problem Formulation

A. Motivation

A circuit-optimization problem typically entails the minimization of the area or power consumption of a circuit under given timing and other constraints. To avoid functional failures due to coupling, optimizations are performed to constrain the maximal induced noise on each net of the circuit. However, these optimizations should not violate the given timing constraints. Successive iterations of optimization for noise reduction and timing therefore become necessary, and result in suboptimal solutions. This motivates the development of an optimization approach for noise reduction, which explicitly considers the timing constraints of the circuit and yields better solutions than the approach of independent iterations of optimization for noise reduction and timing.

B. Circuit Modeling

We model a circuit as a directed acyclic graph (DAG). Nodes of the DAG represent gates in the circuit and the edges represent the corresponding nets. A pseudo primary-output node (PO henceforth) is added to represent a single output node having incoming edges from all output nodes of the DAG. Similarly, a pseudo primary-input node (PI henceforth) having outgoing edges to all input nodes of the DAG is added. Nodes of the graph are topologically sorted with PO having index 0 and PI having the largest index for ease of notation.

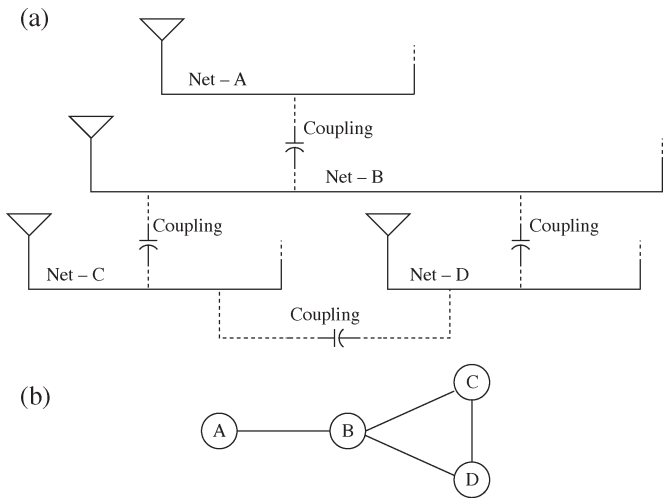

Fig. 1. (a) Circuit with four coupled nets. (b) Corresponding coupling graph.

Gates and nets are modeled as standard switch-level and π-type *RC* circuits, respectively. G is used to denote the set of all gates (nodes) in the circuit (graph). The size of a gate i is denoted by $s(i)$. Gate sizes in the circuit are collectively represented as a gate-size vector S, comprising of all gate sizes in the circuit. Formally, we define

$$S \triangleq [s(i) : i \in G].$$

$\hat{r_i}$ and $\hat{c_i}$ represent unit gate-size output resistance and input capacitance per unit gate size, respectively. As a result, output resistance of gate i is expressed as $r_i = \hat{r_i}/s(i)$, and the capacitance of an input pin of the gate is expressed as $c_i = \hat{c_i}s(i) + f(i)$, where $f(i)$ represents the gate perimeter capacitance, which is independent of the gate size. For simplicity, we assume input capacitances of all input pins of a gate to be the same and ignore intrinsic gate delay. The proposed algorithm can be simply extended to handle them. We use the Elmore delay model for timing estimations in this paper.

An undirected coupling graph containing coupling information is superimposed on the DAG. Nodes in the coupling graph represent the nets of the circuit. Edges in the coupling graph are called coupling edges, and are introduced between nodes corresponding to significantly coupled nets. Fig. 1(a) shows a circuit with nets having significant coupling between them. Fig. 1(b) shows the corresponding coupling graph where the nodes represent the nets of the circuit. The coupling-graph model is used for noise estimation and reduction in our proposed approaches.

C. Problem Definition

Given the size of gates in a circuit, the noise induced on each net can be calculated using a noise model. It is not necessary

that the noise model be an analytical one for our problem formulation. Coupling noise $N(i)$ on the fan-out net of a gate i is formally represented as follows:

$$N(i) \triangleq f_i\left(s(i), s(i_1), \ldots, s(i_k)\right)$$

where $s(i)$ represents the size of gate i, $[i_1, \ldots, i_k]$ represent driving gates of nets that are coupled to the fan-out net of i, and $[s(i_1), s(i_2), \ldots, s(i_k)]$ represent their sizes, respectively.

Given nonnegative weights w_i for each gate i, the objective of the gate-size-optimization problem under noise and timing constraints is to find the minimal weighted sum of gate sizes under the constraints that the maximal arrival time at the PO with respect to the PI is bounded by a given value A_0, noise $N(i)$ on the fan-out net of every gate i is upper bounded by a given value $U(i)$, and the size of each gate i is lower and upper bounded by given values $l(i)$ and $u(i)$, respectively. The noise upper bound $U(i)$ represents the maximum noise the given net can tolerate. Size bounds $l(i)$ and $u(i)$ are given by physical constraints. The timing constraints require that for all paths p from PI to PO, the sum of the delays across components in each path p must be at most A_0. Formally, we want to solve the following problem:

$$\begin{aligned} \min \quad & \sum_{i \in G} w_i s(i) \\ \text{s.t.} \quad & \\ & \sum_{i \in p} D_i \le A_0 && \forall p \in P \\ & N(i) \le U(i) && \forall i \in G \\ & l(i) \le s(i) \le u(i) && \forall i \in G \end{aligned} \tag{1}$$

where D_i is the delay associated with component i and P is the set of all paths p from PI to PO. Since the total number of paths in P can be exponential, the above formulation is impractical for analysis and optimization. We therefore associate a variable a_i to each component that represents the output arrival time for that component. This gives an alternate formulation of the above problem, which is formally expressed as the following:

$$\begin{aligned} \min \quad & \sum_{i \in G} w_i s(i) \\ \text{s.t.} \quad & \\ & a_j \le A_0 && \forall j \in \text{input(PO)} \\ & a_j + D_i \le a_i && \forall i \wedge j \in \text{input}(i) \\ & N(i) \le U(i) && \forall i \in G \\ & l(i) \le s(i) \le u(i) && \forall i \in G \end{aligned} \tag{2}$$

where input(i) gives the set of components that are inputs to component i.

If the weighted sum of gate sizes is considered as an area or power-consumption metric, the formulated problem is interpreted as a circuit-area or power optimization problem under constraints on noise, timing, and gate sizes.

III. Gate-Size Optimization Under Noise Constraints

A. Problem Definition

In this section, we consider a subproblem of the gate-size-optimization problem by temporally ignoring the explicit constraints on the arrival times. We further generalize the objective of the problem to determine a gate-size vector S that minimizes any arbitrary cost function $C(S)$ that is a monotonically nondecreasing function of S. We note that the objective function of our original problem $\sum_{i \in G} w_i s(i)$ is a monotonically nondecreasing function of S for nonnegative weights w_i. The subproblem is formally defined as the following:

$$\begin{aligned} \min \quad & C(S) \\ \text{s.t.} \quad & \\ & N(i) \le U(i) && \forall i \in G \\ & l(i) \le s(i) \le u(i) && \forall i \in G. \end{aligned} \tag{3}$$

B. Gate-Sizing Transformation

Irrespective of the noise model being used, we consider the noise on any net to be monotonically nonincreasing with its driver size and monotonically nondecreasing with the driver size of each of its coupled nets. Thus, for any gate i, the noise function $f_i(s(i), s(i_1), s(i_2), \ldots, s(i_k))$ is considered to be monotonically nonincreasing on $s(i)$ and monotonically nondecreasing on $s(i_j)$ for any $1 \le j \le k$. This assumption is conservative and is satisfied by any reasonable noise function. Similar assumptions based on evidence are used in [8]. The monotonic properties of any noise function are formally expressed as the following:

$$\begin{aligned} & s(i) < s_1(i) \\ & \quad \Rightarrow f_i\left(s(i), s(i_1), \ldots\right) \ge f_i\left(s_1(i), s(i_1), \ldots\right) \end{aligned} \tag{4}$$

$$\begin{aligned} & s(i_j) < s_1(i_j) \\ & \quad \Rightarrow f_i\left(s(i), \ldots, s(i_j), \ldots\right) \le f_i\left(s(i), \ldots, s_1(i_j), \ldots\right). \end{aligned} \tag{5}$$

These properties are used to formulate a gate-sizing transformation. We define a transformation g_i that gives the minimal size (within size constraints) of a gate i such that there are no noise violations (NVs) on its fan-out net. The transformation is a function of $s(i_1), \ldots, s(i_k)$. Formally, we define g_i as the following:

$$\begin{aligned} & g_i\left(s(i_1), s(i_2), \ldots, s(i_k)\right) \triangleq \min\ x \\ & \text{s.t.} \\ & \quad f_i\left(x, s(i_1), s(i_2), \ldots, s(i_k)\right) \le U(i) \\ & \quad l(i) \le x \le u(i). \end{aligned} \tag{6}$$

A system of equations is formed when all gate sizes are combined

$$s(i) = g_i\left(s(i_1), s(i_2), \ldots, s(i_k)\right) \qquad \forall i \in G. \tag{7}$$

We use $\mathcal{G} \triangleq [g_i : i \in G]$ to denote the vector transformation and write (7) in the vector form as the following:

$$S = \mathcal{G}(S). \tag{8}$$

A solution to (8) is a gate-size vector S, which is called a fixpoint of $\mathcal{G}$. The following theorem proves that a solution to (8) is a necessary condition for a solution to the gate-size-optimization subproblem under noise constraints, as defined in (3).

Theorem 1: If there is a solution to (3), there is also a solution to both (3) and (8).

Proof: Let S' be the solution to (3) and $S'' = \mathcal{G}^k(S')$ such that $\mathcal{G}(S'') = S''$, for some $k > 0$. $\mathcal{G}^k(S)$ represents successive applications of transformation $\mathcal{G}$ on the vector S. Transformation $\mathcal{G}^k$ on vector S', comprising gate sizes $s'(i)$, $\forall i \in G$, which satisfies the noise constraints and the size constraints, yields a vector S'' comprising gate sizes $s''(i)$, such that $s''(i) \leq s'(i)$, $\forall i \in G$. This follows from the definition of g_i in (6), which ensures that a gate is never oversized if it satisfies the noise and size constraints. The obtained relation between the gate-size vectors is expressed as $S'' \leq S'$. It is known that the cost function is monotonically nondecreasing with S. We therefore have the following:

$$C(S'') \leq C(S').$$

Since S' is a solution to (3), $C(S')$ is the minimum value of $C(S)$ for all S, which satisfies the noise and size constraints. However, it is known that even S'' satisfies the given constraints and that $C(S'') \leq C(S')$. This implies that

$$C(S'') = C(S').$$

S'' is therefore shown to be a solution to (3), and is also a solution to (8) from definition. ■

It is, thus, established that if a solution to the gate-size-optimization subproblem exists, so does a solution to the same problem, which is also a fixpoint of $\mathcal{G}$. The converse is, however, not true. An arbitrary fixpoint of $\mathcal{G}$ is not necessarily the solution to our subproblem since it may not yield the minimum $C(S)$. A fixpoint of $\mathcal{G}$ can be the optimal solution to our subproblem if and only if it yields the minimal cost function among all other fixpoints. Consequently, we aim to obtain a fixpoint of $\mathcal{G}$, which is also a solution to the subproblem defined in (3).

We next formally define the relation $\leq$ over two gate-size vectors $X \triangleq [x(i) : i \in G]$ and $Y \triangleq [y(i) : i \in G]$.

Definition 1:

$$X \leq Y \triangleq x(i) \leq y(i) \qquad \forall i \in G.$$

Since $\leq$ satisfies the properties of reflexivity, transitivity, and antisymmetry, it also forms a partial order on the family of all gate-size vectors.

Theorem 2: $\mathcal{G}$ is a monotonic transformation, that is, if $S_1 \leq S_2$, then $\mathcal{G}(S_1) \leq \mathcal{G}(S_2)$.

Proof: From Definition 1, $S_1 \leq S_2$ implies the following:

$$s_1(i) \leq s_2(i) \qquad \forall i \in G. \tag{9}$$

We use contradiction to prove our claim. If $\mathcal{G}(S_1) \leq \mathcal{G}(S_2)$ is not true, then there must be at least a gate j, the fan-out net of which couples to fan-out nets of gates $j_1, \ldots, j_k$, such that

$$m > n \tag{10}$$

where $m \triangleq g_j(s_1(j_1), s_1(j_2), \ldots, s_1(j_k))$ and $n \triangleq g_j(s_2(j_1), s_2(j_2), \ldots, s_2(j_k))$. From the monotonic property of f_j in (5), it is known that

$$f_j\left(n, s_1(j_1), \ldots, s_1(j_k)\right) \leq f_j\left(n, s_2(j_1), \ldots\right). \tag{11}$$

From the definitions of n and g_i, we have the following from (11):

$$\begin{aligned} n &= g_j\left(s_2(j_1), s_2(j_2), \ldots, s_2(j_k)\right) \\ &\Rightarrow f_j\left(n, s_2(j_1), s_2(j_2), \ldots\right) \leq U(j) \\ &\Rightarrow f_j\left(n, s_1(j_1), s_1(j_2), \ldots\right) \leq U(j). \end{aligned}$$

However, from the definitions of m and g_i, we obtain the following:

$$\begin{aligned} m &= g_j\left(s_1(j_1), s_1(j_2), \ldots, s_1(j_k)\right) \\ &\Rightarrow m = \min\left\{x : \left(f_j\left(x, s_1(j_1), s_1(j_2), \ldots\right) \leq U(j)\right\}\right. \\ &\Rightarrow m \leq y \qquad \forall y : f_j\left(y, s_1(j_1), s_1(j_2), \ldots, s_1(j_k)\right) \leq U(j) \\ &\Rightarrow m \leq n \end{aligned}$$

which contradicts (10) and proves our claim. ■

According to lattice theory [21], a partially ordered set forms a complete lattice if it has a least upper bound and a greatest lower bound on any subset of its elements. A lattice is constructed of all gate-size vectors that satisfy the size constraints. The bottom element of the lattice is a gate-size vector, comprising gate sizes of which are equal to their individual lower bounds. The top element of the lattice is a virtual gate-size vector, having at least one upper gate-size-bound violation. It is a virtual vector as it is a mapping of all gate-size vectors in the partially ordered set with at least one upper size-bound violation. This makes the family of gate-size vectors with the $\leq$ relation a complete lattice.

Given a subset S_A of elements in a complete lattice L, $\bigvee S_A$ is used to represent the least upper bound of elements in S_A. The existence of a fixpoint in the system is guaranteed by the following theorem due to Knaster and Tarski [21].

Theorem 3: Let L be a complete lattice and $\mathcal{G} : L \rightarrow L$ an order-preserving map. Then

$$\bigvee \{x \in L : x \leq \mathcal{G}(x)\} \in \text{fix}(\mathcal{G})$$

where $\text{fix}(\mathcal{G})$ is the set of fixpoints of $\mathcal{G}$.

The above theorem is not employed to compute a fixpoint since it is not feasible to compute the set $\{x \in L : x \leq \mathcal{G}(x)\}$. Furthermore, it is not known whether such a fixpoint is also the

smallest solution to the gate-sizing problem. People usually use the iterative method (also called successive approximation) to find a fixpoint. In this method, one selects an initial solution X_0 and iteratively computes $X_1 = \mathcal{G}(X_0), X_2 = \mathcal{G}(X_1), \ldots$ in the hope of finding an X_n such that $X_n = \mathcal{G}(X_n)$. However, the procedure cannot be fulfilled by starting from any initial point. Fortunately, the bottom and the top elements are good candidates for that.

Following a tradition in lattice theory, $\perp$ and $\top$ are used to represent the bottom and top elements of the complete lattice, respectively. $\perp$ is defined as the vector of lower bound gate sizes comprising $l(i)$, $\forall i \in G$. Since $\perp \leq \mathcal{G}(\perp)$, based on the monotonic property of $\mathcal{G}$, there is an ascending chain $\perp \leq \mathcal{G}(\perp) \leq \mathcal{G}^2(\perp) \leq \ldots$. If the chain has only finite elements, which is true on any finite solution space, the process eventually reaches a fixpoint. The only property that is used of $\perp$ is $\perp \leq \mathcal{G}(\perp)$. Consequently, any solution X_0 such that $X_0 \leq \mathcal{G}(X_0)$ can be used as an initial solution to reach a fixpoint. Since we aim to find the least fixpoint, we start our iterations with the $\perp$ element as our initial solution. The fixpoint obtained is the optimal solution to our subproblem.

Theorem 4: If $\text{fix}(\mathcal{G}) = \{S_{f_1}, \ldots, S_{f_l}\}$ denotes the set of fixpoints of $\mathcal{G}$, then there exists a least fixpoint $S_{f_L} \in \text{fix}(\mathcal{G})$, defined as $S_{f_L} \triangleq [s_{f_L}(i) = \min(s_{f_1}(i), \ldots s_{f_l}(i)) : i \in G]$, such that $S_{f_L} \leq S_{f_j}, \forall j \in [1, l]$.

Proof: From the definition of S_{f_L}, there must exist a fixpoint S_{f_j} with the same size for an arbitrary gate k, such that $s_{f_L}(k) = s_{f_j}(k)$ and $s_{f_j}(i) \geq s_{f_L}(i), \forall i \in G, i \neq k$. Since S_{f_j} is a fixpoint, the noise constraint on all its gates are satisfied. Thus, given the monotonic properties of the noise function in (5), S_{f_L} must also satisfy the noise constraints. Additionally since it is the lower bound on all the fixpoints, transformation $\mathcal{G}$ on S_{f_L} yields S_{f_L}. It is thus a fixpoint of $\mathcal{G}$. ■

Corollary 4.1: The fixpoint reached starting from the $\perp$ element of the lattice is the least fixpoint and is the optimal solution to the gate-size-optimization problem for noise reduction, provided the fixpoint $\neq \top$, which implies no solution.

Corollary 4.2: The solution to the gate-sizing problem minimizes any cost function $C(S)$, which is monotonically nondecreasing with S. It can therefore minimize a set of cost functions simultaneously. Individual gate sizes can be consider to be a cost function each, since they are monotonically nondecreasing with S. This implies that the solution to the gate-sizing problem yields a gate-size vector S, which consists of the smallest possible sizes of individual gates, such that coupling noise on every net is upper bounded and the circuit satisfies given physical constraints.

Corollary 4.3: If timing constraints are incorporated in the gate-size bounds after timing budgeting, the solution reached using the proposed approach is the optimal gate-size vector that minimizes the weighted sum of gate sizes and satisfies both the noise and timing constraints.

The least fixpoint reached is the lower bound on the gate-size vectors that are fixpoints of $\mathcal{G}$, and thus, for nonnegative weights $w(i)$, yields the minimal weighted sum of gate sizes. It is thus the solution to the gate-sizing problem, as defined in (3). The least fixpoint obtained is the optimal solution and is independent of the given weights $[w(i) : i \in G]$.

C. Scheme of Chaotic Iterations

Before a good iterative order for updates is defined, it is important to establish its theoretical validity. The scheme of chaotic iterations [22] ensures that the process will always converge to the same fixpoint, irrespective of the order being used. Transformation $\mathcal{G}$ is composed of a set of partial transformations $g_1, g_2, \ldots, g_n$. In each step, one or more partial transformations are applied to update gate sizes of some gates, while sizes of all other gates are kept the same. $\mathcal{G}_A$ is used to represent such a partial transformation done in one step, where A represents the points where gate sizes are updated.

Lemma 1:

$$(X \leq \mathcal{G}(X)) \Rightarrow (X \leq \mathcal{G}_A(X) \leq \mathcal{G}(X)) \\ \wedge\ (X \geq \mathcal{G}(X)) \Rightarrow (X \geq \mathcal{G}_A(X) \geq \mathcal{G}(X)). \quad (12)$$

The above lemma states that no matter what evaluation order is used, the generated sequence is monotonic in the same direction and it will not overshoot the fixpoint generated by $\mathcal{G}$. This guarantees reaching the same fixpoint irrespective of the iteration scheme, starting with the same initial point in the lattice.

D. Iterative Sizing Algorithm

We present an algorithm to solve the subproblem of gate-size optimization under noise constraints. The algorithm is transparent to the timing constraints whether it is being incorporated in the size bounds or not. In the former case, the optimal solution attained also guarantees that the timing constraints of the circuit are met, given that the optimal solution does exist. Timing constraints can be incorporated in the size bounds by timing budgeting, as in [8]. In the latter case, gate-size bounds are given by physical constraints only.

Layout extraction is performed on a given circuit and is used to construct the corresponding coupling graph. The size of each gate is initialized to its lower size bound ($s(i) = l(i)$, $\forall i \in G$). Updates ($\mathcal{G}$ transformations) are performed iteratively until all NVs are eliminated, or it is found that gate sizing cannot remove all violations. In the latter case, no optimal solution to gate sizing under noise constraints is declared. For the updates, the noise on the fan-out net of node i is calculated based on a noise model. Gate sizes of nodes whose fan-out nets couple to the fan-out net of i are used to determine the value of $N(i)$. If $N(i)$ exceeds $U(i)$, the size of node i is increased optimally such that $N(i) \leq U(i)$. No update is performed if the calculated noise is less that $U(i)$. If sizing up violates size constraints, no solution for complete noise elimination is declared. Iterations may optionally be continued to remove other NVs. Binary search (for continuous gate sizing) or linear search (for discrete gate sizing) is used to find the optimal gate size during an update. Given n gates in a circuit, and that there exist at most k alternative gates that can be mapped to the same node, the theoretical upper bound on the number of gate-sizing iterations is nk. The pseudocode of the proposed algorithm is shown in Fig. 2.

- Algorithm: Optimal gate-sizing under noise constraints
- **Input:** Layout extraction results
- **Output:** Optimized gate-size vector, if solution exists
- begin
  1) construct coupling-graph from layout extraction
  2) $s(i) = l(i),\ \forall i \in G$
  3) do
  4) for each gate $i \in G : N(i) > U(i)$
  5) $s(i) = g_i(s(i_1), s(i_2), \ldots, s(i_k))$
  6) while ($(\exists i \in G : N(i) > U(i)) \ \wedge \ (\forall i \in G : s(i) \leq u(i))$)
- end

Fig. 2. Optimal algorithm for gate-size optimization under noise constraints.

The order of the iterations may only alter the rate of convergence, but it is certain to reach the optimal solution in any case, provided it exists. We present two possible iterative schemes for node updates next.

1) List traversal: A list of all nodes is maintained and necessary updates are performed sequentially. This is repeated until no updates are found necessary during the entire traversal to yield the optimal solution. However, if a gate exceeds given size constraints, iteration is stopped and no solution for the circuit is declared. Alternately, that node is ignored in future iterations and the algorithm tries to eliminate NVs on other nodes. In either case, no solution to complete NV elimination by gate sizing is declared.
2) Queue traversal: In this scheme, a queue is initially filled with nodes whose fan-outs have NVs. As a node i is popped from the queue, it is sized up to eliminate the NV on its fan-out net. All nodes with fan-out nets having a coupling edge (in the coupling graph) from the fan-out net of node i are pushed in the queue, if they have NVs and are not already in the queue. Iteration stops either when the queue is empty or when a driver size exceeds the given constraint and no solution is concluded. As in list traversal, iterations may be continued optionally to remove remaining violations.

E. Results of Gate-Size Optimization Under Noise Constraints

Experimental results of the proposed algorithm are presented for the ISCAS benchmarks and three larger circuits, namely CKT_1, CKT_2, and CKT_3, respectively. The parameters for all benchmarks are randomly generated with realistic parameters from an 0.18-μm technology library. The 2π model [23] is used as the noise model for all simulations. For the test circuits, the driver resistance $R_{\rm d}$ is from 20–2000 Ω, loading capacitance $C_{\rm l}$ is from 4–50 fF, and the slew is from 10–300 ps. The gate-size bounds used do not incorporate timing constraints.

Table I presents the results obtained using our proposed algorithm. The queue-traversal method is used in the experiments. Optimization using list traversal yields similar results, but the queue-traversal method is faster on the average. For all benchmarks, we present the number of nodes, the number of coupling edges (C Edges), and the number of nets having NV before and after optimization. We present noise reduction as a percentage of nets with NVs fixed with reference to the initial number of nets with violations. The number of initial violations in the table represents the number of nets having violations in the original circuit. It does not reflect the number of nets with violation after gates have been sized down in the initial phase of the algorithm. The increase in the weighted sum of gate sizes for unit weights is found to be less than 2% as compared to the original circuit. In some cases, and under tighter noise bounds, an optimal solution to the subproblem does not necessarily exist. The noise-reduction result in this case reflects the number of violations the algorithm removes when it quits.

TABLE I
NOISE-REDUCTION RESULTS: $U(i) = 0.2V_{\rm dd}$

Circuit	# of Nodes	# of C Edges	# of Initial Violations	# of Final Violations	% Noise Reduction
C432	198	553	2	1	100
C499	245	621	3	0	100
C880	445	1240	3	0	100
C1355	589	1653	12	0	100
C1908	915	2655	10	0	100
C2670	1428	3851	32	1	97
C3540	1721	5086	16	3	81
C5315	2487	7076	17	0	100
C6288	2450	7239	15	0	100
C7552	3721	10823	33	0	100
CKT_1	15K	42017	67	0	100
CKT_2	18K	53114	83	1	99
CKT_3	20K	59389	140	0	100

Runtimes for the benchmarks range from 0.1–10.9 s using the queue traversal and from 0.5–24 s for list traversal. Noise analysis based on the 2π model is quick. More accurate models can be used in our approach, but would increase the runtime significantly. The algorithm and the framework are written in C language. Results presented are obtained by simulations on a Pentium 2.4-GHz Xeon processor server, having 1-GB RAM and running RedHat Linux 8.0.

IV. GATE-SIZE OPTIMIZATION UNDER TIMING CONSTRAINTS

Based on the notations defined earlier, the gate-size-optimization subproblem under timing constraints is defined as the following:

$$\begin{aligned} \min \quad & \sum_{i \in G} w_i s(i) \\ \text{s.t.} \quad & \\ & a_j \leq A_0 && \forall j \in \text{input(PO)} \\ & a_j + D_i \leq a_i && \forall i \wedge j \in \text{input}(i) \\ & l(i) \leq s(i) \leq u(i) && \forall i \in G. \end{aligned} \tag{13}$$

The problem is often considered to be a geometrical programming problem. An extended problem is the gate and wire-sizing problem under timing constraints, which is solved optimally in [14]. We use their approach in solving this subproblem and combine our approach to gate-size optimization under noise constraints in the next section to solve the circuit-optimization problem, under noise and timing constraints defined in (2), in an LR framework.

V. Optimization Under Noise and Timing Constraints by LR

A. LR

LR is a widely acclaimed technique for solving constrained optimization problems. In LR, "troublesome" constraints are "relaxed" and incorporated into the objective function after multiplying them with nonnegative constants called Lagrange multipliers, one multiplier for each constraint. For each fixed vector λ of the Lagrange multiplier introduced, we obtain an easier subproblem called the LR subproblem associated with $\lambda(\mathcal{LRS}/\lambda)$. The problem of finding a λ such that the solution to the subproblem is also the solution to the original constrained problem is called the Lagrangian dual problem or $\mathcal{LDP}$.

Kuhn–Tucker conditions [17] are used to simplify $\mathcal{LRS}/\lambda$ to a simpler subproblem denoted by $\mathcal{LRS}/\mu$. We use the method of subgradient optimization [15], [17] to solve the $\mathcal{LDP}$.

We relax the constraints on the arrival time of the gates since they are difficult to handle. Analytical noise models are complex in form and cannot be expressed as posynomials. We leave the noise and size-bound constraints unrelaxed. This facilitates the approach of gate sizing under noise constraints as a fixpoint computation, which we described earlier. In addition, the objective function of $\mathcal{LRS}/\lambda$ can be expressed as a posynomial.

Following the LR procedure, we introduce a Lagrange multiplier for each constraint on arrival time. For all $j \in$ input(PO), we introduce λ_{j0} for the constraint $a_j \leq A_0$. For all $j \in$ input$(i) \wedge i \neq 0$, we introduce λ_{ji} for the constraint $a_j + D_i \leq a_i$. We assume that the arrival time of PI is 0. We denote the vector of Lagrange multipliers introduced by λ. The vector of arrival times is denoted by $A \triangleq [a_i : i \in G]$.

The LR subproblem associated with λ $(\mathcal{LRS}/\lambda)$ is defined as the following:

$$\begin{aligned} &\min\ L_\lambda(S, A) \\ &\text{s.t.} \\ &\quad N(i) \leq U(i) \qquad \forall i \in G \\ &\quad l(i) \leq s(i) \leq u(i) \qquad \forall i \in G \end{aligned} \tag{14}$$

where $L_\lambda(S, A)$ is given by

$$\sum_{i \in G} w_i s(i) + \sum_{j \in \text{input(PO)}} \lambda_{j0}(a_j - A_0) + \sum_{i \in G} \sum_{j \in \text{Input}(i)} \lambda_{ji}(a_j + D_i - a_i).$$

Kuhn–Tucker [17] conditions imply the following optimality conditions on λ. The proof is presented in [14]

$$\sum_{i \in \text{output}(k)} \lambda_{ki} = \sum_{j \in \text{input}(k)} \lambda_{jk} \qquad \forall k.$$

The above conditions on λ are used to simplify $\mathcal{LRS}/\lambda$. We define $\Omega_\lambda = (\lambda \geq 0 : \lambda)$ to denote the set of λ that satisfies the above conditions. For any $\lambda \in \Omega_\lambda$, $\mathcal{LRS}/\lambda$ is equivalent to solving a new subproblem (proof in [14]) called $\mathcal{LRS}/\mu$, which is defined formally as the following:

$$\begin{aligned} &\min\ L_\mu(S) \\ &\text{s.t.} \\ &\quad N(i) \leq U(i) \qquad \forall i \in G \\ &\quad l(i) \leq s(i) \leq u(i) \qquad \forall i \in G \end{aligned} \tag{15}$$

where vector $\mu \triangleq [\mu_i : i \in G]$, $\mu_i \triangleq \sum_{j \in \text{input}(i)} \lambda_{ji}$, and $L_\mu(S) \triangleq \sum_{i \in G} \mu_i D_i + \sum_{i \in G} w_i s(i)$. We solve $\mathcal{LRS}/\lambda$ by solving $\mathcal{LRS}/\mu$ instead. Vector A is evaluated by topologically considering each variable a_i from the PI and setting it to the smallest possible value that satisfies the timing constraints.

B. Solving $\mathcal{LRS}/\mu$

A local refinement function is derived to solve the Lagrangian subproblem and is applied to gates iteratively until convergence. We define upstream(i) to be the set of resistor indexes on the path(s) from gate i to the nearest upstream gate(s) or input driver(s). Let $R_i = \sum_{j \in \text{upstream}(i)} \mu_j r_j$ and C_i denote the weighted upstream resistance and the downstream capacitance of gate i, respectively.

We define h_i to be a function of gate sizes in a circuit that returns the optimal size of gate i, which locally minimizes L_μ, given other gate sizes. This is similar to the local refinement function derived for timing optimization in [14]. Formally, this is represented as

$$h_i(S) \triangleq (s(i) : \min L_\mu). \tag{16}$$

Based on the Elmore delay model, L_μ can be expressed as a function of $s(i)$, assuming all other gate sizes are fixed

$$L_\mu = A_i(S) \cdot s(i) + \frac{B_i(S)}{s(i)} + E_i(S) \tag{17}$$

where $A_i(S)$, $B_i(S)$, and $E_i(S)$ are functions of gate sizes of the circuit, but independent of $s(i)$. $A_i(S)$ is given by $\hat{c}_i R_i + w_i$, and $B_i(S)$ is given by $\mu_i \hat{r}_i C_i$ [14]. From the above equation, the optimal value of $s(i)$ that minimizes L_μ can be obtained. Mathematically, we evaluate h_i as follows:

$$h_i(S) = \sqrt{\frac{B_i(S)}{A_i(S)}} = \sqrt{\frac{\mu_i \hat{r}_i C_i}{\hat{c}_i R_i + w_i}}. \tag{18}$$

We observe that L_μ is a convex function in $s(i)$, and achieves the minimum at $s(i) = h_i(S)$. However, the size bounds on gate i constrain the feasible region to $s(i) \in [l(i), u(i)]$. If $h_i(S)$ lies outside the feasible region, the optimal value of $s(i)$ that minimizes L_μ under the size bounds is one of the size bounds depending on the side of the feasible region that $h_i(S)$ lies in. To satisfy the noise constraint in addition to the size bounds, we further constrain the feasible region to the right of our previously derived transformation g_i for noise reduction. From the monotonic property of noise functions, it is known that any $s(i)$ larger than g_i satisfies the noise constraint on the

- Algorithm: *Solve* $\mathcal{LRS}/\mu$
- **Output:** Optimized gate-size vector S
- begin
  1) $s(i) = l(i),\ \forall i \in G$
  2) do
  3) for each $i \in G$ in topological order
  4) evaluate C_i
  5) for each $i \in G$ in reverse topological order
  6) evaluate R_i
  7) $s(i) = \phi_i(S)$
  8) until (no further improvement)
- end

Fig. 3. Algorithm to solve the $\mathcal{LRS}/\mu$.

fan-out of i. The optimal value of $s(i)$ that minimizes L_μ under the noise and size constraints is therefore given by the larger of g_i and h_i but constrained within the size bounds.

We next define $\phi_i(S)$ to be the refinement function for a gate i, which returns its optimal size that minimizes L_μ, aims to satisfy the noise constraint on its fan-out net, and is within its size constraints. The noise constraint is satisfied if $g_i \leq s(i)$. Formally, we express ϕ_i as the following:

$$\phi_i(S) \stackrel{\Delta}{=} \min\left(u(i), \max\left(l(i), g_i(S), h_i(S)\right)\right). \tag{19}$$

$\mathcal{LRS}/\mu$ is solved by a greedy algorithm based on iteratively resizing the gates. Local refinements of $\phi_i(S)$ are performed on every gate iteratively assuming all other gate sizes are fixed. R_i and C_i are evaluated for each gate by traversing the circuit in a reverse topological order and in a topological order, respectively. g_i for every gate is evaluated based on the noise model used. We present the pseudocode of the iterative algorithm in Fig. 3.

A system of equations is formed when locally refined gate sizes for all nets are combined as $s(i) = \phi_i(S),\ \forall i \in G$. The transformation is denoted in vector form as $\Phi \stackrel{\Delta}{=} [\phi_i : i \in G]$ and we represent the system of equations in a vector form as the following:

$$S = \Phi(S). \tag{20}$$

A solution to (20) is a gate-size vector S, called a fixpoint of Φ. We prove in the following theorem that Φ is a monotonic and, under constraints, a convergent transformation.

Theorem 5: $\Phi(S)$ is a monotonic transformation over the partial order $\leq$, that is, for two gate-size vectors S_1 and S_2, given $S_1 \leq S_2$ implies $\Phi(S_1) \leq \Phi(S_2)$.

Proof: We use contradiction to assert the theorem. If $\Phi(S_1) \leq \Phi(S_2)$ is not true, then there must be at least a gate j, such that

$$\begin{aligned}
&\phi_j(S_1) > \phi_j(S_2)\\
&\quad\Rightarrow \min\left(u(j), \max\left(l(j), g_j(S_1), h_j(S_1)\right)\right)\\
&\qquad > \min\left(u(j), \max\left(l(j), g_j(S_2), h_j(S_2)\right)\right)\\
&\quad\Rightarrow \max\left(g_j(S_1), h_j(S_1)\right) > \max\left(g_j(S_2), h_j(S_2)\right)\\
&\quad\Rightarrow \left(g_i(S_1) > g_i(S_2)\right) \vee \left(h_i(S_1) > h_i(S_2)\right).
\end{aligned} \tag{21}$$

However, g_i and h_i are monotonic, as shown earlier and in [14], respectively. This is formally expressed as follows:

$$\left(g_i(S_1) \leq g_i(S_2)\right) \wedge \left(h_i(S_1) \leq h_i(S_2)\right) \qquad \forall i \in G.$$

The monotonic properties of g_i and h_i contradict the results obtained above (21). This proves that $\Phi(S)$ is a monotonic transformation. ■

C. Solving the $\mathcal{LDP}$

Let the function $Q(\lambda)$ be the solution to $\mathcal{LRS}/\lambda$. We define the $\mathcal{LDP}$ as the following:

$$\begin{aligned}
&\max\ Q(\lambda)\\
&\text{s.t.}\\
&\quad \lambda \geq 0.
\end{aligned} \tag{22}$$

$Q(\lambda)$ is not differentiable in general. Methods like steepest descent, which depend on gradient directions, are not necessarily applicable. The subgradient-optimization method is used instead, and can be viewed as a generalization of the steepest descent method in which the gradient direction is substituted by a subgradient-based direction [17]. This method however, cannot guarantee the optimal solution in all cases.

Starting with an arbitrary value of λ, the method iteratively moves from the current point to a new point following the subgradient direction. At step k, we solve the Lagrangian subproblem using the Solve $\mathcal{LRS}/\mu$ algorithm. Subsequently, for each relaxed constraint, we define the subgradient to be the right side minus the left-hand side of the constraint, evaluated at the current solution. The subgradient direction is the vector of all the subgradients. We move to a new point by multiplying a step size ρ_k to the subgradient direction and adding it to λ. Finally, λ is projected back to the nearest point in Ω_λ, so that we can solve $\mathcal{LRS}/\mu$ instead of $\mathcal{LRS}/\lambda$ in the next iteration. This is repeated till convergence.

If the step-size sequence ρ_k satisfies the two conditions $\lim_{k\to\infty} \rho_k = 0$ and $\sum_{k=1}^{\infty} \rho_k = \infty$, then the subgradient-optimization method will always converge to the optimal solution. The approach to solving $\mathcal{LDP}$ by subgradient optimization is a standard approach and is similar to the approach in [14]. The pseudocode of the Solve $\mathcal{LDP}$ algorithm is given in Fig. 4.

VI. Gate-Size-Optimization Algorithm

Based on the problem formulation and the LR procedure described in the previous section, we present the pseudocode of a gate-size-optimization algorithm that minimizes the weighted sum of gate sizes under given noise, timing, and physical gate-size constraints in Fig. 5.

Lemma 2: The solution S_{opt} obtained by the gate-size-optimization algorithm under noise constraints (Fig. 2) is a lower bound to the optimal solution of the gate-size-optimization problem under noise and timing constraints.

- Algorithm: *Solve* $\mathcal{LDP}$
- **Output:** λ that maximizes $\mathcal{LRS}/\lambda$
- begin
  1) $k = 1$
  2) $\lambda =$ arbitrary initial vector in Ω_λ
  3) $\mu_i = \sum_{j \in input(i)} \lambda_{ji}, \quad \forall i \in G$
  4) call *Solve* $\mathcal{LRS}/\mu$
  5) evaluate $A = [a_i : i \in G]$
  6) for each gate $i \in G$
  7) for all $j \in input(i)$
  8) $\lambda_{ji} = \begin{cases} \lambda_{ji} + \rho_k(a_j - A_0) & \text{if } i = PO \\ \lambda_{ji} + \rho_k(a_j + D_i - a_i) & \text{otherwise} \end{cases}$
  9) project λ to the nearest point in Ω_λ
  10) $k = k + 1$
  11) Repeat steps 3-10 until $(\sum_{i \in G} w_i s(i) - Q(\lambda)) \leq$ Error bound.
- end

Fig. 4. Algorithm to solve the $\mathcal{LDP}$.

- Algorithm: Gate-size optimization under noise and timing constraints
- **Input:** Layout extraction results
- **Output:** Optimized gate-size vector S
- begin
  1) construct DAG for circuit
  2) superimpose coupling-graph from layout extraction
  3) call *Solve* $\mathcal{LDP}$ to get optimized λ
  4) evaluate μ from λ ($\mu_i = \sum_{j \in input(i)} \lambda_{ji}, \forall i \in G$)
  5) call *Solve* $\mathcal{LRS}/\mu$ to get optimized S
- end

Fig. 5. Algorithm for gate-size optimization under noise and timing constraints.

The developed algorithm for gate-size optimization under noise constraints is used to obtain the updated lower size bound of every gate in the original gate-size-optimization problem under noise and timing constraints. This is because $S_{\rm opt}$ gives the minimum size of each gate for which NVs do not exist. Here, we assume that the initial size bounds used in both the algorithms are the same.

Lemma 3: If the optimal solution S^* to the gate-size-optimization subproblem under timing constraints (13) has a partial order over the relation $\leq$ with the optimal solution S^{**} to our gate-size-optimization problem under noise and timing constraints, the converged solution of our proposed algorithm is guaranteed to be the optimal solution.

We define $H(S) \triangleq [h_i(S) : i \in G]$ as the vector transformation of the local refinement function $h_i(S)$ defined in (16). Since S^* is the optimal solution to the gate-size-optimization subproblem under timing constraints, we have $S^* = H(S^*)$ under size constraints. Given that $S^* \leq S^{**}$, the monotonic property of H implies that $H(S^*) \leq H(S^{**})$. We therefore obtain $S^* \leq H(S^{**}) \leq S^{**}$. In addition, it is intuitive that $\mathcal{G}(\mathcal{S}^{**}) \leq S^{**}$. This implies that the optimal solution S^{**} is a fixpoint of Φ, that is $\Phi(S^{**}) = S^{**}$, since S^{**} yields the minimum sum of gate sizes. Given the monotonic property of $\Phi(S)$, it is proved that our algorithm yields the optimal solution.

VII. Experimental Results

We present results of the proposed algorithm for gate-size optimization under noise and timing constraints on the ISCAS'85 benchmarks [20] and three larger benchmarks, the parameters of which are randomly generated with realistic values in an 0.18-μm technology. We use the 2-π noise model [23] for noise estimations. The upper noise bound $U(i)$ on the nets is set to $0.2V_{\rm dd}$. For comparison, we consider four optimization stages, namely Unoptimized, Timing, Timing $\rightarrow$ Noise, and Timing + Noise.

Table II presents the number of nodes (# of Nodes), number of coupling edges in the coupling graph for the circuit (# of C Edges), and results obtained for the four stages mentioned above. Results for each stage include the weighted sum of all gate sizes (Area), the arrival time at the PO of the circuit ($T_{\rm a}$), and number of NVs in the circuit.

Unoptimized represents a circuit stage satisfying a given timing constraint, but without area optimization or noise consideration. For ease of comparison, we normalize the Area and $T_{\rm a}$ values for this stage and the corresponding values for the other stages are given with respect to this stage.

Timing represents the circuit stage after it is optimized for minimal weighted sum of gate sizes under given timing constraints. The approach is similar to the work in [14]. As expected, this stage is usually the optimal in terms of area, but has the most number of NVs. Since reaching the exact optimal using the LR approach is time consuming, we accept a small error bound for the loop termination in the Solve $\mathcal{LDP}$ algorithm. This makes the final solution close to the optimal, but not necessarily the exact optimal.

Timing $\rightarrow$ Noise represents the circuit stage after gates of circuit in Timing are sized up using the algorithm for gate-size optimization under noise constraints (Fig. 2). We constrain the size bounds such that the timing constraints are not violated during sizing. Results demonstrate an increase in area during this stage, and a decrease in number of NVs, with little effect on timing. In a few cases, the circuit is timed faster, but experiments do not show any timing violation caused by the optimal coupling-noise-reduction algorithm.

Timing + Noise represents the circuit stage after the Unoptimized circuit is optimized with our proposed algorithm for gate-size optimization under noise and timing constraints. Though the area obtained is found to be more than the Timing stage on the average, it is found to be very effective in reducing NVs and has lower area penalty in comparison to Timing $\rightarrow$ Noise. However, since this approach is not an optimal one, we observe that the final area obtained exceeds that obtained using Timing $\rightarrow$ Noise for the benchmark C3570, with both approaches resulting in the same number of final NVs. For the benchmark CKT_2, though the area is increased, this approach is able to remove all NVs, while Timing $\rightarrow$ Noise is unable to do the same.

Table III presents similar results for a further constrained noise upper bound : $U(i) = 0.1V_{\rm dd}$. We observe that gate sizing

TABLE II
NOISE-REDUCTION RESULTS: $U(i) = 0.2V_{\rm dd}$

Circuit	# of Nodes	# of C Edges	Unoptimized			Timing			Timing → Noise			Timing + Noise		
			Area	Ta	NV	Area	Ta	NV	Area	Ta	NV	Area	Ta	NV
C432	198	553	1.00	1.00	3	0.54	0.99	2	0.60	0.99	1	0.54	0.99	0
C499	245	621	1.00	1.00	4	0.43	0.99	3	0.49	0.99	0	0.43	0.99	0
C880	445	1240	1.00	1.00	2	0.75	0.99	3	0.76	0.99	0	0.73	1.00	0
C1355	589	1653	1.00	1.00	6	1.11	0.99	12	1.13	0.99	0	1.12	0.99	0
C1908	915	2655	1.00	1.00	10	0.66	0.95	10	0.67	0.94	0	0.66	0.96	0
C2670	1428	3851	1.00	1.00	14	0.58	1.00	32	0.60	1.00	1	0.60	1.00	1
C3540	1721	5086	1.00	1.00	10	0.78	0.96	16	0.79	0.93	3	0.80	0.92	3
C5315	2487	7076	1.00	1.00	15	0.50	1.00	17	0.50	1.00	0	0.50	1.00	0
C6288	2450	7239	1.00	1.00	16	0.86	0.99	15	0.87	0.99	0	0.87	0.99	0
C7552	3721	10823	1.00	1.00	20	0.49	1.00	33	0.50	1.00	0	0.50	1.00	0
CKT_1	15K	42017	1.00	1.00	53	0.21	1.00	67	0.22	1.00	0	0.22	1.00	0
CKT_2	18K	53114	1.00	1.00	73	0.23	0.98	83	0.23	0.98	1	0.24	1.00	0
CKT_3	20K	59389	1.00	1.00	116	0.28	0.99	140	0.28	0.99	0	0.28	0.99	0

TABLE III
NOISE-REDUCTION RESULTS: $U(i) = 0.1V_{\rm dd}$

Circuit	# of Nodes	# of C Edges	Unoptimized			Timing			Timing → Noise			Timing + Noise		
			Area	Ta	NV	Area	Ta	NV	Area	Ta	NV	Area	Ta	NV
C432	198	553	1.00	1.00	24	0.54	0.99	28	0.78	0.97	5	0.70	0.99	3
C499	245	621	1.00	1.00	32	0.67	1.00	35	0.84	0.96	12	0.75	1.00	9
C880	445	1240	1.00	1.00	37	0.75	0.99	52	0.87	0.99	15	0.80	0.99	10
C1355	589	1653	1.00	1.00	43	1.11	0.99	65	1.22	0.92	21	1.16	0.99	21
C1908	915	2655	1.00	1.00	88	0.66	0.95	111	0.93	0.94	27	0.64	0.96	25
C2670	1428	3851	1.00	1.00	146	0.58	1.00	172	0.74	1.00	40	0.62	1.00	30
C3540	1721	5086	1.00	1.00	166	0.78	0.96	220	0.88	0.93	61	0.82	0.98	39
C5315	2487	7076	1.00	1.00	264	0.50	1.00	297	0.66	0.99	50	0.54	1.00	38
C6288	2450	7239	1.00	1.00	220	0.86	0.99	282	0.99	0.90	80	0.93	0.94	62
C7552	3721	10823	1.00	1.00	360	0.49	1.00	414	0.65	0.99	72	0.56	1.00	57
CKT_1	15K	42017	1.00	1.00	1136	0.32	1.00	1272	0.42	0.89	108	0.32	1.00	87
CKT_2	18K	53114	1.00	1.00	1544	0.23	0.98	1573	0.34	0.92	100	0.32	0.98	94
CKT_3	20K	59389	1.00	1.00	1783	0.28	0.99	1860	0.40	0.95	158	0.35	0.99	132

is inadequate in removing all NVs in this case. This implies that gate sizing for noise reduction is a good technique for postroute noise fixes on some nets, and it is important to consider alternate noise-reduction techniques in earlier stages of design. Experimental results prove the effectiveness of our algorithm over the approach of optimizing for area and then for noise. In comparison, the proposed approach is shown to reduce larger number of NVs, which is significant in cases of constrained $U(i)$ and is efficient for area optimization. The algorithm and the framework are written in C language. The algorithm takes 0.8 s for the smallest benchmark and 1.4 min for the largest benchmark on a Pentium 2.4-GHz Xeon processor server, having 1-GB RAM and running RedHat Linux 8.0.

VIII. CONCLUSION

This paper presents a gate-sizing algorithm for coupling-noise reduction, which optimizes the area or power consumption (represented as a weighted sum of gate sizes) of a circuit while ensuring that its timing constraints are met. We formulate a problem for gate-size optimization under coupling-noise and timing constraints, and break it into two subproblems of gate-size optimization under noise and timing constraints, respectively. The subproblem of gate-size optimization under noise constraints is solved as a fixpoint computation problem on a complete lattice. The proposed algorithm to solve this problem is guaranteed to yield the optimal solution, provided it exists. The subproblem for circuit optimization under timing constraints is considered as a geometrical programming problem. The solutions to the two problems are finally combined to solve the original problem in an LR framework. Experimental results validate the efficacy of the proposed approach over an alternative approach of independent gate sizing for timing and noise reduction successively.

REFERENCES

[1] S. I. Association, *National Technology Roadmap for Semiconductors*, 1999.
[2] R. Levy, D. Blaauw, G. Braca, A. Dasgupta, A. Grinshpon, C. Oh, B. Orshav, S. Sirichotiyakul, and V. Zoloto, "Clarinet: A noise analysis tool for deep submicron design," in *Proc. Design Automation Conf.*, Los Angeles, CA, 2000, pp. 233–238.
[3] K. L. Shepard and V. Narayanan, "Noise in deep submicron digital design," in *Proc. Int. Conf. Computer Aided Design*, San Jose, CA, 1996, pp. 524–531.
[4] T. Xiao and M. Marek-Sadowska, "Gate sizing to eliminate crosstalk induced timing violation," in *Proc. Int. Conf. Computer Design*, Austin, TX, 2001, pp. 186–191.
[5] H. Zhou and D. F. Wong, "Global routing with crosstalk constraints," in *Proc. Design Automation Conf.*, San Francisco, CA, 1998, pp. 374–377.
[6] P. Saxena and C. L. Liu, "Crosstalk minimization using wire perturbations," in *Proc. Design Automation Conf.*, New Orleans, LA, 1999, pp. 100–103.
[7] C. Alpert, A. Devgan, and S. T. Quay, "Buffer insertion for noise and delay optimization," in *Proc. Design Automation Conf.*, San Francisco, CA, 1998, pp. 362–367.
[8] T. Xiao and M. Marek-Sadowska, "Crosstalk reduction by transistor sizing," in *Proc. Asian and South Pacific Design Automation Conf.*, Wanchai, Hong Kong, 1999, pp. 137–140.
[9] M. Hashimoto, M. Takahashi, and H. Onodera, "Crosstalk noise optimization by post-layout transistor sizing," in *Int. Symp. Physical Design*, Del Mar, CA, 2002, pp. 126–130.
[10] M. R. Becer, D. Blauuw, I. Algor, R. Panda, C. Oh, V. Zolotov, and I. N. Hajj, "Post-route gate sizing for crosstalk noise reduction," in *Proc. Design Automation Conf.*, Anaheim, CA, 2003, pp. 954–957.

[11] H. Zhou, "Timing analysis with crosstalk is a fixpoint on a complete lattice," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 22, no. 9, pp. 1261–1269, Sep. 2003.
[12] D. Sinha, H. Zhou, and C. Chu, "Optimal gate sizing for coupling noise reduction," in *Proc. Int. Symp. Physical Design*, Phoenix, AZ, 2004, pp. 176–181.
[13] D. Sinha and H. Zhou, "Yield driven gate sizing for coupling-noise reduction under uncertainty," in *Proc. Asian and South Pacific Design Automation Conf.*, Shanghai, China, 2005, pp. 192–197.
[14] C. Chen, C. Chu, and D. F. Wong, "Fast and exact simultaneous gate and wire sizing by Lagrangian relaxation," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 18, no. 7, pp. 1014–1025, Jul. 1999.
[15] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Application*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
[16] M. L. Fisher, "An applications oriented guide to Lagrangian relaxation," *Interfaces*, vol. 15, no. 2, pp. 10–21, Mar./Apr. 1985.
[17] D. G. Luenberger, *Linear and Nonlinear Programming*. Reading, MA: Addison-Wesley, 1984.
[18] I. H. Jiang, Y. Chang, and J. Jou, "Crosstalk-driven interconnect optimization by simultaneous gate and wire sizing," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 19, no. 9, pp. 999–1010, Sep. 2000.
[19] D. Sinha and H. Zhou, "Gate sizing for crosstalk reduction under timing constraints by Lagrangian relaxation," in *Proc. Int. Conf. Computer-Aided Design*, San Jose, CA, 2004, pp. 14–19.
[20] F. Brglez and H. Fujiwara, "A neutral netlist of 10 combinatorial benchmark circuits," in *Proc. Int. Symp. Circuits and Systems*, Kyoto, Japan, 1985, pp. 695–698.
[21] B. A. Davey and H. A. Priestley, *Introduction to Lattices and Order*. Cambridge, U.K.: Cambridge Univ. Press, 1990.
[22] P. Cousot and R. Cousot, "Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints," in *Proc. ACM Symp. Principles Programming Languages*, Los Angeles, CA, Jan. 1977, pp. 238–252.
[23] J. Cong, D. Z. Pang, and P. V. Srinivas, "Improved crosstalk modeling for noise constrained interconnect optimization," in *Proc. Asia South Pacific Design Automation Conf.*, Yokohama, Japan, 2001, pp. 373–378.

Debjit Sinha received the B.Tech. (honors) degree in electrical engineering from the Indian Institute of Technology, Kharagpur, India, in 2001. He is currently a Ph.D. degree candidate of computer engineering at Northwestern University, Evanston, IL.

From 2001 to 2002, he worked on a project involving research and development of software tools for Compact RISC processors sponsored by National Semiconductor Corporation, USA. In 2002, he joined Northwestern University. His research interests include algorithms and computer-aided design for very large scale integrated circuits, especially deep-submicrometer physical design.

Hai Zhou (M'04–SM'04) received the B.S. and M.S. degrees in computer science and technology from Tsinghua University, Bejing, China, in 1992 and 1994, respectively, and the Ph.D. degree in computer sciences from the University of Texas, Austin, in 1999.

Before he joined the faculty of Northwestern University, he was with the Advanced Technology Group, Synopsys, Inc. He is currently an Assistant Professor of Electrical and Computer Engineering at Northwestern University, Evanston, IL. His research interests include very large scale integrated computer-aided design, algorithm design, and formal methods.

Prof. Zhou served on the technical program committees of the ACM International Symposium on Physical Design and the IEEE International Conference on Computer-Aided Design. He is a recipient of the CAREER Award from the National Science Foundation in 2003.