Timing Optimization During the Physical Synthesis of Cell-Based VLSI Circuits

Aluno: Vinícius dos Santos Livramento[1]
Orientador: Luiz Cláudio Villar dos Santos[1] **Coorientador: José Luís Güntzel**[2]

[1]Programa de Pós-Graduação em Engenharia de Automação e Sistemas (PGEAS)
[2]Programa de Pós-Graduação em Ciência da Computação (PPGCC)
[1,2]Universidade de Santa Catarina (UFSC) - Florianópolis - SC - Brasil
vinilivramento@gmail.com, luiz.santos@ufsc.br, j.guntzel@ufsc.br

1. Introduction, Identification of the Target Problems, and Publications

The evolution of CMOS technology made possible integrated circuits with billions of transistors assembled into a single silicon chip, giving rise to the jargon Very-Large-Scale Integration (VLSI). VLSI circuits span a wide range class of applications, including Application Specific Circuits and Systems-On-Chip. The latter are responsible for fueling the consumer electronics market, especially in the segment of smartphones and tablets, which are responsible for pushing hardware performance requirements every new generation. The required clock frequency affects the performance of a VLSI circuit and induces timing constraints that must be properly handled by synthesis tools. This thesis focuses on techniques for timing closure of cell-based VLSI circuits, i.e. techniques able to iteratively reduce the number of timing violations until the synthesis of the synchronous digital system reaches the specified target frequency.

The design of most digital VLSI circuits follows the cell-based methodology, where the circuit description is mapped into a library of pre-characterized cells, placed in the chip surface, and routed. During the physical synthesis of VLSI circuits, several optimization techniques are used to iteratively reduce the number of timing violations until the target clock frequency is met. Although there has been research in timing optimization techniques for more than 30 years, the evolution and changes in the semiconductor industry is very rapid and new research and techniques must consider the challenges to be tackled for contemporary physical synthesis. Among the challenges to be tackled, three can highlighted: 1) most physical synthesis problems belong to the NP-complete class; 2) the number of cells in modern circuit grows at a steep rate so as to address the increasing demand for new functionalities, and techniques must handle circuits with millions of cells; enough to handle circuits with millions of cells; 3) the time-to-market pressure leads to very fast turnaround time requirements. In short, very large instances of NP-complete problems must be solved quickly to cope with the runtime budget required by contemporary physical synthesis. This renders the need of very efficient algorithms and heuristics that must take advantage of problem specific characteristics and technology parameters. [Alpert et al. 2008]. This thesis targets two important timing optimization problems that affect the interconnect synthesis of VLSI circuits:

- **Problem 1: Incremental Timing-Driven Placement (ITDP)** aims is to relocate a subset of placed cells so as to minimize timing violations trying to achieve the specified clock period (to be able to run at the target clock frequency), while satisfying placement legality constraints [Kahng et al. 2011].
- **Problem 2: Incremental Timing-Driven Layer Assignment (ITLA)** aims to re-assign the routing metal layers of critical and non-critical net (interconnect) segments so as to minimize the circuit timing violations, while satisfying the routing capacity constraints [Alpert et al. 2012].

The solutions presented in this thesis bring innovations that were recognized by the scientific community, leading to 3 article publications [Livramento et al. 2014, Livramento et al. 2016b, Livramento et al. 2016a] and 5 conference papers [Guth et al. 2015, Livramento et al. 2015, Netto et al. 2016c, Netto et al. 2016b, Netto et al. 2016a]. Besides, the implementation of the proposed technique for Problem 1 was submitted to the international contest on Computer-Aided Design to allow a direct comparison of their results with other groups worldwide. This effort resulted in one first place award in the ACM/SIGDA ICCAD Contest on Incremental Timing-Driven Placement 2015 among 42 teams [Kim et al. 2015].

2. Related Work, and Scientific Contributions

2.1. Problem 1: Incremental Timing-Driven Placement

ITDP techniques can be classified into path-based and net-based. **Path-based** approaches minimize timing violations through accurate modeling of the timing of circuit paths, generally using linear programming. The main drawback of path-based techniques is their scalability in face of large circuits, due to the exponential number of paths to be optimized. **Net-based** techniques translate timing information into net-weights or net-length constraints and minimize a weighted wirelength objective. The basic idea of net-weighting techniques is to assign higher weights to critical nets so as to guide the placement engine towards shorter nets. Net-length techniques specify constraints on critical nets so as to bound their maximum lengths. Net-based techniques can be either static or dynamic. The main drawback of most net-based techniques is to rely on a single step to generate the net-weights, which may become inaccurate during the optimization process. In addition, most works on TDP found in the literature place sequential and combinational cells indistinguishably, overlooking the impact of register placement on the clock tree distribution. Since register placement largely affects both clock tree synthesis and timing closure, it should be properly coupled with ITDP so that the optimization obtained from one technique does not undermine the other's, avoiding disruptions on the quality of solution. Thus, the main contribution for Problem 1 is:

- **A novel incremental timing-driven placement formulation based on Lagrangian Relaxation**: A new Lagrangian Relaxation formulation for ITDP that minimizes the total negative slack for both setup and hold timing violations, where Lagrange multipliers are used as net weights and are dynamically updated with an accurate timing analyzer. To solve the formulation, this work proposes a technique that relies on a novel discrete search and employs Euclidean distance to define a proper neighborhood.

2.2. Problem 2: Incremental Timing-Driven Layer Assignment

Several layer assignment techniques perform **net-by-net** iterative improvement steps to accomplish the overall timing optimization. The main limitation of all such techniques results exactly from their net-by-net approach, which may lead to locally-optimal solutions, as highlighted in [Yu et al. 2015]. The very limited availability of wide and thick wires may lead to poor timing optimization when an inadequate net ordering is adopted. Besides, some of those techniques assume that all segments of a given net must share the same layer, which may induce over-allocation. A few techniques perform **all-net** simultaneous optimization to overcome the limitations of net-by-net strategies. Different models like network flow and semidefinite programming were employed. Unfortunately, the objective functions adopted in such works, namely the sum of net delays or the maximum net delay, turns out limiting potential improvements, since large net delays might not lead to a timing violation. Most importantly, the main limitation of all incremental layer assignment techniques reported so far lies in the simplified timing model adopted to guide the optimization. The mismatch between the estimated and the actual timing reported in industrial timing engines ranges may achieve up to 400%. Despite the clear inadequacy of overly pessimistic engines, the accurate timing analyzers available from conventional EDA packages were never used by any technique reported so far. We put this down to the opacity of such analyzers, which do not report timing information for inner net segments to protect their intellectual property. We realized that the key to overcoming their lack of inner observability is the exploitation of flow conservation conditions so as to extract inner timing information. Therefore, the main contribution for Problem 2 is:

- **A novel incremental layer assignment technique driven by an industrial timing engine**: The new approach handles simultaneously critical and non-critical segments and exploits flow conservation conditions to extract timing information for each net segment individually, thereby enabling the use of an external timing engine.

3. Proposed Problem Formulations

3.1. Problem 1: Incremental Timing-Driven Placement

The first target problem can be formulated as follows: **Clock-tree-aware ITDP:** Given a placement solution P, find a new placement solution P^* that minimizes clock tree capacitance and the total negative slack. We propose an approach to solve the target problem by solving two subproblems, which can be formulated as follows. **Subproblem 1 (IRP):** Given a placement solution P, find a register relocation solution that minimizes clock tree capacitance and induces a new placement P'. **Subproblem 2 (ITDP):** Given a placement P', find a new placement P^* that minimizes the total negative slack. To properly address the target problem via decomposition, we tailored instances of the subproblems by appropriate choices of **constraints** and **objective functions**. To make sure that both subproblems find a legal placement solution, we ensure that the new cell locations are aligned to a standard cell site and to a row and do not overlap. To avoid that significant disruptions may impair the quality of the solutions provided by upstream optimization steps, an upper bound is enforced for the maximum displacement of every cell with respect to its initial location. To avoid major changes that may compromise the quality of solutions in upcoming optimization steps (e.g. routing), upper bounds are defined for signal wirelength and placement density increase.

As objective function for IRP, we adopt the local clock-tree wirelength, which comprises the wiring between local clock buffers and every sequential element driven by them. As objective function for ITDP, we adopt the sum of total negative slacks at timing endpoints ($\mathcal{TE}$) by taking into account both setup (late) and hold (early) timing violations. However, the evaluation of the latter objective function is more elaborate than the former. Let r_j^L (r_j^E) and a_j^L (a_j^E) denote the required and arrival times for late (early) scenarios. The objective function for ITDP could use them directly in the evaluation of negative slacks as follows: $\sum_{j \in PO} \min\ 0, r_j^L - a_j^L\ + \sum_{j \in PO} \min\ 0, r^E - a_j^E\ $. Unfortunately, the evaluation of arrival timing constraints at each primary output is hard to handle. That is why we cast the resulting instance of ITDP into a Lagrangian Relaxation formulation, a technique that approximates the optimal solution by removing the hard constraints and incorporating them into the cost function, as penalty terms, weighted by coefficients (λ) known as Lagrange Multiplier (LM). Besides, instead of using arrival times, we apply the well-known KKT conditions to simplify the objective function ($L\ $) so as to rely on late ($d_{i,j}^L$) and early ($d_{i,j}^E$) delays of a given cell j with respect to a cell i from its fanins ($\mathcal{F}_j$), as shown in Equation (1). Note that, in this case, each LM can be interpreted as a net-weight representing the criticality of a net i, j.

$$L\ : \sum_{c_j \in C} \sum_{i \in \mathcal{F}_j} \lambda_{i,j}^L\ d_{i,j}^L\ + \sum_{i \in \mathcal{F}_j} \lambda_{i,j}^E\ -d_{i,j}^E \tag{1}$$

3.2. Problem 2: Incremental Timing-Driven Layer Assignment

The layer assignment problem is defined over a 3D routing grid that can be modeled as a graph G $V, E\ $, where each vertex represents a *G-cell* and each edge represents the connectivity between two adjacent *G-cells*. The set of edges is a partition $E\quad E^w + E^v$, where E^w is the set of edges induced by the boundaries between *G-cells* in the same plane and E^v is the set of edges induced by vias. Each edge in E^w has a capacity that represents the number of detailed routing tracks allowed to pass through that edge. Therefore, assuming an initial 3D global routing solution, the incremental layer assignment problem can be stated as follows: given a set $\mathcal{S}$ of net segments and a set $\mathcal{L}$ of routing layers, re-assign the segment layers in order minimize timing violations. The problem is cast into a binary integer formulation that uses Lagrange Multipliers (λ_j^s) as weights to indicate the criticality of the nets. Therefore, the objective function in Equation (2) aims to assign one metal layer for each net segment to minimize the weighted summation of segment delay and lagrange multipliers. The Equation (3) encodes the binary decisions variables, while the Equations (4) ensures that each segment is assigned to one and only one layer. Finally, the constraint (5) ensures that the edge routing capacity between two adjacent *G-cells* in the same layer is not exceeded, where $\mathcal{R}_i^k$ denotes the set of indices to each net segment routed through the edge e_i on layer l_q.

$$Minimize \quad : \quad \lambda_j^s \cdot \sum_{l_q \in \mathcal{L}} \delta_j^s \; q \; \cdot \alpha_{j,q} \; , \; \forall s_j \in \mathcal{S} \tag{2}$$

$$Subject\ to \quad : \quad \alpha_{j,q} \quad \begin{cases} 1, & \text{if } s_j \in \mathcal{S} \text{ is assigned to } l_q \in \mathcal{L} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

$$: \quad \sum_{l_q \in \mathcal{L}} \alpha_{j,q} \quad 1, \; \forall s_j \in \mathcal{S} \tag{4}$$

$$: \quad \sum_{j \in \mathcal{R}_i^k} \alpha_{j,q} \leq c_{i,q}^e, \; \forall e_i \in E^w, \text{and } \forall l_q \in \mathcal{L} \tag{5}$$

4. Proposed Techniques

4.1. Problem 1: Incremental Timing-Driven Placement

Since the proposed technique solves the Clock-Tree-Aware ITDP problem via decomposition, the question becomes what subproblem should be solved first. As the percentage of registers in a circuit is significant (10-15% on average), a large number of cells can be expected to be relocated during register placement to obtain a compact clock tree. This is likely to largelly affect routability and circuit timing (as a result of the re-wiring between sequential and combinational elements). As a consequence, it would not be pragmatic to solve ITDP first because IRP would largely touch the standard cells (this would probably require ITDP to be applied anew). On the other hand, after incremental register placement, only the registers in critical paths would be touched by ITDP. Therefore, when ITDP is applied after register placement, little impact can be expected on clock tree wirelength. Since the latter order reduces the influence between optimization subproblems, that is the ordering adopted in our approach, whose overview is illustrated in Fig. 1 (a).

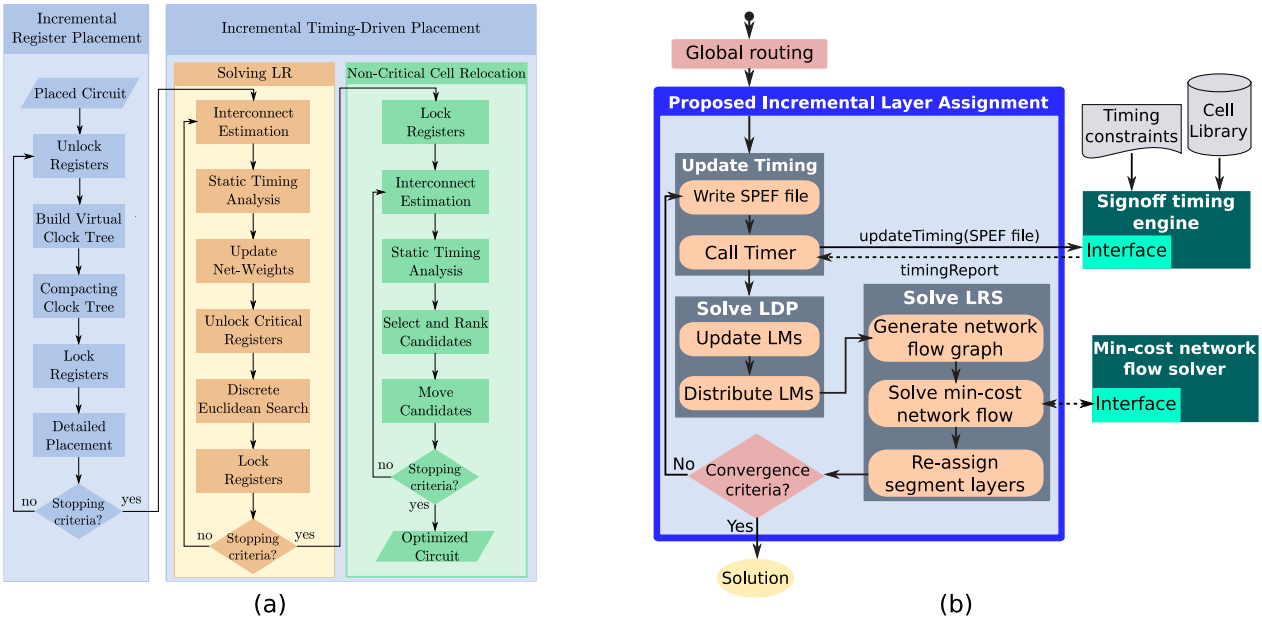

Figure 1. Flowcharts for the proposed techniques for (a) Problem 1 and (b) Problem 2.

The IRP subproblem employs a force-directed placement method to iteratively move each register to its ideal minimum-energy location. Such target location is obtained as the center of mass for parent and siblings. Since this step moves a large number of cells, it has a large impact on signal wirelength and placement density, affecting the circuit routability. Therefore, the proposed technique applies a detailed placement step to combinational cells only to keep registers out of congested areas and to recover from signal wirelength degradation. Our technique ensures that, after IRP, only critical registers are movable, i.e. those with negative slacks. This allows for relocating imbalanced latches from a positive slack side to a negative slack side. By doing so, we can improve the timing during ITDP with few register moves, thereby preserving the quality of

the previously obtained register placement solution. A static timing analysis tool is used to update the circuit timing information, which are used to update the net-weights to guide cell relocations, while a new discrete Euclidean search algorithm induces cell relocations. The last part of the proposed technique aims to exploit the optimization potential left behind by the ITDP technique. The basic idea is to move non-critical cells and reduce the capacitive load of cells belonging to the critical paths.

4.2. Problem 2: Incremental Timing-Driven Layer Assignment

Figure 1 (b) gives an overview of the proposed incremental layer assignment framework. It receives as input a 3D global routing solution. The incremental layer assignment problem is solved in three major steps. 1) *Update Timing* step first writes a parasitics file containing the distributed RC network information for the circuit. Then *Call Timer* invokes the external timing engine through a Tcl-socket. 2) *Solve LDP* updates LMs so as to increase or decrease their values proportionally to the severity of timing violations measured from the reported slacks. Then it distributes the LMs so as to comply with flow conservation conditions. 3) The last step selects critical and non-critical net segments and their respective target layers to generate the network flow graph. Then it invokes the network flow solver. Finally, it accomplishes the optimal assignment found by the min-cost flow solver. The three explained steps are repeated until predefined convergence criteria is reached.

5. Experimental Results and Comparisons with Related Works

The experimental validation of the proposed techniques relied on the ICCAD 2015 Contest infrastructure [Kim et al. 2015], which consists of consists of 8 industrial circuits with sizes between 768k and 1.93M cells.

5.1. Problem 1: Incremental Timing-Driven Placement

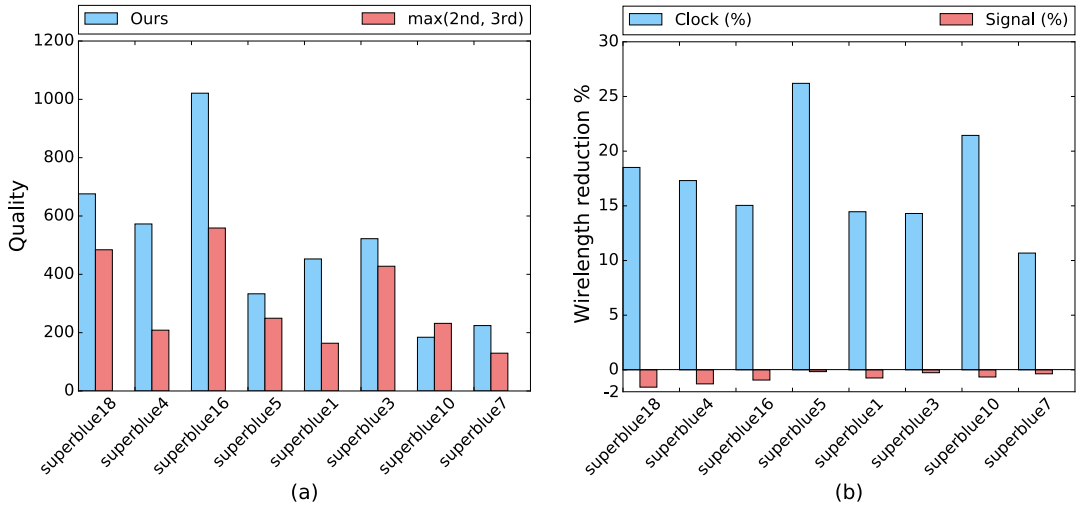

Figure 2. Comparison with related techniques under (a) quality metric and (b) wirelength reduction.

The proposed technique alone was submitted to the ICCAD Contest 2015 and produced results that were awarded the first place in that contes. That is why the experimental validation herein presented compares the proposed technique with those that obtained the 2^{nd} and 3^{rd} place in the ICCAD 2015 Contest, which represented the state-of-the-art of academic techniques. Figure 2 compares the proposed technique with the best results over the $2nd$ and $3rd$ techniques in the contest under 2 different metric. Figure 2 (a) employs the contest quality metric [Kim et al. 2015] which casts both timing violation reduction and density penalty into a single number, where the proposed technique obtained obtain average improvements of 72%. Figure 2 (b) compares clock and signal wirelength. It shows that the proposed technique obtained average improvements in clock-tree wirelength of 17%, with low penalty of 0.7% in signal wirelength.

5.2. Problem 2: Incremental Timing-Driven Layer Assignment

The proposed technique was compared with the works TILA [Yu et al. 2015] and CPLA [Liu et al. 2016], which represented the state-of-the-art of academic works in timing-driven layer assignment. Figures 3 (a)

and (b) compares the obtained timing violation reductions under worst and total negative slack metrics, respectively. The comparison shows that the proposed technique obtained around 50% and 35% less timing violations than the two related techniques.

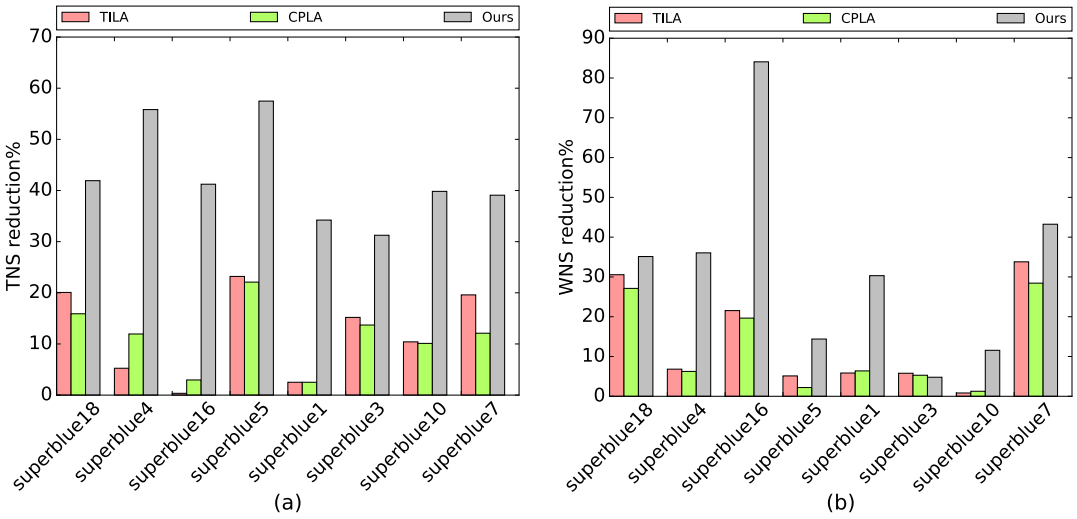

Figure 3. Comparison with related works under worst negative slack and total negative slack metrics.

6. Conclusion

The experimental results using benchmark suites derived from industrial circuits showed the effectiveness of the proposed techniques when compared with related works. Such results also reveal that LR is a powerful modeling technique for timing optimization. Since LR allows for decoupling optimization from timing analysis, it is possible to employ simplified delay models inside the optimization engine while still being guided by accurate slacks computed inside the timing analyzer.

References

Alpert, C., Li, Z., Nam, G.-J., Sze, C. N., Viswanathan, N., and Ward, S. I. (2012). Placement: hot or not? In *ACM International Conference on Computer-Aided Design*, pages 283–290.

Alpert, C. J., Mehta, D. P., and Sapatnekar, S. S. (2008). Handbook of algorithms for physical design automation. CRC Press.

Guth, C., Livramento, V., Netto, R., Fonseca, R., Güntzel, J. L., and Santos, L. (2015). Timing-driven placement based on dynamic net-weighting for efficient slack histogram compression. In *ACM International Symposium on Physical Design*, pages 141–148. (Best Paper Candidate).

Kahng, A. B., Lienig, J., Markov, I. L., and Hu, J. (2011). Vlsi physical design: From graph partitioning to timing closure. page 310. Springer.

Kim, M.-C., Hu, J., Li, J., and Viswanathan, N. (2015). Iccad-2015 cad contest in incremental timing-driven placement and benchmark suite. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 921–926.

Liu, D., Yu, B., Chowdhury, S., and Pan, D. Z. (2016). Incremental layer assignment for critical path timing. In *Design Automation Conference*, pages 85:1–85:6. ACM.

Livramento, V., Guth, C., Netto, R., Güntzel, J. L., and dos Santos, L. C. (2015). Exploiting non-critical steiner tree branches for post-placement timing optimization. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 528–535.

Livramento, V., Liu, D., Chowdhury, S., Yu, B., Xu, X., Pan, D. Z., Guntzel, J. L., and dos Santos, L. C. (2016a). Incremental layer assignment driven by an external signoff timing engine. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*.

Livramento, V., Netto, R., Guth, C., Güntzel, J. L., and Dos Santos, L. C. (2016b). Clock-tree-aware incremental timing-driven placement. *ACM Transactions on Design Automation of Electronic Systems*, 21(3):38.

Livramento, V. S., Guth, C., Güntzel, J. L., and Johann, M. O. (2014). A hybrid technique for discrete gate sizing based on lagrangian relaxation. *ACM Transactions on Design Automation of Electronic Systems*, 19(4):40.

Netto, R., Guth, C., Livramento, V., Castro, M., Pilla, L. L., and Güntzel, J. L. (2016a). Exploiting parallelism to speed up circuit legalization. In *IEEE International Conference on Electronics, Circuits and Systems*, pages 624–627.

Netto, R., Livramento, V., Guth, C., dos Santos, L. C., and Güntzel, J. L. (2016b). Evaluating the impact of circuit legalization on incremental optimization techniques. In *IEEE Symposium on Integrated Circuits and Systems Design*, pages 1–6.

Netto, R., Livramento, V., Guth, C., dos Santos, L. C., and Guntzel, J. L. (2016c). Speeding up incremental legalization with fast queries to multidimensional trees. In *IEEE Computer Society Annual Symposium on VLSI*, pages 36–41.

Yu, B., Liu, D., Chowdhury, S., and Pan, D. Z. (2015). TILA: Timing-driven incremental layer assignment. In *IEEE/ACM International Conference on Computer-Aided Design*, pages 110–117.