

# An Improved Objective for Cell Placement \*

Yu-Wen Tsay Hsiao-Pin Su Youn-Long Lin

Department of Computer Science

Tsing Hua University

Hsin-Chu, Taiwan 300, Republic of China

## Abstract

*To estimate the wiring area needed by the router to connect a signal net, most placement tools measure one half of the perimeter of the minimum rectangle enclosing all terminals of the net. In the past, this approach is reasonable because the half-perimeter value correlates well with the wiring area. As we are entering the deep-submicron era, the approach is no longer appropriate because the wiring delay must be characterized based on a distributed-RC model, in which not only the wiring area but also the wiring topology affects the wiring delay. In this paper, we show that the half-perimeter metric does not correlate well with the wiring delay under the distributed-RC model. We show that the radius of a net estimates the wiring delay more accurately than the half-perimeter metric does. We expand the acceptance criteria of a simulated annealing based placement tool to include moves that do not improve on the wiring length but do reduce the radius. Over all, for a set of benchmark circuits the critical path delays are improved up to 15%.*

## 1 Introduction

Placement is one of the most important tasks in the automatic layout design of cell-based VLSI (i.e., standard cell and gate array). It determines the geometric location of every cell in the netlist and is to be followed by the global and detailed routers. Because it largely decides on the size and performance of the chip layout, a high-quality placement tool is essential to the competitiveness of a VLSI implementation.

The exact quality metric (i.e., size and performance) of the layout will not be available until the routing is completed. Because the routing process is very time consuming, most placement tools that evaluate a large number of placement solutions cannot afford performing actual routing on every one of the solutions. Therefore, they need an approximating metric that is not only easy to calculate but also well-correlated with the final layout quality.

Until now, the most widely used approximating metric is the lower bound on the total wire length needed to complete the routing. The minimum

amount of wire needed to route a signal net using the Manhattan style is one half of the perimeter of the minimum-sized rectangle enclosing all terminals of the net. This metric has been shown to correlate well with the final layout area. It is satisfactory for designs targeting towards processing technology of 0.8  $\mu\text{m}$  or older, in which wiring-induced delay is relatively insignificant compared with the gate delay.

As the industry moves towards high performance deep-submicron design ( $< 0.5 \mu\text{m}$ ), the wiring delay becomes significant and even dominant because of both the increasing in the wiring resistance and the decreasing in the gate delay. Therefore, there are many performance-driven routers [1, 2, 3] being developed recently.

In the deep-submicron era, the wiring delay can be estimated no longer with the simple lumped-RC model accurately. Instead, a more accurate distributed-RC model, such as the Elmore delay model [4, 5] is necessary. The distributed wiring delay is dependent on not only the wiring length but also the routing tree topology. The half-perimeter metric predicts only the wiring length. Hence, it is inadequate for the modern placement tools. Therefore, we are motivated to reinvestigate the objective function for placement.

In this contribution, we first show that the wiring delay under the distributed Elmore model does not correlate well with the half-perimeter value. Instead, it correlates better with the radius of a net for a given half-perimeter value. To take the radius into account, we expand the acceptance criteria of a simulated annealing based placement tool to include moves that do not improve on the wiring length but do reduce the radius of signal nets.

## 2 The Traditional Objective

As demonstrated in many previous papers [6, 7], the traditional half-perimeter metric predicts the wiring length very well. However, to the best of our knowledge, no evaluation has been performed on whether it also predicts the wiring delay accurately under the distributed-RC model despite there have been many performance-driven routers [1, 2, 3] trying to reduce the wiring delay via topological design and wiring width sizing.

To evaluate the goodness of the half-perimeter metric, we measure the layout of a large set of benchmark netlists. The placement is done with the TimberWolf

---

\*This work was supported in part by the National Science Council, R.O.C., under contracts No. NSC85-2221-E-007-032 and NSC86-2221-E-007-019.

6.0 tool [6]. The routing is completed by using a performance-driven tree constructor. The wiring delay is calculated based on the distributed Elmore delay model [4] targeting towards a 0.6  $\mu m$  processing technology.

Given a routing tree  $T$  rooted at  $v_0$ , let  $e_i$  denote the edge from  $v_i$  to its immediate parent. The Elmore delay  $D(v_0 \rightarrow v_i)$  from the source node,  $v_0$ , to a sink node,  $v_i$ , is

$$D(v_0 \rightarrow v_i) = R_s \times C_0 + \sum_{e_j \in \text{path}(v_0 \rightarrow v_i)} r_j \times (c_j/2 + C_j),$$

where  $R_s$  is the driving resistance of the source terminal,  $C_j$  the total capacitance of the subtree rooted at  $v_j$ ,  $r_j$  the parasitic resistance of edge  $e_j$ , and  $c_j$  the parasitic capacitance of edge  $e_j$ .

We assume that all sources have identical driving capability, that is all  $R_s$ s are equal, and all sink terminals have equal load capacitances<sup>1</sup>.

We use benchmark S5378 to illustrate the correlation. After placement, routing and delay calculation, we observe that the correlation between the wiring length and the half-perimeter value is very good as expected. We use the simple linear regression model to investigate the data. The *coefficients of determination*  $r^2$  are 100%, 96.8%, 94.9%, 91.2%, 88.5% and 91.2% for 2, 3, 4, 5, 6 and over-6-terminal nets, respectively. The higher the value  $r^2$ , the more successful is the linear regression model in explaining data variation. However, the correlation between the wiring delay and the half-perimeter value is not clear (The *coefficients of determination*  $r^2$  are 52.0%, 81.2%, 81.2%, 81.7%, 89.6% and 69.7%, respectively.) , especially, when the number of terminals is large ( $> 6$ ). Therefore, we may say that the half-perimeter metric is not adequate if we are to do performance-driven placement.

### 3 Motivation

The motivation behind this work comes from the calculation of the Elmore delay. Under the Elmore model, the delays of two nets of equal wiring length could be different due to their topological difference. Let's consider those two 3-terminal nets depicted in Figure 1(a)&(b). Suppose terminal  $p_0$  is the source and sinks  $p_1$  and  $p_2$ . The RC trees of those nets are depicted in Figure 1(c) and (d), respectively. When the Elmore's distributed-RC delay model is used, the source-to-sink delays for the net shown in Figure 1(a) are calculated.

Under the Elmore model, the interconnection delay of the net of Figure 1(b) is smaller than that of the net shown in Figure 1(a). Traditional half-perimeter-based objective function cannot tell this difference because those two nets have enclosing rectangles of the same size.

<sup>1</sup>Specifications from the technology files: source driving resistance = 100 $\Omega$ ; sink loading capacitance = 15.3fF; wire resistance = 0.03 $\Omega/\mu m$ ; wire capacitance = 0.352fF/ $\mu m$ .

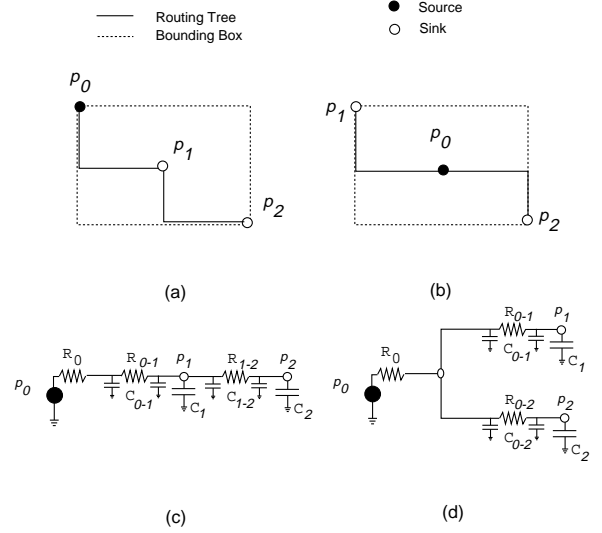


Figure 1: Two nets having equal half-perimeter values but different wiring delays.

This example tells us that it is inadequate to measure only the lower bound on the wire length. Rather, we have to consider the source/sinks distribution as well.

We observe that the radius of a net correlates to the wiring delay much better than the half-perimeter value does. Given a placement, the radius of a net is defined as the maximum Manhattan distance between its source terminal and any of its sink terminals, that is the radius of the net  $n_i$  is

$$R(n_i) = \text{MAX}_{p_j \in n_i} (|X_{sp_i} - X_{p_j}| + |Y_{sp_i} - Y_{p_j}|),$$

where  $X_{sp_i}$  and  $Y_{sp_i}$  are  $xy$  locations for the source terminal  $sp_i$ , and  $X_{p_j}$  and  $Y_{p_j}$  are for other sink terminals  $p_j$  for the net  $n_i$ .

The correlation between the wiring delay and the radius (The *coefficients of determination*  $r^2$  are 52.0%, 84.8%, 82.2%, 85.7%, 94.6% and 78.4%, respectively.) is much better than that between the wiring delay and the half-perimeter. Based on this observation, we conclude that it is good for a performance-driven placement tool to take into account the radius of a net while exploring the solution space.

### 4 The Proposed Approach

An intuitive approach to take the net radius factor into consideration is to add it to the objective function. We have tried using numerous combinations of the half-perimeter and the radius as the objective function for the placement tool to optimize. However, none of them leads to significant improvement over that uses only the original half-perimeter metric. Finally, we realize that the half-perimeter and the radius are not independent of each other. If we emphasize too much on the radius, we will sacrifice the half-perimeter

(wiring length). The longer wiring length introduces the longer wiring delay. Therefore, the half-perimeter and the radius cannot be combined straightforwardly.

Presently, we treat the half-perimeter as the primary objective and the radius secondary. We expand the acceptance criterion of the TimberWolfSC[6]. TimberWolfSC is a standard cell placement tool based on the simulated annealing optimization technique. It iteratively improves the placement quality by generating a move and deciding on whether to accept or reject the move. Since TimberWolfSC is well known to the CAD community, we will not repeat its description here.

Our expansion is quite simple. We give each move two chances to be accepted. First, a move is accepted if it is accepted by the original criterion; otherwise, we examine its contribution to the radius reduction. We change a rejected move to an accepted one if it decreases the summation of the radius of all involved nets. Below is a pseudo code description of the proposed implementations.

1. **While** the outer loop termination not reached
2.     **While** the inner loop termination not reached
3.         do a move and evaluate the cost
4.         **If**  $\text{Accept}(\Delta Cost_L, T)$  **OR**  $(\Delta Cost_R < 0)$
5.             accept the move
6.         **Else**
7.             reject the move
8.         **Endif**
9.         update the temperature  $T$
10.     **Endwhile**
11. **Endwhile**

In the pseudo code,  $\Delta Cost_L$  and  $\Delta Cost_R$  are the changes to the total wiring length and radius, respectively.

We treat two-terminal nets differently from other multiple-terminal nets for the following reason. Since the Manhattan length of the radius of a two-terminal net is equal to the length of the half-perimeter of its bounding box, any reduction in the length can also reduce the radius. Therefore, we let the radius of a two-terminal nets always be zero in order not to over-estimate their impact on the overall quality.

## 5 Experimental Results

We have used the **TimberWolfSC v6.0** [6] program to experiment with the proposed idea. TimberWolfSC v6.0 is a cell-based placement package based on the simulated annealing optimization algorithm. We have no access to the source code of the latest **TimberWolfSC v7.0** [7]. However, the proposed idea is applicable to any other placement tools based on the half-perimeter cost function as well.

Our simplified target library has the following technology specifications: wire resistance =  $0.072\Omega/\mu m$  and wire capacitance =  $0.2fF/\mu m$ .

After placement and performance-driven routing, the wiring delay of every net is calculated under the Elmore Distributed-RC delay model. We are interested in the average and maximum net delay, as well as

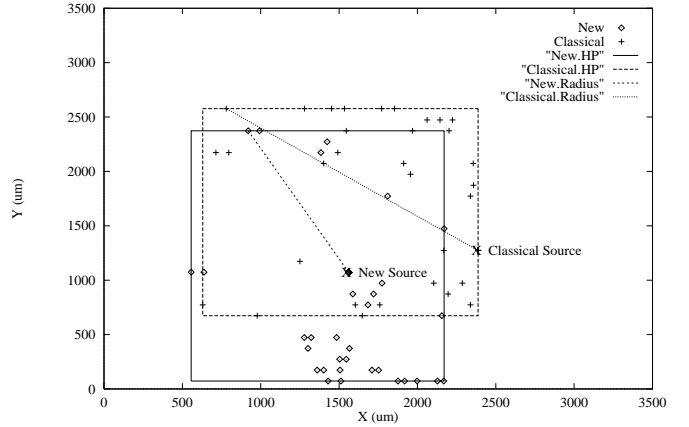


Figure 2: Terminals distribution for a net of circuit S9234 with the classical approach (half-perimeter =  $3661\mu m$ , radius =  $2911\mu m$ , delay =  $2.91ns$ ) and with the new approach (half-perimeter =  $3941\mu m$ , radius =  $1941\mu m$ , delay =  $2.52ns$ )

the circuit performance in terms of the critical path delays and the maximum single net delay.

We have laid out 6 benchmark circuits S1196, S5378, S9234, S13207, S15850 and S38417 from the MCNC Table 1 Column Size summarizes the characteristics of the benchmarks, including the number of cells, the number of nets and the number of 2-terminal nets.

Table 1 shows the average wiring delay per net for all nets, all multi-terminal nets, and CPU time required by each run of each benchmark. Since most of the nets (almost 80%) are two-terminal, the improvement in the average delay over all nets is very small (less than 3%). However, the case of multiple-terminal nets is very remarkable (up to 14%). As expected, the new objective function takes longer to compute. On the average, 5% more CPU time is required. This computation overhead is quite acceptable considering the gain we have achieved in circuit performance.

Table 2 shows the effect of the proposed idea on the placed circuits' performance, including critical path delay and single maximum net delay. We back-annotate the wiring delay values to the original netlist and use the sequential synthesis system SIS [8] to calculate the worst case clock period of each circuit. The reductions range from 5% to 15%. As for the maximum single net delay, we are able to achieve improvement ranging from 3% to 9%.

Table 3 shows that the proposed idea indeed leads to reduction in the radius of the signal nets.

We plot in Figure 2 the terminal distributions of the largest net (33 terminals) of benchmark S9234. It is shown that the source location by the new approach is closer to the center compared with that by the classical approach. The radius by the new approach is 33% shorter than that by the classical approach. Although

Table 1: Average wiring delay per net and CPU time in comparison with classical (Class.) model

Circuit	Size			All Nets(ns)			Multi-terminal Nets(ns)			CPU(s)		
	#Cell	#Net	#2-term	Class.	New	$\frac{New}{Class.}$	Class.	New	$\frac{New}{Class.}$	Class.	New	$\frac{New}{Class.}$
S1196	444	458	353	0.356	0.357	1.00	0.763	0.682	0.89	38.5	40.6	1.05
S5378	1149	1184	779	0.308	0.305	0.99	0.422	0.370	0.88	114.2	123.9	1.08
S9234	1260	1296	842	0.246	0.241	0.98	0.353	0.332	0.94	140.3	145.4	1.04
S13207	3126	3188	2201	0.279	0.275	0.99	0.435	0.419	0.96	560.6	575.8	1.03
S15850	3148	3225	2193	0.311	0.301	0.98	0.482	0.451	0.94	567.6	581.6	1.03
S38417	10079	10107	7557	0.273	0.263	0.96	0.659	0.577	0.86	2000.0	2080.2	1.04

its half-perimeter is 7% longer, the new approach improves the delay by 14%.

Table 2: Performance (ns) after placement

Circuit	Critical path delay			Max. single net delay		
	Class.	New	$\frac{New}{Class.}$	Class.	New	$\frac{New}{Class.}$
S1196	28.68	26.54	0.93	4.633	4.291	0.93
S5378	24.38	22.20	0.91	6.142	5.673	0.92
S9234	29.05	24.56	0.85	3.143	3.041	0.97
S13207	44.11	43.43	0.98	5.502	5.437	0.99
S15850	57.20	52.01	0.91	12.049	10.907	0.91
S38417	92.00	88.20	0.96	13.348	12.086	0.91

Table 3: Reduction in the total radius

Circuit	Total Radius(M)		
	Class.	New	$\frac{New}{Class.}$
S1196	0.093	0.079	0.85
S5378	0.304	0.250	0.82
S9234	0.277	0.259	0.94
S13207	0.753	0.706	0.94
S15850	0.981	0.898	0.92
S38417	3.347	2.800	0.84

Beside the computation overhead, the proposed idea has other side effect. The delay of multi-terminal nets are improved at the expense of slowing down 2-terminal nets. For example, we plot for circuit S9234 in Figure 3 the average delays of net sets with different number of terminals. It is clear that with the new approach, multi-terminal nets are sped up significantly.

## 6 Conclusions

We have proposed for performance-driven cell placement a new objective targeting toward the Elmore RC-tree delay model. We have shown that the radius of a net is an important factor in determining the wiring delay of the net. A simple modification to the acceptance criteria of a simulated annealing-based placement tool leads to significant improvement in the circuit performance. The idea is also applicable to other placement tools as well.

In our experiment, we use a  $0.6\mu\text{m}$  technology. As we move deeper into the deep submicron era to implement circuits, the performance improvement will be more significant, because the wiring delay will become more dominant.

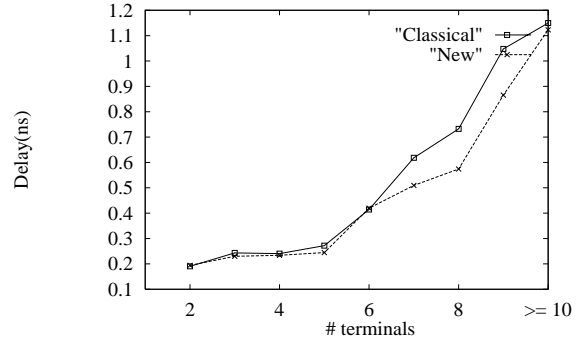


Figure 3: Average delays vs. the number of terminals for circuit S9234

## References

- [1] K. D. Boese, A. B. Kahng, and G. Robins, "High-performance routing trees with identified critical sinks," Design Automation Conference (DAC), pp. 182-187, Jun. 1993.
- [2] J. Cong, A. B. Kahng, G. Robins, M. Sarrafzadeh and C. K. Wong, "Provably good performance-driven global routing," IEEE Trans. on Computer-Aided Design, vol. CAD-11, pp. 739-752, Jun. 1992.
- [3] J. Cong, K. S. Leung, and D. Zhou, "Performance-driven interconnect design-based on distributed RC delay model," Design Automation Conference (DAC), pp. 660-611, Jun. 1993.
- [4] W. C. Elmore, "The transient response of damped linear networks with particular regard to wide band amplifiers," J. Appl. Phys., vol.19, pp. 55-63, 1948.
- [5] T. Koide, M. Ono, S. Wakabayashi, Y. Nishimaru and N. Yoshida, "A new performance-driven placement method with the Elmore delay model for row based VLSIs," Asia and South Pacific Design Automation Conference (ASP-DAC), pp. 405-412, Aug. 1995.
- [6] C. Sechen, "TimberWolf6.0: Mixed macro/standard cell floor planning, placement and routing package," User's manual, Yale University, Sep., 1991.
- [7] W. J. Sun, and C. Sechen, "Efficient and effective placement for very large circuits," IEEE Trans. on Computer-Aided Design, vol. CAD-14, pp. 349-359, Mar. 1995.
- [8] SIS: A system for sequential circuit synthesis, Department of electrical engineering and computer science, University of California, Berkeley.