Fast, Accurate A Priori Routing Delay Estimation

Jinhai Qiu
Implementation Group
Synopsys Inc.
Mountain View, CA
Jinhai.Qiu@synopsys.com

Sherief Reda
Division of Engineering
Brown University
Providence, RI 02912
sherief_reda@brown.edu

Soha Hassoun
Computer Science Dept
Tufts University
Medford, MA 02155
soha@cs.tufts.edu

ABSTRACT

We propose in this paper a novel approach for speeding timing closure. We focus on the problem of accurate post-routing delay estimation from a given placement. Post-routing delays differ from placement delays due to factors such as net topology, layer assignment and congestion. Fundamental to our approach is utilizing an existing base design to predict future designs. We present four wire-delay estimation techniques based on: delay fitting, Steiner-aware delay fitting, Steiner-aware RC sampling, and scaled Steiner-aware RC sampling. We apply our techniques to several designs, and using an industrial flow, we demonstrate that it is possible to estimate the routing delays with an average estimation error of 16% on benchmark circuits. These results are of practical value, and improve on the state-of-the-art industrial estimation capabilities.

Categories & Subject Descriptors
B.7.1 [Integrated Circuits]: Types and Design Styles.
General Terms: Design, Performance, Algorithms.
Keywords: Interconnects, delay estimation, placement and routing.

1. INTRODUCTION

The objective of a priori wirelength estimation is to predict the length of a net route before the computationally intensive routing stage. The wirelengths are used to compute delays, which are utilized to speed up physical synthesis and to reduce the number of required design iterations. Several wirelength estimation techniques have been investigated. The main hypothesis motivating previous work in wirelength estimation is that wirelength correlates well with post-route delay. However, post-route delays differ from placement estimates mainly due to net topology, routing congestion and layer assignment. Congestion occurs due to blockages and finite routing resources, which changes the topology of nets to avoid congestion. Layer assignment can significantly change delay. For example, a net of estimated length of 100 μm will have a delay of 0.9 *ps* if routed in the fourth metal layer in a typical 90 nm process, but it will have a delay of 1.3 *ps* if routed in the first metal layer. The layer choice results in a delay difference of 44%.

SLIP'10, June 13, 2010, Anaheim, California, USA.

The problem we address in this paper is *a priori wire delay estimation* where the estimation occurs after placement but prior to routing. Our underlying approach is based on characterizing a known design, referred to as the *base* design, to predict delays for a design variant or other designs that utilize the same router. Our estimation approach captures empirically the router characteristics as encountered in the base design and this empirical information is used to estimate the delays of other designs. Our approach is practical as designers typically use the same router in either designing a next-generation product based on one with similar function (a design variant), or designing a new integrated circuit using the same design technology (different design, same technology).

We present in this paper four techniques to estimate routing delays. We demonstrate the results of applying these techniques to estimate the delays of design variants and completely different designs than a base case. We highlight our contributions:

- We propose four delay estimators that vary in complexity and in estimation capabilities. These estimators incorporate many factors from the base design to determine the final pin-to-pin routing delays. These factors include net degree, Steiner wirelength, and the general impact of topology and congestion.
- Our estimators empirically capture the routing characteristics of a given router and thus the estimation results are likely to occur when the same router is utilized. Thus, the applicability of our approach is ideal in an industrial setting where the same router is used across different designs.
- Our estimators use lookup tables, therefore quickly providing estimation results. The estimation runtime is essentially proportional to the number of the pin-to-pin paths in a design. Our approach is scalable and adaptable to very large designs.
- Utilizing an industrial design flow, we validate the accuracy of our delay estimates against the actual detailed routes. Our experimental results indicate that we are able to achieve an average 16% estimation error across a number of different designs and design variants.

The paper is organized as follows. In Section 2, we give an overview of previous related work. In Section 3, we present our four estimation techniques outlining the construction of the lookup tables and the information needed to utilize the tables. We conclude this section with a summary of our techniques. In Section 4, we present experimental data that validate our approach and contrast our techniques. Section 5 summarizes the contributions of our work and provides direction for future work.

2. PREVIOUS WORK

The prediction of wirelength characteristics has been an active research topic for more than three decades. The interest in wirelength prediction stems from two reasons: (1) total wirelength determines the die area and consequently the cost of the circuit, and (2) wirelength, in part, determines the delay and consequently the performance of the circuit. Thus, it is important to evaluate wirelength as early as possible.

A great amount of work focuses on the problem of *a priori* estimation of the placement wirelength before layout and routing [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]. A popular estimation technique uses Rent's rule to deduce various circuit statistics [11, 1, 5]. While techniques based on Rent's rule are successful in calculating overall or average wirelength statistics, they are inadequate in predicting individual wire length. It is also possible to estimate wirelength characteristics through the use of lookup tables (or wire-load models) built from knowledge of previously routed circuits [12, 13]. While wire-load models are very fast estimators, they are typically inaccurate. Another popular way to deduce wirelength statistics is through topological analysis of the circuit netlist [7, 8, 9, 10] with successful techniques such as mutual contraction [8], edge separability [9] and intrinsic shortest path length [10]. This estimation can be aided through modeling of the underlying placement and routing tools [3].

While delay is generally correlated with wirelength, the exact delay of a wire depends on many factors including the exact topology of the net taking into account congestion and layer assignment. Thus, it is possible to find two wires with the same length in placement yet with completely different delays in routing. Congestion arises from the contention of design nets on the limited metal layer resources. Before routing takes place, congestion maps can be built through probabilistic analysis of the placement [14, 15] and through analysis of the netlist characteristics and topology [16]. Even with congestion information, the exact wire delay cannot be accurately calculated due to the impact of metal layer assignment in determining the final resistance and capacitance of wires. In general, long wires tend to reside on top metal layers and short wires on bottom metal layers. In addition, wires that belong to the critical path(s) are given preference to follow their shortest routing connections on the layers that lead to the minimum delay.

The focus of this paper is fast estimation of routing delay given the placement of a circuit. Our work differs from previous approaches as we focus directly on delay calculation through estimation of the routing resistance and capacitance while taking into account net topology, the impact of congestion, and layer assignment. Our methods use elaborate lookup tables to estimate the routing delay in a fast manner.

3. ROUTING DELAY ESTIMATION

The final pin-to-pin delay in a routed design could differ from their placement-based, pre-routing estimates for a number of factors. These factors include the following:

- Layer assignment offers trade-offs between interconnect delay and metal utilization. Layer assignment for the different segments of a net will impact the final pin-to-pin delay. Layer assignment also impact the number of inserted vias, which also impacts delay.
- Competition among the different nets for the available metal resources could lead to net topologies with detours. Thus, the final net topologies could be different than their Steiner-based placement counterparts.

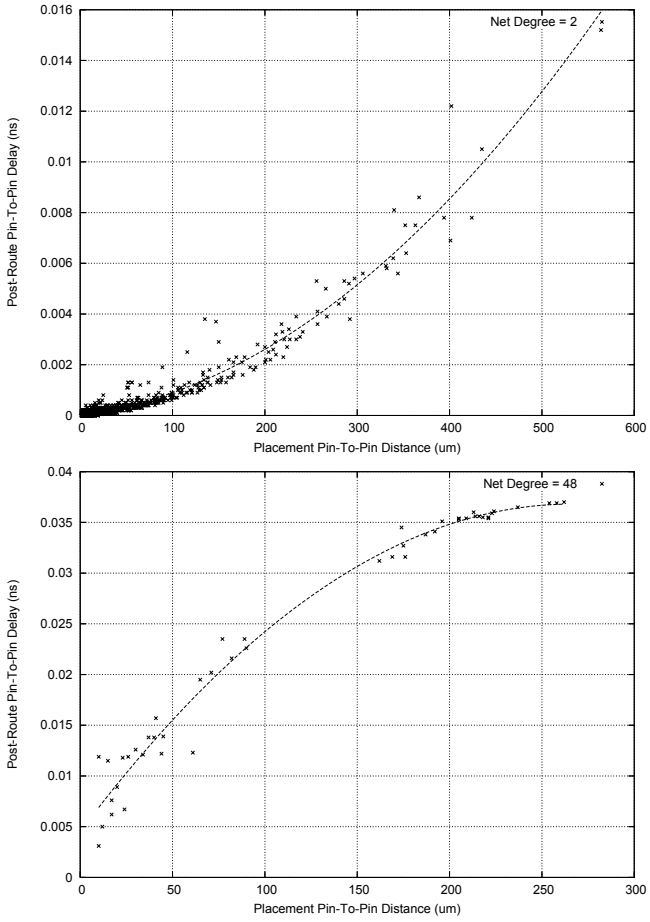

Figure 1: The quadratic fitting between the pin-to-pin placement distance and the post-route delay for two-pin nets and 48-pin nets for one of our benchmarks.

- Blockages in the design (e.g., arising from the use of IP blocks) could also lead to final net topologies that are different from their placement counterparts.

Our delay estimation techniques attempt to incorporate the impact of these factors into their estimates. The underlying approach in our estimation techniques is characterizing a known base design(s) to predict wire delays of design variants or other designs. Our techniques empirically capture the characteristics of a router as encountered in the base design. For any net, we identify five features in a final routed design that are meaningful for delay estimation purposes. These five features we capture are: (1) net degree, (2) pin-to-pin route length, (3) routed Steiner tree wirelength of the net, (4) sink pin capacitance, and (5) congestion and layer assignment as measured by the difference between placement-based and routing-based delays. Our methods offer a trade-off between their ability to capture these features, and hence their modeling accuracy, and the complexities of the lookup tables. Our techniques are as follows.

1. Delay Sampling (DS): Our first technique is motivated by the observation that source-pin to sink-pin final routing delays depend quadratically on the pin-to-pin placement length. For example, Figure 1 plots the relationship between the pin-to-pin placement length and final routing delays for one of our benchmarks for nets of de-

gree 2 and nets of degree 48. The plots show that the relationship between placement length and final delay can be reasonably captured with a quadratic fitting function.

To build our model, our estimation technique analyzes the base design(s) and develops for each net degree, g, a quadratic function of the pin-to-pin placement length l. The empirical data from the base design(s) are used to compute of three coefficients $c_0(g)$, $c_1(g)$, and $c_2(g)$, and use them to compute the estimated wire delay according to:

$$d = c_0(g) + c_1(g) \times l^1 + c_2(g) \times l^2. \quad (1)$$

To estimate the delay of a pin-to-pin wire of a variant or a different design, the estimator uses Equation (1) to find the final delay as a function of the wire's length and its net's degree.

2. Steiner-Aware Delay Sampling (SDS): While delay sampling captures the impact of net degree on wire delay prediction, it lumps all other delay differentiation factors (e.g., the net's total wirelength, its topology and its metal layers) in the fitting coefficients. Delay sampling does not distinguish between two same-length pin-to-pin wires each belonging to nets with similar degrees yet having Steiner trees with different total wirelengths. The difference in total wirelength translates to larger interconnect capacitance which impacts the pin-to-pin delay. Our *Steiner-Aware Delay Sampling (SDS)* technique improves over our first approach (DS) by incorporating a net's total wire length into its post-route pin-to-pin delay estimations.

Using a base design, we sample the delay for each pin-to-pin wire, group and then average the delays according to net degree, pin-to-pin distance, and the Steiner-tree wirelength. We thus build a lookup table that predicts pin-to-pin delay for each unique combination of (1) net degree, (2) Steiner tree wire length and (3) pin-to-pin distance. Given a pin-to-pin wire that is part of a net, we use the delay value indexed by the net's degree, the Steiner tree wirelength of the net, and the wire's pin-to-pin distance as the estimated delay. Data from the lookup tables are interpolated and extrapolated as needed.

3. Steiner-Aware RC Sampling (SRCS): With the SDS technique, we accounted for a net's total wirelength and its impact in on estimating its pin-to-pin delays. One of the problems of the first two techniques is that they do not differentiate between wires that see different gate capacitances at the sink node (assuming all other factors are the same). In the previous two techniques, the delay contributions from both the wire and sink pin capacitances are lumped. Indexing occurs only by wire length, net degree and total wirelength. The gate capacitance at the sink node is an important factor that could differentiate pin-to-pin delays. The *Steiner-Aware RC Sampling (SRCS)* technique accounts for the sink's capacitive load. Instead of sampling the delay as in the previous two methods, we sample for a base design(s) the resistance and the capacitance of each net. We create lookup values containing a resistance, $R(g,w)$, and a capacitance, $C(g,w)$, for each unique combination of net degree g and Steiner tree wirelength w.

To estimate the delay d for a pin-to-pin wire of length l that belongs to a net n, we first obtain $R(g,w)$ and $C(g,w)$ from the lookup table using n's net degree g and its placement's Steiner tree wirelength w. We then use Equation (2) to calculate the delay estimation. In Equation (2), we make the conservative assumption that each unit resistance sees all distributed capacitance (the first term in the equation) and C_{sink} (the second term in the equation), where C_{sink} is the sink pin's gate capacitance. We set the resistance of the pin-to-pin wire to be equal the total resistance of the net $R(g,w)$ multiplied by the ratio of the placement length of the wire to the net's placement Steiner wirelength w.

$$d = R(g,w) \times \frac{l}{w} \times (\frac{C(g,w)}{2} + C_{sink}) \quad (2)$$

While we use an Elmore delay based formula for delay estimation, Elmore delay can be easily fitted with great accuracy to the final routing delay of the pin-to-pin wire [17]. A fitted Elmore delay based on the final routes is equal to

$$d_{final} = a \times R_{l\_ex} \times C_{n\_ex} + b \times R_{l\_ex} \times C_{sink}, \quad (3)$$

where d_{final} is the final routing delay as reported in a SDF file, R_{l_ex} and C_{n_ex} are the actual post-route wire resistance and total net capacitance as reported by post-route RC extraction tool, and a and b are fitting coefficients. The fitting coefficients a and b can be incorporated in Equation (2) as follows

$$d = R(g,w) \times \frac{l}{w} \times (a \times \frac{C(g,w)}{2} + b \times C_{sink}) \quad (4)$$

to improve the accuracy of our post-placement delay estimator.

4. Scaled SRCS: One of the problems in any lookup table methodology is that there were will be estimation instances where the indexes into the lookup table do not lead to a valid record. In this case the delay estimate can be set equal to the nearest existing record in the lookup table or it can be set equal to an interpolation of the nearest existing records in the lookup table. In this technique, *Scaled Steiner Aware RC Sampling (Scaled SRCS)*, we attempt to improve the delay estimates for nets with characteristics (e.g., net degree, Steiner length) that do not lead to a match in the lookup table.

We introduce *scaling factors* that indicate on average the change in delay when the same net n is routed only on the first metal layer using the shortest path. Note that our choice of the first metal layer is arbitrary. We define s_r as the *resistive scaling factor* for net n. s_r indicates the relative magnitude of the resistance of the net in the actual layout when compared to the its resistance if it is routed entirely in metal 1. Similarly we define s_c as the *capacitive scaling factor* for net n. s_c indicates the relative magnitude of the capacitance of the net in the actual layout when compared to the its capacitance if it is routed entirely in metal 1. When obtaining the reference metal1 route from the original route for n, we assume the following:

- All routes on metal 1 use the shortest pin-to-pin path.
- Since all metal segments are on metal 1, there are no metal segments on different metal layers. All vias are removed.

The first step to build the lookup tables for this technique is to use the base design(s) to extract the total resistance R_n and capacitance, C_n for every net n and compute the net's resistance $R_{n\ metal1}$ and capacitance $C_{n\ metal1}$ as if the net is entirely routed using a shortest path, and using only metal 1. We then calculate the resistive and capacitive scaling actors s_r and s_c of net n as follows:

$$s_r = \frac{R_{n\ actual}}{R_{n\ metal1}} \quad (5)$$

$$s_c = \frac{C_{n\ actual}}{C_{n\ metal1}} \quad (6)$$

(A) When building the estimation function or the lookup table, we use the following information:

	net degree	route length	net delay	route steiner tree length
DS	x	x	x	
SDS	x	x	x	x
SRCS	x			x
Scaled SRCS	x	x		x

(B) When computing the stimated delay, we utilize the following information:

	net degree	placement length	placement steiner tree length	C_sink
DS	x	x		
SDS	x	x	x	
SRCS	x	x	x	x
Scaled SRCS	x	x	x	x

Table 1: Summary of each estimation technique's main characteristics.

To build our lookup table, we compute the scaling factors for the different nets and then group and average them according to net degree and Steiner tree wire length. Thus, for each unique combination of net degree and Steiner tree wire length, our lookup table stores a resistive scale factor $s_r(g,w)$ and a capacitive scaling factor $s_c(g,w)$.

To estimate the delay d for a pin-to-pin connection with length l belonging to net n of degree g and Steiner length w, we compute the net's resistance, $CR(w)$ and capacitance, $CC(w)$, assuming a shortest path metal 1 route and using the placement's Steiner tree, and then adjusting these delay using $s_r(g,w)$ and $s_c(g,w)$

$$d = CR(w) \times \frac{l}{w} \times s_r(g,w) \times (a \times CC(w) \times s_c(g,w) + b \times C_{sink}), \quad (7)$$

where a and b are the fitting coefficients from Equation (3) that better fit Elmore delay estimate to the actual delays. The main advantage of estimating the delay using Equation (7) in comparison with Equation (4) is that the delay estimate directly takes into account the placement delay in its computation which provides a better estimation in case the net characteristics do not match any of the entries in the lookup table. This is evident when comparing the resistive and capacitive terms in each equation.

Summary of Techniques: We summarize the four estimation techniques in Table 1(A), and Table 1(B). The techniques vary in four ways. First, each technique utilizes a subset of information from the base design(s) to build an estimation function or lookup table. Second, each technique's lookup table provides unique information. The first two techniques provide delay information. The third technique provides resistive and capacitive values. The fourth technique provides scaling factors. Third, the index into the lookup table utilizes some or all placement data. Fourth, we utilize the lookup information along with placement information to compute the delays. For example, for our SRCS technique, we use the net degree and Steiner tree length from the base design to build the lookup table. The lookup table provides two numbers: a resistance value and a capacitance value. To compute the delay estimate for a pin-to-pin wire, we use the net degree and the placement Steiner tree length to look up the information in the table. We use this information along with the placement length and the sink capacitance to compute the estimated delay. Our Scaled SRCS technique uses the same input information as the SRCS technique to obtain the scaling factors. Scaled SRCS then uses a different formula to compute the estimated delay. Scaled SRCS offers an advantage when a variant or new design contains many nets with characteristics that do not match the entries in the constructed lookup tables.

4. EXPERIMENTAL RESULTS

We verify the accuracy of our delay estimators within an industrial design flow. We use Cadence First Encounter v4.1 for placement, routing, and RC extraction. We also utilize an industrial 90 nm technology library together with four benchmark circuits. The benchmark statistics are summarized in Table 2. We first place the circuits and then predict the routing delay. The predicted delay is compared against the reference routing delay and the error in delay estimation is reported. The estimation error is defined as absolute difference between the estimated delay and the actual routing delay normalized by the routing delay as given by the following equation:

$$\text{estimation error} = \left| \frac{\text{estimated delay from placement} - \text{routing delay}}{\text{routing delay}} \right|$$

We propose two methods to test the validity of the proposed estimators. In the first method, we create *variants* of the base circuit and test the accuracy of the delay models calculated from the given

Circuit	# Nets	# Cells	core area (um^2)
A (des)	27478	27104	285861
B (aes_cipher)	15880	15265	160769
C (s38417)	8558	8529	69713
D (s13207)	2302	2240	21895

Table 2: Relevant statistics of used benchmark circuits.

circuit in predicting the delay of the variant circuits. Thus, the first method can be considered as a way to measure the *stability* of the proposed estimators. In the second method, we use the estimator models calculated from the base circuit to predict the delay of completely different circuits. Thus, the second method is used to measure the *universality* or the generality of our estimators.

4.1 Estimator Stability

To test the stability of the proposed estimation methods, we synthesize design *variants*. The variants are derived by placing blockage areas in the base design. Blocking forces route and/or layer changes. The new routes lead to consumption of metal resources elsewhere which impacts other nets, leading to a "ripple" effect impacting the delays of a large number of nets on different layers. The impact of various blocking geometries on changing the wirelength has been previously evaluated [18].

Our blockages are 4*um* wide and range in length on a designated metal layer. Carefully selecting the metal layer and size of the blockage controls the extent of variant. After creating the blockage, we re-route the design and calculate the percentage of nets that have a change in delay compared to the original design. We create two design variants: one where 10% of the nets have different delays, and one where 40% of the nets have different delays. For example, Figure 2 compares the pin-to-pin delay (in *ns*) of the original design (x-axis) and the 40% variant design on the y-axis for all benchmark circuits. The figure shows that our blockage insertion yields substantial changes into the pin-to-pin delays. We use the estimation model from each of the benchmark circuits, compare its estimation against the reference design, the 10% design variant, and the 40% design variant. We tabulate the results of our delay estimations of the original and variant circuits in Table 3 and summarize the averages in Table 4.

Among the four techniques, the RC sampling techniques (SRCS and Scaled SRCS) predict wire delays more accurately then the delay-based methods (DS and SDS). All sampling techniques experience accuracy degradation as the extent of variant increases. As congestion increases, due to the introduction of blockages, more changes in layer assignment and topology occur, which deteriorates the results of those techniques. In general RC sampling provides better results than delay sampling because congestion does not alter sink capacitance.

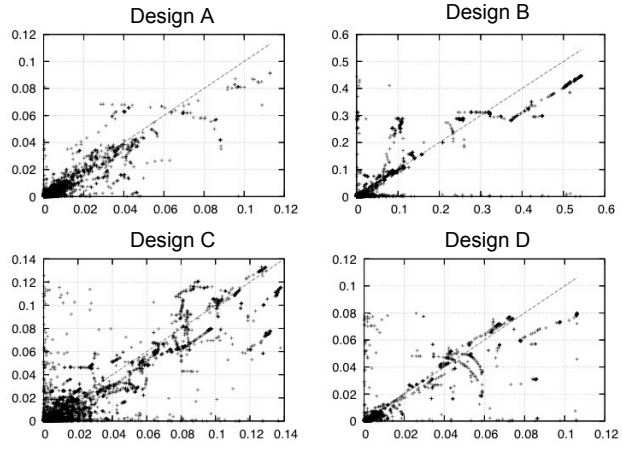

Figure 2: The post-route pin-to-pin delay of the original (x-axis *ns*) vs. the 40% variant (y-axis *ns*) for each benchmark circuit.

Circuit	Estimation Technique	Original	Variants	
			10%	40%
A	DS	46%	47%	47%
	SDS	25%	28%	38%
	SRCS	10%	10%	18%
	Scaled SRCS	13%	13%	18%
B	DS	51%	51%	52%
	SDS	13%	21%	41%
	SRCS	9%	10%	22%
	Scaled SRCS	11%	12%	21%
C	DS	29%	29%	34%
	SDS	9%	11%	35%
	SRCS	4%	13%	25%
	Scaled SRCS	5%	5%	20%
D	DS	34%	35%	40%
	SDS	6%	8%	23%
	SRCS	3%	5%	23%
	Scaled SRCS	5%	5%	18%

Table 3: Compare the stability among techniques.

4.2 Estimator Universality

In the second series of experiments, we validate the universality of our estimator. We use the models computed from one circuit design to estimate the delay of other circuit designs. The estimation error for each of the four benchmark circuits is presented in Table 5 using the circuit itself and each of the other circuits as a reference. For thoroughness, we report the results using every potential reference circuit. Table 6 summarizes the results across all reference circuits for all of our techniques.

Scaled SRCS outperforms other methods with a capability to predict delay with an average estimation error of 16% across all references. The better performance of scaled SRCS is mainly attributed to its use of individual resistive and capacitive scaling factors that are able to do a better job when the estimated designs have nets with different characteristics than those encountered in the base design. On the other hand, delay estimation methods (DS and SDS) that attempt to predict delay directly without estimating the individual resistances and capacitances perform poorly when predicting other designs. Circuits A and B are larger than circuits C and D, and contain a larger variety of net degrees. Circuits A and B are thus better reference circuits than the two smaller circuits C and D.

The accuracy of our sampling techniques means that it is possible to obtain accurate routing delay estimations using a simple and fast method. Our method involves the use of pre-constructed lookup tables together with the layer and topology compensation factors. These numbers are calculated once and used as a "golden" reference to compute the delay of other circuits. Our results show that this approach gives good results. It is also possible to obtain higher estimation accuracy by averaging the results of a number of constructed tables to obtain the golden standard.

Estimation Technique	Original	Variants	
		10%	40%
DS	40.0%	40.5%	43.3%
SDS	13.3%	17.0%	34.3%
SRCS	6.5%	9.5%	22.0%
Scaled SRCS	8.5%	8.8%	19.3%

Table 4: Average stability results.

Ref. Circuit	Estimation Technique	Estimation Error			
		A	B	C	D
A	DS	46%	56%	NA	NA
	SDS	25%	70%	65%	64%
	SRCS	10%	18%	23%	26%
	Scaled SRCS	13%	16%	18%	22%
B	DS	46%	51%	NA	NA
	SDS	80%	13%	56%	59%
	SRCS	26%	9%	15%	20%
	Scaled SRCS	18%	11%	16%	17%
C	DS	136%	157%	29%	NA
	SDS	150%	176%	9%	49%
	SRCS	50%	63%	4%	28%
	Scaled SRCS	20%	20%	5%	22%
D	DS	132%	157%	NA	34%
	SDS	137%	160%	62%	6%
	SRCS	70%	114%	21%	3%
	Scaled SRCS	19%	18%	17%	5%

Table 5: Estimation error reported for circuits A, B, C, and D with each circuit acting as a reference.

Estimation Technique	Reference circuit				Average
	A	B	C	D	
DS	51%	49%	107%	108%	79%
SDS	56%	52%	96%	96%	75%
SRCS	19%	18%	36%	52%	31%
Scaled SRCS	17%	16%	17%	15%	16%

Table 6: Average universality results.

5. CONCLUSION & FUTURE WORK

In this paper we have addressed the problem of accurately estimating routing delay from circuit placement using fast techniques. The main contributor factors to the discrepancy between placement and routing delays are net topology, layer assignment, and congestion. To take these factors into account, we have proposed a number of different techniques. Our techniques construct lookup tables or calculate scaling factors to arrive at the estimated routing delay. Using an industrial flow, we have verified the accuracy of our techniques on a number of circuit designs. Our results show that it is possible to estimate the routing delay with an average estimation error of 16%. Our methods can be incorporated into placers to directly derive their objective to optimize the routing delay or can be used post-placement to better predict circuit timing.

A number of future research directions are outlined:

1. Designers are particularly interested in estimating delays of critical paths. Thus, it is possible to refine our techniques to focus on delay estimation of critical paths.

2. In an industrial setting, our estimation techniques can continuously evolve and improve their accuracy by further populating their lookup tables with encountered routing results.

3. Our approach can utilize the results of a fast global routing stage to further tune its detailed routing estimates.

Acknowledgments: We thank the anonymous reviewers for their comments that helped improve this manuscript.

6. REFERENCES

[1] W. E. Donath, "Placement and Average Interconnection Lengths of Computer and Systems," *IEEE Trans. Circuits and Systems*, vol. 26), pp. 272–277, 1979.

[2] M. Pedram and B. Preas, "Accurate Prediction of Physical Design Characteristics for Random Logic," in *International Conference on Computer Design*, 1989, pp. 100–108.

[3] S. Bodapati and F. N. Najm, "Pre-Layout Estimation of Individual Wire Lengths," in *Workshop on System Level Interconnect Prediction*, 2000, pp. 91–96.

[4] P. Zarkesh-Ha, J. A. Davis, and J. A. Meindl, "Prediction of Net-Length Distribution for Global Interconnects in a Heterogeneous System-On-a-Chip," *IEEE Trans on VLSI Systems*, vol. 8(6), pp. 649–659, 2000.

[5] D. Stroobandt, *A Priori Wire Length Estimates for Digital Design*. Kluwer Academic Publishers, 2001.

[6] J. Dambre, D. Stroobandt, and J. V. Campenhout, "Toward the Accurate Prediction of Placement Wirelength Distributions in VLSI Circuits," *IEEE Transactions on VLSI Systems*, vol. 12(4), pp. 339–348, 2004.

[7] T. Hamada, C.-K. Cheng, and P. M. Chau, "A Wire Length Estimation Technique Utilizing Neighborhood Density Equations," in *Design Automation Conference*, 1992, pp. 57–61.

[8] B. Hu and M. Marek-Sadowska, "Wire length prediction based clustering and its application in placement," in *Design Automation Conference*, Anaheim, CA, USA, 2003, pp. 800 – 5.

[9] J. Cong and S. K. Lim, "Edge separability based circuit clustering with application to circuit partitioning," in *Asia and South Pacific Design Automation Conference*, Yokohama, Japan, 2000, pp. 429 – 34.

[10] A. B. Kahng and S. Reda, "Intrinsic shortest path length: A new, accurate a priori wirelength estimator," in *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, ICCAD*, San Jose, CA, United States, 2005, pp. 173 – 180.

[11] B. Landman and R. Russo, "On a Pin versus Block Relationship for Partitions of Logical Graphs," *IEEE Transactions on Computers*, vol. 20(C), pp. 793–813, 1971.

[12] C. E. Cheng, "RISA: Accurate and Efficient Placement Routability Modeling," in *International Conference on Computer Aided Design*, 1994, pp. 690–695.

[13] W. Gosti, S. P. Khatri, and A. L. Sangiovanni-Vincentelli, "Addressing the Timing Closure Problem by Intgerating Logic Optimization and Placement," in *International Conference on Computer Aided Design*, 2001, pp. 224–231.

[14] A. Kahng and X. Xu, "Accurate Pseudo-Constructive Wirelength and Congestion Estimation," in *Workshop on System Level Interconnect Prediction*, 2003, pp. 61–68.

[15] J. Westra, C. Bartel, and P. Groeneveld, "Probabilistic Congestion Prediction," in *International Symposium on Physical Design*, 2004, pp. 204–209.

[16] P. Kudva, A. Sullivan, and W. Dougherty, "Metrics for Structural Logic Synthesis," in *International Conference on Computer Aided Design*, 2002, pp. 551–556.

[17] A. Abou-Seido, B. Nowak, and C. Chu, "Fitted Elmore Delay: A Simple and Accurate Interconnect Delay Model," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 7, pp. 691 – 696, 2004.

[18] C.-K. Cheng, A. Kahng, L. Bao, and D. Stroobandt, "Toward better wireload models in the presence of obstacles," in *Asia and South Pacific Design Automation Conference*, 2001, pp. 527–32.