A Tale of Two Nets: Studies of Wirelength Progression in Physical Design

Andrew B. Kahng
CSE and ECE Departments
University of California, San Diego
La Jolla, CA 92093
abk@ucsd.edu

Sherief Reda
CSE Department
University of California, San Diego
La Jolla, CA 92093
sreda@cs.ucsd.edu

ABSTRACT

At every stage in physical design, engineers are faced with many different objectives and tools to develop, optimize, and evaluate their design. Each choice of a tool or an objective to optimize can potentially lead to a completely different final physically designed circuit. Furthermore, some of the objectives optimized by the tools are not necessarily the best or right objectives, but rather compromised objectives; for example, placers optimize the half-perimeter wirelength rather than the routed wirelength. The contributions of this paper are twofold. First, we define and use a metric to measure the consistency of optimizing wirelength during the different stages of physical design. Our main technique is based on tracing the relative lengths of two nets - or more accurately pairs of nets - as they progress through the physical design flow. Second, we propose a simple method to quantify the similarity between the results of different tools. Our empirical results point out to the physical design stages where vulnerability can occur from optimizing compromised objectives.

Categories and Subject Descriptors: B.7.2 [Design Aids]: Placement and routing.
General Terms: Measurement, performance, standardization.
Keywords: Consistency, similarity, benchmarking, placer suboptimality, wirelength.

1. INTRODUCTION

The sheer size of today's designs and the large number of optimizations that have to be applied on them have posed great challenges to various physical design algorithms. To reduce this sheer complexity, physical design is usually divided into separate steps where the output of each step is the input to the next step as shown in Figure 1. For example, routing is executed after placement, where ideally, placement and routing should proceed at the same time. For the past and the present, this division is essential because of our inability to technically optimize ideal objectives.

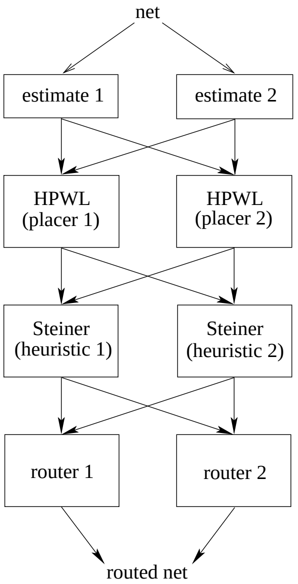

Figure 1: Basic steps on the physical design flow which affect the length of a net.

The division of physical design into a number of steps, together with the inability to find optimal solutions for most steps, have led to the presence of a number of heuristics with suboptimal behavior. Each heuristic optimizes a compromised objective that is a substitute for the ideal objective (routed wirelength for pure wirelength-driven physical design). One question is how reasonable it is to optimize a compromised objective instead of the ideal objective. Another fundamental question is how far the suboptimal solution is from the optimal solution. This question has received recent attention in the placement literature [12, 5, 9, 16, 26]. Another question is how similar the results of various tools are, or even the similarity of the results of the same tool [15, 1, 3]. Answering this question is important because it allows some stability during incremental optimization, as well as tool interoperability.

In this paper, we are going to examine the "life-cycle" of two nets - or more accurately pairs of nets - from the moment

SLIP'06, March 4–5, 2006, Munich, Germany.
Copyright 2006 ACM 1-59593-255-0/06/0003 ...$5.00.

they are “born” in logic synthesis to their last “life stage” in routing. As an interconnect “grows”, we will trace its wirelength which starts as an estimate, then a Half-Perimeter Wirelength (HPWL), then a Steiner length, and finally a routed length. Tracing the “life cycle” of two nets sheds light on two important aspects: (1) the consistency of wirelength optimization, which shows the reasonability of optimizing a compromised object, e.g., HPWL, at one stage versus optimizing Steiner tree length or routed wirelength, and (2) the similarity of physical design results from various possible tools. The life-cycle of a net is illustrated in Figure 1, where at each stage there are many possible optimization objectives as well as tools.

The organization of this paper is as follows. Section 2 gives the motivation for this work. Section 3 gives the results of tracing the relative net length between every two consecutive stages in the physical design flow. Section 4 gives the overall picture by giving “inter-stage” tracing of net length. Finally, Section 5 summarizes the contributions of this paper.

2. MOTIVATION AND METHODOLOGY

In this section we motivate and define the two concepts of *consistency* and *similarity*. The former concept is concerned with the consistency of wirelength optimization in physical design, while the latter is concerned with how similar the results of different tools are, and how similar different results by the same tool are. We also overview our experimental infrastructure.

2.1 Consistency of Optimization

Physical design tools work hard on optimizing wirelength (in this paper we are only concerned with pure wirelength-driven physical design). Generally, tools are not optimizing the best objective, which is the routed wirelength, but rather a “compromised” objective. This motif of optimizing a compromised objective is recurrent at every physical design stage. For example: (1) a logic synthesis tool might be optimizing depending on an estimate to wirelength, (2) a placer might be optimizing HPWL or weighted wirelength as a compromise to routed wirelength, and (3) a Steiner tree algorithm is optimizing the length of each net on its own, but not all routed nets together.

The main underlying assumption behind different physical design optimizations is that optimizing the compromised objective will lead to good final results, or in other words, there is *consistency* in the optimization objectives. The best way to test this assumption is to (1) place a design optimally accordingly to HPWL and then route it, (2) place a design optimally according to routed wirelength, and (3) compare the routed wirelength of step (1) with the wirelength of step (2). This method is certainly the ideal way to test the assumption, but it is not feasible. The HPWL minimization placement problem is NP-hard [22] and inapproximable [20]. Furthermore, attempts to benchmark it either report apparently loose lower bounds [26], or suboptimality bounds for constructed instances [12, 5, 16]. The problems with HPWL benchmarking - let alone optimal routed wirelength benchmarking - are certainly a technical difficulty in quantifying the effect of compromised objectives.

In this paper we try to “indirectly” measure the effect of optimizing compromised objectives. Our main idea is to compare the relative lengths of two nets i and j after two physical design stages, and where i and j have the same number of pins. If the length of j is greater than that of i after some stage k, but the length of j is less than that of i after stage $k+1$, then this indicates an *inconsistency*, where the tool at stage k was not optimizing the relative lengths of i and j in the best way, and that the tool at stage $k+1$ ended up reducing the length of j in comparison to i. This is illustrated in Figures 2 and 3. In Figure 2 we see a consistent case where the relative lengths of i and j stay the same across all physical design stages. In Figure 3 we see an inconsistent case where nets i and j relative lengths change as physical design progresses.

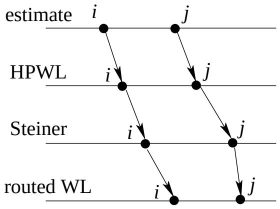

Figure 2: A Consistent situation: The relative lengths are unchanged during physical design. i and j are two nets with the same number of pins.

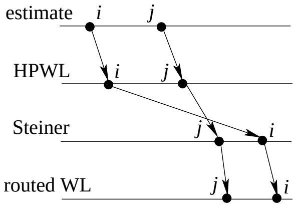

Figure 3: An inconsistent situation: The relative lengths are changed during physical design. i and j are two nets with the same number of pins.

Definition 1. Consistency Test: Let i and j denote two different nets with the same number of pins, a and b denote two physical design stages, and $l_a(i)$ denote the length of net i measured at stage a. We say i and j are *consistent* between a and b if and only if:

- $l_a(i) \geq l_a(j)$ and $l_b(i) \geq l_b(j)$
- or $l_a(i) \leq l_a(j)$ and $l_b(i) < l_b(j)$

A *consistency experiment* consists of testing the consistency of pairs of nets until the average consistency is *stable*, i.e., the fluctuation in the numerical value of consistency is less than 0.01% and after a minimum number of tests are executed. Now it is worthwhile to look at what should be least achievable in a consistency experiment.

Lemma 1. Achieving a consistency of 50% between the results of two stages of physical design is trivial.
Proof: Suppose we have the results from one optimization stage, say placement, on a given netlist. If we pick two nets from the netlist, then a comparison between their lengths could either yield the relationship $\{\geq, \leq\}$. Suppose we construct a completely random placement of the same netlist. Then comparing the length of the same two nets in

the random placement yields either '$\geq'$ or '$\leq'$ with equal probability. Thus we expect a consistency of 50% between the results of actual placement and the results of the random placement. Thus, achieving 50% consistency is a trivial achievement, and any consistency above 50% is desirable.

As an extension to our consistency definition, we calculate how much *tolerance* in the lengths of nets i and j is required to make them consistent. Under the presence of tolerance, $0 \leq tol \leq 1$, two nets i and j are consistent if and only if:

- $l_a(i) \geq l_a(j)$ and ($l_b(i) \geq l_b(j)$ or $\frac{|l_b(i)-l_b(j)|}{max(l_b(i),l_b(j))} < tol$)
- or $l_a(i) \leq l_a(j)$ and ($l_b(i) < l_b(j)$ or $\frac{|l_b(i)-l_b(j)|}{max(l_b(i),l_b(j))} < tol$)

In an ideal scenario, the relative lengths of all nets would stay consistent as the design progress toward routing. This would lead to perfect consistency. However, as mentioned earlier, because of optimizing some compromised objectives, relative lengths change in the physical design flow. We think that the closer a compromised objective is to the actual objective, the better will be our consistency.

2.2 Similarity

Another related issue we treat in this paper is the *similarity* between the outputs of different tools. The subject of placement similarity has been raised recently [15, 1, 3]. We first note that placement is a means towards an end, where the end is determining the length of the interconnects. From this point of view, we consider two placements to be identical only if they yield the same individual interconnect lengths, even if the spatial locations of the objects in the two placements are completely different. Note that such simple notation of similarity takes care of cases where *mirroring* of placement occurs. For example, if a grid placement is rotated 90, 180, or 270 degrees, it will be still similar, under our notion, to the original placement. Our notion of similarity is formalized as follows.

Definition 2. Similarity: The similarity of the length of net i between two different tools p and q is defined as

$$s_{p,q}(i) = \frac{\min(l_p(i), l_q(i))}{\max(l_p(i), l_q(i))} \quad (1)$$

The similarity between two placements (or routings) is the average similarity of all nets. Note that our proposed definition satisfied the three properties proposed by [3]:

- **Reflexive:** $s_{p,p}(i) = 1$.
- **Symmetry:** $s_{p,q}(i) = s_{q,p}(i)$.
- **Triangle Inequality:** for three tools p, q, and t : $s_{p,q}(i) \times s_{q,t}(i) \leq s_{p,t}(i)$. This can be proved in a straight-forward fashion by enumerating all possible scenarios of the relative lengths of $l_p(i), l_q(i)$, and $l_t(i)$.[1]

Having established definitions for consistency and similarity, we examine next the tools and benchmarks to be used in our experiments.

2.3 Infrastructure for Experiments

To carry out our experimental study, the following methods and tools are used.

- Estimates: A number of *a priori* estimators have been recently proposed [13, 10, 19]. We use mutual contraction (MC) [13, 19] and intrinsic shortest path length (ISPL) [18].
- Placers: There have been a recent proliferation in the number of available placers [25, 14, 11, 24, 2, 21, 17]. We choose two placers: Capo9.3 [21] as representative of multi-level min-cut placers, and APlace2.0 [17] as a representative of analytical placers.
- Steiner tree heuristics: FLUTE [7, 8].
- Routers: Cadence Nano Router (version 4.10).

We use the IBMV2 (easy instances) [27] as our benchmarks.

3. CONSISTENCY AND SIMILARITY EXPERIMENTS

In this section we quantify the wirelength consistency between every two consecutive physical design stages. This includes the consistency between estimates and HPWL, between HPWL and Steiner length, and between Steiner length and routed length. We also quantify placer similarity, whether similarity of a placer's own placements, or similarity of placements from different placers.

3.1 A Priori Wirelength Estimates

In this subsection we carry out an experiment to quantify the consistency between *a priori* wirelength estimates and HPWL. We use two *a priori* estimates: mutual contraction [13] and intrinsic shortest path length [18].

Mutual Contraction (MC) [13] is a fast *a priori* predictor. Before calculation, it requires hypergraphs to be transformed to graphs by replacing every hyperedge e_k of cardinality $|e_k|$ with a clique of edges, where each edge has the weight $\frac{2}{|e_k|(|e_k|-1)}$. The mutual contraction of a connection $\{u, v\}$ is defined as

$$MC(u, v) = \frac{w(u, v)}{\sum_{x \in adj(u)} w(u, x)} \times \frac{w(v, u)}{\sum_{x \in adj(v)} w(v, x)}. \quad (2)$$

The mutual contraction of a group of nodes U, that belong to a multi-pin net for example, is defined as

$$MC(U) = \prod_{u \in U} \sum_{v \in U, v \neq u} MC(u, v). \quad (3)$$

Intrinsic Shortest Path Length (ISPL) [18] is a new individual and aggregate wirelength estimator. In ISPL, each hyperedge e_k is first assigned a weight or a length of $\frac{|e_k|}{2}$. The ISPL of a net $\{u, v\}$ is the shortest path length between u and v in $H' = (V, E \backslash \{u, v\})$. That is, the shortest path is computed after temporarily deleting $\{u, v\}$ from H. The ISPL of a hyperedge, or a multi-pin net, is the maximum length of any intrinsic shortest path taken over every pair of nodes in the hyperedge.

[1] Note our different version of the definitions of the three properties in comparison to [3], where the definition of triangle inequality in [3] is incorrect.

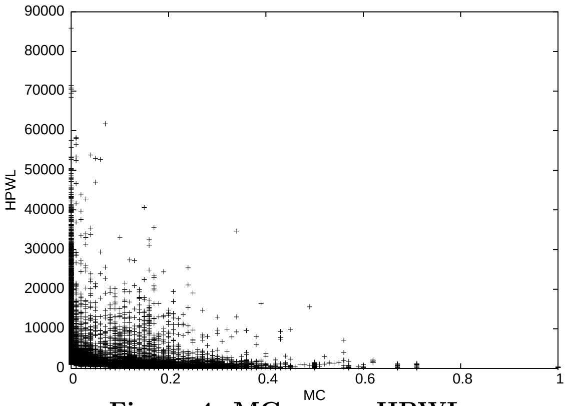

Figure 4: MC versus HPWL.

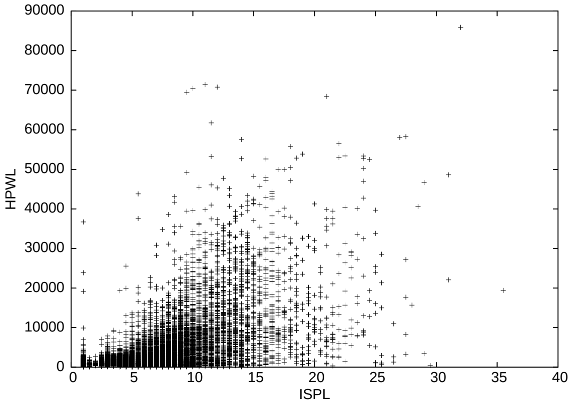

Figure 5: ISPL versus HPWL.

Before we start our experiment, we first plot MC versus HPWL for the ibm01 benchmark in Figure 4 and ISPL versus HPWL in Figure 5. From the figures and by definition, as MC increases, HPWL decreases, while as ISPL increases, HPWL increases. To handle this special relationship between MC and HPWL, we introduce a small modification in our definition of consistency as follows. Nets i and j are consistent between MC and HPWL if and only if:

- $l_a(i) \leq l_a(j)$ and $l_b(i) \geq l_b(j)$
- or $l_a(i) \geq l_a(j)$ and $l_b(i) < l_b(j)$

We give the consistency results between MC and HPWL under various tolerance amounts in Table 1, and the consistency results between ISPL and HPWL in Table 2. All HPWL results are obtained using Capo. Given the trivial performance bound of 50%, we see that both MC and ISPL have poor or average consistency with HPWL. Furthermore, both MC and ISPL seem to get roughly the same consistency numbers with HPWL. We have also tried the same consistency experiment with APlace HPWL, and similar results were attained.

3.2 Half-Perimeter Wire Length

In this subsection we carry out two experiments to (a) quantify the consistency between HPWL and Steiner length, and (b) quantify the similarity of placements produced from different placers, and the similarity of placements produced from the same placer.

circuit	tolerance				
	0	20	40	60	80
ibm01	61.85	68.21	76.81	85.78	93.74
ibm02	61.50	67.12	75.45	83.97	93.03
ibm07	63.68	68.08	74.92	82.60	90.96
ibm08	61.28	65.77	73.59	82.06	90.87
ibm09	64.53	69.31	76.64	84.08	92.08
ibm10	65.05	70.35	77.66	85.59	92.76
ibm11	64.69	69.19	76.01	83.81	91.78
ibm12	52.32	54.44	57.47	61.15	65.59

Table 1: Consistency between MC and HPWL using CAPO placements. All numbers are in percentage.

circuit	tolerance				
	0	20	40	60	80
ibm01	62.67	68.60	77.39	86.73	94.89
ibm02	60.81	66.53	75.06	84.27	93.88
ibm07	60.41	64.82	71.76	80.26	90.23
ibm08	63.82	68.36	76.33	85.17	94.32
ibm09	57.06	61.70	69.51	78.89	90.01
ibm10	64.54	69.55	77.13	85.92	93.90
ibm11	59.38	63.84	70.86	79.66	89.96
ibm12	76.31	78.58	82.23	86.63	91.98

Table 2: Consistency between ISPL and HPWL using CAPO placements. All numbers are in percentage.

Exp 3.2.a Consistency between HPWL and Steiner length: In the first experiment we quantify the consistency between HPWL and Steiner length for Capo placements. The consistency results under various tolerances are given in Table 3. The results show excellent consistency between HPWL and Steiner length. Since the HPWL and Steiner length of all 2-pin and 3-pin nets are essentially identical, we re-execute the consistency experiments with nets of 4 or more pins. The results on Capo's placements are given in Table 4. **Number** gives the percentage of nets with 4 or more pins. **HPWL contribution** gives the HPWL contribution of those nets. **Steiner contribution** gives the Steiner contribution of those nets. It is natural to find that the Steiner contribution is higher than the HPWL contribution. **Consistency** gives the consistency between HPWL and Steiner length for those nets with 4 or more pins under zero tolerance. The excellent consistency between HPWL and Steiner length supports that optimizing HPWL is a quite reasonable placement objective.

Exp 3.2.b Similarity of HPWL results between placers: In this experiment, we report the similarity between (1) the placements of Capo and APlace, (2) different placements of Capo using different random seeds, and (3) different placement of APlace placements using different random seeds. Results in Table 5 show that placements between different placers exhibit little similarity, and that the same can be said for results from the same placer. Figure 6 plots the similarity in the lengths of nets between APlace's placements, and Figure 7 plots the similarity in the lengths of nets between Capo's placements. In the plot, the x-axis gives the

circuit	tolerance				
	0	20	40	60	80
ibm01	99.81	100.00	100.00	100.00	100.00
ibm02	99.72	99.99	100.00	100.00	100.00
ibm07	99.91	100.00	100.00	100.00	100.00
ibm08	99.89	100.00	100.00	100.00	100.00
ibm09	99.90	100.00	100.00	100.00	100.00
ibm10	99.82	99.99	100.00	100.00	100.00
ibm11	99.86	99.99	100.00	100.00	100.00
ibm12	70.21	76.04	82.29	88.88	95.06

Table 3: Consistency between HPWL and Steiner tree length using Capo placements. All numbers are in percentage.

bench	Nets with more than 3 pins			
	Number	HPWL contribution	Steiner contribution	Consistency
ibm01	31.44	70.60	74.64	96.24
ibm02	37.09	80.83	83.80	96.83
ibm07	27.91	69.22	72.51	97.60
ibm08	30.16	74.46	78.65	97.32
ibm09	27.37	70.26	74.00	96.60
ibm10	37.10	72.96	76.02	97.35
ibm11	28.54	63.99	67.23	97.27
ibm12	35.44	72.73	74.84	75.35

Table 4: Contribution of nets with cardinality greater than or equal to 4. All numbers are in percentage.

length of a net in the first placement, and the y-axis gives the length of a net in the second placement.

While min-cut placers (esp. Capo [1]) have been cited before for poor similarity, it also seems some analytical placers, e.g., APlace, also suffer from this as well. This is primarily because APlace establishes an initial random placement to initialize gradients for cell movement [14]. However from the table of results (and visually from the plots), APlace's own placements are more similar than Capo's own placements.

3.3 Steiner Tree Length

In this subsection we carry out two experiments to (1) quantify the consistency between Steiner length and routing length, and (2) determine how accurate are weighted wirelength optimization objectives.

circuit	APlace vs. Capo	Capo vs. Capo	APlace vs. APlace
ibm01	64.92	64.61	71.17
ibm02	63.06	65.10	79.34
ibm07	60.28	59.79	68.95
ibm08	62.38	61.93	65.31
ibm09	58.74	61.54	62.46
ibm10	63.27	64.49	67.56
ibm11	59.90	60.95	64.83
ibm12	61.26	61.25	69.83

Table 5: Similarity in Capo and APlace placements. All numbers in percentages.

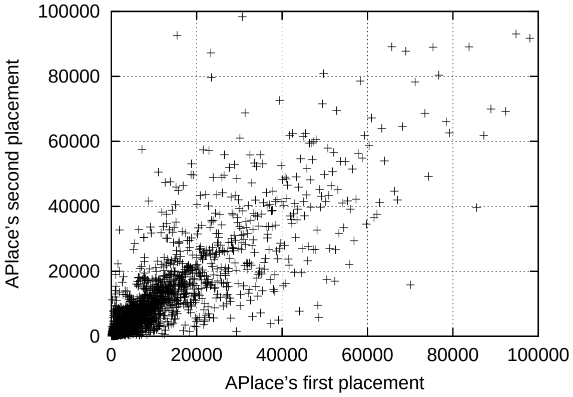

Figure 6: APlace vs APlace.

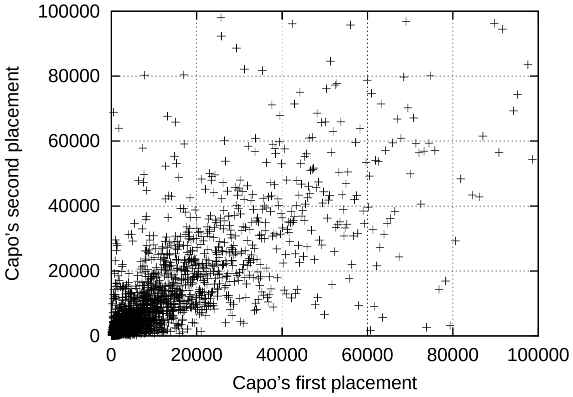

Figure 7: Capo vs Capo.

Exp 3.3.a Consistency between Steiner length and routing: In this experiment we study the change in relative lengths from Steiner trees to routing. We first construct the Steiner trees of all nets using FLUTE, and then route the design using Cadence's Nanoroute. We report our consistency results in Table 6. While the consistency results are very good, they are not as excellent as between HPWL and Steiner length. Figure 8 plots the consistency between Steiner length and routing under various tolerances for the ibm01 benchmark. Consistency increases proportionally as the tolerance increases toward 100%, indicating that large deviations continue to exist between the Steiner length and routed length for some pairs of nets.

Some placers try to minimize the Steiner tree length approximately by using a weighted wirelength (WWL) objective [6, 4]. In WWL, each net length is multiplied by a weight that depends on its degree and perimeter. In the following experiment, we quantify how accurate it is to treat nets with the same length and degree in the same manner.

Exp 3.3.b – Accuracy of WWL lookup tables: We pick two placed nets i and j with the same HPWL and degree, and then use FLUTE to construct the Steiner tree for both nets. We then examine the difference in Steiner tree length between the two nets. Note that under any WWL lookup table, both i and j will have the same WWL. We define the *skew* of two nets i and j as the ratio between their Steiner

circuit	Consistency tolerance				
	0	20	40	60	80
ibm01	77.73	84.32	89.70	93.86	98.58
ibm02	86.35	91.14	95.03	97.76	99.37
ibm07	84.52	89.48	93.93	97.50	99.63
ibm08	84.69	89.76	94.07	97.15	98.99
ibm09	86.01	90.90	94.92	97.99	99.68
ibm10	84.36	89.30	93.45	96.84	99.26
ibm11	85.75	90.61	94.74	97.82	99.58
ibm12	86.35	90.63	94.24	97.24	99.44

Table 6: Consistency between Steiner tree lengths and final routing using APlace placements. All numbers are in percentage.

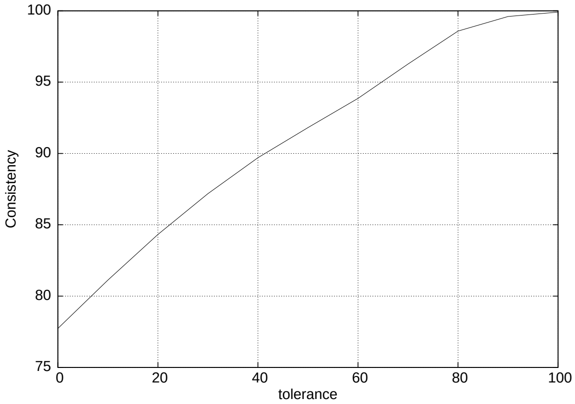

Figure 8: Consistency between Steiner and routed wirelength under variable tolerance for the ibm01 benchmark.

tree lengths (we choose the larger value as the numerator and the smaller value as the denominator). We plot the skew results of ibm01 in Figure 9, where the x-axis gives the skew value, and the y-axis gives the fraction of nets with skew less than or equal that of the x-axis value. The plot shows a serious skew in the Steiner tree length for nets that started with the same HPWL.

3.4 Routers

Routers usually operate in two separate phases: global and detailed. One possible experiment is to measure the consistency between global route lengths and detailed route length; however, commercial tools do not output their global routing results, and thus there is no way for us to measure the consistency in this case.

Regarding similarity experiments, there are two possibilities: (1) measure the similarity of two routings using two different routers but with the same placement, and (2) measure the similarity of two routings using the same router but using different placements of the same design. Since we have only one commercial router, we will not be able to execute possibility (1), but we will be able to execute possibility (2).

Exp 3.4.a Similarity of routings from different placers: We route both the placement of APlace and Capo, and measure the similarity in the net length across the two routings. We report the results in Table 7. We notice that there is little similarity between the two routings; however, our results are no surprise given the little similarity between the initial placements in Table 5.

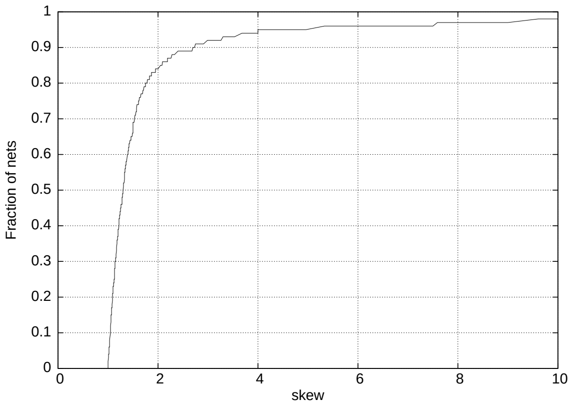

Figure 9: The skew between nets with the same pins and same HPWL and their Steiner length.

circuit	similarity
ibm01	55.54
ibm02	54.41
ibm07	54.00
ibm08	54.90
ibm09	53.21
ibm10	56.03
ibm11	53.97
ibm12	55.72

Table 7: Similarity of routing results between Capo and APlace. All numbers are in percentage.

4. THE BIG PICTURE

In the previous section we have quantified the consistency results between every two consecutive stages. We now study consistency in a more "global" way by calculating the consistency between non-consecutive stages. This can give us an insight, for example, if there is any consistency between *a priori* wirelength estimates and the final routing wirelength [23].

Before giving our summarized results, we establish a *grading* system according to the consistency percentage. With the trivial achievable consistency of 50% - as established by Lemma 1 - in mind, we give a descriptive grading system in Table 10.

We summarize all our consistency experiments under zero tolerance results for Capo in Table 8 and for APlace in Table 9. We also summarize the results descriptively in Table 11. From our results, we come up with the following remarks:

- Individual *a priori* placement estimates have average consistency with HPWL, but in comparison with routed wirelength, they are of essentially little use. However, aggregating individual estimates can smooth out vari-

circuit	ISPL estimate vs.			HPWL vs.		Steiner vs.
	HPWL	Steiner	routed	Steiner	routed	routed
ibm01	61.36	58.63	58.63	99.81	77.89	77.72
ibm02	60.01	56.67	56.67	99.72	82.25	82.35
ibm07	59.60	57.40	57.40	99.91	83.84	84.08
ibm08	62.50	59.75	59.75	99.89	82.82	83.12
ibm09	55.73	53.09	53.09	99.90	81.43	81.35
ibm10	63.61	60.88	60.88	99.82	81.66	81.78
ibm11	58.41	56.03	56.03	99.86	84.76	84.86
ibm12	76.49	34.76	34.76	70.21	61.56	84.55

Table 8: Summary of consistency at zero tolerance for Capo placements.

circuit	ISPL estimate vs.			HPWL vs.		Steiner vs.
	HPWL	Steiner	routed	Steiner	routed	routed
ibm01	60.48	57.92	57.92	99.78	77.87	78.17
ibm02	60.14	56.79	56.79	99.74	86.40	86.50
ibm07	58.77	56.54	56.54	99.92	84.38	84.69
ibm08	62.03	59.25	59.25	99.88	84.50	84.69
ibm09	55.57	52.89	52.89	99.94	85.84	86.24
ibm10	63.11	60.33	60.33	99.83	84.35	84.57
ibm11	58.98	56.60	56.60	99.87	85.37	85.60
ibm12	77.36	50.53	50.53	99.87	86.32	86.48

Table 9: Summary of consistency results at zero tolerance for APlace placements.

Consistency	Grade
90%-100%	Excellent
80%-90%	Very good
70%-80%	Good
60%-70%	Average
50%-60%	Poor

Table 10: A grade system for consistency and similarity experiments.

	HPWL	Steiner	routed
estimate	poor/average	poor	poor
HPWL	-	excellent	good/very good
Steiner	-	-	very good

Table 11: Grade assignments for consistency between wirelength metrics at different stages of physical design.

ations and yield reasonable total or average wirelength estimates [18].

- HPWL has excellent consistency with Steiner length, supporting the common wisdom that optimizing HPWL (or WWL) during placement is a good compromise. However, HPWL does not exhibit the same excellent consistency with routed wirelength.

- In contrast to the excellent consistency between HPWL and Steiner length, consistency between Steiner length and routing is far from excellent.

- Our consistency results seem to hold regardless of the placement methodology used (whether Capo or APlace).

- There is little similarity between placements of different placers, or even different placements of the same placer.

5. CONCLUSIONS

In this paper we have studied two topics: consistency of optimizations in physical design, and similarity of physical design results. We have motivated and defined the concept of consistency and quantified it in both local and global senses. Our method is based on comparing the lengths of net pairs having equal numbers of pins. By tracing the relative lengths through the physical design, we have shed light on which optimizations stay relevant as physical design progresses. We have also proposed a simple idea for quantifying the similarity between the results of physical design tools. Our results indicate that (i) *a priori* estimates are poor estimator of final routing wirelength, (ii) HPWL is an excellent compromise for Steiner length, (iii) routing wirelength has discrepancy compared to Steiner length, and (iv) placers have little similarity in comparison to each other, or in comparison to their own placements.

From this work, we think that the main areas that need improvement and more research are (i) individual *a priori* wirelength estimators, and (ii) better integration between placement and routing.

6. REFERENCES

[1] S. Adya, I. Markov, and P. Villarrubia, "On Whitespace and Stability in Mixed-Size Placement," in *Proc. IEEE International Conference on Computer Aided Design*, 2003, pp. 311–318.

[2] A. Agnihotri, S. Ono, and P. Madden, "Recursive Bisection Placement: Feng Shui 5.0 Implementation Details," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2005, pp. 230–232.

[3] C. J. Alpert, G.-J. Nam, P. Villarubia, and M. C. Yildiz, "Placement Stability Metrics," in *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2005, pp. 1144–1147.

[4] A. E. Caldwell, A. B. Kahng, S. Mantik, I. L. Markov, and A. Zelikovsky, "On Wirelength Estimation for Row-Based Placement," in *Proc. ACM/IEEE International Symposium on Physical Design*, 1998, pp. 4–11.

[5] C. Chang, J. Cong, and M. Xie, "Optimality and Scalability Study of Existing Placement Algorithms," in *Proc. IEEE Asia and South Pacific Design Automation Conference*, 2003, pp. 621–627.

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

[7] C. Chu, "FLUTE: Fast Lookup Table Based Wirelength Estimation Technique," in *Proc. IEEE International Conference on Computer Aided Design*, 2004, pp. 696–701.

[8] C. Chu and Y.-C. Wong, "Fast and Accurate Rectilinear Steiner Minimal Tree Algorithm for VLSI Design," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2005, pp. 28–35.

[9] J. Cong, T. Kong, J. R. Shinnerl, M. Xie, and X. Yuan, "Large-Scale Circuit Placement: Gap and Promise," in *Proc. IEEE International Conference on Computer Aided Design*, 2003, p. to appear.

[10] J. Cong and S. K. Lim, "Edge Separability-Based Circuit Clustering With Application to Multilevel Circuit Partitioning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23(3), pp. 346–357, 2004.

[11] J. Cong, J. R. Shinnerl, M. Xie, T. Kong, and X. Yuan, "Large-Scale Circuit Placement," *ACM Transactions on Design Automation of Electronic Systems*, vol. 10(2), pp. 389–430, 2005.

[12] L. W. Hagen, D. J. H. Huang, and A. B. Kahng, "Quantified Suboptimality of VLSI Layout Heuristics," in *Proc. ACM/IEEE Design Automation Conference*, 1995, pp. 216–221.

[13] B. Hu and M. Marek-Sadowska, "Wire Length Prediction based Clustering and Its Application in Placement," in *Proc. ACM/IEEE Design Automation Conference*, 2003, pp. 800–805.

[14] A. Kahng and Q. Wang, "Implementation and Extensibility of an Analytical Placer," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2004, pp. 18–25.

[15] A. B. Kahng and S. Mantik, "Measurement of Inherent Noise in EDA Tools," in *International Symposium on Quality in Electronic Design*, 2002, pp. 206–211.

[16] A. B. Kahng and S. Reda, "Evaluation of Placer Suboptimality Via Zero-Change Netlist Transformations," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2005, pp. 208–215.

[17] A. B. Kahng, S. Reda, and Q. Wang, "Architecture and Details of a High Quality, Large-Scale Analytical Placer," in *Proc. IEEE International Conference on Computer Aided Design*, 2005, pp. 891–898.

[18] A. B. Kahng and S.Reda, "Intrinsic Shortest Path Length: A New, Accurate A Priori Wirelength Estimator," in *Proc. IEEE International Conference on Computer Aided Design*, 2005, pp. 173–180.

[19] Q. Liu and M. Marek-Sadowska, "Pre-Layout Wire Length and Congestion Estimation," in *Proc. ACM/IEEE Design Automation Conference*, 2004, pp. 582–587.

[20] M. Queyranne, "Performance Ratio of Polynomial Heuristics for Triangle Inequality Quadratic Assignment Problem," *Operations Research Letters*, vol. 4, p. 1986, 231-342.

[21] J. Roy *et al.*, "Capo: Robust and Scalable Open-Source Min-Cut Floorplacer," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2005, pp. 224–226.

[22] S. Sahni and T. Gonzalez, "P-Complete approximation problems," *Journal of the ACM*, vol. 23, pp. 555–565, 1976.

[23] L. Scheffer and E. Nequist, "Why Interconnect Prediction Doesnot Work," in *Workshop on System Level Interconnect Prediction*, 2000, pp. 139–144.

[24] T. Taghavi, X. Yang, B. K. Choi, M. Wang, and M. Sarrafzadeh, "DRAGON2005: Large-Scale Mixed-Size Placement Tool," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2001, pp. 245–247.

[25] N. Viswanathan and C. Chu, "FastPlace: Efficient Analytical Placement Using Cell Shifting, Iterative Local Refinement and a Hybrid Net Model," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2004, pp. 26–33.

[26] Q. Wang, D. Jariwala, and J. Lillis, "A Study of Tighter Lower Bounds in LP Relaxation Based Placement," in *Proc. IEEE Great Lakes Symposium on VLSI*, 2005, pp. 498–502.

[27] X. Yan, B. K. Choi, and M. Sarrafzadeh, "Routabtility Driven White Space Allocation for Fixed-Die Standard-Cell Placement," in *Proc. ACM/IEEE International Symposium on Physical Design*, 2002, pp. 42–47.