Pre-Layout Estimation of Individual Wire Lengths

Srinivas Bodapati and Farid N. Najm

ECE Dept. and Coordinated Science Lab.
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801, USA

ECE Department
University of Toronto
Toronto, Ontario M5S-3G4, Canada

ABSTRACT – We present a novel technique for estimating individual wire lengths in a given standard-cell-based design during the technology mapping phase of logic synthesis. The proposed method is based on creating a black box model of the place and route tool as a function of a number of parameters which are all available before layout. The place and route tool is characterized, only once, by applying it to a set of typical designs in a certain technology. We also propose a net bounding box estimation technique based on the layout style and net neighborhood analysis. We show that there is inherent variability in wire lengths obtained using commercially available place and route tools - wire length estimation error cannot be any smaller than a lower limit due to this variability. The proposed model works well within these variability limitations.

1. INTRODUCTION

In deep submicron technology, interconnect delay accounts for a significant part of signal delay. There is a need to predict interconnect delay before layout, ideally during logic synthesis. Traditionally, *wire load* models have been used during synthesis (before layout) in order to predict the capacitive load on a net. These models give the load as a function of circuit size and fanout of the net. Wire load models have been found useful for predicting the *average* load for nets with a given fanout, but cannot predict the individual load of a net [1]. Today, accurate interconnect delay is only available after layout and routing. This leads to a situation where the synthesis/layout/routing has to be repeated a number of times before the design meets the timing constraints. Problems arise when this process does not converge [2].

Therefore, a much closer interaction is needed between the synthesis and place and route tool, so that accurate wire length estimates can be provided to the synthesis engine during the technology mapping phase to achieve timing convergence. Heineken et. al [3] have proposed such a technique; they do not report individual wire lengths, instead they provide a method for obtaining wire length distributions. Hamada [4] and Pedram et. al [5] also provide models for wire length estimation. However, they also do not report individual wire lengths, instead they give one (average) wire length for all nets having the same number of pins.

Since wire length is a function of several parameters, including the algorithm used by the final placement and routing tool, we need a wire length estimation techinque that can be adapted to any given placement and routing engine. Therefore, in this paper we present a wire length estimation method based on place and route tool characterization. We use structural characteristics obtained from the standard cell netlist, physical cell characteristics obtained from the standard cell library, and use linear regression to build a wire length model. We have also proposed a method for estimating the bounding box of nets with large fanouts (more than 7), which we use to estimate their wire length. Several methods have been proposed to estimate the bounding box of a net. In [6] the authors have given a brief overview of several such methods, and have proposed a net bounding box estimation based on a Uniform Pin Distribution Model. We have used net neighborhood analysis to estimate the bounding box, and the analysis in [6] to estimate the wire lengths for nets having large fanouts. We also show the inherent noise present in place and route tools (commercial) which makes it impossible to predict wire lengths beyond a certain accuracy.

2. METHODOLOGY

The methodology used to develop the wire length model is as follows:

1. A set of standard cell verilog netlists were selected for analysis. These were placed and routed using Cadence (Silicon Ensemble) and an abstract view of the layout was created. The layouts of these design were analyzed to identify some of the salient features of the standard cell layout (discussed further in section 3), which could affect wire length. A common standard cell library was used for all the designs
2. Wire lengths for individual nets were extracted from the layout using Silicon Ensemble. Other parameters like number of cells and cell types, etc. were extracted from the netlist and standard cell library. The aim was to identify only those parameters which can be obtained without actual place and route and thus can be used to estimate individual wire lengths before layout.
3. The extracted parameters were then classified as *local* and *global*. Global parameters are those parameters which remain constant for a given design, such as the number of cells. Local parameters are those which vary within a given design and are associated with each net in the design such as the number of pins on a net, number of two pin nets in the neighborhood, etc..
4. The global parameters were used to calculate the number of standard cell rows and number of core sites in each standard cell row. These physical parameters were then used in conjunction with the extracted local parameters to define some congestion metrics which quantify the salient features of the layout identified in step 1.
5. The congestion metrics calculated above were then analyzed to determine if some of them could be used as the significant variables of the model, i.e., if any of them correlated well with wire length and if it is possible to express wire length as a function of them.
6. A wire length estimation model was developed using these congestion metrics, calculated for a set of benchmark circuits which can account for wire length variations among nets as well as across designs. This model was then verified by estimating the wire lengths of other designs and comparing the estimates with actual wire lengths obtained by the place and route tool.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SLIP 2000, San Diego, CA.
Copyright 2000 ACM 1-58113-249-2/00/0004...$5.00

3. MODEL PARAMETERS

We have used knowledge of the layout style to define placement *congestion metrics*. These metrics, along with the number of pins on the net, were then considered as the potential variables of the wire length estimation model to be constructed. These congestion metrics are based on local and global parameters, to be defined below.

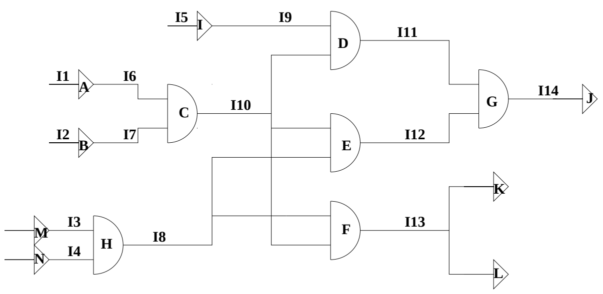


Figure 3.1 Circuit used for defining the various parameters.

3.1 Local Paramters

A key local parameter was identified as:

- *Number of pins on the net* - P_{net}. For example, the number of pins on net I10 in Fig 3.1 is 4, corresponding to cells C, D, E, and F.

Other local parameters were also identified to be useful. They are related to the structure of the *neighborhood* of the net, defined as follows (similar to [3]). The *first level neighborhood* $N_{h1}(i)$ of a net i is defined as the set of all the nets which are connected to cells to which net i is also connected. The *second level neighborhood* $N_{h2}(i)$ of a net i is defined as $N_{h2}(i) = \bigcup_{k \in N_{h1}(i)} N_{h1}(k)$. The *neighborhood* of a net i is then defined as $N_h(i) = N_{h1}(i) \cup N_{h2}(i)$. For example, in Fig 3.1, for net I10, we have $N_{h1}(I10) = \{I6, I7, I9, I11 \dots\}$ and $N_{h2}(I10) = \{I14, I3, I4 \dots\}$. Given this, another local parameter was identified as:

- *Number of two-pin nets in the neighborhood of the given net* - N_{2net}. Since each net corresponds to a cell, this gives a measure of the number of cells having 2-pin nets in the neighborhood of i.

Likewise, we define:

- *Number of three-pin nets in the neighborhood of the given net* - N_{3net}.
- *Number of four-pin nets in the neighborhood of the given net* - N_{4net}.
- *Number of five-pin nets in the neighborhood of the given net* - N_{5net}.
- *Number of six- or more pin nets in the neighborhood of the given net* - N_{6net}.

Finally, we define N_{net} to be the total number of nets in the neighborhood, i.e, $N_{net} = |N_h(i)| = \sum_{k=2}^{6} N_{knet}$

3.2 Global Parameters

The global parameters identified were:

- *Number of Cells in the design* - N_c.
- *Number of two-pin nets in the design* - N_{2agg}. Since each net corresponds to a cell, this gives us the number of cells having two-pin nets at their output.

Likewise, we define:

- *Number of three-pin nets in the design* - N_{3agg}.
- *Number of four-pin nets in the design* - N_{4agg}.
- *Number of five-pin nets in the design* - N_{5agg}.
- *Number of six- or more pin nets in the design* - N_{6agg}.

In every case, this is also the number of cells whose output is tied to a net with so many pins. Other important global parameters include:

- *Expected Row utilization factor* - U. This is a user-specified parameter which specifies how much of a row of standard cells is to be used for cell placement. It determines the size of the design. Higher row utilization factors lead to more compact designs, but routing may be difficult as the number of feedthroughs is reduced at higher U values. Placement of cells may not be possible at higher row utilization factors if the cells are very wide. This parameter was kept relatively constant in our experiments.
- *Aspect ratio of the design* - R. This variable was kept constant for all our experiments and its value was fixed at 1, which is a reasonable assumption. If this value is changed, wire lengths do change. This parameter can be included in our model as it is user-specified and therefore known beforehand.
- *Average Width of the Cells in the design* - W_{avg}. This is given by $\left(\sum_{i=1}^{t} n_i W_i\right)/N_c$ where n_i is the number of cells having width W_i.

There are other parameters (like timing constraints etc.) which are given to place and route engines, but we have developed our model in an unconstrained environment. By this we mean that we have not provided any delay constraints to individual nets in the layout.

4. MODEL DEVELOPMENT

The wire length model development involves the creation of a black box model of the place and route engine as a function of the congestion metrics computed using the global and local parameters. The philosophy behind partitioning the model parameters as global and local is to capture the variation in wire lengths across designs as well as within a design. In our model, each wire length is considered to be function of P_{net} and the congestion metrics calculated for that net, to be defined below.

Our model is motivated by empirical observations, some of which will be presented below. A word is in order about the designs, the library, and the layout style that was used. The designs used for model characterization were obtained after performing placement and routing of the ISCAS and MCNC benchmark circuits shown in Table 4.1. Placement and routing was done using Cadence Silicon Ensemble. A four-metal-layer library having 1.40 micron metal pitch and 102 cells was used. This library contains the abstract cell views of the various cells (in library exchange format called LEF). The minimum width of a core site in this library is 1.4 microns (which is less than the width of a minimum sized inverter available in this library, all cell widths are integral multiples of 1.4). The standard cell height in this library was 12.60 micron. A constant U of 0.85 and $R = 1$ was used for all these designs.

4.1 Base Length

Net length is known to be a strong function of P_{net}. Therefore, we use P_{net} to define a base wire length which along with the congestion metrics form the variables of the model. We define the base wire length as the average of the two lengths obtained by placing all the cells on the net in a single row adjacent to each other or vertically on top of each other in single column. Thus, the base length, which is denoted by L_{nbase}, is given by:

$$L_{nbase} = \frac{1}{2}\left(P_{net} H_{cell} + \frac{P_{net} W_{avg}}{U}\right)$$

where H_{cell} is the height of a cell. Wire length exhibits a strong dependence on L_{nbase}. It was also observed that $2 * L_{nbase}$ comes very close to the actual wire length for nets with 4 to 7 pins, but not for nets with more than 7 pins. In fact, nets with more than 7 pins require special treatment. We refer to them as long nets, and we will introduce below (section 5) a bounding-box based scheme for predicting their length.

Table 4.1. ISCAS and MCNC benchmark circuits used for model characterization

Ckt	#I	#O	#Comp	#Nets	W_{avg}	N_{rows}
random8	8	1	158	280	3.65	8
s1494	14	21	674	690	3.68	18
s510	25	13	248	275	3.66	11
s832	23	18	302	327	3.62	12
c1355	41	32	434	477	4.39	16
s1196	31	24	559	592	3.66	16
c6288	32	32	2274	2309	3.82	34
c1908	33	25	411	446	4.84	16
s820o	23	15	181	206	4.45	10
s641o	52	28	116	170	4.02	7
s298o	17	12	71	90	4.35	6

4.2 Congestion Metrics

The congestion metrics used in our model were derived after analyzing several layouts. It was observed that the majority of the cells connected by two- or three-pin nets were placed close to each other and that the most common configurations were as shown in Fig 4.1a and Fig 4.1b. Consider a net i with $N > 3$ pins, and having $N_{2net} = k$ and $N_{3net} = l$. Consider the k 2-pin nets in the neighborhood. In Cadence, these nets are laid out first, presumably because there are so many of them and they can each be made very short (as in Fig. 4.1a) so that this leads to significant reduction in total line length. If there is a large number of possible locations for placing the cells connected by these 2-pin nets (in the configurations of Fig. 4.1a), then they may end up being spread out over a large area of the layout. This causes the N-pin net to be also spread out over a large area, so that it becomes a long net. This behavior has been observed in practice, and motivates our definitions of the congestion metrics given below.

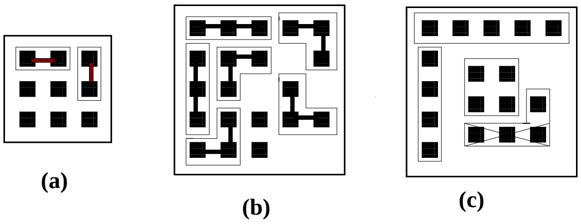


Figure 4.1 Possible placement configurations for 2, 3 and 4 pin nets.

We start with a congestion metric that is related to 2-pin nets, which we call *2-pin congestion*, denoted by P_{2con}, and which is defined as follows. Consider that the sum total of the lengths of all the rows in the standard-cell layout can be computed in two ways. One way is the obvious $N_c W_{avg}/U$. Another way is $N_{rows}(RN_{rows}H_{cell})$, where N_{rows} is the number of standard cell rows in the design. The second expression is true because R is given by row length divided by $N_{rows}H_{cell}$. Equating these two expressions yields $N_{rows} = \sqrt{\frac{N_c W_{avg}}{H_{cell}UR}}$. The number of core sites would, therefore, be $N_{core} = N_{rows}H_{cell}R/W_{core}$. As mentioned previously, based on empirical observations, we will assume that 2-pin nets can only be placed in either of the two configurations shown in Fig 4.1a. If only the horizontal configuration is used, then one can show that the number of possible ways to lay out a 2-pin net is given by $P_{2conb} = \left(\frac{N_{core}}{W_{avg}} - 1\right) UN_{rows}$. If only the vertical configuration is used, then the number of ways to layout a 2-pin net is given by $P_{2cona} = (N_{rows} - 1)UN_{core}/W_{avg}$. To factor in both possibilities, we simply take the sum $(P_{2cona} + P_{2conb})$. Since the design has a total of N_{2agg} 2-pin nets, the number of possible ways to layout each of them is given by $(P_{2cona} + P_{2conb})/N_{2agg}$ (this is an approximation, it assumes no obstructions on the layout surface). This, in a sense, is a measure of the degrees of freedom enjoyed by each 2-pin net. For a net i with N_{2net} 2-pin nets in the neighborhood, the number of possible locations in which the cells tied to these 2-pin nets can be placed (in the configurations of Fig. 4.1a), is a measure of how spread out these 2-pin nets (and therefore the neighborhood cells themselves) may be on the layout surface. Based on this, we define our first congestion metric, related to 2-pin nets in the neighborhood, as $P_{2con} = (P_{2cona} + P_{2conb})N_{2net}/N_{2agg}$. We will refer to this as the *2-pin congestion*.

In a similar way, for three-pin nets we define *3-pin congestion*, denoted by P_{3con}. Skipping the details, this leads to the following:

$$P_{3con} = \frac{(P_{3cona} + P_{3conb} + P_{3conc})(N_{3net} + N_{2net})}{N_{2agg} + N_{3agg}}$$

where:

$$P_{3cona} = \left(\frac{N_{core}}{W_{avg}} - 2\right) UN_{rows}$$

$$P_{3conb} = \frac{(N_{rows} - 2)UN_{core}}{W_{avg}}$$

$$P_{3conc} = 4\left(\frac{N_{core}}{W_{avg}} - 1\right) U(N_{rows} - 1)$$

These computations take into account (approximately) the presence of 2-pin nets.

Similarly we define *4-pin congestion* P_{4con}, *5-pin congestion* P_{5con} and *6-pin congestion* P_{6con}. In *4-pin congestion* we consider 4-pin nets in the neighborhood. In *5-pin congestion* we consider 5-pin nets, but in *6-pin congestion* we consider nets with 6 or more pins. Of these metrics, it was found that *4-pin congestion* is dominant (in terms of how it affects wire-length) in comparison to the other two variables, though its influence is less compared to 2- and 3-pin congestion. This is why we simply lumped all nets with six or higher pins into a single measure.

For 4-pin congestion, only the three basic configurations shown in Fig. 4.1c were considered. For 5 and 6 pin congestion metrics, we do not consider any possible placement configurations. These three congestion metrics (again, skipping the details for brevity) are defined as:

$$P_{4con} = \frac{(P_{4cona} + P_{4conb} + P_{4conc})(N_{2net} + N_{3net} + N_{4net})}{(N_{2agg} + N_{3agg} + N_{4agg})}$$

where:

$$P_{4cona} = \frac{(N_{rows} - 3)UN_{core}}{W_{avg}}$$

$$P_{4conb} = \left(\frac{N_{core}}{W_{avg}} - 3\right) UN_{rows}$$

$$P_{4conc} = (N_{rows} - 1)\left(\frac{N_{core}}{W_{avg}} - 1\right) U$$

$$P_{5con} = \left(N_{rows}\frac{N_{core}}{W_{avg}}U - (N_{2agg} + N_{3agg} + N_{4agg})\right)\frac{N_{5net}}{N_{5agg}}$$

$$P_{6con} = \left(N_{rows}\frac{N_{core}}{W_{avg}}U - (N_{2agg} + N_{3agg} + N_{4agg})\right)\frac{N_{6net}}{N_{6agg}}$$

4.3 One More Variable

Finally, one more variable was found to be required in order to reflect the presence of a large number of 2-pin nets in some cases. Cells that are joined by 2-pin nets are placed close together in order to optimize the total wire length, and if there is a large number of them (which is typically the case) they can end up being "in the way" and can cause the layout of a net to result in bigger length. Basically, we need a measure of the number of 2-pin nets that may end up being placed among (i.e. in the same general layout area as) cells that belong to the neighborhood of the net in question. We propose to use the following measure $N_{2oth} = (N_{2agg} - N_{2net}) \frac{N_{net}}{N_c}$. This measure works as follows. Consider that each net corresponds uniquely to a cell in the netlist, the cell that drives it. For a given net, $N_{2agg} - N_{2net}$ gives the total number of cells in the design that drive 2-pin nets that are outside the neighborhood of this net. A certain fraction of this total will end up being placed "in the way" and we estimate this fraction as the ratio of the size of the neighborhood to the size of the whole design. Thus, this gives a measure of how many remaining 2-pin nets or cells (which are not in the neighborhood) may end up being placed among those neighborhood cells.

4.4 The Model

The wire length model is expressed as a function:

$$L_{net} = f\left(L_{nbase}, P_{2con}, P_{3con}, P_{4con}, P_{5con}, P_{6con}, N_{2oth}\right)$$

Since our intention is to develop a model that is closely coupled to a given place and route engine, and at the same time adaptable to different place and route tools, we developed the model for L_{net} by fitting a general polynomial function. It was found that a general second order or cubic polynomial is sufficient. However, in order to reduce the complexity of this 7-variable model, we ignored all cross-product terms except those with L_{nbase}, since L_{nbase} was found to be the most significant variable. For a cubic polynomial, this reduces the number of terms from about 40 to just 20, without significantly impacting the quality of the fit. The coefficients of the polynomial are obtained using least squares fitting, based on the circuits in Table 4.1.

5. NET BOUNDING BOX ESTIMATION

The model presented so far works well for the majority of nets, but it does not apply very well to nets with large fanouts (more than 7). Since nets with fewer pins are routed first, these long nets end up being spread over a much larger area on the layout, than it would be for nets with less fanout. Thus, we make special-case treatment, by first estimating the dimensions of the *bounding box* for each of these nets. In the following, we will present ways of estimating the area of the bounding box, then its width and height, then the net length.

5.1 Area Of The Bounding Box

If N_{box} is the number of cells in the bounding box, then one way of estimating its area is as follows $B_{boxarea} = \frac{N_{box} W_{avg} H_{cell}}{U}$ The cells in the bounding box will certainly include those in the first level neighborhood. In addition, this being a long net, most of the nets in the second level neighborhood will also belong to the bounding box, because they will typically be placed first. Therefore, we will simply consider that the neighborhood (containing a total of N_{net} cells) belongs to the bounding box.

Apart from the cells in the neighborhood, other cells (most importantly, those driving two or three pin nets) could be in the bounding box as well. We will actually focus only on cells having 2 and 3-pins, because these cells are placed first, as mentioned previously, and they end up being "in the way" as mentioned in section 4.3. Thus we will estimate the total number of cells in the bounding box as:

$$N_{box} = N_{net} + N_{2oth} + N_{3oth}$$

where N_{2oth} is same as defined in section 4.3, and N_{3oth} is similarly defined as:

$$N_{3oth} = (N_{3agg} - N_{3net}) \frac{N_{net}}{N_c}$$

5.2 Dimensions Of The Bounding Box

Now we can estimate the dimensions of the bounding box if we take into account a simple observation made while analyzing the results of the P&R tool. It was found that in majority of cases each cell of a large fanout net was placed in a different row. Thus if P_{net} is the number of pins on a net, then the net would span at least P_{net} rows of the design (if $P_{net} \le N_{rows}$), thus we assume this to be the height of the bounding box. We found this to be a good approximation in practice. Thus, if $P_{net} \le N_{rows}$, we estimate the height of the bounding box as $B_{boxht} = P_{net} H_{cell}$. If $P_{net} > N_{rows}$ then the number of standard cell rows limits the height of the bounding box and it becomes $B_{boxht} = N_{rows} H_{cell}$. With this, the width of the box becomes simply $B_{boxwd} = \frac{B_{boxarea}}{B_{boxht}}$.

5.3 Wire Length Estimation

Finally, given the bounding box dimensions, we estimate the net length. In [6], the expected cost (total net length) of a Rectilinear Steiner Minimal Tree (RStMT) routing of a given net is explored, for various scenarios. It is shown that, when P_{net} is large, this length is given by $\sqrt{B_{boxarea} P_{net}}$, provided that the aspect ratio ($R_{box} = \frac{B_{boxwd}}{B_{boxht}}$) is less than 1. They have also shown that when $R_{box} > 1$, then the length deviates substantially from $\sqrt{B_{boxarea} P_{net}}$.

We use this result in our estimation. When $R_{box} \le 1$, we compute the length as $\sqrt{B_{boxarea} P_{net}}$. When $R_{box} > 1$, we will use the half-perimeter length of the bounding box as a measure of length, i.e., $B_{boxwd} + B_{boxht}$. The comparisons based on this approach will be shown in section 7 (Figure 7.7).

6. NOISE IN ACTUAL WIRE LENGTH

The model obtained after characterization on a set of designs was used to estimate the wire lengths for a *different* set of designs. The estimates were found to be good in some cases but also very poor in some other cases. Upon analysis of the Verilog netlists, we found that the naming convention followed for the cell names was different from what was used in the circuits used in the characterization runs. If the names were changed, agreement became very good! Likewise, individual wire lengths were found to depend on other parameters which are not strictly under user control. Some of these parameters were the *names of the cells*, the *order of the cells*, and the *names of the nets*. Changes in the above parameters lead to changes in the individual wire lengths, even though the average wire length remains relatively constant. The most plausible reason for this might be that the data structures for place and route tools are built using the string variables (say cell names,) and, depending on the strings, the organization of data changes under different naming conventions leading to a different placement. This does not represent a problem with the place and route tool, because muliple placement solutions may be equally good, due to the fact that the tool aims to minimize the *total* wire length. We consider such variations in wire length as *noise* because one cannot control this inherent variability of a P&R engine and cannot possibly hope to account for it in the general case. A typical example of this behavior is show in Fig. 6.1, where the noise in the wire lengths in alu2o is shown. In this figure, both axes represent extracted wire lengths from actual layouts done with different cell names.

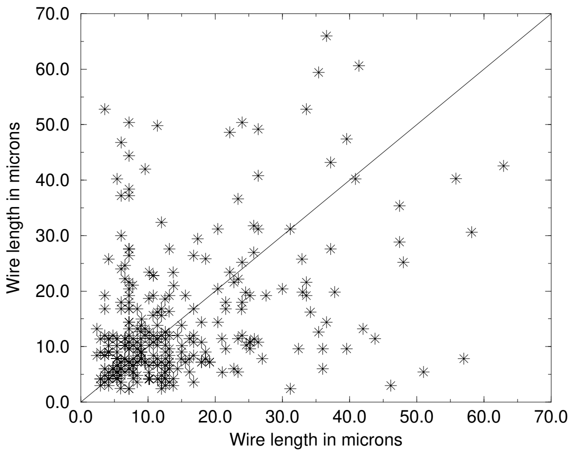


Figure 6.1 Almost random variations in wire lengths less than 70 microns due to changes in cell name

Variations (noise) in wire length due to the above mentioned parameters were very strong for wire lengths less than 70 microns (see Fig 6.1, for alu2o). Hence, we consider that estimation for short wires is pointless. Fortunately, it is more important to estimate wire length accurately on long wires. Thus, we have applied our model to only the set of wires that are longer than 70 microns. Even for wire lengths above 70 micron, a change in cell names does cause some variation in individual wire length, as shown in Figs. 7.1–7.7. Thus, individual wire lengths cannot be estimated beyond a certain accuracy - the *noise floor* depends on the specific place and route tool.

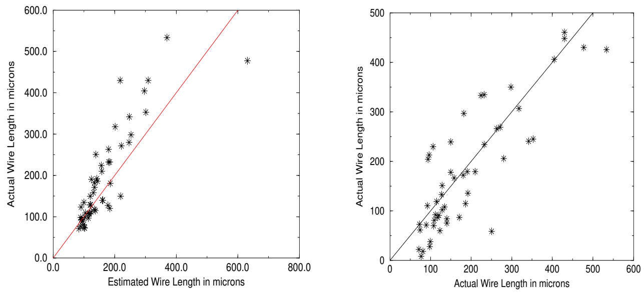
Table 7.1. ISCAS and MCNC benchmark circuits used in Figs.7.1–7.7

Circuit	#I	#O	#Comp	#Nets	W_{avg}	N_{rows}
alu2o	10	4	368	380	3.65	13
s1238o	31	22	331	364	4.55	14
apex6o	135	85	775	912	3.52	19
frg2o	142	109	451	595	4.39	16
x3o	135	89	792	929	3.54	15
random10_325	10	1	487	499	3.73	15
c1355	41	32	434	477	4.39	16
c2670	157	64	425	579	4.41	16
c5315	178	123	1009	1264	4.76	28
s1494	14	21	674	690	3.68	18

7. EXPERIMENTAL RESULTS

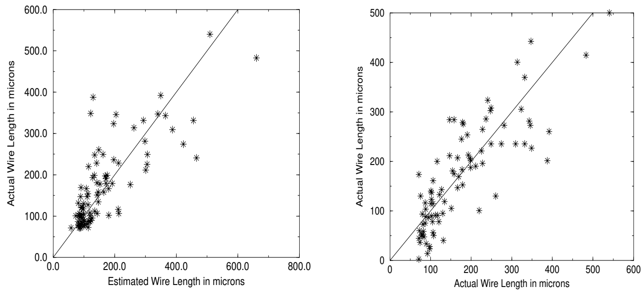
The above modeling technique was implemented in a tool called WLE (wire length estimator). The wire length estimates obtained using our model, for wire lengths above 70 micron, are within the noise limit inherent to the place and route system, as shown in Figs. 7.1–7.7 and Table 7.2. The plots shown in Fig 7.1–7.7 were obtained from ISCAS and MCNC benchmark circuits shown in Table 7.1.

In Fig 7.7 we have shown the wire length estimates for nets having more than 7 pins using the bounding box based wire length estimation technique. All the designs in Table 7.1 were used to get this plot. The library used was the same library described in section 4.



(a) (b)

Figure 7.1 For alu2o, (a) actual v.s. estimated wire length using WLE, and (b) inherent noise in the actual wire length.



(a) (b)

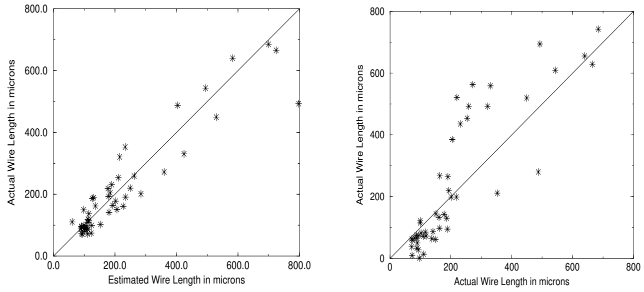
Figure 7.2 For s1238o, (a) actual v.s. estimated wire length using WLE, and (b) Inherent Noise in the actual wire length.

8. CONCLUSION

We have presented a model for estimating individual wire lengths. A significant aspect of this technique is that it can be adapted to a given place and route engine (in this case Silicon Ensemble) by a one-time process of characterization. Since the wire length estimation technique is based on creating a black box model of the place and route engine by characterization, it might be possible to create similar models for other place and route systems. Thus, we have proposed a technique which can be used to provide a close interaction between the synthesis stage and the final place and route which is needed in order to achieve timing convergence. Moreover, we have shown that there is an inherent noise in place and route tools which causes variations in the wire length for the same netlist when some insignificant parameters in the netlist file are changed, which makes it difficult to predict individual wire lengths, most notably for short wires, and to a lesser extent for long wires.

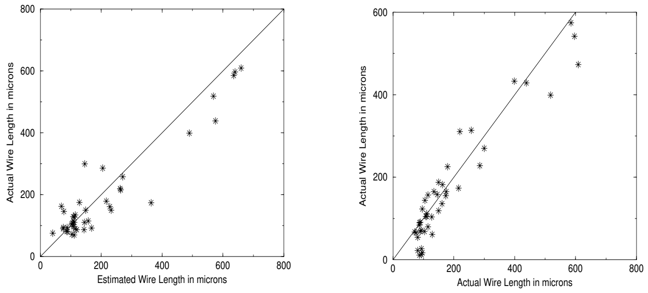
Table 7.2. Average estimation error v.s. average noise in the system

Circuit	Avg Estimation Error	Avg Noise
alu2o	21.75%	33.76%
s1238o	26.46%	31.11%
apex6o	22.15%	42.48%
frg2o	27.05%	25.07%
x3o	22.53%	47.00%
random10_325	26.55%	35.55%
longwires	31.00%	26.92%



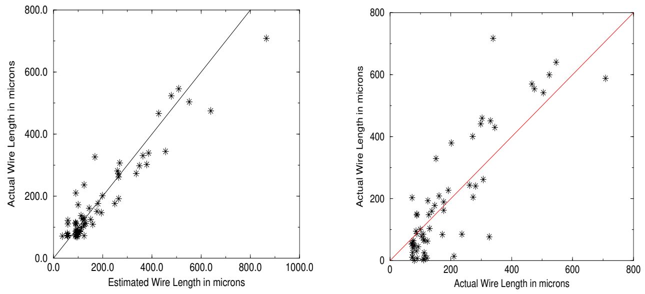
(a) (b)

Figure 7.3 For apex6o, (a) actual v.s. estimated wire length using WLE, and (b) inherent noise in the actual wire length.



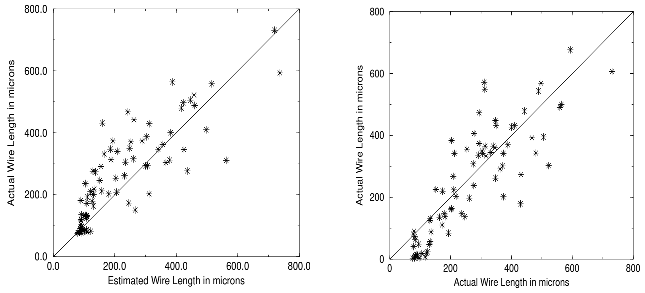
(a) (b)

Figure 7.4 For frg2o, (a) actual v.s. estimated wire length using WLE, and (b) inherent noise in the actual wire length.



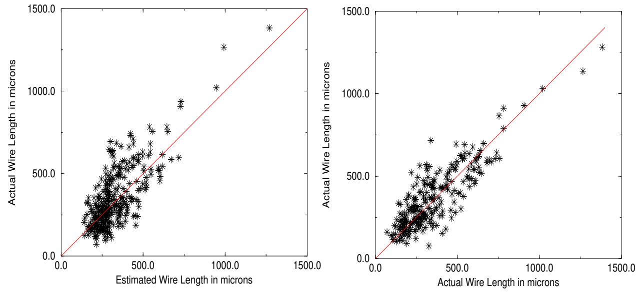
(a) (b)

Figure 7.5 For x3o, (a) actual v.s. estimated wire length using WLE, and inherent noise in the (b) actual wire length.



(a) (b)

Figure 7.6 For random10, (a) actual v.s. estimated wire length using WLE, and (b) inherent noise in the actual wire length.



(a) (b)

Figure 7.7 (a) Actual v.s. estimated wire length for long wires using Bounding Box technique, and (b) inherent Noise in the actual wire length.

9. REFERENCES

[1] H. Kapadia and M. Horowitz, "Using partitioning to help convergence in the standard-cell design automation methodology," *Design Automation Conference*, pp. 592–597, June 1999.

[2] Paul McLellan, "The radical changes wrought by deep submicron IC technology," *EDA Today Summary Report*, vol. 1, no. 6, June 1995.

[3] H. T. Heineken and W. Maly, "Standard cell interconnect length prediction from structural circuit attributes," *Proc. Custom Integrated Circuits Conference*, pp. 167–170, May 1996.

[4] T. Hamada, C.-K. Cheng and P. M. Chau "A wire length estimation technique utilizing neighborhood density equations," *IEEE Transactions on Computer-Aided Design*, vol. 15 pp. 912–922, August 1996.

[5] M. Pedram and B. T. Preas, "Interconnection length estimation for optimized standard cell layouts," *International Conference on Computer Design*, pp. 390–393, October 1989.

[6] A. E. Caldwell, A. B. Kahng, S. Mantik, I. L. Markov and A. Zelikovsky, "On wirelength estimations for row-based placement," *IEEE Transactions on Computer-Aided Design*, pp. 1265–1278, Sep 1999.