Wire-length Prediction using Statistical and Probabilistic Techniques

Jennifer L. Wong[‡], Azadeh Davoodi[†], Vishal Khandelwal[†], Ankur Srivastava[†], Miodrag Potkonjak[‡]
[‡]University of California, Los Angeles, CA 90095,{jwong,miodrag}@cs.ucla.edu
[†]University of Maryland, College Park, MD 20742, {azade,vishalk,ankurs}@glue.umd.edu

UCLA Technical Report # 040027

July 22, 2004

1 Abstract

We address the classic wire-length estimation problem and propose a new statistical wire-length estimation approach that captures the probability distribution function of net lengths after placement and before routing. These types of models are highly instrumental in formalizing a complete and consistent probabilistic approach to design automation and design closure where along with optimizing the pertinent cost function, the associated prediction error is also considered.

The wire-length prediction model was developed using a combination of parametric and non-parametric statistical techniques. The model predicts not only the length of the net using input parameters extracted from the floorplan of a design, but also probability distributions that a net with given characteristics obtained after placement will have a particular length. The model is validated using both learn-and-test and resubstitution techniques.

The model can be used for a variety of purposes, including the generation of a large number of statistically sound and therefore realistic instances of designs. We applied the net models to the probabilistic buffer insertion problem and obtained substantial improvement in net delay after routing ($\sim$40%) when compared to a traditional bounding box-based buffer insertion strategy.

2 Introduction

Wire-length has become one of the most critical metrics in physical design primarily due to the rise of the deep submicron era. Therefore, there is a strong need for early estimation and optimization of this design parameter. A lot of research has been directed towards accurate models for estimation of this important design objective. Accurate timing and routability estimation relies heavily on these models.

Estimating the exact wire-length for each net in the circuit is a very difficult problem. There is a large number of different parameters and constraints, such as the bounding box of the net, number of routing grids and the grid capacity, total number of nets routed in the vicinity of the pertinent net, that are all potentially relevant, but typically are very hard to capture into consistent wire-length model. Hence, estimating an exact value for the wire-length is a very hard problem. Similar difficulty in estimation has also been widely recognized for other critical metrics of deep submicron designs such as power, delay, noise immunity, and crosstalk. Therefore, synthesis optimization is typically performed in the presence of high degrees of estimation inaccuracy. The optimization decisions taken in such a scenario are typically sub-optimal and often result in failure of design closure. In order to solve this problem, a new design automation paradigm is gaining steam in which unpredictable design objectives are modelled probabilistically and the overall design is optimized probabilistically too. For the success of such an approach, we need accurate models which probabilistically estimate the critical design objectives. In order to address this need, we have developed a novel statistical modeling methodology for capturing wire-length in the post placement pre-routing phase.

The model uses data that can be extracted once the placement of the designs is completed. In order to build the model we used a combination of parametric and non-parametric techniques [4, 9]. Since the new model development approach is generic and can be applied to other early estimation tasks in synthesis, we provide a detailed description of how the models were derived. Although statistical technique have demonstrated their potential in many fields, they have rarely been used in synthesis and CAD tools. That is surprising when we consider their advantages. They produce models that are both mathematically sound and that extract the maximal possible amount of information from the collected data. Note that non-parametric statistical techniques are applicable on any set of data with no prior assumptions about their distribution. Furthermore, statistical techniques provide a means for evaluation and validation of obtained models as well as techniques and tools for establishing intervals of confidence about the overall model and any of its subparts. The standard and practical references for parametric and non-parametric statistical techniques that explain in detail many of the concepts, techniques, and algorithms used in this paper include [5, 11, 17]. Although our overall statistical modeling approach is new and several steps are unique, other steps are adopted from modern statistical practice. Finally, it is important to emphasize that the developed statistical model is validated both statistically and through a driver application - buffer insertion for clock cycle

optimization.

Statistical estimation and prediction methodology and models can be used in many ways. For example, one can use the prediction information to evaluate the suitability of a particular floorplan for obtaining final routing where nets satisfy a particular user specified condition. For instance, the goal can be to determine which among a number of competing floorplans is mots likely to result in final design with a few long nets or overall small sum of wirelengths. They are also a natural component of the overall probabilistic design automation methodology. One such probabilistic algorithm is [12] which performs buffer insertion assuming wire-lengths which are estimated as distributions. We used our models in the probabilistic buffer insertion approach of [12] and obtained massive improvements in net delay ($\sim$40%) after routing when compared with a traditional bounding box-based traditional bounding box strategies [2, 13].

The rest of paper is organized in the following way. In order to provide global picture and make the paper self-contained we start by summarizing the probabilistic synthesis paradigm. Next, we describe our statistical modeling procedure and present the developed wire-length estimation model. After that the model is evaluated using both learn-and-test and resubstitution validation methodology. Finally, we present the application of our model to the probabilistic buffer insertion task.

3 Probabilistic Synthesis Paradigms

Automation of integrated systems is marred with estimation inaccuracies which occur due to a combination of many factors. Unawareness of exact layout information such as routing, placement, and exact logic structure are among prominent reasons. In addition, recently fabrication uncertainties have also begun to get considerable weight primarily due to increasing complexity and aggressive scaling of the fabrication process. In the light of such unpredictabilities, a traditional deterministic approach towards design automation becomes incapable and obsolete. Basically, a deterministic approach assigns a fixed value to the cost function (like area, delay, power, wirelength) and does not consider the error associated with the estimation of this cost function. Hence, very little can be said about the optimality of the final design especially if the estimation was erroneous. This issues calls for the development of a probabilistic approach towards design optimization. Such an approach models the cost functions as probability distributions and optimizes the design probabilistically, hence maximizing the likelihood of satisfying design constraints. A number of researchers have suggested that the importance of such an approach [1, 19, 3, 10, 16, 6] because estimation inaccuracies (both due to fabrication variability and layout unawareness) are becoming major bottlenecks in design closure. The main advantage of such an approach is faster design closure, better fabrication yield (since fabrication variability would have been accounted for during designing) and improved robustness.

The main prerequisite for the application of a probabilistic synthesis technique which considers uncertainties, is the availability of accurate prediction techniques. Currently, these models are build mainly manually using deep insights into design process. However, the non-statistical method are rarely statistically tested for their accuracy. We propose the use of modern statistical techniques not only to automatize the development of models and the selection of the most accurate models, but also to provide sound mathematical estimates of their accuracy.

4 Statistical Modelling for Wire-length Prediction

In this Section, we present a statistical approach for predicting the length of a given net on a given chip that is characterized using a set of features that can be rapidly obtained after floorplanning. We start by identifying the objectives and constraints. After that we discuss a set of net and chip features that are used as predictors to our model. The heart of the Section is the procedure that was used for the development of the model. Additionally, the three phases of the procedure (robust linear regression [4], outlier detection, and establishment of probability distribution) are discussed. We then present a model for mapping between different designs. Finally, the evaluation of the proposed models is conducted using learn-and-test and resubstitution techniques [7, 8, 9].

4.1 Problem Formulation

Our primary objective is to predict the length of each net given a set of features that can be rapidly extracted from the floorplan of a chip. The goal is not only predict the length, but also to quantitatively characterize the probability that the net will have a particular length after routing. Furthermore, the operational constraint is to use only features that can be extracted with low computational effort and statistical techniques that can be rapidly applied. The final major objective is to statistically validate all obtained results and to establish intervals on confidence on all deduced models and their parameters.

4.2 Characterization of Nets and Designs

The starting point for model development was the definition of relevant features of nets that are available after placement. We used two types of features: atomic and composite. Atomic features are ones that are directly extracted from the design. Composite features where created by combining atomic features using simple rules. Most often the composite rules were ratios of two atomic features.

We used a state of the art commercial placement and routing tool (Cadence) to collect data that is used to build our statistical models. We use the post placement information as input parameters for building the model for each net. The objective is to identity metrics that influence the post routing wirelength for each net. The basic intuition lies in the fact that the

net length is inversely proportional to the amount of routing area available and directly proportional to the routing hardness. Furthermore, a net is hard to route if its available routing area is being claimed by other neighboring nets. The goal is to build a statistical model using only a small set of parameters that can be easily and rapidly extracted from the placement. While computation of some features is straightforward, the computation of other parameters requires the use of several basic procedures from computational geometry [14]. For example, procedure Locate-Point-Neighbor (p, S) takes a point p and a set of rectangles S and calculates the subset of rectangles which overlap with this point. Procedure Locate-Rectangle-Neighbor(R, S) takes a rectangle R and a set of rectangles S and calculates the rectangles in S that overlap on R. Note that all used properties can be rapidly computed in low polynomial time. We have considered the following post placement properties of the nets.

- Number of Net Terminals. The higher the number of terminals, most often the harder it is to route the net.
- Bounding Box (BBOX) for net i. The BBOX is easy to compute and provided a lower bound on the real wirelength. However, this property does not capture the number of terminals well. More importantly, the bounding box is a function of only a small set of terminals.
- Minimal Spanning Tree (MST). MST is calculated using standard Kruskal's or Prim's algorithm. The property captures the best case scenario for routing while taking into account all terminals.
- Convex Hull (CHULL) of net i's terminals. CHULL envelopes the terminals. Many algorithms, including standard Graham's scan, can be used to calculate CHULL of a net. Runtime is $O(nlogn)$ if n is the number of net terminals. CHULL is in a sense a generalization of BBOX. Note that both MST and CHULL are often very strongly correlated with BBOX.
- Number of different terminals in the bounding rectangle of the net i. This property aims to predict routing difficulty by analyzing the number of terminals from other nets that compete for the same routing resources - space.
- Space Utilization Factor (SUF) for the net i. The bounding area of a net is divided into rectilinear regions based on the number of overlapping neighbors on each region. SUF is calculated using the following formula

$$SUF(Net_i) = NT_i * \sum_{\forall R_j \in net_i} (\frac{OV_{ij} * A_{ij} * P}{A_i}) \quad (1)$$

$$\text{where } P = \sum_{\forall v} (1 - \frac{WS_k}{A_k}) \quad (2)$$

where
v is neighbors $k \in$ region $j, k \notin i$
R is set of all regions
A_i is Bounding Box Area (BBOX) for Net_i
NT_i is total number of terminals in the bounding box of the net
$A_{ij}$is area for all R_j in Net_i
OV_{ij} is number of nets that overlap in R_j of net i (excluding Net_i)
WS_k is white Space of net k which is one of the nets that fall over R_j

The key intuition behind this metric is the fact that more overlapped regions on a net bounding box increase its routing hardness. Moreover, if these neighbors have smaller white space (which means they are themselves congested) then they will make the pertinent net congested too. This metric is calculated using the following procedure. First, we identify regions on the layout based on overlapping net bounding boxes. This is accomplished using an iterative execution of the procedure Locate-Point-Neighbor(p, S) for all grid points. Therefore, the running time of the procedure is proportional to $Grid_x Grid_y$T(Locate-Point-Neighbor). Note that, if the total number of grids is high, the procedure is relatively slow. In this case, we impose a coarser grid resulting in a faster runtime, however at the loss of accuracy. This procedure can be followed by calculating the parameter P (see the equations above) for all regions and summing them up for the net.

- White Space of Net_i. This is a region on the design defined with respect to the BBOX of net i that does not overlap with the BBOX of other nets. The metrics can be calculated using a strategy similar to the one used for the calculation of the previous property. The intuition is simple and clear: large white space is well correlated with higher chances for efficient routing of the net.
- Resource Competition Metric (RCM) for Net_i. This is a composite property that aims to capture the congestion in regions where net i is most likely to be routed. We consider the set of regions, R, that is created after the bounding box of the net is split by considering overlaps with bounding boxes of other nets. If we denote neighbors as $neigh$ and use notation introduced for calculating SUF, the RCM is calculated using the following formula.

$$\sum_{\forall R_j \in net_i} (\frac{A_{ij}}{Area_i} - \sum_{\forall neigh_k \ of R_j} \frac{A_{ij}}{Area_k})) \quad (3)$$

Recall that the regions can be identified in $Grid_x Grid_y$T(Locate-Point-Neighbor) runtime and the above parameter can be calculated for each region and added up for the net. The key intuition behind this parameter is that if the value $\frac{A_{ij}}{Area_i}$ is high then the net i has a larger share of the region where as if $\frac{A_{ij}}{Area_k}$ for the neighbor k is high then that neighbor has a larger share of the region. Larger the RCM for a net more is its share of the available routing area in the nets bounding rectangle.

- Total number of overlapping neighbors. This property can be calculated using the procedure Locate-Rectangle-Neighbor(R, S) and is trying to estimate the number of nets that compete with a given net for routing resources.

- RCM for overlapping neighbors of Net_i. The property is calculated using the RCM procedure. The intuition is that if neighboring nets are very congested, they will induce higher difficulty of routing the pertinent net i.

- Sum of RCMs for all neighbors of overlapping neighbors. This complex measure enhances the scope of the previous metrics.

- Amount of overlapping area with the net for all neighboring nets. The rationale is that a higher ratio of the overlap area indicates increased hardness to route.

- The number of common terminals of neighboring nets to Net_i. This measure is positively correlated with the difficulty of routing of net i.

- Neighbor utilization factor (NUF) that is defined in the following way

$$\sum_{\forall neigh_k \text{of } NET_i} \frac{\text{common terminals} * \text{common area}}{\text{neighbor area}} \tag{4}$$

- Neighbor hardness factor (NHF) defined in the following way

$$\sum_{\forall neigh_k} \text{common terminals} * \text{common area} * RCM(k) \tag{5}$$

The last two properties aim to quantify the competition of neighboring nets with the net under consideration.

4.3 Overall Flow

In this Subsection, we present the overall flow of our statical modeling procedure. Figure 1 summarizes the flow of the developed statistical modeling technique for prediction of the wire-lengths of the nets. The first step is the identification of relevant net properties. Two types of net properties are employed. The first group consists of properties related to the net itself. The second group consist of metrics that aim at predicting encountered congestion during routing of a given net due the routing requirement of neighboring nets. On all properties we also applied a number of nonlinear transformations (e.g. application of logarithm function) in order to obtain better prediction abilities. Interestingly, while it is often reported in other fields that the use of statistical techniques and nonlinear transformations often greatly enhance accuracy, for our model and our set of properties this was not the case.

The second step was data collection. All designs were routed using the Cadence placement and routing tool. Once the data was available, we started with a randomly selected design and built a number of models. We used only 60% of all nets for this propose and preserved the rest of data for conducting statistical test and validation procedures. It was immediately apparent that each of the following three features (bounding box, minimum spanning tree, and convex hull) predicts the length of a majority of nets remarkably well. Each of them had R^2 value above 0.8 individually. The R^2 value is square of residuals, i.e. difference between the predicted variable and its predicted value using an individual property. The statistical t-test indicates that the probability that this correlation is accidental is less than 10^{-16}.

While independently each of the measures (bounding box, minimum spanning tree, and convex hull) are strong predictors, their combination results in only marginally better prediction. Therefore, we decided to use bounding box as the basis of our model because of its low computational cost. Closer examination of the data, indicated that short nets (ones with a bounding box value less than 6,000) and long nets (bounding boxes larger than 6,000) had very different properties. Most importantly, the first group, short nets, had significantly better statistical fits and essentially no outliers. Therefore, we decided to treat these two sets of nets separately. Statistical t-test indicates that correlation is significantly higher for the separated sets than for the overall set. Once the data was divided into two sets, we conducted a linear regression-based procedure for fitting data for different percentiles. For each percentile (in range 10% to 90%) a separate fit is obtained and validated using the t-test. After that, to further enhance the accuracy of our model, we conduct an outliers detection procedure that identified a small subset of data that required specialized models. For this purpose we have developed a CART model [4]. The next step was to repeat linear regressions on the data after outliers were removed.

The next two steps were dedicated to the development of a PDF and CDF for wire-length prediction and to interchip prediction. The goal of interchip prediction is to use global parameters of the chip in order to predict how features, such as global congestion and the number of nets and terminals, impact to PDF for wire-length distribution. Finally, we conducted extensive model evaluation using learn-and-test and the resubstitution procedure in order to verify that the developed model is sound and no overfitting was done. In the rest of this Section, we elaborate on several key steps of the procedure.

4.4 Outlier detection

Outliers can be defined as nets that are not predicted well using a given set of features without significantly changing the complexity of the model. We detected the outliers using the following procedure. We begin by building our preliminary models. As candidates for outliers, we analyzed all points that differ from their prediction by more than k%. In our experimentation, we set k = 20%. Next, all the outlier candidate points are characterized according to each property. The separation value

```
1. Feature Definition;
2. Feature Extraction;
3. Preliminary Data Exploration;
4. Features Evaluation and Normalization
   and Compound Feature Selection;
5. Net_Characterization {
6.      Nets Categorization;
7.      Preliminary Linear Regression on percentiles;
8.      Outliers Detection;
9.      Outliers Modelling;
10.     Final Linear Regression on percentiles; }
11. CDF and PDF model generation;
12. Chip characterization;
13. Development of Mapping Function to New Designs;
14. Evaluation and Validation;
```

Figure 1: Modeling Approach Overall Flow.

for each property is set in such a way that it maximizes the ratio of outliers versus well predicted nets for the nets above (or below) the separation value. Note, that a linear-time sweep is sufficient to find this separation value.

All properties with their corresponding separation values are used as inputs to the non-parametric classification and regression tree (CART) software [4] to provide compact characterization of all outliers. The CART procedure resulted in the model where all nets are separated in three groups according to the number of terminals. The first group consisted of all nets with two terminals, the second with three, four, and five terminals, and the last group contained all other nets.

The final CART model used the following features: number of terminals, RCM of the net, RCM of overlapping neighbors, total number of overlapping neighbors, and the number of common terminals for a given net. The last four features were normalized against the area of the bounding box in order to achieve better separation. The overall misclassification rate for the detection of outliers was 6.7%. For the outlier nets, we build a separate linear regression fit, that had R^2 = 0.83. The t-test indicates that probability of accidental fit was less than 10^-16, clearly indicating the soundness of the model. It is interesting and important to emphasize that all outliers were corresponding to nets that were longer than standard predictions. This phenomenon can be easily explained by the intrinsic nature of the modeling problem. Relatively short nets for a given size of the bounding box (or MST or CHULL) are those that are routed using interconnect that is close to their theoretically possible minimum when no other nets cause congestion. In all designs for all values of bounding boxes, the number of nets with these properties was relatively large. The very high RCM was the best predictor that net will be routed using significantly higher length, in particular if the number of terminals was high.

4.5 CDF and PDF Generation

The goal of this phase is to find accurate cumulative distribution (CDF) and probability distribution functions (PDF) for the length of a net given the size of a corresponding bounding

Bench	# layers	# nets	Area	Net $\overline{Area}$	Total Term
ibm01	8	11507	5.89E+09	1.95E-06	44266
ibm02	10	18429	7.65E+09	2.41E-06	78171
ibm07	10	44394	1.63E+10	2.73E-06	164369
ibm08	10	47944	1.76E+10	2.73E-06	198180
ibm10	10	64227	2.97E+10	2.16E-06	269000
ibm11	10	67016	2.31E+10	2.90E-06	231819
ibm12	10	67739	3.44E+10	1.97E-06	284398

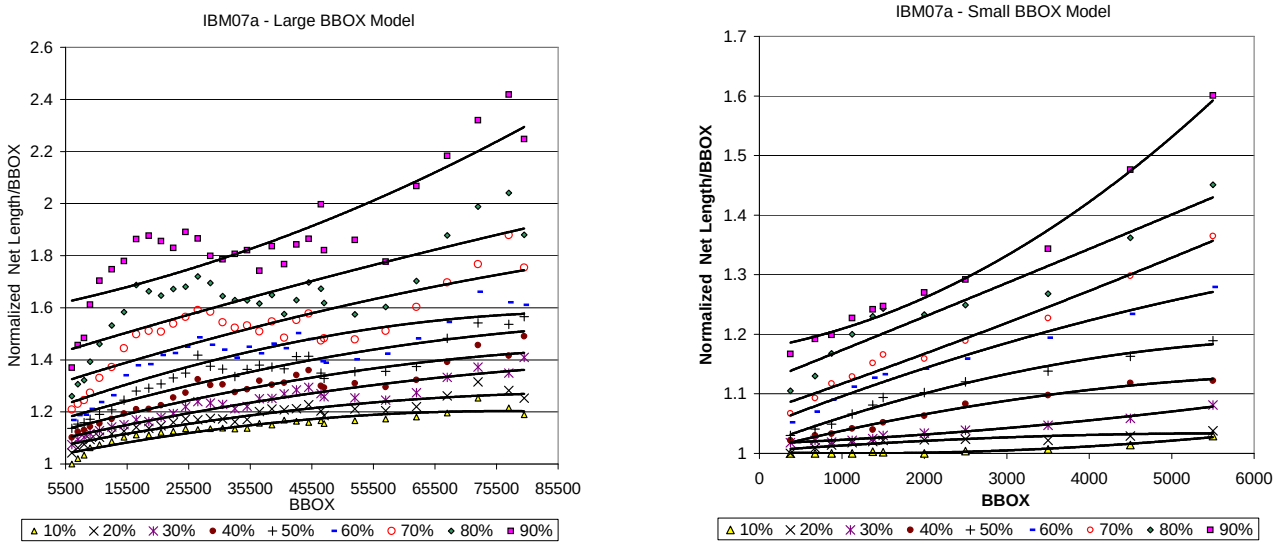
Table 1: Chip level characteristics for ibm designs obtained using Cadence routing and placement tool.

Bench	BBOX	MST	CHULL
ibm01a	9.01E-07	1.09E-06	1.12E-06
ibm02a	6.44E-07	1.39E-06	2.03E-06
ibm07a	5.23E-07	6.06E-07	6.38E-07
ibm08a	4.78E-07	6.05E-07	5.98E-07
ibm10a	3.33E-07	3.91E-07	4.13E-07
ibm11a	3.35E-07	3.79E-07	3.98E-07
ibm12a	4.06E-07	4.74E-07	5.06E-07

Table 2: Congestion metrics for ibm designs.

bounding box. Note that partial information about the PDF and CDF is already contained in percentiles and therefore it is also contained in the percentile-based linear fit models. Therefore, the starting point for the PDF derivation was the percentile models for the ratio of the wire length versus bounding box as a function of the size of the bounding box. For both small and large bounding box data, we used a resubstitution-based technique to obtain the CDF. Note that the PDF can be easily obtained from the CDF using either symbolic or numeric differentiation.

The PDF is built using the following procedure. First a subset of k nets are randomly selected for short nets. In our experimentation, we used value $k = 50\%$. The data is separated in bins that are dictated by BBOX values. The size of bin was determined in such a way that all bins contain the same number of points. The total number of bins was 10. The randomly selected subset of data is used to establish new percentile points for each bin containing data. All percentile points are normalized against the bounding box with shortest nets. The normalization is done in such a way that the average discrepancy between the values that correspond to the identical percentile is minimized. The data is fit using polynomials of low degree (three and four in our experimentation). The procedure is repeated a large number of times, the average value for each of percentile is calculated and fit using a least linear squares approach. This process was terminated once the percentile validation method indicated that we achieved user specified intervals of confidence for the PDF model. The same procedure is repeated for long nets. Figures 2 and 3 show intermediate and final results of the PDF derivation procedure.



(a) Large BBOX

(b) Small BBOX Model

Figure 2: Linear Regression Model for ibm07 design using Cadence Router.

4.6 Interdesign Modelling

Note that our goal is to predict distribution of expected wire length for nets of the design that are not used to build the statistical model. Therefore, once a model was built and validated for a single design, we have to establish means for rapid re-mapping of the wire-length model to other chips.

For this task, we considered the following atomic chip properties: (i) the area of the chip; (ii) the number of nets; (iii) the average and median of bounding box areas, MST, and convex hulls for all nets (iv) the average number of terminals per net; (v) the percentage of the number of nets with a small number of terminals (two , three, or four). The composite chip metrics included ratios of all atomic chip properties and their simple statistical measures such as moments of low orders.

Table 1 shows the chip level characteristic of the designs. The first column denotes the name of the benchmark, followed by the number of chip layers and the number of nets in the benchmark. The fourth column denotes the total area of the chip. The overall congestion of the design is denoted in the fifth column by the total number of nets over the area of the design. The final column specifies the total number of terminals in the benchmark. Table 2 denotes the normalized average size of the bounding box, MST and CHULL for each net for each design. The statistics are normalized against the area of the chip.

We denote by c_i and c_j the overall congestion of designs i and j measured by the normalized sum of convex hull area for each design divided by the total area of the design. Furthermore, we denote by NL_i and NL_j the number of layers used in designs i and j. Our model indicates that the length of the net in design i (L_i) can be calculated using the length of the net with the same BBOX in design j (L_j) using the following formula $L_i = L_j \frac{NL_j}{NL_i} * (\frac{C_i}{C_j})^{0.48}$. This model is built using least linear squares data fitting approach [15]. We build this model using a randomly selected subset of four designs. The model was validated against the remaining design, as well as using resubstitution procedure as explained in the next Subsection.

4.7 Evaluation and Validation

The last step of the modeling procedure was dedicated to the evaluation of the accuracy of the developed models. We followed two paradigms: learn-and-test and resubstitution [7, 8, 9]. In the case of the former procedure, we selected a subset of nets for building the model. This procedure was properly applicable only on modeling done on a single design, since the total number of available designs was too small statistically for sound application of this type of analysis on interchip models. Nevertheless, the application of the learn-and-test procedure on the interchip model indicates very high consistency, strongly implying that different designs follow very similar distributions of the wire lengths for nets characterized by the selected features.

We have applied the learn-and-test validation technique to both trend modeling and outlier identification. In both cases, for single chip models, we obtained predictions with 3% of accuracy for more than 96% of instances.

Resubstitution is the technique that effectively resamples the available data in order to ensure that overfitting is not conducted. It was applied to modeling at both levels of abstractions: interchip and intrachip. We created 100 different subsets of data using uniform random sampling of the data. For the interchip modeling, we select 70% of data for each subset and build a separate model using the developed procedure. The percentile analysis indicates that for all results, the interval of confidence is less than $\pm 3\%$ with probability higher than 97%. For the interchip modelling, we selected a random subset that contained between three and five designs. We repeated this procedure 100 times. The interval of confidence was $\pm 10\%$ with probability higher than 86%. This relatively lower probability was the direct consequence of the fact that from a statis-

tical point of view relatively few design were available. Nevertheless, the percentile analysis [9] strongly validates the approach and indicates that the statistical trends have less than one in billion of chance to be accidental.

5 Statistical Wire-length PDF and CDF Models

In this Section, we present the obtained statistical wire length model. We present the parameters of the model, obtained PDF and CDF, and summarize model evaluation results. Although we present a single final model, it is important to emphasize that the procedure presented in the previous Section resulted in a large number of competitive models that differed relatively little with respect to their accuracy and interval of confidence. The model that we present was mainly selected due to its low conceptual complexity and through the use of a set of features that can be rapidly extracted from the post-placement designs.

The prediction abilities of the model are illustrated in Figures 2(a)-5(b). The demonstration example used for the development of the model is imb07. It is important to emphasize that the model was actually developed using only 60% of randomly selected nets. Figures 2(a) and 2(b) show the normalized net length with respect to BBOX for different sizes of BBOX. The continuous lines in these two figures indicate the prediction models for small and large BBOX respectively. The bottom line correspond to 10% percentile and the top line to 90% percentile value. All other lines indicate the value of expected length for percentiles that differ by 10% increments. Tables 3 and 4 present the parameters of the models and the obtained R^2 values. That that the square of residuals is consistently high. t-test indicates that for both sets the probability of accidental coincidence is less than 10^{-18}. Therefore, it is clear that the model is both theoretically and practically sound.

As can be seen from the table, the variability of the net lengths is well captured as indicated by the high value of the R^2 coefficient, in particularly for the small BBOX model. There are two main reasons why it is much easier to accurately predict short nets. The first one is that there are significantly more short nets than long nets and, therefore, the statistical model can be developed using a much larger number of samples. The second reason is that short nets usually have significantly fewer terminals, simple structure, and can leverage on relatively small areas of white space in their vicinity. For longer wires, we see that the prediction of nets that are almost as short as their lower bound indicated by BBOX is more accurate than nets that are long. For the long nets, the model relies on the CART model presented in the previous Section that has very high consistency. The CART model-based removal of nets that are predicted to be significantly longer than the BBOX-bound, improves the R^2 for all percentiles to above 0.95 level essentially matching the accuracy of the model for short nets. The CART model correctly identifies very long nets with accuracy better than 90%. More importantly, less than 1/than indicated by BBOX linear regression-model is not detected by the CART model. Finally, note that no short net (with BBOX value less than 6,000) was identified as outlier.

ibm07a - Small BBOX Linear Regression Models

%ile	a	b	c	R^2
90	9E-09	2E-05	1.1758	0.9876
80	9E-10	5E-05	1.0686	0.9762
70	2E-10	6E-05	1.1175	0.9184
60	-2E-09	5E-05	1.0439	0.9804
50	-4E-09	5E-05	1.0131	0.9849
40	-2E-09	3E-05	1.0055	0.9876
30	1E-09	6E-06	1.0160	0.9854
20	-9E-10	1E-05	1.0038	0.8049
10	1E-09	-3E-06	1.0023	0.9702

Table 3: Linear Regression Fit Parameters and R^2 for Small BBOX of ibm07 design. Coefficients a, b, and c are used for the quadratic model of the form $ax^2 + bx + c$.

ibm07a - Large BBOX Linear Regression Models

%ile	a	b	c	R^2
90	5E-11	5E-06	1.5944	0.7185
80	-1E-11	7E-06	1.3948	0.6268
70	-2E-11	8E-06	1.2767	0.6890
60	-5E-11	9E-06	1.1828	0.7460
50	-3E-11	7E-06	1.1383	0.8111
40	-3E-11	6E-06	1.0981	0.8655
30	-2E-11	5E-06	1.0720	0.9109
20	-3E-11	5E-06	1.0476	0.9033
10	-3E-11	5E-06	1.0135	0.8862

Table 4: Linear Regression Fit Parameters and R^2 for Large BBOX of ibm07 design. Coefficients a, b, and c are used for the quadratic model of the form $ax^2 + bx + c$.

Figure 3 and 4 show a cumulative distribution function (CDF) and a probability distribution function (PDF) for short and long nets. The x-axis indicates the normalized discrepancy against the most likely values. Again, the continuous line indicates the prediction provided by the model and each plot point corresponds to the length of the nets in a particular bounding box bin selected by the resubstitution procedure. From the PDF figures we can conclude that the majority of nets are routed using a wire-length that is close to theoretical minimum and that longer nets are statistically rare.

We evaluated accuracy and consistency of PDF and CDF using the resubstitution procedure. We generated 100 different subset that contain 60% of initial date and build PDF and CDF of the wire length model. For a hundred randomly selected points their PDF and CDF values were recorded for each of the resubstitution models. The non-parametric interval of confidence was calculated for each point and for the overall probability and cumulative distribution functions. The analysis indicates that with a probability larger than 96% the model is accurate within $\pm$7%. It is interesting to note that interval of confidence was sharper for the CDF than for the PDF most likely as consequence that CDF integrates discrepancies of PDF.

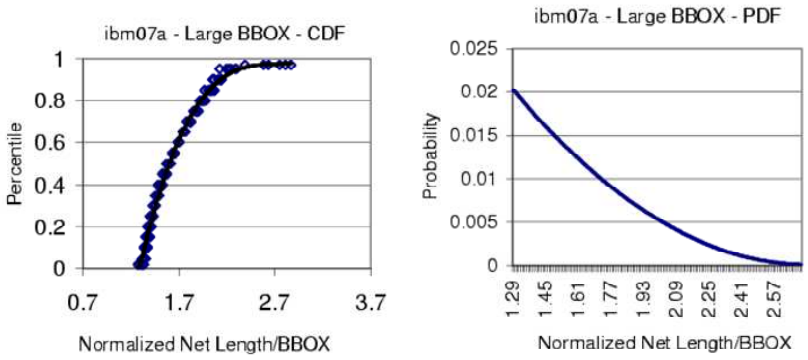


Figure 3: Cumulative Distribution Function and Probability Distribution Function for Large BBOX nets in ibm07 design for Cadence Router.

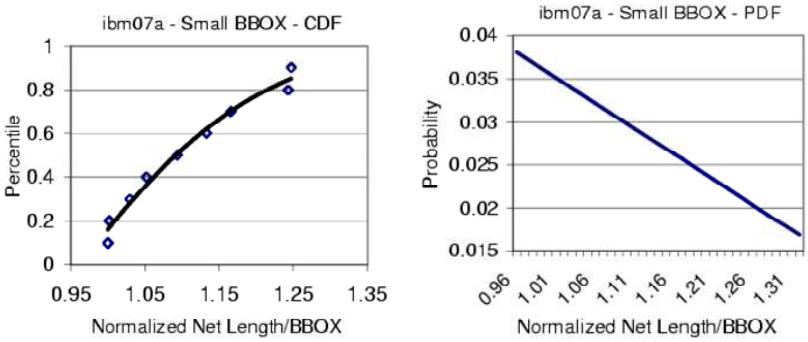


Figure 4: Cumulative Distribution Function and Probability Distribution Function for Small BBOX nets in ibm07 design for Cadence Router.

Finally, Figures 5(a) and 5(b) show a 3-dimensional representation of histograms that is formed by selecting bins according to their ratio of normalized net length versus BBOX and the size of BBOX on the other axis. The z-axis indicates instead of the conventional number of nets which belong to a particular bin, the logarithm of this value in order to provide better visual insight in to the distribution of wire-lengths of the net for all lengths. Data in Figure 5(a) was collected after using the Cadence routing tool. Data in Figure 5(b) is generated using the developed prediction model. It is easy to see that there exists close correspondence and high correlation between data in the two figures, except that for a small subset of bins in the true data that have statistical anomalies due to the specifics of the actual design.

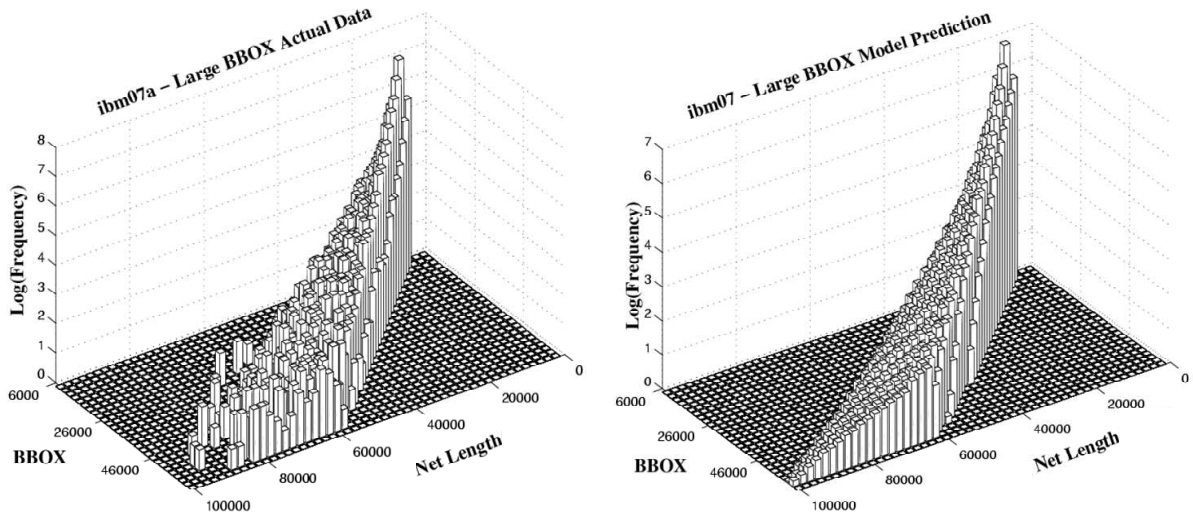
6 Application of Statistical Wire Length Model to Probabilistic Buffer Insertion

In this section, we describe a few applications of the presented wire length model. The common underlying idea is to demonstrate the superiority of statistical estimation and probabilistic optimization over the traditional deterministic approach to design automation. In order to accomplish this objective, we applied the developed statistical models to the probabilistic buffer insertion problem.

The buffer insertion problem can be formally stated in the following way. *Given the fan-out wiring tree with parasitic resistances and capacitances, wire-lengths, potential buffer locations, sink required times, sink capacitive loads and a delay constraint at the driving gate, the problem is to place buffers into the tree such that the required arrival time at the input of the driving gate is maximum. We also consider the optimization of the number of buffers used to satisfy the delay constraint.*

The buffer insertion problem was formalized by [18] and models the fan-out wiring tree as a set of distributed RC sections. The Elmore Delay model [10] is used to compute the delay of such a wiring tree.

In order to estimate the parasitics for each wire-segments we need to determine the exact wire-lengths. Now let us suppose that this optimization is being performed during the inplace mode during which the exact wire-length is not available. The only available information is about the bounding box of the nets. Using the placement information we can generate the probability distributions of individual wire segments of the wiring tree and perform buffer insertion probabilistically. [12] proposed such a probabilistic approach to buffer insertion. For brevity, we omit the details of that algorithm. We ran probabilistic buffer insertion on a placed net (placed using Cadence Qplace) and also traditional buffer insertion [18] assuming bounding box as the net length estimate. After buffer insertion, the entire circuit was routed and the net delay was



(a) Actual Data

(b) Length generated using Prediction Model

Figure 5: Logarithm of Histogram of Number of Nets of given Length and given BBOX for ibm07 Large BBOX.

	Prob.		BBox	
	Delay	# Buf	Delay	# Buf
Net1	8.31	14	9.59	9
Net2	6.0	20	8.74	17
Net3	6.4	22	8.51	17

Table 5: Post Routing Comparison: Prob. vs BBox based Buffer Insertion on ibm08 design.

computed using real wire delay values.

The following table compares the post routing net delays from probabilistic and traditional buffer insertion. Table 5 reports the comparison. It can be seen that post routing, the probabilistic approach produces significantly better results than bounding box based approach indicating the effectiveness of our models and also the superiority of a probabilistic approach.

7 Conclusion

We have built a compact statistical model that predicts the probability that a given net will have a particular wire-length. The model is characterized using a small set of parameters that are easily extracted from the design's floorplan. The run-time of the model is less than one second even for the largest designs. The model was validates using both learn-and-test and resubstitution evaluation techniques.

The proposed net length models have large range of applicability in emerging probabilistic approaches to design automation that are rapidly gaining acceptance. We demonstrated the effectiveness of our model through extensive experimentation with state of the art commercial and academic tools.

References

[1] A. Agarwal, D. Blaauw, V. Zolotov, and S. Vrudhula. Computation and refinement of statistical bounds on circuit delay. In *ACM/IEEE Design Automation Conference*, pages 348–353, 2003.

[2] C. J. Alpert and A. Devgan. Wire segmenting for improved buffer insertion. In *ACM/IEEE Design Automation Conference*, pages 588–593, 1997.

[3] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De. Parameter variations and impact on circuits and microarchitecture. In *ACM/IEEE Design Automation Conference*, pages 338–342, 2003.

[4] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Chapman and Hall, 1984.

[5] P. Dalgaard. *Introductory Statistics with R*. Springer-Verlag, New York, NY, 2002.

[6] A. Davoodi and A. Srivastava. Voltage scheduling under unpredictabilities: A risk management paradigm. In *ACM/IEEE Int'l Symposium on Low Power Electronics and Design*, pages 302–305, August 2003.

[7] B. Efron. Bootstrap methods: another look at the jackknife. *Annals of Statistics*, 7:1–26, 1979.

[8] B. Efron. *The Jackknife, the Bootstrap, and Other Resampling Plans*. S.I.A.M., Philadelphia, 1982.

[9] B. Efron and R. Tibshirani. *An introduction to the bootstrap*. Chapman & Hall, 1993.

[10] W.C. Elmore. The transient analysis of damped linear networks with particular regard to wideband amplifiers. In *Journal of Applied Physics*, volume 19 of *1*, 1948.

[11] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, New York, NY, 2001.

[12] V. Khandelwal, A. Davoodi, A. Nanavati, and A. Srivastava. A probabilistic approach to buffer insertion. In *IEEE International Conference on Computer Aided Design*, pages 560–567, Nov 2002.

[13] J. Lillis, C. K. Cheng, and T. T. Y. Lin. Optimal wire sizing and buffer insertion for low power and a generalized delay model. In *IEEE International Conference on Computer Aided Design*, pages 138–143, 1995.

[14] F. P. Preparata and M. I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, New York, NY.

[15] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical recipes in C (2nd ed.): the art of scientific computing*. Cambridge University Press, 1992.

[16] A. Srivastava, E. Kursun, and M. Sarrafzadeh. Predictability driven binding: Methodologies and tradeoffs. In *Journal of Circuits, Systems and Computers, Special Issue on Low Power IC Designs*, volume 11 of *4*, pages 223–232, August 2002.

[17] Ronald A. Thisted. *Elements of statistical computing*. Chapman & Hall, Ltd., 1986.

[18] L.P.P.P. van Ginneken. Buffer placement in distributed rc-tree networks for minimal elmore delay. In *Int'l Symposium on Circuits and Systems*, pages 865–868, December 1990.

[19] C. Visweswariah. Death, taxes and failing chips. In *ACM/IEEE Design Automation Conference*, pages 343–347, 2003.