Interconnection Analysis for Standard Cell Layouts

Massoud Pedram
Department of Electrical Engineering – Systems
University of Southern California
Los Angeles, CA 90089

Bryan T. Preas
Computer Science Laboratory
Xerox, Palo Alto Research Center
Palo Alto, CA 94304

Abstract— **We present an accurate model and procedures for predicting the common physical design characteristics of standard cell layouts (i.e., the interconnection length and the chip area). The predicted results are obtained from analysis of the net list only, that is, no prior knowledge of the functionality of the design is used. Random and optimized placements, global routing, and detailed routing are each abstracted by procedural models that capture the important features of these processes, and closed-form expressions that define these procedural models are presented. We have verified both the global characteristics (total interconnection length and layout area) and the detailed characteristics (wire length and feedthrough distributions) of the model. On the designs in our test suite, the estimates are very close to the actual layouts.**

I. Introduction

Interconnection analysis addresses two related problems: the wire (interconnection) length estimation problem and the wiring area estimation problem. Many researchers have addressed the area estimation problem and have proposed efficient and accurate techniques for it. The wire length estimation problem has however not been solved satisfactorily. Interconnection length studies tend to be either *theoretical* and hence not applicable to specific designs, or *empirical* such as those relating the Rent's rule parameters to the average wire length. Early research into interconnection length estimation, although of theoretical interest, is too general to be useful for specific designs. Later work, which produces results that have the appropriate level of detail, requires knowledge of the *Rent's exponent*, or assumes particular wire length distributions. In practice, assumptions about wire length distributions are either not verified or require fitting curves to the actual layout data.

A method which derives the wire length distribution as a function of the logic structure of a given design, the physical features of primitive logic cells in the Standard Cell library, and the algorithmic features of physical design processes which are used to construct the placement and routing solutions, is needed. This requirement encourages development of *procedural* models which produce interconnection length estimates with a high degree of accuracy without making arbitrary assumptions about the wire length distribution or fitting curves to the data.

Interconnection length models have many uses as summarized below.

- They can help evaluate the fit of a logic design to a fabrication technology. More precisely, the models determine routability of the given logic design, subject to the constraints of the technology, and therefore, help the system designers trade off aspects of the design and the technology.
- The area required for interconnections within a circuit layout largely depends on the total length of wire that must be accommodated. Accurate estimation of total interconnection length is, therefore, an essential part of any area estimation procedure. In fact, many area estimation techniques require the wire length for the logic design as an input parameter [11], [14].
- Interconnection estimates are useful during the technology mapping phase of the logic synthesis since they can predict the cost of various implementations [17].
- Interconnect estimation has taken on new importance due to 1) technology migration to deep submicron regime where interconnect effects by far dominate gate effects in terms of circuit area and delay [2] and 2) emergence of low power dissipation as a major driver of the CAD tools and design techniques [3]. To expand on this point, it should be noted that most of the ongoing research work on power estimation in CMOS circuits focuses entirely on estimating the switching activity inside a circuit under different input characteristics and thereby ignores the other (equally important) contributing factor, namely, the physical capacitance [16]. This capacitance is itself dominated by the interconnect capacitance, thus, having an accurate interconnect model will improve the accuracy of the power estimation techniques and the effectiveness of the power optimization steps that rely on these estimations.

A. Prior Work

Interconnection analysis models are divided into three categories: empirical, theoretical, and procedural. Empirical studies produce expressions for physical characteristics by extracting information from actual designs and fitting curves to the data. Theoretical studies produce closed form expressions by making simplifying assumptions about the interconnection structure. Procedural models consider properties of the actual physical design processes which are used in the design flow, physical characteristics of the primitive cells in the library, and interconnection structure of the design to improve the accuracy of the predictions.

A.1 Empirical Models

The initial work on the wiring requirements was performed by Rent in the early 1960's. He derived *Rent's rule* which is a relationship between the I/O count and the cell count of a design by fitting curves to the empirical data

from various computer designs. Rent's rule is given by

$$ioCount = (averageCellSize) \cdot (cellCount)^r$$

where r is *Rent's exponent*. Landman and Russo [15] studied the relation between cell count versus I/O count and Rent's exponent. They showed that two different values of Rent's exponent must be used depending on the number of cells; that is, circuits with larger cell counts and smaller package counts have smaller Rent's exponents. Donath [10] reported that values of Rent's exponent ranged as high as 0.75 for highly parallel designs and as low as 0.47 for highly serialized designs. Sastry and Parker [21] derived an interconnection length distribution that fits actual designs.

These models require knowledge of empirical parameters (such as Rent's exponent) which are computed from actual design instances. An implicit assumption is that the design instances which were used in deriving the values of these parameters exhibit the same interconnection structure and design characteristics as those of the design under consideration. This assumption limits the applicability of the empirical formulas.

A.2 Theoretical Models

Theoretical models produce closed form, mathematical descriptions of the physical characteristics from logic designs and physical implementation technologies. These models provide general trends but lack sufficient detail to accurately represent individual designs. They are useful when little is known about the actual design process. These models are divided into two categories: deterministic and stochastic.

Deterministic models rely on parameters extracted from actual designs. The effects of the physical design processes are characterized by simple, measurable parameters. As an example, Donath [8] devised a plausible structure for a logic design which conforms to Rent's rule. He assumed a hierarchical structure where only a fraction of the pins inside a cell are connected to pins outside the cell (the "encoding" assumption). He showed that such a structure exhibits Rent's rule. He also demonstrated that a randomly constructed design does not conform to Rent's rule.

A major thrust in stochastic approaches models the interconnection characteristics of the design as a stationary process. The wiring requirements are computed by making assumptions about the probability distributions of wires. An early attempt to formalize the characteristics of computer logic designs was published by Donath [9]. He defined a top-down hierarchical design approach in which each step of the expansion of the hierarchy is modeled by the substitution of a pattern of interconnected cells for each block. These patterns are selected randomly from a fixed pattern library by a stochastic process. Based on this model, Donath established the relation between the cell-to-pin ratio and performance.

Heller et al. [13] addressed the problem of estimating wiring space requirements. He modeled interconnection wires as independent two-point wires originating stochastically (with a Poisson distribution) at some cell, covering a random distance (an average interconnection length), and terminating at some second cell. Based on this model, he derived the probability of wiring completion of some number of cells in a limited number of wiring tracks. His model correctly predicts the relative difficulty of wiring completion in various designs. El Gamal [11] refined Heller's model. His model assumes a regular two-dimensional array of cells where the generation process and the length of interconnecting wires are modeled as in Heller's work. The path traveled by each wire is established randomly, with the restriction that its endpoints be separated by a Manhattan distance which is equal to the path length. El Gamal derived from this model the minimum number of wire segments, and hence the minimum wiring area required for the square array of cells. He concluded that the overall minimum wiring area is of order $N^2 log^2 N$ where all cells have been placed in an $N \times N$ array.

Sastry and Parker [21] used a model similar to El Gamal's. They modeled interconnections as independent two-point wires covering an average length and derived expressions for channel widths, probability of routing completion, and wire lengths. They showed that wire length distribution has the form of a Weibull distribution with location and shape parameters. These parameters must be computed based on the net lengths obtained from actual layouts. Kurdahi and Parker [14] presented an area estimator for standard cell layouts. They assumed rows of equal size, double entry cells, constant pin pitch, two-pin nets, and minimum rectilinear connection paths. Their model assumes *birth* of a wire at pin slot i and length of a wire l are independent random variables with probabilities $p_B(i)$ and $p_L(l)$. They suggested uniform distribution for $p_B(i)$ and geometric distribution for $p_L(l)$. Based on these assumptions, the required routing area is estimated. This model, however, requires knowledge of average interconnection length which is computed by fitting curves to known data.

These models, although of great theoretical interest, are too general to be useful for specific design decisions. They require knowledge of empirical parameters or hypothetical wire length distributions. Assumptions about wire length distributions are either not verified in practice or require fitting curves to the actual layout data. Many area estimators require wire lengths as input. The accuracy of the area estimates is, therefore, bounded by the accuracy of interconnection length estimates. To be useful for design work, however, highly accurate estimates are needed. To achieve this level of accuracy, proper abstractions to model layout processes and physical structures and careful analysis of the interconnection structure of the design under consideration, are necessary. Theoretical models lack this level of detail and therefore produce results that are not accurate enough for today's design work.

A.3 Procedural Models

Procedural models incorporate greater detail and a lower level of abstraction compared to other models. They rely on relations derived from knowledge of the actual design

processes, interconnection structure of the design, physical layouts of the leaf cells, and layout rules. These models extract interconnection characteristics of the design and combine them with abstractions of the placement and routing processes to give estimates without need for arbitrary wire length distribution assumptions or empirical parameters.

Sechen [22] presented an interconnection length estimator which gives accurate estimates for small designs. He assumed square cells which are placed on a square, two-dimensional grid. For each size of net, the half perimeter of the smallest rectangle enclosing all pins on the net is computed by assuming that a sample cell is placed randomly within a square array of area equal to the average number of cells connected to the cell. Various scenarios and a look up table are used to determine all possible arrangements of cells which establish a given bounding box. Total interconnection length is then computed by summing (over all nets) the half perimeter lengths of the rectangles enclosing pins on the nets. Sechen's abstraction of the layout surface makes his model most applicable to the "sea-of-gates" style. His approximation of total interconnection length for nets with large number of pins (> 4) is not accurate enough.[1] We implemented an interconnection length estimator based on half perimeter lengths of net bounding boxes. For the circuits in the test suite, errors up to 30-40% were observed.

Chen and Bushnell [4] introduced an area estimator for random placement with the assumption that wires do not share tracks. They derived the expected number of wiring tracks and feedthroughs in the central row, and thereby, estimated the chip width and height. The authors did not attempt to model global and detailed routing processes, and did not differentiate between designs based on their interconnection structures. Their estimated chip area for small designs is 40-70% over the actual chip area, and the number of wiring tracks is overestimated by a factor of 2-3. No data is presented for medium or large size designs.

B. Overview

To obtain accurate area estimates, it is necessary to achieve a high degree of accuracy in estimating the wire length. This task is accomplished by a *procedural* model which will be presented in this paper. The model captures the properties of the physical design processes (placement, global routing, and detailed routing), the characteristics of the primitive library cells, and the structural features of the logic design to accurately estimate interconnection length for standard cell layouts. Since interconnection length is a strong function of the structure of a given design, the first task is to extract relevant features which account for the wiring requirements of the design. A metric which captures the local influence of other nets over a net under consideration is introduced. This is a more pertinent and effective metric, as far as interconnection length estimation is concerned, compared to other metrics such as average pin per cell or average number of connected cells to any cell in the design.

The predicted results are obtained from analysis of the net list. No prior knowledge of the functionality of the design is used. The model considers multi-pin nets directly, and does not preprocess them into sets of two-pin nets, as is often the case. Using these wire length estimates, the chip width and height can be computed by a statistical area estimation technique described in Section II, or by a random offset track packing technique presented in Section III.

Two interconnection models, which were originally presented in [18], [19], are discussed. The *basic* model features a random placement, but optimized global and detailed routing. Since the random placement process can be characterized accurately, the effects of placement and routing within the overall model can be separated. The *improved* model extends the basic model by including optimized placement and is used in production. Optimized global and detailed routing abstractions from the basic model are retained.

In the remainder of this paper, the following notation is used. All scalar variable names are written with lower-case first character, e.g., *foo*; vector variables are denoted by *foo[.]*; function names are written with capital first character, e.g., *Foo(.)*.

Given knowledge of standard cell layouts and model assumptions, the model equations follow logically without reference to any empirical or arbitrary parameters.

II. The Basic Interconnection Model

The inputs to the area estimation model are the logical design specification and (primitive) leaf cells included in the specification. Following the standard cell model, double entry cells are placed in rows and interconnected in routing channels among the rows. Outputs of the estimation model are the estimated total wire length, wire length distribution, the estimated total number of feedthroughs, the feedthrough distribution, chip width and height, and chip area.

A standard cell layout is modeled as a regular $w \times n$ array, where n is the number of rows and w ($= numCells/n$) is the average number of cells per row. Wires follow rectilinear paths with horizontal segments on one layer (called *metal1* or $M1$) and vertical segments on another (called *metal2* or $M2$). The average cell width is computed from the cells actually used in the design. There exists an explicit feedthru cell for crossing cell rows.

The basic model assumes a random placement but optimized global and detailed routing processes. The following important aspects of the algorithms have been incorporated. The placement process uniformly distributes cells on the $w \times n$ grid. The global router finds a minimum rectilinear spanning tree to connect pins of nets. Wiring for a net does not meander outside the bounding box defined by the pins on the net. Feedthroughs are placed at the intersections of cell rows and the edges in the rectilinear spanning tree connecting pins on the net. No feedthrough

[1] Chung [5] showed that the worst case length of a minimal rectilinear Steiner tree connecting d pins of a net tends to be $(\sqrt{d}+1)/2$ of the half perimeter length of the smallest rectangle enclosing pins of that net.

is added to a row which contains a pin on the net. Each net contributes at most one feedthrough to each cell row. The channel router finds the shortest path inside the channel to connect pins on the net. The route does not meander outside the box enclosing these pins. Inside the channel each net is connected with trunks with no overlap along the length of the channel. Branches connect trunks to the pins. All branch layer conflicts can be resolved by adding horizontal jogs. Over-the-cell routing is not considered.

The assumption of independent nets allows us to compute the wire length and feedthrough contribution of each net separately. The random placement assumption implies uniform pin distribution over the layout surface, and is captured in the $FrM1L$ and $FrM2L$ equations. Consider a net with d pins uniformly distributed on a $w \times n$ *frame* (abbreviated as prefix Fr in the function names). Let

$$z = Min(d, n).$$

Sum of the lengths of *metal1* wires connecting all pins on the net (in units of cell pitch) is computed as

$$FrM1L(d, w, n) = (\frac{1}{n})^d \cdot \sum_{i=1}^{z} \left(\begin{array}{c} n \\ i \end{array} \right) \cdot A(i, d, w).$$

The term before summation gives the probability of placing d pins on some subset of n rows. The first term after summation gives the number of ways i rows can be selected from among n rows, and $A(i, d, w)$ gives the contribution of a d-pin net occupying exactly i rows to the *metal1* length.

To compute $A(i, d, w)$, all different configurations (groupings) of d pins on i rows (numbered from 1 to i) are examined. In particular, the following integer equation must be solved

$$\sum_{j=1}^{i} x_j = d \qquad x_j \geq 1.$$

The solution to this equation returns a list of *sets*. Each set represents a *distinct* pin configuration describing the distribution of pins on rows. For example, if $i = 3$, $d = 6$, $w = 60$, then solution to the integer equation is $((1, 1, 4), (1, 2, 3), (2, 2, 2))$. Elements in each set are non-decreasing in magnitude, that is, $(1, 1, 4)$ is an acceptable pin set, but $(1, 4, 1)$ is not. This equation is efficiently solved by a recursive procedure. The cardinality of the solution (list of sets) strongly affects the run time of the model since the number of solutions grows rapidly with d and i. Therefore, results for very large nets are approximated by dividing large nets into cliques of smaller nets. Now,

$$A(i, d, w) = \sum_{sets} A_1(i, d, set) \cdot A_2(i, d, set) \cdot CoL(i, w, set)$$

$$A_1(i, d, set) = \prod_{k=0}^{d-1} \left(\begin{array}{c} i - \sum_{j=1}^{k} rows[j] \\ rows[k+1] \end{array} \right) = \frac{i!}{\prod_{k=1}^{d} rows[k]!}$$

$$A_2(i, d, set) = \prod_{k=0}^{i-1} \left(\begin{array}{c} d - \sum_{j=1}^{k} pins[j] \\ pins[k+1] \end{array} \right) = \frac{d!}{\prod_{k=1}^{i} pins[k]!}$$

where $A_1(i, d, set)$ is the number of distinguishable row arrangements for a given pin set and $A_2(i, d, set)$ is the number of distinguishable pin distributions for a given row arrangement and a given pin set. $rows[k]$ is the number of rows with k pins, and $pins[k]$ is the number of pins on the kth row. For example, if $set = (1, 1, 4)$, then $rows[1] = 2$ and $pins[1] = 1$. For this pin set, distinguishable row arrangements are $(1, 1, 4)$, $(1, 4, 1)$ and $(4, 1, 1)$. For row arrangement $(1, 1, 4)$ and assuming pins are numbered as $p1, \cdots, p6$, then $((p1), (p2), (p3, p4, p5, p6))$, $((p2), (p1), (p3, p4, p5, p6))$, $((p2), (p3), (p1, p4, p5, p6))$ and so forth are distinguishable pin distributions.

CoL(i,w,set) gives the expected length of the net if it assumes the distribution of pins described by a particular set; It is an abstraction of the global router and assumes that pins of a net which are distributed on some set of rows do not share routing channels with each other. It is calculated as

$$CoL(i, w, set) = \begin{cases} WL(pins[1], w) & if \text{ i} = 1 \\ \sum_{k=1}^{i} WL(pins[k] + 1, w) & otherwise \end{cases}$$

where $pins[k]$ is sorted in increasing magnitude. These equations model a minimum rectilinear spanning tree global router.

$WL(m, w)$ gives the expected length of the net which has m pins on one side of a routing channel ($1 \leq m \leq w$), and is calculated as

$$WL(m, w) = \frac{\sum_{l=m-1}^{w-1} (w - l) \cdot \left(\begin{array}{c} l - 1 \\ m - 2 \end{array} \right) \cdot l}{\left(\begin{array}{c} w \\ m \end{array} \right)}.$$

The numerator is a sum over all possible spans of the m randomly placed pins on a row with w cells whereas the denominator is the number of ways m cells can be chosen from among w cells. The first term in the numerator is the number of ways spans of l cell pitches can be established within the channel, the second term in the numerator is the number of ways the remaining $m-2$ pins can be placed on the remaining $l - 2 + 1 = l - 1$ cells, and l is the cell span established by the pins. Note that $WL(2, w) = (w + 1)/3$.

Under the assumption of a single wire segment per track (i.e., $\forall m \; WL(m, w) = w - 1$), the equation for $FrM1L(d, w, n)$ reduces to the following

$$A(i, d, w) = B(i, d) \cdot i \cdot (w - 1)$$

$$B(i, d) \equiv i^d - (\sum_{j=1}^{i-1} \left(\begin{array}{c} i \\ j \end{array} \right) \cdot B(j, d)) \qquad B(1, d) = 1.$$

$B(i, d)$ which is defined recursively gives the number of ways of placing d pins on exactly i rows. Note that

$$(\frac{1}{n})^d \cdot \sum_{i=1}^{z} \left(\begin{array}{c} n \\ i \end{array} \right) \cdot B(i, d) \equiv 1.$$

Sum of the lengths of *metal2* wires connecting all pins of the net (in units of channel height) is computed as

$$FrM2L(d,n) = (\frac{1}{n})^d \cdot \sum_{i=1}^{z} \left(\begin{array}{c} n \\ i \end{array} \right) \cdot FrChSp(i,n) \cdot B(i,d)$$

$$FrChSp(i,n) = \frac{\sum_{l=i-1}^{n-1} (n-l) \cdot \left(\begin{array}{c} l-1 \\ i-2 \end{array} \right) \cdot l}{\left(\begin{array}{c} n \\ i \end{array} \right)}$$

where $FrChSp(i,n)$ is the expected number of channels spanned by a d-pin net (occupying i rows). The first term in the numerator is the number of ways spans of l channels can be established within the chip, the second term in the numerator is the number of ways the remaining $i-2$ rows can be chosen from among $l-2+1 = l-1$ rows, and l is the channel span. The denominator is the number of ways i rows can be chosen from among n rows.

The expected number of feedthroughs contributed by a d-pin net is computed next

$$FrFTH(d,n) = (\frac{1}{n})^d \cdot \sum_{i=1}^{z} \left(\begin{array}{c} n \\ i \end{array} \right) \cdot NuFTs(i,n) \cdot B(i,d)$$

where $NuFTs(i,n)$ is the expected number of feedthroughs added by a net which is occupying exactly i rows. It is given by an expression identical to that for $FrChSp$ with $l-i+1$ (number of feedthroughs) replacing l (channel span). This is because the global router does not add a feedthrough to a row which contains a pin on the net.

The total interconnection length required to connect all the nets and the total number of feedthroughs contributed by all the nets are

$$totM1L = \sum_{nets} nets[d] \cdot FrM1L(d,w,n)$$

$$totM2L = \sum_{nets} nets[d] \cdot FrM2L(d,n)$$

$$totFTs = \sum_{nets} nets[d] \cdot FrFTH(d,n)$$

where $nets[d]$ represents the number of nets with d pins. Note that the distributions of wire lengths and feedthroughs as a function of the number ($nets[d]$) or size (d) of nets in the logic design have been computed as well.

The abstraction of the channel routing process is composed of two components: the wire length abstraction captured by $WL(m,w)$ equation given previously and the segment packing into tracks abstraction described below. In order to develop a statistical channel density estimator, we make some simplifying assumptions here. See Section III for a more accurate estimator.

The average number of segments in a routing channel and the average segment length (over all nets) are computed as

$$avgChSegments = \frac{totSegments}{(n-1)}$$

$$avgSegmentL = \frac{totM1L}{totSegments}$$

where

$$totSegments = \sum_{d} FrSegments(d,n).$$

We assume that the *metal1* length for each net is divided equally into a number of segments as determined by the expected number of wire segments (trunks) for each size of net, that is

$$FrSegments(d,n) = (\frac{1}{n})^d \cdot \sum_{i=1}^{z} \left(\begin{array}{c} n \\ i \end{array} \right) \cdot i \cdot B(i,d).$$

Because of random placement process, one may argue that the segments in the channel originate according to a Poisson distribution with density λ (this is the density of the random Poisson points and in this context is equal to the average number of wire segments originated at each slot). We have

$$\lambda = \frac{avgChSegments}{w}$$

Next, define parameter α of the Poisson distribution as

$$\alpha = \lambda \cdot avgSegmentL = \frac{totM1L}{w \cdot (n-1)}$$

where α gives the number of segments crossing a particular slot in the routing channel.

From this and a confidence level of c (for example, 0.999) for routing completion, the required number of wiring tracks per channel is approximated as follows

$$\sum_{k=1}^{chTracks} \frac{e^{-\alpha} \cdot \alpha^k}{k!} \leq c$$

$$totTracks = (n-1) \cdot chTracks.$$

After computing the total number of wiring tracks required by the detailed router, the number of feedthroughs crossing each of the rows is computed. The probability of a feedthrough crossing row i (rows are numbered from bottom to top starting from 1) is given by

$$PFTOnRow(i,n) = \sum_{j=1}^{d-1} (\frac{i-1}{n})^j \cdot (\frac{n-i}{n})^{d-j} \cdot \left(\begin{array}{c} d \\ j \end{array} \right).$$

From the d pins on the net, assume that j are placed in rows below the ith row and $d-j$ pins are placed in rows above the ith row. $(i-1)/n$ is, then, the probability that one pin is placed in rows below row i and $(n-i)/n$ is the probability that another pin is placed in rows above row i. Note that if at least one pin on the net is placed on row i, then $PFTOnRow(i) = 0$. This is consistent with the assumption that the global router does not add feedthroughs to a row which contains some pin on the net.

For randomly placed designs, the number of feedthroughs crossing the central row is the largest. To compute the probability that a d-pin net will contribute a feedthrough to the central row, $i = (n+1)/2$ is used. Now,

$$FrCFTs(d,n) = PFTOnRow(\frac{n+1}{2}, n) \cdot FrFTH(d,n)$$

$$cFTs = \sum_{nets} nets[d] \cdot FrCFTs(d,n).$$

The *chanHeight* is computed as

$$chanHeight = \frac{totTracks \cdot trackSpacing}{n-1}$$

and finally, chip width, chip height, and actual *metal1* and *metal2* lengths (in μmeters) are computed as

$$cellPitch = cellWidth + \frac{cFTs \cdot ftWidth}{w}$$
$$chipWidth = w \cdot cellPitch$$
$$chipHeight = n \cdot cellHeight + (n-1) \cdot chanHeight$$
$$actualM1L = totM1L \cdot cellPitch$$
$$actualM2L = totFTs \cdot cellHeight + totM2L \cdot chanHeight.$$

III. The Improved Interconnection Model

The improved model assumes a placement optimization process, a minimum rectilinear spanning tree global router, and a left edge channel router. The features of the algorithm classes which are captured by the interconnection model are the following. The placement optimizer minimizes the sum over all the nets of the half perimeter length of the rectangle enclosing pins of each net. Pins inside the placement bounding box for the net are not optimized for that net. The global router approximates a minimum rectilinear spanning tree to connect pins on each net while accounting for the possibility to share a routing channel between two groups of pins. The same channel routing paradigm is assumed as in Section II.

Interconnection length and feedthrough count for each size of net is estimated and then summed over all the nets. *Metal1* wire length is expressed in units of average cell pitch, and *metal2* wire length is expressed in units of average channel height. The average interconnection lengths are computed by spatially restricting the possible positions of the pins on the net to a bounding box within the $w \times n$ grid. Considering feasible aspect ratios for this bounding box and various pin configurations within the box and averaging over all such states, the average interconnection lengths and feedthrough count for the net are computed. By summing over all nets, the total interconnection length and the total number of feedthroughs are computed. There is no explicit dependence on a particular cell library or fabrication technology for estimation of wire length. However, such information is required when the total interconnection length is used to estimate the chip width and height.

The interconnection structure of a design is characterized by *net neighborhood populations* (*NNP* s) which account for the local influence of other nets over a net in question. The *NNP* for a net is the number of *distinct* primary inputs/outputs (I/Os) and cells which are at distance zero or one from the net. To compute the *NNP* referenced to a particular net, we find all the cells and I/Os connected by this net (i.e. at distance 0 from the net). Every other net which is connected to these cells is followed until all the cells and primary I/Os which are at distance one from the net are visited. The *NNP* for the net is the total number of distinct cells and I/Os encountered in this manner. In the *NNP* computation, nets that connect more than 25% of the cells in the design, or have more than 40 pins, are ignored. (These are typically power and clock nets that go everywhere.) This procedure is repeated for all nets of given size resulting in the average neighborhood population for each size of net. At the end, *NNP[d]* contains the average neighborhood population for nets with d pins.

The *NNP* for a net reflects the conflicting demands on a placement optimizer that is attempting to optimally place the cells directly connected to the net. To clarify this notion, assume that the placement optimizer is seeking a placement of d cells connected by exactly one net. The optimizer will cluster these cells in a bounding box of minimum half perimeter length. In reality, however, it is not possible to place cells connected to each net in such a minimum length bounding box due to competition from other nets. The placement and routing of the cells directly connected to a net of size d (to a first approximation) is influenced by a cell and I/O population of size *NNP[d]*.

The abstraction of the placement optimizer is described next. Consider a d-pin net with pins on a two-dimensional, $w \times n$ grid. The d pins on the net can be placed within an $x \times y$ bounding box where y ranges from k_1 to k_2 and $x = \lceil NNP[d]/y \rceil$. k_1 is given by $\lfloor NNP[d]/XSpan(d,n) \rfloor$ and k_2 is equal to the $\lceil YSpan(d,n) \rceil$ where *XSpan(d,n)* and *YSpan(d,n)* are the expected cell span and the expected row span of the net if the net pins are randomly placed on the $w \times n$ grid. Due to the placement process which minimizes the half perimeter length of the rectangle enclosing all pins on the net, and due to conflicting demands of other nets, the d pins are uniformly distributed inside the $x \times y$ bounding box. Now,

$$W(x,y) = \frac{(w-x+1) \cdot (n-y+1)}{x \cdot cellWidth + (\gamma \cdot (y-1) + y) \cdot cellHeight}$$

where the numerator gives the count of all feasible sub-grids of size $x \times y$ in a grid of size $w \times n$, and the denominator gives the half perimeter length of the $x \times y$ grid. γ is the ratio of the expected channel height to the cell height.

The average length of the net with d pins is given by

$$M1L(d) = \frac{\sum_{y=k_1}^{k_2} W(x,y) \cdot FrM1L(d,x,y)}{\sum_{y=k_1}^{k_2} W(x,y)}$$

where *FrM1L(d,x,y)* is the expected length of the net if it is restricted to $x \times y$ bounding box and is given in Section II. Recall that this length depends on *CoL(i,x,set)* which gives the expected length of the net when it assumes the configuration described by a particular pin set. In the following

we address the *channel sharing* problem, i.e., given a particular pin configuration what is the probability that these pins are facing the same channel. This issue is important because pins on two adjacent rows can be connected within the shared channel.

Let *Sh(i,y)* denote the probability of a pair of rows sharing a routing channel when i rows are selected randomly from among y rows. This probability is calculated as

$$Sh(i,y) = \frac{\sum_{j=0}^{y-i} \left(\begin{array}{c} i-2+j \\ i-2 \end{array} \right)}{\left(\begin{array}{c} y \\ i \end{array} \right)} = \frac{i}{y}$$

where the numerator is the number of ways two out of i selected rows can share some channel, and the denominator is the number of ways i rows can be selected from among y rows. This equation is not exact and underestimates the sharing probability for $i > 2$. Let $pins[i,j] = pins[i] + pins[j]$. We now have

$$\begin{aligned} &CoL(i,x,set) = \\ &\quad (1 - Sh(i,y)) \cdot \{\textstyle\sum_{k=1}^{i} WL(pins[k]+1, x)\} + \\ &\quad Sh(i,y) \cdot \{\textstyle\sum_{k=1}^{i/2} TWL(pins[2k-1,2k]+1, x) + \\ &\quad (if\ \ IsOdd(i)\ \ then\ \ WL(pins[i]+1,x)\ \ else\ \ 0)\} \end{aligned}$$

where *WL(m,x)* was given in Section II and *TWL(m,x)* gives the expected length of m-pin portion of the net when all m pins lie on the two sides of a routing channel ($2 \leq m \leq 2x$) and is given by

$$TWL(m,x) = \frac{\sum_{l=(m-1)/2}^{x-1} 4(x-l) \cdot \left(\begin{array}{c} 2l \\ m-2 \end{array} \right) \cdot l}{\left(\begin{array}{c} 2x \\ m \end{array} \right)}.$$

The numerator is a sum over all possible spans of the m randomly placed pins on a channel with w cells on each side, and the denominator is the number of ways m cells can be chosen from among $2w$ cells. The first term in the numerator is the number of ways spans of l cell pitches can be established within the channel, the second term in the numerator is the number of ways the remaining $m-2$ pins can be placed on the remaining $2l$ cells, and l is the cell span established by the pins. Note that

$$TWL(2,x) = \frac{2(x^2-1)}{3(2x-1)}$$

which is equal to $x/3$ for large x.

Sum of the lengths of *metal2* wires connecting all the pins, the expected number of feedthroughs added to all rows and to the central row by a net of size d are computed in a manner similar to $M1L$ calculation. That is,

$$M2L(d) = \frac{\sum_{y=k_1}^{k_2} W(x,y) \cdot FrM2L(d,x,y)}{\sum_{y=k_1}^{k_2} W(x,y)}$$

$$FTH(d) = \frac{\sum_{y=k_1}^{k_2} W(x,y) \cdot FrFTH(d,y)}{\sum_{y=k_1}^{k_2} W(x,y)}$$

where *FrM2L(d,x,y)* and *FrFTH(d,y)* are given in Section II. The total *metal1* and *metal2* lengths required to connect all nets, the total number of feedthroughs crossing all rows, $totFTs$, and those crossing the central row, $cFTs$, are also calculated as in Section II.

Given average wire length and wire length distribution, the statistical technique of Section II or other statistical area estimation techniques can be exploited to estimate the total chip area and aspect ratio [13], [14], [21]. Here, we describe a new technique based on *random offset track packing* to model the detailed routing process.

The *metal1* length for each net is divided equally into a number of segments as determined by the expected number of wire segments (trunks) for each size of net. The average number of routing segments of a net with d pins is calculated as

$$Segments(d) = \frac{\sum_{y=k_1}^{k_2} W(x,y) \cdot FrSegments(d,y)}{\sum_{y=k_1}^{k_2} W(x,y)}$$

The expression for *FrSegments(d,y)* is similar to that given in Section II, except that we must now consider the effect of *channel sharing*. More precisely, if two rows share the same channel, they use a single wire segment (trunk); otherwise, they use two wire segments. Consequently, the average number of wire segments used to connect the pins distributed on i rows is

$$i \cdot (1 - Sh(i,y)) + \frac{i}{2} \cdot Sh(i,y) = i \cdot (1 - \frac{Sh(i,y)}{2}).$$

Therefore, term i in the expression for *FrSegments(d,y)* must now be replaced with $i \cdot (1 - \frac{Sh(i,y)}{2})$.

The number and lengths of all segments for each size of net lying in each channel are given by

$$chSegments[d] = \lceil \frac{nets[d] \cdot Segments(d)}{n-1} \rceil$$

$$segmentL[d] = \lceil \frac{M1L[d]}{Segments(d)} \rceil$$

The *track packing* problem, in the absence of vertical and horizontal constraints, is defined as follows: Given t segments which must be placed in tracks of equal length $w-1$ and given that segment i requires l_i units of track length, the objective is to determine the minimum number of tracks needed to accommodate all segments. This is the well known bin packing problem and is NP-complete [12]. Many heuristics exist which obtain packings with a "small" fraction of tracks more than the optimal packing.

The simplification made by assuming that no horizontal constraints exist on the wire segments causes underestimation of the routing area. By generating a uniformly distributed offset for each wire segment in the channel, this shortcoming is remedied. One could build a horizontal constraint graph for these randomly positioned wire segments. The assignment of tracks to wire segments corresponds to the proper coloring of this constraint graph (which is by

construction an interval graph) [24]. In the absence of vertical constraints, efficient optimal algorithms exist for coloring the interval graphs. The task at hand, however, is much easier because only density of the channel must be computed and this can be accomplished by a simple plane sweep technique. The total density of the standard cell layout (*totDensity*) is the sum of channel densities over all channels. Ignoring vertical constraints in the area estimation model produces small errors because modern dogleg routers often route channels at density. Then,

$$chanHeight = \frac{totDensity \cdot trackSpacing}{n-1}.$$

Finally, chip width, chip height, and actual *metal1* and *metal2* lengths (in μmeters) are computed as in Section II.

IV. Complexity Analysis

k	1	2	3	4	5	6	7	8	9	10	11
C(k)	1	2	3	5	7	11	15	22	30	42	56

TABLE I

$C(k)$ values for k ranging from 1 to 11

The complexity of the model presented above is

$$O(n \cdot \sum_{k=2}^{d_{max}} C(k))$$

$$C(k) = \sum_{i=1}^{k} \|Sets(i,k)\|$$

Table I gives values of $C(k)$ for k ranging from 1 to 15.

The run time is relatively independent of the size of design, but is strongly influenced by the maximum size of net being considered (d_{max}). For this reason, nets with large number of pins are divided into cliques of smaller nets. This division introduces little error because, typically, there are few large nets. The complexity of the random offset track packing problem is linear in the number of segments in each channel which is in turn proportional to the number of nets in the circuit.

V. Experimental Results

The interconnection model has been implemented in the Cedar language running on Xerox Dorado Workstations (2-MIPS machines) and incorporated the model into the DATools system developed at Xerox PARC [1].

Table II describes the examples used to test the model's predictions. The counter and the adder are circuits synthesized by the DATools system when no performance requirements are imposed. The adders are simple ripple-carry designs, the counters are carry-look-ahead designs. The *RSD* is part of a Reed-Solomon error correction circuit, and the *SnprCtl* is part of a cache controller. Primary1SC and Primary2SC are the benchmarks from the physical design workshop [20]. The placement is obtained by TimberWolfSC version 4.1 [23]. The global and detailed routers are discussed in [6], [7].

example	cells	I/Os	nets	pins
16b adder	144	52	177	546
SnprCtl	95	30	114	331
RSD	210	89	211	670
64b counter	478	130	585	1537
Primary1	750	73	903	2801
Primary2	2907	107	3029	8758

TABLE II

Summary of the example circuits used for the area estimator

Table III compares the model's wire length, area and aspect ratio estimates with the actual results obtained after placement, global, and detailed routing of the circuits. On this set of benchmarks, the estimates are within a 10% accuracy rate. Note that our program runs between two to three orders of magnitude faster than the actual place and route tools while achieving this level of accuracy.

The detailed characteristics of the model were verified by collecting data and generating statistics for the actual interconnection length and feedthrough count for each size of net, and comparing it with the model's estimated values. Table IV gives this comparison for Primary1SC.

	predicted			actual		
example	M1L	M2L	area	M1L	M2L	area
SnprCtl	24.1	20.1	0.81	22.1	17.5	0.76
16b adder	24.0	23.6	0.81	22.1	21.5	0.76
RSD	59.6	52.2	1.70	62.1	48.3	1.61
64b cntr	226.2	254.6	5.84	238.6	238.2	5.36
Primary1	714.0	545.5	27.1	782.7	491.2	26.9
Primary2	3958	3422	109	4300	3050	113

TABLE III

Comparison of estimates versus the actual results of wire length (mm), area (mm^2) and aspect ratio

pins	estimated		actual	
	M1L	FT Cnt	M1L	FT Cnt
2	484	0.353	506	0.484
3	843	0.797	835	0.597
4	1073	1.026	1100	0.530
5	1417	1.114	1474	0.846
7	2028	1.106	2630	0.833
12	4280	0.852	5689	1.330

TABLE IV

Detailed comparison of *metal1* length and feedthrough count for various sizes of nets for Primary1SC with 14 rows

We briefly note some sources of error. A large portion

of our 10% error budget arises from the fact that we operate on average behavior rather than worst case behavior. Although large scale features of the actual layouts (that is, layout area and aspect ratio, total *metal1* and *metal2* lengths and total feedthrough counts) remain relatively constant, the detailed wire length and feedthrough distributions as a function of number of pins on nets vary as much as 20-40% from one layout of the same circuit to the next. Our estimates of these two distributions are close to the average over several layouts. Another source of error is the incomplete characterization of the physical design processes. We do not capture some aspects of the processes. For example, our placement abstractions do not fully capture the interdependence of nets[2], our global routing abstractions exclude an improvement global routing phase targeted toward reducing local congestions, and our channel router abstractions ignore vertical constraints among various net segments in the channel.

VI. Concluding Remarks

We presented an interconnection model that predicts interconnection lengths and layout areas for standard cell layouts. The procedural model abstracts the important features of the physical design processes for standard cell layout (placement, global and detailed routing). The equations that define this model are based on the functions performed by the design processes rather than on unsubstantiated statistical distributions or on arbitrary parameters. We extract the relevant features (interconnection structure and leaf cell layouts) of the logic design to provide parameters for the equations. The detailed information (wire length and channel span distributions) abstracted from the logic design allows us to transform the two-dimensional area estimation problem into two one-dimensional problems. Technology independence is achieved through parameterizing the layout design rules. These predictions are within approximately 10% of the actual lengths and areas over a wide range of layout aspect ratios, and over a range of logic design characteristics. The prediction process is very efficient; it takes less than one minute to analyze each of the circuits posed as examples in this paper.

References

[1] R. Barth, L. Monier, and B. Serlet. Patchwork: Layout from schematic notations. In *Proceedings of the 25th Design Automation Conference*, pages 250–255, June 1988.

[2] M. T. Bohr. Interconnect scaling - the real limiter to high performance ulsi. In *Proceedings of IEDM*, pages 241–244, 1995.

[3] A. P. Chandrakasan, S. S. Scheng, and R. W. Brodersen. Low power CMOS digital design. IEEE *Journal of Solid State Circuits*, 27(4):473–483, April 1992.

[4] X. Chen and M. L. Bushnell. A module area estimator for VLSI layout. In *Proceedings of the 25th Design Automation Conference*, pages 54–59, June 1988.

[5] F. R. K. Chung and F. K. Hwang. The largest minimal rectilinear steiner trees for a set of n points enclosed in a rectangle with given perimeter. *Networks*, 9:19–36, 1979.

[6] J. Cong and B. T. Preas. A new algorithm for standard cell global routing. In *Proceedings of the* IEEE *International Conference on Computer Aided Design*, pages 176–180, November 1988.

[7] D. N. Deutsch. A 'dogleg' channel router. In *Proceedings of the 13th Design Automation Conference*, pages 425–433, June 1976.

[8] W. E. Donath. Hierarchical structure of computers. Technical Report RC 2392, IBM T. J. Watson Research Center, March 1969.

[9] W. E. Donath. Stochastic model of the computer logic design process. Technical Report RC 3136, IBM T. J. Watson Research Center, November 1970.

[10] W. E. Donath. Placement and average interconnection lengths of computer logic. IEEE *Transactions on Circuits and Systems*, CAS-26(4):272–277, April 1979.

[11] A. A. El Gamal. Two dimensional stochastic model for interconnections in master slice circuits. IEEE *Transactions on Circuits and Systems*, CAS-28:127–138, February 1981.

[12] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-completeness*. W. H. Freeman and Company, 1979.

[13] W. R. Heller, W. F. Mikhail, and W. E. Donath. Prediction of wiring space requirements for LSI. In *Proceedings of the 14th Design Automation Conference*, pages 32–42, June 1977.

[14] F. J. Kurdahi and A. C. Parker. Techniques for area estimation of VLSI layouts. IEEE *Transactions on Computer-Aided Design of Integrated Circuits and Systems*, CAD-8(1):81–92, January 1989.

[15] B. S. Landman and R. L. Russo. On a pin versus block relationship for partitions of logic graphs. IEEE *Transactions on Computers*, C-20(12):1469–1479, December 1971.

[16] M. Pedram. Power minimization in IC design: principles and applications. ACM *Trans. on Design Automation of Electronic Systems*, 1(1):1–54, January 1996.

[17] M. Pedram and N. Bhat. Layout driven technology mapping. In *Proceedings of the 28th Design Automation Conference*, pages 99–105, June 1991.

[18] M. Pedram and B. T. Preas. Accurate prediction of physical design characteristics of random logic. In *Proceedings of the International Conference on Computer Design*, pages 100–108, October 1989.

[19] M. Pedram and B. T. Preas. Interconnection length estimation for optimized standard cell layouts. In *Proceedings of the International Conference on Computer Design*, pages 390–393, October 1989.

[20] B. T. Preas. Benchmarks for cell-based layout systems. In *Proceedings of the 24th Design Automation Conference*, pages 319–320, June 1987.

[21] S. Sastry and A. C. Parker. Stochastic models for wirability analysis of gate arrays. IEEE *Transactions on Computer-Aided Design of Integrated Circuits and Systems*, CAD-5(1):52–65, January 1986.

[22] C. Sechen. Average interconnection length estimation for random and optimized placements. In *Proceedings of the* IEEE *International Conference on Computer Aided Design*, pages 190–193, November 1987.

[23] C. Sechen and K. W. Lee. An improved simulated annealing algorithm for row-based placement. In *Proceedings of the* IEEE *International Conference on Computer Aided Design*, pages 478–481, 1987.

[24] T. Yoshimura and E. S. Kuh. Efficient Algorithms for Channel Routing. In IEEE *Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pages 25–35, January 1982.

[2]Note that the structure of a network can be characterized more accurately by a growing sequence of *multi-level* net neighborhood populations.