A Stochastic Model for Interconnections in Custom Integrated Circuits

ABBAS A. EL GAMAL, MEMBER, IEEE, AND ZAHIR A. SYED, STUDENT MEMBER, IEEE

Abstract—**A stochastic model for interconnections in integrated circuits composed of unequal size logic blocks separated by routing channels is described. An algorithm, based on the model, is given for estimating channel widths and chip area. The effectiveness of the algorithm is tested through an example. Applications of the model to placement and routing of integrated circuits are discussed.**

I. Introduction

WE CONSIDERED in [1] the problem of estimating interconnections area for master slice integrated circuits. We obtained estimates for routing channels width in terms of the number of logic blocks, the average number of interconnections emanating form the logic blocks, and the average interconnection length. The goal of this paper will be to extend the work in [1] to integrated circuits composed of arbitrary size logic blocks with routing channels running in the space separating the blocks as depicted in Fig. 1.

We begin in Section II by describing a statistical interconnections model for integrated circuits of the type shown in Fig. 1. This model is essentially a generalized version of the model in [1].

In Section III, an algorithm is provided for estimating routing channel widths. This is done in two steps. First, the number of interconnections entering each channel from other neighboring channels is estimated. Then a simple diffusion model for interconnection generation and death is used to estimate the congestion in every channel. An example is given in Section IV. In Section V, we discuss some applications of our statistical models to the layout of large complex integrated circuits.

II. Description of the Model

We assume given is a placement of a finite number of unequal size nonoverlapping polygons, as shown in Fig. 1. The placement allows only for horizontal and vertical orientations of the sides of the polygons. The space between two sides of any two adjacent polygons is referred to as a *routing channel*. We denote by a *channel graph*, the undirected connected graph with edges representing the center lines of the routing channels. The nodes of the channel graph are the intersection points of the different channels center lines. The channel graph of the placement in Fig. 1 is given in Fig. 2.

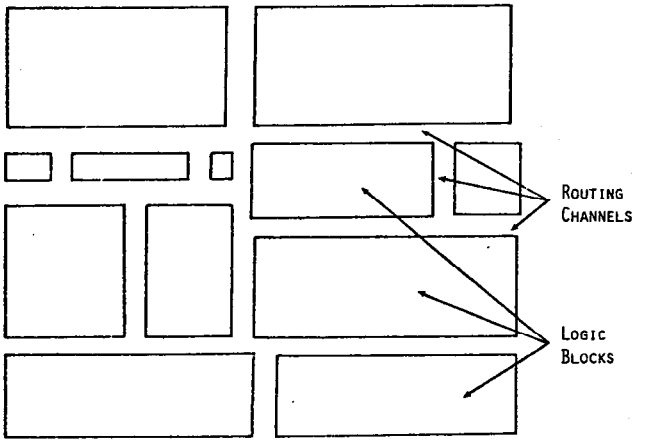


Fig. 1. A general integrated circuit organization.

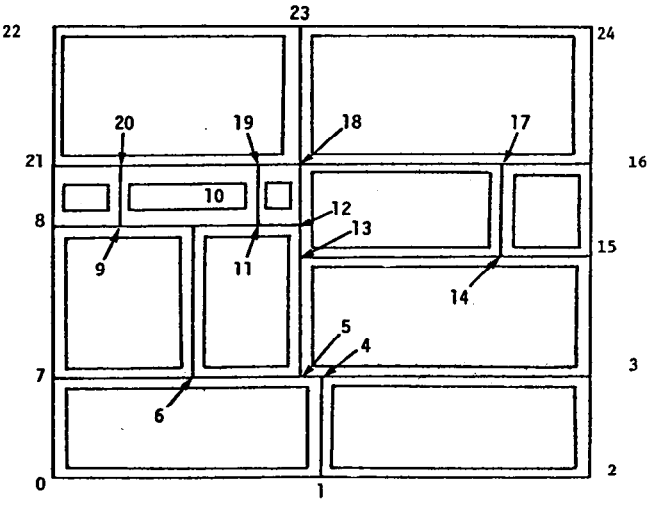


Fig. 2. Channel graph for Fig. 1.

Let the nodes of the channel graph be denoted by $1, 2, \cdots, N$ and denote an edge connecting nodes i and j by (i, j). The lengths of the routing channels are assigned to corresponding edges in the channel graph. Thus every edge (i, j) is assigned a positive number l_{ij} denoting the length of channel (i, j).

Manuscript received August 18, 1980; revised March 16, 1981. This work is supported in part by the Defense Advanced Research Projects Agency (DARPA) under Contract MDA 903-79-C-0680 and by the Joint Services Electronics Program (JSEP) through the Air Force Office of Scientific Research under Contract F 44620-76-C-0061.

A. A. El Gamal is with the Information System Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA 94305.

Z. A. Syed is with the University of Southern California, University Park, Los Angeles, CA.

0098-4094/81/0900-0888$00.75

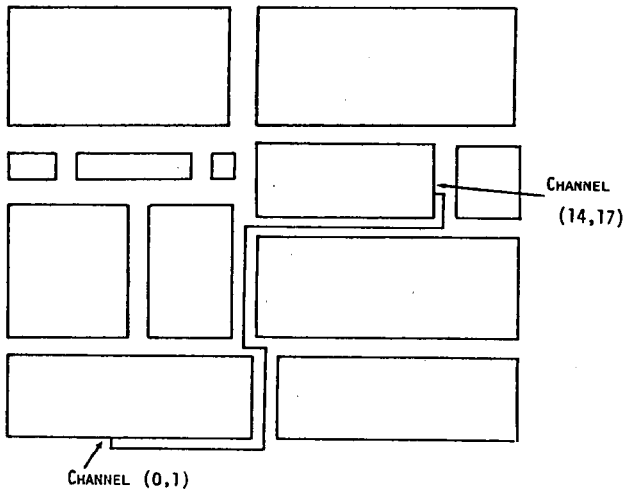


Fig. 3. Minimum length trajectory from channel (0, 1) to channel (14, 17).

Interconnection Generation: We assume that interconnections are randomly generated along the edges of the channel graph according to a Poisson process with parameter λ interconnections per unit channel length. The number of interconnections generated along channel (i, j) is, therefore, Poisson with parameter λl_{ij}.

Interconnection Path Description: Interconnections move along minimum distance paths only. For example, in Fig. 3 an interconnection originating at channel (0, 1) and moving to channel (14, 17) can only choose the shown minimum length trajectory along the channel graph.

Starting from any channel (l, k) the permissible shortest distance paths along the channel graph can be labeled (this will be made more precise in the next section). An interconnection originating at channel (l, k) and arriving at node i, chooses with equal probability (except for "dead ends" as defined in Section III) among the feasible directions which are determined by the labeling procedure.

Interconnection Length Distribution: Each interconnection is assumed to have a random length drawn according to an exponential distribution with parameter α. The average interconnection length is equal to $1/\alpha$ in units of channel length.

Remark: The interconnections model described is a generalization of the model in [1]. If the logic block are of equal size, then the parameter λ in [1] could be easily related to λ in this paper. Also the average interconnection length $\bar{R}$ in [1], which was measured in terms of the number of steps on a lattice, could be related to the average interconnection length $1/\alpha$. However, as will be explained in Section V, the motivating application of the model in this paper is different from that in [1], and the asymptotic types of results discussed in [1] are not of interest here.

In order to use the described model to estimate channel widths, we now define the following random variables:

1) $X_{(i,j)}^{(l,k)}$ = number of interconnections generated in channel (i, j) and entering channel (l, k) at node l. The random variable $X_{(i,j)}^{(l,k)}$ has a Poisson distribution with mean that depends on λ, α and the topology of the channel graph. For any two channels $(i, j) \neq (i', j')$, $X_{(i,j)}^{(l,k)}$ and $X_{(i',j')}^{(l,k)}$ are independent. Also, $X_{(i,j)}^{(l,k)}$ and $X_{(i,j)}^{(k,l)}$ are independent. In general, any two random variables $X_{(i,j)}^{(l,k)}$ and $X_{(i,j)}^{(l',k')}$ are independent if no interconnection originating at channel (i, j) can have a minimum distance trajectory that passes through both channels (l, k) (from $l \to k$) and (l', k') (from $l' \to k$).

2) $$X_{lk} = \sum_{\substack{\text{all channels } (i,j) \\ \text{other than } (l,k)}} X_{(i,j)}^{(l,k)}$$
$$= \text{total number of interconnections entering channel } (l, k) \text{ at } l. \quad (1)$$

Each random variable X_{lk} is the sum of independent Poisson distributed random variables. Thus X_{lk} is Poisson with parameter equal to the sum of the means of these random variables.

3) Z_{lk} = the number of interconnections generated in channel (l, k) and that leave the channel from node k. This excludes interconnections that are generated in (l, k) but end within the channel and the interconnections that leave the channel from node l. The random variables Z_{lk} are independent and Poisson distributed.

Now suppose that the means for the random variables X_{lk}, X_{kl}, Z_{lk}, and Z_{kl} are given. How does one estimate the width of channel (l, k)?

A Diffusion Model for Routing Channels: Consider a channel (l, k) with length l_{lk}, and define:

1) $X_{lk}(t)$, $0 \leq t \leq l_{lk}$ to be the number of interconnections at t that have entered the channel at l. Thus $X_{lk}(0) = X_{lk}$ (the left or lower node of any channel is considered at $t=0$). For any $0 \leq t \leq l_{lk}$ the random variable $X_{lk}(t)$ is Poisson distributed with parameter $= \bar{X}_{lk} \cdot e^{-\alpha t}$, where $\bar{X}_{lk}$ is the mean of X_{lk}. Similarly, define $X_{kl}(t)$, $0 \leq t \leq l_{lk}$ to be the number of interconnections at t that have entered the channel at node k. The random variable $X_{kl}(t)$ is Poisson distributed with parameter $= \bar{X}_{kl} \cdot e^{-\alpha(l_{lk}-t)}$, where $\bar{X}_{kl} = EX_{kl}$. Also, $X_{kl}(l_{lk}) = X_{kl}$.

2) Let $Z_{lk}(t)$, $0 \leq t \leq l_{lk}$ be the number of interconnections at t that have originated in the interval $(0, t)$ and that move towards k. Assume that interconnections originate uniformly over $(0, t)$ and that any interconnection originating at $\tau \in (0, t)$ chooses to move toward k with probability p_{lk}. Since the interconnection length is exponentially distributed, the probability that an interconnection starting at $\tau \in (0, t)$ have length greater than $(t-\tau)$ is $e^{-\alpha(t-\tau)}$. It then follows that

$$\bar{Z}_{lk}(t) = EZ_{lk}(t) = \int_0^t \lambda p_{lk} e^{-\alpha(t-\tau)} d\tau$$
$$= \frac{\lambda p_{lk}}{\alpha}(1 - e^{-\alpha t}). \quad (2)$$

Similarly,

$$\bar{Z}_{kl}(t) = EZ_{kl}(t) = \int_t^{l_{lk}} \lambda(1 - p_{lk}) e^{-\alpha(\tau - t)} d\tau$$
$$= \frac{\lambda(1 - p_{lk})}{\alpha}(1 - e^{-\alpha(l_{lk}-t)}) \quad (3)$$

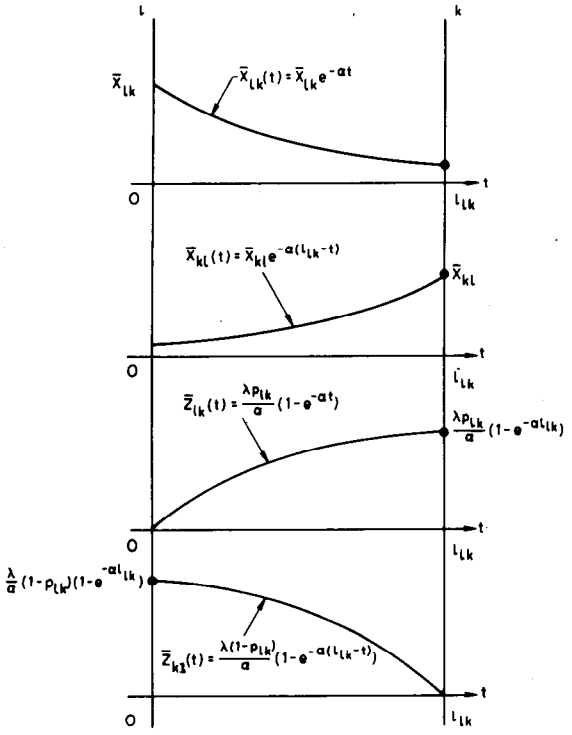


Fig. 4. Plot of $\bar{X}_{lk}(t)$, $\bar{X}_{kl}(t)$, $\bar{Z}_{lk}(t)$, and $\bar{Z}_{kl}(t)$.

$\bar{X}_{lk}(t)$, $\bar{X}_{kl}(t)$, $\bar{Z}_{lk}(t)$, and $\bar{Z}_{kl}(t)$ are sketched in Fig. 4, as a function of t. The width of the channel at t can now be defined as

$$W_{lk}(t)=X_{lk}(t)+X_{kl}(t)+Z_{lk}(t)+Z_{kl}(t). \tag{4}$$

$W_{lk}(t)$ is the sum of independent Poisson random variables. Thus $W_{lk}(t)$ is Poisson with mean

$$EW_{lk}(t)=\bar{X}_{lk}e^{-\alpha t}+\bar{X}_{kl}e^{-\alpha(l_{lk}-t)}+\frac{\lambda p_{lk}}{\alpha}(1-e^{-\alpha t})+\frac{\lambda(1-p_{lk})}{\alpha}(1-e^{-\alpha(l_{lk}-t)}). \tag{5}$$

The average width of channel (l, k) denoted as $\overline{W}_{lk}$ can be easily obtained by finding the maximum of (5) in t.

III. An Algorithm for Estimating Channel Widths

We now provide a simple algorithm for estimating $\overline{W}_{lk}$. This algorithm consists of the following two main steps:

1) Use Algorithm 1 below to compute $\bar{X}_{lk}$ and $\bar{X}_{kl}$, for all channels $(1, k)$.

2) Evaluate

$$\overline{W}_{lk}=\max_{0\leq t\leq l_{lk}} EW_{lk}(t), \qquad \text{for all channels } (l,k). \tag{6}$$

Step 2 involves simple computation based on (5) and (6). Thus no further details are presented for this step.

Algorithm 1:

A) For each channel (i, j)

1) Find the shortest paths to (l, k), for all channels $(l, k)\neq(i, j)$. This is done by labeling the channel graph using a shortest path algorithm, such as Algorithm B in [2]. Instead of labeling the nodes, however, the edges are labeled. The label w_{lk} attached to edge (l, k) is the shortest distance from (i, j) to node l or k whichever is farthest from (i, j). A path $p=\{p_1, p_2, \cdots, p_m\}$ is a shortest distance path from node p_1 to p_m if

$$w_{p_ip_{i+1}}=w_{p_{i-1}p_i}+l_{p_ip_{i+1}}, \qquad \forall\, 2\leq i\leq m-1.$$

2) Using Algorithm 1.1, obtain, for all channels $(l, k)\neq(i, j)$

$$\bar{X}_{(i,j)}^{(l,k)} \text{ and } \bar{X}_{(i,j)}^{(k,l)}$$

where

$$\bar{X}_{(i,j)}^{(l,k)}=EX_{(i,j)}^{(l,k)}.$$

B) Evaluate, for all channels (l, k)

$$\bar{X}_{lk}=\sum_{\forall(i,j)\neq(l,k)}\bar{X}_{(i,j)}^{(l,k)} \text{ and}$$

$$\bar{X}_{kl}=\sum_{\forall(i,j)\neq(l,k)}\bar{X}_{(i,j)}^{(k,l)}.$$

Step B above is actually accomplished in Step A.2 by accumulating

$$\bar{X}_{lk}\leftarrow\bar{X}_{lk}+\bar{X}_{(i,j)}^{(l,k)}, \qquad \forall(l,k)\neq(i,j)$$

everytime a new channel (i, j) is processed.

Algorithm 1.1: (For channel (i, j))

1) From (2) and (3) obtain $\bar{Z}_{ij}(l_{ij})$ and $\bar{Z}_{ji}(0)$. The probability $p_{ij}=n_j/(n_i+n_j)$ where n_i is the number of edges connected to node i excluding (i, j), and $n_j=$number of edges connected to j excluding (i, j).

2) $\bar{X}_{(i,j)}^{(l,k)}\leftarrow\bar{Z}_{ji}(0)/n_i$ for all edges (l, k) connected to node i excluding (i, j).

$\bar{X}_{(i,j)}^{(l,k)}\leftarrow\bar{Z}_{ij}(l_{ij})/n_j$ for all edges (l, k) connected to node j excluding (i, j).

3) For each channel (l, k) connected to channel (i, j)

$$\bar{X}_{lk}\leftarrow\bar{X}_{lk}+\bar{X}_{(i,j)}^{(l,k)}.$$

Insert $\{s,(l, k)\}$ at the bottom of a circular queue Q, where s is the side l or k from which the interconnections generated in channel (i, j) enter (l, k).

4) While $Q\neq\phi$ do

5) Begin

6) Select element $\{s,(l, k)\}$ from the top of Q

7) Let s correspond to side l and let l be the lower or left node of channel (l, k). Then

$$\bar{X}_{(i,j)}^{(l,k)}(l_{lk})\leftarrow\bar{X}_{(i,j)}^{(l,k)}(0)e^{-\alpha l_{lk}}$$

$\bar{X}_{(i,j)}^{(l,k)}(t)$denotes the average number of interconnections at t; $0\leq t\leq l_{lk}$ generated in channel (i, j) and entered channel (l, k) at l.

$$\bar{X}_{(i,j)}^{(l,k)}(0)=\bar{X}_{(i,j)}^{(l,k)} \text{ and } \bar{X}_{(i,j)}^{(l,k)}(l_{lk})=\bar{X}_{(i,j)}^{(k,l)}$$

8) For all channels (k, p), $p\neq l$ using the labeling done in Step A.1 of Algorithm 1, determine

(a) $m=$number of channels for which $w_{kp}=w_{lk}+l_{kp}$. We denote these as *shortest path channels.*

(b) $m^*=$number of *dead end channels*$=$number of channels (out of m found above) (k, p) for which there is no channel (p, q) such that $w_{pq}=w_{kp}+l_{pq}$. The remaining $m-m^*$ channels are called *through channels.*

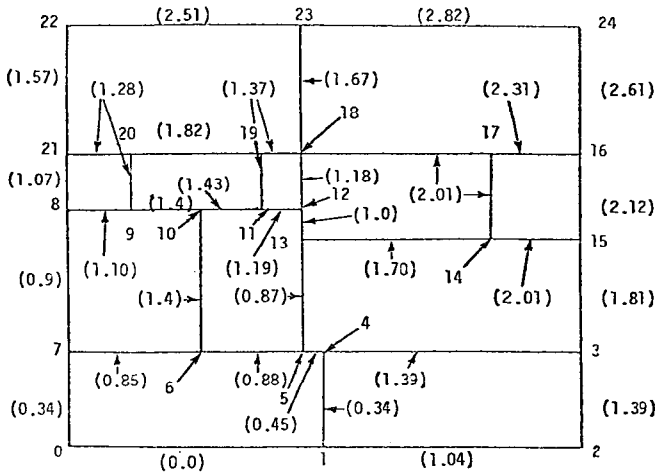


Fig. 5. Labeling of the channel graph starting from channel (0, 1).

9) If $m^*=0$ or $m=m^*$ then

$\bar{X}_{(i,j)}^{(k,p)} \leftarrow \bar{X}_{lk}(l_{lk})/m$, for all m shortest path channels

10) Else
11) Begin
12) $\bar{X}_{(i,j)}^{(k,p)} \leftarrow \bar{X}_{lk}(l_{lk})/c$, for all dead end channels
13) $\bar{X}_{(i,j)}^{(k,p)} \leftarrow \bar{X}_{lk}(l_{lk})(1-m^*/c)/(m-m^*)$, for all through channels where c is a constant
14) End
15) $\bar{X}_{kp} \leftarrow \bar{X}_{kp} + \bar{X}_{(i,j)}^{(k,p)}$, for all shortest paths channels
16) For all through channels (k, p)
17) Insert $\{k,(k, p)\}$ at bottom of Q
18) Delete $\{s,(l, k)\}$ from top of Q
19) End.

Remark: In steps 7–17 s was taken to be the lower or left node l of channel (l, k). Similar computations are done for $s=k$.

The above algorithm contains the most important steps necessary to compute $\bar{W}_{lk}$. We, therefore, give some numerical examples to help explain how it is used to compute $\bar{X}_{(i,j)}^{(l,k)}$ and $\bar{X}_{(i,j)}^{(k,l)}$.

We consider $(i, j)=(0,1)$. The channel graph is first labeled using the shortest path algorithm (see Fig. 5). The needed channel lengths are given in Table I.

Now $\bar{Z}_{10}(0)$ and $\bar{Z}_{01}(l_{01})$ are computed as stated in Steps 1 and 2 of Algorithm 1.2.

Let $\lambda=6.198$ pins per mm

$\alpha=2.385$ per mm

$l_{01}=1.04$ mm, and $p_{01}=2/3$. From (2) and (3) we obtain

$$\bar{Z}_{10}(0)=\frac{\lambda(1-p_{01})}{\alpha}(1-e^{-\alpha l_{01}})$$
$$=0.794$$

and

$$\bar{Z}_{01}(l_{01})=\frac{\lambda p_{01}}{\alpha}(1-e^{-\alpha l_{01}})$$
$$=1.589.$$

Hence

$$\bar{X}_{(0,1)}^{(0,7)}=\bar{Z}_{10}(0)$$
$$=0.794$$

and

$$\bar{X}_{(0,1)}^{(1,2)}=\bar{X}_{(0,1)}^{(1,7)}=Z_{01}(l_{01})/2.$$
$$=0.794.$$

TABLE I
ESTIMATED VERSUS MEASURED CHANNEL WIDTHS

CH	LENGTH (mm)	$\bar{W}_{ij}$	$\bar{W}_{ij}$	W_{ij}	$\frac{\lvert W_{ij}-\bar{W}_{ij}\rvert}{\sqrt{\bar{W}_{ij}}}$	$\frac{\lvert W_{ij}-\bar{W}_{ij}\rvert}{\sqrt{\bar{W}_{ij}}}$
(0,1)	1.045	3.64	3.13	2	0.86	0.64
(0,7)	0.343	3.36	3.73	1	1.29	1.42
(7,6)	0.507	3.06	2.96	3	0.03	0.02
(6,5)	0.433	3.00	2.46	4	0.58	0.98
(5,4)	0.105	2.52	2.93	4	0.26	0.62
(1,4)	0.343	3.33	2.70	1	1.28	1.03
(1,2)	1.045	3.71	3.33	2	0.89	0.73
(2,3)	0.343	3.34	2.95	3	0.18	0.03
(4,3)	1.045	2.85	2.52	4	0.68	0.93
(3,15)	0.418	3.51	3.33	3	0.27	0.18
(14,15)	0.313	2.85	2.98	3	0.09	0.01
(13,14)	0.836	2.92	2.93	3	0.05	0.04
(5,13)	0.418	3.14	2.78	4	0.48	0.73
(13,12)	0.134	3.22	3.45	2	0.68	0.78
(11,12)	0.194	2.72	3.50	5	1.38	0.80
(10,11)	0.239	3.21	4.56	6	1.56	0.67
(6,10)	0.552	2.89	3.62	2	0.52	0.85
(9,10)	0.299	2.98	3.93	7	2.33	1.55
(8,9)	0.209	2.79	3.83	4	0.72	0.86
(7,8)	0.552	3.34	4.39	3	0.19	0.66
(8,21)	0.179	2.92	3.15	3	0.05	0.08
(21,20)	0.209	2.93	3.85	3	0.04	0.43
(9,20)	0.179	2.47	3.35	2	0.30	0.74
(20,19)	0.537	2.92	5.11	5	1.22	0.05
(11,19)	0.179	2.49	3.33	1	0.94	1.28
(19,18)	0.194	3.27	4.00	3	0.15	0.50
(12,18)	0.179	2.94	3.15	4	0.62	0.48
(18,17)	0.836	3.33	2.79	4	0.37	0.73
(14,17)	0.313	2.64	2.92	3	0.22	0.05
(17,16)	0.313	2.82	2.07	2	0.49	0.05
(15,16)	0.313	2.94	2.59	5	1.20	1.49
(16,24)	0.493	3.37	2.44	5	0.89	1.64
(23,24)	1.149	3.61	1.90	3	0.32	0.80
(18,23)	0.493	3.59	2.85	3	0.31	0.09
(22,23)	0.940	3.56	1.95	2	0.83	0.04
(21,22)	0.493	3.39	2.68	3	0.21	0.19

W_{ij} = Measured width of channel (i, j) in terms of number of tracks.
$\bar{W}_{ij}$ = Estimated width of channel (i, j) in terms of number of tracks when each channel has an individual generation rate λ_{ij}.

We now consider computations for the intermediate channel (0,7) (i.e., Step 7 of Algorithm 1.2). Using the diffusion model presented in Section II, the expected number of interconnections leaving node 7 is given by

$$\bar{X}^{(0,7)}_{(0,1)}(l_{07}) = \bar{X}^{(0,7)}_{(0,1)} e^{-\alpha l_{07}} = 0.35.$$

Each interconnection, leaving node 7, chooses one of the two available channels (7,6) and (7,8) with probability 1/2. Therefore,

$$\bar{X}^{(7,6)}_{(0,1)} = \bar{X}^{(7,8)}_{(0,1)} = 0.175.$$

Channels (5,6) and (6,10) are examples of what we call dead end channels. This can be seen for channel (6,10) by investigating the labels of channels (6,10), (9,10) and (10,11). Note that $w_{9\,10} \neq w_{6\,10} + l_{9\,10}$ and $w_{10\,11} \neq w_{6\,10} + l_{10\,11}$. Similarly for channel (5,6), $w_{6\,10} \neq w_{6\,5} + l_{6\,10}$ and $w_{7\,6} \neq w_{6\,5} + l_{7\,6}$ (refer to Step 8.b of Algorithm 1.2).

Since channel (5,6) is a dead end, we assume that an interconnection leaving node 5 will enter channel (5,6) with a small probability $1/c$. For example, with

$\bar{X}^{(5,4)}_{(0,1)}(0) = 0.136$ and $c = 100$, it follows that

$$\bar{X}^{(5,6)}_{(0,1)} = \bar{X}^{(4,5)}_{(0,1)}/100 = 1.364 \times 10^{-3} \text{ and}$$

$$\bar{X}^{(5,13)}_{(0,1)} = \bar{X}^{(4,5)}_{(0,1)}(1 - 1/100) = 0.135.$$

IV. An Example

We now apply our algorithm to estimate channel widths for the chip in Fig. 1. The layout of this chip is due to Preas [3].

Following are the major characteristics of the chip:

1) Number of devices: 1127.
2) Metal to metal spacing: 18 μm
   Polysilicon to polysilicon spacing: 23 μm
3) Dimensions (Die size: from scribe line to scribe line):

   Area = Length × Height

   = 3.025 by 2.575 = 7.79 mm^2.

4) Parameters λ and α:

   The parameters obtained from this chip are

   $\lambda = 6.198$ pins per mm

   $\alpha = 2.385$ per mm.

First we test our Poisson assumption of interconnections generation, and the exponential assumption for interconnections length.

The parameter λ is estimated by simply dividing the number of pins P by twice the total length of the channels. Thus

$$\lambda = P \Big/ \left(2 \sum_{(i,j)} l_{ij} \right). \tag{7}$$

Remark: It has been pointed out by one of the referees that (7) may not be a good estimate for λ when a sizable fraction of the interconnections connect more than two pins. He suggested the following generalization:

$$\lambda = (P - N) \Big/ \sum_{\forall (i,j)} l_{ij} \tag{7'}$$

where N = number of interconnections.

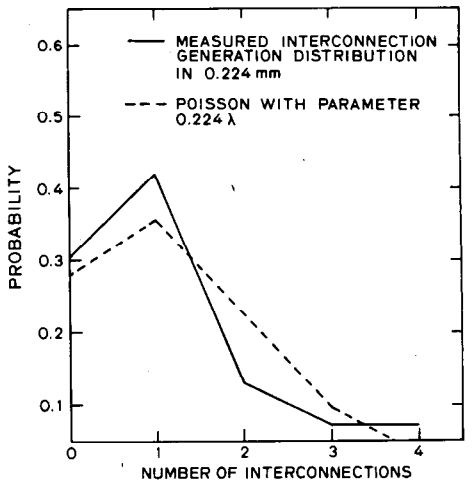


Fig. 6. Interconnection generation distribution.

The measured (solid curve) and the model predicted (broken curve) distributions for the number of interconnections generated in 0.224 mm of channel length are shown in Fig. 6. The model predicted curve is obtained by plotting

$$\frac{(0.224\lambda)^k}{k!} e^{-0.224\lambda}$$

while the measured curve is obtained by dividing the channels into subintervals of length 0.224 mm and counting the number of pins in each subinterval. The measured mean of this sample is 1.188, and the measured variance is 1.341 which is within the 95-percent confidence interval of the hypothesized variance.

Model predicted (broken curve) and measured (solid curve) cumulative interconnection length distributions are given in Fig. 7. The model predicted curve is a plot of $1 - e^{-\alpha l}$, whereas the measured curve is derived from a histogram obtained by measuring the lengths of all wires. The parameter α is determined by computing the reciprocal of the sample average of all wire lengths.

In Sections II and III, we have assumed that interconnection generation rate λ is uniform over the channel graph. To obtain better channel width estimates (at the expense of increased storage requirement) we also consider the case when each channel (i, j) is assigned an individual interconnection generation rate λ_{ij} = half number of pins in (i, j). Thus for every channel (i, j), λ in (2) and (3) is replaced by λ_{ij}.

Table I gives the actual and estimated channel widths (units used are number of tracks) for both the individual and the average generation rate case. The last two columns show the absolute deviation of the estimated from the

TABLE II
COMPARISON OF MEASURED AND ESTIMATED CHIP DIMENSIONS

Chip Dimension	Actual	Estimated	
		Average λ	λ_{ij}
Width mm	2.287	2.326	2.313
Height mm	1.946	1.967	1.899

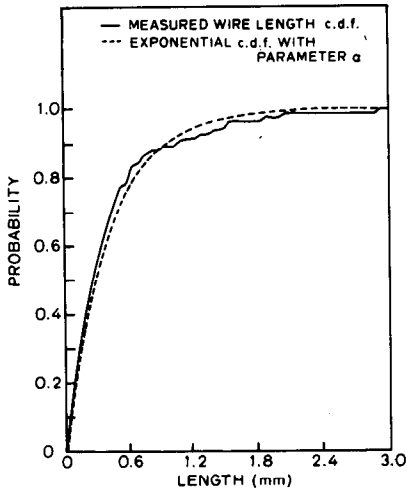


Fig. 7. Interconnections length distribution.

actual widths in units of standard deviations. Table II compares the estimated and the actual dimensions of the bounding rectangle (the smallest length and height rectangle that can enclose the logic blocks and their interconnections).

V. APPLICATIONS TO LAYOUT

We have described in this paper and in [1] statistical models for interconnections in large integrated circuits. We have demonstrated that using two parameters; namely average interconnection length and average wire generation, as well as certain simple rules for interconnection trajectories, it is possible to obtain satisfactory and quick estimates of chip area and routing channel widths. We now discuss some applications of these results to the layout of complex integrated circuits.

The application of the model in [1] to master slice layout is well known [4]. A good prediction of the maximum width of routing channels is used to determine the dimensions of a "universal" two-dimensional $N \times N$ array of logic blocks. This array could then be routed in different ways to construct different computing machines. One advantage of this design methodology is the direct applicability of many placement and routing algorithms that have been developed for printed circuit boards. However, two serious problems are frequently encountered. The first is the failure to complete the routing due to unavoidable routing congestions. This could turn the layout process into a "computational nightmare." The second problem is that of inefficient utilization of chip area which is caused by the rigid master slice layout.

One way to overcome the master slice layout problems is by tailoring (customizing) the layout to the specific design. First, decompose the design into reasonable size logic blocks. Find compact layouts for them. Then tightly place and interconnect the blocks to yield the final layout. The statistical model described in this paper can be used to assist this layout process as follows:

(i) *Chip Planning:* Given the relative positions of the logic blocks, in an early stage of the chip design, one could use the statistical models to predict the chip area.

(ii) *Comparison of Placements:* Given several feasible placements of the logic blocks, the statistical models could be used to determine which placement (or placements) yields the smallest overall area.

(iii) *Routing:* Given a placement of the logic blocks how should the routing be done? One obvious routing strategy would be to assign arbitrarily large widths to the routing channels. But then we may either waste too much area, or run into routing difficulties, or both. An alternative routing methodology would be to constantly change the placement each time an interconnection or a group of interconnections is routed. This is likely (but not guaranteed) to produce a good compacted layout, but at the expense of excessive computations. A third routing strategy which was given in [3] is to first estimate the widths of the channels, do partial routing, then modify the placement and so on until routing is completed satisfactorily. Unfortunately, the proposed methods in [3] for estimating channel widths (in addition to the attempt to minimize total chip area) require too much computing time.

One way of improving the computational efficiency of the routing algorithm described in [3] would be to use the statistical estimates of channel widths, at least as an initial step. Other improvements are also possible, but will not be discussed here.

A Remark on Estimating λ and α: For the above applications it is assumed that the model parameters λ and α are known. These parameters can be estimated as follows: The number of interconnections generated per unit channel length λ is simply estimated by dividing the total number of pins by twice the sum of the lengths of the channels (as in Section IV). The average interconnection length $1/\alpha$ could be estimated by routing a randomly chosen sample of interconnections and finding the sample average of their lengths.

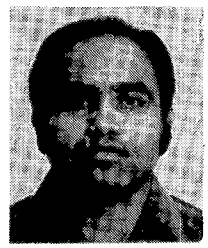
REFERENCES

[1] A. El Gamal, "Two-dimensional stochastic model for interconnections in master slice integrated circuits," *IEEE Trans. Circuits Syst.*, vol. CAS-28, pp. 127–138, Feb. 1981.
[2] D. G. Johnson, "Efficient algorithms for shortest paths in sparse networks," *J. Ass. Comput. Mach.*, vol. 24, no. 1, pp. 1–13, Jan. 1977.
[3] B. T. Preas, "Placement and routing algorithms for hierarchical integrated circuit layout," Tech. Rep. 180, Computer System Lab., Stanford Univ., Stanford, CA, Aug. 1979.
[4] W. Heller, W. Mikhail, and W. Donath, "Prediction of wiring space requirements for LSI," in *Design Automation & Fault-Tolerance Computing*, 1978, pp. 117–144.

Abbas A. El Gamal (S'71–M'78), for a photograph and biography please see page 138 of the February 1981 issue of this TRANSACTIONS.

✤

Zahir A. Syed (S'75) was born in Karachi, Pakistan, on December 25, 1949. He received the B.S. degree from Karachi University in 1974 and the M.S. degree from Massachusetts Institute of Technology, Cambridge, in 1978; both in electrical engineering. He is currently a candidate for the



Ph.D. degree in electrical engineering at University of Southern California, Los Angeles.

From 1974 to 1975 he was Lecturer in the faculty of Electrical Engineering at Karachi University, Pakistan. During 1978 and 1979 he was a Research Assistant at the University of Southern California, University Park, Los Angeles. Currently he is a Research Associate at the Stanford Information Systems Laboratory. His research area is Design Automation, specially, layout of integrated circuits.

Controlled Rounding Arithmetics, for Second-Order Direct-Form Digital Filters, that Eliminate All Self-Sustained Oscillations

DEBASIS MITRA, MEMBER, IEEE, AND VICTOR B. LAWRENCE, MEMBER, IEEE

Abstract— **Quantization often allows a recursive second-order filter to oscillate even when the underlying linear model is absolutely stable and there is no external signal present to excite the filter. One of the significant innovative ideas recently introduced to combat this condition is "controlled rounding" which utilizes memory in the rounding decision process. Specifically, the selection at time $(n+1)$ of the state variable of filter, x_{n+1}, depends not only on the quantity to be rounded, but also on the two previous outputs of the rounding operation, x_n and x_{n-1}; the latter two quantities are anyhow readily available at the time of decision.**

This paper gives controlled rounding arithmetics that suppress all self-sustained oscillations in all direct-form second-order filters for which the underlying ideal linear model is stable. Not just rounding but also overflow, and hence all the features that make a digital filter a finite state machine, are taken into account. The incremental cost of the hardware which is mostly on account of the additional logic is reckoned to be slight.

I. INTRODUCTION

THE FACT that a digital filter is a finite state machine is the common cause for a variety of discrepancies between the filter behavior and that of the underlying ideal linear model which the filter is intended to imitate. One of these forms of distortion is of primary concern here and it is the widely studied topic of self-sustained oscillations in the basic modular unit of most present day digital filters, the direct-form second-order digital filter [1]–[7] shown in Fig. 1. Even when the filter is intended to be idle,[1] not filtering, the compounding of

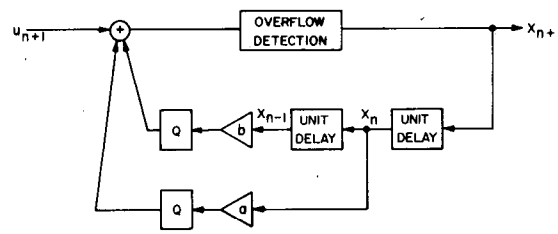


Fig. 1. Schematic of the recursive part of second-order direct-form digital filter.

feedback with the nonlinearity of the quantization operation often allows the filter to oscillate. Such oscillations can be a severe handicap in many applications; this is especially so in applications where the idle channel noise is a source of great concern. Therefore, the suppression of self-sustained oscillations without incurring substantial disadvantages in other respects is important.

Recently various significant innovative ideas have been introduced to combat this condition [3]–[11]. The one that concerns us directly is the notion of controlled rounding [3]. In controlled rounding the decision involved in rounding utilizes memory. Specifically, (see Fig. 2) the selection at time $(n+1)$ of the value for x_{n+1} depends not only on the quantity that is to be rounded, but also on the two previous outputs of the rounding operation, x_n and x_{n-1}; the latter two quantities are anyhow readily available at the time of decision $(n+1)$. It is fortuitous that recent innovations in hardware make the acceptance of controlled rounding more attractive now than in the past. The partic-

Manuscript received April 26, 1978; revised February 18, 1981.
D. Mitra is with Bell Laboratories, Murray Hill, NJ 07974.
V. B. Lawrence is with Bell Laboratories, Holmdel, NJ 07733.

[1] That is, $u_n \equiv 0$ in Fig. 1. We call oscillations in such conditions self-sustained oscillations.

0098-4094/81/0900-0894$00.75