Area Estimation for DSP Algorithms

Khaled M. Elleithy
Computer Sc. & Eng. Department
University of Bridgeport
Bridgeport, CT 06601
elleithy@bridgeport.edu

Alaa A. Amin
Computer Eng. Department
King Fahd University
Dhahran 31261, Saudi Arabia
amin@ccse.kfupm.edu.sa

Abstract-- In this paper we present a method to estimate the layout area of DSP algorithms that are designed using the standard cell methodology. The circuit description is given as a netlist of standard cell library modules. The area occupied by the circuit can be estimated prior to the actual layout phase. Area estimation before final layout is important for design evaluation and for the prediction of the chip floorplan.

THE MATHEMATICAL FRAMEWORK

Constructive and analytical approaches can be used to estimate the area occupied by a standard cell design. In constructive approach, the design is modeled as a data structure on which partitioning algorithms are applied to estimate the total area. In analytical approach, probabilistic models for the behavior of the wiring different between cells are used [1-2]. Analytical approach is much faster in terms of computation requirement and easier to program. The analytical model used in this work is an extension of the work in [2]. For the wire average length estimation, there are also constructive and analytical approaches for its estimation. For the analytical case the estimation of the average wire-length [3] and its distribution have been published [4]. For the constructive case there also published results [5]. For this work, analytical approach for the wire length estimation is also used.

Earlier work on area estimation has been done through analytical, nondeterministic models. One of the early attempts was the observation of E. F. Rent of IBM who developed an empirical formula, which relates the number of components in partitioned subcircuit to the number of external components of that subcircuit. The relation is known as Rent’s rule [6] and is given by:

\[ P = KB^r \]  

where \( P \) is the number of external connections to the subcircuit, \( K \) is the average number of pins per block, and \( r \) is Rent’s constant and \( B \) is the number of components in the subcircuit.

Heller et al. [7], developed a stochastic model for the prediction of the wiring space needed for a 1-D placement of cells. The model predicts the successful routability of the placement. They proposed a heuristic to extend the model for 2-D placement.

Donath [8] has shown from simple theoretical considerations that the distribution \( f_k \) of the wire lengths for a good two-dimensional placement on a square Manhattan grid is of the form:
where $\gamma$ is related to the Rent partitioning exponent $p$ given by the relation
\[ 2p + \gamma = 3. \]

Feuer [3] predicted the wire length distribution and estimated the average wire length in an IC laid out with Standard Cell approach. The main results of his work is that if the partitioning of a logic graph follows Rent's rule, then the average wire length distribution takes the forms:

\[ \bar{L}_i = \sqrt{2} \frac{\alpha (5-\alpha)}{(3-\alpha)(4-\alpha)} \frac{C^{5/2}}{(1-C^{\alpha-1})} \]
\[ \bar{L}_e = \sqrt{2} \frac{(1-\alpha)(5-\alpha)}{(3-\alpha)(4-\alpha)} \frac{C^{5/2}}{C^2} \]

where $p$ is Rent's constant and $C$ is the number of components in a subcircuit contained in a region of radius $R$ and $\alpha = 2(1-p)$. $L_i$ and $L_e$ are the average wire lengths for internal and external connections respectively. The average wire length including both internal and external connections is given by:

\[ \bar{L} = \sqrt{2} \frac{(2-\alpha)(5-\alpha)}{(3-\alpha)(4-\alpha)} \frac{C^{5/2}}{(1+C^{\alpha-1})} \]

El-Gamal [1] has extended the work of Heller et al., to accommodate 2-D gate arrays. A two-dimensional stochastic model for gate array channel wiring is studied. The gate array is modeled as a 2-D lattice. Wire segments emerge from lattice points and assumed to follow a Poisson distribution of the form:

\[ P(X=k) = \frac{\lambda^k e^{-\lambda}}{k!}, \quad k = 0, 1, 2 \ldots \]

where $X$ is the number of wires emanating from a grid point, $\lambda$ is the Poisson parameter and $k$ is the number of wires emanating from a grid point. Wires are assumed to travel either horizontally or vertically with no jogging or diagonal runs. The direction taken by each wire is determined by flipping a fair coin. Many wire length distributions such as Poisson, exponential and geometric have been suggested in the literature.

Sastry [9] proved that the "ideal" distribution in the limiting case of optimum placement is the exponential distribution in the continuous domain. Since the geometric distribution is the counter part of the exponential distribution in the discrete case:

\[ P(L=l) = pq^{l-1}, \text{ with } 1/p = \bar{L} \text{ and } q = 1-p \]

where $\bar{L}$ is the average wire length. Based on these assumptions the upper bound on the channel density is:

\[ \bar{W} = \frac{\bar{L} \lambda}{2} \]

where $\bar{W}$ is the average wire length and $\lambda$ is the parameter for the Poisson distribution.
Sastry [4] modeled a gate array as a grid of channel intersection and the problem of wiring space requirement estimation is reduced to the estimation of the dimension of these intersections. This was achieved by classifying the wires at an intersection into six different types, each modeled as a Poisson distributed random variable. He also established an equivalence relation between Rent’s rule and the wire length. The distribution was found to be of Weibull family given by:

\[ E = \frac{1}{p} \left( \frac{1}{\gamma} \right)^{1/p} \Gamma \left( \frac{1}{p} \right) \]

where \( p \) is Rent’s constant, \( \gamma \) is the average number of pins per cell, and \( \Gamma(x+1) = x! \) is the gamma function. The asymptotic channel width for \( N \to \infty \), \( (N \) is the number of logic modules in the array) was found to be:

\[ W = 2\mu + \sqrt{\frac{\mu}{\pi}} \]

where \( \mu = \frac{\gamma}{3} (E[L^2] - 1) \) and \( E[L^2] \) is the second moment of the wire length random variable \( L \). The routability is defined as the probability that neither the vertical nor the horizontal channel width requirements are exceeded. These statistically dependent events are approximated events modeled by two independent events and an asymptotic probability is derived for each of them. The vertical routability is found to be:

\[ R_v = \frac{1}{2\pi \sqrt{3}} \int \int e^{-\frac{1}{2\mu}(x^2+y^2)} dx dy \]

where \( a = \frac{w - 2\mu}{\sqrt{2\mu}} \).

The horizontal routability is symmetrically defined.

Kurdahi [2] proposed a model for standard cell area estimation, which is accurate within 10 percent of the actual areas of designs.

**AREA ESTIMATION FOR STANDARD CELL LAYOUTS**

The estimation of the layout area occupied by standard cell design is composed of three parts:

1. The active area which is readily known from the netlist description of the circuit.
2. The area of the pads which are also known from the circuit description.
3. The area of the interconnect, which is not trivial and needs a probabilistic modeling approach to be estimated.

**Placement**

The standard cell layout consists of two phases:

1. **Partitioning** the design by assigning cells to row.
2. Placement by assigning locations to cells.

There are two phases in placement:

- a) Initial placement phase.
- b) Iterative improvement phase.

Examples of the placement techniques such as row assignment, pair linking, clustering, force directed, bipartitioning and row folding are reported in literature. In this work, row folding is used for area estimation. The placement is done as follows:

1. Perform 1-D placement as an initial placement such that the average wire length is minimized.
2. Fold the single row placement such that area and shape constraints are satisfied.

The model assumptions are as follows:

- Cells are placed in rows of roughly the same size.
- Equivalent pins are available on both sides of each row.
- All cells have equal distance between two consecutive equivalent pins or pin pitch.
- The cell width is a multiple of pin pitches.
- All nets are two-pin nets. An n-pin net can be decomposed in an n-1 two pin nets or an appropriate wire length distribution is used such as negative binomial.
- The total number of cells in the block is $N$.
- The total number of pin slots in the block is $w$.
- Placement is done using the 1-D place and fold technique.

The Single Row Case

The following assumptions are done for single row placement:

- Pin slots are numbered 1, 2, 3, ..., $w$ from left to right.
- Wires travel from left to right.
- The length of a wire is assumed to be a random variable $L$, with geometric probability density function $p_L(l) = Pr\{L=l\} = pq^{l-1}$.

The density at point $x$, $d(x)$ is defined as the number of wires crossing a vertical cutline. The expected value of the density function is given by

$$E(d(x)) = \frac{N}{wpq} (1 - q^x)(1 - q^{w-x+1}) \text{ (13)}$$

The maximum local density occurs at $x_{\text{max}} = \left\lfloor \frac{w + 1}{2} \right\rfloor$.

The Multiple Row Model

The single row model is extended to multiple row placement by folding the 1-D placement to $n$ row 2-D placement. The initial row is “snacked” into $n$ rows. The number of pin slots per row is $r = \frac{w}{n}$.

The average value, $E(W,x)$, of track density at $x$ is given by:
For \( n \) even,
\[
E(W_n(x)) = \frac{1}{w p q (1 - q^{-2r})} \left[ 2(1 - q^w)(1 - q^{2r-2+2s+2}) - q^w + q^{2r-2+2s+2} \right] \\
(1 - q^{2s})(1 - q^{2r-2+2s+2}) \left[ n - 2q^{2r} \frac{1 - q^{-2r}}{1 - q^{2r}} \right] 
\]

For \( n \) odd,
\[
E(W_n(x)) = \frac{1}{w p q (1 - q^{-2r})} \left[ 2(1 - q^{w-1})(1 - q^{r-1}) - q^w + q^{r-1} \right] \\
(1 - q^{2s})(1 - q^{2r-2+2s+2}) \left[ n - 2q^{2r} \frac{2 - q^{-w-1} - q^{-2r}}{1 - q^{2r}} \right] 
\]

The above equations give an estimate for \( d_n(x) \) at a cutline \( x \). The track requirement is approximated by:
\[
\max_{1 \leq s \leq r} \{ E[W_n(x)] \} 
\]

The maximum density is done by setting the derivative of \( E(W_n(x)) \) to zero and solve for \( x \). The derivation that will follow is for \( n \) even. The same steps can be followed for the case where \( n \) is odd.

\[
g(x) = \frac{d}{d(x)} [E(W_n(x))] \]

where:
\[
g(x) = k_1 q^r + k_2 q^{2r} + k_3 q^{2r-2+2s+2} + k_4 q^{2r-2+2s+2} q^r 
\]

\[
k_1 = k_2 = k_4 = k_5 = 0
\]

The solution for the equation
\[
x_{max} = \frac{\ln(\rho)}{\ln(q)} 
\]

where \( \rho \) is the root of the equation:
\[
k_1 Z^3 + k_2 Z^4 + k_3 (q^r)^2 Z^2 + Z k_4 q^2 (q^r)^2 = 0 
\]

which can be solved using Newton-Raphson (slope) search method or any other numerical method such as the Bisection method. Parameters \( k_1, k_2, k_3 \) and \( k_4 \) depend on the design for which the area has to be estimated. Once this equation is solved for \( x \), it is substituted in Equation 14 to find \( \max \{E_n(x)\} \).
Estimating the Feedthroughs Space

The Feedthroughs account for space needed in the horizontal direction. A feedthrough cell is inserted in a row when a wire runs more than one row. The estimation of the space by the vertical wires requires estimating the number of Feedthroughs in the cell rows. The expected number of feed through in row \( k \) is given by:

\[
E(FT) = \frac{N}{w} q^r (1 - q^{(n-k)r})(1 - q^{(k-1)r})
\]

The maximum number of Feedthroughs \( \max[E(FT)] \) occurs in row

\[
k_{\text{max}} = \frac{n+1}{2}
\]

The maximum number of Feedthroughs is given by:

\[
\max[E(FT)] = \frac{N}{w} q^r (1 - q^{(n-k)\max})(1 - q^{(k-1)\max})
\]

The maximum width contributed by the feedthrough cells is given by:

\[
W(FT) = FT_{\text{width}} \cdot \max[E(FT)]
\]

Where \( FT_{\text{width}} \) is the width of the feedthrough cells which are assumed to be the same for all cells.

CHIP TOTAL AREA

The chip total area is given by:

\[
\text{Chip Area} = \text{Active Area} + \text{Routing Area}
\]

The IO_Area is readily known from the circuit netlist and is not counted for in the area estimation. The chip area is given by:

\[
\text{Chip Area} = H \cdot W
\]

As an approximation the channel width and cell width are equal. In that case, the height of the chip is given by:

\[
H = d \cdot \max[E(W_x(s))] + n \cdot \text{Cell Height}
\]

where \( d \) is the track width, \( n \) the number of rows and \( \text{Cell Height} \) is the height of standard cells. The width of the chip is given by:

\[
W = W(FT) + r \cdot \text{Cell Pitch}
\]

Where \( \text{Cell Pitch} \) is the distance between two consecutive input/output pins of a cell. This distance is assumed to be constant for all cells. For a square shape: \( W = H \)

When this condition is satisfied, the number of rows is given by:

\[
n = \frac{(W(FT) - d \cdot \max[E(W_x(s))]) \pm \sqrt{(W(FT) - d \cdot \max[E(W_x(s))])^2 + 4d \cdot \text{Cell Height} \cdot \text{Cell Pitch}}}{2 \cdot \text{Cell Height}}
\]

The area of the chip is

\[
\text{Chip Area} = \{d \cdot \max[E(W_x(s))] + n \cdot \text{Cell Height}] \cdot (W(FT) + r \cdot \text{Cell Pitch})
\]
The unknown parameters in this expression are:

- $d$: The physical track width.
- $Cell\_Height$: The cell height is given in the input netlist.
- $Cell\_Pitch$: The cell pitch is given in the input netlist.
- $\text{Max}[E(W(x))]$: the maximum channel density.

The unknown parameters, which are used indirectly in the expression, are:

- $P$: Rent's constant which is assumed to take the same value in each design $P=0.65$.
- $L$: The average wire length.

$$\alpha = 2(1-P) \quad p = \frac{1}{L}, \quad q = 1-p. \quad (30)$$

- $N$: The total number of cells in the design.
- $w$: The total number of pin slots in the design.
- $n$: The number of rows in the design
- $r = \frac{w}{n}$: The number of pin slots per row.
- $C$: The total number of standard cells in the design.

**AREA ESTIMATION ALGORITHM**

The algorithm to estimate the area occupied by a design starts with extraction of the parameters specific to the design.

$\text{Extract\_Circuit\_Parameters}(C)$

**Input:** Circuit netlist $C$

**Output:** Number of cells $N$, the total number of pin slots $w$, Average wirelength $L$, $Cell\_Height$, $Cell\_Pitch$, the physical track width $d$.

1. Read the circuit netlist $C$;
2. Record the total number of cells $N$;
3. Compute the number pin slots $w$;
4. Record the cell height ($Cell\_Height$);
5. Record the cell pitch ($Cell\_Pitch$);
6. Record the physical track dimension $d$
7. Extract_Rent_Parameter($C$); /* $P$ assumed to be 0.65)
8. Calculate the average wire length $L$

Rent's parameter $P$ is assumed to be constant for all designs ($P=0.65$). For proper area estimation, a procedure Extract_Rent_Parameter is devised to extract $P$ for each design. The procedure to estimate the total area of the design is given below.
**Estimate_Area**\((N, w, \bar{L}, \text{Cell\_Pitch}, \text{Cell\_Height}, d, FT_0, W_0)\)

**Input:** Number of cells \(N\), the total number of pin slots \(w\), Average wirelength \(\bar{L}\), Cell\_Height, Cell\_Pitch, the physical track width \(d\), An initial estimate of the track density \(W_0\), and an initial estimate of the number of Feedthroughs \(FT_0\) in the layout.

<table>
<thead>
<tr>
<th>Calculate (p) and (q): /*use equation 30 */</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\max[E{W_\ell(x)}] = W_0)</td>
</tr>
<tr>
<td>(\max[E{F_{\ell,\ell}}] = FT_0)</td>
</tr>
<tr>
<td>Compute the initial Number of rows /*Equation 29 */</td>
</tr>
<tr>
<td>/*The Objective is to have a square layout */</td>
</tr>
<tr>
<td>While (H \neq W /*Equations 26 and 27 */</td>
</tr>
<tr>
<td>Find (x_{\text{max}}: /*equation 17 */</td>
</tr>
<tr>
<td>Find (\max[E{W_\ell(x)}] /<em>equation 14</em>/</td>
</tr>
<tr>
<td>Compute (\max[E{F_{\ell,\ell}}] /* Equation 22 */</td>
</tr>
<tr>
<td>Compute (H /*Equation 26 */</td>
</tr>
<tr>
<td>Compute (W /*Equation 27 */</td>
</tr>
<tr>
<td>end /*While */</td>
</tr>
</tbody>
</table>

**CASE STUDY**

The following algorithm for Concurrent Read Concurrent Write (CRCW) sorting of \(N\) elements is based on the idea of sorting by enumeration [10]. The position of a given element of \(S\) in the sorted sequence is determined by computing \(c_i\), the number of elements smaller than \(s_i\). If two elements \(s_i\) and \(s_j\) are equal, then \(s_i\) is taken to be the larger of the two if \(i > j\); otherwise \(s_j\) is the larger. Once all the \(c_i\) have been computed for all elements, each element \(s_i\) is placed in the \((1+c_i)^{th}\) position of the sorted sequence.

**CRCW Algorithm**

**Step 1:** for \(i=1\) to \(N\) do in parallel
    for \(j=1\) to \(N\) do in parallel
        if \((s_i > s_j)\) or \((s_i = s_j\) and \(i > j\)) then
            \(P(i,j)\) writes 1 in \(c_i\)
        else
            \(P(i,j)\) writes 0 in \(c_i\)
        end-if
    end-for
end-for

**Step 2:** for \(i=1\) to \(n\) do in parallel
    \(P(i,1)\) stores \(s_i\) in position \(1+c_i\) of \(S\)
end-for
VLSI Mapping to Event Logic

Mapping of the algorithm to event logical based on the procedure given in [11,12]. In realizing the CRCW sorting algorithm in event logic, the following assumptions are made:

1. The algorithm is realized for the case of four 4-bit numbers.
2. The inputs of the circuit are:
   - Four 4-bit numbers (Ext. <1..4>, <5..8>, <9..12>, <13..16>)
   - A global reset signal is routed to all "CLR" inputs of various elements, e.g. C-Element, G-Latch, etc.
   - An External input Request "R" latches all numbers into a set of 4-input registers (IN1 - IN4).
3. The outputs of the circuit are:
   - The sorted four numbers as stored in 4-output registers (S1 - S4). These outputs are connected to external pads.
   - A global acknowledge <Final-Ack> signal which is also connected to an external output pad.
4. The pinout of the chip is thus as follows:

<table>
<thead>
<tr>
<th>Pin</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Vcc</td>
</tr>
<tr>
<td>1</td>
<td>GND</td>
</tr>
<tr>
<td>16</td>
<td>The input four 4-bit numbers</td>
</tr>
<tr>
<td>1</td>
<td>Input Request</td>
</tr>
<tr>
<td>16</td>
<td>The sorted four output 4-bit numbers</td>
</tr>
<tr>
<td>1</td>
<td>Output Acknowledge</td>
</tr>
<tr>
<td>1</td>
<td>Global Reset Input</td>
</tr>
</tbody>
</table>

The actual implementation of the chip and the application of the estimation algorithm show realistic overall chip height and hence a better area estimate within 10% for this example and several other examples.

CONCLUSIONS

In this paper, an algorithm for estimating the area for standard cell designs is presented. The algorithm is an extension of the work presented by Kurdahi[2]. The extension presented in this paper is used to estimate the total chip area taking into consideration the maximum channel density and the maximum number of feedthroughs. We have used the average wire length model developed by Feuer [3], which is more suitable for use in standard cell design style as compared to the gate arrays average wire length model adopted in [2]. We have further developed the average track density function of Kurdahi to derive a formula for the location of the highest track density ($x_{max}$). Another formula for the number of rows in a standard cell design "n" which results in a more realistic overall chip height and hence a better area estimate was developed.
ACKNOWLEDGEMENTS

The second author would like to acknowledge the support of King Fahd University of Petroleum and Minerals.

References


