A Novel Metric for Interconnect Architecture Performance*

Parthasarathi Dasgupta‡, Andrew B. Kahng‡ ¶, and Swamy Muddu¶

‡CSE Department, UCSD, La Jolla, CA 92093-0114
¶ECE Department, UCSD, La Jolla, CA 92093-0407
partha@cs.ucsd.edu, {abk, smuddu}@ucsd.edu

Abstract

We propose a new metric for evaluation of interconnect architectures. This metric is computed by *optimal assignment* of wires from a given wire length distribution (WLD) to a given interconnect architecture (IA). This new metric, the *rank* of an IA, is a *single* number that gives the number of connections in the WLD that meet a specific target delay when embedded in the IA. A dynamic programming algorithm is presented to exactly compute the rank of an IA with respect to a given WLD within practical runtimes. We use our new IA metric to quantitatively compare impacts of geometric parameters as well as process and material technology advances. For example, we observe that 42% reduction in Miller coupling factor achieves the same rank improvement as a 38% reduction in inter-layer dielectric permittivity for a 1M gate design in the 130nm technology.

1 Introduction

Performance evaluation of interconnect architectures (IA) is typically made with respect to delay, crosstalk noise, number of interconnection layers and congestion. These factors are often studied with respect to global lines which are critical to meet performance requirements. However, such studies often fail to consider factors such as via blockage and repeater insertion in semi-global and local layers[1]. Previous IA quality measures are also typically independent of design parameters (e.g., Rent parameter or wire length distribution) and do not permit quantified comparison of different types of IA improvements (materials, dimensions, etc.)

In this paper, we propose a novel metric for interconnect architecture performance which returns a *single* number for given IA and wire length distribution (WLD). Our metric is the *rank* of the IA with respect to a WLD. The metric is computed by *optimal assignment* of connections (and repeater from a fixed repeater resource) from the WLD to the IA with the objective of maximizing the number of longest wires that meet their clock frequency dependent target delays. The rest of the paper is organized as follows. Section 2 describes the previous works on performance evaluation and optimization of IA's. Section 3 introduces the proposed metric and Section 4 gives a dynamic programming (DP) algorithm for computation of rank by optimal assignment of wires. Section 5 describes our experimental setup and results for rank-based comparison of various IA improvements. Section 6 provides conclusions and future research directions.

2 Related Works

Performance evaluation of IA's has been extensively studied in recent years. [6] presents a study of scaling interconnect parameters on delay and signal integrity. They perform delay and noise analysis of global lines over different technology nodes with varying *critical* line lengths. Similar studies in [10] and [2] study the effect of changing geometric parameters and technology constraints on delay, crosstalk and signal integrity. However, these works do not consider local and semi-global lines in performance measurement. [11] evaluates delay of global lines and prescribes design techniques for improving delay. [5] also computes the delay of global lines, but considers repeater insertion to minimize total delay. In [1] and [13], geometric parameters of interconnect lines in local, semi-global and global layers are optimized to increase routing density while minimizing area and delay. [13] gives an optimal top to bottom design methodology for minimizing delay, number of wiring layers, and area. Repeaters are inserted to minimize the delay of global lines within a given repeater area resource. In most of these recent studies, effects of via blockage and repeater insertion on design are not considered when measuring performance. In [7] and [3], these factors are shown to strongly affect the number of layers needed to achieve a given IA quality. Repeater insertion is also shown to severely limit the delay performance of an IA in [13]. We take these factors into consideration when defining a new IA metric.

3 A New Metric for Interconnect Architecture Evaluation

Ultimately, quality of an IA should reflect how well the IA allows designers to meet both performance requirements and manufacturing constraints. We seek a quality metric which is simple, efficiently computable, design-dependent, frequency-dependent and sensitive to interconnect geometric parameters as well as material properties. Given an interconnect architecture and a wire length distribution[2], our new metric determines the quality of the IA by *optimal assignment* of wires from the WLD to the IA subject to the constraints that (i) longer wires are assigned to higher layers and (ii) longer wires are buffered first to meet their target delays. The quality of an IA is determined by the rank of the first wire assignment that fails to meet its target delay (Definition 2).

Definition 1 *The rank of a wire is its index in the wire length distribution (WLD), where the wires have been arranged in order of non-increasing lengths.*

Definition 2 *The rank of an interconnect architecture α, denoted by $r(\alpha)$, is a non-negative integer, and is given by the index of the highest-rank wire in the WLD that fails to meet its target delay, within a specified repeater area budget, subject to the condition that all the wires of the WLD can be assigned in the given architecture.*

Definition 3 *An interconnect architecture α has a rank $r(\alpha) = 0$ if not all the wires in the given WLD can be assigned to its layer-pairs even without meeting the delay requirements.*

The assumptions made for rank computation are as follows. (see Figure 1)

*This work was partially supported by Cadence Design Systems, Inc. and the MARCO Gigascale Silicon Research Center. Parthasarathi Dasgupta's permanent address: Dept. of Management Information Systems, Indian Institute of Management. Email: partha@iimcal.ac.in

[1] Via blockage effect decreases the total wiring area available in a layer. Repeater insertion in global layers increases the via blockage in local layers. These two effects thus increase the number of layers required for routing a design and must be taken into account during IA evaluation.

[2] The WLD used in this study is the stochastic wire length distribution of [4].

- The given IA is characterized by layer-pairs. All wires in a layer-pair have identical values of width and thickness. The spacing between any two adjacent wires in a given layer-pair and the height of the inter-layer dielectric (ILD) between any two consecutive layer-pairs are constants.
- All wires in the architecture are "L"-shaped. Each segment of an "L"-shaped wire is routed in one layer of a layer-pair. Via area for the "L", and of the ends of the "L" segments, is computed as a part of the wire.
- Longer wires are routed on upper layer-pairs and shorter wires are routed on lower layer-pairs.
- The maximum area available for repeaters is specified as a percentage of total die area[3]. All gates are placed evenly in the entire die.
- Repeaters used in all wires of a given layer-pair are of uniform size.
- Repeaters are inserted starting from the longer wires and proceeding to the shorter wires.

Then, the problem of computing the *rank* of an IA can be well-defined as follows:

Input: (see Table 1 for notation)

- Interconnect architecture α with fixed number of layer-pairs and fixed values of width, spacing, height, and thickness per layer.
- WLD w containing n wires.
- Available repeater area A_R.
- Upper bounds d_i on the maximum permissible delay of wire i, $i = 1, \ldots, n$ in the WLD w.

Objective: Assign wires from the given WLD to the layer-pairs of the given architecture α, and insert repeaters, such that $r(\alpha)$ is maximum.

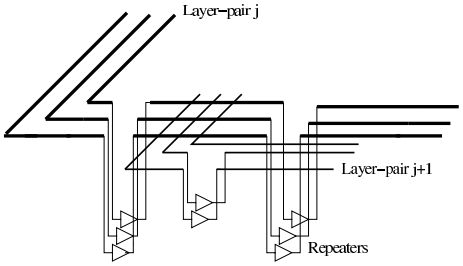

Figure 1: Longer wires are assigned to higher layer-pairs. Shorter wires are assigned to lower layer-pairs. Repeaters are inserted in longer wires first to meet the target delay requirements.

4 Rank Computation using Dynamic Programming

To compute the rank of the IA, a maximum number of wires should be assigned to its layer-pairs, satisfying delay requirements. To achieve this, we require an optimal combination of wires assigned to layer-pairs, repeaters inserted in the wires, and vias. Such an optimal combination is not guaranteed by greedy top-down assignment of wires to layer-pairs with repeater insertion. Figure 2 shows a counter example with all four wires to be assigned having equal length. *RC* delay of the upper layer-pair is much larger than that of the bottom layer-pair. Greedy wire assignment assigns two wires to the upper layer-pair and repeaters to meet target delay, but this exhausts the repeater budget of eight repeaters. Wires assigned to the lower layer-pair thus fail to meet target delay. The optimal solution has rank 4 while the greedy solution has rank 2.

[3] In the current version of our implementation, we do not reconcile implied driver and receiver sizing with total gate area budget. However, the DP algorithm can be extended to address this.

Notation	Description
α	Architecture with m layer-pairs, fixed width, spacing, and thickness
η_i	Repeater count in wire i
ν	Number of vias contributed by a wire in α
τ_j	Delay of wire segment between two consecutive repeaters in layer-pair j
a	Switching constant of repeater
A_d	Die area
A_R	Maximum repeater area
$A_{w,j}$	Total wiring area in layer-pair j
$A_{v,j}$	Total area allocated in layer-pair j for vias from wires assigned to layer-pairs $1,\ldots,j-1$.
$A_{u,j}$	Total area allocated in layer-pair j for vias from repeaters used in layer-pairs $1,\ldots,j-1$.
b	Switching constant of repeater
B_j	Available area for wire assignment in layer-pair j
c_o	Input capacitance of minimum-sized inverter
$\bar{c}_j$	Capacitance per unit length of wire on layer-pair j
C_L	Load capacitance
c_p	Parasitic capacitance of transistor in driver
d_i	Target delay of wire i in WLD
D_i	Delay of wire i
f_c	Target clock frequency
i	Index of wire in WLD
i'	Index of a wire in WLD that meets target delay
j	Index of layer-pairs in IA
l_i	Length of wire i
l_{max}	Maximum wire length
m	Number of layer-pairs in IA α
M	Array storing feasibility of wire assignment
n	Number of wires in WLD
p	Variable representing wire index
q	Variable representing layer-pair index
r	Variable representing repeater area
r_o	Output resistance of minimum-sized inverter
$\bar{r}_j$	Resistance per unit length of wire on layer-pair j
s_j	Repeater size corresponding to layer-pair j
S_j	Spacing between adjacent wires in layer-pair j
$s_{opt,j}$	Optimal repeater size in layer-pair j
v_a	Area of a via (obtained from process parameters)
w	Wire length distribution
W_j	Width of wire in layer-pair j
z_r	Number of repeaters used for area r

Table 1: Table of notations.

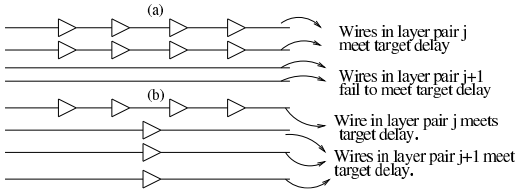

Figure 2: Suboptimality of greed. (a) shows the greedy wire assignment to two consecutive layer-pairs achieving rank = 2. (b) shows the optimal wire assignment achieving rank = 4.

Exhaustive search over all possible layer-assignments and repeater configurations is impractical. We now give a dynamic programming (DP) algorithm that performs optimal rank computation in reasonable time. The DP runs in m stages, where m = total number of layer-pairs in the architecture.

The problem of rank computation is considered as a collection

of subproblems, where each subproblem is characterized by the four parameters (i) number of wires to be assigned, (ii) number of layer-pairs used for assigning the wires, (iii) repeater area used to satisfy the delay constraints, and (iv) number of wires assigned that meet delay constraints. Let i, j, r and i' respectively denote the elements of the 4-tuple in the order. A four-dimensional boolean array M of cells is defined with dimensions corresponding to i, j, r and i'. If i wires can be assigned to j layer-pairs, such that i' wires meet their target delay using at most r repeater area and the remaining $n-i$ wires can be assigned to $m-j$ layer-pairs (ignoring delay requirements), then the value of $M[i,j,r,i']$ is 1. If the assignment is infeasible, then $M[i,j,r,i']$ is 0.

The DP populates the cells of M according to the recurrence relation given by Equation (1), where $1 \leq i, i', i'_1, i'_2 \leq n$, $1 \leq j \leq m$, $1 \leq r, r_1, r_2 \leq A_R$, and z_{r_1} and z_{r_2} are the number of repeaters corresponding to repeater areas r_1 and r_2 respectively. The definitions of the terms 1, 2, 3 in Equation (1) are as follows.

- $M[i'_1, j, r_1, i'_1]$ correspond to previously computed entries of M.
- $M'(i'_1, j+1, z_{r_1}, r-r_1, r_2, i'_2, i)$ indicates whether it is possible to assign wires $i'_1+1, \ldots, i'_1+i'_2$ meeting delay requirements to the $(j+1)^{st}$ layer-pair using at most $r-r_1$ repeater area, given that i'_1 wires have already been assigned to layer-pairs $1,\ldots,j$ using r_1 repeater area, and also that i - i'_1 - i'_2 wires fit into layer-pair $j+1$, ignoring the delay constraints. $r_2 \leq r-r_1$ denotes the actual repeater area used for assigning i'_2 wires to $(j+1)^{st}$ layer-pair. z_{r_1} is used to compute the via area used in $(j+1)^{st}$ layer-pair due to r_1 repeater area used in layer-pairs $1,\ldots,j$. $M'(.)$ is 1 if the assignment is feasible, and 0 otherwise.
- $M''(n, i, m, j+1, z_{r_1}+z_{r_2})$ indicates whether it is possible to assign (n - i) wires to the (remaining) last (m - (j + 1)) layer-pairs ignoring the delay requirements, given that r_1+r_2 repeater area has been used in wires in layer-pairs $1,\ldots,j$ + 1. $z_{r_1}+z_{r_2}$ is the repeater count corresponding to r_1+r_2 repeater area used in layer-pairs $1,\ldots,j+1$. $M''(.)$ is 1 if the assignment is feasible, and 0 otherwise.

For rank computation, we need to find the maximum value of i' for which $M[i,j,A_R,i']$ is 1 for $i=1,\ldots,n$ and $j=1,\ldots,m$.

Algorithm 1: Rank computation

Input: number of wires n, number of layer-pairs m, maximum repeater area A_R
Output: Rank of architecture $r(\alpha)$

1. Initialize_M(n, A_R)
2. update_M(n, m, A_R) // this is the key step in rank computation
3. for $j=m$ to 1
4. for $i=n$ to 1
5. for $i'=i$ to 1
6. if $M[i, j, A_R, i']$ == 1 then
7. return(i')
8. return(0)

Figure 3: Algorithm for computation of IA rank.

The DP algorithm `update_M` starts wire assignment from the topmost layer-pair and proceeds to the lower layer-pairs. Longer wires are assigned to the higher layer-pairs and shorter wires to the lower layer-pairs. In each iteration of the DP (Steps $8-10$ in Figure 5), we compute a binary value that indicates the feasibility of wire assignment to a sequence of layer-pairs. Starting with the longest wire and the topmost layer-pair, we compute the feasibility of assigning n wires to m layer-pairs using at most A_R repeater area. Wire assignment with repeater insertion in a specific layer-pair is performed by a function $M'(.)$. Wire assignment to a sequence of layer-pairs without considering delay requirements is performed by $M''(.)$. The algorithm starts with an initial set of values of the boolean array $M[i',j,r,i']$, for $i'=1,\ldots,n$, $j=1$, and $r=1$ to A_R, set by the function `Initialize( )` in Step 1 of Figure 4. At the first iteration, the value of $M[i',j+1,r,i']$ is computed for $i'=r=1$ and for $j=1$. Subsequent iterations will compute the new values of $M[i',j+1,r,i']$ from the pre-computed values of $M[\,]$, and the values returned by $M'(.)$ and $M''(.)$. The time complexities of the procedures given above are as follows. $M'(.)$ has a worst-case complexity of $O(n+A_R)$. $M''(.)$ has a worst-case complexity of $O(n)$. The function `update_M(.)` has a worst-case complexity of $O(m \times n^4 \times A_R^3)$. Overall, the worst-case time complexity of our algorithm for rank computation (Figure 3) is $O(m \times n^4 \times A_R^3)$.

Algorithm 2: Initialize_M

Input: number of wires n, maximum repeater area A_R
Output: Initialized Boolean Array M

1. for $i=1$ to n
2. for $r=1$ to A_R
3. if $M'(0,2,0,r,r_2,i,i)$ == 1 and $M''(n,i,m,2,z_{r2})$ == 1 then $M[i][1][r]=1$
4. else $M[i][j][r]=0$

Figure 4: Initialization of data structure M.

Algorithm 3: DP (update_M)

Input: number of wires n, number of layer-pairs m, maximum repeater area A_R
Output: Boolean array M

1. for $i=2$ to n
2. for $j=1$ to m - 1
3. for $i'=1$ to i
4. for $r=1$ to A_R
5. $M[i,j+1,r,i']=0$
6. for $i'_1=1$ to i'
7. for $r_1=1$ to r
8. $p = M[i'_1,j,r_1,i'_1] \wedge M'(i'_1,j+1,z_{r_1},r-r_1,r_2,i'-i'_1,i) \wedge M''(n,i,m,j+1,z_{r_1}+z_{r_2})$
9. $M[i,j+1,r,i'] = M[i,j+1,r,i'] \vee p$
10. if $M[i,j+1,r,i']==1$ then goto step (4)

Figure 5: Procedure for updating boolean array M.

4.1 Delay Computation and Repeater Insertion

Target delay for wire i in the WLD is defined as $d_i = (l_i/l_{max}) \times (1/f_c)$ where d_i represents the normalized (with respect to length) delay of the wire i, l_i is the length of wire i, and l_{max} is the maximum wire length in WLD. Longer wires have a larger value of d_i and hence have more stringent delay requirements than shorter wires. The actual delay D_i of a wire i depends on the layer-pair to which it is assigned. Repeater insertion is performed according to the following rules.

- Longer wires are buffered before shorter wires.
- Incremental insertion of repeaters is performed until target delay is met or repeaters cannot be placed at appropriate intervals for a wire.

$$M[i,j+1,r,i'] = \bigvee \{ \overbrace{M[i'_1,j,r_1,i'_1]}^{1} \wedge \overbrace{M'(i'_1,j+1,z_{r_1},r-r_1,r_2,i'_2,i)}^{2} \wedge \overbrace{M''(n,i,m,j+1,z_{r_1}+z_{r_2})}^{3} \mid r_1+r_2 \le r, i'_1+i'_2 = i' \} \quad (1)$$

- Repeaters inserted in all wires of a layer-pair are of uniform size. The optimal repeater size $s_{opt,j}$ is determined using constants $\bar{r}_j$ and $\bar{c}_j$ of a layer-pair. Thus, the number of repeater types is equal to the number of layer-pairs.

The number and size of repeaters inserted in a wire depends upon the layer-pair[4] to which the wire is assigned (as well as the wire length and delay constraints). The delay of wire i assigned to layer-pair j is computed from the model of interconnect given in [15]. Specifically, delay of a wire segment of length l between any two consecutive repeaters is given by [15]:

$$\tau_j = bR_{tr}(C_L + c_p) + b(\bar{c}_j R_{tr} + \bar{r}_j C_L)l + a\bar{r}_j\bar{c}_j l^2 \quad (2)$$

where R_{tr} is transistor equivalent resistance, a and b are constants[5] that depend on the switching model of the repeater, and C_L and c_p are load and parasitic capacitances respectively. Also, $\bar{r}_j$ and $\bar{c}_j$ are determined completely by the wire width, spacing and thickness of a layer-pair. Repeater size s_j is expressed as a multiple of the minimum inverter size. The size of the repeater required to minimize total wire delay is a function of wire parameters and is determined by $R_{tr} = r_o/s_j$ and $C_L = s_j \times c_o$, where r_o is the output resistance and c_o is the input capacitance of a minimum-sized inverter. On layer-pair j, the total delay of a wire of length l_i with η_i repeaters, each of size s_j, is given by the following equation [15]:

$$\begin{aligned} D_i &= \eta_i \tau_j \\ &= \eta_i \left[br_o(c_o + c_p) + b(\bar{c}_j \frac{r_o}{s_j} + \bar{r}_j c_o s_j)\frac{l_i}{\eta_i} + \bar{r}_j\bar{c}_j a \frac{l_i^2}{\eta_i^2} \right] \\ &= br_o(c_o + c_p)\eta_i + b(\bar{c}_j \frac{r_o}{s_j} + \bar{r}_j c_o s_j) l_i + \bar{r}_j\bar{c}_j a \frac{l_i^2}{\eta_i} \end{aligned} \quad (3)$$

To make $D_i \le d_i$, we insert η_i repeaters each of size s_j in wire i. A closed form solution for η_i and s_j cannot be obtained by solving $D_i = d_i$. Instead, we (i) determine optimum repeater size s_j for layer-pair j to minimize delay [14] and (ii) insert repeaters[6] of size s_j incrementally in a wire i until $D_i \le d_i$. Optimum repeater size required to minimize total delay D_i is obtained by setting $\frac{\partial D_i}{\partial s_j} = 0$ and is given by

$$s_{opt,j} = \sqrt{\frac{\bar{c}_j r_o}{c_o \bar{r}_j}} \quad (4)$$

To compute wire area available for routing in a layer-pair, wire count and the number of repeaters inserted in the wires are required. We compute the number of repeaters corresponding to a repeater area as the ratio of repeater area to the repeater size.

$$z_{r_1} = \frac{r_1}{s_j} \quad (5)$$

In Subsections 4.2 and 4.3, we describe the evaluation of $M'(.)$ and $M''(.)$. In these procedures, Equation (5) is used to obtain the number of vias to be allocated in a given layer-pair for repeaters already inserted in higher layer-pairs.

4.2 Wire Assignment to Layer-Pair With Delay Requirements

We now explain key aspects of the procedure `wire_assign` for computing $M'(.)$ in the Equation (1) recurrence. This procedure returns a boolean value indicating the feasibility of assignment of wires to a layer-pair considering delay requirements.

[4]Resistance per unit length, capacitance per unit length, and ground capacitance depend on parameters of the layer-pair.

[5]$a = 0.4$ and $b = 0.7$ for wire delay computation [15].

[6]In this work we assume uniform size repeaters in all wires of a layer-pair.

Algorithm 4: Wire assignment (with delay requirements) wire_assign

Input: number of wires i'_1 above layer-pair j, current layer-pair j, number of repeaters z_{r_1} used for wires in layer-pairs $1,\ldots,j-1$, repeater area r_3 available for assignment in layer-pair j, number of wires i'_2 required to meet target delay in layer-pair j, total number of wires i to be assigned up to current layer-pair, die area A_d, repeater size $s_{opt,j}$, WLD w, target delay d_i of wires.
Output: Boolean value $M'(i'_1,j,z_{r_1},r_3,r_2,i'_2,i)$, r_2 = repeater area actually used in current layer-pair j

1. $B_j = A_d$ - $A_{v,j-1}$ - $A_{u,j-1}$
// $A_{v,j-1}, A_{u,j-1}$ are computed from i'_1 and z_{r_1} respectively
2. $p = i'_1 + 1$
3. while $p \le i'_1 + i'_2$
4. wire_area = $l_p \times (W_j + S_j)$
5. if (wire_area $\le B_j$) then goto step (6) else return(0)
6. assign wire p to layer-pair j
7. $B_j = B_j$ - wire_area
8. while ($D_p > d_p$ AND repeater_area $< r_3$)
9. compute D_p
10. repeater_area = repeater_area + $s_{opt,j}$
11. if (repeater_area $= r_3$) then return(0)
12. $p = p + 1$
13. if wires $i'_1 + i'_2 + 1$, ..., i cannot be assigned then return(0)
14. return(1)

Figure 6: Algorithm for assignment of wires to single layer-pair considering delay requirements.

- It assumes that i'_1 wires are assigned to layer-pairs $1,\ldots,j-1$ meeting delay requirements using z_{r_1} number of repeaters.
- $i - i'_1 - i'_2$ wires are to be assigned to the current layer-pair (j) of which i'_2 wires should meet the target delay within an available repeater area r_3.
- Initially, the available area (B_j) for assignment of wires in layer-pair j is computed from die area A_d, via area ($A_{v,j-1}$) used by wires in layer-pairs $1,\ldots,(j-1)$, and via area ($A_{u,j-1}$) used by repeaters inserted in wires on the layer-pairs $1,\ldots,(j-1)$.
- Wires are assigned incrementally in the current layer-pair until either no more area is available for assignment of wires, or the number of wires assigned is equal to the specified count. The procedure returns 0 if the former condition is satisfied. For each wire i assigned, its actual delay D_i is computed, and is compared with its target delay d_i.
- If $D_i > d_i$, then repeaters of size $s_{opt,j}$ are inserted incrementally until $D_i \le d_i$ or repeater area used is not less than the available area r_3. The procedure returns 0 if the available area for repeaters is used up before the delay in the wire reaches the desired bound.

If the procedure is able to successfully assign i'_2 wires within the available repeater area, it next attempts to assign the remaining $i - i'_1 - i'_2$ wires to the current layer-pair ignoring delay constraints. If the assignment is unsuccessful, it returns 0. If all the above assignments can be done successfully, the procedure returns 1.

Algorithm 5: Greedy assignment (greedy_assign)

Input: total number of wires n, index of last wire i assigned so far, number of layer-pairs $m-(j+1)$, number of repeaters $z_{r_1}+z_{r_2}$ used for repeater area (r_1+r_2) for wires in layer-pairs $1,\ldots,j+1$, die area A_d.
Output: Boolean value $M''(n,i,m,j+1,z_{r_1}+z_{r_2})$

1. for $q=m$ to $j+2$ //q is the layer-pair index
2. compute $B_q = A_d - ((z_{r_1}+z_{r_2})+\nu\times i)\times v_a$
3. $q=m$ // start with bottommost layer-pair
4. $p=n$ // start with smallest wire
5. while$(q>(j+1))$
6. if$(p==i)$ then return(1)
7. $A_{w,q} = l_p \times (W_q + S_q)$
8. $A_{v,q} = 0$
9. while$(A_{w,q}+A_{v,q} \le B_q)$
10. assign wire p to layer-pair q
11. compute $A_{w,q} = A_{w,q} + l_p \times (W_q + S_q)$
12. compute $A_{v,q} = (p-i)\times\nu\times v_a$
13. $p=p-1$
14. if$(p==i)$ then return(1)
15. $q=q-1$
16. return(0)

Figure 7: Greedy algorithm for assignment of wires to layer-pairs without considering delay bounds.

4.3 Wire Assignment to Layer-Pairs Without Delay Requirements

The procedure `greedy_assign` computes the feasibility of assigning $n-i$ wires to $m-(j+1)$ layer-pairs without considering delay requirements. Wire assignment is performed to layer-pairs greedily in a bottom-up manner until all the layer-pairs are full. Salient aspects are as follows.

- Repeaters assigned to wires in layer-pairs $1,\ldots,j$ are routed using vias passing through all the layer-pairs below. Wire assignments in all the layer-pairs $(j+2),\ldots,m$ take into account area occupied by repeater vias. This is computed in Steps 9 and 10 of procedure `greedy_assign`.
- The area remaining in layer-pair j for wire assignment is $A_d - A_{u,j}$ (after removing area corresponding to repeater vias).
- Wires are assigned bottom-up starting from layer-pair m, and from wire n. Incremental assignment is performed to one layer-pair at a time until $A_{w,j}+A_{v,j} = A_d - A_{u,j}$. If the available area in a layer-pair is zero, then wire assignment starts from the next higher layer-pair.

The procedure returns 1 if all $n-i$ wires can be assigned within the available $m-(j+1)$ layer-pairs. It returns 0 otherwise.

Lemma 1 *`greedy_assign` is optimal.*

Proof. The procedure `greedy_assign` has the following characteristics: (i) wires are assigned in ascending order of their lengths, starting from the shorter wires at the bottom layers; (ii) it uses strictly more wires in the lower layers; (iii) it has strictly more wiring resource in the higher layers and (iv) it has less wiring demand in the upper layers. Thus, at any stage, if there is some extra space in any lower layer-pair, then some wires can always be moved from a higher layer-pair to the lower layer-pair to get an improved solution. This procedure thus attempts to pack wires in the layer-pairs strictly in a bottom-up manner, and uses optimum number of layer-pairs. □

5 Performance Studies

We study the variation in rank for different IA's and WLD's. Architectures chosen for study are based on TSMC parameters for the 180nm, 130nm and 90nm technology nodes (given in Table 3) [12]. WLDs are generated for 1M, 4M and 10M gate designs using the method of [4] and Rent parameter p = 0.6. Variation of rank is studied with varying ILD permittivity, Miller coupling factor, target clock frequency, and maximum repeater area. To reduce runtime, we perform *coarsening* of the WLD for large instances.

5.1 Coarsening of the WLD

The time complexities of our proposed algorithms are very large. For the large gate counts used in our studies, naive implementation of the basic algorithm requires exorbitant runtime. We reduce instance complexity by forming *bunches* of connections given by the WLD, such that each bunch is a collection of wires of uniform size, and assignment of the connections to the layer-pairs is done in bunches of several wires instead of the simple method of one wire at a time. The rank of the architecture is determined by the actual number of wires present in the maximum set of bunches that can be feasibly assigned to the layer-pairs. Hence, error in rank computation due to bunching can be at most the size of the maximum bunch formed from the given WLD. The relation between maximum number of bunches, bunch size and the number of wires is given by $max.\ number\ of\ bunches = \lceil number\ of\ wires / bunch\ size \rceil$.

In our bunching procedure, all the wires of a bunch are of uniform length. For instance, for a set of 100 wires of identical size, if the bunch size is specified as 40, we generate three bunches, of sizes 40, 40 and 20 respectively. Delay considerations for a bunch can be easily obtained from those of a single wire in the bunch. However, our proposed bunching scheme may not be appropriate for an input WLD with very few wires of identical size.[7]

5.2 Experimental Results and Implications

Dielectric constant, Miller coupling factor, target clock frequency and maximum repeater area are varied to study the effect of these parameters on rank. A baseline design is chosen and each of the above parameters is varied one at a time to observe variation in rank. We performed experiments with baseline designs of 4M gates in the 90nm, 1M gates in the 130nm, and 1M gates in the 180nm technology nodes. For space reasons, here we report experiments with a single baseline design of 1M gates in the 130nm technology node. The baseline parameters are given in Table 2.

Parameter	Baseline value
k	3.9
Miller coupling factor	2
Repeater area fraction	0.4
Semi-global layer-pairs	2
Global layer-pairs	1
Target clock frequency	500MHz

Table 2: Baseline parameters for the 180nm, 130nm and 90nm technology node designs.

WLDs are generated for the 1M and 4M gate design based on the stochastic WLD model of [4] with Rent parameter p = 0.6. Initially, die area is computed based on the gate pitch (g), number of gates (N) as $Die\ Area\ due\ to\ gates = g^2 N$. Maximum repeater allocation (A_r) is specified as a fraction of the die area and is added

[7] For further reduction of runtime, we also use a different, and orthogonal, instance size reduction from the bunching technique. In this *binning* technique, we replace a group of wires with a single wire whose length is the mean of all wire lengths in the group. Thus, for example, if we have a set of wires of lengths 5996, 5997, 5998, 5999, and 6000 of counts 3, 2, 2, 1 and 1 respectively, then the binning procedure will reduce this set to a single wire length of 5998 with a count of 9. This reduces the size of the distribution by factor of 5. Binning can also be used on bunched wires. While this separate coarsening technique is also available, we did not use it because practical runtimes were achievable using only bunching. (However, since our present results may be partly compromised by the effects of large bunch size, our ongoing experimentation is incorporating a reduced bunch size in conjunction with binning.) It is important to note that bunching and binning do not change the time complexity of our DP algorithm.

to the actual die area. Then, the actual die area used is given by Equation (6).

$$A_r = Max.\ repeater\ fraction \times A_d \quad (6)$$
$$A_d = A_r + Die\ Area\ due\ to\ gates$$

The gate pitch for computing the actual wire lengths in the WLD is then obtained by distributing gates evenly in the actual die area A_d. A clock frequency of 1.7GHz is chosen for 130nm (maxmum MPU clock frequency based on ITRS 2001 [8]). The *bunch size* used is 10000. The technology parameters chosen for the study of variation of rank are given in Table 3. For die area computation, gate pitch is taken as $12.6 \times Tech\ Node$ (based on empirical data from ITRS [8]). The variation of rank with dielectric constant, Miller coupling factor, target clock frequency, and repeater area is given in Table 4.

Parameter	180nm	130nm	90nm
M_1 minimum width	0.230μm	0.160μm	0.120μm
M_1 minimum spacing	0.230μm	0.180μm	0.12μm
M_1 thickness	0.483μm	0.336μm	0.26μm
M_X minimum width	0.280μm	0.200μm	0.14μm
M_X minimum spacing	0.280μm	0.210μm	0.14μm
M_X thickness	0.588μm	0.340μm	0.30μm
M_t minimum width	0.440μm	0.440μm	0.42μm
M_t minimum spacing	0.460μm	0.460μm	0.42μm
M_t thickness	0.960μm	1.020μm	0.88μm
V_1 minimum width	0.260μm	0.190μm	0.13μm
V_{x-1} minimum width	0.260μm	0.260μm	0.13μm
V_{t-1} minimum width	0.360μm	0.360μm	0.36μm

Table 3: Technology parameters used for study of variation of rank for the 130*nm* technology node. Parameters for the 180*nm* and 90*nm* technology nodes are also given. For 180*nm*, x = 2, 3, 4, 5 and t = 6. For 130*nm*, x = 2, 3, 4, 5, 6 and t = 7. For 90*nm*, x = 2, 3, 4, 5, 6, 7 and t = 8.

K		M		C		R	
3.90	0.397288	2.00	0.397288	5.00e+08	0.397288	0.10	0.117438
3.80	0.402596	1.95	0.401711	6.00e+08	0.391980	0.20	0.210967
3.70	0.407019	1.90	0.407019	7.00e+08	0.388441	0.30	0.303728
3.60	0.413212	1.85	0.412327	8.00e+08	0.385787	0.40	0.397288
3.50	0.418520	1.80	0.418520	9.00e+08	0.384018	0.50	0.491019
3.40	0.424713	1.75	0.423828	1.00e+09	0.382249		
3.30	0.430021	1.70	0.429136	1.10e+09	0.309706		
3.20	0.437098	1.65	0.435329	1.20e+09	0.309706		
3.10	0.444175	1.60	0.441521	1.30e+09	0.309706		
3.00	0.450368	1.55	0.449483	1.40e+09	0.309706		
2.90	0.458330	1.50	0.456561	1.50e+09	0.309706		
2.80	0.465364	1.45	0.463594	1.60e+09	0.235608		
2.70	0.474210	1.40	0.471556	1.70e+09	0.235608		
2.60	0.482172	1.35	0.479518				
2.50	0.491904	1.30	0.488365				
2.40	0.501635	1.25	0.498096				
2.30	0.512251	1.20	0.507828				
2.20	0.522867	1.15	0.518444				
2.10	0.534368	1.10	0.529060				
2.00	0.547637	1.05	0.540560				
1.80	0.575947	1.00	0.553830				
1.90	0.560907						

Table 4: Variation of rank for the 130nm, 1M gate design. The second sub-column in each column corresponds to normalized rank. Legend: K = ILD permittivity; M = Miller coupling factor; C = target clock frequency; R = maximum repeater fraction of die area.

We observe that reduction in dielectric constant enables reduction in coupling capacitance and delay. For the 130nm technology node (Table 4), reduction of 38% in k produces the same increase in rank as 42.5% change in Miller coupling factor[8]. For ease of comparison, we normalized rank with respect to the total number of wires in the WLD. Simulations were performed on a dual-processor Intel Xeon system with 2GB of memory, running Linux OS. No rank computation has runtime greater than 200s in our implementation.

6 Conclusions and Future Directions

In this paper, we have proposed a new metric for evaluation of quality of interconnect architectures. A dynamic programming method for rank computation is presented. The variation of rank with K, M, C and R (see Table 3 for notation) for different geometric parameters and technology nodes is studied. Results show, in general, an improvement of rank with decreasing values of K, M, C, and increasing values of R. Comparison of trends in variation of rank for the 130nm technology node for a 1M gate design indicate that reductions in M can have almost the same performance impact as reduction in K. The variation of rank with several geometric and technology parameters show the need to "co-optimize" across several material, process, and design characteristics to achieve high-rank embeddings of future WLDs in future interconnect architectures. In other words, it is not possible to enable future MPU-class designs by material improvements alone.

In our study, the delay requirement of wires in the WLD is assumed to be linear in wire length. This requirement becomes unreasonable since the actual delay of the connections in the IA is proportional to the square of length. Thus, we are currently studying alternative models for per-connection delay requirement. We are also pursuing direct optimization of interconnect architectures according to our proposed metric, with the goal of evaluating ITRS and foundry BEOL architectures.

[8] In our experiments, we considered the minimum value of Miller coupling factor to be 1.0 . This can be achieved by double-sided shielding of lines.

References

[1] M. B. Anand, M. Kukumu and H. Shibita, "Multiobjective Optimization of VLSI Interconnect Parameters", *IEEE Trans. on CAD of Integrated Circuits and Systems*, 17(12), 1998, pp. 231-239.

[2] M. T. Bohr, "Interconnect Scaling - The Real Limiter to High Performance ULSI", *IEDM Tech Dig.*, 1995, pp. 241-244.

[3] Q. Chen, J. A. Davis, J. D. Meindl and P. Zarkesh-Ha, "A Compact Physical via Blockage Model", *IEEE Trans. on VLSI Systems*, 8(6), 2000, pp. 689-692.

[4] J. A. Davis, V. K. De and J. D. Meindl, "A Stochastic Wire-length Distribution for Gigascale Integration (GSI) - Part 1: Derivation and Validation", *IEEE Trans. on Electron Devices*, 45(3), 1998, pp. 580-589.

[5] M. Edahiro, Y. Hayashi and S. Takahashi, "Interconnect Design Strategy: Structures, Repeaters and Materials with Strategic System Performance Analysis (S^2PAL) Model", *IEEE Trans. on Electron Devices*, 48(2), 2001, pp. 345-356.

[6] C. Hu, S. -Y. Oh, O. S. Nakagawa and D. Sylvester, "Interconnect Scaling: Signal Integrity and Performance in Future High-Speed CMOS Designs", *Symposium on VLSI Technology: Digest of Technical Papers*, 1998, pp. 45-47.

[7] A. B. Kahng, S. Mantik, D. Stroobandt, "Toward Accurate Models of Achievable Routing", *IEEE Trans. on CAD of Circuits and Systems*, 20(5), 2001, pp. 648-659.

[8] *The International Technology Roadmap for Semiconductors*, 2001 edition, International Sematech, Austin, Texas, December 2001. http://public.itrs.net/

[9] A. B. Kahng, D. Stroobandt, "Wiring layer assignment with consistent stage delays", *Proc. Intl. Workshop SLIP*, 2000, pp. 115-122.

[10] K. Rahmat, O. S. Nagakawa, S. -Y. Oh and J. Moll, "A Scaling Scheme for interconnect in deep submicron". *HP technical literature*, ULSI Laboratory, Hewlett-Packard Co., Palo Alto, CA 94304.

[11] S. Odanaka and K. Yamashita, "Interconnect Scaling Scenario Using a Chip Level Interconnect Model", *IEEE Trans. on Electron Devices*, 47(1), 2000, pp. 151-162.

[12] Taiwan Semiconductor Manufacturing Company Ltd. http://www.tsmc.com/

[13] R. Venkatesan, J. A. Davis, K. A. Bowman, and J. D. Meindl, "Optimal $n-tier$ Multilevel Interconnect Architecture for Gigascale Integratioin (GSI)", *IEEE Trans. on VLSI Systems*, 9(6), 2001, pp. 899-912.

[14] H. B. Bakoglu, "Circuits, Interconnections, and Packaging for VLSI", *Addison-Wesley*, 1990.

[15] R. H. J. M. Otten and R. K. Brayton, "Planning for Performance", *Proc. of DAC*, 1998, pp. 122-127.