CONTENTS

Available online at

ScienceDirect

www.sciencedirect.com

# Timing-driven via placement heuristics for three-dimensional ICs

Vasilis F. Pavlidis*, Eby G. Friedman

*Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627, USA*

## Abstract

The dependence of the interconnect delay on the interplane via location in three-dimensional (3-D) ICs is investigated in this paper. The delay of these interconnects can be significantly decreased by optimally placing the interplane vias. The via locations that minimize the propagation delay of two-terminal interconnects consisting of multiple interplane vias under the distributed Elmore delay model are determined. For interconnect trees, the interplane via locations that minimize the summation of the weighted delay of the sinks of the tree are also determined. For these interconnect structures, the interplane via locations are obtained both through geometric programming and near-optimal heuristics. Placement constraints are imposed such that the path is negligibly affected. The proposed heuristics are used to implement efficient algorithms that exhibit lower computational times as compared to general optimization solvers with negligible loss of optimality. Various interplane via placement scenarios are considered. Simulation results indicate delay improvements for relatively short point-to-point interconnects of up to 32% with optimally placed interplane vias. For interconnect trees, the maximum improvement in delay for optimally placed interplane vias is 19%. The proposed algorithms can be integrated into a design flow for 3-D circuits to enhance placement and routing where timing is a primary design criterion.

## 1. Introduction

Technology scaling has enabled an increase in integration density and a considerable decrease in the intrinsic gate delay through smaller and faster devices. Higher integration densities require both a greater number and longer interconnects. Therefore, as the device delay is reduced, the performance of integrated circuits has become dominated by the interconnect delay. In addition, other interconnect-related issues, such as power consumption and signal integrity, have become more pronounced with technology scaling. To manage these issues, a variety of techniques have been developed such as tapered buffers, repeater insertion, wire sizing, and shielding, to name only a few. Nonetheless, these techniques increase silicon area and power consumption and do not mitigate the primary issue, which is the increase in interconnect length. As a result, innovative technologies and design techniques are required to satisfy the ever increasing demand for greater performance.

Three-dimensional (3-D) or volumetric integration is such a promising alternative which offers the opportunity to relieve the deleterious effects of long interconnects [1]. Another important characteristic of 3-D structures is that these systems can include different technologies such as GaAs and SiGe, and design disciplines such as analog, digital, RF circuits, and MEMS implemented within a single 3-D multiplane system, where each of the system components is fabricated with a high yield manufacturing process. Such technological diversity extends the capabilities of 3-D systems over a conventional CMOS platform, expanding the boundaries of the IC design space.

Three-dimensional systems can be conceptualized both at the package and wafer level [1–3]. Although package level techniques for 3-D circuits reduce the off-chip interconnect distances and overcome the limitations of systems-on-chip

*Corresponding author. Tel.: +1 585 275 1606.

*E-mail addresses:* pavlidis@ece.rochester.edu (V.F. Pavlidis), friedman@ece.rochester.edu (E.G. Friedman).

(SoC), such as process yield and compatibility issues, these methods do not significantly decrease the length of the on-chip interconnects, a problematic issue in high performance circuits. The on-chip interconnect length can be greatly decreased by employing those 3-D technologies where the interplane interconnections are not limited to the die periphery.

Beam recrystallization [4], silicon epitaxial growth [5], solid phase crystallization [6], and processed wafer bonding [7] are examples of wafer level fabrication techniques that have been proposed for 3-D circuits, where the interconnections among devices located on different planes are implemented with vertical interplane interconnects. All but the last one of these techniques involve the growth of the devices of the stack on a bulk silicon layer. The primary disadvantage of these fabrication techniques except for the wafer bonding technique, however, is the difficulty in providing high quality devices.

The processed wafer bonding technique differs from the other fabrication processes in that this technique supports the processing of each plane as an independent wafer [7–10]. These alternatives naturally involve high quality devices since each device layer is developed as a standard two-dimensional (2-D) IC, producing high yield for each individual die within the stack. A major limitation of these techniques is the misalignment of the planes during the bonding process, placing constraints on the minimum size of the interplane vias. Another important concern is the substrate thickness of the upper planes of the 3-D structure, which primarily determines the length of the interplane vias. Although wafer thinning is applied, sufficient substrate thickness is required to sustain the mechanical stresses developed during the stacking of the 3-D system. Alternatively, silicon-on-insulator (SOI) technology can be used for the upper planes of the 3-D structure, drastically reducing the interplane via length [11–13]. Thermal effects are significantly pronounced for this technology, however, due to the low thermal conductivity of $SiO_2$ [13].

In this paper, wafer bonding is considered to be the target technology for 3-D systems. A schematic of a 3-D circuit is illustrated in Fig. 1, where two physical planes are bonded with adhesive materials or metal pads [8]. As illustrated in Fig. 1, the physical planes are bonded face to face. Back-to-face bonding can also be utilized. Each physical plane of the stack is similar to a conventional 2-D circuit, in that a plane includes a device layer and multiple metal layers to connect individual circuits located on the same physical plane (the intraplane interconnects). Communication among circuits on different physical planes (the interplane interconnects) is implemented by interplane vias, which are called vias here for brevity.

To fully exploit the potential of 3-D circuits, sophisticated placement and routing algorithms are required. A channel routing methodology suitable for 3-D standard cell and gate array circuits has been presented in Ref. [14], while a thermal aware placement technique for the same type of circuits is presented in Ref. [15]. The routing problem for 3-D FPGAs has also been addressed in Ref. [16]. Early 2-D CAD algorithms have recently been adapted for 3-D circuits [17–23]. In all of these algorithms, however, the particular traits of the interplane interconnects, such as the non-uniform impedance and the location of the via, are not considered. In addition, in some of these approaches, the interplane vias are completely ignored [23] or considered equivalent to the intraplane interconnects [21], an assumption that does not apply to every form of 3-D integration, such as system-in-package.

Zhang et al. [24] consider the effect of the vertical vias on the interplane interconnects in their delay expression by modeling the line with different impedances; however, the authors apply two restrictive assumptions. One assumption is that the via is always placed at the center of the line, independent of line length, and the second assumption is that each horizontal segment of the interconnect has the same impedance characteristics. The former assumption can lead to severe performance inaccuracies, while the latter assumption does not accurately depict the physical nature of the interplane interconnects as described in Ref. [25]. Randomly placing the vias can result in significant performance degradation.

The via locations that yield the minimum propagation delay of interplane interconnects are determined in this work. In the case of two-terminal interconnects with multiple vias, the via locations are determined through geometric programming and non-convex quadratic programming, where globally optimum solutions are obtained. These solutions are compared in terms of optimality with a proposed heuristic. Another heuristic for the near-optimal via location of multi-terminal interconnect trees is also introduced. Optimization algorithms based on the proposed heuristics exhibit lower
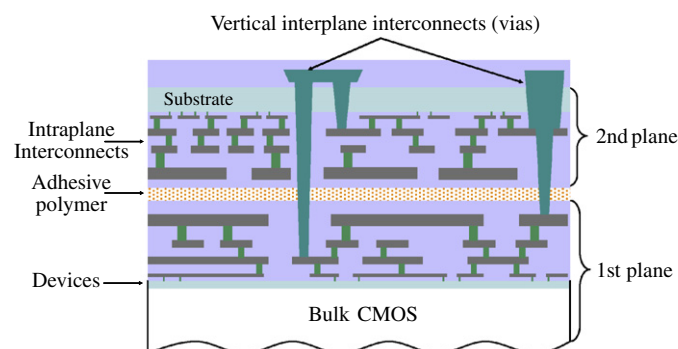


Fig. 1. Schematic of a three-dimensional circuit [8] where face-to-face bonding is employed.

computational times as compared to general optimization solvers. Finally, the effect of the impedance characteristics of the interconnect segments on the improvement in delay achieved by the proposed via placement method is investigated. Simulation results demonstrate the integration of variable via locations into placement and routing algorithms for 3-D circuits, which can considerably enhance the performance of a 3-D design flow.

The rest of the paper is organized as follows. In Section 2, the problem of via placement for two- and multi-terminal interconnects is defined and specific characteristics of interconnects in 3-D circuits are discussed. A heuristic for determining the near-optimal via location of interplane interconnects that comprise more than one via is presented in Section 3. The via locations for multi-terminal nets that minimize the summation of the weighted delay of the branches of the tree are determined in Section 4. Efficient algorithms based on the proposed heuristics are described in Section 5. Simulation results are presented in Section 6, illustrating the performance enhancements that can be achieved by optimally placing vias in 3-D circuits. Finally, in Section 7, some conclusions are offered.

## 2. Via placement problem formulation

The timing-driven via placement problem for two-terminal nets and interconnect trees under the distributed Elmore delay model is formulated in this section. In conventional 2-D circuits, a two-terminal net such as the structure shown in Fig. 2 is usually modeled as a line with uniform impedance characteristics, while the vias are either ignored or considered as small lumped resistive loads. The heterogeneity of 3-D circuits, however, does not support a uniform line model. In 3-D systems, circuits from different and disparate technologies are integrated onto a single multiplane system. As a characteristic example of heterogeneous 3-D systems, consider the 3-D SOI process developed by the M.I.T. Lincoln Laboratory [26]. This process includes wafer bonding of three planes with three metal layers available for each plane, where the sheet resistance of the topmost plane is approximately an order of magnitude smaller than that of the other metal layers. The difference in the impedance characteristics of the interconnects in 3-D systems is also the result of process variations that exist among dies of the same wafer (interdie variation) and among dies of different wafers (wafer-to-wafer variations). The interplane interconnects are, therefore, modeled as wire segments with non-uniform impedance characteristics.

In order to analyze the delay of a line, the distributed Elmore delay model has been adopted due to the simplicity and high fidelity of this model [27]. The accuracy of the model can be further improved as discussed in Ref. [28]. However, unlike a single plane, more than one set of fitting coefficients is required in a 3-D system. Alternatively, higher order models with greater accuracy as compared to the Elmore delay model can be utilized to characterize the delay of the interplane nets. Due to the particular traits of the interplane nets in 3-D circuits, however, the optimization problem can be non-convex even for the simple Elmore delay model. Employing higher order delay models further exacerbates the difficulty of optimizing the interconnect delay as the convexity of these timing models cannot be easily proved. In addition, these models may not be in suitable form to be solved as a geometric programming problem, which can yield globally optimum solutions. Consequently, any solutions based on these models can produce local minima, possibly creating inferior solutions than that produced by the less accurate Elmore delay model. An increase in the computational time should also be considered as a natural tradeoff for greater accuracy when utilizing such models. Two-terminal interplane nets comprising multiple vias are considered in Section 2.1. The more complex task of via placement for interplane interconnect trees in 3-D circuits is introduced in Section 2.2.

### 2.1. Two-terminal net with multiple vias

The problem of timing-driven via placement for two-terminal nets is formulated in this section. Consider the interplane interconnect shown in Fig. 2 that connects two circuits located $n$ physical planes apart. The horizontal segments of the line are connected through the vias, which can traverse more than one plane. Consequently, the number of horizontal segments within the interconnect is smaller than or at most equal to the number of physical planes between the two circuits, i.e., $n \geqslant m$, where the equality only applies when each of the vias connects metal layers from two adjacent physical planes.

Each horizontal segment $j$ of the line is located on a different physical plane with length $l_j$. The vias are denoted by the index of the first of the two connected segments. For example, if a via connects segment $j$ and $j+1$, the via is denoted as $v_j$
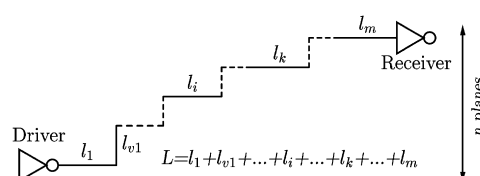


Fig. 2. Interplane interconnect consisting of $m$ segments connecting two circuits located $n$ planes apart.

with length $l_{vj}$. Note that planes $j$ and $j+1$ are not necessarily physically adjacent. The total length of the line $L$ is equal to the summation of the length of the horizontal segments and vias

$$L = l_1 + l_{v1} + \cdots + l_j + l_{vj} + \cdots + l_m. \tag{1}$$

The length of each horizontal segment of the line is bounded

$$l_{j\,\min} \leqslant l_j \leqslant l_{j\,\min} + \Delta x_j, \tag{2}$$

or, alternatively, the via placement is constrained

$$0 \leqslant x_j \leqslant \Delta x_j, \tag{3}$$

where $l_{j\,\min}$ is the minimum length of the interconnect segment on plane $j$, and $\Delta x_j$ is the length of the interval in which the via that connects planes $j$ and $j+1$ is placed. This interval length is called the "allowed interval" here for clarity. $x_j$ is the distance of the via location from the edge of the allowed interval.

$l_{j\,\min}$ is the length of an interconnect segment connecting two allowed intervals or an allowed interval and a placed cell. These lengths are considered fixed. Alternatively, the routing path of a net is not altered except for the via location within the allowed intervals. Each horizontal segment is assumed to be laid out on a single metal layer within the physical plane. In the case where a horizontal segment is on more than one layer, as the outcome of a layer assignment algorithm [29], the problem can be approached in two different ways. The intraplane vias can be treated as additional variables where the location of these vias also needs to be determined. This formulation, however, requires the additional allowed intervals be determined specifically for the intraplane vias. Alternatively, the first and last section of the segment connected to the interplane vias remains as a variable while the remaining sections of that horizontal segment constitute the minimum length of segment $l_{j\,\min}$, which is constant, as previously discussed.

The distributed Elmore delay model is used to determine the delay of these interconnects. The corresponding electrical model of the line is depicted in Fig. 3. The related notation is listed in Table 1. The distributed Elmore delay of a two-terminal interconnect in matrix form is

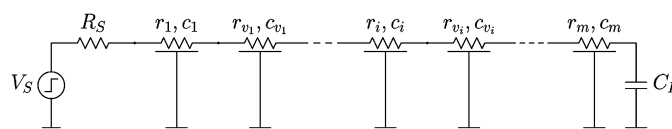$$T(\mathbf{l}) = 0.5\mathbf{l}^{\mathrm{T}}\mathbf{Al} + \mathbf{bl} + D, \tag{4}$$



Fig. 3. Interplane interconnect model composed of a set of non-uniform distributed $RC$ segments.

Table 1
Notation for two-terminal nets and interconnect trees

| Notation | Definition |
| --- | --- |
| $R_{\mathrm{S}}$ | Driver resistance |
| $C_{\mathrm{L}}$ | Load capacitance |
| $r_j$ ($c_j$) | Resistance (capacitance) per unit length of horizontal segment $j$ |
| $r_{vj}$ ($c_{vj}$) | Resistance (capacitance) per unit length of interplane via $v_j$ |
| $R_j$ ($C_j$) | Total interconnect resistance (capacitance) of horizontal segment $j$ |
| $R_{vj}$ ($C_{vj}$) | Total interconnect resistance (capacitance) of interplane via $v_j$ |
| $R_{u_j}$ | Upstream resistance of the allowed interval of via $v_j$ |
| $R_{u_{ij}}$ | Common upstream resistance of the allowed interval of via $v_i$ and $v_j$ |
| $d_i$ | Candidate direction for a $type$-2 move |
| $C_{d_j}$ | Total downstream capacitance of the allowed interval of via $v_j$ (in every direction $d_i$) |
| $P_{s_{pq}}$ | Path from root of the tree to sink $s_{pq}$ |
| $P_{s_{pq}v_j}$ | Path to sink $s_{pq}$ including $v_j$ in every candidate direction |
| $U_{kj}$ | Set of vias located upstream $v_j$ up to $v_k$, including $v_k$ and belonging to at least one path $P_{s_{pq}v_j}$ |
| $\overline{P_{s_{pq}v_j}}$ | Path to sink $s_{pq}$ that does not include $v_j$ |
| $\overline{P_{s_{pq}U_{kj}}}$ | Path to sink $s_{pq}$ that does not include any of the vias in the set $U_{kj}$ |
| $P_{s_{pq}v_jd_i}$ | Path to sink $s_{pq}$ that includes $v_j$ and belongs to direction $d_i$ |
| $\overline{P_{s_{pq}v_jd_i}}$ | Path to sink $s_{pq}$ including $v_j$ in every candidate direction except for $d_i$ |
| $C_{dv_jd_i}$ | Downstream capacitance of the allowed interval of via $v_j$ for the paths $P_{s_{pq}v_jd_i}$ |
| $C_{dv_j\overline{d_i}}$ | Downstream capacitance of the allowed interval of via $v_j$ for the paths $\overline{P_{s_{pq}v_jd_i}}$ |

$$\mathbf{l} = \begin{bmatrix} l_1 & l_2 & \cdots & l_{m-1} & l_m \end{bmatrix}^{\mathrm{T}}, \tag{5}$$

$$\mathbf{A} = \begin{bmatrix} r_1 c_1 & r_1 c_2 & r_1 c_3 & \cdots & r_1 c_m \\ r_1 c_2 & r_2 c_2 & r_2 c_3 & \cdots & r_2 c_m \\ \vdots & \vdots & \cdots & \cdots & \vdots \\ r_1 c_{m-1} & r_2 c_{m-1} & r_3 c_{m-1} & \cdots & r_{m-1} c_m \\ r_1 c_m & r_2 c_m & r_3 c_m & \cdots & r_m c_m \end{bmatrix}, \tag{6}$$

$$\mathbf{b} = \begin{bmatrix} r_1 \left( \sum\limits_{i=1}^{m-1} c_{vi} l_{vi} + C_L \right) + c_1 R_S \\ \vdots \\ r_m C_L + c_m \left( R_S + \sum\limits_{i=1}^{m-1} r_{vi} l_{vi} \right) \end{bmatrix}^{\mathrm{T}}, \tag{7}$$

$$D = R_{\mathrm{S}} \sum_{i=1}^{m-1} c_{vi} l_{vi} + C_{\mathrm{L}} \sum_{i=1}^{m-1} r_{vi} l_{vi} + \frac{1}{2} \sum_{i=1}^{m-1} r_{vi} c_{vi} l_{vi}^2 + R_{\mathrm{S}} C_{\mathrm{L}}. \tag{8}$$

Note that Eq. (5) includes only the length of the horizontal segments of the interconnect $l_j$, as the length of the vias $l_{vj}$ is considered constant. Since Eq. (8) is a constant quantity, the optimization problem can be described as follows:

(P)    minimize $T(\mathbf{l}) = 0.5 \mathbf{l}^{\mathrm{T}} \mathbf{A} \mathbf{l} + \mathbf{b} \mathbf{l}$
    subject to (1), (2), and (3).

As described by the following theorem, the primal problem (P) is typically not convex and, therefore, convex quadratic programming optimization techniques are not directly applicable.

**Theorem 1.** *The primal optimization problem (P) is convex iff*

$$r_{i+1} c_i - r_i c_{i+1} > 0. \tag{9}$$

**Proof. A** is a positive definite matrix if all subdeterminants are positive. By elementary row operations, the subdeterminants of **A** are positive *iff* (9) applies. If (9) applies, **A** is positive definite and (P) is a convex optimization problem.□

Note that Eq. (9) should be satisfied for every horizontal segment of the interconnect such that **A** is a positive definite matrix.

### 2.2. Interconnect trees

Timing-driven via placement for interplane interconnect tree structures is formulated in this section. A simple interplane interconnect tree (also called an interconnect tree for simplicity) is illustrated in Fig. 4a, while some related terminology is listed in Table 1. The sinks of the tree are located on different physical planes within a 3-D stack. Sub-trees not directly
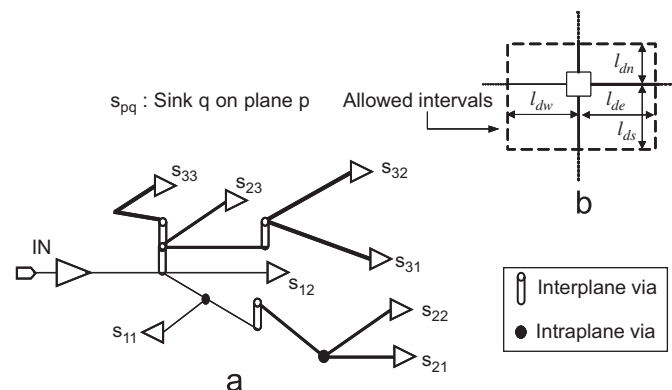


Fig. 4. Interplane interconnect tree: (a) typical interplane interconnect tree and (b) intervals and directions that the interplane via can move.

connected to the interplane vias and that do not contain any interplane vias (i.e., intraplane trees) are also shown. The interconnect segment from each physical plane is denoted by a solid line of varying thickness. Different objective functions can be applied to optimize such an interconnect structure. In this paper, the weighted summation of the distributed Elmore delay of the branches of an interconnect tree is considered as the objective function

$$T_w = \sum_{\forall s_{pq}} w_{s_{pq}} T_{s_{pq}}, \tag{10}$$

where $w_{s_{pq}}$ and $T_{s_{pq}}$ are the weight and distributed Elmore delay of sink $s_{pq}$, respectively. The weights are assigned to the sinks according to the criticality of the net. The criticality of the nets is user defined. Alternatively, for a specific circuit, the identified critical paths are assigned higher weights. The criticality of the sinks can also be determined either by adopting a uniform distribution or, alternatively, as adopted here, the criticality of the paths can be based on the length of the specific paths. Consequently, the longer nets are assigned greater weights, which is a reasonable and practical approach.

For a via connecting multiple interconnect segments, or equivalently, for a via with degree greater than two, there are several candidate directions $d_i$s along which the delay can be decreased. The placement of vias along these directions is constrained by the $l_{di}$s, as shown in Fig. 4b, where the length $l_{di}$s are not generally equal. In addition, vias can span more than one physical plane. For example, consider the via connecting sinks $s_{23}$ and $s_{33}$. This via traverses two physical planes, where the allowed interval for placing the via can be different for each plane.

Three different types of moves for an interplane via are defined. A *type*-1 move is shown in Fig. 5a. This type of move requires the insertion of an intraplane via (to preserve connectivity), as depicted by a dot in Fig. 5a. In the following analysis, the effect of these additional intraplane vias on the delay of the tree is assumed to be insignificant, where the impedance characteristics of the intraplane vias are assumed to be considerably lower than the impedance characteristics of the interplane vias [30], particularly if bulk CMOS devices are used for the upper planes. Alternatively, this effect can be included by appropriately shrinking the length of the allowed interval of the interplane via.

A *type*-2 move is shown in Fig. 5b. A *type*-2 move differs from a *type*-1 move in that an additional interconnect segment of length $\Delta l$ is inserted. Although an additional interconnect segment is required for this type of move, a reduction in the delay of the tree can occur. The segments of length $\Delta l$ illustrated in Fig. 5b are located on the same $y$-coordinate but on different physical planes, yet are shown on different coordinates for added clarity.

Another type of move is illustrated in Fig. 5c, where additional interplane vias are inserted, and is denoted as a *type*-3 move. This type of move is not permitted for two reasons. The additional interplane vias outweigh the delay reduction resulting from optimizing the length of the connected segments due to the high impedance characteristics of the interplane vias. Additional interplane vias also increase the vertical interconnect density which is undesirable. The routing congestion also increases as these vias typically can block the metal layers within a plane, adversely affecting the length of the allowed intervals for the remaining nets.

To better explain the different types of moves illustrated in Fig. 5, consider the cross-section of these interplane moves as shown in Fig. 6. In Fig. 6, $l_j$ and $l_{j+1}$ are the length of the interconnect segments in the $x$-direction that belong to the plane $j$ and $j+1$, respectively (not necessarily adjacent). In Figs. 5a and 6a, the interplane via is shifted to the left to decrease the
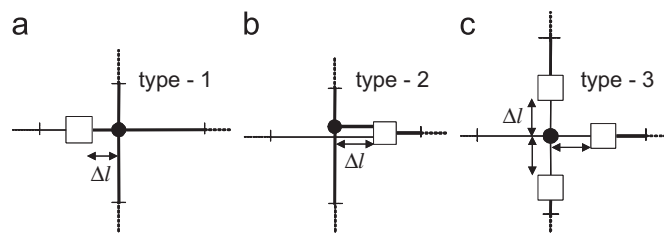


Fig. 5. Different interplane via moves: (a) *type*-1 move (allowed), (b) *type*-2 move (allowed), and (c) *type*-3 move (prohibited).
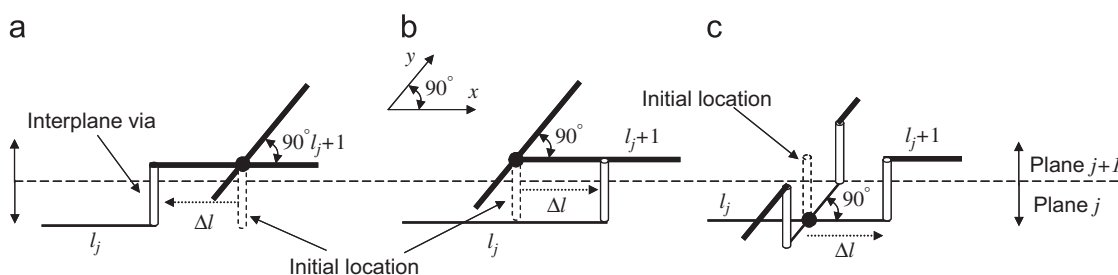


Fig. 6. Cross-section of the interplane moves shown in Fig. 5.

length of segment $l_j$. If $x$–$y$ routing is permitted within one metal layer (i.e., segments in both the $x$- and $y$-direction within the same metal layer are allowed), no intraplane via is required to preserve connectivity. If, however, only one routing direction is allowed in each metal layer, the segments on plane $j+1$ occupy more than one layer and, therefore, an intraplane via (shown by the dot) is required. In Figs. 5b and 6b, the via is shifted by $\Delta l$ to the right, extending segment $l_j$ on plane $j$ by $\Delta l$. An additional segment $\Delta l$ is required on plane $j+1$ to connect the segments routed in the $y$-direction with the interplane via. As previously described, an intraplane via may also be required. Alternatively, the overhead of segment $\Delta l$ on plane $j+1$ can be avoided by adding interplane vias, as shown in Figs. 5c and 6c. Such a type of move, however, is not allowed in order to maintain a low interplane via density.

The constraints described by Eqs. (1) and (2) are extended to each sink and via of the interconnect tree. The constraint in Eq. (1) is adapted to consider any increase in wire length that can result from a *type*-2 move for some branches of the tree. Consequently, for a selected move direction $d_i$ for via $v_j$ and path $P_{s_{pq}v_j}$, Eqs (1) and (2) are, respectively,

$$L_{s_{pq}} = l_1 + l_{v1} + \cdots + l_j + l_{vj} + l_{j+1} + \cdots + l_n, \quad \text{for a } type\text{-1 move.} \tag{11}$$

$$L_{s_{pq}} = l_1 + l_{v1} + \cdots + l_j + l_{vj} + \Delta l + l_{j+1} + \cdots + l_n, \quad \text{for a } type\text{-2 move.} \tag{12}$$

$$l_{j\,\min} \leqslant l_j \leqslant l_{j\,\min} + l_{di}, \quad \text{for a } type\text{-1 move.} \tag{13}$$

$$l_{j\,\min} \leqslant l_j \leqslant l_{j\,\min} + l_{di} + \Delta l, \quad \text{for a } type\text{-2 move.} \tag{14}$$

Consequently, the constrained optimization problem for placing a via within an interplane interconnect tree can be described as

(P1)      **minimize** $T_w$,
           **subject to** (1) $\forall$ sink $s_{pq}$, (11)−(14).

By similar reasoning as for two-terminal nets, (P1) typically includes an indefinite quadratic form $\mathbf{l}^T\mathbf{Al}$, where $\mathbf{A}$ is the matrix described in Eq. (6) adapted for interconnect trees. Certain transformations can be applied to convert (P) and (P1) into a convex optimization problem [31]; the objective functions, however, are no longer quadratic. Alternatively, (P) and (P1) can be cast as a geometric programming problem. Geometric programs include optimization problems for functions and inequalities of the following form:

$$g(y) = \sum_{j=1}^{M} s_j y_1^{a_{1j}} y_2^{a_{2j}} \cdots y_n^{a_{nj}}, \tag{15}$$

$$s_j y_1^{a_{1j}} y_2^{a_{2j}} \cdots y_n^{a_{nj}} \leqslant 1, \tag{16}$$

where the variables $y_j$s and coefficients $s_j$s must be positive and the exponents $a_{ij}$s are real numbers. Although equality constraints are not allowed in standard geometric problems, (P) and (P1) can be solved as generalized geometric programs as described in Ref. [32], where globally optimum solutions are determined. In Section 3, an efficient heuristic for placing vias in two-terminal nets is presented.

## 3. Two-terminal net via placement heuristic

In this section, a heuristic for the near-optimal interplane via placement of two-terminal nets that include several interplane vias is described. The key step in the heuristic is that the optimum via placement depends primarily upon the length of the allowed interval (that is estimated or known after an initial placement) rather than the exact location of the via. Consider the interplane interconnect line shown in Fig. 2, where the optimum location for via $j$ that connects interconnect segments $j$ and $j+1$ is to be determined. With respect to this via, the critical point (i.e., $\partial T_{el}/\partial x_j = 0$) of the Elmore delay is

$$x_j = -\left[\frac{l_{vj}(r_j c_{vj} - r_{vj} c_{j+1} + r_{j+1} c_{j+1} - r_j c_{j+1}) + R_{uj}(c_j - c_{j+1}) + \Delta x_j(r_j - r_{j+1})c_{j+1} + C_{dj}(r_j - r_{j+1})}{r_j c_j - 2r_j c_{j+1} + r_{j+1} c_{j+1}}\right], \tag{17}$$

where $R_{uj}$ and $C_{dj}$ are the upstream resistance and downstream capacitance, respectively, of the allowed interval for via $j$ (see also Table 1), as shown in Fig. 2. The Elmore delay of the line with respect to $x_j$ can be either a convex or a concave function [25]. The remaining discussion in this section applies to the case where the Elmore delay of the line is a convex function with respect to $x_j$. A similar analysis can be applied for the concave case.

In Eq. (17), the optimum via location $x_j^*$ is a monotonic function of $R_{uj}$ and $C_{dj}$. The sign of the monotonicity depends upon the interconnect impedance parameters of the segments $j$ and $j+1$ connected by via $j$. As the length of the allowed intervals for all of the vias is constrained by Eq. (3), the minimum and maximum values of $R_{uj}$ and $C_{dj}$ can be readily

determined, permitting the values of $x_j^*$ for these extrema, $x_{j\,\min}^*$ and $x_{j\,\max}^*$, to be evaluated. Due to the monotonic dependence of $x_j$ on $R_{uj}$ and $C_{dj}$, the optimum location for via $j$, $x_j^*$, lies within the range delimited by $x_{j\,\min}^*$ and $x_{j\,\max}^*$. The following cases can be distinguished, while a proof of the heuristic is provided in Appendix A.

(i) If $x_{j\,\max}^* \leqslant 0$, $x_j^* = 0$, and the optimum via location coincides with the lower bound of the interval as defined by Eq. (3).

(ii) If $x_{j\,\min}^* \geqslant \Delta x_j$, $x_j^* = \Delta x_j$, and the optimum via location coincides with the upper bound of the interval as defined by Eq. (3).

(iii) If $\Delta x_j \geqslant x_{j\,\min}^* \geqslant 0$ and $\Delta x_j \geqslant x_{j\,\max}^* \geqslant 0$, the bounded interval as defined by Eq. (3) reduces to

$$0 \leqslant x_{j\,\min}^* \leqslant x_j \leqslant x_{j\,\max}^* \leqslant \Delta x_j. \qquad (18)$$

In this case, the via location cannot be directly determined. However, by iteratively decreasing the range of values for $x_j^*$, the near-optimal location for via $j$ can be achieved.

The following example is used to demonstrate that the physical domain for $x_j^*$ iteratively decreases to a single point, the optimum via location. Consider segment $i$, $j$, and $k$ in Fig. 2, where segments $i$ and $k$ are located upstream and downstream of segment $j$, respectively. From Eqs. (2) and (3), the minimum and maximum values of $R_{ui}^0$, $R_{uj}^0$, $R_{uk}^0$, $C_{di}^0$, $C_{dj}^0$, and $C_{dk}^0$ are determined, where the superscript represents the number of iterations. Assume that $x_{\min}^{*0}$ and $x_{\max}^{*0}$ are obtained from Eq. (17) to satisfy Eq. (18) for all three segments, $i$, $j$, and $k$. Note that this condition is assumed to illustrate the convergence of the heuristic and is not a requirement for segments $i$ and $k$. Segments $i$ and $k$ can satisfy any of the cases (i)–(iv) of the proposed heuristic. As the range of values for the via location of segments $i$ and $k$ decreases according to Eq. (18), the minimum (maximum) value of the upstream resistance and downstream capacitance of segment $j$ increases (decreases), i.e., $R_{uj\,\min}^0 < R_{uj\,\min}^1$, $C_{dj\,\min}^0 < C_{dj\,\min}^1$, $R_{uj\,\max}^1 < R_{uj\,\max}^0$, and $C_{dj\,\max}^1 < C_{dj\,\max}^0$. Due to the monotonicity of $x_j^*$ on $R_{uj}$ and $C_{dj}$, $x_{j\,\min}^{*0} < x_{j\,\min}^{*1}$ and $x_{j\,\max}^{*1} < x_{j\,\max}^{*0}$. The range of values for $x_j^*$ therefore also decreases and, typically, after two or three iterations, the optimum location for the corresponding via is determined.

(iv) If $x_{j\,\min}^* \leqslant 0$ and $x_{j\,\max}^* \geqslant \Delta x_j$, the via location cannot be directly determined. Additionally, the bounding interval cannot be reduced. Consequently, some loss of optimality occurs. This departure from the optimal, however, is smaller than 0.03% as shown by the optimization results described in Section 6. In all of the simulations, less than 1% of the interconnect instances yield boundary values for $x_j^*$ such that the inequalities $x_{j\,\min}^* \leqslant 0$ and $x_{j\,\max}^* \geqslant \Delta_j$ are satisfied.

The inequalities in (iv) are usually satisfied, where the length of the allowed interval for a via is relatively small as compared to the length of the allowed intervals for the remaining vias. A non-optimal via placement for that interconnect segment does not significantly affect the overall delay of the line. Furthermore, the non-optimal placement of a via does not necessarily affect the optimal placement of the remaining vias. For example, any via placed according to the criteria described in (i) and (ii) is not affected by the placement of the remaining vias. Therefore, as noted earlier, the length of the allowed intervals rather than the exact location of the vias is the key factor in determining the optimum via locations. The same fundamental notion is used in a heuristic to place the interplane vias in the case of interconnect trees in 3-D circuits, which is described in Section 4.

## 4. Multi-terminal net via placement heuristic

A near-optimal heuristic for placing vias in interconnect trees in 3-D circuits is presented in Section 4.1. A variant of the problem in Section 4.1 where the interplane vias are placed to minimize the delay of a single critical branch of a tree is discussed in Section 4.2.

### 4.1. Interconnect trees

In this section, placing an interplane via within an interconnect tree in a 3-D circuit to minimize the summation of the weighted Elmore delay of the branches of the tree is investigated. Since several moves for the interplane vias are possible, as discussed in Section 2.2, the expressions that determine the via location are different in multi-terminal nets. To determine which type of move for those vias with a connectivity degree greater than two can yield a decrease in the delay of a tree, the following conditions apply.

**Condition 1.** If $r_j > r_{j+1}$, only a *type*-1 move for $v_j$ can reduce the delay of a tree.

**Proof.** The proposition is analytically proven in Appendix B. The condition can also be intuitively explained. A *type*-2 move increases by $\Delta l$ the length of segment $l_j$. The reduction in $l_{j+1}$ is counterbalanced by the additional segment with

length $\Delta l$ on the $j+1$ plane (see Fig. 5b). Consequently, the total capacitance of the tree increases. If Condition 1 is satisfied, a *type*-2 move also increases the total resistance of the tree and, therefore, the delay of the tree will only increase by this via move. □

**Condition 2.** For a candidate direction $d_i$, if $r_j < r_{j+1}$ and

$$\sum_{\forall s_{pq} \in P_{s_{pq}v_jd_i}} w_{s_p}(r_j + r_{j+1})C_{dv_j\overline{d_i}} \leqslant \sum_{\forall s_{pq} \in P_{s_{pq}v_jd_i}} w_{s_p}(r_{j+1} - r_j)C_{dv_jd_i} \tag{19}$$

is satisfied, a *type*-2 move can reduce the delay of the tree.

**Proof.** The proof of this condition is also intuitive. All of the interconnect segments located upstream from $v_j$ see an increase in the capacitance by $c_j\Delta l$, increasing the delay of each downstream sink $v_j$. Consequently, only a reduction in the resistance can decrease the delay of the tree. Alternatively, the sinks located downstream from the candidate direction $d_i$ see a reduction in the upstream resistance by $(r_j - r_{j+1})\Delta l < 0$, while the sinks downstream from the other directions see an increase in the upstream resistance by $(r_j + r_{j+1})\Delta l$. For a *type*-2 move, resulting in a decrease in the delay of the tree, both the weighted sum of these two components as determined by the weight of the sinks and the downstream capacitances must be negative. □

Condition 2 is evaluated for each via of a tree with degree greater than two. If Eq. (19) is satisfied for more than one direction, the direction that produces the greatest value of the RHS of Eq. (19) is considered the optimum direction for that via. Finally, note that both conditions 1 and 2 are only necessary and not sufficient conditions. Following the notation listed in Table 1, the critical point for a via connecting two segments on planes $j$ and $j+1$ and satisfying condition 1 is

$$x_{\text{type-1}} = \frac{\left(\sum\limits_{v_i \in U_{1j}} \sum\limits_{s_m \in P_{s_m U_{ij}}} w_{s_m}R_{u_{ij}} + \sum\limits_{s_p \in P_{s_p v_j}} w_{s_p}R_{u_j}\right)(c_{j+1} - c_j) - l_{v_j}\left(r_jc_{v_j} - r_{v_j}c_{j+1}\right) + (r_j - r_{j+1})\left(c_{j+1}l_{d_w} + C_{dv_j}\right)}{\sum\limits_{s_p \in P_{s_p v_j}} w_{s_p}\left(r_jc_j + r_{j+1}c_{j+1} - 2r_jc_{j+1}\right)}. \tag{20}$$

For a *type*-2 move along a candidate direction $d_i$, the critical point for a via connecting two segments on planes $j$ and $j+1$ is

$$x_{\text{type-2}} = \frac{\sum\limits_{s_p \in P_{s_p v_jd_i}} w_{s_p}r_{j+1}\left(C_{dv_jd_i} + c_{j+1}l_{d_i}\right) - \sum\limits_{v_i \in U_{1j}} \sum\limits_{s_m \in P_{s_m U_{ij}}} w_{s_m}R_{u_{ij}}c_k - \sum\limits_{s_p \in P_{s_p v_j}} w_{s_p}\left(r_jc_{v_j} - c_{j+1}l_{d_i} + C_{d_j} + r_{j+1}C_{dv_j\overline{d_i}} + R_{u_j}c_k\right)}{\sum\limits_{s_p \in P_{s_p v_j}} w_{s_p}\left(r_jc_j + r_{j+1}c_{j+1}\right)}. \tag{21}$$

## 4.2. Single critical sink interconnect trees

There are cases where the delay of only one branch of a tree is required to be optimized. Although the heuristic presented in Section 4.1 can be used for this type of tree, a computationally simpler, yet accurate, optimization procedure for single critical net trees is possible and is described here. Denoting by $s_c$, the critical sink of the tree, the weight for this sink $w_{s_c}$ is one, while the assigned weight for the remaining sinks are zero. Consequently, the expression that minimizes delay is significantly simplified. In addition, the approach is different as compared to the optimization problem discussed in Section 4.1. More specifically, the interplane vias that belong to the critical branch (the on-path vias) are optimized according to the heuristic for two-terminal nets. There is no need to test conditions 1 and 2 for these vias, as any *type*-2 move only occurs in the direction that includes the critical sink. Regarding those vias that are not part of the critical path (the off-path vias), these vias are placed to minimize the capacitance of the tree. This situation occurs because the non-critical sinks of the tree only contribute as capacitive loads to the delay of the critical sink.

The location of the off-path vias are readily determined since the impedance characteristics of the interconnect segments are known. Note that in this sense, the placement of the off-path vias is always optimal. Any loss of optimality is caused by the placement of the on-path vias. As the near-optimal two-terminal net heuristic is used for placing the on-path vias, the loss of optimality is negligible. In Section 5, these heuristics are used to develop efficient algorithms for placing vias in two-terminal and multi-terminal nets in 3-D ICs.

## 5. Via placement algorithms

Efficient near-optimal algorithms for placing vias among interplane interconnects are presented in this section. Based on the aforementioned heuristic for two-terminal nets, an efficient algorithm is presented for two-terminal nets in Section 5.1. A second algorithm that considers interplane interconnect trees is presented in Section 5.2. A third algorithm that places interplane vias to minimize the delay for the particular case of interconnect trees with a single critical branch is discussed in Section 5.3.

### 5.1. Two-terminal net near-optimal via placement algorithm (TTVPA)

The heuristic described in Section 3 has been used to implement an algorithm that exhibits near-optimal via placement for two-terminal interplane interconnects in 3-D ICs, producing significantly lower computational time as compared to general optimization solvers. The pseudocode of the heuristic algorithm is illustrated in Fig. 7. The input to the algorithm is the minimum length of the interconnect segments and the length of the allowed intervals. In the first step of the algorithm, the maximum and minimum upstream (downstream) resistance (capacitance) for each allowed interval is determined. In the following steps, the range of values for the optimum via location as given by Eq. (17) is evaluated. In step five, these values are compared to the inequalities described in Section 3. If a via location is determined in this step, the via is marked as processed and the capacitance and resistance arrays are updated. If, after a number of iterations, there are unprocessed vias, the vias are placed, in step 14, at the center of the corresponding allowed intervals and the algorithm terminates.

A bottom-up approach is followed for placing the vias. The downstream capacitance of the allowed interval for the via located on the last level of the tree is constant; therefore, the upstream resistance of this via can only vary due to the location of the other vias. With such an approach, a low computational time is achieved as most of the vias are placed within one iteration of the algorithm. Alternatively, a top-down approach can also be adopted, where the via on the first plane of the 3-D circuit has a constant upstream resistance. The location of the remaining vias affects only the downstream capacitance of this via. If processing the vias in step 4 of the algorithm is randomly performed, additional iterations for placing each via can be required since both the downstream capacitance and upstream resistance can vary.

As discussed in Section 2.2, vias can span more than one physical plane, where the allowed intervals are different for each plane. The location for these vias can be determined by two approaches. First, the smallest allowed interval can be considered as the allowed interval for each plane traversed by the via. Such an approach, however, is not efficient as the delay of a net can be further decreased if different allowed intervals are considered. Specifically, a stacked via can be treated

---

| *Two-Terminal Via Placement Algorithm:* $(\mathbf{l_{min}}, \mathbf{\Delta x})$ |
|---|
| 1.       Determine $\mathbf{C_{dmin}}$, $\mathbf{C_{dmax}}$, $\mathbf{R_{umin}}$, $\mathbf{R_{umax}}$ |
| 2.       while $\mathbf{S} \neq \varnothing$ |
| 3        if iter < max_iter |
| 4.          $s_j \leftarrow$ an unprocessed via |
| 5.          obtain $x^*_{j\min}$ and $x^*_{j\max}$ from eq. (13) |
| 6.          check for the inequalities in *(i) – (iv)* |
| 7.          if $s_j$ is optimized *(cases i-ii)* |
| 8.            store optimum via location |
| 9.            $\mathbf{S} \leftarrow \mathbf{S} - \{s_j\}$ |
| 10.           update $\mathbf{C_{dmin}}$, $\mathbf{C_{dmax}}$, $\mathbf{R_{umin}}$, $\mathbf{R_{umax}}$ |
|          elseif $\Delta x_j$ decreases *(case iii)* |
| 11.           update $\mathbf{l_{jmin}}$, $\mathbf{C_{dmin}}$, $\mathbf{C_{dmax}}$, $\mathbf{R_{umin}}$, $\mathbf{R_{umax}}$ |
|          else *(case iv)* |
| 12.           go to step 3 |
|        else *(the non-optimized vias)* |
| 13.          place via in the center of the allowed interval |
| 14.          store via location |
| 15.          $\mathbf{S} \leftarrow \mathbf{S} - \{s_j\}$ |
| 16.     exit |

Fig. 7. Pseudocode of the proposed near-optimal via placement algorithm for two-terminal nets (TTVPA).

as a set of interplane vias where each via connects two adjacent planes and the minimum length of the interconnect segment between these vias is set to zero (i.e., $l_{min}$ in Eq. (2) is equal to zero). The additional horizontal segments correspond to additional variables; however, the delay of the tree can be further reduced. Note that in either of these two approaches, the algorithm is not modified, only the input vectors are different. Consequently, the proposed algorithm can handle vias spanning multiple physical planes with different allowed intervals on each plane.

## 5.2. Interconnect tree near-optimal via placement algorithm (ITVPA)

The via placement optimization algorithm for multi-terminal nets is presented in this section. The input to the algorithm is an interplane interconnect tree where the minimum length of the segments, the weight of the sinks, and the length of the allowed intervals are provided. Pseudocode of the algorithm is shown in Fig. 8. Due to the different types of moves that are possible in interplane interconnect trees, the candidate direction for via placement is initially determined in steps 1–5. The *move_type* routine operates from the leaf to the root, where the type and direction of the move of each via of the tree has degree greater than two. Conditions 1 and 2 are tested for each via and direction. In step 6, the *optimize_tree_delay* routine places the vias within a tree such that (10) is minimized.

This routine is based on the algorithm used for two-terminal nets as the objective function is of the same form. The process of placing the vias, however, proceeds with certain modifications. A bottom-up approach is applied starting from the last level of the tree towards the root of the tree. For each level of the tree, those vias that belong to this level are successively placed. Within each level, the via being processed is selected according to the order of the paths of the tree produced during the tree generation step. After placing the vias within a level of the tree, the upstream resistance and downstream capacitance matrices are updated. Those vias successfully placed during this iteration of the algorithm are marked as processed. In the case where some vias are not optimally placed and the maximum number of iterations has been reached, the same criterion used in the two-terminal net algorithm is adopted for determining the location of these vias.

## 5.3. Single critical branch near-optimal via placement algorithm (SCBVPA)

Although the heuristic presented in the previous section can be used to improve the delay of trees with a single critical path, a simpler optimization procedure for single critical net trees is described in this section. The input to the proposed algorithm is a description of the interplane interconnect tree where the minimum length of the segments, the weight of the sinks, and the length of the allowed intervals are provided. Pseudocode of the algorithm is shown in Fig. 9. In steps 1–3,

```
Interconnect Tree Via Placement Algorithm: (l_min, Δx, l_di, w_si)

1.    foreach physical plane i, i = n → 1
2.      foreach interplane via j on plane i
3.          if via_degree > 2
4.              move_type(j)
            else
5.              goto step 2
6.    optimize_tree_delay()
7.    exit
```

Fig. 8. Pseudocode of the proposed near-optimal via placement algorithm for interconnect trees (ITVPA).

```
Single Critical Branch Via Placement Algorithm: (l_min, Δx, l_di, w_si)

1.    foreach off path via j
2.      set via j to min. capacitance location
3.    foreach on path via i
4.      direction_move(i)
5.    optimize_tree_delay()
6.    exit
```

Fig. 9. Pseudocode of the proposed near-optimal via placement algorithm for single critical branch interconnect trees (SCBVPA).

each of the off-path vias is placed at the minimum capacitance location within the corresponding allowed interval. The *direction_move* routine sets the direction of the on-path via to that direction, which includes the critical sink of the tree. In step 5, the *optimize_tree_delay* routine is utilized to determine the location of the on-path vias. As previously mentioned, any loss of optimality for this type of tree results from the heuristic used to place the via in two-terminal nets. As shown in Section 6, however, this heuristic produces results similar to optimization solvers, and the proposed algorithm naturally exhibits significantly lower computational time as compared to general purpose solvers. Results for other types of interplane interconnect are also presented.

## 6. Test cases for via placement algorithms

Simulation and analytic results for various interplane interconnects in 3-D ICs are presented in this section. The interplane interconnects for a different number of physical planes are analyzed. The impedance characteristics of the horizontal segments and vias are extracted for several interconnect structures using a commercial impedance extraction tool [33]. Copper interconnect is assumed with an effective resistivity of $2.2\,\mu\Omega$ cm. Based on the extracted impedances, the resistance and capacitance of the horizontal segments range from 25 to $125\,\Omega$/mm and 100 to 300 fF/mm, respectively, for a 90 nm CMOS technology node [34,35]. The cross-section of the vias is $1\,\mu m \times 1\,\mu m$, with 1-$\mu m$ spacing from the surrounding horizontal metal layers, assuming an SOI process as described in Ref. [25]. For all of the interconnect structures, the total and minimum length of each horizontal segment is randomly generated. For simplicity, all of the vias connect the segments of two adjacent physical planes (i.e., $m = n$). In Section 6.1, simulation and analytic results for two-terminal nets with multiple vias are reported. Results for multi-terminal nets and single critical sink interconnect trees are presented in Section 6.2. Limitations of the algorithms and the impact of routing congestion on the quality of the results are discussed in Section 6.3. The savings in delay that can be achieved by optimally placing the vias is demonstrated for different via placement scenarios.

### 6.1. Two-terminal net with multiple vias

In this section, the improvement in delay achieved by placing the vias is demonstrated and simulation results from the proposed via placement algorithms for two-terminal nets (TTVPA) are presented. SPICE delay simulations are reported in Table 2. Each horizontal interconnect segment is modeled as a 50 $RC$ $\pi$-segment, while each interplane via is modeled by ten $RC$ $\pi$-segments. Consequently, if an interplane interconnect traverses four planes, where each via connects the horizontal interconnect segments of two adjacent planes, a total of 230 $RC$ $\pi$-segments are utilized to model the entire interconnect length. The delay of the line $T_1$, where the vias are placed at the center of the allowed intervals, is listed in column 2. The delay $T_2$, listed in column 3, corresponds to the line delay for random via placement. The minimum interconnect delay $T_{min}$, where the vias are optimally placed, is listed in column 4. The via locations or, equivalently, the length of the horizontal segments, are determined from the algorithm described in Section 5.1. The improvement in delay as

Table 2
SPICE simulation results demonstrating the delay savings achieved by near optimal via placement

| Length (μm) | $T_1$ (ps) | $T_2$ (ps) | $T_{min}$ (ps) | Improvement (%) | $n$ |
|---|---|---|---|---|---|
| 1017 | 12.35 | 12.64 | 11.42 | 8.14 (10.68) | 4 |
| 1180 | 13.37 | 14.42 | 12.33 | 8.43 (16.95) | 4 |
| 849 | 11.00 | 11.71 | 10.27 | 7.11 (14.02) | 4 |
| 969 | 13.52 | 14.96 | 12.12 | 11.55 (23.43) | 4 |
| 967 | 12.38 | 12.59 | 11.72 | 5.63 (7.42) | 4 |
| 1612 | 18.54 | 19.85 | 17.24 | 7.54 (15.14) | 5 |
| 1537 | 20.80 | 19.47 | 19.37 | 7.38 (0.52) | 5 |
| 1289 | 17.78 | 18.43 | 16.45 | 8.09 (12.04) | 5 |
| 1443 | 18.77 | 19.54 | 18.07 | 3.87 (8.14) | 5 |
| 1225 | 16.97 | 18.33 | 15.62 | 8.64 (17.35) | 5 |
| 2118 | 30.52 | 34.81 | 26.44 | 15.43 (31.66) | 7 |
| 2130 | 27.92 | 27.32 | 25.94 | 7.63 (5.32) | 7 |
| 1961 | 28.49 | 30.67 | 26.16 | 8.91 (17.24) | 7 |
| 2263 | 35.58 | 40.11 | 31.31 | 13.64 (28.11) | 7 |
| 2174 | 32.31 | 30.34 | 29.16 | 10.80 (4.05) | 7 |
| Average improvement | | | | 8.85 (14.14) | |

The resistance and capacitance per unit length of the vias are $r_{vi} = 6.7\,\Omega$/mm and $c_{vi} = 6$ pF/mm, respectively. The length of the vias is $l_{vi} = 20\,\mu m$. The driver resistance is $R_S = 15\,\Omega$ and the load capacitance is $C_L = 100$ fF. The length of the allowed intervals is $\Delta x_i = 200\,\mu m$.

compared to the case where the vias are placed at the center of the line is listed in column 5. The number in parentheses corresponds to the improvement in delay over a random via placement. Note that the variation in the improvement in delay changes significantly for those listed instances, although the interconnect lengths are similar and the load capacitance and driver resistance are the same. This considerable variation demonstrates the strong dependence of the line delay on the impedance characteristics of the segments of the line and supports modeling the interplane interconnect as a group of non-uniform segments. Additionally, depending upon the impedance characteristics of the line segments, placing a via at the center of the allowed intervals is, for certain instances, near-optimal, explaining why the improvement in delay is not significant in these instances. The same characteristic applies to those cases where a random placement is close to the optimum placement. Nevertheless, as listed in Table 2, an improvement of up to 32% is observed for relatively short interconnects, demonstrating that an optimum via placement can significantly enhance the speed of 3-D circuits (in addition to the primary benefit of reduced wire length and therefore lower power).

The algorithm presented in Section 5.1 is compared both in terms of optimality and efficiency to two optimization solvers. The first solver, YALMIP [36], is a general optimization solver that supports geometric programming while GLOPTIPOLY [37] is an optimization solver for non-convex polynomial functions. YALMIP and GLOPTIPOLY produce identical solutions. Due to the excessive computational time of GLOPTIPOLY (greater than three orders of magnitude as compared to YALMIP), however, only comparisons with YALMIP are reported. Optimization results are listed in Table 3 for different values of $\Delta x$ ranging from 50 to 300 μm.

As reported in columns 8 and 9 of Table 3, TTVPA exhibits high accuracy as compared to YALMIP. These results are independent of the number of planes that comprise the 3-D interconnect, demonstrating that the proposed algorithm yields optimum solutions for most interconnect instances. In addition, for those cases where some of the vias are not optimally placed, the loss of optimality is insignificant (as previously discussed in Section 3). For the interconnects reported in Table 3, the computational time of YALMIP and TTVPA are reported in Table 4. The runtime ratio of YALMIP to TTVPA is listed in column 5. TTVPA is approximately two orders of magnitude faster than YALMIP. The complexity of TTVPA has

Table 3

Optimization results for various two-terminal interplane interconnects and number of physical planes $n$

| $n$ | Average interconnect length (μm) | $\Delta x_i$s (μm) | Delay improvement (%) | | | | Deviation of TTVPA from optimum solution (%) | | Instances |
|---|---|---|---|---|---|---|---|---|---|
| | | | Vias placed in the center | | Random via placement | | Average | Maximum | |
| | | | Average | Maximum | Average | Maximum | | | |
| 3 | 270 | 50 | 3.36 | 11.10 | 5.88 | 22.23 | 0 | 0.005 | 10,000 |
| 3 | 520 | 100 | 4.59 | 17.63 | 8.02 | 35.92 | 0 | 0.008 | 10,000 |
| 3 | 1020 | 200 | 5.90 | 23.12 | 10.27 | 47.07 | 0 | 0.013 | 10,000 |
| 4 | 405 | 50 | 4.02 | 13.01 | 6.00 | 25.97 | 0 | 0.006 | 10,000 |
| 4 | 781 | 100 | 5.26 | 16.95 | 7.91 | 34.11 | 0 | 0.002 | 10,000 |
| 4 | 1155 | 150 | 5.94 | 21.61 | 8.89 | 44.46 | 0 | 0.011 | 10,000 |
| 5 | 540 | 50 | 4.48 | 13.73 | 6.16 | 27.49 | 0 | 0.005 | 10,000 |
| 5 | 1040 | 100 | 5.69 | 17.97 | 7.79 | 35.82 | 0.0001 | 0.012 | 10,000 |
| 5 | 1541 | 150 | 6.35 | 22.36 | 8.63 | 46.26 | 0.0001 | 0.017 | 10,000 |

Table 4

Computational time for placing the vias of those interconnects reported in Table 3

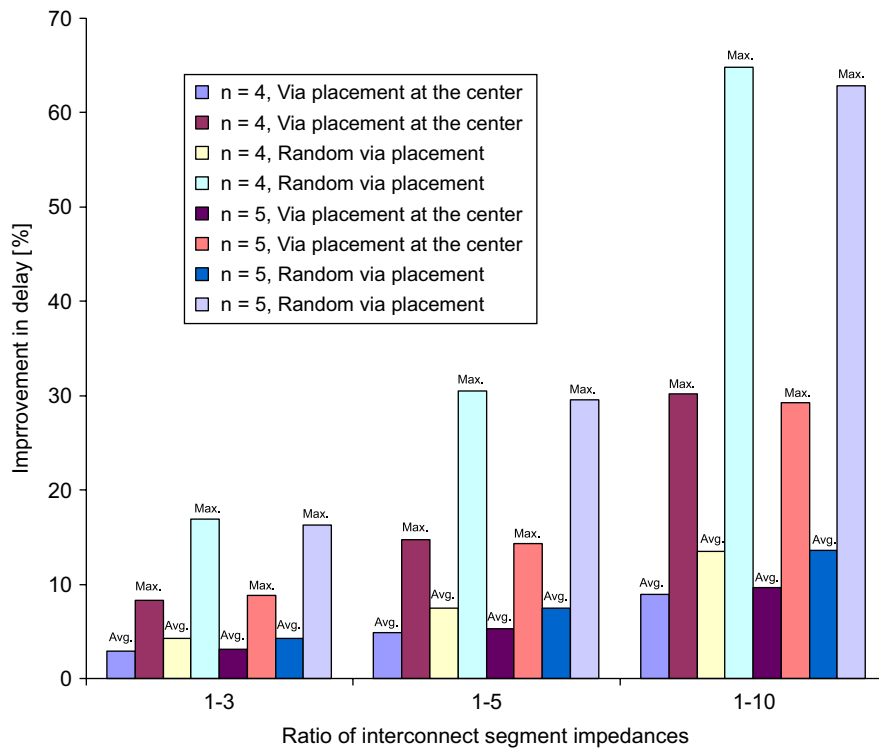| $n$ | Average interconnect length (μm) | Runtime (s) | | Runtime ratio × times | Instances |
|---|---|---|---|---|---|
| | | YALMIP | TTVPA | | |
| 3 | 270 | 1072.34 | 7.57 | 141.69 | 10,000 |
| 3 | 520 | 1051.80 | 7.08 | 148.60 | 10,000 |
| 3 | 1020 | 1076.45 | 7.39 | 145.58 | 10,000 |
| 4 | 405 | 7550.26 | 36.06 | 209.36 | 10,000 |
| 4 | 781 | 19716.58 | 34.33 | 574.30 | 10,000 |
| 4 | 1155 | 1602.73 | 12.78 | 125.38 | 10,000 |
| 5 | 540 | 2247.20 | 20.04 | 112.14 | 10,000 |
| 5 | 1040 | 6340.68 | 18.91 | 335.31 | 10,000 |
| 5 | 1541 | 10437.60 | 19.00 | 549.35 | 10,000 |

Fig. 10. Average and maximum improvement in delay for different range of interconnect segment resistance and capacitance ratios for two different via placement scenarios. The vias are placed at the center of the allowed intervals and the vias are randomly placed.

an almost linear dependence on the number of interplane vias. As depicted in Fig. 7, each via is typically processed once; otherwise, a maximum of two to five iterations are required to place a via.

As shown in Table 3, the savings in delay from the near-optimal via placement strongly depends upon the length of the allowed intervals. For example, doubling the length of the allowed intervals for via placement increases almost two-fold the maximum improvement in delay. As the length of the allowed intervals increases, the constraints in Eq. (2) are relaxed and a greater performance benefit from optimally placing the vias is achieved.

The effect of the non-uniformity of the interplane interconnects on the improvement in delay is graphically illustrated in Fig. 10, where the improvement in delay for interplane interconnects spanning four and five physical planes is depicted. The average savings in delay of highly non-uniform interconnects (i.e., $r_{(i+1)}/r_i = 1$–$10$ and $c_i/c_{(i+1)} = 1$–$10$) can be significant, approaching 10% and 13% for a moderately sized length, where the vias are placed at the center of the allowed intervals and are randomly placed, respectively. The maximum improvement can exceed 60%, as shown in Fig. 10.

## 6.2. Interconnect trees

In Table 4, optimization results for interconnect trees with various number of leaves and planes (i.e., tree depth) are reported. The optimality and efficiency of ITVPA is similar to that of TTVPA, as the optimization routine for ITVPA is approximately the same as for TTVPA. The runtime of ITVPA is similar to that of TTVPA and has a linear dependence on the number of vias within the tree. The complexity of the additional routine that determines the direction and type of move for each via is linear with the number of vias within the tree.

The improvement in the delay of the interconnect trees is listed in columns 6–9 of Table 5. The results are compared to the case where the vias are initially placed at the center of the allowed interval (i.e., $x_i = \Delta x_i/2$) and to the case where the vias are placed at the lower edge of the allowed interval (i.e., $x_i = 0$). The improvement in performance depends upon the length of the allowed interval. This dependence, however, is weak as compared to two-terminal nets. In addition, the improvement in delay is lower than the point-to-point nets for the same allowed length intervals. This reduction in delay improvement occurs for two reasons.

For those vias with degree greater than two, which constitute the majority of interplane vias in interconnect trees, after the type of move for each via is determined, the actual interval length that these vias are allowed to move is $\Delta x_i/2$ and not $\Delta x_i$ (see Fig. 5). Furthermore, in the proposed algorithm, any modifications to the routing tree are strictly confined within the allowed interval such that the routing tree is least affected. This constraint requires an additional interconnect segment for *type*-2 moves. If this constraint is relaxed, an additional interconnect segment is not necessary and the length of the

Table 5
Optimization results for various interplane interconnect trees for different number of sinks and physical planes $n$

| $n$ | No. of sinks | Average branch length (µm) | Average maximum branch length (µm) | $\Delta x_i$s (µm) | Delay improvement (%) | | | | Instances |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | $x_i^* = \Delta x_i/2$ | | $x_i^* = 0$ | | |
| | | | | | Average | Maximum | Average | Maximum | |
| 3 | 4 | 153 | 186 | 50 | 2.72 | 9.33 | 3.79 | 11.25 | 10,000 |
| 3 | 4 | 307 | 376 | 100 | 4.23 | 15.17 | 6.03 | 17.94 | 10,000 |
| 4 | 4 | 208 | 273 | 50 | 1.11 | 3.53 | 2.49 | 5.63 | 5000 |
| 4 | 4 | 828 | 1100 | 200 | 3.12 | 10.29 | 6.42 | 13.50 | 5000 |
| 4 | 4 | 1243 | 1650 | 300 | 4.07 | 14.15 | 7.76 | 19.38 | 5000 |
| 4 | 8 | 431 | 569 | 100 | 3.90 | 13.24 | 7.71 | 19.68 | 10,238 |
| 5 | 4 | 264 | 362 | 50 | 1.25 | 3.83 | 2.40 | 5.89 | 5000 |
| 5 | 4 | 1054 | 1452 | 200 | 3.62 | 11.55 | 6.56 | 12.04 | 5000 |
| 5 | 4 | 791 | 1089 | 300 | 3.90 | 11.61 | 6.95 | 19.34 | 5000 |
| 5 | 8 | 454 | 660 | 50 | 0.90 | 2.69 | 2.27 | 4.98 | 5000 |
| 5 | 8 | 521 | 738 | 100 | 1.78 | 5.55 | 4.33 | 8.40 | 5000 |
| 5 | 8 | 779 | 1111 | 150 | 2.38 | 7.44 | 5.67 | 11.90 | 5000 |
| 5 | 8 | 1038 | 1481 | 200 | 2.91 | 8.71 | 6.74 | 12.58 | 5000 |
| 6 | 8 | 306 | 455 | 50 | 1.11 | 3.17 | 2.36 | 4.89 | 5000 |
| 6 | 8 | 615 | 913 | 100 | 2.00 | 5.44 | 4.09 | 9.85 | 5000 |
| 6 | 8 | 922 | 1373 | 150 | 2.72 | 7.01 | 5.43 | 11.72 | 5000 |
| 6 | 8 | 921 | 1371 | 200 | 3.32 | 10.02 | 6.61 | 14.21 | 5000 |
| 6 | 16 | 555 | 845 | 50 | 0.86 | 2.74 | 2.52 | 4.95 | 4970 |
| 6 | 16 | 637 | 934 | 100 | 1.68 | 4.82 | 4.84 | 9.26 | 5059 |
| 6 | 16 | 953 | 1404 | 150 | 2.28 | 6.10 | 6.32 | 12.96 | 5021 |

Table 6
Optimization results for various single critical sink interconnect trees for different number of sinks and physical planes $n$

| $n$ | No. of sinks | Average branch length (µm) | Average maximum branch length (µm) | $\Delta x_i$s (µm) | Delay improvement (%) | | | | Instances |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | $x_i^* = \Delta x_i/2$ | | $x_i^* = 0$ | | |
| | | | | | Average | Maximum | Average | Maximum | |
| 4 | 4 | 341 | 453 | 50 | 2.72 | 8.95 | 3.70 | 14.63 | 5000 |
| 4 | 4 | 1021 | 1363 | 150 | 1.61 | 5.52 | 2.18 | 11.01 | 5000 |
| 4 | 4 | 1368 | 1821 | 200 | 1.36 | 5.49 | 1.92 | 8.59 | 5000 |
| 5 | 4 | 433 | 595 | 50 | 3.09 | 9.37 | 4.55 | 19.44 | 5000 |
| 5 | 4 | 1299 | 1790 | 150 | 1.80 | 5.55 | 2.85 | 13.42 | 5000 |
| 5 | 4 | 1734 | 2391 | 200 | 1.51 | 5.66 | 2.44 | 11.35 | 5000 |
| 5 | 8 | 427 | 612 | 50 | 2.53 | 8.10 | 4.09 | 16.20 | 5000 |
| 5 | 8 | 853 | 1227 | 100 | 1.94 | 7.22 | 2.98 | 12.63 | 5000 |
| 5 | 8 | 1282 | 1845 | 150 | 1.57 | 6.55 | 2.46 | 14.04 | 5000 |
| 5 | 8 | 1711 | 2461 | 200 | 1.33 | 4.81 | 2.25 | 10.44 | 5000 |
| 6 | 8 | 505 | 753 | 50 | 2.88 | 8.90 | 4.39 | 17.8 | 5000 |
| 6 | 8 | 1009 | 1512 | 100 | 2.14 | 6.10 | 3.45 | 15.20 | 5000 |
| 6 | 8 | 1511 | 2265 | 150 | 1.71 | 5.46 | 2.83 | 12.16 | 5000 |
| 6 | 16 | 523 | 779 | 50 | 2.52 | 8.29 | 4.54 | 13.25 | 4963 |
| 6 | 16 | 1045 | 1564 | 100 | 1.91 | 6.69 | 3.10 | 10.53 | 4977 |
| 6 | 16 | 1563 | 2351 | 150 | 1.55 | 5.36 | 2.96 | 12.37 | 4976 |

interconnect segments can be further reduced, resulting in a considerably greater improvement in performance. Maintaining fixed paths limits the efficiency of the proposed algorithms; however, the algorithms are sufficiently general for use with placement and routing tools for 3-D ICs as long as these tools provide an allowed interval for via placement. In addition, interconnect routing can consider other important design objectives such as thermal effects or routing congestion. The proposed algorithm for placing vias in multi-terminal nets can be applied as a subsequent step without significantly affecting the initial layout produced by existing tools.

In Table 6, optimization results for single critical branch interconnect trees are reported. The improvement in the delay of these trees is listed in columns 6–9 of Table 6, as compared to the situation where the vias are initially placed at the

center of the allowed interval (i.e., $x_i = \Delta x_i/2$) and where the vias are placed at the lower edge of the allowed interval (i.e., $x_i = 0$). This improvement is lower than for those interconnect trees listed in Table 5 as only *type*-1 moves can occur for the off-path vias. Indeed, any *type*-2 move for the off-path via only increases the off-path capacitance and, in turn, the delay of the critical leaf. A smaller number of vias can therefore be relocated to reduce the delay of the single critical sink trees. Alternatively, for off-path vias, placing a via at the center of the allowed interval can produce an optimum placement, resulting in a smaller overall improvement in the delay of this type of tree.

### 6.3. Quality of results

The improvement in the delay of interplane two-terminal and multi-terminal nets achieved by optimally placing the vias is demonstrated in the previous subsections. Typically, the larger the allowed interval, the greater the improvement in delay. Consequently, efficient placement tools for 3-D circuits that generate sufficiently large allowed intervals are desired. These intervals can be available space reserved for interplane interconnect routing. For interconnect trees, the improvement in delay is smaller than for two-terminal nets. This decreased improvement in delay is due to the constraint of placing the vias within the allowed intervals in order to minimally affect the routing of the interconnect tree. If the placement of the vias is permitted within an entire region (e.g., rectangle or polygon), a greater decrease in delay can occur. Assigning such a region for placing vias, however, increases the congestion within a 3-D circuit as the same number of vias will compete for sparser routing resources.

Despite the considerably lower computational time of the proposed algorithms, further speed improvement can be achieved if more than one net is simultaneously processed. Although these algorithms support multiple net optimization without significant modification, a single net at a time approach likely yields improved results as the most critical nets can be routed first. Net ordering algorithms [38] can be used to prioritize the routing of those interconnects, permitting the delay of these nets to be considerably reduced. Additionally, since the number of interplane interconnects is small as compared to the number of intraplane interconnects [39], processing these interconnects one at a time will not significantly increase the total computational time.

Thermal issues are expected to be important in 3-D ICs [15], where additional dummy vias are utilized to control the average and peak temperature of the upper planes within a 3-D system. Additionally, thermally aware cell placement improves the heat distribution and removal characteristics. These two techniques are decoupled from the proposed via placement problem which is considered a later step in the design process. Consequently, thermal issues are not strongly connected with the proposed via placement approach. Alternatively, some of the thermal vias within these available spaces can be replaced with signal vias connecting circuits on different planes. Placing the signal vias prior to placing the thermal vias within these regions results in large allowed intervals, thereby improving the effectiveness of the proposed via placement technique. In this case, where the via placement can significantly enhance the thermal profile of a physical plane, the allowed interval for some vias can be decreased or removed. Such a practice, however, trades off performance for thermal management.

## 7. Conclusions

Significant performance improvements can be achieved by optimally placing interplane vias in 3-D circuits. Employing a distributed Elmore delay model, the task of optimally placing vias in two-terminal nets with multiple vias and interconnect trees is presented as a geometric programming problem. Considering physical constraints, near-optimal heuristics are proposed for two-terminal nets with multiple vias and interconnect trees. A near-optimal and efficient algorithm is also presented for two-terminal nets, which is compared to general optimization solvers in terms of computational time. Near-optimal algorithms for interplane via placement in interconnect trees and trees with a single critical sink are also described.

The dependence of the improvement in delay on the length of the allowed intervals for via placement is investigated. The improvement in delay for various interplane via placement scenarios is considered. Delay improvements of up to 16% and 32% are demonstrated for two-terminal nets where the vias are optimally placed as compared to via placement at the center of the allowed intervals and random placement, respectively. Delay improvements in interconnect trees of up to 14% and 19% are demonstrated where the vias are optimally placed as compared to via placement at the center of the allowed intervals and at the lower edge of the allowed intervals, respectively. The proposed algorithms can be embedded into existing and developmental placement and routing methodologies targeting 3-D circuits.

## Appendix A. Analytic proof of the two-terminal heuristic

A formal proof of the two-terminal heuristic for placing interplane vias is described in this appendix. Consider the following expression that describes the critical point (i.e., the derivative of the delay is set equal to zero) for placing a via $v_j$, as illustrated in Fig. A1:

$$x_j^* = -\left[ \frac{l_{vj}(r_j c_{vj} - r_{vj} c_{j+1} + r_{j+1} c_{j+1} - r_j c_{j+1}) + R_{uj}(c_j - c_{j+1}) + \Delta x_j (r_j - r_{j+1}) c_{j+1} + C_{dj}(r_j - r_{j+1})}{r_j c_j - 2 r_j c_{j+1} + r_{j+1} c_{j+1}} \right]. \tag{A.1}$$

From this expression, the critical point $x_j$ is a monotonic function of the upstream resistance and downstream capacitance of the allowed interval for via $v_j$, denoted as $R_{uj}$ and $C_{dj}$, respectively,

$$x_j^* = f(R_{uj}^*, C_{dj}^*). \tag{A.2}$$

These quantities ($R_{uj}^*$ and $C_{dj}^*$) depend upon the location of the other vias along the net and are unknown. However, as the allowed interval for the vias and the impedance characteristics of the line are known, the minimum and maximum values of these impedances, $R_{uj\,min}$, $R_{uj\,max}$, $C_{dj\,min}$, and $C_{dj\,max}$, can be determined. Without loss of generality, assume that $r_j > r_{j+1}$ and $c_j > c_{j+1}$ (the other cases are similarly treated). For this case, the critical point (i.e., $\partial T/\partial x_j = 0$) is a strictly increasing function of $R_{uj}$ and $C_{dj}$. Consequently, the minimum and maximum value for the critical point $x_{j\,min}^*$ and $x_{j\,max}^*$ is determined from, respectively,

$$x_{j\,min}^* = f(R_{uj\,min}, C_{dj\,min}), \tag{A.3}$$

$$x_{j\,max}^* = f(R_{uj\,max}, C_{dj\,max}). \tag{A.4}$$

The final value of the upstream (downstream) capacitance for via $v_j$, which is determined after placing all of the remaining vias of the net denoted as $R_{uj}^*$ ($C_{dj}^*$) within the range, is

$$R_{uj\,min} < R_{uj}^* < R_{uj\,max}, \quad (C_{dj\,min} < C_{dj}^* < C_{dj\,max}). \tag{A.5}$$

Due to the monotonic relationship of the critical point $x_j$ on $R_{uj}$ and $C_{dj}$,

$$x_{j\,min}^* = f(R_{uj\,min}, C_{dj\,min}) < x_j^* = f(R_{uj}^*, C_{dj}^*) < x_{j\,max}^*$$
$$= f(R_{uj\,max}, C_{dj\,max}). \tag{A.6}$$

Consequently, by iteratively decreasing the range of the $x_j^*$ according to (A.6), the location for $v_j$ can be determined.

To better explain this iterative procedure, consider the vias, $v_i$, $v_j$, and $v_k$, shown in Fig. A1 that have not yet been placed. In this example, vias $v_i$ and $v_k$ are assumed to belong to case (iii) of the heuristic. Since the allowed intervals for vias $v_i$, $v_j$, and $v_k$ and the impedance characteristics of the respective horizontal segments are known, the minimum $x_{min}^{*0}$ and maximum $x_{max}^{*0}$ critical point for all of the segments $i$, $j$, and $k$ are obtained. The minimum and maximum values of $R_{ui}^0$, $R_{uj}^0$, $R_{uk}^0$, $C_{di}^0$, $C_{dj}^0$, and $C_{dk}^0$ are determined, where the superscript represents the number of iterations.

From (A.6), the via location of segments $i$ and $k$ is contained within the limits determined by (A.1). As the interval for placing the vias $v_i$ and $v_k$ decreases, the minimum (maximum) value of the upstream resistance and downstream capacitance of segment $j$ increases (decreases), i.e., $R_{uj\,min}^0 < R_{uj\,min}^1$, $C_{dj\,min}^0 < C_{dj\,min}^1$, $R_{uj\,max}^1 < R_{uj\,max}^0$, and $C_{dj\,max}^1 < C_{dj\,max}^0$. Due to the monotonicity of $x_j^*$ (see (A.2)–(A.4) and (A.6)) on $R_{uj}$ and $C_{dj}$, $x_{j\,min}^{*0} < x_{j\,min}^{*1}$ and $x_{j\,max}^{*1} < x_{j\,max}^{*0}$. The range of values for $x_j^*$,
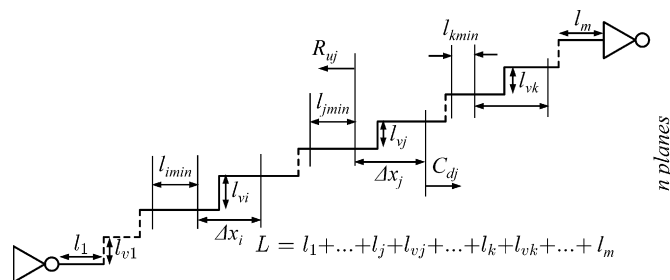


Fig. A1. Interplane interconnect consisting of $m$ segments connecting two circuits located $n$ planes apart.

therefore, also decreases and, typically, after two or three iterations, the optimum location for the corresponding via is determined.

The above example is extended to each of the other possible sub-cases that can occur for segments $i$ and $k$. Specifically,

(a) $i$ and $k$ belong to either case (i) or (ii). Both $R_{uj}$ and $C_{dj}$ are precisely determined or, equivalently, $R_{uj\,min} = R_{uj\,max}$ and $C_{dj\,min} = C_{dj\,max}$. Consequently, the placement of both vias $v_i$ and $v_k$ is known and $x_{j\,min}^{*0} = x_{j\,max}^{*0} = x_j^*$. The placement of $v_j$ is also determined within the first iteration.

(b) $i$ belongs to case (i) or (ii) and $k$ belongs to case (iii). $R_{uj}$ is precisely determined or, equivalently, $R_{uj\,min} = R_{uj\,max}$ and the placement of via $v_i$ is known. Since $v_i$ is placed and $k$ belongs to case (iii), $C_{dj\,min}^0 < C_{dj\,min}^1$ and $C_{dj\,max}^1 < C_{dj\,max}^0$. The placement of via $v_j$ converges faster, as only the placement of segment $k$ remains unknown after the first iteration.

(c) $k$ belongs to case (i) or (ii) and $i$ belongs to case (iii). $C_{dj}$ is precisely determined or, equivalently, $C_{dj\,min} = C_{dj\,max}$ and the placement of via $v_k$ is known. Since $v_k$ is placed and $i$ belongs to case (iii), $R_{uj\,min}^0 < R_{uj\,min}^1$ and $R_{uj\,max}^1 < R_{uj\,max}^0$. The placement of via $v_j$ converges faster as only the placement of segment $i$ remains unknown after the first iteration.

(d) $i$ belongs to any of the cases (i)–(iii) and $k$ belongs to case (iv). $R_{uj}$ is readily determined (cases (i) and (ii)) or converges, as described in the previous sub-case, $R_{uj\,min}^0 < R_{uj\,min}^1$ and $R_{uj\,max}^1 < R_{uj\,max}^0$. As $k$ belongs to case (iv), however, $C_{dj}$ does not change as in the cases above. If the decrease in the upstream resistance is sufficient to determine $x_j^*$ according to (A.1), $v_j$ is marked as processed, otherwise $v_j$ is marked as unprocessed and the algorithm continues to the next via. In the latter case, the placement approach is described by case (iv) of the heuristic.

(e) $k$ belongs to any of the cases (i)–(iii) and $i$ belongs to case (iv). $C_{dj}$ is readily determined (cases (i) and (ii)) or converges, as described in sub-case (b), implying $C_{dj\,min}^0 < C_{dj\,min}^1$ and $C_{dj\,max}^1 < C_{dj\,max}^0$. As $i$ belongs to case (iv), however, $R_{uj}$ does not change as in the aforementioned cases. Overall, if the decrease in the downstream capacitance is sufficient to determine $x_j^*$ according to (A.1), $v_j$ is marked as processed, otherwise $v_j$ is marked as unprocessed and the algorithm continues to the next via. In the latter case, the placement approach is described by case (iv) of the heuristic.

(f) Both $i$ and $k$ belong to case (iv). Therefore, both $R_{uj}$ and $C_{dj}$ cannot be bounded. Consequently, $v_j$ is marked as unprocessed and the next via is processed. Alternatively, this sub-case degenerates to case (iv) of the heuristic presented in Section 3.

## Appendix B. Analytic proof of condition 1

A proof for necessary condition 1 is provided as follows.

**Condition 1.** If $r_j > r_{j+1}$, only a *type*-1 move for $v_j$ can reduce the delay of a tree.

**Proof.** Consider Fig. B1 where the interplane via $v_j$ (the solid square) can be placed in any direction $d_e$, $d_s$, and $d_n$ within the interval $l_{de}$, $l_{ds}$, and $l_{dn}$, respectively. For the tree shown in Fig. B1 and removing the terms that are independent of $v_j$, Eq. (10) is

$$T_w = \sum_{v_i \in U_{0j}} \sum_{s_p \in P_{sp U_{ij}}} w_{s_p} R_{uij}\left(c_{v_j} l_{v_j} + C_{d_j}\right) + \sum_{s_p \in P_{sp v_j}} w_{s_p}\left(R_{uj}(c_{v_j} l_{v_j} + C_{d_j}) + r_{v_j} l_{v_j} C_{d_j} + \frac{r_{v_j} c_{v_j} l_{v_j}^2}{2}\right), \tag{B.1}$$
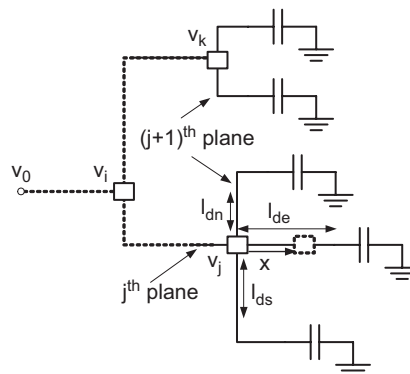


Fig. B1. A portion of an interconnect tree.

where

$$C_{d_j} = \sum_{\forall k} C_{dv_j d_k} + c_{j+1}(l_{d_e} + l_{d_s} + l_{d_n}). \tag{B.2}$$

Suppose that a *type*-2 move is required, shifting $v_j$ by $x$ towards the $d_e$ direction (the dashed square). Expression (10) becomes

$$T'_w = \left( \sum_{v_i \in U_{ij}} \sum_{s_p \in P_{sp} U_{ij}} w_{s_p} R_{uij} + \sum_{s_p \in P_{sp} v_j} w_{s_p} R_{u_j} \right) (c_{v_j} l_{v_j} + c_j x + C_{d_j})$$

$$+ \sum_{s_p \in P_{sp} v_j} w_{s_p}[r_j x (c_{v_j} l_{v_j} + C_{d_j}) + r_{j+1} l_{d_e}(C_{d_j} - (1/2)c_{j+1}l_{d_e}) + (r_j - r_{j+1})xC_{d_j} + (1/2)(r_{v_j} c_{v_j} l_{v_j}^2 + r_j c_j x_j^2)]. \tag{B.3}$$

For a *type*-2 move to reduce the weighted delay of the tree, shifting $v_j$ will decrease $T_w$, or, equivalently, $\Delta T = T'_w - T_w < 0$. Subtracting (B.1) from (B.3) yields

$$\Delta T = \sum_{s_p \in P_{sp} v_j} w_{s_p}\left[ r_j x(c_{v_j} l_{v_j} + C_{d_j}) + R_{u_j} c_j x + r_{j+1} l_{d_e}\left( C_{d_j} - \frac{c_{j+1}l_{d_e}}{2} \right) + (r_j - r_{j+1})xC_{d_j} + \frac{r_j c_j x_j^2}{2} \right]$$

$$+ \sum_{v_i \in U_{ij}} \sum_{s_p \in P_{sp} U_{ij}} w_{s_p} R_{uij} c_j x. \tag{B.4}$$

Since $r_j > r_{j+1}$, (B.4) is always positive and a *type*-2 move cannot reduce the delay of a tree. $\square$

## References

[1] K. Banerjee, S.J. Souri, P. Kapur, K.C. Saraswat, 3-D ICs: a novel chip design for improving deep-submicrometer interconnect performance and systems-on-chip integration, Proc. IEEE 89 (5) (2001) 602–633.

[2] K.M. Brown, System in package: the rebirth of SIP, Proc. IEEE Int. Conf. Custom Integr. Circuits (2004) 681–684.

[3] J. Miettinen, M. Mantysalo, K. Kaija, E.O. Ristolainen, System design issues for 3D System-in-Package (SiP), Proc. IEEE Int. Conf. Electron. Components Technol. (2004) 610–615.

[4] G.T. Goeloe, et al., Vertical single-gate CMOS inverters on laser-processed multilayer structures, Proc. IEEE Electron Device Meeting (1981) 554–556.

[5] G.W. Neudeck, S. Pae, J.P. Denton, T. Su, Multiple layers of silicon-on-insulator for nanostructure devices, J. Vac. Sci. Technol. B 17 (3) (1999) 994–998.

[6] D.N. Kouvatsos, A.T. Voutsas, M.K. Hatalis, Polycrystalline silicon thin film transistors fabricated in various solid phase crystallized films deposited on glass substrates, J. Electron. Mater. 28 (1) (1999) 19–25.

[7] A. Fan, A. Rahman, R. Reif, Copper wafer bonding, Electrochem. Solid-State Lett. 10 (2) (1999) 534–536.

[8] R.J. Gutmann, et al., Three-dimensional (3D) ICs: a technology platform for integrated systems and opportunities for new polymeric adhesives, Proc. Conf. Polym. Adhes. Microelectron. Photon. (2001) 173–180.

[9] L. Xue, C. Liu, S. Tiwari, Multi-layers with buried structures (MLBS): an approach to three-dimensional integration, Proc. IEEE Int. Conf. Silicon Insulator (2001) 117–118.

[10] M. Koyanagi, et al., Future system-on-silicon LSI chips, IEEE Micro. 18 (4) (1998) 17–21.

[11] V. Sutharalingam, et al., Megapixel CMOS image sensor fabricated in three-dimensional integrated circuit technology, Proc. IEEE Int. Conf. Solid-State Circuits (2005) 356–357.

[12] T. Kunio, K. Oyama, Y. Hayashi, M. Morimoto, Three dimensional ICs having four stacked active device layers, Proc. IEEE Electron. Device Meeting (1989) 837–840.

[13] C.C. Liu, et al., Heating effects of clock drivers in bulk, SOI, and 3-D CMOS, IEEE Electron. Device Lett. 23 (12) (2002) 716–718.

[14] C.C. Tong, C.-L. Wu, Routing in a three-dimensional chip, IEEE Trans. Comput. 44 (1) (1995) 106–117.

[15] B. Goplen, S.S. Sapatnekar, Placement of thermal vias in 3-D ICs using various thermal objectives, IEEE Trans. Computer-Aided Design Integr. Circuits Syst. 25 (4) (2006) 692–709.

[16] A. Cohoon, et al., Physical layout for three-dimensional FPGAs, in: Proceedings of the ACM/SIGDA Physical Design Workshop, April 1996, pp. 142–149.

[17] S. Das, et al., Calibration of Rent's rule for three-dimensional integrated circuits, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 12 (4) (2004) 359–366.

[18] W.R. Davis, et al., Demystifying 3D ICs: the pros and cons of going vertical, IEEE Design Test Comput. Mag. 22 (6) (2005) 498–510.

[19] I. Kaya, M. Olbrich, E. Barke, 3-D placement considering vertical interconnects, in: Proceedings of the IEEE International SOC Conference, September 2003, pp. 257–258.

[20] T. Tanprasert, An analytical 3-D placement that preserves routing space, in: Proceedings of the IEEE International Symposium on Circuits and Systems, vol. III, May 2000, pp. 69–72.

[21] S.T. Obenaus, T.H. Szymanski, Gravity: fast placement for 3-D VLSI, ACM Trans. Design Automat. Electron. Syst. 8 (3) (2003) 298–315.

[22] S. Das, A. Chandrakasan, R. Reif, Design tools for 3-D integrated circuits, in: Proceedings of the IEEE Asia and South Pacific Design Automation Conference, January 2003, pp. 53–56.

[23] Y. Deng, W.P. Maly, Interconnect characteristics of 2.5-D system integration scheme, in: Proceedings of the ACM International Symposium on Physical Design, April 2001, pp. 171–175.

[24] R. Zhang, et al., Stochastic modeling, power trends, and performance characterization of 3-D circuits, IEEE Trans. Electron. Devices 48 (4) (2001) 638–652.

[25] V.F. Pavlidis, E.G. Friedman, Interconnect delay minimization through interlayer via placement, in: Proceedings of the ACM Great Lakes Symposium on VLSI, April 2005, pp. 20–25.

[26] Massachusetts Institute of Technology Lincoln Laboratory, FDSOI Design Guide, September 2006.

[27] K.D. Boese, et al., Fidelity and near-optimality of Elmore-based routing constructions, in: Proceedings of the IEEE International Conference on Computer Design, October 1993, pp. 81–84.

[28] A.I. Abou-Seido, B. Nowak, C. Chu, Fitted Elmore delay: a simple and accurate interconnect delay model, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 12 (7) (2004) 691–696.

[29] J.D. Cho, et al., Crosstalk-minimum layer assignment, in: Proceedings of the IEEE Conference on Custom Integrated Circuits, May 1993, pp. 29.7.1–29.7.4.

[30] C. Ryu, et al., High frequency electrical circuit model of chip-to-chip vertical via interconnection for 3-D chip stacking package, in: Proceedings of the IEEE Topical Meeting on Electrical Performance of Electronic Packaging, October 2005, pp. 151–154.

[31] J.G. Ecker, Geometric programming: methods, computations and applications, SIAM Rev. 22 (3) (1980) 338–362.

[32] S. Boyd, S.J. Kim, L. Vandenberghe, A. Hassibi, A tutorial on geometric programming, Optimization Eng. 8 (1) (2007) 67–127.

[33] Metal User's Guide, ⟨www.oea.com⟩.

[34] Predictive Technology Model [online]. Available from: ⟨http://www.eas.asu.edu/~ptm⟩.

[35] W. Zhao, Y. Cao, New generation of predictive technology model for sub-45 nm design exploration, in: Proceedings of the IEEE International Symposium on Quality Electronic Design, March 2006, pp. 585–590.

[36] J. Löfberg, YALMIP: a toolbox for modeling and optimization in MATLAB, in: Proceedings of the IEEE International Symposium on Computer-Aided Control Systems Design, September 2004, pp. 284–289.

[37] D. Henrion, J.B. Lasserre, GloptiPoly: global optimization over polynomials with Matlab and SeDuMi, ACM Trans. Math. Software 29 (2) (2003) 165–194.

[38] K.M. Lepak, I. Luwandi, L. He, Simultaneous shield insertion and net ordering under explicit RLC noise constraint, in: Proceedings of the IEEE/ACM International Conference on Design Automation, June 2001, pp 199–202.

[39] J.W. Joyner, et al., Impact of three-dimensional architectures on interconnects in gigascale integration, IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 9 (6) (2000) 922–927.

**Vasilis F. Pavlidis** received his B.S. and M.Eng. in electrical and computer engineering from the Democritus University of Thrace, Xanthi, Greece, in 2000 and 2002, respectively. He is working toward his Ph.D. degree in the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY.

From 2000 to 2002, he was with INTRACOM S.A., Athens, Greece. In the summer of 2007, he was with Synopsys Inc., Mountain View, California. His current research interests are in the area of interconnect modeling, 3-D integration, networks-on-chip, and related design issues in VLSI.

**Eby G. Friedman** received his B.S. degree from Lafayette College in 1979, and his M.S. and Ph.D. degrees from the University of California, Irvine, in 1981 and 1989, respectively, all in electrical engineering.

From 1979 to 1991, he was with Hughes Aircraft Company, rising to the position of manager of the Signal Processing Design and Test Department, responsible for the design and test of high performance digital and analog ICs. He has been with the Department of Electrical and Computer Engineering at the University of Rochester since 1991, where he is a Distinguished Professor, the Director of the High Performance VLSI/IC Design and Analysis Laboratory, and the Director of the Center for Electronic Imaging Systems. He is also a Visiting Professor at the Technion—Israel Institute of Technology. His current research and teaching interests are in high performance synchronous digital and mixed-signal microelectronic design and analysis with application to high speed portable processors and low power wireless communications.

He is the author of more than 300 papers and book chapters, several patents, and the author or editor of nine books in the fields of high speed and low power CMOS design techniques, high speed interconnect, and the theory and application of synchronous clock and power distribution networks. Dr. Friedman is the Regional Editor of the *Journal of Circuits, Systems and Computers*, a Member of the editorial boards of the *Analog Integrated Circuits and Signal Processing*, *Microelectronics Journal*, *Journal of Low Power Electronics*, and *Journal of VLSI Signal Processing*, Chair of the *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* steering committee, and a Member of the technical program committee of a number of conferences. He previously was the Editor-in-Chief of the *IEEE Transactions on Very Large Scale Integration (VLSI) Systems,* a Member of the editorial board of the *Proceedings of the IEEE* and *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, a Member of the Circuits and Systems (CAS) Society Board of Governors, Program and Technical chair of several IEEE conferences, Guest Editor of several special issues in a variety of journals, and a recipient of the Howard Hughes Masters and Doctoral Fellowships, the University of Rochester Graduate Teaching Award, and a College of Engineering Teaching Excellence Award. Dr. Friedman is a Senior Fulbright Fellow and an IEEE Fellow.