Hierarchical 2-D Field Solution for Capacitance Extraction for VLSI Interconnect Modeling†

E. Aykut Dengi
Motorola Inc.
3501 Ed Bluestein Blvd., Austin, TX 78721

Ronald A. Rohrer
Dept. of ECE, Carnegie Mellon University
5000 Forbes Ave., Pittsburgh, PA 15213

Abstract

A hierarchical two-dimensional field solution technique is introduced for capacitance extraction for VLSI interconnect modeling. As a basis for compromise between the efficiency of Boolean rules-based extraction and the accuracy of flat field solution, this hierarchical approach can handle realistic conductor cross-sections and multiple conformal and/or planarized dielectrics.

1. Introduction

As integrated circuit processing technology marches relentlessly down through deep submicron feature sizes, chip performance limitations, such as system delay and signal integrity, are coming to be determined more by interconnect effects than by active device characteristics [1]. To address the issue in its entirety and obtain the overall chip interconnect coupling capacitance matrix would require prohibitively expensive three-dimensional field solution. Because that is impractical for a VLSI circuit design, significant compromises are effected in commercially available capacitance extraction tools: typically field solution is employed *a priori* to *tune* rules-based extractors. Such rules-based extractors utilize process parameter and field solution derived *models* together with Boolean operations on two-dimensional mask sets to determine capacitance values derived from features such as overlap, lateral proximity, and so on.

Even accurate field solutions can incur severe performance penalties when realistic irregular conductor cross-sections and corresponding conformal dielectrics are introduced (Fig. 1). These process features have an important effect on parasitic values [2]. Rules-based extraction may partially *correct* for these effects during their full field solution pre-processing *tuning* phase. But rules-based extractors ultimately incur large errors due to their necessity to partition any problem domain into very small pieces and then to reassemble the results applying the rules to those several small pieces.

Figure 1. A realistic vertical cross-section of IC interconnect. We see that conductors on layers 1-5 are trapezoidal. The top layer metal has a completely irregular geometry. There are voids between minimum-spaced conductors. There is a conformal layer of dielectric on top of the top layer metal(passivation). SEM picture courtesy of IBM Corp. © Copyright IBM Corp. 1994, 1996.

As an attempt to overcome these limitations this paper proposes a simple hierarchical partitioning approach to field solution [7,8]. This scheme still entails detailed *a priori* field solution, but on a smaller set of parameterized library macromodeling elements. Such a physical macromodel is represented electrically in terms of a capacitance matrix that relates the potential(s) and flux(es) at the artificial boundary of the element to the total charge on the conductor surface. The library element capacitance matrix macromodels can then be combined at runtime to produce accurate field solutions of entire interconnect cross-sections. While less efficient than rules-based Boolean extraction, this approach is much more efficient than flat field solution while maintaining or exceeding its accuracy.

Another recently published approach to hierarchical field solution [13] shares with this work roots in domain decomposition (see, for instance, [12]). Both approaches solve for the field in the non-overlapping subdomains and merge the solutions using compatibility conditions at the interfaces. Whereas [13] has focused more on spectral techniques and

†. This work was supported by the National Science Foundation under Grant MIP-9216942, and by the Semiconductor Research Corporation under Contract DC-068.

DAC 97, Anaheim, California

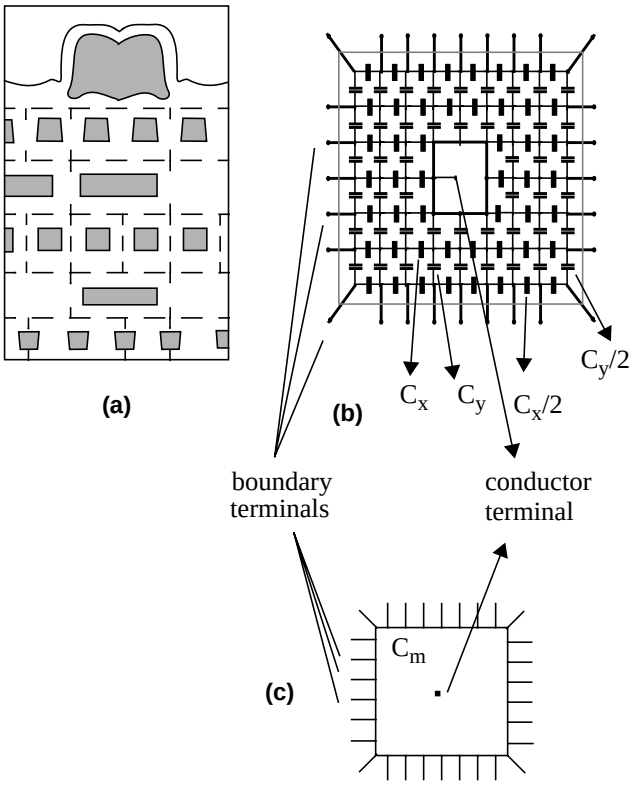


Figure 2. Partitioning and macromodeling: **(a)** The cross section in Fig. 1 is partitioned; **(b)** Each partition can be modeled as a capacitive mesh; **(c)** The internal nodes are eliminated to form a macromodel represented by a capacitance matrix.

limited itself mainly to regular geometries in 3-D, this work has concentrated on the Finite Difference and Boundary Element Method formulations in solving complex geometries typical of IC vertical cross-sections.

The 2-D ideas we present here are useful as a basis for developing a hierarchical 3-D field solution technique, and also useful on their own in the so-called "2.5-D" extraction context. In the confines of this paper, we have been able only to cover the most basic elements of hierarchical 2-D solution for capacitance extraction. Refinements and extensions, including 3-D implications, are covered in detail in [7] and will be elaborated in subsequent papers.

All the feasibility studies shown in this paper are conveniently implemented in *MATLAB*™ [14].

The rest of this paper is organized as follows. In Section 2 we give an overview of the hierarchical 2-D capacitance extraction method and its Finite Difference (FD) implementation. In Section 3, we discuss how we can build a library of elements as a pre-processing step. Finally, we offer some conclusions.

2. Hierarchical 2-D Capacitance Extraction

The 2-D capacitance extraction methodology to be described here has the following features:

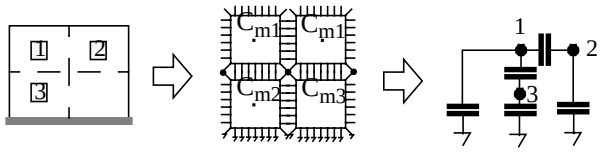


Figure 3. Hierarchical field solution combines the individual macromodels eliminating nodes at their common edges.

1. It uses hierarchical partitioning and library macromodel preprocessing to attain chip extraction runtime efficiency.
2. It can provide total and coupling capacitances related to a single net or the full capacitance matrix associated with a group of nets.
3. It can deal accurately with arbitrary conductor vertical cross-sections and multiple conformal and/or planarized dielectrics.

Although the technique is best implemented in terms of the Boundary Element Method (BEM), it is most easily explained in terms of a finite difference formulation, which we will do here. The BEM formulation can be found in [7] and will be published elsewhere.

The essence of the hierarchical field solution technique can be captured by examining Figures 1, 2 and 3. The vertical cross-section to be analyzed (Fig. 1) is partitioned into distinct regions (Fig. 2 (a)). A library element (Fig. 2 (c)) is formed by solving for a macromodel capacitance matrix for a given partition (Fig. 2 (b)). The capacitance matrix for a vertical cross-section is obtained by combining the appropriate macromodels (Fig. 3).

The nature of the problem with which we are dealing renders the use of macromodels especially advantageous. We can take advantage of the fact that the problem space, defined by all possible geometries found on the interconnect layers of a chip, is very limited. Although the geometry under analysis is complicated by conductors with nontrivial cross-sections and nonhomogeneous dielectric layers (Fig. 1), there is much regularity to be exploited. The space of possible 3-D geometries formed by a manufactured IC is limited by the technology description and design rules. The technology description is determined by the manufacturing process. The design rules represent restrictions due to yield concerns, the manufacturing process or design tools. Therefore we do not need the generality of a generic field solver, and we can avoid some of the inefficiencies that generality imply. We can describe any vertical cross-section with relatively few library elements using our method, as we shall see in Section 3.

Finite Difference Macromodels

For capacitance computation purposes, we can represent the finite difference discretization of an arbitrary domain (of dielectric materials and conductors) as a mesh of capacitors [9]. If we cut an arbitrary patch that includes the conductor from the capacitive grid, we get the sub-mesh depicted in

Fig. 2. In order to get the same result when we put it back, we assign half as much capacitance to the capacitors along the boundary edges. The capacitor terminals that lie along the surface of the internal conductor are all at the same potential and are "shorted out". We can now eliminate the inner nodes from the capacitance formulation of the circuit $Cv = Q$ via a partial LU decomposition, creating an N-port Norton equivalent for this capacitive network. Here, subscript *1* refers to inner nodes, subscript *2* refers to outer nodes and nodes on the conductor; *q* is flux and *v* is potential.

$$\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}, \tag{1}$$

$$q_1 = 0 \text{ (KCL)}. \tag{2}$$

Eliminating internal nodes:

$$(C_{22} - C_{21}(C_{11})^{-1}C_{12})v_2 = q_2,$$
$$C_m = C_{22} - C_{21}C_{11}{}^{-1}C_{12}. \tag{3}$$

Note that the macromodel capacitance matrix C_m is singular because we did not pick a datum node. We did this so that C_m is in the form of a stencil [5] into a global matrix. Note also that an irregular conductor geometry is still represented by only one terminal, although, in a flat field solution, the conductor would have to be represented by many grid points. Grid points on dielectric interfaces inside elements are similarly eliminated. This reduces the number of variables to be considered at runtime. Throughout this paper, the term runtime will be used to indicate the assembly and solution of the global capacitance matrix (see below), as opposed to the "preprocessing" time that must be employed for macromodel library creation.

Global Solution

We will now show how we can obtain the capacitance matrix for a given geometry using macromodel parameters. We first decompose each layer into geometries for which we have macromodels in our library. The macromodels we choose may include no, one or more conductors. We then stencil the macromodel matrices (C_ms) into a global nodal analysis matrix [5], which we shall call the global capacitance matrix. For inter-element boundaries, each node is shared by two macromodels, and the corresponding matrix entries are added. The total flux for inter-element boundaries is set to zero on the RHS of the equation. For the outside boundaries, we need to impose a boundary condition. Setting the flux equal to zero imposes a Neumann boundary condition. Eliminating the columns and rows corresponding to a node is equivalent to imposing a Dirichlet boundary condition (V=0). Or we may use a terminating macromodel to emulate more realistic boundary conditions (see [7]). In this case, there are no outside boundaries, and these nodes are treated as inter-element nodes between the macromodel at the boundary and the terminating macromodel which has the same format as a macromodel. The boundary conditions imposed may have a large impact on the results.

Once the global nodal analysis matrix, C_g is formed, we may re-arrange it as:

$$C_g v = \begin{bmatrix} C_{\text{elim,elim}} & C_{\text{elim,cond}} \\ C_{\text{cond,elim}} & C_{\text{cond,cond}} \end{bmatrix} \begin{bmatrix} v_{\text{elim}} \\ v_{\text{cond}} \end{bmatrix} = \begin{bmatrix} 0 \\ q_{\text{cond}} \end{bmatrix} \tag{4}$$

where the subscript *cond* denotes conductor nodes (one for each conductor), and *elim* denotes all other nodes (at boundaries) to be eliminated (Fig. 4). A partial LU decomposition is performed to obtain:

$$Cv_{\text{cond}} = q_{cond}$$
$$C = C_{\text{cond,cond}} - C_{\text{cond,elim}}C^{-1}_{\text{elim,elim}}C_{\text{elim,cond}} \tag{5}$$

which is the capacitance matrix we have set out to compute. We can also create new larger macromodels by combining previously created ones and eliminating their common boundaries. In this case:

$$\begin{bmatrix} C_{\text{elim,elim}} & C_{\text{elim,bound}} & C_{\text{elim,cond}} \\ C_{\text{bound,elim}} & C_{\text{bound,bound}} & C_{\text{bound,cond}} \\ C_{\text{cond,elim}} & C_{\text{cond,bound}} & C_{\text{cond,cond}} \end{bmatrix} \begin{bmatrix} v_{\text{elim}} \\ v_{\text{bound}} \\ v_{\text{cond}} \end{bmatrix} = \begin{bmatrix} 0 \\ q_{\text{bound}} \\ q_{\text{cond}} \end{bmatrix} \tag{6}$$

and as in Eq. (4), a partial LU decomposition is performed to eliminate this time only the inter-element boundary nodes, denoted by the subscript *elim* (Fig. 4). The new macromodel may then be stored in the element library for future use. Termination macromodels are created in a similar manner. For termination macromodels, certain boundary conditions are assumed for some of the outside boundaries, and only a subset of outside boundaries is retained.

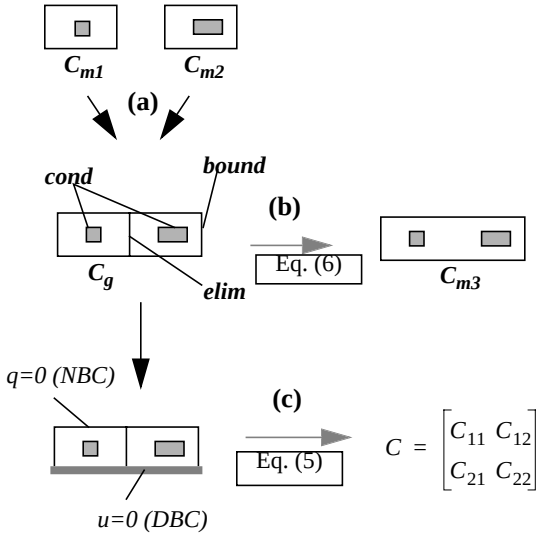


Figure 4. Once a global matrix is formed **(a)** with stencils from macromodels, a macromodel can be obtained for the overall configuration **(b)**, or boundary conditions can be imposed and a capacitance matrix can be obtained **(c)**.

We may not need the full capacitance matrix all the time.

When we want to compute the coupling capacitance to conductor k only, we apply 1 Volt to conductor k, and 0 Volts to all others:

$$v_{\text{cond}} = \begin{bmatrix} 0 & \dots & \underset{\uparrow}{1} & \dots & 0 \\ & & k & & \end{bmatrix}^T \tag{7}$$

We then solve for q_{cond} in Eq. (4):

$$\begin{aligned} v_{\text{elim}} &= -C_{\text{elim,elim}}^{-1} C_{\text{elim,cond}} v_{\text{cond}}, \\ q_{\text{cond}} &= C_{\text{cond,elim}} v_{\text{elim}} + C_{\text{cond,cond}} v_{\text{cond}}. \end{aligned} \tag{8}$$

Note that we do not actually invert $C_{\text{elim,elim}}$ at any point. One way to compute v_{elim} is to perform an LU factorization on $C_{\text{elim,elim}}$ and solve for v_{elim} by forward and backward substitution of the column of $C_{\text{elim,cond}}$ that v_{cond} picked. One of the major advantages of our method is that it yields a block sparse matrix like the Finite Difference method, yet it has few variables, like the Boundary Element Method (Fig. 5). Special numerical methods have been developed to perform the LU factorization of such block sparse matrices efficiently [3]. Alternatively, we could use an iterative method, such as GMRES [4] to solve for v_{elim}, which is feasible owing to the sparse nature of the global matrix.

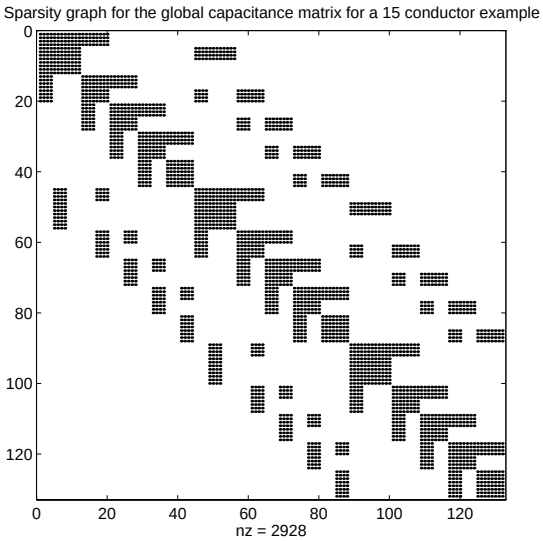


Figure 5. Sparsity graph for a typical global capacitance matrix C_g. This matrix was formed in the solution of a 15 conductor problem.

The number of variables in the above matrix is determined by the number of nodes that are eliminated, i.e., interface nodes and boundary nodes. We can decompose this into horizontal nodes and vertical nodes. Nodes that are grounded never enter the calculation; therefore the number of nodes is (see Figure 6):

$$n_{nodes} = \left(\frac{w_{\text{tot}}}{\Delta_{\text{hor}}} + 1\right) n_{\text{layer}} + (n_{\text{elem}} + n_{\text{layer}}) n_{\text{vert}}. \tag{9}$$

The block-sparse matrix solution is typically $O(n_{\text{nodes}}^x)$, where x is between 1.2 and 1.5. Note that n_{nodes} is not a function of process details such as the irregularity of the conductors and the number of dielectric layers.

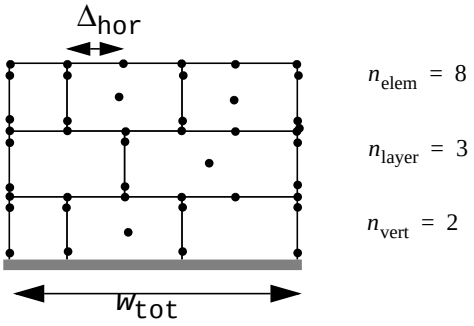


Figure 6. Computing complexity. The variables used in Eq. (9) are indicated for this example.

Convergence

We present a simple example to show the convergence properties of our method. We examine the vertical cross-section in Fig. 7. We shall assume Neumann boundary conditions at the side and top boundaries.

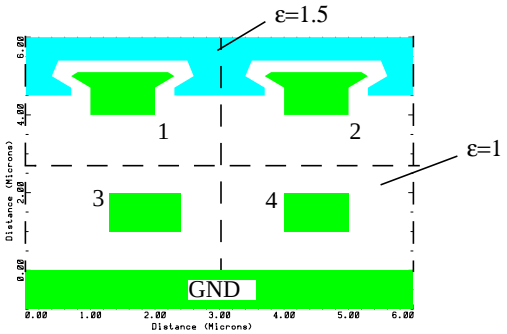


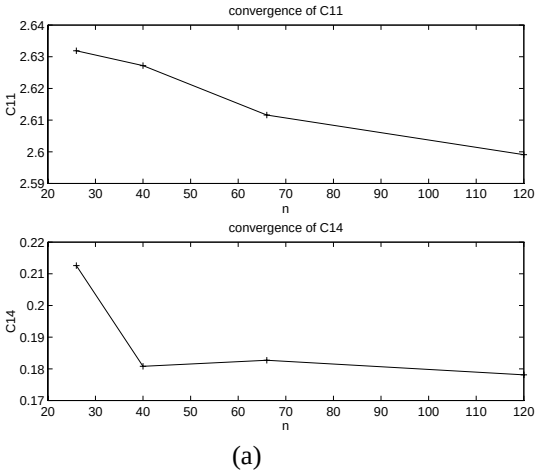
Figure 7. The vertical cross-section for an imaginary process. The dielectric constant is taken as 1. The top metal layer has an irregular cross-section and there is a conformal dielectric with a dielectric constant 1.5. Dashed lines indicate the element boundaries used.

We have simulated this configuration with *Raphael*^{TM} [6], a general purpose Finite Difference Field solver, using 10,000 grid points and the result is taken as "exact". The results obtained using our technique are compared to the *Raphael* results in Fig. 8. The capacitance matrix for $n_{\text{nodes}} = 120$ is tabulated in Table 1. But we can see that even for the lowest level of discretization, the error is within 1.5%.

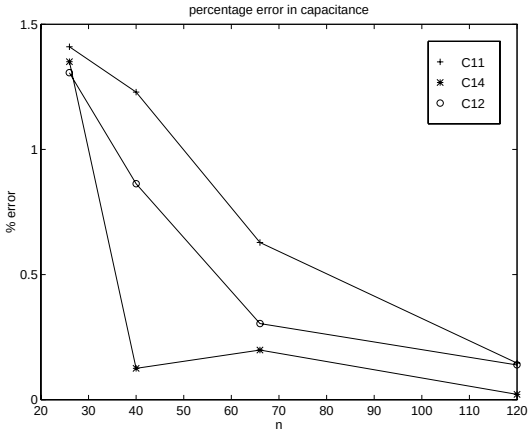
For this example, ignoring the top-layer dielectric and the irregularity in the top-layer conductor cross-section would give rise to 15% error in the total capacitance of conductor 1. In our method, these effects are accounted for at the pre-processing stage and incur no additional cost at runtime.

3. Library Building

We shall assume that conductors at each metallization layer have a constant height from the substrate "to the first order."



(a)



(b)

Figure 8. Convergence with improving discretization. *n* is the number of variables in the global capacitance matrix formed. (a) shows the convergence in absolute terms of *C11*, the total capacitance of conductor 1, and *C14*, the coupling capacitance between conductors 1 and 4. (b) shows the percentage error in some capacitance values. The percentage error is computed as:

$$100((C_x^{\text{approx}} - C_x^{\text{exact}})/C_{11}^{\text{exact}})$$

Table 1. Capacitance values computed for the configuration of Fig. 7

C_{ij}	1	2	3	4
1	2.5991	1.3828	0.9661	0.1781
2		2.6251	0.2571	0.9407
3			4.5312	0.8135
4				4.2258

By "constant to the first order", we mean that the deviation of the height from nominal due to process variations and topography is a small portion of the nominal value. This is true for processes that use advanced planarization techniques such as chemical mechanical polishing (CMP) [11].

In addition to this basic assumption, we make an assumption which is not essential to the operation of the method, but increased efficiency is obtained when it holds: We will assume that conductor vertical cross-sections (together with conformal dielectrics) can be described in a scalable fashion, i.e., the shape of a conductor with arbitrary width may be obtained by slicing a minimum width conductor and extending the middle (Fig. 9). With these assumptions, we construct a library of macromodels that can be combined to describe any vertical cross-section allowed for a given technology.

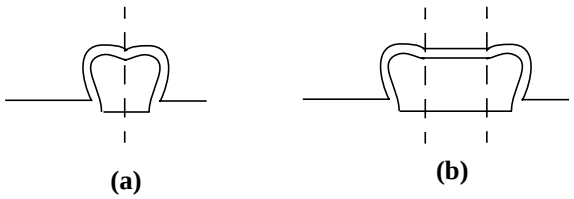


Figure 9. Scalable vertical cross-section with a conformal dielectric. **(a)** is the minimum-width metal. Left and right halves are identical due to symmetry. The irregularity of the conductor is due to the deposition and etching steps that affect mostly the sides. In **(b)**, the medium section is derived by connecting the end-points of each side.

Assembling the macromodels, it appears that the physical locations of the terminals must match at the interfaces in order to use Eq. (5) directly. In this limited sense, we must have 'universally matching' elements (see Fig. 10) in the library, so that the representation of an element is independent of the elements around it. This means that we must use the same discretization for vertical sides of elements that can be adjacent. Some elements cannot be adjacent by rule because that would correspond to an impossible configuration. For the horizontal sides, on the other hand, all segments must be the same size. Fortunately, our method is not limited to 'universally matching' elements, since we can use interpolation and extrapolation to bring together elements with different discretizations in a single global solution using extensions to Eq. (5) [7].

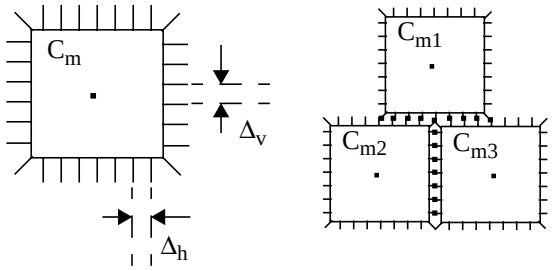


Figure 10. Universally matching elements.

The elements need not have the conductor in the middle. Fig. 11 shows some types of elements we can have. Type II elements allow a smaller library size.

There are trade-offs between runtime extraction efficien-

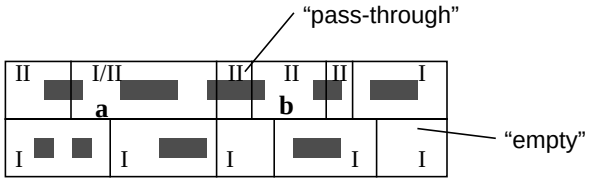


Figure 11. Types of elements and how they combine. Elements with no conductors on the boundary, including "empty" elements are of Type I. Elements with no conductors in the middle, but with one or two conductors on the boundary, including "pass-through" elements are of Type II. Elements with conductors on the boundary as well as the middle are of Type I/II. **(a)** and **(b)** are complex elements that can be used for runtime efficiency and parameterization.

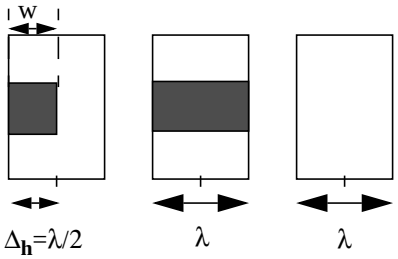


Figure 12. The minimum set of three basic elements that span a layer of a given technology.

cy, accuracy and library size. We shall briefly mention some of these trade-offs. A detailed study can be found in [7].

The discretization of the elements is a major determinant of the accuracy of the global solution. Elements for the same geometry with different levels of discretization can be stored in the library to allow different points on the accuracy-efficiency curve. The largest horizontal discretization (hence, the highest efficiency) is determined by the feature size:

$$\Delta_{h,max} = \lambda_{process} \quad (10)$$

This choice of horizontal discretization makes sure that we can "span" the technology with the minimum set of elements (Fig. 12) per layer. By spanning a technology, we mean having a "sufficient variety of library elements to describe any vertical cross-section that is allowed in a certain process technology. We cannot span a technology only with the elements we have described so far, because they have fixed dimensions. To maintain a compact library in the face of a fine design grid requires parameterized elements [7].

On the other hand, we can create macromodels for more complex elements (e.g., elements a and b in Fig. 11) in order to increase the accuracy we obtain for a given runtime cost. More complex elements increase the library size, but also increase the efficiency at runtime by reducing the number of variables to be eliminated from the global solution matrix. These elements may be constructed from more those primitive ones as shown in Eq. (6), or from scratch using Eq. (3).

4. Conclusion

We have introduced a theoretically founded hierarchical two-dimensional field solution technique. The technique uses preprocessing for runtime efficiency and can deal accurately with realistic VLSI interconnect cross-section configurations. We are presently working on the extension of this technique to 3-D.

References

[1] *The National Technology Roadmap for Semiconductors*, Semiconductor Industry Association, 4300 Stevens Creek Blvd, Suite 271, San Jose, CA 95129, 1994

[2] Wright, P.J.; Shih, Y.-C.A., "Capacitance of top leads metal - comparison between formula, simulation, and experiment," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. (USA)*; vol.12, no.12; Dec. 1993; pp. 1897-902

[3] Kane, J.H.; Kumar, B.L.K.; Saigal, S. "An Arbitrary Condensing, Noncondensing Solution Strategy for Large Scale, Multizone Boundary Element Analysis," *Computer Meth. Appl. Mech. Eng.* 79(1990) pp. 219-244

[4] Saad, Y.; Schultz, M.H., "A generalized minimum residual algorithm for solving non-symmetric linear systems," *SIAM J. Stat. Comp.*, 7:856-869, 1986

[5] Gustavson, F.; Liniger, W.; Willoughby, R. "Symbolic generation of an optimal Crout algorithm for sparse systems of linear equations," IBM Corporation. Yorktown Heights, N.Y., Res. Paper RC 1852, June 1967

[6] *Raphael Interconnect Analysis User's Manual*, Technology Modeling Associates, Inc. Third Floor, 300 Hamilton Ave., Palo Alto, CA 94301, Aug. 1993

[7] Dengi, E.A.; Rohrer, R.A., "Hierarchical 2-D Field Solution for Capacitance Extraction for VLSI Interconnect Modeling," SRC Technical Report #CMUCAD-96-32, July 1996

[8] Dengi, E. A., "A Parasitic Capacitance Extraction Method for VLSI Circuits," PhD Dissertation, Carnegie Mellon University, March 1997.

[9] Kumashiro, S.; Rohrer, R.A.; Strojwas, A.J., "Asymptotic waveform evaluation for transient analysis of 3-D interconnect structures," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. (USA)*, vol.12, no.7; July 1993 pp.; pp. 988-96 pp.

[10]Brebbia, C.A., Telles, J.C.F., Wrobel, L.C. *Boundary Element Techniques*, Springer-Verlag, Berlin, Heidelberg, 1984

[11]Anand, M.B., *et al.*, "Fully Integrated Back End of the Line Interconnect Process," *1994 VLSI Multilevel Interconnect Conference*, June 7-8, 1994, IEEE 1994 pp. 15-21

[12]Smith, B.; "Domain Decomposition Methods for Partial Differential Equations," the Proceedings of ICASE/LaRC Workshop on Parallel Numerical Algorithms, editors D. E. Keyes, A. Sameh, and V.Venkatakrishnan, Kluwer Publishers, 1995.

[13]Hong, W.; Sun, W.; Zhu, Z.; Ji, H.; Song, B.; Dai, W.W.; "A Novel Dimension Reduction Technique for the Capacitance Extraction of 3D VLSI Interconnects," *Proceedings of the IEEE/ACM ICCAD*, Nov 10-14, 1996, pp.381-386

[14]*MATLAB User's Guide*, The Mathworks, Inc. Cochituate Place, 24 Prime Park Way, Natick, MA 01760, Aug. 1992