A Statistical Method for Fast and Accurate Capacitance Extraction in the Presence of Floating Dummy Fills

Shabbir Batterywala, Rohit Ananthakrishna, Yansheng Luo† and Alex Gyure†
ATG, Synopsys (India) Pvt. Ltd., Bangalore, India
Synopsys Inc., Mountain View, CA 94085, USA†
{battery, yansheng, alexg}@synopsys.com, arohit@gmail.com

Abstract— **Dummy fills are being extensively used to enhance CMP planarity. However presence of these fills can have a significant impact on the values of interconnect capacitances. Accurate capacitance extraction accounting for these dummies is CPU intensive and cumbersome. For one, there are typically hundreds to thousands of dummy fills in a small layout region, which stress the general purpose capacitance extractor. Second, since these dummy fills are not introduced by the designers, it is of no interest for them to see the capacitances to dummy fills in the extraction reports; they are interested in *equivalent* capacitances associated with signal power and ground nets. Hence extracting equivalent capacitances across nets of interest in the presence of large number of dummy fills is an important and challenging problem. We present a novel extension to the widely popular Monte-Carlo capacitance extraction technique. Our extension handles the dummy fills efficiently. We demonstrate the accuracy and scalability of our approach by two methods (i) classical and golden technique of finding equivalent interconnect capacitances by eliminating dummy fills through the network reduction method and (ii) comparing extracted capacitances with measurement data from a test chip.**

I. Introduction

Modern fabrication technologies can have up to 10 or 12 routing layers with non-uniform metal density. This results in severe planarity issues which can cause manufacturing problems and timing unpredictability. With the advent of Chemical-Mechanical Polishing (CMP) [1], it has been possible to achieve local as well as global planarity of the wafer surface. However the CMP process can result in dishing[2] of a layer that is being planarized if there is an uneven distribution (empty spaces) of interconnects, thereby diminishing the effectiveness of CMP technology. In order to minimize this dishing phenomenon, dummy fill objects[3] are used. Techniques have been suggested towards placement of these fill objects between signal nets in [4] and [5]. In literature these fill objects are referred to with names like; *fill metals*, *fill nets* or *dummy fills*. Typically these dummy fills are small objects (1um X 1um or so) inserted automatically by layout optimization tools in sparse regions between the signal nets. While achieving the desired CMP planarity these dummies have an adverse impact on coupling capacitances. Their inclusion increases coupling capacitances by large amounts. This necessitates that parasitic extractors should account for these dummies during capacitance extraction. Before going into details of extraction we us first define a few terminologies.

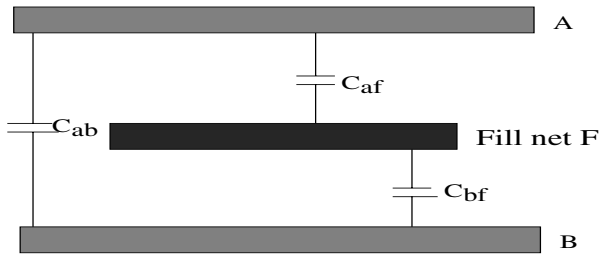

Fig. 1. *Capacitances between 2 interconnects and a fill metal between them*

Consider a simple example consisting of two interconnects and a floating fill net between them as shown in Figure 1. c_{ab}, c_{af} and c_{bf} are the *direct capacitances* between (A,B), (A,F) and (B,F) respectively. However since fill F is floating, the crosstalk between A and B is determined not by c_{ab} alone but by $c'_{ab} = c_{ab} + \frac{c_{af}c_{bf}}{c_{af}+c_{bf}}$. The capacitance c'_{ab} between interconnects in the presence of floating fill net(s) is called the *equivalent capacitance*. Since c_{af} and c_{bf} are positive it is easy to see that $c'_{ab} > c_{ab}$.

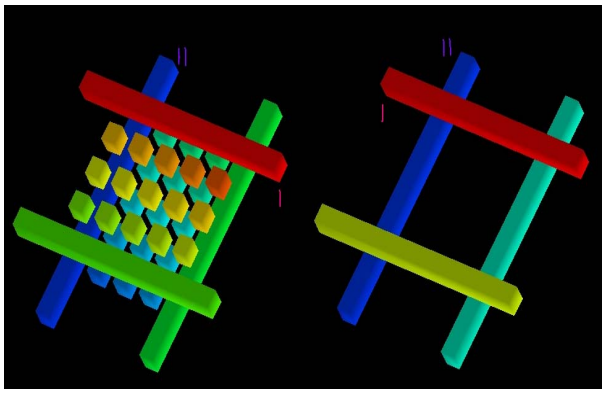

Fig. 2. *2x2 cross bus with and without fill metals. Net I is the top conductor and Net II is the left conductor*

Floating fill nets: As we observed in Figure 1, equivalent capacitance between A & B is more than that of the direct capacitance in the presence of fill net F. Let us look at the impact on equivalent capacitance in the presence of large numbers of floating fills with another example. Figure 2 shows a 2x2 cross bus with and without fill nets. The capacitance between nets I and II without fill nets is $0.465\,fF$ (femto Farads) and with fill nets the equivalent capacitance becomes $0.826\,fF$ which is a 77.6% increase in value. We observed similar behavior in other pairs of capacitances too. He *et. al.* [6] give examples in which the relative change in capacitance can be more than 300%. For a more detailed analysis of capacitance deviation due to floating fill metals, see the work of Lee *et. al.* [7]. To minimize the increase in capacitance because of the floating fills, various fill metal shapes and placement strategies have also been investigated. Towards this, [8] suggests using fill patterns which are plus ('+') shaped and impose restrictions on where the fill nets can be placed. However this still does not eliminate the need to compute accurate equivalent capacitances. Since there could be hundreds to thousands of floating fill nets near a signal net, extracting each net becomes a non-trivial task. Also, as these floating fill nets are not introduced by the designers, they are not interested in the coupling capacitances to these nets; they are interested only in the equivalent capacitances between interconnects. Hence the need to compute equivalent capacitances in the presence of floating fill nets.

Grounded fill nets: Grounded fill nets are not common as they require significant additional routing resources to connect fill structures to nearby ground lines. The presence of grounded fills introduce shielding across neighboring signal wires on either sides, at the expense of increased grounded capacitance for these wires. Since the lumped grounded capacitances increase, signal delay increases for these wires. Since they do not pose the problem of equivalent capacitance extraction, existing capacitance extractor tools can gracefully handle grounded fill nets. The pros and cons of floating and

grounded fills are discussed in [9]. In this paper we consider only floating fill nets and henceforth by fill nets we mean floating fill nets.

Since floating fills increase coupling capacitances across signal lines crosstalk and hence timing is directly affected. It is important to accurately account for the presence of fills during capacitance extraction. The available capacitance extraction techniques (FEM [10], BEM [11], Monte Carlo [12]) do not have specific treatment to dummy fills. [13] gives a scheme to modify the geometry of signal nets and to remove the dummy fills to get approximate capacitance values quickly. However, exact computation of equivalent capacitances in the presence of floating fill nets is not well researched. The most straight forward and classical way would be to find all the coupling capacitances, and then eliminate the fill nets using network reduction techniques, thereby finding the equivalent capacitances. The problem with this approach is that too many coupling capacitances must be computed, as there would be thousands of fill nets. This increases the extraction time by several folds. The subsequent network reduction step is also runtime and memory intensive. Towards this, extensions to BEM method have been reported in [14] and [15] which can directly compute equivalent capacitances. The capacitance obtained by imposing zero-charge on floating conductors contains the equivalent coupling effect. Thus the equivalent capacitance matrix is evaluated without constructing the full bare capacitance matrix. However with this no real runtime advantage is gained, as one still needs to write equations at fill metal surfaces. This makes the overall system matrix too large. [16] uses the so-called fictitious domain method to transform the problem with floating conductors to a one without floating conductors. However the matrix size is still determined by the total number of conductors and floating fills.

We present a novel extension to the technique of Le-Coz and Iverson [12] wherein capacitances are computed by statistical Monte-Carlo integrations though a series of Random Walks (RW). A random walk starts from a conformal surface (called *Gaussian surface*) around a signal net, makes a series of hops in dielectric space, and terminates on another signal net, thereby giving an estimate of coupling capacitance between the two signal nets. The problem arises when a walk hits a floating net and the integral equation to determine the potential at that point can no longer be used. To get around this we use the fact that a fill net is an equi-potential surface and the potential of a fill net can be computed from the potentials of its neighborhood. This results in an additional integral equation along with the equations used in [12]. The addition of this equation modifies the random walk process, and we call this extension the *Modified Random Walk* (MRW) method. With the MRW method, walks can 'snap' across the floating nets, thereby directly computing the equivalent capacitances associated with the signal nets.

In order to establish the accuracy of the MRW method, we use the classical technique of network reduction (NR) to find the true equivalent capacitances. First, we use the standard RW method to determine all the coupling capacitances and then apply network reduction to eliminate nodes corresponding to fill nets. The resulting network contains nodes corresponding to only the signal nets, thereby yielding the equivalent capacitances. Since RW methods report capacitances as a value plus an associated error bound (statistical uncertainty), it becomes imperative to use these error bounds in the network reduction process. We use *real intervals* to represent capacitances, and use *interval arithmetic* in the network reduction process. We use results from statistics to come up with tight bounds for the eventual equivalent capacitances. We use efficient algorithm to do network reduction. With these techniques, we demonstrate the accuracy and scalability of our MRW technique with synthetic and real test cases.

The remainder of this paper is organized as follows. In Section II, we describe the network reduction method and an efficient implementation of it. Section III describes the underlying principle of MRW. Section IV contains experimental results to validate the accuracy and scalability of MRW method. Concluding remarks are given in Section V.

II. The Golden Method - Network Reduction (NR)

In the example used in Section I a simple formula for equivalent capacitance in the presence of a single fill net was used. However in practice, there are hundreds to thousands of fill nets. In this section we first describe underlying theory behind computing equivalent capacitances from direct capacitances in presence of an arbitrary number of fill nets, and then describe an algorithm to efficiently compute it. We also describe how to handle the more general case wherein direct capacitances are intervals instead of real numbers.

Let there be $n+m$ nets in the layout; the first n nets constitute the set of interconnects $\mathcal{I}$ and the remaining m form the set of floating fill nets $\mathcal{F}$. One of the interconnects is the reference *ground* net. The capacitance model for the circuit can be represented as a $n+m$ terminal multi-port, with each net modeled as a terminal or a port. Let i_j be the port current injected through the port j. Let $I_{\mathcal{I}} = [i_1, i_2, \cdots, i_n]^T$ and $I_{\mathcal{F}} = [i_{n+1}, i_{n+2}, \cdots, i_{m+n}]^T$ be the port current vectors. Let v_j be the potential at the port j. Let $\dot{v} = \frac{dv}{dt}$. Then $\dot{V}_{\mathcal{I}} = [\dot{v}_1, \cdots, \dot{v}_n]^T$ and $\dot{V}_{\mathcal{F}} = [\dot{v}_{n+1}, \cdots, \dot{v}_{m+n}]^T$ are derivatives of port voltage vectors. $C_{\mathcal{II}}$ is an $n \times n$ matrix that gives the coupling (mutual) capacitances between the nets in $\mathcal{I}$. The i^{th} row of $C_{\mathcal{II}}$, $C_{\mathcal{II}}[i] = [-c_{i1}, -c_{i2}, \cdots, c_{ii}, \cdots, -c_{in}]$ where $c_{ij} = c_{ji}$ is the coupling capacitance between nets i and j and $c_{ii} = \sum_{p=1, p\neq i}^{n+m} c_{ip}$ is the self (total) capacitance of net i. Similarly we define $C_{\mathcal{IF}}$ and $C_{\mathcal{FF}}$. From the definitions we observe that $C_{\mathcal{II}}$, $C_{\mathcal{FF}}$ and the block capacitance matrix in Equation 1 are all *symmetric*. Port current and voltage derivative vectors are related through C by the matrix equation

$$\begin{bmatrix} I_{\mathcal{I}} \\ I_{\mathcal{F}} \end{bmatrix} = \begin{bmatrix} C_{\mathcal{II}} & C_{\mathcal{IF}} \\ C_{\mathcal{FI}} & C_{\mathcal{FF}} \end{bmatrix} \begin{bmatrix} \dot{V}_{\mathcal{I}} \\ \dot{V}_{\mathcal{F}} \end{bmatrix}. \tag{1}$$

Since no current is injected in any of the floating fills in $\mathcal{F}$, we have $i_{n+1} = \cdots = i_{n+m} = 0$. i.e., $I_{\mathcal{F}} \equiv 0$. Hence if we eliminate $\dot{V}_{\mathcal{F}}$ from row (1) in Equation 1, we get

$$I_{\mathcal{I}} = \left[C_{\mathcal{II}} - C_{\mathcal{IF}} C_{\mathcal{FF}}^{-1} C_{\mathcal{FI}}\right] \dot{V}_{\mathcal{I}}. \tag{2}$$

Equation 2 relates voltage derivative with injected current for interconnect terminals; hence the equivalent capacitance matrix C_{eq} is given by

$$C_{eq} = C_{\mathcal{II}} - C_{\mathcal{IF}} C_{\mathcal{FF}}^{-1} C_{\mathcal{FI}} \tag{3}$$

A. Efficient Implementation through Graphs

Evaluation of C_{eq} by first computing constituent matrices of Equation 1 and then doing the matrix operation of Equation 3 could be very inefficient. However the matrices $C_{\mathcal{II}}, C_{\mathcal{IF}}$ and $C_{\mathcal{FF}}$ are in general very sparse. Bunch and Rose [17] give efficient methods to handle such matrices. Also, the system matrix of Equation 1 is positive definite and one can use graph based techniques for equation solution (George and Liu [18]). In our implementation, we model the symmetric sparse matrix as an undirected graph. Each interconnect net, which is modeled as a terminal i in the multi-port model, is taken as node i in the graph. Coupling capacitance across terminal i and j is modeled as an edge with initial weight c_{ij} across nodes i and j. The initial weight of node i is taken as the self capacitance c_{ii} of terminal i. We do not add zero weight edges to the graph. Also we use thresholding on the self and coupling capacitances so that extremely small values are discarded. The threshold is automatically determined by the error tolerance value and the minimum capacitance value above which the capacitances are of interest to the designer.

Let us interpret the elimination process by examining how we eliminate $\dot{V}_{\mathcal{F}}$ from the first row in Equation 1. Let us consider the interconnect net 1 and a fill net k. Then the network equations for these two nets can be obtained from rows (1) and (k) of the expanded matrix Equation 1, they are,

$$c_{11}\dot{v}_1 - c_{12}\dot{v}_2 - \cdots - c_{1\,n+m}\dot{v}_{n+m} = i_1 \quad (4)$$

$$-c_{k\,1}\dot{v}_1 - \cdots + c_{k\,k}\dot{v}_k - \cdots - c_{k\,n+m}\dot{v}_{n+m} = 0. \quad (5)$$

Then after eliminating $\dot{v}_k$ from Equation 4, coefficients of 4 get updated to

$$(c_{11} - \frac{c_{1k}^2}{c_{kk}})\dot{v}_1 - (c_{12} + \frac{c_{1k}c_{2k}}{c_{kk}})\dot{v}_2 - \cdots - 0\dot{v}_k - \cdots - (c_{1\,n+m} + \frac{c_{1k}c_{n+m\,k}}{c_{kk}})\dot{v}_{n+m} = i_1$$

Generalizing this, we see that after fill net k has been eliminated, the equivalent coupling capacitance between nets, say i and j changes from c_{ij} to $c'_{ij} = c_{ij} + \frac{c_{ik}c_{jk}}{c_{kk}}$ and self capacitance of net i to $c'_{ii} = c_{ii} - \frac{c_{ik}^2}{c_{kk}}$.

The algorithm for computing equivalent capacitance is as follows:

1. Construct the weighted undirected graph
2. Pick a $suitable$ vertex v which corresponds to a fill net. How v is chosen is described later on
3. For all possible pairs of edges $\{(u,v),(w,v)\}$ incident on v, update c_{uw} by $c_{uw} + \frac{c_{uv}c_{wv}}{c_{vv}}$. If (u,w), does not exist ($c_{uw} = 0$), we create an edge from u to w and assign a weight of $\frac{c_{uv}c_{wv}}{c_{vv}}$ to it
4. For all vertices u adjacent to v update c_{uu} by $c_{uu} - \frac{c_{uv}^2}{c_{vv}}$
5. Delete v along with all edges incident on it
6. If there is any fill vertex still remaining go to step (2)
7. The weights on the remaining edges and vertices are the equivalent coupling and self capacitances

Figure 3 illustrates the step when vertex f corresponding to fill net $\mathcal{F}$ is being removed. Initially f has edges to a, b, c and d. When f is removed, it adds edges between (a,b), (a,c), (a,d), (b,c), (b,d) and (c,d). We see that there is already an edge between b and c. The weights on these 2 parallel edges can be added since it represents parallel capacitances.

Choosing v: The complexity of the above algorithm is proportional to the number of edges created in step (3). The objective is to ensure that sparsity of graphs is not lost in the elimination process. Towards this end, various heuristics like *Dynamic Minimum Degree, Dynamic Minimum Fill-in, Nested Dissection* etc. are suggested in [18]. In our implementation we use dynamic minimum degree heuristic.

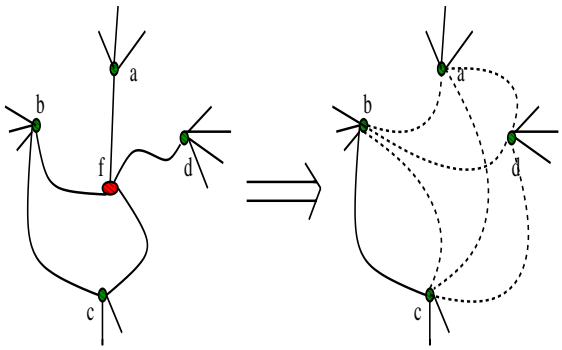

Fig. 3. *Illustration of vertex elimination using Graphs- Before & after vertex f being removed. Dotted arcs indicate the edges that are being added in the process of removing f*

B. Handling Intervals

In the algorithm described above, c_{ij} was a real value. However, RW method outputs an associated error e_{ij} with each c_{ij}. We describe how to update e_{ij} tightly along with c_{ij}.

Let X be a random variable estimated by n random samples $x_1, x_2, \cdots, x_n$. x, the mean of the n samples is an estimate for $E[X]$, the true mean of X. Let σ_x^2 be the the variance of the n samples. Then the standard error $e_x = \frac{\sigma_x}{\sqrt{n}}$ gives the standard deviation of the mean. Typically the error is reported as a 3-sigma confidence interval, which means $E[X]$ will lie within $(x - 3e_x, x + 3e_x)$ with 99.8% confidence. The notation $X = (x, e_x)$ means that x is the estimated mean of X and e_x is the associated standard error.

Let X and Y be two independent random variables which we have already estimated, and suppose now we are interested in $X + Y$. The estimate of the mean of $X + Y$ can be computed easily, and it is $x + y$. It can be verified that $e_{x+y} \leq e_x + e_y$. However using the property of errors of independent random variables, we can show that $e_{x+y} \leq \sqrt{e_x^2 + e_y^2}$. If $e_x = e_y$, then the new error bound is $\frac{1}{\sqrt{2}}$ of the naive bound; hence much tighter.

In the RW method, C_{ij} is a random variable which denotes the coupling capacitance from i to j. RW reports (c_{ij}, e_{ij}). We have to update these (c_{ij}, e_{ij}) pairs when we do the network reduction. Any two distinct C_{ij} and C_{kl} are estimated by independent random walks; hence C_{ij} and C_{kl} are independent. Looking back at the updates done by the network reduction method, we should provide tight bounds for standard errors of $X + Y$, $X - Y$, XY and $1/X$. Theorem II.1 provides the bounds.

Theorem II.1: Let $X = (x, e_x)$ and $Y = (y, e_y)$ be two independent random variables. Then

1. $X \pm Y = (x \pm y, \sqrt{e_y^2 + e_y^2})$
2. $XY = (xy, \sqrt{x^2e_y^2 + y^2e_x^2 + e_x^2e_y^2})$
3. $\frac{1}{X} = (\frac{1}{x}, \frac{1}{x - e_x} - \frac{1}{x})$

Proof: (1) and (2) follow from the definition of standard error and the identities

$$\sigma_{x+y}^2 = \sigma_x^2 + \sigma_y^2$$

$$\sigma_{xy}^2 = x^2\sigma_y^2 + y^2\sigma_x^2 + \sigma_x^2\sigma_y^2$$

respectively. The proof of (3) is straight forward. ■

It is easy to see that if there are no errors associated with X and Y, then there are no errors associated with either of the above operations and hence it boils down to the non-interval case. In our algorithm, whenever we update c_{ij}'s we also update the e_{ij}'s using Theorem II.1.

III. Modified Random Walk (MRW)

A. Monte-Carlo Integration through Random Walks (RW)

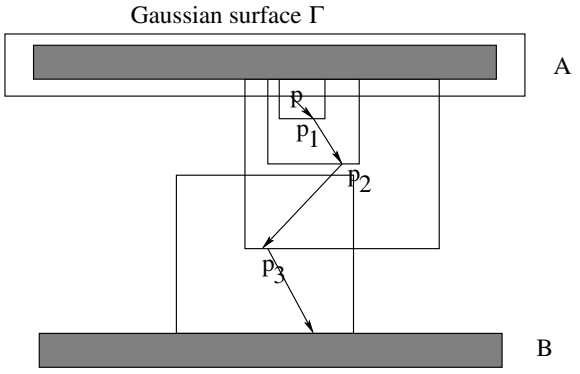

Fig. 4. *Illustration of random walk process.*

In this section we extend the RW method of [12] to handle fill nets by modifying the walk process around a floating metal. But

first we briefly describe the RW method of [12] which computes capacitances indirectly by evaluating the charge integrals by Monte-Carlo sampling. Total charge Q_i on net i can be expressed in terms of the coupling capacitance c_{ij} between nets i and j and the potential vector $(v_1, \ldots, v_n)$ as

$$Q_i = \sum_{j=1, j\neq i}^{n} c_{ij}(v_i - v_j). \quad (6)$$

If we keep net i at 0 volt and all the other nets at 1 volt, we can compute the self capacitance of net i and the coupling capacitances between net i and the other nets in the system. The self capacitance is equal to the total charge on i and the coupling capacitances are computed by linear separation of Q_i in Equation 6.

Basic electrostatic equations used for the evaluation of charge Q are given below. [1]

$$Q = -\int_{\Gamma} \epsilon_p \, \widehat{\mathbf{E}}(p) \cdot \widehat{\eta}(p) \, d\Gamma \quad (7)$$

$$\widehat{\mathbf{E}}(p) = \int_{\mathcal{A}} \widehat{\mathbf{G}}_p(p_1)\phi(p_1) \, d\mathcal{A} \quad (8)$$

$$\phi(p_1) = \int_{\mathcal{A}_1} G_{p_1}(p_2)\phi(p_2) \, d\mathcal{A}_1 \quad (9)$$

where,

Γ is a Gaussian surface

$d\Gamma$ is the elemental area on Γ containing point p

$\mathcal{A}$ is the surface of a maximal conductor free cube centered around p

$\mathcal{A}_1$ is the surface of a maximal conductor free cube centered around p_1

According to Gauss's law, total charge Q stored on a net in a multi-conductor system is evaluated by integrating the dot product of electric field $\widehat{\mathbf{E}}(.)$ and unit normal vector $\widehat{\eta}(.)$ on a surface Γ around the net. This surface is often called a *Gaussian surface*. By definition, Gaussian surfaces *should* completely enclose the net on which total charge is being computed and *should not* contain any other net in the system. Given the geometric description of the interconnect nets, a Gaussian surface can be constructed, and unit normal vectors can be computed at every point on the surface. However, the electric field $\widehat{\mathbf{E}}(p)$ at points p on Gaussian surface Γ needs to be evaluated. $\widehat{\mathbf{E}}(p)$ can be computed in terms of electric potentials around a cube C_p centered around point p. The field is related to boundary potentials through a vector Green's function $\widehat{\mathbf{G}}_p(.)$ as shown in Equation 8. The potential at point p_1 on the surface $\mathcal{A}$ of this cube C_p can again be expressed in terms of boundary potentials around another cube C_{p_1} centered around point p_1. The relationship involves a scalar Green's function $G_{p_1}(.)$ as shown in Equation 9. We evaluate the integral (7) by sampling points p on Γ. We very briefly sketch the algorithm with the help of Figure 4.

1. Build Gaussian surfaces around the nets to be extracted. For example around net A
2. Pick a point p on Γ so as to find an estimate for Equation 7
3. To find an estimate for Equation 7, an estimate for $\widehat{\mathbf{E}}(p)$ as given by Equation 8 is needed. For that, build a maximal conductor free cube C_p centered around p and pick a point p_1 on $\mathcal{A}$, the surface of C_p
4. Now to find an estimate for Equation 8, an estimate for $\phi(p_1)$ as given by Equation 9 is needed. For that, build another maximal conductor free cube C_{p_1} and pick a point p_2 on $\mathcal{A}_1$
5. If p_2 lies on a net say B, the walk ends as $\phi(p_2) = 1$. Otherwise keep generating points $p_3, \cdots$ until it ends in a net say again B.
6. Since the walk started at A and ended at B, the estimate computed above is an estimate of the coupling capacitance between A and B

[1] For notational simplicity we represent surface integral by simple integral sign $\int$.

B. Modified Random Walk

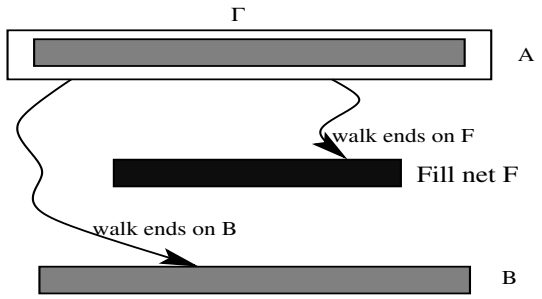

Fig. 5. *Random walk in the presence of fill net*

Consider a simple example shown in Figure 5. It has three nets; A, B and a fill net F. A walk beginning from net A could terminate on A or B or F. If it terminates on A, we get an estimate for the *self* capacitance of A. If it terminates on B, we get an estimate for the *direct coupling* capacitance between A and B. If it terminates on F, we get an estimate for the capacitance between A and F. We need to modify the walks so that (1) we don't compute coupling capacitances to F, and (2) instead of computing the direct coupling capacitance between A and B, we compute the *equivalent capacitance* (which in this simple case is $c_{eq} = c_{ab} + \frac{c_{af}c_{bf}}{c_{af}+c_{bf}}$). We want to have a technique which will work in conductor configurations having any number of complex shaped fill metals placed arbitrarily in a complex dielectric medium. This will enable the technique to be useful in equivalent capacitance extraction problems likely to be encountered in modern technologies using dummy fills.

Let us examine Equation 9, which enables the walk to proceed in the dielectric medium. A random walk at point p_1 in the dielectric medium is extended to the next point, by sampling points on the surface $\mathcal{A}_1$ of a conductor free cube centered at point p_1. The sampling is done treating the Green's function G_{p_1} as a probability density function (pdf) for point selection. This hop process has two prerequisites:

1. The point p_1 is in a dielectric medium with a non-zero size conductor free dielectric cube available centered around it.
2. There is a pre-computed Green's function available for the conductor free cube which can be used as pdf for point selection on its surface $\mathcal{A}_1$.

Clearly, when a walk hits a floating metal at point p_1 there is no conductor free cube possible centered at p_1. However if we use the fact that the fill metal is an equi-potential surface (since it is a piece of metal) we can write another potential Green's function at the fill metal and use that to continue the walk.

The potential $\phi(F)$ on a fill net F can be computed from the potential around its neighborhood as follows:

$$\phi(F) = \frac{1}{K} \int_{\Gamma_f} \frac{\epsilon_p}{x_p} \, \phi(p) d\Gamma_f \quad (10)$$

where p is a point on neighborhood boundary Γ_f around F (see Figure 6) . ϵ_p is the average permittivity over the line joining p and and the nearest point on F. x_p is the length of this line. K is a normalizer and is given by

$$K = \int_{\Gamma_f} \frac{\epsilon_p}{x_p} \, d\Gamma_f. \quad (11)$$

Equation 10 is conceptually similar to the Equation 9. While Equation 9 expresses the potential at a point in dielectric medium

in terms of potentials at a cube shaped boundary through a pre-computed Green's function, Equation 10 expresses potential at a point on a floating metal in terms of potentials on a not-necessarily cube shaped boundary. The boundary can best be thought of as (possibly non-uniform) inflation of floating metal. Also, $\frac{\epsilon_p}{K x_p}$ can be thought of as a Green's function (See Equation 9) for this boundary. Further more, such Green's functions can be pre-computed for each floating metal and stored in the form of a pdf for point selection on their corresponding boundaries.

Equation 10 along with Equations 7, 8 and 9 is interpreted as the *modified random walk (MRW) process.* Whenever a walk lands on a fill net, we use Equation 10 instead of Equation 9 to compute the potential at that point. To do this, we select a point on a neighborhood around the fill net (see Figure 6) and continue. With this modification, walks start and terminate only at non-fill nets (interconnects) and thus we get true equivalent capacitance between non-fill nets, taking into account the presence of all fill metals.

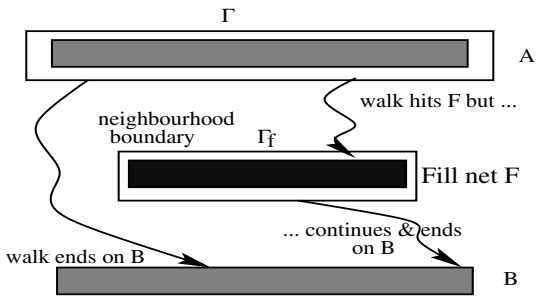

Fig. 6. *Modified random walk*

Neighborhood Γ_f around fill nets are implicitly constructed from inflations of constituent 3D boxes in the fill nets. A weight function is computed based on average permittivity value ϵ_p and average normal distance x_p on each face of the neighborhood. These can be easily computed from the geometric description of the fill net and the dielectric specification of its surroundings. This enables computation of equivalent capacitances in arbitrary dielectric profiles. In particular, this enables equivalent capacitance computation in uniform, layered and conformal dielectric profiles, which are the most commonly encountered situations in modern fabrication technologies.

This natural extension of RW is also well suited for parallel implementation. Hence the technique described in [19] can gracefully adapt to compute equivalent capacitances.

IV. Experimental Results

We present the accuracy and scalability of MRW through four sets of experiments. The first one consists of theoretical example for which the true answer can be arrived at by hand computation. The second set consists of real test structures with fills for which silicon data is available. The third set consists of synthetic examples. Accuracy in this case is established by comparing with the golden method - *NR*. The last one has real large designs with thousands of fills and is used to demonstrate the scalability of MRW. In all the experiments we use a 3-sigma confidence level. All computations were done on Sun-UltraSparc-III machine (1200MHz, 4GB RAM).

TABLE I

Equivalent capacitance between T and B in pF

true value	NR	MRW
19.42	$19.55 \pm 2.41\%$	$19.53 \pm 1.12\%$

Theoretical Example: *Parallel Plates* Figure 7 depicts a parallel plate capacitor with 4 thin parallel plates inserted between the top

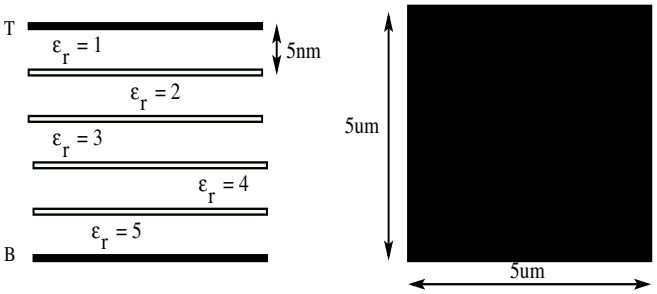

Fig. 7. *Front & Top views of 6 identical & equally spaced parallel plates. Middle 4 are fill nets. Thickness of each plate is 1nm*

(T) and bottom (B) plates. These in-between plates are treated as fill nets. The plates are made very thin (1nm) and have large areas so as to minimize the ratio of fringe capacitance to the overlap capacitance. Thus the equivalent capacitance between T and B is effectively a series combination of 5 capacitances. We put slabs of different dielectric strengths between the plates as shown in the front view. Table I contains true capacitance computed from series combination of 5 parallel plate capacitors along with the equivalent capacitances as computed by NR and MRW methods. One can note a very good correlation across these values. Because of some marginal fringe capacitances not accounted in analytical calculation, the equivalent capacitance between T and B is slightly more than the series approximation and we observe that in the results computed by NR and MRW.

TABLE II

Results of Test Structures - Capacitances in pF

Struc	# fills	SD	NR	MRW	MRW Time
P1S	2119	1.73	$1.73 \pm 3.40\%$	$1.75 \pm 1.96\%$	189s
P31	2119	1.20	$1.17 \pm 5.67\%$	$1.17 \pm 4.31\%$	664s
L2PSb	2552	4.39	$4.51 \pm 3.41\%$	$4.49 \pm 1.99\%$	108s
LA1Pa	2943	4.13	$4.41 \pm 10.91\%$	$4.13 \pm 1.81\%$	87s
FAPSa	2610	10.17	$10.24 \pm 10.30\%$	$10.27 \pm 1.74\%$	110s
FA3Sb	11833	6.224	-	$6.01 \pm 2.00\%$	128s
FA9Sa	27513	6.33	-	$6.52 \pm 2.12\%$	401s
FA5Sb	17782	6.39	-	$6.01 \pm 1.96\%$	325s

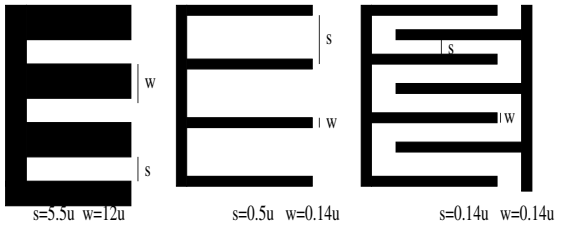

Fig. 8. *TSMC90nm Test Structures*

Real Test Structures: Figure 8 shows the top view of three test structures; plate, line and finger, which were fabricated using TSMC90nm technology. There are two identical plates in different metal layers in each structure. For example, P31 contains identical plates (as shown in left shape in Figure 8) in M3 & M1 and they are connected to pad. Similarly in FAPSa, the finger structure (mid-

dle shape in Figure 8) has no top plate (i.e., air) and a bottom plate is placed in polysilicon (P). Also, the spacing between the lines is 0.14um. Floating fills were added at the database level using the TSMC guidelines from the design rule document. The metal fill dimensions were 2um x 2um with a 2um space.

Table II gives the results. Owing to the large number of fill nets in each structure, the NR method took hours to compute full capacitance network and the equivalent capacitances. In certain test structures final result could not be computed by NR because too much computational effort was involved in computing the full capacitance network. On the contrary MRW could compute these capacitances in reasonable amounts of time. (reported in Table II) One can note very good correlation on equivalent capacitances with silicon data (SD).

TABLE III

Pair-wise equivalent capacitances (fF) between T, B, L and R.

Coupling Cap	NR	MRW
c_{BL}	$0.821 \pm 2.14\%$	$0.829 \pm 1.89\%$
c_{BR}	$0.833 \pm 2.10\%$	$0.826 \pm 1.96\%$
c_{BT}	$0.174 \pm 1.74\%$	$0.175 \pm 4.10\%$
c_{LR}	$0.107 \pm 2.41\%$	$0.109 \pm 5.12\%$
c_{LT}	$0.819 \pm 2.14\%$	$0.826 \pm 1.89\%$
c_{RT}	$0.822 \pm 2.17\%$	$0.830 \pm 1.95\%$

Synthetic Example: *2x2 Cross Bus* This synthetic test case is similar to the one shown in Figure 2. The 2 horizontal conductors of M2 are denoted as T (top) and B (bottom) respectively. Similarly the 2 vertical conductors of M1 are denoted as L (left) and R (right) respectively. There is an array of 11x11 fill metals between the conductors in each layer (total of 242 fill nets). Owing to the smaller number of fill nets, a very accurate capacitance network was computed by NR and one can note strong match across the equivalent capacitances as computed by NR and MRW methods. Table III summarizes the results.

TABLE IV

Results of Real Designs with fills - Coupling capacitance (in fF) between certain pair of nets

Design	# fills	MRW	Time
des1	60816	$5.69 \pm 2.84\%$	36m
des2	10383	$3.77 \pm 1.30\%$	401m

Real Designs: We used 2 real designs with thousands of fills present. Table IV shows the results of coupling capacitance between certain large nets. One can note that even in these large examples MRW can compute equivalent capacitances in reasonable time. For these examples NR results could not be computed, and no silicon data was available for comparison.

As noted above, equivalent capacitance numbers for certain examples were not computed by NR method. This is due to the fact that NR computes all coupling capacitances associated with each fill net within *some* error bounds. However, not all coupling capacitances have the same impact on the equivalent capacitance. A few of them could be computed with looser error bounds, which would still not affect the final outcome. However it is difficult to determine these insignificant capacitances a priori. So we end up computing each coupling capacitance with *same* error bound. This results in over computation in NR which increases its runtime. Since MRW does not try to compute capacitances associated with fill metals separately, it does not suffer from this. Hence runtime of MRW is much better than that of NR. Note that we use the variance reduction techniques reported in [20] in our implementations.

V. Conclusions

We have presented a natural and intuitive extension to the Monte Carlo method to compute equivalent capacitances in the presence of large number of fill metals. The extension is applicable to capacitance extraction problems commonly encountered in modern fabrication technologies using dummy metals. We have demonstrated the accuracy and scalability of our new method with synthetic as well as real examples and test chip measurement data. The method is extremely efficient in runtime and memory utilization. In addition, it also scales very well to perform extraction in presence of thousands of dummy fills in between signal nets.

References

[1] I. Ali, S. Roy, and G. Shinn, "Chemical-mechanical polishing of interlayer dielectric: A review," *Solid State Technol*, vol. 37, no. 10, pp. 63–70, 1994.

[2] K. Smekalin, "CMP dishing effects in shallow trench isolation," *Solid State Technol*, vol. 40, pp. 187–194, 1997.

[3] A.B. Kahng, G. Robins, A. Singh, and A. Zelikovsky, "Filling algorithms and analyses for layout density control," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 18, no. 4, pp. 445–462, April 1999.

[4] C.W. Liu, C.S. Tsai, J.M. Liu, and T. Shih, "Method of planarization using dummy leads," *US Patent 6,156,660*, December 5, 2000.

[5] M.T. Yang and H.T. Pan, "Method of automatically generating dummy metals for multilevel interconnection," *US Patent 5,798,298*, August 25, 1998.

[6] L. He, A. B. Kahng, K. Tam, and J. Xiong, "Design of integrated-circuit interconnects with accurate modeling of chemical-mechanical planarization," *Proceedings of SPIE Microlithography*, March 2005.

[7] Won-Seok Lee, Keun-Ho Lee, Jin-Kyu Park, Tae-Kyung Kim, Young-Kwan Park, and Jeong-Taek Kong, "Investigation of the capacitance deviation due to metal-fills and the effective interconnect geometry modeling," *Proceedings of Fourth International Symposium on Quality Electronic Design, 2003*, pp. 373–376, 24-26 March 2003.

[8] Mark M. Nelson, "Optimized pattern fill process for improved CMP uniformity and interconnect capacitance," *Proceedings of the 15th Biennial University/Government/Industry Microelectronics Symposium, 2003*, pp. 374–275, 30 June - 2 July 2003.

[9] B.E. Stine, D.S. Boning, J.E. Chung, L. Camiletti, F. Kruppa, E.R. Equi, W. Loh, S. Prasad, M. Muthukrishnan, D. Towery, M. Berman, and A. Kapoor, "The physical and electrical effects of metal-fill patterning practices for oxide chemical-mechanical polishing processes," *IEEE Trans. on Electron Devices*, vol. 45, no. 3, pp. 665–679, March 1998.

[10] Tai-Yu Chou and Z.J. Cendes, "Capacitance calculation of IC packages using the finite element method and planes of symmetry," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 13, no. 9, pp. 1159–1166, Sept. 1994.

[11] K. Nabors and J. White, "FastCap: A multipole accelerated 3-D capacitance extraction program," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 10, no. 11, pp. 1447–1459, November 1991.

[12] Y. L. Le Coz and R. B. Iverson, "A stochastic algorithm for high speed capacitance extraction in integrated circuits," *Solid State Electronics*, vol. 53, no. 7, pp. 1005–1012, 1992.

[13] O. Cueto, F. Charlet, and A. Farcy, "An efficient algorithm for 3d interconnect capacitance extraction considering floating conductors," *International Conference on Simulation of Semiconductor Processes and Devices*, pp. 107–110, 2002.

[14] Jin-Kyu Park, Keun-Ho Lee, Joo-Hee Lee, Young-Kwan Park, and Jeong-Taek Kong, "An exhaustive method for characterizing the interconnect capacitance considering the floating dummy-fills by employing an efficient field solving algorithm," *International Conference on Simulation of Semiconductor Processes and Devices, SISPAD 2000.*, pp. 98–101, 6-8 September 2000.

[15] Wenjian Yu, Mengsheng Zhang, and Zeyi Wang, "An efficient algorithm for 3-d interconnect capacitance extraction considering the floating dummy-fills," *Proceedings. 7th International Conference on Solid-State and Integrated Circuits Technology*, vol. 2, no. 18-21, pp. 1038–1041, Oct 2004.

[16] A. Kurokawa, T. Kanamoto, A. Kasebe, Y. Inoue, and H. Masuda, "Efficient capacitance extraction method for interconnects with dummy fills," *Custom Integrated Circuits Conference*, pp. 485–488, 2004.

[17] J. R. Bunch and D. J. Rose Eds., *Sparse Matrix Computations*, Academic Press, New York, NY, USA, 1976.

[18] A. George and J. W-H. Liu, *Computer Solution of Large Sparse Positive Definite Systems*, Prentice-Hall, Englewood Cliffs, NJ 07632, USA, 1981.

[19] Nidhi Sawhney, Shabbir Batterywala, Narendra Shenoy, and Richard Rudell, "Parallelizing a statistical capacitance extractor," *Proceedings of VLSI Design and Test*, pp. 253–267, 2004.

[20] Shabbir Batterywala and Madhav Desai, "Varaiance reduction in Monte Carlo capacitance extraction," *Proceedings of* 18^{th} *VLSI Conference*, pp. 85–90, January 2005.