Enhancing 3-D Random Walk Capacitance Solver with Analytic Surface Green's Functions of Transition Cubes

Jiechen Huang, Wenjian Yu*
Dept. Computer Science & Tech., BNRist, Tsinghua Univ., Beijing 100084, China

ABSTRACT

The complicated dielectric profile under advanced process technologies challenges the accuracy of floating random walk (FRW) based capacitance extraction, as the latter pre-computes the *surface Green's functions* for a finite set of multi-dielectric transition cubes and makes approximations of transition cubes during the FRW process. In this work, we derive analytic surface Green's functions for transition cubes with arbitrary stratified dielectrics and propose a fast algorithm named AGF to compute them. A capacitance solver named FRW-AGF is then proposed to incorporate AGF into the FRW process to accurately model realistic transition cubes. Experimental results show that the proposed AGF is over 100× faster than the state-of-the-art, and FRW-AGF largely improves the accuracy of RWCap4 [3, 16] (making all errors to golden values below 5%) without degrading computational speed and parallel scalability.

ACM Reference Format:
Jiechen Huang, Wenjian Yu. 2024. Enhancing 3-D Random Walk Capacitance Solver with Analytic Surface Green's Functions of Transition Cubes. In *61st ACM/IEEE Design Automation Conference (DAC '24), June 23–27, 2024, San Francisco, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3649329.3658263

1 INTRODUCTION

Advanced process technologies pose challenges to capacitance extraction in many aspects [2, 17, 19], where the increased complexity of dielectrics emerges as a key issue. Examples include thinner layers and deeper stack of dielectrics, high-K materials, and ubiquitous non-stratified dielectrics due to layout-dependent thickness variations and conformal dielectrics (see Fig. 1). These technology features remarkably affect parasitic capacitance, necessitating three-dimensional (3-D) capacitance field solvers with reliable accuracy to support circuit simulation and analysis [18].

Floating random walk (FRW) method is gaining popularity among capacitance field solvers. It offers advantages over the traditional methods like finite difference method (FDM) and finite element method (FEM): better scalability to large structures, reliable and tunable accuracy, and inherent parallelism [20]. However, the FRW method faces a notable weakness in handling complex dielectrics. Unlike FDM or FEM, where dielectrics are directly modeled into linear equations [9], FRW must modify sampling function (i.e. transition probability) and weight value according to the dielectric

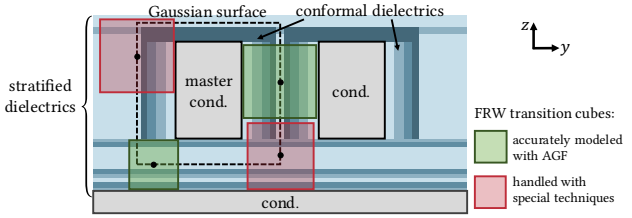

Figure 1: A cross-section view of interconnect structure embedded in stratified dielectrics and non-stratified/conformal dielectrics. The major idea of the proposed FRW-AGF using AGF to accurately model transition cubes is also shown.

configuration in transition cube [20]. They are the surface Green's function and its partial derivatives for 3-D Laplace equation, and can be solved by applying 3-D FDM, but at a high computational cost. Therefore, most existing works perform a pre-computation step to prepare look-up tables for these surface Green's functions, i.e. *Green's function table* (GFT) and *weight value table* (WVT) [11, 13, 16, 20, 21]. This pre-characterizes multi-dielectric transition cubes, but also makes approximation or simplification to avoid excessive memory cost for the GFT and WVT. Typically, transition cubes (called E4L cubes) with equal-thickness four layers of dielectrics and specific permittivity values are pre-characterized [11, 16]. While running the FRW method, each encountered transition cube is approximated by these E4L cubes through calculating equivalent permittivities. Another treatment for a transition cube including non-stratified dielectrics is approximating it by an eight-octant transition cube, whose GFT can be generated on-the-fly [15]. Recently, an approach was proposed to accelerate the pre-characterization of transition cubes with stratified dielectrics (including E4L cubes) and reduce the memory cost for the corresponding tables [16]. However, it is still based on 3-D FDM, and does not help on reducing the error caused by the approximation of using E4L cubes. Another recent work proposes to online generate sampling functions for arbitrary transition cubes with deep neural networks [12]. It improves the accuracy of FRW method for handling structures with complex dielectric environment, but at the cost of slowing down the computation by over 10× [12].

The above discussion outlines the fundamental role of computing surface Green's functions and the relevant issue of FRW method's inaccuracy for structures with complex dielectric environment. In this work, we deal with this weakness of the FRW method through a novel approach for computing the surface Green's functions. The contributions are as follows.

- A novel method (named **AGF**) which computes analytic surface Green's functions for transition cubes with arbitrary stratified dielectrics is proposed. It is over two orders of magnitude faster than the specially-optimized approach in [16] (i.e. TechGFT2 [3]).
- An FRW capacitance solver that invokes AGF on-the-fly (i.e. online and on demand) during the random-walk process, named

*This work is supported by Beijing Natural Science Foundation (Z220003).

DAC '24, June 23–27, 2024, San Francisco, CA, USA

ACM ISBN 979-8-4007-0601-1/24/06
https://doi.org/10.1145/3649329.3658263

FRW-AGF, is proposed. It utilizes the analytic surface Green's functions effectively via an approach with dynamic GFTs and WVTs, and does not rely on any pre-computed data. Its idea is sketched out in Fig. 1. FRW-AGF has been tested with wire and FinFET structures under real technologies. The results show that it is much more accurate than existing FRW methods by removing the systematic errors caused by approximation of transition cubes, and has comparable (or even faster) speed and parallel scalability to the FRW solver using pre-computed tables [3, 16].

2 PRELIMINARIES

2.1 FRW-based Capacitance Extraction

Capacitance extraction aims to compute the coupling capacitances C_{ij} between a master conductor i and all other conductors j. The basic idea of FRW-based extraction is to utilize Monte Carlo integration to evaluate the integral of Gauss's law [14]:

$$Q_i = -\oint_G \varepsilon(\mathbf{r})\nabla\phi(\mathbf{r})\cdot\mathbf{n}(\mathbf{r})\mathrm{d}s \tag{1}$$

where Q_i is the electric charge on i, G is the Gaussian surface. $\varepsilon(\mathbf{r})$ and $\phi(\mathbf{r})$ stand for the dielectric permittivity and electric potential at point $\mathbf{r}$, respectively. The FRW method solves $\phi(\mathbf{r})$ implicitly by sampling a dedicated Markov chain. For any surface S enclosing $\mathbf{r}$, a conditional probability density function (cPDF) $P(\mathbf{r},\mathbf{r}_1)$ exists:

$$\phi(\mathbf{r}) = \oint_S P(\mathbf{r},\mathbf{r}_1)\phi(\mathbf{r}_1)\mathrm{d}s. \tag{2}$$

Substitute (2) into (1), one can rearrange (1) to derive

$$Q_i = -\oint_G \frac{\varepsilon(\mathbf{r})}{F}\oint_{S_1} q(\mathbf{r},\mathbf{r}_1)w(\mathbf{r},\mathbf{r}_1)\oint_{S_2} P(\mathbf{r}_1,\mathbf{r}_2)\phi(\mathbf{r}_2)\mathrm{d}s_2\mathrm{d}s_1\mathrm{d}s \tag{3}$$

where the importance sampling function and weight value is [20]

$$q(\mathbf{r},\mathbf{r}_1)=\frac{|\nabla_{\mathbf{r}}P(\mathbf{r},\mathbf{r}_1)\cdot\mathbf{n}(\mathbf{r})|}{K(\mathbf{r})},\ w(\mathbf{r},\mathbf{r}_1)=\frac{K(\mathbf{r})F\nabla_{\mathbf{r}}P(\mathbf{r},\mathbf{r}_1)\cdot\mathbf{n}(\mathbf{r})}{|\nabla_{\mathbf{r}}P(\mathbf{r},\mathbf{r}_1)\cdot\mathbf{n}(\mathbf{r})|}. \tag{4}$$

The normalizing factors are $K(\mathbf{r}) = \oint_{S_1}|\nabla_{\mathbf{r}}P(\mathbf{r},\mathbf{r}_1)\cdot\mathbf{n}(\mathbf{r})|\mathrm{d}s_1$ and $F=\oint_G\varepsilon(\mathbf{r})\mathrm{d}s$. Based on (3), the Markov chain for estimating Q_i can be established as: 1) Starting point $\mathbf{r}$ is sampled on G according to $\frac{\varepsilon(\mathbf{r})}{F}$, 2) the first transition is based on $q(\mathbf{r},\mathbf{r}_1)$ and then $w(\mathbf{r},\mathbf{r}_1)$ can be calculated, 3) other transitions follow the cPDF $P(\mathbf{r}_{k-1},\mathbf{r}_k)$, and 4) conductor surfaces are the absorbing boundary [20].

In practice, transition domain S_k is designed to be cube-shaped and centered at $\mathbf{r}_{k-1}$, as illustrated in Fig. 2. To support the FRW, *surface Green's function* P and its gradient $\nabla_{\mathbf{r}}P$ must be evaluated [20]. If $\mathbf{r}=(x,y,z)$, $\nabla_{\mathbf{r}}P$ can also be expressed by three partial derivatives $\{\frac{\partial P}{\partial x},\frac{\partial P}{\partial y},\frac{\partial P}{\partial z}\}$. We refer to functions P, $\frac{\partial P}{\partial x}$, $\frac{\partial P}{\partial y}$ and $\frac{\partial P}{\partial z}$ as *surface Green's functions*. With them the weigh value can be easily obtained, and their values depend on the dielectrics inside S_k. Usually, these functions are pre-computed for specific dielectric configurations and stored as two tables, known as *Green's function table* (GFT) and *weight value table* (WVT) [20]. Obviously, items in GFT and WVT, which are used to approximate actual transition probability and weight value, fundamentally govern the accuracy of capacitance results.

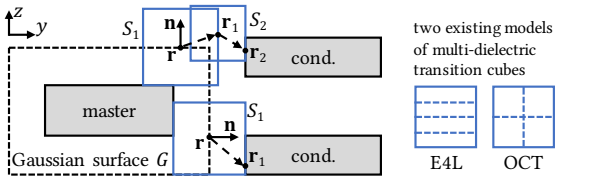

Figure 2: An illustration of two FRW paths. The E4L and OCT transition cubes are also illustrated.

2.2 Techniques for Obtaining the GFTs/WVTs

The configuration and computation of GFTs and WVTs are crucial for the FRW capacitance solver. For single-dielectric transition cube, they can be derived analytically [8, 14]. For multi-dielectric cases, three kinds of approaches can be seen in existing literature, but they all compromise either accuracy or efficiency.

Pre-characterizing specific process technology. Transition cubes are allowed to include only two stratified dielectrics in early works [20], which is later relaxed to four layers of dielectrics in [21]. The dilemma is that: limited hop length dramatically slows down the FRW process for advanced-technology structures, but allowing more dielectric layers leads to explosion of parameter space and oversized GFT/WVTs. Notice this approach is technology-dependent, which largely restricts the practicality of FRW solver.

Technology-independent pre-characterization with dielectric homogenization (E4L and OCT cubes). Two special dielectric configurations of transition cubes are described in [11], i.e., with equal-thickness four layers (E4L) and eight octants (OCT) dielectrics. The four or eight effective permittivity values are assigned by homogenizing the dielectrics in corresponding regions. This scheme is enhanced in [15, 16]. A hybrid scheme allowing three layers or E4L is presented in [13]. Notice the dielectric homogenization enables longer hops but induces systematic error.

Deep learning based surface Green's functions. In a latest work [12], surface Green's functions for complex-dielectric domains (about 50% of all encountered) are approximated with deep neural networks (DNNs). Several special DNN architectures are used and result in satisfactory accuracy for cases with non-stratified dielectrics. However, with this approach the solver is 12× on average slower than the methods with only pre-computed tables [12], no mentioning the generalization concern of the DNN models.

Most existing works pre-compute the GFT/WVTs for some special and empirical configurations of transition cubes to approximate the true sampling functions and weight values. The DNN-based solution is more accurate but too slow for practical use [12]. In this work, we will propose an approach based on accurate analytic surface Green's functions, which improves the accuracy of FRW solver without sacrifice on computational time.

3 ANALYTIC SURFACE GREEN'S FUNCTIONS OF MULTI-DIELECTRIC TRANSITION CUBE

3.1 The Problem and Its Analytic Solutions

We formulate the problem for computing the surface Green's functions and show that they can be derived analytically. Consider a transition cube with side-length L centered at $\mathbf{r}$ which encloses M layers of dielectrics. Suppose the dielectrics are layered along z-axis (see Fig. 3), and the origin of the Cartesian coordinate system is at the cube's corner with minimum x, y and z coordinates. The

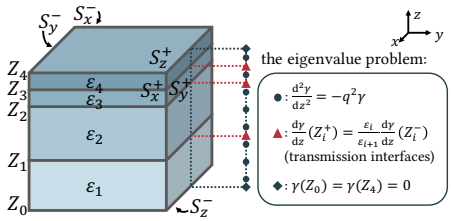

Figure 3: Major computation of AGF algorithm is to solve the eigenvalue problem (15) via 1-D FDM.

z-coordinates depicting the M dielectrics are: $0 = Z_0 < Z_1 < \cdots < Z_M = L$. And, the dielectric permittivity within the cube satisfies:

$$\varepsilon(x,y,z) = \varepsilon(z) = \varepsilon_i,\ \forall z \in (Z_{i-1}, Z_i),\ i = 1, 2, \ldots, M. \tag{5}$$

The cube's surface is denoted by six faces: S_x^+ at $x = L$, S_x^- at $x = 0$, S_y^+ at $y = L$, S_y^- at $y = 0$, S_z^+ at $z = L$, and S_z^- at $z = 0$, with the remaining variables ranging from 0 to L in each case.

The surface Green's function can be derived by solving the following Laplace equation with Dirichlet boundary condition:

$$\begin{cases} \nabla\cdot(\varepsilon\nabla\phi) = 0, & (0 < x,y,z < L), \\ \phi(L,y,z) = \phi_x^+(y,z),\ \phi(0,y,z) = \phi_x^-(y,z), \\ \phi(x,L,z) = \phi_y^+(x,z),\ \phi(x,0,z) = \phi_y^-(x,z), \\ \phi(x,y,L) = \phi_z^+(x,y),\ \phi(x,y,0) = \phi_z^-(x,y), \end{cases} \tag{6}$$

where ϕ_x^+ denotes the boundary function on S_x^+, and $\phi_x^-, \phi_y^+, \phi_y^-, \phi_z^+$ and ϕ_z^- are the functions on the other five faces. The solution of (6) can be expressed as a sum of integrals on the six faces [5], and that at the cube's center $\mathbf{r}$ is of our interest:

$$\phi(\mathbf{r}) = \iint_{S_x^+} P_x^+(u,v)\phi_x^+(u,v)\mathrm{d}u\mathrm{d}v + \ldots + \iint_{S_z^-} P_z^-(u,v)\phi_z^-(u,v)\mathrm{d}u\mathrm{d}v. \tag{7}$$

These six functions $P_x^+, \ldots, P_z^-$ are exactly the surface Green's function P on the corresponding faces (compare (7) with (2)). Similarly, we express the partial derivatives $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\}$ with their values on the six faces: $\frac{\partial P_x^+}{\partial x}, \frac{\partial P_x^-}{\partial x}, \ldots, \frac{\partial P_z^-}{\partial z}$, totally 18 functions.

The Laplace equation in (6) can be treated with separation of variables, resulting in three eigenvalue problems for the x-, y- and z-components of solution. The x- and y-components are simply trigonometric or hyperbolic functions, and the z-component can be expressed with the transmission matrix trick [1, 5]. Due to page limit, we omit the theoretic details, and give the obtained analytic solutions of surface Green's functions as follows. On S_z^+,

$$P_z^+(u,v) = \frac{4}{L^2}\sum_{n,m=0}^{\infty} \frac{\chi(\frac{L}{2}; l_{n,m})}{(-1)^{n+m}\chi(L; l_{n,m})} \sin(k_n u)\sin(k_m v), \tag{8}$$

$$\frac{\partial P_z^+}{\partial x}(u,v) = \frac{4}{L^2}\sum_{n,m=0}^{\infty} \frac{k_n'\chi(\frac{L}{2}; l'_{n,m})}{(-1)^{n+m}\chi(L; l'_{n,m})} \sin(k_n' u)\sin(k_m v), \tag{9}$$

where $k_n = \frac{(2n+1)\pi}{L}, k_n' = \frac{2n\pi}{L}, l_{n,m} = \sqrt{k_n^2 + k_m^2}, l'_{n,m} = \sqrt{{k_n'}^2 + k_m^2}$,

$$\chi(z; l) = \begin{bmatrix} \cosh l(z - Z_{I(z)}) \\ \sinh l(z - Z_{I(z)}) \end{bmatrix}^{\mathrm{T}} \cdot \prod_{i=1}^{I(z)} T_i \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{10}$$

$$I(z) = \max\{i, Z_i < z\}, \tag{11}$$

$$T_i = \begin{bmatrix} \cosh l(Z_i - Z_{i-1}) & \sinh l(Z_i - Z_{i-1}) \\ \frac{\varepsilon_i}{\varepsilon_{i+1}}\sinh l(Z_i - Z_{i-1}) & \frac{\varepsilon_i}{\varepsilon_{i+1}}\cosh l(Z_i - Z_{i-1}) \end{bmatrix}. \tag{12}$$

Due to symmetry, $\frac{\partial P_z^+}{\partial y}(u,v) = \frac{\partial P_z^+}{\partial x}(v,u)$. And, $\frac{\partial P_z^+}{\partial z}$ is of the same form as P_z^+, except that χ in the numerator is replaced with $\frac{\mathrm{d}\chi}{\mathrm{d}z}$. Reverse the order of dielectric layers and then (8)-(9) become solutions on S_z^-. On S_x^+,

$$P_x^+(u,v) = \frac{1}{L}\sum_{n=0}^{\infty}(-1)^n \sin(k_n u)\psi(v; \mathbf{h}_n), \tag{13}$$

$$\frac{\partial P_x^+}{\partial x}(u,v) = \frac{1}{L}\sum_{n=0}^{\infty}(-1)^n k_n' \sin(k_n' u)\psi(v; \mathbf{h}_n'), \tag{14}$$

where $\psi(v; \mathbf{h}_n) = \mathbf{h}_n^{\mathrm{T}}\mathbf{\Gamma}^{-1}\mathbf{v}$, $\mathbf{h}_n[i] = \gamma_i(\frac{L}{2})/\cosh(h_{n,i}\frac{L}{2})$, $\mathbf{\Gamma}[i,j] = \int_0^L \gamma_i(\tau)\gamma_j(\tau)\mathrm{d}\tau$, and $\mathbf{v}[i] = \gamma_i(v)$. For $\mathbf{h}_n$, $h_{n,i} = \sqrt{k_n^2 + q_i^2}$ (replaced with $h'_{n,i} = \sqrt{{k_n'}^2 + q_i^2}$ for $\mathbf{h}_n'$). Here q_i's denote an infinite set of eigenvalues of the following problem, and γ_i's denote the corresponding eigenfunctions.

$$\frac{\mathrm{d}^2\gamma}{\mathrm{d}z^2} = -q^2\gamma, \gamma(0) = \gamma(L) = 0, \frac{\mathrm{d}\gamma}{\mathrm{d}z}(Z_i^+) = \frac{\varepsilon_i}{\varepsilon_{i+1}}\frac{\mathrm{d}\gamma}{\mathrm{d}z}(Z_i^-), i = 1, \ldots, M. \tag{15}$$

And, $\frac{\partial P_x^+}{\partial y}$ and $\frac{\partial P_x^+}{\partial z}$ are of the same form as P_x^+, except that $\mathbf{h}_n$ is replaced with $\mathbf{h}_n^{(y)}$ and $\mathbf{h}_n^{(z)}$ respectively, such that $\mathbf{h}_n^{(y)}[i] = \frac{h_{n,i}\gamma_i(\frac{L}{2})}{\sinh(h_{n,i}\frac{L}{2})}$ and $\mathbf{h}_n^{(z)}[i] = \frac{\frac{\mathrm{d}\gamma_i}{\mathrm{d}z}(\frac{L}{2})}{\cosh(h_{n,i}\frac{L}{2})}$. On faces $S_x^+, S_x^-, S_y^+, S_y^-$, both P and $\frac{\partial P}{\partial z}$ are identical, $\frac{\partial P_x^+}{\partial x} = -\frac{\partial P_x^-}{\partial x} = \frac{\partial P_y^+}{\partial y} = -\frac{\partial P_y^-}{\partial y}$, and $\frac{\partial P_x^+}{\partial y} = \frac{\partial P_x^-}{\partial y} = \frac{\partial P_y^+}{\partial x} = \frac{\partial P_y^-}{\partial x}$. If dielectrics are stratified along x- or y-axis, these functions can be similarly derived.

3.2 Computation of the Analytic Functions

Some details for computing the analytic surface Green's functions (8)-(15) are : 1) The infinite series in them converge rapidly, allowing truncation within just 10 terms. 2) The eigenvalue problem in (15) has no closed-form solution and is solved with one-dimensional (1-D) FDM, as depicted in Fig. 3. In the discrete case, the solution of (15) is a finite set of eigenvalues and eigenfunctions, based on which the matrix $\mathbf{\Gamma}$ and vectors $\mathbf{h}_n, \mathbf{v}$ can be computed.

This algorithm for computing analytic surface Green's functions is named AGF. Consider an $N\times N\times N$-meshed transition cube. The functions are obtained by evaluating P and its partial derivatives on $6N^2$ surface panels. Existing works [13, 16, 20, 21] use N^3-point 3-D FDM to solve them. Although it has been optimized by exploiting symmetry [16], the time complexity is still $O(N^6)$ and memory usage is of $O(N^3)$. In contrast, major computation of AGF is to employ N-point 1-D FDM to solve the eigenvalue problem (15). That is to perform eigendecomposition of an $N\times N$ matrix. Therefore, the time complexity of AGF is $O(N^3)$, and the memory usage is of $O(N^2)$. Since AGF can be highly facilitated by matrix arithmetic, we implement it with BLAS and LAPACK routines provided by Eigen [4]. As an example, Fig. 4 displays surface Green's functions computed by AGF on some faces of a transition cube.

4 THE CAPACITANCE SOLVER FRW-AGF

4.1 The Idea of FRW-AGF

The technology-independent pre-characterization scheme with E4L and OCT transition cubes has been widely employed. However,

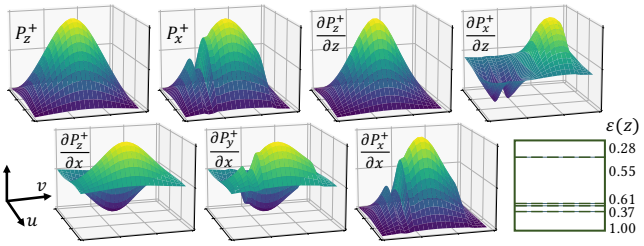

Figure 4: Some results of AGF for a transition cube with five-layer dielectrics, whose cross-section view labeled with normalized permittivities are shown at lower-right corner.

experiments reveal that they fail to well approximate the surface Green's functions in complex dielectric environment and lead to notable error. The AGF algorithm proposed in the last section can accurately produce surface Green's functions for arbitrary stratified dielectrics at a much faster speed than existing techniques. This inspires the idea of using AGF on-the-fly during the FRW procedure. Here "on-the-fly" signifies online and on-demand computation of weight value and transition probability encountered at every random-walk step. We name this enhanced method for capacitance extraction FRW-AGF, which is supposed to be more accurate than the existing method with pre-computed GFT/WVTs.

The first challenge in realizing FRW-AGF is to handle dielectric corners. For structures with non-stratified dielectrics, transition cubes may contain interfaces towards different directions, such as those outlined in red in Fig. 1. In those cases, surface Green's functions are hard to obtain analytically, so we propose special rules to handle dielectric corners. If interfaces are mostly towards one direction, e.g., the x-direction, we divide the cube according to x-direction interfaces and homogenizing dielectrics within each layer. Otherwise, we resort to the OCT model, which also enables on-the-fly sampling [15].

The second challenge is to make FRW-AGF run not slower than the existing method with pre-computed GFT/WVTs. The algorithm usually performs millions of hops to complete a capacitance extraction task. Although AGF is much faster than FDM-based solutions, invoking it at every hop is still expensive. To avoid redundant computation, we propose to store computed functions in hash tables for potential reuse. Different from the pre-computed GFT/WVTs, our tables are **dynamic**, i.e., their sizes grow during random-walk process. We store P in the GFT and $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\}$ in the WVT. With these dynamic tables, the search for surface Green's functions is described as Alg. 1. This offers high-speed retrieval of surface Green's functions, and therefore transition probabilities and weight values.

4.2 The Overall Algorithm and More Details

With the help of AGF and dynamic GFT/WVT, the overall FRW-AGF algorithm is described as Alg. 2.

For transition cubes with stratified dielectrics, the mapping from dielectric configuration to surface Green's functions (e.g. GFT) is

$$\mathcal{F} : \{(Z_1, \varepsilon_1), (Z_2, \varepsilon_2), \dots\} \mapsto P. \tag{16}$$

Notice that P is invariant if ε_i's are scaled by any factor. Also, the discrete version of P is invariant after scaling Z_i's. Therefore, we propose a *normalized description of dielectrics* for a transition cube: $\{(\hat{Z}_1, \frac{\hat{\varepsilon}_1}{\hat{\varepsilon}_{\max}}), (\hat{Z}_2, \frac{\hat{\varepsilon}_2}{\hat{\varepsilon}_{\max}}), \dots\}$, where $\hat{Z}_i$'s have been scaled and discretized into $\{1, 2 \dots, N\}$. Layers thinner than $\frac{L}{N}$ may be merged due to discretization, in which case $\hat{\varepsilon}_i$'s would be averaged. Under normalized description, the number of distinct transition cubes in one extraction is limited, so most surface Green's functions can be kept and reused. Therefore, we implement the GFT and WVT as hash tables with high-performance insertion and retrieval, with normalized descriptions of dielectrics as their keys. Only when the requested item cannot be found in GFT or WVT, AGF will be invoked and its result will be inserted as a new item.

Algorithm 1 Find surface Green's function for a transition cube.

Input: A transition cube S. A hash table T (the dynamic GFT or WVT).
Output: A function P (for GFT) or a set $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\}$ (for WVT).

if S contains dielectric corners **then**
  Perform special techniques in Sec. 4.1 and choose AGF or OCT;
  if OCT is chosen **then**
    return the OCT-based function proposed in [15];
  end if
end if
$\mathcal{D} \leftarrow$ compute the normalized description for dielectrics in S;
$\mathcal{K} \leftarrow$ compute the hash of $\mathcal{D}$;
if $\mathcal{K}$ is not a key of T **then**
  Run AGF algorithm for $\mathcal{D}$ and insert results into T with $\mathcal{K}$ as key;
end if
return the item in T indexed by $\mathcal{K}$;

Algorithm 2 FRW-AGF: the FRW algorithm with on-the-fly surface Green's functions.

Input: A 3-D structure of conductors and dielectrics. Master conductor i.
Output: Capacitance values $C_{ij}, \forall j$.

Construct Gaussian surface G enclosing i;
GFT $\leftarrow \{\}$; WVT $\leftarrow \{\}$; $C_{ij} \leftarrow 0, \forall j$; $n_{\text{path}} \leftarrow 0$;
repeat
  Sample $\mathbf{r}$ on G and determine its normal direction $\mathbf{n}$;
  $S \leftarrow$ generate a maximal conductor-free cube centered at $\mathbf{r}$;
  $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\} \leftarrow$ invoke Alg. 1 with S and WVT as inputs;
  $\mathbf{r}_1 \leftarrow$ sample on S following q in (4), based on $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\}$;
  $\omega \leftarrow$ evaluate w in (4), based on $\mathbf{r}, \mathbf{r}_1$ and $\{\frac{\partial P}{\partial x}, \frac{\partial P}{\partial y}, \frac{\partial P}{\partial z}\}$;
  while $\mathbf{r}_1$ is not on conductor surface **do**
    $S \leftarrow$ generate a maximal conductor-free cube centered at $\mathbf{r}_1$;
    $P \leftarrow$ invoke Alg. 1 with S and GFT as inputs;
    $\mathbf{r}_1 \leftarrow$ sample on S following the cPDF P;
  end while
  $C_{ik} \leftarrow C_{ik} + \omega$; $n_{\text{path}} \leftarrow n_{\text{path}} + 1$; // say $\mathbf{r}_1$ is on conductor k
until the convergence criterion is met
$C_{ij} \leftarrow C_{ij}/n_{\text{path}}, \forall j$.

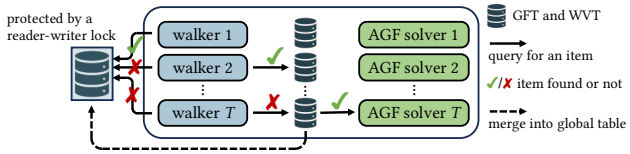

Figure 5: Multi-thread parallel scheme of FRW-AGF: the global tables are shared across T threads, while FRW walkers, buffer tables and AGF solvers are private in each thread.

The multi-thread parallel scheme for FRW-AGF (see Fig. 5) is also designed to inherit the parallel efficiency of FRW algorithm. To minimize redundancy, we share the dynamic GFT and WVT across threads. However, this necessitates mutually exclusive access,

potentially impacting the inherent parallelism. To guarantee low-latency retrieval, we design a *reader-writer lock* to protect the global resources, enabling parallel retrievals [6]. Moreover, we design local buffers to accumulate insertions and make batched updates to the global table. Once a buffer table reaches a size threshold, its contents are transferred to the global table. With these techniques, FRW-AGF maintains good multi-thread scalability as existing FRW algorithms.

5 NUMERICAL RESULTS

We have implemented the proposed algorithms of AGF and FRW-AGF in C++. They are compared with RWCap4 programs [3, 16]. All Experiments are carried out on a Linux server with Intel Xeon 8375C CPUs at 2.9GHz and with serial computing, except for the experiment of parallel computing in Sec. 5.3.

Table 1: The comparison results of TechGFT2 [3] and the proposed AGF on pre-characterizing E4L transition cubes.

N	TechGFT2 [3]		AGF		D_{KL}	RMSE	Sp.
	Time	Mem.	Time	Mem.			
20	33.4 s	19.4 MB	0.30 s	4.32 MB	6.67×10^{-4}	1.28×10^{-5}	**111**×
24	66.4 s	26.4 MB	0.37 s	4.33 MB	4.29×10^{-4}	6.32×10^{-6}	**179**×
32	215 s	53.6 MB	0.56 s	4.23 MB	2.20×10^{-4}	2.09×10^{-6}	**384**×
48	1395 s	199 MB	1.07 s	4.41 MB	9.04×10^{-5}	4.40×10^{-7}	**1304**×
72	3.52 hr	1.13 GB	3.10 s	4.85 MB	3.85×10^{-5}	9.35×10^{-8}	**4088**×

5.1 Using AGF for Pre-characterization

We follow the experimental setting in [16] to compute the GFT/WVT for the E4L transition cubes. The dielectric permittivity in four layers are picked in [0.5,1] with a step of 0.05, resulting in totally 2541 distinct transition cubes (see Sec. V-B in [16]). Techniques in [16] have been implemented in TechGFT2 [3], an FDM-based program accelerated by exploiting symmetry of surface Green's functions. The surface Green's functions obtained with TechGFT2 and AGF are denoted by P_1 and P_2, respectively. We measure their discrepancy with root-mean-square error (RMSE) and KL divergence [10]:

$$D_{\mathrm{KL}}(P_1\|P_2) = \sum_{x\in\{\text{surface panels}\}} P_1(x)\ln\frac{P_1(x)}{P_2(x)}. \tag{17}$$

In Tab. 1, we list the total runtime and peak memory consumed for different discretization numbers on the cube (N). It can be seen that AGF is memory-efficient and faster than TechGFT2, with **111× to 4088× speedup** (approximately of $O(N^3)$). Values of D_{KL} and RMSE are averaged over all surface Green's functions, and indicate that our results are highly consistent with FDM's results.

5.2 Using FRW-AGF for Capacitance Extraction

We use FRW-AGF to extract 10 interconnect structures. Cases A and B are the simple parallel-wire structures obtained from [3]. They also present (i.e. Cases 1 and 4) in [16, 20]. Cases 1 through 6 are the interconnect structures consisting of parallel wires in between two conductor plates. For technology nodes, Cases 1-3 are at 16nm, Case 4 at 28nm, and Cases 5-6 at 55nm. Cases 7 and 8 are two FinFET structures at 16nm technology node. They are all from our industrial partner. All the cases involve multi-layer dielectric stacks and Cases 1 through 8 include non-stratified dielectrics. The FinFET cases include high-K dielectrics as well. We consider C_{self}, C_{c} and C_{bot} (coupling capacitances to the adjacent wire and the bottom plate respectively) for Cases 1-6, and C_{self} and C_1 (the largest coupling capacitance) for Cases 7-8. In all experiments, the termination criterion of FRW is to ensure the standard deviation below 1% on all considered capacitances.

The shared FRW solver RWCap4 at [3] employs static GFTs and WVTs of pre-characterized E4L transition cubes [16], and the OCT-based sampling technique [15]. In addition, we design an FDM solver specially to characterize transition cubes including arbitrary dielectric configuration. We invoke it to make online computation of the surface Green's functions of transition cubes with complex dielectrics. This provides a more accurate FRW solver (named **FRW-FDM**) but with large time cost, for comparison. The following metrics are used to measure the average error of weight values and transition probabilities across all transition cubes:

$$\delta_\omega = [\sum_{i=1}^{n_\omega}(\omega_i-\omega_i')^2/n_\omega]^{\frac{1}{2}},\ \delta_{\mathrm{KL}} = \sum_{i=1}^{n_P} D_{\mathrm{KL}}(P_i\|P_i')/n_P \tag{18}$$

where ω_i and P_i are the i-th weight value and transition probability, ω_i' and P_i' are accurate ones computed with the aforementioned FDM solver, and n_ω and n_P are their total numbers respectively.

For Cases A and B, the golden values of C_{self} are 1.890fF and 0.360fF [16], while our FRW-AGF gets 1.858fF and 0.360fF, respectively. The results on Cases 1-6 are listed in Tab. 2 and those on Cases 7-8 in Tab. 3. The results of Raphael, an FDM-based golden tool [7], are also listed. From them we see that FRW-AGF exhibits better accuracy than RWCap4, whose error is large especially for FinFET. The error of FRW-AGF is **always less than 5%**. Besides, surface Green's functions computed with FRW-AGF are much closer to the accurate ones than those derived from the pre-characterized tables in RWCap4 (see δ_ω and δ_{KL}). Notice that FRW-FDM is also very accurate, but costs a day or more time for each case.

Table 2: The capacitance results (in unit of 10^{-15} F) and relative errors for Cases 1-6. The δ_ω and δ_{KL} (both in unit of 10^{-2}) measure the averaged errors of weight value and transition probability, respectively. (Errors larger than 5% are underlined)

Case	Raphael			FRW-FDM			RWCap4 [3]					FRW-AGF				
	C_{self}	C_{c}	C_{bot}	C_{self}/Err.	C_{c}/Err.	C_{bot}/Err.	C_{self}/Err.	C_{c}/Err.	C_{bot}/Err.	δ_ω	δ_{KL}	C_{self}/Err.	C_{c}/Err.	C_{bot}/Err.	δ_ω	δ_{KL}
1	15.9	1.10	12.9	15.6/1.9%	1.09/0.7%	12.7/2.1%	14.3/10%	1.07/2.7%	11.4/12%	6.99	1.71	15.8/**0.5%**	1.08/**1.6%**	12.9/**0.1%**	1.70	0.28
2	16.5	2.97	8.17	16.7/1.4%	3.06/3.0%	8.25/1.0%	17.1/3.9%	3.00/1.1%	8.91/9.0%	9.10	1.63	16.7/**1.6%**	3.01/**1.5%**	8.37/**2.3%**	1.39	0.08
3	21.6	8.03	3.87	21.3/1.2%	7.97/0.7%	3.76/2.7%	20.8/3.6%	7.92/1.4%	3.36/13%	11.52	2.04	21.1/**2.2%**	7.82/**2.6%**	3.82/**1.2%**	3.88	0.36
4	8.21	0.926	5.69	8.09/1.5%	0.933/0.8%	5.65/0.8%	8.34/1.5%	0.826/11%	6.07/6.5%	12.04	3.88	8.30/**1.2%**	0.933/**0.8%**	5.77/**1.3%**	8.43	2.14
5	14.2	0.978	11.3	14.0/1.5%	0.970/0.8%	11.2/1.4%	14.4/1.6%	0.884/9.6%	11.9/4.8%	0.51	2.77	14.3/**0.4%**	0.980/**0.2%**	11.4/**0.8%**	0.51	1.21
6	9.58	0.786	7.46	9.57/0.1%	0.789/0.5%	7.46/0.0%	10.0/4.6%	0.711/9.5%	8.11/8.6%	0.51	3.11	9.84/**2.6%**	0.799/**1.7%**	7.70/**3.1%**	0.51	1.99
avg.	-	-	-	- /1.27%	- /1.08%	- /1.33%	- /4.22%	- /5.86%	- /8.98%	6.78	2.52	- /**1.39%**	- /**1.40%**	- /**1.49%**	2.74	1.01

Table 3: The capacitance results (in unit of 10^{-18} F) and relative errors for Cases 7-8. Five master conductors are set for each case. (Errors larger than 5% are underlined)

Case	Raphael		RWCap4 [3]				FRW-AGF			
	C_{self}	C_1	C_{self}/Err.	C_1/Err.	δ_ω	δ_{KL}	C_{self}/Err.	C_1/Err.	δ_ω	δ_{KL}
7	933	465	852/8.6%	417/10%	8.13	1.08	911/**2.3%**	452/**2.8%**	3.02	0.09
	967	442	886/8.4%	395/11%	5.99	1.03	934/**3.4%**	421/**4.9%**	3.31	0.14
	720	459	651/9.7%	406/12%	14.07	2.52	708/**1.7%**	453/**1.3%**	3.60	0.31
	1010	437	890/12%	387/11%	15.90	2.02	1012/**0.2%**	432/**1.1%**	3.27	0.18
	877	424	815/7.1%	376/11%	16.30	2.43	861/**1.9%**	417/**1.7%**	3.40	0.29
8	883	449	816/7.6%	410/8.6%	5.85	1.50	868/**1.7%**	444/**1.1%**	3.37	0.11
	886	442	813/8.2%	398/10%	6.38	1.73	872/**1.6%**	436/**1.5%**	3.26	0.17
	675	456	587/13%	403/12%	12.51	2.77	672/**0.4%**	452/**1.0%**	2.87	0.21
	915	446	811/11%	398/11%	12.31	2.71	902/**1.5%**	440/**1.2%**	2.88	0.26
	820	420	725/12%	373/11%	13.75	2.68	821/**0.1%**	415/**1.3%**	2.65	0.24
avg.	-	-	- /9.73%	- /10.75%	11.12	2.05	- /**1.47%**	- /**1.78%**	3.16	0.20

In Tab. 4, we list runtime, memory cost and other details of RWCap4 and FRW-AGF for all the cases. FRW-AGF is overall faster than RWCap4 because: 1) accurate weight values may lead to faster convergence of stochastic error (see #walks), and 2) modeling of E4L cubes in RWCap4 adds computational overhead. The size of dynamic tables (i.e. the number of distinct cubes or AGF calls) is small and always below 0.2% of #hops, which means very effective reuse. The resort to OCT is infrequent, accounting for an average of only 4.5% of all hops. Furthermore, we collect the statistics about items in dynamic tables for Cases 1 and 7, shown in Fig. 6. They suggest that many transition cubes contain over 4 layers of dielectrics. Obviously, RWCap4 would lack accuracy for these cases.

Table 4: Memory (MB), runtime (s), total numbers of executed walks and hops, and the finalized size of dynamic GFT&WVT.

Case	#blks	RWCap4 [3]				FRW-AGF					
		#walks	#hops	Mem.	T_{total}	#walks	#hops	Table Size	Mem.	T_{AGF}	T_{total}
A [3]	4	0.97M	11M	109	6.56	0.98M	11M	1440	**11**	0.13	**5.65**
B [3]	4	0.97M	12M	109	7.00	0.94M	12M	1533	**11**	0.13	**5.77**
1	406	4.6M	43M	110	118	2.3M	41M	19802	**72**	1.79	**65.5**
2	8754	0.88M	7.8M	186	19.4	0.72M	5.9M	6728	**130**	0.57	**11.3**
3	1321	1.0M	8.3M	380	**34.8**	2.0M	42M	18254	**349**	1.65	80.7
4	29	6.5M	46M	109	105	5.5M	39M	26782	**70**	2.47	**64.4**
5	25	11M	66M	109	118	9.4M	60M	28936	**74**	2.75	**78.2**
6	29	6.3M	47M	109	89.5	5.4M	40M	26188	**68**	2.47	**60.9**
7	276	4.2M	31M	**110**	111	4.1M	34M	64498	166	5.43	**94.8**
8	127	3.3M	20M	**109**	73.6	3.2M	24M	48033	126	4.07	**64.7**

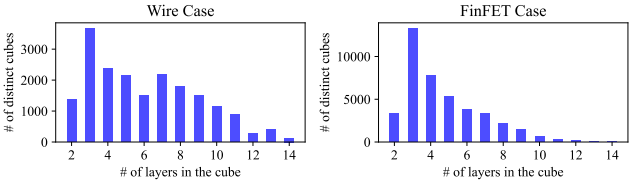

Figure 6: The distribution of distinct cubes in the dynamic GFT and WVT with respect to the number of dielectric layers.

5.3 Parallel Scalability of FRW-AGF

The parallel scalability of FRW-AGF is examined by running it with multiple threads. The results for Case 1 and 7 are shown in Fig. 7. From it we see that the proposed techniques in Sec. 4.2 largely reduce the resource sharing overhead. As the result, the parallel efficiency of FRW-AGF is close to that of the existing FRW solver.

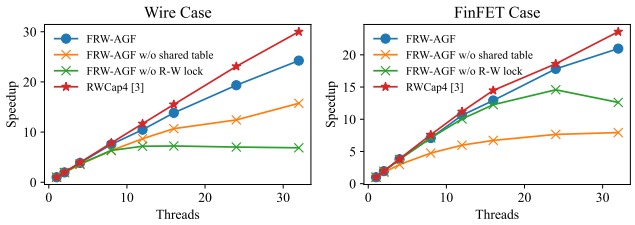

Figure 7: Parallel speedup vs. the number of threads for the proposed FRW-AGF and RWCap4 [3].

6 CONCLUSIONS

The FRW method for multi-dielectric capacitance extraction runs upon a set of surface Green's functions. Based on their analytic solutions we propose an ultra fast algorithm to compute them for transition cubes with arbitrary stratified dielectrics. Incorporating it into the FRW process, we then propose an FRW solver named FRW-AGF, which models the multi-dielectric transition cubes more accurately. Compared to existing methods, FRW-AGF is free of any pre-computed data yet achieves higher accuracy for complex-dielectric structures, and good computational efficiency as well.

REFERENCES

[1] W.C. Chew. 1999. *Waves and Fields in Inhomogenous Media.* Wiley.
[2] K. Chow and C. Relyea. 2018. *Parasitic Extraction Technologies for Advanced Node and 3D-IC Design.* Siemens.
[3] Numbda Group. 2023. *RWCap-v4.* [Online]. Available: http://numbda.cs.tsinghua.edu.cn/download/RWCap_v4_en.html.
[4] G. Guennebaud, B. Jacob, et al. 2010. Eigen v3. http://eigen.tuxfamily.org.
[5] R. Haberman. 2012. *Applied Partial Differential Equations with Fourier Series and Boundary Value Problems.* Pearson Education.
[6] M. Herlihy, N. Shavit, V. Luchangco, and M. Spear. 2020. *The Art of Multiprocessor Programming.* Elsevier Science.
[7] Synopsys Inc. Accessed: 2023. *TCAD-Raphael.* [Online]. Available: https://www.synopsys.com/manufacturing/tcad/interconnect-simulation/raphael.html.
[8] R. B. Iverson and Y. L. Le Coz. 2001. A floating random-walk algorithm for extracting electrical capacitance. *Mathematics and Computers in Simulation* 55, 1 (2001), 59–66.
[9] J.M. Jin. 2015. *Theory and Computation of Electromagnetic Fields.* Wiley.
[10] S. Kullback and R. A. Leibler. 1951. On information and sufficiency. *The Annals of Mathematical Statistics* 22, 1 (1951), 79–86.
[11] G. Rollins. 2010. *SESS Rapid3D 20X Performance Improvement.* Synopsys.
[12] M. Visvardis, P. Liaskovitis, and E. Efstathiou. 2023. Deep-learning-driven random walk method for capacitance extraction. *IEEE TCAD* 42, 8 (2023), 2643–2656.
[13] M. Visvardis, E. Lourandakis, and S. Stefanou. 2015. Pre-computation system and method for transition domain characterization within a floating random walk based multi-dielectric capacitance extraction methodology. US Patent 10,274,533.
[14] Y. L. Le Coz and R. B. Iverson. 1992. A stochastic algorithm for high speed capacitance extraction in integrated circuits. *Solid-State Electron.* 35, 7 (1992), 1005–1012.
[15] M. Yang and W. Yu. 2020. Floating random walk capacitance solver tackling conformal dielectric with on-the-fly sampling on eight-octant transition cubes. *IEEE TCAD* 39, 12 (2020), 4935–4943.
[16] M. Yang, W. Yu, M. Song, and N. Xu. 2022. Volume reduction and fast generation of the precharacterization data for floating random walk-based capacitance extraction. *IEEE TCAD* 41, 5 (2022), 1467–1480.
[17] W. Yu, M. Song, and M. Yang. 2021. Advancements and challenges on parasitic extraction for advanced process technologies. In *Proc. ASP-DAC.* 841–846.
[18] W. Yu and X. Wang. 2014. *Advanced Field-Solver Techniques for RC Extraction of Integrated Circuits.* Springer.
[19] W. Yu, Q. Zhang, Z. Ye, and Z. Luo. 2012. Efficient statistical capacitance extraction of nanometer interconnects considering the on-chip line edge roughness. *Microelectronics Reliability* 52, 4 (2012), 704–710.
[20] W. Yu, H. Zhuang, C. Zhang, G. Hu, and Z. Liu. 2013. RWCap: A floating random walk solver for 3-D capacitance extraction of very-large-scale integration interconnects. *IEEE TCAD* 32, 3 (2013), 353–366.
[21] B. Zhang, W. Yu, and C. Zhang. 2016. Improved pre-characterization method for the random walk based capacitance extraction of multi-dielectric VLSI interconnects. *Int. J. Numer. Model.* 29, 1 (2016), 21–34.