

# Massively Parallel Computation for 3-D Nonlinear Finite Edge Element Problem With Transmission Line Decoupling Technique

Jiacong Li<sup>1</sup>, Peng Liu<sup>1</sup>, and Venkata Dinavahi<sup>1</sup>

Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada

Transmission line method (TLM) has been used in 2-D scalar finite-element (FE) analysis due to its parallelism and constant admittance matrix. In this paper, the TLM is extended for the 3-D nonlinear vector FE problem that is more widely used for electromagnetic apparatus in practice. TLM is specially adapted for tetrahedron edge elements to calculate quasi-static electromagnetic field distribution for eddy current problems, and a dummy scalar gauge is applied to make the reduced magnetic vector FE formulation full-ranked and uniquely solvable by TLM. For each element, the nonlinearity is separated by transmission lines and only local small-scale Newton–Raphson iteration is needed, which is suitable for massive parallelization due to the independence between different elements. The TLM is implemented on a many-core GPU for a nonlinear FE power inductor case study, and the comparison of the results with a commercial FE software shows over 50 times speedup with a relative error of less than 2%.

**Index Terms**—3-D edge element, Coulomb gauge, eddy current, finite-element method (FEM), nonlinear, reduced magnetic vector potential (MVP), transmission line method (TLM).

## I. INTRODUCTION

FINITE-ELEMENT method (FEM) has become one of the most important tools in engineering design and simulation. FEM is widely used for field calculation in electromagnetic apparatus such as transformer, electrical motor, and power inductor, where complex geometry and nonlinearity stemming from ferromagnetic materials are always encountered.

Traditionally, the Newton–Raphson (N-R) method deals with such nonlinearity in FEM formulations. The solution of the nonlinear matrix system is gradually reached by solving the global Jacobian matrix in every iteration. However, the Jacobian matrix must be changed and repeatedly factorized at each iteration step, which slows down the computation dramatically.

On the other hand, with a bottleneck of clock speed, the general trend in hardware development is to increase the number of processing units for better overall performance. Multi-core CPUs and many-core GPUs are widely seen in high-performance computing. For example, the recently released NVIDIA Tesla V100 GPU is equipped with 5120 Cuda cores and 16 GB HBM2 memory [1], which encourages researchers to develop algorithms with high parallelism to exploit such computational power [2].

Under such circumstances, the transmission line method (TLM), which was introduced to the nonlinear circuit network [3], [4] and 2-D FEM [5], [6] decades ago, has witnessed increasing attention recently [7], [8]. When a nonlinear system is solved by TLM, the update happens only inside the system vector in each iteration step. This indicates that the

system matrix remains the same during the whole solution process, and only one-time matrix factorization is necessary. In addition, the parallelism of the TLM-FEM is high because elemental local nonlinearity is separated by transmission lines. By utilizing these properties, the speed of TLM-FEM is fast enough even for real-time simulation [7].

However, all of the above TLM-FEM applications are based on 2-D triangle elements, while in reality, the geometry of an electromagnetic model is in 3-D form but the 3-D-TLM-FEM implementation is never conducted. Moreover, the above FEM-TLM was only implemented for nodal elements in the scalar form; however, nodal elements result in a large inaccuracy at sharp corners of geometry and when materials have large permeability difference, the numerical error becomes substantial [9], [10]. On the contrary, edge elements [11] and vector basis functions, which are widely seen and have been utilized for network equivalence [12], [13] in the 3-D FEM analysis, do not have such problem. Considering the above-mentioned facts, it becomes paramount to explore new algorithms to implement transmission line modeling in 3-D edges elements, especially for nonlinear transient electromagnetic analysis where a large amount of computational resource is needed.

In this paper, for the first time, the transmission line decoupling technique is modified and extended from 2-D nodal scalar elements (NSE) to 3-D edge vector elements (EVE) in a nonlinear electromagnetic field problem. Challenges caused by the difference between 3-D EVE and 2-D NSE are successfully tackled. First, 3-D FE-discretized formulation for reduced magnetic vector potential (MVP) is introduced. In contrast with 2-D triangular NSE, the 3-D EVE formulation is not full-ranked and a gauge is added to make a unique solvable nonlinear matrix system.

An equivalent electrical circuit network (rather than a magnetic equivalent circuit network) is then extracted from the matrix system to facilitate the introduction of transmission

Manuscript received February 25, 2019; revised May 7, 2019 and June 12, 2019; accepted June 13, 2019. Date of publication July 9, 2019; date of current version September 18, 2019. Corresponding author: P. Liu (e-mail: pliu3@ualberta.ca).

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMAG.2019.2923535

0018-9464 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.

See [http://www.ieee.org/publications\\_standards/publications/rights/index.html](http://www.ieee.org/publications_standards/publications/rights/index.html) for more information.

lines. Next, this network is solved with elemental nonlinearity decoupled by the TLM technique. A new concept of scattering box is defined to abstract away elemental local nonlinearity in the form of 21 nonlinear resistors, compared with only 3 nonlinear resistors for 2-D triangular NSE. Also, the process of assigning value after partial N-R iterations is introduced to solve nonlinearity inside each element. The new process eliminates the need for an explicit function of  $\vec{B}$  as a dependent variable of the voltage difference, which is essential in 2-D TLM FEM. Transmission lines are then used to separate the scattering boxes from the linear network, allowing massive parallelism at the elemental level.

This paper is arranged as follows. Section II introduces the gauged eddy current formulation based on reduced MVP, which is discretized by edge element interpolation functions in Section III. Sections IV and V describe the massively parallelized TLM-FEM scheme of the discretized formulation. In Section VI, the massive parallelism is verified by a GPU implementation, and the comparison between the proposed 3-D-TLM-FEM scheme and Comsol indicates an excellent speedup of over 50 while a good precision (2%). Finally, Section VIII gives the main conclusion of the work.

## II. FEM FORMULATION FOR EDDY CURRENT ANALYSIS

### A. Reduced Magnetic Potential Formulation

Quasi-static Maxwell's equations give restriction of magnetic flux density ( $\vec{B}$ ) in eddy current analysis. Through Maxwell's equations, it is obvious that  $\vec{B}$  jumps at the interface between different materials, which causes inconvenience for numerical computation [14]. To deal with such a problem, the well-known  $\vec{A} - \varphi$  formulation was introduced and  $\vec{A}$ ,  $\varphi$  can be made continuous between materials [14]. The  $\vec{A} - \varphi$  formulation can be simplified to the reduced  $\vec{A}$  formulation (1) when the edge element is used, which allows perpendicular jump of  $\vec{A}$  between different conductivities

$$\nabla \times (v \nabla \times \vec{A}) + \sigma \frac{\partial \vec{A}}{\partial t} = \vec{J}_e \quad (1)$$

where  $v$  is the field-dependent reluctivity,  $\sigma$  is the electrical conductivity, and  $\vec{J}_e$  is the impressed current density.

### B. Gauge for Non-Conducting Region

The system in (1) is uniquely solvable if all solution domains are conductors. However, when solution domain ( $\Omega_{\text{all}}$ ) includes non-conduction region ( $\Omega_n$ ), (1) does not have a unique solution because it loses restriction to the divergence of  $\vec{A}$  in  $\Omega_n$ . To get a unique solution, an innovative gauge [15], [16] can be added by introducing a dummy variable  $\chi$  in  $\Omega_n$ , which results in the following:

$$\begin{cases} \nabla \times (v \nabla \times \vec{A}) + \sigma \frac{\partial \vec{A}}{\partial t} - \nabla \chi = \vec{J}_e \\ \nabla \cdot \vec{A} + \chi = 0. \end{cases} \quad (2)$$

Suppose  $\vec{J}_e$  is divergence-free, by taking divergence to both sides of the first equation in (2), one finds that  $\chi$  satisfies the Laplace equation:

$$\nabla^2 \chi = 0 \mid \Omega_n. \quad (3)$$

By forcing  $\chi$  to be zero at the outer boundary of  $\Omega_n$

$$\chi = 0 \mid \Gamma_n \quad (4)$$

where  $\Gamma_n$  is the boundary of  $\Omega_n$ , (3) becomes a simple boundary value problem and the solution is 0 in  $\Omega_n$ . Now that  $\chi$  is 0, considering second equation of (2),  $\nabla \cdot \vec{A}$  becomes 0. Therefore, Coulomb gauge is applied to  $\vec{A}$  to get a unique solution. For simplicity, a homogeneous Dirichlet boundary is imposed to  $\vec{A}$  in this work

$$\vec{A}^{\parallel} = 0 \mid \Gamma_{\text{all}} \quad (5)$$

where  $\Gamma_{\text{all}}$  is the boundary of  $\Omega_{\text{all}}$ . Equations (2), (4), and (5) form the formulation used for the eddy current analysis in this paper.

## III. FINITE-ELEMENT METHOD AND DISCRETIZED FORMULATION

### A. Magnetic Vector Potential Field

FEM can be used to solve the above-mentioned formulation. It simplifies the problem by representing the unknown field with a limited degree of freedom. The solution domain is divided into many subdomains (elements). Within each element, the unknown field is a linear combination of known pattern fields (interpolation functions fully determined by the coordinate of the element, as shown in Fig. 1). The terms ( $A_1, A_2, A_3, \dots$ ) of this linear combination for every element are the degree of freedom to be solved.

The Galerkin method can be used to find those terms. For the method, the weight function is the same as interpolation function. If one uses vector identities and forces the integration of the product of weight function and residual from (2) to be zero within one element, elemental-discretized formulation is obtained as follows:

$$\int_{S_e} (v \nabla \times \vec{A}) \times \vec{d}s \cdot \vec{N}_i + \int_{V_e} (v \nabla \times \vec{A}) \cdot \nabla \times \vec{N}_i dV + \sigma \frac{\partial}{\partial t} \int_{V_e} \vec{N}_i \cdot \vec{A} dV - \int_{V_e} \vec{N}_i \cdot \nabla \chi dV - \int_{V_e} \vec{N}_i \cdot \vec{J}_e dV = 0 \quad (6)$$

$$\int_{S_e} L_i \vec{A} \vec{d}s + \int_{V_e} \vec{A} \cdot \nabla \vec{L}_i dV - \int_{V_e} L_i \chi dV = 0 \quad (7)$$

where  $S_e$  is the surface boundary of the element and  $i$  is integer from 1 to 6. By substituting two FEM-discretized field expressions at the top of Fig. 1 into (6) and (7) and ignoring the first term in (6) and (7), the matrix-form element-discretized formulation is obtained as

$$\begin{bmatrix} [\widehat{A}A_{ij}]_{6 \times 6} & [\widehat{A}\chi_{ij}]_{6 \times 4} \\ [\widehat{\chi}A_{ij}]_{4 \times 6} & [\widehat{\chi}\chi_{ij}]_{4 \times 4} \end{bmatrix} \begin{bmatrix} [A_j]_{6 \times 1} \\ [\chi_j]_{4 \times 1} \end{bmatrix} + \sigma \frac{\partial}{\partial t} \begin{bmatrix} [\widehat{C}C_{ij}]_{6 \times 6} & [0]_{6 \times 4} \\ [0]_{4 \times 6} & [0]_{4 \times 4} \end{bmatrix} \begin{bmatrix} [A_j]_{6 \times 6} \\ [0]_{4 \times 6} \end{bmatrix} = \begin{bmatrix} [b_i]_{6 \times 6} \\ [0]_{4 \times 6} \end{bmatrix} \quad (8)$$

where

$$\widehat{A}A_{ij} = \int_{V_e} (v \nabla \times \vec{N}_i) \cdot (\nabla \times \vec{N}_j) dV, \quad i, j \in [1, 6] \quad (9)$$

$$\widehat{A}\chi_{ij} = \int_{V_e} \vec{N}_i \cdot \nabla \vec{L}_j dV, \quad i, j \in [1, 6] \quad (10)$$

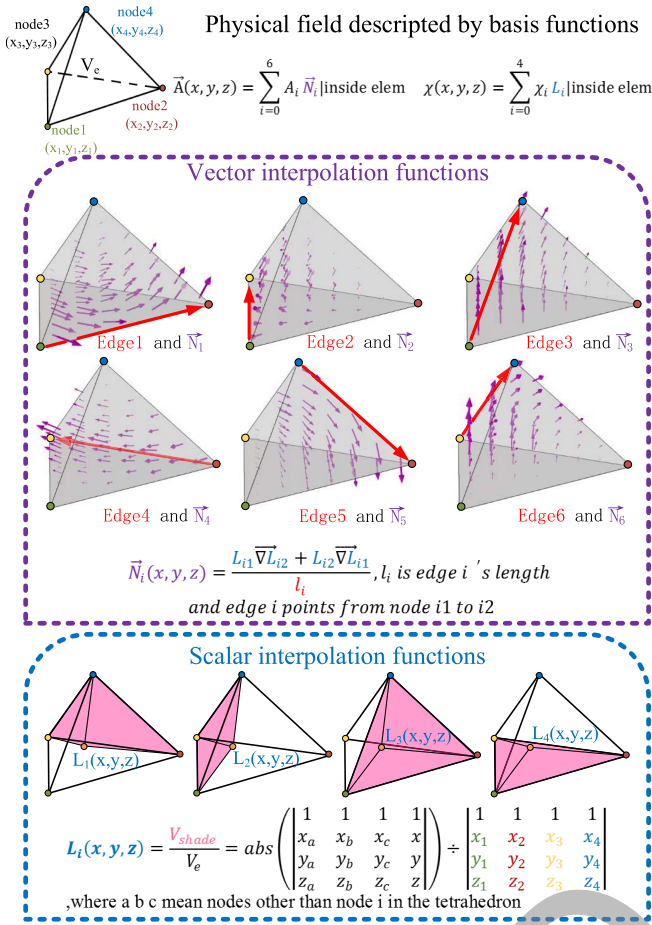


Fig. 1. Scalar nodal and vector edge interpolation functions.

$$\widehat{\chi} A_{ij} = \int_{V_e} \nabla \vec{L}_i \cdot \vec{N}_j dV, \quad i, j \in [1, 6] \quad (11)$$

$$\widehat{\chi} \chi_{ij} = \int_{V_e} L_i L_j dV, \quad i, j \in [1, 6] \quad (12)$$

$$\widehat{C} C_{ij} = \int_{V_e} \vec{N}_i \cdot \vec{N}_j dV, \quad i, j \in [1, 6] \quad (13)$$

$$b_i = \int_{V_e} \vec{N}_i \cdot \vec{J}_e dV, \quad i, j \in [1, 6]. \quad (14)$$

However, (8) only indicates the relationship between edge/node unknowns that belong to the corresponding element. In the assembling phase, the relationship between all unknowns of the solution domain is constructed by adding the contribution of every element to a global matrix. The surface integration in (6) and (7) is not shown in (8). In fact, they are canceled out during the assembling to ensure the continuity of tangential  $v \nabla \times \vec{A}$  and the continuity of perpendicular  $\vec{A}$  at the interface between every element. As a result, the Coulomb gauge is automatically satisfied and every element can have different  $v$  with almost no computational expense. Note that  $A_i$  is not the value of  $x$ ,  $y$ , or  $z$  component of magnetic potential on the corresponding edge  $i$ . In fact,  $A_i$  is the terms of the edge's interpolation function. The value of  $A_i$  is the value of projection of magnetic potential to the edge's direction. According to the definition of edge interpolation

function,  $\vec{B}$  can be expressed by  $A_i$  as

$$\vec{B} = \nabla \times \vec{A}_i = \sum_{i=1}^6 \nabla \times \vec{N}_i = \sum_{i=1}^6 \frac{2 \nabla \vec{L}_{i1} \times \nabla \vec{L}_{i2}}{l_i}. \quad (15)$$

### B. Excitation Current Field

The excitation current field is calculated by solving a 3-D static current conservation problem

$$\begin{cases} \nabla \cdot \vec{J}_e = -\nabla^2 \sigma_{cl} \varphi = 0 | \Omega_{cl} \\ \varphi = 0 | \Gamma_{Gd} \\ \varphi = V(t) | \Gamma_{Vs} \\ \vec{n} \cdot \vec{J}_e = \vec{n} \cdot \nabla \varphi | \Omega_{cl} \end{cases} \quad (16)$$

where  $\Omega_{cl}$  is the coil domain,  $\Gamma_{Gd}$  and  $\Gamma_{Vs}$  are the boundary of coil surface, ground, and excitation voltage.  $V(t)$  is the voltage used to calculate the time-variant excitation current field, and  $\sigma_{cl}$  is the material conductivity of the coil. The coil domain is discretized by tetrahedrons and scalar basis function  $L_i$  shown in Fig. 1. Similarly, (16) is discretized with the Galerkin method. The voltage value at each tetrahedron vertex is then solved. The current inside each element is found by  $\sigma_{cl} \nabla \varphi$ .

## IV. SOLVING THE NONLINEAR SYSTEM

### A. TLM for FEM Analysis

The nonlinear system assembled by (8) is usually solved by N-R or the quasi-Newton method. However, these methods require updating of the global matrix, which slows down the computation. The TLM, in contrast, does not have such a problem.

TLM was already implemented to handle nonlinearity, despite its full-wave physical essence. Although the TLM originates from the Huygens' wave propagation model and was used to solve full-wave problems [4], due to the delaying and isolating functions of the transmission line, TLM was successfully extended to solve nonlinear problems, such as the solution of a nonlinear electrical circuit network [3]. By converting 2-D FEM equations into a nonlinear electrical circuit equivalence, the TLM was further implemented for the 2-D magnetic field analysis [5].

To stay consistent with previous TLM-FEM works, we opted to represent the 3-D FEM equations with a similar nonlinear equivalent circuit model. According to the circuit analysis theory [17], the symmetric matrix system in (8) can be represented by a circuit network. The right-hand side (RHS) corresponds to node injection currents and the unknowns correspond to the circuit node voltage. To the first matrix, the negative value of the matrix element  $ij$  corresponds to the conductance between nodes  $i$  and  $j$ . The sum of the first six elements in row  $i$  corresponds to nonlinear conductance of node  $i$  to the ground, and the sum of the last four elements corresponds to the linear ground conductance of node  $i$ . Similarly, the second matrix can be represented by a linear capacitor network. Since the two admittance matrixes in (8) are added together, the resistive and capacitive networks are, according to the circuit analysis theory, parallel connected to the same circuit topology, which is shown in the next part.



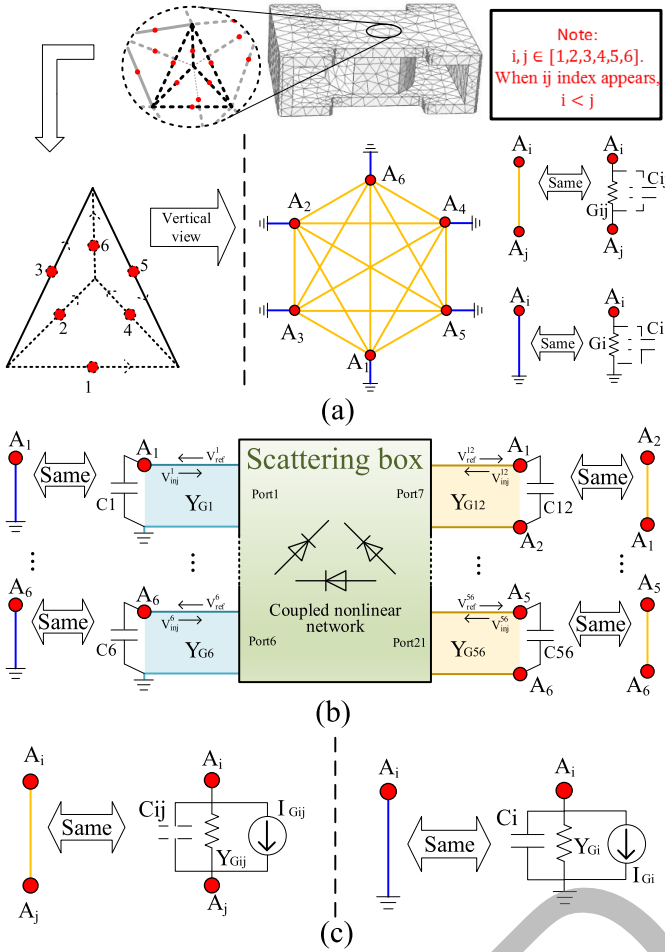


Fig. 2. TLM technique for the nonlinear system in (8) without  $\chi$ . (a) Electrical circuit for element equation without  $\chi$ . (b) Resistors replaced by TL line model in scattering phase. (c) TL lines replaced by the Norton circuit in gathering phase.

Nonlinear resistors in the circuit network are separated from the linear circuit by transmission lines with arbitrary characteristic impedance  $Z_c$ . The linear circuit only sees linear  $Z_c$  and “communicates” with nonlinear resistors by traveling waves on transmission lines. After reflecting between the two line terminals many times, the waves reach steady values. In this way, nonlinearity is replaced by equivalent current sources and the admittance matrix stays the same during each TLM iteration.

It is worth mentioning that  $v$  is constant inside one element, which implies coupling of all nonlinear resistors inside one element. These resistors should be treated as a small subsystem separated by transmission lines for each element. The subsystem is defined as a “scattering box” in this paper. Since these small subsystems (scattering boxes) for each element are separated by transmission lines, it is possible to solve each subsystem independently of others.

### B. Elemental Electrical Circuit and TLM Process

In (8), matrix elements involving  $\chi$  do not include  $v$  and they correspond to linear resistors. For simplicity, they are not shown in the elemental circuit model. Fig. 2(a) shows the

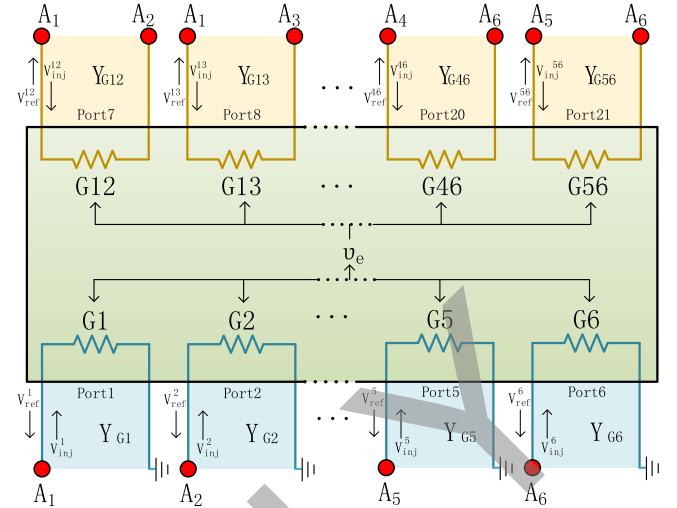


Fig. 3. Inside scattering box: coupled elemental nonlinear network for scattering phase.

nonlinear resistors of the equivalent circuit model of submatrix  $[\bar{A}\bar{A}_{ij}]_{6 \times 6}$ ,  $\sigma[\bar{C}\bar{C}_{ij}]_{6 \times 6}$  in (8). The six red dots are the circuit nodes, and the voltage of the nodes ( $A_i$ ) corresponds to the value of projection of  $\vec{A}$  field to the corresponding edge direction. Yellow lines are the branches between two nodes, and blue lines are the branch between each node and ground. The circuit component of each branch is shown on the right side with the following values:

$$G_{ij} = - \int_{V_e} v_e (\nabla \times \vec{N}_i) \cdot (\nabla \times \vec{N}_j) dV \quad i, j \in [1, 6], i \neq j \quad (17)$$

$$G_i = \sum_{j=1}^6 \int_{V_e} v_e (\nabla \times \vec{N}_i) \cdot (\nabla \times \vec{N}_j) dV, \quad i, j \in [1, 6] \quad (18)$$

$$C_{ij} = -\sigma \widehat{C}\widehat{C}_{ij}, \quad C_i = \sum_{j=1}^6 \widehat{C}\widehat{C}_{ij} \quad i, j \in [1, 6] \quad (19)$$

where  $v_e$  is the unknown reluctivity determined by  $A_i$ .  $v_e$  is to be solved by N-R iteration inside the scattering box.

The TLM process is divided into two major phases: scattering and gathering. In the scattering phase shown in Fig. 2(b), the nonlinear resistors are replaced by transmission lines and a scattering box. The transmission line characteristic conductance is given by

$$Y_{Gij} = - \int_{V_e} v_e^g (\nabla \times \vec{N}_i) \cdot (\nabla \times \vec{N}_j) dV \quad i, j \in [1, 6], i \neq j \quad (20)$$

$$Y_{Gi} = \sum_{j=1}^6 \int_{V_e} v_e^g (\nabla \times \vec{N}_i) \cdot (\nabla \times \vec{N}_j) dV, \quad i, j \in [1, 6] \quad (21)$$

where  $v_e^g$  is the guessed elemental reluctivity value before computation and should be as close as possible to the final solution of  $v_e$ .

In this phase, information of the global circuit is fed to elemental nonlinear resistors through transmission lines. Injection waves affected by nodal voltages ( $A_i$ ) and branch voltages ( $A_i - A_j$ ) enter the scattering box (Fig. 3). Since each

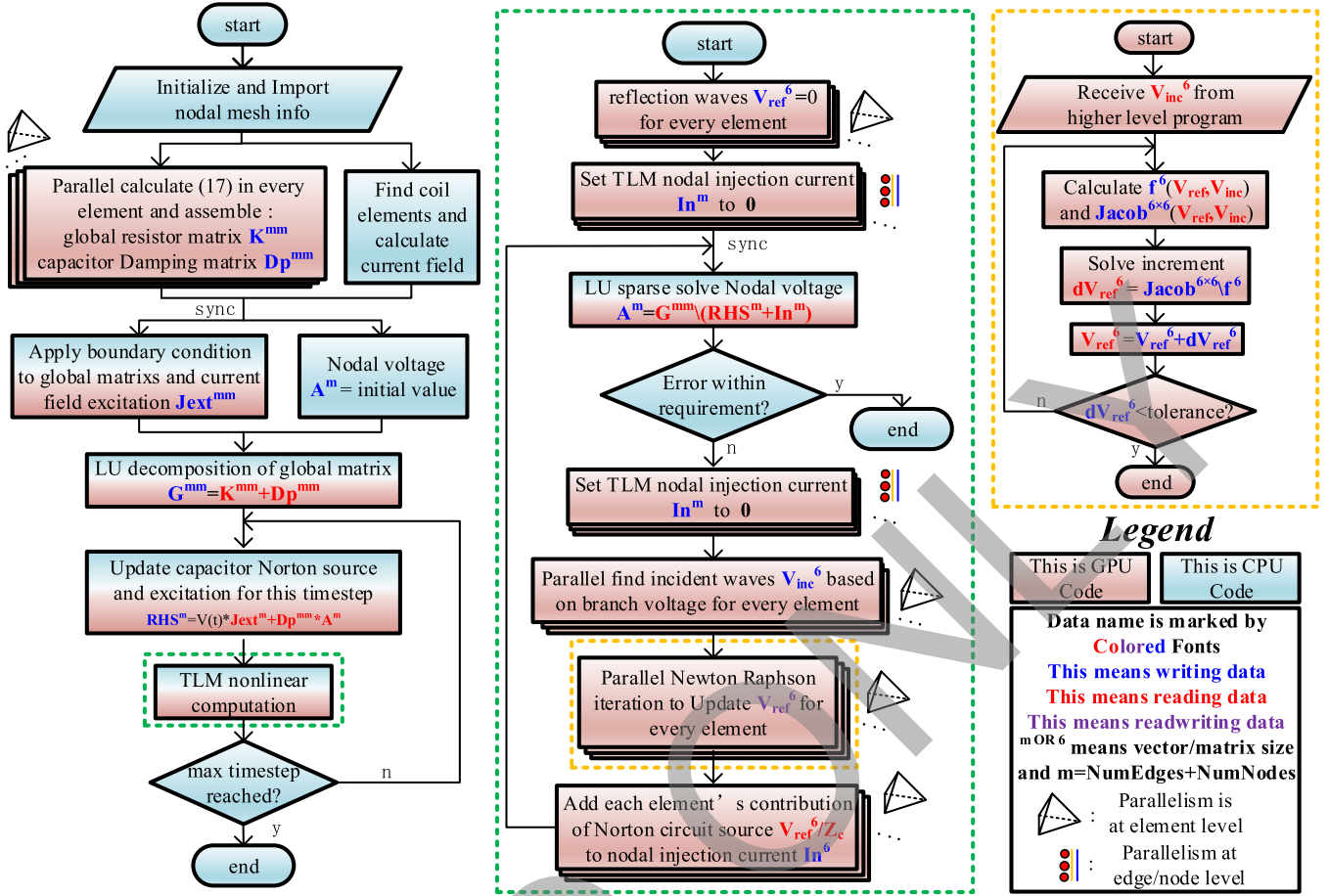


Fig. 4. Detailed massively parallel implementation of 3-D-TLM-FEM method.

scattering box only sees transmission lines and incident waves rather than other scattering boxes, the solution process inside each scattering box is independent of other scattering boxes. Inside each individual scattering box, incident wave at one port is scattered to all ports according to the coupling relation of nonlinear resistors. After scattering, waves leave the box and become reflection waves. For example,  $V_{inj}^1$  enters port 1 and it is distributed and guided toward ports 1–21 based on the relationship between  $G_1$  and other conductance. All incident waves are recombined to form reflection waves. Since  $v_e$  is determined by  $(A_i)$  rather than  $(A_i - A_j)$ , only blue ports participate in the N-R iteration. The nonlinear relationship between  $V_{inj}$  and  $V_{ref}$  is as follows:

$$G_i(V_{inj}^i + V_{ref}^i) - Y_{Gi}(V_{inj}^i - V_{ref}^i) = 0 \quad i \in [1, 6] \quad (22)$$

where  $Y_{Gi}$  are guessed (fixed) but  $G_i$  are the functions of  $V_{inj}^i + V_{ref}^i$ . These functions are determined by (15) and  $B - v/B - H$  curve. To calculate this  $6 \times 6$  system, the N-R iteration is used and the following equation yields the Jacobian matrix:

$$\frac{\partial G_i}{\partial V_{ref}^j} = \frac{\partial G_i}{\partial B^2} \frac{\partial B^2}{\partial V_{ref}^j} \quad i, j \in [1, 6]. \quad (23)$$

After the  $6 \times 6$  N-R iteration,  $v_e$  is found and  $v_e$  is used to calculate the  $G_{ij}$  at ports 7–21. The reflection waves for ports 7–21 can be found by

$$V_{ref}^{ij} = V_{inj}^{ij} \frac{G_{ij} - Y_{Gij}}{G_{ij} + Y_{Gij}} \quad i, j \in [1, 6]. \quad (24)$$

The above-mentioned local N-R solving process does not involve nodal voltage difference ( $A_i - A_j$ ). In other words, the solution process does not need an explicit function of  $B$  as a dependent variable of the voltage difference. Therefore, an extra transformation is unnecessary for (15), which might be extremely difficult.

During the gathering phase [Fig. 2(c)], the reflected waves carrying the information from elemental nonlinear resistors return to the global circuit network. Transmission lines and reflected waves are replaced by the Norton model, and node voltages are then updated for the global circuit network and these voltages are used to generate injection waves for the next iteration. It is worth mentioning that the circuit network in such a phase is a massive network assembled from all elemental circuits in the mesh.

As explained above, each TLM iteration involves many independent parallelizable solutions for scattering boxes during the scattering phase and one solution of the fixed global admittance matrix.

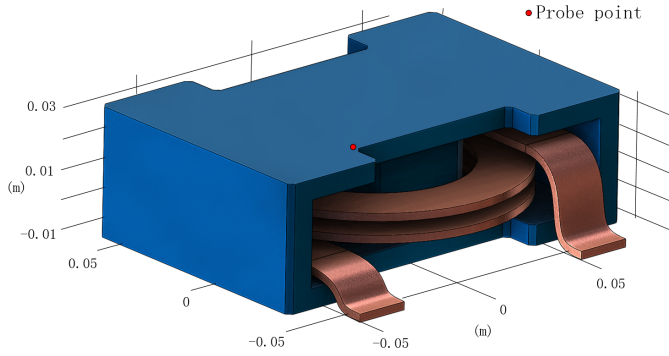


Fig. 5. Dimensions of the studied power inductor.

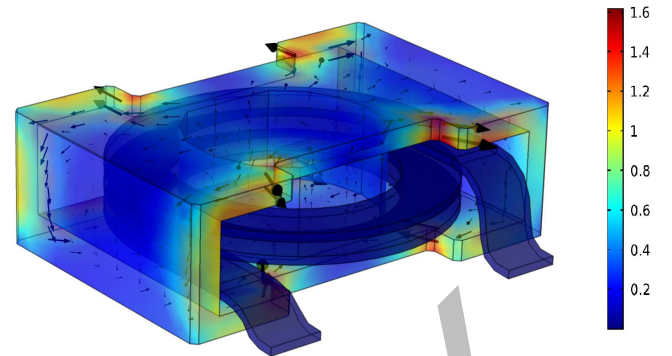


Fig. 6. Magnetic flux density distribution (module and vector) at 0.00667 s.

### C. Time Discretization for Capacitors

As shown in Fig. 2, there are capacitors in parallel with nonlinear resistors. The capacitors involve time derivatives that can be replaced by a resistor and a history current source by different numerical methods (backward Euler rule used in this paper). The admittance and current sources are given by

$$Y_{Cij} = C_{ij}/\Delta t \quad Y_{Ci} = C_i/\Delta t \quad i, j \in [1, 6] \quad (25)$$

$$I_{Cij} = Y_{Cij}(A_i^{(n-1)} - A_j^{(n-1)}) \quad i, j \in [1, 6] \quad (26)$$

$$I_{Ci}^{(n)} = Y_{Ci}A_i^{(n-1)} \quad i, j \in [1, 6] \quad (27)$$

where  $\Delta t$  is the timestep, the upper index ( $n$ ) means the current timestep, and  $(n-1)$  means the previous timestep.

### V. PROGRAM FOR MASSIVELY PARALLEL ARCHITECTURE

As mentioned above, the scale of N-R iteration in the TLM is small ( $6 \times 6$ ). More importantly, the N-R iteration of different elements is independent and thus can be massively parallelized. In addition, TLM iterations do not change the admittance matrix; thus, only one time of admittance matrix inversion is needed for the whole solution process. These properties make it suitable to run the proposed 3-D-TLM edge FEM on massively parallelized architectures such as the GPU. A Cuda C program is developed, and the flowchart is shown in Fig. 4 for GPU implementation.

### VI. CASE STUDY

To demonstrate the precision and efficiency of the 3-D-TLM-FEM method, a power inductor shown in Fig. 5 is studied in comparison with Comsol. The blue iron core has a size of  $0.15 \text{ m} \times 0.1 \text{ m} \times 0.0475 \text{ m}$  and it is surrounded by a coppery solid coil. The test case is implemented on a work station with Intel Xeon E5-2698 v4 CPU and NVIDIA Tesla V100-PCIR-16GB GPU. Material properties and voltage to generate excitation current are given in Table I.

For the given problem definition, computation is carried out based on the following parameters. The relative tolerance for TLM iteration is set to  $10^{-5}$ . Timestep for the case study is set to  $1/1200 \text{ s}$  and the time length is  $0.25 \text{ s}$ .

After post-processing, the magnetic flux density vector field is obtained. The field distribution is shown in Fig. 6 at the time when the maximum value occurs.

TABLE I  
PROBLEM DEFINITION

Material information		
Domain	$\sigma$	B-H curve
Air	0	$H = v_0 B$
Coil	$10^7 \text{ S/m}$	$H = v_0 B$
Core	$10^4 \text{ S/m}$	$H = \begin{cases} v_0 B / 2000 &  B  < 1.3 \\ v_0 [ B  + ( B  - 1.3)^8] \times  B  / 2000 B  B  \geq 1.3 \end{cases}$
$v(t)$		$0.45 \sin(120\pi t) \text{ V}$

TABLE II  
EXECUTION TIME AND SPEEDUP OF 3-D-TLM

Mesh ID	Number of			Execution time		Speed up
	Elements	DOFs	Nonlinear DOFs	Comsol <sup>TM</sup>	TLM	
1	7870	9061	1594	191.1s	3.65s	52.1
2	12533	17423	2604	401.9s	9.69s	41.5
3	20249	27996	4165	736s	24.7s	29.8
4	39258	54559	6109	2636s	162.1s	16.3
5	66706	91664	9236	4750s	342.8s	13.9
6	143964	196973	14283	11464s	912.7s	12.6

A comparison is made with Comsol to verify the precision and efficiency of the proposed algorithm, and the result shows good accuracy with average relative error less than 2% over all space and time span. Figs. 7 and 8 display the field results obtained from both Comsol and the TLM-FEM scheme.

Meanwhile, a significant speedup can be seen for different mesh sizes (Table II). It is not surprising to witness an excellent speedup since the proposed 3-D-TLM-FEM scheme, in nature, has excellent parallelism and the V100 GPU has over 5000 cores. However, it is still worth mentioning that the speedup depends on TLM iterations needed per timestep and may vary for different  $B-H$  curves and excitation amplitudes. Also, different matrix solution algorithms affect the solution time.

### VII. DISCUSSION

As shown in Table II, the speedup decreases as mesh complexity increases. This leads to the following questions and discussion: is the performance hindered by TLM algorithm itself

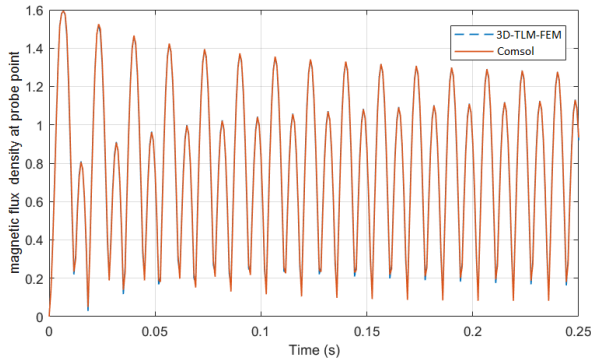


Fig. 7. Comparison of magnetic flux density module at probe point versus time.

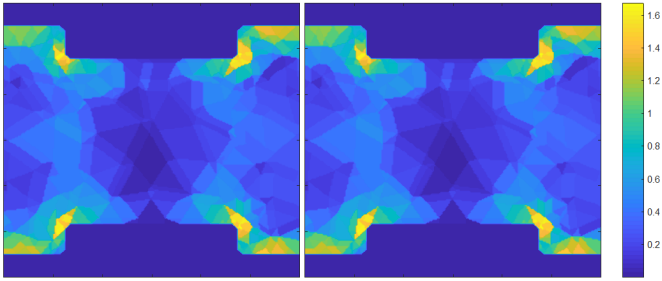


Fig. 8. Comparison of the magnetic flux density module at  $Z = 0.33$  and  $t = 0.00667$  (left from the proposed 3-D-TLM-FEM scheme and right from Comsol).

for the case study? and will the 3-D-TLM-FEM method remain efficient for a larger mesh size? More analysis/discussion is given in the following.

For the time analysis, the following approximation is used:

$$T_{\text{TLM}} \approx N_{\text{TLM}} \times (t_{\text{NR}} + t_{\text{LU}}) \quad (28)$$

where  $T_{\text{TLM}}$  is the total TLM execution time,  $N_{\text{TLM}}$  is the total TLM iterations needed for entire computation,  $t_{\text{NR}}$  is the N-R iteration time at one single iteration, and  $t_{\text{LU}}$  is the (LU) triangle matrix back substitution time.  $t_{\text{NR}}$  and  $t_{\text{LU}}$  stay almost unvaried because LU matrix/routine and scale of N-R iteration stay the same during the entire TLM iteration.

To find the reason for the speedup drop,  $N_{\text{TLM}}$  for the case study is investigated. By comparing information given in Fig. 9 and Table II, it is obvious that the increase in  $T_{\text{TLM}}$  versus DOF is much higher than that of  $N_{\text{TLM}}$ . According to (28), a nonlinear growth of  $t_{\text{NR}} + t_{\text{LU}}$  is seen, and the timeline generated by Cuda Visual Profiler indicates that  $t_{\text{NR}}$  stays almost the same for all meshes. Therefore, the nonlinear increment in  $t_{\text{LU}}$  hinders the computation for a larger mesh, possibly due to the insufficient number of cores to fully parallelize triangle matrix back substitution. It is the LU solving process rather than TLM-FEM scheme that causes the majority of speed reduction for the case study. With better algorithm and more powerful GPU, the LU triangular solution time is likely to approach ideal  $O(\text{DOF})$ , and the performance of the case study will increase dramatically.

$N_{\text{TLM}}$  is not the “black sheep” in the case study, but it is possible that  $N_{\text{TLM}}$  grows impractically large for bigger

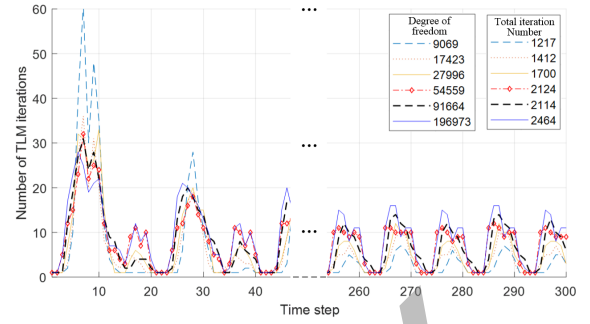


Fig. 9. TLM iteration number at each timestep for the case study.

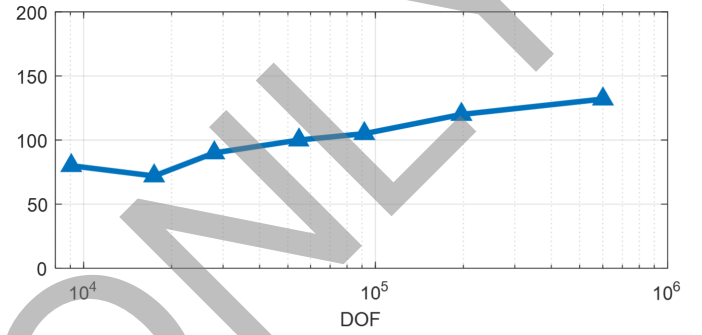


Fig. 10. TLM iterations required to reach tolerance of  $10^{-5}$  in static scenario.

problems? To eliminate the interference of time stepping, another comparison of  $N_{\text{TLM}}$  between different mesh sizes is carried out for a static case where excitation current drives the inductor into deeper saturation, and results in Fig. 10 show that TLM iteration number is not sensitive to mesh DOF. Even an initial descending trend of  $N_{\text{TLM}}$  versus DOF is witnessed. This coincides with the intuition that TLM iteration is much more related to matching than the network complexity.

Based on the above-mentioned discussion, the proposed 3-D-TLM-FEM scheme is promising, especially for large mesh since its algorithm complexity comes close to  $O(\text{DOF})$ . Furthermore, for long-time-span transient simulations, the 3-D-TLM-FEM will gain more advantage— $N_{\text{TLM}}$  reduces at later timesteps and the difference between  $N_{\text{TLM}}$  narrows for different mesh sizes, as shown in Fig. 9. The increment in TLM iterations is much smaller than that of mesh DOF. With a good convergence rate and an iterative nature, the 3-D-FEM-TLM scheme may look similar to the quasi-Newton method; however, they are essentially two different methods.

## VIII. CONCLUSION

In this paper, the TLM was successfully extended for the 3-D nonlinear electromagnetic field analysis using edge vector FEs. The challenges were discussed and conquered, and it turned out that the good features of the TLM such as constant admittance matrix and massive parallelism can also benefit the computation of 3-D nonlinear EVEs in practice.

More specifically, for the first time, the transmission line modeling was applied to the edge-element-discretized formulation to calculate EM field with nonlinearity. To solve



the discretized formulation with excellent parallelism, nonlinearity and matrix rank were properly handled. The concept of scattering box was introduced to model elemental local nonlinearity for the TL-circuit system. Transmission lines in the system successfully decoupled local nonlinearity from the global linear network so that computation can be parallelized at the 3-D element level. In addition, the proper gauge was applied to edge FE formulation, which results in a full ranked matrix system to allow LU factorization of the global linear network, making it possible to carry out matrix factorization only once during the whole computation process.

Due to the above-mentioned property, the proposed 3-D-TLM-FEM is perfectly suitable for high-performance parallel computation, and the comparison between the TLM implementation and Comsol shows excellent speed up (over 50) while maintaining a good precision for the power inductor case study.

#### ACKNOWLEDGMENT

This work was supported by the Natural Science and Engineering Research Council of Canada (NSERC).

#### REFERENCES

- [1] NVIDIA TESLA V100 GPU ACCELERATOR. Accessed: Jan. 20, 2019. [Online]. Available: <https://images.nvidia.com/content/technologies/volta/pdf/tesla-volta-v100-datasheet-letter-fnl-web.pdf>
- [2] P. Liu and V. Dinavahi, "Matrix-free nodal domain decomposition with relaxation for massively parallel finite-element computation of EM apparatus," *IEEE Trans. Magn.*, vol. 54, no. 9, Sep. 2018, Art. no. 7402507.
- [3] P. B. Johns and M. O'Brien, "Use of the transmission-line modelling (t.l.m.) method to solve non-linear lumped networks," *Radio Electron. Eng.*, vol. 50, no. 1.2, pp. 59–70, Jan./Feb. 1980.
- [4] C. Christopoulos, *The Transmission-Line Modeling Method: TLM*, vol. 221. New York, NY, USA: IEEE Press, 1995.
- [5] J. Lobry, J. Trecat, and C. Broche, "The transmission line modeling (TLM) method as a new iterative technique in nonlinear 2-D magnetostatics," *IEEE Trans. Magn.*, vol. 32, no. 2, pp. 559–566, Mar. 1996.
- [6] O. Deblecker, J. Lobry, and C. Broche, "Novel algorithm based on transmission-line modeling in the finite-element method for nonlinear quasi-static field analysis," *IEEE Trans. Magn.*, vol. 39, no. 1, pp. 529–538, Jan. 2003.
- [7] P. Liu and V. Dinavahi, "Real-time finite-element simulation of electromagnetic transients of transformer on FPGA," *IEEE Trans. Power Del.*, vol. 33, no. 4, pp. 1991–2001, Aug. 2018.
- [8] P. Liu, J. Li, and V. Dinavahi, "Matrix-free nonlinear finite-element solver using transmission-line modeling on GPU," *IEEE Trans. Magn.*, vol. 55, no. 7, Jul. 2019, Art. no. 7401605.
- [9] J. P. A. Bastos and N. Sadowski, *Magnetic Materials and 3D Finite Element Modeling*. Boca Raton, FL, USA: CRC Press, 2013.
- [10] J. Jin, *The Finite Element Method in Electromagnetics*. Hoboken, NJ, USA: Wiley, 2015.
- [11] M. L. Barton and Z. J. Cendes, "New vector finite elements for three-dimensional magnetic field computation," *J. Appl. Phys.*, vol. 61, no. 8, pp. 3919–3921, 1987.
- [12] A. Demenko, J. K. Sykulski, and R. Wojciechowski, "On the equivalence of finite element and finite integration formulations," *IEEE Trans. Magn.*, vol. 46, no. 8, pp. 3169–3172, Aug. 2010.
- [13] A. Demenko and J. K. Sykulski, "Geometric formulation of edge and nodal finite element equations in electromagnetics," *COMPEL-Int. J. Comput. Math. Elect. Electron. Eng.*, vol. 31, no. 5, pp. 1347–1357, 2012.
- [14] R. Albanese and G. Rubinacci, "Finite element methods for the solution of 3D eddy current problems," *Adv. Imag. Electron Phys.*, vol. 102, pp. 1–86, 1997.
- [15] Y. Zhao and W. N. Fu, "A novel formulation with Coulomb Gauge for 3-D magnetostatic problems using edge elements," *IEEE Trans. Magn.*, vol. 53, no. 6, Jun. 2017, Art. no. 9100104.
- [16] Y. Zhao and W. N. Fu, "A novel Coulomb-gauged magnetic vector potential formulation for 3-D eddy-current field analysis using edge elements," *IEEE Trans. Magn.*, vol. 53, no. 6, Jun. 2017, Art. no. 9400704.
- [17] J. A. Edminister, *Schaum's Theory and Problems of Electric Circuits*. New York, NY, USA: McGraw-Hill, 1994.

**Jiacong Li** (S'18) received the B.Sc. degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2015. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada.

His current research interests include computational electromagnetics, finite-element analysis, and parallel and distributed processing.

**Peng Liu** (S'15) received the B.Sc. and M.Eng. degrees in electrical engineering from the Harbin Institute of Technology, Harbin, China, in 2013 and 2015, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada.

His current research interests include computational electromagnetics, finite-element analysis, and parallel and distributed processing.

**Venkata Dinavahi** (S'94–M'00–SM'08) received the Ph.D. degree from the University of Toronto, Toronto, ON, Canada, in 2000.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada. His current research interests include real-time simulation of power systems, large-scale system simulation, and parallel and distributed computing.