A Multilevel Newton Method for Mixed-Energy Domain Simulation of MEMS

N. R. Aluru and Jacob White, *Member, IEEE*

Abstract— An efficient black-box algorithm for self-consistent analysis of three-dimensional (3-D) microelectromechanical systems (MEMS) is described. The algorithm is matrix-free based and employs a multilevel Newton technique to solve the coupled electromechanical equations. The new approach is shown to converge very rapidly and is much faster than relaxation algorithm for tightly coupled cases. While this paper focuses on coupled electromechanical analysis, the proposed algorithm can be extended to include several coupled domains typically encountered in MEMS. [355]

Index Terms—MEMS modeling, mixed-energy domain simulation, multilevel Newton method, relaxation technique.

I. INTRODUCTION

ALTHOUGH there are many microelectromechanical system (MEMS) designs that use piezoelectric, thermal, pneumatic, and magnetic actuation, the most popular approach in present day microsensor and microactuator designs is to use electrostatic forces to move micromachined parts. Designers of such microelectromechanical devices need efficient, robust, and easily used computer simulation tools to investigate design alternatives. Since most of the structures of interest are geometrically complicated, electromechanically coupled, and are inherently three-dimensional (3-D), microelectromechanical CAD (MEMCAD) tool developers have been focused on improving the usability, efficiency, and robustness of coupled 3-D electromechanical analysis [8], [14]. In particular, analysis of 3-D electromechanical systems involves two coupled domains: elastomechanics and electrostatics, which have each been studied very extensively in the literature (see, e.g., [4] for elastomechanics and [10] for electrostatics), and commercial simulation tools are available for each domain. This paper proposes a new approach to construct a 3-D coupled electromechanical simulation tool that is convergent, accurate, and stable and utilizes already existing and commercially available simulation tools for each individual domain. The idea of a construction of a MEMCAD tool from existing black boxes is important for very good reasons—many applications in MEMS involve more than two coupled energy domains, and often it is necessary to investigate more than one actuation mechanism or study a different actuation mechanism by replacing an existing one. This flexibility can be achieved by constructing a black-boxes-based simulation tool that is convergent. Even though this paper reviews an approach for coupled electromechanical analysis, the idea can be extended to include several coupled energy domains.

The numerical techniques employed for coupled electromechanical analysis have so far been based on relaxation [5], [7], a form of surface-Newton method [3], [15], and a tightly coupled Newton method [1], [2]. In particular, finite-element-based elastostatic analysis and accelerated boundary-element-based electrostatic analysis have been combined using algorithms based on relaxation, a form of surface-Newton method, and a tightly coupled Newton method. The relaxation technique is easy to program and can be trivially extended to include several coupled domains. However, the relaxation approach fails to converge for strong coupling between energy domains. As a result, the relaxation technique is limited in its applicability for coupled-domain simulation. The surface-Newton technique is also a black-box approach, but requires more effort to program than the relaxation technique. The key idea in the surface-Newton approach is to reduce the dimensionality of the coupled problem from 3-D to two-dimensional (2-D) and to employ a Newton technique to determine the surface node positions. In the case of electromechanical systems, once the surface node positions are known, the surface charges and the interior node positions can be determined by decoupled electrostatic and mechanical analysis, respectively. This approach requires modifications when the coupling between the energy domains is not just through the surface, and, hence, is not a general approach for multiple energy-domain simulation. Most importantly, the surface-Newton approach is matrix-free based and is extremely sensitive to the choice of the matrix-free parameter. A tightly coupled Newton method can be very quick and efficient when compared to the relaxation and surface-Newton approaches for coupled electromechanical analysis. A tightly coupled Newton method, however, is not black-box based and, hence, cannot be extended very trivially to include several coupled domains. In this paper, we present a new multilevel Newton approach for coupled-domain simulation of MEMS. The new technique is black-box based and is shown to be very efficient, convergent even for strong coupling between domains, and can be trivially extended to include several coupled domains.

The rest of the paper is organized as follows. Section II discusses the self-consistent electromechanical analysis, and the black-box-based relaxation technique for electromechani-

N. R. Aluru is with the Beckman Institute and Department of General Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA.

J. White is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA.
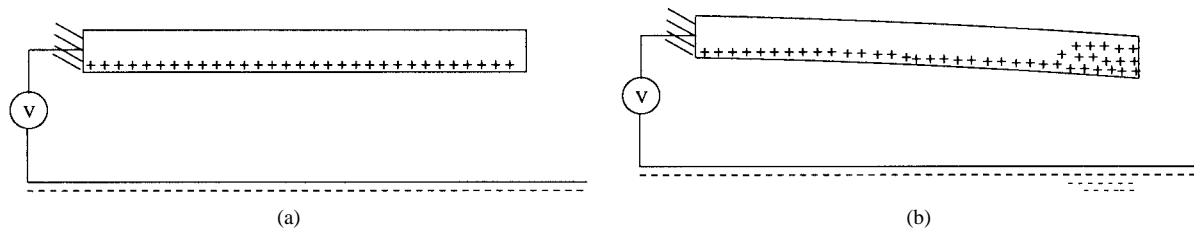
Fig. 1.   Illustration of electromechanical coupling problem through a 2-D beam over a ground plane example: (a) applied voltage causes a charge distribution and (b) the deformed structure with charge redistribution.

cal analysis is described in Section III. The multilevel Newton scheme is introduced in Section IV, and numerical results are presented in Section V comparing the relaxation and multilevel Newton schemes. Finally, conclusions are given in Section VI.

II. SELF-CONSISTENT ELECTROMECHANICAL ANALYSIS

Electromechanical systems typically involve a mechanical structure which undergoes deformation when subjected to electrostatic actuation. The electromechanical coupling is illustrated through a beam over a ground plane example shown in Fig. 1. When a voltage is applied between the beam and the ground plane, a charge distribution is induced on the surface of the beam. This charge distribution causes the beam to deflect and the charge distribution on the beam surface changes because of the beam deflection. An equilibrium state is obtained when the forces due to the beam deflection and the surface charges balance each other. The beam deflection and the electrostatic charges can be obtained by the solution of coupled problems involving mechanical or elastostatic analysis and electrical or electrostatic analysis.

The elastic deformation of the structure can be predicted by studying nonlinear elastostatics (or elastodynamics, as the case may be). A commercial simulator such as ABAQUS[1] is typically employed for the finite-element solution of the nonlinear elastostatic equations. Mathematically, the solution of the elastostatic equations can be represented as

$$u = R_M(P(q)) \tag{1}$$

where $R_M(P(q))$ is a black-box elastostatic solve to compute the structural displacements u, given the pressure P, which is a function of the surface charges q. The details of a finite-element formulation to solve (1) are presented in [2]. Note that the elastostatic equations are nonlinear and typically a Newton method is employed to compute the displacement u.

Given the potentials on the conductors, the charges can be predicted by an electrostatic analysis. Fast simulators, such as FASTCAP [10] can be employed to accurately compute the surface charge distribution. FASTCAP employs a boundary integral formulation with multipole [9] or precorrected fast Fourier transform acceleration techniques [12]. Mathematically, the solution of the electrostatic equations can be represented as

$$q = R_E(u, V) \tag{2}$$

[1] Hibbit, Karlsson, and Sorenson, Inc., Providence, RI.

where $R_E(u, V)$ denotes a black-box electrostatic solve to compute the surface charges q, given the conductor geometry u, and the applied potentials V. The dependence of the electrostatic solve on u is explained by the change in the conductor geometry as it undergoes deformation due to electrostatic forces. The voltages are held fixed during the coupled electromechanical analysis and in subsequent discussion, the dependence of the electrostatic residual on V will be dropped, it being understood that voltages are specified on the conductors for electrostatic analysis. The details of a boundary integral formulation to solve (2) are presented in [10].

Coupled electromechanical analysis requires the self-consistent solution of (1) and (2). With the availability of black-box solvers such as ABAQUS for mechanical analysis and FASTCAP for electrical analysis, coupled electromechanical analysis has been implemented with the black-box algorithms discussed in the next two sections.

III. RELAXATION TECHNIQUE

A simplest black-box approach for coupled electromechanical analysis is the Gauss–Seidel relaxation algorithm [5]. In this approach, the data is passed back and forth between black-box electrostatic and elastostatic analysis programs until a converged solution is obtained. The relaxation procedure is summarized as given below

Algorithm 1: Relaxation procedure for coupled electromechanical analysis.
$\quad k = 1, u^k = 0.$
Repeat
$\quad$ Compute $q^k = R_E(u^k)$.
$\quad$ Compute $u^{(k+1)} = R_M(P(q^k))$.
$\quad k = k + 1;$
until $\|u^k - u^{k+1}\| < \epsilon$ and $\|q^k - q^{k-1}\| < \epsilon$.

As is evident from the above procedure, black-box approaches based on relaxation can be implemented very quickly for coupled electromechanics and for other coupled domain analysis of MEMS. However, as will be shown in the results section, the relaxation algorithm fails to converge for strong coupling between electrical and mechanical domains.

IV. MULTILEVEL NEWTON ALGORITHM

In this section, we present a new multilevel Newton technique for black-box analysis of coupled electromechanical equations. The proposed multilevel Newton technique is
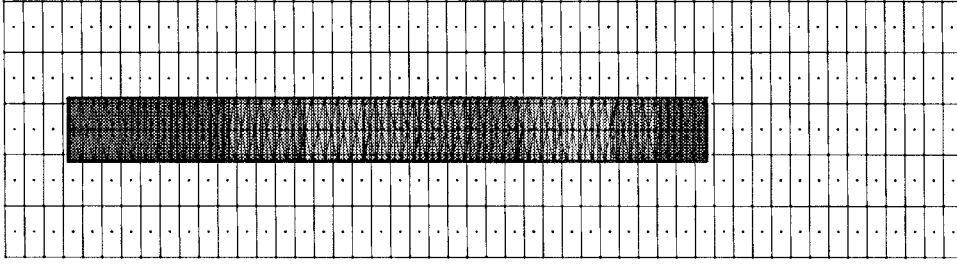
Fig. 2. Top view of a beam over a ground plane example.

matrix-free based, converges very rapidly, and does not suffer from the drawbacks of the surface-Newton or relaxation algorithms. In the multilevel Newton technique, the coupled electromechanical equations are solved by employing a nested Newton–Raphson method. The outer-Newton iteration solves the following residual equation:

$$R(u, q) = \left\{ \begin{matrix} q - R_E(u) \\ u - R_M(q) \end{matrix} \right\}. \qquad (3)$$

In (3), $R_E(u)$ is the charge on the conductors for the conductor geometry defined by u, $R_M(q)$ is the structural displacement because of the electrostatic forces generated by the charge q, and if both $q - R_E(u)$ and $u - R_M(q)$ are zero or approximately zero, then the charge q and the displacement u are a self-consistent solution to the electromechanical system. Note that because the residual R is defined in a certain way it can be computed using black boxes. The Jacobian for the residual given in (3) is given as

$$J(u, q) = \begin{bmatrix} I & -\partial R_E / \partial u \\ -\partial R_M / \partial q & I \end{bmatrix}. \qquad (4)$$

With the definition of the residual and the Jacobian defined through (3) and (4), respectively, the multilevel Newton technique for solving the coupled electromechanical equations can be summarized as follows:

Algorithm 2: Multilevel Newton technique for coupled electromechanical analysis.
  $k = 1, u^k = 0, q^k = 0$.
  Repeat
    Solve

$$J(u^k, q^k) = \left\{ \begin{matrix} \delta_q \\ \delta_u \end{matrix} \right\} = -R(u^k, q^k) \qquad \text{for } \delta_q, \delta_u.$$

    Set $u^{k+1} = u^k + \delta_u$.
    Set $q^{k+1} = q^k + \delta_q$.
    $k = k + 1$;
  until $\|u^k - u^{k+1}\| < \epsilon$ and $\|q^k - q^{k+1}\| < \epsilon$.

The linear system of equations in algorithm 2

$$J \left\{ \begin{matrix} \delta_q \\ \delta_u \end{matrix} \right\} = -R$$

will be solved by employing an iterative solver, such as a generalized conjugate residual (GCR) [6]. The GCR algorithm
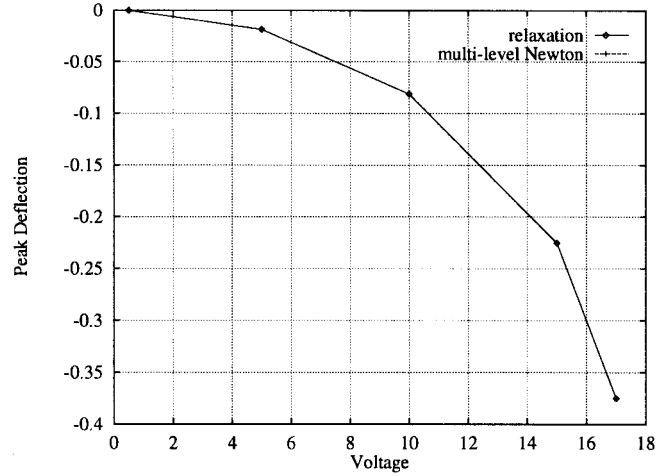

Fig. 3. Comparison of peak deflections from relaxation and coupled algorithms for a beam over a ground plane structure. Note that the two curves overlap.

for the solution of a linear system of equations is summarized below

Algorithm 3: GCR algorithm for solving $Ax = b$
  Make an initial guess to the solution, x^0.
  Set $p^0 = r^0 = b - Ax^0$.
  Set $k = 0$.
  do {
    if $\|r\| < tol$, return x^k as the solution.
    else {

$$\alpha = \frac{\langle Ap^k, r^k \rangle}{\langle Ap^k, Ap^k \rangle}.$$
$$x^{k+1} = x^k + \alpha p^k.$$
$$r^{k+1} = r^k - \alpha Ap^k.$$
$$p^{k+1} = r^{k+1} + \sum_{j=0}^{k} \beta_k^j p^j.$$

      where $\{\beta_k^j\}$ are chosen so that
      $\langle Ap^{k+1}, Ap^j \rangle = 0$ for $0 \leq j \leq k$
      Set $k = k + 1$.
    }
  }

The key step in the GCR algorithm is the computation of the matrix–vector product Ap. The matrix–vector product required in the iterative solver can be computed using finite-difference approximation, i.e.,

$$\frac{\partial R}{\partial u} * r \approx \frac{R(u + \theta * r) - R(u)}{\theta} \qquad (5)$$
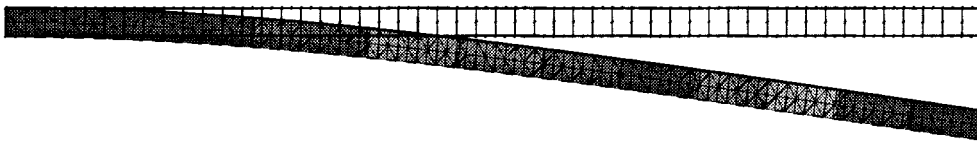
Fig. 4. Deflection of the beam (not to scale) for an applied bias of 17.23 V.

TABLE I
COMPARISON OF RELAXATION AND MULTILEVEL NEWTON
ALGORITHMS FOR NUMBER OF ITERATIONS AND CPU(S)
FOR A THICK BEAM AND GROUND PLANE EXAMPLE

Bias	# Iterations		CPU(s)	
	Relaxation	Multlilevel Newton	Relaxation	Multlilevel Newton
2.0	4	2	283.5	698.7
4.0	5	3	381.0	967.0
6.0	6	3	507.7	1244.9
8.0	7	3	608.4	1079.6
10.0	8	3	710.2	1086.8
12.0	10	3	909.5	1086.7
14.0	13	4	1244.4	1530.7
16.0	20	4	2015.8	1499.0
17.0	41	5	4248.1	1957.0
17.20	94	5	9713.83	2145.7
17.23	200	7	20910.5	2823.5

(a)

(b)

Fig. 5. Convergence of relaxation and multilevel Newton algorithms for a beam and ground plane structure: (a) applied bias is 17.20 V and (b) applied bias is 17.23 V.

Fig. 6. Convergence of the GCR algorithm for coefficient $a = 0.01$ and coefficient $a = 0.255$.

where the matrix-free parameter θ is given as

$$\theta = \operatorname{sign}(u*r) * \min\left(1, \frac{a\|u\|}{\|r\|}, \frac{b\|R(u)\|}{\|r\|}\right)$$
$$a \in (0.01, 0.5) \quad b \in (0.1, 1).$$

For the definition of R given in Algorithm 2, the matrix–vector product can be computed as

$$J\begin{Bmatrix} \delta_q \\ \delta_u \end{Bmatrix} \approx \begin{Bmatrix} \delta_q - \dfrac{1}{\theta}\left(R_E(u+\theta\delta_u) - R_E(u)\right) \\ \delta_u - \dfrac{1}{\theta}\left(R_M(q+\theta\delta_q) - R_M(q)\right) \end{Bmatrix} \quad (6)$$

where θ is a matrix-free parameter which is small and different from zero, δ_q is an increment in the charge vector, and δ_u is an increment in the displacement vector. Both δ_q and δ_u are generated by the GCR algorithm. The choice of θ given above is identical to the one proposed for the surface-Newton method [15]. In the context of the surface-Newton method, the limits on a and b have been obtained through numerical experiments. Noting that the black-box elastostatic and electrostatic solves $R_M(q)$ and $R_E(u)$ can be performed outside the GCR solver, an algorithm for computing the matrix–vector product given in (6) is as follows:
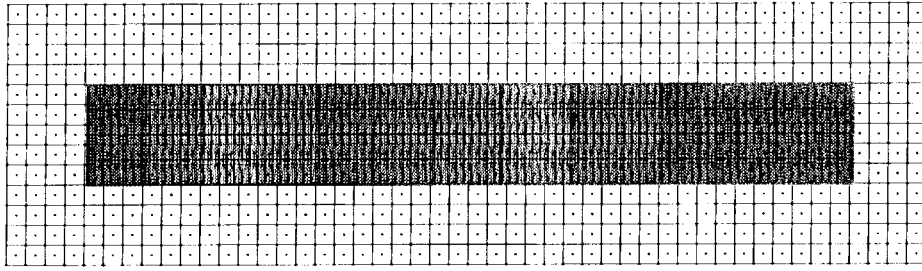
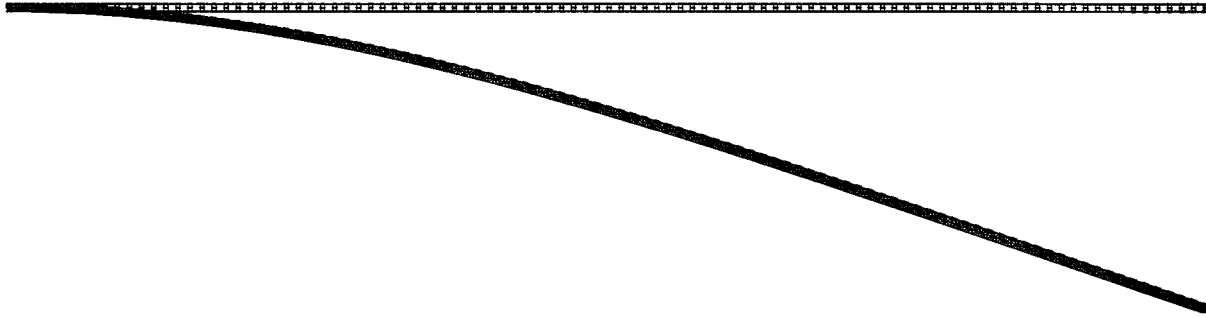Fig. 7. Top view of a beam over a ground plane example.

Fig. 8. Deflection of the beam (not to scale) for an applied bias of 2.38 V.

Algorithm 4: Computation of the matrix–vector product for each GCR iteration.

Assume δ_u, δ_q, and θ are given,
and $u_1 = R_M(q)$, $q_1 = R_E(u)$ are computed outside GCR.
Compute $q_2 = R_E(u + \theta\delta_u)$ by a black box electrostatic solve.
Compute $u_2 = R_M(q + \theta\delta_q)$ by a black box electrostatic solve.
Compute

$$J\begin{Bmatrix} \delta_q \\ \delta_u \end{Bmatrix} = \begin{Bmatrix} \delta_q - \dfrac{1}{\theta}(q_2 - q_1) \\ \delta_u - \dfrac{1}{\theta}(u_2 - u_1) \end{Bmatrix}.$$

As shown in algorithm 4, each iteration of the GCR solver requires two black-box solves. The first black-box solve $R_E(u + \theta\delta_u)$ is to compute the charges on the perturbed geometry and the second black-box solve $R_M(q + \theta\delta_q)$ is to compute the displacements given perturbed electrostatic forces. Since the elastostatic equations are nonlinear, a Newton technique is employed to compute the displacements. The inner Newton loop within each GCR iteration and the outer Newton loop to solve the coupled electromechanical equations makes this approach a multilevel Newton technique.

The coefficients a and b in the matrix-free parameter definition can be picked precisely for the multilevel Newton technique. Specifically, a choice of $a = 0.01$ and $b = 0.1$ are employed for the multilevel Newton technique. While this choice of θ exhibited rapid convergence of the GCR algorithm for all the electromechanical examples considered in this paper, we show in the results section that the multilevel Newton technique can be sensitive to the choice of the parameter a employed.

TABLE II
COMPARISON OF RELAXATION AND MULTILEVEL NEWTON ALGORITHMS FOR
NUMBER OF ITERATIONS AND CPU(S) FOR A BEAM AND GROUND PLANE EXAMPLE

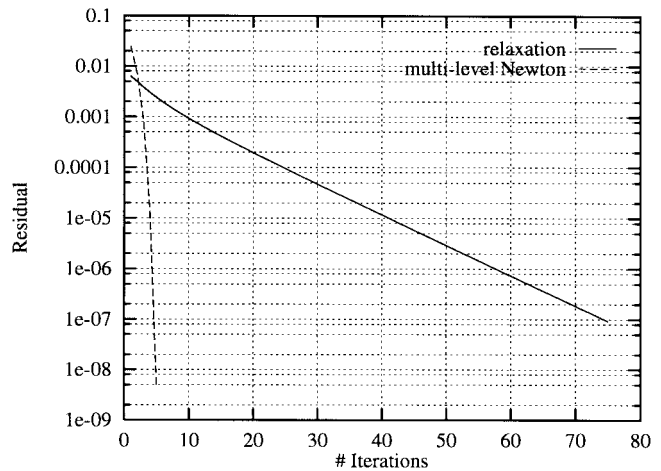| Bias	# Iterations		CPU(s)	
	Relaxation	Multilevel Newton	Relaxation	Multilevel Newton
1.0	6	3	3511.4	5466.7
1.5	8	3	4753.5	5864.4
2.0	13	4	7693.5	7612.8
2.25	20	4	11 756.6	7797.9
2.35	36	5	20 821.9	10 046.8
2.38	75	5	42 749.0	9958.6

Fig. 9. Convergence of relaxation and multilevel Newton algorithms just before pull-in for the beam and ground-plane example.

Remarks:

1) An approach that would eliminate the nonlinear elastostatic and electrostatic solves in each GCR iteration can be developed by defining the residual R in algorithm 2 to be equal to the elastostatic and electrostatic residuals
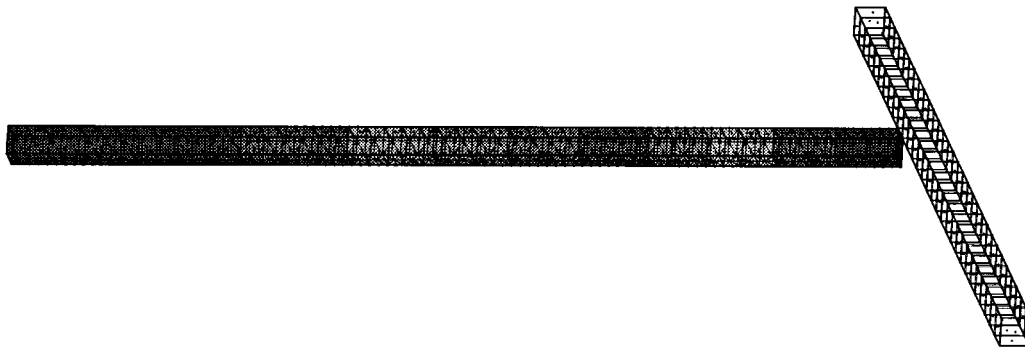
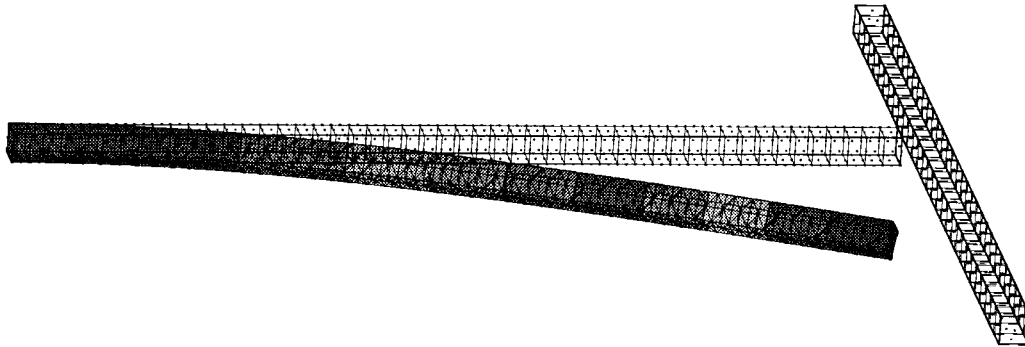Fig. 10. Two silicon bars at a potential difference of 0 V.

Fig. 11. Deflection of the bar (not to scale) for an applied bias of 2000 V.

(see [2]). This is, however, not a black-box approach and one needs to develop in-house code in order to employ this approach.

2) The definition of the residual given in (3) considers the out-of-balance displacements and charges. A multilevel Newton method can also be implemented for out-of-balance forces as the residual. However, care should be exercised as multilevel Newton methods can be very sensitive to the choice of the matrix-free parameter if the residual is not scaled properly.

3) The multilevel Newton technique is guaranteed to converge as long as the initial conditions are not very far off from the final solution. This is the same difficulty with the Newton technique [11]. In order to compute displacements for high applied voltages, zero initial conditions may not converge, in which case voltage stepping can be applied.

4) The cost per iteration of a multilevel Newton method is typically higher when compared to the relaxation method.

V. RESULTS

Numerical results are presented for three examples: beam over a ground plane, two silicon bars positioned perpendicular to each other, and a comb drive structure. A Young's modulus of 169 Gpa and a Poisson's ratio of 0.3 is used for all the examples. The performance of the multilevel Newton and relaxation algorithms is examined for all the examples. In particular, the convergence characteristics and the simulation

TABLE III
COMPARISON OF RELAXATION AND MULTILEVEL NEWTON ALGORITHMS FOR
NUMBER OF ITERATIONS AND CPU(S) FOR THE CROSS BARS EXAMPLE (A *
INDICATES THAT THE ALGORITHM FAILS TO CONVERGE FOR THE BIAS)

Bias	# Iterations		CPU(s)	
	Relaxation	Multilevel Newton	Relaxation	Multilevel Newton
200.0	6	3	449.4	1255.8
400.0	11	4	1154.2	1738.9
600.0	27	4	3666.8	2583.4
700.0	73	3	10 984.2	1992.1
750.0	200	3	44 447.2	1966.1
800.0	*	3	*	1857.7

times are compared. All computations are performed on a Dec Alpha computer.

A. Beam Examples

Two beam examples over a ground plane structure are considered: the beams are of different dimensions and the convergence characteristics of the relaxation and multilevel Newton algorithms are different and these are pointed out in the results.

Example 1: The beam example considered here is 500 μm long, 50 μm wide, 14.35 μm thick, and positioned 1 μm above the ground plane. Fig. 2 shows a top view of the beam example. The beam is discretized into 50 parabolic elements, and the ground plane is discretized into 250 four-node elements. When a positive potential with reference to the ground plane is applied on the beam, the beam deflects toward the ground plane because of the electrostatic force. As the
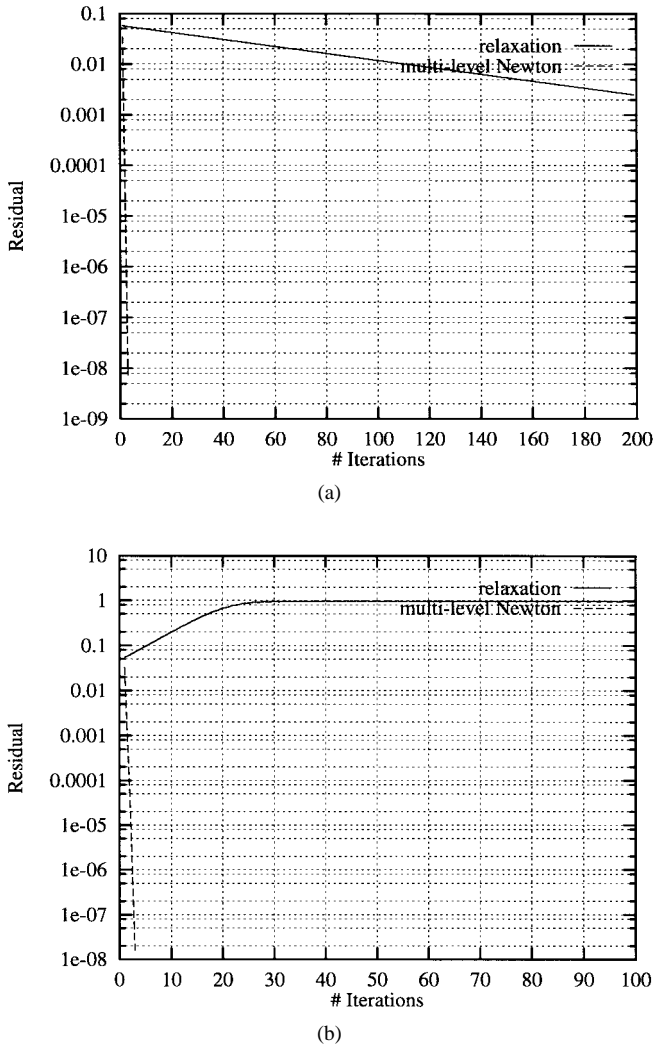
(a)

(b)

Fig. 12. Convergence of relaxation and multilevel Newton algorithms for the silicon bars example: (a) applied bias is 750 V and (b) applied bias is 800 V.

Fig. 13. Comb drive example.

potential difference increases, the tip of the beam approaches the ground plane and touches the ground plane for a certain bias defined as the pull-in voltage. The pull-in voltage for the beam considered here is 17.24 V.

Fig. 3 compares the peak deflection obtained from the relaxation and multilevel Newton algorithms. The results are identical verifying the accuracy of the multilevel Newton solver. The deflection of the beam for an applied bias of 17.23 V is shown in Fig. 4.

The performance of the relaxation and multilevel Newton algorithms for the beam example is summarized in Table I. Observe that the multilevel Newton algorithm takes fewer iterations and is much faster compared to the relaxation algorithm for tightly coupled cases. Fig. 5 compares the convergence of the relaxation and multilevel Newton algorithms for the beam and ground plane example. Note that closer to pull-in the relaxation algorithm converges slowly, but the coupled algorithm converges rapidly. The slow convergence of the relaxation algorithm, near pull-in, is due to the increased coupling between elastostatic and electrostatic systems. As the coupled algorithm accurately accounts for all the coupling it exhibits rapid convergence behavior.
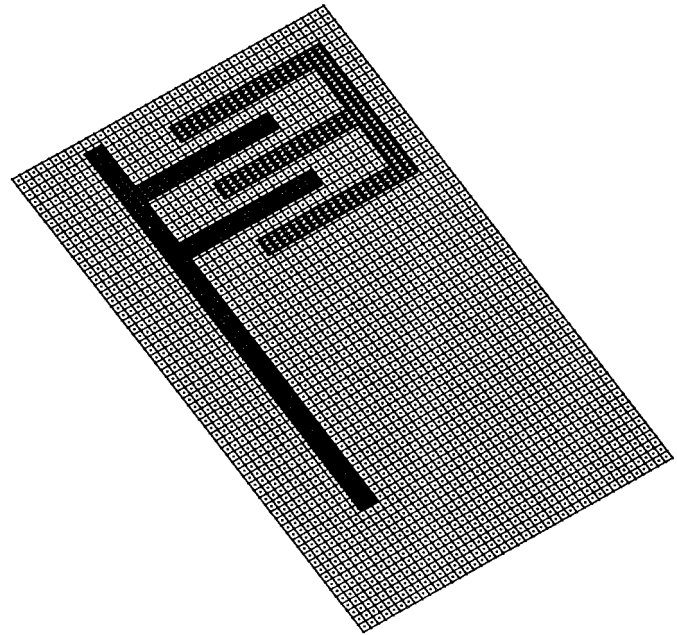
The multilevel Newton algorithm employs a matrix-free approach to compute the matrix–vector product required in the GCR algorithm. The selection of a matrix-free parameter to accurately compute the matrix–vector product is discussed in Section IV. Specifically, the matrix-free parameter involved two coefficients (a, b), and their choices were given as $a = 0.01$ and $b = 0.1$. While the convergence of the GCR algorithm is not sensitive to the value of b, Fig. 6 indicates that the GCR algorithm can converge slowly if the coefficient is picked as $a = 0.255$ instead of $a = 0.01$. This is not very surprising as matrix-free methods can be sensitive to the choice of the matrix-free parameter. The performance of the matrix-free method can be improved significantly by considering nonconstant values of the parameter a. For example, as shown in Fig. 6, when the residual stops to decrease, the value of parameter a can be changed for the next iteration so that the residual starts to decrease again. We are, however, not aware of a mathematically elegant way to adaptively change the value of parameter a so the residual decreases monotonically.

Example 2: The beam structure considered is 80 μm long, 10 μm wide, 0.5 μm thick, and positioned 0.7 μm above the ground plane. Fig. 7 shows a top view of the beam example. The pull-in voltage for the beam considered here is 2.39 V. The beam is discretized into 200 parabolic elements (100 $\times$ 2 $\times$ 1 along length, width, and thickness, respectively) and the ground plane is discretized into 624 four-node elements. The deflection of the beam at 2.38 V, just before pull-in, is shown in Fig. 8.

The comparison of relaxation and multilevel Newton algorithms for the entire bias sweep is summarized in Table II. As the bias voltage approaches pull-in, the relaxation algorithm converges slowly or fails to converge due to the increased coupling between elastostatic and electrostatic equations. The multilevel Newton algorithm, on the other hand, converges faster even as we approach pull-in. Comparison of the sim-

(a)   (b)
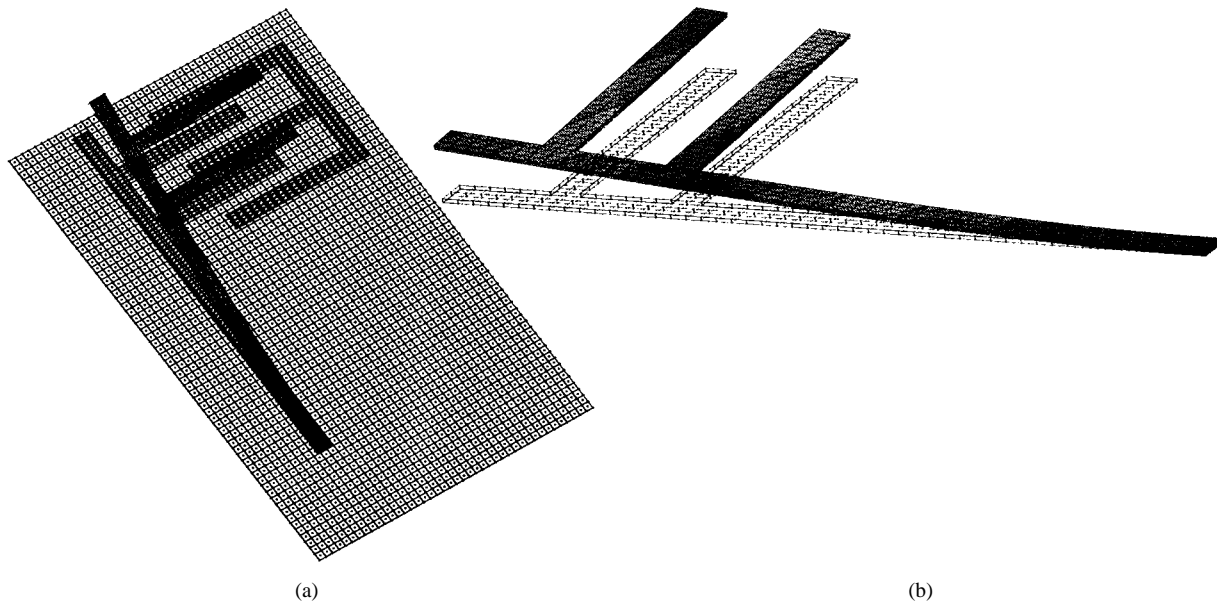
Fig. 14.   Deformation of the comb (not to scale) for an applied bias of 85 V. Shown on the left is the complete structure, and on the right is just the comb.

ulation times reveals that the multilevel Newton algorithm takes a little longer as compared to the relaxation algorithm for low bias voltages. However, closer to pull-in, the multilevel Newton algorithm takes very few iterations and is much faster. To predict the pull-in voltage for the beam structure, relaxation algorithm takes a total of 91285.9 s while the coupled multilevel Newton algorithm takes 46 747.2 s. Hence, the multilevel Newton algorithm is about two times faster than the relaxation algorithm for this example. The convergence of the multilevel Newton and relaxation algorithms for an applied bias of 2.38 V is shown in Fig. 9.

B. Cross Bars Example

The cross bars example consists of two silicon bars positioned as shown in Fig. 10. The beam positioned horizontally is the movable part and is positioned 1 μm above the fixed beam in the z direction. The left end of the horizontal beam is fixed, so that when a potential is applied on the movable beam, the tip of the movable beam deflects in the z direction. The movable beam is discretized with 30 parabolic elements and the fixed beam is discretized with 30 linear elements. The deflection of the beam for a potential difference of 2000 V is shown in Fig. 11. Note that this solution is obtained with the multilevel Newton approach and the relaxation algorithm does not converge for a potential difference of more than 750 V.

The performance of the relaxation and multilevel Newton algorithms for the cross bars example is summarized in Table III. The relaxation algorithm fails to converge for a potential difference of more than 750 V across the bars. The convergence of the relaxation and multilevel Newton algorithms for 750-V potential difference is shown in Fig. 12(a). Even after 200 iterations, the relaxation algorithm fails to converge to less than three orders of magnitude. Fig. 12(b) shows the rapid convergence of the multilevel Newton algorithm and the break down of the relaxation algorithm for an applied bias of 800 V.

Table III indicates that the multilevel Newton algorithm is very efficient as compared to the relaxation algorithm. For an applied bias of 700 V, the multilevel Newton algorithm is about 5.5 times faster, and for an applied bias of 750 V, the multilevel Newton algorithm is about 23 times faster compared to the relaxation algorithm.

C. Comb Drive Example

The comb example consists of a deformable comb structure, a drive structure, and a ground plane. As shown in Fig. 13, the F-shaped finger structure is the movable comb, the E-shaped finger structure is the drive, and the rectangular-shaped structure is the ground plane. The comb is discretized into 172 parabolic elements, the drive is discretized into 144 linear bricks, and the ground plane is discretized into 2688 four-node elements. When a positive potential is applied on the drive structure, and zero potential on the comb and the ground plane, the comb structure deforms out of plane. The deformation of the comb structure for an applied bias of 85 V is shown in Fig. 14. Note that only the comb structure deforms and the drive and the ground plane do not move.

A comparison of the relaxation and multilevel Newton algorithms for the comb example is summarized in Table IV. At low voltages, the deflection of the comb is small, the coupling between the electrical and mechanical systems is weak and the relaxation algorithm works very well. At low voltages, the multilevel Newton algorithm takes fewer iterations compared to the relaxation algorithm but the simulation time for the multilevel Newton algorithm is a little longer. For higher voltages, the multilevel Newton algorithm converges much faster compared to the relaxation algorithm. For a bias of 80 V, the multilevel Newton algorithm is about 7.7 times faster. The convergence of the relaxation and multilevel Newton algorithms at 80-V bias is shown in Fig. 15. For an application of 85 V on the drive, the relaxation algorithm fails to converge

TABLE IV
COMPARISON OF RELAXATION AND MULTILEVEL NEWTON ALGORITHMS FOR
NUMBER OF ITERATIONS AND CPU(S) FOR A COMB DRIVE EXAMPLE (A *
INDICATES THAT THE ALGORITHM FAILS TO CONVERGE FOR THE BIAS)

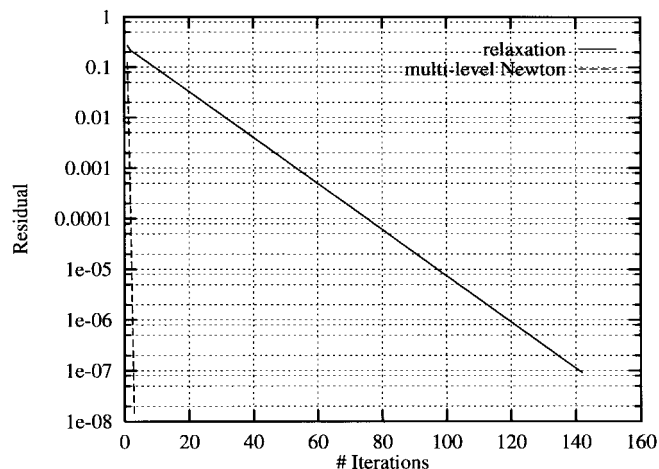| Bias	# Iterations		CPU(s)	
	Relaxation	Multilevel Newton	Relaxation	Multilevel Newton
25.0	7	3	3595.4	5802.2
50.0	16	4	9138.0	10 195.1
75.0	70	4	42 160.3	12 053.2
80.0	142	3	81 827.0	10 660.4
85.0	*	3	*	10 767.8

Fig. 15. Comparison of convergence of relaxation and multilevel Newton algorithms for a comb example at an applied bias of 80 V.
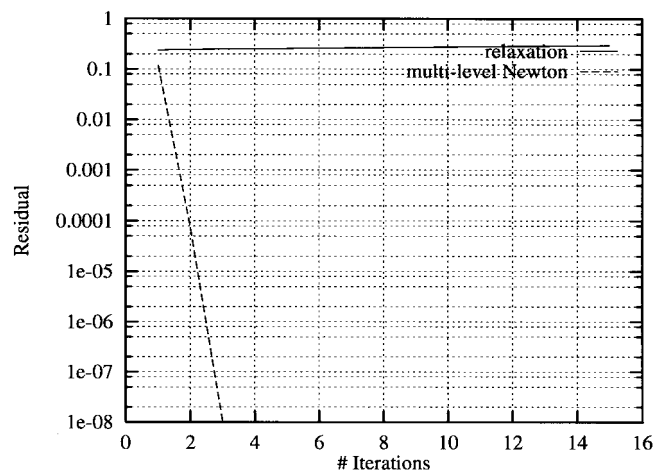

Fig. 16. Comparison of convergence of relaxation and multilevel Newton algorithms for a comb example at an applied bias of 85 V.

and the multilevel Newton algorithm converges very rapidly and takes only three iterations. This is illustrated in Fig. 16.

VI. CONCLUSION

In this paper, we presented a matrix-free-based multilevel Newton coupled algorithm for 3-D electromechanical analysis. Similar to the relaxation algorithm, the multilevel Newton algorithm employs repeated black-box calls to elastostatic and

electrostatic analysis. While the relaxation algorithm fails to converge for tight coupling between mechanical and electrical domains, the multilevel Newton algorithm is shown to converge very rapidly. Numerical results are presented for several electromechanical devices and CPU statistics indicate that the multilevel Newton can be two orders of magnitude faster than relaxation algorithm. The multilevel Newton algorithm has been shown to be sensitive to the choice of the matrix-free parameter. However, the choice of the matrix-free parameter given in this paper works extremely well for coupled electromechanical analysis. The multilevel Newton technique described in this paper can be extended for other coupled domains in MEMS such as microfluids, but the choice of the matrix-free parameter may need more investigation.

REFERENCES

[1] N. R. Aluru and J. White, "Direct-Newton finite-element/boundary-element technique for micro-electro-mechanical analysis," in *Tech. Dig. IEEE Solid-State Sensor and Actuator Workshop,* Hilton Head Island, SC, June 1996, pp. 54–57.
[2] _____, "An efficient numerical technique for electromechanical simulation of complicated micro-electro-mechanical-structures," *Sens. Actuators A,* vol. 58, pp. 1–11, 1997.
[3] M. Bachtold, J. G. Korvink, J. Funk, and H. Baltes, "New convergence scheme for self-consistent electromechanical analysis of iMEMS," in *Proc. Int. Electron Devices Meeting (IEDM),* 1995, pp. 605–608.
[4] K. J. Bathe, E. Ramm, and E. L. Wilson, "Finite element formulation for large deformation dynamic analysis," *Int. J. Num. Meth. Eng.,* vol. 9, pp. 353–386, 1975.
[5] X. Cai, H. Yie, P. Osterberg, J. Gilbert, S. Senturia, and J. White, "A relaxation/multipole-accelerated scheme for self-consistent electromechanical analysis of complex 3-D microelectromechanical structures," in *Proc. 11th Int. Conf. Computer-Aided Design,* Santa Clara, CA, Nov. 1993, pp. 283–287.
[6] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, "Variational methods for nonsymmetric systems of linear equations," *SIAM J. Numer. Anal.,* vol. 20, pp. 345–357, 1983.
[7] J. R. Gilbert, R. Legtenberg, and S. D. Senturia, "3D coupled electromechanics for MEMS: Applications of CoSolve-EM," in *Proc. IEEE Microelectromechanical Systems (MEMS),* 1995, pp. 122–127.
[8] J. Funk, J. G. Korvink, J. Buhler, M. Bachtold, and H. Baltes, "SOLIDIS: A tool for microactuator simulation in 3D," *IEEE J. Microelectromech. Syst.,* vol. 6, no. 1, pp. 70–82, 1997.
[9] L. Greengard, *The Rapid Evolution of Potential Fields in Particle Systems.* Cambridge, MA: MIT Press, 1988.
[10] K. Nabors and J. White, "FastCap: A multipole-accelerated 3-D capacitance extraction program," *IEEE Trans. Computer-Aided Design,* vol. 10, pp. 1447–1459, 1991.
[11] J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables.* San Diego, CA: Academic, 1970.
[12] J. R. Phillips and J. White, "A precorrected-FFT method for capacitance extraction of complicated 3-D structures," in *Proc. 12th Int. Conf. Computer-Aided Design,* San Jose, CA, Nov. 1994, pp. 268–271.
[13] A. Ruehli and P. A. Brennen, "Efficient capacitance calculations for three-dimensional multiconductor systems," *IEEE Trans. Microwave Theory Tech.,* vol. MTT-21, pp. 76–82, Feb. 1973.
[14] S. D. Senturia, R. M. Harris, B. P. Johnson, S. Kim, K. Nabors, M. A. Shulman, and J. K. White, "A computer-aided design system for microelectromechanical systems (MEMCAD)," *IEEE J. Microelectromech. Syst.,* vol. 1, pp. 3–13, Mar. 1992.
[15] H. Yie, X. Cai, and J. White, "Convergence properties of relaxation versus the surface-Newton generalized-conjugate residual algorithm for self-consistent electromechanical analysis of 3-D micro-electromechanical structures," in *Proc. Numerical Process and Device Modeling (NUPAD) V,* Honolulu, HI, 1994, pp. 137–140.

N. R. Aluru received the Bachelor of Engineering degree with honors and distinction from the Birla Institute of Technology and Science, Pilani, India, in 1989, the M.S. degree from Rensselaer Polytechnic Institute, Troy, NY, in 1991, and the Ph.D. degree from Stanford University, Stanford, CA, in 1995.

He was a Post-Doctoral Associate at the Massachusetts Institute of Technology (MIT), Cambridge, from 1995 to 1997. In 1998, he joined the Department of General Engineering, University of Illinois, Urbana-Champaign (UIUC), as an Assistant Professor. He is also affiliated with the Beckman Institute for Advanced Science and Technology, Department of Electrical and Computer Engineering, and Mechanical and Industrial Engineering Departmentat at UIUC (http://www.staff.uiuc.edu/~aluru/).

Jacob White (S'80–M'83) received the B.S. degree in electrical engineering and computer science from the Massachusetts Institute of Technology (MIT), Cambridge, and the S.M. and Ph.D. degrees in electrical engineering and computer science from the University of California, Berkeley.

He was at the IBM T. J. Watson Research Center from 1985 to 1987 and was the Analog Devices Career Development Assistant Professor at MIT from 1987 to 1989. He is currently a Professor at MIT, and his research interests are in serial and parallel numerical algorithms for problems in circuit, interconnect, device, and microelectromechanical system design. He was an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN from 1992 to 1996.

Dr. White was a 1988 Presidential Young Investigator. He was Technical Program Chair and General Chair of the International Conference on Computer-Aided Design in 1998 and 1999, respectively.