# Advanced Numerical Methods and Software Approaches for Semiconductor Device Simulation

Graham F. Carey*, A. L. Pardhanani* and S. W. Bova†



**Abstract**

In this article we concisely present several modern strategies that are applicable to drift-dominated carrier transport in higher-order deterministic models such as the drift-diffusion, hydrodynamic, and quantum hydrodynamic systems. The approaches include extensions of "upwind" and artificial dissipation schemes, generalization of the traditional Scharfetter-Gummel approach, Petrov-Galerkin and streamline-upwind Petrov Galerkin (SUPG), "entropy" variables, transformations, least-squares mixed methods and other stabilized Galerkin schemes such as Galerkin least squares and discontinuous Galerkin schemes. The treatment is representative rather than an exhaustive review and several schemes are mentioned only briefly with appropriate reference to the literature. Some of the methods have been applied to the semiconductor device problem while others are still in the early stages of development for this class of applications. We have included numerical examples from our recent research tests with some of the methods. A second aspect of the work deals with algorithms that employ unstructured grids in conjunction with adaptive refinement strategies. The full benefits of such approaches have not yet been developed in this application area and we emphasize the need for further work on analysis, data structures and software to support adaptivity. Finally, we briefly consider some aspects of software frameworks. These include dial-an-operator approaches such as that used in the industrial simulator PROPHET, and object-oriented software support such as those in the SANDIA National Laboratory framework SIERRA.

**Keywords:** semiconductor TCAD, device modeling, drift-diffusion, hydrodynamic, finite element, adaptive grids, software frameworks.

## 1 Introduction

The dramatic advances in microelectronics during the past two decades are largely a result of "shrinking" the technology. The semiconductor device is an integral part of the

*ASE-EM, TICAM, The University of Texas at Austin, Austin, Texas 78712. Phone: 512/471-4676. Email: carey@cfdlab.ae.utexas.edu. Fax: 512/232-3357.

†Sandia National Laboratories, Albuquerque, NM 87185. Phone: 508/844-6093. Email: swbova@sandia.gov.

## DISCLAIMER

## DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

hardware, and device size is now well below a micron with channel lengths from source to drain less than 0.1 micron and gate oxide layers less than 10 nanometers in thickness. For given voltage bias and operating conditions, as device size shrinks the local field strength inside the device increases and the interior layers in the property fields become more abrupt. Other physical effects such as quantum tunneling in the inversion layer become significant and several numerical difficulties commonly arise. These numerical difficulties are typically associated with the following issues: (1) an inadequate physical model that ignores physics (such as the quantum effect) that was negligible at the previous scale; (2) numerical effects associated with the high local gradients in the solution that adversely impact the convergence of the nonlinear iterative solver; (3) other numerical effects such as oscillations in the approximate solutions that are intrinsically tied to the resolution of the underlying grid and the stability of the chosen discretization scheme. These three difficulties are obviously not unique to the semiconductor device problem. In fact, they are endemic to numerical simulation of convection-dominated transport processes. Yet the device problem does embody some of the most extreme behaviors one may encounter in this class of problems. Part of the difficulty has to do with the scale and the reliability of a deterministic mathematical model such as those based on augmented drift-diffusion or hydrodynamic PDE systems. This is particularly evident in the vicinity of a charge accumulation region at the gate-oxide interface. Other discrete models using Monte Carlo solutions are possible but still not a practical alternative for the device designer's needs. Extending the deterministic models to include quantum tunneling in this regime is one approach that is proving very useful for present generation technology. For example, the use of WKB asymptotic expansion solutions to Schroedinger's equation extends the applicability of drift-diffusion and hydrodynamic models to smaller length scales[41, 42]. Since the issue of multiscale capability is a topical research subject in a number of modeling applications areas, we suggest that this is a good framework for interpreting the above problems – that is, different microscale (here quantum level) and macroscale (carrier transport by drift diffusion) effects need to be accommodated. This concept has not yet been explored to develop alternative simulation

models and strategies and remains an open research opportunity.

The second difficulty – convergence of the nonlinear scheme – is also more sensitive in the device simulation application [40] than in many other convection-diffusion problems. This is partly due to the strength of the solution gradients but a more significant issue appears to be the nonlinear source terms that describe carrier recombination (a reaction-like term [74]). Finally, the local strength of the electric field in the drift terms is extreme relative to the practical grid size, again because of the small length scales over which significant solution variations and sharp gradients occur. (The doping concentration of modern devices varies by several orders of magnitude over very short lengths.) Likewise, the potential and carrier concentration solutions to the device equations will have abrupt interior layers. This implies the "usual" stabilization needs for treating convection (the drift term) as well as a strong recommendation for graded meshes to avoid excessive dissipation introduced by the stabilization mechanisms. These last two items – stabilization of deterministic models and adaptive grid refinement – are the focus of the technical discussion in Sections 3 and 4.

As the scale of the devices shrinks, more complex models are needed and different types of analysis components are being linked to integrate simulation and design capability. This latter aspect implies a greater demand for computational flexibility and interoperability between analysis modules or simulation models and systems for both process and device simulation. Likewise, there is a need for improved pre- and post- processing, from geometric modeling and CAD, to automated grid generation, through integrated simulations, to collaborative visualization and visual steering. In section 5 we sketch some recent and ongoing ideas related to software frameworks to support these endeavors. Here we briefly describe the dial-an -operator approach in the industrial simulator PROPHET developed at Lucent and the work in progress on the SANDIA national Laboratory system SIERRA. Another consequence of the growing complexity of the PDE systems and the need for rapid simulation of high resolution grids is the increase in computational requirements to solve the problems. Advances in commodity processor speed have been instrumental in providing the needed computing power economically via desktop systems, area networks, tightly linked

clusters of personal computers or workstations, and large scale distributed supercomputer systems. Affordable parallel shared and distributed memory systems are now available and are beginning to be used for semiconductor applications. In section 6 we briefly summarize some recent developments related to parallel computation.

## 2 Transport Equations

The best known models for device simulations are based on the stationary drift-diffusion (DD) equations for electrostatic potential $\psi$ and carrier concentrations $n$ and $p$

$$\begin{aligned} -\nabla \cdot (\epsilon \nabla \psi) &= q\,(p - n + C) \\ \nabla \cdot (\boldsymbol{J}_n) &= q\,R(\psi, n, p) \\ \nabla \cdot (\boldsymbol{J}_p) &= -q\,R(\psi, n, p) \end{aligned} \tag{1}$$

with

$$\begin{aligned} \boldsymbol{J}_n &= q\mu_n n\boldsymbol{E} + qD_n \nabla n \\ \boldsymbol{J}_p &= q\mu_p p\boldsymbol{E} - qD_p \nabla p \end{aligned} \tag{2}$$

where $\mu_n$ and $\mu_p$ are electron and hole mobilities, $D_n$ and $D_p$ are corresponding diffusivities, $\boldsymbol{E} = -\nabla\psi$ is the electric field, $q$ is the unit charge, $C$ is the electrically active impurity concentration, and $R$ is the electron-hole recombination term [27, 53, 55].

An augmented system that permits a simplified treatment of hot electron effects can be constructed by making the mobility $\mu_n$ a function of the local field gradient, such as $\mu_n = \mu_n(\frac{\partial |E|}{\partial s})$ where $\frac{\partial E}{\partial s}$ corresponds to the gradient of $E$ in the field direction. A more rigorous "hydrodynamic" model for hot carriers can be obtained by including transport of energy density $w$ as an additional (the third-order) moment of the Boltzman transport equation [52]. Introducing the electron temperature $T$ and a closure relation for the heat flux in the moment system we obtain an additional convection-diffusion equation for $T$ of the form

$$\frac{\partial T}{\partial t} + v \cdot \nabla T = -\frac{2}{3}(\nabla \cdot v)T + \frac{2}{3n}\nabla \cdot \boldsymbol{Q} + \left(\frac{\partial T}{\partial t}\right)_c \tag{3}$$

where the subscript $c$ denotes collision contributions and the closure relation for the heat flux $Q$ is an "appropriate" constitutive relation (a Fourier type relation $Q = -k\nabla T$ is frequently assumed with some question as to both the validity of this form and the value of $k$ [52]).

Quantum effects can be included by modifying these underlying hydrodynamic transport equations using potential solutions for Schroedinger's equation in subregions or by more general treatments. For example, the momentum displaced Wigner distribution function may be introduced in the moment expansion [33, 39].

These PDE systems are discretized and the resulting nonlinear system is solved for a sequence of applied voltages to determine the $I-V$ curve for the device design in question and to analyze other effects concerning the performance or breakdown of the device. The potential and transport systems are often solved in an iteratively decoupled form (Gummel iteration) but other algorithms are also applicable and used.

## 3 Stabilization

It is easy to verify that the central difference scheme permits oscillatory approximations to a monotone analytic solution of the model 1-D drift-diffusion problem with zero source when the grid is not adequately refined. To see this, consider the 1-D convection-diffusion equation expressed in the form $u'' - \frac{E}{D}u' = 0$ on the domain $0 < x < 1$. A standard Galerkin scheme with linear elements or the standard central difference scheme both yield the same 3-point difference equation at a representative node $i$ of a uniform mesh. By writing this difference equation in terms of slopes $s+$ and $s-$ to the right and left of the center node $i$ of this difference patch and simplifying we find that the sign of the ratio $s+/s-$ is negative, so the computed solution will be oscillatory if the mesh size $h$ exceeds $2D/E$. This classical result is known as the cell Peclet condition ($\frac{hE}{D} \leq 2$).

As with many other transport problems, in the semiconductor device problem it is the convective or 'drift' term in the carrier and energy transport equations that leads to the use of so-called 'stabilization' strategies to suppress numerical oscillations. In the case of

the device problem the difficulty is compounded by the fact that these oscillatory errors have large gradients that alternate in sign and usually promote divergence of the nonlinear iterative solution scheme for the system [54]. Since there is natural diffusion in the system, this suggests that stabilization may not be necessary provided the grid is graded to a sufficiently fine resolution into the regions where there are high solution gradients. The adaptive grid strategies that we discuss later provide a logical approach to achieve this goal. Nevertheless some form of stabilization that accommodates the near-hyperbolic nature of the problem is desirable, especially since the graded mesh and adequate initial solution iterate on that mesh are not known *a priori.* Instead, they must be arrived at incrementally from an initial coarse grid and iterate using some form of continuation process. In our studies we have achieved some success by the following type of continuation process: generate an initial coarse grid and solve a simpler problem at low applied voltage (e.g. the potential problem at zero bias); improve the grid, adjust the model to drift-diffusion and use a strongly stabilized scheme to compute the next iterate; improve the grid and adjust the model to hydrodynamic and use a moderately stabilized scheme to solve the problem; repeat the last step to convergence at this applied voltage with adaptive refinement and modified stabilization; apply Euler-Newton continuation [64] to obtain the starting iterate for the next point on the $I-V$ curve and proceed in a similar vein to the last part of the previous $I-V$ step.

It is clear from the above that stabilization and mesh adjustment are key components of a successful algorithm. The "near hyperbolic" nature of the transport problem implies that the convective term should be treated with care. This is well known both in the device simulation community and in other flow and transport applications such as high speed gas dynamics where convective effects are strong and shock-like layers can arise [49, 56]. In fact, the behavior of the charge carriers in the device problem is analogous to an "electron gas" [32, 34]. Classical approaches for ensuring stability of hyperbolic problems involve "upstream" or "upwind" differencing to incorporate the directional property of the drift, ideally within some modified method of characteristics formulation. These upwind schemes

introduce numerical diffusion corresponding to the leading order truncation error terms from the discretization. This artificial diffusion is a function of the grid size and convective coefficient. One consequence is that approximations on coarse grids may be very dissipative and the layers are smeared. Since the numerical or artificial diffusion added by these schemes is much greater than any physical diffusion in the original problem this often makes the solution of little practical use other than as a starting iterate for a new solution on an improved grid. However, at a sufficiently fine grid resolution, such as the 1-D resonant tunneling QHD studies in Gardner [33], a uniform fine grid can be used to obtain good results, but solutions in 2D or 3D can not be efficiently obtained.

Degradation of the physical layers (smearing over several grid cells) can be mitigated by the introduction of higher-order upwind schemes that still suppress oscillations, such schemes being frequently adapted from computational fluid dynamics (CFD). Some examples are the flux limiter schemes [73] and similar total variation diminishing (TVD) or bounded (TVB) schemes [21, 46, 78] or the non-oscillatory ENO schemes [19, 70]. This latter strategy has been applied to simple diode simulations to obtain non-oscillatory approximations with sharp fronts [29, 35].

The idea of discretizing the drift term to better reflect the underlying physics is also the key to the well known Scharfetter-Gummel strategy [68]. A superior one-dimensional upwind difference approximation is constructed by first assuming the electric field $E = -\psi'$ is constant on each cell (or element) and analytically solving a simplified differential equation locally for the current density $J_n$ in terms of grid point (nodal) electron concentrations

$$J_n|_{i+1/2} = D_n \left( \frac{2\alpha_{i+1/2}}{h_i} \right) \left( \frac{n_{i+1} e^{\alpha_{i+1/2}} - n_i e^{-\alpha_{i+1/2}}}{e^{\alpha_{i+1/2}} - e^{-\alpha_{i+1/2}}} \right) \tag{4}$$

where $\alpha_{i+1/2} = \frac{1}{2} E_{i+1/2} h_i = -\frac{1}{2}(\psi_{i+1} - \psi_i)$ with $h_i = x_{i+1} - x_i$ the cell length between nodes $i$ and i+1.

Then setting $[\![J_n]\!] = 0$ at $x = x_i$, where $[\![.]\!]$ denotes the jump in the quantity, yields the desired upwind weighted three-point difference approximation. Note that the resulting scheme involves exponentials (or hyperbolic cotangents) in the difference coefficients and

that these can be interpreted in terms of the local Green's function for the linearized equations on the cell. We remark also that these same hyperbolic cotangents arise in the choice of weights for exact superconvergence of the Galerkin method with linear elements applied to the model convection-diffusion equations.

The exponential weighting in the Scharfetter-Gummel weighted difference scheme can be interpreted as related to the underlying Green's function for the linear drift-diffusion operator. That is, introducing the integrating factor $e^{-\frac{E}{D}x}$ and simplifying, we can obtain the corresponding Green's function for a source $\delta(x-\xi)$. By incorporating the local effect of the approximate Green's function at the cell or element level, the Scharfetter-Gummel difference weighting provides a good stabilization and suppresses oscillations that would otherwise arise from a standard central difference treatment of the drift-diffusion equations in one dimension.

Motivated by the success of upwind differencing the convective term, analogous upwind-biased Galerkin schemes have been introduced and refined in various ways. These usually are posed as a Petrov-Galerkin formulation in which the test functions are weighted in the upstream direction. A simple construction for the model problem that is conceptually linked to the previous Green's function ideas is to introduce an integrating factor to express the problem in self-adjoint form and then write the standard Galerkin formulation of this transformed problem to obtain

$$\int_0^1 u'(e^{-\frac{E}{D}x}v')dx = \int_0^1 f(e^{-\frac{E}{D}x}v)dx \tag{5}$$

which yields a non-oscillatory approximate method . This scheme can be related to exponentially upwinding the test function [12].

In higher dimensions the upwinding issue is more complicated. A common approach for finite difference schemes on Cartesian grids has been to apply the 1D formula in the respective coordinate directions. However, since the field $\boldsymbol{E}$ is in general oriented at some angle to the axes, simply using its components as the associated 1D drift components will yield a scheme with excessive cross-dissipation. (Similar observations apply to other

transport applications in CFD where this approach has been followed.) An alternative is to rotate the coordinate frame into a "streamline-normal" system and apply the 1D formula in the streamline direction, then rotate back to the original frame. Similarly, in the higher dimensional Petrov-Galerkin scheme the upwind bias for the test function can be constructed to be aligned with the field vector. The resulting scheme is then a streamline upwind Petrov-Galerkin (SUPG) form [9, 44].

A streamline diffusion scheme can be constructed simply by adding artificial diffusion in the streamline direction. For our model steady drift-diffusion equation we then obtain the following weak statement: find $u$ such that

$$\int_\Omega (\hat{e} \cdot \nabla u) w dv + \frac{D}{|\boldsymbol{E}|} \int_\Omega \nabla u \cdot \nabla w dv + \gamma(|\boldsymbol{E}|, h) \int_\Omega u_{\xi\xi} w dv = 0 \tag{6}$$

holds for all admissible $u$, where $\hat{e}$ is the unit vector in the direction of $\boldsymbol{E}$, the artificial diffusivity $\gamma$ is a function of the electric field strength and $\xi$ is in the direction of the field. (In two dimensions $u_\xi = u_x x_\xi + u_y y_\xi = E_1 u_x + E_2 u_y$ where $E_1$, $E_2$ are the field components.) This weak statement is equivalent to solving the "dissipative" differential equation:

$$u_\xi - \frac{D}{|\boldsymbol{E}|} \Delta u - \gamma u_{\xi\xi} = 0 \tag{7}$$

The function coefficient $\gamma$ in the stabilization term is chosen to satisfy $\gamma = 0$ if $\epsilon = D/|\boldsymbol{E}| \geq h$; that is, the mesh size is such that there are grid points in the solution layers. Hence we may set $\gamma = h - \epsilon$ if $h > \epsilon$ and $\gamma = 0$ otherwise to get a viable stabilization scheme. However, since this dissipation is an $O(h)$ modification of the original problem it is not surprising that the asymptotic accuracy is now only $O(h)$. That is, this scheme is only first-order accurate.

The streamline upwind Petrov-Galerkin construction recovers the second-order accuracy: First, consider the degenerate hyperbolic problem obtained by letting $D \to 0$. We then have, $\boldsymbol{E} \cdot \nabla u = f$ or, equivalently, $u_\xi = f/|\boldsymbol{E}|$. Biasing the weight in the upstream direction $-\hat{e}$ we set $\tilde{w} = w + \gamma b$ with $b = w_\xi$, and amplitude $\gamma$ is to be specified. Then the weak statement becomes: find $u$ satisfying

$$\int_\Omega u_\xi (w + \gamma w_\xi) dv = \int_\Omega \frac{f}{|\boldsymbol{E}|} (w + \gamma w_\xi) dv \tag{8}$$

Re-introducing the diffusion term ($D \neq 0$) we write similarly,

$$\int_\Omega (u_\xi - \epsilon \Delta u)(w + \gamma w_\xi) dv = \int_\Omega \frac{f}{|\boldsymbol{E}|}(w + \gamma w_\xi) dv \tag{9}$$

and reorder terms to get

$$\int_\Omega u_\xi w dv + \gamma \int_\Omega u_\xi w_\xi dv + \epsilon \int_\Omega \nabla u \cdot \nabla w dv - \epsilon\gamma \int_\Omega \Delta u w_\xi dv = \int_\Omega \frac{f}{|\boldsymbol{E}|} w dv + \gamma \int_\Omega \frac{f}{|\boldsymbol{E}|} w_\xi dv \tag{10}$$

The construction is completed by "interpreting" the last term on the left as a sum of element integral contributions. (Note that this implies that interface jumps are ignored in computing this contribution and that for linear elements $\Delta u_e = 0$ in the element interior implies that this term is zero.) In our opinion this is a less than satisfactory situation, but the fact that $\epsilon\gamma$ scales the term with both $\epsilon$ and $\gamma$ small implies that asymptotically the "correction" is valid. It should also be kept in mind that we are, in fact, perturbing the original problem in the sense that a higher-order artificial dissipation may be associated with the upstream bias term.

The time dependent SUPG scheme follows in a similar fashion with $\boldsymbol{E} \cdot \nabla u$ replaced by $u_t + \boldsymbol{E} \cdot \nabla u$ and the remainder of the formulation as above. This, however, also allows us to introduce other treatments that have been the subject of recent study in other applications areas. Of particular interest are the space-time and discontinuous Galerkin methods [2, 6, 22, 47].

Let us consider the 1-D case. Introducing the "pseudo-material derivative" $Du/Dt = u_t + \boldsymbol{E} \cdot \nabla u = u_\xi$, then in the new $\xi(x,t)$ space-time frame we have

$$u_\xi - D u_{xx} = f, \quad u_\xi = u_t + E u_x \tag{11}$$

Adding an artificial diffusion $\gamma u_{xx}$ we can construct a corresponding space-time Galerkin Scheme

$$\int_0^T \int_0^1 u_\xi w dx dt + \epsilon \int_0^T \int_0^1 u_x w_x dx dt + \gamma \int_0^T \int_0^1 u_x w_x dx dt = \int_0^T \int_0^1 f w dx dt \tag{12}$$

and similarly, a space-time SUPG scheme follows of the form

$$\int_0^T \int_0^1 u_\xi (w + \gamma w_\xi) dx dt - \int_0^T \int_0^1 \epsilon w_{xx} (w + \gamma w_\xi) dx dt = \int_0^T \int_0^1 f (w + \gamma w_\xi) dx dt \tag{13}$$

or in higher dimensions

$$\int_0^T \int_\Omega u_\xi(w+\gamma w_\xi)d\Omega dt - \epsilon \int_0^T \int_\Omega \nabla u \cdot \nabla v(w+\gamma w_\xi)d\Omega dt = \int_0^T \int_\Omega f(w+\gamma w_\xi)d\Omega dt \quad (14)$$

The space-time formulations can also be applied on individual time strips $S_n = \Omega \times [t_n, t_{n+1}]$ where $\Delta t_n = t_{n+1} - t_n$ is the time interval of interest and $\Omega$ is the spatial domain ($\Omega = [0,1]$ in the example above). For example, a Petrov-Galerkin space-time formulation could be introduced of the form: find $u$ satisfying the "initial" condition at $t = t_n$ and the essential boundary conditions on $\Gamma_n = \partial\Omega \times [t_n, t_{n+1}]$ and such that

$$\int_{S_n} (u_t w + \boldsymbol{E} \cdot \nabla u w + D\nabla u \cdot \nabla w)d\Omega dt = \int_{S_n} f w d\Omega dt \quad (15)$$

holds for all admissible test functions $w(\Omega, t)$. (Here, for convenience, we have taken $u(\Omega, t)$ to be specified on $\Gamma_n$, so $w = 0$ on this part of the strip boundary as well as on the surface $\Omega$ of the strip at $t = t_n$.) A finite element strip method follows on discretizing the space time strip as a single layer of elements and introducing an appropriate basis for the approximation trial and test functions. For example, a tensor product bilinear basis for trial and test functions on a strip of rectangular elements will yield an implicit difference scheme similar to those encountered in the Crank Nicolson difference method and the Crank Nicolson Galerkin semidiscrete strip schemes. In a similar manner, a strip of triangular elements with a continuous conforming basis might be applied to yield a more general fully discrete space-time algebraic system for the strip, and the strip need not be of fixed width in time. Other generalizations are also feasible. For instance, the test functions in the previous tensor product discretization of rectangles may be constant in time and continuous piecewise linear in space. This implies a test basis that is discontinuous across the time interfaces between adjacent strips. This Petrov Galerkin scheme also yields an implicit system to be solved for each time interval. Error estimates and superconvergence properties (in time) have been shown for this type of scheme applied to the model diffusion equation [3].

The continuity of the trial space across the strip interfaces at $t_n$ can also be weakened to obtain a discontinuous-in-time Galerkin or Petrov-Galerkin formulation. Let $u+$ and $u-$

denote the approximations as $t_n$ is approached from above or below, respectively. Then the strip interface jump condition $[\![u]\!] = 0$ on the initial surface can be enforced weakly in the variational statement on each strip. That is, we add surface integrals to the variational problem on $S_n$ of the form $\int_\Omega [\![u]\!]^2 d\Omega$. Note that in this scheme the trial and test spaces as well as the mesh need not be conforming across time strip boundaries. These schemes can also be extended to include upwind strategies and additional stabilisation treatments such as SUPG to accommodate the drift term in the semiconductor device problem. A further generalisation to treat drift-dominated and hyperbolic PDE problems is to use discontinuous Galerkin schemes at the individual element level. In this case the "inflow" and "outflow" boundaries are identified for a specified field direction and an arbitrarily oriented element. The approximation and test functions are now discontinuous across interelement faces with jump conditions enforced weakly using the correct directional drift inclusion. This concept is applicable to arbitrary space-time elements but has not apparently been investigated to date in this broader context.

The time-dependent transport problem also can be treated by introducing an equivalent numerical dissipation term via the time discretization. This is, in fact, the basis of the familiar Lax-Wendroff approach for hyperbolic problems. The basic idea is to use the differential equation as an auxiliary relation. This relation can be differentiated and manipulated to express the leading time truncation error as a spatial dissipation term that can in turn be differenced or similarly discretized. A simple illustrative example is afforded by the fundamental drift equation $u_t = -Eu_x$. Forward differencing with respect to time we get

$$\frac{u^{n+1}(x) - u^n(x)}{\Delta t} = -Eu_x(x, t_n) + \frac{\Delta t}{2} u_{tt}(x, t_n) + \cdots \tag{16}$$

Differentiating $u_t = -Eu_x$ with respect to $t$ and simplifying we have $u_{tt} = E^2 u_{xx}$ so that the time discretized equation becomes

$$\frac{u^{n+1}(x) - u^n(x)}{\Delta t} = -Eu_x(x, t_n) + E^2 \frac{\Delta t}{2} u_{xx}(x, t_n) + O((\Delta t)^2) \tag{17}$$

Neglecting terms of $O((\Delta t)^2)$ and differencing centrally in $x$ yields the Lax-Wendroff dissipative stabilized scheme.

In like fashion, we can take this resulting dissipative form and integrate against a test function $w(x)$ to obtain a Taylor-Galerkin scheme. This idea has been generalized to construct a number of higher-order stabilized difference schemes [71, 72]. Note that the artificial dissipation in $x$ varies as the square of $E$ and linearly with $\Delta t$. This implies that care must be exercised that both the scheme not be excessively dissipative ($E^2$ and $\Delta t$ large) and that $\Delta t$ not be so small that oscillations arise. Results for a simple $0.4\mu$m silicon diode are shown in Figure 1 (from [9]). Here the doping is $2 \times 10^{18}$ cm$^{-3}$ at source and drain with $2 \times 10^{15}$ cm$^{-3}$ in the channel. The solution shown is obtained by time-marching to a steady state.

Other variants of the SUPG scheme may also be developed. For example, in [10] a transformation to "entropy" variables is introduced and used to symmetrize the flux jacobian matrix for the hydrodynamic system. Consider, for example, the non-parabolic energy band transport system

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}_1(\boldsymbol{U})\frac{\partial \boldsymbol{U}}{\partial x_1} + \boldsymbol{A}_2(\boldsymbol{U})\frac{\partial \boldsymbol{U}}{\partial x_2} = \boldsymbol{S}(\boldsymbol{U}) + \nabla \cdot k\nabla \boldsymbol{U} \tag{18}$$

Introducing a change of variables $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{V})$ we have

$$\boldsymbol{A}_0(\boldsymbol{V})\frac{\partial \boldsymbol{V}}{\partial t} + \hat{\boldsymbol{A}}_1(\boldsymbol{V})\frac{\partial \boldsymbol{V}}{\partial x_1} + \hat{\boldsymbol{A}}_2(\boldsymbol{V})\frac{\partial \boldsymbol{V}}{\partial x_2} = \hat{\boldsymbol{S}}(\boldsymbol{V}) \tag{19}$$

where now the entropy variable transformation is constructed such that $\boldsymbol{A}_0 = \frac{\partial \boldsymbol{U}}{\partial \boldsymbol{V}}$ is symmetric positive definite and $\hat{\boldsymbol{A}}_i = \boldsymbol{A}_i\boldsymbol{A}_0$ are symmetric. The SUPG scheme can then be formulated for the transformed system.

For example, in the above case we introduce the transformation for $\boldsymbol{U} = (n, nu, nv, nw)^T$ to $\boldsymbol{V} = (5/3 - s - \frac{(u^2+v^2)}{2w_i}, u/w_i, v/w_i, -1/w_i)^T$ with $s = \ln(p/n^{5/3})$, $w_i = w - (u^2 + v^2)/2$, where $p = 2/3nw_i$ arises from the Wiedemann-Franz assumption for $\boldsymbol{Q} = -k\nabla \boldsymbol{U}$.

In fact, SUPG is a member of a broader class of stabilized finite element methods known as Galerkin/Least-Squares (GLS) [45]. In this approach, the symmetric form of the governing differential equation may be written as

$$\mathcal{L}\boldsymbol{V} = \boldsymbol{A}_0(\boldsymbol{U})\frac{\partial \boldsymbol{V}}{\partial t} + \hat{\boldsymbol{A}}_1(\boldsymbol{V})\frac{\partial \boldsymbol{V}}{\partial x_1} + \hat{\boldsymbol{A}}_2(\boldsymbol{V})\frac{\partial \boldsymbol{V}}{\partial x_2} - \hat{\boldsymbol{S}}(\boldsymbol{V}) = \boldsymbol{0}. \tag{20}$$

Then Galerkin's method may be written as

$$\int_\Omega W^t \mathcal{L} V d\Omega = 0. \tag{21}$$

The GLS method adds a functional to (21) of least-squares form, namely

$$\int_\Omega W^t \mathcal{L} V d\Omega + \int_\Omega (\mathcal{L} W)^t \tau \mathcal{L} V d\Omega = 0, \tag{22}$$

where $\tau$ is a matrix of local element intrinsic time scales which is constructed from the modal matrix of the differential operator. In (22), if $\mathcal{L}W$ is replaced by the convective part of the operator $\hat{A} \cdot \nabla V$, then the SUPG method is recovered. Similarly, GLS and SUPG coincide for purely convective, steady-state problems.

There is a famous theorem of Godunov [36] which asserts that, in general, no linear numerical scheme for hyperbolic problems may simultaneously be monotonic and better than first-order accurate in space. Hence, modern numerical methods for capturing steep layers in convection-dominated problems typically incorporate some sort of nonlinear feedback mechanism to enforce monotonicity of the solution. In this way, the magnitude of the artificial dissipation is made proportional to the solution gradients. This is the motivation behind the so-called "discontinuity capturing operators" in the SUPG/GLS literature, and slope limiters for TVD and ENO methods. In this way, the formal order of accuracy of the numerical solution may be improved in regions where the solution is smooth, and reduced in regions of large, local gradients to suppress spurious oscillations and enforce monotonicity of the solution.

The first order system for carrier transport in (1)-(2) can also be approximated directly using a mixed method [31] by introducing the current densities $J_n$ and $J_p$ as additional variables. This increases the number of nodal unknowns and therefore the size of the algebraic system to be solved so these ideas have not been pursued in practice. However, in other applications areas such as coupled fluid flow and transport mixed methods are receiving increased attention because the flux quantities are approximated directly to greater accuracy. Moreover, local conservation can be enforced at the element level using appropriate elements such as the Raviart-Thomas family. This approach can also be applied to

the electrostatic potential equation and, for simplicity of exposition, we will describe the formulation for this case. Accordingly, let us first write the electrostatic equation as a first order system by introducing a flux $\sigma$ to obtain

$$\begin{aligned} \frac{1}{\epsilon}\sigma &= -\nabla\psi \\ \nabla\cdot\sigma &= f \end{aligned} \tag{23}$$

where we have also set $f = q(n - p + C)$. In the following we assume that $f$ is known , as in a block iteration where carrier concentration iterates are available from the previous step with the decoupled carrier transport equations. Note that the full system could also be considered as a single large first-order system using the mixed Galerkin formulation. The Galerkin statement for the mixed formulation follows after introducing test functions $w$ and $v$ corresponding to the variations of $\sigma$ and $\psi$ in a weighted residual statement: find $\sigma \in \boldsymbol{H}_{\mathrm{div}}(\Omega)$ and $\psi \in \boldsymbol{L}^2(\Omega)$ such that

$$\begin{aligned} \int_\Omega \epsilon^{-1}\sigma\cdot w dx - \int_\Omega \psi\nabla\cdot w dx &= \int_{\partial\Omega}\psi w\cdot n ds \\ \int_\Omega(\nabla\cdot\sigma)v dx &= \int_\Omega f v dx \end{aligned} \tag{24}$$

hold for all admissible $w \in \boldsymbol{H}_{\mathrm{div}}(\Omega)$ and $\psi \in \boldsymbol{L}^2(\Omega)$. (The function space notation implies that both $\psi$ and $\nabla\cdot w$ are square integrable.) Introducing the approximation subspaces and simplifying , this saddle point problem yields a block system of the form

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}^T \\ \boldsymbol{B} & \boldsymbol{0} \end{bmatrix}\begin{bmatrix} \boldsymbol{S} \\ \boldsymbol{\psi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{G} \\ \boldsymbol{F} \end{bmatrix} \tag{25}$$

which can be solved for the nodal values $\boldsymbol{S}$, $\boldsymbol{\psi}$ of $\sigma$ and $\psi$. Note that this system can be reduced by static condensation to the Schur's complement system

$$(\boldsymbol{B}\boldsymbol{A}^{-1}\boldsymbol{B}^T)\boldsymbol{\psi} = \boldsymbol{F} + \boldsymbol{B}\boldsymbol{A}^{-1}\boldsymbol{G} \tag{26}$$

which corresponds to the symmetric positive system obtained using the standard Galerkin method for the second order electrostatic potential equation in (1). The low order Raviart-Thomas spaces are frequently chosen to develop the system in (26). For a rectangular

element this implies a constant electrostatic potential approximation on each element and a corresponding center node . The flux components are approximated using tensor products of linears in the component direction and by constants in the remaining directions. That is $\sigma_{ih} = \alpha_i x_i + \beta_i$ are linear in the $i-$th coordinate direction and constant in the other directions. ( e.g., This implies nodes for $\sigma_i$ at the midpoint pair in direction $i$ on opposite sides of a rectangle.) For the triangle, the electrostatic potential is again constant on the element with a node at the centroid, and the flux is linear of the form $\sigma_{ih} = \alpha x_i + \beta_i$ with flux nodal values normal to the edges at the edge center nodes such that $\int_{\partial\Omega_e} \sigma_h \cdot n ds = 0$. The mixed treatment for the carrier transport equations in (1)-(2) follows in similar fashion: the weighted residual statement for the first order system is constructed and the approximation spaces introduced to give a saddle point problem and algebraic system of the same structure as that appearing previously in (25), where the electrostatic potential and electric field are now presumed known from the above electrostatic Gummel step.

The Galerkin approach leading to a saddle point problem is not the only mixed weighted residual formulation that can be developed. One such alternative is to use a least squares minimization formulation for the residuals in the first order problem . This avoids the restrictions associated with a saddle point formulation and generates a mixed system that is symmetric positive definite. The analysis of this type of treatment and variations of this approach are under investigation. For the electrostatic potential problem (23) a least squares residual functional can be easily written

$$I = \alpha \int_\Omega (\nabla \cdot \sigma - f)^2 dx + (1-\alpha) \int_\Omega (\frac{1}{\epsilon}\sigma + \nabla\psi)^T (\frac{1}{\epsilon}\sigma + \nabla\psi) dx \qquad (27)$$

where $\alpha$ is a weight and we seek a minimizer of $I$ for $(\sigma, \psi)$ in $\boldsymbol{H}_{\rm div} \times L^2$. The approximate problem is obtained by introducing finite element expansions $\sigma_h$ and $\psi_h$ in (27) and setting the first variation of $I$ to zero to obtain a system for the nodal vectors that is now symmetric and positive [14, 15].

# 4 Adaptive Grids

We have commented previously that locally adapting the grid can be very beneficial. Obviously, grading the mesh into the layers will improve accuracy and efficiency. Further, numerical oscillations arise as a consequence of discretizing on a grid that is too coarse. While adding dissipation or applying other strategies to suppress oscillations is important, the underlying difficulties stem from the mesh. The best approach is to combine a stabilized scheme with mesh adaption.

The main approaches for adapting the mesh are by (1) redistributing the nodes or (2) refining cells. The redistribution approach is best suited to computations using stencil-based finite difference schemes on mapped structured grids. For example, in a recent study we developed a mapped formulation for the device problem as follows: (1) first a coordinate mapping is introduced between a Cartesian reference grid and a topologically equivalent curvilinear graded mesh in the physical domain; (2) then the governing device transport equations are mapped to the reference domain and differenced (together with the metric coefficient introduced by the inverse map) on the Cartesian grid. This is standard practice for similar discretization schemes on structured grids in aerodynamics and CFD. The distinction now is that we have extended the Scharfetter-Gummel discretization procedure to the mapped equation and along the edges of the grid cells in the reference domain [57].

For example, consider the following hydrodynamic model for electron transport [7, 8]

$$\begin{aligned} \nabla \cdot (\boldsymbol{J}_n) &= 0 \\ \nabla \cdot (\boldsymbol{S}_n) + \boldsymbol{J}_n \cdot \nabla \psi &= -\frac{n(w - w_0)}{\tau_w} \end{aligned} \tag{28}$$

where $\boldsymbol{J}_n$ and $\boldsymbol{S}_n$ respectively denote the current density and energy flux, which are defined by the following equations

$$\begin{aligned} \boldsymbol{J}_n &= \tau_p q \left[ \frac{2}{3m^*} \nabla (B(w) n w) - \frac{q}{m^*} n \nabla \psi \right] \\ \boldsymbol{S}_n &= \boldsymbol{Q} - \Omega w \boldsymbol{J}_n / q \end{aligned} \tag{29}$$

The momentum and energy relaxation times, $\tau_p$ and $\tau_w$, are empirical functions, which are

chosen from the work of Bordelon et al. [8] as

$$\tau_p = \frac{0.007\,q}{w} \times 10^{-12} \text{ s}$$
$$\tau_w = 0.46 \times 10^{-12} \text{ s} \tag{30}$$

The system is closed by assuming a Fourier type constitutive relation for the heat flux, $Q$, of the form

$$Q = -\frac{2}{3}\left(\frac{\gamma R}{K_B}\right) n \nabla w \tag{31}$$

The other quantities in equations (28) - (31) are defined as follows: $B(w) = (1+\alpha\frac{w}{q})/(1+2\alpha\frac{w}{q})$, $q$ = electron charge = $1.602 \times 10^{-19}$ C, $m^*$ = effective mass = $2.367 \times 10^{-31}$ kg, $\epsilon$ = permittivity of silicon = 11.9 $\epsilon_0$, $\epsilon_0 = 8.854 \times 10^{-12}$ $C^2$/(joule m), $K_B$ = Boltzmann constant = $1.381 \times 10^{-23}$ joule/kelvin, $w_0 = \frac{3}{2}K_B T_L$ joule, $T_L$ = lattice temperature in kelvin, and $\alpha = 0.5eV^{-1}$, $\Omega = 1.3$, $\gamma = 4.2 \times 10^{-26}$ (watt $m^2$)/kelvin, $R = 0$ to 0.5 are empirical constants.

The mapped Scharfetter-Gummel approach is derived for the current-density and energy-flux terms in this hydrodynamic system. Details of this derivation for the case of a general coordinate mapping are given in [57]. Here we state these results in the original coordinate system, and give the usual one-dimensional version that is used in a finite-volume setting to discretize the $\nabla \cdot J_n$ and $\nabla \cdot S_n$ terms. Accordingly, if we use subscript $i$ to denote quantities at node $i$, the (constant) current-density and energy flux components on the mesh segment between nodes $i$ and $i+1$ are given by

$$\frac{J_n}{q} = \frac{2}{3}\frac{C_\tau B(w_i^{av})}{m^* \Delta x}\left[\frac{n_{i+1}}{w_{i+1}}\beta(\mathcal{X}) - \frac{n_i}{w_i}\beta(-\mathcal{X})\right]\frac{(w_{i+1}-w_i)}{\ln(w_{i+1}/w_i)} \tag{32}$$
$$S_n = \nu[w_{i+1}\beta(\mathcal{Y}) - w_i\beta(-\mathcal{Y})] \tag{33}$$

where

$$\mathcal{X} = \left[\frac{3}{2}\frac{q}{B(w_i^{av})}\frac{(\phi_{i+1}-\phi_i)}{(w_{i+1}-w_i)} - 2\right]\ln(w_{i+1}/w_i)$$
$$\mathcal{Y} = \frac{\Omega J_n/q}{\nu}$$
$$\nu = \frac{H}{\Delta x}\ln\left(\frac{n_{i+1}}{n_i}\right)\frac{n_{i+1}n_i}{n_{i+1}-n_i}$$

and $\beta(x) = x/(e^x - 1)$ denotes the Bernoulli function. The other quantities introduced in (32) - (33) are

$$\begin{aligned} C_\tau &= \text{coefficient of } \tau_p = 0.007 \times 10^{-12}, \\ \Delta x &= \text{local mesh spacing } = (x_{i+1} - x_i), \\ w_i^{av} &= \text{average energy along edge } \Delta x = \frac{1}{2}(w_{i+1} + w_i) \end{aligned}$$

Note that the transformation takes into account the way the mesh is graded and this permits grading the mesh into regions where solution gradients or errors are large.

The second approach for improving the grid is to add new grid points locally to enrich the mesh in certain regions and simultaneously remove grid points in those parts of the domain where they are not needed. The 1D FETG result in Figure 1 was first computed on a uniform grid of 100 elements and generated solution profiles with strong oscillations near regions of large solution gradient. The non-oscillatory results in the Figure are for a final nonuniform grid of 150 elements obtained by point insertion. The situation in higher dimensions is obviously more difficult. However, points can be added conveniently as part of a Delaunay triangulation process [13]. The Delaunay triangulation is optimal in the sense that it connects the nodes (grid points) so that the local triangle shape is the best possible in a certain geometric sense (in the sense of maximizing the minimum angle via edge swaps). The refinement algorithm for grid point insertion is very straightforward: assume a new point $p$ is to be added in a designated element based on the solution behavior or a local error "indicator"; add the point and identify any neighboring triangles that will be influenced by the Delaunay process; remove the corresponding interior edges to define a "cavity" around $p$ and connect $p$ to the vertices of the cavity. This process is repeated recursively until the point insertion is completed. Similarly a vertex center of an interior patch can be deleted to form a polygonal cavity which can then be retriangulated to again meet the Delaunay requirement. Local coarsening can thus be achieved by successive point deletions, again guided by a local error or feature indicator. The approach can be extended to point insertion into tetrahedral grids.

Not only is this point insertion strategy simple, but it also incurs little overhead to the data structure, using only the edge neighbor information available from the Delaunay process. Yet, surprisingly, it appears to be little used for adaptive grid enrichment. In the case of existing industrial software that uses unstructured triangulation, this point insertion approach is appealing because it is relatively straightforward to retrofit the adaptive component to the analysis software. Hence, this is the easiest path for upgrading existing device and process simulators to include adaptivity and yield more accurate, reliable simulations that are more stable and not oscillatory.

A more common approach for adaptive refinement that does not require a Delaunay property is to simply insert points at the midpoints of specified edges of a triangulation. For example, we can refine a designated triangle to a quartet of similar subtriangles by simply connecting new nodes at the midpoints of the three sides. The neighbor triangles can then be refined by connecting these midpoint nodes to the opposing vertices or a similar strategy. This idea has been applied by Bank *et al* [4, 5] to device simulations using adaptive refinement with a multigrid solver. The approach could also be combined with Delaunay swaps to help avoid generating poorly-shaped sub-triangles. Such a scheme relies on an element-based data structure using an element error indicator. A variant of this method is to split designated edges (rather than elements) guided by an edge-based error indicator. A point is then inserted in a given edge and the adjacent triangles are appropriately subdivided. (Clearly this can also be done using the previous Delaunay procedure). These strategies and variants of them can easily be generalized to tetrahedra. For example, see Plaza and Carey [62, 63] for recent tesselations based on longest edge bisection of tetrahedra using the skeleton triangulation. The sketch in Figure 2 shows a tetrahedron refined in this manner.

The approach of subdividing the triangle to a quartet of sub-triangles or the tetrahedron to an octet of sub-tetrahedra generates respectively a quadtree and an octree data structure that can be exploited in the simulation to yield a more efficient adaptive algorithm [18]. Rather than reconnect midside nodes of edges shared by unrefined neighbor elements, one

can include constraints on the solution behavior along these edges to ensure appropriate continuity or smoothness of the approximation (conformity). This approach can also be applied to quadrilateral and hexahedral (quadrilateral brick) elements with their associated quadtree and octree data structures. One of the first studies of this type for device simulation used quadrilateral elements for a MOSFET simulation [17, 69]. A sketch of the potential field from that early calculation is given in Figure 3. More recently Dutton *et al* [20, 26, 37] and Hitschfield *et al* [43] have developed adaptive schemes for process and device simulations respectively.

Instead of refining the mesh to improve accuracy, one may increase the order of the difference scheme or the degree $p$ of the finite element basis. The latter $p$-type or spectral element approach is particularly appealing, since the grid with mesh parameter $h$ remains fixed and the element degree can be increased as needed. For elliptic boundary value problems with smooth solutions, a polynomial element basis of degree $p$ will yield a global asymptotic rate of convergence in the $L^2$ norm that is $O(h^{p+1})$. This high accuracy and rapid convergence can be achieved by increasing $p$ to the necessary level on a relatively coarse grid. The element matrices increase with $p$ and the bandwidth grows correspondingly, but the high accuracy implies that these methods will be more efficient when the solution is sufficiently regular. However, for problems with singularities, the reduced global regularity restricts the rate of convergence: if the solution is in $H^r$ then the rate in $L^2$ becomes $\mu = \min(p+1, r)$ , so $r$ limits the rate of a $p$ scheme. In this case it is desirable to adapt by refining the mesh towards singularities (local $h$ refinement) and increase the polynomial degree on elements remote from the singularities. Such schemes are called adaptive $hp$ methods and have not yet been applied to semiconductor device or process simulation although they are used in other field problems in engineering mechanics and electromagnetics.

A uniform $p$ refinement scheme has been applied to compute solutions to the augmented drift-diffusion equations mentioned previously. This study involved the use of parallel multilevel iterative solution techniques in which the level corresponded to the degree of the element polynomial basis [24]. Adaptive $p$ schemes can be constructed in a manner similar

to the adaptive $h$ schemes where now the polynomial degree varies across the elements of the discretization. This implies that, as in the case of $h$-refinement, a strategy is needed to permit transition between refined and unrefined elements. In the adaptive $h$ scheme on simplices this can be achieved by connecting the "hanging" node on the element interface to the opposite vertices of the adjacent unrefined element. For hexahedral grids, special transition elements can be constructed or techniques for locally constraining the solution by penalties or Lagrange multiplier methods can be introduced. Similarly, in the adaptive $p$ scheme, continuity of the approximation across the element interface can be enforced by constraining the higher degree basis functions on the element interface. If a hierarchic basis is used, then this simply implies that the appropriate degree of freedom be set to zero at the interface node of the refined element. Further details on adaptive $p$ and $hp$ strategies are provided in [60].

# 5 Software Frameworks

The use of more sophisticated algorithms such as those with complex data structures for grid adaption has increased the complexity of the associated software. In addition there is a desire from the applications area to be able to handle a more diverse class of problems in more general settings (variable spatial dimensions, coupled fields, etc.). Finally, and this is particularly the case for the device problem, the formulation, algorithms and software should be easily extensible to treat new models or a gradation of models. We have seen that, depending on the application, one may wish to solve in order of increasing complexity the potential problem, the drift-diffusion system, the hydrodynamic system, quantum hydrodynamic systems or higher order models. Moreover, the "constitutive" models for mobility (augmented mobility models), inversion layer treatment, thermal closure etc. should be encompassed within the one software framework. These requirements imply a concomitant demand on the software design and the use of higher level programming languages and tools to facilitate a flexible, extensible package.

The use of object-oriented software paradigms certainly facilitates design of such systems

[25, 28, 30, 80] and this approach has been applied in both the commercial and university sectors to varying degrees (Avanti, Floods, Maple, Mathematica) [16, 51]. The Stanford University ALAMODE simulator [79] is another example of an object oriented dial-an-operator tool for TCAD simulations.

Symbolic manipulators have also been receiving increasing attention as a mechanism for expressing differential equations explicitly in software using, for instance, Mathematica or Maple. While symbolic manipulation does incur a modest overhead, it facilitates design of a framework which can accommodate a broad applications set. This implies that changes to the differential equation system may often be possible directly in the higher level symbolic language without affecting the discretization procedure, data structure and solvers.

Part of our recent work on device simulation has involved the industrial simulator PROPHET. This software framework was originally developed for semiconductor process simulation and we have been collaborating with the Lucent developers and colleagues at Stanford on the extension of the capability to device analysis [67]. One long-term goal is to provide a single integrated framework for both process and device simulation. PROPHET embodies some of the features mentioned above - in particular, it provides a "dial-an-operator" capability that allows the user to "build" a differential equation system. This implies that the analyst may even construct mathematical models for other classes of applications beyond the process and device problems of immediate interest [59].

Of course, the ability to handle very general differential systems at the symbolic level presumes that individual differential operators such as div, grad, curl and integral operators can be discretized appropriately. This, in turn, places considerable demand on the data structure at the next lower level. Some operators will require element or cell information, others edge or patch information and so on. For example, discretization strategies for stabilization such as streamline upwinding or exponential weighting via a local Green's function may use patch or edge data structures in specific ways.

The basic strategy used in PROPHET consists of decomposing equations into terms, and treating each term as a combination of a geometric and a physical operator. New

application models can be treated by either combining the predefined geometric and physical operators to construct new PDE systems, or by creating new physical operators via a well-defined interface. The package also includes a database library which enables easy access to any coefficients, parameter values or other properties that pertain to pre-configuring the supported applications. More comprehensive details regarding the set up of PDE systems and the structure of the database library are discussed in the references [65, 66].

The analyst may 'interact' with the PROPHET framework in three main modes: (1) at the top-most application level; (2) at the middle "dial-an-operator" level; and (3) at the lower-most discretization level. For instance, adding the new drift capability (grad operator) for the device problem using the mapped S-G approach on edges requires expanding the discretization capability at the lowest level, whereas modifying the differential equations with operators from the existing library involves the mid level and simpler parameter changes involve only the top level. An example of a MOSFET simulation with PROPHET is given in Figure 4. Further details are provided in [58].

Over the past three years the SIERRA C++ framework [75] has been under development at Sandia National Laboratories as part of the U.S. Department of Energy's Accelerated Strategic Computing Initiative (ASCI). The goals of this project are to provide software support services that are common to finite element applications. An important aspect of this effort is the construction of a set of high-level abstractions that allows the details of services such as adaptive mesh refinement, cache management, message-passing, linear solvers, and so forth to be hidden from the applications developer.

The basic paradigm is that a finite element code is a set of nested computational mechanics objects. The highest level, called a domain, is essentially a container for one or more procedures that manages the time integration of a set of regions. Each region is responsible for solving nonlinear sets of strongly coupled equations at a single timestep. Different sets of physics correspond to loosely coupled regions. In turn, the region contains lower-level element mechanics, which perform the actual element integrations, etc. At the lowest level, the element mechanics may contain nested material mechanics that provide the constitutive

relations such as stress-strain, thermal conductivity, etc. Boundary conditions are typically implemented as mechanics classes that are owned by the region and hence are peers of the element mechanics.

In this way, separate physics may be loosely coupled at the procedure level, by defining a region that contains each distinct set. For example, a microelectromechanical systems (MEMS) problem might consist of several regions: one (or more) in which fluid motion is modeled, another in which structural deformation is modeled, a region for electromagnetic field calculations and regions for heat transfer and radiation modelling. Data transfer among regions occurs at the procedure level via abstract transfer objects that hide the details of the mesh projections from the developer. Regions may overlap and it is not necessary that the two meshes align. As another example, a transistor might be modeled as two non-overlapping regions: the first could be the oxide, in which the nonlinear Poisson equation is solved for the electric field, and the second might contain the rest of the device, in which the coupled Poisson equation and the charge transport equations are solved. Alternatively, if Gummel iteration is used to decouple the electrostatic potential and transport calculations, then the first region corresponding to the nonlinear Poisson equation could span the entire device, and thus overlap with the second, in which only transport equations are solved. In this case, the transfer object would pass the electric field from the first region to the second, and also pass the carrier concentrations from the second region to the Poisson region. In this latter strategy, adaptive refinement could be used to obtain two separate meshes in the semiconductor, one optimized for the potential solution and the other optimized for the concentration solution. The details of determining the mesh intersections on distributed memory computers and projecting solutions from one mesh to another are transparent to the application developer.

# 6 Parallel and distributed computing

The need for more sophisticated physical models leading to larger coupled PDE systems together with the requirement of high resolution grids has fostered interest in parallel de-

vice simulation [23, 76]. Traditionally, the device designer has depended on uniprocessor workstation capability, but the recent emergence of multithreaded parallel shared-memory workstations and of tightly-coupled distributed parallel PC workstation clusters provides an inexpensive means of scaling up the application and reducing run time [38].

A standard approach for parallelizing PDE simulation is via domain decomposition: The domain and grid are partitioned to a set of subdomains and corresponding subdomain grids. Both overlapping and non-overlapping domain decomposition strategies are applicable. Typically, the subdomain calculations are carried out on their respective processors with overlap or interface communication using MPI between adjacent processors [1, 11, 23, 38]. To date there has been little use of these ideas for device and process simulation except in a few university research studies [61, 77], but but they are now widely used for other engineering applications. We will see more widespread interest and use in the near future, particularly as software frameworks are developed to support parallelism for process and device simulation.

We are particularly interested in parallel solution strategies that can accommodate unstructured grids. Hence, it is important that the grid partitioning problem be efficiently treated and that the resulting partition have good load balancing properties for parallel computation. Sandia software package CHACO [50] provides several algorithms of varying complexity ranging from simple inertial bisection to more costly spectral schemes. These approaches have been continued in the METIS software which is widely available [48]. Both software systems provide effective means for partitioning unstructured grids in applications to the stationary device equations.

A representative example is included here from a recent study. The test problem was the n-channel depleted MOSFET described in [11] using the drift-diffusion model on a mesh of 7722 triangles generated by Delaunay triangulation. A partitioning of the unstructured grid to 6 subdomains is indicated in Figure 5. See [11] for numerical results as well as more details of the algorithm and implementation.

# 7 Concluding remarks

Advances in models, methodology, software and computer processing hardware continue to enhance our ability to provide reliable and detailed simulation capabilities for device analysis and design. However, the continuing trends to shrink technology make this area one of great challenge in all these respects. At the same time, the need to shorten the design cycle and accelerate delivery of new products to market places a greater weight on the use of simulation technology. There is ample evidence that increased research and development funding at the universities and in industry is needed in this area if it is to achieve these goals. It is also clear that in many respects semiconductor modeling and simulation lags similar work in other engineering areas and that this problem is increasing. It is surprising that this situation has arisen, given the importance of microelectronics to the high technology and the information Technology infrastructure.

# Acknowledgments

This work has been supported in part by NSF Grant 791AT-51067A under the NPACI program and by ASCI grant B347883. We would like to express our appreciation to R. Dutton, Conor Rafferty, Z. Ping Yu and A. Tasch for their comments and encouragement of this endeavor. Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under contract DE-AL04-98AL85000.

# References

[1] N. R. Aluru, K. H. Law, and R. W. Dutton. Simulation of the hydrodynamic device model on distributed memory parallel computers. *IEEE Trans. CAD*, 15:1029–1047, 1996.

[2] N. R. Aluru, K. H. Law, P. M. Pinsky, A. Raefsky, R. J. G. Goossens, and R. W. Dutton. Space-time galerkin/least-squares finite element formulation for the hydrodynamic

device equations. *IEICE Transactions on Electronics*, E77-C(2):227–235, 1994.

[3] A. K. Aziz and P. Monk. Continuous finite elements in space and time for the heat equation. *Math. Comp.*, 52:255–274, 1989.

[4] R. E. Bank. *PLTMG: A Software Package for Solving Elliptic Partial Differential Equations, User's Guide 6.0.* SIAM, Philadelphia, 1990.

[5] R. E. Bank, D. J. Rose, and W. Fichtner. Numerical methods for semiconductor device simulation. *IEEE Transactions on Electron Devices*, ED-30(9):1031–1041, 1983.

[6] K. S. Bey and J. T. Oden. Hp-version discontinuous galerkin methods for hyperbolic conservation law. *Computer Methods in Applied Mechanics & Engineering*, 133(3-4):259–286, 1996.

[7] T. J. Bordelon, X.-L. Wang, C. M. Maziar, and A. F. Tasch. An efficient non-parabolic formulation of the hydrodynamic model for silicon device simulation. *IEDM Technical Digest*, pages 353–356, 1990.

[8] T. J. Bordelon, X.-L. Wang, C. M. Maziar, and A. F. Tasch. Accounting for band-structure effects in the hydrodynamic model: a first-order approach for silicon device simulation. *Solid-State Electronics*, 35(2):131–139, 1992.

[9] S. Bova and G. F. Carey. A Taylor-Galerkin finite element method for the hydrodynamic semiconductor equations. *IEEE Trans. CAD*, 14(12):1437–1444, 1995.

[10] S. W. Bova. *Finite Element Solution of Hyperbolic Transport Systems Using Adaptive Methods.* PhD thesis, University of Texas at Austin, Austin, TX, 1994.

[11] S. W. Bova and G. F. Carey. A distributed memory parallel element-by-element scheme for semiconductor device simulation. *Computer Methods in Applied Mechanics and Engineering*, 181:403–423, 1999.

[12] G. F. Carey. Exponential upwinding and integrating factors for symmetrization. *Communications in Applied Numerical Methods*, 1(2):57–60, 1985.

[13] G. F. Carey. *Computational Grids: Generation, Adaptation, and Solution Strategies.* Taylor and Francis, 1997.

[14] G. F. Carey, A. Pehlivanov, Y. Shen, A. Bose, and K. C. Wang. Least squares finite elements for fluid flow and transport. *International Journal for Numerical Methods in Fluids*, 27:97–107, 1998.

[15] G. F. Carey and A. I. Pehlivanov. Local error estimation and adaptive remeshing scheme for least-squares mixed finite elements. *Computer Methods in Applied Mechanics and Engineering*, 150:125–131, 1997.

[16] G. F. Carey, J. Schmidt, V. Singh, and D. Yelton. A prototype scalable, object-oriented finite element solver on multicomputers. *Journal of Parallel & Distributed Computing*, 20(3):357–379, 1994.

[17] G. F. Carey and M. Sharma. Semiconductor device modelling using flux upwind finite element. *COMPEL*, 8(4):219–224, 1989.

[18] G. F. Carey, M. Sharma, and K. C. Wang. A class of data structures for 2-D and 3-D adaptive mesh refinement. *International Journal for Numerical Methods in Engineering*, 26:2607–2622, 1988.

[19] J. Casper and H. L. Atkins. A finite-volume high-order eno scheme for two-dimensional hyperbolic systems. *Journal of Computational Physics*, 106(1):62–76, 1993.

[20] T. Chen, D. W. Yergeau, and R. W. Dutton. Efficient 3d mesh adaptation in diffusion simulation. *SISPAD '96 Proceedings*, pages pp.171–172, 1996.

[21] B. Cockburn and Chi-Wang Shu. Tvb runge-kutta local projection discontinuous galerkin finite element method for conservation laws ii. general framework. *Mathematics of Computation*, 52(186):411–435, 1989.

[22] B. Cockburn and Chi-Wang Shu. The local discontinuous galerkin method for time-dependent convection-diffusion system. *SIAM Journal on Numerical Analysis*, 35(6), 1998.

[23] R. K. Coomer and I. G. Graham. Massively parallel methods for semiconductor device modelling. *Computing*, 56(1):1–27, 1996.

[24] B. Davis and G. F. Carey. Multilevel solution of augmented drift-diffusion equations. *COMPEL*, 15(2):4–18, 1996.

[25] Y. Dubois-Pelerin and T. Zimmermann. Object-oriented finite element programming: Iii. an efficient implementation in c++. *Computer Methods in Applied Mechanics & Engineering*, 108(1-2):165–183, 1993.

[26] R. W. Dutton and E. C. Kan. Hierarchical process simulation for nano-electronic. *VLSI Design*, 6:385–391, 1998.

[27] R. W. Dutton and Z. Yu. *IC Processes and Devices.* Kluwer, Boston, 1993.

[28] D. Eyheramendy and T. Zimmermann. Object oriented finite element programming: an interactive environment for symbolic derivations, application to an initial boundary value problem. *Advances in Engineering Software*, 27(1-2):3–10, 1996.

[29] E. Fatemi, C. L. Gardner, J. W. Jerome, S. Osher, and D. J. Rose. Simulation of a steady-state electron shock wave in a submicron semiconductor device using high-order upwind methods. In *Computational Electronics: Semiconductor Transport and Device Simulation.* Kluwer Academic Publishers, Boston, 1991.

[30] B. W. R. Forde, R. O. Foschi, and S. F. Stiemer. Object-oriented finite element analysis. *Computers & Structures*, 34(3):355–374, 1990.

[31] F. Brezzi; M. Fortin. *Mixed and hybrid finite element methods.* Springer-Verlag, New York, 1991.

[32] C. L. Gardner. Numerical simulation of a steady-state electron shock wave in a submicrometer semiconductor device. *IEEE Transactions on Electron Devices*, 38(2):392–398, 1991.

[33] C. L. Gardner. The quantum hydrodynamic model for semiconductor devices. *SIAM J. Applied Mathematics*, 54(2):409–427, 1994.

[34] C. L. Gardner, J. W. Jerome, and D. J. Rose. Numerical methods for the hydrodynamic device model: subsonic flow. *IEEE Trans. CAD*, 8(5):501–507, 1989.

[35] C. L. Gardner, J. W. Jerome, and Chi-Wang Shu. The eno method for the hydrodynamic model for semiconductor devices. In A. M. Tentner, editor, *Grand Challenges in Computer Simulation: Proceedings of the 1993 High Performance Computing Symposium*, pages 96–101, 1993.

[36] S. K. Godunov and V. S. Ryabenkii. *Theory of Difference Schemes: An Introduction.* North-Holland, Amsterdam, 1964. Translated by E. Godfredsen.

[37] N. A. Golias and R. W. Dutton. Delaunay triangulation and 3d adaptive mesh generation. *Finite Elements in Analysis & Design*, 25(3-4):331–341, 1997.

[38] W. Gropp, E. Lusk, and A. Skjellum. *Using MPI: portable parallel programming with the message-passing interface.* MIT Press, Cambridge, Mass., 1994.

[39] H. L. Grubin and J. P. Kreskovsky. Quantum moment balance equations and resonant tunneling structures. *Solid State Elecron.*, 32:1971–1975, 1989.

[40] H. K. Gummel. A self-consistent iterative scheme for one-dimensional steady state transistor calculations. *IEEE Transactions on Electron Devices*, ED-11:455–465, 1964.

[41] S. Harland, M. Manassian, W-K. Shih, S. Jallepalli, H. Wang, G. L. Chindalore, A. F. Tasch, and C. M. Maziar. Computationally efficient models for quantization effects in mos electron and hole accumulation layers. *IEEE Transactions on Electron Devices*, 45(7):1487–1493, 1998.

[42] S. Harland, A. F. Tasch, and C. M. Maziar. A new structural approach for reducing punchthrough current in deep submicron MOSFETs and extending MOSFET scaling. *Electronics Letters*, 29:1894–1896, 1993.

[43] N. Hitschfield, P. Conti, and W. Fichtner. Mixed element trees: a generalization of modified octrees for the generation of meshes for the simulation of complex 3-D semiconductor device structures. *IEEE Trans. CAD*, 12(11):1714, 1993.

[44] T. J. R. Hughes and A. Brooks. A theoretical framework for Petrov-Galerkin methods with discontinuous weighting functions – application to the streamline upwind procedure. In Gallagher et al, editor, *Finite Elements in Fluids*, volume 55, pages 47–65. 1982.

[45] T. J. R. Hughes, L. P. Franca, and G. M. Hulbert. A new finite element formulation for computational fluid dynamics: the galerkin/least squares method for advective diffusive systems. *Computer Methods in Applied Mechanics and Engineering*, 58:173–189, 1986.

[46] C. J. Hwang and S. Y. Yang. Locally implicit total variation diminishing schemes on mixed quadrilateral-triangular meshes. *AIAA Journal*, 31:2008–2015, 1993.

[47] O. Karakashian and C. Makridakis. A space-time finite element method for the nonlinear schrodinger equation: the discontinuous galerkin method. *Mathematics of Computation*, 67(222):479–499, 1998.

[48] G. Karypis and V. Kumar. Fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, 20(1):359–392, 1998.

[49] P. Lax. Hyperbolic systems of conservation laws and the mathematical theory of shock waves. In *SIAM*. Philadelphia, PA, 1972.

[50] R. Leland and B. Hendrickson. Empirical study of static load balancing algorithms. In *Proceedings of the Scalable High-Performance Computing Conference*, pages 682–685, 1994.

[51] M. Liang and M. E. Law. An object-oriented approach to device simulation - FLOODS. *IEEE Trans. Computer-Aided Design*, 13(10):1235–1240, 1994.

[52] M. Lundstrom. *Transport Fundamentals for Device Applications.* Addison-Wesley, Reading, MA, 1990.

[53] P. Markowich. *The Stationary Semiconductor Device Equations.* Springer-Verlag, Wien, Austria, 1986.

[54] P. Markowich, C. Ringhofer, and S. Selberherr. A singular perturbation approach for the analysis of the fundamental semiconductor equations. *IEEE Transactions on Electron Devices*, ED-30:1165–1180, 1983.

[55] M. S. Mock. *Analysis of Mathematical Models of Semiconductor Devices.* Boole press, Dublin, 1983.

[56] S. Osher and S. Charkravarthy. Upwind schemes and boundary conditions with applications to Euler equations in general geometries. *Journal of Computational Physics*, 50:445–481, 1983.

[57] A. L. Pardhanani and G. F. Carey. A mapped scharfetter-gummel formulation for the efficient simulation of semiconductor device models. *IEEE Trans. CAD*, 16(10):1227–1233, 1997.

[58] A. L. Pardhanani and G. F. Carey. Mapped discretization strategies for curvilinear adaptively redistributed grids in semiconductor device modeling. *Computer Methods in Applied Mechanics and Engineering*, 1999. (in press).

[59] A. L. Pardhanani and G. F. Carey. Multidimensional semiconductor device and microscale thermal modeling using the prophet simulator with dial-an-operator framework. *Computer Modeling in Engineering and Science*, 1999. (in press).

[60] A. Patra and J. T. Oden. Computational techniques for adaptive hp finite element methods. *Finite Elements in Analysis & Design*, 25(1-2):27–39, 1997.

[61] T. F. Pena, E. L. Zapata, and D. J. Evans. Finite element simulation of semiconductor devices on multiprocessor computers. *Parallel Computin*, 20(8):1129–1159, 1994.

[62] A. Plaza and G. F. Carey. Local refinement of simplicial grids based on the skeleton. *Applied Numerical Mathematics*, 1999. (in press).

[63] A. Plaza, M. A. Padron, and G. F. Carey. A 3d refinement/derefinement algorithm for solving evolution problems. *Applied Numerical Mathematics.* (in press).

[64] P. H. Rabinowitz. *Applications of Biffurcation Theory.* Academic Press, New York, 1977.

[65] C. S. Rafferty. Programmer's guide to the prophet database. Technical memorandum, Bell Laboratories, Lucent Technologies, 1996.

[66] C. S. Rafferty and R. K. Smith. Solving partial differential equations with the prophet simulator. Technical memorandum, Bell Laboratories, Lucent Technologies, 1996.

[67] C. S. Rafferty, Zhiping Yu, A. L. Pardhanani, G. F. Carey, and R. W. Dutton. Semiconductor device simulation using prophet. Ticam report, University of Texas at Austin, Austin, TX, 1999. (in preparation).

[68] D. L. Scharfetter and H. K. Gummel. Large-signal analysis of a silicon read diode oscillator. *IEEE Transactions on Electron Devices*, ED-16(1):64–77, 1969.

[69] M. Sharma and G. F. Carey. Semiconductor device simulation using adaptive refinement and flux upwinding. *IEEE Trans. CAD*, 8(6):590–598, 1989.

[70] T. Sonar. On families of pointwise optimal finite volume eno approximations. *SIAM Journal on Numerical Analysis*, 35(6), 1998.

[71] W. F. Spotz and G. F. Carey. High-order compact scheme for the stream-function vorticity equations. *International Journal for Numerical Methods in Engineering*, 38(20):3497–3512, 1995.

[72] W. F. Spotz and G. F. Carey. A high-order compact formulation for the 3D Poisson equation. *Numerical Methods for Partial Differential Equations*, 12:235–243, 1996.

[73] P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, 21(5):995–1011, 1984.

[74] S. M. Sze. *Physics of Semiconductor Devices.* John Wiley, 1981.

[75] Lee M. Taylor, H. Carter Edwards, and James R. Stewart. Functional requirements for SIERRA version 1.0 beta. SAND Report SAND99-2587, Sandia National Laboratories, Albuquerque, NM, 1999.

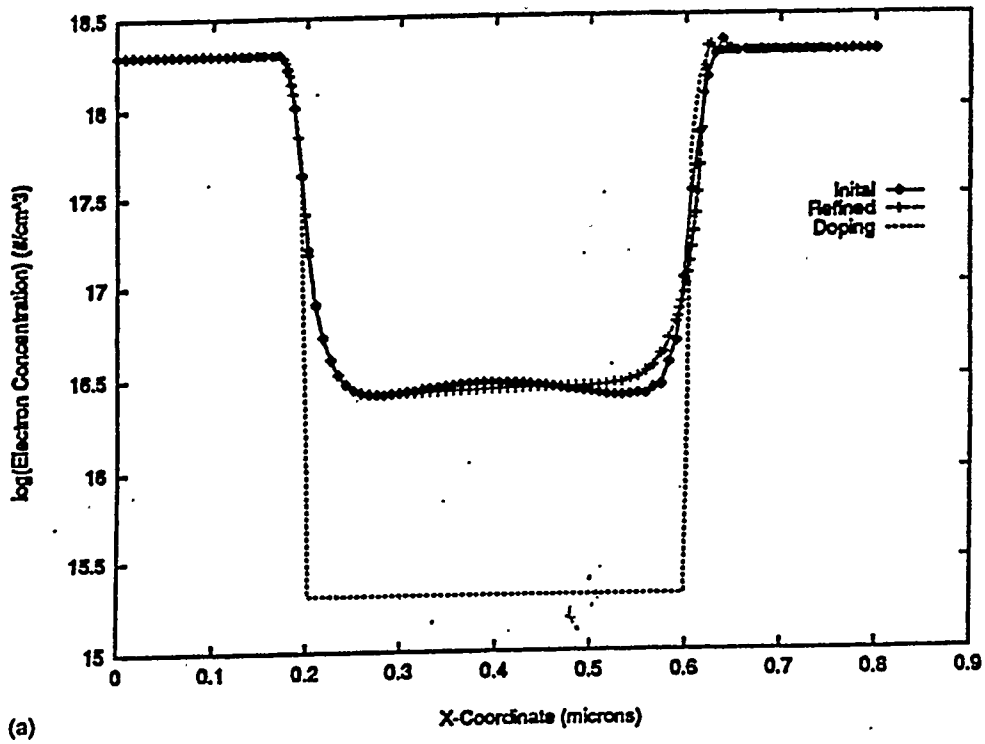
[76] E. Tomacruz, J. V. Sanghavi, and A. Sangiovanni-Vincentelli. Algorithms for drift-diffusion device simulation using massively parallel processors. *IEICE Transactions on Electronics*, E77-C(2):248–254, 1994.

[77] C. S. Tsang-Ping, C. M. Snowden, and D. M. Barry. Parallel implementation of an electrothermal simulation for gaas mesfet devices. *IEEE Trans. CAD*, 15(3):308–316, 1996.

[78] H. C. Yee and A. Harten. Implicit tvd schemes for hyperbolic conservation laws in curvilinear coordinates. *AIAA Journal*, 25:266–274, 1987.

[79] D. W. Yergeau, E. C. Kan, M. J. Gander, and R. W. Dutton. Alamode: A layered model development environment. In *Simulation of Semiconductor Devices and Processes*, pages 66–69, Wien, Austria, 1995.

[80] T. Zimmermann, Y. Dubois-Pelerin, and P. Bomme. Object-oriented finite element programming: I. governing principle. *Computer Methods in Applied Mechanics & Engineering*, 98(2):291–303, 1992.


18.5
18
17.5
17
16.5
16
15.5
15
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
log(Electron Concentration) (#/cm^3)
X-Coordinate (microns)
Initial
Refined
Doping

(a)

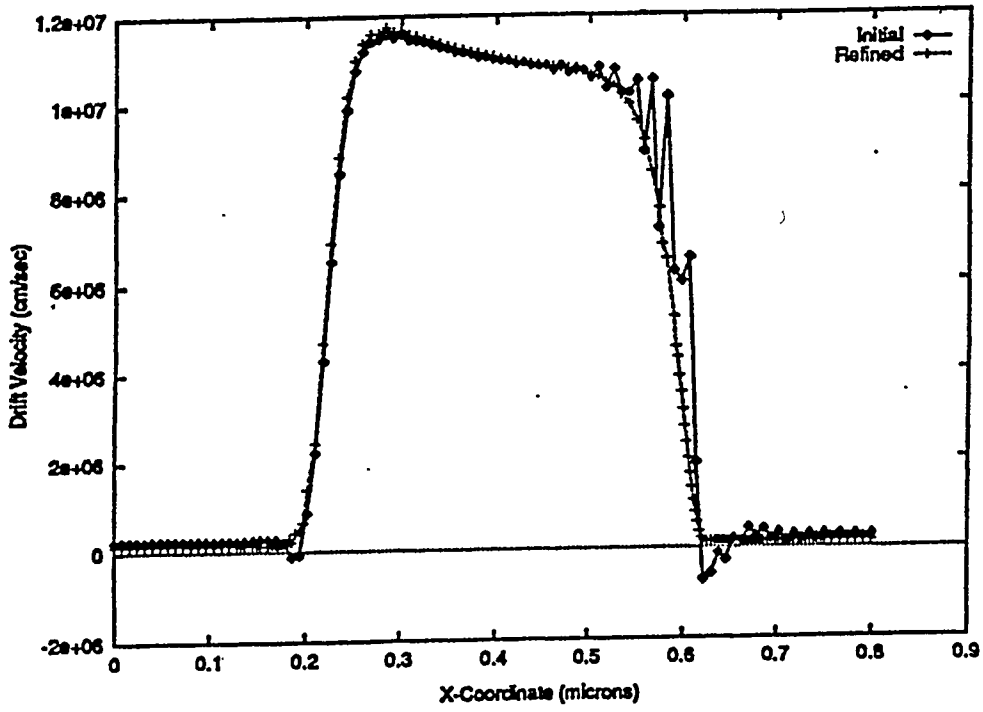

1.2e+07
1e+07
8e+06
6e+06
4e+06
2e+06
0
-2e+06
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
Drift Velocity (cm/sec)
X-Coordinate (microns)
Initial
Refined

Fig 1

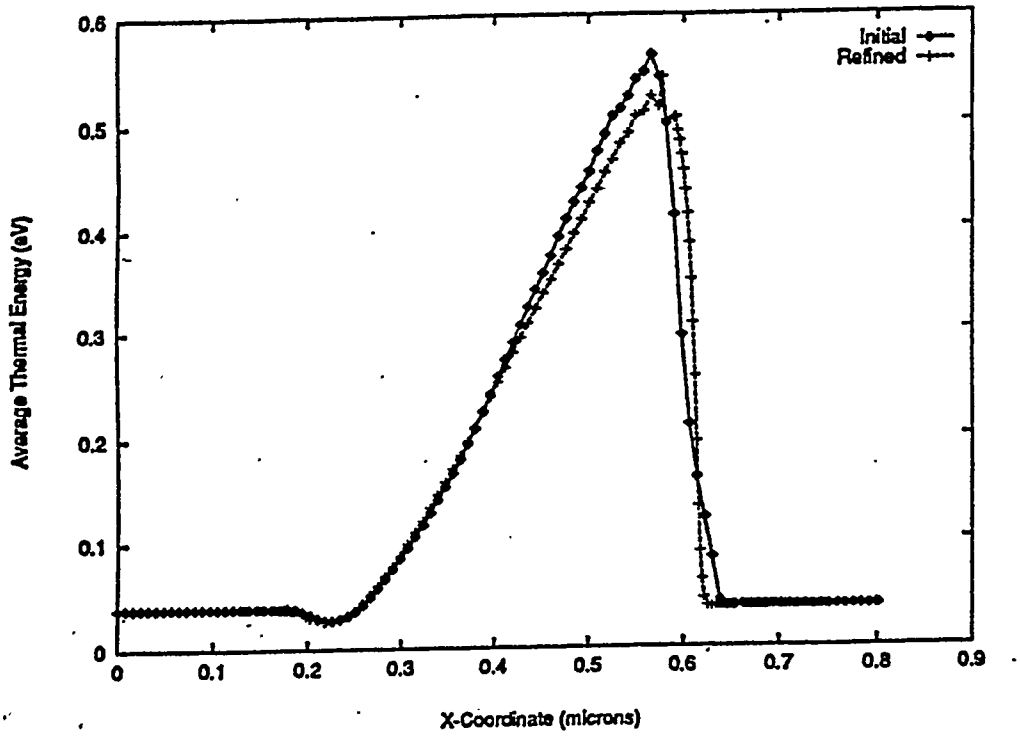
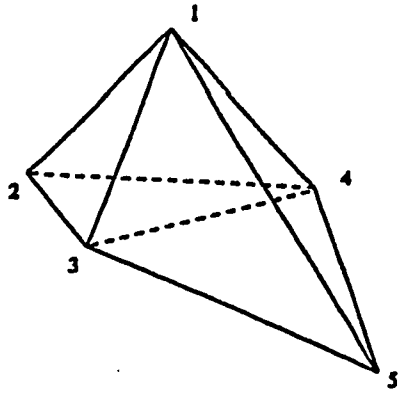
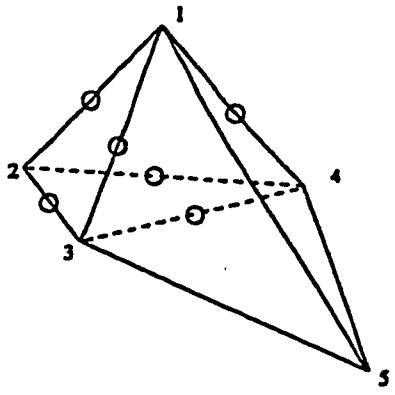

Initial
Refined
Average Thermal Energy (eV)
X-Coordinate (microns)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

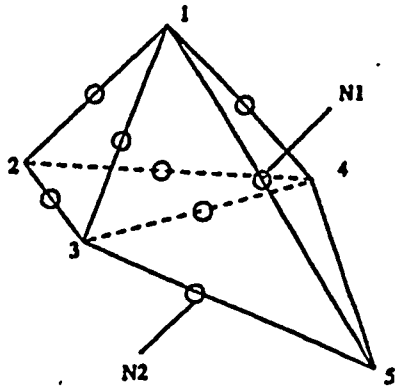
Fig 1 (cont)



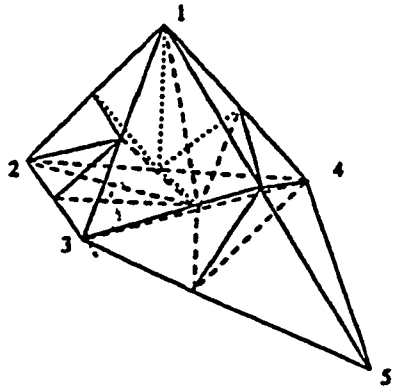

(a) Initial mesh



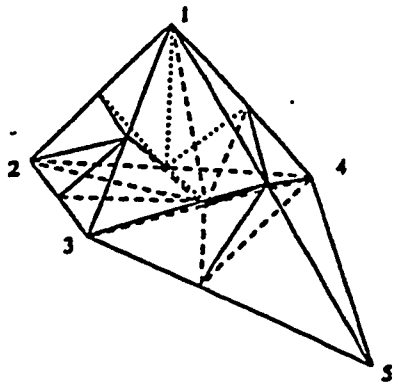

(b) Step 1: Edge subdivision




(c) Step 2: Conformity




(d) Step 3: Skeleton division




(e) Final mesh

Fig 2

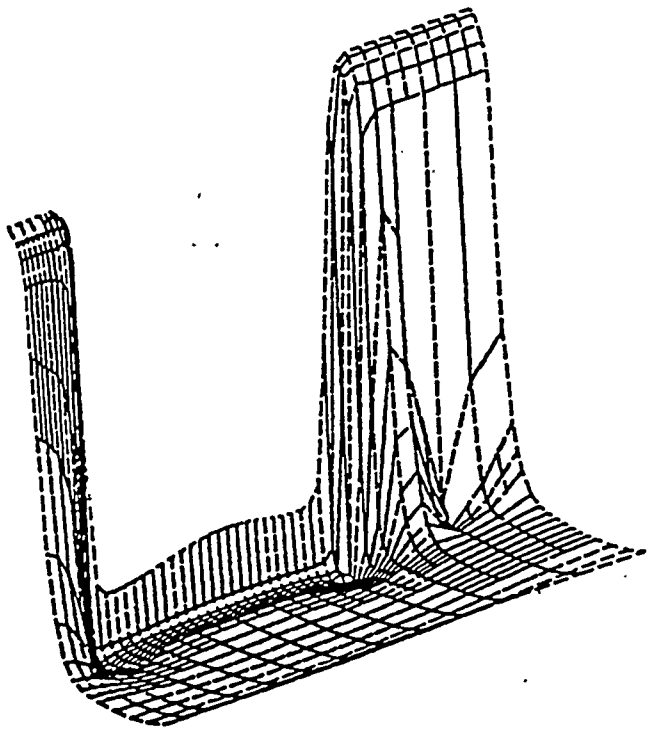


Fig 3

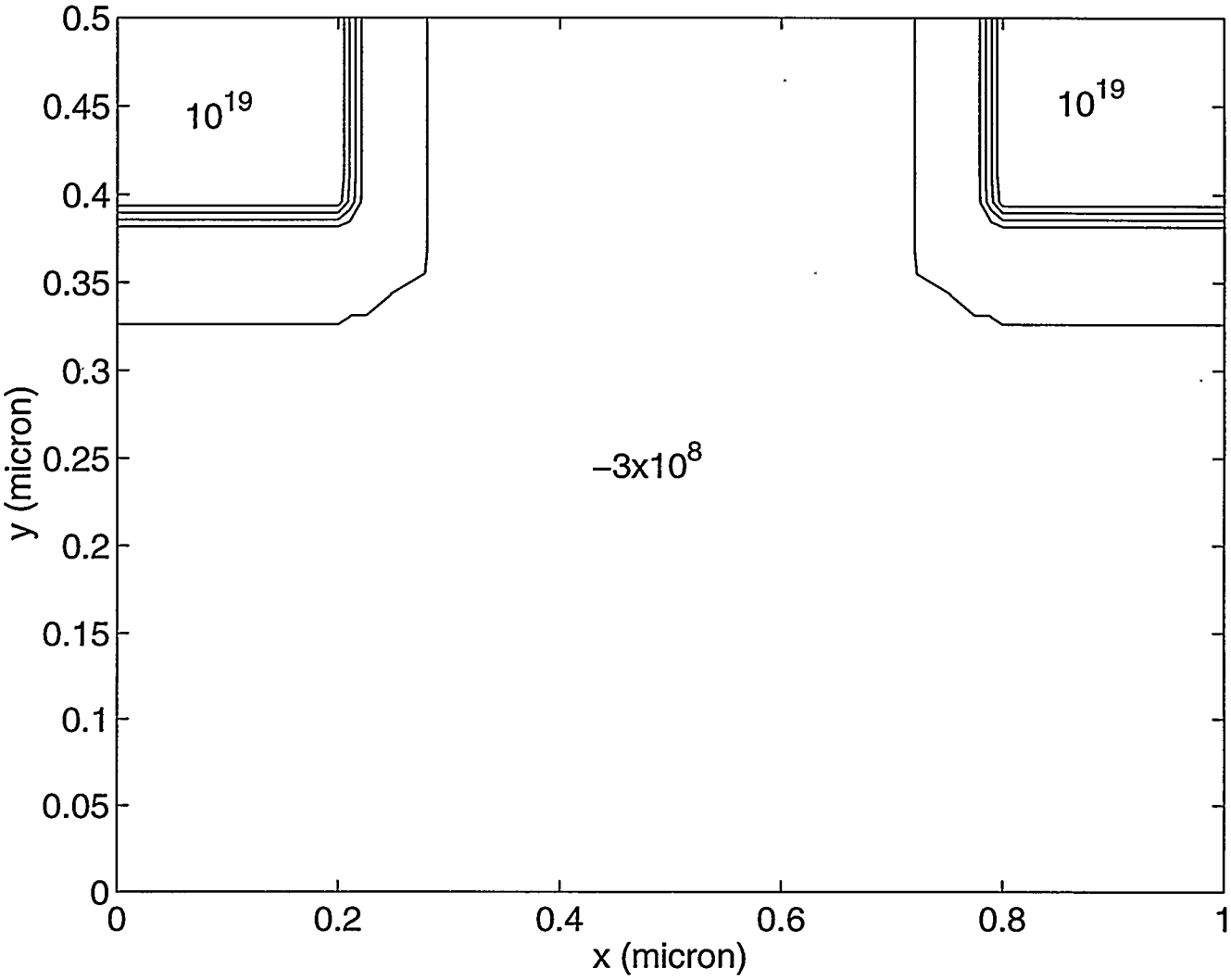

0.5
0.45
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
y (micron)
$10^{19}$
$10^{19}$
$-3x10^{8}$
0
0.2
0.4
0.6
0.8
1
x (micron)

Fig. 4

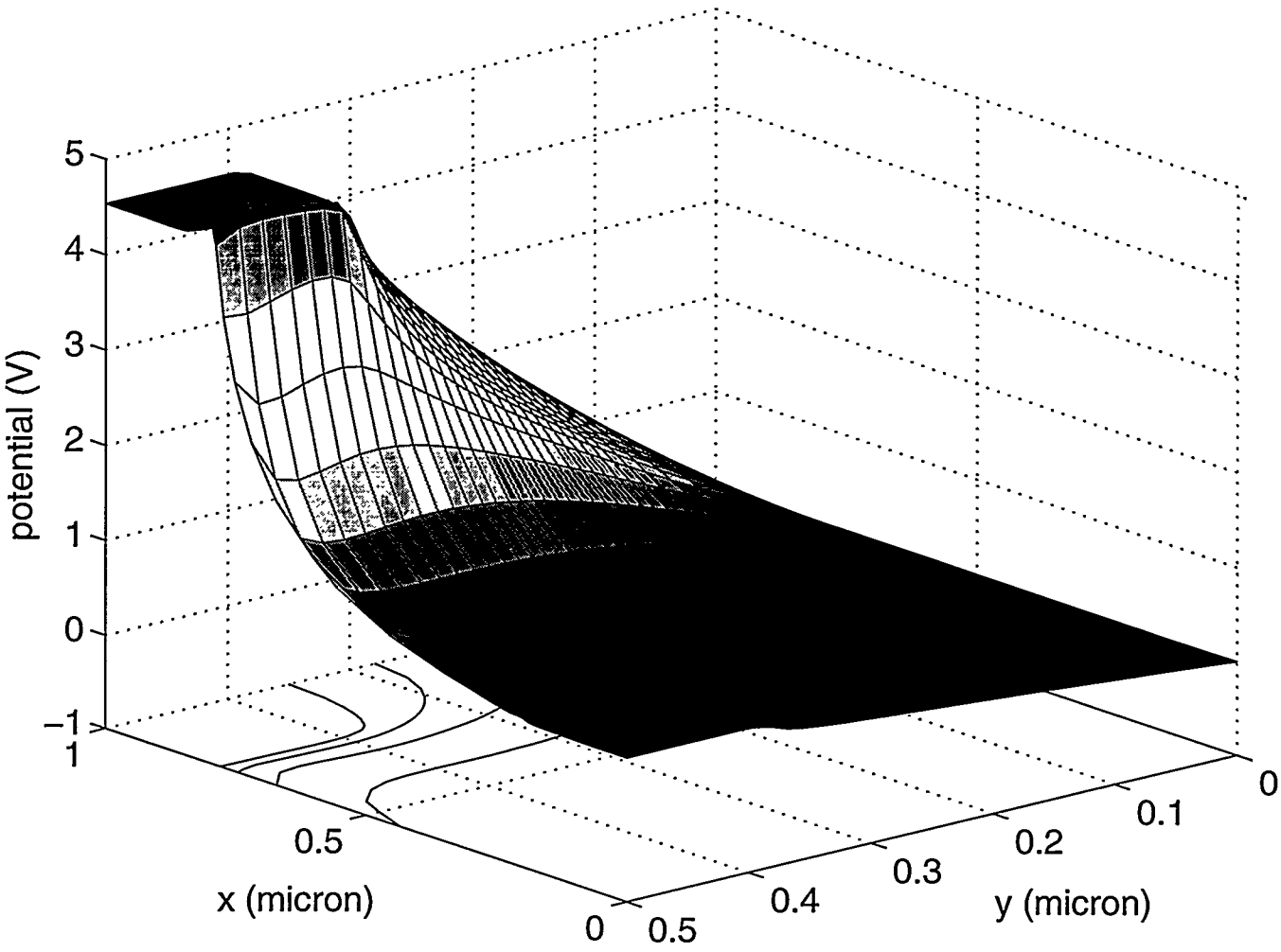



Fig 4 (cont)
b

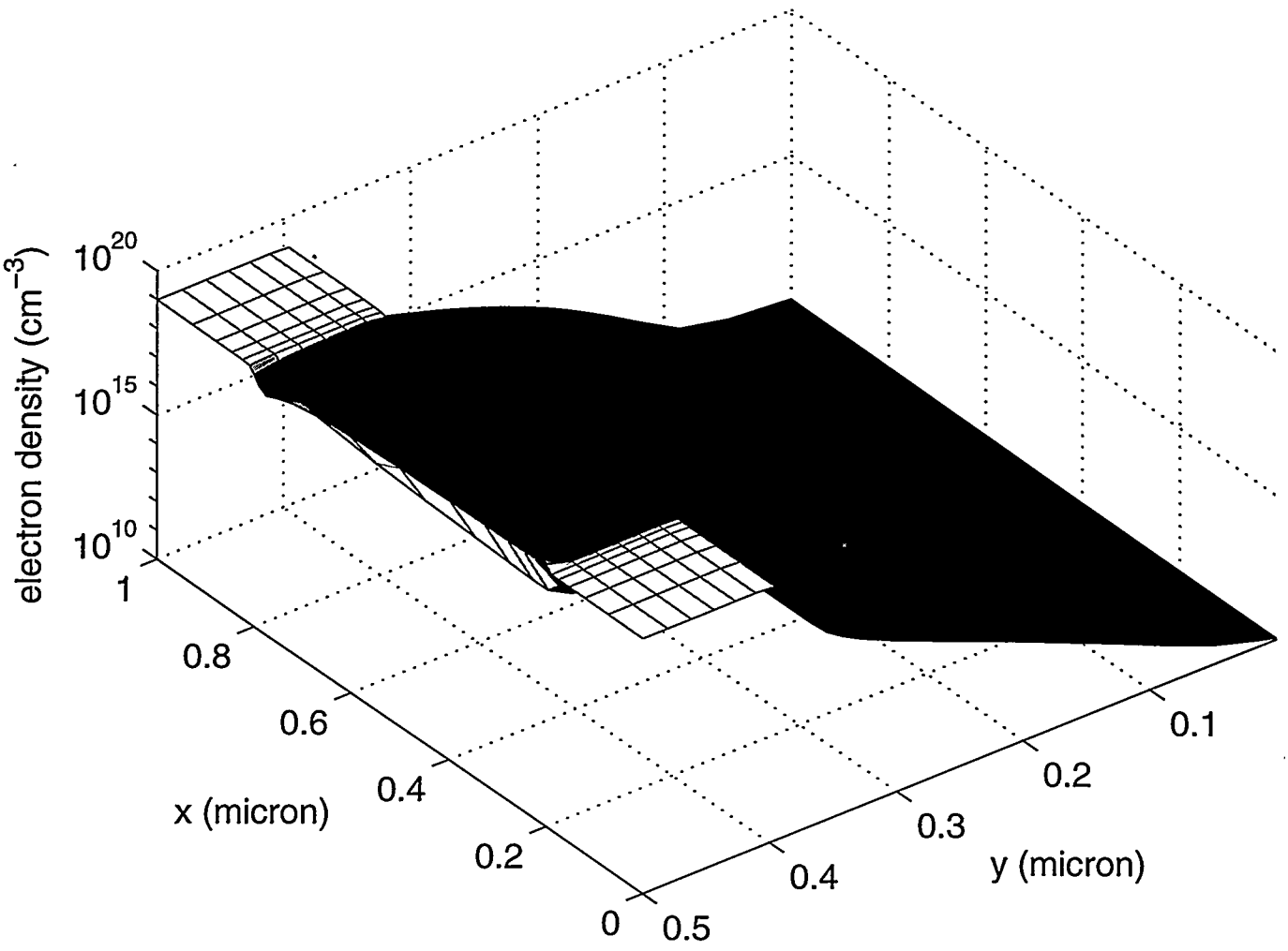

electron density (cm$^{-3}$)
$10^{20}$
$10^{15}$
$10^{10}$
1
0.8
0.6
0.4
0.2
0
x (micron)
0.5
0.4
0.3
0.2
0.1
y (micron)

Fig 4 (cont)
c

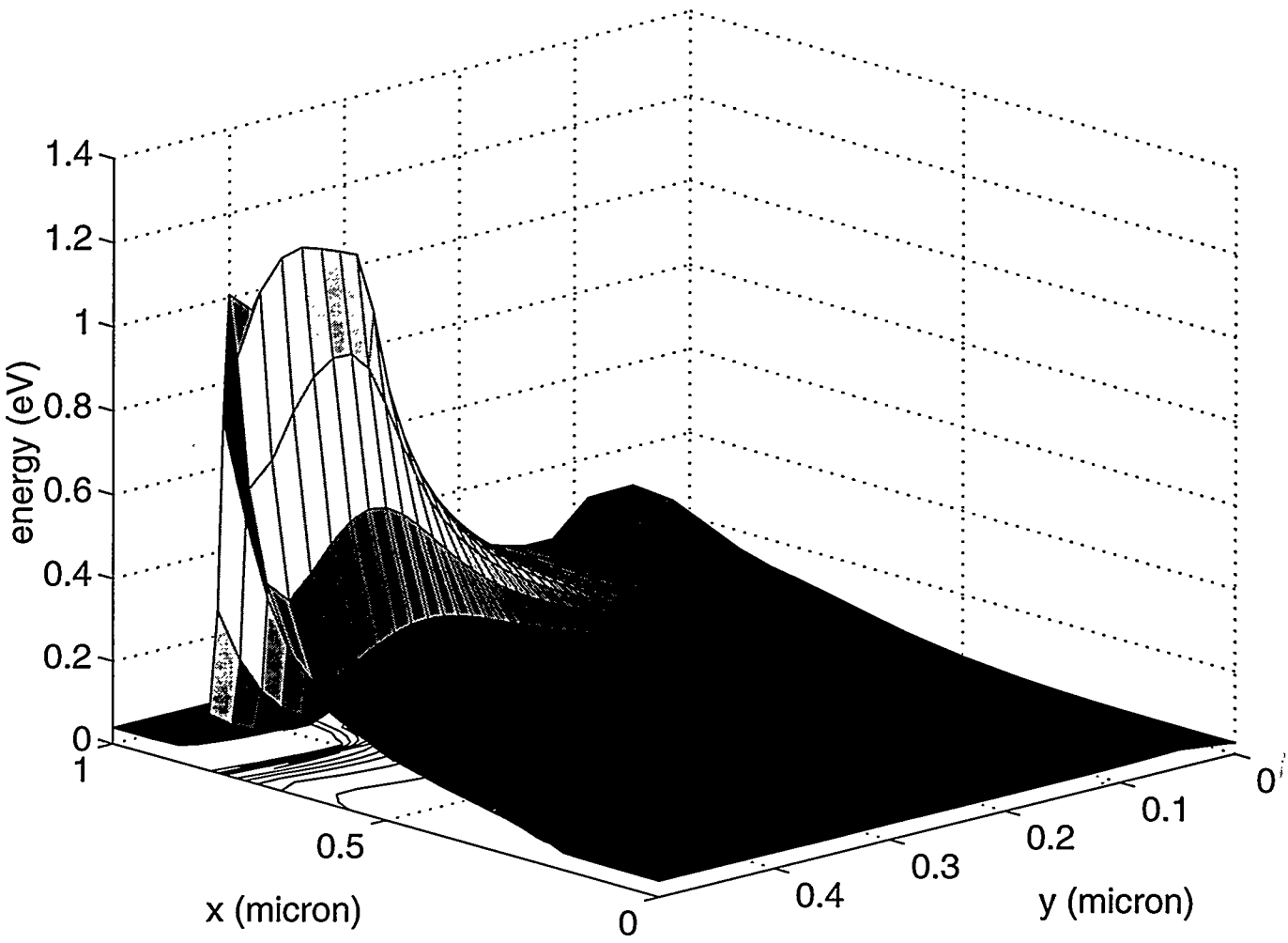



Fig 4 (cont)
d

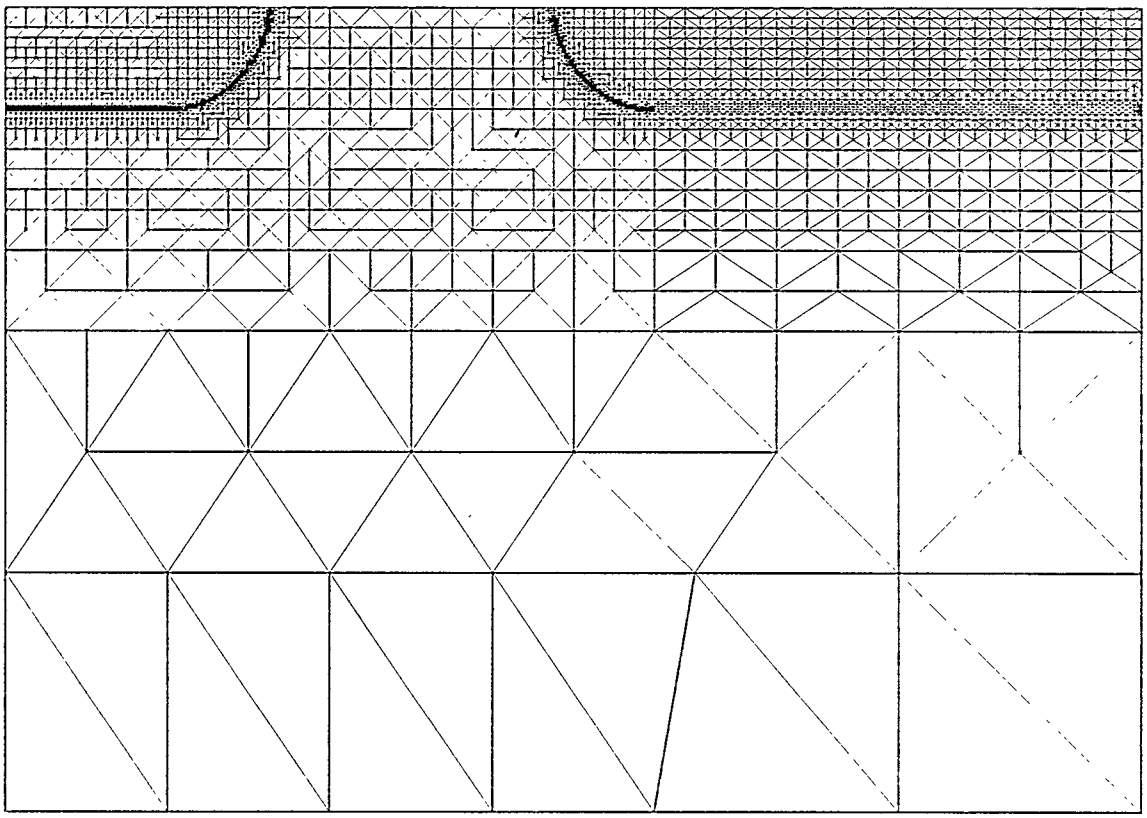


Fig 5 (a)

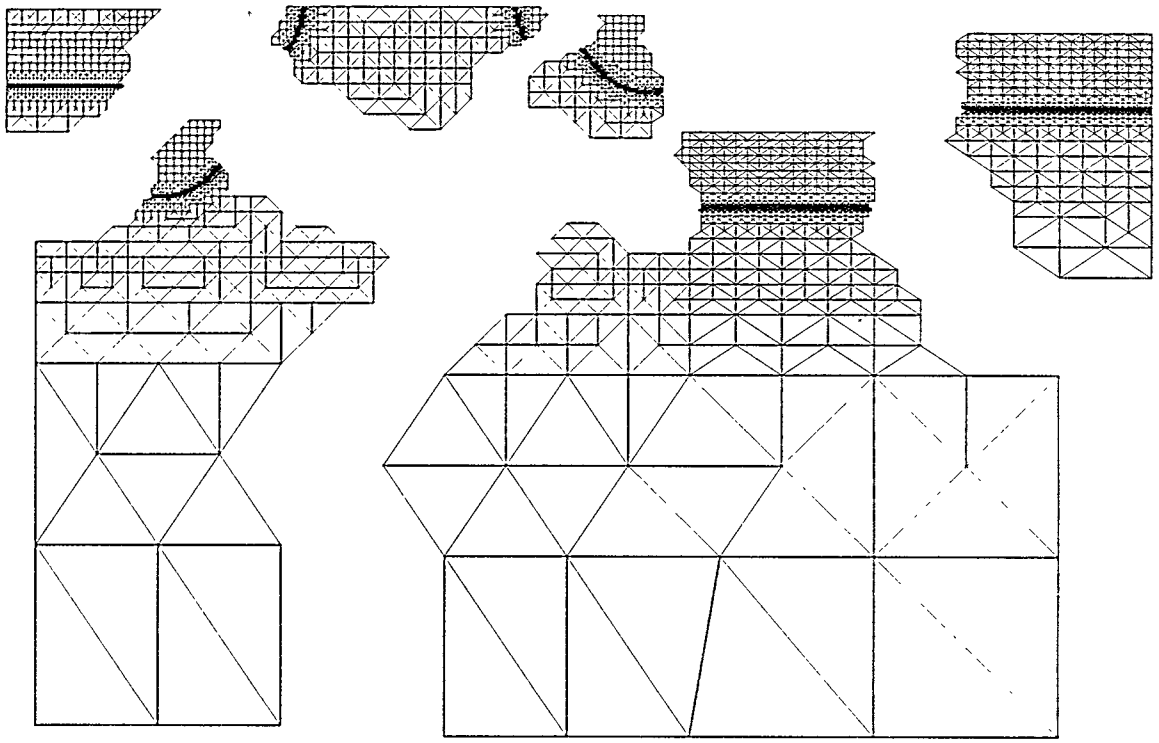


Fig 5(b)

# List of figures