VLSI DESIGN
1998, Vol. 6, Nos. (1–4), pp. 91–95
Reprints available directly from the publisher
Photocopying permitted by license only

Topologically Rectangular Grids in the Parallel Simulation of Semiconductor Devices

A. ASENOV, A.R. BROWN, S. ROY and J.R. BARKER

Department of Electronics and Electrical Engineering, University of Glasgow, Glasgow, G12 8QQ, Scotland, UK

Topologically rectangular grids offer simplicity and efficiency in the design of parallel semiconductor device simulators tailored for mesh connected MIMD platforms. This paper presents several approaches to the generation of topologically rectangular 2D and 3D grids. The effects of the partitioning of such grids on different processor configurations are studied. A simulated annealing algorithm is used to optimise the partitioning of 2D and 3D grids on two dimensional arrays of processors. Problems related to the discretization, parallel matrix generation and solution strategy are discussed. The use of topologically rectangular grids is illustrated through the example of power electronic device simulation.

Keywords: Simulation, Numerical, Semiconductors, Devices, Parallel, Grids

INTRODUCTION

The development of parallel device simulators is a widely accepted method for overcoming the speed and memory restrictions of a single processor system [1]. This is particularly important for 3D problems [2] where the speed of existing 3D single-processor simulators significantly restricts their use in practical device design. To achieve maximum speed-up and efficiency, parallel code design should reflect the architecture of the parallel platform, while remaining scalable and portable in this changing world of parallel systems.

Topologically rectangular (structured) two and three-dimensional grids are attractive for parallel simulation of semiconductor devices because they are easily partitioned onto processor arrays, enhancing the scalability and portability of the simulator code. There are speed-up and efficiency advantages when large topologically rectangular grids are partitioned onto mesh connected arrays of processors using physical domain decomposition instead of logical spectral methods [3].

In this paper we report on different aspects of the implementation of topologically rectangular grids in the parallel simulation of semiconductor devices including grid generation, optimum partitioning, discretization and solution strategy

TOPOLOGICALLY RECTANGULAR GRIDS

Structured topologically rectangular grids allow index ordering preserving the number of grid nodes in each one of the index directions. In addition, nodes with neighbouring indices are physically adjacent in the grid. Most finite difference grids are inherently topologically rectangular. However it is also possible to construct topologically rectangular finite element (FE) grids.

Topologically rectangular grids can be easily partitioned not only onto pipelines but also onto 2D and 3D arrays of mesh connected processors. Such partitioning only requires nearest-neighbour communication in the design of iterative solvers, because adjacent nodes in adjacent partition subdomains appear on neighbouring processors. Topologically rectangular grids lead to a regular band structure in the discretization matrix which may also be advantageous in the design of parallel direct solvers. They are also ideal for the design of multigrid solvers. However because of the constraints of grid generation a topologically rectangular grid may require more nodes than unstructured FE grids.

One simple approach to the generation of topologically rectangular grids is the deformation of an originally rectangular grid to the shape of the solution domain boundaries. Such an approach is used in the FE Monte Carlo simulation of recess gate compound FETs [4]. Fig.1(a,b) illustrates the generation of a quadrilateral grid in the recess region of a MESFET by deformation, following the shape of the recess and the mushroom gate.

A more complex approach can also be used for the generation of the grids. Here, non-equidistantly spaced guiding contours resembling the shape of the domain boundary control the position of the grid nodes. The number of nodes on each guiding contour vary to satisfy the criteria for a topologically rectangular grid. This second approach allows better control over the density of nodes and the shape of the finite elements. The quadrilateral elements of the grid could be further subdivided into two triangles if simplex type triangulation is required. Attention must be paid to avoid obtuse triangles and if necessary some nodes may be slid along the guiding contours. A similar strategy may be applied to 3D grid design where the guiding contours are replaced by guiding surfaces following the domain boundary.

PARTITIONING

In the domain decomposition approach a topologically rectangular 2D grid may be partitioned on 1D (pipeline) or 2D arrays of processors, and a 3D grid may be partitioned on 1D, 2D or 3D arrays of processors. This is done by dividing the number of grid nodes in each index direction by the number of processors available in the processor array in that direction. For a 3D topologically rectangular grid, Fig.2 shows the theoretical speed-up of a hypothetical linear equation solver as a function of problem size for a pipeline, 2D and 3D array of processors. The calculations are based on detailed performance theory [5] with local and global communication times typical for

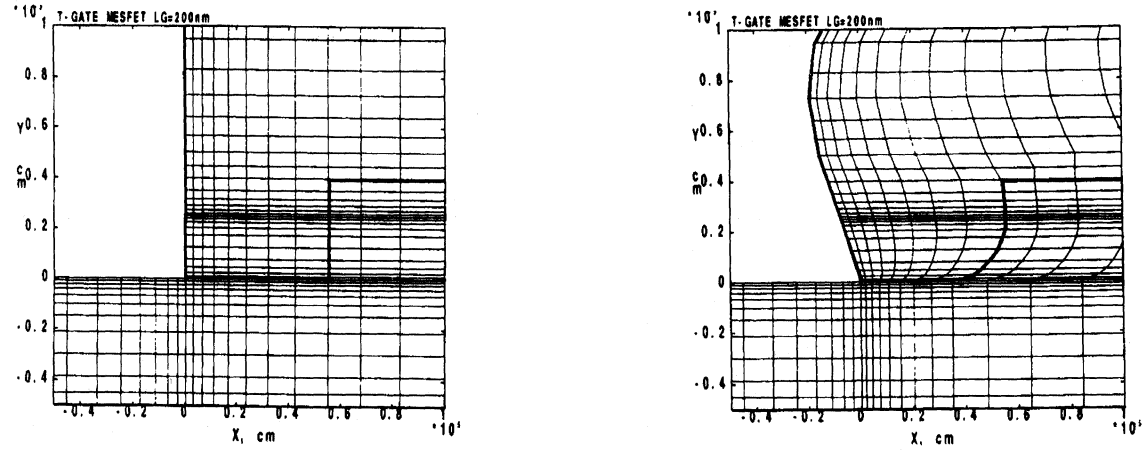

FIGURE 1 Generation of a quadrilateral grid in the recess region of a MESFET

the Parsytec Supercluster Model 64. Clearly the best processor configuration for the domain decomposition of topologically rectangular 3D grids is a 3D array of mesh connected processors. This minimises the ratio between the surface and volume of each subdomain and hence the proportion of communications in the global processing time. However Parsytec systems are physically restricted to 2D processors arrays.

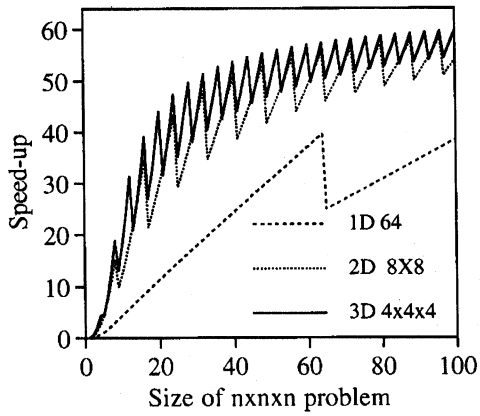

FIGURE 2 The theoretical speed-up of a hypothetical linear equation solver as a function of problem size for a pipeline, 2D and 3D array of processors

Fig. 2 also highlights some of the problems arising from the 'naïve' rectilinear partitioning of the grid. First-order load balancing only occurs when the number of grid nodes is exactly divisible by the number of processors in each dimension of the processor array. Otherwise deep oscillations in speed-up and efficiency occur. We apply a simulated annealing algorithm to improve the first-order load balancing of such a partition. In the case of a 2D array of processors, every grid partitioning (state) has an associated score function $E(V,S_x,S_y,r)$ where V and S are the number of grid points assigned to each partition subdomain and the partition x,y edge lengths respectively. r is the processing/comms ratio for the device simulation implementation. State transformations have been developed (Fig.3) that allow transitions into states which preserve the four way connectivity of the partition subdomains. Transformations are chosen randomly, and only followed if they decrease the score

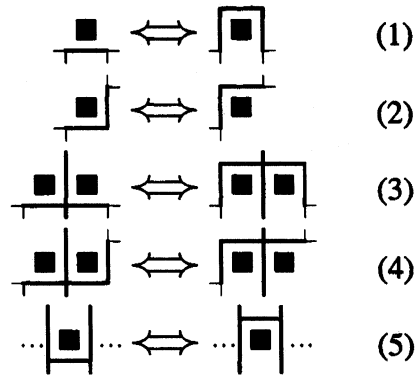

FIGURE 3 Non-repeating partition transformations which preserve processor mesh connectivity. Subdomain boundaries marked by heavy lines

function (i.e. $\Delta E<0$), or if $random(0,1) < \exp(-\Delta E/T)$ (T is the annealing temperature). If transformations merely perturb the score function, and enough of them occur at each T to form a Markov chain in equilibrium, then an appropriate schedule of T reductions will result in convergence towards an optimal final state [6]. Fig.4 shows the annealed partitioning of a 19×19 grid over a 3×3 array of processors. Fig.5 shows the theoretical speed-up of a coloured SOR parallel solver for rectilinear and annealed partitionings as a function of problem size.

To enhance the portability of our parallel device solvers, the code is split in two parts: a hardware

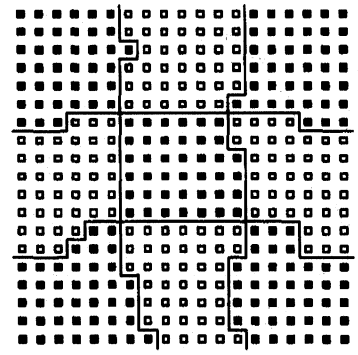

FIGURE 4 Partitioning of a 19×19 grid on a 3×3 array of processors after simulated annealing

dependent communication harness and the simulation engine. The communication harness, written using PARIX message passing primitives, provides all global and local communications between the processors required for the operation of the simulation engine on 1D, 2D and 3D arrays of processors.

PARALLEL DISCRETIZATION AND SOLUTION

The Galerkin finite element approach has been adopted to solve the Poisson equation. The calculation of matrix coefficients is carried out by an isoparametric mapping of the quadrilateral elements into rectangles (in the 2D case) or the distorted brick elements into cubes (in 3D). For parallel matrix generation and assembly a node based approach is used where the solution subdomain on each processor is scanned node by node. For each node only the contributions of the elements to this particular node are calculated. This leads to almost 100% efficiency for annealed partitioning. For the non-linear Poisson equation we use a four colour Block Newton SOR scheme.

A modified control volume approach has been developed for the discretization of the current continuity equation using quadrilateral elements in the 2D case. Alternatively the quadrilaterals may be subdivided into triangles and the standard control volume procedure applied. In the 3D case a control volume approach amenable to distorted bricks is in the process of development. However each distorted brick may be subdivided into six tetrahedral elements and a standard control volume approach applied. A parallel implementation of the BiCGSTAB(2) method with polynomial preconditioning has been adopted for the current continuity equations.

EXAMPLE

An illustration of the use of topologically rectangular grids is the finite element 3D simulation of cellular power IGBTs. The complex shape and doping distribution of these devices requires a 3D finite element discretization. This, together with the computational complexity of power device simulation, makes the problem ideal for parallel processing. Because of symmetry only one quarter of the cell has been discretized for calculation of the in-cell breakdown after punchthrough stopper implantation. The grid at this stage (Fig.6) conforms to the shape of the gate and the metallurgical *p-n* junctions at the cell surface.

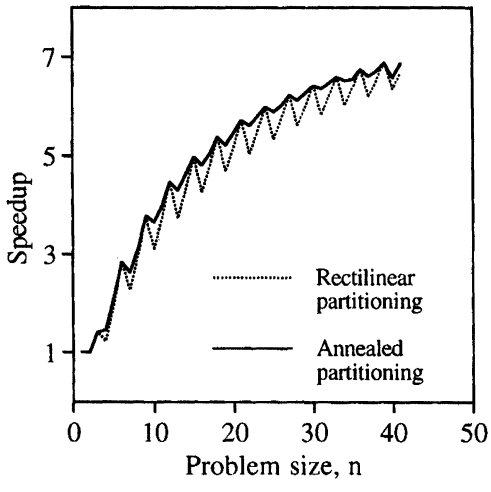

FIGURE 5 Theoretical speed-up of a coloured SOR parallel solver for rectilinear and annealed partitions as a function of problem size

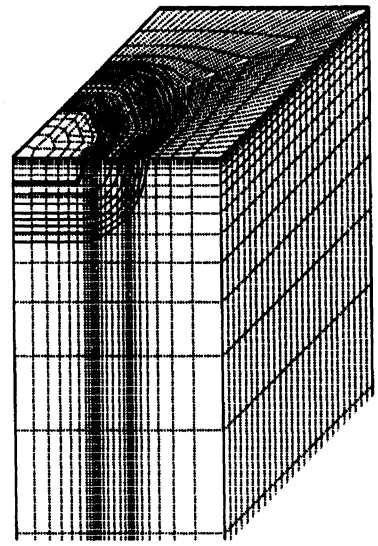

FIGURE 6 Topologically rectangular grid for a cellular IGBT

CONCLUSION

In this paper we have presented a method of using topologically rectangular grids in the simulation of semiconductor devices. Such grids facilitate the development of parallel codes, enhancing their scalability and portability. Two procedures for the automated generation of topologically rectangular grids are outlined. Based on detailed performance theory we have demonstrated that under domain decomposition, optimum performance is achieved when the processor array topology reflects the dimensionality of the simulation problem.

References

[1] T.F. Pana, E.L. Zapata and D.J. Evans, "Finite Element Simulation of Semiconductor Devices on multiprocessor computers", *Parallel Computing,* vol. 20, pp. 1130-1159, 1993.

[2] U.A. Ranawake, C. Huster, P.M. Lenders and S.M. Goodnick, "PMC-3D : A Parallel Three-Dimensional Monte Carlo Semiconductor Device Simulator", *IEEE Trans. Computer Aided Design of IC and Systems,* vol. 13, pp. 712-723, 1994.

[3] N. Frolos, J.S. Reeve, J. Clinkemaillie, S. Vlahoutsis and G. Lonsdale, "Comparative Efficiencies of Domain Decompositions", *Parallel Computing,* July, 1995

[4] S. Babiker, A. Asenov, J.R. Barker and S.P. Beaumont, "Finite Element Monte Carlo Simulation of Recess Gate FETs" in *Simulation of Semiconductor Devices and Processes,* Vol.6, eds. H. Ryssel and P. Pichler, Springer-Verlag, 1995.

[5] A. Asenov, D. Reid and J.R. Barker, "Speed-up of Scalable Iterative Linear Solvers Implemented on an Array of Transputers", *Parallel Computing,* vol. 21, pp. 669-682, 1995

[6] Metropolis, N., *et al.,* "Equation of State Calculations by Fast Computing Machines" *Journal of Chemical Physics,* 1953. vol. 21 : pp. 1087-1092.

Biographies

Asen Asenov had 10 years industrial experience as a head of the Process and Device Modelling Group in IME - Sofia. In 1985 he developed one of the first integrated process and device CMOS simulators IMPEDANCE. In 1989 - 1991 he was a visiting professor at the Physics Department of TU Munich. He joined the Department of Electronics and Electrical Engineering at Glasgow University in 1991. As a leader of the Device Modelling Group he has a major contribution in the development of the 2D and 3D device simulators and their application in the design of compound FETs, SiGe MOSFETs and IGBTs. He is also working on the design and performance of parallel algorithms.

Andrew Brown is a research assistant in the Department of Electronics & Electrical Engineering at the University of Glasgow. He received his B.Eng. (Hons) in Electronics and Electrical Engineering from the University of Glasgow in 1992. He is currently working on the numerical simulation of Insulated Gate Bipolar Transistors as part of his doctoral research. His research interests include device modelling, parallel computing, and visualisation.

Scott Roy is a researcher in the Department of Electronics & Electrical Engineering. He received his B.Sc. (Hons) in Physics and Electronic Engineering from the University of Glasgow in 1987. He is completing doctoral research in the study of single electronic systems at the Nanoelectronic Research Centre, and working on the design and construction of a hypermesh based parallel processing system. His present interests include ultrasmall digital systems, computer architecture, high speed interconnect networks and parallel algorithms.

John Barker has been a Professor in the Department of Electronics and Electrical Engineering since 1985. He has a long standing interest in computational methods, device modelling and transport theory. From 1970-78, 1980-85 he was a member of the Theory group in the Dept. of Physics, University of Warwick. During 1978 and 1979 he worked at the IBM T.J. Watson Laboratory, Yorktown Heights, North Texas State University and Colorado State University. From 1987 to 1989 he was academic director of the IBM UK/Glasgow University Kelvin Project on Numerically Intensive Parallel Computing. He is academic director of the Parsytec Parallel Processing Centre at the University of Glasgow.