GPGPU Accelerated Sparse Linear Solver for Fast Simulation of On-Chip Coupled Problems

Christian R. Bahls* Sebastián Gim Jakub Pas** Rolf-W. Frey** Gabriela Ciuprina****

** Institut Allgemeine Elektrotechnik, Fakultät für Informatik und Elektrotechnik, Universität Rostock, Germany. Christian.bahls@gmx.de*

*** Centrul de Inginerie Electrică Asistată de Calculator (CIEAC), Universitatea Politehnica Bucureşti, Romania. (kuba,gabriela,rolf,seb)@lmn.pub.ro*

Abstract: Continued device scaling into the nanometer region has given rise to new effects that previously had negligible impact but now present greater challenges to designing successful mixed-signal silicon. Design efforts are further exacerbated by unprecedented computational resource requirements for accurate design simulation and verification. This paper presents a general purpose graphic processing unit (GPGPU) accelerated sparse linear solver for fast simulation of on-chip coupled problems using nVIDIA and ATI GPGPU accelerators on a multi-core computational cluster and evaluates parallelization strategies from a computational perspective.

Keywords: Coupled Problems (Electronic Circuits), Software Methodology (Software Design), Cluster Computing (GPGPU Acceleration).

1. INTRODUCTION

Incessant miniaturization of the transistor according to Moore's Law has lead to generational improvements in microprocessor technology [1]. However, continued scaling of devices into the nanometer region has given rise to new effects that had previously negligible impact but now present challenges to continued scaling. This has resulted in an increased complexity in engineering resources essential for a successful design. The ITRS roadmap suggests extreme scaling of CMOS technology until the 10 nm region and operating frequencies of up to 60 GHz in future generation devices [2]. At such close dimensions, fabrication process variations, substrate noise and electromagnetic coupling between circuit components make mixed-signal RF silicon designs extremely challenging.

Because of this, the CHAMELEON-RF project was conceived as part of an initiative to address these issues [2]. The project is a research platform for the development of prototype tools and methodologies for comprehensive high accuracy modeling of on-chip electromagnetic effects using the domain decomposition (DD) approach and the concept of electromagnetic interconnectors or 'hooks', which are essentially boundary conditions between electromagnetic circuit elements (EMCEs) [3], in order to manage the unprecedented complexity faced when designing next generation highly integrated mixed-signal architectures and System on Chip (SoC) applications.

The CHAMELEON-RF nano-EDA research prototype platform called Chamy which is being developed here at CIEAC incorporates an EM field simulator based on the Finite Integral Technique (FIT) [4] with systematic all level model order reduction to keep complexity manageable [5]. The method allows tractable multi-scale parameterized model extraction of coupled structures with possibility of sensitivity analysis [6]. This approach has advantages over alternative approaches such as FEM, BEM etc. in full 3D field simulators, which although enables greater accuracy, is intractable for most practical real world designs.

In this paper, we present recent developments within the CHAMELEON-RF project, in particular, novel techniques employing GPGPU acceleration of sparse linear solvers which accelerates the computational kernel of Chamy and evaluate parallelization strategies for distributed processing on a multi-core computational cluster from a computational perspective. The next section presents a theoretical overview of numerical methods used in Chamy to construct compact state space matrices for fast EM simulation of on chip coupled problems. Various approaches such as domain decomposition, multigrid techniques or reduced order modeling techniques can be used to reduce the degrees of freedom (DoFs) of a simulation. However this comes at the cost of reduced accuracy because there is often the trade off between accuracy and simulation time. GPGPUs have been identified as the enabling technology for peta-scale computing. With the use of GPGPUs, high accuracy simulations of complex systems can be obtained within a tractable amount of time. The remaining sections of this paper are organized as follows. Section 3 presents the key points gained from experience in developing GPGPU accelerated solvers for large, sparse, unsymmetrical systems of linear equations akin to the matrices encountered in CHAMELEON-RF using ATI Brook+. Numerical results from benchmarks of the algorithm and code parallelization issues are discussed. The paper is finally concluded in Section 4.

2. SYSTEM STATE SPACE MATRIXES

Numerical methods are employed in computational electromagnetics (CEM) to efficiently approximate solutions to real world problems of electromagnetic interaction between objects in the real world environment where analytical or closed form solutions to the classical Maxwell equations (1-4) are not readily derivable [7].

$$curl\,\mathbf{H} = \mathbf{J} + \frac{\partial \mathbf{D}}{\partial t} \tag{1}$$

$$curl\,\mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{2}$$

$$div\,\mathbf{D} = \rho \tag{3}$$

$$div\,\mathbf{B} = 0 \tag{4}$$

$$\mathbf{n}\,curl\,\mathbf{E}(P,t) = 0 \quad for \quad \forall P \in \Sigma - S_k^{''} \tag{5}$$

$$\mathbf{n}\,curl\,\mathbf{H}(P,t) = 0 \quad for \quad \forall P \in \Sigma - S_k^{''} \tag{6}$$

$$\mathbf{n} \times \mathbf{E}(P,t) = 0 \; for \; \forall P \in \bigcup S_k^{'} \tag{7}$$

$$\mathbf{n} \times \mathbf{H}(P,t) = 0 \; for \; \forall P \in \bigcup S_k^{''} \tag{8}$$

Generally, various approaches within CEM involve the discretization of a continuous domain of interest by a grid (with appropriate boundary conditions) before iteratively or otherwise solving the Maxwell equations for each point in the grid. Both orthogonal and non orthogonal grids are possible depending on application scenario. Likewise, multigrid and flexible grid methods for structured and unstructured grids are also available and either the integral or differential form could be used for solutions in time or frequency domain over n dimensions.

$$\begin{cases}(2)\\(4)\end{cases} \Rightarrow \begin{cases}\oint \mathbf{E}d\mathbf{r} = -\iint \frac{\partial \mathbf{B}}{\partial t} d\mathbf{A} \\ \oiint \mathbf{B}d\mathbf{A} = 0\end{cases} \Rightarrow \begin{cases}\mathbf{C}v = -\frac{d\varphi}{dt} \\ \mathbf{D}'\varphi = 0\end{cases}$$

$$\begin{cases}(1)\\(3)\end{cases} \Rightarrow \begin{cases}\oint \mathbf{H}d\mathbf{r} = \iint (\mathbf{J} + \frac{\partial \mathbf{D}}{\partial t}) d\mathbf{A} \\ \oiint \mathbf{D}d\mathbf{A} = \iiint \boldsymbol{\rho} d\mathrm{v}\end{cases} \Rightarrow \begin{cases}\mathbf{C}'\mathbf{u} = \mathbf{i} + \frac{d\boldsymbol{\psi}}{dt} \\ \mathbf{D}\boldsymbol{\psi} = \mathbf{q}\end{cases} \tag{9-13}$$

$$\Rightarrow div\mathbf{J} = -\frac{\partial \rho}{\partial t} \Rightarrow \iint \mathbf{J}d\mathbf{A} = -\iiint \frac{\partial \rho}{\partial t} d\mathrm{v} \Rightarrow \mathbf{Di} = -\frac{d\mathbf{q}}{dt}$$

$$\begin{cases}\mathbf{B} = \mu\mathbf{H} \\ \mathbf{D} = \varepsilon\mathbf{E} \\ \mathbf{J} = \sigma\mathbf{E}\end{cases} \Rightarrow \begin{cases}\varphi = \mathbf{M}_{\boldsymbol{\mu}}\mathbf{u} = \mathbf{M}_{\mathbf{v}}^{-1}\mathbf{u} \\ \boldsymbol{\psi} = \mathbf{M}_{\boldsymbol{\varepsilon}}\mathbf{v} \\ \mathbf{i} = \mathbf{M}_{\boldsymbol{\sigma}}\mathbf{v}\end{cases} \tag{14-16}$$

Chamy uses the concept of the EMCE [3] (Figure 1) which imposes the boundary conditions (5-8). FIT [4] is then used to obtain a set of discrete algebraic equations (9-13) with associated material properties (14-16). State space matrices to describe the system of the form (17-18) can then be constructed and solved using an appropriate direct or iterative solver. The solution of such a system consisting of large, sparse, unsymmetrical matrixes is computationally the most costly step of the entire process.

$$\begin{cases}C\frac{d\mathbf{x}}{dt} = -G\mathbf{x} + B\mathbf{u} \\ \mathbf{y} = L\mathbf{x}\end{cases} \tag{17-18}$$

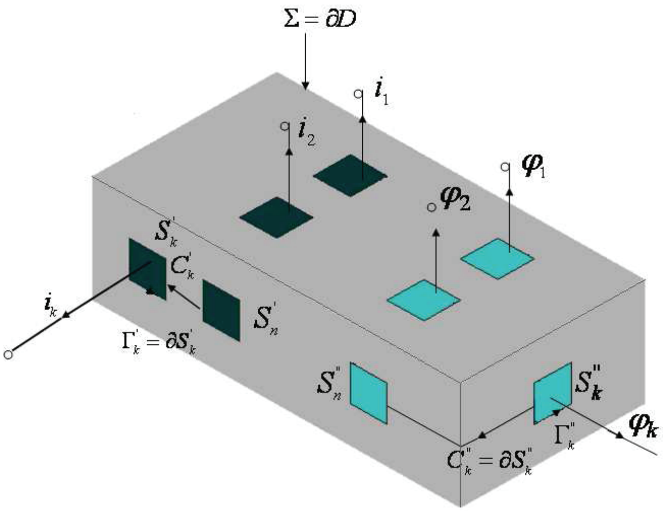


Fig 1: Electromagnetic Circuit Element (EMCE)

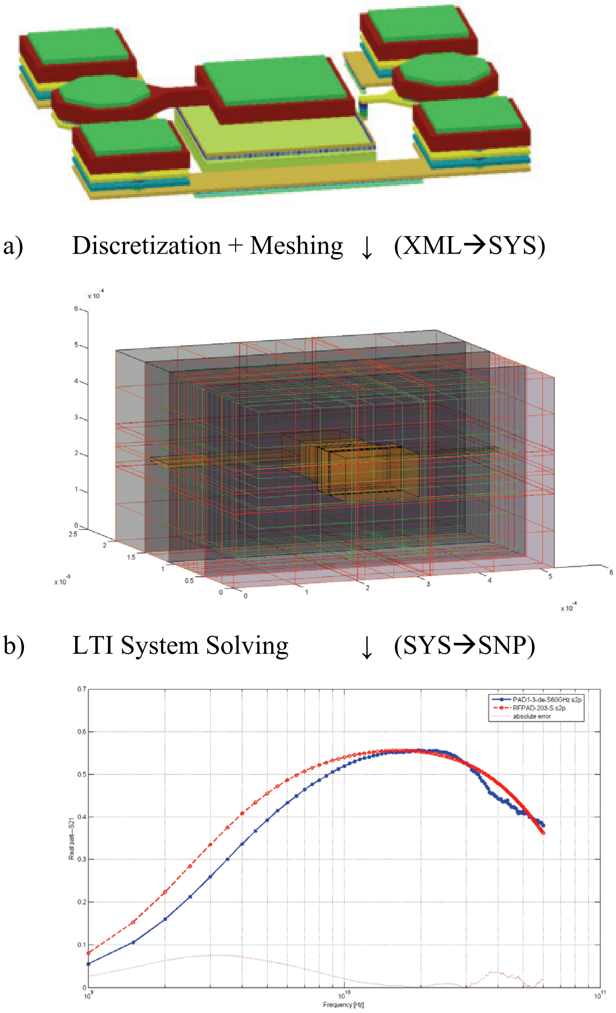
Creative use of domain decomposition, multi grid techniques or reduced order modelling techniques (ROM) can be selectively applied at all levels of the process to efficiently prune down DoFs. However, the simulation of complex systems within a reasonable amount of time remains a computational challenge.

3. GPGPU ACCELERATION

Shortly after the turn of the 21st century, market driven demand for graphics rendering performance lead to the widening performance gap between graphics processing units (GPUs) and general purpose processors (GPPs) [8]. Almost a decade later, the wide availability of commodity GPUs with high floating point arithmetic performance (circa 1 Tflop region) coupled with fast memory bandwidths (approaching 100 GBps) far outperforms contemporary commercially available GPPs. This however comes at the cost of increased complexity in algorithm design and device programmability. Device programmability is being addressed by major GPGPU vendors with the release of ATI Brook+ and nVIDIA CUDA suites. Algorithm design to harness the latent computational potential of GPGPUs however, remains an art with careful host – GPU partitioning of code and many iterations of benchmarking, application profiling and tuning. This section describes key issues behind the algorithm development of a GPGPU accelerated solver.

3.1 Chameleon-RF workflow

The overall Chameleon-RF workflow involves discretization and meshing of an on-chip structure to construct the system state space matrices (XML→SYS).


a) Discretization + Meshing ↓ (XML→SYS)

b) LTI System Solving ↓ (SYS→SNP)

c) Validation with measurement

Fig 2: Chameleon RF workflow illustrated using RFPAD test structure. a) constructing the state space matrices (SYS) from a layout file (XML). b) numerically linear time invariant (LTI) solving state space (SYS) matrices to yield a frequency response file (SNP). c) validation with measurement.

Once the multiple-input multiple-output (MIMO) state space system linear time invariant (LTI) (Eq. 1-16) matrices (SYS) have been constructed, it then can be solved by an appropriate direct or iterative numerical method to yield the systems frequency response over a range of desired frequencies (SNP). The resulting simulation can then be calibrated against actual physical measurements and added to the design library as a reusable RF block. Figure 2 illustrates the Chameleon-RF workflow for a typically encountered structure within the semiconductor industry. The RFPAD structure is a full stack (four metal layers) RF pad placed over a 3 μm thick nwell layer that acts as shielding. RF pads are a widely used structure in the semiconductor industry for off-chip packaging connections via wire-bond leads or flip chip technology and also functions as a filter for high frequency components of the signal.

3.2 Accelerated Solver

Within this workflow, the SYS→SNP process which involves solving the LTI matrices is the apparent bottleneck where solutions of systems with > 100,000 DoFs being a challenge. The pseudo-code of this process is as follows:

```
sys2snp (SysFilename, SnpInFilename, AFSinfo,
         SnpOutFilename) {
  [C,G,B,L,F]= ReadMatrices (SysFilename)

  if (AutomaticFrequencySelection == false) {
    [freq_list]=ReadFrequencies (SnpInFilename)
  }
  else {
    [freq_list]=AutomaticFrequencySelection ()
  }

  for (idx_f=1:length(freq_list) {
       ComputeFrequencyResponse (idx_f, C,G,B,L,F)
  }

  WriteResponse (SysOutFilename)
}
```

Profiling the application identified that computing the frequency response is computationally the most expensive step and the algorithm was subject to GPGPU acceleration attempts. Initial experiments using Brook+ and CUDA to accelerate supernodal [9, 10] and multifrontal [11] direct solvers indicated however that BLAS 3 supernodal updates for EMCE type matrices which are unsymmetric, non-hermitian, ill-conditioned and nearly indefinite were not large enough to merit GPGPU processing. Supernodal occurrences for EMCE type matrices were observed to be only in the 10-30 column region. Further experimentation indicated that supernodal occurrences needed to be around > 128 columns before GPGPU acceleration became advantageous. Depending on the matrix structure (Eg. for non EMCE type matrices), this may be the case and the direct solution of offloading BLAS calls to GPGPU is advantageous. It was also observed that the fill in for unordered EMCE type matrices was rather large (200+ fold). Several possible techniques to mitigate these observations were considered. The supernodal occurrence threshold could be lowered by rewriting the existing graph elimination tree scheduler for maximum data resuse within the GPGPU. Also, the fill-in could potentially be reduced by METIS [12] like pre-ordering packages to pack non zero occurances closer to the diagonal. However, heuristic pre-ordering packages are mostly non-deterministic polynomial type problems that incur additional computational cost. Because of this, iterative solvers were investigated as they were deemed more compatible to mapping with GPGPU architecture.

Using the RFPAD test structure as a sample system, several iterative methods [13] consisting of the derivatives of

Bi-Conjugate Gradient [14] (BiCG Stabilized, QMRCG Stabilized) and Generalized Minimum Residual Solver [15] (FGMRES, DQGMRES) with and without Incomplete LU preconditioning (ILU0, ILU Threshold) were tested. Analyzing the result of these numerical experiments, it became clear that a generalized minimum residual (GMRES) like derivative Krylov subspace iterative method, the Direct-Quasi GMRES (DQGMRES) [16] with variable pre-conditioners in each step would lend itself to GPGPU acceleration. DQGMRES only performs an incomplete orthogonalization by truncating the solution process to the last k vectors at each step which increases efficiency.

3.3 Numerical Results

The DQGMRES method (without pre-conditioner at the moment) was implemented in C for the host CPU and GPU using ATI Brook+. Two test matrices were selected to benchmark the implementation. The first were matrices of increasing sizes arising from 3D discrete Laplacians with Dirichlet boundary conditions using Finite Difference (FD) method. The second were matrices of increasing sizes arising from 3D Maxwell equation using the FIT. Both test matrices were benchmarked on the host CPU and an ATI Radeon 4870 GPU.

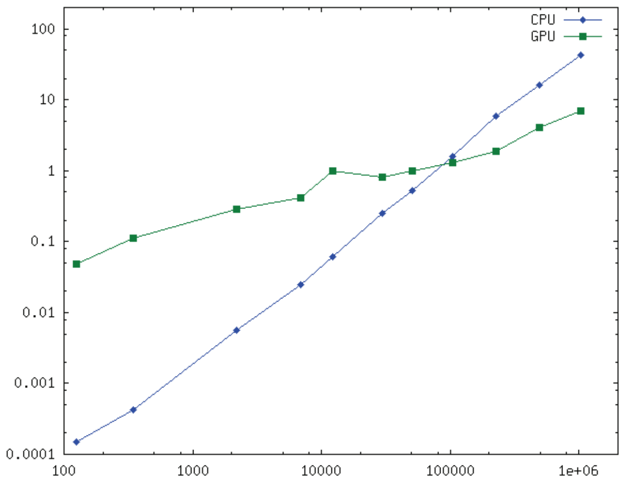


Fig 3: Log-log plot of 3D Laplacian DoFs vs simulation time (seconds).

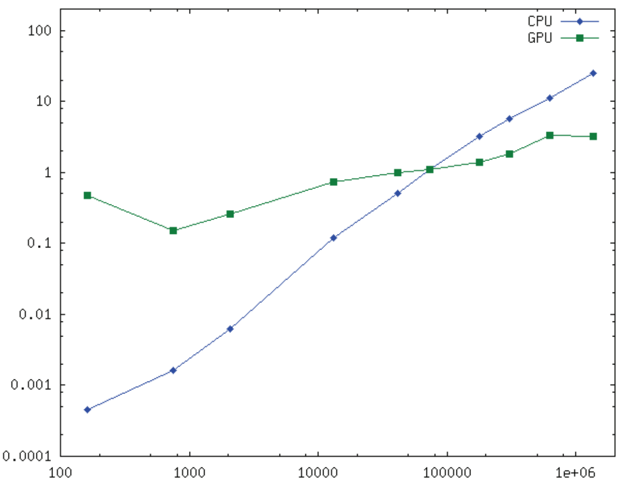


Fig 4: Log –log plot of 3D Maxwell DoFs vs simulation time (seconds).

It was observed that for large sparse matrices with > 100,000 DoFs, GPGPU acceleration significantly improved simulation time compared with conventional CPU implementations.

4. CONCLUSION

Promising initial results of un-tuned code on test matrices indicate the latent computational potential of GPGPUs. Further speed up should be obtainable using code tuned to the underlying architecture. The key insight it was observed, for successfully developing algorithms for GPGPU acceleration that efficiently utilize the GPGPU computing capability, is to select algorithms which map well or could be adapted to the underlying GPGPU architecture.

GPGPU technology is currently at its infancy and evolving. The work presented here is part of on-going research activities at CIEAC into numerical methods in electrical engineering. As part of the Marie Curie FP6 Transfer of Knowledge for nano Electronic Design Automation (ToK4nEDA) project, a high performance computing research cluster is being commissioned for research into state of the art computing techniques and electromagnetic simulation methods. The Advanced Technology Laboratory Server (ATLAS) is a hybrid cluster consisting of 2-socket Opteron hosts with GPGPU accelerators from nVidia and ATI. ATLAS host nodes are connected by a high bandwidth low latency Infiniband interconnect switch fabric with Remote Direct Memory Access (RDMA) capabilities. A clustered file system Lustre is mounted and configured for redundancy using a combination of hardware and software RAID techniques.

Ongoing activities and future work at the author’s institution include research into Out-of-Core techniques for large matrices which do not fit entirely into memory, mixed precision computational techniques, multiple GPGPU devices and distributed memory scenarios based upon the new approaches experimented with here.

ACKNOWLEDGEMENT

The authors would like to gratefully acknowledge support and insightful discussion with Prof. Daniel Ioan and Assc. Prof. Gabriela Ciuprina. The authors would also like to acknowledge the financial support offered by the European Commission under the Marie Curie Fellowship (FP6) programme.

The views stated herein reflect only the authors views and the European Comission is not liable for any use that may be made of the information contained.

REFERENCES

[1] E. Mollick, "Establishing Moore's Law," *IEEE Annals of the History of Computing*, vol. 28, pp. 62-75, 2006.

[2] J. Niehof, H. Janssen, and W. Schilders, "Comprehensive High-Accuracy Modeling of ELectromagnetic Effects in Complete Nanoscale RF blocks: CHAMELEON RF," presented at Signal Propagation on Interconnects, 2006. IEEE Workshop on, 2006.

[3] I. M. D. Ioan, "Missing Link Rediscovered: The Electromagnetic Circuit Element Concept," *JSAEM Studies in Applied Electromagnetics and Mechanics*, vol. Vol 8, pp. pp. 302-320, 1999.

[4] T. Weiland, "A Discretization Method for the Solution of Maxwell's Equations for Six-Component Fields," *Electronics and Communication.*, vol. 31, pp. 116, 1977.

[5] D. Ioan, G. Ciuprina, M. Radulescu, and E. Seebacher, "Compact modeling and fast simulation of on-chip interconnect lines," *Magnetics, IEEE Transactions on*, vol. 42, pp. 547-550, 2006.

[6] D. Ioan, G. Ciuprina, and M. Radulescu, "Theorems of parameter variations applied for the extraction of compact models of on-chip passive structures.," presented at Signals, Circuits and Systems, 2005. ISSCS 2005. International Symposium on, 2005.

[7] M. Clemens, "Large systems of equations in a discrete electromagnetism: formulations and numerical algorithms," *Science, Measurement and Technology, IEE Proceedings* -, vol. 152, pp. 50-72, 2005.

[8] D. L. John D. Owens, Naga Govindaraju, Mark Harris, Jens Krüger, Aaron E. Lefohn, Timothy J. Purcell " A Survey of General-Purpose Computation on Graphics Hardware " *Computer Graphics Forum*, vol. 26, pp. 80-113, 2007.

[9] S. C. E. James W. Demmel, John R. Gilbert, Xiaoye S. Li, Joseph W. H. Liu, "A supernodal approach to sparse partial pivoting," *SIAM J. Matrix Analysis and Applications*, vol. 20, pp. 720-755, 1999.

[10] J. R. G. a. X. S. L. James W. Demmel, "An Asynchronous Parallel Supernodal Algorithm for Sparse Gaussian Elimination," *SIAM J. Matrix Analysis and Applications*, vol. 20, pp. 915-952, 1999.

[11] A. D. Timothy, "Algorithm 832: UMFPACK V4.3---an unsymmetric-pattern multifrontal method," *ACM Trans. Math. Softw.*, vol. 30, pp. 196-199, 2004.

[12] G. K. a. V. Kumar, "METIS: Unstructured graph partitioning and sparse matrix ordering system.," *Technical Report, Department of Computer Science, University of Minnesota*, 1995.

[13] Y. Saad, *Iterative Methods for Sparse Linear Systems*: Society for Industrial and Applied Mathematics, 2003.

[14] H. Martin Becker and S. Manfred, "A Variant of the Biconjugate Gradient Metho Suitable for Massively Parallel Computing," in *Proceedings of the 4th International Symposium on Solving Irregularly Structured Problems in Parallel*: Springer-Verlag, 1997.

[15] S. Youcef and H. S. Martin, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM J. Sci. Stat. Comput.*, vol. 7, pp. 856-869, 1986.

[16] K. W. Yousef Saad, "DQGMRES: a direct quasi-minimal residual algorithm based on incomplete orthogonalization," *Numerical Linear Algebra*, vol. 3, pp. 320-329, 1996.