Frequency Interpolation Methods for Accelerating Parallel EMC Analysis

K. Homma, M. Noro
Fujitsu Laboratories Ltd

K. Nagase, T. Yamagajo
Fujitsu Ltd

Peter Strazdins*
Australian National University

http://cs.anu.edu.au/∼Peter.Strazdins/seminars/ParFFS

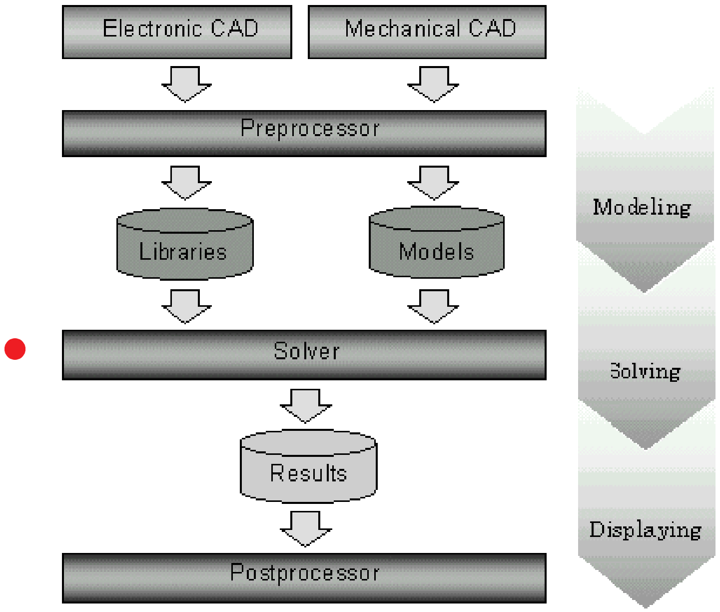
April 27, 2001

1 Talk Outline

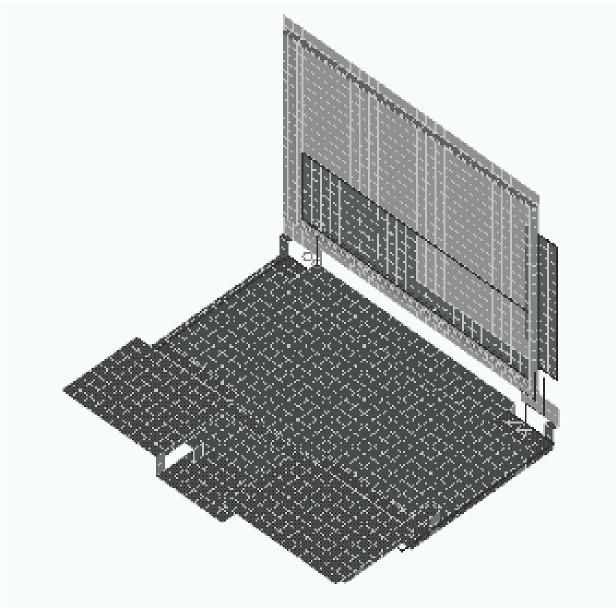
- introduction to EMC analysis and ACCUFIELD
- fast frequency stepping in ACCUFIELD
- direct solution methods for symmetric indefinite systems
- iterative solution methods via fast frequency stepping
- parallel performance and convergence issues
- related work in EMC
- conclusions

2 Introduction to EMC and ACCUFIELD

- Electromagnetic Compatibility (EMC) issues becoming increasingly important
  - $\uparrow$ GHz (microwave) clock speeds, increasingly complex electrical systems
  - health & system malfunction issues due to electromagnetic interference (EMI)
    - $\Rightarrow$ strict EMC regulations imposed by most countries
- the ACCUFIELD application (Fujitsu) predicts EMI at the system design stage



ACCUFIELD architecture



Note PC model

- used mainly for relatively simple electric devices, such as printed circuit boards
- uses the moment & transmission line approximation methods to calculate the EM field in the frequency domain
- models: break conductive & dielectric surfaces / wires into discrete elements

- given an incident electric field V (known), find the induced current I from $ZI = V$, where:
  - Z is a dense complex symmetric (indefinite) $N \times N$ matrix
    - assembled using the relatively compact model description
  - for Note PC model, $N = 7041$, but $N \approx 30000$ possible for larger systems
  - direct solution via $Z \rightarrow PLDL^T P^T$ requires $\frac{N^3}{3}$ FLOPs, $\approx \frac{N^2}{2}$ storage
    - store Z in lower triangular half; overwrite with L and D
    - $\Rightarrow$ a large $||$ parallel machine is required eg. AP3000, 24 nodes @ 300 MHz, 1.5GB
  - EMC analysis is usually required at different frequencies ω
    - here $V = V(\omega), Z = Z(\omega)$; the direct solution at ω_c can be used to precondition an $O(N^2)$ iterative solution for ω near ω_c
  - using solution vector I, can determine the induced surface fields, current emissions, the effectiveness of filters and shielding, etc

3 Fast Frequency Stepping in ACCUFIELD

- consider a || analysis over set of frequencies $\Omega = \{\omega_1, \omega_2, \ldots, \omega_{n_f}\}$, $\omega_i < \omega_{i+1}$
- for linear system assembly:
  - the model's information is broadcast to all cells (MPI is used throughout)
  - explicitly assemble $Z(\omega_i)$ for $\omega_i \in \Omega_s \subseteq \Omega$ (little communication is required)
  - for $\omega_i \notin \Omega_s$, interpolate $Z(\omega_i)$ from $Z(\omega_j), Z(\omega_k)$, where $j < i < k$ and $\omega_j, \omega_k \in \Omega_s$
  - in each case, each cell independently generates the matrix elements it 'owns', according to an $r \times r$ block-cyclic matrix distribution over a $P \times Q$ logical grid
- for linear system solution:
  - choose a central frequency ω_c, eg. $\omega_c \in \Omega_s$, and obtain a direct solution here
  - the factorized $Z(\omega_c)$ is used to precondition the iterative solutions for ω_i to left *and* right of ω_c
  - do this until the iterative solution time $>$ half the direct solution time
- these methods represents space-time tradeoff

4 Direct Solution of Symmetric Indefinite Systems

- a direct solution for I is the most general and accurate method
- stable algorithms exist that exploit symmetry to require only $\frac{N^3}{3} + O(N^2)$ FLOPS in the factorization:
  - (row/column) symmetric interchanges are required to eliminate some columns
  - the Bunch-Kaufman LDLT factorization algorithm $Z \rightarrow PLDL^TP^T$ has been shown to be very competitive!
  - we have developed especially efficient parallelizations:
    - advanced communication algorithms to minimize communication startups
    - a reduced interchange variant algorithm was developed
    - for serial execution, out-performs highly competitive LAPACK `zysv()`
    - for $||$ execution, the algorithmic blocking factor can be chosen independently of r for optimum performance

- back-solve computation is designed to perform: $I = V;\ I \leftarrow P^{-T}L^{-T}D^{-1}L^{-1}P^{-1}V$
  - has only $O(N^2)$ FLOPs, but high $O(N)$ associated $||$ overheads
  - can minimize by:
    - blocking messages $(I \leftarrow L^{-1}I)$
    - caching D on each cell from the factorization stage (for $I \leftarrow D^{-1}I$)
    - minimizing symmetric interchanges $(I \leftarrow P^{-1}I)$
  - is also used for preconditioning in our iterative solver
- the DBLAS Distributed BLAS Library (ANU) interfacing to MPI and serial BLAS is used for all computational components in the solvers

5 Iterative Solution for Fast Frequency Stepping

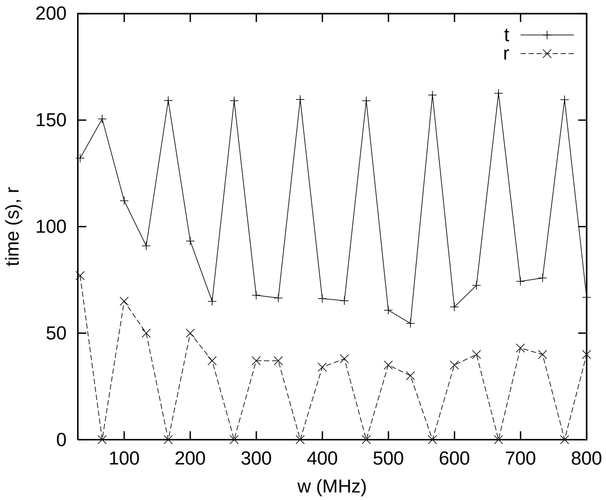
- use a preconditioned conjugate gradient method, adapted for indefinite symmetric systems
  - nb. ACCUFIELD usually generates nearly definite systems
- each iteration requires:
  - a symmetric matrix-vector multiply $r \leftarrow r - \alpha Z(\omega_i)p$
  - a back-solve via the factored preconditioner $p \leftarrow \beta p + Z^{-1}(\omega_c)r$
  - miscellaneous vector operations (adding a small amount of extra overhead)
- $Z(\omega_i)$ and the factored $Z(\omega_c)$ are stored in an $N \times (N+1)$ rectangular matrix
- for large matrices, eg. $\frac{N}{P} \geq 1000$, time is dominated by the matrix-vector multiply
- for smaller matrices, time is dominated by communication startups

6 Performance: Convergence and Computational Speed

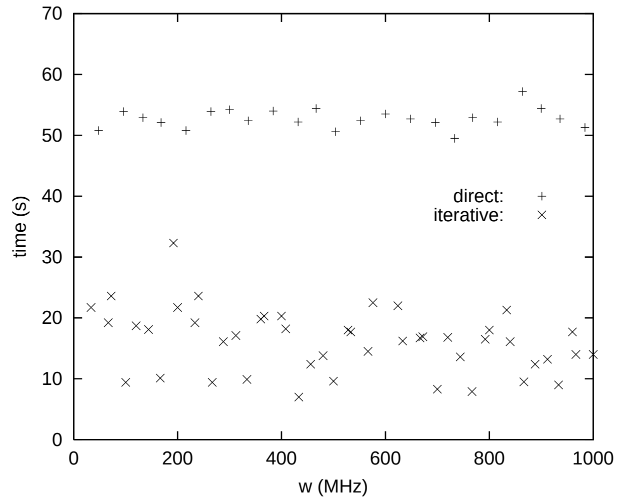
- results here are for an 8-node Fujitsu AP3000 with 360 MHz US-2 processors (startup cost $\approx 20\mu s$, transfer rate of up to 80 MB/s)
- for analysis of the Note PC model using direct solution:

# cells :	1	4	8
times (s):	293	103	59

- parallel execution times (r is the convergence rate):



POS terminal, $N = 7017$
(overall FFS speedup of 1.65;
at most 2 iterative solves per ω_c)



Note PC model, $N = 4608$
(overall FFS speedup of 2.18;
nearby ω ‘pairs’ converged rapidly)

7 Related Work in EMC

- frequency interpolation proposed for non-symmetric systems (non-||) by Hoyler and Unbehauen (1996)
  - favorable comparisons with other (eg. block-Jacobi and GMRES) preconditioning techniques
  - FFS proceeds only in a single direction
- FFS has been encapsulated in a more general framework called Model-Based Parameter Estimation (Miller, 1998)
  - shows how interpolation can reduce computational work on both the linear system assembly and solution
- parallel iterative methods for EMC reported by Jakobus (2000)
  - used a conjugate gradient method with a block-Jacobi preconditioner
  - improvements reported for non-symmetric systems
  - no frequency interpolation was used

8 Conclusions

- EMC analysis using the moment method can be highly accurate
- but the resulting linear systems can be very large: $||$ processing is required!
- have developed scalable & efficient algorithms for both assembly and solution
- EMC analysis over multiple frequencies $\Rightarrow$ the use of (potentially) much faster iterative solution methods (some extra memory costs)
- for exploiting symmetry: $\approx \frac{1}{2}$ the FLOPs (direct solution only) and storage but have $\approx 1.5\times$ the communication startups
- for FFS, $||$ overheads are relatively high $\Rightarrow$ only efficient for 3–4 interpolations for moderate-sized systems
  - FFS speedups much greater for serial computations
- future work: improving convergence rates of FFS, further refinement of communication performance of the back-solve computation
- parallel EMC: the increasing complexity of electrical systems $\Rightarrow$ Moore's Law may never catch up!