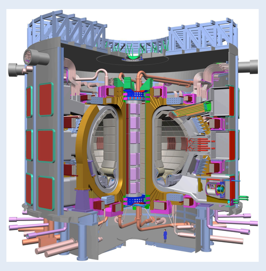
SAND2010-2797P

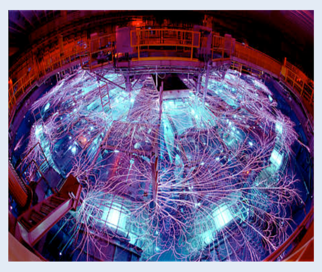
Large-Scale Parallel Performance of Multiphysics Applications on Multicore Architectures

Paul T. Lin and John N. Shadid
Computational Science R&D Group, Sandia National Labs
Albuquerque NM 87122; e-mail: ptlin@sandia.gov

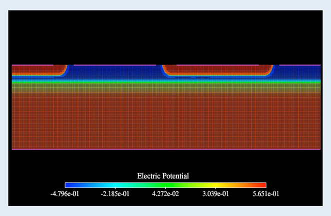
Abstract: The efficient computation of multiphysics simulations is challenging for several reasons: strong coupling of the multiphysics systems, high nonlinearity and a large spectrum of interacting length and time scales. We examine the parallel performance of two multiphysics application codes for large-scale simulations. The first simulates resistive magnetohydrodynamics (MHD), which describes the dynamics of charged fluids in the presence of electromagnetic fields. Important applications include fusion energy devices such as tokamak reactors and Z-pinch devices. The second application code involves the simulation of semiconductor devices via the drift-diffusion equations. An important application is the simulation of the response of electronics in radiation environments. For the two multiphysics application codes, a stabilized finite element method is used to discretize the system of PDEs, which are then solved by Newton-Krylov methods. Two major challenges for these large-scale simulations are the scalability of the solvers and efficiency of the algorithm on multicore processors. The choice of the preconditioner is critical to the parallel scaling and to reducing the solution time for these linear systems. We have been investigating multigrid preconditioners as well as approximate block factorization and physics-based preconditioners.
To examine parallel scalability, studies have been performed on an IBM Blue Gene/P platform for problems as large as two billion unknowns on 100k cores. We have also examined scaling on a Cray XT3/4 platform. We performed studies on various platforms with 4-16 cores per compute node and found that the application codes could use all the cores with reasonable efficiency. However, the decrease in efficiency with increasing core count makes it clear that a hybrid approach will be needed in the future.



ITER



SNL Z-machine



Bipolar Junction Transistor

Resistive MHD Model

(With R. Pawlowski, E. Cyr, R. Tuminaro, L. Chacon, J. Banks)

Navier-Stokes + Electromagnetics

$$\rho\frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla\mathbf{u}) - \nabla\cdot(\mathbf{T} + \boxed{\mathbf{T_M}}) - \rho\mathbf{g} = 0 \qquad \mathbf{T} = -(P + \frac{2}{3}\mu(\nabla\cdot\mathbf{u}))\mathbf{I} + \mu[\nabla\mathbf{u} + \nabla\mathbf{u}^T]$$

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\mathbf{u}) = 0 \qquad \mathbf{T_M} = \frac{1}{\mu_0}\mathbf{B}\otimes\mathbf{B} - \frac{1}{2\mu_0}\|\mathbf{B}\|^2\mathbf{I}$$

$$\rho C_p\left[\frac{\partial T}{\partial t} + \mathbf{u}\cdot\nabla T\right] + \nabla\cdot\mathbf{q}\boxed{-\eta\|\mathbf{J}\|^2} = 0$$

$$\boxed{\frac{\partial\mathbf{B}}{\partial t} - \nabla\times(\mathbf{u}\times\mathbf{B}) + \nabla\times(\frac{\eta}{\mu_0}\nabla\times\mathbf{B}) = \mathbf{0}}$$

2D

$$\mathbf{B} = \nabla\times\mathbf{A}$$

$$R_{A_z} = \frac{\partial A_z}{\partial t} + \mathbf{u}\cdot\nabla A_z - \frac{\eta}{\mu_0}\nabla^2 A_z + E_z^0 = 0$$

General case a strongly coupled, multiple time- and length-scale, nonlinear, nonsymmetric system with parabolic and hyperbolic character

Semiconductor Drift-Diffusion Model

(With R. Hoekstra, G. Hennigan, J. Castro, D. Fixel, E. Phipps, L. Musson, T. Smith, E. Keiter)

Electric potential

$$-\nabla\cdot\epsilon\nabla\psi = q\,(p - n + C)$$

$$\nabla\cdot\mathbf{J}_n - qR = q\frac{\partial n}{\partial t} \qquad \mathbf{J}_n = -qn\mu_n\nabla\psi + qD_n\nabla n$$

$$-\nabla\cdot\mathbf{J}_p - qR = q\frac{\partial p}{\partial t} \qquad \mathbf{J}_p = -qp\mu_p\nabla\psi - qD_p\nabla p$$

Each additional species adds an additional equation (also modifies equation for electric potential)

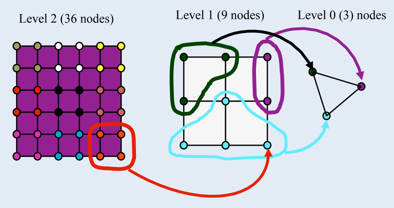
$$-\nabla\cdot\mathbf{J}_i - q_iR_i = q_i\frac{\partial X_i}{\partial t} \qquad \mathbf{J}_i = -q_i\mu_iX_i\nabla\psi - q_iD_i\nabla X_i \qquad \mu_i = \frac{q_iD_i}{kT}$$

$$-\nabla\cdot\epsilon\nabla\psi = q\,(p - n + C) + \sum_{i=1}^{n} q_iX_i \qquad q_i \equiv Z_iq$$

Drift-Diffusion uses stabilized FEM with Newton-Krylov solver as in MHD case; both use same Trilinos solvers

Trilinos ML Library: Algebraic Multilevel Preconditioners (Tuminaro, Hu, Sala, et al.)

Aggregates to produce a coarser operator
Restriction/prolongation operators
Construction of A_{k-1} as $A_{k-1} = R_k A_k P_k$



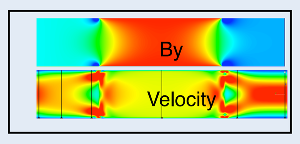
- Nonsmoothed aggregation (NSA)
- Petrov-Galerkin smoothed aggregation (PGSA)
  - Restriction smoothing
  - Calculate damping parameter to minimize P_l and R_l

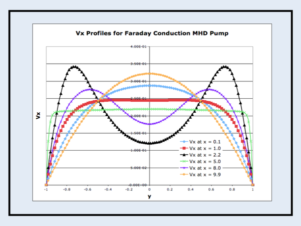
$$P_\ell = (I - \bar{\Omega}D_\ell^{-1}A_\ell)P_\ell^{(t)}$$

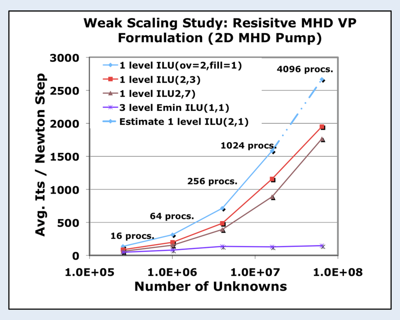
$$R_\ell = [P_\ell^{(t)}]^T(I - A_\ell D_\ell^{-1}\hat{\Omega})$$

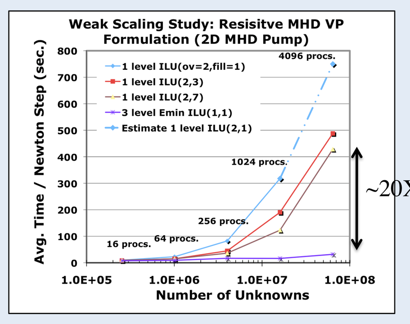
Large-Scale Parallel Performance

Scaling Performance for Fully-Coupled Resistive MHD: 2D MHD Faraday Conduction Pump

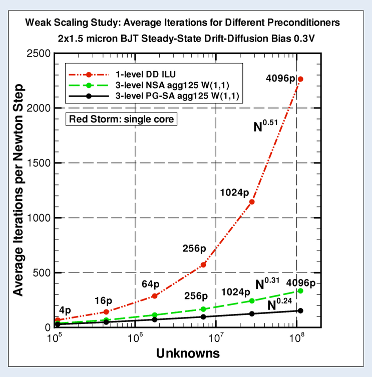


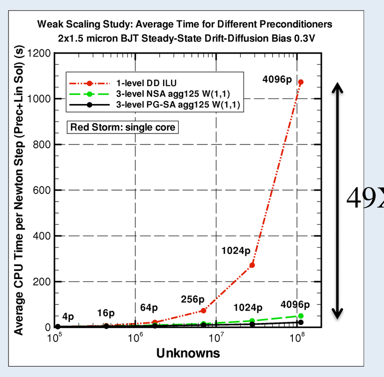




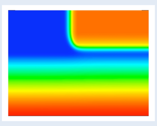


Scaling Performance for Drift-Diffusion: 2D Steady-State 2x1.5 μm BJT





Recall: Drift-Diffusion uses the same solvers as MHD



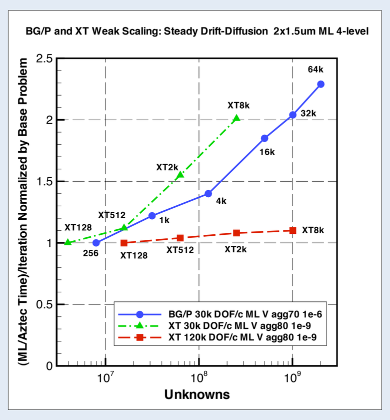
GMRES ML PGSA 3-level 125 nodes/ aggregate

For 110M DOF run on 4096 cores Cray XT: PGSA 2.3 times faster than NSA; 49 times faster than 1-level DD ILU

Steady-state Solution Time for 1B DOF on 24k cores of Cray XT3/4

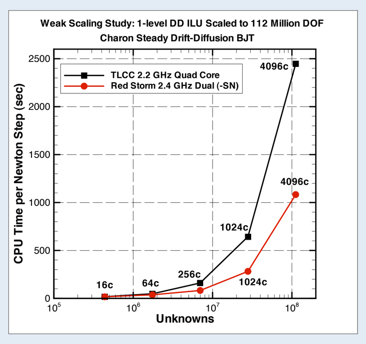
	Fine DOF	Avg iter/N	Time/Newton step (s)				Total time (min)
			Prec	Lin sol	Jac	Total	
MHD	1.05B	86 [18]	63	24	12	99	33
Drift-Diff	1.01B	243 [11]	50	216	20	243	47

Weak Scaling to 64k Cores on IBM BG/P

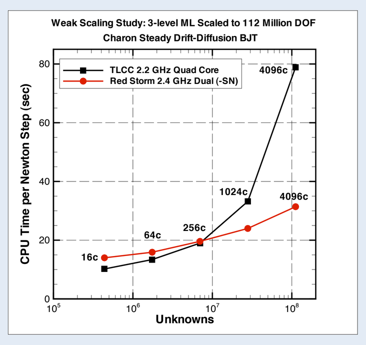


2x1.5 μm BJT steady-state drift-diffusion
Problem size increased by factor of 256 to two billion DOF on 65536 cores
Used all four cores per BG/P node; 30k DOF/core
TFQMR with ML PGSA 4-level
Comparison with 30k and 120k DOF/core for Cray XT3/4; as expected, better scaling with increased work per core
2 billion DOF problem successfully run on 100k cores

Large-Scale Simulations Need Capability Platforms



- 1-level ILU Preconditioner



- Multigrid Preconditioner (critical for scalability) stresses network more
- Scales much better on Cray XT > 1000 cores

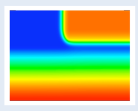
2x1.5 μm BJT Steady-state Drift-Diffusion
TLCC: quad-socket/quadcore 2.2 GHz AMD Barcelona; InfiniBand

Red Storm: dualcore 2.4 GHz AMD Opteron

Future Hardware Trends and Effect on Application Codes: More Sockets per Node, more Cores/Socket

- How efficiently do existing application codes use multiple cores?
- How long will MPI-only programming model work? Future programming paradigm? So far, single-level flat MPI approach still OK, but efficiencies are dropping. Clearly a hybrid approach will be needed in future. The decrease in memory bandwidth per core is not good; need to exploit locality.

Single Node Multicore Efficiency: Quad-socket, Quadcore CPU



core	DOF	linear sys solve		Jacobian		total	
		time(s)	η	time(s)	η	time(s)	η
1	28K	9.71	Ref	3.52	Ref	14.6	Ref
4	110K	10.7	91	3.48	1.01	15.4	94
8	219K	11.6	84	3.45	1.02	16.3	89
12	329K	13.2	74	3.46	1.02	17.9	81
16	438K	15.8	61	3.13	1.12	20.1	73

2x1.5 μm BJT steady drift-diffusion
2.2 GHz AMD Barcelona
Weak scaling: 28k DOF/core
Time per Newton step
Linear solve time (preconditioner setup and ML/Aztec) efficiencies problematic
Code performance significantly affected by memory BW

Single Node Multicore Efficiency: Dual-socket, 6-core CPU

core	DOF	linear sys solve		Jacobian		total	
		time(s)	η	time(s)	η	time(s)	η
1	28K	5.38	Ref	2.46	Ref	8.72	Ref
2	55K	5.83	92	2.46	100	9.19	95
4	110K	6.78	79	2.50	98	10.2	86
6	165K	7.65	70	2.55	96	11.1	78
8	219K	8.78	61	2.52	98	12.2	71
10	273K	9.77	55	2.52	98	13.2	66
12	329K	10.97	49	2.55	96	14.5	60

2x1.5 μm BJT steady drift-diffusion
2.6 GHz AMD Istanbul
Weak scaling: 28k DOF/core
Linear solve time (preconditioner setup and ML/Aztec) efficiencies problematic

Multicore Efficiency Study: Network and Nodes

configuration	54.5K DOF/core		218K DOF/core	
	time(s)	η	time(s)	η
128n 1ppn	25.9	Ref	147	Ref
32n 4ppn	26.0	100	152	97
16n 8ppn	27.1	96	163	90
10.5n 12ppn	30.3	86	194	76
8n 16ppn	35.5	73	229	64

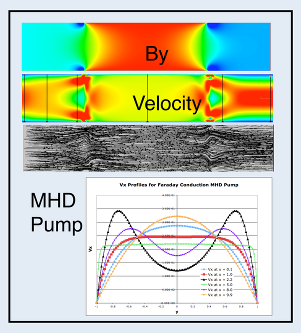
Combines effects of network and node architecture: vary nodes and cores/node for total of 128 cores
TLCC: quad-socket, quadcore 2.2 GHz AMD Barcelona; InfiniBand
Use all 16 cores per node
2x1.5 μm BJT steady drift-diffusion

configuration (quad core)	Newton Step	
	time(s)	η
4096n 1ppn	28.7	Ref
2048n 2ppn	31.2	92
1366n 3ppn	35.4	81
1024n 4ppn	39.3	73

Efficiency of using Cray XT3/4 quadcore 2.2 GHz Budapest (4096 cores)
2x1.5 μm BJT steady drift-diffusion

Multicore Efficiency Study: MHD Pump on Cray XT

Nodes	Cores	Compute Jac +Prec		Linear Solve		Total	
		Time (sec)	η (%)	Time (sec)	η (%)	Time (sec)	η (%)
4096	1	16.9	---	4.3	---	21.2	---
2048	2	18.2	93	4.5	95	22.6	94
1024	4	17.7	95	4.9	88	22.6	94



References:
[1] Lin, Shadid, Sala, Tuminaro, Hennigan, Hoekstra, "Performance of a Parallel Algebraic Multilevel Preconditioner for Stabilized FE Semiconductor Device Modeling," JCP, Vol 228, Issue 17, pp. 6250-6267, 2009
[2] Lin, Shadid, "Performance of an MPI-only Semiconductor Device Simulator on a Quad Socket/Quad Core InfiniBand Platform," Sandia National Laboratories Technical Report SAND 2009-0179, Jan 2009; expanded version submitted to JCP.
[3] Lin, Shadid, Tuminaro, Sala," Performance of a Petrov-Galerkin Algebraic Multilevel Preconditioner for Finite Element Modeling of the Semiconductor Device Drift-Diffusion Equations," IJNME 2010, published online in Wiley InterScience (www.interscience.wiley.com), DOI: 10.1002/nme.2902
[4] Lin, Shadid, Tuminaro, Sala, Hennigan, Pawklowski, "Parallel Fully-Coupled Algebraic Multilevel Preconditioner Applied to Multiphysics PDE Applications: Drift-diffusion, Flow/transport/reaction, Resistive MHD," submitted to IJNMF
[5] Shadid, Pawlowski, Banks, Charon, Lin, Tuminaro, "Towards a Scalable Fully-Implicit Fully-coupled Resistive MHD Formulation with Stabilized FE Methods," in review, JCP
[6] Shadid, Cyr, Pawlowski, Tuminaro, Charon, Lin, "Initial Performance of Fully-Coupled AMG and Approximate Block Factorization Preconditioners for Solution of Implicit FE Resitive MHD," in proceedings ECCOMAS CFD, June 2010

The authors gratefully acknowledge funding from the DOE Office of Science Advanced Scientific Computing Research (ASCR) and DOE NNSA Advanced Simulation & Computing (ASC) programs

National Nuclear Security Administration

ASC DOE / NNSA

Office of Science ASCR

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's NNSA under contract DE-AC04-94AL85000.

Sandia National Laboratories