Algebraic multigrid methods for mechanical engineering applications

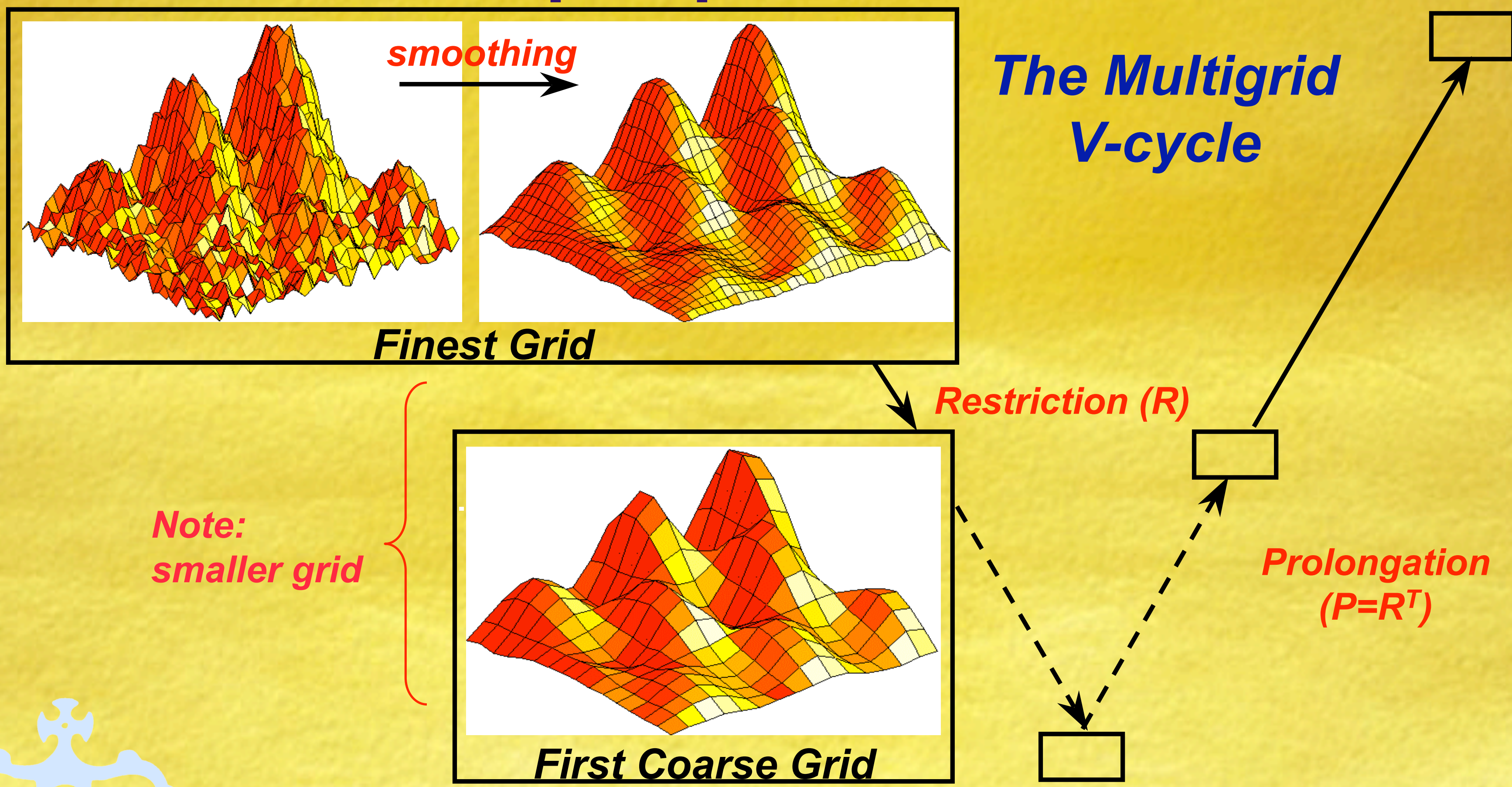
Mark F. Adams

St. Wolfgang/Strobl Austria- 3 July 2006

Outline

- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Multigrid smoothing and coarse grid correction (projection)



Multigrid components

- Smoother $\mathbf{S}^{\nu}(f,u_0)$, ν iterations of simple PC (Schwarz)
  - Multiplicative: great theoretical properties, parallel problematic
  - Additive: requires damping (eg, Chebyshev polynomials)
- Prolongation (interpolation) operator **P**
  - Restriction operator **R** ($\mathbf{R} = \mathbf{P}^T$)
    - Map residuals from fine to coarse grid
  - Columns of **P**: discrete coarse grid functions on fine grid
- **Algebraic** coarse grid (Galerkin) $\mathbf{A}_H = \mathbf{R}\mathbf{A}_h\mathbf{P}$
- AMG method defined by **S** and **P** operator

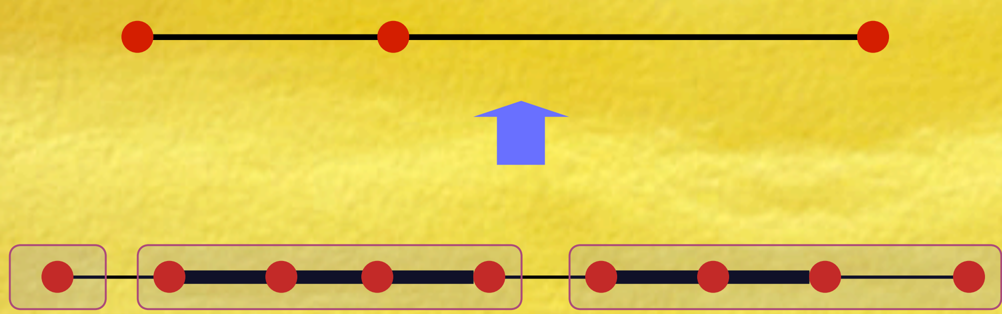
Outline

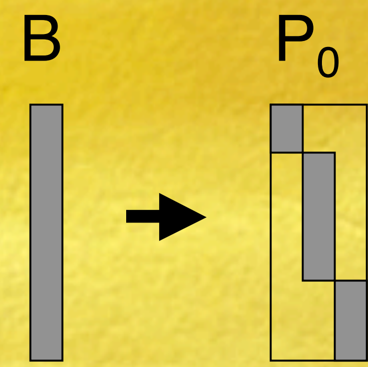
- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Smoothed Aggregation

Piecewise constant function: “Plain” agg. (P_0)

- Start with kernel vectors B of operator
  - eg, 6 RBMs in elasticity
- Nodal aggregation





“Smoothed” aggregation: lower energy of functions

- One Jacobi iteration: $P \leftarrow (I - \omega D^{-1} A) P_0$

Outline

- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

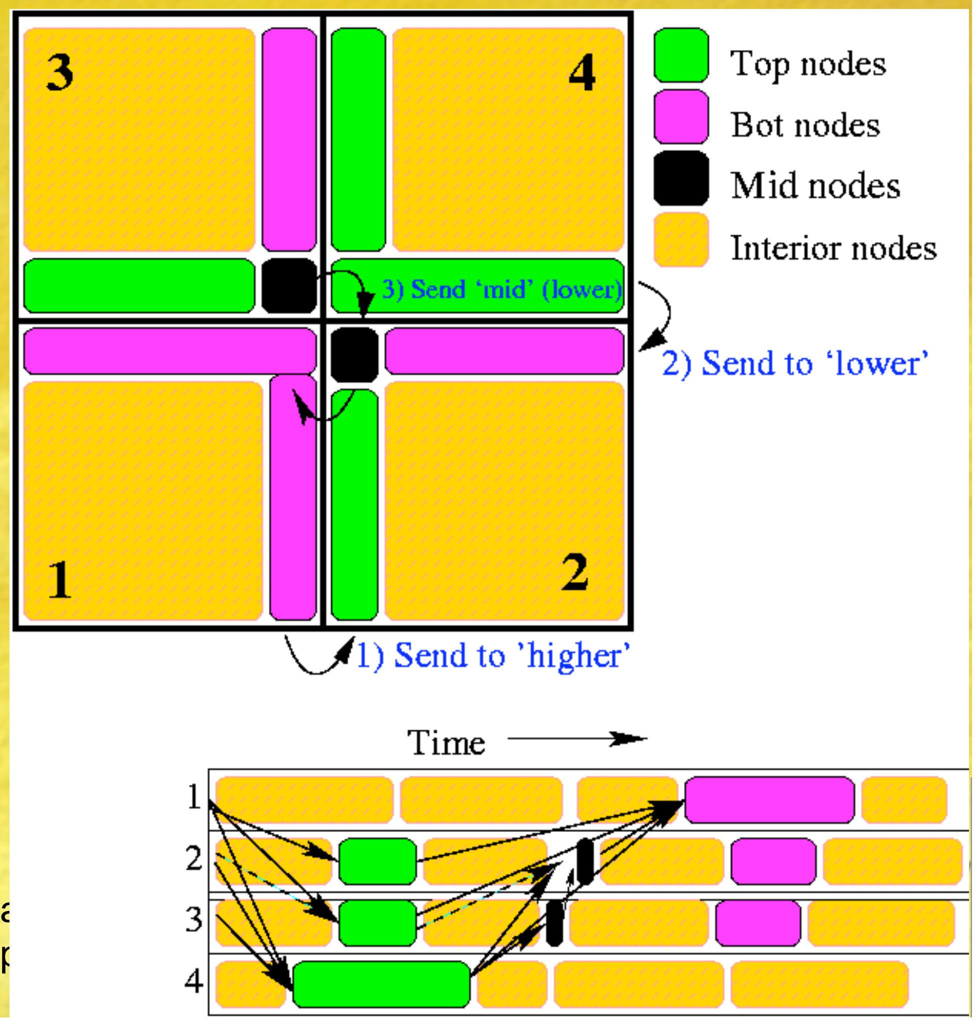
Smoothers

- CG/Jacobi: Additive
  - Essentially damped by CG - Adams SC1999
  - Dot products, non-stationary
- Gauss-Seidel: multiplicative (Optimal MG smoother)
  - Complex communication and comput. - Adams SC2001
- Polynomial Smoothers: Additive
  - Chebyshev ideal for MG - Adams et.al. JCP 2003
  - Chebychev chooses *p(y)* such that
    - $|1 - p(y)\, y| = min$ over interval $[\lambda^*, \lambda_{max}]$
    - Estimate of λ_{max} easy
    - Use $\lambda^* = \lambda_{max} / C$ (No need for lowest eigenvalue)
      - C related to rate of grid coarsening

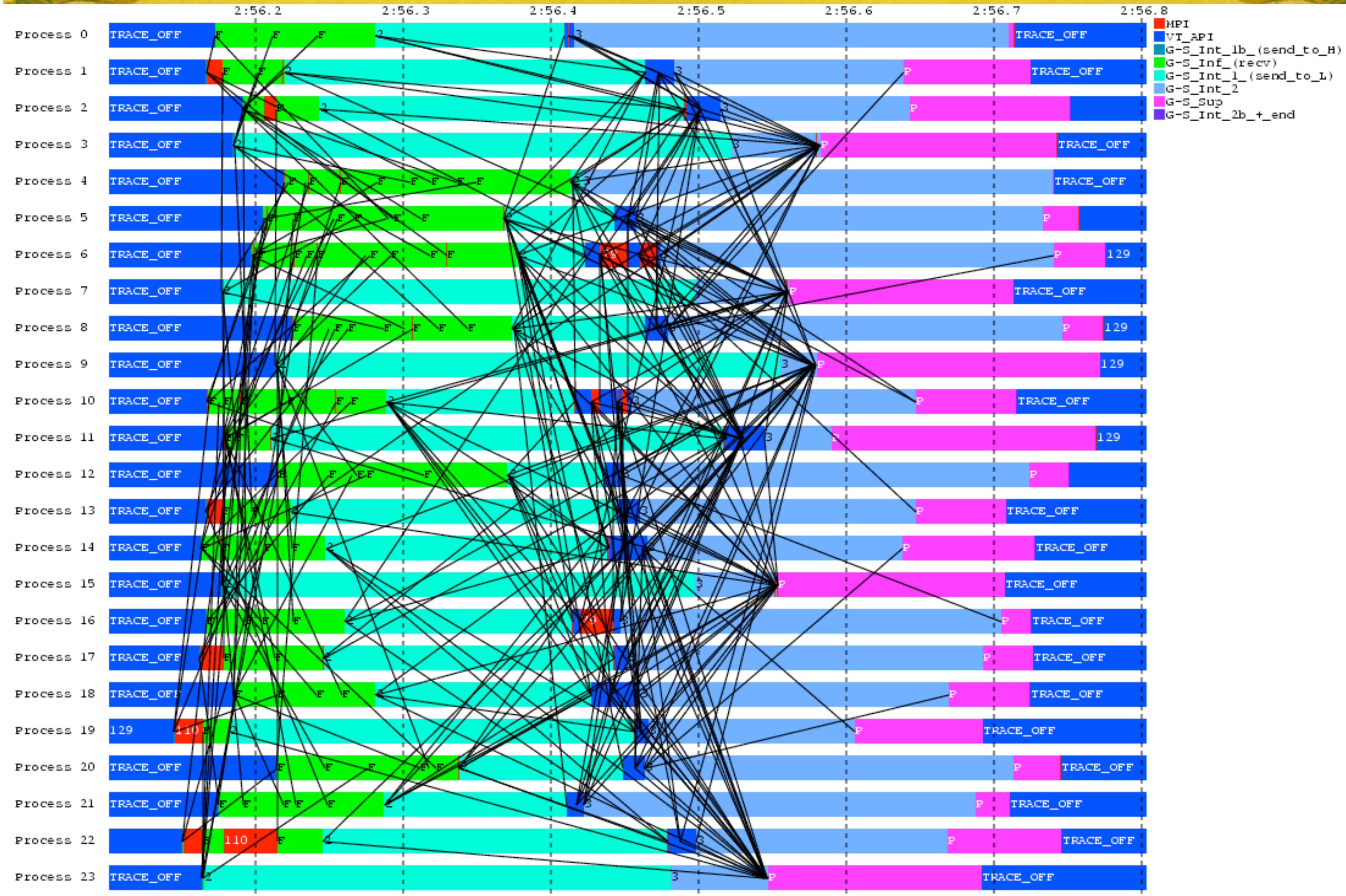
Parallel Gauss-Seidel

- Multiplicative smoothers
  - (+) Powerful
  - (+) Great for MG
  - (-) Difficult to parallelize
- Ideas:
  - Use processor partitions
  - Use ‘internal’ work to hide communication
  - Symmetric!

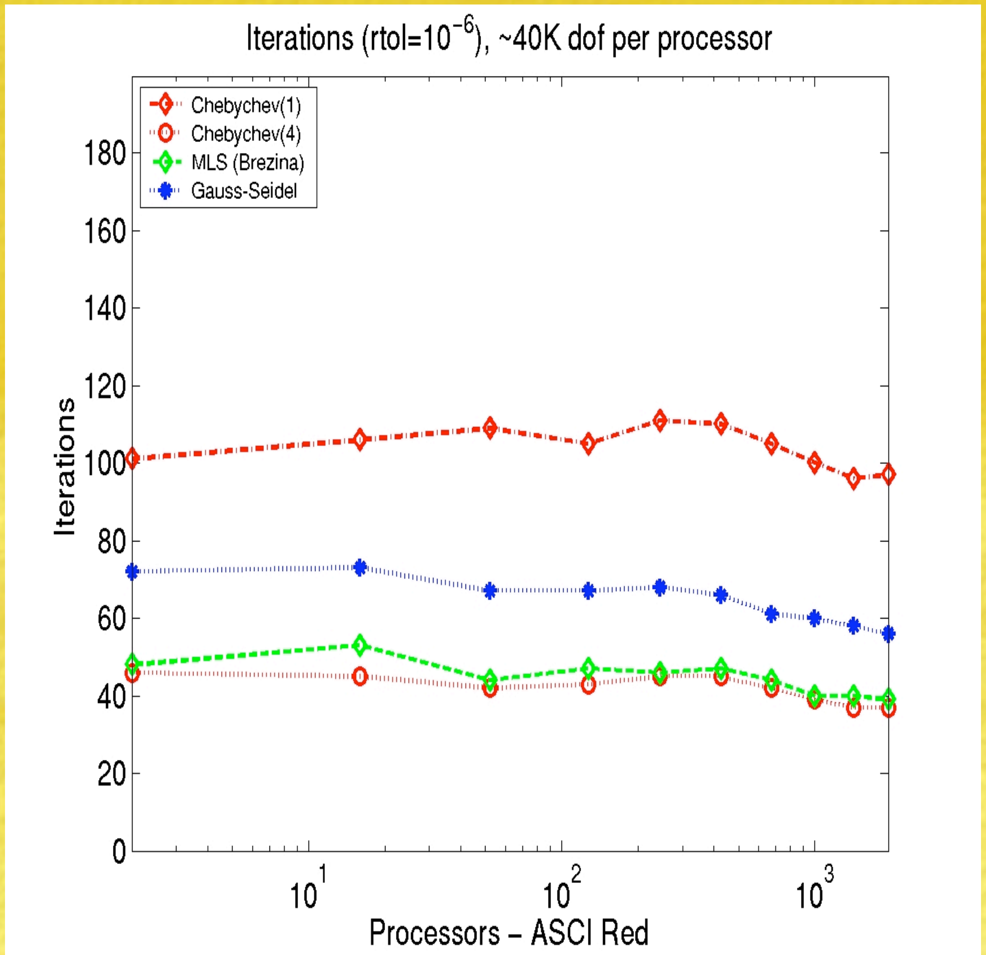
Example: 2D, 4 proc



Cray T3E - 24 Processors – About 30,000 dof Per Processor



Iteration counts (80K to 76M equations)

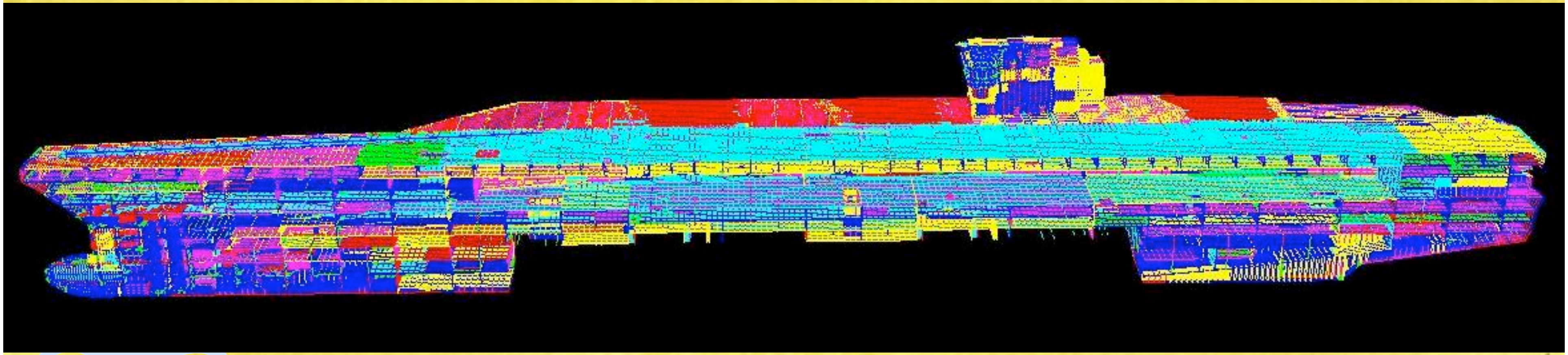


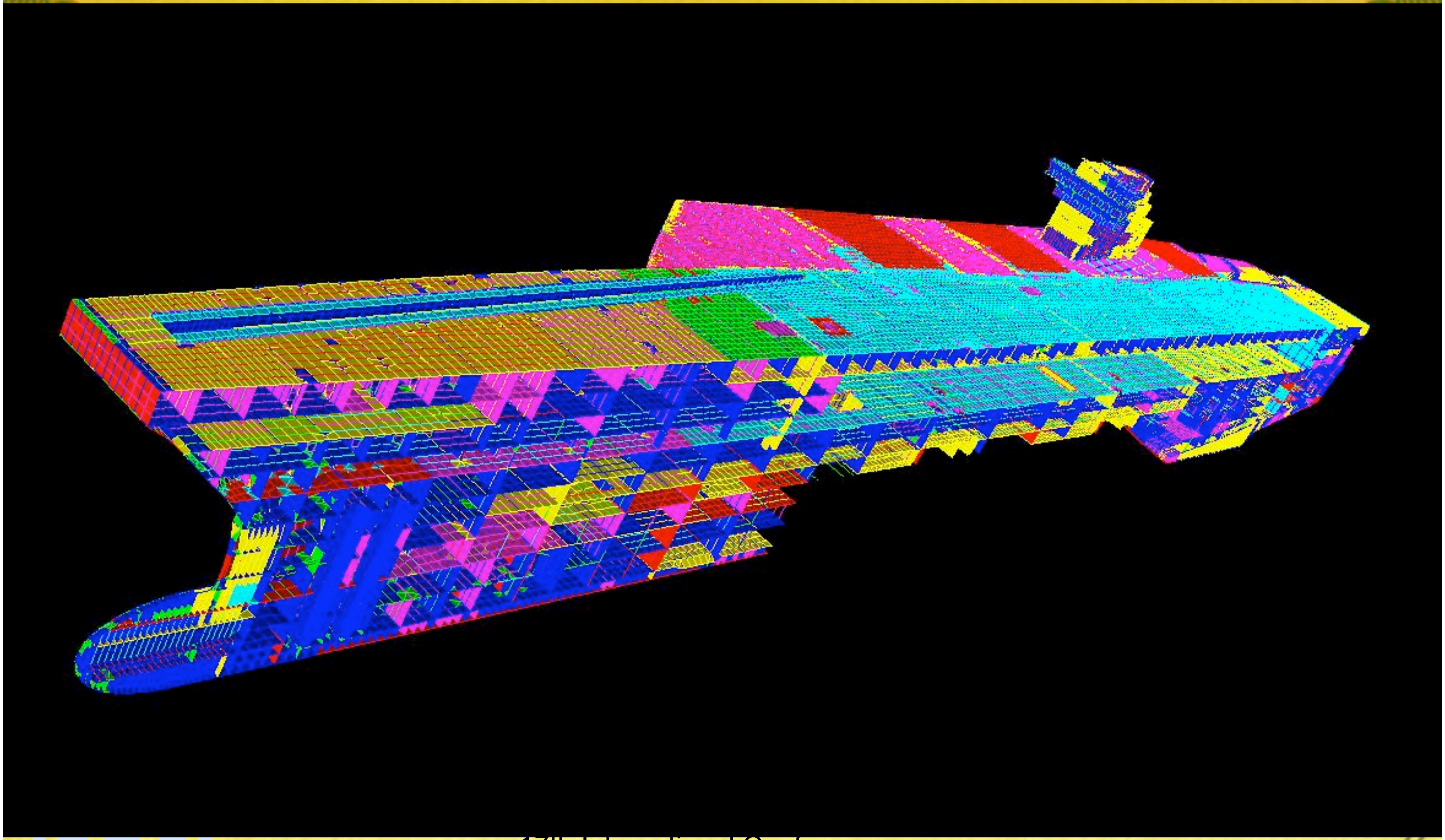
Outline

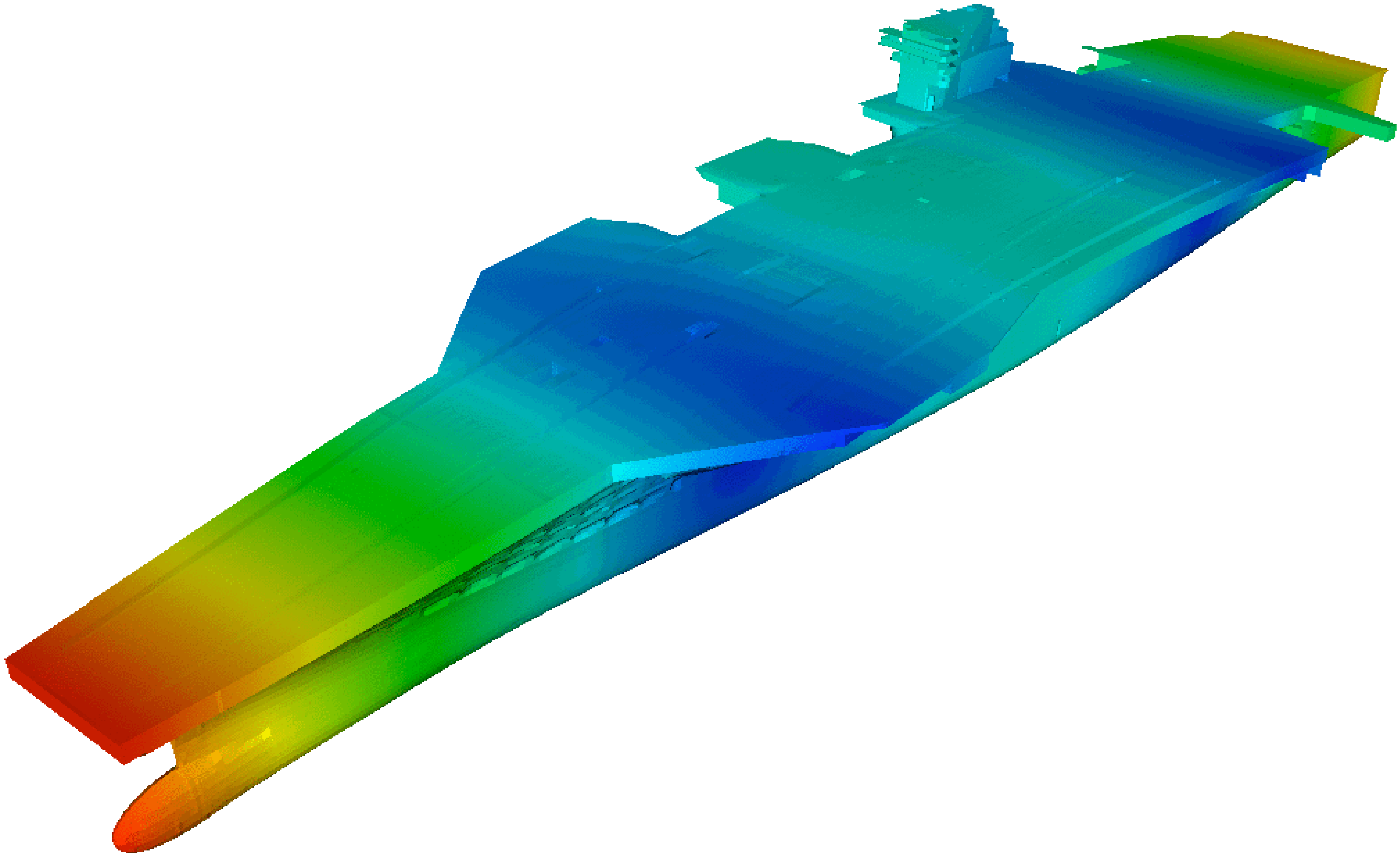
- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Aircraft carrier

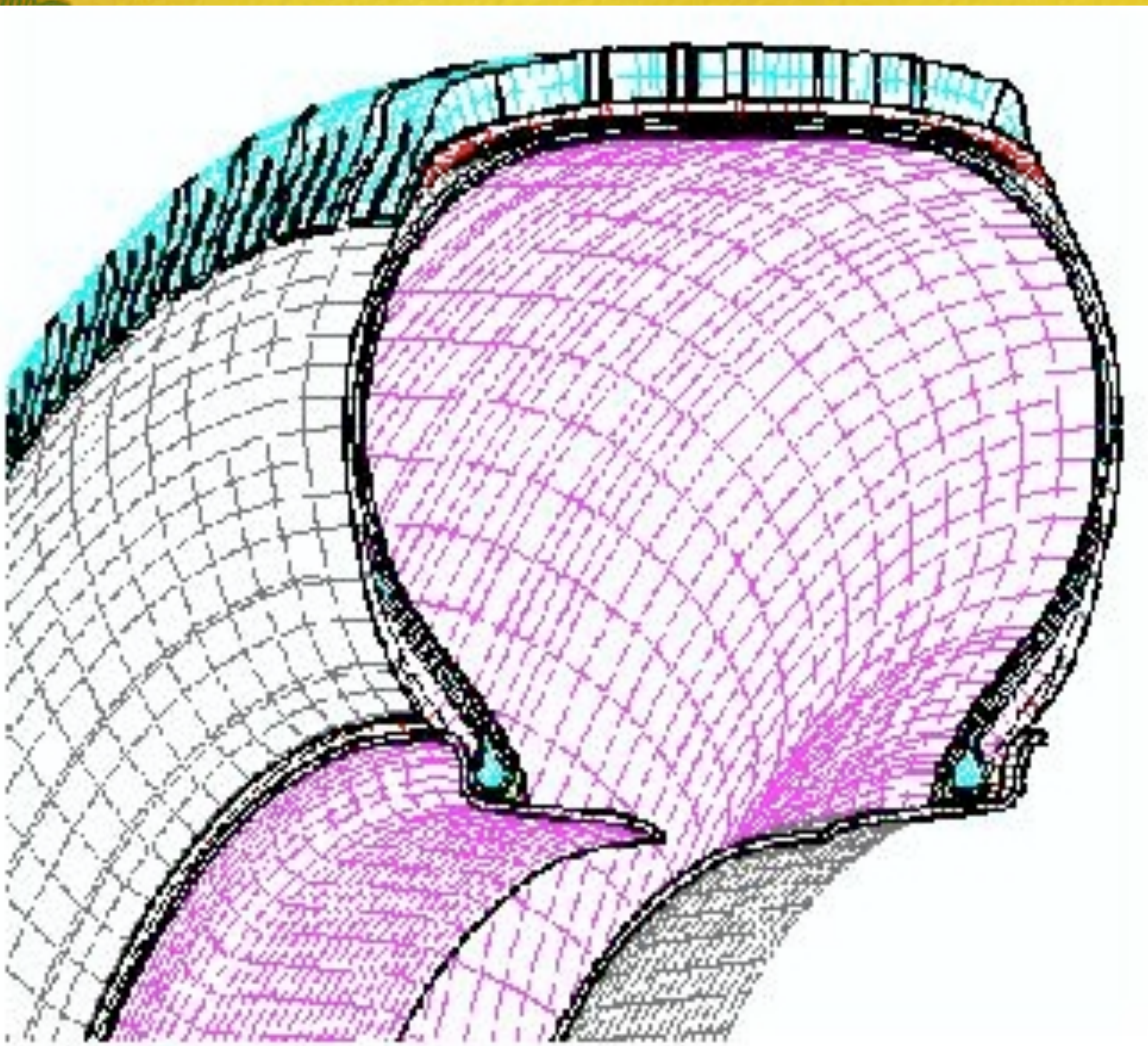
- 315,444 vertices
- Shell and beam elements (6 DOF per node)
- Linear dynamics – transient (time domain)
- About 1 min. per solve (rtol=10^{-6})
  - 2.4 GHz Pentium 4/Xenon processors
  - Matrix vector product runs at 254 Mflops



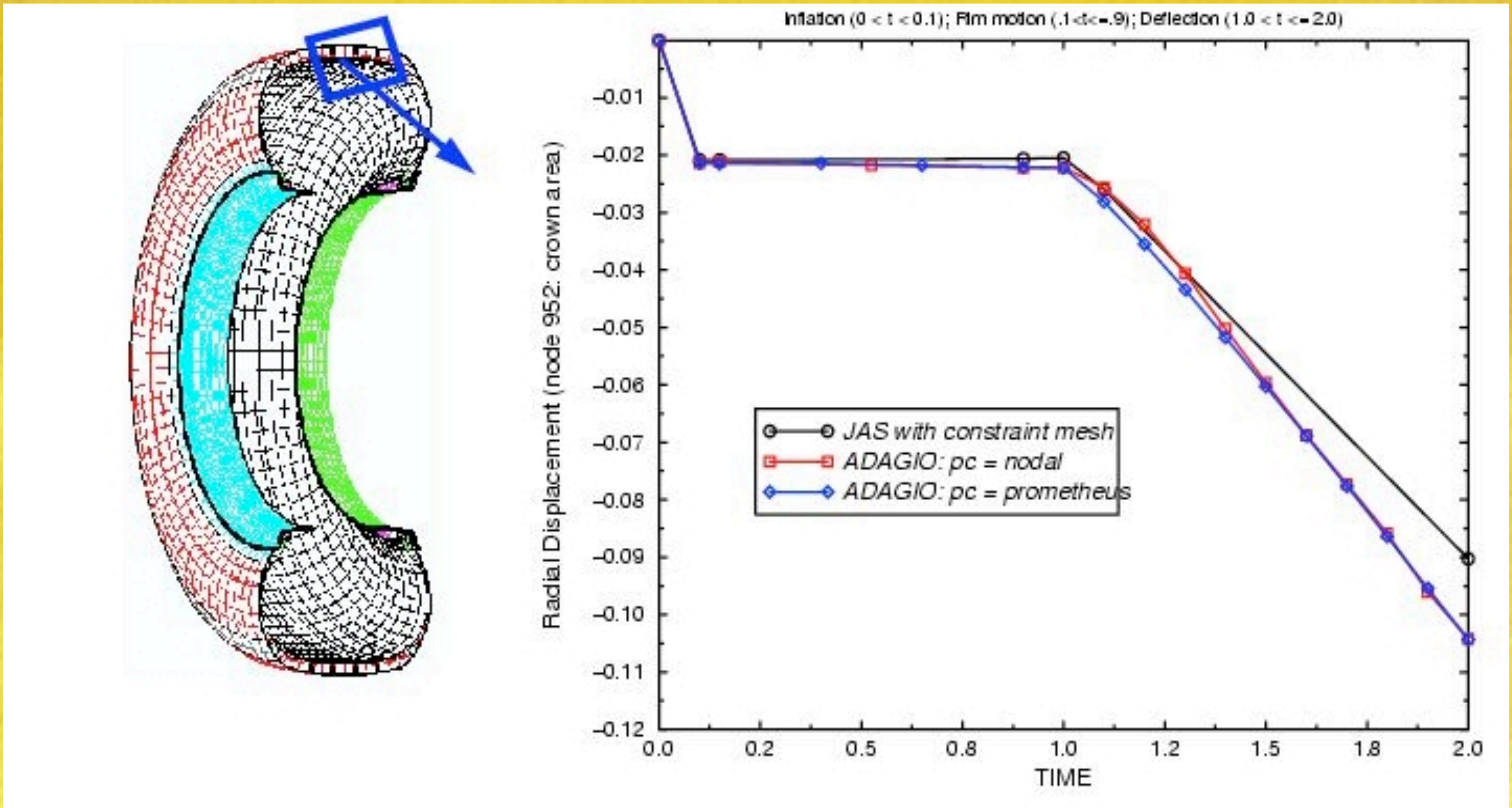




"BR" tire



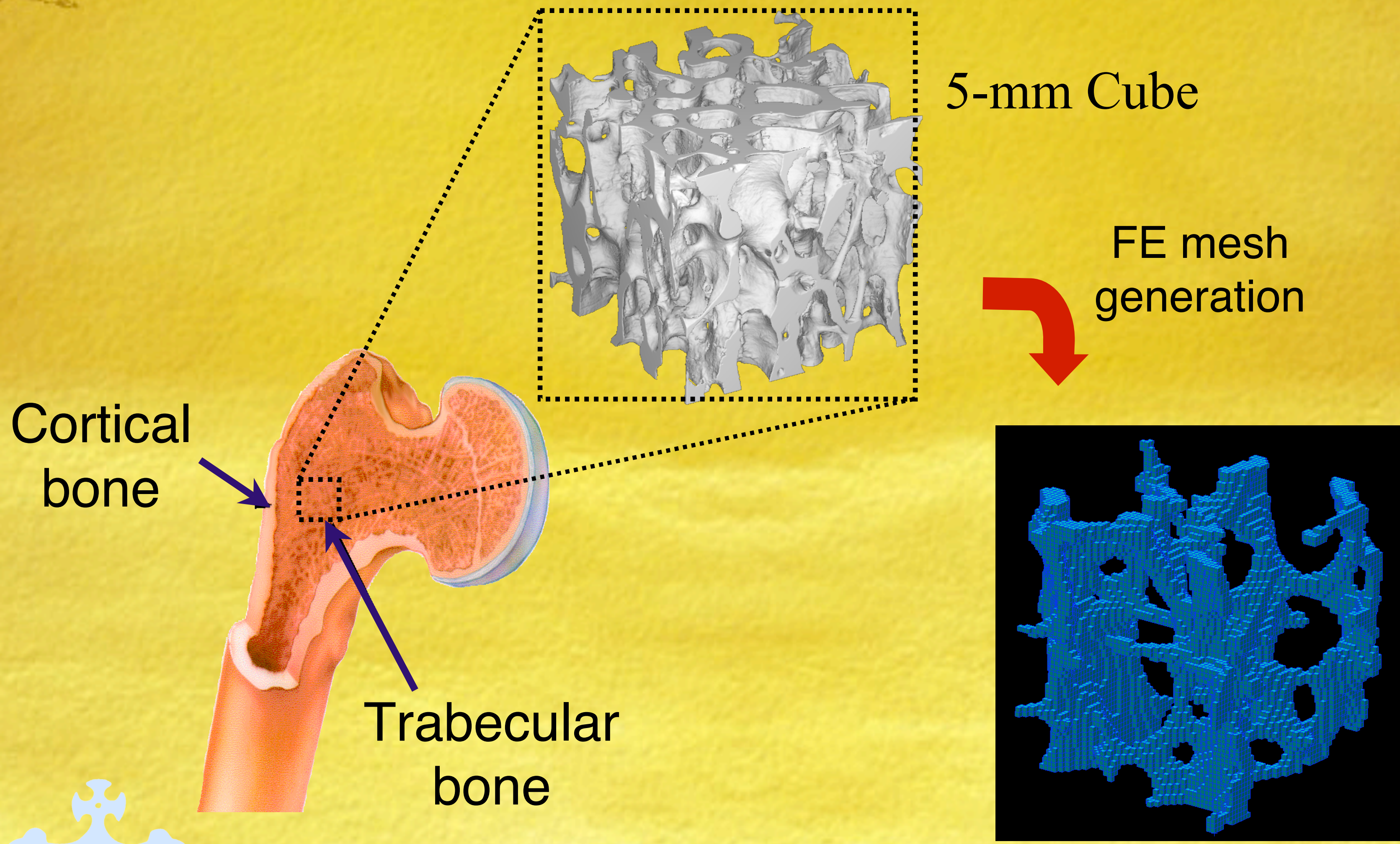
Math does matter!



Outline

- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Trabecular Bone



Outline

- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Computational Architecture

Athena: Parallel FE

ParMetis

  Parallel Mesh Partitioner *(Univerisity of Minnesota)*

Prometheus

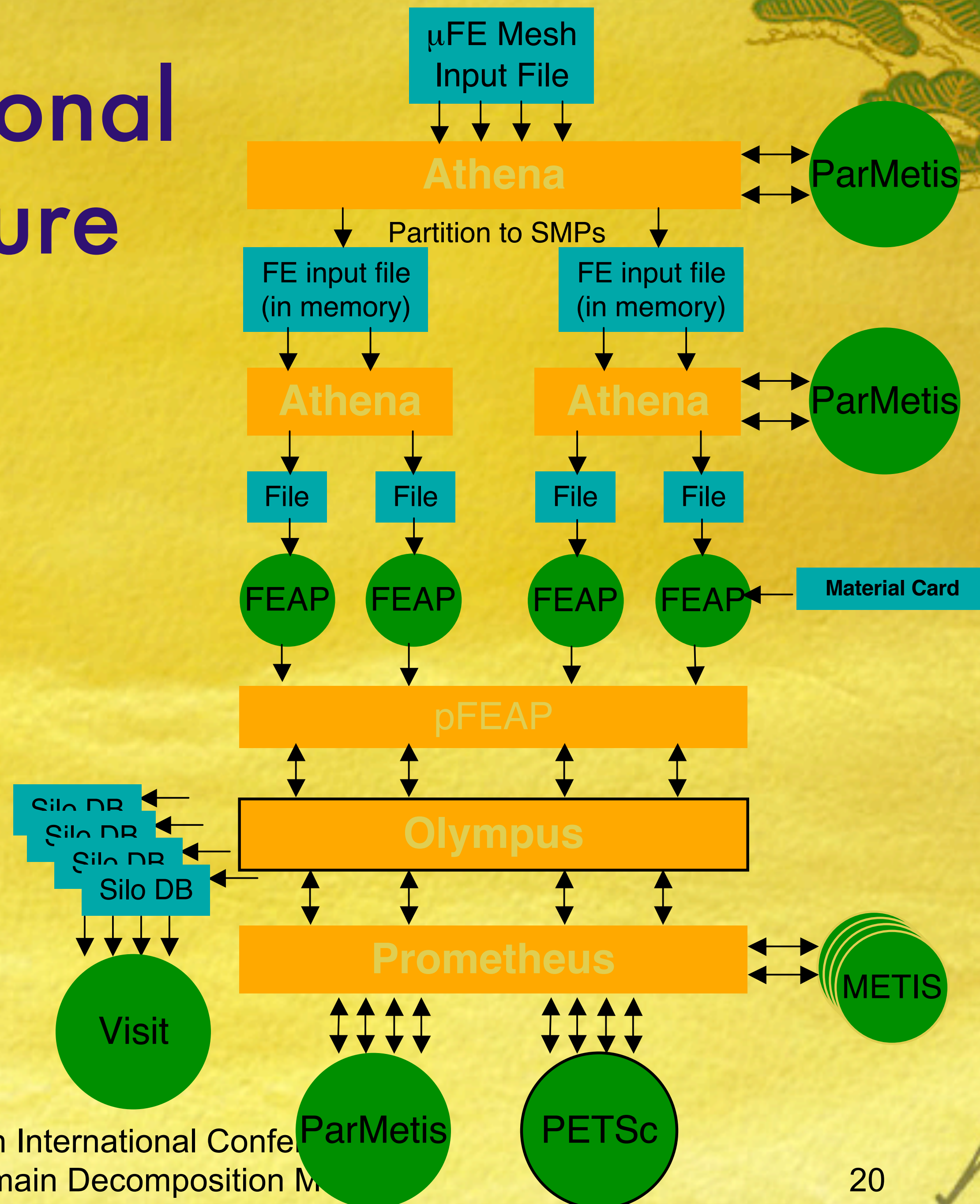
  Multigrid Solver

FEAP

  Serial general purpose FE application *(University of California)*

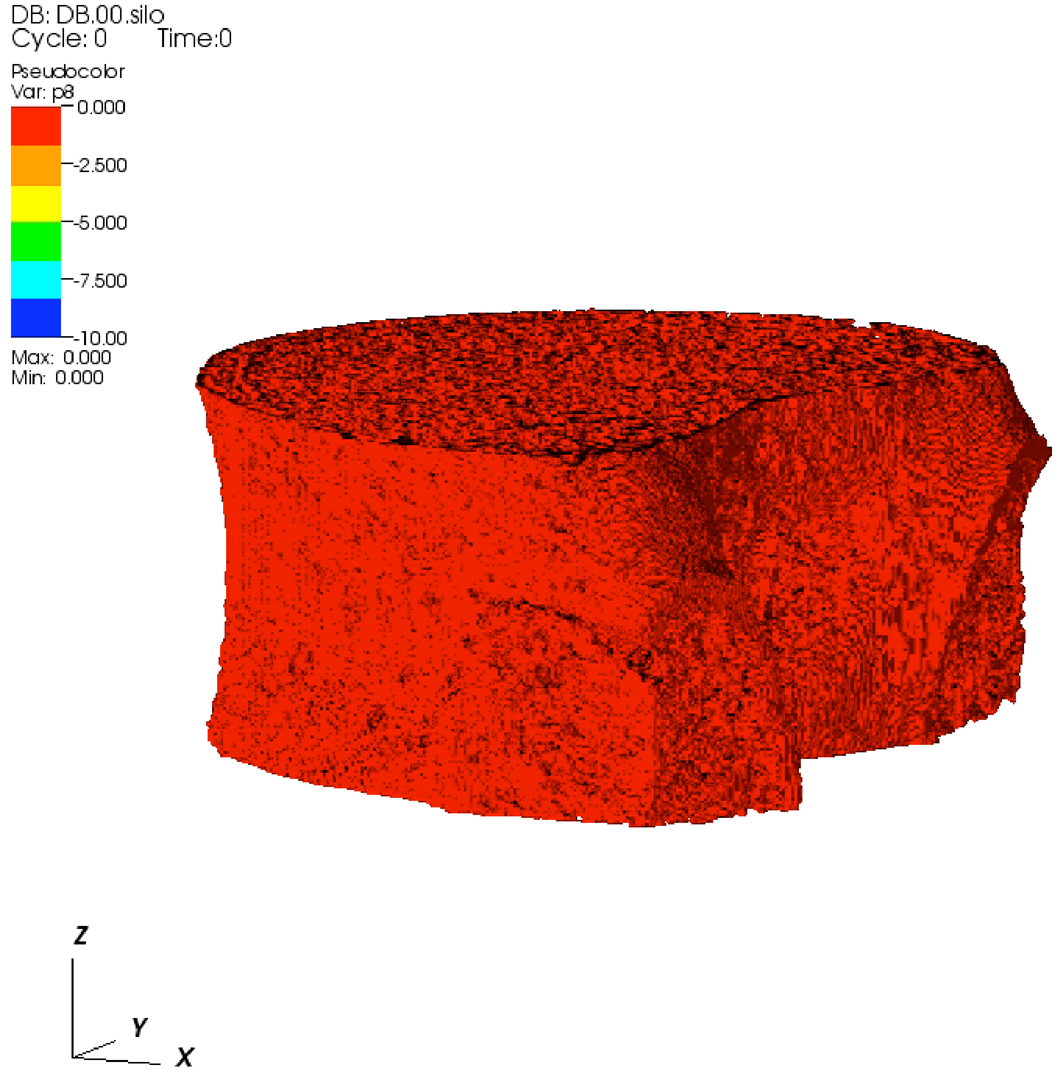
PETSc

  Parallel numerical libraries *(Argonne National Labs)*



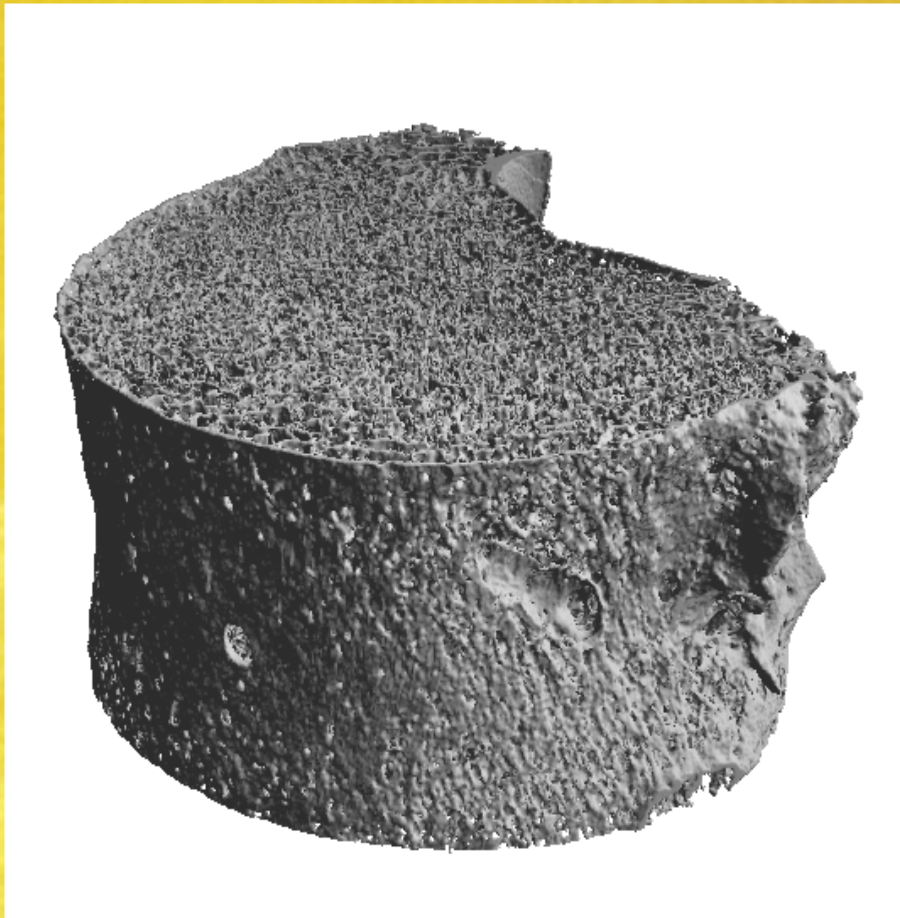
Viz:

- Geometric & Material non-linear
- 2.25% strain
- 8 procs. DataStar (SP4 at UCSD)



Scalability: Vertebral Body

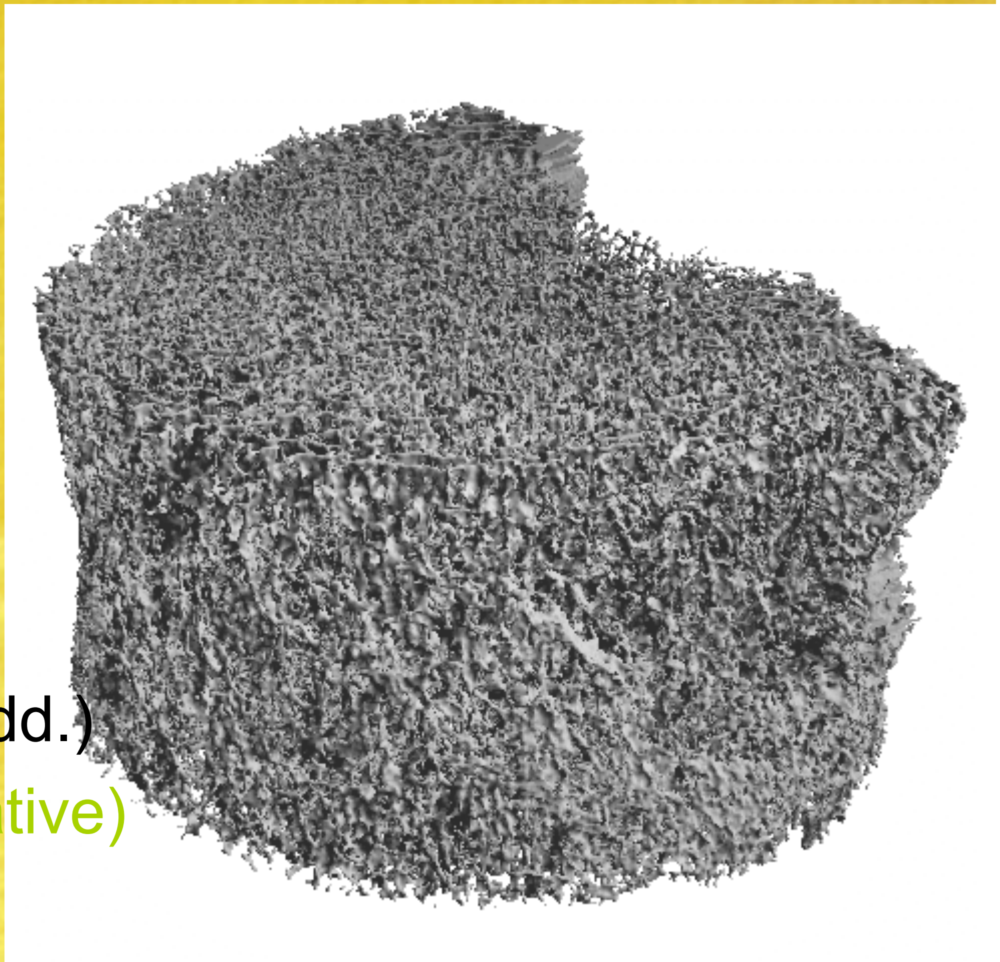
- Large deformation elast.
- 6 load steps (3% strain)
- Scaled speedup
  - ~131K dof/processor
- 7 to 537 million dof
- 4 to 292 nodes
- IBM SP Power3
  - 14 of 16 procs/node used
- Double/Single Colony switch



80 μm w/ shell

Scalability

- Inexact Newton
- CG linear solver
  - Variable tolerance
- Smoothed aggregation AMG preconditioner
- (vertex block) Diagonal smoothers:
  - 2nd order Chebeshev (add.)
  - Gauss-Seidel (multiplicative)

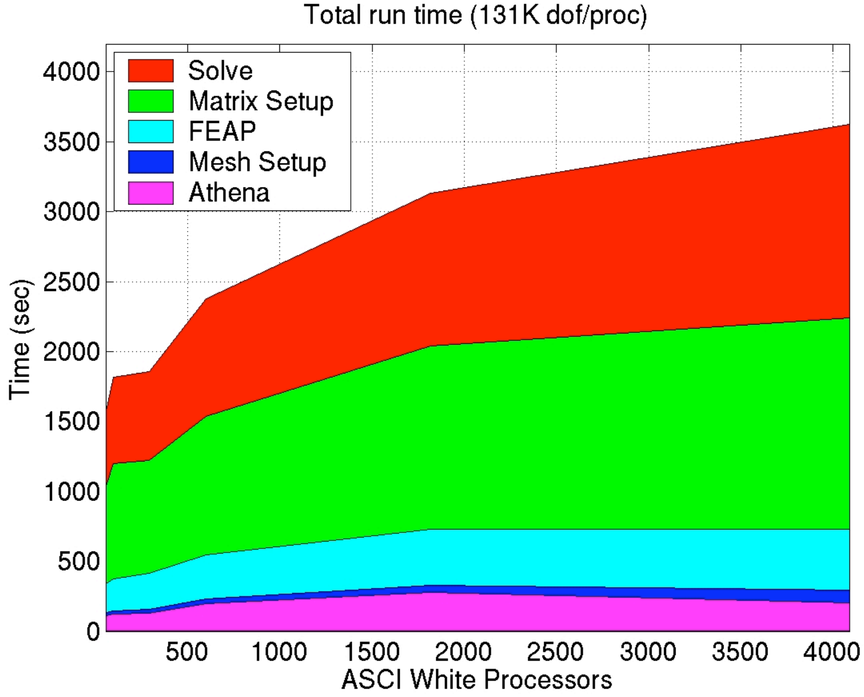


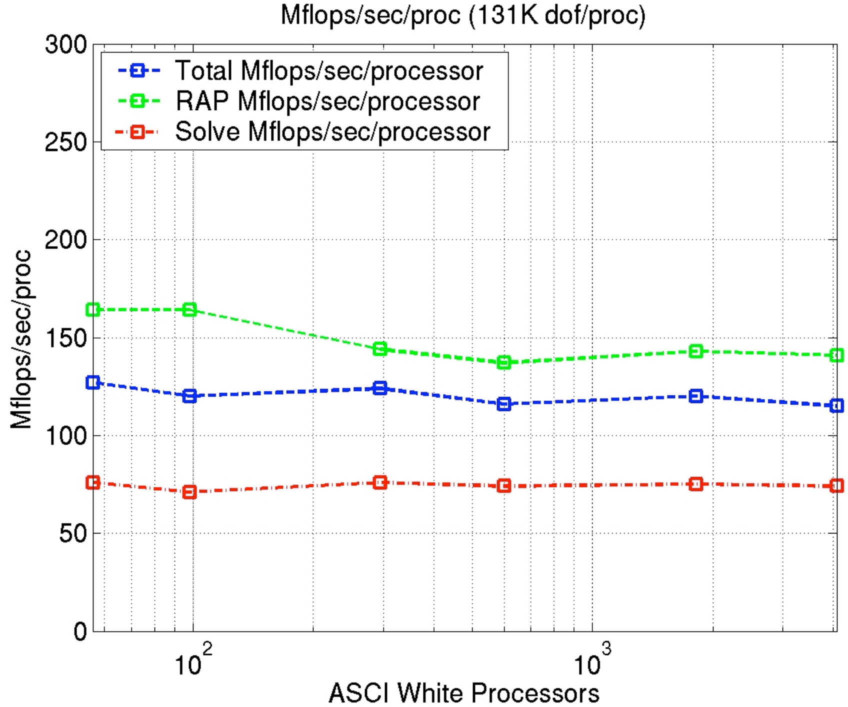
80 μm w/o shell

Computational phases

- Mesh setup (per mesh):
  - Coarse grid construction (aggregation)
  - Graph processing
- Matrix setup (per matrix):
  - Coarse grid operator construction
    - Sparse matrix triple product RAP (expensive for S.A.)
  - Subdomain factorizations
- Solve (per RHS):
  - Matrix vector products (residuals, grid transfer)
  - Smoothers (Matrix vector products)

131K dof/proc - Flops/sec/proc
.47 Teraflop/s - 4088 processors





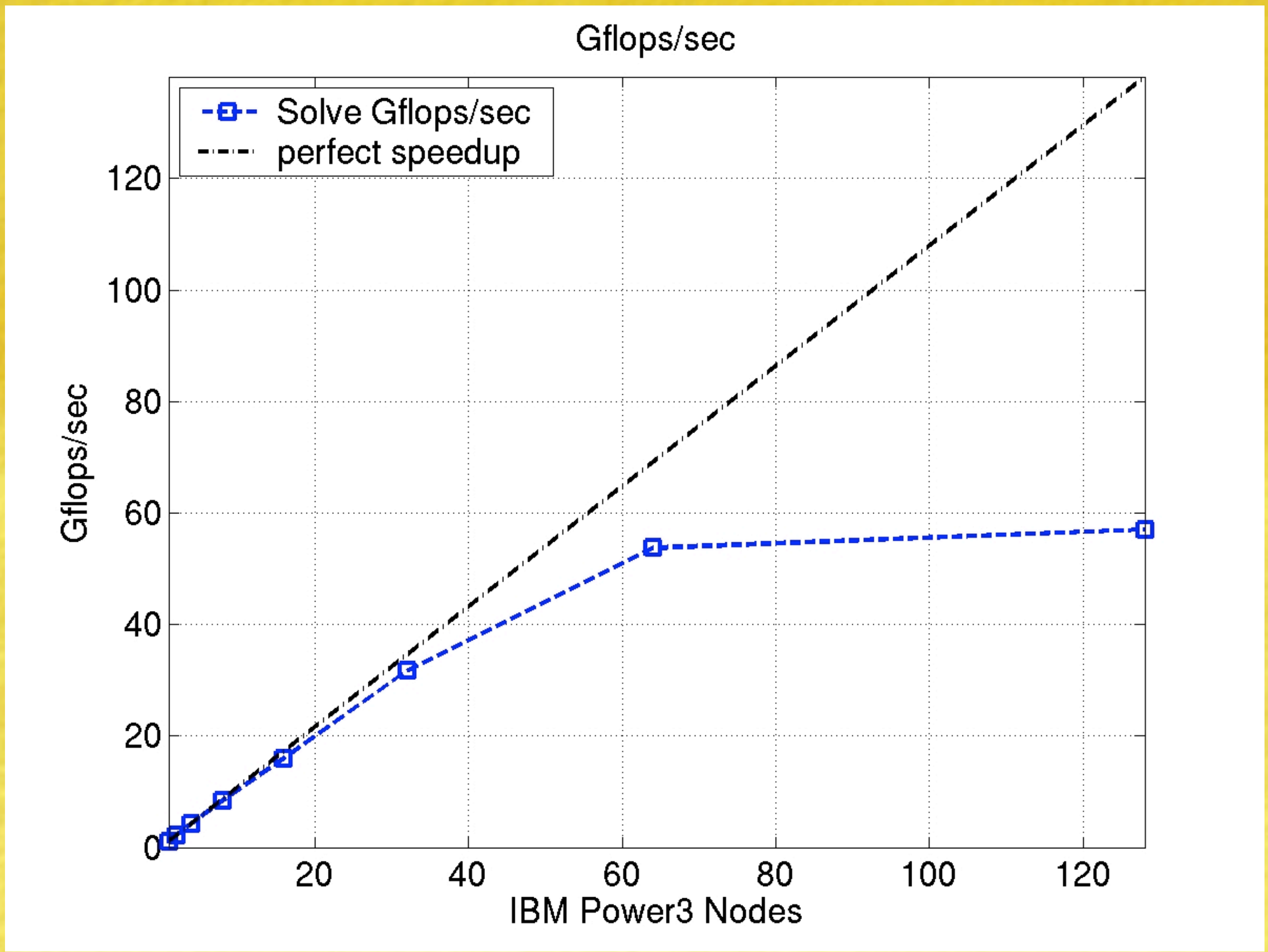
Sources of inefficiencies: Linear solver iterations

Load \ Newton	Small (7.5M dof)					Large (537M dof)					
	1	2	3	4	5	1	2	3	4	5	6
1	5	14	20	21	18	5	11	35	25	70	2
2	5	14	20	20	20	5	11	36	26	70	2
3	5	14	20	22	19	5	11	36	26	70	2
4	5	14	20	22	19	5	11	36	26	70	2
5	5	14	20	22	19	5	11	36	26	70	2
6	5	14	20	22	19	5	11	36	26	70	2

Sources of scale inefficiencies in solve phase

	7.5M dof	537M dof
#iteration	450	897
#nnz/row	50	68
Flop rate	76	74
#elems/pr	19.3K	33.0K
model	1.00	2.78
Measured run time	1.00	2.61

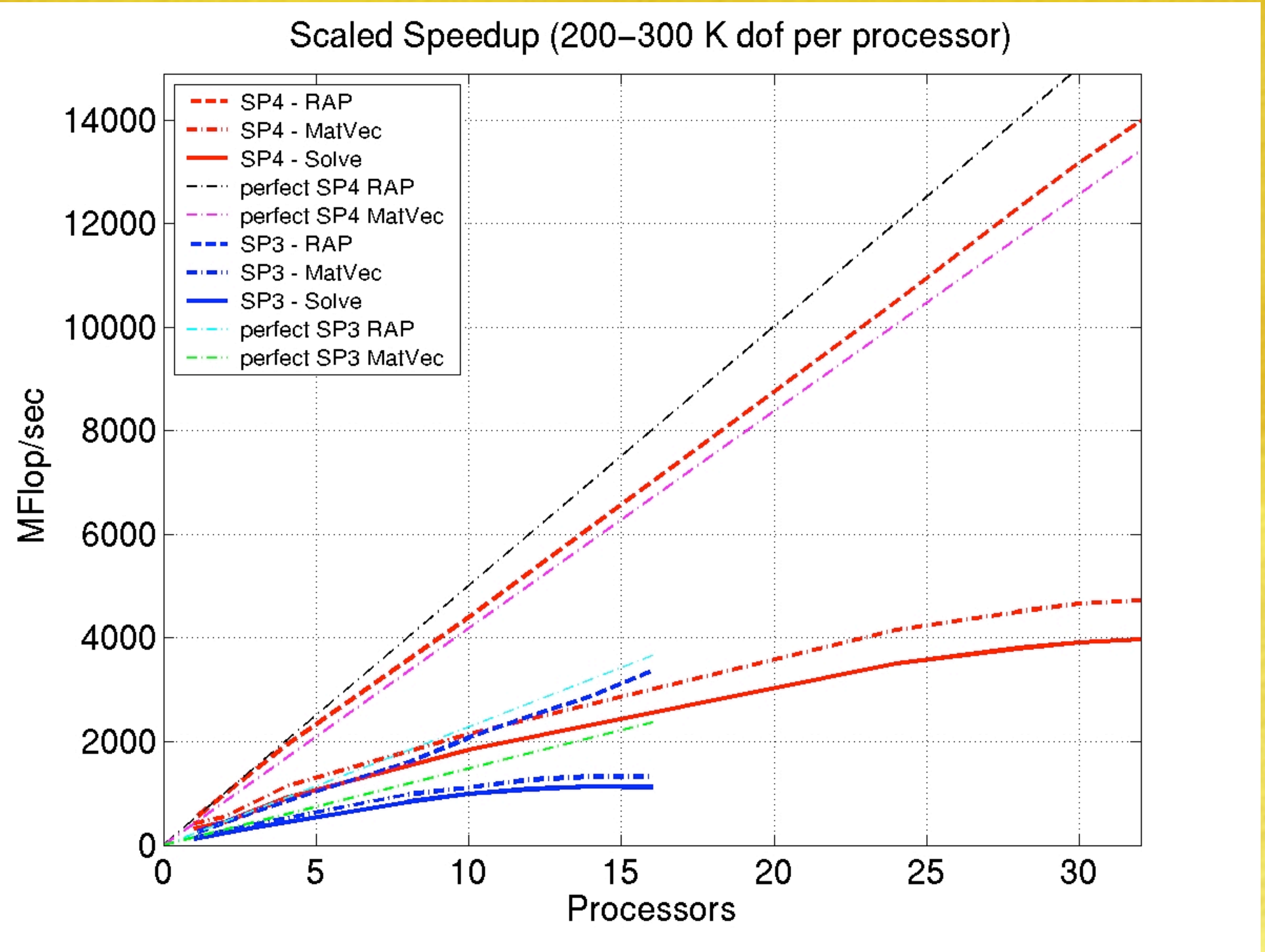
Strong speedup: 7.5M dof (1 to 128 nodes)



Nodal Performance of IBM SP Power3 and Power4

- IBM power3, 16 processors per node
  - 375 Mhz, 4 flops per cycle
  - 16 GB/sec bus (~7.9 GB/sec w/ STREAM bm)
    - Implies ~1.5 Gflops/s MB peak for Mat-Vec
    - We get ~1.2 Gflops/s (15 x .08Gflops)
- IBM power4, 32 processors per node
  - 1.3 GHz, 4 flops per cycle
  - Complex memory architecture

Speedup



Scaled Speedup (200–300 K dof per processor)
SP4 - RAP
SP4 - MatVec
SP4 - Solve
perfect SP4 RAP
perfect SP4 MatVec
SP3 - RAP
SP3 - MatVec
SP3 - Solve
perfect SP3 RAP
perfect SP3 MatVec
MFlop/sec
14000
12000
10000
8000
6000
4000
2000
0
0
5
10
15
20
25
30
Processors

Outline

- Algebraic multigrid (AMG)
  - Coarse grid spaces
  - Smoothers: Add. (Cheb.) and Mult. (G-S)
  - Industrial applications
- Micro-FE bone modeling
  - Scalability/performance studies
    - Weak and strong (scaled/unscaled) speedup
- Multigrid algorithms for KKT system
  - New AMG framework for KKT systems

Constrained Linear Systems With Lagrange Multipliers

- Solid mechanics
  - **Contact**, tied meshes
  - RBE3s …
- Mixed methods
- Incompressible flow
- Optimization problems ….

$$\begin{pmatrix} \mathbf{K} & \mathbf{C}^{\mathrm{T}} \\ \mathbf{C} & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \lambda \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}$$

AMG for Constrained Systems (KKT-AMG)

$$\begin{pmatrix} \mathbf{K}_{i+1} & \mathbf{C}_{i+1}^{T} \\ \mathbf{C}_{i+1} & \mathbf{0} \end{pmatrix} \Leftarrow \begin{pmatrix} \mathbf{R}_i & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{R}}_i \end{pmatrix} \begin{pmatrix} \mathbf{K}_i & \mathbf{C}_i^{T} \\ \mathbf{C}_i & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{R}_i^{T} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{R}}_i^{T} \end{pmatrix}$$

1) Use Identity: $\mathbf{I}$

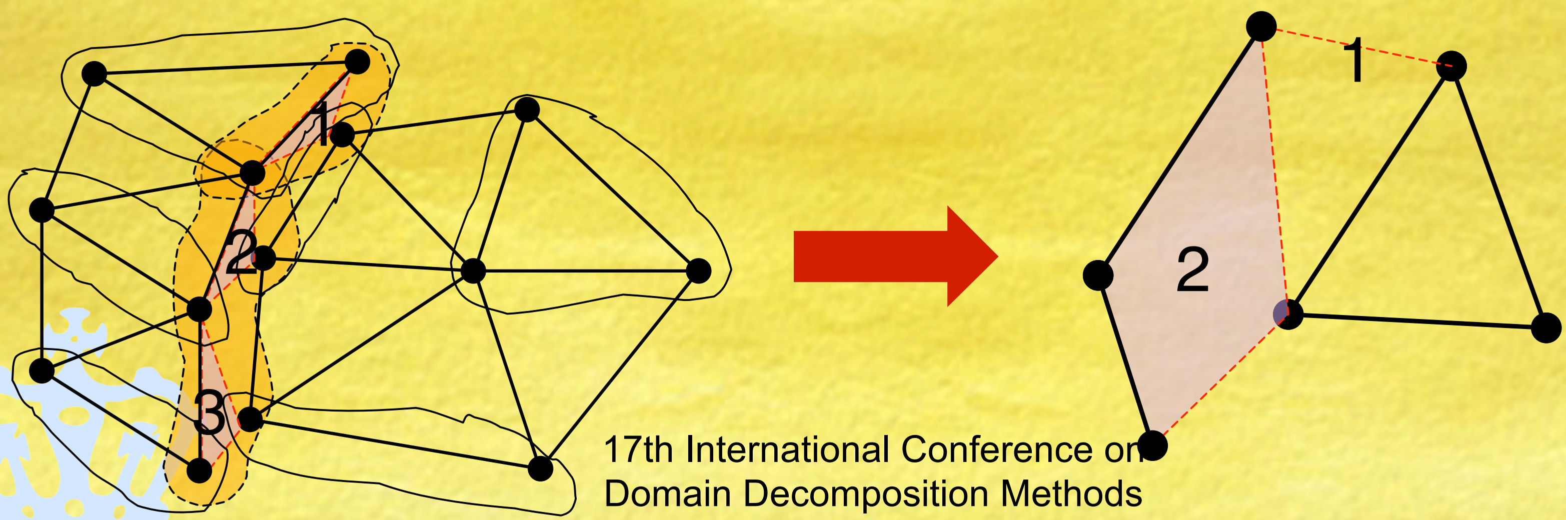
$$\begin{pmatrix} \mathbf{K}_{i+1} & \mathbf{C}_{i+1}^{T} \\ \mathbf{C}_{i+1} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{R}\mathbf{K}_i\mathbf{P} & \mathbf{R}\mathbf{C}_i^{T} \\ \mathbf{C}_i\mathbf{P} & \mathbf{0} \end{pmatrix} \Leftarrow \begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{K}_i & \mathbf{C}_i^{T} \\ \mathbf{C}_i & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{P} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{pmatrix}$$

2) Constraint coarsening: $\overline{\mathbf{R}}_{i+1}^{i}$

$$\begin{pmatrix} \mathbf{K}_{i+1} & \mathbf{C}_{i+1}^{T} \\ \mathbf{C}_{i+1} & \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{R}_i^{i+1}\mathbf{K}_i\mathbf{P}_{i+1}^{i} & \mathbf{R}_i^{i+1}\mathbf{C}_i^{T}\overline{\mathbf{P}}_{i+1}^{i} \\ \overline{\mathbf{R}}_i^{i+1}\mathbf{C}_i\mathbf{P}_{i+1}^{i} & \mathbf{0} \end{pmatrix} \Leftarrow \begin{pmatrix} \mathbf{R}_{i+1}^{i} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{R}}_{i+1}^{i} \end{pmatrix} \begin{pmatrix} \mathbf{K}_i & \mathbf{C}_i^{T} \\ \mathbf{C}_i & \mathbf{0} \end{pmatrix} \begin{pmatrix} \mathbf{P}_{i+1}^{i} & \mathbf{0} \\ \mathbf{0} & \overline{\mathbf{P}}_{i+1}^{i} \end{pmatrix}$$

Coarse Grid Space

- Start simple: plain aggregation
- Standard AMG graph problem
  - Aggregate strongly connected domains
- But what graph T? (ie, symmetric matrix)
  - **$T=CC^T$**
  - **$T=CYC^T$**
    - **$Y=PP^T$: $T = CPP^TC^T = (CP)(CP)^T$**



Motivation for T = CPP^{T}C^T (with piecewise const. L.M. spaces)

Consider $\mathbf{C}_{l+1}\mathbf{C}_{l+1}^T = \left(\overline{\mathbf{P}}^T \mathbf{C}_l \mathbf{P}\right)\left(\mathbf{P}^T \mathbf{C}_l^T \overline{\mathbf{P}}\right) = \overline{\mathbf{P}}^T \mathbf{T}\, \overline{\mathbf{P}}$

Recall $\mathbf{C}_{l+1} = \overline{\mathbf{P}}^T \mathbf{C}_l \mathbf{P}$

Thus $\mathbf{C}_{l+1}\mathbf{C}_{l+1}^T[I,J] = \overline{\mathbf{P}}_{iI} \mathbf{T}_{ij} \overline{\mathbf{P}}_{jJ} = \sum_{i\in I,\, j\in J} \mathbf{T}_{ij}$

$$\cos(\theta_{\mathrm{IJ}}) \le \max_{i\in I,\, j\in J} \mathbf{T}_{ij}$$

Smoothers for Constrained Systems

- Constraint centric Schwarz (Vanka)
  - Multiplicative
  - Additive
- Segregated, PC Uzawa (Braess,…)
  - Additive
- ILU (Shultz, Taskflow)
  - Level fill (1)

Segregated Smoothers for Constrained Systems

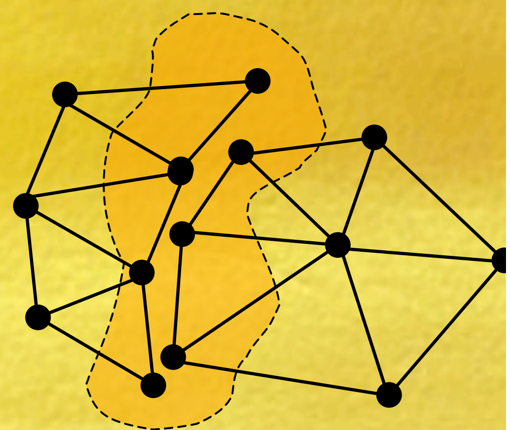
$$\begin{pmatrix} K & C^T \\ C & 0 \end{pmatrix} = \begin{pmatrix} K & 0 \\ C & -S \end{pmatrix} \begin{pmatrix} I & K^{-1}C^T \\ 0 & I \end{pmatrix} \qquad S = CK^{-1}C^T$$

$$\begin{pmatrix} K & 0 \\ C & -S \end{pmatrix}^{-1} \begin{pmatrix} K & C^T \\ C & 0 \end{pmatrix} = \begin{pmatrix} I & K^{-1}C^T \\ 0 & I \end{pmatrix}$$

- Primal smoother M^{-1} for K^{-1}
- Dual smoother Q^{-1} for S^{-1}
  - $u \leftarrow u_0 + M^{-1} (f - C^T p_0 - A u_0)$
  - $p \leftarrow p_0 + Q^{-1} (C u - g)$
  - Preconditioned Uzawa
  - Symmetrize: $u \leftarrow u + M^{-1} C^T(p_0 - p)$
- $Q = C D^{-1}C^T$, Processor block Jacobi

Constraint Centric Schwarz Smoothers

- Use Domain Decomposition (Schwarz) ideas (overlapping)
- Aggregate constraint equations into non-overlapping subdomains
  - Generalized Vanka (additive)
- For each constraint domain (processor)
  - Add primal eqs in support of constraints
  - Well posed KKT matrix (exact solve)



Constraint Centric Domain Decomposition Smoothers

- Multiplicative Schwarz
  - Additive constraint part: $B_d = B_1 + B_2 + \ldots B_k$
    - $B_i = R_i^T (R_i A R_i^T)^{-1} R_i$, exact subdomain solves
  - B_p: Multiplicative smoother for primal eqs.
    - $R^T M^{-1} R$, $R = [I\ 0]$, smoother M^{-1} for primal eqs
    - M^{-1}: Parallel Gauss-Seidel
  - $x \leftarrow x_0 + (B_p + B_d - B_d A B_p)(b - Ax_0)$
- Additive Schwarz (B_p: add. smoother)
  - $x \leftarrow x_0 + (B_p + B_d)(b - Ax_0)$

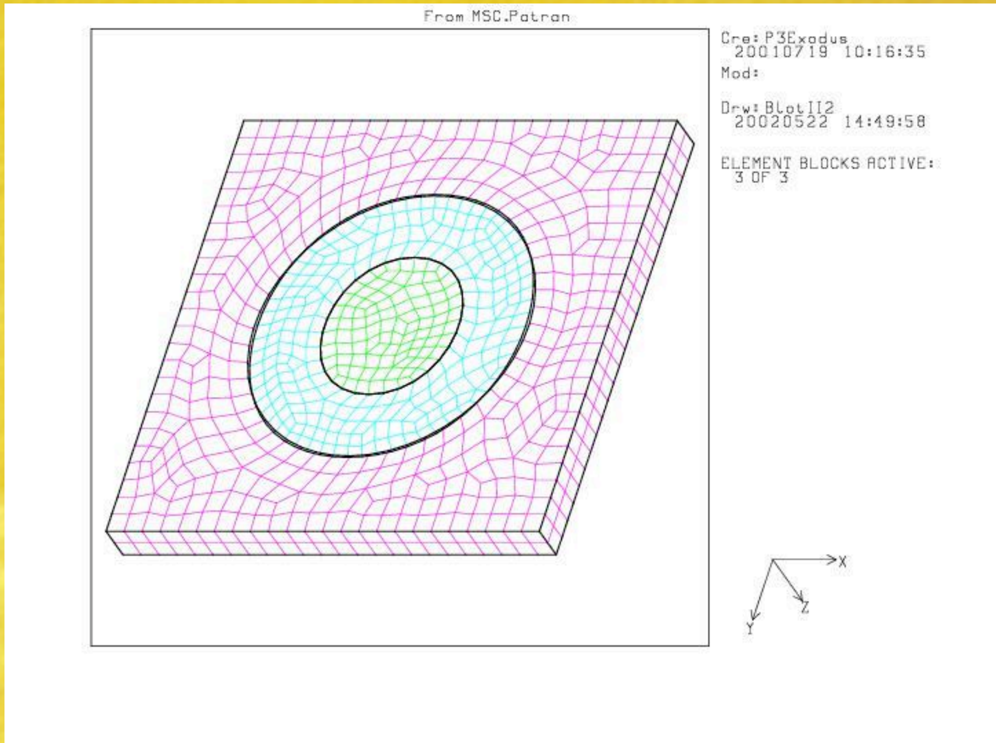
Numerical results

- Krylov solvers preconditioned
  - One V-cycle multigrid primal preconditioner
- Primal Smoother: Nodal diagonal PC
  - 1st order Chebychev (additive)
  - 1 iteration symmetric Gauss-Seidel (multiplicative)

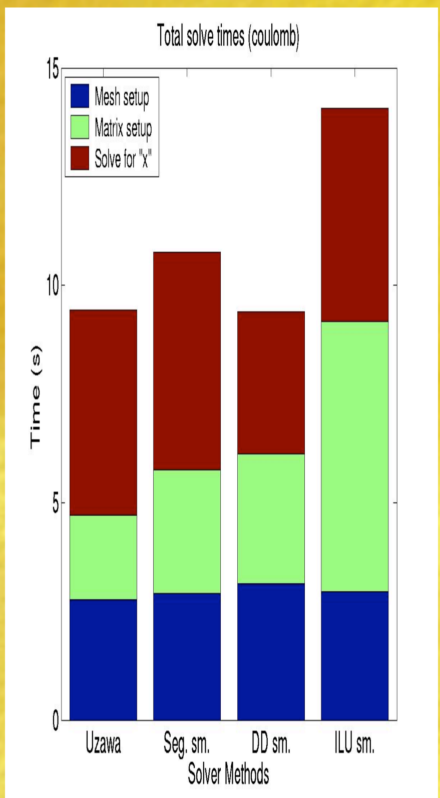
1) GMRES / KKT-AMG
   - Constraint centric DD
     - Additive
     - Multiplicative
   - Symmetric preconditioned Uzawa (segregated) iteration
     - Processor block Jacobi solver for $CD^{-1}C^T$
   - ILU level fill (1)
2) Uzawa outer iterations / CG inner iterations
   - Highly optimized production code

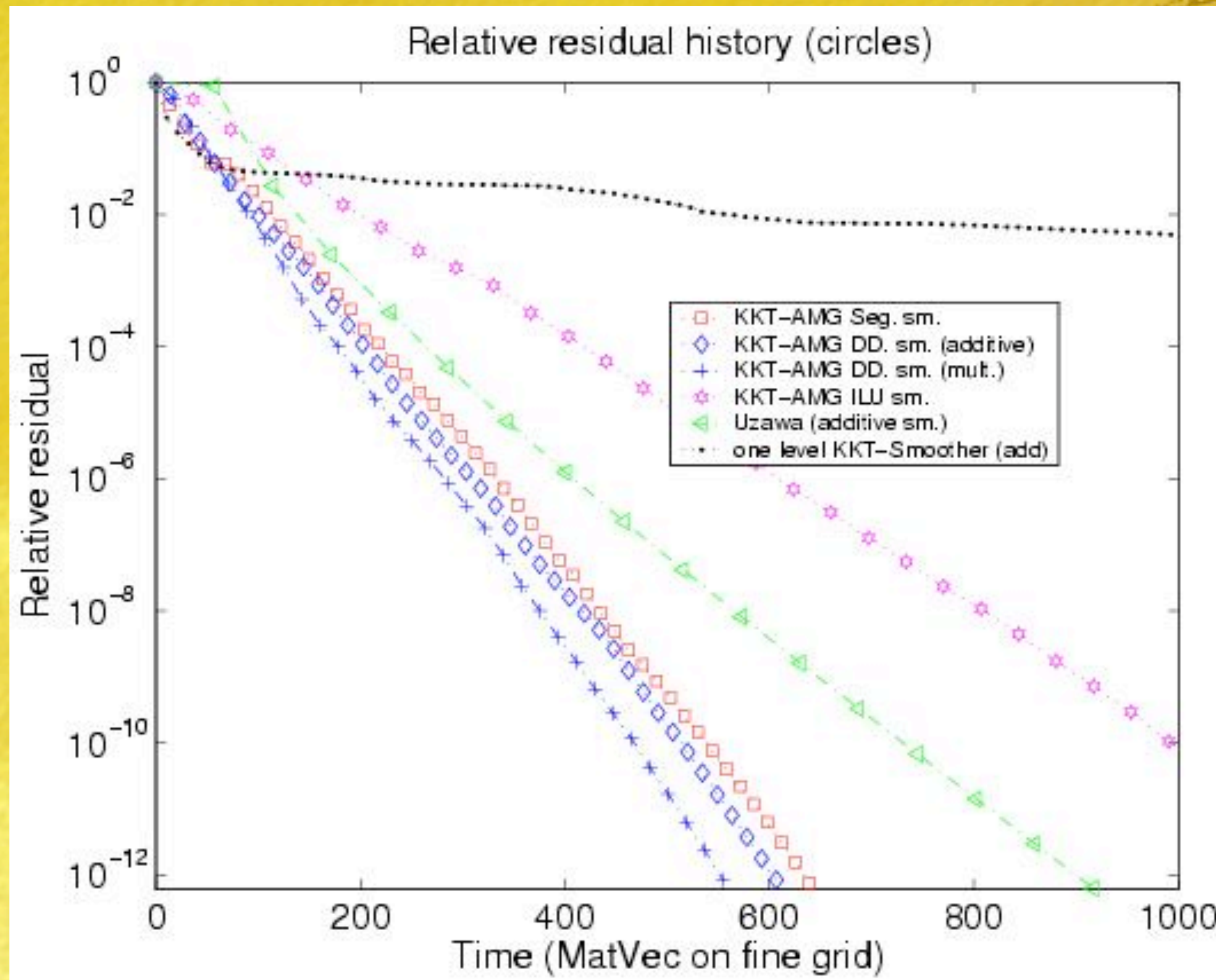
Adagio "Circles" Problem

- ~7200 dof
- 2 contact surfaces
- 2 layers of 1st order hexehedra displacement FE
- 10^2 ratio in elastic modulus
- Contact w/ friction
  - Results in tied mesh
  - 480 constraint eqs.



Adagio "Circles" Problem

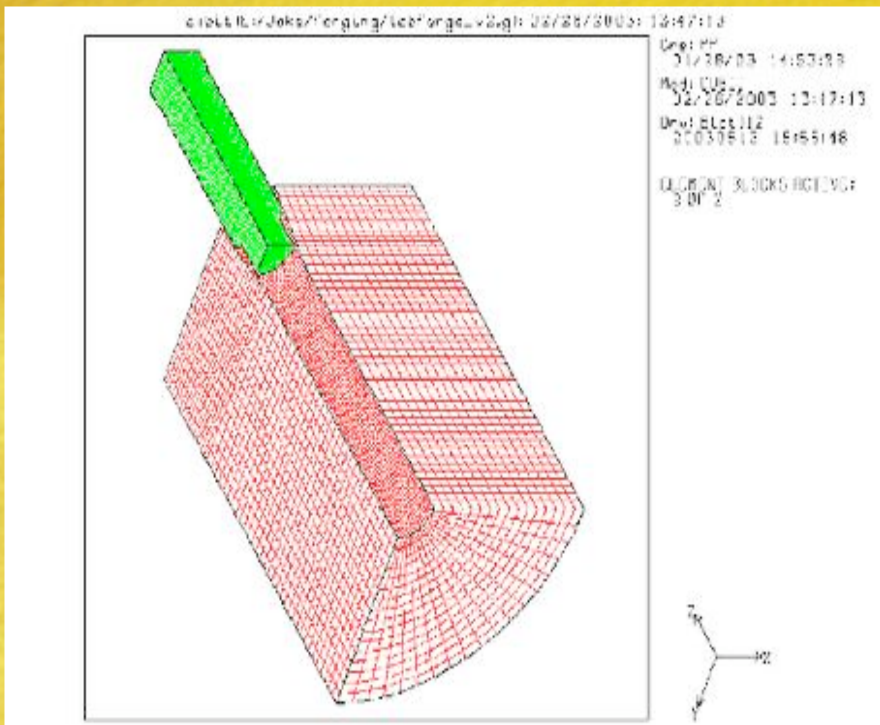




Independence of smoothers (processor) domain size

Iteration counts	# domains (processors)			
Smoothers	1	2	4	8
Segregated	51	57	52	47
CCS (additive)	47	43	47	42
CCS (multiplicative)	35	34	37	31
ILU	30	33	35	32
Uzawa	100	122	127	106

Adagio forging problem



- ~128K dof
- 64 Lagrange multipliers
- 5 “time” steps
- 1st contact in 3rd step
- 16 processors
  - SGI 2000

Solve Times (sec)

Smoothers	end-to-end	setup	solve (# of solves)
Segregated	1060	259	293 (309)
CCS (add)	1400	255	654 (313)
CCS(mult)	910	257	182 (306)
ILU	2096	607	1025 (304)
Uzawa	1057	238	355 (304)

Mesh independence

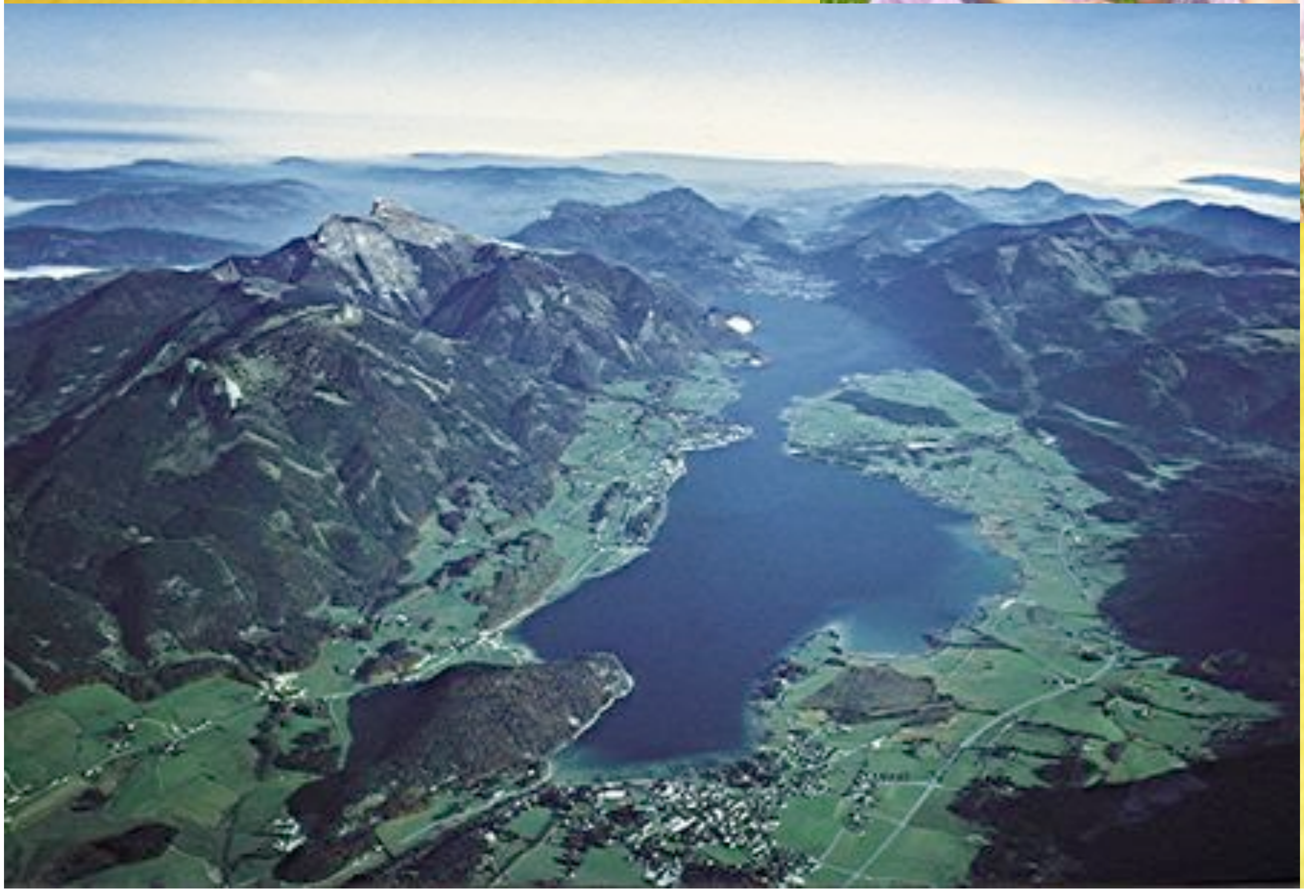
	Iterations (1st solve)				Dof (approx.)	
Levels	Uzawa	CCS (add.)	CCS (mult.)	Segre-gated	primal	L.M
2	29	7	4	8	8016	17
3	29	8	5	9	284	4
4	28	10	4	8	54	2

Thank You

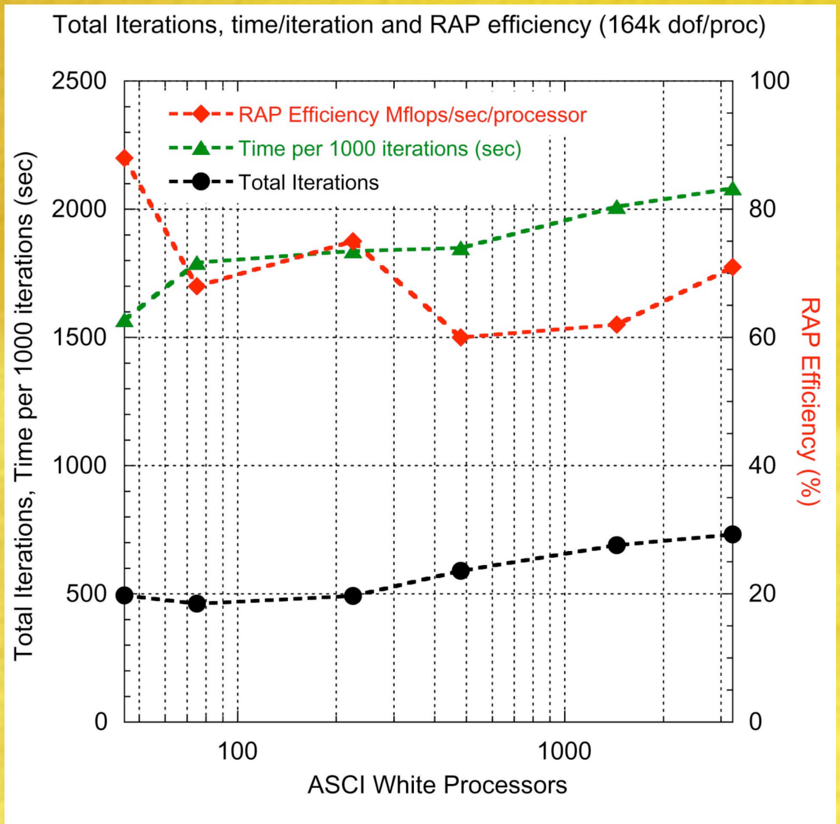
Ultrascalable implicit finite element analyses in solid mechanics with over a half a billion degrees of freedom

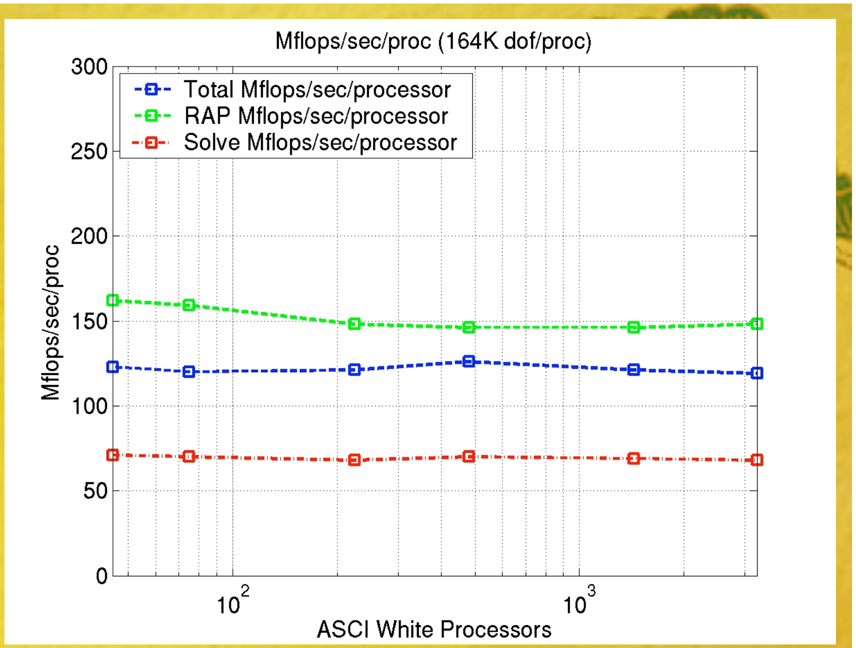
M.F. Adams, H.H. Bayraktar,T.M. Keaveny, P. Papadopoulos

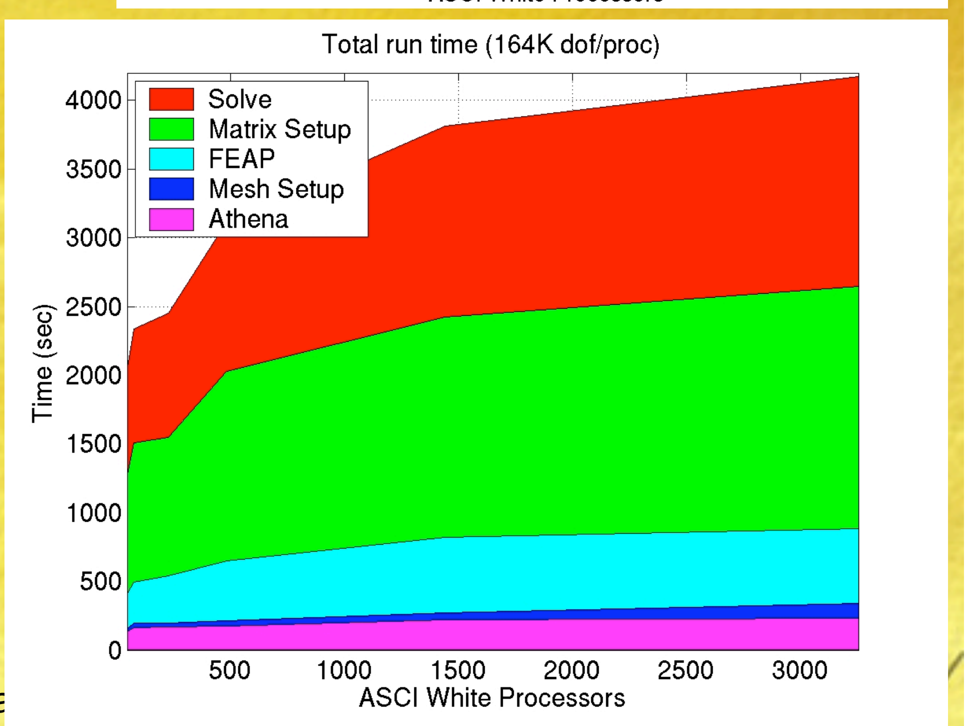
ACM/IEEE Proceedings of SC2004: High Performance Networking and Computing



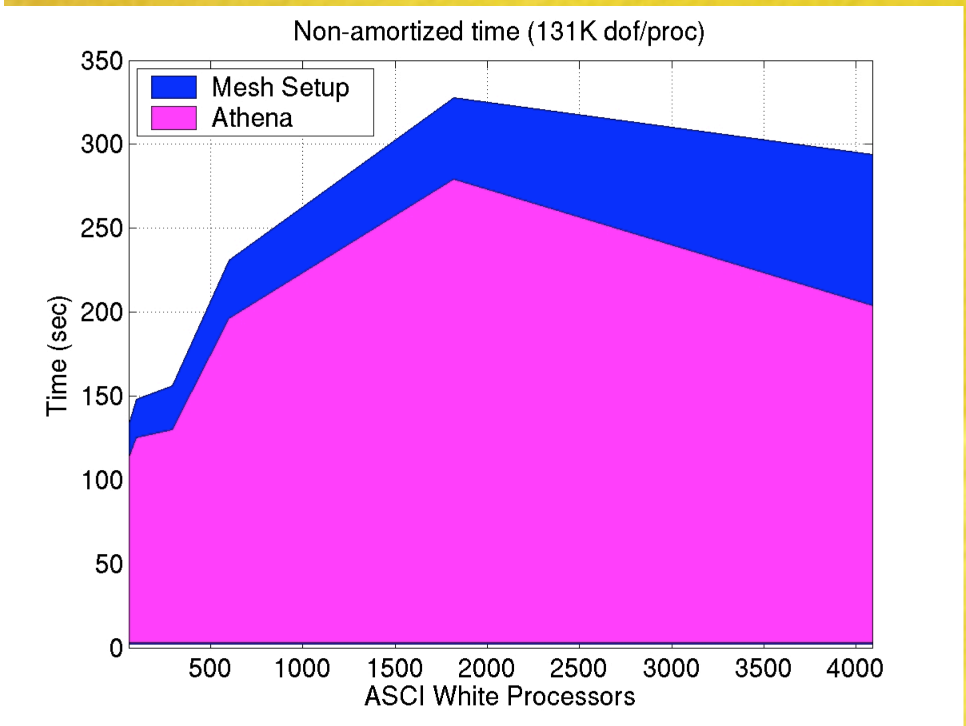
164K dof/proc

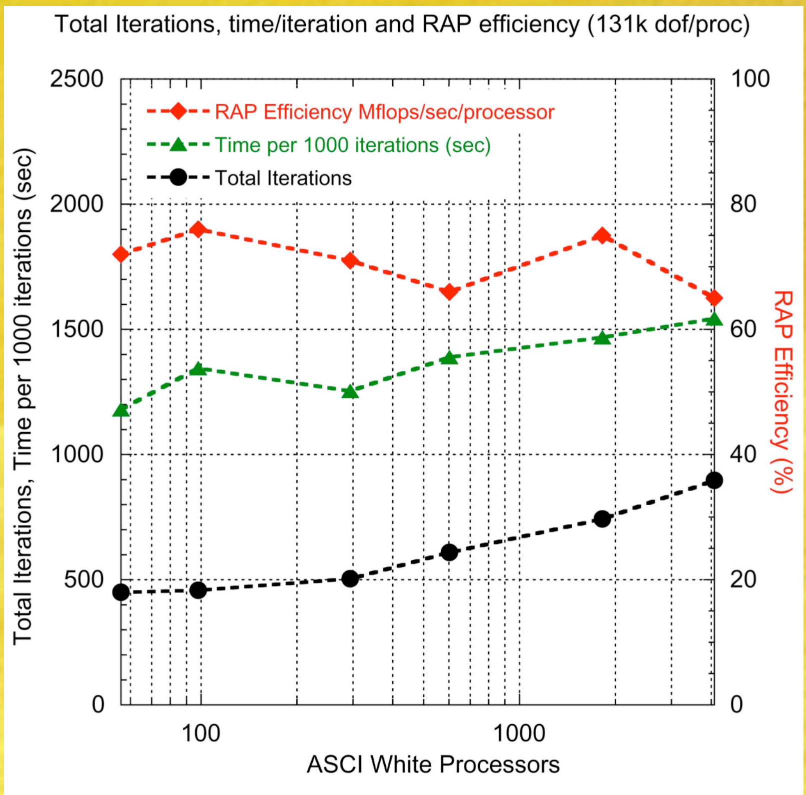




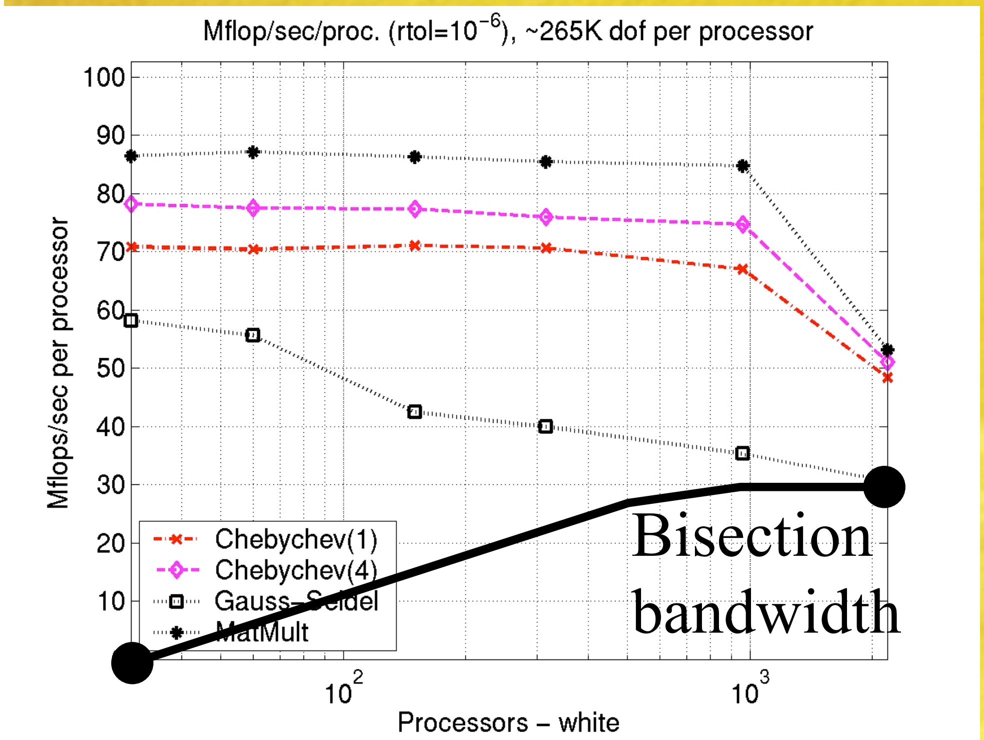


End to end times and (in)efficiency components





First try: Flop rates (265K dof/processor)



265K dof per proc.

IBM switch bug

- Bisection bandwidth plateau 64-128 nodes

Solution:

- use more processors
- Less dof per proc.
- Less pressure on switch

Multigrid V(ν_1,ν_2) - cy cle

- Given smoother S and coarse grid space (P)
  - Columns of "prolongation" operator P, discrete rep. of coarse grid space
- Function u = **MG-V**(A,f)
  - if A is small
    - $u \leftarrow A^{-1}f$
  - else
    - $u \leftarrow S^{\nu 1}(f, u)$ -- $\nu 1$ steps of smoother (pre)
    - $r_H \leftarrow P^T(f – Au)$
    - $u_H \leftarrow$ **MG-V**(P^TAP, r_H) -- recursion (Galerkin)
    - $u \leftarrow u + Pu_H$
    - $u \leftarrow S^{\nu 2}(f, u)$ -- $\nu 2$ steps of smoother (post)
- Iteration matrix: $T = S (I - P(RAP)^{-1}RA) S$
  - *multiplicative*

Polynomial Smoothers - Additive preconditioners

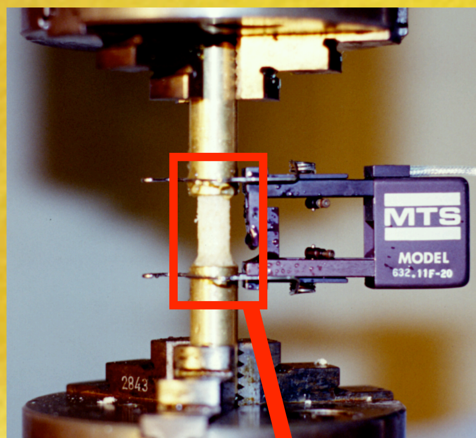
- Additive preconditioners (M) parallelize well
  - Polynomial methods are additive
  - $x^{(m+1)} = x^{(m)} + p(MA)\,(Mb - MA\, x^{(m)})$
- Chebychev is ideal for multigrid smoothers
- Chebychev chooses *p(y)* such that
  - $|1 - p(y)\, y| = min$ over interval $[\lambda^*, \lambda_{max}]$
- No need for lowest eigenvalue
- Estimate of λ_{max} is straight forward
  - Use $\lambda^* = \lambda_{max} / C$
    - C related to rate of grid coarsening

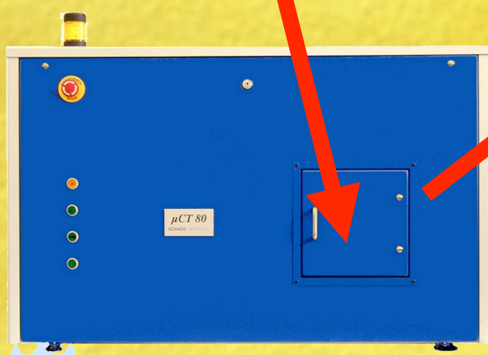
2D Laplacian

Order	Smoother	Iterations
1	lex. Gauss-Seidel	28
2	lex. Gauss-Seidel	16
3	lex. Gauss-Seidel	13
1	red-black Gauss-Seidel	20
2	red-black Gauss-Seidel	11
3	red-black Gauss-Seidel	10
1	damped Jacobi/Cheb.	53
2	damped Jacobi	27
2	Chebyshev	19
3	Chebyshev	13

Methods: μFE modeling

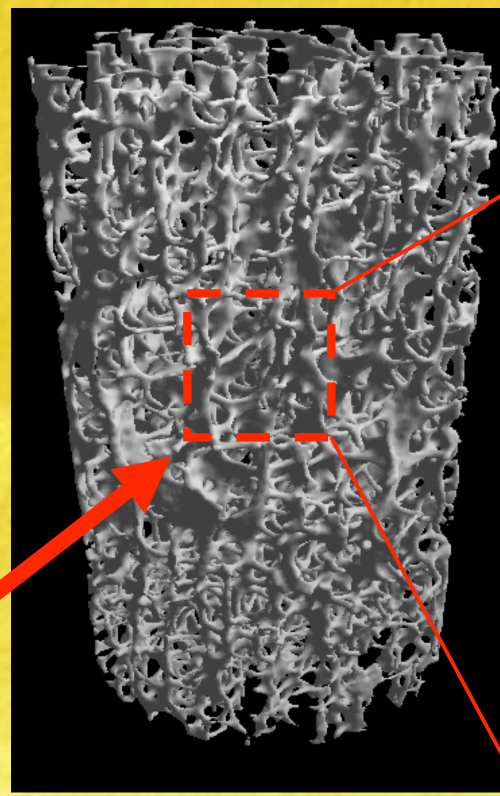
Mechanical Testing
E, ε_{yield}, σ_{ult}, etc.



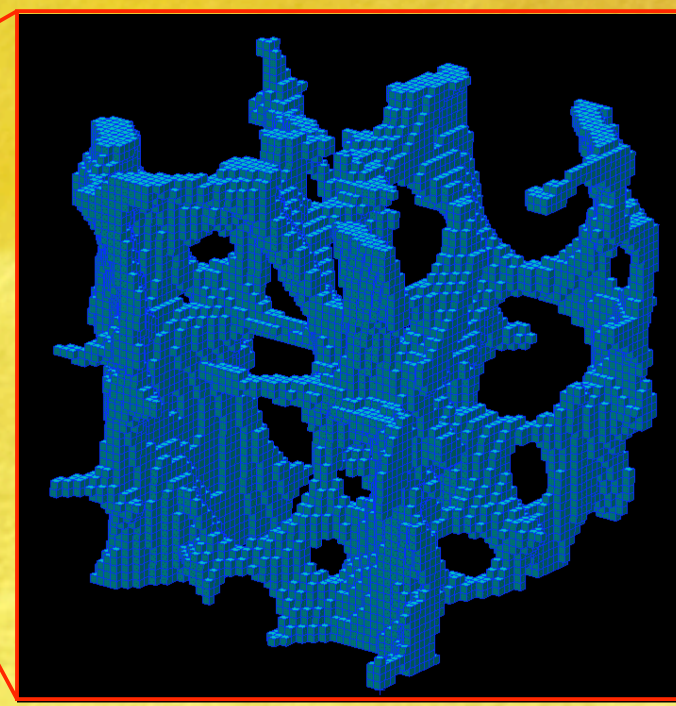


Micro-Computed Tomography
μCT @ 22 μm resolution

3D image



μFE mesh

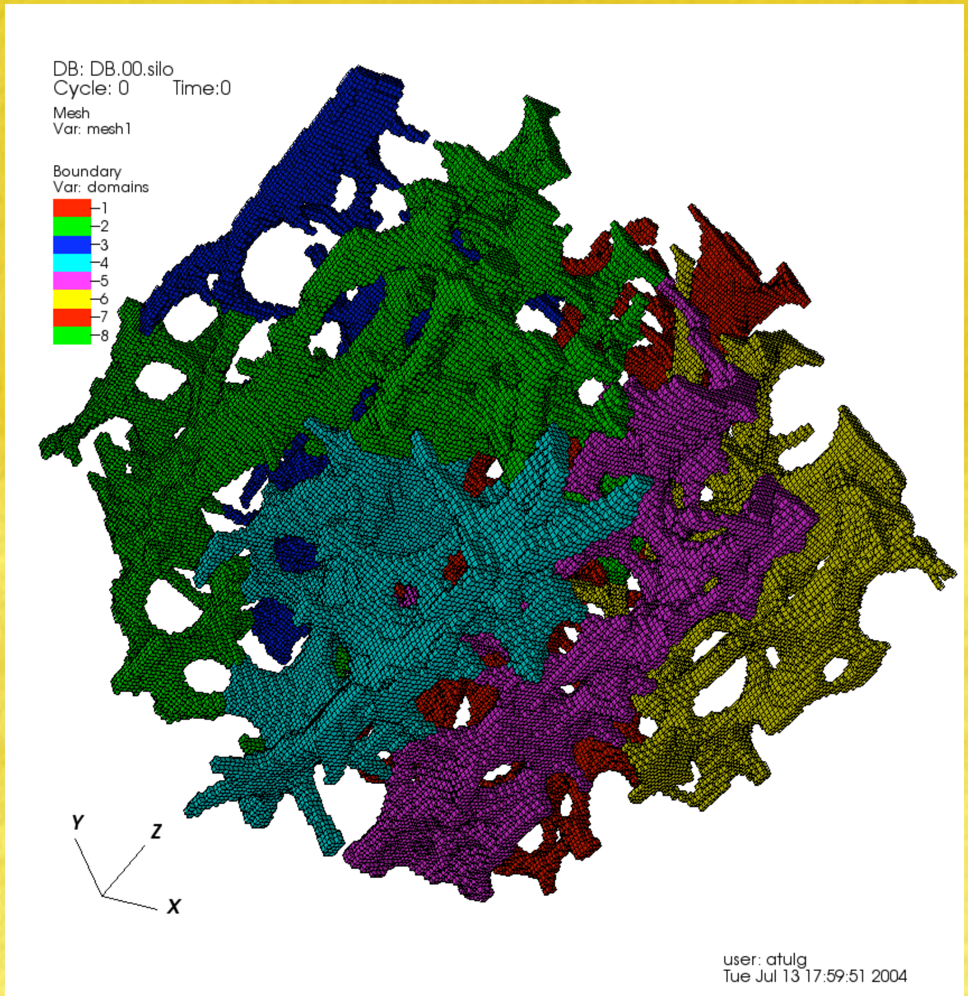


2.5 mm cube
44 μm elements

Motivation

- Calibrate material models for continuum elements
  - eg, explicit computation of a yield surface)
- Highly accurate modeling capabilities for low order model validation
- Investigation of effects that are not accessible with lower order models
  - role of cortical shell in load carrying of vertebra
  - effects of drug treatment on continuum properties

ParMetis partitions

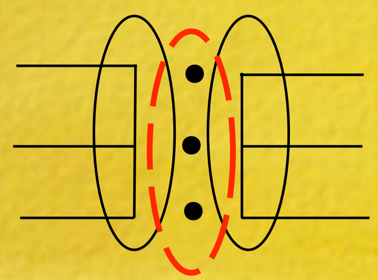


Common Solution Methods for KKT systems

- Uzawa (augmented Lagrange)
  - Richardson iteration on Schur complement
  - $(\mathbf{C}\,\mathbf{K}^{-1}\mathbf{C}^T)\,\lambda = \mathbf{C}\mathbf{K}^{-1}f - g$
- Constraint (Schur) reduction
  - Static condensation
    - Eliminate a 'slave' variable in each constraint
- Projection methods:
  - Krylov method with $\mathbf{P}^T\mathbf{K}\mathbf{P}\,u' = f_p$
    - $\mathbf{P} = \mathbf{I}\text{-}\mathbf{C}^T(\mathbf{C}\mathbf{Q}\mathbf{C}^T)^{-1}\mathbf{C},\ f_p = \mathbf{P}(f - \mathbf{K}u_0),\ u_0 = \mathbf{C}^T(\mathbf{C}\mathbf{Q}\mathbf{C}^T)^{-1}g,\ u = u_0 + u'$
    - Precondition with *anything* (ie, constraint oblivious)

Motivation for $Y = PP^T$

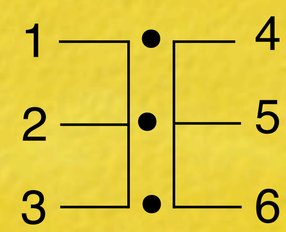
Aggressive primal coarsening



$$P = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix}$$

$$T = CPP^TC^T = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

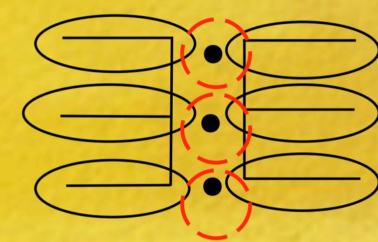
2D Model Problem



$$C = \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{pmatrix}$$

$$\mathbf{CC}^T = \mathbf{I}$$

Non-aggressive primal coarsening



$$P = I_{6X6}$$

$$T = CPP^TC^T = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$