Sparse Days
June 5-6th, 2014
CERFACS, Toulouse, France

Towards a Hybrid AMG-HSS Linear Solver

Osni Marques
Lawrence Berkeley National Laboratory
OAMarques@lbl.gov

Joint work with Sherry Li, Alexander Druinsky, Brian Austin, Eric Roman (LBNL); Panayot Vassilevski, Andrew Barker and Umberto Villa (LLNL)

Challenges for Next Generation Solvers (100PF – and beyond)

- Extreme levels of concurrency
  - millions of nodes with thousands of lightweight cores
  - hundreds of thousands of nodes with more aggressive cores
- Resilience and non-deterministic behavior
  - hard interrupts (failure of a device)
  - soft errors (change of a data value due to faults in logic latches)
- Reduced memory sizes per core
  - more computation on local data, minimization of synchronization
  - shift the focus from the usual weak scaling to strong scaling
- Data storage and movement
  - on a node, data movement will be much more costly, than other operations
  - data access will be much more sensitive to data layout
- Deep memory hierarchies
  - solvers may need to be hierarchical (e.g. cache-oblivious)
- Portability with performance
  - current programming possibilities are not interoperable
  - abstractions

An Optimal Order Resilient Solver (for PDEs)

- High order elements to discretize the PDEs and algebraic multigrid (AMG) method
  - element-agglomeration AMG (or AMGe)
  - assume access to the local element matrices at the finest mesh
  - global (assembled) coarse-level matrices and local coarse-level element matrices
- Reduced-rank sparse factorizations
  - Hierarchically Semi-Separable (HSS) matrices
  - HSS-sparse component
  - enhance scalability via randomization
- *Resilience*
  - *Algorithmic-based features*
  - *Containment Domains*

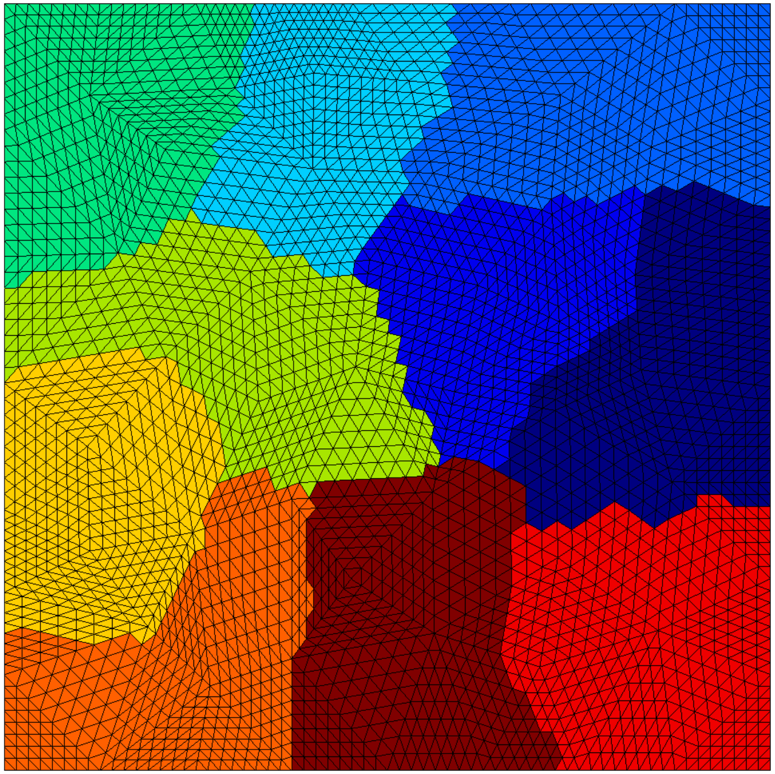
Two-grid Adaptive Smoothed Aggregation Spectral AMG Method

(Marian Brezina, Panayot Vassilevski, Delyan Kalchev)

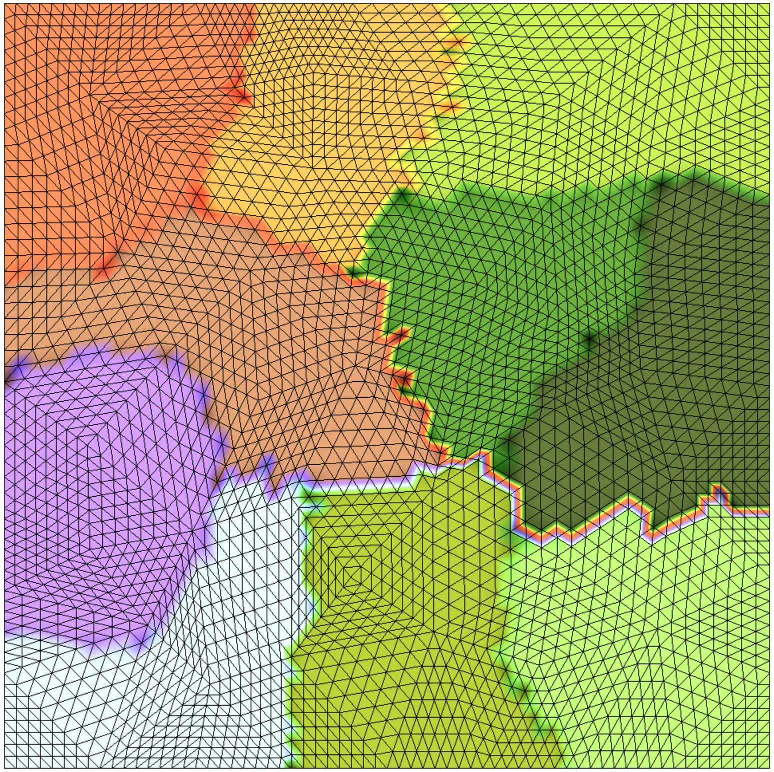
- Pre-smoothing
  - intermediate iterate: $y = x_i + M^{-1}(b - Ax_i)$
- Coarse grid correction
  - restrict the residual: $r_c = P^T(b - Ay)$
  - coarse grid equation: $A_c\, x_c = r_c$
  - update fine grid iterate: $z = y + Px_c$
- Post-smoothing
  - next two grid iterate: $x_{i+1} = z + M^{-T}(b - Az)$

- *Large jumps in the PDE coefficients*
- *Elliptic stochastic PDEs (typically solved by a Monte Carlo method)*

Agglomeration and Aggregation



elements → agglomerates



vertices → aggregates

Interpolation and Preconditioner

- Tentative interpolation operator

$$\overline{P} = \begin{bmatrix} \overline{P}_1 & \cdots & \\ & \overline{P}_2 & \\ \vdots & \ddots & \vdots \\ & \cdots & \overline{P}_{na} \end{bmatrix}$$

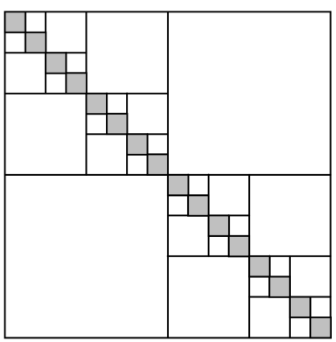
- $\overline{P}_i$ is obtained by means of the eigenvalue problem $A_i q = \lambda D_i q$ (for each agglomerate A_i with diagonal D_i)
- Final interpolation operator: $P = S\overline{P}$, where S (smoother) is a matrix polynomial (Chebyshev-based, sparse)
- M is a polynomial smoother (Chebyshev-based, sparse)
- *Brezina and Vassilevski, Smoothed Aggregation Spectral Element Agglomeration AMG: SA-ρAMGe,* LNCS 7116, *2012.*

HSS-embedded Low-rank Sparse Solver and Preconditioners

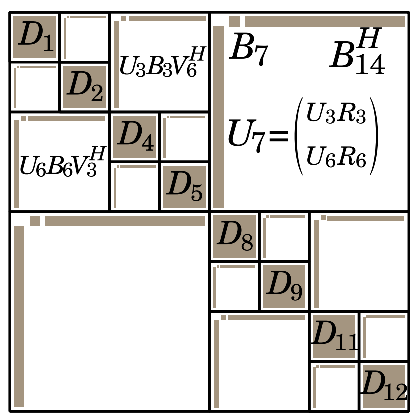
(Sherry Li, Artem Napov, Francois-Henry Rouet, Jianlin Xia)

- Hierarchically Semi-Separable structure for dense, but data-sparse structured matrices
  - Examples: BEMs, Integral equations, PDEs with smooth kernels
  - Matrix off-diagonal blocks are rank deficient
  - Recursion leads to hierarchical partitioning, nested bases lead to nearly linear complexity

hierarchical partition



SVD with nested bases

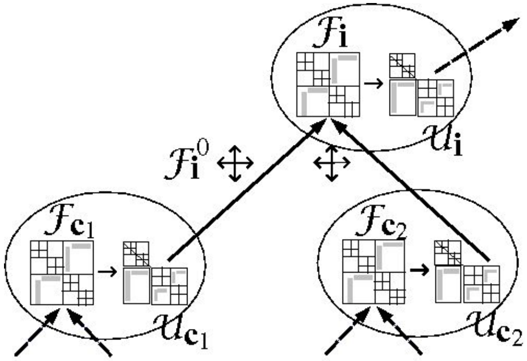


Matrix block representation

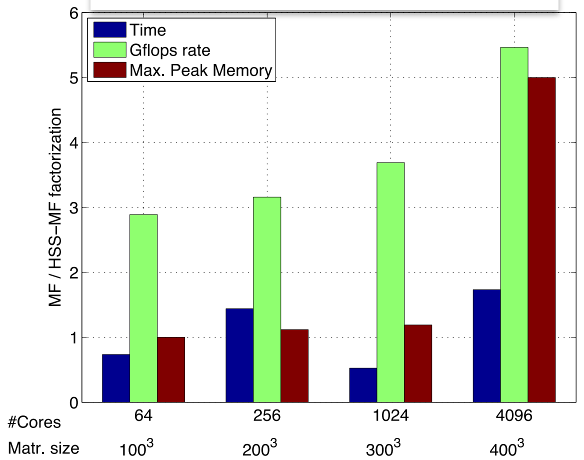
$$A = \left[\begin{array}{cc|cc} D_1 & U_1B_1V_2^T & \multicolumn{2}{c}{\multirow{2}{*}{$U_3B_3V_6^T$}} \\ U_2B_2V_1^T & D_2 & & \\ \hline \multicolumn{2}{c|}{\multirow{2}{*}{$U_6B_6V_3^T$}} & D_4 & U_4B_4V_5^T \\ & & U_5B_5V_4^T & D_5 \end{array}\right]$$

HSS sparse solver and preconditioners (Hsolver)

- For sparse matrices: apply HSS to dense submatrices
  - Exploit nested tree parallelism: outer separator tree from nested dissection partitioning, inner HSS tree
  - Apply HSS to separators of multifrontal method
  - Randomized sampling as new compression kernel



Performance ratio of Traditional over New



- Parallel Performance: compare to traditional algorithm
  - 3D seismic imaging: Helmholtz equations up to 600^3 cubic grids (216M equations)
  - 16,000+ cores, 2x faster, uses 1/5 of memory

S. Wang, X.S. Li, F.-H. Rouet, J. Xia, and M. de Hoop, “A Parallel Geometric Multifrontal Solver Using Hierarchically Semiseparable Structure”, ACM TOMS, revised, 2014.
A. Napov, X.S. Li and M. Gu, “An algebraic multifrontal preconditioner that exploits the low-rank property”, Numerical Lin. Alg. & Apps, submitted, 2014.

Hybrid solver: possible usage patterns

Input: discretized PDE with high order elements in parallel format (distributed matrix)

AMGe module

i. Generates intermediate coarse-level reduced dimension (upscaled) problem with guaranteed approximation properties
ii. Provides AMGe hierarchy used to build the 1st component of the hybrid AMGe-HSS solver

HSS solver module

Input: reduced dimension (intermediate coarse-level) problem from the AMGe module
Output: 2nd component of the hybrid AMGe-HSS solver

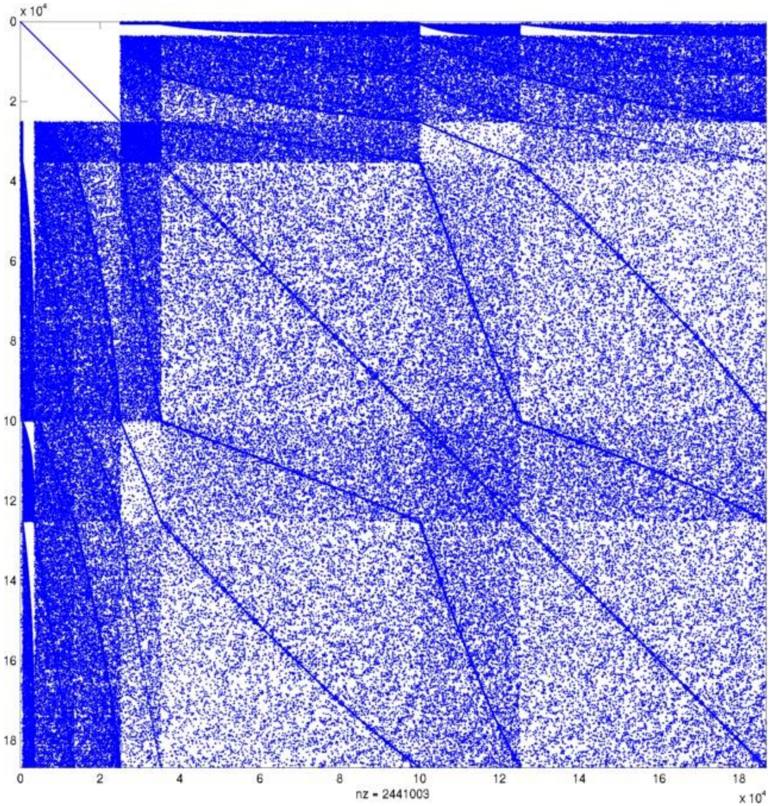
Preconditioned Krylov solver

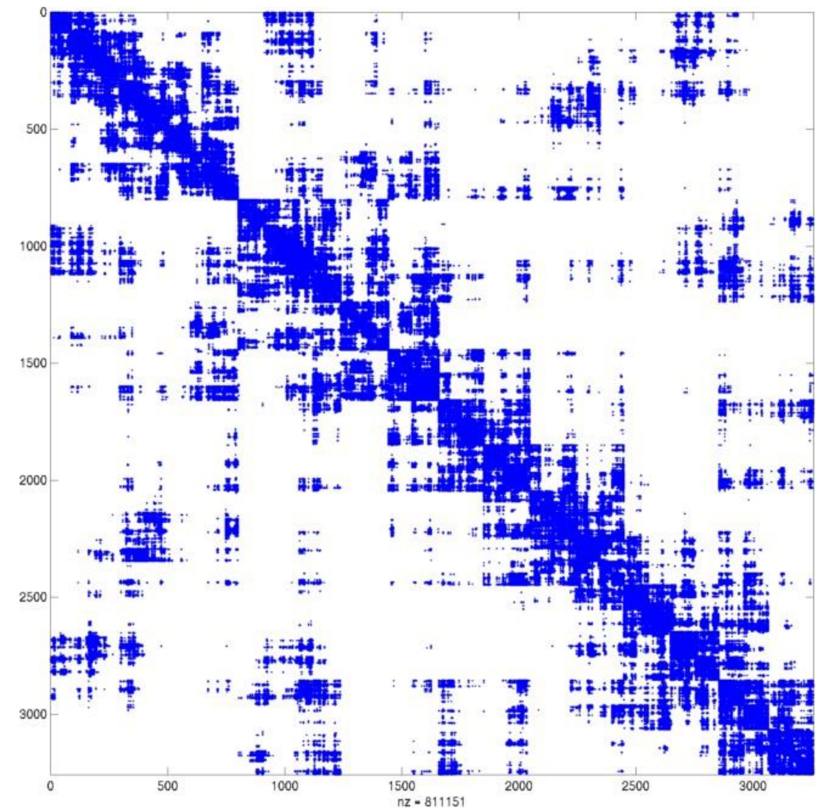
1) hybrid AMGe-HSS (solid lines) ✓
2) standalone AMGe ✓
3) standalone HSS ✓
4) AMGe preconditioned Krylov
5) HSS preconditioned Krylov
6) AMGe-HSS preconditioned Krylov ✓

Computational Settings and Experiments

- SAAMGE Code (D. Kalchev)
  - MFEM
  - LAPACK
  - Metis
  - default: 400 elements per agglomerate; tol = 10^{-10}; max iter = 100 …
- HSS Code (A. Napov)
  - Scotch
  - Metis (for nodes ordering within a separator)
- Cases 1-3
  - AMG as a preconditioner for CG
  - Coarse grid solver: PCG/Gauss-Seidel or FGMRES/StruMF
  - Intel Xeon @ 2.53GHz (Westmere)
- Case 4:
  - Intel Xeon @ 2.40 GHz (Ivy Bridge)
- MKL BLAS

Case 1: cube

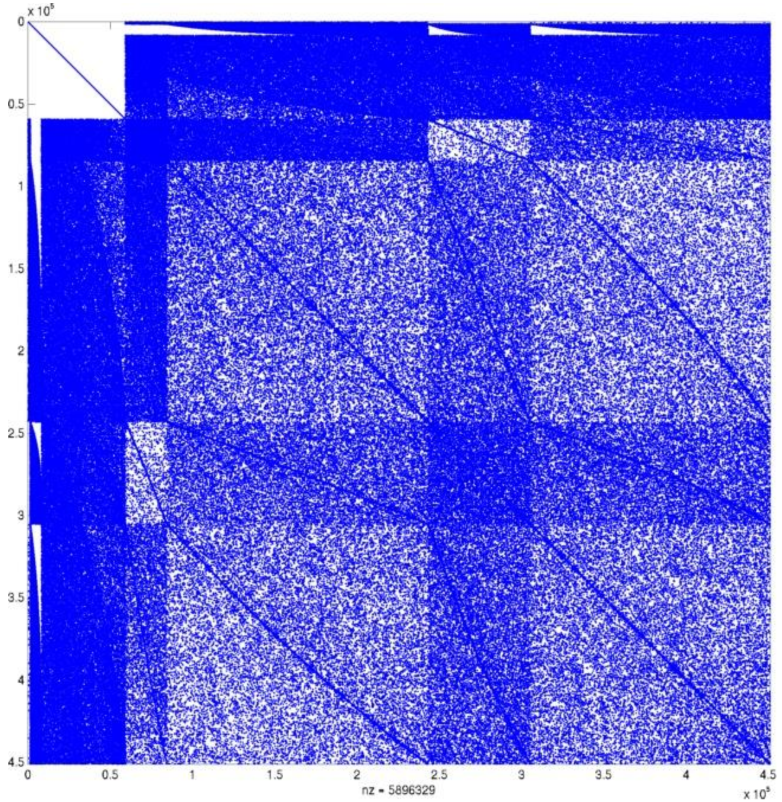


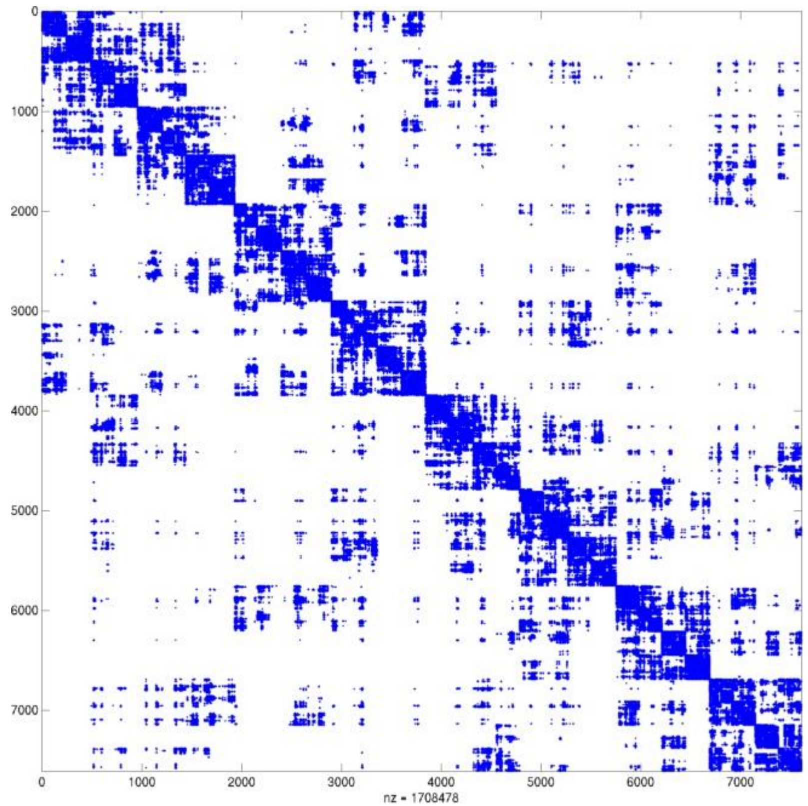


fine grid
n = 186677
nnz = 2441003
nnz/n ≈ 13

coarse grid
n = 3257 (~57x)
nnz = 811151
nnz/n ≈ 249

Case 2: cube and cylinder

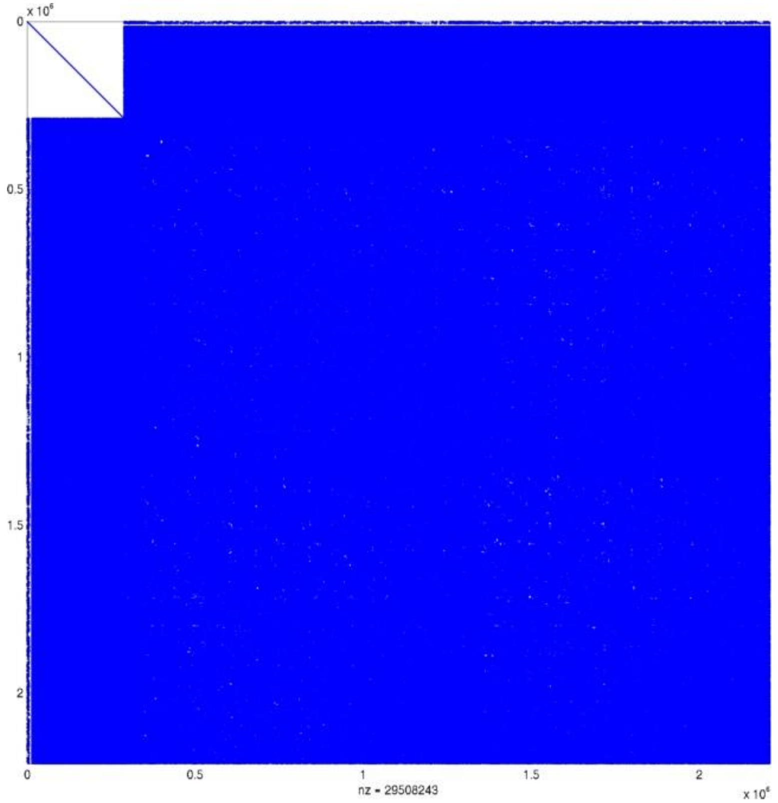


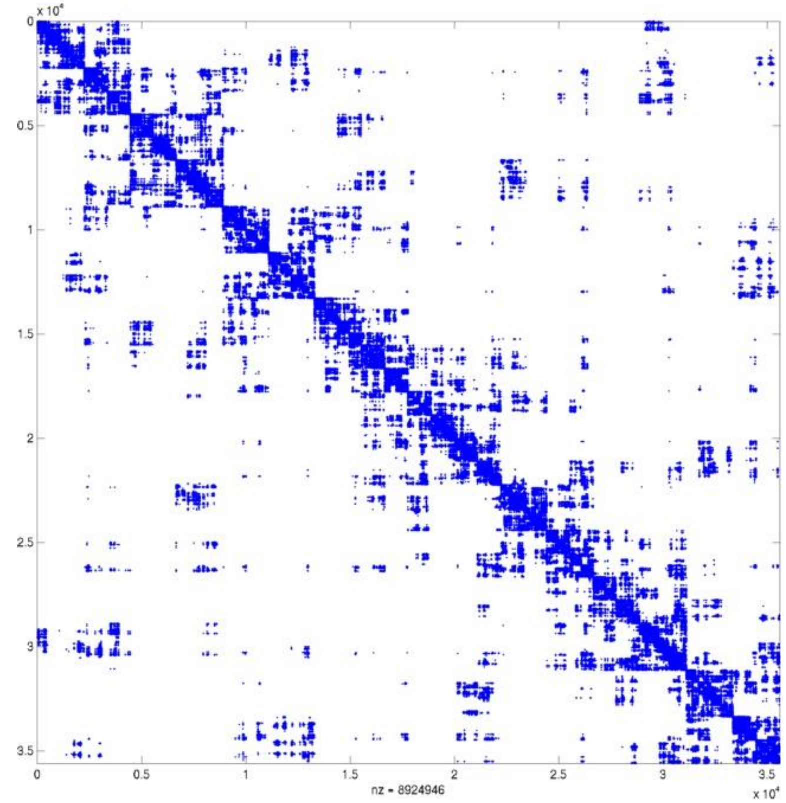


fine grid
n = 451053
nnz = 5896329
nnz/n ≈ 13

coarse grid
n = 7606 (~59x)
nnz = 1708478
nnz/n ≈ 224

Case 3: cube and cylinder





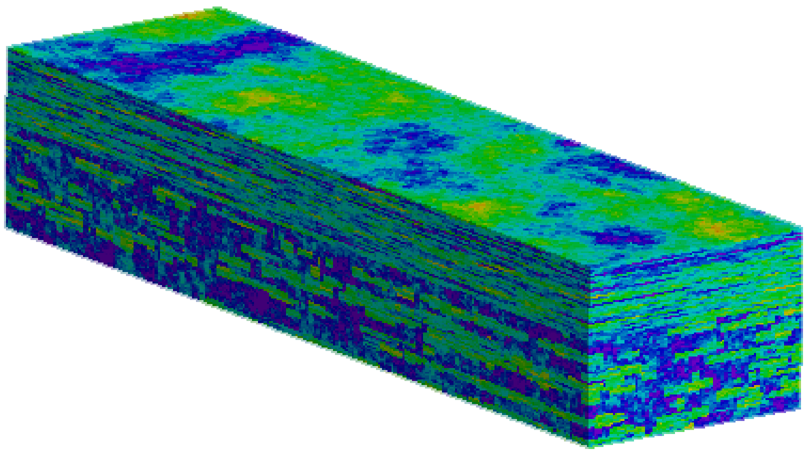
fine grid
n = 2209165
nnz = 29508243
nnz/n ≈ 13

coarse grid
n = 35590 (~62x)
nnz = 8924946
nnz/n ≈ 250

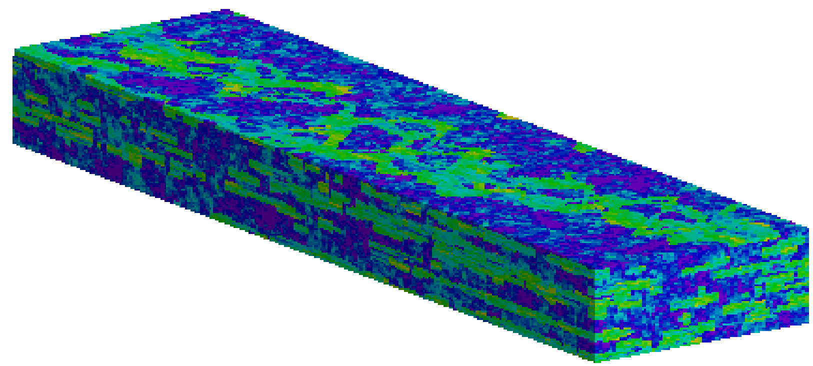
Case 4: SPE10 Benchmark (model 2)

http://www.spe.org/web/csp/datasets/set02.htm

- Model of a formation in the Brent oil field
- 1200 x 2200 x 170 ft
- The top 70 ft (35 layers) represents the Tarbert formation; the bottom 100 ft (50 layers) represents Upper Ness (fluvial).
- The fine scale cell size is 20 ft x 10 ft x 2 ft.



porosity of the whole model

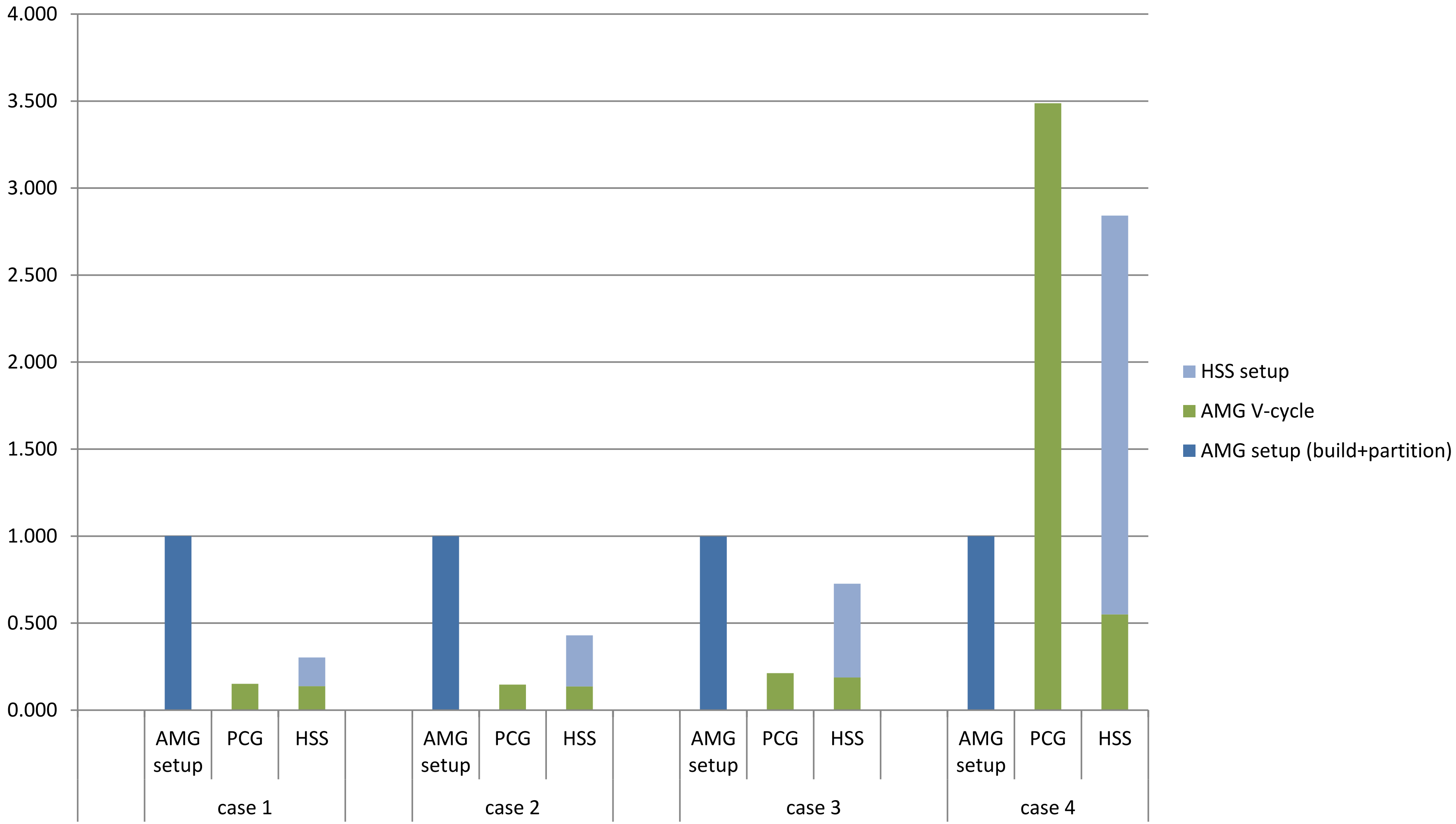


part of the Upper Ness sequence

fine grid: n = 1159366; nnz = 30628096; nnz/n ≈ 26

coarse grid: n = 60043 (~19x); nnz = 79825303; nnz/n ≈ 1329

Results (normalized times)



Cases 1-3: AMG-PCG/HSS preconditioned Krylov

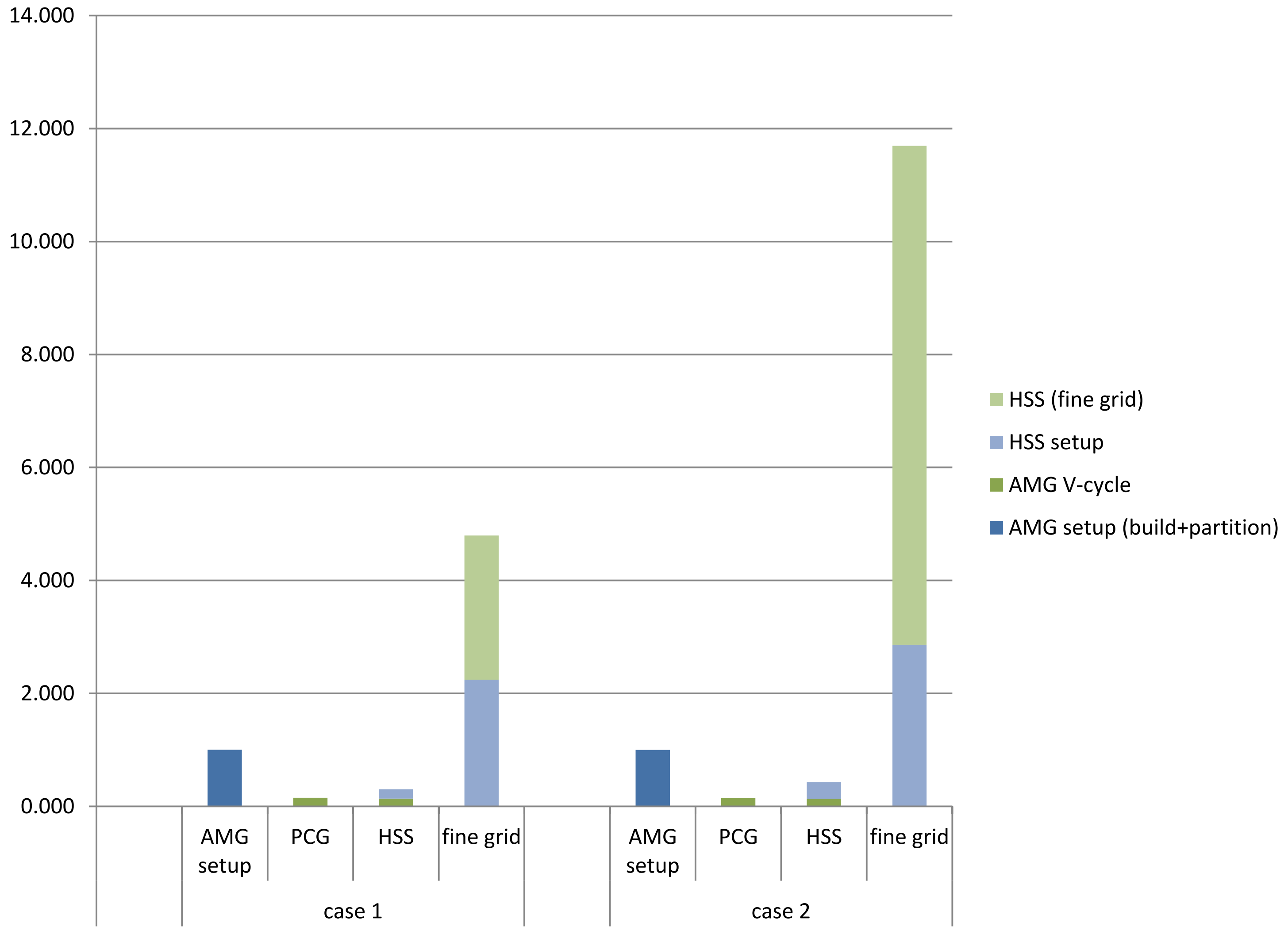
SPE10: SuperLU_DIST and StruMF

StruMF: Artem and Li, An algebraic multifrontal preconditioner that exploits the low-rank property, NLAA, submitted.

fine grid: n = 1159366; nnz = 30628096; nnz/n ≈ 26
coarse grid: n = 60043; nnz = 79825303; nnz/n ≈ 1329

			seconds					
SuperLU_DIST (1 core)	**nnz(L+U)**	**fill-ratio**	**Num. factor**	**Solve**	**Metis**	**Symbolic**		
fine	**2.35E+09**	**76.7**	**1547.2**	**4.8**	**19.5**	**6.8**		
coarse	**5.44E+08**	**6.8**	**313.4**	**0.8**	**1.9**	**1.5**		
StruMF (coarse, 1 core)	**nnz(ULV)**	**fill-ratio**	**Num. factor**	**Solve**	**Partition**	**Symbolic**	**Max rank**	**Avg rank**
multifrontal	**5.71E+08**	**7.2**	**162.6**	**0.8**	**97.6**	**3.5**		
non-sym	**3.46E+08**	**4.3**	**866.5**	**3.6**	**97.6**	**3.4**	**962**	**106**
symmetric	**3.95E+08**	**4.9**	**759.3**	**3.5**	**97.6**	**3.5**	**963**	**143**

Results (normalized times, standalone HSS)



Final Remarks

- For well conditioned coarse grids direct methods do not seem to be competitive
- Current HSS implementation does not handle very coarse grids well, e.g. as in SPE10 (separator is reordered to reduce the rank)
- Codes are still sequential, parallel version (in the works) is expected to give better insight about HSS strategy on larger/harder problems (large jumps in the PDE coefficients)