AMES: A recursive multilevel approximate inverse-based preconditioner for solving general linear systems

Yiming Bu, Bruno Carpentieri and Tingzhu Huang

University of Groningen
Johann Bernoulli Institute for Mathematics and Computer Science

Eindhoven, The Netherlands
June 17, 2015

Outline

1. Motivation
2. The Algebraic Mulitilevel Explicit Solver (AMES)
3. Experiments: comparison against state-of-the-art solvers
4. Concluding remarks

Outline

1. Motivation
2. The Algebraic Mulitilevel Explicit Solver (AMES)
3. Experiments: comparison against state-of-the-art solvers
4. Concluding remarks

Problem

- Solving systems of linear equations of the form

$$Ax = b$$

where $A \in \mathbb{R}^{n \times n}$ is large, sparse and nonsymmetric,
$x \in \mathbb{R}^n$ is the unknown vector and $b \in \mathbb{R}^n$ the right hand side.

Goal: an ideal solver

- numerically stable, economical to construct, cheap to apply.
- good degree of inherent parallelism.

Some state-of-the-art solvers

- **The algebraic recursive multilevel solver ARMS**:
  Schur-complement based multilevel ILU factorization.

  Y. Saad and B. Suchomel,
  ARMS : An algebraic recursive multilevel solver for general sparse linear systems. NLAA, 9:359-378 (2002).

- **The sparse approximate inverse preconditioner SPAI**:
  Sparse approximate inverse preconditioner.

  M. Grote and T. Huckle,
  Parallel preconditionings with sparse approximate inverses. SIAM J. Scientific Computing 18 838-853 (1997).

- **Multilevel ILU factorization algorithms in ILUPACK**:
  Multilevel preconditioners constructed from inverse-based factorizations.

  M. Bollhoefer and Y. Saad,
  ILUPACK - preconditioning software package.

Outline

The Algebraic Mulitilevel Explicit Solver (AMES)

The multilevel framework:

1. Scaling phase

$$D_1^{1/2} A D_2^{1/2} y = D_1^{1/2} b, \quad x = D_1^{1/2} y,$$

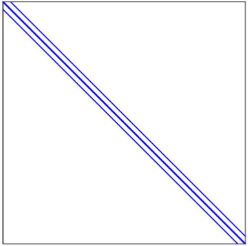
where

$$D_1(i,j) = \begin{cases} \dfrac{1}{\max\limits_i |a_{ij}|} & , \text{ if } i = j \\ 0 & , \text{ if } i \neq j \end{cases}, \quad D_2(i,j) = \begin{cases} \dfrac{1}{\max\limits_j |a_{ij}|} & , \text{ if } i = j \\ 0 & , \text{ if } i \neq j \end{cases}.$$

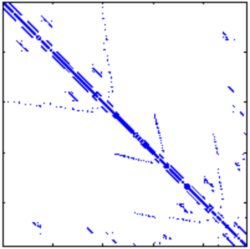
The AMES method

2. Preordering phase

$$\tilde{A} = P^T A P = \begin{pmatrix} B_1 & & & & F_1 \\ & B_2 & & & F_2 \\ & & \ddots & & \vdots \\ & & & B_p & F_p \\ E_1 & E_2 & \cdots & E_p & A_S \end{pmatrix}$$



The original structure of *rdb2048*.

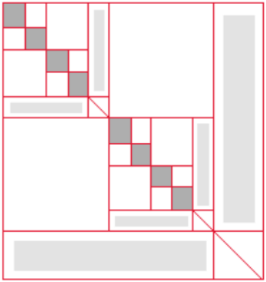


The structure of reordered *rdb2048*.

G. Karypis, V. Kumar. Metis: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices version 4.0.

The AMES method

3. Analysis phase



Tree data structure

- In $\tilde{A}$, the blocks B at the last level, as well as the blocks E and F at each level, are stored in compressed sparse row format in a tree data structure.
- Auxiliary vectors are computed for exploiting the sparsity of the blocks E and F, in preparation to the following factorization phase.

The AMES method

4. Factorization phase

Compute the approximate inverse factors of $\tilde{A}$:

$$\tilde{L}^{-1} \approx \begin{pmatrix} U_1^{-1} & & & & W_1 \\ & U_2^{-1} & & & W_2 \\ & & \ddots & & \vdots \\ & & & U_p^{-1} & W_p \\ & & & & U_S^{-1} \end{pmatrix}, \; \tilde{U}^{-1} \approx \begin{pmatrix} L_1^{-1} & & & & \\ & L_2^{-1} & & & \\ & & \ddots & & \\ & & & L_p^{-1} & \\ G_1 & G_2 & \cdots & G_p & L_S^{-1} \end{pmatrix},$$

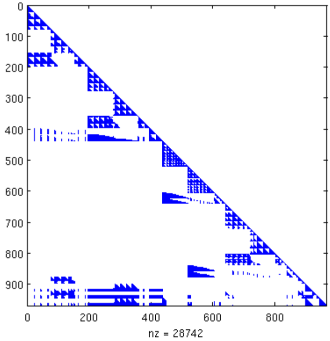
where $B_i^{-1} = U_i^{-1} L_i^{-1}$, $W_i = -U_i^{-1} L_i^{-1} F_i U_S^{-1}$, $G_i = -L_S^{-1} E_i U_i^{-1} L_i^{-1}$, and L_S, U_S are the triangular factors of the Schur complement matrix

$$S = A_S - \sum_{i=1}^{p} E_i B_i^{-1} F_i.$$

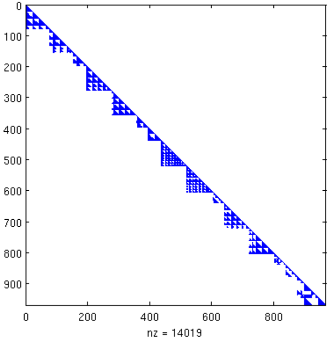
The AMES method

4. Factorization phase

Compute and store the inverse factorizations of
1) the blocks B at last level,
2) the Schur complements at each level.



The inverse factor.



The entries actually stored in memory.

The AMES method

5. Solve phase

Compute $x = Mb$, where

$$M \cdot b \approx A^{-1} \cdot b = \begin{pmatrix} \tilde{B}^{-1} + \tilde{B}^{-1} F \tilde{S}^{-1} E \tilde{B}^{-1} & -\tilde{B}^{-1} F \tilde{S}^{-1} \\ -\tilde{S}^{-1} E \tilde{B}^{-1} & \tilde{S}^{-1} \end{pmatrix} \cdot \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

Algorithm 1 Preconditioning operation

1: $p_1 = \tilde{B}^{-1} b_1$
2: $[p_2, p_3] = \tilde{S}^{-1} [E \cdot p_1, b_2]$
3: $[p_4, p_5] = \tilde{B}^{-1} [F \cdot p_2, F \cdot p_3]$
4: $x_1 = p_1 + p_4 - p_5$
5: $x_2 = p_3 - p_2$

Outline

Numerical tests

Matrix problem	n	Field	$nnz(A)$
cz20468	20,468	Closest Point Method	206,076
raefsky3	21,200	Fluid Structure Interaction	1,488,768
ABACUS_shell_ud	23,412	ABAQUS benchmark	218,484
sme3Db	29,067	3D structural mechanics problem	2,081,063
viscoplastic2	32,769	FEM discretization	381,326
cz40948	40,948	Closest Point Method	412,148
rma10	46,835	3D CFD Model	2,374,001

Table : Set and characteristics of the test matrix problems.

solver	nrestart	tol	max it
GMRES	500	1.0e-12	5000

Table : Parameters for the tests.

Numerical tests against other state-of-the-art solvers

Matrix	Method	$\frac{nnz(M)}{nnz(A)}$	Its	t_p	t_f	t_s
cz20468	AMES_ILU	1.26	187	0.3	0.2	4.2
	ILUPACK	1.24	2500	-	0.4	40.3
	ARMS	1.16	+5000	-	0.1	+6.5
	SPAI	1.64	+5000	-	4.0	+8.0
raefsky3	AMES_ILU	0.54	235	2.4	3.7	10.0
	ILUPACK	0.55	1224	-	2.8	25.2
	ARMS	2.38	+5000	-	2.4	+23.5
	SPAI	1.83	+5000	-	5040	+243
ABACUS shell ud	AMES_ILU	1.79	453	0.3	0.8	22.1
	ILUPACK	1.82	1411	-	0.5	26.6
	ARMS	1.88	+5000	-	0.2	+7.6
	SPAI	2.41	+5000	-	11.0	+12.0
sme3Db	AMES_ILU	0.85	407	3.5	8.4	39.3
	ILUPACK	0.74	1210	-	4.1	41.4
	ARMS	5.61	+5000	-	39.0	+54.9
	SPAI	1.23	+5000	-	3360	+123

Table : Comparison against the unpreconditioned system and the SPAI

** t_p, t_f and t_s are the time costs for the pre-processing phase, factorization phase and solve phase.*

Numerical tests against other state-of-the-art solvers

Matrix	Method	$\frac{nnz(M)}{nnz(A)}$	Its	t_p	t_f	t_s
viscoplastic2	AMES_ILU	3.07	78	0.9	14.3	3.9
	ILUPACK	4.00	1627	-	1.6	70.0
	ARMS	3.02	+5000	-	0.9	+10.9
	SPAI	3.37	+5000	-	244	+24.0
cz40948	AMES_ILU	1.41	170	0.7	0.4	7.4
	ILUPACK	1.48	1627	-	1.0	51.1
	ARMS	1.70	+5000	-	0.9	+21.8
	SPAI	1.64	+5000	-	8.5	+17.2
rma10	AMES_ILU	2.33	164	3.9	13.1	34.5
	ILUPACK	2.27	1242	-	8.6	82.9
	ARMS	14.30	+5000	-	203.9	+111
	SPAI	4.84	+5000	-	11280	+180

Table : Comparison against the unpreconditioned system and the SPAI

** t_p, t_f and t_s are the time costs for the pre-processing phase, factorization phase and solve phase.*

Using direct method as the local solver

Matrix	$\frac{nnz(M_L+M_U)}{nnz(A)}$	Its	t_p	t_f	t_s
cz20468	1.28	2	0.7	0.4	1.5
raefsky3	2.74	1	3.4	11.1	1.3
cz40948	1.87	2	1.2	0.3	0.2
rma10	3.01	1	5.2	11.6	0.8

Table : Experiments on the effect of utilizing direct solver (MA38) within AMES.

Application on the electromagnetics problems

Table : Set and characteristics of test matrix problems.

Matrix problem	Size	nnz(A)	Field
dw2048	2,048	10,114	Square dielectric waveguide
dw8192	8,192	41,746	Square dielectric waveguide
utm3060	3,060	42,211	Uedge Test Matrix
utm5940	5,940	83,842	Uedge Test Matrix

Numerical tests against other state-of-the-art solvers

Table : Performance comparison of the multilevel approximate inverse preconditioner against other iterative solvers.

Matrix	Method	$\frac{nnz(M)}{nnz(A)}$	Its	t_p	t_f	t_s
dw2048	AMES	1.90	17	0.02	0.03	0.05
	ARMS	2.24	418	-	0.01	0.12
	SPAI	3.12	+5000	-	0.21	+1.76
dw8192	AMES	3.31	52	0.08	0.30	0.94
	ARMS	4.26	1063	-	0.05	4.32
	SPAI	6.98	+5000	-	5.78	+35.59
utm3060	AMES	2.79	125	0.15	0.28	0.73
	ARMS	2.93	+5000	-	0.05	+9.54
	SPAI	3.24	+5000	-	6.06	+5.91
utm5940	AMES	3.50	267	0.27	0.84	5.73
	ARMS	6.15	+5000	-	0.35	+27.69
	SPAI	3.86	+5000	-	21.88	+23.22

** t_p, t_f and t_s are the time costs for the pre-processing phase, factorization phase and solve phase.*

Parameter setting

1. The number of the sub-blocks each block B has, p.

p	Its	t_p	t_f	t_s	t_{per_it}	$\frac{sizeB}{sizeAS}$
5	332	0.5	0.9	14.2	0.043	39.2
15	329	0.5	1.0	13.9	0.042	4.3
30	317	0.5	1.1	12.9	0.041	1.1
50	400	0.5	1.6	18.4	0.046	0.4

Table : Different number of children, with the same $\frac{nnz(M)}{nnz(A)} = 1.17$ (cz20468).

Remark: AMES performs better when the size of block B is close to the size of the Schur complement.

* t_{per_it} *is the time cost for each iteration.*

Parameter setting

2. The number of levels of B, n_{levB}.

n_{levB}	Its	t_p	t_f	t_s	t_{per_it}
1	315	1.11	0.48	20.3	0.065
2	278	1.27	0.48	24.5	0.088
3	275	1.33	0.52	47.5	0.17
4	266	1.46	0.53	105.7	0.40

Table : The impact of the number of levels, with the same $\frac{nnz(M)}{nnz(A)} = 1.33$ (cz40948).

Remark: For the problems that t_{per_it} is sensitive to n_{levB}, a smaller number of levels would be a better choice.
As the number of levels increases while with the same memory ratio, the time of first two phases increases a little, and that of the solve phase increases a lot.

Outline

Concluding Remarks

Conclusions:

- We propose a new multilevel solver AMES for preconditioning sparse linear systems.
- AMES is based on a distributed Schur complement formulation, which expoits sparsity and efficiency.
- AMES has numerically stability, robustness and inherent parallelism.
- Experiments shows AMES is competitive wrt state-of-the-art solvers on some large sparse problems.

Future work:

block implementation, complex arithmetic, OpenMP, ...

References

Yiming Bu, Bruno Carpentieri
A recursive multilevel approximate inverse-based preconditioner for solving general linear system. Under revision.