

ELMRES

Elementary Residual Method

Gary Howell and Desi Stephens

Mathematical Sciences
Florida Institute of Technology
Melbourne, FL 32901
e-mail howell@zach.fit.edu

Department of Mathematics
Florida A&M
Tallahassee, FL
e-mail dstephen@zach.fit.edu

ELMRES and GMRES

- Krylov subspace methods
- GMRES is an orthogonal projection
- ELMRES is an oblique projection.
- GMRES corresponds to reduction to a similar Hessenberg matrix by orthogonal similarity transformations.
- ELMRES corresponds to reduction of to a similar Hessenberg matrix by elementary similarity transformations.

Continued Comparison

- The cost for m columns for GMRES is $2m^2n$ flops (modified Gram-Schmidt without reorthogonalization).
- The cost for m columns for ELMRES is $m^2n - n^3/3$ flops.
- Parallel GMRES is fine grain, requiring parallel inner products.
- Parallel ELMRES is medium grain, requiring a triangular solve.

The m th step of GMRES

$w_m = A * v_m$; /* enlarge subspace

For $i = 1 : m$,

$h_{im} = w_m^T * v_i$; /* modified Gram-Schmidt

$w_m = w_m - h_{im} * v_i$;

End;

$h_{m+1,m} = \|w_m\|_2$;

$v_{m+1} = w_m / h_{m+1,m}$ /* normalization step

At the m th step

y_m minimizes $\|\beta e_1 - \tilde{H}_m y\|_2$

$x_m = x_0 + V_m y_m$;

The m th step of ELMRES

$y_m = A * l_m$; /* enlarge subspace

For $i = 1 : m$,

$$h_{im} = y_m(i);$$

$$y_m = y_m - h_{im} * l_i;$$

/* produce a zero in $y_m(i)$

End;

$h_{m+1,m} = \|y_m\|_\infty$; /* partial pivot

$l_{m+1} = y_m / h_{m+1,m}$ /* normalization step

At the m th step

z_m minimizes $\|\beta e_1 - \tilde{H}_m y\|_2$

$$x_m = x_0 + L_m y_m ;$$

Parallel GMRES

Suppose we partition by rows. Store V and A with the same partition. For the m th step

1. Broadcast new v_m to each processor.
 - (a) matrix multiply performed on individual processors

2. For $i = 1 : m$,
 - (a) each processor communicates its bit of $w_m^t v_i$ (parallel dot product)
 - (b) one proc broadcasts $\alpha = w_m^t v_i$ to all, then perform parallel $w_m - \alpha v_i$

3. End Do

GMRES – fine grain

If message latency is significant, parallel dot products and axpys are slow for vectors of size less than 10-20K .

The number of messages is proportional to basis size, limiting basis size and requiring restarts.

Frequent restarts may prevent and/or slow convergence.

Parallel ELMRES

Suppose we partition A and L by rows. For the m th step

1. Broadcast the new vector l_m to each processor
 - (a) matrix multiply performed on individual processors
2. Broadcast $h_{1m} \dots h_{mm}$ to all processors
3. Each processor sends its candidate pivot to root.
4. Notify winning row
5. Send winning row to root.

ELMRES – medium grain

Efficient even with high latency and relatively few rows per processor.

The number of messages is almost independent of the number of basis vectors.

Can use the memory available on many processors to enlarge size of basis.

A larger basis means fewer restarts and more reliable convergence.

Assumptions for estimating parallel performance

- Latency = 3000 computations (time to start message)
- Time to transmit one floating point = 5 computations.
- 100 nonzero entries per row of a 100K matrix.
- Neglect contention

Hessenberg's Algorithm

- A rearrangement of elementary similarity transformations to work column by column and take advantage of sparsity.
- Reduces sparse matrix to similar Hessenberg form in $2/3n^3$ flops.
- EISPACK elmres uses $5/3n^3$ flops.
- LAPACK dgehrd uses $10/3n^3$ flops .

Backward error for Hessenberg's algorithm

For LU decomposition,

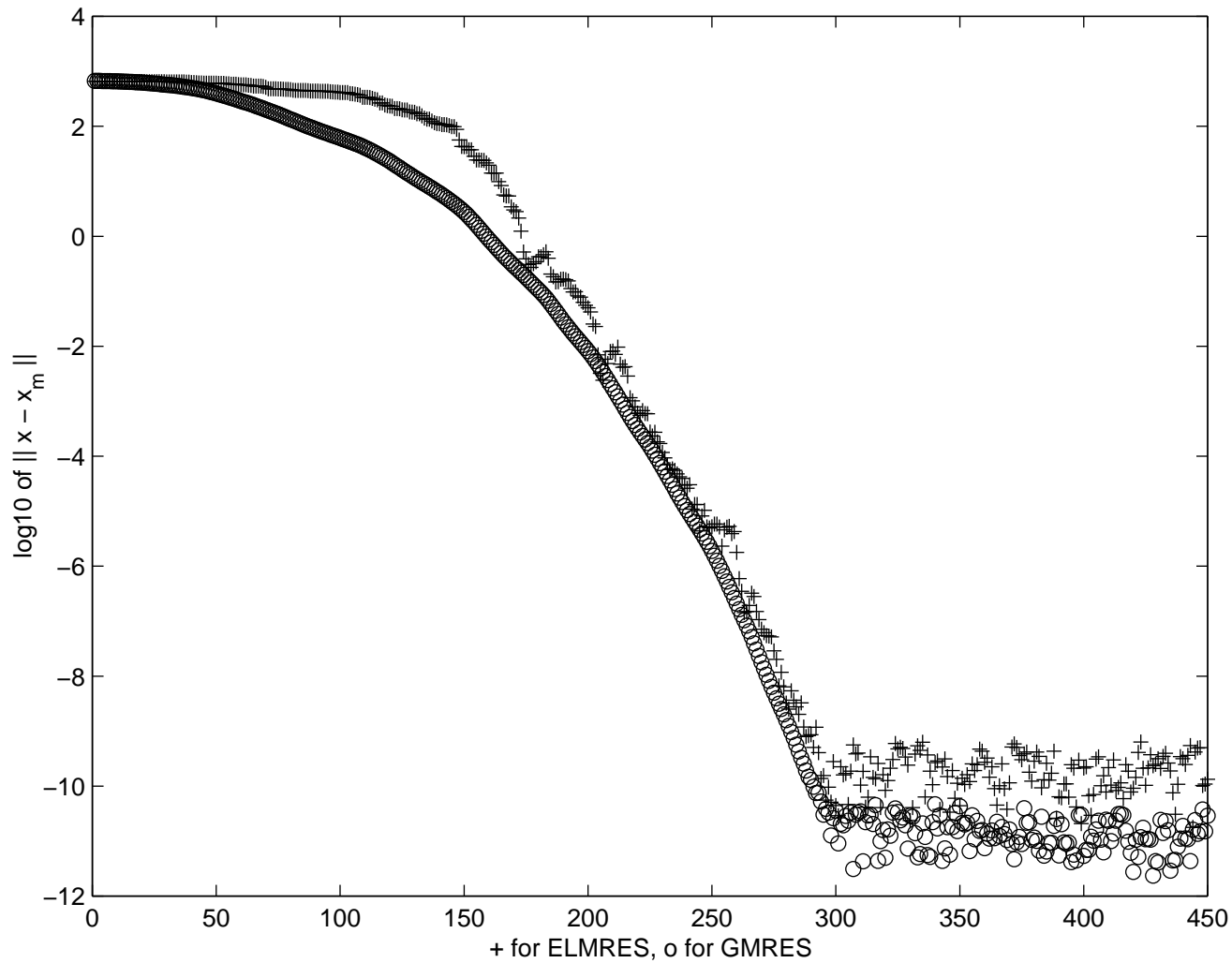
$$LU = A + \Delta A, |\Delta A| \leq \gamma_n |L| |U|$$

For Hessenberg's algorithm,

$$H = L(A + \Delta A)L^{-1}, |\Delta A| \leq \gamma_n (|L| |H| + |A| |L|)$$

As in LU decomposition, large backward errors occur only in the rare event that element growth is large.

Convection Diffusion Errors -- Size 640



Parallel Speed-Ups for GMRES and ELMRES

