Highly Accurate Doubling Algorithms for $M-$matrix Algebraic Riccati Equations

Jungong Xue, Fudan University
(Joint work with Ren-cang Li)

The First Annual Meeting of Applied Mathematics: Frontier Aspects of Applied Mathematics, Dec 6-7, 2015, CASTS

Outline

Matrix Computation

- Input matrix A, compute function $f(A)$
- $f(A)$ may be
  - Solution of linear system $Ax = b$
  - Solution of eigenvalue problem $Ax = \lambda x$
  - $\cdots$
- In general, conventional algorithms are at best backward stable

$$f(A)_{\text{computed}} = f(A+E), \quad \|E\| = O(\mathrm{u})\|A\|$$

Accuracy of Backward Stable Algorithms

- Perturbation analysis

$$\frac{\|f(A+E)-f(A)\|}{\|f(A)\|} \leq \kappa(A)\frac{\|E\|}{\|A\|}$$

- If $\kappa(A)$ is large, backward stable algorithms can't produce accurate solution.

- If $\kappa(A)$ is not large, backward stable algorithms produce accurate solution, but in norm-wise sense. The relative accuracy of small entries in $f(A)$ can't be warranted.

CAE: Compute Accurately and Efficiently

- Accurately: each entry of $f(A)$ is computed with guaranteed relative accuracy
- Efficiently: computation is in polynomial time
- Why CAE?
  - Needed in some applications.
  - It is worthwhile to compute to the accuracy warranted by input data.

`Demmel, ICM 2002, ICIAM 2003, Acta Numerica 2008`

Necessary Condition for CAE

- Entry-wise perturbation analysis: for $|E| \le \epsilon \cdot |A|$,

$$|f(A+E) - f(A)| \le \nu(A)\epsilon \cdot |f(A)|$$

- If $\nu(A) < \infty$ and isn't large, then small relative changes in entries of A cause small relative changes in entries of $f(A)$
- Under traditional model of floating point arithmetic:
  "*The class that we can CAE appears to be identical to the class where all the outputs are in fact accurately determined by the inputs*"
  ——`Demmel, ICIAM 2003`

$M-$MATRIX

- $A \in \mathbb{R}^{n\times n}$ is called an $M-$matrix if

$$A = \rho I - B,$$

  where B is nonnegative and $\rho \geq \rho(B)$, the spectral radius of B.

- If A is a nonsingular $M-$matrix, then A^{-1} is entrywise nonnegative.

$M-$matrix Algebraic Riccati Equation

- An *M-Matrix Algebraic Riccati Equation*(MARE) is the matrix equation

$$XDX - AX - XB + C = 0,$$

for which

$$W = \begin{bmatrix} B & -D \\ -C & A \end{bmatrix},$$

is a nonsingular or an irreducible singular M-matrix.

- MAREs have wide applications in applied probability, transportation theory ...

Minimal Nonnegative Solution

- MARE
$$XDX - AX - XB + C = 0$$
has a unique minimal nonnegative solution Φ, i.e.,
$$\Phi \le X \quad \text{for any other nonnegative solution } X.$$
- The dual equation
$$D - YA - BY + YCY = 0$$
is a MARE and has the minimal nonnegative solution Ψ.

Entrywise Perturbation Analysis for MARE

Let Φ and $\widetilde{\Phi}$ be minimal nonnegative solutions to MAREs

$$XDX - AX - XB + C = 0, \qquad X\widetilde{D}X - \widetilde{A}X - X\widetilde{B} + \widetilde{C} = 0.$$

Suppose $|W - \widetilde{W}| \leq \epsilon |W|$, then

$$\begin{aligned} |(\Phi - \widetilde{\Phi}) \oslash \Phi| &\leq \epsilon \Upsilon \oslash \Phi + O\left(\epsilon^2\right) \\ &\leq \gamma\epsilon \mathbf{1}_{n\times m} + O\left(\epsilon^2\right), \end{aligned}$$

where $\oslash$ denotes the entrywise division, Υ and γ are defined by

$$(A - \Phi D)\Upsilon + \Upsilon(B - D\Phi) = D_A\Phi + \Phi D_B, \qquad \gamma = \max_{i,j}(\Upsilon \oslash \Phi)_{(i,j)}.$$

Xue, Xu and Li, 2012

Doubling Algorithms

- Initialization: constructing E_0, F_0, X_0 and Y_0;
- For $k \geq 0$, iterate

$$\begin{aligned} E_{k+1} &= E_k(I - Y_kX_k)^{-1}E_k, \\ F_{k+1} &= F_k(I - X_kY_k)^{-1}F_k, \\ Y_{k+1} &= Y_k + E_k(I - Y_kX_k)^{-1}Y_kF_k, \\ X_{k+1} &= X_k + F_k(I - X_kY_k)^{-1}X_kE_k. \end{aligned}$$

- Different initializations result in
  - SDA `Guo, Lin and Xu, 2006`
  - SDA-ss `Bini, Meini and Poloni, 2010`
  - ADDA `W.-G. Wang, W.-C. Wang and R.-C. Li, 2012`

Convergence of Doubling Algorithms

If the initial matrices are properly selected,

- all E_k, F_k, X_k, and Y_k are entrywise nonnegative
- all $I - X_k Y_k$ and $I - Y_k X_k$ are nonsingular M-matrices
- All X_k have the same entrywise nonzero pattern as Φ, and all Y_k have the same entrywise nonzero pattern as Ψ
- the sequences $\{X_k\}$ and $\{Y_k\}$ converge increasingly and quadratically to Φ and Ψ, respectively.

Why Highly Accurate Doubling Algorithms

- Small relative perturbations to the entries of A, B, C and D introduces small relative changes to the entries of Φ.
- The doubling algorithms are very efficient for computing Φ.
- It is desirable to design highly accurate doubling algorithms which compute Φ as accurately as the input data deserves.

Ideas for Highly Accurate Doubling Algorithms

- The algorithm should be cancellation-free.
- A proper stopping criterion which guarantees small entrywise relative error.

Triplet Representation of $M-$matrices

- A triplet representation $\{N_A, \boldsymbol{u}, \boldsymbol{v}\}$ of an $M-$matrix $A \in \mathbb{R}^{n\times n}$ consists of

$$N_A = \mathrm{diag}(A) - A,\ 0 < \boldsymbol{u} \in \mathbb{R}^n, \text{ and } \boldsymbol{v} = A\boldsymbol{u} \geq 0,$$

- Triplet representation is either known explicitly or has to be computed.

`Alfa, Xue and Ye, 2002; Xue, Xu and Li, 2012`

Entrywise Perturbation Analysis via Triplet Representation

If

$$|N_A - N_{\tilde{A}}| \le \epsilon N_A, \quad |\boldsymbol{u} - \tilde{\boldsymbol{u}}| \le \epsilon \boldsymbol{u}, \quad |\boldsymbol{v} - \tilde{\boldsymbol{v}}| \le \epsilon \boldsymbol{v},$$

then

$$|A^{-1} - \tilde{A}^{-1}| \le ((2n-1)\epsilon + O(\epsilon^2))A^{-1}$$

`Alfa, Xue and Ye, 2002`

GTH-like Algorithm

- Let A has triplet representation $(N_A, \boldsymbol{u}, \boldsymbol{v})$
- One step of Gaussian elimination

$$A = \begin{bmatrix} a_{11} & -\boldsymbol{a}^T \\ -\boldsymbol{b} & A_1 \end{bmatrix} = \begin{bmatrix} 1 & \\ -\frac{1}{a_{11}}\boldsymbol{b} & I \end{bmatrix} \begin{bmatrix} a_{11} & -\boldsymbol{a}^T \\ & A^{(1)} \end{bmatrix}$$

with $A^{(1)} = A_1 - \frac{1}{a_{11}}\boldsymbol{b}\boldsymbol{a}^T$

- Let $\boldsymbol{u} = \begin{bmatrix} u_1 \\ \bar{\boldsymbol{u}} \end{bmatrix}, \boldsymbol{v} = \begin{bmatrix} v_1 \\ \bar{\boldsymbol{v}} \end{bmatrix}$

$$A^{(1)}\bar{\boldsymbol{u}} = \bar{\boldsymbol{v}} + \frac{v_1}{a_{11}}\boldsymbol{b}$$

GTH-LIKE ALGORITHM

- Construct the triplet representation $(N_{A^{(1)}}, \bar{\boldsymbol{u}}, \boldsymbol{v}^{(1)})$
  - Compute the off-diagonal entries of $N_{A^{(1)}}$

$$|a_{ij}^{(1)}| = |a_{ij}| + \frac{b_i a_j}{a_{11}}, \qquad i \neq j$$

  - Compute

$$\boldsymbol{v}^{(1)} =: A^{(1)}\bar{\boldsymbol{u}} = \bar{\boldsymbol{v}} + \frac{v_1}{a_{11}}\boldsymbol{b}$$

- No substraction of same signed numbers
- Compute A^{-1} with entrywise relative accuracy $O(\mathfrak{u})$.

`Alfa, Xue and Ye, 2002`

GTH-like Algorithms in Doubling Algorithms

- Construct triplet representations of M-matrices $I - X_kY_k$ and $I - Y_kX_k$.
- Compute $(I - X_kY_k)^{-1}$ and $(I - Y_kX_k)^{-1}$ using the GTH-like algorithm.

Old Method of Constructing Triplet Representation

At each step, construct triplet representations of $I - X_k Y_k$ and $I - Y_k X_k$ by solving some linear systems,

- Time-consuming
- Not cancellation-free
- The entrywise relative accuracy of the computed $(I - X_k Y_k)^{-1}$ and $(I - Y_k X_k)^{-1}$ depends on some condition number

`Xue, Xu and Li, 2012`

Nguyen and Poloni's Work

For the special case $W\mathbf{1} = 0$, they develop a method to construct triplet representations for $I - X_k Y_k$ and $I - Y_k X_k$ in a cancellation-free manner.

`Nguyen and Poloni, 2015`

Our Contributions

- Extends Nguyen and Poloni's work to all MAREs.
- Proposing an entrywise relative residual which reveals the entrywise relative accuracy of all entries.

Old Initialization of ADDA

- Select

$$\hat{\alpha} \geq \max_{1\leq i\leq m} A_{(i,i)}, \quad \hat{\beta} \geq \max_{1\leq j\leq n} B_{(j,j)},$$

- Set

$$\begin{aligned} A_{\hat{\beta}} &= A + \hat{\beta} I_n, & B_{\hat{\alpha}} &= B + \hat{\alpha} I_m, \\ U_{\hat{\alpha}\hat{\beta}} &= A_{\hat{\beta}} - C B_{\hat{\alpha}}^{-1} D, & V_{\hat{\alpha}\hat{\beta}} &= B_{\hat{\alpha}} - D A_{\hat{\beta}}^{-1} C, \end{aligned}$$

- Set

$$\begin{aligned} \hat{E}_0 &= -I_m + (\hat{\alpha} + \hat{\beta}) V_{\hat{\alpha}\hat{\beta}}^{-1}, & \hat{F}_0 &= -I_n + (\hat{\alpha} + \hat{\beta}) U_{\hat{\alpha}\hat{\beta}}^{-1}, \\ \hat{Y}_0 &= (\hat{\alpha} + \hat{\beta}) B_{\hat{\alpha}}^{-1} D U_{\hat{\alpha}\hat{\beta}}^{-1}, & \hat{X}_0 &= (\hat{\alpha} + \hat{\beta}) U_{\hat{\alpha}\hat{\beta}}^{-1} C B_{\hat{\alpha}}^{-1}. \end{aligned}$$

W.-G. Wang, W.-C. Wang and R.-C. Li, 2012

Compact Form of Old Initialization

Original initialization of ADDA can be combined into

$$\begin{bmatrix} \hat{E}_0 & \hat{Y}_0 \\ \hat{X}_0 & \hat{F}_0 \end{bmatrix} = \begin{bmatrix} B + \hat{\alpha} I_m & -D \\ -C & A + \hat{\beta} I_n \end{bmatrix}^{-1} \begin{bmatrix} \hat{\beta} I_m - B & D \\ C & \hat{\alpha} I_n - A \end{bmatrix}.$$

`Poloni and Reis, 2011`

Nguyen and Poloni's Initialization

- Set $\alpha = \hat{\alpha}^{-1}, \beta = \hat{\beta}^{-1}$ and let

$$\begin{bmatrix} E_0 & Y_0 \\ X_0 & F_0 \end{bmatrix} = \begin{bmatrix} \alpha I & \\ & \beta I \end{bmatrix} \begin{bmatrix} \hat{E}_0 & \hat{Y}_0 \\ \hat{X}_0 & \hat{F}_0 \end{bmatrix} \begin{bmatrix} \hat{\beta} I & \\ & \hat{\alpha} I \end{bmatrix},$$

- For $k \geq 0$

$$E_k = \left(\frac{\alpha}{\beta}\right)^{2^k} \hat{E}_k, \quad F_k = \left(\frac{\beta}{\alpha}\right)^{2^k} \hat{F}_k, \quad \hat{X}_k = X_k, \quad \hat{Y}_k = Y_k.$$

- Unify three main doubling algorithms: SDA ($\alpha = \beta$), SDA-ss ($\alpha = 0$ or $\beta = 0$), ADDA (in general).

`Nguyen and Poloni, 2015`

Compact Form of New Initialization

Nguyen and Poloni's initialization can be combined into

$$\begin{bmatrix} E_0 & Y_0 \\ X_0 & F_0 \end{bmatrix} = \begin{bmatrix} \alpha B + I_m & -\beta D \\ -\alpha C & \beta A + I_n \end{bmatrix}^{-1} \begin{bmatrix} I_m - \beta B & \alpha D \\ \beta C & I_n - \alpha A \end{bmatrix}.$$

Triplet Representation of W

The triple representation $\{N_W, \boldsymbol{u}, \boldsymbol{v}\}$ of W, i.e.,

$$\boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} > 0, \quad \boldsymbol{v} = W\boldsymbol{u} = \begin{bmatrix} B & -D \\ -C & A \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix} \geq 0,$$

is either known explicitly or has to be computed

`Xue, Xu and Li, 2012`

Uniformly Bounded E_k and F_k

Let E_0, F_0, Y_0, X_0 be constructed by Poloni and Reis's method. Then

$$\begin{bmatrix} E_k & Y_k \\ X_k & F_k \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} \le \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} \quad \text{for all } k \ge 0.$$

In particular, if $\boldsymbol{v} = 0$, then

$$\begin{bmatrix} E_k & Y_k \\ X_k & F_k \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} \quad \text{for all } k \ge 0.$$

Triplet Representations via Auxiliary Vectors

Let

$$\begin{bmatrix} \boldsymbol{w}_1^{(k)} \\ \boldsymbol{w}_2^{(k)} \end{bmatrix} := \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} - \begin{bmatrix} E_k & Y_k \\ X_k & F_k \end{bmatrix} \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \end{bmatrix} \geq 0,$$

Since

$$(I - X_k Y_k)\boldsymbol{u}_2 = \underbrace{\boldsymbol{w}_2^{(k)} + F_k \boldsymbol{u}_2 + X_k(E_k \boldsymbol{u}_1 + \boldsymbol{w}_1^{(k)})}_{=: \boldsymbol{v}_2^{(k)}} \geq 0,$$

$$(I - Y_k X_k)\boldsymbol{u}_1 = \underbrace{\boldsymbol{w}_1^{(k)} + E_k \boldsymbol{u}_1 + Y_k(F_k \boldsymbol{u}_2 + \boldsymbol{w}_2^{(k)})}_{=: \boldsymbol{v}_1^{(k)}} \geq 0,$$

Triplet Representation

$$I - Y_k X_k = \{N_{I - Y_k X_k}, \boldsymbol{u}_1, \boldsymbol{v}_1^{(k)}\},$$

$$I - X_k Y_k = \{N_{I - X_k Y_k}, \boldsymbol{u}_2, \boldsymbol{v}_2^{(k)}\}.$$

Update of Auxiliary Vectors

- Initial auxiliary vector $\begin{bmatrix} \boldsymbol{w}_1^{(0)} \\ \boldsymbol{w}_2^{(0)} \end{bmatrix}$ can be calculated in a cancellation-free manner.
- The auxiliary vectors can be computed recursively

$$\boldsymbol{w}_1^{(k+1)} = \boldsymbol{w}_1^{(k)} + E_k(I - Y_k X_k)^{-1}[\boldsymbol{w}_1^{(k)} + Y_k \boldsymbol{w}_2^{(k)}],$$
$$\boldsymbol{w}_2^{(k+1)} = \boldsymbol{w}_2^{(k)} + F_k(I - X_k Y_k)^{-1}[X_k \boldsymbol{w}_1^{(k)} + \boldsymbol{w}_2^{(k)}].$$

Remark. As $E_k(I - Y_k X_k)^{-1}$ and $F_k(I - X_k Y_k)^{-1}$ has been calculated during the doubling procedure, the cost of update of residual $\boldsymbol{v}_i^{(k)}$ is negligible

Entrywise Relative Residual

Splitting

$$A = D_A - N_A, \quad D_A = \mathrm{diag}(A),$$
$$B = D_B - N_B, \quad D_B = \mathrm{diag}(B).$$

Define

$$\mathcal{R}_L(X) \equiv XDX + N_A X + XN_B + C, \quad \mathcal{R}_R(X) \equiv D_A X + XD_B,$$

Let $\widetilde{\Phi}$ be a nonnegative approximation of Φ, define

$$\mathrm{ERRes}(\widetilde{\Phi}) = \max_{i,j} \frac{|\mathcal{R}_L(\widetilde{\Phi}) - \mathcal{R}_R(\widetilde{\Phi})|_{(i,j)}}{[\mathcal{R}_R(\widetilde{\Phi})]_{(i,j)}}.$$

Entrywise Relative Error

Theorem

Let $\widetilde{\Phi} \approx \Phi$ satisfy $0 \le \widetilde{\Phi} \le \Phi$ and that $\widetilde{\Phi}$ and Φ share the same entrywise nonzero pattern. If ERRes *is no bigger than ϵ and if ϵ is sufficiently tiny, then*

$$\begin{aligned} |(\Phi - \widetilde{\Phi}) \oslash \Phi| &\le \epsilon \Upsilon \oslash \Phi + O\left(\epsilon^2\right) \\ &\le \gamma \epsilon \mathbf{1}_{n \times m} + O\left(\epsilon^2\right), \end{aligned}$$

where $\oslash$ denotes the entrywise division, Υ and γ are defined by

$$(A - \Phi D)\Upsilon + \Upsilon(B - D\Phi) = D_A \Phi + \Phi D_B, \quad \gamma = \max_{i,j} (\Upsilon \oslash \Phi)_{(i,j)}.$$

Stopping Criterion

- First check Kahan's criterion

$$\frac{(X_{k+1} - X_k)_{ij}^2}{(X_k - X_{k-1})_{ij} - (X_k - X_{k+1})_{ij}} \le \epsilon \cdot (X_{k+1})_{ij} \text{ for all } i \text{ and } j$$

- If Kahan's criterion is satisfied, check (probably with a different ϵ) if

$$\text{ERRes}(X_{k+1}) \le \epsilon$$

Algorithms Compared

- accADDA: use GTH-like algorithm together with cancellation-free triplet representation construction to compute all the inverses
- *plain* ADDA: simply use the usually Gaussian elimination with partial pivoting, such as MATLAB's operators "\" and "/", to compute all the inverses

Entrywise Relative Error

Let $\widetilde{\Phi}$ be a nonnegative approximation of Φ, define

$$\mathrm{ERRrr}(\widetilde{\Phi}) = \max_{i,j} \frac{|(\widetilde{\Phi} - \Phi)_{(i,j)}|}{\Phi_{(i,j)}}.$$

Remark. The 'Exact' Φ is either known explicitly or computed by *Maple* with 100 decimal digits.

Example 1

$$A = 18 \cdot I_2, \quad B = 180002 \cdot I_{18} - 10^4 \cdot \mathbf{1}_{18\times 18}$$

$$C = \mathbf{1}_{2\times 18}, \qquad D = C^{\mathrm{T}}.$$

Example 2

$$B = \begin{bmatrix} 3+\delta & -1 & & \\ & 3+\delta & \ddots & \\ & & \ddots & -1 \\ -1 & & & 3+\delta \end{bmatrix} \in \mathbb{R}^{100\times 100}$$

$$C = 2I_{100}, \quad A = B, \quad D = C,$$

where $\delta = 2^{-24}$

Example 3

$$A = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 15+\delta & -5 \\ 0 & -5 & 15 \end{bmatrix}, \qquad B = \frac{1}{1.001}\begin{bmatrix} 15 & -5 & 0 \\ -5 & 15 & 0 \\ 0 & 0 & 5 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 4 \\ 5 & 5 & \delta \\ 5 & 5 & 0 \end{bmatrix}, \qquad D = \frac{1}{1.001}\begin{bmatrix} 0 & 5 & 5 \\ 0 & 5 & 5 \\ 4 & 1 & 0 \end{bmatrix},$$

where $\delta = 10^{-8}$.

Numerical Results

Eg.	accADDA		plain ADDA		γ
	ERRrr	ERRes	ERRrr	ERRes	
1	$1.2 \cdot 10^{-15}$	$3.9 \cdot 10^{-16}$	$4.5 \cdot 10^{-13}$	$3.9 \cdot 10^{-16}$	$1.0 \cdot 10^{4}$
2	$2.1 \cdot 10^{-15}$	$1.7 \cdot 10^{-15}$	$5.9 \cdot 10^{-12}$	$1.7 \cdot 10^{-14}$	$1.1 \cdot 10^{3}$
3	$4.3 \cdot 10^{-16}$	$3.1 \cdot 10^{-16}$	$3.5 \cdot 10^{-10}$	$4.5 \cdot 10^{-11}$	$6.2 \cdot 10^{2}$

Conclusion

- Construct triplet representation for all involved M-matrices in doubling algorithms in a cancellation-free manner for all MAREs.
- Propose an entrywise relative residual that reflects relative accuracy for all entries.
- New entrywise perturbation analysis is required.