Highly accurate Numerical Linear Algebra via Rank Revealing Decompositions

Froilán M. Dopico

Department of Mathematics and ICMAT, Universidad Carlos III de Madrid, Spain

joint work with

Nieves Castro-González (Universidad Politécnica de Madrid, Spain)

Johan Ceballos (Universidad Carlos III de Madrid, Spain)

Plamen Koev (San José State University, USA)

Juan M. Molera (Universidad Carlos III de Madrid, Spain)

SIAM Conference on Applied Linear Algebra. Minisymposium on "Different perspectives of conditioning and numerical stability". Universitat Politècnica de València. June 18th-22nd, 2012

Abstract (I)

- Given the factors of a rank revealing decomposition (RRD)

$$A = XDY,$$

where X and Y are well-conditioned, and D is diagonal and non-singular (and, so, it inherits the potential ill-conditioning of A).

- we present briefly algorithms developed by

1. Demmel, Gu, Eisenstat, Slapničar, Veselić, and Drmač, LAA, 1999.
2. D., Koev, and Molera, Numer. Math., 2009.
3. D. and Molera, IMA J. Numer. Anal., 2011.
4. Castro-González, Ceballos, D., Molera, to be submitted July-2012.

Abstract (I)

- Given the factors of a rank revealing decomposition (RRD)

$$A = XDY,$$

where X and Y are well-conditioned, and D is diagonal and non-singular (and, so, it inherits the potential ill-conditioning of A).

- we present briefly algorithms developed by

1. Demmel, Gu, Eisenstat, Slapničar, Veselić, and Drmač, LAA, 1999.
2. D., Koev, and Molera, Numer. Math., 2009.
3. D. and Molera, IMA J. Numer. Anal., 2011.
4. Castro-González, Ceballos, D., Molera, to be submitted July-2012.

Abstract (II)

- These algorithms allow us to compute (in the same order)
  - SVD of $A = XDY$,
  - Eigenvalues and eigenvectors of symmetric $A = XDX^T$,
  - Solution of linear system $(XDY)x = b$,
  - Solution of least squares problem $\min_x \|b - (XDY)x\|_2$,

  with much more accuracy than standard algorithms for very ill-conditioned matrices.

- This accuracy means in most cases relative errors bounded by

$$O(\mathtt{u}),$$

where $\mathtt{u}$ is the unit roundoff,

- but, more precisely, bounded by

$$O(\kappa\, \mathtt{u}),$$

where κ is a relevant condition number for each problem with respect perturbations of the factors. κ is usually of order $O(1)$ and much smaller than the traditional condition number of $A = XDY$.

Abstract (II)

- These algorithms allow us to compute (in the same order)
  - SVD of $A = XDY$,
  - Eigenvalues and eigenvectors of symmetric $A = XDX^T$,
  - Solution of linear system $(XDY)x = b$,
  - Solution of least squares problem $\min_x \|b - (XDY)x\|_2$,

  with much more accuracy than standard algorithms for very ill-conditioned matrices.

- This accuracy means in most cases relative errors bounded by

$$O(\mathtt{u}),$$

  where $\mathtt{u}$ is the unit roundoff,

- but, more precisely, bounded by

$$O(\kappa\,\mathtt{u}),$$

  where κ is a relevant condition number for each problem with respect perturbations of the factors. κ is usually of order $O(1)$ and much smaller than the traditional condition number of $A = XDY$.

Abstract (II)

- These algorithms allow us to compute (in the same order)
  - SVD of $A = XDY$,
  - Eigenvalues and eigenvectors of symmetric $A = XDX^T$,
  - Solution of linear system $(XDY)x = b$,
  - Solution of least squares problem $\min_x \|b - (XDY)x\|_2$,

  with much more accuracy than standard algorithms for very ill-conditioned matrices.
- This accuracy means in most cases relative errors bounded by

  $$O(\mathtt{u}),$$

  where $\mathtt{u}$ is the unit roundoff,
- but, more precisely, bounded by

  $$O(\kappa\,\mathtt{u}),$$

  where κ is a relevant condition number for each problem with respect perturbations of the factors. κ is usually of order $O(1)$ and much smaller than the traditional condition number of $A = XDY$.

Abstract (III)

- **Most important message of this talk:** From a RRD $A = XDY$ of a matrix, that may be arbitrarily ill-conditioned, it is possible to solve (almost) all basic problems of Numerical Linear Algebra with relative errors of $O(\mathtt{u})$ and with computational costs of the same order as traditional algorithms.
- Only the nonsymmetric eigenvalue problem is excluded from this approach.
- Therefore, for those classes of matrices for which RRDs can be accurately computed, it is possible to solve accurately (almost) all basic problems of Numerical Linear Algebra, independently of the magnitude of the traditional condition number of the matrix.
- Key idea: Algorithms for RRDs never form the matrix. They work directly on the factors and deal with the ill-conditioned diagonal factor D in a special and highly accurate way.

Abstract (III)

- **Most important message of this talk:** From a RRD $A = XDY$ of a matrix, that may be arbitrarily ill-conditioned, it is possible to solve (almost) all basic problems of Numerical Linear Algebra with relative errors of $O(\mathtt{u})$ and with computational costs of the same order as traditional algorithms.
- Only the nonsymmetric eigenvalue problem is excluded from this approach.
- Therefore, for those classes of matrices for which RRDs can be accurately computed, it is possible to solve accurately (almost) all basic problems of Numerical Linear Algebra, independently of the magnitude of the traditional condition number of the matrix.
- Key idea: Algorithms for RRDs never form the matrix. They work directly on the factors and deal with the ill-conditioned diagonal factor D in a special and highly accurate way.

Abstract (III)

- **Most important message of this talk:** From a RRD $A = XDY$ of a matrix, that may be arbitrarily ill-conditioned, it is possible to solve (almost) all basic problems of Numerical Linear Algebra with relative errors of $O(\mathtt{u})$ and with computational costs of the same order as traditional algorithms.
- Only the nonsymmetric eigenvalue problem is excluded from this approach.
- Therefore, for those classes of matrices for which RRDs can be accurately computed, it is possible to solve accurately (almost) all basic problems of Numerical Linear Algebra, independently of the magnitude of the traditional condition number of the matrix.
- Key idea: Algorithms for RRDs never form the matrix. They work directly on the factors and deal with the ill-conditioned diagonal factor D in a special and highly accurate way.

Abstract (III)

- **Most important message of this talk:** From a RRD $A = XDY$ of a matrix, that may be arbitrarily ill-conditioned, it is possible to solve (almost) all basic problems of Numerical Linear Algebra with relative errors of $O(\mathtt{u})$ and with computational costs of the same order as traditional algorithms.
- Only the nonsymmetric eigenvalue problem is excluded from this approach.
- Therefore, for those classes of matrices for which RRDs can be accurately computed, it is possible to solve accurately (almost) all basic problems of Numerical Linear Algebra, independently of the magnitude of the traditional condition number of the matrix.
- Key idea: Algorithms for RRDs never form the matrix. They work directly on the factors and deal with the ill-conditioned diagonal factor D in a special and highly accurate way.

Outline

1. Accurate rank revealing decompositions (RRDs)
2. Accurate solution of linear systems
3. Accurate solution of least squares problems
4. Accurate eigenvalues/vectors of symmetric matrices
5. Accurate SVD
6. Conclusions and open problems

Outline

1. **Accurate rank revealing decompositions (RRDs)**
2. Accurate solution of linear systems
3. Accurate solution of least squares problems
4. Accurate eigenvalues/vectors of symmetric matrices
5. Accurate SVD
6. Conclusions and open problems

Definition of RRD

Definition (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

An RRD of $A \in \mathbb{C}^{m\times n}$ is a factorization

$$A = XDY,$$

where $X \in \mathbb{C}^{m\times r}$, $D = \mathrm{diag}(d_1, d_2, \ldots, d_r) \in \mathbb{C}^{r\times r}$ is nonsingular, and $Y \in \mathbb{C}^{r\times n}$ are such that

- $\mathrm{rank}\, A = \mathrm{rank}\, X = \mathrm{rank}\, D = \mathrm{rank}\, Y = r$, and
- X and Y are well conditioned.

$$A = XDY = \begin{bmatrix} \times & \times \\ \times & \times \\ \times & \times \\ \times & \times \\ \times & \times \end{bmatrix} \begin{bmatrix} \times & \\ & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \end{bmatrix}$$

Definition of RRD

Definition (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

An RRD of $A \in \mathbb{C}^{m\times n}$ is a factorization

$$A = XDY,$$

where $X \in \mathbb{C}^{m\times r}$, $D = \operatorname{diag}(d_1, d_2, \ldots, d_r) \in \mathbb{C}^{r\times r}$ is nonsingular, and $Y \in \mathbb{C}^{r\times n}$ are such that

- $\operatorname{rank} A = \operatorname{rank} X = \operatorname{rank} D = \operatorname{rank} Y = r$, and
- X and Y are well conditioned.

$$A = XDY = \begin{bmatrix} \times & \times \\ \times & \times \\ \times & \times \\ \times & \times \\ \times & \times \end{bmatrix} \begin{bmatrix} \times & \\ & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times \\ \times & \times & \times \end{bmatrix}$$

Definition of accurate RRD

Definition (Demmel et al, LAA, 1999)

Let $\widehat{X} \in \mathbb{C}^{m\times r}$, $\widehat{D} = \mathrm{diag}(\widehat{d}_1, \widehat{d}_2, \ldots, \widehat{d}_r) \in \mathbb{C}^{r\times r}$, and $\widehat{Y} \in \mathbb{C}^{r\times n}$ be the factors of an RRD $A = XDY$ computed by a certain algorithm. We say that $\widehat{X}\widehat{D}\widehat{Y}$ has been accurately computed if

$$\frac{\|\widehat{X} - X\|_2}{\|X\|_2} = O(\mathtt{u}), \quad \frac{\|\widehat{Y} - Y\|_2}{\|Y\|_2} = O(\mathtt{u}), \quad \text{and}$$

$$\frac{|\widehat{d}_i - d_i|}{|d_i|} = O(\mathtt{u}), \quad i = 1 : r.$$

- This is the accuracy that we need to apply the algorithms of this talk to $\widehat{X}\widehat{D}\widehat{Y}$ and to perform accurate Numerical Linear Algebra on A.
- This accuracy can be obtained only for special types of matrices through highly structured implementations of Gaussian elimination with complete pivoting (GECP) (and for one class via QRCP).

Definition of accurate RRD

Definition (Demmel et al, LAA, 1999)

Let $\widehat{X} \in \mathbb{C}^{m\times r}$, $\widehat{D} = \mathrm{diag}(\widehat{d}_1, \widehat{d}_2, \ldots, \widehat{d}_r) \in \mathbb{C}^{r\times r}$, and $\widehat{Y} \in \mathbb{C}^{r\times n}$ be the factors of an RRD $A = XDY$ computed by a certain algorithm. We say that $\widehat{X}\widehat{D}\widehat{Y}$ has been accurately computed if

$$\frac{\|\widehat{X} - X\|_2}{\|X\|_2} = O(\mathtt{u}), \quad \frac{\|\widehat{Y} - Y\|_2}{\|Y\|_2} = O(\mathtt{u}), \quad \text{and}$$

$$\frac{|\widehat{d}_i - d_i|}{|d_i|} = O(\mathtt{u}), \quad i = 1 : r.$$

- This is the accuracy that we need to apply the algorithms of this talk to $\widehat{X}\widehat{D}\widehat{Y}$ and to perform accurate Numerical Linear Algebra on A.
- This accuracy can be obtained only for special types of matrices through highly structured implementations of Gaussian elimination with complete pivoting (GECP) (and for one class via QRCP).

Definition of accurate RRD

Definition (Demmel et al, LAA, 1999)

Let $\widehat{X} \in \mathbb{C}^{m\times r}$, $\widehat{D} = \mathrm{diag}(\widehat{d}_1, \widehat{d}_2, \ldots, \widehat{d}_r) \in \mathbb{C}^{r\times r}$, and $\widehat{Y} \in \mathbb{C}^{r\times n}$ be the factors of an RRD $A = XDY$ computed by a certain algorithm. We say that $\widehat{X}\widehat{D}\widehat{Y}$ has been accurately computed if

$$\frac{\|\widehat{X} - X\|_2}{\|X\|_2} = O(\mathtt{u}), \quad \frac{\|\widehat{Y} - Y\|_2}{\|Y\|_2} = O(\mathtt{u}), \quad \text{and}$$

$$\frac{|\widehat{d}_i - d_i|}{|d_i|} = O(\mathtt{u}), \quad i = 1 : r.$$

- This is the accuracy that we need to apply the algorithms of this talk to $\widehat{X}\widehat{D}\widehat{Y}$ and to perform accurate Numerical Linear Algebra on A.
- This accuracy can be obtained only for special types of matrices through highly structured implementations of Gaussian elimination with complete pivoting (GECP) (and for one class via QRCP).

Matrices for which accurate RRDs can be computed

- Cauchy, Scaled-Cauchy, Vandermonde (DFT + GECP). [Demmel]
- Diagonally Dominant M-Matrices. [Demmel and Koev, Peña]
- Polynomial Vandermonde. [Demmel and Koev]
- Well Scalable Symmetric Positive Definite. [Demmel and Veselić]
- Some well Scalable Symmetric Indefinite. [Slapničar and Veselić]
- Scaled Diagonally Dominant. [Barlow and Demmel]
- Acyclic Matrices (include bidiagonal). [Demmel and Gragg]
- Diagonally Dominant. [Ye, D. and Koev]
- Totally Nonnegative. [D. and Koev]
- DSTU. [Demmel]
- Graded Matrices. [Demmel et al.] [Higham]
- Symmetric versions. [D. and Koev] [D., Molera, Ceballos] [Peláez and Moro]
- ....

Outline

Algorithm (D. & Molera, IMA J. Numer. Anal., 2011)

- Input: $A \in \mathbb{C}^{n \times n}$ (nonsingular), $b \in \mathbb{C}^n$
- Output: x solution of $Ax = b$

1. Compute an accurate RRD of $A = XDY$
2. Solve the three systems

$$\begin{array}{ccc} X\,s = b & \longrightarrow & s \\ D\,w = s & \longrightarrow & w \\ Y\,x = w & \longrightarrow & x \end{array}$$

- $X\,s = b$ and $Y\,x = w$ are solved by any backward stable method.
- $w_i = s_i / d_{ii}, \quad i = 1:n.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(n^3)$ flops.

The algorithm for Linear Systems

Algorithm (D. & Molera, IMA J. Numer. Anal., 2011)

- Input: $A \in \mathbb{C}^{n\times n}$ (nonsingular), $b \in \mathbb{C}^n$
- Output: x solution of $Ax = b$

1. Compute an accurate RRD of $A = XDY$
2. Solve the three systems

$$\begin{array}{lcc} X\,s = b & \longrightarrow & s \\ D\,w = s & \longrightarrow & w \\ Y\,x = w & \longrightarrow & x \end{array}$$

- $X\,s = b$ and $Y\,x = w$ are solved by any backward stable method.
- $w_i = s_i / d_{ii}, \quad i = 1 : n.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(n^3)$ flops.

The algorithm for Linear Systems

Algorithm (D. & Molera, IMA J. Numer. Anal., 2011)

- Input: $A \in \mathbb{C}^{n\times n}$ (nonsingular), $b \in \mathbb{C}^n$
- Output: x solution of $Ax = b$

1. Compute an accurate RRD of $A = XDY$
2. Solve the three systems

$$\begin{array}{rcl} X\,s = b & \longrightarrow & s \\ D\,w = s & \longrightarrow & w \\ Y\,x = w & \longrightarrow & x \end{array}$$

- $X\,s = b$ and $Y\,x = w$ are solved by any backward stable method.
- $w_i = s_i/d_{ii}, \quad i = 1:n$.
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(n^3)$ flops.

The algorithm for Linear Systems

Algorithm (D. & Molera, IMA J. Numer. Anal., 2011)

- Input: $A \in \mathbb{C}^{n\times n}$ (nonsingular), $b \in \mathbb{C}^n$
- Output: x solution of $Ax = b$

1. Compute an accurate RRD of $A = XDY$
2. Solve the three systems

$$\begin{array}{ccc} X\,s = b & \longrightarrow & s \\ D\,w = s & \longrightarrow & w \\ Y\,x = w & \longrightarrow & x \end{array}$$

- $X\,s = b$ and $Y\,x = w$ are solved by any backward stable method.
- $w_i = s_i / d_{ii}, \quad i = 1:n.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(n^3)$ flops.

The algorithm for Linear Systems

Algorithm (D. & Molera, IMA J. Numer. Anal., 2011)

- Input: $A \in \mathbb{C}^{n\times n}$ (nonsingular), $b \in \mathbb{C}^n$
- Output: x solution of $Ax = b$

1. Compute an accurate RRD of $A = XDY$
2. Solve the three systems

$$\begin{array}{ccc} X\,s = b & \longrightarrow & s \\ D\,w = s & \longrightarrow & w \\ Y\,x = w & \longrightarrow & x \end{array}$$

- $X\,s = b$ and $Y\,x = w$ are solved by any backward stable method.
- $w_i = s_i/d_{ii}, \quad i = 1:n.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(n^3)$ flops.

Theorem (D. & Molera, IMA J. Numer. Anal., 2011)

If $\widehat{x}$ is the solution of $Ax = b$ computed by the algorithm in previous slide ($A = XDY$), then

$$\frac{\|\widehat{x}-x\|_2}{\|x\|_2} \leq O(\mathtt{u}) \max\{\kappa_2(X), \kappa_2(Y)\} \frac{\|A^{-1}\|_2 \|b\|_2}{\|x\|_2}$$

To be compared with

$$\frac{\|\widehat{x}-x\|_2}{\|x\|_2} \leq O(\mathtt{u})\, \kappa_2(A),$$

that holds for traditional algorithms as GEPP, GECP, QR,...

The error for Linear Systems

Theorem (D. & Molera, IMA J. Numer. Anal., 2011)

If $\widehat{x}$ is the solution of $Ax = b$ computed by the algorithm in previous slide ($A = XDY$), then

$$\frac{\|\widehat{x} - x\|_2}{\|x\|_2} \leq O(\mathtt{u}) \max\{\kappa_2(X)\,,\, \kappa_2(Y)\} \frac{\|A^{-1}\|_2 \|b\|_2}{\|x\|_2}$$

To be compared with

$$\frac{\|\widehat{x} - x\|_2}{\|x\|_2} \leq O(\mathtt{u})\, \kappa_2(A) \quad ,$$

that holds for traditional algorithms as GEPP, GECP, QR,...

$\left(\|A^{-1}\|_2\|b\|_2 \,/\, \|x\|_2\right)$ is "almost always" a moderate number (Chan & Foulser, 1988)

- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \kappa_2(A)$, but even more **"almost always"**
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \ll \kappa_2(A)$, if A very ill-conditioned because,
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \dfrac{1}{\cos\theta(u_n, b)}$,

  where u_n left-singular vector of A of smallest singular value.

Example in MATLAB:

```
>> V = vander(randn(20,1)); b=randn(20,1);
>> cond(V)= 7.1021e+11
>> x = V\b;
>> norm(inv(V))*norm(b)/norm(x) = 8.4317
```

$(\|A^{-1}\|_2\|b\|_2 \,/\, \|x\|_2)$ is "almost always" a moderate number (Chan & Foulser, 1988)

- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \kappa_2(A)$, but even more **"almost always"**
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \ll \kappa_2(A)$, if A very ill-conditioned because,
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \dfrac{1}{\cos\theta(u_n, b)}$,

  where u_n left-singular vector of A of smallest singular value.

Example in MATLAB:

```
>> V = vander(randn(20,1)); b=randn(20,1);
>> cond(V)= 7.1021e+11
>> x = V\b;
>> norm(inv(V))*norm(b)/norm(x) = 8.4317
```

$(\|A^{-1}\|_2\|b\|_2 \,/\, \|x\|_2)$ is "almost always" a moderate number (Chan & Foulser, 1988)

- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \kappa_2(A)$, but even more **"almost always"**
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \ll \kappa_2(A)$, if A very ill-conditioned because,
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \dfrac{1}{\cos\theta(u_n, b)}$,

  where u_n left-singular vector of A of smallest singular value.

Example in MATLAB:

```
>> V = vander(randn(20,1)); b=randn(20,1);
>> cond(V)= 7.1021e+11
>> x = V\b;
>> norm(inv(V))*norm(b)/norm(x) = 8.4317
```

$(\|A^{-1}\|_2\|b\|_2 \,/\, \|x\|_2)$ is "almost always" a moderate number (Chan & Foulser, 1988)

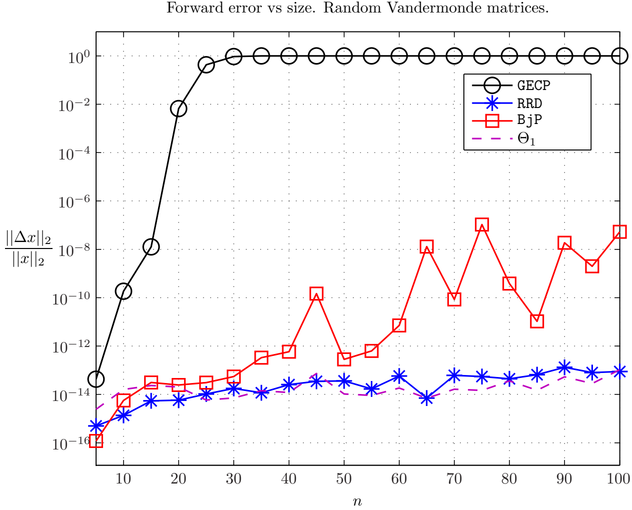
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \kappa_2(A)$, but even more **"almost always"**
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \ll \kappa_2(A)$, if A very ill-conditioned because,
- $\dfrac{\|A^{-1}\|_2\|b\|_2}{\|x\|_2} \leq \dfrac{1}{\cos\theta(u_n, b)}$,

  where u_n left-singular vector of A of smallest singular value.

Example in MATLAB:

```
>> V = vander(randn(20,1)); b=randn(20,1);
>> cond(V)= 7.1021e+11
>> x = V\b;
>> norm(inv(V))*norm(b)/norm(x) = 8.4317
```

Numerical tests for linear systems: Random Vandermonde Matrices. (computing RRD as Demmel, SIMAX, 1999)



Outline

The algorithm for Least Squares Problems

Algorithm (Castro, Ceballos, D., Molera, to be submitted, 2012)

- Input: $A \in \mathbb{C}^{m \times n}$, $b \in \mathbb{C}^m$
- Output: x_0 minimum 2-norm solution of $\min_{x \in \mathbb{C}^n} \|b - A\,x\|_2$

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following steps
   1. Compute the solution x_1 of $\min_{x \in \mathbb{C}^r} \|b - X\,x\|_2$ using QR Householder.
   2. Solve the diagonal linear system, $D\,x_2 = x_1$.
   3. Compute the minimum 2-norm solution x_0 of the underdetermined linear system $Yx = x_2$ using QR Householder on Y^*.

- $x_2(i) = x_1(i)/d_{ii}, \quad i = 1 : r.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(m\,n^2)$ flops.

The algorithm for Least Squares Problems

Algorithm (Castro, Ceballos, D., Molera, to be submitted, 2012)

- Input: $A \in \mathbb{C}^{m \times n}$, $b \in \mathbb{C}^m$
- Output: x_0 minimum 2-norm solution of $\min_{x \in \mathbb{C}^n} \|b - A\,x\|_2$

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following steps
   1. Compute the solution x_1 of $\min_{x \in \mathbb{C}^r} \|b - X\,x\|_2$ using QR Householder.
   2. Solve the diagonal linear system, $D\,x_2 = x_1$.
   3. Compute the minimum 2-norm solution x_0 of the underdetermined linear system $Y x = x_2$ using QR Householder on Y^*.

- $x_2(i) = x_1(i)/d_{ii}$, $\quad i = 1 : r$.
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(m\,n^2)$ flops.

The algorithm for Least Squares Problems

Algorithm (Castro, Ceballos, D., Molera, to be submitted, 2012)

- Input: $A \in \mathbb{C}^{m \times n}$, $b \in \mathbb{C}^m$
- Output: x_0 minimum 2-norm solution of $\min_{x \in \mathbb{C}^n} \|b - A\,x\|_2$

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following steps
   1. Compute the solution x_1 of $\min_{x \in \mathbb{C}^r} \|b - X\,x\|_2$ using QR Householder.
   2. Solve the diagonal linear system, $D\,x_2 = x_1$.
   3. Compute the minimum 2-norm solution x_0 of the underdetermined linear system $Yx = x_2$ using QR Householder on Y^*.

- $x_2(i) = x_1(i)/d_{ii}, \quad i = 1 : r.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(m\,n^2)$ flops.

The algorithm for Least Squares Problems

Algorithm (Castro, Ceballos, D., Molera, to be submitted, 2012)

- Input: $A \in \mathbb{C}^{m \times n}$, $b \in \mathbb{C}^m$
- Output: x_0 minimum 2-norm solution of $\min_{x \in \mathbb{C}^n} \|b - A\,x\|_2$

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following steps
    1. Compute the solution x_1 of $\min_{x \in \mathbb{C}^r} \|b - X\,x\|_2$ using QR Householder.
    2. Solve the diagonal linear system, $D\,x_2 = x_1$.
    3. Compute the minimum 2-norm solution x_0 of the underdetermined linear system $Yx = x_2$ using QR Householder on Y^*.

- $x_2(i) = x_1(i)/d_{ii}, \quad i = 1 : r.$
- **Intuition:** the ill-conditioned linear system is solved very accurately.
- **Cost:** $O(m\,n^2)$ flops.

Theorem (Castro, Ceballos, D., Molera, to be submitted, 2012)

If $\widehat{x}_0$ is the minimum 2-norm solution of $\min_{x\in\mathbb{C}^n} \|b - A\,x\|_2$ computed by the algorithm in previous slide ($A = X D Y$) and x_0 is the exact one, then

$$\frac{\|\widehat{x}_0 - x_0\|_2}{\|x_0\|_2} \leq O(\mathtt{u})\ \max\{\kappa_2(Y), \kappa_2(X)\}\ \frac{\|A^\dagger\|_2\,\|b\|_2}{\|x_0\|_2}\,,$$

$A^\dagger$ is the Moore-Penrose pseudo-inverse of A and $\kappa_2(Y) = \|Y\|_2\|Y^\dagger\|_2$.

To be compared with

$$\frac{\|\widehat{x}_0 - x_0\|_2}{\|x_0\|_2} \leq O(\mathtt{u})\left(\kappa_2(A) + \frac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2\,\frac{\|b - Ax_0\|_2}{\|A\|_2\,\|x_0\|_2}\right),$$

that holds for traditional algorithms as Householder-QR, SVD,...

The error for Least Squares Problems

Theorem (Castro, Ceballos, D., Molera, to be submitted, 2012)

If $\widehat{x}_0$ is the minimum 2-norm solution of $\min_{x\in\mathbb{C}^n} \|b - A\,x\|_2$ computed by the algorithm in previous slide ($A = X\,D\,Y$) and x_0 is the exact one, then

$$\frac{\|\widehat{x}_0 - x_0\|_2}{\|x_0\|_2} \leq O(\mathtt{u})\ \max\{\kappa_2(Y), \kappa_2(X)\}\ \frac{\|A^\dagger\|_2\ \|b\|_2}{\|x_0\|_2},$$

$A^\dagger$ is the Moore-Penrose pseudo-inverse of A and $\kappa_2(Y) = \|Y\|_2 \|Y^\dagger\|_2$.

To be compared with

$$\frac{\|\widehat{x}_0 - x_0\|_2}{\|x_0\|_2} \leq O(\mathtt{u}) \left(\kappa_2(A) + \frac{\|A^\dagger\|_2 \|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2 \frac{\|b - Ax_0\|_2}{\|A\|_2\ \|x_0\|_2} \right),$$

that holds for traditional algorithms as Householder-QR, SVD,...

Define

$$\kappa_{LS}(A,b) := \left(\kappa_2(A) + \frac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2\,\frac{\|b - Ax_0\|_2}{\|A\|_2\,\|x_0\|_2}\right),$$

- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \kappa_{LS}(A,b)$, but even more **"almost always"**
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \ll \kappa_{LS}(A,b)$, if A very ill-conditioned because,
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \dfrac{1}{\cos\theta(u_r,b)}$,

where u_r left-singular vector of A of smallest nonzero singular value.

Define

$$\kappa_{LS}(A,b) := \left(\kappa_2(A) + \frac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2\,\frac{\|b - Ax_0\|_2}{\|A\|_2\,\|x_0\|_2}\right),$$

- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \kappa_{LS}(A,b),$ but even more **"almost always"**
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \ll \kappa_{LS}(A,b)$, if A very ill-conditioned because,
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \dfrac{1}{\cos\theta(u_r,b)}$,

where u_r left-singular vector of A of smallest nonzero singular value.

$\left(\|A^\dagger\|_2\|b\|_2 \,/\, \|x_0\|_2\right)$ is "almost always" a moderate number

Define

$$\kappa_{LS}(A,b) := \left(\kappa_2(A) + \frac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2\,\frac{\|b - Ax_0\|_2}{\|A\|_2\,\|x_0\|_2}\right),$$

- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \kappa_{LS}(A,b)$, but even more **"almost always"**
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \ll \kappa_{LS}(A,b)$, if A very ill-conditioned because,
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \dfrac{1}{\cos\theta(u_r,b)}$,

where u_r left-singular vector of A of smallest nonzero singular value.

$\left(\|A^\dagger\|_2\|b\|_2 \,/\, \|x_0\|_2\right)$ is "almost always" a moderate number

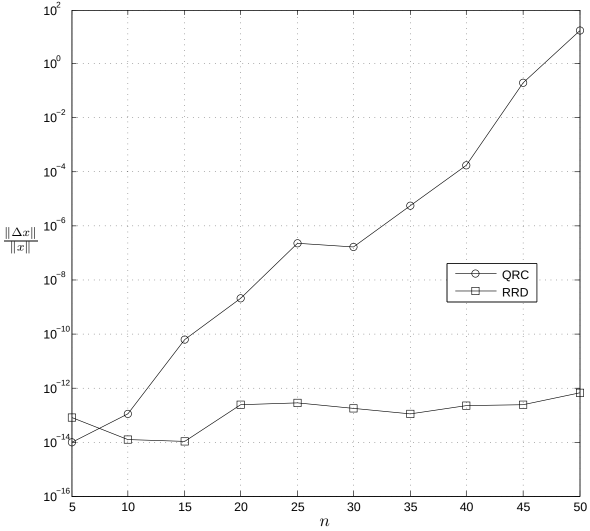
Define

$$\kappa_{LS}(A,b) := \left(\kappa_2(A) + \frac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} + \kappa_2(A)^2\,\frac{\|b - Ax_0\|_2}{\|A\|_2\,\|x_0\|_2}\right),$$

- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \kappa_{LS}(A,b)$, but even more **"almost always"**
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \ll \kappa_{LS}(A,b)$, if A very ill-conditioned because,
- $\dfrac{\|A^\dagger\|_2\|b\|_2}{\|x_0\|_2} \leq \dfrac{1}{\cos\theta(u_r,b)}$,

  where u_r left-singular vector of A of smallest nonzero singular value.

Numerical tests for Least Squares: $50 \times n$ graded matrices $S_1 B S_2$, with $\kappa_2(B) = 10$ and $\kappa_2(S_1) = \kappa_2(S_2) = 10^{(2:2:16)}$ (comp. RRD as Higham, 2000)



Outline

1. Accurate rank revealing decompositions (RRDs)
2. Accurate solution of linear systems
3. Accurate solution of least squares problems
4. **Accurate eigenvalues/vectors of symmetric matrices**
5. Accurate SVD
6. Conclusions and open problems

Algorithm (IMPLICIT JACOBI. (D., Koev, Molera, Numer. Math., 2009))

- Input: $A = A^T \in \mathbb{R}^{n \times n}$
- Output: e-values, λ_i, and matrix of e-vectors, U, of A

1. Compute an accurate symmetric RRD of $A = XDX^T$
2. Apply **implicit Jacobi to** XDX^T, i.e.,

```
U = I_n
repeat
    for i < j
        compute a_ii, a_ij, a_jj of A = XDX^T
        compute Jacobi Rotation R s.t.  a_ij = 0 by similarity
        X = R* X
        U = U R
    endfor
until convergence ( |a_ij| / sqrt(|a_ii a_jj|) <= tol = O(u)  for all i < j )
compute λ_k = a_kk for k = 1 : n.
```

$U = I_n$

repeat

for $i < j$

compute a_{ii}, a_{ij}, a_{jj} of $A = XDX^T$

compute Jacobi Rotation R s.t. $a_{ij} = 0$ by similarity

$X = R^* X$

$U = U R$

endfor

until convergence $\left(\frac{|a_{ij}|}{\sqrt{|a_{ii} a_{jj}|}} \leq \mathtt{tol} = O(\mathtt{u}) \quad \text{for all } i < j \right)$

compute $\lambda_k = a_{kk}$ for $k = 1 : n$.

Comments on the IMPLICIT JACOBI algorithm

- The ill-conditioned matrix D **is never modified**.
- **Only the well-conditioned factor X is transformed** in the process.
- This is the reason why high relative accuracy is obtained.
- **Cost**: $O(n^3)$ flops.
- **Efficient implementation requires preconditioning** via QR factorization with column pivoting of $X\sqrt{|D|}$ (this was suggested by Drmač).

Comments on the IMPLICIT JACOBI algorithm

- The ill-conditioned matrix D **is never modified**.
- **Only the well-conditioned factor** X **is transformed** in the process.
- This is the reason why high relative accuracy is obtained.
- **Cost**: $O(n^3)$ flops.
- **Efficient implementation requires preconditioning** via QR factorization with column pivoting of $X\sqrt{|D|}$ (this was suggested by Drmač).

Comments on the IMPLICIT JACOBI algorithm

- The ill-conditioned matrix D **is never modified**.
- **Only the well-conditioned factor X is transformed** in the process.
- This is the reason why high relative accuracy is obtained.
- **Cost**: $O(n^3)$ flops.
- **Efficient implementation requires preconditioning** via QR factorization with column pivoting of $X\sqrt{|D|}$ (this was suggested by Drmač).

Comments on the IMPLICIT JACOBI algorithm

- The ill-conditioned matrix D **is never modified**.
- **Only the well-conditioned factor X is transformed** in the process.
- This is the reason why high relative accuracy is obtained.
- **Cost**: $O(n^3)$ flops.
- **Efficient implementation requires preconditioning** via QR factorization with column pivoting of $X\sqrt{|D|}$ (this was suggested by Drmač).

Comments on the IMPLICIT JACOBI algorithm

- The ill-conditioned matrix D **is never modified**.
- **Only the well-conditioned factor X is transformed** in the process.
- This is the reason why high relative accuracy is obtained.
- **Cost**: $O(n^3)$ flops.
- **Efficient implementation requires preconditioning** via QR factorization with column pivoting of $X\sqrt{|D|}$ (this was suggested by Drmač).

Theorem (D., Koev, Molera, Numer. Math., 2009)

Let $\widehat{\lambda}_i$ be the eigenvalues of $A = XDX^T$ computed by the Implicit Jacobi Algorithm and λ_i the exact ones. Let $\widehat{v}_i$ and v_i be the corresponding eigenvectors. Then

$$\frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq O(\mathtt{u})\,\kappa_2(X) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathtt{u})\,\kappa(X)}{\min_{j\neq i} \left|\frac{\lambda_i - \lambda_j}{\lambda_i}\right|} \quad \textit{for all} \quad i,$$

To be compared with

$$\frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq O(\mathtt{u})\,\kappa_2(A) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathtt{u})}{\frac{\min_{j\neq i}|\lambda_i - \lambda_j|}{\max_i |\lambda_i|}} \quad \text{for all} \quad i,$$

that hold for traditional algorithms as QR, divide and conquer,...

Theorem (D., Koev, Molera, Numer. Math., 2009)

Let $\widehat{\lambda}_i$ be the eigenvalues of $A = XDX^T$ computed by the Implicit Jacobi Algorithm and λ_i the exact ones. Let $\widehat{v}_i$ and v_i be the corresponding eigenvectors. Then

$$\frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq O(\mathtt{u})\, \kappa_2(X) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathtt{u})\, \kappa(X)}{\min_{j \neq i} \left| \frac{\lambda_i - \lambda_j}{\lambda_i} \right|} \quad \textit{for all} \quad i,$$

To be compared with

$$\frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} \leq O(\mathtt{u})\, \kappa_2(A) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathtt{u})}{\frac{\min_{j \neq i} |\lambda_i - \lambda_j|}{\max_i |\lambda_i|}} \quad \text{for all} \quad i,$$

that hold for traditional algorithms as QR, divide and conquer,...

EXAMPLE: Symmetric INDEFINITE 100×100 Cauchy matrix A

$$a_{ij} = \frac{1}{x_i + x_j}, \quad \text{with} \quad \begin{cases} x_i = i - 0.5 \quad for \quad i = 1 : 99 \\ x_{100} = -99.5 \end{cases}$$

- $\kappa(A) = 3.5 \cdot 10^{147}$
- **Errors in RRR + Imp. Jacobi** compared to 200-decimal digits MATLAB's `eig` command

$$\max_i \frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 1.2 \cdot 10^{-13} \quad \text{and} \quad \max_i \|\widehat{v}_i - v_i\|_2 = 5.7 \cdot 10^{-14}$$

- **Errors in MATLAB's `eig` function**

$$\max_i \frac{|\widehat{\lambda}_i - \lambda_i|}{|\lambda_i|} = 1.84 \cdot 10^{132} \quad \text{and} \quad \max_i \|\widehat{v}_i - v_i\|_2 = 1.41$$

Outline

Algorithm (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

- Input: $A \in \mathbb{C}^{m \times n}$
- Output: $U\Sigma V^*$ SVD of A

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following algorithm
   1. QR with column pivoting of $XD = QRP$ (so $A = QRPY$)
   2. Multiply to get $W = RPY$ (so $A = QW$)
   3. Compute SVD of $W = \bar{U}\Sigma V^*$ with **one-sided Jacobi** (so $A = Q\bar{U}\Sigma V^*$)
   4. Multiply $U = Q\bar{U}$ (so $A = U\Sigma V^*$)

- The one-sided Jacobi step can be performed with new fast and accurate Jacobi algorithm by Drmač and Veselić, SIMAX, 2008.
- **Cost:** $O(mn^2)$ flops.

The algorithm for SVD

Algorithm (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

- `Input:` $A \in \mathbb{C}^{m\times n}$
- `Output:` $U\Sigma V^*$ `SVD of` A

1. `Compute an accurate RRD of` $A = XDY$
2. `Apply to` XDY `the following algorithm`
    1. `QR with column pivoting of` $XD = QRP$ `(so` $A = QRPY$`)`
    2. `Multiply to get` $W = RPY$ `(so` $A = QW$`)`
    3. `Compute SVD of` $W = \bar{U}\Sigma V^*$ `with` **one-sided Jacobi** `(so` $A = Q\bar{U}\Sigma V^*$`)`
    4. `Multiply` $U = Q\bar{U}$ `(so` $A = U\Sigma V^*$`)`

- The one-sided Jacobi step can be performed with new fast and accurate Jacobi algorithm by Drmač and Veselić, SIMAX, 2008.
- **Cost:** $O(mn^2)$ flops.

The algorithm for SVD

Algorithm (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

- Input: $A \in \mathbb{C}^{m\times n}$
- Output: $U\Sigma V^*$ SVD of A

1. Compute an accurate RRD of $A = XDY$
2. Apply to XDY the following algorithm
   1. QR with column pivoting of $XD = QRP$ (so $A = QRPY$)
   2. Multiply to get $W = RPY$ (so $A = QW$)
   3. Compute SVD of $W = \bar{U}\Sigma V^*$ with **one-sided Jacobi** (so $A = Q\bar{U}\Sigma V^*$)
   4. Multiply $U = Q\bar{U}$ (so $A = U\Sigma V^*$)

- The one-sided Jacobi step can be performed with new fast and accurate Jacobi algorithm by Drmač and Veselić, SIMAX, 2008.
- **Cost:** $O(mn^2)$ flops.

Theorem (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

Let $\widehat{\sigma}_i$ be the singular values of $A = XDY$ computed by the algorithm in previous slide and σ_i the exact ones. Let $\widehat{u}_i$ and u_i be the corresponding left singular vectors and $\widehat{v}_i$ and v_i the right ones. Then

$$\frac{|\widehat{\sigma}_i - \sigma_i|}{|\sigma_i|} \leq O(\mathrm{u}) \max\{\kappa_2(X), \kappa_2(Y)\} \quad \text{and}$$

$$\theta(v_i, \widehat{v}_i) \leq \frac{O(\mathrm{u}) \max\{\kappa_2(X), \kappa_2(Y)\}}{\min_{j \neq i} \left| \frac{\sigma_i - \sigma_j}{\sigma_i} \right|} \quad \textit{for all} \quad i,$$

To be compared with

$$\frac{|\widehat{\sigma}_i - \sigma_i|}{|\sigma_i|} \leq O(\mathrm{u})\, \kappa_2(A) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathrm{u})}{\frac{\min_{j \neq i} |\sigma_i - \sigma_j|}{\max_i |\sigma_i|}} \quad \text{for all} \quad i,$$

that hold for traditional algorithms as QR, divide and conquer,...

The errors in accurate SVD

Theorem (Demmel, Gu, Eisenstat, Slapničar, Veselić, Drmač, LAA, 99)

Let $\widehat{\sigma}_i$ be the singular values of $A = XDY$ computed by the algorithm in previous slide and σ_i the exact ones. Let $\widehat{u}_i$ and u_i be the corresponding left singular vectors and $\widehat{v}_i$ and v_i the right ones. Then

$$\frac{|\widehat{\sigma}_i - \sigma_i|}{|\sigma_i|} \leq O(\mathbf{u}) \max\{\kappa_2(X), \kappa_2(Y)\} \quad \text{and}$$

$$\theta(v_i, \widehat{v}_i) \leq \frac{O(\mathbf{u}) \max\{\kappa_2(X), \kappa_2(Y)\}}{\min_{j \neq i} \left| \frac{\sigma_i - \sigma_j}{\sigma_i} \right|} \quad \textit{for all} \quad i,$$

To be compared with

$$\frac{|\widehat{\sigma}_i - \sigma_i|}{|\sigma_i|} \leq O(\mathbf{u})\, \kappa_2(A) \quad \text{and} \quad \theta(v_i, \widehat{v}_i) \leq \frac{O(\mathbf{u})}{\frac{\min_{j \neq i} |\sigma_i - \sigma_j|}{\max_i |\sigma_i|}} \quad \text{for all} \quad i,$$

that hold for traditional algorithms as QR, divide and conquer,...

Outline

Conclusions and open problems

- Rank-Revealing decompositions (RRDs) may be computed with high accuracy for many classes of structured matrices.
- This allows us to perform with high accuracy almost all basic tasks of Numerical Linear Algebra for these structured matrices.
- The only basic task that is not included in this framework is the nonsymmetric eigenvalue problem.
- This problem will be the subject of future work.

- Rank-Revealing decompositions (RRDs) may be computed with high accuracy for many classes of structured matrices.
- This allows us to perform with high accuracy almost all basic tasks of Numerical Linear Algebra for these structured matrices.
- The only basic task that is not included in this framework is the nonsymmetric eigenvalue problem.
- This problem will be the subject of future work.

Conclusions and open problems

- Rank-Revealing decompositions (RRDs) may be computed with high accuracy for many classes of structured matrices.
- This allows us to perform with high accuracy almost all basic tasks of Numerical Linear Algebra for these structured matrices.
- The only basic task that is not included in this framework is the nonsymmetric eigenvalue problem.
- This problem will be the subject of future work.

Conclusions and open problems

- Rank-Revealing decompositions (RRDs) may be computed with high accuracy for many classes of structured matrices.
- This allows us to perform with high accuracy almost all basic tasks of Numerical Linear Algebra for these structured matrices.
- The only basic task that is not included in this framework is the nonsymmetric eigenvalue problem.
- This problem will be the subject of future work.