Perturbation theory of LDU factorization and accurate computations for diagonally dominant matrices

Froilán M. Dopico[1]

Plamen Koev[2]

[1] Departamento de Matemáticas, Universidad Carlos III de Madrid, Spain

[2] Department of Mathematics, San José State University, California, USA

2º Encuentro de Álgebra Lineal, Análisis Matricial y Aplicaciones, ALAMA2010, Valencia, June 2-4, 2010

Outline

Outline

1. Introduction
2. Perturbation theory for LDU factorization
3. Error analysis
4. Conclusions

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n \times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L - \widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U - \widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii} - \widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \ \ \forall i,$$
  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n\times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
-
$$\frac{\|L - \widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U - \widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii} - \widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$
  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n \times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L - \widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U - \widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii} - \widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n \times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L - \widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U - \widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii} - \widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n\times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L-\widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U-\widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n \times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L - \widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U - \widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii} - \widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU Rank Revealing Decompositions (RRD)

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n\times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L-\widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U-\widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Let $A = LDU$ and $\widehat{L}\widehat{D}\widehat{U}$ be, respectively, the exact and computed LDU factorizations of a matrix $A \in \mathbb{R}^{n\times n}$.

If these factorizations satisfy

- L and U are well-conditioned (this happens if complete pivoting is used),
- $$\frac{\|L-\widehat{L}\|}{\|L\|} = O(\epsilon), \quad \frac{\|U-\widehat{U}\|}{\|U\|} = O(\epsilon), \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} = O(\epsilon) \;\; \forall i,$$

  where ϵ is machine precision (this can be guaranteed only for some types of matrices through special implementations of GECP),

then there are algorithms that use the factors $\widehat{L}$, $\widehat{D}$, $\widehat{U}$ for

- **computing the SVD of A very accurately** (Demmel et al. 1999), and
- **computing very accurately the solution of $Ax = b$ for almost every b** (D-Molera this conference).

independently of the traditional condition number of A.

Accurate LDU RRDs of diagonally dominant matrices

- Q. Ye, Math. Comp. (2008), developed a very ingenuous algorithm for computing accurately in $2n^3$ flops the LDU factorization with complete pivoting of row diagonally dominant matrices...
- that are parameterized in a particular way, but
- best error bounds that have been proved after considerable efforts are

$$\frac{\|L-\widehat{L}\|_\infty}{\|L\|_\infty} \leq 6\, n\, 8^{(n-1)}\epsilon, \quad \frac{\|U-\widehat{U}\|_\infty}{\|U\|_\infty} \leq 6 \cdot 8^{(n-1)}\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 5 \cdot 8^{(n-1)}\epsilon,$$

  where $n \times n$ is the size of the matrix.
- $\epsilon = 2^{-53}$ in double precision, **so the bounds are useless for $n > 20$**...
- However, numerical experiments indicate accuracy....

Accurate LDU RRDs of diagonally dominant matrices

- Q. Ye, Math. Comp. (2008), developed a very ingenuous algorithm for computing accurately in $2n^3$ flops the LDU factorization with complete pivoting of row diagonally dominant matrices...
- that are parameterized in a particular way, but
- best error bounds that have been proved after considerable efforts are

$$\frac{\|L-\widehat{L}\|_\infty}{\|L\|_\infty} \leq 6\,n\,8^{(n-1)}\epsilon, \quad \frac{\|U-\widehat{U}\|_\infty}{\|U\|_\infty} \leq 6\cdot 8^{(n-1)}\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 5\cdot 8^{(n-1)}\epsilon,$$

where $n \times n$ is the size of the matrix.

- $\epsilon = 2^{-53}$ in double precision, **so the bounds are useless for $n > 20$**...
- However, numerical experiments indicate accuracy....

Accurate LDU RRDs of diagonally dominant matrices

- Q. Ye, Math. Comp. (2008), developed a very ingenuous algorithm for computing accurately in $2n^3$ flops the LDU factorization with complete pivoting of row diagonally dominant matrices...
- that are parameterized in a particular way, but
- best error bounds that have been proved after considerable efforts are

$$\frac{\|L-\widehat{L}\|_\infty}{\|L\|_\infty} \leq 6\, n\, 8^{(n-1)}\epsilon, \quad \frac{\|U-\widehat{U}\|_\infty}{\|U\|_\infty} \leq 6 \cdot 8^{(n-1)}\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 5 \cdot 8^{(n-1)}\epsilon,$$

where $n \times n$ is the size of the matrix.

- $\epsilon = 2^{-53}$ in double precision, **so the bounds are useless for** $n > 20$...
- However, numerical experiments indicate accuracy....

Accurate LDU RRDs of diagonally dominant matrices

- Q. Ye, Math. Comp. (2008), developed a very ingenuous algorithm for computing accurately in $2n^3$ flops the LDU factorization with complete pivoting of row diagonally dominant matrices...
- that are parameterized in a particular way, but
- best error bounds that have been proved after considerable efforts are

$$\frac{\|L-\widehat{L}\|_\infty}{\|L\|_\infty} \leq 6\,n\,8^{(n-1)}\epsilon, \quad \frac{\|U-\widehat{U}\|_\infty}{\|U\|_\infty} \leq 6\cdot 8^{(n-1)}\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 5\cdot 8^{(n-1)}\epsilon,$$

  where $n \times n$ is the size of the matrix.
- $\epsilon = 2^{-53}$ in double precision, **so the bounds are useless for $n > 20$**...
- However, numerical experiments indicate accuracy....

Accurate LDU RRDs of diagonally dominant matrices

- Q. Ye, Math. Comp. (2008), developed a very ingenuous algorithm for computing accurately in $2n^3$ flops the LDU factorization with complete pivoting of row diagonally dominant matrices...
- that are parameterized in a particular way, but
- best error bounds that have been proved after considerable efforts are

$$\frac{\|L-\widehat{L}\|_\infty}{\|L\|_\infty} \leq 6\,n\,8^{(n-1)}\epsilon, \quad \frac{\|U-\widehat{U}\|_\infty}{\|U\|_\infty} \leq 6\cdot 8^{(n-1)}\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 5\cdot 8^{(n-1)}\epsilon,$$

  where $n\times n$ is the size of the matrix.
- $\epsilon = 2^{-53}$ in double precision, **so the bounds are useless for $n > 20$**...
- However, numerical experiments indicate accuracy....

Goal of the talk

- **Is Q. Ye's algorithm accurate?**
- **I will prove satisfactory error bounds for Q. Ye's algorithm by using a new Perturbation Theory for the LDU of diagonally dominant matrices.**

$$\frac{\|L-\widehat{L}\|}{\|L\|} \leq 14\,n^3\epsilon, \quad \frac{\|U-\widehat{U}\|}{\|U\|} \leq 14\,n^3\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 14\,n^3\epsilon \;\; \forall i$$

- **Fundamental consequence: We can compute very accurately the solution of linear systems and SVDs for diagonally dominant matrices in $O(n^3)$ flops for arbitrarily ill-conditioned matrices.**
- **Diagonally dominant matrices appear in many applications.**

Goal of the talk

- **Is Q. Ye's algorithm accurate?**
- **I will prove satisfactory error bounds for Q. Ye's algorithm by using a new Perturbation Theory for the LDU of diagonally dominant matrices.**

$$\frac{\|L-\widehat{L}\|}{\|L\|} \leq 14\,n^3\epsilon, \quad \frac{\|U-\widehat{U}\|}{\|U\|} \leq 14\,n^3\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 14\,n^3\epsilon \;\; \forall i$$

- **Fundamental consequence: We can compute very accurately the solution of linear systems and SVDs for diagonally dominant matrices in $O(n^3)$ flops for arbitrarily ill-conditioned matrices.**
- **Diagonally dominant matrices appear in many applications.**

Goal of the talk

- **Is Q. Ye's algorithm accurate?**
- **I will prove satisfactory error bounds for Q. Ye's algorithm by using a new Perturbation Theory for the LDU of diagonally dominant matrices.**

$$\frac{\|L-\widehat{L}\|}{\|L\|} \leq 14\, n^3 \epsilon, \quad \frac{\|U-\widehat{U}\|}{\|U\|} \leq 14\, n^3 \epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 14\, n^3 \epsilon \ \ \forall i$$

- **Fundamental consequence: We can compute very accurately the solution of linear systems and SVDs for diagonally dominant matrices in $O(n^3)$ flops for arbitrarily ill-conditioned matrices.**
- **Diagonally dominant matrices appear in many applications.**

Goal of the talk

- **Is Q. Ye's algorithm accurate?**
- **I will prove satisfactory error bounds for Q. Ye's algorithm by using a new Perturbation Theory for the LDU of diagonally dominant matrices.**

$$\frac{\|L-\widehat{L}\|}{\|L\|} \leq 14\,n^3\epsilon, \quad \frac{\|U-\widehat{U}\|}{\|U\|} \leq 14\,n^3\epsilon, \quad \frac{|d_{ii}-\widehat{d}_{ii}|}{|d_{ii}|} \leq 14\,n^3\epsilon \;\; \forall i$$

- **Fundamental consequence: We can compute very accurately the solution of linear systems and SVDs for diagonally dominant matrices in $O(n^3)$ flops for arbitrarily ill-conditioned matrices.**
- **Diagonally dominant matrices appear in many applications.**

Parameterizing row diagonally dominant matrices (Q. Ye)

- Assume from now on that $A \in \mathbb{R}^{n\times n}$ satisfies $a_{ii} \geq 0$ for all i (no restriction for SVD or linear systems).
- Define $v = (v_1, v_2, \ldots, v_n)$ where

$$v_i := a_{ii} - \sum_{j \neq i} |a_{ij}|$$

- A is row diagonally dominant if and only if $v_i \geq 0$ for all i.
- $A_D := \begin{cases} 0 & \text{for } i = j \\ a_{ij} & \text{for } i \neq j \end{cases}$
- The pair (A_D, v) allows us to recover the matrix A and we parameterize the set of $n \times n$ matrices through pairs of this type. A matrix A parameterized is this way will be denoted as

$$A = \mathcal{D}(A_D, v)$$

Parameterizing row diagonally dominant matrices (Q. Ye)

- Assume from now on that $A \in \mathbb{R}^{n \times n}$ satisfies $a_{ii} \geq 0$ for all i (no restriction for SVD or linear systems).
- Define $v = (v_1, v_2, \ldots, v_n)$ where

$$v_i := a_{ii} - \sum_{j \neq i} |a_{ij}|$$

- A is row diagonally dominant if and only if $v_i \geq 0$ for all i.
- $A_D := \begin{cases} 0 & \text{for } i = j \\ a_{ij} & \text{for } i \neq j \end{cases}$
- The pair (A_D, v) allows us to recover the matrix A and we parameterize the set of $n \times n$ matrices through pairs of this type. A matrix A parameterized is this way will be denoted as

$$A = \mathcal{D}(A_D, v)$$

Parameterizing row diagonally dominant matrices (Q. Ye)

- Assume from now on that $A \in \mathbb{R}^{n\times n}$ satisfies $a_{ii} \geq 0$ for all i (no restriction for SVD or linear systems).
- Define $v = (v_1, v_2, \ldots, v_n)$ where

$$v_i := a_{ii} - \sum_{j\neq i} |a_{ij}|$$

- A is row diagonally dominant if and only if $v_i \geq 0$ for all i.
- $A_D := \begin{cases} 0 & \text{for } i = j \\ a_{ij} & \text{for } i \neq j \end{cases}$
- The pair (A_D, v) allows us to recover the matrix A and we parameterize the set of $n \times n$ matrices through pairs of this type. A matrix A parameterized is this way will be denoted as

$$A = \mathcal{D}(A_D, v)$$

Parameterizing row diagonally dominant matrices (Q. Ye)

- Assume from now on that $A \in \mathbb{R}^{n\times n}$ satisfies $a_{ii} \geq 0$ for all i (no restriction for SVD or linear systems).
- Define $v = (v_1, v_2, \ldots, v_n)$ where

$$v_i := a_{ii} - \sum_{j\neq i} |a_{ij}|$$

- A is row diagonally dominant if and only if $v_i \geq 0$ for all i.
- $A_D := \begin{cases} 0 & \text{for } i = j \\ a_{ij} & \text{for } i \neq j \end{cases}$
- The pair (A_D, v) allows us to recover the matrix A and we parameterize the set of $n \times n$ matrices through pairs of this type. A matrix A parameterized is this way will be denoted as

$$A = \mathcal{D}(A_D, v)$$

Parameterizing row diagonally dominant matrices (Q. Ye)

- Assume from now on that $A \in \mathbb{R}^{n\times n}$ satisfies $a_{ii} \geq 0$ for all i (no restriction for SVD or linear systems).
- Define $v = (v_1, v_2, \ldots, v_n)$ where

$$v_i := a_{ii} - \sum_{j\neq i} |a_{ij}|$$

- A is row diagonally dominant if and only if $v_i \geq 0$ for all i.
- $A_D := \begin{cases} 0 & \text{for } i = j \\ a_{ij} & \text{for } i \neq j \end{cases}$
- The pair (A_D, v) allows us to recover the matrix A and we parameterize the set of $n \times n$ matrices through pairs of this type. A matrix A parameterized is this way will be denoted as

$$A = \mathcal{D}(A_D, v)$$

Example: Two types of small ($\approx 10^{-3}$) relative componentwise perturbations of a row diagonally dominant matrix A:

$$A = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 0 \\ 0.002 \\ 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.001 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(B) = \begin{bmatrix} 0 \\ 0.001 \\ 0 \end{bmatrix}$$

$$v(C) = \begin{bmatrix} 0 \\ 0.002002 \\ 0 \end{bmatrix}, \quad c_{12} = -1.5015 \Longrightarrow C = \begin{bmatrix} 3.0015 & -1.5015 & 1.5 \\ -1 & 2.002002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}$$

Singular values of A, B and C (no theorem)

	A	B	C
σ_1	4.641	4.640	4.642
σ_2	2.910	2.909	2.910
σ_3	$6.663 \cdot 10^{-4}$	$3.332 \cdot 10^{-4}$	$6.673 \cdot 10^{-4}$

Good perturbation properties of this parametrization

Example: Two types of small ($\approx 10^{-3}$) relative componentwise perturbations of a row diagonally dominant matrix A:

$$A = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 0 \\ 0.002 \\ 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.001 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(B) = \begin{bmatrix} 0 \\ 0.001 \\ 0 \end{bmatrix}$$

$$v(C) = \begin{bmatrix} 0 \\ 0.002002 \\ 0 \end{bmatrix}, \quad c_{12} = -1.5015 \Longrightarrow C = \begin{bmatrix} 3.0015 & -1.5015 & 1.5 \\ -1 & 2.002002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}$$

Singular values of A, B and C (no theorem)

	A	B	C
σ_1	4.641	4.640	4.642
σ_2	2.910	2.909	2.910
σ_3	$6.663 \cdot 10^{-4}$	$3.332 \cdot 10^{-4}$	$6.673 \cdot 10^{-4}$

Good perturbation properties of this parametrization

Example: Two types of small ($\approx 10^{-3}$) relative componentwise perturbations of a row diagonally dominant matrix A:

$$A = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 0 \\ 0.002 \\ 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 3 & -1.5 & 1.5 \\ -1 & 2.001 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}, \qquad v(B) = \begin{bmatrix} 0 \\ 0.001 \\ 0 \end{bmatrix}$$

$$v(C) = \begin{bmatrix} 0 \\ 0.002002 \\ 0 \end{bmatrix}, \quad c_{12} = -1.5015 \Longrightarrow C = \begin{bmatrix} 3.0015 & -1.5015 & 1.5 \\ -1 & 2.002002 & 1 \\ 2 & 0.5 & 2.5 \end{bmatrix}$$

Singular values of A, B and C (no theorem)

	A	B	C
σ_1	4.641	4.640	4.642
σ_2	2.910	2.909	2.910
σ_3	$6.663 \cdot 10^{-4}$	$3.332 \cdot 10^{-4}$	$6.673 \cdot 10^{-4}$

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from** $\mathcal{D}(A_D^{(k)}, v^{(k)})$ **as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

Key features of Q. Ye's algorithm for LDU of diag. dominant

- INPUT: $\mathcal{D}(A_D, v)$ with $v \geq 0$ *(not the matrix A)*.
- It performs Gaussian elimination with complete (diagonal) pivoting.
- If we denote $A^{(1)} := A$ and $A^{(k)}$ is the matrix obtained after $k-1$ steps of Gaussian elimination are performed, then the algorithm iterates

$$\mathcal{D}(A_D^{(1)}, v^{(1)}) \to \mathcal{D}(A_D^{(2)}, v^{(2)}) \to \cdots \to \mathcal{D}(A_D^{(k)}, v^{(k)}) \to \cdots$$

- $v^{(k+1)}$ **is obtained from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ as a sum of nonnegative terms. There are no cancellation errors in this part!!**
- $A_D^{(k+1)}$ is computed from $\mathcal{D}(A_D^{(k)}, v^{(k)})$ by applying the usual Gaussian elimination process. So cancellation errors may appear but they are bounded in an absolute sense.

What happens if the vector v in $\mathcal{D}(A_D, v)$ is not known?

- If only the entries of the starting matrix A are known, then one can compute with the usual *recursive summation* method

$$v_i := a_{ii} - \sum_{j \neq i} |a_{ij}| \qquad \text{for all } i,$$

  but it may produce large relative cancellation errors if $a_{ii} \approx \sum_{j \neq i} |a_{ij}|$ and this would spoil the accuracy of the whole computation.

- In case of severe cancellation, one can compute the v_i with *doubly compensated summation* (Priest, 1992) that computes the sum of n numbers with relative error 2ϵ with cost of $10(n-1)$ flops.

What happens if the vector v in $\mathcal{D}(A_D, v)$ is not known?

- If only the entries of the starting matrix A are known, then one can compute with the usual *recursive summation* method

$$v_i := a_{ii} - \sum_{j \neq i} |a_{ij}| \qquad \text{for all } i,$$

  but it may produce large relative cancellation errors if $a_{ii} \approx \sum_{j \neq i} |a_{ij}|$ and this would spoil the accuracy of the whole computation.

- In case of severe cancellation, one can compute the v_i with *doubly compensated summation* (Priest, 1992) that computes the sum of n numbers with relative error 2ϵ with cost of $10(n-1)$ flops.

What happens if the vector v in $\mathcal{D}(A_D, v)$ is not known?

- If only the entries of the starting matrix A are known, then one can compute with the usual *recursive summation* method

$$v_i := a_{ii} - \sum_{j \neq i} |a_{ij}| \qquad \text{for all } i,$$

  but it may produce large relative cancellation errors if $a_{ii} \approx \sum_{j \neq i} |a_{ij}|$ and this would spoil the accuracy of the whole computation.

- In case of severe cancellation, one can compute the v_i with *doubly compensated summation* (Priest, 1992) that computes the sum of n numbers with relative error 2ϵ with cost of $10(n-1)$ flops.

The absence of cancellation does not imply accuracy (I)

- Best available error bounds for Q. Ye's algorithm increase exponentially with the dimension $6 \cdot n \cdot 8^{(n-1)}\epsilon$.
- This algorithm avoids partially cancellation, but I will assume a much more favorable scenario to show why a *direct forward error analysis* produces exponential error bounds in the dimension.
- Assumption: There is no cancellation at all in the whole process of Gaussian elimination, so, in every step $k \longrightarrow k+1$ and in every update

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}}, \qquad (k+1) \leq i, j \leq n,$$

$a_{ij}^{(k+1)}$ is computed as a sum of two numbers with the same sign.

- Let the relative errors in the computed entries of iterate $A^{(k)}$ be

$$\widehat{a}_{ij}^{(k)} = a_{ij}^{(k)} \langle p_k \rangle \qquad k \leq i, j \leq n,$$

where $$\langle p_k \rangle := \prod_{i=1}^{p_k} (1+\delta_i)^{\pm 1}, \qquad |\delta_i| \leq \epsilon \qquad \text{(Stewart's notation)}$$

The absence of cancellation does not imply accuracy (I)

- Best available error bounds for Q. Ye's algorithm increase exponentially with the dimension $6 \cdot n \cdot 8^{(n-1)}\epsilon$.
- This algorithm avoids partially cancellation, but I will assume a much more favorable scenario to show why a *direct forward error analysis* produces exponential error bounds in the dimension.
- Assumption: There is no cancellation at all in the whole process of Gaussian elimination, so, in every step $k \longrightarrow k+1$ and in every update

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}}, \qquad (k+1) \leq i, j \leq n,$$

  $a_{ij}^{(k+1)}$ is computed as a sum of two numbers with the same sign.
- Let the relative errors in the computed entries of iterate $A^{(k)}$ be

$$\widehat{a}_{ij}^{(k)} = a_{ij}^{(k)} \langle p_k \rangle \qquad k \leq i, j \leq n,$$

where $\qquad \langle p_k \rangle := \prod_{i=1}^{p_k} (1+\delta_i)^{\pm 1}, \qquad |\delta_i| \leq \epsilon \qquad$ (Stewart's notation)

The absence of cancellation does not imply accuracy (I)

- Best available error bounds for Q. Ye's algorithm increase exponentially with the dimension $6 \cdot n \cdot 8^{(n-1)}\epsilon$.
- This algorithm avoids partially cancellation, but I will assume a much more favorable scenario to show why a *direct forward error analysis* produces exponential error bounds in the dimension.
- Assumption: There is no cancellation at all in the whole process of Gaussian elimination, so, in every step $k \longrightarrow k+1$ and in every update

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}}, \qquad (k+1) \leq i, j \leq n,$$

$a_{ij}^{(k+1)}$ is computed as a sum of two numbers with the same sign.

- Let the relative errors in the computed entries of iterate $A^{(k)}$ be

$$\widehat{a}_{ij}^{(k)} = a_{ij}^{(k)} \langle p_k \rangle \qquad k \leq i, j \leq n,$$

where $\langle p_k \rangle := \prod_{i=1}^{p_k} (1+\delta_i)^{\pm 1}, \qquad |\delta_i| \leq \epsilon \qquad$ (Stewart's notation)

The absence of cancellation does not imply accuracy (I)

- Best available error bounds for Q. Ye's algorithm increase exponentially with the dimension $6 \cdot n \cdot 8^{(n-1)}\epsilon$.
- This algorithm avoids partially cancellation, but I will assume a much more favorable scenario to show why a *direct forward error analysis* produces exponential error bounds in the dimension.
- Assumption: There is no cancellation at all in the whole process of Gaussian elimination, so, in every step $k \longrightarrow k+1$ and in every update

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}}, \qquad (k+1) \leq i, j \leq n,$$

  $a_{ij}^{(k+1)}$ is computed as a sum of two numbers with the same sign.
- Let the relative errors in the computed entries of iterate $A^{(k)}$ be

$$\widehat{a}_{ij}^{(k)} = a_{ij}^{(k)} \langle p_k \rangle \qquad k \leq i, j \leq n,$$

where $\quad \langle p_k \rangle := \prod_{i=1}^{p_k} (1+\delta_i)^{\pm 1}, \qquad |\delta_i| \leq \epsilon \qquad$ (Stewart's notation)

The absence of cancellation does not imply accuracy (I)

- Best available error bounds for Q. Ye's algorithm increase exponentially with the dimension $6 \cdot n \cdot 8^{(n-1)}\epsilon$.
- This algorithm avoids partially cancellation, but I will assume a much more favorable scenario to show why a *direct forward error analysis* produces exponential error bounds in the dimension.
- Assumption: There is no cancellation at all in the whole process of Gaussian elimination, so, in every step $k \longrightarrow k+1$ and in every update

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}}, \qquad (k+1) \leq i, j \leq n,$$

$a_{ij}^{(k+1)}$ is computed as a sum of two numbers with the same sign.

- Let the relative errors in the computed entries of iterate $A^{(k)}$ be

$$\widehat{a}_{ij}^{(k)} = a_{ij}^{(k)} \langle p_k \rangle \qquad k \leq i, j \leq n,$$

where $$\langle p_k \rangle := \prod_{i=1}^{p_k} (1+\delta_i)^{\pm 1}, \qquad |\delta_i| \leq \epsilon \qquad \text{(Stewart's notation)}$$

The absence of cancellation does not imply accuracy (II)

- Computed entries of $A^{(k+1)}$

$$\widehat{a}_{ij}^{(k+1)} = \widehat{a}_{ij}^{(k)} \langle 1 \rangle - \frac{\widehat{a}_{ik}^{(k)} \widehat{a}_{kj}^{(k)}}{\widehat{a}_{kk}^{(k)}} \langle 3 \rangle = a_{ij}^{(k)} \langle p_k \rangle \langle 1 \rangle - \frac{a_{ik}^{(k)} \langle p_k \rangle a_{kj}^{(k)} \langle p_k \rangle}{a_{kk}^{(k)} \langle p_k \rangle} \langle 3 \rangle$$

-
$$\widehat{a}_{ij}^{(k+1)} = \left(a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}} \right) \langle 3\, p_k + 3 \rangle = a_{ij}^{(k+1)} \langle 3\, p_k + 3 \rangle$$

- We have proved that
$$p_{k+1} = 3\, p_k + 3$$

-
$$p_1 = 0 \Longrightarrow p_n = 3^n \left(\frac{3^n - 1}{2 \cdot 3^{n-1}} - 1 \right) \Longrightarrow p_n \approx \frac{3^n}{2}$$

- **Relative error bounds for LDU without cancellation:** $3^n \epsilon$

The absence of cancellation does not imply accuracy (II)

- Computed entries of $A^{(k+1)}$

$$\widehat{a}_{ij}^{(k+1)} = \widehat{a}_{ij}^{(k)} \langle 1 \rangle - \frac{\widehat{a}_{ik}^{(k)} \widehat{a}_{kj}^{(k)}}{\widehat{a}_{kk}^{(k)}} \langle 3 \rangle = a_{ij}^{(k)} \langle p_k \rangle \langle 1 \rangle - \frac{a_{ik}^{(k)} \langle p_k \rangle a_{kj}^{(k)} \langle p_k \rangle}{a_{kk}^{(k)} \langle p_k \rangle} \langle 3 \rangle$$

- $$\widehat{a}_{ij}^{(k+1)} = \left(a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}} \right) \langle 3\, p_k + 3 \rangle = a_{ij}^{(k+1)} \langle 3\, p_k + 3 \rangle$$

- We have proved that

$$p_{k+1} = 3\, p_k + 3$$

- $$p_1 = 0 \Longrightarrow p_n = 3^n \left(\frac{3^n - 1}{2 \cdot 3^{n-1}} - 1 \right) \Longrightarrow p_n \approx \frac{3^n}{2}$$

- **Relative error bounds for LDU without cancellation: $3^n \epsilon$**

The absence of cancellation does not imply accuracy (II)

- Computed entries of $A^{(k+1)}$

$$\widehat{a}_{ij}^{(k+1)} = \widehat{a}_{ij}^{(k)} \langle 1 \rangle - \frac{\widehat{a}_{ik}^{(k)} \widehat{a}_{kj}^{(k)}}{\widehat{a}_{kk}^{(k)}} \langle 3 \rangle = a_{ij}^{(k)} \langle p_k \rangle \langle 1 \rangle - \frac{a_{ik}^{(k)} \langle p_k \rangle a_{kj}^{(k)} \langle p_k \rangle}{a_{kk}^{(k)} \langle p_k \rangle} \langle 3 \rangle$$

-

$$\widehat{a}_{ij}^{(k+1)} = \left(a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}} \right) \langle 3\, p_k + 3 \rangle = a_{ij}^{(k+1)} \langle 3\, p_k + 3 \rangle$$

- We have proved that

$$p_{k+1} = 3\, p_k + 3$$

-

$$p_1 = 0 \Longrightarrow p_n = 3^n \left(\frac{3^n - 1}{2 \cdot 3^{n-1}} - 1 \right) \Longrightarrow p_n \approx \frac{3^n}{2}$$

- **Relative error bounds for LDU without cancellation: $3^n \epsilon$**

The absence of cancellation does not imply accuracy (II)

- Computed entries of $A^{(k+1)}$

$$\widehat{a}_{ij}^{(k+1)} = \widehat{a}_{ij}^{(k)} \langle 1 \rangle - \frac{\widehat{a}_{ik}^{(k)} \widehat{a}_{kj}^{(k)}}{\widehat{a}_{kk}^{(k)}} \langle 3 \rangle = a_{ij}^{(k)} \langle p_k \rangle \langle 1 \rangle - \frac{a_{ik}^{(k)} \langle p_k \rangle a_{kj}^{(k)} \langle p_k \rangle}{a_{kk}^{(k)} \langle p_k \rangle} \langle 3 \rangle$$

-

$$\widehat{a}_{ij}^{(k+1)} = \left(a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}} \right) \langle 3\, p_k + 3 \rangle = a_{ij}^{(k+1)} \langle 3\, p_k + 3 \rangle$$

- We have proved that

$$p_{k+1} = 3\, p_k + 3$$

-

$$p_1 = 0 \Longrightarrow p_n = 3^n \left(\frac{3^n - 1}{2 \cdot 3^{n-1}} - 1 \right) \Longrightarrow p_n \approx \frac{3^n}{2}$$

- Relative error bounds for LDU without cancellation: $3^n \epsilon$

The absence of cancellation does not imply accuracy (II)

- Computed entries of $A^{(k+1)}$

$$\widehat{a}_{ij}^{(k+1)} = \widehat{a}_{ij}^{(k)} \langle 1 \rangle - \frac{\widehat{a}_{ik}^{(k)} \widehat{a}_{kj}^{(k)}}{\widehat{a}_{kk}^{(k)}} \langle 3 \rangle = a_{ij}^{(k)} \langle p_k \rangle \langle 1 \rangle - \frac{a_{ik}^{(k)} \langle p_k \rangle a_{kj}^{(k)} \langle p_k \rangle}{a_{kk}^{(k)} \langle p_k \rangle} \langle 3 \rangle$$

- $$\widehat{a}_{ij}^{(k+1)} = \left(a_{ij}^{(k)} - \frac{a_{ik}^{(k)} a_{kj}^{(k)}}{a_{kk}^{(k)}} \right) \langle 3\, p_k + 3 \rangle = a_{ij}^{(k+1)} \langle 3\, p_k + 3 \rangle$$

- We have proved that

$$p_{k+1} = 3\, p_k + 3$$

- $$p_1 = 0 \Longrightarrow p_n = 3^n \left(\frac{3^n - 1}{2 \cdot 3^{n-1}} - 1 \right) \Longrightarrow p_n \approx \frac{3^n}{2}$$

- **Relative error bounds for LDU without cancellation:** $3^n \epsilon$

Outline

We need the help of perturbation theory...

Theorem

Let $A = \mathcal{D}(A_D, v) \in \mathbb{R}^{n\times n}$ and $\widetilde{A} = \mathcal{D}(\widetilde{A}_D, \widetilde{v}) \in \mathbb{R}^{n\times n}$ be row diagonally dominant matrices, and $A = LDU$ and $\widetilde{A} = \widetilde{L}\,\widetilde{D}\,\widetilde{U}$ be their factorizations. If

$$|\widetilde{v} - v| \leq \delta\, v \quad \text{and} \quad |\widetilde{A}_D - A_D| \leq \delta |A_D|, \quad \text{with} \ \ \delta < 1,$$

then

- For $i = 1 : n$

$$\widetilde{d}_{ii} = d_{ii} \frac{(1+\eta_1)\cdots(1+\eta_i)}{(1+\alpha_1)\cdots(1+\alpha_{i-1})} \qquad |\eta_k| \leq \delta, \ |\alpha_k| \leq \delta.$$

- For $i < j$

$$|\widetilde{u}_{ij} - u_{ij}| \leq 3\, i\delta$$

Recall: $\max_{ij} |u_{ij}| = \max_{ii} |u_{ii}| = 1$.

Perturbation of the L factor.

Theorem (continuation)

- For $i > j$,

$$|\widetilde{\ell}_{ij} - \ell_{ij}| \leq |\ell_{ij}| \left(\frac{1}{(1-\delta)^j} - 1 \right) + 2 \, \frac{(1+\delta)^j - 1}{(1-\delta)^j} \left| \frac{a_{ii}^{(j)}}{a_{jj}^{(j)}} \right|$$
$$= (\, j\delta + O(\delta^2)\,) \left(|\ell_{ij}| + 2 \left| \frac{a_{ii}^{(j)}}{a_{jj}^{(j)}} \right| \right),$$

  where $A^{(j)}$ is the matrix obtained after $(j-1)$ steps of Gaussian elimination.

- If the matrix A is ordered for complete pivoting, then $|\ell_{ij}| \leq 1$, $|a_{ii}^{(j)}| \leq |a_{jj}^{(j)}|$ and

$$|\widetilde{\ell}_{ij} - \ell_{ij}| \leq 3\, j\delta + O(\delta^2)$$

Perturbation of the L factor.

Theorem (continuation)

- For $i > j$,

$$|\widetilde{\ell}_{ij} - \ell_{ij}| \leq |\ell_{ij}| \left(\frac{1}{(1-\delta)^j} - 1 \right) + 2\, \frac{(1+\delta)^j - 1}{(1-\delta)^j} \left| \frac{a_{ii}^{(j)}}{a_{jj}^{(j)}} \right|$$
$$= (\, j\delta + O(\delta^2)\,) \left(|\ell_{ij}| + 2 \left| \frac{a_{ii}^{(j)}}{a_{jj}^{(j)}} \right| \right),$$

  where $A^{(j)}$ is the matrix obtained after $(j-1)$ steps of Gaussian elimination.

- If the matrix A is ordered for complete pivoting, then $|\ell_{ij}| \leq 1$, $|a_{ii}^{(j)}| \leq |a_{jj}^{(j)}|$ and

$$|\widetilde{\ell}_{ij} - \ell_{ij}| \leq 3\, j\delta + O(\delta^2)$$

Complete pivoting is essential for good behavior of L: Example

Matrix ordered according to a pivoting strategy designed to make the **factor L column diagonally dominant** and as much as possible.

$$A = \begin{bmatrix} 1000 & 100 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 400 \\ 0.05 \\ 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70 \end{bmatrix} \begin{bmatrix} 1 & 0.1 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Example: $\delta \approx 10^{-2}$ perturbation in $\mathcal{D}(A_D, v)$.

$$\widetilde{A} = \begin{bmatrix} 1000 & 101 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 399 \\ 0.05 \\ 10 \end{bmatrix}$$

$$\widetilde{A} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70.05 \end{bmatrix} \begin{bmatrix} 1 & 0.101 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Complete pivoting is essential for good behavior of L: Example

Matrix ordered according to a pivoting strategy designed to make the **factor L column diagonally dominant** and as much as possible.

$$A = \begin{bmatrix} 1000 & 100 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 400 \\ 0.05 \\ 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70 \end{bmatrix} \begin{bmatrix} 1 & 0.1 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Example: $\delta \approx 10^{-2}$ perturbation in $\mathcal{D}(A_D, v)$.

$$\widetilde{A} = \begin{bmatrix} 1000 & 101 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 399 \\ 0.05 \\ 10 \end{bmatrix}$$

$$\widetilde{A} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70.05 \end{bmatrix} \begin{bmatrix} 1 & 0.101 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Complete pivoting is essential for good behavior of L: Example

Matrix ordered according to a pivoting strategy designed to make the **factor L column diagonally dominant** and as much as possible.

$$A = \begin{bmatrix} 1000 & 100 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 400 \\ 0.05 \\ 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70 \end{bmatrix} \begin{bmatrix} 1 & 0.1 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Example: $\delta \approx 10^{-2}$ perturbation in $\mathcal{D}(A_D, v)$.

$$\widetilde{A} = \begin{bmatrix} 1000 & 101 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 399 \\ 0.05 \\ 10 \end{bmatrix}$$

$$\widetilde{A} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70.05 \end{bmatrix} \begin{bmatrix} 1 & 0.101 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Complete pivoting is essential for good behavior of L: Example

Matrix ordered according to a pivoting strategy designed to make the **factor L column diagonally dominant** and as much as possible.

$$A = \begin{bmatrix} 1000 & 100 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 400 \\ 0.05 \\ 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70 \end{bmatrix} \begin{bmatrix} 1 & 0.1 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Example: $\delta \approx 10^{-2}$ perturbation in $\mathcal{D}(A_D, v)$.

$$\widetilde{A} = \begin{bmatrix} 1000 & 101 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 399 \\ 0.05 \\ 10 \end{bmatrix}$$

$$\widetilde{A} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70.05 \end{bmatrix} \begin{bmatrix} 1 & 0.101 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Complete pivoting is essential for good behavior of L: Example

Matrix ordered according to a pivoting strategy designed to make the **factor L column diagonally dominant** and as much as possible.

$$A = \begin{bmatrix} 1000 & 100 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 400 \\ 0.05 \\ 10 \end{bmatrix}$$

$$A = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & \boxed{0} & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70 \end{bmatrix} \begin{bmatrix} 1 & 0.1 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Example: $\delta \approx 10^{-2}$ perturbation in $\mathcal{D}(A_D, v)$.

$$\widetilde{A} = \begin{bmatrix} 1000 & 101 & 500 \\ 0 & 0.1 & 0.05 \\ 100 & 10 & 120 \end{bmatrix}, \qquad v(A) = \begin{bmatrix} 399 \\ 0.05 \\ 10 \end{bmatrix}$$

$$\widetilde{A} = \begin{bmatrix} 1 & & \\ 0 & 1 & \\ 0.1 & \boxed{-1} & 1 \end{bmatrix} \begin{bmatrix} 1000 & & \\ & 0.1 & \\ & & 70.05 \end{bmatrix} \begin{bmatrix} 1 & 0.101 & 0.5 \\ & 1 & 0.5 \\ & & 1 \end{bmatrix}$$

Outline

New error bounds for Q. Ye's algorithm

Theorem

Let us apply Ye's algorithm with complete pivoting on $A = \mathcal{D}(A_D, v) \in \mathbb{R}^{n\times n}$ row diagonally dominant matrix to compute $\widehat{L}$, $\widehat{D}$ and $\widehat{U}$ with machine precision ϵ. If L, D and U are the exact factors then:

- *For* $i > j$
$$|\widehat{\ell}_{ij} - \ell_{ij}| \leq 14\, n\, j^2\, \epsilon < 14\, n^3\, \epsilon.$$
- *For* $i = 1, \ldots, n$
$$|\widehat{d}_{ii} - d_{ii}| \leq |d_{ii}| \frac{6\, n\, i^2\, \epsilon}{1 - 6\, n\, i^2\, \epsilon} \leq |d_{ii}| \frac{6\, n^3\, \epsilon}{1 - 6\, n^3\, \epsilon}.$$
- *For* $i < j$
$$|\widehat{u}_{ij} - u_{ij}| \leq 8\, n\, i^2\, \epsilon < 8\, n^3\, \epsilon.$$

Recall: $\max_{ij} |u_{ij}| = \max_{ij} |\ell_{ij}| = 1$.

$O(n^3\, \epsilon)$ error bounds, no exponential growth with the dimension.

New error bounds for Q. Ye's algorithm

Theorem

Let us apply Ye's algorithm with complete pivoting on $A = \mathcal{D}(A_D, v) \in \mathbb{R}^{n\times n}$ row diagonally dominant matrix to compute $\widehat{L}$, $\widehat{D}$ and $\widehat{U}$ with machine precision ϵ. If L, D and U are the exact factors then:

- *For* $i > j$
$$|\widehat{\ell}_{ij} - \ell_{ij}| \leq 14\, n\, j^2\, \epsilon < 14\, n^3\, \epsilon.$$
- *For* $i = 1, \ldots, n$
$$|\widehat{d}_{ii} - d_{ii}| \leq |d_{ii}|\, \frac{6\, n\, i^2\, \epsilon}{1 - 6\, n\, i^2\, \epsilon} \leq |d_{ii}|\, \frac{6\, n^3\, \epsilon}{1 - 6\, n^3\, \epsilon}.$$
- *For* $i < j$
$$|\widehat{u}_{ij} - u_{ij}| \leq 8\, n\, i^2\, \epsilon < 8\, n^3\, \epsilon.$$

Recall: $\max_{ij} |u_{ij}| = \max_{ij} |\ell_{ij}| = 1$.

$O(n^3\, \epsilon)$ **error bounds, no exponential growth with the dimension.**

Outline

Conclusions

- The satisfactory error analysis that we have presented is possible because a structured perturbation theory has been developed.

- This error analysis proves rigorously that for any diagonally dominant matrix A, there are algorithms that
  - compute its SVD with high relative accuracy, (Ye's + Demmel et al)
  - compute accurately the solution of $Ax = b$ for almost every b, (Ye's + D-Molera)

  with cost $O(n^3)$ and independently of the traditional condition number of A.

- The perturbation theory for LDU can be combined with results from Demmel et al. to obtain very good relative perturbation bounds for the SVD depending on $\kappa(L)$ and $\kappa(U)$. Can this be improved?