

# Numerical Linear Algebra

## Review

Gerard Sleijpen and Martin van Gijzen

September 7, 2011

# The instructor & Co-author

Dr. Gerard Sleijpen

Utrecht University

Mathematical Institute

Budapestlaan 6. room 504

Utrecht

Tel: +31-30-253 1732

Email: [G.L.G.Sleijpen@uu.nl](mailto:G.L.G.Sleijpen@uu.nl)

<http://www.staff.science.uu.nl/~sleij101>

Dr. ir. Martin van Gijzen

Delft University of Technology

Faculty EWI

Mekelweg 4, room HB 07.260

2628 CD Delft

Tel: +31 15-2782519

E-mail: [M.B.vanGijzen@TUDelft.nl](mailto:M.B.vanGijzen@TUDelft.nl)

<http://ta.twi.tudelft.nl/nw/users/gijzen/>

# Program September 7

- Overview of the course
- Useful references
- A motivating example
- Review of basic linear algebra concepts
- Inner products, vector norms and matrix norms
- Condition number, finite precision arithmetic

# Goals of the course

To provide theoretical insight and to develop practical skills for solving numerically large scale linear algebra problems.

Particular emphasis lies on large-scale linear systems and on eigenvalue problems.

At the end of the course you will

- understand the principles behind modern solution techniques for linear algebra problems;
- be able to implement them and to understand their behaviour;
- and you will be able to select (and adapt) a suitable method for you problem.

# Topics per day

- Day 1 and 2: review of linear algebra
- Day 3: direct solution methods
- Day 4: basic iterative methods for linear systems and eigenvalue problems
- Day 5-11: Krylov methods for linear systems and eigenvalue problems
- Day 12 and 13: Multigrid, preconditioning, parallel implementation, special topics
- Day 14: Eigenvalue problems, special topics
- **Note: no lecture on November 9**

# Examination

- On Day 3: **mandatory** linear algebra review test
- Every week: homework assignment, to be handed in. The homework assignment must be made individually.
- Day 14: final project assignment You are allowed to do the project assignment in pairs. The report has to be handed in on January 31, 2012 at the latest. After handing it in you have to make an appointment to defend your report.

# Recommended literature

- Gene H. Golub and Charles F. Van Loan. Matrix Computations. 3rd ed. The Johns Hopkins University Press, Baltimore, 1996.
- Henk van der Vorst, Iterative methods for large linear systems. Cambridge press, 2003
- Richard Barrett et al. Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods. 2nd edition, SIAM, 1994.
- Zhaojun Bai et al. Templates for the Solution of Algebraic Eigenvalue Problems 1st edition, SIAM, 2000

# Useful webpages

- <http://www.netlib.org/>: a wealth of information to numerical software and other information, e.g.
  - LAPACK, BLAS: dense linear algebra
  - NETSOLVE: grid computing
  - TEMPLATES: the book plus software
  - MATRIX MARKET: matrices
- <http://www.math.uu.nl/people/vorst/>: manuscript of the book, software
- <http://www-math.mit.edu/gs/>: Gilbert Strang's homepage: video course, demos ...

# Motivation of the course

Many applications give rise to large linear algebra problems.

Typically these problems involve matrices that are

- Large,  $10^8$  unknowns are not exceptional anymore;
- Sparse, only a fraction of the entries of the matrix is nonzero;
- Structured, the matrix often has a symmetric pattern and is banded.

Moreover, the matrix can have special numerical properties, e.g it may be symmetric, Toeplitz, or the eigenvalues may all be in the right-half plane.

# An application: ocean circulation (1)

Physical model: balance between

- Wind force
- Coriolis force
- Bottom friction.

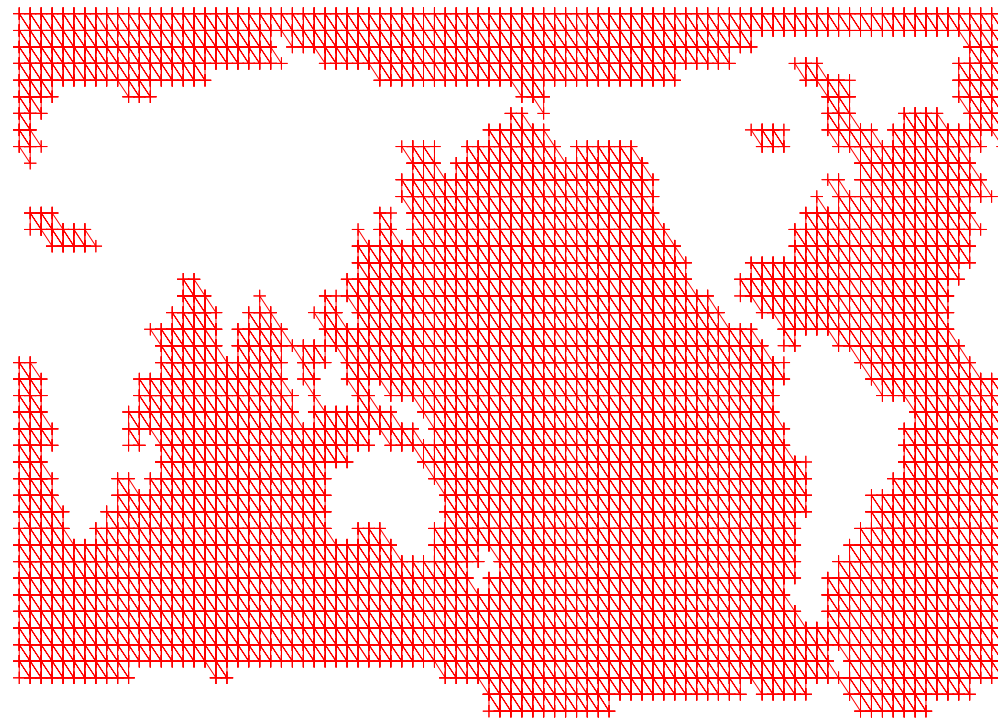
# An application: ocean circulation (2)

Mathematical model

$$r\nabla^2\psi + \beta\frac{\partial\psi}{\partial x} = \nabla \times \mathbf{F} \quad \text{in } \Omega,$$

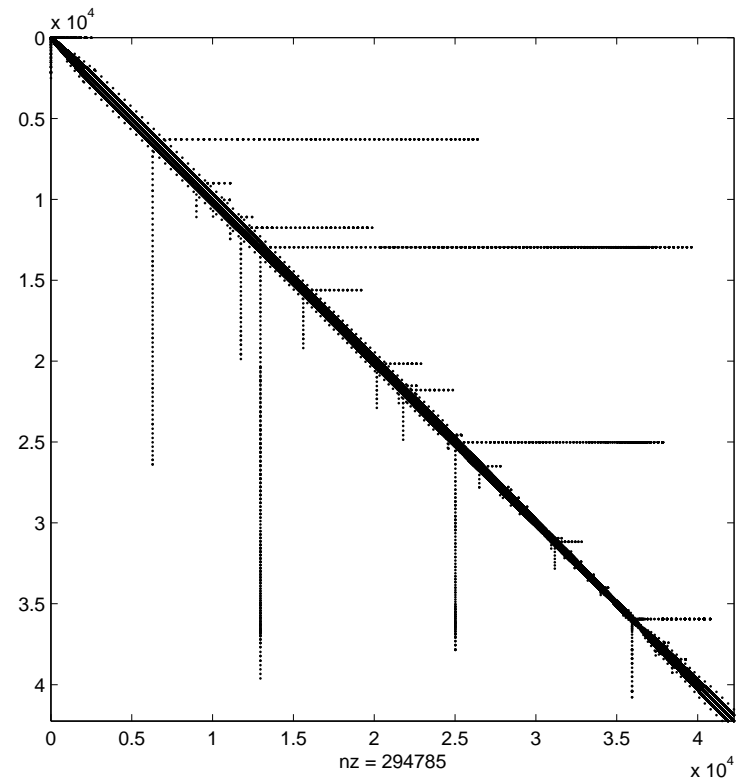
- $\psi$ : streamfunction
- $r$ : bottom friction parameter
- $\beta$ : Coriolis parameter
- $\mathbf{F}$ : Wind stress

# An application: ocean circulation (3)



Numerical model: discretisation with FEM

# An application: ocean circulation (4)



The nonzero pattern of the resulting matrix  $A$

# Solving the resulting system

In order to be able to solve this problem you have to consider many questions:

- How can you exploit the sparsity of the matrix?
- Can you make use of the arrow structure?
- Is the matrix symmetric? Can you exploit this?
- Is the matrix close to singular? Is your solution algorithm sensitive to (numerical) errors?
- Can your solution method exploit the available (parallel) hardware?
- ...

# Assignment 1a

The matrix

$$\begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}$$

is the discretisation of  $-\frac{d^2y}{dx^2}$ , with bc  $\frac{dy}{dx}(0) = \frac{dy}{dx}(1) = 0$ .

List as many characteristics of this matrix as possible.

# Assignment 1a

The matrix

$$\begin{pmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}$$

is the discretisation of  $-\frac{d^2y}{dx^2}$ , with bc  $\frac{dy}{dx}(0) = \frac{dy}{dx}(1) = 0$ .

List as many characteristics of this matrix as possible.

- Symmetry? Positive definite?
- Rank? Range? Nullspace?
- Eigenvalues? Eigenvectors?

# Assignment 1b

The matrix

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

is the (upwind) discretisation of  $\frac{dy}{dx}$ , with bc  $y(0) = 0$ .

List as many characteristics of this matrix as possible.

# Inner products

The inner product is a function  $(\cdot, \cdot): \mathbb{C}^n \times \mathbb{C}^n \rightarrow \mathbb{C}$  that satisfies the following properties:

$$\text{i) } (x, y) = \overline{(y, x)} \quad x, y \in \mathbb{C}^n ,$$

$$\text{ii) } (x + y, z) = (x, z) + (y, z), \quad x, y, z \in \mathbb{C}^n ,$$

and  $(\alpha x, y) = \alpha(x, y), \quad \alpha \in \mathbb{C}, x, y \in \mathbb{C}^n ,$

$$\text{iii) } (x, x) \geq 0, \quad (x, x) = 0 \iff x = 0, \quad x \in \mathbb{C}^n .$$

# Vector norms (1)

A vector norm on  $\mathbb{C}^n$  is a function  $\|\cdot\| : \mathbb{C}^n \rightarrow \mathbb{R}$  that satisfies the following properties:

i)  $\|x\| \geq 0$   $x \in \mathbb{C}^n$  , and  
 $\|x\| = 0 \iff x = 0$ ,

ii)  $\|x + y\| \leq \|x\| + \|y\|$   $x, y \in \mathbb{C}^n$  ,

iii)  $\|\alpha x\| = |\alpha| \|x\|$   $\alpha \in \mathbb{C}$  ,  $x \in \mathbb{C}^n$  .

# Vector norms (2)

An important class of vector norms are the so-called  $p$ -norms (Hölder norms) defined by

$$\|x\|_p = (|x_1|^p + \dots + |x_n|^p)^{1/p} \quad p \geq 1.$$

The 1,2, and  $\infty$  norms are the most commonly used

$$\|x\|_1 = |x_1| + \dots + |x_n|$$

$$\|x\|_2 = (|x_1|^2 + \dots + |x_n|^2)^{1/2} = (x^H x)^{1/2}$$

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i|.$$

# Assignment 2

Let the matrix  $A$  be Hermitian and Positive Definite.

- Is  $(x, y)_A = x^H A y$  an inner product?
- Is the norm that is induced by this inner product a proper norm? Here  $x^H$  denotes the conjugate transpose of  $x$ .

# Orthogonality

Two vectors  $x$  and  $y$  are orthogonal with respect to an inner product if

$$(x, y) = 0.$$

The notation  $x \perp y$  means that  $x$  and  $y$  are orthogonal.

If the inner product is not specified the standard inner product ( $p = 2$ ) is assumed.

Two space  $U$  and  $V$  are orthogonal if for every  $u \in U$  and  $v \in V$  we have

$$(u, v) = 0$$

# Orthogonal matrices

A matrix is called orthogonal if

- All columns of the matrix are orthogonal (with respect to the standard inner product),
- and the columns are normalised.

Hence for an orthogonal matrix  $Q$  we have  $Q^H Q = I$ , with  $I$  the identity matrix.

# Matrix norms (1)

The analysis of matrix algorithms frequently requires use of matrix norms.

For example, the quality of a linear system solver may be poor if the matrix of coefficients is "nearly singular".

To quantify the notion of near-singularity we need a measure of distance on the space of matrices. Matrix norms provide that measure.

# Matrix norms (2)

A matrix norm on  $\mathbb{C}^{m \times n}$  is a function  $\|\cdot\| : \mathbb{C}^{m \times n} \rightarrow \mathbb{R}$  that satisfies the following properties:

$$\text{i) } \|A\| \geq 0 \quad A \in \mathbb{C}^{m \times n}, \quad \text{and} \\ \|A\| = 0 \iff A = 0,$$

$$\text{ii) } \|A + B\| \leq \|A\| + \|B\| \quad A, B \in \mathbb{C}^{m \times n},$$

$$\text{iii) } \|\alpha A\| = |\alpha| \|A\| \quad \alpha \in \mathbb{C}, x \in \mathbb{C}^{m \times n}.$$

The most commonly used matrix norms are the  $p$ -norms induced by the vector  $p$ -norms.

$$\|A\|_p = \sup_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p} = \max_{\|x\|_p=1} \|Ax\|_p \quad p \geq 1.$$

# Matrix norms (3)

Below we list some properties of vector and matrix  $p$ -norms

- $\|AB\|_p \leq \|A\|_p \|B\|_p \quad A \in \mathbb{C}^{m \times n}, \quad B \in \mathbb{C}^{n \times q}$
- $\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^m |a_{ij}| \quad A \in \mathbb{C}^{m \times n}$
- $\|A\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^n |a_{ij}| \quad A \in \mathbb{C}^{m \times n}$
- $\|A\|_2$  is equal to the square root of the largest eigenvalue of  $A^H A$ .
- All norms are equivalent, meaning that there are  $m, M > 0$  such that  $m\|A\|_p \leq \|A\|_q \leq M\|A\|_p$ .

# Matrix norms (4)

Matrix norms that are not induced by a vector norm also exist. One of the best known is the Frobenius norm. The Frobenius norm of an  $m \times n$  matrix  $A$  is given by

$$\|A\|_F = \sqrt{\sum_{i=1}^m \sum_{j=1}^n |a_{ij}|^2}$$

This is equal to

$$\|A\|_F = \sqrt{\text{Tr}(AA^H)}$$

In which  $\text{Tr}(A)$  is the trace of  $A$ , which is the sum of the main diagonal elements.

# Condition number (1)

The condition number plays an important role in numerical linear algebra since it gives a measure of how perturbations in  $A$  and  $b$  affect the solution  $x$ .

The condition number  $K_p(A)$ , for a nonsingular matrix  $A$ , is defined by

$$K_p(A) = \|A\|_p \|A^{-1}\|_p.$$

A low condition number means that small perturbations in the matrix or right-hand side give small changes in the solution.

A large condition number means that a small perturbation in the problem may give a large change in the solution.

# Condition number (2)

Suppose  $Ax = b$ ,  $A \in \mathbb{R}^{n \times n}$  and  $A$  is nonsingular,  $0 \neq b \in \mathbb{R}^n$ , and  $A(x + \Delta x) = b + \Delta b$ , then

$$\frac{\|\Delta x\|_p}{\|x\|_p} \leq K_p(A) \frac{\|\Delta b\|_p}{\|b\|_p} .$$

## Condition number (2)

Suppose  $Ax = b$ ,  $A \in \mathbb{R}^{n \times n}$  and  $A$  is nonsingular,  $0 \neq b \in \mathbb{R}^n$ , and  $A(x + \Delta x) = b + \Delta b$ , then

$$\frac{\|\Delta x\|_p}{\|x\|_p} \leq K_p(A) \frac{\|\Delta b\|_p}{\|b\|_p}.$$

Proof: From the properties of the norms it follows that

$$\|b\|_p = \|Ax\|_p \leq \|A\|_p \|x\|_p, \text{ so } \frac{1}{\|x\|_p} \leq \|A\|_p \frac{1}{\|b\|_p}.$$

We know that  $A\Delta x = \Delta b$ , so  $\Delta x = A^{-1}\Delta b$ . Furthermore

$$\|\Delta x\|_p = \|A^{-1}\Delta b\|_p \leq \|A^{-1}\|_p \|\Delta b\|_p.$$

Combination of these inequalities proves the theorem.

# Condition number (3)

Suppose you want the solution  $x$  of

$$Ax = b, \quad A \in \mathbb{C}^{n \times n} \text{ nonsingular}, \quad 0 \neq b \in \mathbb{C}^n$$

You actually solve the perturbed system

$$(A + \Delta A)(x + \Delta x) = b + \Delta b, \quad \Delta A \in \mathbb{C}^{n \times n}, \quad \Delta b \in \mathbb{C}^n$$

with  $\|\Delta A\|_p \leq \delta \|A\|_p$  and  $\|\Delta b\|_p \leq \delta \|b\|_p$ .

*When has this system a (unique) solution?*

# Condition number (3)

Suppose you want the solution  $x$  of

$$Ax = b, \quad A \in \mathbb{C}^{n \times n} \text{ nonsingular}, \quad 0 \neq b \in \mathbb{C}^n$$

You actually solve the perturbed system

$$(A + \Delta A)(x + \Delta x) = b + \Delta b, \quad \Delta A \in \mathbb{C}^{n \times n}, \quad \Delta b \in \mathbb{C}^n$$

with  $\|\Delta A\|_p \leq \delta \|A\|_p$  and  $\|\Delta b\|_p \leq \delta \|b\|_p$ .

*When has this system a (unique) solution?*

If  $K_p(A)\delta = r < 1$  then  $A + \Delta A$  is nonsingular and

$$\frac{\|\Delta x\|_p}{\|x\|_p} \leq \frac{2\delta}{1-r} K_p(A).$$

Proof: see Golub and Van Loan, p.83.

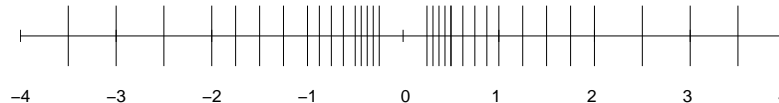
# Finite precision arithmetic

A computer stores real numbers as

$$f = \pm 0.d_1d_2\dots d_t \cdot \beta^e \quad d_1 > 0 \quad 0 \leq d_i < \beta, \quad L \leq e \leq U$$

$f$  is called a floating point number, and  $F$  is the set of floating point numbers.

- $\beta$ : base
- $t$ : precision
- $[L, U]$ : exponent range



*Note: floating point numbers are not equally spaced*

# Computing with floating point numbers

Each nonzero  $f \in F$  satisfies

$$m \leq |f| \leq M \text{ where } m = \beta^{L-1} \text{ and } M = \beta^U (1 - \beta^{-t}).$$

To have a model of computer arithmetic the set  $G$  is defined by

$$G = \{x \in \mathbb{R} \mid m \leq |x| \leq M\} \cup \{0\},$$

and the operator  $fl(\text{float})$ :  $G \rightarrow F$ , where  $fl$  maps a real number from  $G$  to a floating point number. The  $fl$  operator satisfies

$$fl(x) = x(1 + \epsilon), \quad |\epsilon| \leq u, \quad x \in G,$$

where  $u$  (unit roundoff) is defined by  $u = \frac{1}{2}\beta^{1-t}$

# Overflow and underflow

Let  $a$  and  $b$  be elements of  $F$ . If  $|a * b| \notin G$  then an arithmetic fault occurs if:

- $|a * b| > M$ : overflow, or
- $0 < |a * b| < m$ : underflow.

If  $a * b \in G$  then we assume that the computed version of  $a * b$  is given by  $fl(a * b)$  which is equal to  $(a * b)(1 + \epsilon)$  with  $|\epsilon| < u$ .

# Roundoff in basic operations

The following result for  $\alpha, A \in F$  is easy to show

$$fl(\alpha A) = \alpha A + E \quad |E| \leq u|\alpha A|$$

Here  $B = |A|$  means  $b_{ij} = |a_{ij}|$ ,  $i = 1, \dots, m$ ,  $j = 1, \dots, n$ .

Similarly we have

$$fl(A + B) = (A + B) + E \quad |E| \leq u|A + B|$$

and for the scaled vector update (called GAXPY) we get

$$fl(\alpha x + y) = \alpha x + y + e \quad |e| \leq u(2|\alpha x| + |y|) + O(u^2)$$

Similar results exist for all basic matrix and vector operations.

# Forward and backward error analysis

The previous results are obtained using a forward error analysis, which determines the error in the solution as a result of perturbations in the data. For the analysis of algorithms one commonly uses a backward error analysis. In this approach the solution is considered the exact solution of a perturbed problem, and the question is how big the perturbations are. For example, an algorithm for solving the linear system  $Ax = b$  yields a solution  $\tilde{x}$ .

Forward analysis: what is an upperbound for  $\|e\| = \|x - \tilde{x}\|$ ?

Backward analysis:  $\tilde{x}$  is solution of  $(A + \Delta A)\tilde{x} = (b + \Delta b)$ .

What are  $\Delta A$  and  $\Delta b$ ?

# Concluding remarks

Today we saw some of the concepts that will allow us to answer questions like:

- How sensitive is my problem to perturbation?
- What is the effect of finite precision arithmetic? How sensitive is my algorithm for finite precision calculations?
- How accurate is my solution?

In the following lessons these and similar questions will play a crucial role.

Further reading: Golub and van Loan, Page 48 - 68