Introduction to Scientific Computing

(Lecture 2: Machine precision and condition number)

Bojana Rosić

Institute of Scientific Computing

November 1, 2016

General information :)

- 13 homeworks (HW)
- Work in groups of 2 or 3 people
- Each HW brings maximally 36 points =468 points
- Exam requirement: $> 50\%$ or approx. 234 points
- Please, do not copy or you will get 0 points!!!
- Assignments and other information you may find on web-page: https://www.tu-braunschweig.de/wire/lehre/ws15/ode1
- Open office hours:
  - Dr. Moshagen, Tuesday 14:00-15:00
  - Dr. Rosić, Wednesday, 14:00-15:00

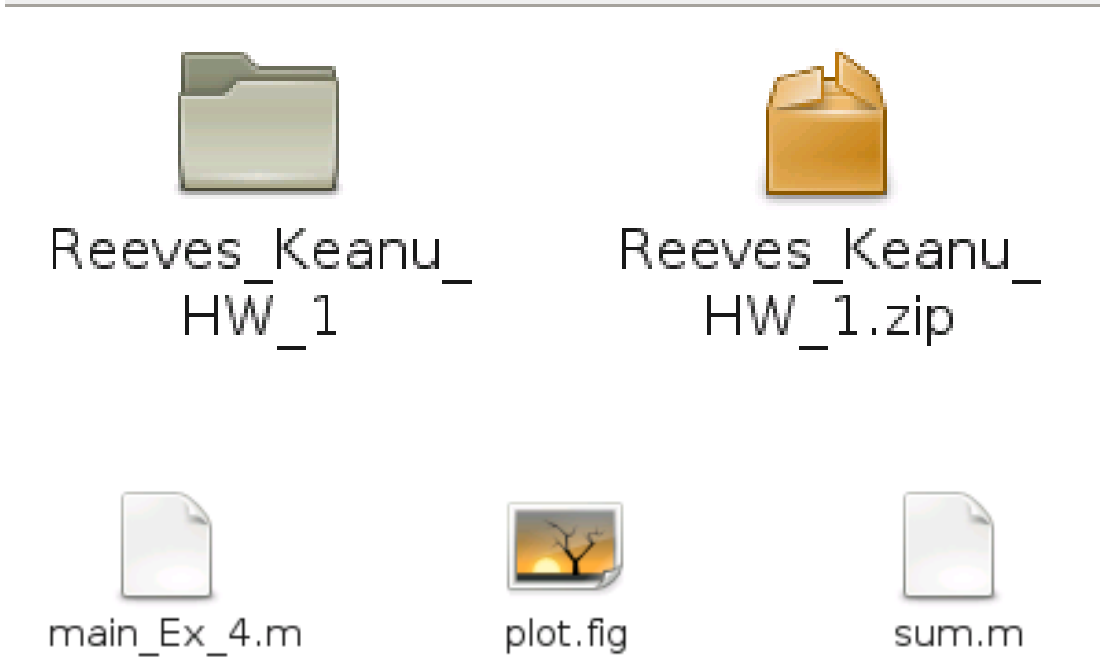
General information :)

- You are allowed to program in: Matlab (Octave),C, Fortran etc.
- We recommend Matlab (Octave) as easiest
- All program codes have to be sent per email:

  howisc15@gmail.com

- Other things than code in paper
  (don't forget to write your student ID and name)
- HW submit on tutorials
  (deadline: each next Friday/Monday before exercises)

HW format


Reeves_Keanu_
HW_1
Reeves_Keanu_
HW_1.zip
main_Ex_4.m
plot.fig
sum.m

HW format

```
%
%  This is the main file for Ex. 4
%
%  The script calculates the sum of array
%
%
% Student:
% Reeves Keanu
% ID 12345667

% Initialisation of variables

a=[ 1 2 3 4 5 6 7];

% Call the function

s=sum(a);
```

HW format

```
function s=sum(a)

% Function s=sum(a)
%
%  computes the sum of elements of array a
%
% Input:
%   a - array
%
% Output:
%
% s - sum
%
%  Student:
% Reeves Keanu
% ID 12345667
%


 % Initisalise sum

 s=0;

 % Find number of elements of a

 nelem=length(a);

 % Compute the sum

 for i=1:nelem
   s=s+a(i);
 end


end
```

- Function has to be written in a separate file.
- Function file has to be followed by main file
- Do not send only function file!
- Plot results!! Send all plots attached!

I Motivation for today's lecture: Ariane 5

Ariane 5 rockets are manufactured under the authority of the European Space Agency (ESA) and the Centre National d'Etudes Spatiales. Airbus Defence and Space is the prime contractor for the vehicles, leading a consortium of subcontractors. Ariane 5 is operated and marketed by Arianespace as part of the Ariane programme. Astrium builds the rockets in Europe and Arianespace launches them from the Guiana Space Centre in French Guiana.

Total cost: $500 million

Time of development: a decade

@stff.co.uk

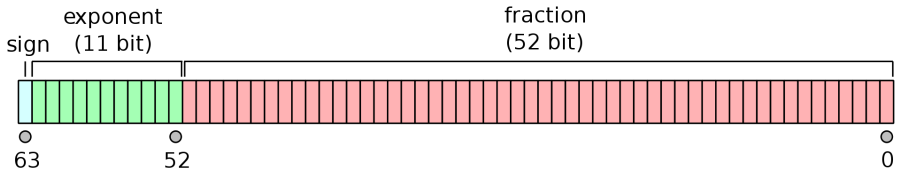
and ca. 40 seconds later...

@esa

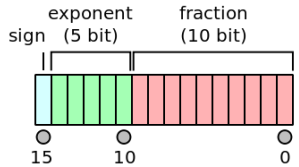
On **4 June 1996**, the maiden flight of the Ariane 5 launcher ended in a failure. Only about 40 seconds after initiation of the flight sequence, at an altitude of about 3700 m, the launcher veered off its flight path, broke up and exploded.

So, what happened?

Only one bug: computer program tried to stuff a 64-bit number into a 16-bit space



64 bit floating point number



16 bit integer number

This caused that the measured velocities and altitude of Ariane 5 were read wrong. Such measures were followed by extreme control instructions onto the engines, which resulted in self-destruction.

Computer Arithmetic

The number $x \in \mathbb{R}$ can be written as:

$$x = (sign)m \cdot b^c$$

where m is called mantissa, c is exponent and b is basis.
For example, the number 117 in **decimal system** reads

$$-117 = -1.17 \cdot 10^2 = -(1 \cdot 10^0 + 1 \cdot 10^{-1} + 7 \cdot 10^{-2}) \cdot 10^2$$

where $(sign) = -1$, mantissa $m = 1.17$, $b = 10$ and $c = 2$.

However, -117 can be also written in **binary system** as:

$$-117 = -1.110101 \cdot 2^6$$

where $(sign) = -1$, $m = 1.110101$, $b = 2$ and $c = 6$.

Computer Arithmetic

$$-117 = -1.110101 \cdot 2^6$$

In this case mantissa is:

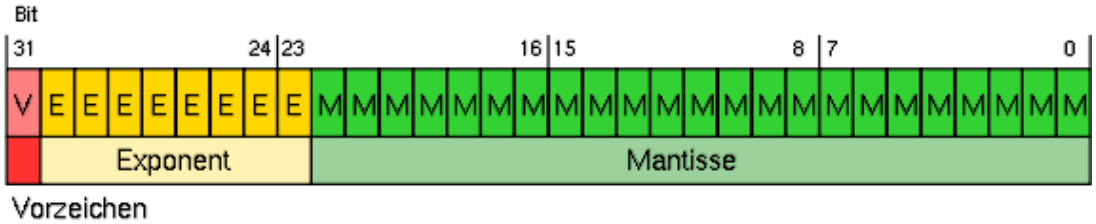
$$1.110101 = 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 0 \cdot 2^{-3} + 1 \cdot 2^{-4} + 0 \cdot 2^{-5} + 1 \cdot 2^{-6}$$

such that:

$$-117 = -(1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0)$$

Computer Arithmetic

Every computer has only fixed number of places to store sign, mantissa, basis and exponent.



Example: 32 bit number

Thus, every computer can represent all numbers x between x_{min} called underflow level and x_{max} called overflow level.

IEEE-Standard

- single-precision floating-point numbers

$$x = (-1)^v (0.1 m_2 m_3 \ldots m_{24})_2 2^{c-126}, \quad c := (c_7 \ldots c_0)_2 = \sum_{i=0}^{7} c_i 2^i$$

- double-precision floating-point numbers

$$x = (-1)^v (0.1 m_2 m_3 \ldots m_{53})_2 2^{c-1021}, \quad c := (c_{10} \ldots c_0)_2 = \sum_{i=0}^{10} c_i 2^i$$

Single-precision numbers

$$x = (-1)^v (0.1 m_2 m_3 \ldots m_{24})_2 2^{c-126}$$

Special cases: $c = 0$ (underflow) and $c = 255$ (overflow) $\Rightarrow$ NaN.

- The smallest and largest representable number:

$$a_{min}^{\pm} \approx 1.2 \cdot 10^{-38}, \quad a_{max}^{\pm} \approx \pm 3.4 \cdot 10^{38}$$

- The machine precision is

$$eps = \frac{1}{2} 2^{-23} = 2^{-24} \approx 10^{-7}$$

- 6 up to 7 places precision!!

Double-precision numbers

$$x = (-1)^v (0.1 m_2 m_3 ... m_{53})_2 2^{c-1021}$$

Special cases: $c = 0$ (underflow) and $c = 2047$ (overflow).

- The smallest and largest representable number:

$$a^{\pm}_{min} \approx 2.2 \cdot 10^{-308}, a^{\pm}_{max} \approx \pm 1.8 \cdot 10^{308}$$

- The machine precision is

$$eps = \frac{1}{2} 2^{-52} \approx 10^{-16}$$

- 15 up to 16 places precision!!

Dangers of computer arithmetic

We would like to add two numbers

$$117000 = 0.1170 \cdot 10^6$$

and

$$323 = 0.3230 \cdot 10^3$$

on computer which stores mantissa on 4 decimal spaces and exponent on 2.

Note: The computer can't add two numbers with different exponents, so it converts the smaller number into a form with the same exponent as the big number.

Dangers of computer arithmetic

Thus,

$$\begin{aligned} 117000 &= 0.1170 \cdot 10^6 \\ 323 &= 0.3230 \cdot 10^3 = 0.0003 \cdot 10^6. \end{aligned}$$

After summing, one obtains:

$$0.1170 \cdot 10^6 + 0.0003 \cdot 10^6 = 0.1173 \cdot 10^6$$

Finally, result is

$$s = 117300$$

and correct result is:

$$s_t = 117323.$$

Conclusion: Buy new computer!

Dangers of computer arithmetic

Finally, our computer made:

- absolute error

$$\Delta s = |s_t - s| = |117323 - 117300| = 23$$

- relative error

$$\epsilon_s = \frac{\Delta s}{|s_t|} = \frac{23}{117323} = 1.9604e - 04$$

Rounding operation

Let some number $x \in \mathbb{R}$ be given as:

$$x = \sigma m 10^c$$

where the number of digits in mantissa is equal to l. Then, the rounding of x follows from:

$$\tilde{m} = m_1 \cdot 10^{-1} + ... + m_l \cdot 10^{-l} + r$$

where $r = 0$ when $m_{l+1} \leq 4$ and 10^{-l} when $m_{l+1} \geq 5$. Following this:

$$rd(x) = \sigma \tilde{m} 10^c, \quad rd(x) \in \mathbb{M}$$

Rounding error

Thus, rounding produces error:

$$\Delta x = |x - rd(x)|$$

$$\begin{aligned} rd(0.551) &= 0.55 \Rightarrow \Delta x = 0.001 \\ rd(0.547) &= 0.55 \Rightarrow \Delta x = 0.003 \end{aligned}$$

However,

$$\begin{aligned} rd(0.551 \cdot 10^{20}) \Rightarrow 0.55 \cdot 10^{20} \Rightarrow \Delta x = 1 \cdot 10^{17} \\ rd(0.547 \cdot 10^{20}) \Rightarrow 0.55 \cdot 10^{20} \Rightarrow \Delta x = 3 \cdot 10^{17}. \end{aligned}$$

Imagine this is money!!

Floating point arithmetic

Each arithmetic operation $\star \in \{+, -, \cdot, /\}$ is substituted by machine operation $\otimes \in \{\oplus, \ominus, \odot, \oslash\}$ such that:

$$a \otimes b := rd(a \star b) = (a \star b)(1 + \epsilon), \quad |\epsilon| \leq eps$$

$$a \oplus (b \oplus c) \neq (a \oplus b) \oplus c$$

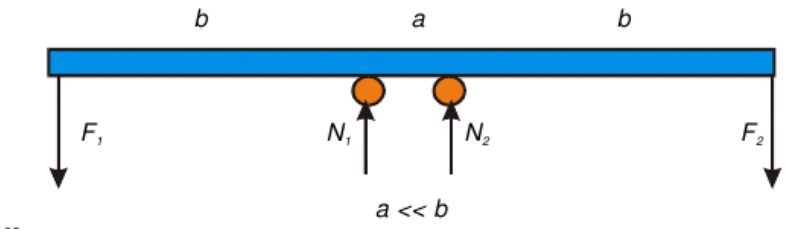
In the same way elementary functions

$$\varphi(x) \in \{sin(x), \sqrt{x}, \exp(x), ...\}$$

are substituted by machine functions:

$$\tilde{\varphi}(x) = \varphi(x)(1 + \epsilon), \quad |\epsilon| \leq eps$$

II Motivation for today's lecture:



Physics:

The beam is in equilibrium when $F_1 = F_2 = F$.

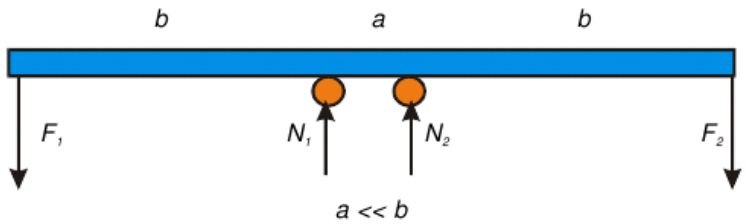
Solving the system:

$$\begin{pmatrix} F_2(a+2b) \\ F_1(a+2b) \end{pmatrix} = \begin{pmatrix} b & (a+b) \\ (a+b) & b \end{pmatrix} \begin{pmatrix} N_1 \\ N_2 \end{pmatrix}$$

reactions become

$$N_1 = N_2 = F$$

II Motivation for today's lecture:



Numerics:

It can happen that our forces F_1 and F_2 are not correct (not equal to the real ones) due to rounding, measurement, truncation or approximation errors. Then, one may face the case

$$F_1 = F(1+\varepsilon), \quad F_2 = F(1-\varepsilon)$$

For example: $F = 1, F_1 = 1.01, F_2 = 0.99 \Rightarrow \varepsilon = 10^{-2}$. Then, our system gives $N_1 = F(1+\frac{a+2b}{a}\varepsilon)$ and $N_2 = F(1-\frac{a+2b}{a}\varepsilon)$. This is different from reality especially when $a << b$. Think that $b = 10, a = 0.1$. Then, $\frac{a+2b}{a}\varepsilon = 201 \cdot 10^{-2} = 2.01$, and hence $N_1 = 3.01, N_2 = -1.01$.

Errors

Due to this reason one has to consider sensitivity of our numerics with respect to

- rounding errors
- measurement errors
- errors from earlier computation (programing and truncation errors)

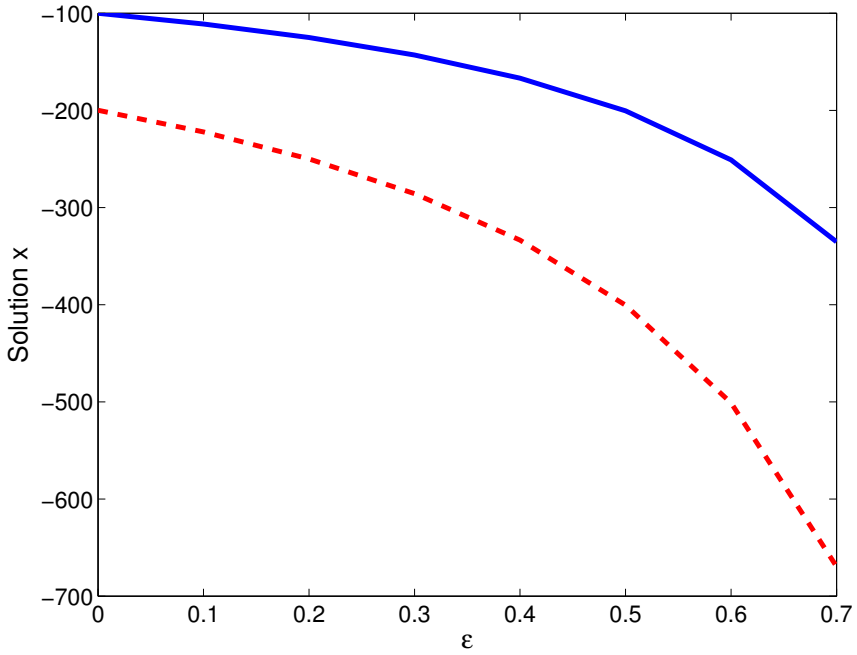
Example

By solving the system with an existing error ε

$$\begin{pmatrix} 400+\varepsilon & -201 \\ -800 & 401 \end{pmatrix} x = \begin{pmatrix} 200 \\ -200 \end{pmatrix}$$

one obtains different solutions for slight perturbations ε.

Question is whether we can trust the solution obtained by solving the system with some value of ε we do not know? For example, $400+\varepsilon$ is obtained from another computation and it is black-box for us (we do not see it, the value is obtained from another subroutine or measurement)



To answer this question one has...

to introduce the notion of sensitivity of the system on perturbations in the data, also known as **the condition of the problem**. In this respect we may distinguish

- **well-conditioned** problem if small perturbations in the data cause small perturbations in the solution
- **ill-conditioned** problem if small perturbations in the data cause large perturbations in the solution

Condition number of the function

The computer representation of a number is

$$x \Rightarrow \tilde{x}$$

We would like to evaluate $y = f(x)$. To achieve this, the computer does the following:

$$\tilde{y} = f(\tilde{x}).$$

Question: Can we tell how big is:

$$\Delta y = \tilde{y} - y$$

based on knowledge of

$$\Delta x = \tilde{x} - x \text{ and } \frac{\Delta x}{x}?$$

Condition number of the function

Let us assume that $|\Delta x|$ is small enough such that Taylor expansion

$$\begin{aligned} \tilde{y} &= f(\tilde{x}) \\ &= f(x + \Delta x) \\ &= f(x) + \frac{\partial f}{\partial x}\Delta x \\ &= y + \frac{\partial f}{\partial x}\Delta x \end{aligned}$$

holds. Thus,

$$\begin{aligned} \Delta y &= \tilde{y} - y \\ &= \frac{\partial f}{\partial x}\Delta x \end{aligned}$$

Condition number of the function

The relative error of the output is

$$\epsilon_y = \frac{\Delta y}{y} = \frac{\partial f}{\partial x}\frac{\Delta x}{y}$$

i..e.

$$\epsilon_y = \frac{\partial f}{\partial x}\frac{x}{y}\frac{\Delta x}{x}$$

$$\epsilon_y = k(x)\frac{\Delta x}{x} = k(x)\epsilon_x$$

The number

$$k(x) = \frac{\partial f}{\partial x}\frac{x}{f(x)}$$

is called condition number. This number predicts the relative error in output ϵ_y if the relative error in input ϵ_x is known.

Condition number of the function

In practice this number is often defined as:

$$k(x) = \left| \frac{\partial f}{\partial x} \frac{x}{f(x)} \right|$$

since ϵ_y and ϵ_x are very often given in terms of absolute relative errors, i.e.

$$\epsilon_y = \frac{|\Delta y|}{|y|} \text{ and } \epsilon_y = \frac{|\Delta x|}{|x|}.$$

Note: in case of the multivariate function (function which depends on the set of arguments $\mathbf{x} = \{x_1, x_2, \cdots\}$) one may introduce the condition number:

$$k_j(x) := \left| \frac{\partial f}{\partial x_j}(\mathbf{x}) \frac{x_j}{f(\mathbf{x})} \right|$$

Example

$$f(x) = \frac{x}{x-1} \text{ for } x = 0.93 \text{ is } f(x) = 13.28$$

$$f(x) = \frac{x}{x-1} \text{ for } x = 0.94 \text{ is } f(x) = 15.66$$

Slight change in argument x produces jump in function value. This means that the function is ill-conditioned in $x = 0.93$. On the other hand,

$$f(x) = \frac{x}{x-1} \text{ for } x = -0.93 \text{ is } f(x) = -0.4818$$

$$f(x) = \frac{x}{x-1} \text{ for } x = -0.94 \text{ is } f(x) = -0.4845$$

Slight change in argument x does not greatly influence the function value. This means that the function is well-conditioned in -0.93.

Example

The condition number of function $f(x) = \frac{x}{x-1}$ is

$$\begin{aligned} k(x) &= \left| \frac{\partial f}{\partial x} \frac{x}{f(x)} \right| \\ &= \left| \frac{1}{(1-x)^2} \frac{x}{\frac{x}{x-1}} \right| = \frac{1}{|1-x|} \end{aligned}$$

Thus,

$$k(0.93) = \frac{1}{|1-0.93|} = 14.28$$

$$k(-0.93) = \frac{1}{|1+0.93|} = 0.52$$

ill-conditioned part

The error in the input $x = 0.93$ is

$$\Delta x = |0.94 - 0.93| = 0.01$$

i.e.

$$\epsilon_x = \frac{\Delta x}{|x|} = \frac{0.01}{0.93} = 0.0108 \approx 1\%,$$

hence

$$\epsilon_y = k(0.93)\epsilon_x = 14.28 \cdot 0.0108 = 0.1535 \approx 15\%$$

well-conditioned part

The error in the input $x = -0.93$ is

$$\Delta x = |-0.94 + 0.93| = 0.01$$

i.e.

$$\epsilon_x = \frac{\Delta x}{|x|} = \frac{0.01}{0.93} = 0.0108 \approx 1\%,$$

hence

$$\epsilon_y = k(-0.93)\epsilon_x = 0.52 \cdot 0.0108 = 0.0056 \approx 0.5\%$$

Condition number of the system of equations

Let us assume the following problem

$$Ax = b \Rightarrow x = A^{-1}b$$

in which the matrix A is non-singular (i.e. it is invertible and has non-zero determinant). If the right hand side is slightly perturbed by Δb, then one obtains new system of equations

$$A\tilde{x} = \tilde{b} = b + \Delta b$$

$$\tilde{x} = A^{-1}(b + \Delta b) = A^{-1}b + A^{-1}\Delta b = x + \Delta x$$

Thus, the change in solution is

$$\Delta x = A^{-1}\Delta b$$

Condition number of the system of equations

To judge whether the value of Δx is small or large, let us observe

$$\Delta x = A^{-1}\Delta b$$

and use the following inequality

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

in which $\|\cdot\|$ denotes some kind of norm. Then one may conclude the following:

If $\|A^{-1}\|$ is small, then Δx will be small only if Δb is small. If $\|A^{-1}\|$ is large, then Δx can be large even if Δb is small.

Condition number of the system of equations

Following this, let us evaluate

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

From original system $Ax = b$ we have

$$\|A\|\|x\| \geq \|b\|$$

which leads

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\|A\|\|A^{-1}\|\|\Delta b\|}{\|A\|\|x\|} \leq \frac{\|A\|\|A^{-1}\|\|\Delta b\|}{\|b\|}$$

i.e.

$$\frac{\|\Delta x\|}{\|x\|} \leq \|A\|\|A^{-1}\|\frac{\|\Delta b\|}{\|b\|} = k(A)\frac{\|\Delta b\|}{\|b\|}$$

Condition number of the system of equations

The condition number $k(A)$ in

$$\frac{\|\Delta x\|}{\|x\|} \leq k(A) \frac{\|\Delta b\|}{\|b\|}$$

finally gives us the rule of checking if our system is well-conditioned or not:

- if $k(A)$ is small, then the system is well-conditioned
- otherwise, if $k(A)$ is large then the system is ill-conditioned

Condition number of the system of equations

Similar conclusion can be made if one adds perturbations ϵ_A to the matrix itself:

$$(A + \epsilon_A)\tilde{x} = \tilde{b} = b + \Delta b$$

$$(A + \epsilon_A)(x + \Delta x) = \tilde{b} = b + \Delta b$$

$$\Delta x = A^{-1}(\Delta b - \epsilon_A x - \epsilon_A \Delta x)$$

This leads to inequality

$$\|\Delta x\| = \|A^{-1}\|(\|\Delta b\| - \|\epsilon_A\|\|x\| - \|\epsilon_A\|\|\Delta x\|)$$

i.e.

$$(1 - \|A^{-1}\|\|\epsilon_A\|)\|\Delta x\| \leq \|A^{-1}\|(\|\Delta b\| - \|\epsilon_A\|\|x\|)$$

Dividing both sides by $\|x\|$ one obtains

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\epsilon_A\|}\left(\frac{\|\Delta b\|}{\|x\|} - \|\epsilon_A\|\right)$$

Condition number of the system of equations

The inequality

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\|A^{-1}\|}{1 - \|A^{-1}\|\|\epsilon_A\|} \left(\frac{\|\Delta b\|}{\|x\|} - \|\epsilon_A\| \right)$$

further can be modified as

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{\|A\|\|A^{-1}\|}{1 - \|A\|\|A^{-1}\| \frac{\|\epsilon_A\|}{\|A\|}} \left(\frac{\|\Delta b\|}{\|A\|\|x\|} - \frac{\|\epsilon_A\|}{\|A\|} \right)$$

which leads to

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{k(A)}{1 - k(A) \frac{\|\epsilon_A\|}{\|A\|}} \left(\frac{\|\Delta b\|}{\|b\|} - \frac{\|\epsilon_A\|}{\|A\|} \right)$$

Condition number of the system of equations

Assuming that the perturbation ϵ_A is small enough such that

$$\|A^{-1}\|\|\epsilon_A\| << 1$$

holds, one may simplify

$$\frac{\|\Delta x\|}{\|x\|} \leq \frac{k(A)}{1 - k(A)\frac{\|\epsilon_A\|}{\|A\|}} \left(\frac{\|\Delta b\|}{\|b\|} - \frac{\|\epsilon_A\|}{\|A\|} \right)$$

to

$$\frac{\|\Delta x\|}{\|x\|} \leq k(A) \left(\frac{\|\Delta b\|}{\|b\|} - \frac{\|\epsilon_A\|}{\|A\|} \right)$$

Hence, again the same conclusion

- if $k(A)$ is small, then the system is well-conditioned
- otherwise, if $k(A)$ is large then the system is ill-conditioned

Example

The system

$$\begin{pmatrix} 400 & -201 \\ -800 & 401 \end{pmatrix} x = \begin{pmatrix} 200 \\ -200 \end{pmatrix}$$

is ill-conditioned because

$$k(A) = \|A\|\|A^{-1}\| = 1.0006 \cdot 10^3 \cdot 2.5015 = 2.5030 \cdot 10^3$$

Hence, the system is sensitive to both perturbations in the right hand side and matrix itself.

Literature

- http://people.ds.cam.ac.uk/nmm1/arithmetic/na1.pdf
- http://homerreid.dyndns.org/teaching/18.330/Notes/MachineArithmetic.pdf
- Roland Angst. The Condition of a System of Linear Equations: Alternative Derivation
- http://www.cs.cornell.edu/~bindel/class/cs4220-s15/lec/2015-02-04-notes.pdf
- http://nucinkis-lab.cc.ic.ac.uk/HELM/workbooks/workbook_30/30_4_matrx_norms.pdf

Any questions?