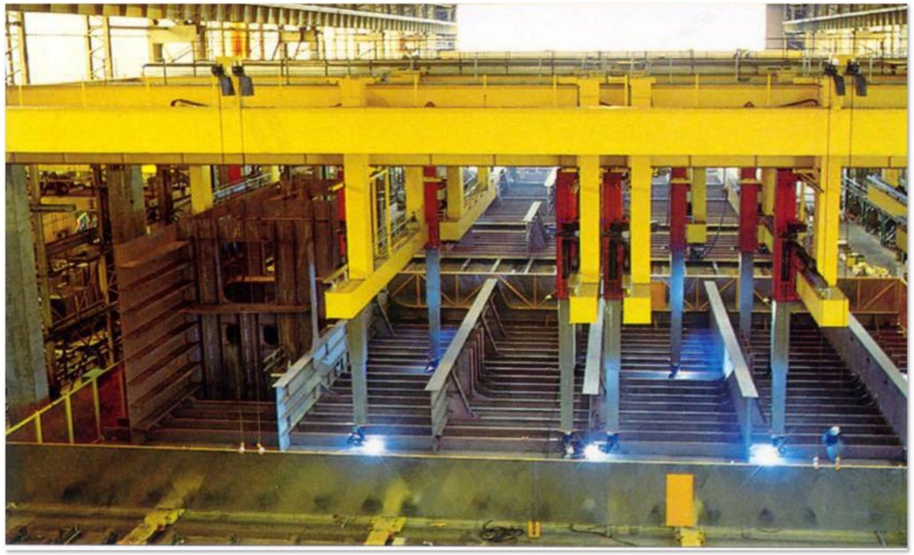
Numerical Methods

Professor, Head of Section I4.0Lab, Henrik Gordon Petersen
University of Southern Denmark
Email: hgp@mmmi.sdu.dk
URL: robotics.sdu.dk

Worked with robotics since the early 1990’s

Odense Steelshipyard (OSS)



AMROSE (Autonomous Multiple Robot Operations in Structured Environments). Project with OSS (1991-1997)

Automatic programming of welding operations and movements between weldings (up to 5 hours of execution). Based on CAD models and corrections by touching operations.

Primarily based on this collaboration, a foundation at A.P. Møller financed the establishment of the Maersk Institute. Without this, we would not have had the institute and the robotics educations.

Current work (a lot to do like my colleagues)

- ~~Responsible for a staff of 50+ people~~
- ~~Responsible for a section economy of more than 30Mkr per year~~
- Responsible for 4 large external projects
- Grant writing
- Interactions and bilateral projects with companies
- And many other things

- If you have questions, please email rather than just showing up at my office. Questions to exercises should be sent to Jens.

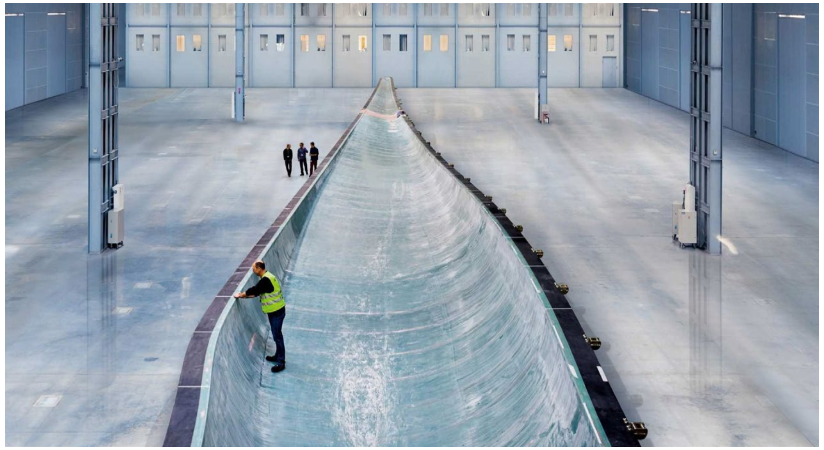
Manufacturing Academy of Denmark

Agile production systems

Today, many manufacturing processes are manual, for example product assembly. This is due to the majority of Danish manufacturing companies producing small batches of customer specific products where traditional automation solutions are not suited. To maintain competitiveness and ensure efficiency, Danish manufacturing companies want to optimize their production and utilization of human resources.

Listen to the leader of the strategic effort Agile Production Systems Henrik Gordon Petersen explain the effort.

Involved with robotics for the green transition

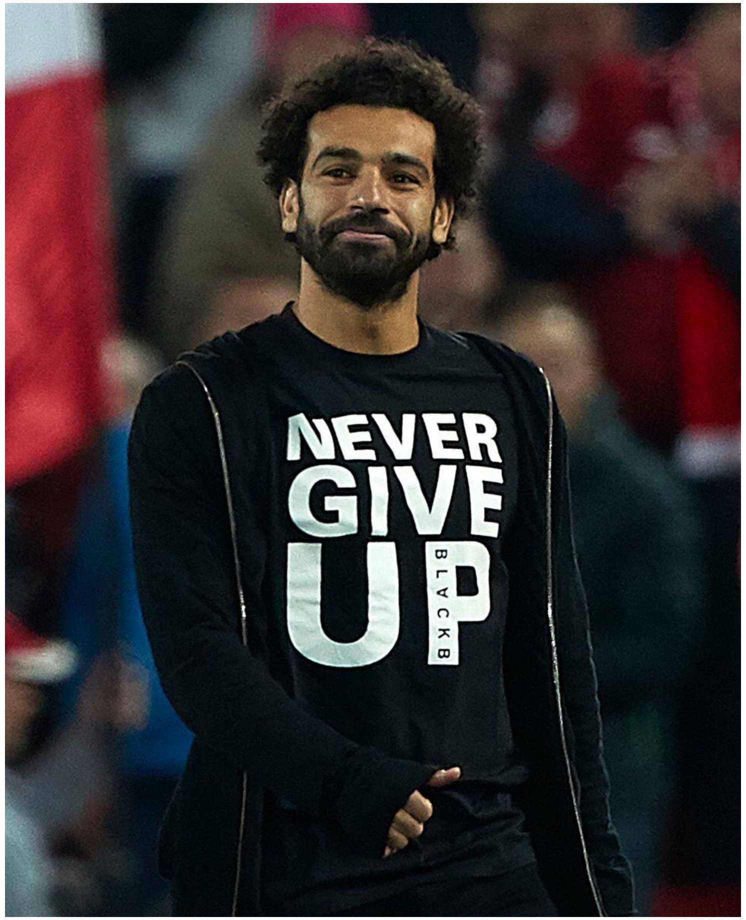


Automated production of blades for
wind turbines (with Siemens Gamesa)

Automatic disassembly of products for
component reuse (reducing material waste)

Robotic Lab Automation for deriving
new Power2X materials (Pioneer Center)

Passionated Liverpool supporter



”Never give up” attitude can bring you a lot forward as engineers and in life.

You constantly encounter troubles

- Programming troubles
- Hardware and electronics troubles
- Mathematics troubles

Never give up !!!

If you work less than 45 hours per week on your study, you are not exploiting your full potential

Numerical Methods. Subjects and goals:

Problems:

- Solving systems of linear equations
- Solving systems of non-linear equations
- Numerical integration and integral equations
- Numerical solution of ordinary differential equations
- Numerical solution of partial differential equations

For each problem, we will discuss

- examples of applications in which the problem occur
- outline the most important numerical solution methods including when to apply which method
- pitfalls in numerical solutions

All numerically generated results must have an associated estimate of the accuracy !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

The use of generative artificial intelligence (generative AI) at SDU

https://mitsdu.dk/en/mit_studie/kandidat/datalogi_kandidat/vejledning-og-support/aipaasdu

Digital tools and services based on generative artificial intelligence (generative AI), such as ChatGPT, can be a useful resource for you as a student at SDU. Below, you can read the basic guidelines on how you can use generative AI.

For the autumn semester 2023 and the corresponding exam period, SDU has decided on the following guidelines regarding the use of generative AI at SDU. The guidelines may be adjusted with effect from the start of a new semester.

Generative AI in classes

In certain courses or for certain assignments, the course description will state if the use of generative AI, such as ChatGPT is not permitted when its use may be inappropriate for the learning process.

Generative AI, including ChatGPT, Bard and others, can be useful for your learning. For example:

- review and feedback on text/language
- structuring ideas or content
- summarising large amounts of information
- brainstorming and idea generation

As with other sources or aids you use for assignments on your study programme, it is important that you follow the rules of good academic practice so that there is no doubt about who is the author of the output.

If you are allowed to use ChatGPT or other in a submitted assignment, you must specify where and how you used it. You must cite generative AI when you:

- rewrite a text that a generative AI has written (paraphrasing).
- cite or use content created by generative AI in your work (whether text, images, data, video or other).
- have used the tool, for instance, for idea generation, transcription or a literature search.

When and when not to

In more and more subjects, students will find that they have an ongoing discussion in class about how and whether AI can be used in the subject. In general, it is important to constantly consider whether it is a good idea to use AI in your subject. AI has both strengths and weaknesses. Below is a list of things you would do well to remember when using AI:

- Be critical of AI output. AI does not guarantee accuracy, so the results are not always reliable or relevant.
- Recognise that AI can be biased. Critically assess the output.
- Consider the ethical aspects of using AI.
- Avoid using mainstream AI tools like ChatGPT or Bard for article searches. They tend to combine different articles and form one source.
- Remember, you are the student, not AI. The intellectual labour has to come from you. AI should only be seen as an aid.

Principles

Generative AI and confidentiality

You should be aware that information you enter (e.g. on ChatGPT) is not treated confidentially and that generative AI may share the information you have entered with others. This requires special attention if your work involves business or research collaboration. It is your responsibility to only enter information that either has been published or may be published. SDU cannot be held responsible for your use of generative AI.

As ChatGPT is not centrally authorised by SDU due to data processing reasons, your lecturer cannot require you to use it for teaching purposes. But it may be recommended.

Generative AI for exams – on-site written exams, assignments and projects

The use of generative AI is not permitted for on-site written or oral exams. This also applies to on-site written/oral exams in which all aids are permitted.

You are allowed to use generative AI for assignments and projects, including final projects, if you follow good academic practice as described above.

If you use generative AI in violation of the guidelines, it will be considered as communication with others and thus cheating in the exam. This also applies to on-site written/oral exams in which 'all aids are permitted, unless otherwise specifically stated in the course description or exam assignments.

From the first ItsLearning announcement:

I am open to allowing you to use any other publicly available software tools for solving the exercises. However, there will be no support for using this from our side, and it is 100% your responsibility that the answers you provide are correct if you use other tools than those from Numerical Recipes.

However it will NOT be allowed to use software from other former or current Numerical Recipes students, i.e. software directly linked to the course.

System of linear equations:

$$
\begin{aligned}
a_{00}x_0 + a_{01}x_1 + a_{02}x_2 + \cdots + a_{0,N-1}x_{N-1} &= b_0 \\
a_{10}x_0 + a_{11}x_1 + a_{12}x_2 + \cdots + a_{1,N-1}x_{N-1} &= b_1 \\
a_{20}x_0 + a_{21}x_1 + a_{22}x_2 + \cdots + a_{2,N-1}x_{N-1} &= b_2 \\
\cdots \qquad\qquad\qquad\qquad & \quad \cdots \\
a_{M-1,0}x_0 + a_{M-1,1}x_1 + \cdots + a_{M-1,N-1}x_{N-1} &= b_{M-1}
\end{aligned}
$$

Here the N unknowns x_j, $j = 0, 1, \ldots, N-1$ are related by M equations. The coefficients a_{ij} with $i = 0, 1, \ldots, M-1$ and $j = 0, 1, \ldots, N-1$ are known numbers, as are the *right-hand side* quantities b_i, $i = 0, 1, \ldots, M-1$.

Matrix-vector notation:

$$
\begin{aligned}
a_{00}x_0 + a_{01}x_1 + a_{02}x_2 + \cdots + a_{0,N-1}x_{N-1} &= b_0 \\
a_{10}x_0 + a_{11}x_1 + a_{12}x_2 + \cdots + a_{1,N-1}x_{N-1} &= b_1 \\
a_{20}x_0 + a_{21}x_1 + a_{22}x_2 + \cdots + a_{2,N-1}x_{N-1} &= b_2 \\
\cdots \qquad\qquad\qquad\qquad & \cdots \\
a_{M-1,0}x_0 + a_{M-1,1}x_1 + \cdots + a_{M-1,N-1}x_{N-1} &= b_{M-1}
\end{aligned}
$$

$$
\mathbf{A} = \begin{bmatrix} a_{00} & a_{01} & \ldots & a_{0,N-1} \\ a_{10} & a_{11} & \ldots & a_{1,N-1} \\ & \cdots & & \\ a_{M-1,0} & a_{M-1,1} & \ldots & a_{M-1,N-1} \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \cdots \\ b_{M-1} \end{bmatrix}
$$

$$
\mathbf{A} \cdot \mathbf{x} = \mathbf{b}
$$

We start with: $N = M$

General Numerical Recipes notation:

$$\mathbf{C} = \mathbf{A} \cdot \mathbf{B} \quad \Longleftrightarrow \quad c_{ik} = \sum_j a_{ij} b_{jk}$$

$$\mathbf{b} = \mathbf{A} \cdot \mathbf{x} \quad \Longleftrightarrow \quad b_i = \sum_j a_{ij} x_j$$

$$\mathbf{d} = \mathbf{x} \cdot \mathbf{A} \quad \Longleftrightarrow \quad d_j = \sum_i x_i a_{ij}$$

$$q = \mathbf{x} \cdot \mathbf{y} \quad \Longleftrightarrow \quad q = \sum_i x_i y_i$$

Boldface capital letters are matrices, boldface small letters are vectors, non-boldface small letters are scalars

Review of Gaussian Elimination+Backsubstitution (example):

$$\begin{array}{rcrcrcrcr} 3x_1 & + & 4x_2 & + & x_3 & + & 2x_4 & = & 18 \\ 6x_1 & + & 6x_2 & + & 3x_3 & + & 8x_4 & = & 45 \\ 12x_1 & + & 12x_2 & + & 7x_3 & + & 18x_4 & = & 97 \\ 3x_1 & + & 0x_2 & + & 4x_3 & + & 15x_4 & = & 52 \end{array}$$

$$\left(\begin{array}{cccc|c} 3 & \boxed{4 & 1 & 2 & 18} \\ \boxed{6} & 6 & 3 & 8 & 45 \\ 12 & 12 & 7 & 18 & 97 \\ 3 & 0 & 4 & 15 & 52 \end{array}\right)$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & -4 & 3 & 10 & 25 \\ 0 & -4 & 3 & 13 & 34 \end{array}\right)$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 1 & 5 & 16 \end{array}\right)$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 0 & 3 & 9 \end{array}\right)$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 0 & 3 & 9 \end{array}\right) \quad x_4 = 9/3 = 3$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 0 & 3 & 9 \end{array}\right) \quad x_3 = (7 - 2x_4)/1 = (7 - 6)/1 = 1$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 0 & 3 & 9 \end{array}\right) \quad x_2 = (9 - x_3 - 4x_4)/(-2) = (9 - 1 - 12)/(-2) = 2$$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -2 & 1 & 4 & 9 \\ 0 & 0 & 1 & 2 & 7 \\ 0 & 0 & 0 & 3 & 9 \end{array}\right) \quad x_1 = (18 - 4x_2 - x_3 - 2x_4)/3 = (18 - 8 - 1 - 6)/3 = 1$$

LU decomposition:

Suppose we are able to write the matrix $\mathbf{A}$ as a product of two matrices,

$$\mathbf{L} \cdot \mathbf{U} = \mathbf{A} \tag{2.3.1}$$

where $\mathbf{L}$ is *lower triangular* (has elements only on the diagonal and below) and $\mathbf{U}$ is *upper triangular* (has elements only on the diagonal and above). For the case of a 4×4 matrix $\mathbf{A}$, for example, equation (2.3.1) would look like this:

$$\begin{bmatrix} \alpha_{00} & 0 & 0 & 0 \\ \alpha_{10} & \alpha_{11} & 0 & 0 \\ \alpha_{20} & \alpha_{21} & \alpha_{22} & 0 \\ \alpha_{30} & \alpha_{31} & \alpha_{32} & \alpha_{33} \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \beta_{01} & \beta_{02} & \beta_{03} \\ 0 & \beta_{11} & \beta_{12} & \beta_{13} \\ 0 & 0 & \beta_{22} & \beta_{23} \\ 0 & 0 & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\mathbf{L} \cdot \mathbf{y} = \mathbf{b}$$

$$\mathbf{A} \cdot \mathbf{x} = (\mathbf{L} \cdot \mathbf{U}) \cdot \mathbf{x} = \mathbf{L} \cdot (\mathbf{U} \cdot \mathbf{x}) = \mathbf{b}$$

$$\alpha_{ii} \equiv 1 \qquad i = 0, \ldots, N-1$$

$$\mathbf{U} \cdot \mathbf{x} = \mathbf{y}$$

Solve for L and U:

$$\begin{bmatrix} \boxed{1 \quad 0 \quad 0 \quad 0} \\ \alpha_{10} \quad 1 \quad 0 \quad 0 \\ \alpha_{20} \quad \alpha_{21} \quad 1 \quad 0 \\ \alpha_{30} \quad \alpha_{31} \quad \alpha_{32} \quad 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \beta_{01} & \beta_{02} & \beta_{03} \\ 0 & \beta_{11} & \beta_{12} & \beta_{13} \\ 0 & 0 & \beta_{22} & \beta_{23} \\ 0 & 0 & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} \boxed{a_{00}} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\beta_{00} = a_{00}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha_{10} & 1 & 0 & 0 \\ \alpha_{20} & \alpha_{21} & 1 & 0 \\ \alpha_{30} & \alpha_{31} & \alpha_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \beta_{01} & \beta_{02} & \beta_{03} \\ 0 & \beta_{11} & \beta_{12} & \beta_{13} \\ 0 & 0 & \beta_{22} & \beta_{23} \\ 0 & 0 & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ \boxed{a_{10}} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\alpha_{10} = \frac{a_{10}}{\beta_{00}}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha_{10} & 1 & 0 & 0 \\ \alpha_{20} & \alpha_{21} & 1 & 0 \\ \alpha_{30} & \alpha_{31} & \alpha_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \beta_{01} & \beta_{02} & \beta_{03} \\ 0 & \beta_{11} & \beta_{12} & \beta_{13} \\ 0 & 0 & \beta_{22} & \beta_{23} \\ 0 & 0 & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ \boxed{a_{20}} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\alpha_{20} = \frac{a_{20}}{\beta_{00}}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha_{10} & 1 & 0 & 0 \\ \alpha_{20} & \alpha_{21} & 1 & 0 \\ \alpha_{30} & \alpha_{31} & \alpha_{32} & 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \beta_{01} & \beta_{02} & \beta_{03} \\ 0 & \beta_{11} & \beta_{12} & \beta_{13} \\ 0 & 0 & \beta_{22} & \beta_{23} \\ 0 & 0 & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ \boxed{a_{30}} & a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$\alpha_{30} = \frac{a_{30}}{\beta_{00}}$$

$$\begin{bmatrix} \boxed{1 \quad 0 \quad 0 \quad 0} \\ \alpha_{10} \quad 1 \quad 0 \quad 0 \\ \alpha_{20} \quad \alpha_{21} \quad 1 \quad 0 \\ \alpha_{30} \quad \alpha_{31} \quad \alpha_{32} \quad 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \boxed{\begin{matrix}\beta_{01}\\ \beta_{11}\\ 0\\ 0\end{matrix}} & \beta_{02} & \beta_{03} \\ 0 & & \beta_{12} & \beta_{13} \\ 0 & & \beta_{22} & \beta_{23} \\ 0 & & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & \boxed{a_{01}} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \beta_{01} = a_{01}$$

$$\begin{bmatrix} 1 \quad 0 \quad 0 \quad 0 \\ \boxed{\alpha_{10} \quad 1 \quad 0 \quad 0} \\ \alpha_{20} \quad \alpha_{21} \quad 1 \quad 0 \\ \alpha_{30} \quad \alpha_{31} \quad \alpha_{32} \quad 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \boxed{\begin{matrix}\beta_{01}\\ \beta_{11}\\ 0\\ 0\end{matrix}} & \beta_{02} & \beta_{03} \\ 0 & & \beta_{12} & \beta_{13} \\ 0 & & \beta_{22} & \beta_{23} \\ 0 & & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & \boxed{a_{11}} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \beta_{11} = a_{11} - \alpha_{10}\beta_{01}$$

$$\begin{bmatrix} 1 \quad 0 \quad 0 \quad 0 \\ \alpha_{10} \quad 1 \quad 0 \quad 0 \\ \boxed{\alpha_{20} \quad \alpha_{21} \quad 1 \quad 0} \\ \alpha_{30} \quad \alpha_{31} \quad \alpha_{32} \quad 1 \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \boxed{\begin{matrix}\beta_{01}\\ \beta_{11}\\ 0\\ 0\end{matrix}} & \beta_{02} & \beta_{03} \\ 0 & & \beta_{12} & \beta_{13} \\ 0 & & \beta_{22} & \beta_{23} \\ 0 & & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & \boxed{a_{21}} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix} \qquad \alpha_{21} = \frac{a_{21} - \alpha_{20}\beta_{01}}{\beta_{11}}$$

$$\begin{bmatrix} 1 \quad 0 \quad 0 \quad 0 \\ \alpha_{10} \quad 1 \quad 0 \quad 0 \\ \alpha_{20} \quad \alpha_{21} \quad 1 \quad 0 \\ \boxed{\alpha_{30} \quad \alpha_{31} \quad \alpha_{32} \quad 1} \end{bmatrix} \cdot \begin{bmatrix} \beta_{00} & \boxed{\begin{matrix}\beta_{01}\\ \beta_{11}\\ 0\\ 0\end{matrix}} & \beta_{02} & \beta_{03} \\ 0 & & \beta_{12} & \beta_{13} \\ 0 & & \beta_{22} & \beta_{23} \\ 0 & & 0 & \beta_{33} \end{bmatrix} = \begin{bmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & \boxed{a_{31}} & a_{32} & a_{33} \end{bmatrix} \qquad \alpha_{31} = \frac{a_{31} - \alpha_{30}\beta_{01}}{\beta_{11}}$$

For each $j = 0, 1, 2, \ldots, N-1$ do these two procedures:

for $i = 0, 1, \ldots, j$ $$\beta_{ij} = a_{ij} - \sum_{k=0}^{i-1} \alpha_{ik}\beta_{kj}$$

for $i = j+1, j+2, \ldots, N-1$ $$\alpha_{ij} = \frac{1}{\beta_{jj}} \left(a_{ij} - \sum_{k=0}^{j-1} \alpha_{ik}\beta_{kj} \right)$$

What if $\beta_{jj} = 0$? (also a problem for a "small" β_{jj})

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & 0 & \boxed{1 & 4 & 9} \\ 0 & -4 & 3 & 10 & 25 \\ 0 & -4 & 3 & 13 & 34 \end{array}\right)$$

Swap rows $\updownarrow$

$$\left(\begin{array}{cccc|c} 3 & 4 & 1 & 2 & 18 \\ 0 & -4 & 3 & 10 & 25 \\ 0 & 0 & 1 & 4 & 9 \\ 0 & -4 & 3 & 13 & 34 \end{array}\right)$$

Pivoting (row swaps) like in Gaussian elimination. Terrible to bookkeep, but implemented in the Numerical Recipes routines Ludcmp.

Review of Gaussian Elimination+Backsubstitution (example):

$$
\begin{array}{rcrcrcrcr}
3x_1 & + & 4x_2 & + & x_3 & + & 2x_4 & = & 18 \\
6x_1 & + & 6x_2 & + & 3x_3 & + & 8x_4 & = & 45 \\
12x_1 & + & 12x_2 & + & 7x_3 & + & 18x_4 & = & 97 \\
3x_1 & + & 0x_2 & + & 4x_3 & + & 15x_4 & = & 52
\end{array}
$$

$$
\left(\begin{array}{cccc|c}
3 & \boxed{4 & 1 & 2 & 18} \\
6 & 6 & 3 & 8 & 45 \\
12 & 12 & 7 & 18 & 97 \\
3 & 0 & 4 & 15 & 52
\end{array}\right)
$$

$$
\left(\begin{array}{cccc|c}
3 & 4 & 1 & 2 & 18 \\
0 & -2 & 1 & 4 & 9 \\
0 & -4 & 3 & 10 & 25 \\
0 & -4 & 3 & 13 & 34
\end{array}\right)
$$

$$
\left(\begin{array}{cccc|c}
3 & 4 & 1 & 2 & 18 \\
0 & -2 & 1 & 4 & 9 \\
0 & 0 & 1 & 2 & 7 \\
0 & 0 & 1 & 5 & 16
\end{array}\right)
$$

$$
\left(\begin{array}{ccccc}
3 & 4 & 1 & 2 & 18 \\
0 & -2 & 1 & 4 & 9 \\
0 & 0 & 1 & 2 & 7 \\
0 & 0 & 0 & 3 & 9
\end{array}\right)
$$

LU-decomposition:

$$
\begin{pmatrix}
3 & 4 & 1 & 2 \\
6 & 6 & 3 & 8 \\
12 & 12 & 7 & 18 \\
3 & 0 & 4 & 15
\end{pmatrix}
=
\begin{pmatrix}
1 & 0 & 0 & 0 \\
2 & 1 & 0 & 0 \\
4 & 2 & 1 & 0 \\
1 & 2 & 1 & 1
\end{pmatrix}
\boxed{\begin{pmatrix}
3 & 4 & 1 & 2 \\
0 & -2 & 1 & 4 \\
0 & 0 & 1 & 2 \\
0 & 0 & 0 & 3
\end{pmatrix}}
$$

What about L?

Computational complexity (big O() notation)

LU decomposition:

For each $j = 0, 1, 2, \ldots, N-1$ do these two procedures:

$$\text{for } i = 0, 1, \ldots, j \qquad \beta_{ij} = a_{ij} - \sum_{k=0}^{i-1} \alpha_{ik}\beta_{kj}$$

O(N^3)

$$\text{for } i = j+1, j+2, \ldots, N-1 \qquad \alpha_{ij} = \frac{1}{\beta_{jj}}\left(a_{ij} - \sum_{k=0}^{j-1} \alpha_{ik}\beta_{kj}\right)$$

Forward substitution using L (solve Ly=b):

$$y_0 = \frac{b_0}{\cancel{\alpha_{00}}}$$

$$y_i = \cancel{\frac{1}{\alpha_{ii}}}\left[b_i - \sum_{j=0}^{i-1} \alpha_{ij} y_j\right] \qquad i = 1, 2, \ldots, N-1$$

O(N^2)

Back substitution using U (solve Ux=y):

$$x_{N-1} = \frac{y_{N-1}}{\beta_{N-1,N-1}}$$

$$x_i = \frac{1}{\beta_{ii}}\left[y_i - \sum_{j=i+1}^{N-1} \beta_{ij} x_j\right] \qquad i = N-2, N-3, \ldots, 0$$

O(N^2)

Gaussian elimination is also O(N^3)

Advantage of LU decomposition when comparing to Gaussian elimination?

Assume we have to solve a system of linear equations with the **same matrix** and a **new right hand side.**

Then we only need to do forward and back substitutions (assuming we stored L and U). Hence O(N^2) rather than O(N^3).

Exercises

1) Consider an NxN matrix A where we have computed L and U. How can we then easily compute the determinant of A ?
2) Assume that we have 2 NxN matrices A and B, and we have LU-decompositions A=LU and B=L'U'. Let now C=AB. How can we solve Cx=b and what is the resulting computational complexity ?
3) Compute by hand the LU decomposition of

$$A = \begin{pmatrix} 1 & 3 & 5 \\ -2 & 0 & -1 \\ 2 & 3 & 1 \end{pmatrix}$$

4) Programming exercise using the NR routine *Ludcmp* (Jens).