Advanced Numerical Analysis

Lecture Notes for the Sommersemester 2023

by

Lukas **Exl**, Norbert J. **Mauser**, Hans Peter **Stimming**

Research platform MMM ”Mathematics-Magnetism-Materials”
and
Fak. f. Mathematik, Univ. Wien

This course is designed for the Master of Mathematics at Univ. Wien, it fits also for the Master Computational Sciences and the Master Data Science, as well as for master and PhD students in all MINT fields with a focus on numerics, like Computational Physics, Computational Astrophysics / Geology / Meteorology, Informatics, etc.
The prerequisite for this course is a sound understanding of analyis and linear algebra, and a basic numerics course containing standard subjects like numerical linear algebra etc. Good students can learn that on the fly in this course using e.g. the introductory lecture notes provided by the teachers.
The exercise classes in parallel to the lecture are strongly recommended since understanding numerical methods is hard without practical application.

1 Iterative methods for large linear systems

Iterative methods try to find an approximate solution $x \in \mathbb{R}^n$ to the linear equation $Ax = b$, where $A \in \mathbb{R}^{n\times n}$ is typically a *sparse matrix* and n a very large integer. In these cases, so-called iterative methods might be preferred over direct methods (which are based on matrix factorization) because their iterations mainly rely on efficient (sparse) matrix-vector evaluation. Large and sparse matrices appear frequently in the applied sciences e.g. as finite difference or stiffness matrices in discretizations of partial differential equations (PDEs).

An extensive discourse of the topic is given in the book of Y. Saad [1]. We will cover the most important aspects of this topic including the most insightful proofs.

With increasing computational power larger numerical problems can be tackled. Many of these problems ultimately reduce to or rely on the task of solving (many) systems of linear equations of large sizes, e.g. $n > 10^7$. Direct solution of such large systems are almost never efficient or even intractable. However, if special structure or, specifically, sparsity (many matrix entries are zero) can be imposed, iteration methods can tackle such tasks. An easy model example for the occurrence of large and sparse linear systems is the finite difference or finite element discretization of PDEs such as the Poisson equation. We will show these examples in the next introductory section of this chapter. They already involve important features like sparsity but also an typically ill-conditioned system matrix. Also many other (spatial) discretizations of continuous problems in mathematical sciences ultimately reduce to subproblems which are large linear systems. Often linear systems occur as local approximations to nonlinear problems (like in nonlinear equations, optimization or differential equations), where linearizations often yield approximate subproblems whose successive solution yields an approximation to the original problem. The key subproblem is often a large (sparse) linear system which occurs each iteration, and hence, has to be solved many times for the approximate solution of the main problem. It is therefore very important to be able to efficiently solve the linear systems.
Depending on features of the system matrix A different methods and versions of methods yield appropriate efficiency. Such features can be the sparsity pattern, condition number, matrix properties like symmetry or spectrum (e.g. positive eigenvalues). An important consideration in a concrete application case might also be the desired error of the approximate solution and its definition or measurement, e.g. via residual (minimum residual methods) or absolute/relative error.
However, the unprepared use of a certain iterative method for solving linear systems might not be effective or even be practically impossible because of a too large condition number of the system matrix. Therefore, so-called preconditioning has to be considered. We will introduce basic techniques for preconditioning linear systems in the end of this chapter.

1.1 Introductory examples

We briefly show the occurrence of large sparse linear systems in a model problem, which here shall be the Poisson problem in two dimensions with Dirichlet boundary conditions on $\Omega = (0,1)^2$. This is the following equation for the scalar function $u = u(x,y)$ induced by the density $f = f(x,y)$:

$$\begin{aligned} -\Delta u &= f, \\ u_{|\partial\Omega} &= 0. \end{aligned} \tag{1}$$

A finite difference method would try to solve the above differential equation on an equidistant grid defined by the grid points (nodes) $x_{ij} = (x_i, y_j) = (ih, jh),\ i,j = 0\ldots n+1$ with the grid spacing $h = 1/(n+1)$. The Laplace operator $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is approximated by a (second order) finite difference formula. Incorporating the given values on the boundary (zeros) this yields the approximations u_{ij} to the values $u(x_i, x_j)$ according to the solution of

$$4\,u_{ij} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1} = h^2\, f_{ij} \quad i,j = 1\ldots n, \tag{2}$$

where $f_{ij} = f(x_i, x_j)$.
The equations (2) can be written in matrix form $A\widetilde{u} = b$ by (e.g.) 'lexicographic' re-ordering of the

involved variables: make 'long vectors' by setting $\widetilde{u}_{i+(j-1)n} = u_{ij}$ and $b_{i+(j-1)n} = h^2 f_{ij}$. The matrix $A \in \mathbb{R}^{N\times N}$ with $N = n^2$ (K_{2d} matrix) is sparse (see Figure 1) and can be generated by so-called Kronecker products $A = K_{2d} := K_{1d} \otimes I + I \otimes K_{1d} \in \mathbb{R}^{N\times N}$, where $K_{1d} \in \mathbb{R}^{n\times n}$ is the tridiagonal matrix arising from the central difference quotient, i.e., $K_{1d} = \text{tridiag}(-1, 2, -1)$ and $I \in \mathbb{R}^{n\times n}$ the identity (diag(1)). The matrix A has 4's in the diagonal, -1's in off diagonals and is sparse (many zeros). It is also symmetric and positive definite (positive eigenvalues), commonly abbreviated by SPD. The eigenvalues satisfy $\lambda_{min} = 8 \sin^2(\frac{\pi}{2}h) = O(h^2)$ and $\lambda_{max} = 8\cos^2(\frac{\pi}{2}\,h) = O(1)$ and hence, the condition number is $\kappa_2 = O(h^{-2})$.

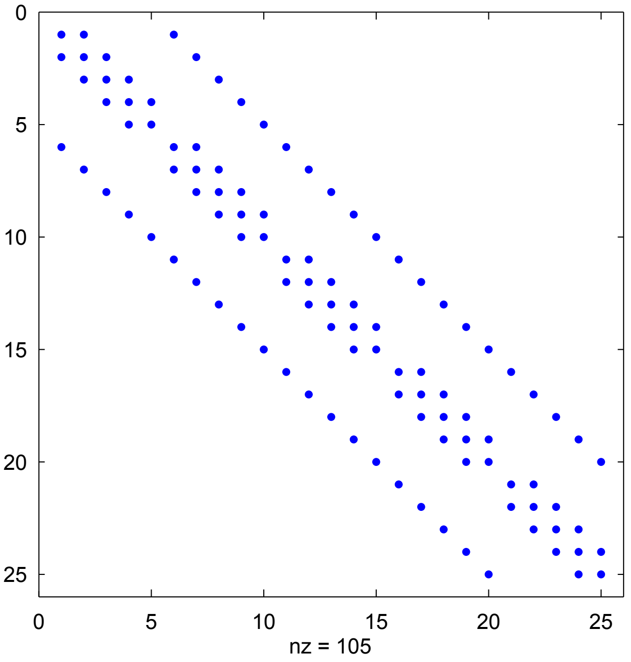

Figure 1: Sparsity pattern of K_{2d} matrix

It is a typical example of a matrix (arising from a (spatial) discretization) which gets increasingly ill-conditioned as the original continuous problem gets more and more resolved (h smaller). We remark that A is a so-called Toeplitz matrix, that is a (band) matrices with constant entries along the (off-) diagonal. These matrices have eigenvalues which can be expressed by trigonometric functions and systems which involve Toeplitz matrices can be efficiently solved by FFT methods. Anyhow, the special structure normally gets lost as soon as more general geometries (and/or boundary conditions) or variable coefficients in the differential equation are considered.

Sparse systems also arise in so-called finite element methods (FEM) which are often used e.g. in engineering to numerically compute problems formulated as differential equations on more general geometries. The arising sparsity of the systems is the key consideration of the choice of FE basis functions as opposed to the more global Ritz/Galerkin methods of which FEM is a special version. We briefly illustrate the so-called variational procedure of FEM by a simple example in one dimension. The key derivation is similar in higher dimensions and general geometries. Consider the boundary value problem

$$-u'' = f, \quad u(0) = u(1) = 0.$$

The variational scheme now consists of multiplying both sides of the equation with functions which incorporate the boundary conditions and followed by integration (by parts):

$$a(u,v) := \int_0^1 u'(x)v'(x)\,dx = \int_0^1 f(x)v(x)\,dx =: b(v)$$
$$v\,:\,v(0) = v(1) = 0$$

Now the solution is approximated by an ansatz in a finite dimensional space X_n, e.g. piecewise linear, $u^*(x) \approx \sum_{j=1}^n c_j\,\varphi_j(x)$ $(\varphi_j \in X_n)$. Inserting the ansatz into the variational form and varying $v = \varphi_i$, $i = 1,\dots,n$ in X_n yields a linear system $Ac = b$ for the coefficients

$$\sum_{j=1}^n \underbrace{a(\varphi_i,\varphi_j)}_{=:a_{ij}\hookrightarrow A\in\mathbb{R}^{n\times n}} \underbrace{c_j}_{\hookrightarrow c\in\mathbb{R}^n} = \underbrace{b(\varphi_i)}_{=:b_i\hookrightarrow b\in\mathbb{R}^n}\,, \quad i = 1,\dots,n. \tag{3}$$

In the FEM the approximation and ansatz space X_n is chosen in a way such that the values of the bilinear form $a(\varphi_i,\varphi_j)$, and hence the matrix A, vanish most of the time. It is usually called the *stiffness matrix.*

1.2 Sparse formats

Here we consider a few of the straightforward formats for sparse matrix storage. The reason is twofold: (i) the sparse storage decreases memory requirements, (ii) the frequently occurring matrix-vector multiplication can be treated in an effective way.

1.2.1 Coordinate format

The coordinate format consists of three arrays (see figure 2):

- AA: an array of nonzero elements of the matrix A,
- IR: row indices array (integers) of nonzero elements,
- IC: column indices array (integers) of nonzero elements.

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 4 & -2 & 0 & 0 \\ 0 & -2 & 4 & -3 & 0 \\ 0 & 0 & -3 & 6 & -2 \\ 0 & 0 & 0 & -2 & 4 \end{pmatrix}$$

$$\begin{array}{lcl} \text{AA} & = & [2 \;\; -1 \;\; -1 \;\; 4 \;\; -2 \;\; -2 \;\; 4 \;\; -3 \;\; -3 \;\; 6 \;\; -2 \;\; -2 \;\; 4] \\ \text{IR} & = & [1 \;\; 1 \;\; 2 \;\; 2 \;\; 2 \;\; 3 \;\; 3 \;\; 3 \;\; 4 \;\; 4 \;\; 4 \;\; 5 \;\; 5] \\ \text{IC} & = & [1 \;\; 2 \;\; 1 \;\; 2 \;\; 3 \;\; 2 \;\; 3 \;\; 4 \;\; 3 \;\; 4 \;\; 5 \;\; 4 \;\; 5] \end{array}$$

Figure 2: Example for coordinate format taken from [3].

1.2.2 Compressed Sparse Row (CSR) and Column (CSC) formats

More compression than in the coordinate form is achieved by only storing integer pointers for the positions where a row (resp. column) starts, since the nonzeros can be arranged in the nonzero-array row-by-row (resp. column-by-column).
The CSR format consists of the following three arrays (CSC is analogue), compare with figure 3 :

- AA: an array of nonzero elements of the matrix $A \in \mathbb{R}^{n\times n}$,

$$A = \begin{pmatrix} 2 & -1 & 0 & 0 & 9 \\ -1 & 4 & -2 & 0 & 0 \\ 0 & -2 & 4 & -3 & 0 \\ 0 & 0 & -3 & 6 & -2 \\ 0 & 0 & 0 & -2 & 4 \end{pmatrix}$$

$$\begin{array}{llcccccccccccccccl}
\text{AA} & = [& 2 & -1 & 9 & -1 & 4 & -2 & -2 & 4 & -3 & -3 & 6 & -2 & -2 & 4 &] & \\
\text{PR} & = [& 1 & & & 4 & & & 7 & & & 10 & & & 13 & & & 15] \\
\text{IC} & = [& 1 & 2 & 5 & 1 & 2 & 3 & 2 & 3 & 4 & 3 & 4 & 5 & 4 & 5 &] &
\end{array}$$

Figure 3: Example for compressed sparse row (CSR) format taken from [3].

- IC: column indices array (integers) of nonzero elements,
- PR: array of $n+1$ integers. Entry $PR(i)$ points to positions in AA and IC where the i-th row of A starts. Nonzeros of i-th row are $AA(PR(i) : PR(i+1)-1)$ and their column indices are $IC(PR(i) : PR(i+1)-1)$. The last entry $PR(n+1) = nz+1$, where nz is the number of nonzeros.

A description of matrix-vector multiplication, which efficiently uses the CSR format is given in Alg. 1.

Algorithm 1: Matrix vector multiplication in the compressed sparse row format.

```
Data: CSR for $A \in \mathbb{R}^{n \times n}$: $AA, IC, PR$, vector $x \in \mathbb{R}^n$
Result: vector $y \in \mathbb{R}^n$
for $i = 1 : n$ do
    $k_1 = PR(i)$
    $k_2 = PR(i+1) - 1$
    $y(i) = AA(k_1 : k_2) * x(IC(k_1 : k_2))^T$
end
```

1.3 Basic iterative methods

In the following we will describe basic methods, which aim at solving a given linear system $Ax = b$ with invertible matrix $A \in \mathbb{R}^{n \times n}$ (can also be $\mathbb{C}$) iteratively. The matrices can be considered to be large, e.g. $n > 10^7$, which often occurs in applications. The iteration methods will take the following fixed point form:

$$x_{k+1} = \Phi(x_k, b) \quad \text{starting at some } x_0. \tag{4}$$

Iterative methods are designed to be faster than direct methods in the case that iterations are cheaply computable. This is the case if the involved matrix-vector products are cheap, since these are the main operations in iterative methods for linear systems.

In practice, one would first choose a certain available numerical method, ideally based on mathematical analysis or other understanding of the problem, and then prescribe an approximation accuracy, which should be reached for some error measure (stopping criterion). The 'converged solution' is only an approximation to the true solution of the nonsingular system. Practitioners still might be satisfied with this situation since direct solution (exact within computer accuracy) simply might be too expensive.

We emphasize that although the requirements for convergence might be perfectly fulfilled theoretically in a concrete problem case, the practical performance still depends on e.g. condition number, preconditioning of the problem, the chosen numerical method, starting point, stability of the algorithm and its computer realization, error measurement, size of the problem and other factors.

We will start with a very brief theoretical background on so-called linear fixed point iterations, which are the theoretical basis for the splitting methods described afterwards.

1.3.1 Linear fixed point iteration and convergence

We assume here a linear system

$$Ax = b, \tag{5}$$

with invertible matrix $A \in \mathbb{R}^{n\times n}$ or $\mathbb{C}^{n\times n}$. Many iterative methods for solving (5) have the form of a so-called *fixed point iteration*

$$x_{k+1} = \Phi(x_k, b), \tag{6}$$

for some function Φ (iteration mapping), see Alg. 2.

```
Algorithm 2: (Stationary) iterative methods
  Data: Matrix $A \in \mathbb{R}^{n\times n}$ nonsing., r.h.s. $b \in \mathbb{R}^n$, initial vector $x_0 \in \mathbb{R}^n$
  Result: $x \in \mathbb{R}^n$ approximate solution
  Initialization: Set $x = x_0$
  while Convergence criterion not satisfied do
      $x \leftarrow \Phi(x, b)$   with $\Phi$ according to chosen method
  end
```

Definition 1.1 (fixed point iteration). *Solutions to the equation $x = \Phi(x, b)$ are called fixed points of* (6)*. The fixed point iteration is*

- ***consistent**, if for every b the solution $x_* = A^{-1}b$ is a fixed point,*
- ***(globally) convergent**, if for every b there is a x_* such that for every starting vector x_0 the sequence $(x_k)_{k\in\mathbb{N}}$ defined by the iteration* (6) *converges to x_*,*
- ***linear**, if the iteration mapping has the special form*

  $$\Phi(x, b) = Mx + Nb \tag{7}$$

  the matrix M is called iteration matrix,
- ***symmetric**, if it is linear, the matrix A symmetric and N is SPD.*

We will consider linear fixed point iterations in the following.

Theorem 1.2. *The linear fixed point iteration* (7) *is consistent if and only if there holds*

$$M = I - NA. \tag{8}$$

Proof. Exercise. □

The convergence of the linear fixed point iteration (7) depends on the iteration matrix M. This can be seen by the sufficient contraction condition for unique convergence of the *Banach fixed point theorem* which is $\|\Phi(x) - \Phi(y)\| \leq q\|x - y\|$ for $q < 1$ for all $x, y \in \mathbb{R}^n$. In the present case we have $\|M(x-y)\| \leq \|M\|\|x - y\| \leq q\|x - y\|$, which shows that $\|M\| < 1$ guarantees convergence.
In fact this statement can be strengthened [1] to

Theorem 1.3. *Let $\rho(M) < 1$. Then there exists a unique fixed point x_* of* (6) *(with Φ from* (7)*) and the iteration* (6) *converges to x_* for any starting value x_0.*

Lemma 1.4. *Let $A \in \mathbb{K}^{n\times n}$, where $\mathbb{K} = \mathbb{R}$ or $\mathbb{C}$. Then, there holds:*

(i) $\rho(A^m) = \rho(A)^m$ for all $m \in \mathbb{N}$,

[1] $\rho(M) \leq \|M\|$, only for normal M we have $\rho(M) = \|M\|_2$.

(ii) For any norm $\|.\|$ on $\mathbb{K}^n$ we have $\rho(A) \leq \|A\|$,

(iii) For every $\varepsilon > 0$ there exists a norm $\|.\|_\varepsilon$ on the vector space $\mathbb{K}^n$ such that

$$\rho(A) \leq \|A\|_\varepsilon \leq \rho(A) + \varepsilon,$$

(iv) For any norm $\|.\|$ on $\mathbb{K}^n$ we have $\rho(A) = \lim_{m\to\infty} \|A^m\|^{1/m}$. □

Proof Th. 1.3. Since $\rho(M) < 1$, we conclude that $I - M$ is regular ($1 \notin \sigma(M)$), and so the fixed point equation $x = Mx + Nb$ has a unique solution x_*. When we now subtract the fixed point equation and the linear iteration $x_{k+1} = Mx_k + Nb$, denoting the error of the k-th iterate with $\varepsilon_k = x_* - x_k$, we get

$$\varepsilon_{k+1} = M\varepsilon_k = M^2\varepsilon_{k-1} = \ldots = M^{k+1}\varepsilon_0. \tag{9}$$

By Lemma (1.4) we can define a norm $\|.\|_\epsilon$ such that $\|M\|_\epsilon \leq \rho(M) + \epsilon < 1$, where we get

$$\|\varepsilon_{k+1}\|_\epsilon = \|M^{k+1}\varepsilon_0\|_\epsilon \leq \|M^{k+1}\|_\epsilon \|\varepsilon_0\|_\epsilon \leq \|M\|_\epsilon^{k+1}\|\varepsilon_0\|_\epsilon \to 0, \quad \text{as } k \to \infty.$$

This convergence is true in any norm, since every norm is equivalent in $\mathbb{K}^n$ ($\mathbb{K} = \mathbb{C}$ or $\mathbb{R}$). □

Corollary 1.5. *Let the linear fixed point iteration* (6) *(with Φ from* (7)*) be consistent with the invertible matrix A and $\rho(M) < 1$. Then for every starting point the iteration converges to the solution of $Ax = b$.* □

We remark that the invertibility of N is necessary for convergence, because otherwise (by consistency) the matrix $I - M = NA$ has an eigenvalue $\lambda = 0$, which contradicts $\rho(M) < 1$.

We also have that $\rho(M) < 1$ is necessary for convergence:

Theorem 1.6 (Proof [1, 3]). *Consider the linear fixed point iteration* (6) *with Φ from* (7)*. Let further $\rho(M) \geq 1$. Then the iteration* (6) *does not converge, i.e., either there are starting vectors x_0 and r.h.s. b such that the sequence (x_k) is not convergent or the iterates corresponding to different starting values converge to different fixed points.* □

Ad Proof. We can choose $b = 0$ and pick $\lambda \in \sigma(M)$ with $|\lambda| = \rho(M) \geq 1$ with associated eigenvector $x_0 \neq 0$. Then we have $x_k = M^k x_0 = \lambda^k x_0$. If $|\lambda| > 1$ then $|x_k| \to \infty$. In the case $|\lambda| = 1$ with $\lambda \neq 1$ we have $\lambda = e^{i\varphi}$ with $\varphi \in (0, 2\pi)$ and $x_k = e^{ik\varphi}x_0$, which is not converging (no Cauchy sequence). In the final case of $\lambda = 1$ we get the trivial iterates $x_k \equiv x_0$ which 'converge' to different limits as the starting value would change. □

The contraction property, used in convergence studies of fixed point iterations, indicates a reduction of the error by a factor $q \in (0, 1)$ each iteration. Since the norm of the error of the k-th iterate $\varepsilon_k = x_* - x_k$ is $\|\varepsilon_k\| = \|M^k\varepsilon_0\| \leq \|M^k\|\|\varepsilon_0\|$, the quantity $q := \limsup_{k\to\infty}(\|\varepsilon_k\|/\|\varepsilon_0\|)^{1/k} = \limsup_{k\to\infty}\|M^k\|^{1/k} = \rho(M)$ determines the (asymptotic) error reduction each step (convergence factor).
It should be mentioned here that the size of $\rho(M)$ only determines the asymptotic behavior of convergence [2], whereas the empirically observed convergence rate might be very different. For instance, in the case of non-normal M the norm $\|M\|_2$ [3] might be greater than 1.

1.3.2 Splitting methods: Fixed point and residual form

Consider the splitting of the matrix A by

$$A = G - H. \tag{10}$$

To construct a consistent fixed point iteration we start with

$$Ax_* = b \quad \to \quad Gx_* = Hx_* + b. \tag{11}$$

[2] $\rho(M)$ is also called *convergence factor*, while $-\ln\rho(M)$ is often referred to as *convergence rate*.

[3] Compare with Banach fixed point criterion, where $\|M\|$ is connected with the 'rate of contraction'.

References

[1] Y. Saad. *Iterative Methods for sparse linear systems - Second Edition.* SIAM 2003.

[2] Y. Saad. *Numerical Methods for Large Eigenvalue Problems - Second Edition.* SIAM 2011.

[3] W. Auzinger and J.M. Melenk. *Iterative Solution of Large Linear Systems..* Lecture notes, TU Wien, 2009.

[4] O. Scherzer. *CO-MAT1.* Lecture notes, Uni Wien, 2014/15.

[5] L. Exl. *Numerical Mathematics II.* Lecture notes in Physics, Uni Wien, 2016-2023.

[6] L. Exl. *Iterative Methoden für große dünnbesetzte lineare Gleichungssysteme.* Lecture notes, Universität Hamburg, 2012.

[7] L. Exl. *Mathematik 1 für Industrial Simulation* Lecture notes, FH St. Pölten, 2011.

[8] L. Exl. *Mathematik 2 für Industrial Simulation* Lecture notes, FH St. Pölten, 2012.

[9] N. Trefethen and D. Bau III. *Numerical Linear Algebra.* SIAM, Philadelphia, 1997.

[10] A. Quarteroni, R. Sacco, and F. Saleri *Numerical Mathematics.* Springer Verlag, Berlin, 2000.

[11] J. Nocedal and S. Wright. *Numerical Optimization - Second Edition.* Springer Verlag, Science & Business Media, 2006.