



- 1 We are given the linear system of  $n$  equations

$$\sum_{j=1}^n a_{ij}x_j = b_i, \quad i = 1, 2, \dots, n, \text{ or in short as } \mathbf{Ax} = \mathbf{b}.$$

This can be solved by fixed point iterations as follows. Note that the *system* of equations is a fixed point formulation, not each equation on its own:

1. Solve equation  $i$  with respect to  $x_i$  for  $i = 1, 2, \dots, n$ . The result is the fixed point formulation

$$x_i = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{ij}x_j - \sum_{j=i+1}^n a_{ij}x_j \right), \quad i = 1, 2, \dots, n. \quad (1)$$

2. The corresponding fixed point iteration scheme (Jacobi-iterations) is given by

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right), \quad i = 1, 2, \dots, n. \quad (2)$$

for  $k = 0, 1, 2, 3, \dots$ , together with an appropriate choice of starting values  $\mathbf{x}^{(0)}$ .

3. This can be improved by using the updated value of  $x_j$  for  $j < i$ , resulting in the scheme (Gauss-Seidel iterations):

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left( b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right), \quad i = 1, 2, \dots, n.$$

If no starting value is given, start with  $\mathbf{x}^{(0)} = \mathbf{0}$ .

- a) (H) Do two iterations on each of the two iteration schemes on the linear systems of equations

$$\mathbf{Ax} = \mathbf{b}$$

with

$$A = \begin{bmatrix} 3 & 1 & 1 \\ 1 & 3 & -1 \\ 3 & 1 & -5 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 5 \\ 3 \\ -1 \end{bmatrix}$$

starting with  $\mathbf{x}^{(0)} = \mathbf{0}$ .

A matrix  $A$  is called *strictly diagonally dominant* if

$$|a_{ii}| > \sum_{j=1}^{i-1} |a_{ij}| + \sum_{j=i+1}^n |a_{ij}|, \quad \text{for all } i = 1, 2, \dots, n$$

**b)** (H) Is the matrix  $A$  from **a)** strictly diagonal dominant?

It can be proved that strictly diagonally dominant matrices are always nonsingular, so the system  $A\mathbf{x} = \mathbf{b}$  will always have a unique solution.

**c)** The next task is to prove that the Jacobi-iterations will converge for all choices of starting values if the coefficient matrix  $A$  is diagonal dominant. Let  $\mathbf{x}$  be the exact solution, then the error after  $k$  iterations is

$$\mathbf{e}^{(k)} = \mathbf{x} - \mathbf{x}^{(k)},$$

or in element form:

$$e_i^{(k)} = x_i - x_i^{(k)}, \quad i = 1, 2, 3, \dots, n.$$

The max-norm of a vector  $\mathbf{y} \in \mathbb{R}^n$  is  $\|\mathbf{y}\|_\infty = \max_{i=1, \dots, n} |y_i|$  (see *Preliminaries*). Your task is to prove that in the case of Jacobi iterations there is an  $L$  such that

$$\|\mathbf{e}^{(k+1)}\|_\infty \leq L \|\mathbf{e}^{(k)}\|_\infty$$

and that when  $A$  is strictly diagonally dominant, we have  $L < 1$ . This can be done following the steps below.

- Take the difference between (1) and (2) in order to express  $e_i^{(k+1)}$  as some linear combinations of  $e_i^{(k)}$ , that is

$$e_i^{k+1} = \sum_{j=1, j \neq i}^n d_{ij} e_j^{(k)}, \quad i = 1, 2, \dots, n.$$

- Take the absolute value on both sides, and use the triangle inequality  $|a + b| \leq |a| + |b|$ , together with the fact that  $|e_i^{(k)}| \leq \|\mathbf{e}^{(k)}\|_\infty$  to find an  $L$  such that

$$\|\mathbf{e}^{(k+1)}\|_\infty \leq L \|\mathbf{e}^{(k)}\|_\infty$$

The constant  $L$  will be some combination of the absolute values of the coefficients of  $A$ .

- Use the definition of a strictly diagonally matrix to prove that  $L < 1$ .

#### Remarks:

- Also Gauss-Seidel iterations converges when  $A$  is strictly diagonal dominant.
- Both Jacobi and Gauss-Seidel are examples of a classical iterative methods for linear systems. There are a lot more of them.
- Small systems of linear equations should always be solved by some version of Gauss-elimination (Mathematics 3). Iterative methods are only practical for very large systems of equations where most of the elements in the matrix  $A$  are zero.

- 2 (H) Consider the initial value problem

$$y' - xy^2 = 0, \quad y(0) = 1.$$

- a) Find the exact solution to the equation.
- b) Do 4 steps of Euler's method with  $h = 0.1$ . Compute the error at the last step.
- c) Do 2 steps of Heuns method with  $h = 0.2$ . Compute the error at the last step
- d) Do 1 step of the 4th order Runge-Kutta method (see Problem 4) with  $h = 0.4$ . Compute the error at the end.

In each case, 4 function evaluations have been done. Which of the methods performed best?

- 3 Consider two charges (in one dimension), with positions  $u_1(x)$  and  $u_2(x)$  at time  $x > 0$ . The positions of the charges is described by a system of two second order differential equations:

$$\begin{aligned} u_1''(x) &= -\frac{1}{(u_1(x) - u_2(x))^2}, \\ u_2''(x) &= \frac{1}{(u_1(x) - u_2(x))^2}, \end{aligned} \quad (3)$$

with initial values  $u_1(0) = 0, u_1'(0) = 1, u_2(0) = 1, u_2'(0) = 0$ .

- a) (H) Rewrite (3) to a system of first order differential equations.
- b) (H) Do one step by Heun's method, using  $h = 0.1$  to solve this system of equations.  
Hint: This is messy, and it is easy to make mistakes. To check your hand calucations, do one step with the function `heun` in the Jupyter notebook, and print out all intermediate vectors. But do remember that this notebook is not available at the exam!
- c) (J) Solve the problem numerically over the interval  $[0, 1]$ , by a method of your own choice.

- 4 a) (J) Implement the classical Runge–Kutta method (RK4):

$$\begin{aligned} \mathbf{k}_1 &= \mathbf{f}(x_n, \mathbf{y}_n) \\ \mathbf{k}_2 &= \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_1\right) \\ \mathbf{k}_3 &= \mathbf{f}\left(x_n + \frac{h}{2}, \mathbf{y}_n + \frac{h}{2}\mathbf{k}_2\right) \\ \mathbf{k}_4 &= \mathbf{f}(x_n + h, \mathbf{y}_n + h\mathbf{k}_3) \\ \mathbf{y}_{n+1} &= \mathbf{y}_n + \frac{h}{6}(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4). \end{aligned}$$

and verify numerically that the order is 4 (see `oving10_python_files.ipynb` for how to do this).

b) (J) Use RK4 to solve the Lotka–Volterra equation:

$$\begin{aligned}y_1' &= 2.0 y_1 - y_1 y_2, & y_1(0) &= 2.0 \\y_2' &= 0.5 y_1 y_2 - y_2, & y_2(0) &= 0.5\end{aligned}$$

over the interval  $[0, 10]$ . Try both the stepsize  $h = 0.1$  and  $h = 0.5$ .

What happens if you try to solve the same problem with the same stepsizes with Heuns method?

NB! If you have solved the previous points correctly, you should have, for  $h = 0.5$  got a numerical solution corresponding to the  $\bullet$ 's in Figure 1. The dotted lines serves as the exact solution, calculated by a numerical method with high accuracy, and are here only for demonstration purpose. We observe that the solution is quite accurate in the points  $(x_n, \mathbf{y}_n)$ , but we are loosing useful information between.



Figure 1: The Lotka–Volterra equation, where  $\bullet$  represents the solution from RK4 with  $h = 0.5$  and  $--$  the exact solution.