24 Basic iterative methods

Today we start a new section of the course: the solution of linear systems by iterative techniques. The notes given here are not a complete record of what we did in class. For more details, see your lecture notes or [Saa03, Chap. 4].

The methods we'll look at first are classical: the Jacobi, Gauss-Seidel and SOR methods. Next week, we'll look at more modern Kirov subspace methods.

24.1 The basic idea

As usual, we want to find the vector $\mathbf{x} \in \mathbb{R}^m$, such that

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

where $A \in \mathbb{R}^{m \times m}$ and $b \in \mathbb{R}^m$. We will try to do this by constructing a sequence of vectors

$$\{\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots\},$$

which, we hope, will be successively better approximations to $\mathbf{x}$. For example, it might be that

$$\|\mathbf{x} - \mathbf{x}^{(k+1)}\| \leqslant \|\mathbf{x} - \mathbf{x}^{(k)}\|,$$

for some norm, or $\mathbf{x}^{(k)} \to \mathbf{x}$ as $k \to \infty$.

24.2 The residual

In both theory and practice, one of the most important quantities to work with is the *residual*:

$$\mathbf{r}^{(k)} = \mathbf{b} - A - \mathbf{x}^{(k)}.$$

Obviously, if $\mathbf{r}^{(k)} = \mathbf{0}$, we are done. So much of the focus can be on trying to minimise $\mathbf{r}^{(k)}$. Furthermore, in practical computations, we do not compute an infinite number of $\mathbf{x}^{(k)}$. Instead we would like to iterate until $\|\mathbf{x} - \mathbf{x}^{(k)}\|$ is smaller than some predetermined value. But since we can't compute $\mathbf{x} - \mathbf{x}^{(k)}\|$, we rely on making $\|\mathbf{r}^{(k)}\|$ small enough.

24.3 Jacobi's method

Suppose we want to make $\mathbf{r}_i^{(k)} = 0$ then this leads (See notes from class!) to *Jacobi's method*

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\big(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}\big). \qquad (24.1)$$

Example 24.1. If we take $A = T_N$, for $N = 4$, and apply the Jacobi method to solve $A\mathbf{x} = h^2$, a summary of the results are as shown in Table 24.1. After 100 iterations, the error is approximately 10^{-10}.

Rather than (24.1), we shall prefer to write the method using a matrix-formulation. Let D, E and F be matrices in $\mathbb{R}^{m \times m}$ such that

- D is diagonal;
- E is strictly lower triangular (i.e., it is lower triangular, and $E_{ii} = 0$;
- F is strictly upper triangular.
- $A = D - E - F$.

Then we can write Jacobi's method as

$$\mathbf{x}^{(k+1)} = D^{-1}(E + F)\mathbf{x}^{(k)} + D^{-1}b.$$

k	$\|\mathbf{x} - \mathbf{x}^{(k)}\|_2$	$\|\mathbf{r}^{(k)}\|_2$
0	3.187e-01	1.250e-01
10	3.825e-02	1.461e-02
20	4.595e-03	1.755e-03
30	5.519e-04	2.108e-04
40	6.628e-05	2.532e-05
50	7.961e-06	3.041e-06
60	9.562e-07	3.653e-07
70	1.149e-07	4.387e-08
80	1.379e-08	5.269e-09
90	1.657e-09	6.329e-10
100	1.990e-10	7.601e-10

Table 24.1: Solving the problem in Example 24.1 using Jacobi's method

24.4 Gauss-Seidel method

As we shall see, the Gauss-Seidel method is not very efficient. A better option (*see notes from class*!) would be the Gauss-Seidel method:

$$\mathbf{x}^{(k+1)} = (D - E)^{-1}F\mathbf{x}^{(k)} + (D - E)^{-1}b.$$

If we apply this technique to the problem in Example 24.1 we get the results in Table 24.2. We see that we achieve machine precision after about 80 iterations.

k	$\|\mathbf{x} - \mathbf{x}^{(k)}\|_2$	$\|\mathbf{r}^{(k)}\|_2$
0	3.187e-01	1.250e-01
10	5.385e-03	2.473e-03
20	7.769e-05	3.568e-05
30	1.121e-06	5.148e-07
40	1.617e-08	7.426e-09
50	2.333e-10	1.071e-10
60	3.365e-12	1.545e-12
70	4.856e-14	2.231e-14
80	6.790e-16	3.310e-16

Table 24.2: Solving the problem in Example 24.1 using the Gauss-Seidel method