

Appendix A

Tridiagonal matrix algorithm

The tridiagonal matrix algorithm (TDMA), also known als *Thomas algorithm*, is a simplified form of Gaussian elimination that can be used to solve tridiagonal system of equations

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = y_i, \quad i = 1, \dots, n, \quad (\text{A.1})$$

or, in matrix form ($a_1 = 0, c_n = 0$)

$$\begin{pmatrix} b_1 & c_1 & 0 & \dots & \dots & 0 \\ a_2 & b_2 & c_2 & \dots & \dots & 0 \\ 0 & a_3 & b_3 & c_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & c_{n-1} & \dots \\ 0 & \dots & \dots & 0 & a_n & b_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_n \end{pmatrix}$$

The TDMA is based on the Gaussian elimination procedure and consist of two parts: a forward elimination phase and a backward substitution phase [8]. Let us consider the system (A.1) for $i = 1 \dots n$ and consider following modification of first two equations:

$$\text{Eq}_{i=2} \cdot b_1 - \text{Eq}_{i=1} \cdot a_2$$

which relults in

$$(b_1 b_2 - c_1 a_2) x_2 + c_2 b_1 x_3 = b_1 y_2 - a_2 y_1.$$

The effect is that x_1 has been eliminated from the second equation. In the same manner one can eliminate x_2 , using the modified second equation and the third one (for $i = 3$):

$$(b_1 b_2 - c_1 a_2) \text{Eq}_{i=3} - a_3 (\text{mod. Eq}_{i=2}),$$

which would give

$$(b_3(b_1 b_2 - c_1 a_2) - c_2 b_1 a_3) x_3 + c_3(b_1 b_2 - c_1 a_2) x_4 = y_3(b_1 b_2 - c_1 a_2) - (y_2 b_1 - y_1 a_2) a_3$$

If the procedure is repeated until the n 'th equation, the last equation will involve the unknown function x_n only. This function can be then used to solve the modified equation for $i = n - 1$ and so on, until all unknown x_i are found (backward

substitution phase). That is, we are looking for a backward ansatz of the form:

$$x_{i-1} = \gamma_i x_i + \beta_i. \quad (\text{A.2})$$

If we put the last ansatz in Eq. (A.1) and solve the resulting equation with respect to x_i , the following relation can be obtained:

$$x_i = \frac{-c_i}{a_i \gamma_i + b_i} x_{i+1} + \frac{y_i - a_i \beta_i}{a_i \gamma_i + b_i} \quad (\text{A.3})$$

This relation possesses the same form as Eq. (A.2) if we identify

$$\boxed{\gamma_{i+1} = \frac{-c_i}{a_i \gamma_i + b_i}, \quad \beta_{i+1} = \frac{y_i - a_i \beta_i}{a_i \gamma_i + b_i}}. \quad (\text{A.4})$$

Equation (A.4) involves the recursion formula for the coefficients γ_i and β_i for $i = 2, \dots, n-1$. The missing values γ_1 and β_1 can be derived from the first ($i = 1$) equation (A.1):

$$x_1 = \frac{y_1}{b_1} - \frac{c_1}{b_1} x_2 \Rightarrow \gamma_2 = -\frac{c_1}{b_1}, \beta_2 = \frac{y_1}{b_1} \Rightarrow \boxed{\gamma_1 = \beta_1 = 0}.$$

The last what we need is the value of the function x_n for the first backward substitution. We can obtain it if we put the ansatz

$$x_{n-1} = \gamma_n x_n + \beta_n$$

into the last ($i = n$) equation (A.1):

$$a_n(\gamma_n x_n + \beta_n) + b_n x_n = y_n,$$

yielding

$$x_n = \frac{y_n - a_n \beta_n}{a_n \gamma_n + b_n}.$$

One can get this value directly from Eq. (A.2), if one formally puts

$$x_{n+1} = 0.$$

Altogether, the TDMA can be written as:

Algorithm:

1. Set $\gamma_1 = \beta_1 = 0$;
2. Evaluate for $i = 1, \dots, n-1$

$$\gamma_{i+1} = \frac{-c_i}{a_i \gamma_i + b_i}, \quad \beta_{i+1} = \frac{y_i - a_i \beta_i}{a_i \gamma_i + b_i};$$

3. Set $x_{n+1} = 0$;
4. Find for $i = n+1, \dots, 2$

$$x_{i-1} = \gamma_i x_i + \beta_i.$$

Use "for" loop here: for (i=1; i<=n-1; i++)

After finding these values,
the "for" loop is finished

Stage 4: Begin new "for" loop

The algorithm admits $\mathcal{O}(n)$ operations instead of $\mathcal{O}(n^3)$ required by Gaussian elimination.

for (i=n+1; i>=2; i=i-1)

Limitation

The TDMA is only applicable to matrices that are diagonally dominant, i.e.,

$$|b_i| > |a_i| + |c_i|, \quad i = 1, \dots, n.$$