Finite differences for second order linear PDE in 2 variables

PDE are classified into three types:

- **elliptic** (example: Poisson equation)
- **parabolic** (example: heat equation)
- **hyperbolic** (example: wave equation)

Discretization of PDE (inside the given domain) consists of the three following steps:

1. Choosing the step-size in both directions and constructing the grid.
2. Expressing the equation at every grid node (inside the domain).
3. Substitution of derivatives with the finite differences.

Caution: All terms of the equation have to be expressed or approximated at the same grid node.

Heat equation

Mixed problem for heat equation

We are seeking a function $u \equiv u(x,t)$ which satisfies

$$\frac{\partial u}{\partial t} = p\,\frac{\partial^2 u}{\partial x^2} + f(x,t) \quad \text{in the domain } \Omega = (a,b) \times (0,T) \;, \tag{1}$$

has prescribed initial condition at time $t = 0$

$u(x,0) = \phi(x)$ for $x \in \langle a,b \rangle$

and has prescribed boundary values for $t > 0$

$u(a,t) = \alpha(t)\;, \quad u(b,t) = \beta(t)\;\;.$

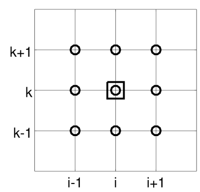
Coefficient of thermal diffusivity p is supposed to be constant.
Initial and boundary conditions have to satisfy *conditions of compatibility*:

$\phi(a) = \alpha(0)\;, \quad \phi(b) = \beta(0)\;\;.$

Discretization:

1. Variables x and t in the equation now represent different entities (x usually represent spatial direction and t represents time). So it is natural to choose different step-sizes and construct a rectangular grid of nodes over Ω with equal mesh spacing h in x direction and τ for t.

Scheme of the grid around a grid node P_i^k:



Notation:

$P_i^k \equiv [x_i, t_k]$ ... grid nodes, where

x_i ... x-coordinates of the nodes: $h = x_{i+1} - x_i$

t_k ... t-coordinates of the nodes: $\tau = t_{k+1} - t_k$

$u(x,t)$ ... function of two variables defined in Ω, $u(P_i^k) \equiv u(x_i, t_k)$

$U_i^k \approx u(P_i^k)$ ... approximate value of $u(x,t)$ at a grid node P_i^k

Explicit method

2. Express the equation (1) at every node $P_i^k = [x_i, t_k],\ k = 0, 1, \ldots$:

$$\frac{\partial u}{\partial t}(P_i^k) \ = \ p\,\frac{\partial^2 u}{\partial x^2}(P_i^k) + f(P_i^k) \tag{2}$$

3. Use the second central difference for approximation of the partial derivative with respect to x at the node P_i^k (see Figure 1) as

$$\frac{\partial^2 u}{\partial x^2}(P_i^k) = \frac{u(P_{i-1}^k) - 2u(P_i^k) + u(P_{i+1}^k)}{h^2} + \mathcal{O}(h^2) \approx \frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2}$$

and the first forward difference for approximation of the partial derivative with respect to t as

$$\frac{\partial u}{\partial t}(P_i^k) = \frac{u(P_i^{k+1}) - u(P_i^k)}{\tau} + \mathcal{O}(\tau) \approx \frac{U_i^{k+1} - U_i^k}{\tau}$$

Substitution of these differences into (2):

$$\frac{U_i^{k+1} - U_i^k}{\tau} \ = \ p\,\frac{U_{i-1}^k - 2U_i^k + U_{i+1}^k}{h^2} + f(P_i^k)\ .$$

After rearrangig this leads to equation for 4 unknowns:

$$U_i^{k+1} = \ \sigma U_{i-1}^k + (1 - 2\sigma)\,U_i^k + \sigma U_{i+1}^k + \tau\, f(P_i^k), \tag{3}$$

where $\sigma = \dfrac{p\,\tau}{h^2}$.

Condition of stability for explicit method: $\sigma \leq 0.5$.

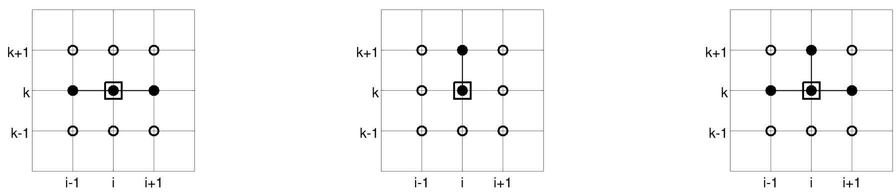


Figure 1: Grid nodes used for finite differences centered at the node P_i^k. Left: 2-nd central difference with respect to x. Center: 1-st forward difference with respect to t. Right: Four-point stencil for explicit method.

Numerical solution is evaluated one time level after another: from known values at k-th time level, values at $(k+1)$-st time level are computed, one node after another, using the formula (3). Values at left and right boundaries are given by the boundary conditions $\alpha(t)$ and $\beta(t)$, respectively. Values at the initial time level are given by the initial condition $\phi(x)$.

Implicit method

2. Express the equation (1) at every node $P_i^{k+1} = [x_i, t_{k+1}],\ k = 0, 1, \ldots$:

$$\frac{\partial u}{\partial t}(P_i^{k+1}) \;=\; p\,\frac{\partial^2 u}{\partial x^2}(P_i^{k+1}) + f(P_i^{k+1}) \tag{4}$$

3. Use the second central difference for approximation of the partial derivative with respect to x at the node P_i^{k+1} as

$$\frac{\partial^2 u}{\partial x^2}(P_i^{k+1}) = \frac{u(P_{i-1}^{k+1}) - 2u(P_i^{k+1}) + u(P_{i+1}^{k+1})}{h^2} + \mathcal{O}(h^2) \approx \frac{U_{i-1}^{k+1} - 2U_i^{k+1} + U_{i+1}^{k+1}}{h^2}$$

and the first backward difference for approximation of the partial derivative with respect to t as

$$\frac{\partial u}{\partial t}(P_i^{k+1}) = \frac{u(P_i^{k+1}) - u(P_i^k)}{\tau} + \mathcal{O}(\tau) \approx \frac{U_i^{k+1} - U_i^k}{\tau}$$

and substitute these differences into (4):

$$\frac{U_i^{k+1} - U_i^k}{\tau} \;=\; p\,\frac{U_{i-1}^{k+1} - 2U_i^{k+1} + U_{i+1}^{k+1}}{h^2} + f(P_i^{k+1})\ .$$

After rearrangig this leads to equation for 4 unknowns:

$$-\sigma U_{i-1}^{k+1} + (1 + 2\sigma)\,U_i^{k+1} - \sigma U_{i+1}^{k+1} \;=\; U_i^k + \tau\, f(P_i^{k+1}) \tag{5}$$

where $\sigma = \dfrac{p\,\tau}{h^2}$.

The discretization is performed at every inner node of the $(k+1)$-st time level, so a system of linear equations is obtained, from which values at the $(k+1)$-st time level can be computed. Values at the initial time level are given by the initial condition, values at left and right boundaries are given by the boundary conditions.

Implicit scheme is **unconditionally stable.**

Figure 2: Grid nodes used for finite differences. Left: Four-point stencil for explicit method (it is centered at the node P_i^k). Right: Four-point stencil for implicit method (it is centered at the node P_i^{k+1}).

Problem 1

Find numerical solution of the heat equation

$$\frac{\partial u}{\partial t} = 0.3\,\frac{\partial^2 u}{\partial x^2} + x \quad \text{in domain } \Omega = (0,1)\times(0,0.4)$$

with initial condition $u(x,0) = x^2$ for $x \in \langle 0,1\rangle$

and boundary conditions $u(0,t) = 0\ ,\ \ u(1,t) = 1$ for $t > 0$.

a) Choose the spatial step-size $h = 0.25$ and use the time step-size τ as big as possible, provided it still leads to the stable explicit method. Then use the explicit method.

b) Choose the step-size h and the time step-size twice as big as before and use the implicit method.

The solution

First of all, let us check the compatibility of the initial and boundary conditions: both conditions are equal to zero for $x = 0$, $t = 0$ and both conditions are equal to one for $x = 1$, $t = 0$. We can see that the initial and boundary conditions are compatible.

a) We are seeking the maximal time step-size τ such that the explicit method is stable, i.e. $\sigma \le 0.5$:

$$\sigma = \frac{p\,\tau}{h^2} \le 0.5 \quad \Leftrightarrow \quad \tau \le 0.5\,\frac{h^2}{p} = 0.5\,\frac{0.25^2}{0.3} = 0.10417$$

In order to solve the problem within the time interval $\langle 0,0.4\rangle$, choose τ such that the end time value 0.4 is its multiple: set $\tau = 0.1$, then

$$\sigma = \frac{0.3\cdot 0.1}{0.25^2} = 0.48\ .$$

Let us prepare a table, which then will be subsequently filled by rows, starting from the initial time level (in the bottom). The layout of the table:

t_4	0.4	U_0^4	U_1^4	U_2^4	U_3^4	U_4^4
t_3	0.3	U_0^3	U_1^3	U_2^3	U_3^3	U_4^3
t_2	0.2	U_0^2	U_1^2	U_2^2	U_3^2	U_4^2
t_1	0.1	U_0^1	U_1^1	U_2^1	U_3^1	U_4^1
t_0	0.0	U_0^0	U_1^0	U_2^0	U_3^0	U_4^0
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

Values of t are in the first column of the table, x-coordinate nodes are in the bottom row. We start by filling the values given by the initial condition into the row for $t = 0$ (blue) and the values given by the boundary conditions into the second and the last columns (red). Values at the corners (violet) should be the same whether they are computed from the initial condition or from the boundary condition. Values of the solution we are searching for are inside the table (black). In the beginning, after filling in the initial condition $u(x,0) = x^2$ and the boundary conditions $u(0,t) = 0$ and $u(1,t) = 1$, we have

t_4	0.4	0.0000				1.0000
t_3	0.3	0.0000				1.0000
t_2	0.2	0.0000				1.0000
t_1	0.1	0.0000				1.0000
t_0	0.0	0.0000	0.0625	0.2500	0.5625	1.0000
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

Now let us subsequently compute values at particular time level, using the already computed values from the previous time level:

The first level ($t_1 = 0.1$):

$$U_1^1 = (1-2\sigma)\,U_1^0 + \sigma(U_0^0 + U_2^0) + \tau\, f(x_1, t_0) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.0625 + 0.48(0 + 0.25) + 0.1 \cdot 0.25 = 0.1475$$
$$U_2^1 = (1-2\sigma)\,U_2^0 + \sigma(U_1^0 + U_3^0) + \tau\, f(x_2, t_0) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.25 + 0.48(0.0625 + 0.5625) + 0.1 \cdot 0.50 = 0.36$$
$$U_3^1 = (1-2\sigma)\,U_3^0 + \sigma(U_2^0 + U_4^0) + \tau\, f(x_3, t_0) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.5625 + 0.48(0.25 + 1) + 0.1 \cdot 0.75 = 0.6975$$

The second level ($t_2 = 0.2$):

$$U_1^2 = (1-2\sigma)\,U_1^1 + \sigma(U_0^1 + U_2^1) + \tau\, f(x_1, t_1) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.1475 + 0.48(0 + 0.36) + 0.1 \cdot 0.25 = 0.2037$$
$$U_2^2 = (1-2\sigma)\,U_2^1 + \sigma(U_1^1 + U_3^1) + \tau\, f(x_2, t_1) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.36 + 0.48(0.1475 + 0.6975) + 0.1 \cdot 0.50 = 0.47$$
$$U_3^2 = (1-2\sigma)\,U_3^1 + \sigma(U_2^1 + U_4^1) + \tau\, f(x_3, t_1) =$$
$$= (1 - 2 \cdot 0.48) \cdot 0.6975 + 0.48(0.36 + 1) + 0.1 \cdot 0.75 = 0.7557$$

The third and the fourth level ($t = 0.3$ and $t = 0.4$) can be computed similarly. The resulting table with approximate values of the solution at the interior nodes:

t_4	0.4	0.0000	0.2894	0.5846	0.8415	1.0000
t_3	0.3	0.0000	0.2587	0.5293	0.8108	1.0000
t_2	0.2	0.0000	0.2037	0.4700	0.7557	1.0000
t_1	0.1	0.0000	0.1475	0.3600	0.6975	1.0000
t_0	0.0	0.0000	0.0625	0.2500	0.5625	1.0000
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

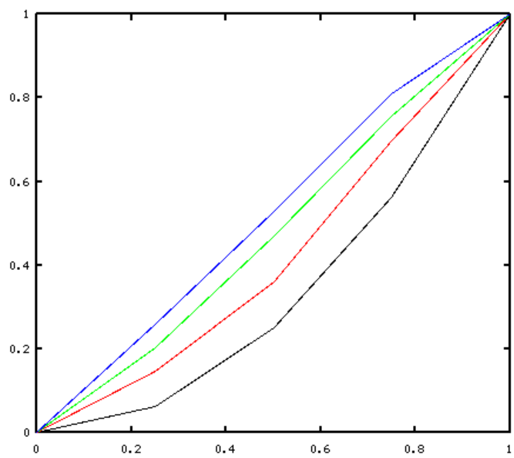


Figure 3: **Problem 1:** Graphs of the solution at time steps from 0 to 0.3 – black, red, green and blue, in succession. The horizontal axis is x, the vertical axis is $u(x,t)$.

b) For $\tau = 0.2$, $h = 0.25$ we have $\sigma = \dfrac{0.3 \cdot 0.2}{0.25^2} = 0.96$. In matrix form, the implicit method can be written as

$$\begin{bmatrix} 1+2\sigma & -\sigma & 0 \\ -\sigma & 1+2\sigma & -\sigma \\ 0 & -\sigma & 1+2\sigma \end{bmatrix} \begin{bmatrix} U_1^{k+1} \\ U_2^{k+1} \\ U_3^{k+1} \end{bmatrix} = \begin{bmatrix} \sigma U_0^{k+1} + U_1^k + \tau\, f(x_1, t_{k+1}) \\ U_2^k + \tau\, f(x_2, t_{k+1}) \\ \sigma U_4^{k+1} + U_3^k + \tau\, f(x_3, t_{k+1}) \end{bmatrix}$$

The first time level ($t_1 = 0.2$):

$$\begin{bmatrix} 2.92 & -0.96 & 0 \\ -0.96 & 2.92 & -0.96 \\ 0 & -0.96 & 2.92 \end{bmatrix} \begin{bmatrix} U_1^1 \\ U_2^1 \\ U_3^1 \end{bmatrix} = \begin{bmatrix} 0 + 0.0625 + 0.2 \cdot 0.25 \\ 0.2500 + 0.2 \cdot 0.50 \\ 0.96 + 0.5625 + 0.2 \cdot 0.75 \end{bmatrix} = \begin{bmatrix} 0.1125 \\ 0.3500 \\ 1.6725 \end{bmatrix}$$

$U^1 = [\, 0.1731,\ 0.4093,\ 0.7074]^T$.

The second time level ($t_2 = 0.4$):

$$\begin{bmatrix} 2.92 & -0.96 & 0 \\ -0.96 & 2.92 & -0.96 \\ 0 & -0.96 & 2.92 \end{bmatrix} \begin{bmatrix} U_1^2 \\ U_2^2 \\ U_3^2 \end{bmatrix} = \begin{bmatrix} 0 + 0.1731 + 0.2 \cdot 0.25 \\ 0.4093 + 0.2 \cdot 0.50 \\ 0.96 + 0.7074 + 0.2 \cdot 0.75 \end{bmatrix} = \begin{bmatrix} 0.2231 \\ 0.5093 \\ 1.8174 \end{bmatrix}$$

$U^2 = [\, 0.2459,\ 0.5156,\ 0.7919]^T$.

With the implicit method, the double step-size does not lead to unstability, although the the error is probably about twice as big as in a).

Problem 2

Consider the heat equation

$$\frac{\partial u}{\partial t} = 0.2\,\frac{\partial^2 u}{\partial x^2} + 2t + x \quad \text{in domain } \Omega = (0,1)\times(0,T)$$

with initial condition $\quad u(x,0) = 0 \;$ for $x \in \langle 0,1\rangle$

and boundary conditions $\quad u(0,t) = 0\;,\;\; u(1,t) = 3t$ pro $t > 0\;.$

Choose the spatial and time step-sizes $h = 0.25$ and $\tau = 0.1$, respectively. Verify the stability of the explicit method and compute an approximate value of $u(0.75, 0.4)$.

The solution

$$\sigma = \frac{p\,\tau}{h^2} = \frac{0.2\cdot 0.1}{0.25^2} = 0.32 \le 0.5$$

For the given combination of spatial and time step-sizes, the explicit method is stable.

Let us prepare a table, which then will be subsequently filled by rows from the initial time level, as the time levels are computed one after another. In order to compute the approximate value U_3^4 of $u(0.75, 0.4)$, there is no need to compute all the values inside the domain – the "pyramid" designated in the table will be sufficient:

t_4	0.4				U_3^4	
t_3	0.3			U_2^3	U_3^3	U_4^3
t_2	0.2		U_1^2	U_2^2	U_3^2	U_4^2
t_1	0.1	U_0^1	U_1^1	U_2^1	U_3^1	U_4^1
t_0	0.0	U_0^0	U_1^0	U_2^0	U_3^0	U_4^0
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

After filling in the initial condition $u(x,0) = 0$ and boundary conditions $u(0,t) = 0$ and $u(1,t) = 3t$ we have

t_4	0.4					
t_3	0.3					0.9000
t_2	0.2					0.6000
t_1	0.1	0.0000				0.3000
t_0	0.0	0.0000	0.0000	0.0000	0.0000	0.0000
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

Now let us subsequently compute values at particular time levels:

The first level ($t_1 = 0.1$):

$U_1^1 = (1-2\sigma)\,U_1^0 + \sigma(U_0^0 + U_2^0) + \tau\, f(x_1, t_0) =$
$\quad = (1 - 2\cdot 0.32)\cdot 0 + 0.32(0+0) + 0.1(2\cdot 0 + 0.25) = 0.025$

$U_2^1 = (1-2\sigma)\,U_2^0 + \sigma(U_1^0 + U_3^0) + \tau\, f(x_2, t_0) =$
$\quad = 0.36\cdot 0 + 0.32(0+0) + 0.1(2\cdot 0 + 0.5) = 0.05$

$U_3^1 = (1-2\sigma)\,U_3^0 + \sigma(U_2^0 + U_4^0) + \tau\, f(x_3, t_0) =$
$\quad = 0.36\cdot 0 + 0.32(0+0) + 0.1(2\cdot 0 + 0.75) = 0.075$

The second level ($t_2 = 0.2$):

$U_1^2 = (1-2\sigma)\,U_1^1 + \sigma(U_0^1 + U_2^1) + \tau\, f(x_1, t_1) =$
$\quad = 0.36 \cdot 0.025 + 0.32(0 + 0.05) + 0.1(2 \cdot 0.1 + 0.25) = 0.07$
$U_2^2 = (1-2\sigma)\,U_2^1 + \sigma(U_1^1 + U_3^1) + \tau\, f(x_2, t_1) =$
$\quad = 0.36 \cdot 0.05 + 0.32(0.025 + 0.075) + 0.1(2 \cdot 0.1 + 0.5) = 0.12$
$U_3^2 = (1-2\sigma)\,U_3^1 + \sigma(U_2^1 + U_4^1) + \tau\, f(x_3, t_1) =$
$\quad = 0.36 \cdot 0.075 + 0.32(0.05 + 0.3) + 0.1(2 \cdot 0.1 + 0.75) = 0.234$

Te third level ($t_3 = 0.3$):
$U_2^3 = (1-2\sigma)\,U_2^2 + \sigma(U_1^2 + U_3^2) + \tau\, f(x_2, t_2) =$
$\quad = 0.36 \cdot 0.12 + 0.32(0.07 + 0.234) + 0.1(2 \cdot 0.2 + 0.5) = 0.2305$
$U_3^3 = (1-2\sigma)\,U_3^2 + \sigma(U_2^2 + U_4^2) + \tau\, f(x_3, t_2) =$
$\quad = 0.36 \cdot 0.234 + 0.32(0.12 + 0.6) + 0.1(2 \cdot 0.2 + 0.75) = 0.4296$

The fourth level ($t_4 = 0.4$):
$U_3^4 = (1-2\sigma)\,U_3^3 + \sigma(U_2^3 + U_4^3) + \tau\, f(x_3, t_3) =$
$\quad = 0.36 \cdot 0.4296 + 0.32(0.2305 + 0.9) + 0.1(2 \cdot 0.3 + 0.75) = 0.6514$

The resulting table with approximate values of the solution:

t_4	0.4				0.6514	
t_3	0.3			0.2305	0.4296	0.9000
t_2	0.2		0.0700	0.1200	0.2340	0.6000
t_1	0.1	0.0000	0.0250	0.0500	0.0750	0.3000
t_0	0.0	0.0000	0.0000	0.0000	0.0000	0.0000
		0.00	0.25	0.50	0.75	1.00
		x_0	x_1	x_2	x_3	x_4

The approximate value of $u(0.75, 0.4)$ is equal to 0.6514 .