RUSSIAN ACADEMY OF SCIENCES
BUDKER INSTITUTE OF NUCLEAR PHYSICS
SIBERIAN BRANCH

S.V. Miginsky

AN EFFECTIVE NUMERICAL SCHEME FOR PARABOLIC PARTIAL DIFFERENTIAL EQUATIONS

Budker INP 2011-25

Novosibirsk
2011

An Effective Numerical Scheme
for Parabolic Partial Differential Equations

S.V. Miginsky
Budker Institute of Nuclear Physics, 630090, Novosibirsk, Russia

Abstract

Parabolic partial differential equations frequently arise in computational physics. For instance, a nonstationary heat conduction equation and diffraction one in a paraxial approach are of this type. A system of ordinary differential equations obtained from the initial equation by discretization of the Laplacian is stiff or has rapidly oscillating parasitic solutions, so an A-stable scheme is to be used. All these schemes include inversion of a huge dimension matrix, so can not be effective. In addition, a conventional three-node scheme provides only second approximation order that is also not effective.

An effective scheme for parabolic equations is proposed and investigated in the paper. It is based on the second order Rosenbrock scheme for the independent coordinate with a special procedure of matrix pseudoinversion and a three node one with a Numerov's corrector for the Laplacian.

Эффективная численная схема
для параболических дифференциальных уравнений
в частных производных

С.В. Мигинский
Институт ядерной физики им. Г.И. Будкера, 630090, Новосибирск, Россия

Аннотация

Параболические уравнения в частных производных достаточно часто встречаются в вычислительной физике, например – это нестационарное уравнение теплопроводности и уравнение дифракции волны в параксиальном приближении. Трудность численного решения такого уравнения состоит в том, что система обыкновенных дифференциальных уравнений, полученная дискретизацией лапласиана в исходном, является либо жесткой, либо имеет быстроосциллирующие паразитные решения. Соответственно, необходимо использовать A-устойчивый метод решения, который всегда требует обращения матрицы большого размера. Кроме того, стандартная трехточечная схема вычисления лапласиана имеет всего лишь второй порядок аппроксимации, что не весьма эффективно.

В работе предложена и исследована эффективная численная схема решения таких уравнений на основе метода Розенброка второго порядка по независимой координате с особым алгоритмом псевдообращения матрицы и трехточечной схемы с корректором Нумерова для вычисления лапласиана.

1. Introduction

Parabolic partial differential equations (PPDE) often arise in computational physics. Their general form is

$$\frac{\partial}{\partial t}\varphi(x,t) = \nabla\big(D\nabla\varphi(x,t)\big) + f(x,t), \tag{1.1}$$

where φ, x, and f can be scalars of vectors. We shall consider D = const further, so the equation can be simplified:

$$\frac{\partial}{\partial t}\varphi(x,t) = D\Delta\varphi(x,t) + f(x,t). \tag{1.2}$$

Note that D can be real, as in the diffusion equation or imaginary as in the paraxial diffraction one. If one discretizes the right part using, say, a conventional three-node scheme, he obtains a set of ordinary differential equations (SODE) to be solved using some numerical scheme along t. The problem is that the SODE obtained is stiff (if D is real) or has rapidly oscillating parasitic solutions (if D is imaginary). Let us demonstrate it. Consider unidimensional scalar PPDE:

$$\frac{\partial}{\partial t}\varphi(x,t) = D\frac{\partial^2}{\partial x^2}\varphi(x,t) + f(x,t). \tag{1.3}$$

Generating a uniform grid of a step Δx and applying a three-node approximation for the Laplacian

$$\frac{\partial^2}{\partial x^2}\varphi(x) \sim \frac{\varphi(x+\Delta x) + \varphi(x-\Delta x) - 2\varphi(x)}{(\Delta x)^2} \tag{1.4}$$

one obtains the SODE:

$$\frac{\partial}{\partial t}\varphi(x_i,t) = D\frac{\varphi(x_{i+1}) + \varphi(x_{i-1}) - 2\varphi(x_i)}{(\Delta x)^2} + f(x_i,t), \tag{1.5}$$

where i is the node index. The corresponding homogeneous equation set

$$\frac{\partial}{\partial t}\varphi(x_i,t) = D\frac{\varphi(x_{i+1}) + \varphi(x_{i-1}) - 2\varphi(x_i)}{(\Delta x)^2} \tag{1.6}$$

has the following full set of solutions:

$$\varphi(x,t) = A\exp\left(ikx + \lambda t\right), \tag{1.7}$$

where A is an arbitrary amplitude, and

$$\lambda = \frac{2D}{(\Delta x)^2}\left(\cos\left(k\Delta x\right) - 1\right), \quad 0 \le k \le \frac{\pi}{\Delta x}. \tag{1.8}$$

Thus, λ varies from 0 to $-4D/(\Delta x)^2$. If one would like to improve accuracy and decrease Δx, the minimum λ/D ratio tends to minus infinity. We do not need these rapidly decreasing (if D is real) or oscillating (if D is imaginary) solutions, but they exist and worsen the stability of conventional numerical schemes. It forces to decrease the integration step $\propto \Delta x^2$ and makes the procedure very ineffective [1]. One can use a dedicated scheme for stiff equations, but all of them need to calculate a Jacobian matrix and to invert some other matrix, so their efficiency for huge dimension system is extremely poor.

2. Numerical scheme

2.1. Basic scheme

First of all, consider a simplest numerical scheme for the two-dimensional equation (1.2)

$$\left[\frac{\partial}{\partial t} - D\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)\right]\varphi(x,y,t) = f(x,y,t). \tag{2.1}$$

Next, generate a regular rectangular grid with steps Δx and Δy for x and y respectively and replace the continuous Laplacian by the digital one

$$\left[\frac{d}{dt} - D\left(\frac{\mathbf{A_x}}{\Delta x^2} + \frac{\mathbf{A_y}}{\Delta y^2}\right)\right]\vec{\varphi}(t) = \mathbf{f}(t), \tag{2.2}$$

where $\vec{\varphi}$ and $\mathbf{f}$ are vectors, while $\mathbf{A_x}$ and $\mathbf{A_y}$ are basis matrices of the digital Laplacian. The forms of the matrices depend on the method of numeration of nodes in the grid. All their diagonal elements are –2, and they have not more than two offdiagonal elements $A_{ij} = 1$, where j is the number of a neighbour by x and y respectively. For example, if one numbers nodes consequently within each raw (y = const) and then rows (also consequently), $\mathbf{A_x}$ will be tridiagonal. Each $\mathbf{A_x}$ and $\mathbf{A_y}$ can be reduced to the tridiagonal form by simultaneous permutation of rows and columns (that is change of numeration of grid nodes), but not simultaneously. Both are always sparse.

Next choose a step τ for t and apply an A-stable one-stage Rosenbrock scheme of the second order [2]

$$\vec{\varphi}(t+\tau) = \vec{\varphi}(t) + \tau\left(\mathbf{I} - \frac{\tau}{2} D\left(\frac{\mathbf{A_x}}{\Delta x^2} + \frac{\mathbf{A_y}}{\Delta y^2}\right)\right)^{-1} D\left(\frac{\mathbf{A_x}}{\Delta x^2} + \frac{\mathbf{A_y}}{\Delta y^2}\right)\vec{\varphi}(t) + \tag{2.3}$$

$$+ \tau\mathbf{f}(t+\tau/2).$$

The part $\mathbf{f}(t)$ not depending on $\vec{\varphi}$ does not affect the stability, so the Rosenbrock scheme does not have to cover it. One needs to factorize $\left(\mathbf{I} - \frac{\tau}{2} D\left(\frac{\mathbf{A_x}}{\Delta x^2} + \frac{\mathbf{A_y}}{\Delta y^2}\right)\right)$ to make a step by this scheme. Although the matrix is sparse, it corresponds to a flat graph, and the procedure seems to be too expensive. Let us try to find a more effective way to solve (2.3). Decompose Rosenbrock corrector into two parts:

$$\vec{\varphi}(t+\tau) = \vec{\varphi}(t) + \tag{2.4}$$

$$+ \tau\left(\mathbf{I} - \frac{\tau}{2} D\frac{\mathbf{A_x}}{\Delta x^2}\right)^{-1}\left(\mathbf{I} - \frac{\tau}{2} D\frac{\mathbf{A_y}}{\Delta y^2}\right)^{-1} D\left(\frac{\mathbf{A_x}}{\Delta x^2} + \frac{\mathbf{A_y}}{\Delta y^2}\right)\vec{\varphi}(t) + \tau\mathbf{f}(t+\tau/2).$$

Check stability properties first. $\mathbf{f}(t)$ does not affect stability, so we omit it. As usually, regard the grid as infinite by both coordinates. Consider a space harmonic $(k_x)^i (k_y)^j$, where $|k_x| = |k_y| = 1$, and i and j are indices of nodes by x and y respectively. Then a step (2.4) is equivalent to multiplication of $\vec{\varphi}$ by

$$\alpha = 1 + D\tau\frac{(k_x - 1)^2/(k_x\Delta x^2) + (k_y - 1)^2/(k_y\Delta y^2)}{\left(1 - D\tau(k_x - 1)^2/2(k_x\Delta x^2)\right)\left(1 - D\tau(k_y - 1)^2/2(k_y\Delta y^2)\right)} = \tag{2.5}$$

$$= \frac{(1 - D_x)(1 - D_y)}{(1 + D_x)(1 + D_y)},$$

where $D_{x,y} = D\tau(1 - \mathrm{Re}\, k_{x,y})/(\Delta x, y)^2$. Then

$$|\alpha| = \left|\frac{(1 - D_x)(1 - D_y)}{(1 + D_x)(1 + D_y)}\right|. \tag{2.6}$$

Note that (2.6) consists of two multipliers depending on k_x or k_y, so we can analyze them separately and similarly.

$$\left|\frac{(1-D_x)}{(1+D_x)}\right| = \frac{\sqrt{\xi^2-\xi+\eta^2+1}}{\sqrt{\xi^2+\xi+\eta^2+1}}, \tag{2.7}$$

where $\xi = \mathrm{Re}\, D_x$ and $\eta = \mathrm{Im}\, D_x$. It is obvious that the numerator is always less than or equal to the denominator if $\xi \geq 0 \Leftrightarrow \mathrm{Re}\, D \geq 0$. In other case the initial equation (2.1) has infinitely increasing solutions. Thus $|\alpha| \leq 1$ and the modified scheme (2.4) is A-stable.

Let us check the approximation order of (2.4).

$$\left(\mathbf{I}-\frac{\tau}{2}(\mathbf{J_1}+\mathbf{J_2})\right)^{-1} = \left(\left(\mathbf{I}-\frac{\tau}{2}\mathbf{J_1}\right)\left(\mathbf{I}-\frac{\tau}{2}\mathbf{J_2}\right)-\frac{\tau^2}{4}\mathbf{J_1}\mathbf{J_2}\right)^{-1} \cong \tag{2.8}$$

$$\cong \left(\mathbf{I}-\frac{\tau}{2}\mathbf{J_2}\right)^{-1}\left(\mathbf{I}-\frac{\tau}{2}\mathbf{J_1}\right)^{-1},$$

where $\mathbf{J_1} = D\mathbf{A_y}/\Delta y^2$ and $\mathbf{J_2} = D\mathbf{A_x}/\Delta x^2$ are the Jacobian matrices of the spitted right part. The neglected term has the order τ^3 (together with τ combined with the right part in the scheme), and does not affect the approximation order. This is true only if $\mathbf{J_1}$ and $\mathbf{J_2}$ have equal eigenvectors, $(k_x)^i(k_y)^j$ in our case. Than also $\mathbf{J_1}\mathbf{J_2} = \mathbf{J_2}\mathbf{J_1}$. If $\mathbf{J_1}$ and $\mathbf{J_2}$ have eigenvalues g_1 and g_2 for some eigenvector than the residual part of the scheme is $\tau^3(g_1^3+g_2^3)/12$.

2.2. Factorization of infinite matrices

Suppose that the matrices in (2.8) are infinite, namely the area in xy-plane is infinite. Let us apply Gaussian elimination method by rows top-down to the each of them. If the initial matrix is

$$\begin{pmatrix} . & . & . & . & . & . & . & . \\ \dots & 0 & b & a & b & 0 & 0 & \dots \\ \dots & 0 & 0 & b & a & b & 0 & \dots \\ . & . & . & . & . & . & . & . \end{pmatrix}, \tag{2.9}$$

and the stationary state after elimination of some part of upper rows is

$$\begin{pmatrix} . & . & . & . & . & . & . \\ \dots & 0 & c & b & 0 & 0 & \dots \\ \dots & 0 & b & a & b & 0 & \dots \\ . & . & . & . & . & . & . \end{pmatrix}, \tag{2.10}$$

where the third row is to be eliminated in this step. The condition

$$a - b^2 / c = c \tag{2.11}$$

is equivalent to the process stationarity. (2.11) has two solutions

$$c = 1/2\left(a \pm \sqrt{a^2 - 4b^2}\right), \tag{2.12}$$

and only one with "+" sign gives a stable process. In our case

$$a = 1 + \frac{\tau D}{\Delta\xi^2}, b = -\frac{\tau D}{2\Delta\xi^2}, \tag{2.13}$$

where ξ is x or y.

Thus we do not need to factorize the matrix in (2.3) explicitly. We only are to apply the following procedure for x and y coordinates consequently instead:

$$\left.\begin{array}{l} 1^0.\ \varphi_i \leftarrow \varphi_i / c; \\ 2^0.\ \varphi_{i+1} \leftarrow \varphi_{i+1} - b\varphi_i; \end{array}\right\}\ i = 1...N \\ 3^0.\ \varphi_i \leftarrow \varphi_i - b/c \cdot \varphi_{i+1};\ i = N...1 \tag{2.14}$$

where φ_i and φ_{i+1} means the function values in the neighboring nodes spaced by Δx or Δy. "←" means "set the value to the storage cell". The procedure is applied to each line of the grid by x and y independently. Steps 1^0 and 2^0 are executed in one direction, while 3^0 in the opposite. The order of x- and y-stages effects on the result, but not on the approximation order.

In the long run, we obtained the algorithm of N time complexity, where N is the number of nodes in the grid. The method can be easily extended to any spatial dimension, and the time complexity remains N. Compare to the most effective algorithms for sparse matrices [3]: the factorization time complexities of sparse matrices are $N^{3/2}$ for 2D grids, N^2 for 3D grids, etc. (Table 4.1). Note that boundary conditions can not be applied reasonably in our algorithm, as even a huge finite matrix being factorized differs from infinite one at the edges.

2.3. Boost of the approximation order

The approximation order of our algorithm is two that seems not so good. Let us try to improve this feature using the Numerov's formula [4] Chapter III (10.8)

$$\frac{1}{12}u''(x-\Delta x)+\frac{5}{6}u''(x)+\frac{1}{12}u''(x+\Delta x)\sim\frac{u(x-\Delta x)-2u(x)+u(x+\Delta x)}{\Delta x^2}. \qquad (2.15)$$

The formula permits to calculate the second derivative of a function and has fourth approximation order. Its drawback is that it is implicit. In our case this drawback does not matter at all, as application of (2.15) to (2.1) means change of matrices coefficients in (2.4) only.

Let us define two Numerov's matrices $\mathbf{A}_x^N$ and $\mathbf{A}_y^N$ similar to $\mathbf{A_x}$ and $\mathbf{A_y}$ above: all their diagonal elements are 5/6, and they have not more than two offdiagonal elements $A_{ij} = 1/12$, where j is the number of a neighbour by x and y respectively. Then a forth order approximation of the second derivative is

$$\frac{\partial^2 u}{\partial\xi^2}\sim\left(\mathbf{A}_\xi^N\right)^{-1}\frac{\mathbf{A}_\xi}{\Delta\xi^2}, \qquad (2.16)$$

where ξ is x or y. Substituting (2.12) into (2.1) one obtains

$$\left[\frac{d}{dt}-D\left(\left(\mathbf{A}_x^N\right)^{-1}\frac{\mathbf{A_x}}{\Delta x^2}+\left(\mathbf{A}_y^N\right)^{-1}\frac{\mathbf{A_y}}{\Delta y^2}\right)\right]\vec{\varphi}(t)=\mathbf{f}(t) \qquad (2.17)$$

instead of (2.2). The Rosenbrock scheme transforms it into

$$\begin{aligned}\vec{\varphi}(t+\tau)=\vec{\varphi}(t)+\tau&\left(\mathbf{I}-\frac{\tau}{2}D\left(\left(\mathbf{A}_x^N\right)^{-1}\frac{\mathbf{A_x}}{\Delta x^2}+\left(\mathbf{A}_y^N\right)^{-1}\frac{\mathbf{A_y}}{\Delta y^2}\right)\right)^{-1}\times\\ &\times D\left(\left(\mathbf{A}_x^N\right)^{-1}\frac{\mathbf{A_x}}{\Delta x^2}+\left(\mathbf{A}_y^N\right)^{-1}\frac{\mathbf{A_y}}{\Delta y^2}\right)\vec{\varphi}(t)+\\ &+\tau\mathbf{f}(t+\tau/2).\end{aligned} \qquad (2.18)$$

and

$$\vec{\varphi}(t+\tau)=\vec{\varphi}(t)+\tau\left(\mathbf{I}-\frac{\tau}{2}D\left(\mathbf{A}_x^N\right)^{-1}\frac{\mathbf{A_x}}{\Delta x^2}\right)^{-1}\left(\mathbf{I}-\frac{\tau}{2}D\left(\mathbf{A}_y^N\right)^{-1}\frac{\mathbf{A_y}}{\Delta y^2}\right)^{-1}\times \tag{2.19}$$

$$\times D\left(\left(\mathbf{A}_x^N\right)^{-1}\frac{\mathbf{A_x}}{\Delta x^2}+\left(\mathbf{A}_y^N\right)^{-1}\frac{\mathbf{A_y}}{\Delta y^2}\right)\vec{\varphi}(t)+\tau\mathbf{f}(t+\tau/2)=$$

$$=\vec{\varphi}(t)+\tau\left(\mathbf{A}_x^N-\frac{\tau}{2}D\frac{\mathbf{A_x}}{\Delta x^2}\right)^{-1}\left(\mathbf{A}_y^N-\frac{\tau}{2}D\frac{\mathbf{A_y}}{\Delta y^2}\right)^{-1}\times$$

$$\times D\left(\mathbf{A}_y^N\frac{\mathbf{A_x}}{\Delta x^2}+\mathbf{A}_x^N\frac{\mathbf{A_y}}{\Delta y^2}\right)\vec{\varphi}(t)+\tau\mathbf{f}(t+\tau/2).$$

after decomposition. Commutativity of *x*- and *y*- matrices was used above. The coefficients are now

$$a=\frac{5}{6}+\frac{\tau D}{\Delta\xi^2},b=\frac{1}{12}-\frac{\tau D}{2\Delta\xi^2}\,. \tag{2.20}$$

Two additional products by $\mathbf{A}_x^N$ and $\mathbf{A}_y^N$ are to be calculated in this case.

In comparison with (2.4), we only improved the approximation order for spatial coordinates, so we do not need to check those of the whole scheme once more. However, we should analyze its stability properties. Let us make a similar formula manipulation as (2.4) → (2.5).

$$\alpha=1+\frac{D\tau}{12}\frac{(k_y^2+10k_y+1)(k_x-1)^2/(k_xk_y\Delta x^2)+}{\left((k_x^2+10k_x+1)/12k_x-D\tau(k_x-1)^2/(2k_x\Delta x^2)\right)\times}\rightarrow \tag{2.21}$$

$$\rightarrow\frac{+(k_x^2+10k_x+1)(k_y-1)^2/(k_yk_y\Delta y^2)}{\times\left((k_y^2+10k_y+1)/12k_y-D\tau(k_y-1)^2/(2k_y\Delta y^2)\right)}=$$

$$=\frac{(N_x-D_x)(N_y-D_y)}{(N_x+D_x)(N_y+D_y)},$$

where $D_{x,\ y}$ were defined above, and $N_{x,y}=5/6+1/6\,\mathrm{Re}\,k_{x,y}$. Substituting $d_x=D_x/N_x$ and $d_y=D_y/N_y$ we find

$$|\alpha|=\left|\frac{(1-d_x)(1-d_y)}{(1+d_x)(1+d_y)}\right|. \tag{2.22}$$

$\mathrm{Re}\,d_{x,y}\geq 0$ for the same reason as $\mathrm{Re}\,D_{x,y}\geq 0$, so the new scheme is also A-stable.

3. Numerical test

Let us use the well-known equation for paraxial wave propagation [5] (2.21)

$$\left[2ik\frac{\partial}{\partial z}+\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)\right]\varphi=0\,. \tag{3.1}$$

in free space without sources. This equation has an analytical solution for the fundamental Gaussian mode, so we can easily evaluate the accuracy. The exact solution is

$$|u|=u_0\frac{w}{w_0}\exp(-r^2/w^2)\,, \tag{3.2}$$

where u_0 is the initial amplitude, w_0 is the initial size, $r^2=x^2+y^2$, and

$$w^2=w_0^2\left[1+\left(\frac{2z}{kw_0^2}\right)^2\right]. \tag{3.3}$$

The radius of curvature of the wavefront is

$$R=z\left[1+\left(\frac{kw_0^2}{2z}\right)^2\right]. \tag{3.4}$$

In the one-dimensional case the equation is

$$\left(2ik\frac{\partial}{\partial z}+\frac{\partial^2}{\partial x^2}\right)\varphi=0\,, \tag{3.5}$$

and its exact solution

$$|u|=u_0\sqrt{\frac{w}{w_0}}\exp(-x^2/w^2)\,. \tag{3.6}$$

Every time numerical simulation was conducted in the interval $z=[0,kw_0^2/2]$ and the grid half-size was $4.5w_0$. The purpose of the numerical test was not detailed investigation of the algorithm properties for various equations, but only a proof of its workability and stability, and an estimation of its accuracy. The results for the basic scheme and the improved one are collected in Table 3.1 and Table 3.2 respectively.

Table 3.1. Accuracy of the basic scheme.

Nodes per grid half-size	Steps by z	$\lvert\Delta u/u\rvert$
One dimension		
60	32	$8\cdot10^{-4}$
60	25	$8.3\cdot10^{-4}$
60	16	$1\cdot10^{-3}$
60	8	$2\cdot10^{-3}$
60	4	$6\cdot10^{-3}$
60	2	$1.8\cdot10^{-2}$
60	1	$4.6\cdot10^{-2}$
15	32	$1.1\cdot10^{-2}$
30	32	$3\cdot10^{-3}$
120	32	$2.5\cdot10^{-4}$
Two dimensions		
60	32	$1\cdot10^{-3}$
60	16	$1.2\cdot10^{-3}$
60	8	$1.9\cdot10^{-3}$
42	32	$2\cdot10^{-3}$
30	32	$4\cdot10^{-3}$
15	8	$1.7\cdot10^{-2}$
84	32	$5.6\cdot10^{-4}$

It is clear from Table 3.1 that the basic scheme is stable for all the reasonable ratios $\Delta z/\Delta x$. For example, $\Delta z / k\Delta x^2 \approx 90$ for one step by z and 60 nodes per grid half-size, and the scheme still gives quite reasonable solution in this case. For reference, the Euler method applied to (3.5) → (1.6) is unstable for any $\Delta z / k\Delta x^2$, as in this case

$$\max|\alpha| = \left|1 + 2i\Delta z / k\Delta x^2\right| > 1 . \tag{3.7}$$

For the most well-known Kutta scheme [4] Chapter II Table 1.2

$$|\alpha| = \frac{1}{24}\sqrt{\zeta^8\left(\frac{\Delta z}{k\Delta x^2}\right)^8 - 8\zeta^6\left(\frac{\Delta z}{k\Delta x^2}\right)^6 + 576} , \tag{3.8}$$

where $\zeta \in [0,2]$. In this case $\max|\alpha| > 1$ if $\Delta z / k\Delta x^2 > \sqrt{2}$.

The dependence of the relative accuracy of the basic scheme on the number of steps is placed in Fig. 3.1. The data from Table 3.1 were used. One can see the error decreases a little more slowly than the number of steps to the minus two. It means the "practical" approximation order in this case is a little less than two. The

dependence of the accuracy on the number of nodes is represented in Fig. 3.2. The "practical" approximation order here is exactly four.

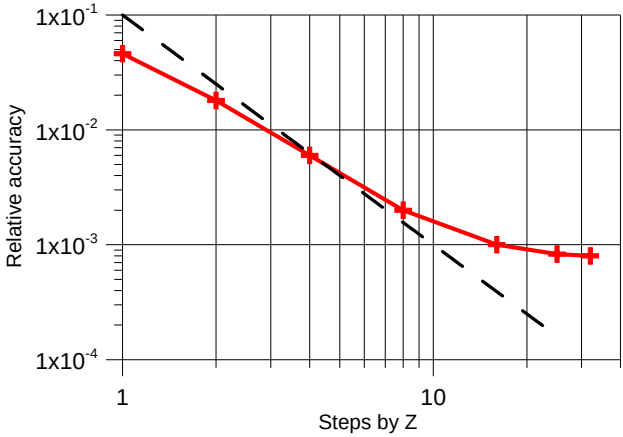

Fig. 3.1. Basic scheme accuracy vs number of integration steps (solid). $0.1/N^2$ (dashed).

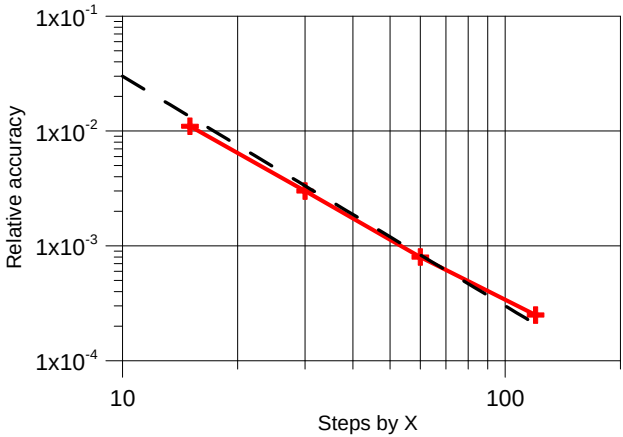

Fig. 3.2. Basic scheme accuracy vs number of nodes per grid half-size (solid). $3/N^2$ (dashed).

The results for the improved scheme are placed in Table 3.2 and Fig. 3.3. One can see that the "practical" approximation for *xy* is four and the accuracy is limited by Δz.

Table 3.2. Accuracy of the improved scheme. Two dimensions.

Nodes per grid half-size	Steps by z	$\lvert\Delta u/u\rvert$
60	32	$8.8\cdot10^{-5}$
30	32	$1.2\cdot10^{-4}$
15	32	$6.5\cdot10^{-4}$
8	32	$8.3\cdot10^{-3}$

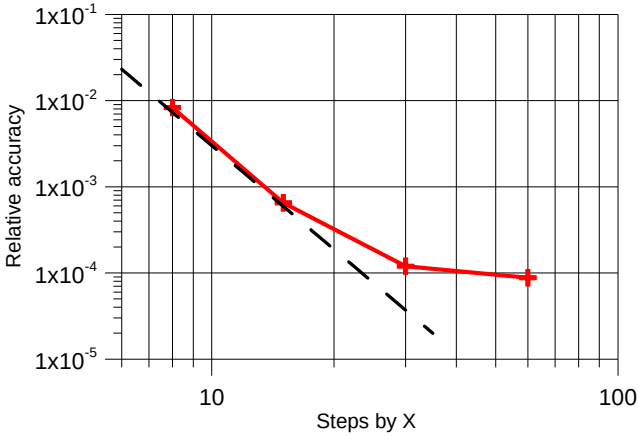

Fig. 3.3. Improved scheme accuracy vs number of nodes per grid half-size (solid). $30/N^4$ (dashed).

4. Discussion and conclusions

Thus we obtained an effective numerical scheme for parabolic partial differential equations of arbitrary dimension and no boundary condition. Its spatial approximation order is four and the temporal one is two. It is stable for arbitrary ratio of spatial and temporal steps. The scheme has been numerically tested, and the tests proved its operability.

The spatial approximation order can be enhanced using, say, five-node numerical Laplacian approximation with a Numerov-like corrector. Hardly it makes any practical sense. The temporal order can be easily raised using a higher order Rosenbrock scheme. In this case it is not clear how to include the part $\mathbf{f}(t)$ not depending on $\vec{\varphi}$ with the same order.

References

[1] E. Hairer, G. Wanner. Solving Ordinary Differential Equation II: Stiff and Differential-Algebraic Problems. - Springer-Verlag, 1996.

[2] H.H. Rosenbrock. Some general implicit processes for the numerical solution of differential equations. Comput. // J., **5** (1963), 329-330.

[3] S. Pissanetzky. Sparse matrix technology. // Academic Press, New York, USA, 1984.

[4] E. Hairer, S.P. Nørsett, G. Wanner. Solving ordinary differential equations I: Nonstiff problems. - Berlin, New York: Springer-Verlag, 1993.

[5] M. Levy. Parabolic Equation Methods for Electromagnetic Wave Propagation. - The Institution of Electrical Engineers, London, UK, 2000.

S.V. Miginsky

An Effective Numerical Scheme
for Parabolic Partial Differential Equations

С.В. Мигинский

Эффективная численная схема
для параболических дифференциальных уравнений
в частных производных

ИЯФ 2011-25

Ответственный за выпуск А.В. Васильев
Работа поступила 1.09. 2011 г.

Сдано в набор 5.09. 2011 г.
Подписано в печать 5.09. 2011 г.
Формат 60x90 1/16 Объем 0,9 печ.л.,0.7 уч.-изд.л.
Тираж 90 экз. Бесплатно. Заказ № 25

Обработано на РС и отпечатано
на ротапринте «ИЯФ им. Г.И. Будкера» СО РАН,
Новосибирск, 630090, пр. Академика Лаврентьева, 11