Journal of Computational Physics **173,** 716–729 (2001)
doi:10.1006/jcph.2001.6909, available online at http://www.idealibrary.com on IDEAL®

Simple and Accurate Scheme for Nonlinear Convection–Diffusion Equation

A. A. Kulikovsky[1]

Institute for Materials and Processes in Energy Systems (IWV-3), Research Center
"Jülich," D-52425 Jülich, Germany
E-mail: akul@srcc.msu.su

Received March 9, 2001; revised August 6, 2001

A simple nonlinear scheme for iterative solution of nonlinear convection–diffusion equation is described. The scheme is tested by solutions of three nonlinear steady-state model equations and linear nonstationary transport equation. The feature of the scheme is transition from a second-order accuracy on coarse grids to a first-order on fine grids. © 2001 Academic Press

1. INTRODUCTION

In many areas of physics, a flux is determined by diffusion and convection (or drift). The diffusion coefficient and convection velocity may depend on an unknown variable, which introduces nonlinearity to the problem. Examples include nonlinear heat transfer problems, transport of charged species in semiconductors and plasmas, and the Fokker–Plank equation [1–3].

Recently, it became clear that water transport through the polymer electrolyte membranes (PEM) obeys a similar equation [4, 5]. The latter problem is of large interest since PEM is a key component of fuel cells, which are expected to perform revolution in mobile power sources including vehicles.

The steady-state nonlinear convection–diffusion equation is basically the conservation law

$$\nabla \cdot \mathbf{N} = R, \tag{1}$$

where the flux $\mathbf{N}$ has a form

$$\mathbf{N} = -D(c)\nabla c + \mathbf{U}(c). \tag{2}$$

[1] On leave from Moscow State University, Research Computing Center (NIVC), 119899 Moscow, Russia.

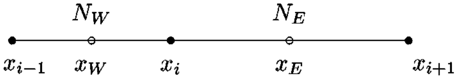

FIG. 1. Computational cell. Black dots are grid nodes; hollow dots denote cell surfaces.

Here c is usually a concentration or number density of species, D is the diffusion coefficient, $\mathbf{U}$ is convective flux, and R represents sources. The numerical solution of (1) and (2) has been the subject of numerous works (see, e.g., [3, 6] and cited literature therein).

In modeling of semiconductors and gas discharges, a diffusion–drift model is widely used, which leads to the equation of the type of (1) and (2), with given D and $\mathbf{U}(c) = \mathbf{W}(\mathbf{r})c$, where $\mathbf{W}(\mathbf{r})$ is a drift velocity. A one-dimensional variant of the flux in the diffusion–drift model is

$$N = -D(x)\frac{\partial c}{\partial x} + v(x)c, \tag{3}$$

where v is the x-component of drift velocity. Some thirty years ago, Scharfetter and Gummel [7] proposed a scheme for the numerical solution of Eq. (1) with the flux (3). Their scheme is based on the following idea.[2]

Let a nonuniform grid with the nodes $\{x_i, i = 0, \ldots, M-1\}$ covers computational domain. The computational cell is bounded by the “west” and “east” points x_W, x_E, respectively (Fig. 1). The expression for the flux (3) is considered as the equation for $c(x)$ on the interval between two adjacent nodes $[x_i, x_{i+1}]$. The coefficients in (3) and the flux N are assumed to be constant on this interval and equal to their values at the “east” boundary of the cell: $N = N_E$, $v = v_E$, and $D = D_E$ (Fig. l).

Solving (3) with the boundary condition $c(x_i) = c_i$ and substituting $x = x_{i+1}$, $c = c_{i+1}$ into the solution, one obtains the Scharfetter–Gummel (SG) flux

$$N_E^{SG} = \frac{v_E}{\exp\left(\frac{v_E h_i}{D_E}\right) - 1}\left[\exp\left(\frac{v_E h_i}{D_E}\right)c_i - c_{i+1}\right]. \tag{4}$$

This scheme was widely used in semiconductor devices simulation [10]. Recently, it was introduced into gas-discharge physics [11]. A remarkable property of that scheme is that it gives a monotonic solution [12]. A more accurate adaptive variant of the SG scheme was offered in [13] and used in modeling of high-voltage breakdown in air [14] .

In this work we present a new scheme for solution of (1) and (2), which employs an extension of the Allen/Scharfetter–Gummel idea to the nonlinear variant of the convection–diffusion equation. We expand the functions $D(c)$ and $U_x(c)$ in (2) (U_x is the x-component of convective flux) near the node x_i, and the resulting equation is then solved on the interval $[x_i, x_{i+1}]$. This gives expression to the flux, which contains derivatives $\partial D/\partial x$ and $\partial U_x/\partial x$. Equation (1) is then solved by iterations, with the derivatives calculated using values from the previous iteration. Four tests demonstrate the properties of the scheme.

[2] This idea was first offered by Allen [8] in 1955 for fluid dynamics equations and then rediscovered in other fields (see [9]).

2. THE PHYSICAL BACKGROUND: WATER TRANSPORT IN MEMBRANE

First, we will construct the scheme for the following simplest physical model. In polymer electrolyte fuel cell water transport through the membrane is determined by diffusion because of the concentration gradient and electroosmotic drag [15]. The latter arises due to the flux of protons through the membrane, which forces water to move in the same direction.

The flux **N** of water in membrane is

$$\mathbf{N} = -D(c)\nabla c + n_d(c)\frac{\mathbf{j}}{F}, \tag{5}$$

where $D(c)$ is the diffusion coefficient of water in the membrane, c is the water molar concentration, n_d is the drag coefficient, $\mathbf{j}$ is the local proton current density, and F is the Faraday constant.

The water is neither produced nor consumed in the membrane, and hence the mass conservation equation reads as

$$\nabla \cdot \mathbf{N} = 0. \tag{6}$$

Both the diffusion and the drag coefficients are proportional to water content λ, a number of water molecules per one sulfonic group SO_3^- in membrane [15, 16],

$$D(c) = D^T\lambda(c), \quad n_d(c) = n_d^0\lambda(c), \tag{7}$$

where n_d^0 is constant and the factor D^T describes the temperature dependence of diffusion coefficient (hereinafter D^T is constant). In polymer electrolyte membranes, λ is a function of water molar concentration c [15].

Taking (7) into account, (5) transforms to

$$\frac{\mathbf{N}}{D^T} = -\lambda(c)\nabla c + \mathbf{b}\lambda(c), \tag{8}$$

where $\mathbf{b} = n_d^0\mathbf{j}/(D^TF)$.[3]

3. THE SCHEME

3.1. Flux Through the Cell Surface

We start with the one-dimensional variant of (8),

$$-\lambda(c)\frac{\partial c}{\partial x} + b\lambda(c) = \frac{N}{D^T}, \tag{9}$$

where $b = n_d^0 j_x/(D^TF)$.

Let the nonuniform grid with the nodes $\{x_i, i = 0, \ldots, M-1\}$ cover the computational domain for (9). The distance between adjacent nodes is $h_i = x_{i+1} - x_i$. The computational cell is bounded by the points x_W and x_E, separated by a distance $(h_i + h_{i-1})/2$ (Fig. 1).

[3] In general, current density also depends on water content, so that $\mathbf{b} = \mathbf{b}(c)$. This case is considered in Section 3.2.

Numerical approximation of the flux through the cell surface is based on the following idea. Since the cell size is small, the variation of λ across the cell is not large. We expand the function $\lambda(c)$ near the node x_i:

$$\lambda \simeq \lambda_i + q_i(x - x_i), \quad q_i \equiv \left.\frac{\partial \lambda}{\partial x}\right|_{x=x_i}. \tag{10}$$

Substituting (10) for (9) we get

$$-(\lambda_i + q_i(x - x_i))\frac{\partial c}{\partial x} + b(\lambda_i + q_i(x - x_i)) = \frac{N}{D^T}. \tag{11}$$

Following the idea of Scharfetter and Gummel [7], we consider (11) as ODE with respect to $c(x)$ on the interval $[x_i, x_{i+1}]$, and we assume that $b = b_E$ and $N = N_E$, where b_E and N_E are the values at the "east" cell boundary (Fig. 1). The solution of this equation, subject to boundary condition $c(x_i) = c_i$, is

$$c(x) = c_i + b_E(x - x_i) - \frac{N_E}{D^T q_i}\ln\left(1 + \frac{q_i(x - x_i)}{\lambda_i}\right). \tag{12}$$

Substituting $x = x_{i+1}$ into (12), we obtain the flux N_E:

$$N_E = -\frac{D^T q_i}{\ln\left(1 + \frac{q_i h_i}{\lambda_i}\right)}(c_{i+1} - b_E h_i - c_i). \tag{13}$$

Since q_i depends on c, this is nonlinear flux.

When

$$\frac{q_i h_i}{\lambda_i} \to 0,$$

the logarithm in the denominator of (13) can be expanded, and we get

$$N_E = -\frac{D^T \lambda_i}{h_i}(c_{i+1} - c_i) + D^T b_E \lambda_i, \tag{14}$$

which is simply (9) with the calculation of convective term at the node i ("upwind" scheme). This expansion is valid if either $q_i \to 0$ or $h_i \to 0$. Note that here $q_i = 0$ means that both the diffusion coefficient and the convective flux are constant. In that case, for any h_i, (14) gives the *exact* value of convective flux and provides a second-order approximation of diffusion flux, i.e., (14) has second-order accuracy.

However, if $q_i \neq 0$, then for small h_i the scheme transforms to the first-order (upwind) one. Below, this property of the scheme is considered in more detail.

3.2. A More General Form of the Flux

A more general form of the flux (9) is

$$-\lambda(c)\frac{\partial c}{\partial x} + b\gamma(c) = \frac{N}{D^T}, \tag{15}$$

that is, dependencies of the diffusion coefficient and the convective flux on c are different. Expanding both $\lambda(c)$ and $\gamma(c)$ near x_i and taking $b = b_E$, $N = N_E$, we get

$$-(\lambda_i + q_i(x - x_i))\frac{\partial c}{\partial x} + b_E(\gamma_i + p_i(x - x_i)) = \frac{N_E}{D^T}, \tag{16}$$

where

$$q_i \equiv \left.\frac{\partial \lambda}{\partial x}\right|_{x=x_i}, \quad p_i \equiv \left.\frac{\partial \gamma}{\partial x}\right|_{x=x_i}. \tag{17}$$

Solving this equation and substituting $c = c_{i+1}$ and $x = x_{i+1}$ in the solution, we obtain the general expression for the flux:

$$N_E = -\frac{D^T q_i}{\ln\left(1 + \frac{q_i h_i}{\lambda_i}\right)}\left(c_{i+1} - b_E h_i \frac{p_i}{q_i} - c_i\right) + D^T b_E\left(\gamma_i - \frac{p_i}{q_i}\lambda_i\right). \tag{18}$$

As q_i tends to zero (constant diffusion coefficient), the flux (18) reduces to

$$N_E^{q\to 0} = \lim_{q_i \to 0} N_E = -\frac{D^T \lambda_i}{h_i}(c_{i+1} - c_i) + D^T b_E\left(\gamma_i + \frac{p_i h_i}{2}\right), \tag{19}$$

which can also be received directly from (16) with $q_i = 0$.

In the limit of the constant diffusion coefficient, the scheme, hence, always has second-order accuracy. Indeed, the diffusion flux (first term on the right-hand side of (19)) is approximated with the centered formula, which gives second-order accuracy on the uniform grid. The second (convective term) in (19) contains

$$\gamma_i + \left.\frac{\partial \gamma}{\partial x}\right|_i \frac{h_i}{2},$$

which evidently has second-order accuracy. One may expect, that second-order accuracy is retained for finite q_i.

If $q_i \neq 0$ and $h_i \to 0$, the logarithm in (18) can be expanded and we obtain

$$N_E^{h\to 0} = \frac{D^T \lambda_i}{h_i}(c_i - c_{i+1}) + D^T b_E \gamma_i, \tag{20}$$

which is a first-order upwind scheme, since the convective term is approximated at the node i. The relations (19) and (20) suggest that if the diffusion coefficient is not constant, accuracy of the scheme changes from second order to first order as the cell size diminishes. Numerical examples (Section 4) confirm this assumption.

3.3. Model Equation of Water Transport in Membrane

Consider a model equation

$$\frac{\partial}{\partial x}\left(-c\frac{\partial c}{\partial x} + c\right) = 0 \tag{21}$$

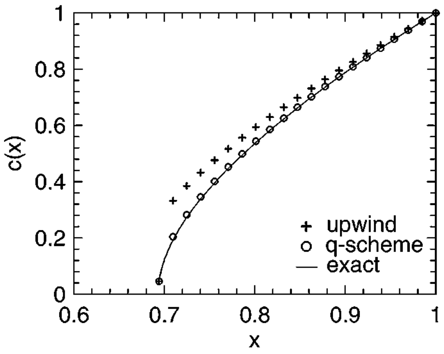

FIG. 2. Solutions of the model problem (21). Solid line: exact solution; crosses: upwind scheme; circles: q-scheme. The domain is [ln 2 + 0.001, 1]; uniform grid; number of cells 20; number of iterations 10.

on the interval $x \in [0, 1]$. We seek the solution of (21) subject to initial conditions

$$\left.\frac{\partial c}{\partial x}\right|_{x=1} = 2, \quad c(1) = 1. \tag{22}$$

Equation (21) is a 1D version of (6) and (9), with $b = D^T = 1$ and $\lambda(c) \equiv c$. Physically (21) means that the diffusion coefficient is proportional to the local value of concentration c. Initial conditions (22) fix the concentration and flux at $x = 1$.

The first integral of (21) is

$$-c\frac{\partial c}{\partial x} + c = -1, \quad c(1) = 1. \tag{23}$$

The solution of (23) cannot be expressed in terms of elementary functions. However, there exists a simple solution of the problem for function $x(c)$. The equation for $x(c)$ immediately follows from (23),

$$(1 + c)\frac{\partial x}{\partial c} - c = 0, \quad x(1) = 1,$$

with the solution

$$x = c - \ln(1 + c) + \ln 2. \tag{24}$$

The function $c(x)$, obtained from (24), is shown in Fig. 2. The feature of that problem is that below the point $x_{\lim} = \ln 2$, the problem has no solutions. At $x_{\lim}$, the concentration vanishes along with the diffusion coefficient. The transport below $x_{\lim}$ is, therefore, forbidden.

4. NUMERICAL RESULTS

4.1. Problem 1

To demonstrate the properties of the scheme, we slightly reformulate problems (21) and (22). In practice, the flux at $x = 1$ is unknown, and the concentrations at the boundaries are given instead. This leads to the boundary value problem. For test purposes we will formulate the boundary value problem for (21), whose solution coincides with the solution of the Cauchy problem (23). Namely, we will seek the solution of (21) in the domain $[\ln 2 + \varepsilon, 1]$, subject to boundary conditions

$$c(\ln 2 + \varepsilon) = c_0; \quad c(1) = 1, \tag{25}$$

with $\varepsilon = 0.001$. The small positive ε excludes the point $x = \ln 2$, with the infinite derivative $\partial c/\partial x$ from computational domain. The point with coordinates $x_0 = \ln 2 + 0.001$ and $c_0 = 0.0453905$ lies on the curve (24), that is, the solution of problems (21) and (25) coincides with the solution of (24).

We introduce grid $\{x_i, i = 0, \ldots, M-1\}$ in the domain $x \in [\ln 2 + \varepsilon, 1]$. To solve (21) numerically, we will use two schemes. The first one, which will be referred to as the *q-scheme* is based on (13). Zero divergence means that in every cell $N_E - N_W = 0$. Taking (13) into account we get

$$-F_{i-1}^m\, c_{i-1}^{m+1} + \left(F_{i-1}^m + F_i^m\right) c_i^{m+1} - F_i^m\, c_{i+1}^{m+1} = F_{i-1}^m h_{i-1} - F_i^m h_i, \quad i = 1, \ldots, M-2, \tag{26}$$

where m enumerates iterations,

$$F_i^m = \frac{q_i^m}{\ln\left(1 + \frac{q_i^m h_i}{\lambda_i^m}\right)}, \tag{27}$$

and

$$\lambda_i^m \equiv \lambda\left(c_i^m\right), \quad q_0^m = \frac{c_1^m - c_0^m}{h_0}, \quad q_i^m = \frac{c_{i+1}^m - c_{i-1}^m}{h_{i-1} + h_i} \quad (i = 1, \ldots, M-2).$$

The “upwind” scheme (14) leads to the same system (26) with coefficients

$$F_i^m = \frac{\lambda_i^m}{h_i}. \tag{28}$$

Boundary conditions (25) give two lacking equations for $i = 0$ and $i = M-1$. The relations (26) along with the boundary conditions form a system of linear equations with a three-diagonal matrix, which can be solved on each iteration step by the Thomas algorithm.

The solutions are shown in Fig. 2 along with the exact solution (24). It is seen that the q-scheme provides much better accuracy.

4.2. Problem 2

As a second test, we consider a problem with quadratic dependence $\lambda = c^2$,

$$\frac{\partial}{\partial x}\left(-c^2 \frac{\partial c}{\partial x} + c^2\right) = 0; \quad \left.\frac{\partial c}{\partial x}\right|_{x=1} = 2, \quad c(1) = 1, \tag{29}$$

on the interval $x \in [0, 1]$. The analysis, quite analogous to those in previous section, gives the implicit solution

$$x = c - \arctan(c) + \frac{\pi}{4}. \tag{30}$$

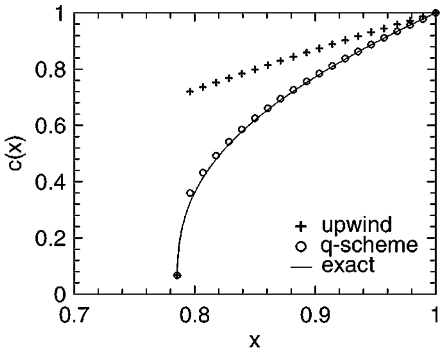

FIG. 3. Solutions of problem (29). Solid line: exact solution; crosses: upwind scheme; circles: q-scheme. The domain is $[\frac{\pi}{4}+0.001, 1]$: uniform grid; number of cells 20; number of iterations 10.

The solution exists on the interval

$$x \in \left[\frac{\pi}{4}, 1\right];$$

below

$$x_{\text{lim}} = \frac{\pi}{4}$$

the problem has no solutions. The quadratic dependence of diffusion coefficient on c leads to a faster drop of $c(x)$ as $x \to x_{\text{lim}}$.

For test purposes, we replace the Cauchy problem (29) with the boundary value problem

$$\frac{\partial}{\partial x}\left(-c^2\frac{\partial c}{\partial x}+c^2\right)=0, \quad c(\pi/4+\varepsilon)=c_0, \quad c(1)=1, \tag{31}$$

which has the same solution. Here $\varepsilon = 0.001$, and numerical value of c_0 is obtained from (30). Solutions of (31), obtained with the q-scheme and upwind scheme, are compared in Fig. 3. Again the q-scheme gives much better accuracy.

4.3. Problem 3

A third test is the equation with different dependence of diffusion coefficient and convective flux on c:

$$\frac{\partial}{\partial x}\left(-c\frac{\partial c}{\partial x}+c^2\right)=0; \quad \left.\frac{\partial c}{\partial x}\right|_{x=1}=2, \quad c(1)=1, \quad x\in[0,1]. \tag{32}$$

The solution to that equation is given by

$$x = 1+\ln\sqrt{1+c^2}-\ln\sqrt{2}, \tag{33}$$

that is, the solution exists in the domain $x \in [1-\ln\sqrt{2}, 1]$. We take the computational domain $[1-\ln\sqrt{2}+\varepsilon, 1]$, with $\varepsilon = 0.001$. The fluxes N_E and N_W now are given by (18)

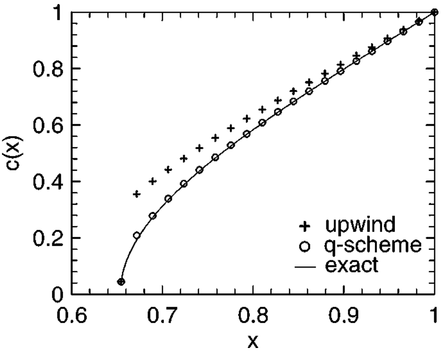

FIG. 4. Solutions of problem (32). Solid line: exact solution; crosses: upwind scheme; circles: q-scheme. The domain is $[1 - \ln\sqrt{2} + 0.001, 1]$; uniform grid; number of cells 20; number of iterations 10.

and condition $N_E - N_W = 0$ leads to the system

$$\begin{aligned} &-F_{i-1}^m c_{i-1}^{m+1} + \left(F_{i-1}^m + F_i^m\right) c_i^{m+1} - F_i^m c_{i+1}^{m+1} \\ &\quad = F_{i-1}^m \frac{p_{i-1}^m}{q_{i-1}^m} h_{i-1} - F_i^m \frac{p_i^m}{q_i^m} h_i + \frac{q_{i-1}^m \gamma_{i-1}^m - p_{i-1}^m \lambda_{i-1}^m}{q_{i-1}^m} - \frac{q_i^m \gamma_i^m - p_i^m \lambda_i^m}{q_i^m}, \\ &\qquad i = 1, \ldots, M-2, \end{aligned} \tag{34}$$

with coefficients (27). The derivatives q_i^m and p_i^m are evaluated using centered differences

$$q_i^m = \frac{\lambda_{i+1}^m - \lambda_{i-1}^m}{h_{i-1} + h_i}, \quad p_i^m = \frac{\gamma_{i+1}^m - \gamma_{i-1}^m}{h_{i-1} + h_i}. \tag{35}$$

The upwind scheme leads to the system

$$\begin{aligned} -U_{i-1}^m c_{i-1}^{m+1} + \left(U_{i-1}^m + U_i^m\right) c_i^{m+1} - U_{i+1}^m c_{i+1}^{m+1} &= \gamma_{i-1}^m h_{i-1} - \gamma_i^m h_i, \\ i = 1, \ldots, M-2, \end{aligned} \tag{36}$$

with $U_i^m = \lambda(c_i^m)$, $\gamma_i^m = \gamma(c_i^m)$. Boundary conditions (32) close the systems (34) and (36). The solutions are compared in Fig. 4.

4.4. Accuracy and Rate of Convergence

Table I shows behavior of an error of the numerical solution in all three tests for various grids. It is seen that doubling of the number of cells M until $M \leq 160$ reduces the error by a factor of almost 4. On coarse grids, the scheme, therefore, exhibits almost second-order accuracy. Changing M from 160 to 320 and from 320 to 640, one obtains only two- to three-fold decrease in error (two last rows of Table I). This illustrates the transition of the scheme to first-order accuracy under small h.

Figure 5 shows the rate of convergence of iterations. It displays nonlinear residual

$$\sqrt{\frac{1}{M} \sum_i (N_E - N_W)^2},$$

TABLE I

Error of Numerical Solution $\sqrt{\frac{1}{M}\sum_i (c_i - \bar{c}(x_i))^2}$ for Various Uniform Grids ($\bar{c}$ is Exact Solution)

	Problem 1		Problem 2		Problem 3	
Number of cells	Error	Factor	Error	Factor	Error	Factor
20	3.72×10^{-3}	—	2.65×10^{-3}	1	4.45×10^{-3}	1
40	1.06×10^{-3}	3.5	7.95×10^{-4}	3.3	1.29×10^{-3}	3.45
80	2.36×10^{-4}	4.5	1.89×10^{-4}	4.2	2.97×10^{-4}	4.3
160	6.30×10^{-5}	3.7	5.36×10^{-5}	3.5	7.52×10^{-5}	4.0
320	2.85×10^{-5}	2.2	2.26×10^{-5}	2.4	3.02×10^{-5}	2.5
640	1.07×10^{-5}	2.7	8.46×10^{-6}	2.7	1.03×10^{-5}	2.9

Note. "Factor" is error reduction factor with respect to the grid on the previous row.

Here N_E, N_W (given by (13) for problems 1 and 2, and by (18) for problem 3) are calculated with c_i, q_i, p_i upon completion of the iteration step. In all cases, six to seven iterations are necessary to reduce the error by three orders of magnitude, although problem 2 exhibits a somewhat lower rate of convergence.

Figure 6 illustrates the rate of convergence in problem 2 for the two types of grids. First is the uniform grid, with 20 cells and $h \simeq 0.011$ (same as in Fig. 6). The second is an "exponential" 20-cells grid $h_{i+1} = 2h_i$, with $h_1 = 10^{-6}$, $h_{20} \simeq 0.1$. It is seen that with the nonuniform grid, the rate of convergence is not dramatically lower. Just two iterations more are required to lower the residual by three orders of magnitude, as compared to the uniform grid.

To accelerate convergence, Newton's method can be used. However, the convection–diffusion equations are usually part of a large system of equations. Managing the Newton's method for the whole system is a cumbersome procedure, since the physical model frequently needs to be updated, and each time it is necessary to rewrite and reprogram the Jacobian matrix. The "local" Newton's method, formulated for each equation separately, usually does not give essential benefits. Picard iterations are still the "working horse" in research practice.

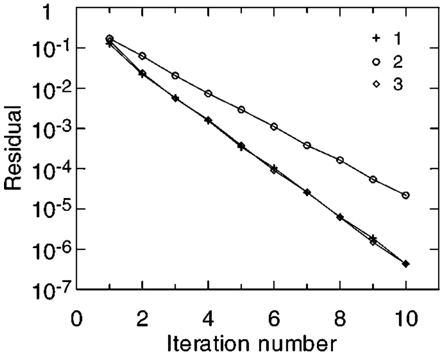

FIG. 5. Nonlinear residual $\sqrt{\frac{1}{M}\sum_i (N_E - N_W)^2}$ as a function of number of iteration. Uniform grid, number of cells 20. Crosses: problem 1; circles: problem 2; diamonds: problem 3.

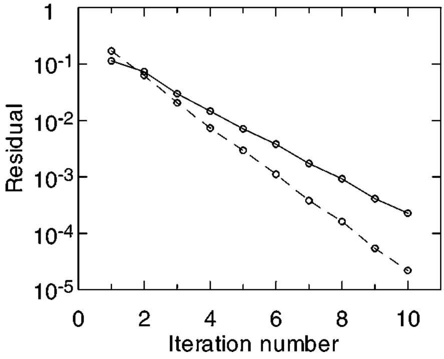

FIG. 6. Nonlinear residual as a function of iteration number for problem 2. Dashed curve: uniform grid; number of cells 20; $h \simeq 0.011$. Solid curve: "exponential" grid $h_{i+1} = 2h_i$; number of nodes 20; $h_1 = 10^{-6}$; $h_{20} \simeq 0.1$. Note that in the case of exponential grid, h_{20} is 10 times larger than h on uniform grid.

4.5. Problem 4: Nonstationary Transport Equation

To demonstrate the use of the q-scheme in nonstationary problems, we consider the simplest convection–diffusion transport equation,

$$\frac{\partial c}{\partial t} + \frac{\partial N}{\partial x} = 0, \quad N = -D\frac{\partial c}{\partial x} + vc, \tag{37}$$

with constant v and D on the interval $x \in [0, 1]$. Implicit, first order in time, and centered in space approximation of (37) on uniform grid, $\{x_i = ih, i = 0, \ldots, M-1\}$ is

$$\frac{c_i^{k+1} - c_i^k}{\tau} + \frac{N_E^{k+1} - N_W^{k+1}}{h} = 0, \tag{38}$$

where k enumerates time level.

Since $D = \text{const}$, the q-scheme for (38) reduces to (19), with

$$\lambda_i = 1, \quad D^T = b_E = 1, \quad \gamma_i = vc_i, \quad p_i = v\left.\frac{\partial c}{\partial x}\right|_i .$$

This leads to a following three-diagonal system

$$\begin{aligned} -\left(\frac{D\tau}{h^2} + \frac{v\tau}{h}\right)c_{i-1}^{k+1} + \left(1 + 2\frac{D\tau}{h^2} + \frac{v\tau}{h}\right)c_i^{k+1} - \frac{D\tau}{h^2}\,c_{i+1}^{k+1} \\ = \frac{v\tau}{2}\left(\left.\frac{\partial \tilde{c}}{\partial x}\right|_{i-1} - \left.\frac{\partial \tilde{c}}{\partial x}\right|_{i}\right) + c_i^k, \end{aligned} \tag{39}$$

where the values $\tilde{c}$ on the right-hand side are obtained by iterations.[4]

For comparison we will use the second-order QUICK scheme [17], which on a uniform grid may be written as

$$\frac{c_i^{k+1} - c_i^k}{\tau} + v\frac{c_E^{k+1} - c_W^{k+1}}{h} - D\frac{c_{i+1}^{k+1} - 2c_i + c_{i+1}^{k+1}}{h^2} = 0, \tag{40}$$

[4] Starting from $\tilde{c} = c^k$, one has to solve (39) repeatedly, taking $\tilde{c} = c^{k+1}$ until the residual becomes small enough. Note that in this process c_i^k on the right-hand side does not change.

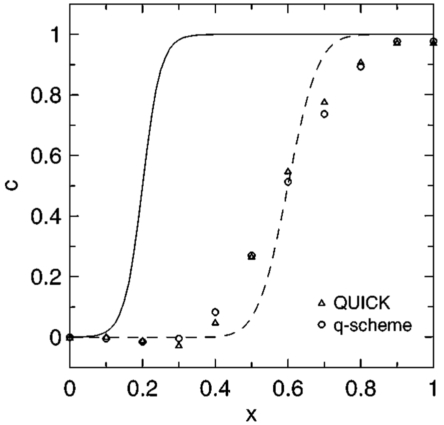

FIG. 7. Transport of steep gradient. Solid line: initial condition; dashed line: solution, obtained with the q-scheme on a very fine grid ("exact" solution); dots: results from the q-scheme (circles) and QUICK scheme (triangles). Uniform grid; number of cells 10; CFL number 0.4; 10 time steps.

where

$$c_E^{k+1} = \frac{1}{2}\left(c_{i+1}^{k+1} + c_i^{k+1}\right) - \frac{1}{8}\left(c_{i+1}^k - 2c_i^k + c_{i-1}^k\right). \tag{41}$$

The correction term on the right-hand side of (41) is calculated on the k-layer to retain the three-diagonal structure of the linear system.

Equation (37) is solved for $v = 1$ and $D = 0.001$, which forms a convection-dominated problem. A function $f(x, t = 0) = 0.5[1 + \tanh((x - 0.2)/0.05)]$ was taken as the initial condition, which models the steep gradient (Fig. 7). Boundary conditions

$$c|_{x=0} = 0, \quad \left.\frac{\partial c}{\partial x}\right|_{x=1} = 1$$

were used. The time step was taken to be $\tau = C \min\{h/v, h^2/D\}$, with the Courant number $C = 0.4$; 10 steps were performed. Note that the scheme (39) is fully implicit and in fact allows one to perform calculations with $C > 1$.

Figure 7 shows a comparison of the results. Taking into account the very coarse grid (10 cells), we conclude that both schemes satisfactorily describe the transport and both produce a nonphysical valley near $x = 0.3$ (Fig. 7). Such a "ripple" is a characteristic feature of second-order schemes. Although QUICK gives a somewhat less diffusive solution, the q-scheme produces a smaller valley and better transports median value $C = 0.5$ (Fig. 7).

5. DISCUSSION

Transition of the q-scheme from second order on the coarse grid to first order on the fine grid can be useful in problems with steep gradients. Grid refinement is usually used to capture the steep gradient. One may expect dumping of nonphysical oscillations in a region where the grid is fine and where the q-scheme reduces to a first-order scheme.

To dump nonphysical oscillations, one may construct the FCT scheme, based on a combination of first-order (upwind) and second-order (q) schemes. This procedure is described in [18]. The FCT scheme, however, utilizes a nonlinear filter, which is cumbersome and rather time-consuming. An alternative approach is offered in [19], where a linear combination of low- and high-order solutions was constructed to preserve monotonicity of transported value.

The linear problem (37) was taken to simplify comparison with the popular QUICK scheme, since in the nonlinear case, the latter would require linearization. However, the advantage of the q-scheme manifests itself in nonlinear problems: It does not require linearization and hence is easier to implement.

In all tests, the derivatives $\frac{\partial\lambda}{\partial x}$ and $\frac{\partial\gamma}{\partial x}$ were calculated using direct numerical differentiation (35). If the analytical formula for $\lambda(c)$ and/or for $\gamma(c)$ is known, a more accurate result gives the product

$$\frac{\partial\lambda}{\partial c}\frac{\partial c}{\partial x}$$

with analytical expression for $\frac{\partial\lambda}{\partial c}$ and centered formula for $\frac{\partial c}{\partial x}$.

The q-scheme can be easily extended for 2D and 3D problems. The fluxes through the surfaces of the 3D cell can be written in the form of (18), where c_i and c_{i+1} are values in the adjacent nodes along a given direction. The divergence of total flux $\nabla\cdot\mathbf{N}$ is then constructed based on the Gauss theorem, which in Cartesian coordinates has a form

$$\nabla\cdot\mathbf{N}\simeq\frac{1}{\delta P_{ijk}}\sum N_\perp\delta S_\perp,$$

where δP_{ijk} is a cell volume, $N_\perp$ is given by (18), and $\delta S_\perp$ is the respective surface of the cell. This procedure leads to a system of nonlinear equations with a 5-diagonal matrix. The coefficients of that system (the derivatives in (18)) are evaluated using values from the previous iteration. The resulting system of linear equations can then be solved by any standard procedure.

6. ACKNOWLEDGMENTS

The author is grateful to Prof. Alexei Kornyshev for discussion of polymer electrolyte membrane physics. The author is grateful to the referees for useful discussion and suggestions, which helped to clarify the properties of the q-scheme.

REFERENCES

1. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow* (Hemisphere, New York, 1980).
2. E. S. Oran and J. P. Boris, *Numerical Simulation of Reactive Flow* (Elsevier, New York, 1987).
3. K. W. Morton, *Numerical Solution of Convection–Diffusion Problems* (Chapman & Hall, London, 1996).
4. S. Gottesfeld and T. A. Zawodzinski, Polymer electrolyte fuel cells, in *Advances in Electrochemical Science and Engineering*, edited by R. C. Alkire, H. Gerischer, D. M. Kolb, and Ch. W. Tobias (Wiley–VCH, Weinheim, 1997), vol. 5, pp. 195–301.
5. K. Dannenberg, P. Ekdunge, and G. Lindbergh, Mathematical model of the PEMFC. *J. Appl. Electrochem.* **30**(12), 1377 (2000).
6. A. A. Samarskii and P. N. Vabishchevich, *Numerical Methods for Solution of Convection–Diffusion Problems* (Editorial URSS, Moscow, 1999) [in Russian].
7. D. L. Scharfetter and H. K. Gummel, Large-signal analysis of a silicon Read diode oscillator, *IEEE Trans. Electr. Dev.* **16**(1), 64 (1969).

8. D. N. de G. Allen and R. V. Southwell, Relaxation methods applied to determine the motion, in two dimensions, of a viscous fluid past a fixed cylinder, *Quart. J. Mech. Appl. Math.* **8**, 129 (1955).

9. B. P. Leonard and J. E. Drummond, Why you should not use “hybrid,” “power-law” or related exponential schemes for convective modelling—There are much better alternatives, *Int. J. Num. Meth. Fluids* **20**, 421 (1995).

10. M. Kurata, *Numerical Analysis for Semiconductor Devices* (Heath., Lexington, 1982).

11. A. Fiala, L. C. Pitchford, and J.-P. Boeuf, Two-dimensional, hybrid model of low–pressure glow discharge, *Phys. Rev. E* **49**, 5607 (1994).

12. Yu. A. Berezin and M. P. Fedoruk, *Numerical Simulation of Nonstationary Processes in Plasmas* (Nauka, Novosibirsk, 1993).

13. A. A. Kulikovsky, A more accurate Scharfetter–Gummel algorithm of electron transport for semiconductor and gas discharge simulation, *J. Comput. Phys.* **119**, 149 (1995).

14. A. A. Kulikovsky, Positive streamer in a weak field in air: a moving avalanche-to-streamer transition, *Phys. Rev. E* **57**, 7066 (1998).

15. T. E. Springer, T. A. Zawodzinski, and S. Gottesfeld, Polymer electrolyte fuel cell model, *J. Electrochem. Soc.* **138**, 2334 (1991).

16. T. F. Fuller, *Solid Polymer Electrolyte Fuel Cells*, Ph.D. thesis (University of California, 1995).

17. B. P. Leonard, A stable and accurate convective modelling procedure based on quadratic upstream interpolation, *Comput. Meth. Appl. Mech. Eng.* **19**, 59 (1979).

18. S. T. Zalezak, Fully multidimensional flux-corrected transport algorithm for fluids, *J. Comput. Phys.* **31**, 335 (1979).

19. T. W. H. Sheu, S. K. Wang, and S. F. Tsai, Development of a high-resolution scheme for a multi-dimensional advection–diffusion equation, *J. Comp. Phys.* **144**, 1 (1998).