Further Comparisons of Finite Difference Schemes for Computational Modelling of Biosensors[*]

D. Britz[1], R. Baronas[2,3], E. Gaidamauskaitė[2], F. Ivanauskas[2]

[1]Department of Chemistry, Aarhus University
Langelandsgade 140, 8000 Århus C, Denmark
britz@chem.au.dk

[2]Faculty of Mathematics and Informatics, Vilnius University
Naugarduko str. 24, LT-03225 Vilnius, Lithuania

[3]Institute of Mathematics and Informatics
Akademijos str. 4, LT-08663 Vilnius, Lithuania

Received: 2009-07-16 **Revised:** 2009-09-04 **Published online:** 2009-11-11

Abstract. Simulations are presented for a reaction-diffusion system within a thin layer containing an enzyme, fed with a substrate from the surrounding electrolyte. The chemical term is of the nonlinear Michaelis-Menten type and requires a technique such as Newton iteration for solution. It is shown that approximating the nonlinear chemical term in these systems by a linearised form reduces both the accuracy and, in the case of second-order methods such as Crank-Nicolson, reduces the global error order from $O(\delta T^2)$ to $O(\delta T)$. The first-order methods plain backwards implicit with and without linearisation, and Crank-Nicolson with linearisation are all of $O(\delta T)$ and very similar in performance, requiring, for a given accuracy target, an order of magnitude more CPU time than the efficient methods backward implicit with extrapolation and Crank-Nicolson, both with Newton iteration to handle the nonlinearity. Steady state computations agree with expectations, tending to the known solutions for limiting cases. The Crank-Nicolson method shows some concentration oscillations close to the outer layer boundary but this does not propagate to the inner boundary at the electrode. The backward implicit methods do not result in such oscillations and if concentration profiles are of interest, may be preferred.

Keywords: numerical simulation, computational electrochemistry, reaction-diffusion, amperometric sensor.

1 Introduction

This paper describes algorithms for the simulation of chronoamperometry at a thin enzyme layer on an electrode. The thin layer contains an enzyme that converts a substrate

[*]This work was partially supported by Lithuanian State Science and Studies Foundation, project No. N-08007.

S into a product P, the substrate diffusing into the layer from an electrolyte outside the layer. The product P is electroactive at the electrode and the electrode potential is set such that the resulting current is limited by diffusive transport of P to the electrode, setting the boundary condition that the concentration of P is zero at the electrode. The electrolyte outside the enzyme layer is well stirred and contains S at some bulk concentration, this being the boundary concentration at the outer layer surface. The electrolyte does not initially contain P and is assumed to have a much greater volume than the layer, ensuring that the concentration of P remains virtually at zero; this is the other boundary condition for P at the outer layer surface. The system tends towards a steady state. There are analytical solutions for the current for some parameter limits but no general solution exists, which is why a simulation is needed.

Enzyme kinetics was first described by Michaelis and Menten in 1913 [1] and their equation was confirmed by Briggs and Haldane in 1925 [2]. The enzyme electrode was suggested by Updike and Hicks in 1967 for the first time [3], followed by other early works [4–7]. These papers attempted to solve the mathematics of the relevant kinetics. The electrodes can either be run in the amperometric mode, in which the product is electrolysed at the electrode, or in the potentiometric mode, where no current is drawn at the electrode and the electrode potential is used to measure the product concentration. A useful review was written in 1990 by Schulmeister [8], who described mathematical and numerical approaches to solving the kinetics of these electrodes. Kulys et al. presented some steady state solutions for the amperometric mode of these electrodes [9], which are used for comparison in the present paper. Some numerical solutions were presented for the amperometric and potentiometric cases [10–12], and most recently in 2007 [13, 14], where the nonlinear chemical homogeneous terms were approximated by linearised forms. In this paper, we present further algorithms of greater efficiency, not avoiding the nonlinearity.

2 Theory

The chemical reactions in the layer are

$$\mathrm{S} + \mathrm{E} \rightleftharpoons \mathrm{ES} \rightarrow \mathrm{E} + \mathrm{P}, \tag{1}$$

where E refers to the enzyme, ES is a transitory complex assumed to be at a steady concentration, and P is the product [15]. This is a catalytic reaction, the enzyme itself not being used up.

Let the layer be of thickness d along the coordinate x. The partial differential equations describing the kinetics of the two substances S and P within the layer are [8,14]

$$\frac{\partial s}{\partial t} = D_s \frac{\partial^2 s}{\partial x^2} - V_m \frac{s}{s + K_M}, \tag{2a}$$

$$\frac{\partial p}{\partial t} = D_p \frac{\partial^2 p}{\partial x^2} + V_m \frac{s}{s + K_M}\ , \quad 0 < x < d,\ \ t > 0, \tag{2b}$$

in which s and p are the concentrations of S and P, respectively, t is the time variable, x the coordinate across the layer, D_s and D_p are the two diffusion coefficients, V_m is the maximal rate of the enzymatic reaction, and K_M is the Michaelis constant. The chemical reaction term is the Michaelis-Menten term, and gives rise to problems in the simulation, as it is nonlinear. The bulk concentration of S is s_0. Initially, there is neither S nor P within the enzyme layer. S diffuses into that layer and is converted to P there. Also, because the bulk volume is assumed large and well stirred, there is never appreciable P in the bulk outside the enzyme layer.

Initial and boundary conditions are

$$t = 0,\;\; 0 \le x \le d\colon \quad s = 0,\;\; p = 0, \tag{3}$$

$$t > 0,\;\; x = 0\colon \quad \frac{\partial s}{\partial x} = 0,\;\; p = 0, \tag{4a}$$

$$t > 0,\;\; x = d\colon \quad s = s_0,\;\; p = 0. \tag{4b}$$

P is thus held at zero concentrations at both ends of the layer, but will attain finite concentrations within it. (In potentiometric mode, the last condition at the electrode becomes the zero-flux condition, $x = 0\colon \partial p/\partial x = 0$, and as here, $x = d\colon p = 0$).

It is convenient to normalise the variables,

$$X = \frac{x}{d}\,, \quad T = \frac{D_s t}{d^2}\,, \quad S = \frac{s}{s_0}\,, \quad P = \frac{p}{s_0}\,, \quad r = \frac{D_p}{D_s}\,. \tag{5}$$

This leads to a dimensionless form of the kinetic equations,

$$\frac{\partial S}{\partial T} = \frac{\partial^2 S}{\partial X^2} - \mu \frac{S}{S+\kappa}, \tag{6a}$$

$$\frac{\partial P}{\partial T} = r\frac{\partial^2 P}{\partial X^2} + \mu \frac{S}{S+\kappa}, \quad 0 < X < 1,\;\; T > 0, \tag{6b}$$

where we have two new symbols

$$\mu = \frac{V_m d^2}{D_s s_0} \quad \text{and} \quad \kappa = K_M/s_0. \tag{7}$$

Following Kulys et al. [9], we also define the diffusion module

$$\sigma^2 = \frac{V_m d^2}{D_s K_M} = \mu/\kappa\,. \tag{8}$$

The new initial and boundary conditions are

$$T = 0,\;\; 0 \le X \le 1\colon \quad S = 0,\;\; P = 0, \tag{9}$$

$$T > 0,\;\; X = 0\colon \quad \frac{\partial S}{\partial X} = 0,\;\; P = 0, \tag{10a}$$

$$T > 0,\;\; X = 1\colon \quad S = 1, \quad P = 0. \tag{10b}$$

The current density g occuring at the electrode due to reduction or oxidation of P is given by

$$g = nFD_p \left.\frac{\partial p}{\partial x}\right|_{x=0}, \tag{11}$$

where n is the number of electrons transferred and F is the Faraday constant. This becomes the dimensionless current density G, simply given by

$$G = \left.\frac{\partial P}{\partial X}\right|_{X=0}. \tag{12}$$

The system tends towards a steady state at long times ($T \gg 1$). There are analytical solutions for the steady state current density [9] only for the extreme cases of $\kappa \ll 1$,

$$G_0 = \mu/2\,, \tag{13}$$

and $\kappa \gg 1$,

$$G_\infty = 1 - \frac{1}{\cosh\sigma}. \tag{14}$$

These can be used for comparison with simulated values.

In what follows, both steady state and chronoamperometric solutions will be demonstrated by digital simulation [16].

3 Discretisation and simulation methods

The domain $0 \le X \le 1$ is divided into N equal intervals H and the time dependent simulations proceed by steps of length δT. The notation used is that we have concentration samples $S_i,\ P_i,\ i = 0, \ldots, N$ at the points $X_i = iH,\ i = 0, \ldots, N$ ($X_0 = 0,\ X_N = 1$), and that the plain symbols S_i and P_i correspond to present, known values at time T and the symbols S'_i and P'_i are the next values, at $T + \delta T$, to be computed. It is the aim of this work to achieve global concentration and current errors of $O(\delta T^2, H^2)$. Second order with respect to H is easily achieved by a central difference approximation to the spatial second derivative, and will not be included in order statements hereafter, as the focus is on the order with respect to δT. Achievement of second order error is critically dependent on how the nonlinear term in (6) is discretised.

The system (6) has one convenient property. Although it is a coupled pair of equations, only the second equation, for P, depends on the other. Thus we can compute all S values first, handling the nonlinearity in some manner, and then compute all P values, where the nonlinearity is now simply a function of known old and new S values.

3.1 Steady state

In order to compute the steady state, one can use the time-marching method [14], driving the simulation to times so long that no further change is noted. This is indeed a good check on the correctness of the simulations, but an easier method to achieve steady state is to set the time derivatives on the left-hand side of (6a) and (6b) to zero, discretise the right-hand side and solve. For S, the discrete expression at the point X_i becomes

$$0 = \frac{S_{i-1} - 2S_i + S_{i+1}}{H^2} - \mu \frac{S_i}{S_i + \kappa}, \quad i = 1, \dots, N-1. \tag{15}$$

At the electrode surface, we invoke the zero flux condition (4a). In order to achieve a global error of $O(H^2)$ here, a three-point forward approximation is used (see $y'_1(3)$ in [16, p. 281]),

$$-3S_0 + 4S_1 - S_2 = 0, \tag{16}$$

and at $i = N$, condition (10b),

$$S_N = 1\ . \tag{17}$$

These equations now form a system of nonlinear equations to be solved, which we express as

$$\mathbf{F} = 0 \tag{18}$$

with

$$\begin{aligned}
F_0 &= -3S_0 + 4S_1 - S_2, \\
F_1 &= \frac{S_0 - 2S_1 + S_2}{H^2} - \mu \frac{S_1}{S_1 + \kappa}, \\
&\vdots \\
F_i &= \frac{S_{i-1} - 2S_i + S_{i+1}}{H^2} - \mu \frac{S_i}{S_i + \kappa}, \\
&\vdots \\
F_{N-1} &= \frac{S_{N-2} - 2S_{N-1} + S_N}{H^2} - \mu \frac{S_{N-1}}{S_{N-1} + \kappa}, \\
F_N &= S_N - 1.
\end{aligned} \tag{19}$$

Newton's method [17] suggests itself here, with an initial guess at the vector $\mathbf{S} = 0$, and computing at each Newton iteration a correction vector $\delta\mathbf{S}$ given as the solution of

$$\mathbf{J}\delta\mathbf{S} = -\mathbf{F}, \tag{20}$$

in which $\mathbf{J}$ is the Jacobian matrix of $\mathbf{F}$. This is an iterative process, and generally, only 3–4 iterations are required for convergence. This was set to the condition that the Euclidian norm $||\delta\mathbf{S}|| < 10^{-8}$.

3.2 Chronoamperometry

For time-marching problems, the left-hand side of (6a) and (6b) must be included in the simulation. This is always approximated here as

$$\frac{\partial Z_i}{\partial T} \approx \frac{Z'_i - Z_i}{\delta T}, \tag{21}$$

where Z stands for either S or P. This approximation takes different orders, depending on which position in the interval δT it refers to. Second order errors with respect to δT are achieved if (21) refers to the midpoint of the time interval; this is the case if Crank-Nicolson is used [18], where the right-hand sides of the partial differential equations are also discretised referred to the midpoint (see below). For the explicit simulation method [19] , the approximation is a forward difference, and for the fully implicit method (backward implicit BI or Laasonen method [20]) it becomes a backward difference, both producing global errors of $O(\delta T)$ (see also below).

We now outline several possible methods of solution. In the previous works [13, 14], the nonlinearity was avoided by an approximation. It will be seen that this leads to an order reduction to $O(\delta T)$, even for methods that otherwise achieve global errors of $O(\delta T^2)$. Nonlinearities often occur in connection with homogeneous chemical terms in electrochemical simulation problems, and in many cases [16, pp. 135–136] second-order linear approximations can be found. This is not possible here, so we must cope with the nonlinearity in some manner. We explore the use of the backward difference or Laasonen method [20] (here called BI) and of the Crank-Nicolson method [18], here called CN. The explicit method [19] is not used here, as it has stability limitations, nor is hopscotch [21, 22] used, although it is a stable method, because of the "propagational inadequacy" problem, pointed out by Feldberg [23].

In all cases below, use is made of the definition

$$\lambda = \delta T / H^2 \tag{22}$$

for convenience, and we assume $r = 1$, that is, the two species S and P share the same diffusion coefficient.

3.3 Method BI1, backward implicit with linearisation

This is the use of the BI or Laasonen method, but avoids the nonlinearity, as was done in [13]. The discretisations for S are

$$\begin{aligned}
&-3S'_0 + 4S'_1 - S'_2 = 0,\\
&\frac{1}{\lambda}\big(S'_1 - S_1\big) = S'_0 - 2S'_1 + S'_2 - \mu H^2 \frac{S'_1}{S_1 + \kappa},\\
&\ \vdots\\
&\frac{1}{\lambda}\big(S'_i - S_i\big) = S'_{i-1} - 2S'_i + S'_{i+1} - \mu H^2 \frac{S'_i}{S_i + \kappa},\\
&\ \vdots\\
&\frac{1}{\lambda}\big(S'_{N-1} - S_{N-1}\big) = S'_{N-2} - 2S'_{N-1} + S'_N - \mu H^2 \frac{S'_{N-1}}{S_{N-1} + \kappa},\\
&S'_N = 1.
\end{aligned} \tag{23}$$

Note that the nonlinearity has been removed. BI is a first-order method, so this might not matter. It will be seen in Section 5 that it works as well as BI2 as described below. The system can be rearranged in an obvious manner and solved using the Thomas algorithm [16, 24, 25]. The boundary value S_0 is computed using the "u-v" mechanism [16, pp. 87–89], which is simply a convenient way to solve the small linear system comprised of the boundary condition equation (the first in (23)) together with the next two equations, after they have been reduced to two unknowns each by the first (backward) Thomas sweep (for details, see [16, pp. 86–89]). Having computed all S values, a similar system is written for P,

$$\begin{aligned}
&P_0 = 0,\\
&\frac{1}{\lambda}\left(P_1' - P_1\right) = P_0' - 2P_1' + P_2' + \mu H^2 \frac{S_1'}{S_1 + \kappa},\\
&\quad\vdots\\
&\frac{1}{\lambda}\left(P_i' - P_i\right) = P_{i-1}' - 2P_i' + P_{i+1}' + \mu H^2 \frac{S_i'}{S_i + \kappa},\\
&\quad\vdots\\
&\frac{1}{\lambda}\left(P_{N-1}' - P_{N-1}\right) = P_{N-2}' - 2P_{N-1}' + P_N' + \mu H^2 \frac{S_{N-1}'}{S_{N-1} + \kappa},\\
&P_N' = 0,
\end{aligned} \tag{24}$$

where both the old S_i and the new S_i' values are used. Again, the Thomas algorithm is used. Lastly, G is computed using a three-point current approximation, known to have errors of $O(H^2)$ [16, 26].

3.4 Method BI2, backward implicit without linearisation

With this method, the nonlinearity is not removed. Then the ith equation in (26) becomes

$$\frac{1}{\lambda}\left(S_i' - S_i\right) = S_{i-1}' - 2S_i' + S_{i+1}' - \mu H^2 \frac{S_i'}{S_i' + \kappa} \tag{25}$$

and similarly for P, where the nonlinear term is positive. Here, the Thomas algorithm cannot be used. There are two main possibilities.

The nonlinear system can be solved using the Rosenbrock method [27], described for electrochemical digital simulations by Bieniasz [28] and described in detail in [16, pp. 167–172]. This evaluates concentrations at the next time level without iteration, and will handle nonlinearities without problems. There are several variants, of different error orders and unconditional stability.

Another way to handle the nonlinearity is, at each step, to use the Newton method as was done above for the steady state computation, computing a correction set in each iteration. This is necessary only for S, after which the known new values S' can be used for the computation of P, again using the Thomas algorithm.

For S, the system of equations like (23) but using (25), is cast into a form similar to

(19), solving for the correction vector. The system corresponding to (20) is the following:

$$\begin{bmatrix} -3 & 4 & -1 & & & \\ 1 & a_1 - \frac{\mu H^2 \kappa}{(S_1+\kappa)^2} & 1 & & & \\ & \ddots & & & & \\ & & 1 & a_1 - \frac{\mu H^2 \kappa}{(S_i+\kappa)^2} & 1 & \\ & & & \ddots & & \\ & & & & & a_1 - \frac{\mu H^2 \kappa}{(S_{N-1}+\kappa)^2} \end{bmatrix} \begin{bmatrix} \delta S_0 \\ \delta S_1 \\ \vdots \\ \delta S_i \\ \vdots \\ \delta S_{N-1} \end{bmatrix} = \begin{bmatrix} -b_0 \\ -b_1 \\ \vdots \\ -b_i \\ \vdots \\ -b_{N-1} \end{bmatrix} \tag{26}$$

where

$$\begin{aligned} &a_1 = -(2 + 1/\lambda), \\ &b_0 = -3S_0 + 4S_1 - S_2, \\ &b_i = -S_i/\lambda \quad (1 \le i \le N-2), \\ &b_{N-1} = -S_i/\lambda - S_N. \end{aligned} \tag{27}$$

The system is solved iteratively until $\delta\mathbf{S}$ converges, each time correcting the vector $\mathbf{S}$ by this amount. Since changes from one time step to the next are rather small, in most cases only 2–3 iterations are required in order to achieve convergence. Then one solves for $\mathbf{P}$ using the equations as in (24), except that the chemical term now contains only terms in (now known) S'_i.

3.5 Method BI3, backward implicit without linearisation and using extrapolation

The BI method is a first-order method with respect to time intervals. It has the advantage of a smooth error response in cases of a sharp initial step, such as we have in the present system, where concentration s_0 (or in dimensionless terms, unity S) is applied at a boundary at $T = 0$. We shall show below that Crank-Nicolson reacts with oscillations to such sharp transients; this can be damped in several ways [29] but it is often more convenient to use BI with extrapolation [30, 31], which preserves the smooth error response and increases the order. Extrapolation was introduced to electrochemical simulations by Strutwolf and Schoeller [32]. In the second-order variant, each step is first taken with the full time interval δT, and the new concentrations stored. This is denoted as the operation L_1. Then the calculation is repeated using two steps of length $\delta T/2$, denoted as L_2^2, and the resulting concentrations combined with those from the first step according to

$$\mathbf{Z}' = \left(2L_2^2 - L_1\right)\mathbf{Z} \tag{28}$$

($\mathbf{Z}$ again denotes either $\mathbf{S}$ or $\mathbf{P}$). This eliminates the first-order error term, leaving the term in $O(\delta T^2)$. Method BI2 was modified in this sense.

3.6 Method CN1, Crank-Nicolson with linearisation

The essence of CN is to match the time derivative approximation, left-hand side of (6a) and (6b), with right-hand side terms referred to the midpoint across the time step. This makes the time derivative approximation a central second-order difference. CN produces global errors of $O(\delta T^2)$. As mentioned above, CN has the problem of reacting to sharp changes in an oscillatory manner, which can be a disadvantage. The oscillations can be damped [29] with some programming effort. In the present case, as will be shown below, there are oscillations in concentrations, but they do not appear to matter, as they do not propagate to the electrode, where the current is computed. It is possible that some parameter choices could produce oscillations even on this side of the enzyme layer and if this happens, then damping techniques can be used.

In [14], CN1 was used, discretising at the ith point for S as

$$\frac{2}{\lambda}\left(S'_i - S_i\right) = S'_{i-1} - 2S'_i + S'_{i+1} + S_{i-1} - 2S_i + S_{i+1} - 2\mu H^2 \frac{S_i}{S_i + \kappa}. \tag{29}$$

The discrete expressions for the boundary conditions at $X = 0$ are, in the CN manner,

$$-3S'_0 + 4S'_1 - S'_2 - 3S_0 + 4S_1 - S_2 = 0 \tag{30}$$

and the condition at $X = 1$ is the same as that for BI1, Section 3.3. The system is easily solved by the Thomas algorithm. Then we have for P, at the ith point,

$$\frac{2}{\lambda}\left(P'_i - P_i\right) = P'_{i-1} - 2P'_i + P'_{i+1} + P_{i-1} - 2P_i + P_{i+1} - 2\mu H^2 \frac{S'_i}{S'_i + \kappa}, \tag{31}$$

now using the new values for S. Again, this is readily solved.

3.7 Method CN2, Crank-Nicolson without linearisation

Using the CN idea, a more consistent discretisation is the following. The first and last discrete equations, expressing boundary conditions, are those for CN1, but the discretisation at point i becomes

$$\begin{aligned}\frac{2}{\lambda}\left(S'_i - S_i\right) = S'_{i-1} - 2S'_i + S'_{i+1} + S_{i-1} - 2S_i + S_{i+1} \\ - \mu H^2\left(\frac{S'_i}{S'_i + \kappa} + \frac{S_i}{S_i + \kappa}\right).\end{aligned} \tag{32}$$

This is a nonlinear system, and can be solved by the Newton procedure as described above. For P, we then have, at point i,

$$\begin{aligned}\frac{2}{\lambda}\left(P'_i - P_i\right) = P'_{i-1} - 2P'_i + P'_{i+1} + P_{i-1} - 2P_i + P_{i+1} \\ + \mu H^2\left(\frac{S'_i}{S'_i + \kappa} + \frac{S_i}{S_i + \kappa}\right),\end{aligned} \tag{33}$$

where now both S and S' are known.

4 Estimating error orders

Except at extreme values of κ, there are no analytical solutions, so that if we wish to compute error orders, we cannot use known solutions for the estimation. However, there is a way, due to Østerby [33], which does not require exact values to compare with. In a given simulation, one first takes N steps of length h, with the result R_1. Then the simulation is repeated using $N/2$ steps of length $2h$, with result R_2, followed by a third using $N/4$ steps of length $4h$, with result R_4. These are combined to produce the factor q, given by

$$q = \frac{R_4 - R_2}{R_2 - R_1} \tag{34}$$

and the order is then $\ln q / \ln 2$. This was carried out in the present work.

5 Results and discussion

Except for the steady state computations, all simulations were driven at equal space intervals of 0.001, that is $N = 1000$, which was found to be adequate, and at various numbers of time intervals δT always to $T = 1$.

The computations were carried out under Linux Suse 10.2 using the Intel Fortran 90/95 compiler and IEEE 754 standard double precision, giving roughly 16 decimal digit precision.

The orders were computed by the procedure mentioned above, and are tabulated in Table 1. We note that there are only two methods, BI3 and CN2, that produce global errors of second order, the others all having errors of $O(\delta T)$, even CN1. Thus it is seen that eliminating the nonlinearity reduces the error order for CN.

Table 1. Error orders with respect to δT for the methods, $\mu = \kappa = 10$, $N = 1000$

Method	Order of S_0	Order of G
BI1	1	1
BI2	1	1
BI3	2	2
CN1	1	1
CN2	2	2

Steady state currents G were computed as described for a range of κ and three values of μ, and are shown in Fig. 1. They are all referred to both G_0 given in (13) and G_∞ given in (14). Note the convergence to unity at the respective ends of the κ scale.

Errors in the current densities were computed relative to converged current values, obtained by increasing the number of time steps in a simulation using CN2, until there were no further changes in the value to 8 decimals. The errors are expressed as relative errors e, given by

$$e = \frac{G - G_c}{G_c}, \tag{35}$$

where G_c is the converged current. This had to be computed for any given parameter set (μ, κ). It was found that $N_T = 4096$ was sufficient for convergence to 8 decimal places. The Newton iterations can easily achieve much better convergence than this, and it is indeed N_T here which limits convergence. The errors are shown in Fig. 2. The figure reflects the orders from Table 1 by the slopes and it is seen that all three first-order methods BI1, BI2 and CN1 have very similar errors, CN1 being only a slight improvement on BI1 and BI2. BI2, which accounted for the nonlinearity, might have been thought to show smaller errors than BI1, despite the same order, but did not. Clearly, CN2 and BI3 have the smallest errors and a second-order slope.

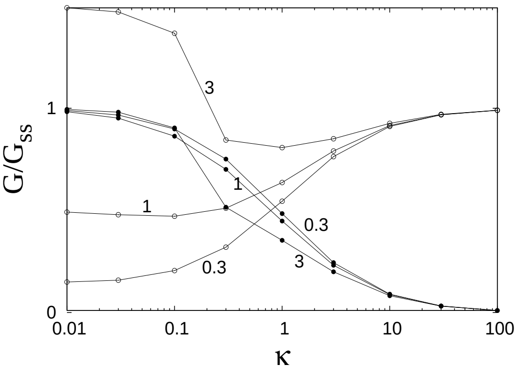

Fig. 1. Steady state responses. The numbers show values of μ. The curves converging to unity at the left-hand edge are G/G_0, while those converging to unity at the right-hand edge are G/G_∞.

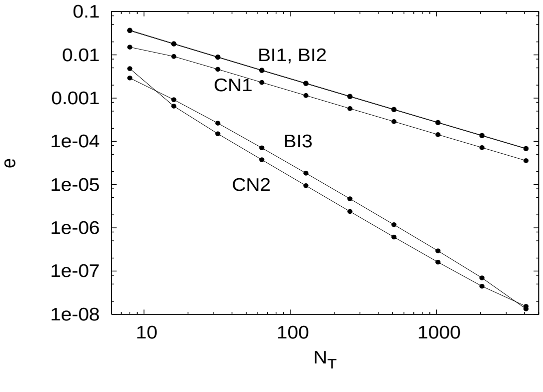

Fig. 2. Error e vs. N_T for the methods as shown, $N = 1000$, $\mu = \kappa = 10$.

It is of some interest to compare the efficiency of the methods, in terms of CPU time needed to achieve a target accuracy. We choose the target 0.01% or a relative error of 10^{-4} here, for easy comparison, based on Fig. 2. Table 2 shows the results. The two best methods, BI3 and CN2 are an order of magnitude faster than the other three. In practice, such a small error is not needed for comparison with experimental data, nor are the CPU times significant. However, if the simulation method were to be applied to longer simulations such as linear sweep voltammetry and/or if experimental parameters such as μ and κ or diffusion coefficients are to be fitted by doing many runs, then efficiency becomes important.

Table 2. N_T to $T = 1$ needed and CPU use to achieve a relative error e of 10^{-4}

Method	N_T	CPU/s
BI1	3000	0.45
BI2	3000	0.86
BI3	50	0.049
CN1	1500	0.22
CN2	40	0.025

It was mentioned above that Crank-Nicolson has an oscillatory response to sharp initial transients. This is the case here, where the initial condition $S(X = 1) = 1$ is applied, all other S values initially being zero. Normally, the oscillations that ensue result also in oscillatory currents, but here the sharp transient takes place at the opposite end of the enzyme layer from that at which the current is generated. Fig. 3 shows a time development of the S profile for 50 steps in time with $\delta T = 0.02$, and the oscillations in S are clearly seen at the outer plane of the layer, slowly damped with time.

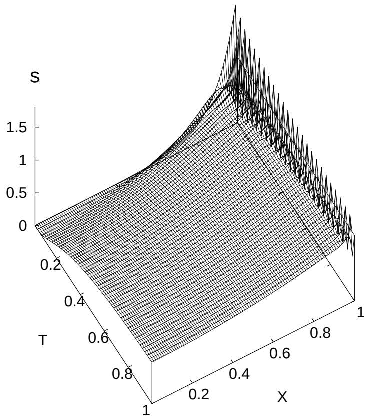

Fig. 3. Concentration profile vs. X and T for method CN2. Parameters: $\mu = \kappa = 10$, $N = 100$, $\delta T = 0.02$, $N_T = 50$.

It is also clear, however, that these oscillations do not propagate to the electrode, where the profiles are smooth. This is reflected in the current, which also has a smooth time response. However, if one is interested in the concentrations themselves, then the oscillations might not be desirable. In that case, method BI3 offers a similar efficiency and a smooth response, as seen in Fig. 4.

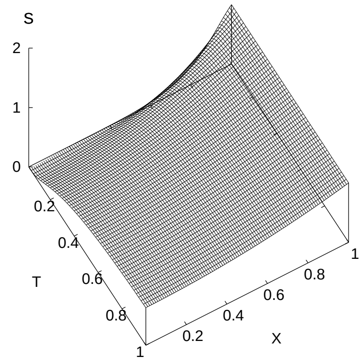

Fig. 4. Concentration profile vs. X and T for method BI3. Parameters: $\mu = \kappa = 10$, $N = 100$, $\delta T = 0.02$, $N_T = 50$.

6 Conclusions

It is seen that attempting to approximate the nonlinear chemical term by a linear form reduces both the accuracy and, in the case of second-order error methods such as Crank-Nicolson, the order from $O(\delta T^2)$ to $O(\delta T)$. The first-order methods plain backwards implicit with and without linearisation, and Crank-Nicolson with linearisation all have global errors of $O(\delta T)$ and are very similar in performance, requiring, for a given accuracy target, an order of magnitude more CPU time than the efficient methods backward implicit with extrapolation and the Crank-Nicolson, both with Newton iteration to handle the nonlinearity.

Steady state computations agree with expectations, tending to the limiting cases for small and large κ.

Crank-Nicolson shows some concentration oscillations close to the outer layer boundary, $X = 1$, but this does not propagate to the inner boundary at the electrode, so it may not matter. Backward implicit methods do not result in such oscillations and if concentration profiles are of interest, may be preferred.

References

1. L. Michaelis, M. L. Menten, Die Kinetik der Invertinwirkung, *Biochem. Z.*, **49**, pp. 333–369, 1913.

2. G.E. Briggs, J.B.S. Haldane, A note on the kinetics of enzyme action, *Biochem. J.*, **19**, pp. 338–339, 1925.

3. S.J. Updike, G.P. Hicks, The enzyme electrode, *Nature*, **214**, pp. 986–988, 1967.

4. G.G. Guilbault, Enzyme electrode probes, *Pure Appl. Chem.*, **25**, pp. 727–740, 1971.

5. W.J. Blaedel, T.R. Kissel, R.C. Boguslaski, Kinetic behavior of enzymes immobilized in artificial membranes, *Anal. Chem.*, **44**(12), pp. 2030–2037, 1972.

6. P.W. Carr, Fourier analysis of the transient response of potentiometric enzyme electrodes, *Anal. Chem.*, **49**(6), pp. 799–802, 1977.

7. H.F. Hameka, G.A. Rechnitz, Theory of the biocatalytic membrane electrode, *J. Phys. Chem.*, **87**(7), pp. 1235–1241, 1983.

8. T. Schulmeister, Mathematical modelling of the dynamic behaviour of amperometric enzyme electrodes, *Select. Electr. Rev.*, **12**(2), pp. 203–260, 1990.

9. J.J. Kulys, M.V. Pesliakiene, A.S. Samalius, 401 – the development of bienzyme glucose electrodes, *Bioelectroch. Bioener.*, **8**(1), pp. 81–88, 1981.

10. L.D. Mell, J.T. Maloy, A model for the amperometric enzyme electrode obtained through digital simulation and applied to the immobilized glucose oxidase system, *Anal. Chem.*, **47**(2), pp. 299–307, 1975.

11. P. Jochum, B.R. Kowalski, A coupled two-compartment model for immobilized enzyme electrodes, *Anal. Chim. Acta*, **144**, pp. 25–38, 1982.

12. A. Bergel, M. Comtat, Theoretical evaluation of transient responses of an amperometric enzyme electrode, *Anal. Chem.*, **56**(14), pp. 2904–2909, 1984.

13. R. Baronas, F. Ivanauskas, J. Kulys, Computational modelling of the bahaviour of potentiometric membrane biosensors, *J. Math. Chem.*, **42**(3), pp. 321–336, 2007.

14. E. Gaidamauskaitė, R. Baronas, A comparison of finite difference schemes for computational models of biosensors, *Nonlinear Anal. Model. Control*, **12**(3), pp. 359–369, 2007.

15. A.P.F. Turner, I. Karube, G.S. Wilson, *Biosensors: Fundamentals and Applications*, Oxford University Press, Oxford, 1987.

16. D. Britz, *Digital Simulation in Electrochemistry*, 3rd ed., Springer, Berlin, 2005.

17. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in Fortran*, 2nd ed., Cambridge University Press, Cambridge, 1986.

18. J. Crank, P. Nicolson, A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type, *Proc. Cambridge Phil. Soc.*, **43**, pp. 50–67, 1947.

19. S.W. Feldberg, Digital simulation: a general method for solving electrochemical diffusion-kinetic problems, in: *Electroanal. Chem.*, Vol. 3, A.J. Bard (Ed.), Marcel Dekker, New York, pp. 199–296, 1969.

20. P. Laasonen, Über eine Methode zur Lösung der Wärmeleitungsgleichung, *Acta Math.*, **81**, pp. 309–317, 1949.

21. A. R. Gourlay, Hopscotch: a fast second-order partial differential equation solver, *J. Inst. Maths. Appl.*, **6**(4), pp. 375–390, 1970.

22. D. Shoup, A. Szabo, Hopscotch: An algorithm for the numerical solution of electrochemical problems, *J. Electroanal. Chem.*, **160**, pp. 1–17, 1984.

23. S. W. Feldberg, Propagational inadequacy of the hopscotch finite difference algorithm: the enhancement of performance when used with an exponentially expanding grid for simulation of electrochemical diffusion problems, *J. Electroanal. Chem.*, **222**(1–2), pp. 101–106, 1987.

24. L. H. Thomas, *Elliptic Problems in Linear Difference Equations over a Network*, Watson Scientific Computing Laboratory, Columbia University, New York, 1949.

25. L. H. Thomas, Elliptic problems in linear equations over a network, *Special Collection Dept. of the North Carolina State Univ. Libraries*, The Llewellyn Hilleth Thomas Papers 1921–1989, item 210.8-13.8, 1949.

26. D. Britz, Investigation of the relative merits of some n-point current approximations in digital simulations. Application to an improved algorithm for quasireversible systems, *Anal. Chim. Acta*, **193**, pp. 277–285, 1987.

27. H. Rosenbrock, Some general implicit processes for the numerical solution of differential equations, *Computer J.*, **5**(4), pp. 329–330, 1962/3.

28. L. K. Bieniasz, Finite-difference electrochemical kinetic simulations using the Rosenbrock time integration scheme, *J. Electroanal. Chem.*, **469**(2), pp. 97–115, 1999.

29. D. Britz, O. Østerby, J. Strutwolf, Damping of Crank-Nicolson error oscillations, *Comput. Biol. Chem.*, **27**(3), pp. 253–263, 2003.

30. J. D. Lawson, J. L. Morris, The extrapolation of first order methods for parabolic partial differential equations I, *SIAM J. Numer. Anal.*, **15**(6), pp. 1212-1224, 1978.

31. A. R. Gourlay, J. L. Morris, The extrapolation of first order methods for parabolic partial differential equations II, *SIAM J. Numer. Anal.*, **17**(5), pp. 641–655, 1980.

32. J. Strutwolf, W. W. Schoeller, Digital simulation of potential step experiments using the extrapolation method, *Electroanal.*, **9**(8), pp. 1403–1408, 1997.

33. O. Østerby, *The error of the Crank-Nicolson method for linear parabolic equations with a derivative boundary condition*, Technical Report PB-534, DAIMI, Aarhus University, 1998.