ACCURACY CONTOURS IN (n_T, λ) SPACE IN ELECTROCHEMICAL DIGITAL SIMULATIONS*

Dieter BRITZ[a] and Merete F. NIELSEN[b]

[a] *Kemisk Institut, Aarhus Universitet, 8000 Aarhus C, Denmark*
[b] *Department of General and Organic Chemistry,*
The H.C. Ørsted Institute, Universitetsparken 5, 2100 Copenhagen, Denmark

Received April 27, 1990
Accepted June 12, 1990

Dedicated to the memory of Prof. J. Heyrovský on the occasion of his centenary.

In finite difference simulations of electrochemical transport problems, it is usually tacitly assumed that λ, the stability factor $D\delta t/\delta x^2$, should be set as high as possible. Here, accuracy contours are shown in (n_T, λ) space, where n_T is the number of finite difference steps per unit (dimensionless) time. Examples are the Cottrell experiment, simple chronopotentiometry and linear sweep voltammetry (LSV) on a reversible system. The simulation techniques examined include the standard explicit (point- and box-) methods as well as Runge–Kutta, Crank–Nicolson, hopscotch and Saul'yev. For the box method, the two-point current approximation appears to be the most appropriate. A rational algorithm for boundary concentrations with explicit LSV simulations is discussed. In general, the practice of choosing as high a λ value when using the explicit techniques, is confirmed; there are practical limits in all cases.

Some simple electrochemical experiments are simulated here: the Cottrell experiment, simple chronopotentiometry and linear sweep voltammetry (LSV) on a reversible system, all with known solutions. This is a rather restricted set and the question arises, how useful the results will be in general. One would like to include, for example, homogeneous kinetics. This would lead to many more figures and will be left to future studies.

The diffusion equation,

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} \qquad (1)$$

is suitably normalised[1] such that the new time variable T has the range $0 < T \leqq 1$ (except with LSV, see below) and new concentrations C_i (species i) are referred to the bulk concentration of the main species; the new (one-dimensional) space variable X is commonly given the range $0 \leqq X \leqq 6$ (except in LSV), corresponding to about 3

* Presented at the J. Heyrovský Centennial Congress on Polarography, Prague, August 1990.

Nernst diffusion layer thicknesses. In LSV, normalisation is such that one sweeps over a time period $T > 1$ and the Nernst layer (dimensionless), $\sqrt(\pi T)$, grows larger than in the other cases. This will be referred to again below.

There are three simulation variables: δT, the time interval (given by the number of time steps n_T); δX, the space interval; and the stability factor, also called dimensionless diffusion coefficient[2] λ, defined as

$$\lambda = \delta T/(\delta X)^2 \,. \tag{2}$$

Of these three, any two determine the third. For explicit simulation methods, stability dictates that $\lambda \leqq 0{\cdot}5$; implicit methods such as hopscotch[3], Crank–Nicolson[4] and Saul'yev[5], are stable for all λ. This does not mean, however, that all λ values are practical.

It has mostly been assumed that in a given simulation, λ should be set as high as possible, since this either maximises δT (minimises n_T) or minimises δX, giving better spatial resolution and thus smaller discretisation errors. For the explicit techniques, $\lambda = 0{\cdot}4$ or $0{\cdot}45$ is traditionally chosen, while no such tradition exists for the implicit methods. In this paper, simulation accuracy contours in (n_T, λ) (logarithmic) space are examined to attempt to arrive at some general guide lines. As well, the effect of n, the number of points involved in the current approximation[6], is considered: the dimensionless current G is given as

$$G = \left.\frac{\partial C}{\partial X}\right|_{X=0} = \frac{1}{a_n \delta X} \sum_{i=0}^{n-1} b_i C_i \tag{3}$$

with the coefficients a and b tabulated[6,1].

Finally, for LSV (and in fact any system where time-varying boundary concentrations are found) it is important to use a rational sequence of steps at each iteration, to be discussed below. The importance of correct ordering of steps was discussed previously[7,1], in the context of homogeneous reactions.

THEORETICAL

In the Cottrell experiment, the boundary concentration of the electroactive species, C_0, is held at zero. At $T = 1$, then, expression (3) can be compared with the known exact solution

$$G_{\text{anal}} = 1/\sqrt{\pi} \quad (T = 1)\,. \tag{4}$$

The error displayed in the contour plots is the relative error e,

$$e = \frac{G - G_{\mathrm{anal}}}{G_{\mathrm{anal}}} . \qquad (5)$$

For simple chronopotentiometry, if we normalise time by the transition time, the boundary concentration C_0 of the electroactive species should reach zero at $T = 1$ and the constant current G is $\frac{1}{2}\sqrt{\pi}$. The error in this case is that in transition time found, T_0, giving

$$e = T_0 - 1 . \qquad (6)$$

For simple LSV, we have the reversible redox couple

$$\mathrm{A} + ze \Leftrightarrow \mathrm{B} \qquad (7)$$

(z being the number of electrons transferred) and the time interval by which we normalise time is the (sweep) time during which one potential unit p,

$$p = zFE/RT \qquad (8)$$

is swept. Then, the peak current $G = 0{\cdot}44629$ occurs[8] at $p = -1{\cdot}1090$ or $-28{\cdot}49$ mV (if $z = 1$). Since peak potential here is the more sensitive quantity (and in many simulations, a more important result than the peak current), the error is defined as

$$e = p_{\max} - (-1{\cdot}1090) \qquad (9)$$

but expressed, for convenience, in mV in the plots. Initially, all $C_{\mathrm{A},i} = 1$, $C_{\mathrm{B},i} = 0$ and at all p, the Nernst equation,

$$C_{\mathrm{A},0}/C_{\mathrm{B},0} = \mathrm{e}^p \qquad (10)$$

holds, as well as equality of the flux magnitudes or

$$G_{\mathrm{A}} = -G_{\mathrm{B}} \qquad (11)$$

if the diffusion coefficients of A and B are assumed equal. The sweep starts at p_0 and $\delta p < 0$, equal to $-1/n_T$ or $-\delta T$. In these investigations, $p_0 = +8$ was used for accuracy (about 200 mV for a one-electron reaction), so that when the peak is reached, about 9 p- or T- units have passed. The required X-range is some multiple of the Nernst layer thickness $\sqrt{(\pi T)}$; as mentioned above, at $T = 1$ (Cottrell and chronopotentiometry simulations), $0 \leqq X \leqq 6$ is sufficient, being about $3\sqrt{(\pi T)}$; since T in LSV is larger by a factor of 9 or so, an X-range larger by a factor of about 3 should be used in these LSV simulations. This can of course be automated and one could use an expanding n_X as T increases but this was not done here.

Algorithms

For the Cottrell simulations, the current G is expressed as in Eq. (*3*). There is a tactical problem with this method: what should C_0 be initially? Mathematically, the boundary condition is that $C_0 = 0$ for $T > 0$ and one has the choice of initialising C_0 as zero or unity (equal to all other C_i). In the latter case (as used, for example, by Bard and Faulkner[9]), there is no change in any concentrations during the first simulation step, so one effectively wastes one iteration. Setting C_0 initially to zero imparts an initial finite flux; from the exact solution of the Cottrell experiment, one can calculate what time.this flux corresponds to (I. Ruzić, Ruder Bosković Institute, Zagreb, private comunication 1985). One finds that this comes to about $\frac{1}{4}\delta T$, depending on the choice of λ and which simulation method one is using. In both cases, it can be argued that one in fact starts the simulation at a time $T \neq 0$. Some workers such as Bard and Faulkner[9] therefore compensate for this effect by correcting all times by a small amount. In the present work, we initialise C_0 as zero and assign to this state $T = 0$; at the end of k iterations, T is assigned the value $k\delta T$, without any corrections (*method 1*). There is another way of handling this (M. F. Nielsen, unpublished results): if one computes the flux using the next C_0 value and present C_i $(i > 0)$ values (*method 2*), then the fact that – implicitly – a unit $C_0(T = 0)$ value is used does not waste an iteration in the box method, since the change in C_1 is computed from the flux and the initial C_0 is never used. This gives – for a Cottrell simulation – a small difference in the results compared to method 1, initially setting C_0 to zero, letting this drive the first diffusion iteration and computing the flux at the end of that iteration. However, initially setting $C_0 = 1$, $T = 0$ in method 1 gives the same results, for a Cottrell simulation, as method 2. In the case of LSV, the two different methods give practically the same results.

For chronopotentiometry, one needs to devise a rational strategy. At any given time T, we have a set of known "present" concentrations C_i $(i = 1 \ldots n_X$, n_X being the number of points in space). Applying the condition of constant $G(\frac{1}{2}\sqrt{\pi})$ and inverting Eq. (*3*) then yields C_0 corresponding to the present concentration set. This value C_0 then drives the diffusion calculation for the explicit simulation, so that for the new value C_1' at $T' = T + \delta T$ we have (point method[10])

$$C_1' = C_1 + \lambda(C_0 - 2C_1 + C_2) \qquad (12)$$

and so on for other i. The same thing holds for the Runge–Kutta method RKI (ref.[11]) for the first half-step, after which a new intermediate C_0 value must be used for consistency[11,1].

For the transition time, the simulation stops when a C_0 value reaches or goes below zero. Linear interpolation between this and the previous C_0 value then determines the exact transition time.

For the Crank–Nicolson method, there is a problem with derivative boundary conditions[12,13], which is overcome by including future C'_0 values in the implicit discretisation[14]. Thus, for the first equation of the system of implicit discrete equations, we have

$$C'_1 = C_1 + \tfrac{1}{2}\lambda(C_0 - 2C_1 + C_2 + C'_0 - 2C'_1 + C'_2)\,, \tag{13}$$

where C'_0 is as yet unknown and can be solved for, if we have another equation involving it. This is furnished by the known current, that is, Eq. (3) at T'

$$G_A = \frac{1}{a_n \delta X} \sum_{i=0}^{n-1} b_i C'_i\,. \tag{14}$$

This technique, here called ICN, has been described for several systems[14].

The stable Saul'yev method[15] comes in several variants. Lapidus and Pinder give a good description[16] of the basic asymmetric LR and RL algorithms which, although λ-stable, are not very accurate. Saul'yev himself[15] and Larkin[17] suggest balancing the asymmetries with either alternating between LR and RL on successive iterations, or taking, at each iteration, the average of an LR and an RL sweep. This latter technique, which we call ⟨LR, RL⟩, is used here. It shares with Crank–Nicolson the derivative boundary problem and it can be solved in a similar manner, giving what we might denote as the I⟨LR, RL⟩ method, to be described elsewhere[18].

The rational algorithm for LSV goes as follows. At time T, potential p, we have two sets of known discrete concentrations $C_{A,i}$ and $C_{B,i}$ ($i = 1 \ldots n_X$). The boundary concentrations $C_{A,0}$ and $C_{B,0}$ are given, for this p value, by a combination of the Nernst equation (*10*) and the equal and opposite fluxes condition (*11*). Substituting Eq. (*3*) for G_A and G_B in Eq. (*11*) and solving, we obtain

$$C_{A,0} = \frac{-\sum_{i=0}^{n-1} b_i(C_{A,i} + C_{B,i})}{b_0(1 + e^{-p})}\,, \tag{15}$$

$$C_{B,0} = C_{A,0}e^{-p}\,. \tag{16}$$

The $C_{A,0}$ value then also yields the present G_A (flux) value, and both C_0 values drive the diffusion step to $p' = p + \delta p$. Feldberg[2] has used a similar expression, although he mixes present and future C-values in his flux expressions, using the two-point discretisation natural to the box method,

$$G_A = (C_{A,1} - C'_{A,0})/\tfrac{1}{2}\delta X\,, \tag{17}$$

but another sequence of steps in each calculation, resulting in (almost) identical current at each time step. The mixing turns out to have little if any effect. Britz[1] has previously described an inconsistent algorithm (for both point- and box methods), in which $C_{A,0}$ and $C_{A,0}$ appropriate to p', the new p, are first computed and then used to drive the diffusion. In addition, the current G_A is then computed using all the new C values. This procedure is neither rational nor consistent, and causes grave convergence problems, especially with respect to peak potential. Britz consequently found that up to 10^4 steps per potential unit were required for reasonable accuracy in peak potential, compared with only about 100 for Crank–Nicolson or, for that matter (see Results section) the plain explicit methods using the correct algorithm, if an accuracy of about 0·1 mV is demanded. This inconsistency was pointed out to one of the authors (D.B.) by M. F. Nielsen and O. Hammerich (Copenhagen, private communication 1988). Thus, the previously called box method[1] was not in fact that method used by the Feldberg school, and previous efficiency calculations[1,11] are incorrect on this point.

Similarly as in chronopotentiometry, implicit Crank–Nicolson or Saul'yev expressions, involving the unknown future $C'_{A,0}$ and $C'_{B,0}$ are used for LSV; there are no special complications.

The peak potential is calculated by fitting an exact parabola to the three current points standing on the peak, and calculating the p value corresponding to the parabola's maximum. In this way, peak potential resolution is better than the interval δp which can be as large as 0·01 or about 0·3 mV.

Numerical Experiments

As well as the error, cpu time contours are plotted. In general, one decides on an acceptable error (for example, 10^{-4} relative, or 0·1 mV in LSV were chosen here), and the most efficient simulation technique is that which yields this error or less in the shortest possible cpu time.

The error limit of 10^{-4} relative, or 0·01%, might be considered too severe, since measurement accuracy rarely goes below 1%. Previous studies, however, have tended to show that when comparing simulation methods, the error limit is not critical. The choice is somewhat arbitrary and our present choice was a matter of convenience, leading to a convenient range of cpu times for comparison purposes.

Computer programs were written in FORTRAN, using double precision (only in order to obtain reliable error estimates) and were run on a Digital Equipment VAX 6210 computer under the VMS operating system.

RESULTS AND DISCUSSION

In the following, the aim was a relative error $|e| \leqq 10^{-4}$ for the Cottrell and chronopotentiometry simulations, and $|e| \leqq 0{\cdot}1$ mV in peak potential for LSV. The areas corresponding to these errors have been shaded in all the figures. The names EX, BOX and RK2 have been assigned to, respectively, the explicit point-, box- and second-order Runge–Kutta methods; the names ICN, ⟨LR, RL⟩ and I⟨LR, RL⟩ have been defined above.

Cottrell Simulations

Figure 1 shows these, using the point method (EX) for the 2-, 3, and 6-point current approximations. For $n = 6$, a zero-error contour is seen, that is, the error crosses żero, at about $\lambda = 0{\cdot}06$. One might expect such a contour at $\lambda = 0{\cdot}16$, where the second-order discretisation error fortuitously goes close to zero[16]. Possibly the residual errors even with the 6-point current approximation, or the particular boundary condition acts to shift the zero crossing region.

Given the contours in Fig. 1, what should one choose as operating point, having decided on the acceptable relative error? Clearly, a high-n current approximation is desirable, so we look at Fig. 1c. One might choose a point on the zero-error line, such as point X at ($\log n_Y$, $\log \lambda$) about ($2{\cdot}4$, $-1{\cdot}7$); however, the shaded area is very narrow there and there are uncertainties in the parameters. If one wants a reasonably wide region of safety, the circled region A might be better or (in log units), roughly ($2{\cdot}8$, $-1{\cdot}25$). One cannot rely, however, on this to be valid if conditions change slightly: for example if a chemical reaction takes place in solution. One might conclude that region B is safest, where the cpu time is now about two orders of magnitude greater than for the other two points, though still at a reasonable 7 s or so. This is at a λ-value of about 0·4, close to the usual. On the above grounds, the 3- or even 2-point current approximation are not much less efficient. Note also that, as expected, the error contours become erratic at λ exceeding 0·45, so this is a practical maximum.

Figure 2 shows the corresponding plots for the box method, for $n = 2$, 3 and 5 (for $n = 5$, the b-coefficients are quite large[6] so this is the limit). There is a zero error line for all three and any n appears to be usable with this method. The uncertainty in λ for this zero-error line confirms our choice of region A, with a cpu time of about 7 s. The box method, as traditionally used[2], has been said[1] to use an incorrect discretisation for the change in C_1: the Feldberg formula is

$$C'_1 = C_1 + \lambda(2C_0 - 3C_1 + C_2)\,, \tag{18}$$

based on fluxes in and out of boxes, whereas Taylor analysis of the same geometry seen as points, gives[1]

$$C'_1 = C_1 + \tfrac{4}{3}\lambda(2C_0 - 3C_1 + C_2)\,. \tag{19}$$

The "correct" factor 4/3 has been used in Fig. 3. Region A (for $n = 2$) is about the best but now we use >10 s cpu. This supports the intuitive fluxes-between-boxes Feldberg approach with the box method.

Figure 4, for the RK2 (that is, second-order RKI) method[1,11], leaves little doubt about the choice of region A, with a cpu time of around 1 s. The λ limit is clearly a little over 0·4; RKI is not λ-stable for λ > 0·5.

Figure 5, for the Crank–Nicolson method, extends to high λ, up to 100. Evidently there is a practical λ limit, depending on n_T but it is also clear that high λ is not desirable. If we look for a minimum-cpu point on the 10^{-4} error contour, we arrive

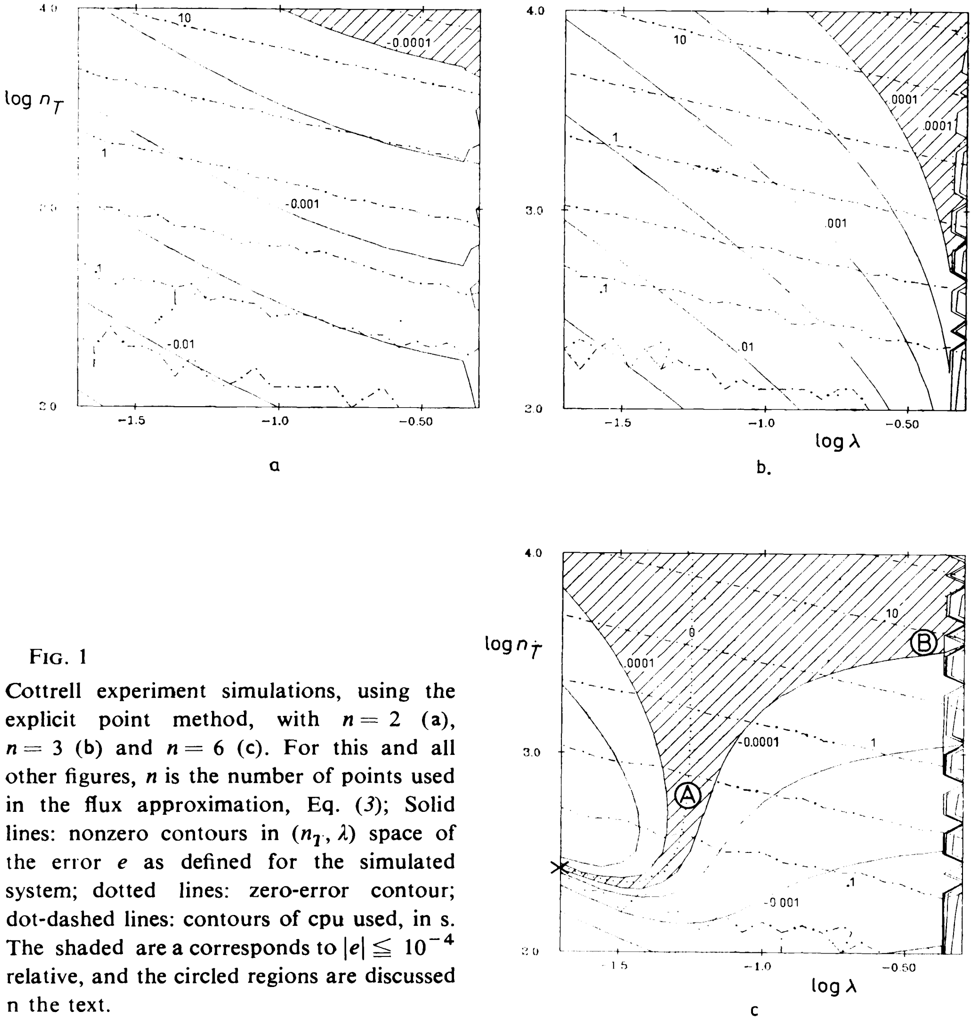


FIG. 1

Cottrell experiment simulations, using the explicit point method, with $n = 2$ (a), $n = 3$ (b) and $n = 6$ (c). For this and all other figures, n is the number of points used in the flux approximation, Eq. (3); Solid lines: nonzero contours in (n_T, λ) space of the error e as defined for the simulated system; dotted lines: zero-error contour; dot-dashed lines: contours of cpu used, in s. The shaded are a corresponds to $|e| \leq 10^{-4}$ relative, and the circled regions are discussed n the text.

at about point A, not far from $(n_T, \lambda) = (100, 3)$ (as found previously[1]), where the required cpu time is only about 0·3 s or so, making CN highly efficient. For this method, not λ but δX forms a limit; the edge of the dense contour region on the right of Fig. 5 leans at the constant-δX slope.

The hopscotch methcd has been promoted[3] for electrochemical simulations. It is stable for all λ. Fig. 6, however, makes the point that this does not imply accuracy at all λ. One could choose operating region A, with the risk that the zero-error contour

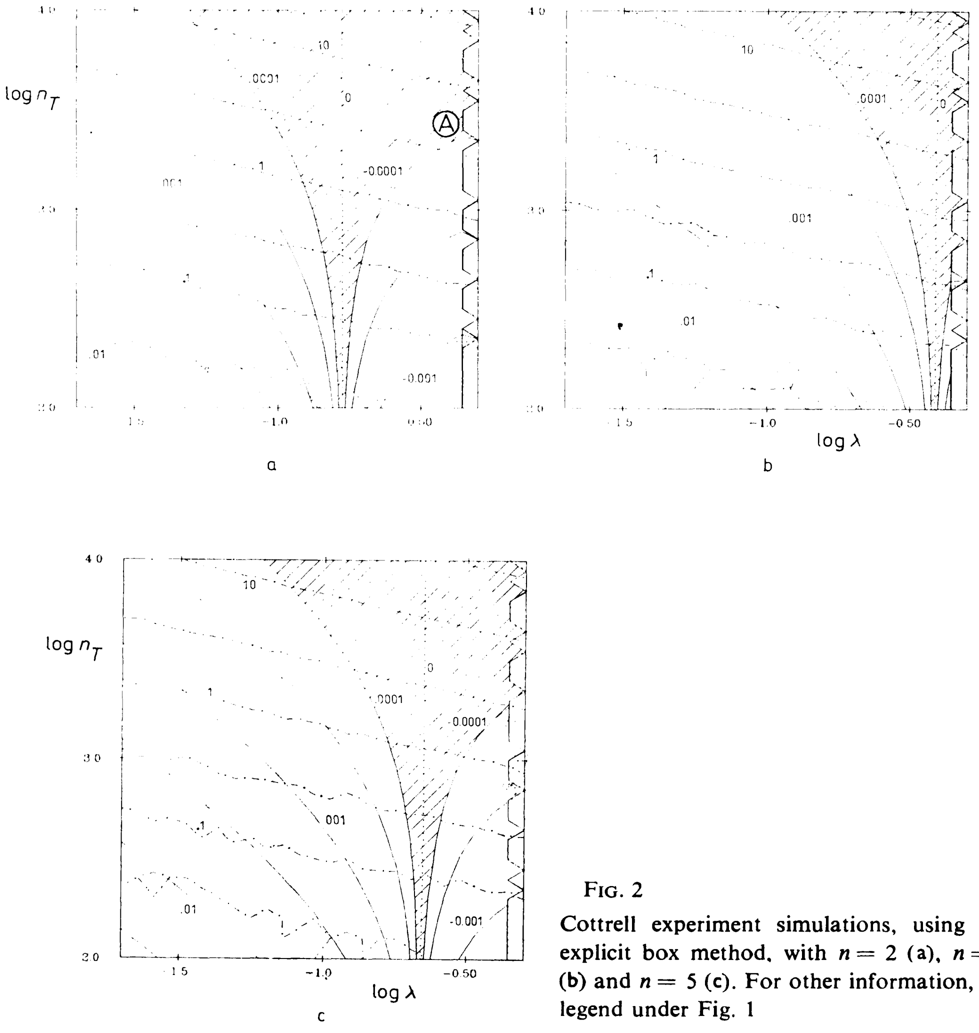


FIG. 2

Cottrell experiment simulations, using the explicit box method, with $n = 2$ (a), $n = 3$ (b) and $n = 5$ (c). For other information, see legend under Fig. 1

might shift under changed conditions; or region B, with a cpu of about 10 s, gaining nothing over the usual explicit methods and in fact using a similar λ.

Figure 7 shows the results for the best of the Saul'yev variants, $\langle$LR, RL$\rangle$ (refs[17,18]). As with hopscotch, the unrestricted λ is of little use although at "normal" λ values, we here lie close to the zero-error line (region A, cpu time 0·3 s). As with hopscotch, it is probably safer to move into a broader part of the acceptable error region, point B, close to $\lambda = 1$ and cpu time about 3 s.

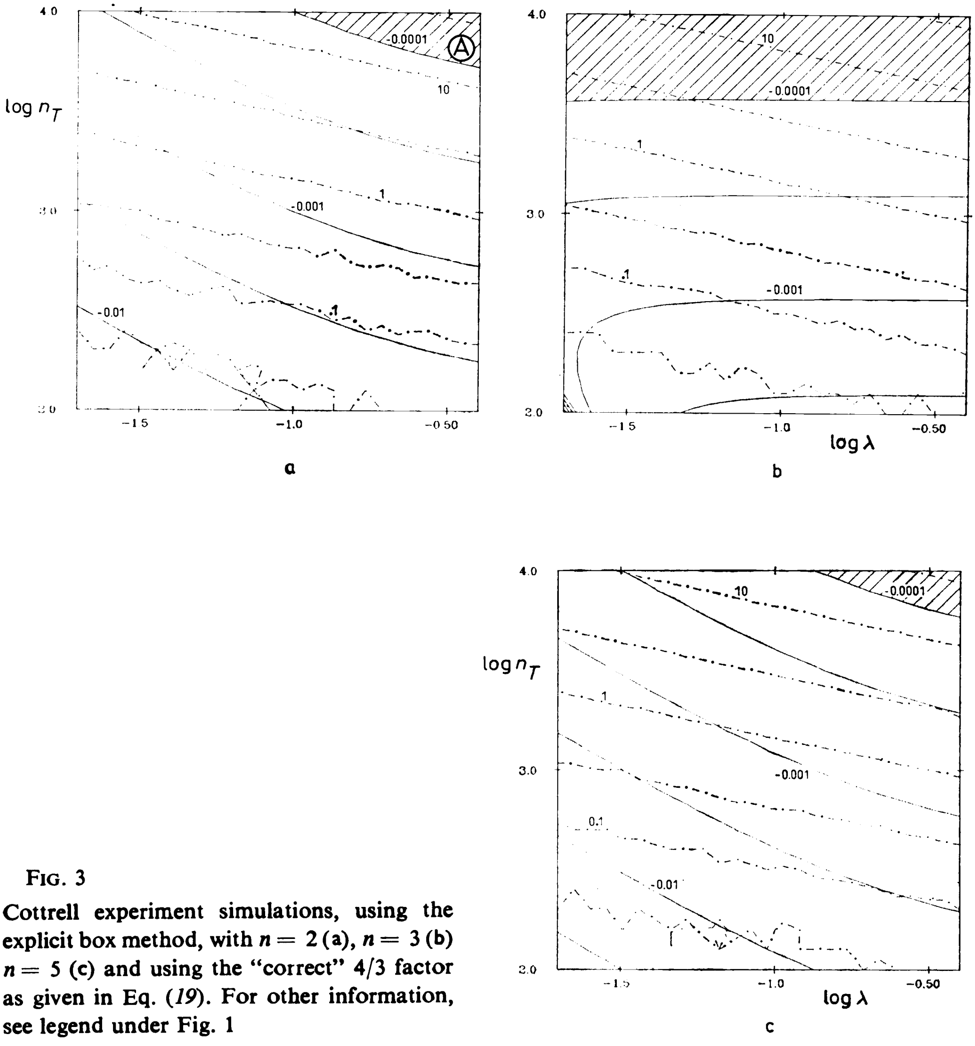


FIG. 3

Cottrell experiment simulations, using the explicit box method, with $n = 2$ (a), $n = 3$ (b) $n = 5$ (c) and using the "correct" 4/3 factor as given in Eq. (*19*). For other information, see legend under Fig. 1

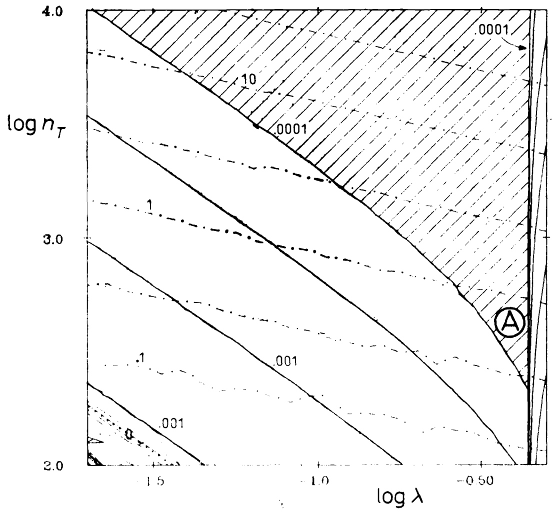


FIG. 4

Cottrell experiment simulations, using the second-order Runge–Kutta method, with $n = 6$. For other information, see legend under Fig. 1

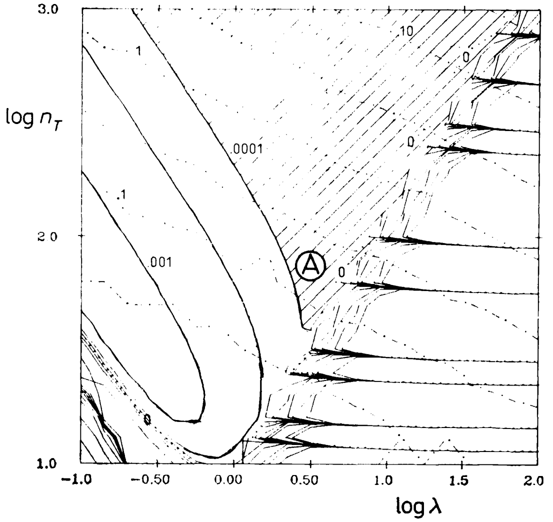


FIG. 5

Cottrell experiment simulations, using the Crank–Nicolson method, with $n = 6$. For other information, see legend under Fig. 1

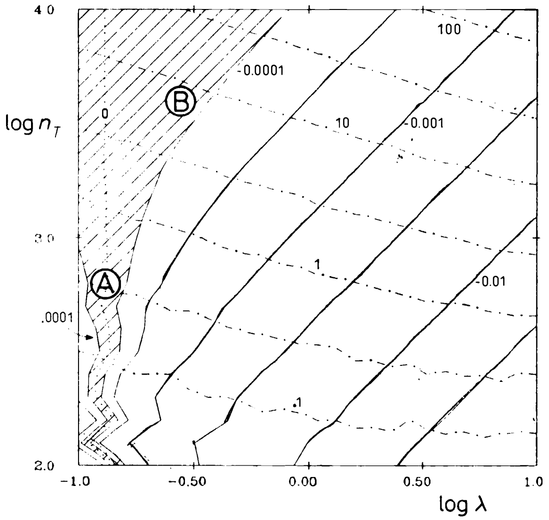


FIG. 6

Cottrell experiment simulations, using the hopscotch method, with $n = 6$. For other information, see legend under Fig. 1

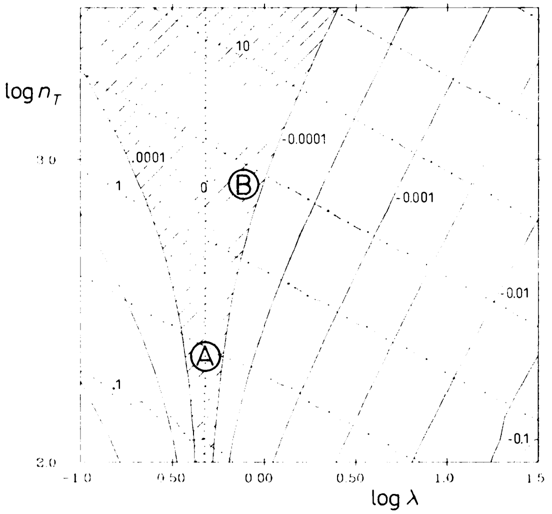


FIG. 7

Cottrell experiment simulations, using the Saul'yev variant ⟨LR, RL⟩ (see text), with $n = 6$. For other information, see legend under Fig. 1

In summary (Table I), the box- and point-methods are about equally efficient (but a low-n current approximation must be used with the box method), RK2 and Saul'yev ⟨LR, RL⟩ are mildly better by about a factor of 3–7, and Crank–Nicolson is very efficient. Hopscotch shows up poorly here but it must be pointed out that it is intended[3] for two-dimensional simulations, where it may perform better.

Chronopotentiometry

Figure 8 shows error contours in the calculated transition time, for the explicit point method EX, for two n values (for $n = 2$, results are very poor and not included). Our choice of operating point is quite clear here: $n = 5$ (Fig. 8b) at region A, with cpu time of about 10 s. The box method plots are shown in Fig. 9 and here, as for the Cottrell simulations, the "poor" current approximation, $n = 2$, is once again the best – no others are shown. The chosen operating region A is here a little less efficient than for the point method but comparable at a cpu use of about 20 s.

RK2 (Fig. 10) does better; in region A only 1 s cpu is needed. The Crank–Nicolson method ICN (that is, with implicit C_0'), Fig. 11, again shows erratic behaviour for very high λ but region A seems a reasonable choice and gives 0·3 s cpu. Again, the δX

TABLE I

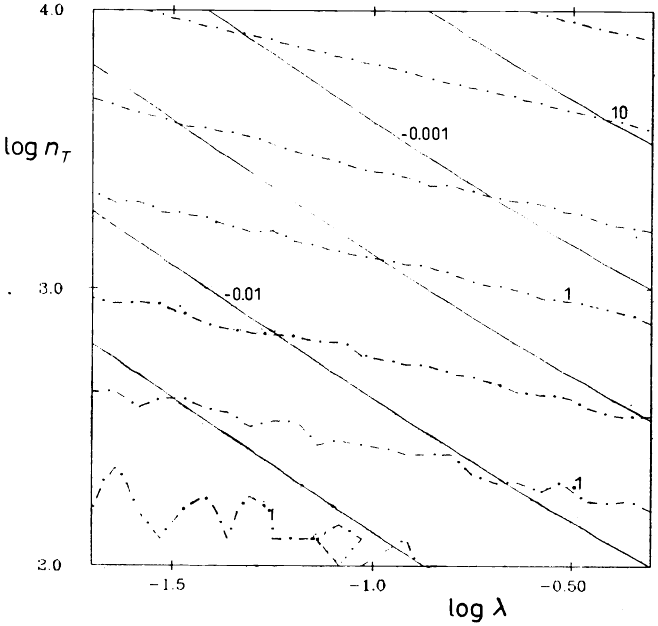
Relative efficiencies of some simulation methods, applied to the Cottrell experiment; operating regions as shown in Figs 1–7

Method	Fig. no.	Region	n_T	λ	cpu, s	Eff.
Explicit point, $n = 6$	1c	B	4 000	0·4	7	1
Box, $n = 2$	2a	A	4 000	0·4	7	1
Box with 4/3 factor, $n = 2$	3a	A	6 000	0·4	30	0·3
RK2, $n = 6$	4	A	1 200	0·4	1	7
CN, $n = 6$	5	A	50	3	0·3	23
Hopscotch, $n = 6$	6	B	5 000	0·3	10	0·7
Saul'yev ⟨LR, RL⟩, $n = 6$	7	B	1 000	1	3	2·3

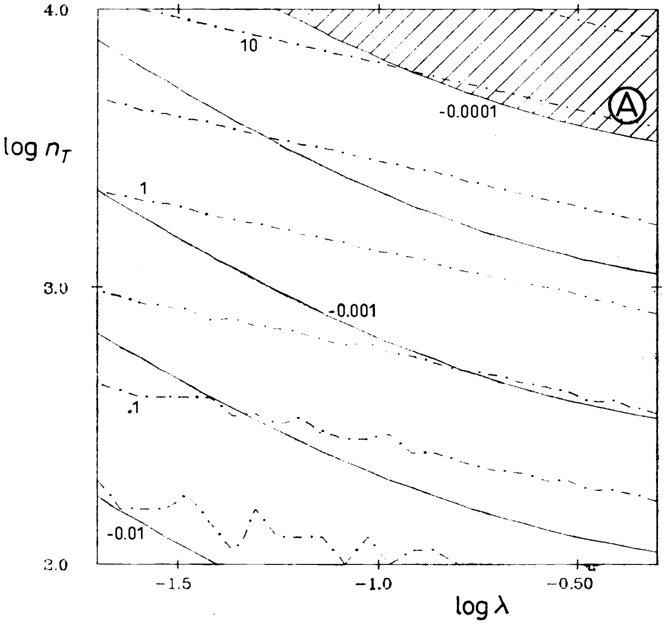
⟶

FIG. 8

Chronopotentiometry simulations, using the explicit point method, with $n = 2$ (a) and $n = 5$ (b). For other information, see legend under Fig. 1


4.0
3.0
2.0
log n_T
−1.5
−1.0
−0.50
log λ
10
-0.001
1
-0.01
.1
1

a


4.0
3.0
2.0
log n_T
−1.5
−1.0
−0.50
log λ
10
-0.0001
A
1
-0.001
.1
-0.01

b

limit, rather than a λ-limit, is seen. The Saul'yev variant I⟨LR, RL⟩, also with implicit C_0' (Fig. 12) shows a zero-error contour and we choose region A to be safe, where cpu use is 3 s – although one might opt for region B or even C, at unusually low λ values. Table II summarises these results and shows again the superiority of Crank–Nicolson, properly applied. The Saul'yev variant is about as good as RK2, which however is easier to program.

Linear Sweep Voltammetry

Figure 13 shows mV-error contours for an LSV simulation by the explicit point method ($n = 5$). Choosing a ±0·1 mV peak potential error, and avoiding the zero-error line leads to the choice of region A at $\lambda = 0{\cdot}4$ and a cpu use of about 3 s. This operating region stands in sharp contrast with previous claims[1], where the incorrect algorithm was used; Fig. 14 shows this and we note that the ±0·1 mV error contour does not appear in the plot, so the cpu use would be >100 s.

The box method (for $n = 2$ only) is shown in Fig. 15. As for the other experimental techniques, $n = 2$ is optimum and region A gives a cpu use of less than 1 s, in contrast with previous findings[1,11]. This is the same operating point as chosen for the point method; the cpu difference must be due to the difference in current approximations. Normally, the choice of n has very little effect on cpu but at small n_T as used here (100) and thus small n_X, the high-n calculation begins to take an appreciable cpu time.

Figures 16 and 17 show the results for RK2 and ICN, respectively; RK2 shows a very narrow ±0·1 mV region over the $\lambda = 0{\cdot}4$ line, and >100 s cpu use if we wish to stay away from this narrow area (region A). If we were to choose this narrow region, however, cpu times as low as 0·2 s are possible (region B); such an operating point was fortuitously chosen in a previous work[11]. For ICN, region A at $\lambda \simeq 1$ appears a safe choice and gives a cpu use of 1 s. It would be possible to go to $n_T < 10$ but, re-

TABLE II

Relative efficiencies of some simulation methods, applied to chronopotentiometry; operating regions as shown in Figs 8—12 ($n = 5$ or 6 assumed unless otherwise stated)

Method	Fig. no.	Region	n_T	λ	cpu, s	Eff.
Point	8b	A	5 000	0·4	10	1
Box, $n = 2$	9a	A	6 000	0·4	20	0·5
RK2	10	A	1 600	0·4	3	3
ICN	11	A	100	10	0·3	30
Saul'yev	12	A	1 000	1	3	3
I⟨LR, RL⟩		B	2 000	0·1	3	3
		C	1 000	0·1	1	10

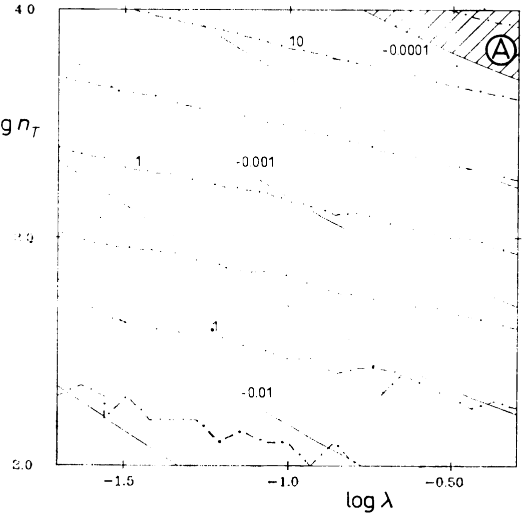


FIG. 9

Chronopotentiometry simulations, using the explicit box method, with $n = 2$. For other information, see legend under Fig. 1

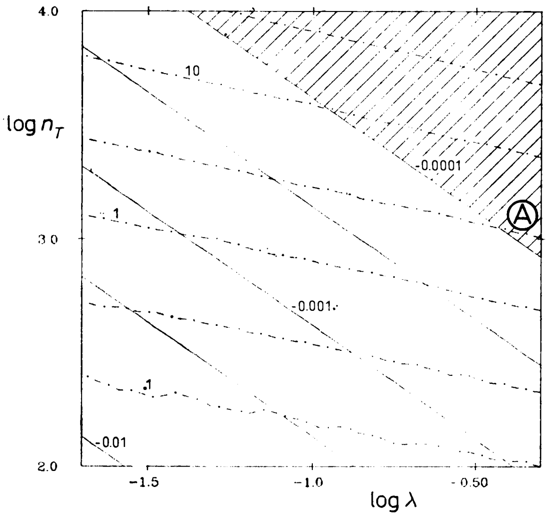


FIG. 10

Chronopotentiometry simulations, using the second-order Runge–Kutta method, with $n = 6$. For other information, see legend under Fig. 1

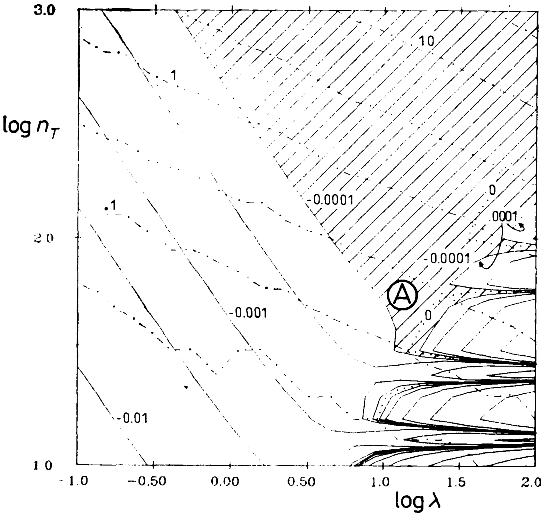


Fig. 11

Chronopotentiometry simulations, using the Crank–Nicolson method with implicit C_0' (ICN), with $n = 6$. For other information, see legend under Fig. 1

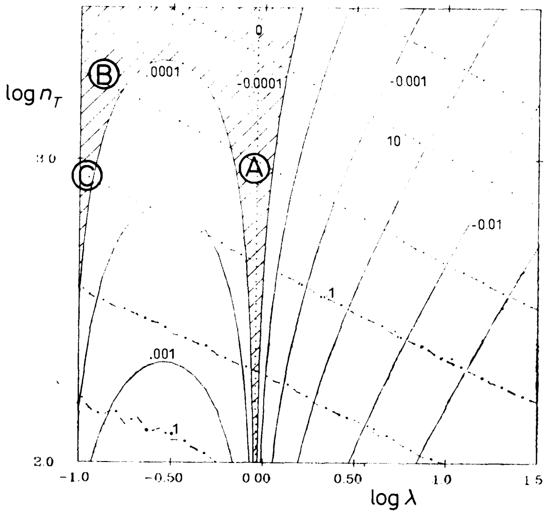


Fig. 12

Chronopotentiometry simulations, using the Saul'yev variant with implicit C_0' (I⟨LR, RL⟩), with $n = 6$. For other information, see legend under Fig. 1

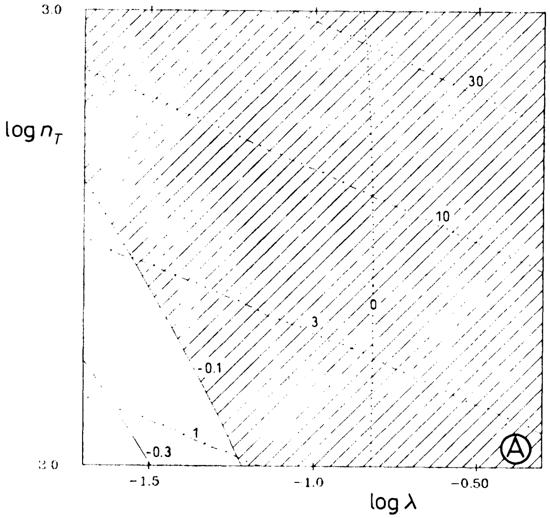


FIG. 13

LSV simulations, using the explicit point method with $n = 5$. For other information, see legend under Fig. 1

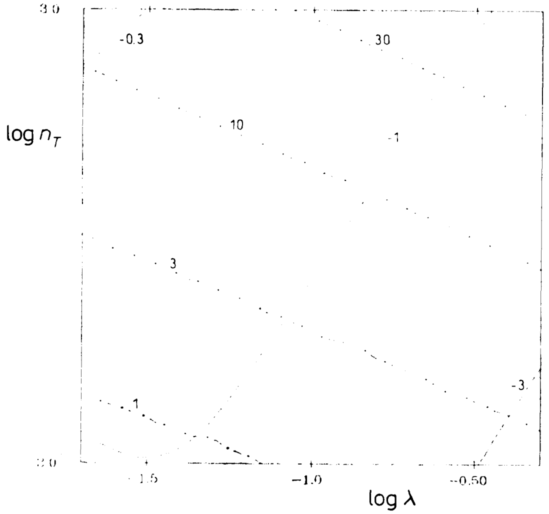


FIG. 14

LSV simulations, using the explicit point method with $n = 5$, but with the previously given[1] inconsistent algorithm. For other information, see legend under Fig. 1.

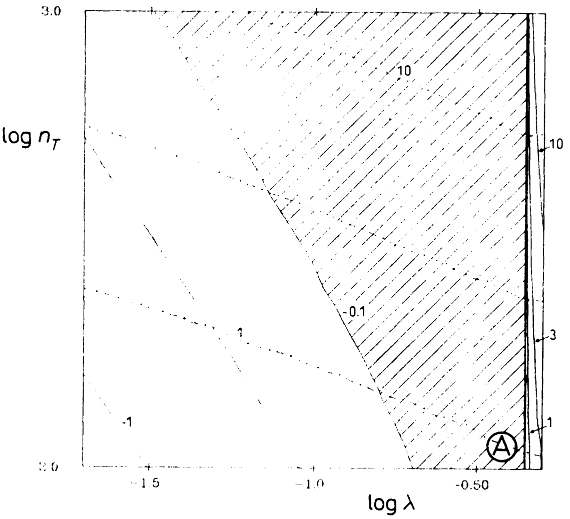


Fig. 15

LSV simulation, using the explicit box method with $n = 2$. For other information, see legend under Fig. 1

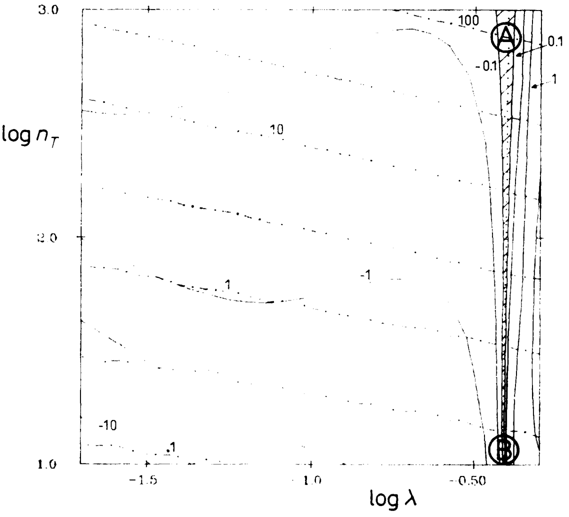


Fig. 16

LSV simulation, using the second-order Runge–Kutta method with $n = 5$. For other information, see legend under Fig. 1

garding the $-0{\cdot}03$ mV contour, this would likely lead to much the same cpu use. In any case, it might be difficult to convince a practising simulator to use such large δT (or δp) values.

Table III summarises the LSV findings. The ICN result is not surprising but the poor performance of RK2, and the high efficiency of the box method, might be unexpected.

TABLE III

Relative efficiencies of some simulation methods, applied to simple reversible LSV; operating regions as shown in Figs 13—17 ($n = 5$ or 6 assumed unless otherwise stated)

Method	Fig. no.	Region	n_T	λ	cpu, s	Eff.
Point	13	A	100	0·4	3	1
Box, $n = 2$	15a	A	100	0·4	· 1	3
RK2	16	A?	1 000	0·4	>100	<0·03
ICN	17	A	10	1	1	3

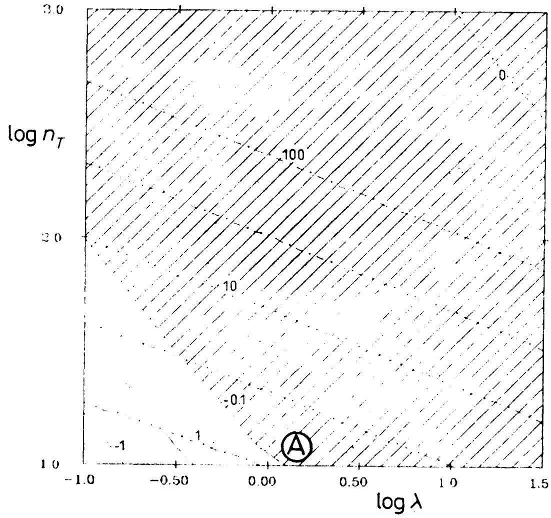


FIG. 17

LSV simulations, using the Crank–Nicolson method, with implicit C_0' (ICN), and $n = 5$. For other information, see legend under Fig. 1

CONCLUSIONS

If one can trust that the results outlined are fairly general, we might conclude that in all cases, the Crank–Nicolson method is more efficient than the explicit point method by a factor of about 20 – 30, at λ-values of 1 – 10, and given that – for derivative boundary conditions – the C'_0 values are included in the implicit discretisations. A surprising finding is that the standard Feldberg (box-) method works best with the "poor" flux approximation $n = 2$ and then works at an efficiency about equal to that of the point method, except in the case of simple LSV, where it matches the Crank–Nicolson method. This must be regarded as a fortuitous result.

The Runge–Kutta integration method[11] is either moderately efficient or, in the case of LSV, rather inefficient, unless one chooses a very narrow operating region. This was inadvertently done in previous work[11], where the efficiency of RK2 in LSV appeared to be very high.

For these one-dimensional simulations, hopscotch shows no advantages and the best Saul'yev variant, ⟨LR, RL⟩, is seen to be mildly more efficient than the explicit point method. Both hopscotch and Saul'yev, however, are expected to be strong contenders for two-dimensional simulations, where no exhaustive comparisons have been made to our knowledge.

The question of what λ value to choose can be answered fairly rationally. We must ignore the tempting zero-error contour lines and we are then left with the result that for the explicit methods EX, BOX and RKI, a λ-range 0·3 – 0·4, as traditionally used, is about ideal. For Crank–Nicolson, somewhat higher λ values are optimal, $\lambda = 3$ being about an ideal all-round point. The upper limit depends on δT but in general, for practical δT (or n_T) values, $\lambda = 30$ or so should not be exceeded.

Lastly, a rational algorithm is essential for simulations with time-dependent boundary concentrations; small inconsistencies in the algorithm can lead to very slow convergence indeed.

The authors gratefully acknowledge discussions with Dr Ole Østerby, of the Computer Science Department, Aarhus University.

REFERENCES

1. Britz D.: *Digital Simulation in Electrochemistry,* 2nd ed. Springer, Heidelberg 1988.
2. Feldberg S. W.: Electroanal. Chem. (Ed.: Bard) *3,* 199 (1969).
3. Shoup D., Szabo A.: J. Electroanal. Chem. Interfacial Electrochem. *199,* 437 (1982).
4. Crank J., Nicolson P.: Proc. Cambridge Phil. Soc. *43,* 50 (1947).
5. Marques da Silva B., Avaca L. A., Gonzalez E. R.: J. Electroanal. Chem. Interfacial Electrochem. *250,* 457 (1988).
6. Britz D.: Anal. Chim. Acta *193,* 277 (1987).
7. Nielsen M. F., Almdal K., Hammerich O., Parker V. D.: Acta Chem. Scand., A *41,* 423 (1987).
8. Oldham K. B.: J. Electroanal. Chem. Interfacial Electrochem. *105,* 373 (1979).

9. Bard A. J., Faulkner L. R.: *Electrochemical Methods*. Wiley, New York 1980.
10. Britz D.: Anal. Chim. Acta *122*, 331 (1980).
11. Britz D.: J. Electroanal. Chem. Interfacial Electrochem. *240*, 17 (1988).
12. Keast P., Mitchell A. R.: Comput. J. *9*, 110 (1966).
13. Britz D., Thomsen K. N.: Anal. Chim. Acta *194*, 317 (1987).
14. Britz D., Heinze J., Mortensen J., Störzbach M.: J. Electroanal. Chem. Interfacial Electrochem., *240*, 27 (1988).
15. Saul'yev V. K.: *Integration of Equations of Parabolic Type by the Method of Nets*. Pergamon Press, New York 1964.
16. Lapidus L., Pinder G. F.: *Numerical Solution of Partial Differential Equations in Science and Engineering*. Wiley, New York 1982.
17. Larkin B. K.: Math. Comput. *18*, 196 (1964).
18. Britz D., Marques da Silva B., Avaca L. A., Gonzalez E. R.: Anal. Chim. Acta *239*, 87 (1990).