Physica D 49 (1991) 61–70
North-Holland

A model for fast computer simulation of waves in excitable media

Dwight Barkley

Applied and Computational Mathematics Department, Princeton University, Princeton, NJ 08544, USA

Starting from a two-variable system of reaction–diffusion equations, an algorithm is devised for efficient simulation of waves in excitable media. The spatio-temporal resolution of the simulation can be varied continuously. For fine resolutions the algorithm provides accurate solution of the underlying reaction–diffusion equations. For coarse resolutions, the algorithm provides qualitative simulations at small computational cost.

1. Introduction

Wave propagation in excitable media provides an important and beautiful example of spatio-temporal self-organization. Spiral waves in unstirred Belousov–Zhabotinsky reagent and impulse propagation along nerve axons are two well-known examples of this phenomenon. Numerous other examples can be found throughout the literature[#1]. Given the widespread interest in these waves, numerical techniques for fast computer simulations of excitable media are of considerable importance. This is particularly true if numerical studies are to be made of large two- and three-dimensional excitable systems.

The fundamental difficulty in simulating excitable media is the separation of spatio-temporal scales in such systems. The time scale on which variables change as the system becomes locally excited is typically several orders of magnitude faster than the time scale on which interesting behavior occurs in the extended medium. Similarly, spatial gradients at fronts where excitation occurs are vastly larger than gradients elsewhere in the medium. If a numerical scheme is to accurately simulate a given excitable system, then it must resolve the fast dynamics of excitation, and unfortunately this requires a fine spatial mesh and a time step which is very small in comparison with the time scale of ultimate interest.

Cellular-automaton models have been proposed as method of circumventing the scale-separation problem [4–10]. In effect, these models reduce the fast dynamics of excitation to a single, discontinuous, jump in some state variable. Recently, very impressive high-speed simulations of waves in excitable media have been obtained using the cellular-automaton approach [8–10]. Despite the success of automaton models, they have some disadvantages: (1) they are governed by somewhat ad hoc rules, and (2) because they do not resolve the fast dynamics, they cannot obtain arbitrarily fine spatio-temporal resolution. This second point means that it is difficult to assess the validity of results from automaton simulations without resorting to simulations by other methods.

Here we propose a simple numerical model which offers most of the advantages of cellular automata, but in addition has the following virtues: (1) it is based directly on a system of reaction–diffusion equations and (2) the spatio-temporal resolution can be adjusted continuously. For coarse resolutions, the fast dynamics is not accurately resolved by the model, but the impor-

[#1] These Proceedings give the most recent review of the field. Refs. [1–3] are other reviews of interest.

0167-2789/91/$03.50

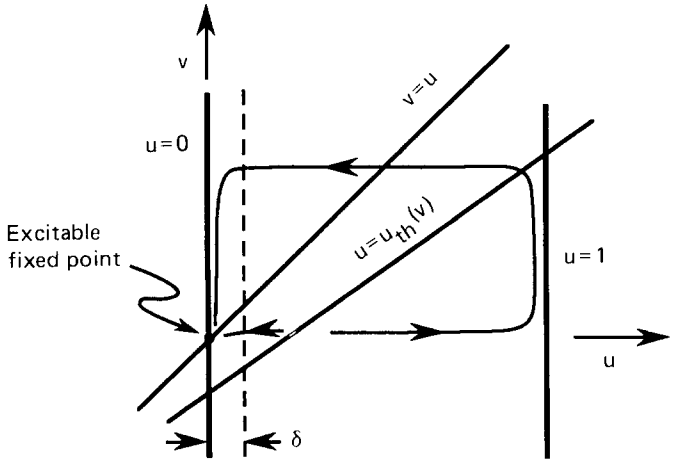


Fig. 1. Illustration of the local dynamics. The axes are the variables u and v. Shown are the system nullclines: the v nullcline $g(u,v)=0$ is the line $v=u$, and the u nullcline $f(u,v)=0$ consists of three lines: $u=0$, $u=1$, and $u=u_{\rm th}(v)=(v+b)/a$. An excitable fixed point sits at the origin where the u and v nullclines intersect. $u_{\rm th}$ is the excitability threshold for the fixed point. Initial conditions near the fixed point and to the left of the threshold, decay directly to the fixed point. Initial conditions to the right of the threshold undergo a large excursion before returning to the fixed point. δ denotes a small "boundary layer" flanking the left branch of the u nullcline. If the system is outside the boundary layer then is excited, otherwise it is recovering.

tant characteristics of excitability are maintained[#2]. In the limit of a fine resolution, the numerical scheme provides accurate solutions to the underlying reaction–diffusion equations. Thus low-resolution simulations can be used to gain insight and explore parameter space at small computational expense, while high-resolution simulations are available with the same compute code and can be used to assess the validity of low-resolution results.

2. The model

Our starting point is a two-variable system of reaction–diffusion equations modeling the dynamics of an excitable medium:

$$\frac{\partial u}{\partial t}=f(u,v)+\nabla^2 u,\quad \frac{\partial v}{\partial t}=g(u,v),\qquad (1)$$

where the functions $f(u,v)$ and $g(u,v)$ express the local kinetics of the two variables u and v. By choice of length scales, the diffusion coefficient for the u-variable is scaled to unity. For simplicity, we assume that v does not diffuse and discuss the applicability of this approximation at the conclusion. At present the spatial domain is arbitrary; later we shall specialize to a two-dimensional square domain with Neumann boundary conditions.

We model the local kinetics with equations

$$f(u,v)=\epsilon^{-1}u(1-u)\left[u-u_{\rm th}(v)\right],$$

$$g(u,v)=u-v,\qquad (2)$$

where $u_{\rm th}(v)=(v+b)/a$, and a, b, and ϵ are parameters, with ϵ generally small. The local dynamics, that is the dynamics in the absence of diffusion, is illustrated in fig. 1 with a typical nullcline picture of phase space. The system has a stable but excitable fixed point at the origin where the nullclines, $f(u,v)=0$ and $g(u,v)=0$, intersect. The variables u and v are known as the excitation and recovery variables, respectively. Two-variable models of this kind are ubiquitous

[#2] For coarse spatio-temporal discretizations, it is perhaps appropriate to consider our model as a coupled-map lattice, see e.g. ref. [11].

in the study of excitable systems [2, 3][#3]; these particular local kinetics have been used recently in the study of the periodic–quasiperiodic transition for spiral waves [12]. Only three parameters appear in the local kinetics and this is of considerable advantage when it comes to exploring parameter space and classifying the model dynamics. We return to this point at the conclusion.

Any numerical scheme for solving (1) should recognize and take advantage of the two distinct dynamical states of the system: excitation and recovery. To be precise, consider a small "boundary layer", of size δ, flanking the left branch of the u nullcline as shown in fig. 1. For our purposes here, we call a given point in the spatial domain excited if (u,v) lies outside the boundary layer, i.e. if $u > \delta$. A point in the spatial domain is recovering if it is within the boundary layer. Regions of excitation and recovery for two cases are shown in plates Ic and Ie discussed below[#4].

The following features of excitation and recovery should be considered in the design of efficient numerical schemes. (i) Within the excited region, the time scale of u can be very fast due to the small parameter ϵ. In particular, transitions between the vertical branches of the u-nullcline are very quick. Thus, either small time steps must be taken to accurately resolve the fast dynamics, or if large (less accurate) time steps are to be used, then care must be taken to prevent numerical instabilities and to preserve the qualitative character of the fast jumps in u. (ii) Within the recovery region the dynamics is very simple. The excitation variable u is essentially zero, so that the local dynamics effectively reduces to exponential decay of the recovery variable v. Moreover, in the interior of recovery regions, diffusion is negligible because the spatial profile of u is basically flat. With these points in mind, we design an algorithm for solving (1).

[#3] It should be noted that problems can arise in our model if the system gets close to the "corners" where the diagonal segment of the u-null-cline intersects the vertical segments. There are no such difficulties for the spiral waves presented here.

[#4] For some purposes, one might want to called a point excited if it lies to the right of the excitability threshold u_{th}, and recovering if it lies to the left of the threshold. Also, in some cases the term "boundary layer" is used for regions of space where u makes fast jumps.

2.1. Local dynamics

For the moment we ignore the diffusion term and consider a scheme for time stepping the local dynamics. With the approximation that $u = 0$ within the boundary layer, we obtain the following simple algorithm for the local dynamics:

if $u^n < \delta$

$$u^{n+1} = 0,$$

$$v^{n+1} = (1 - \Delta t)v^n,$$

else

$$u_{\text{th}} = (v^n + b)/a,$$

$$v^{n+1} = v^n + \Delta t\,(u^n - v^n),$$

$$u^{n+1} = F(u^n, u_{\text{th}}),$$

where u^n and v^n are the value of variables u and v at the nth time step (at some point in the spatial domain) and Δt is the time step. The function F gives the time stepping of u outside the boundary layer and will be discussed momentarily. Inside the boundary layer, u is simply set to zero.

The variable v is stepped both inside and outside the boundary layer by the explicit-Euler method, with the condition that u is set to zero within the boundary layer. The time scale of v is so slow in comparison with the time scale of u, that even with time steps very large relative to the u time scale, explicit-Euler stepping of v is both stable and accurate.

The function F for stepping the u-dynamics may be either of explicit or implicit form. With explicit-Euler time stepping, we have found that $\Delta t \approx \epsilon$ is the maximum Δt for obtaining reasonably accurate results in the fast regions. This Δt

is also near the stability limit for an explicit-Euler step. In order to take very large, less accurate, time steps a semi-implicit form for F can be used. This is obtained from:

$$u^{n+1} = u^n + (\Delta t/\epsilon)u^{n+1}(1-u^n)(u^n - u_{th})$$

if $u^n \le u_{th}$,

$$= u^n + (\Delta t/\epsilon)u^n(1-u^{n+1})(u^n - u_{th})$$

if $u^n > u_{th}$,

where u at the future time is used in those factors on the right-hand side which undergo largest change as the system approaches the stable branches of the u-nullcline. This semi-implicit form keeps the time stepping of the kinetics from overshooting the stable branches of the u-nullcline even if a large time step is taken in the fast region. Solving the above expressions for u^{n+1}, we obtain

$$F(u^n, u_{th}) = \frac{u^n}{1-(\Delta t/\epsilon)(1-u^n)(u^n - u_{th})}$$

if $u^n \le u_{th}$,

$$= \frac{u^n + (\Delta t/\epsilon)u^n(u^n - u_{th})}{1+(\Delta t/\epsilon)u^n(u^n - u_{th})}$$

if $u^n > u_{th}$. (3)

For small $\Delta t/\epsilon$ the denominators in the above expression can be expanded to recover the explicit-Euler form for F:

$$F(u^n, u_{th}) = u^n + (\Delta t/\epsilon)u^n(1-u^n)(u^n - u_{th}) + \mathcal{O}(\Delta t^2).$$

For large $\Delta t/\epsilon$, F goes over to the limit

$$F(u^n, u_{th}) = 0 \quad \text{if } u^n < u_{th},$$
$$= u_{th} \quad \text{if } u^n = u_{th},$$
$$= 1 \quad \text{if } u^n > u_{th},$$

while remaining continuous for any finite Δt. This large-Δt (or small-ϵ) limit models the fast excitation dynamics in the crudest way possible, and is reminiscent of cellular automata [9] in which the u-variable takes on just two values, 0 and 1. Thus, (3) is a representation for u-dynamics which in the small-$\Delta t/\epsilon$ limit gives an accurate time step, while in the large-$\Delta t/\epsilon$ limit preserves the important characteristics of excitability.

2.2. *Diffusion*

Having presented a scheme for stepping the local dynamics, we turn to the efficient treatment of the diffusion term. With the approximation that $u = 0$ within the boundary layer, the u-field is flat in the interior of recovery regions, and hence the Laplacian is zero there and need not be evaluated. To avoid unnecessary computation, the Laplacian can be evaluated "actively" rather than "passively". By this we mean the following. Consider the five-point finite-difference Laplacian formula

$$h^2\nabla^2 u_{ij} = u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{ij},$$

where u_{ij} with the value of u at grid point (i, j) and h is the grid spacing. (We now restrict attention to regular square lattices.) Passive evaluation of the Laplacian is obtained by looping grid indices and evaluating directly the above formula at each point in the spatial domain, that is,

for each i, j

$$\text{lap}_{ij} \leftarrow u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{ij}.$$

The factor of h^2 is absorbed into the diffusion coefficient.

Alternatively, active evaluation is obtained by considering the contribution that each point

makes to the Laplacian of nearby points, that is,

for each i, j

$\text{lap}_{ij} \leftarrow 0,$

for each i, j

$\text{lap}_{ij} \leftarrow \text{lap}_{ij} - 4u_{ij},$

$\text{lap}_{i+1,j} \leftarrow \text{lap}_{i+1,j} + u_{ij},$

$\text{lap}_{i-1,j} \leftarrow \text{lap}_{i-1,j} + u_{ij},$

$\text{lap}_{i,j+1} \leftarrow \text{lap}_{i,j+1} + u_{ij},$

$\text{lap}_{i,j-1} \leftarrow \text{lap}_{i,j-1} + u_{ij}.$

Clearly the two methods of evaluating the Laplacian give the same result. What makes active evaluation of the Laplacian desirable is that it can be incorporated into the algorithm for the local dynamics in such a way that unnecessary computation is avoided at points which make zero contribution to the Laplacian of the u-field. Specifically, the following algorithm updates a single grid point in the spatial domain and computes its contribution to the Laplacian of neighboring points for use at the next time step:

if $u_{ij} < \delta$

$u_{ij} \leftarrow D\,\text{lap}_{kij},$

$v_{ij} \leftarrow (1 - \Delta t)v_{ij}$

else

$u_{\text{th}} \leftarrow (v_{ij} + b)/a,$

$v_{ij} \leftarrow v_{ij} + \Delta t\,(u_{ij} - v_{ij}),$

$u_{ij} \leftarrow F(u^n, u_{\text{th}}) + D\,\text{lap}_{kij},$

$\text{lap}_{k'ij} \leftarrow \text{lap}_{k'ij} - 4u_{ij},$

$\text{lap}_{k',i+1,j} \leftarrow \text{lap}_{k',i+1,j} + u_{ij},$

$\text{lap}_{k',i-1,j} \leftarrow \text{lap}_{k',i-1,j} + u_{ij},$

$\text{lap}_{k',i,j+1} \leftarrow \text{lap}_{k',i,j+1} + u_{ij},$

$\text{lap}_{k',i,j-1} \leftarrow \text{lap}_{k',i,j-1} + u_{ij},$

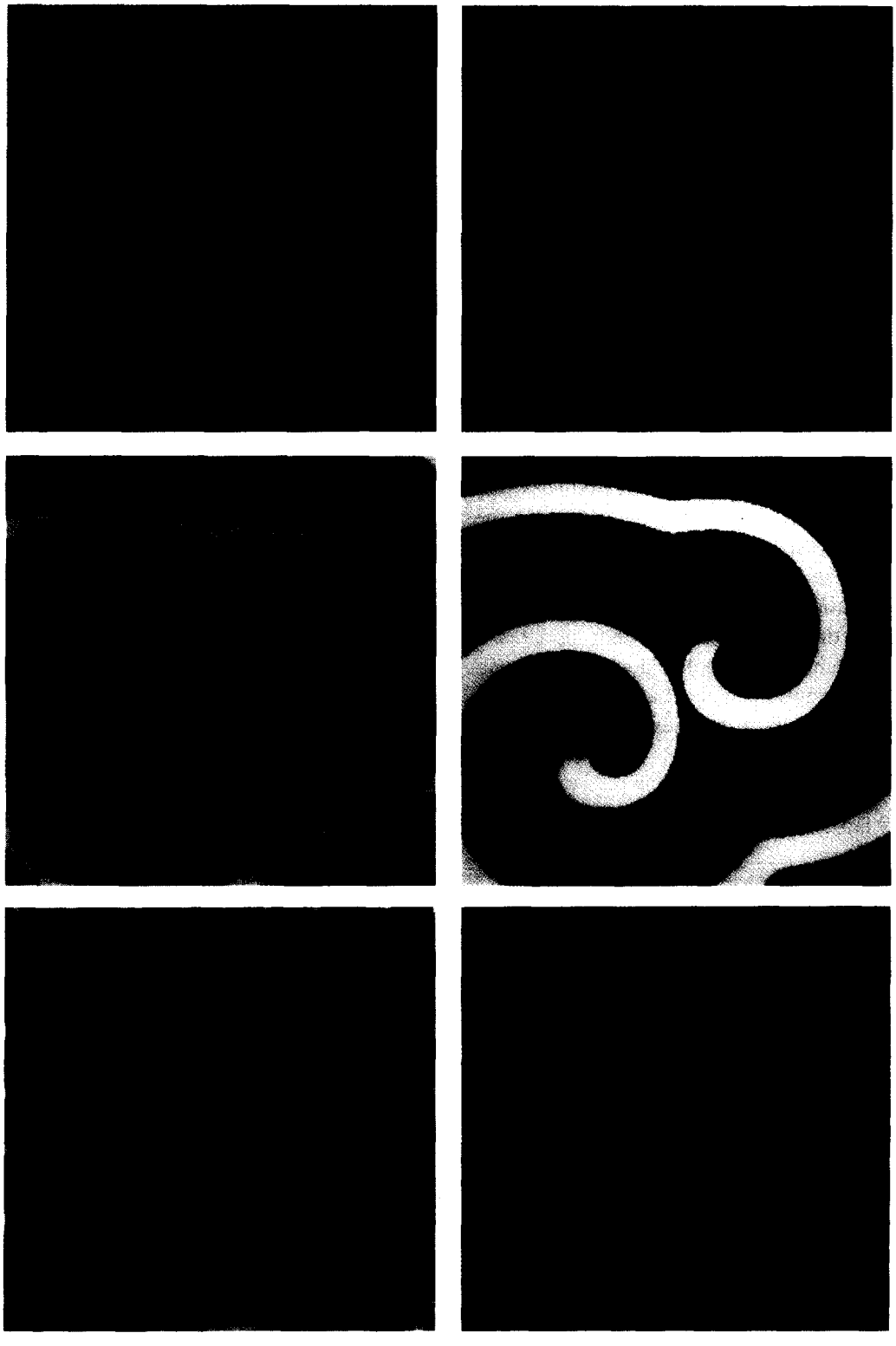
$\text{lap}_{kij} \leftarrow 0$ (4)

where $D = \Delta t/h^2$. The Laplacian $\text{lap}_{k,i,j}$ has three subscripts, the first of which takes on just two values (zero and one, say). The values of k and k' are interchanged at every time step. In effect, there are two Laplacian fields which are used in alternation: the first (unprimed) is used to update the u-field at the current step and is then set to zero for use at the next time step. The second (primed) Laplacian is computed for use at the next time step. A complete subroutine for taking one time step is given in the appendix. In the limit $\Delta t, h, \delta \to 0$, and with appropriate boundary conditions, we expect the numerical solution obtained from (4) to converge to the solution of PDE (1) [13].

3. Results

We have simulated spiral waves using the subroutine given in the appendix with the implicit form for the function F. The spatial domain is a square grid of area L^2 containing N^2 grid points. Hence, the grid spacing is $h = L/(N-1)$. No-flux boundary conditions are imposed on the domain boundary. There are seven parameters for the problem: the four "physical parameters", a, b, ϵ, and L, and three "numerical parameters", Δt, δ, and N. As a practical matter, we have found it convenient to fix the relationship between spatial and temporal discretizations. For all results reported here $\Delta t = L^2/5(N-1)^2$.

Plate I shows some representative results from our model. Plates Ia and Ib show a single spiral wave at two different resolutions. The parameter values for (a) are: $a = 0.3$, $b = 0.01$, $1/\epsilon = 200$, $L = 40$, $N = 81$, and $\delta = 10^{-4}$. With these parameters, $h = 0.5$ and $\Delta t/\epsilon = 10$, and this is the coarsest resolution found to produce meaningful results. For coarser resolutions grid effects become dominant and there is significant slowing in the wavespeed along the grid diagonals. Even at the resolution of Ia significant grid effects are



sometimes observed[#5]. Note that the time step is an order of magnitude larger than that which is possible with explicit time stepping of the reaction kinetics.

Simulations at the resolution of plate Ia are extremely fast: the execution time required for one wave rotation is about 2 s using single precision arithmetic on a Silicon Graphics 4D/200 series workstation with a Mips R3000 processor. This execution time is comparable to that recently reported for cellular-automata simulations [9]. An exact comparison of computational speeds is not possible, however, because of differences between cellular-automaton approach and that used here. While we do not expect our simulations to achieve quite the speed of well-optimized cellular automata (due primarily to the number of floating-point operations required to compute the function F), we do expect our model to be competitive on machines with good floating-point hardware.

The advantage of our model is that the spatio-temporal resolution of the simulation can be increased in a well controlled manner. For example, the spiral tip in plate Ia exhibits complex motions (meandering [14]), but at the resolution of the figure it is not possible to obtain quantitative results on this motion. Plate Ib shows the same situation as Ia except that the number of grid points had been increased to $N = 121$ and the box size has been decreased to $L = 25$ so that the motion of the spiral tip is easily discernible. The white curve shows the path of the spiral tip over the two wave rotations leading to the state shown. The spiral tip is defined as the intersection of the two contours $u = 1/2$ and $f(u = 1/2, v) = 0$, where f is given in (2). Increasing the resolution further produces only small quantitative changes in the wave dynamics[#6].

Shown in plates Ic and Id are contours for the excitation and recovery variables for a pair of co-rotating spiral waves. The parameter values are the same as in Ia except that $N = 241$. This gives $h = 0.167$ and $\Delta t/\epsilon \simeq 1$, corresponding to a well resolved simulation. The point we wish to make here is that even at fine resolutions our algorithm is very efficient. Throughout the blue region in Ic the system is within the boundary layer illustrated in fig. 1. Hence throughout this region (more than 70% of the domain), our algorithm requires just one conditional evaluation and two floating-point multiplications per grid point per time step. The execution time on the aforementioned Silicon Graphics workstation for one wave rotation is only 80 s at this resolution.

Plates Ie and If show representative simulations in large boxes containing many spiral waves. The parameter values for Ie are the same as for Ic except that $a = 0.4$, $1/\epsilon = 150$, $L = 100$ and $\delta = 10^{-3}$. The waves shown were obtained by breaking the waves of Ic many times. (We have found that wave breaking is easily accomplished by transposing the right and left halves of the domain.) The parameter values for If are the same as for Ic except that $a = 0.5$, $L = 200$, $N =$

[#5]Grid anisotropy can be reduced by using a nine-point Laplacian formula and operator splitting as discussed in ref. [12].

[#6]The path of the spiral tip provides a very stringent diagnostic of numerical solutions. For completely converged, anisotropic results the resolution must be increased beyond that of plate Ib.

◀ Plate I. Representative results from model simulations. Parameter values are given in the text. (a) and (b) show the v-field for a single spiral wave at two different resolutions. Colors range from red at the minimum value of v to blue at the maximum value of v. The white curve in (b) is the path of the spiral tip during two wave rotations. (c) and (d) show the u- and v-fields, respectively, for a pair of co-rotating spiral waves. Those points in (c) which are within the boundary layer of fig. 1 are shown in blue, points outside the boundary layer, but to the left of the threshold are green, points to the right of the threshold are yellow. Colors in (d) range from white at the minimum value of v to read for the maximum value of v. (e) and (f) show two different simulations in large domains. In (e) the u-field is shown with the same color map as (c). In (f) the v-field is shown with the same color map as (a).

401, and $\delta = 10^{-3}$. The initial condition for If was a random (u, v) field. Both Ie and If are coarse-resolution simulations and are quite fast: the execution time per wave rotation is about 25 s in the case of Ie and about 55 s in the case of If. These low-resolution simulations illustrate how results can be obtained for large domains at relatively high speed using our model. This, in turn, will make the future investigation of such large domains practical.

4. Discussion and conclusion

A few comments on the applicability of our model are in order. The model, specifically the local dynamics (2), has been chosen both for simplicity and for efficient numerical implementation. While the model is not derived from any particular excitable system, it is based on widely recognized characteristics of excitability and the model dynamics should be generally representative of excitable media. There are only three model parameters, a, b, and ϵ, by which the properties of the medium can be adjusted, thus making a complete classification of parameter space a possibility. By the same token, certain details of any particular system might elude our model, for there undoubtedly is not enough freedom with only three parameters to accurately match the detailed characteristics of any specific medium. For this it will be necessary to change the form of the local kinetics.

The numerical parameters δ, Δt, and h enter the model parameter space only in so far as results might vary somewhat with these parameters if they are not sufficiently small for results to be converged to the solution of the underlying PDE. When using low spatio-temporal resolution to explore parameter space, one hopes that the same structure (bifurcation points, etc.) exists at low resolutions as at high resolutions and that only quantitative variation arises as the resolution is varied. This must be verified in practice and this is easily addressed using the proposed model.

It should be kept in mind that in PDE (1) only the excitation variable diffuses. This is appropriate for simulating systems such as neuro-muscular tissue [1–3, 15] and catalytic surfaces [16, 17] in which the recovery variable is immobile. However, caution should be observed in using (1) to simulate systems, such as chemical reactions, in which all species diffuse with approximately the same diffusion coefficient. Our motivation for leaving v-diffusion out of the model is that the simulations are considerably faster with only the excitation variable diffusing. The hope is that under many circumstances neglecting diffusion of the recovery variable does not result in much quantitative error when simulating chemical systems such as the Belousov–Zhabotinsky reaction. In any case, it is a simple matter to add v-diffusion to the simulations[#7].

In conclusion, we have presented an efficient algorithm for simulating waves in excitable media and have shown a variety of results from such simulations. We have focused on the ability of our model to provide both high-speed qualitative results and accurate simulations to the underlying system of reaction–diffusion equations. The numerical scheme is based on a model which is simple and yet contains the essential features of a broad class of excitable systems. The algorithm itself is easily implemented and requires less than 50 lines of computer code for the time-stepping subroutine. Thus, the model presented here offers much in terms of simplicity, speed, and wide applicability.

Acknowledgements

I wish to thank M. Kness, L.S. Tuckerman, and G. Zanetti for helpful discussions, and A. Konstantinov for assistance with color photography. This work has been supported by DARPA grant number N00014-86-K-0759.

[#7] We note that the character of the transition from simple to compound rotation can be affected critically by diffusion of v, and probably should be included when modeling this transition in chemical systems.

Appendix

The following is a complete subroutine, in the language C, for taking one time step of the model.

```
float u[N+1][N+1], v[N+1][N+1], lap[2][N+2][N+2] ;

void step()
{
int i, j ;
float u_th ;

/* interchange k and kprm  */
ktmp = kprm ;
kprm = k ;
k = ktmp ;

/* main loop */
for( i=1; i<=N; i++ )
  for( j=1; j<=N; j++ )
    {
    if ( u[i][j] < DELTA )
      {
        u[i][j] = D * lap[k][i][j] ;
        v[i][j] = one_m_dt * v[i][j] ;
      }
    else
      {
        u_th = one_o_a * v[i][j] + b_o_a ;
        v[i][j] = v[i][j] + dt * ( u[i][j] - v[i][j] ) ;
        u[i][j] = F( u[i][j], u_th ) + D * lap[k][i][j] ;
        lap[kprm][i][j] = lap[kprm][i][j] - 4.* u[i][j] ;
        lap[kprm][i+1][j] = lap[kprm][i+1][j] + u[i][j] ;
        lap[kprm][i-1][j] = lap[kprm][i-1][j] + u[i][j] ;
        lap[kprm][i][j+1] = lap[kprm][i][j+1] + u[i][j] ;
        lap[kprm][i][j-1] = lap[kprm][i][j-1] + u[i][j] ;
      }
    lap[k][i][j] = 0. ;
    }

/* impose no-flux boundary conditions */
for( i=1; i<=N; i++ )
  {
   lap[kprm][i][1] = lap[kprm][i][1] + u[i][2] ;
   lap[kprm][1][i] = lap[kprm][1][i] + u[2][i] ;
   lap[kprm][i][N] = lap[kprm][i][N] + u[i][N-1] ;
   lap[kprm][N][i] = lap[kprm][N][i] + u[N-1][i] ;
  }
}
```

where the parameters have the following meanings: `dt` $= \Delta t$, `one_m_dt` $= 1 - \Delta t$, `one_o_a` $= 1/a$, `b_o_a` $= b/a$.

Note that while the grid indices run from 1 to N, the second two subscripts on the variable lap run from 0 to N+1. This over-dimensioning of the Laplacian is necessary in active Laplacian evaluation to keep the subscripts from going out of bounds at the domain boundaries. For simplicity, the variables u and v are also over-dimensioned (because C uses zero-based indexing).

The function F is only used symbolically in the preceding subroutine. We suggest in-line coding of the function in the following way:

```
#if EXPLICIT  /* explicit form for F  */

   u[i][j] = u[i][j] + dt_o_eps * u[i][j] * (1.0-u[i][j]) * (u[i][j] - u_th)
             + D * lap[k][i][j];

#else         /* implicit form for F  */

   if( u[i][j] < u_th )
     u[i][j] = u[i][j] / (1.- dt_o_eps * (1.0-u[i][j]) * (u[i][j] - u_th) )
               + D * lap[k][i][j];
   else
     {
       temp = dt_o_eps * u[i][j] * (u[i][j] - u_th) ;
       u[i][j] = (u[i][j] + temp) / (1. + temp ) + D * lap[k][i][j];
     }

#endif
```

where dt_o_eps $= \Delta t/\epsilon$. The explicit form should be used for dt_o_eps $\lesssim 1$ because it is more efficient than the implicit form. The implicit form should be used for larger time steps.

References

[1] A.T. Winfree, When Time Breaks Down (Princeton Univ. Press, Princeton, 1987).

[2] V.S. Zykov, Modelling of Wave Processes in Excitable Media (Manchester Univ. Press, Manchester, 1988).

[3] J.J. Tyson and J.P. Keener, Physica D 32 (1988) 327.

[4] N. Wiener and A. Rosenblueth, Arch. Inst. Cardiol. Mex. 16 (1946) 205.

[5] V.I. Krinsky, Biophysics (USSR) 11 (1966) 776.

[6] J.M. Greenberg and S.P. Hastings, SIAM J. Appl. Math. 34 (1978) 515.

[7] V.S. Zykov and A.S. Mikhailov, Sov. Phys. Dokl. 31 (1986) 51.

[8] M. Gerhardt and H. Schuster, Physica D 36 (1989) 209.

[9] M. Gerhardt, H. Schuster and J.J. Tyson, Science 247 (1990) 1563; Physica D 46 (1990) 392, 416.

[10] M. Markus and B. Hess, Nature 347 (1990) 56.

[11] M.-N. Chee, R. Kapral and S.G. Whittington, J. Chem. Phys. 92 (1990) 7315, and references therein;
D. Barkley, in: Nonlinear Structures in Physical Systems, eds. L. Lam and H.C. Morris (Springer, New York, 1990) p. 192.

[12] D. Barkley, M. Kness and L.S. Tuckerman, Phys. Rev. A 42 (1990) 2489.

[13] G.D. Smith, Numerical Solution of Partial Differential Equations: Finite Difference Methods (Clarendon Press, Oxford, 1985).

[14] A.T. Winfree, Science 181 (1973) 937.

[15] A.T. Winfree, J. Theor. Biol. 138 (1989) 353.

[16] J.R. Brown, G.A. D'Netto and R.A. Schmitz, in: Temporal Order, eds. L. Rensing and N.I. Jaeger (Springer, Berlin, 1985) p. 86.

[17] J. Maselko and K. Showalter, Nature 339 (1989) 609; Physica D 49 (1991) 21–32, these Proceedings.