International Journal of Modern Physics C, Vol. 0, No. 0 (1999) 000–000

AN INTEGRATION SCHEME FOR REACTION-DIFFUSION MODELS

MASSIMO NITTI
Dipartimento di Sistemi e Informatica, via S. Marta 3, I-50139 Firenze, Italy
E-mail: massimo@lyapu.dsi.unifi.it

ALESSANDRO TORCINI
INFM, Udr Firenze, L.go E. Fermi 5, I-50125 Firenze, Italy
Dipartimento di Energetica, via S. Marta 3, I-50139 Firenze, Italy
E-mail: torcini@fi.infn.it

STEFANO RUFFO
Dipartimento di Energetica, via S. Marta 3, I-50139 Firenze, Italy
INFN, sez. Firenze
E-mail: ruffo@fi.infn.it

Received (May 07,1999)
Revised (May 12,1999)

A detailed description and validation of a recently developed integration scheme is here reported for one- and two-dimensional reaction-diffusion models. As paradigmatic examples of this class of partial differential equations the complex Ginzburg-Landau and the Fitzhugh-Nagumo equations have been analyzed. The novel algorithm has precision and stability comparable to those of pseudo-spectral codes, but it is more convenient to employ for systems with quite large linear extention L. As for finite-difference methods, the implementation of the present scheme requires only information about the local enviroment and this allows to treat also system with very complicated boundary conditions.

Keywords: Partial Differential Equations; Reaction-Diffusion Models; Integration Schemes.

1. Introduction

The study and development of efficient and accurate integration algorithms for reaction-diffusion partial differential equations (RDPDE) is a compelling task. Because reaction-diffusion equations are used to represent quite a large class of systems ranging from chemical, to biological and physical ones [1]. Within the field of physics deterministic reaction-diffusion models have been used to mimick front-propagation, surface- and interface-growths, turbulent behaviour, pattern formation, excitable media, etc. [2,3,4].

A paradigmatic example of RDPDE is the well-known Fitzhugh-Nagumo equation (FHNE) used to reproduce the onset of solitary waves, periodic wave trains, circular and spiral waves in excitable media [5,6]

$$u_t = D\nabla^2 u + u(u-a)(1-u) - v \quad ; \quad v_t = \varepsilon(\beta u - v) \tag{1}$$

where $u = u(\mathbf{r}, t)$ and $v = v(\mathbf{r}, t)$ are real fields and represent the activator and the inhibitor, respectively. The appearance of different dynamical behaviours and of the associated patterns is ruled by the values assumed by the real parameters D, a, ε and β and by the dimensionality of the system (see Refs. [5,6] for a review).

Another widely studied reaction diffusion model is represented by the complex Ginzburg-Landau equation (CGLE), that is particularly relevant for the description of the dynamics of spatially extended systems both from an experimental and from a theoretical point of view. This because any spatially extended systems that undergoes a Hopf bifurcation from a stationary to an oscillatory state is described by the CGLE sufficiently close to the bifurcation point [7]. The CGLE can be written as

$$A_t = (1 + jc_1)\nabla^2 A + A + (1 - jc_3)|A|^2 A \tag{2}$$

where c_1 and c_3 are real positive numbers, while $A(\mathbf{r}, t) = \rho(\mathbf{r}, t) \exp[j\psi(\mathbf{r}, t)]$ is a complex field of amplitude ρ and phase ψ. Here j is the imaginary unit with $j^2 = -1$. Equation (2) exhibits several different stationary and turbulent regimes and a quite rich literature has been devoted to the description of its phases [8,17]. In the limit $(c_1, c_3) \to (\infty, \infty)$ the CGLE reduces to the nonlinear Schrödinger equation, while the real Ginzburg-Landau equation is recovered in the opposite limit $(c_1, c_3) \to (0, 0)$ [7].

In the present paper we describe in detail the implementation and the performances of a new integration scheme for RDPDE. In particular, we deal with the FHNE and the CGLE in one and two dimensions for different types of boundary conditions. In Section II the new integration algorithm is explained and its application to the FHNE and to the CGLE is described in detail. The treatment of different boundary conditions (namely, periodic, no-flux and fixed ones) is explained in Section III. In Sect. IV the precision and the performances of the present scheme are compared with those of usual pseudo-spectral codes for one- and two-dimensional systems. A brief final discussion is reported in Sect. V.

2. The integration scheme

The algorithm here reported is a modification of the well known "leap-frog" method applied to a spatially extended system, whose evolution is represented by a partial differential equation (PDE). The leap-frog algorithm was originally devised for Hamiltonian systems [10] and heavily employed for classical molecular dynamics simulations [11]. But it can be also implemented for any ordinary differential equation, when the time evolution of a given dynamical variable $\mathbf{z} = \mathbf{z}(t)$ can be written as

$$\dot{\mathbf{z}}(t) = A \ \mathbf{z}(t) + B \ \mathbf{z}(t) \tag{3}$$

where A and B are two operators, generically non linear, that typically do not commute. Suppose now that one is able to integrate separately the two contributions associated to A and B in Eq. (3), but not Eq. (3) as a whole. Therefore one would like to rewrite the formal solution of (3) $\mathbf{z}(\mathbf{t}) = \exp[t(A+B)]\mathbf{z}(\mathbf{0})$ as a product of terms $\exp[a_k tA]$ and $\exp[b_k tB]$. A vast literature has been devoted to the choice of the coefficients a_k and b_k that minimizes the errors done in rewriting the formal solutions in such an approximate way [12,13]. The best known approximation is the so-called Trotter formula

$$\mathrm{e}^{\tau(A+B)} = \mathrm{e}^{\tau A/2}\mathrm{e}^{\tau B}\mathrm{e}^{\tau A/2} + o(\tau^3) \quad , \tag{4}$$

where τ is the time step. For a time interval $t = \tau n$, one can combine more efficiently $n-1$ half steps and obtains

$$\mathrm{e}^{t(A+B)} \simeq \mathrm{e}^{\tau A/2} \left[\mathrm{e}^{\tau B}\mathrm{e}^{\tau A}\right]^{(n-1)} \mathrm{e}^{\tau B}\mathrm{e}^{\tau A/2} \tag{5}$$

which essentially corresponds to the leap-frog (or Verlet) algorithm.
Let us now consider a generic reaction diffusion equation

$$\dot{\mathbf{V}} = \mathbf{G}(\mathbf{V}) + D\nabla^2\mathbf{V} \tag{6}$$

where $\mathbf{V} = \mathbf{V}(\mathbf{x}, t)$ is a classical field. The most used techniques to integrate such a partial differential equation (PDE) are the so-called time-splitting pseudo-spectral codes [14]. These schemes are among the most stable and efficient and we will illustrate it in the case of the leap-frog time splitting (5)[15]. Referring to (3), the operators are identified in the following manner

$$A\mathbf{V} = \mathbf{G}(\mathbf{V}) \quad , \quad B\mathbf{V} = D\nabla^2\mathbf{V} \tag{7}$$

where now A is a non-linear local (in space) operator and B a linear non-local one. One should now solve separately the evolution equation involving the operator A and B and then apply formula (5). The solution of the non-linear part is usually performed by standard integration techniques for ODE, but in some cases it can be even solved exactly (as it will be shown for the CGLE). To integrate the part involving the spatial derivatives, we consider the Fourier transform of the field $\tilde{\mathbf{V}}(\mathbf{p}, t)$ and of the corresponding evolution equation

$$\dot{\tilde{\mathbf{V}}} = -p^2 D\tilde{\mathbf{V}} \tag{8}$$

whose solution is simply

$$\tilde{\mathbf{V}}(\mathbf{p}, t+\tau) = \exp[-p^2 D\tau]\tilde{\mathbf{V}}(\mathbf{p}, t) \tag{9}$$

In order to obtain the solution in the real space it is now sufficient to back Fourier transform the signal. Therefore, an integration over a time Δt with discrete time step $\tau = \Delta t/n$ and adopting a leap-frog scheme can be summarized as

$$\mathbf{V}(\mathbf{r}, t+\Delta t) = \left\{ \mathrm{e}^{\frac{\tau}{2}A} \left[F^{-1} \mathrm{e}^{-p^2 D\tau} F \mathrm{e}^{\tau A} \right]^{(n-1)} F^{-1} \mathrm{e}^{-p^2 D\tau} F \mathrm{e}^{\frac{\tau}{2}A} \right\} \mathbf{V}(\mathbf{x}, t) \quad (10)$$

where F indicates the Fourier transform and F^{-1} the reverse transform.
The integration method we propose follows exactly the same scheme as the one just shown, but the step involving spatial derivatives is directly solved in the real space. As previously noticed, integration in Fourier space reduces to a simple product (see (9)); in the direct space this corresponds to perform a convolution integral of the signal with an appropriate kernel

$$\mathbf{V}(\mathbf{r}, t+\tau) = \int d\mathbf{s} K(\mathbf{s}, \tau) \mathbf{V}(\mathbf{r}-\mathbf{s}, t) \quad , \qquad (11)$$

where $\mathbf{r} = (x_1, x_2, x_3)$ in the three dimensional case. The expression of the kernel in 3d for the reaction-diffusion systems is straightforward

$$K(\mathbf{r}, t) = \frac{1}{(4D\pi t)^{3/2}} \exp\left[\frac{-|r|^2}{4Dt} \right] \quad . \qquad (12)$$

In order to implement this scheme on a computer, we have to deal with a discrete time step τ and with a discrete spatial resolution $\Delta = (\Delta_1, \Delta_2, \Delta_3)$. The discrete version of the convolution integral (11) reads

$$\mathbf{V}(\mathbf{l}, i+1) = \sum_{n_1, n_2, n_3} K(\mathbf{n}) \mathbf{V}(\mathbf{l}-\mathbf{n}, i) \qquad (13)$$

where $\mathbf{r} = (\Delta_1 n_1, \Delta_2 n_2, \Delta_3 n_3)$ and $\mathbf{n} = (n_1, n_2, n_3)$ is the spatial index, while i is the temporal one ($t = i\tau$) and the sum runs over all the lattice. This approach maintains exactly the same accuracy of the spectral codes, but it is inefficient in terms of CPU time. Since the kernel decays rapidly along any spatial direction, a first naive improvement could consist in evaluating the sum in (13) only over q nearest neighbours of each considered site, but this approach, for precisions comparable to those of pseudo-spectral codes, is also quite time consuming [16].
A better strategy consists in truncating the sum in Eq. (13) and in substituting the discretized kernel with a set of unknown coefficients $C(\mathbf{n})$, that should be conveniently determined [17]

$$\mathbf{V}(\mathbf{l}, i+1) = \sum_{n_1, n_2, n_3 = -N_c}^{N_c} C(\mathbf{n}) \mathbf{V}(\mathbf{l}-\mathbf{n}, i) \qquad (14)$$

where N_c is the number of convolution channels considered along each spatial direction. We have devised the following method to determine the $(2N_c+1)^d$ coefficients $C(\mathbf{n})$ for a d-dimensional lattice:

- let us first make the *ansatz* that the field can be well approximated, on the considered grid, by decomposing it on $(2N_c+1)^d$ elements of the following Fourier basis
$$\left\{ \exp\left[j \sum_h k_h \alpha_h x_h \right] \right\}$$
where $k_h = -N_c, \ldots, N_c$ and α_h is a set of parameters to be determined. On this basis the field can be expressed as
$$\mathbf{V}(\mathbf{l}, i) = \sum_{\{k_h\}} \tilde{\mathbf{V}}(\{k_h\}, i) \ \exp\left[j \sum_{h=1}^{d} k_h \alpha_h l_h \Delta_h \right] \quad ;$$
- the time evolution of each Fourier components is determined by the time propagator (8)
$$\exp\left[-\sum (k_h \alpha_h)^2 D \tau \right] \quad ;$$
- substituting in eq. (13) the field decomposed on the Fourier basis and imposing that the kernel should give the exact time evolution on these modes, one is left with the following set of equations :
$$\sum_{\mathbf{n}} \exp\left[-j \sum_{h=1}^{d} k_h \alpha_h n_h \Delta_h \right] C(\mathbf{n}) = \exp\left[-(\sum (k_h \alpha_h)^2 D \tau \right] \tag{15}$$
$\forall k_h = -N_c, \ldots, N_c$, from which one can obtain the coefficients $C(\mathbf{n})$.

In isotropic problems the spatial resolution does not depend on the considered axis (i.e. $\Delta_h = \Delta \ \forall h$) and in that case symmetry reasons suggest that $\alpha_h = \alpha \forall h$. Moreover, the symmetry of the kernel allows to reduce the $(2N_c+1)^d$ system to a system of $\{(N_c+d)!/[N_c!d!] - 1\}$ equations. This because the elements of the kernel are related by the following symmetry relations

$$C(n_1, n_2, n_3) = C'([n_1^2 + n_2^2 + n_3^2]) \tag{16}$$

and by the normalizations condition

$$C(0,0,0) = 1 - \sum_{n_1, n_2, n_3 \neq (0,0,0)} C(n_1, n_2, n_3) \tag{17}$$

However, the problem of the choice of the parameters $\{\alpha_h\}$ is left unsolved. Its values are determined by demanding minimization of the error due to the use of a Fourier basis that is not the complete one. The α_h-parameters should be determined for each choice of the time step, of the spatial resolution and of the diffusion coefficient. An empirical strategy to choose the optimal α_h-values, that turns out to be essentially correct, consists in finding the values that minimize the quadratic sum $SSQ = \sum_{n_1,n_2,n_3=-N_c}^{N_c} |C(n_1,n_2,n_3)|^2$, with the additional constraint that the first 6 cumulants of the discretized kernel should coincide (within an arbitrary precision) with those of the true kernel (12). In Fig. 1 it is shown that such empirical recipe indeed coincides with the request of a minimal integration error.

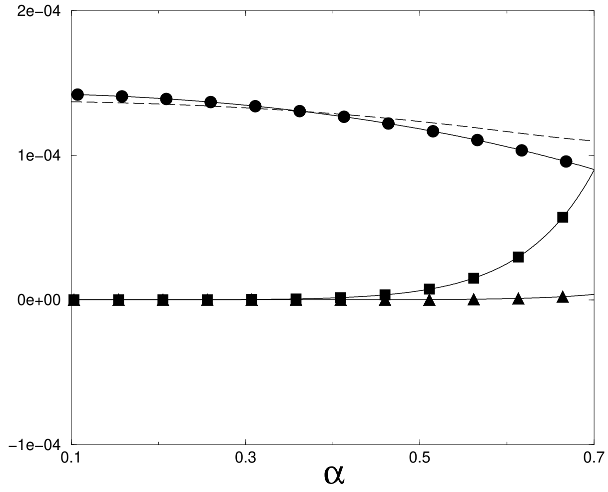

Figure 1: Dependence of the integration error (dashed line) and of the SSQ (reported in arbitrary units) (circles) on the choice of the parameter α for the one-dimensional FHNE, with $\Delta_x = 0.5$, $\tau = 0.10$, $N_c = 5$ and no-flux boundary conditions. The differences between the values of the 4^{th}- (triangles) and 6^{th}-momenta (squares) of the corresponding kernel and those of the true one (12) are also reported. Numerical instabilities have been usually found for small α-values, that in the present case correspond to $\alpha < 0.08$.

3. Specific applications

In this Section we will describe in detail the implementation of our integration scheme for the CGLE and the FHNE, both in one and two dimensions.
Let us first consider the FHNE, in this case we have chosen values of the parameter for which the system is in an excitable state, namely $a = 0.015$, $\beta = 1$, $\varepsilon = 0.006$ and $D = 1$. We have performed the integration of the nonlinear part with a simple Euler scheme, while for what concerns the diffusive part of the equation we have simply to integrate the equation:

$$u_t = D\nabla^2 u \tag{18}$$

This can be done by adopting our scheme where the kernel is given by the real function reported in (12). In Fig. 2 a comparison between the "real" kernel

$K(\mathbf{r})$ and its optimized expression $C(\mathbf{k})$ on the spatio-temporal grid, obtained by solving the system (15), is reported.
A more interesting situation is represented by the CGLE, because this time one is able to solve exactly the differential equation involving the nonlinear operator. This result was firstly reported in [18], where the authors have shown that, at least for the CGLE, the precision of standard time-splitting pseudo-spectral integration schemes is superior to that associated with predictor-corrector and Runge-Kutta algorithms. In order to solve such ODE one has to write separately the following equations for the phase

$$\frac{\partial \psi(\mathbf{r},t)}{\partial t} = c_3 \rho^2(\mathbf{r},t) \tag{19}$$

and for the amplitude

$$\frac{\partial \rho^2(\mathbf{r},t)}{\partial t} = 2[\rho^2(\mathbf{r},t) - \rho^4(\mathbf{r},t)] \quad . \tag{20}$$

The solution of Eq. (20) is simply given by

$$\rho(\mathbf{r},\tau) = 1/\sqrt{\mathrm{e}^{-2\tau}(1/\rho^2(\mathbf{r},0) - 1) + 1} \tag{21}$$

and inserting this into eq. (19) one obtains

$$\psi(\mathbf{r},\tau) = \psi(\mathbf{r},0) + c_3 \left\{\tau + \ln[\rho(\mathbf{r},0)/\rho(\mathbf{r},\tau)]\right\} \quad . \tag{22}$$

For what concerns the integration of the diffusive part, it should be noticed that this time the kernel is complex and it has the expression

$$K(\mathbf{r}) = \left(\frac{\gamma}{\pi}\right)^{d/2} \mathrm{e}^{-\gamma_r |r|^2} \left[\cos(\gamma_i |r|^2) - j \sin(\gamma_i |r|^2)\right] \tag{23}$$

where $\gamma = \gamma_r - j\gamma_i = 1/[4\tau(1 + jc_1)]$.

4. Treatment of boundary conditions

The most commonly used boundary conditions are: periodic (P), no-flux (NF) (where the first derivative of the field vanishes at the boundaries) and fixed (F0) (where the field should have a fixed value at the boundaries, typically set to zero). To treat these different cases with pseudo-spectral codes, one is obliged to use different Fourier basis [14] : namely, the complex basis for P, the cosine basis for NF and the sine basis for F0. Instead, with the present algorithm the same kernel can be used for each of the above cited cases and different boundary conditions are treated just manipulating in different ways the field values at boundaries. This allows to treat, e.g. a discretized two-dimensional field with different boundary conditions on each border or even with various boundary conditions on each single border.

Let us consider a classical field $\{\mathbf{V}(k,i)\}$, with $k = 0, \ldots, N-1$, discretized on a one-dimensional grid of resolution Δx at a fixed time i. Problems arise in the evaluation of the convolution (14) when $k \geq N - N_c - 1$ and $k \leq N_c$, when points "outside" the chain should be considered. For P conditions the field is replicated out of the chain following the simple rules $\mathbf{V}(N+m,i) = \mathbf{V}(m,i)$ and $\mathbf{V}(-m,i) = \mathbf{V}(N-m,i)$. In such case it should be noticed that the first point of the grid is located at any point of the chain, while the last point is placed at a distance L, where $N \times \Delta_x = L$. The NF conditions are treated differently, in this case we have $\mathbf{V}(N+m,i) = \mathbf{V}(N-m+1,i)$ and $\mathbf{V}(-m,i) = \mathbf{V}(m+1,i)$, moreover the first and the last point of the grid are located inside the chain at a distance $(\Delta_x)/2$ from its extrema. Finally, the P0 conditions are implemented as follows : $\mathbf{V}(N+m,i) = -\mathbf{V}(N-m,i)$ for $m \neq 0$ and $\mathbf{V}(N,i) = 0$, while $\mathbf{V}(-m,i) = -\mathbf{V}(m,i)$. In this last situation, the first point of the grid is located at the beginning of the chain, while the last point is within the chain at a distance Δ_x from the extremum.

5. Precision and Performances of the Code

In the present Section we will compare our algorithm with standard time splitting codes [18] for the one- and two-dimensional FHNE and CGLE. It should be stressed that the integration algorithm introduced here and the usual time splitting codes [18] differ only in the treatment of the spatial derivatives. Therefore for what concerns the analysis of the precision and of the performances we will concentrate on such integration step.

In order to give a measure of the precision of the considered algorithms, we have estimated the mean square deviation (MSD) between an orbit obtained by the algorithm under test and a reference orbit, that is assumed to be "exact". In particular, the MSD has been evaluated over a fixed time interval τ and averaged over a total integration time $T = n\tau$. Once the test orbit has been periodically synchronized at the reference one at times $t = m\tau$ $(m = 1, \ldots, n)$.

In the considered time-split integration schemes we can identify two different type of errors, one due to the spatial resolution and the other to the temporal one. The second kind of error can be measured considering a test and reference orbit obtained with the same spatial resolution but with different time integration steps τ (obviously that of the reference orbit should be taken much smaller, typically we have considered $\simeq \tau/5$). The "temporal" MSD are obviously identical for our algorithm and for the standard pseudo-spectral ones. Moreover, with the spatial resolution considered (that corresponds to typical values employed in literature) this error is dominant with respect to the "spatial" one. Data are reported in Table I.

For what concerns the MSD associated to the spatial resolution, it is clear from the data reported in Table II, III, IV and V that this error is different for the 2 algorithms and it depends strongly on the number of convolution

τ	1d	2d
0.1	.164E-03	.742E-02
0.05	.741E-04	.301E-03

Table 1: MSD associated to the temporal integration for the FHNE: the data refers to a spatial resolution $\Delta_x = 0.5$ and no-flux boundary conditions.

channels N_c (when our algorithm is considered). Our algorithm being devised in order to reproduce the precision of pseudo-spectral codes (that we will term FFT), one expects that the MSD associated to our algorithm will be bigger than that associated to the FFT code for any value of N_c. However, already for $N_c \simeq 20 - 30$ in the one- and two-dimensional cases the precision of the FFT code is recovered.

Δ_x	1.0	2.0
$N_c = 5$	2.961E-07	9.399E-05
$N_c = 10$	7.770E-08	8.570E-05
$N_c = 20$	6.593E-08	7.289E-05
FFT	6.727E-08	7.513E-05

Table 2: MSD associated to the spatial integration relative to our algorithm for various values of N_c, the last row refers to usual pseudo-spectral code. The data have been obtained for the one-dimensional FHNE with time step $\tau = 0.05$ and no-flux boundary conditions.

Δ_x	1.0	2.0
$N_c = 3$	9.115E-07	7.619E-05
$N_c = 5$	1.256E-07	4.913E-05
$N_c = 6$	6.884E-08	4.264E-05
$N_c = 7$	5.128E-08	4.292E-05
$N_c = 8$	3.596E-08	4.066E-05
$N_c = 10$	2.838E-08	4.102E-05
FFT	1.942E-08	2.835E-05

Table 3: MSD associated to the spatial integration relative to our algorithm for various values of N_c, the last row refers to usual pseudo-spectral code. The data have been obtained for the two-dimensional FHNE with time step $\tau = 0.10$ and no-flux boundary conditions.

The CPU time required by the pseudo-spectral code will increase with the linear dimension N (where the total number of mesh points in d dimension is N^d) as $N^d \ln(N)$, instead our algorithm will scale as $N^d f(N_c)$. In particular

Δ_x	2.00	1.00	0.50
$N_c = 6$	0.125E-04	0.266E-06	0.123E-08
$N_c = 10$	0.955E-05	0.187E-07	0.120E-11
$N_c = 20$	0.227E-05	0.109E-08	
$N_c = 30$	0.211E-05	0.617E-09	0.109E-13
$N_c = 40$	0.455E-05	0.525E-09	0.921E-14
$N_c = 50$	0.212E-05	0.521E-09	0.866E-14
FFT	0.212E-05	0.521E-09	0.866E-14

Table 4: MSD associated to the spatial integration relative to our algorithm for various values of N_c, the last row refers to usual pseudo-spectral code. The data have been obtained for the one-dimensional CGLE with time step $\tau = 0.05$, periodic bounday conditions and for parameter values $c_1 = 3.5$ and $c_3 = 0.9$. In this case the temporal MSD is $0.161E-02$. The data of this Table have been previously reported in [17].

Δ_x	0.25	0.50	0.75
$N_c = 3$		0.510E-04	0.316E-03
$N_c = 5$		0.463E-05	0.724E-04
$N_c = 10$	0.654E-06	0.359E-06	0.264E-04
$N_c = 15$	0.265E-07	0.308E-06	0.239E-04
$N_c = 20$	0.113E-10	0.225E-06	0.724E-04
FFT	0.254E-13	0.214E-06	0.256E-04

Table 5: MSD associated to the spatial integration relative to our algorithm for various values of N_c, the last row refers to usual pseudo-spectral code. The data have been obtained for the two-dimensional CGLE with time step $\tau = 0.05$, periodic boundary conditions and for parameter values $c_1 = 3.5$ and $c_3 = 0.9$. In this case the temporal MSD is $0.502E-03$.

for the FHNE in the one-dimensional case the ratio between the CPU time requested by the code introduced here an that associated to a standard FFT routine is $\gamma N_c/\ln(N)$, where $\gamma \sim 0.15$. Therefore for typical values ($N_c = 10$, $N = 2048$) such ratio is ~ 0.2. In the two-dimensional case, the ratio can be expressed as $\gamma[(N_c+1)(N_c+2)/2-1]/\ln(N)$ and for the FHNE we have $\gamma \sim 0.3$. In this last situation for $N_c = 4$ and $N = 2048$, the CPU time ratio has a value ~ 0.6. Thus our algorithm is noticeably faster than usual pseudo-spectral codes in one dimension, while in two dimension it becomes convenient to use for grids larger than $N \sim 2048$.

6. Conclusions

In the present paper, an innovative integration scheme for reaction-diffusion PDEs has been reported. We have shown that such scheme is competitive, both in one and two dimensions, with respect to usual time-splitting pseudo-spectral codes. It combines high levels of accuracy with reduced cost in terms of CPU time and an extreme portability, being applicable with a quite limited effort to systems with different boundary conditions. Future improvements of the present algorithm should consist in determining an automatic procedure for the choice of the α_h-parameters. We plan to implement this algorithm in a parallel enviroment and to extend its applicability also to more general classes of PDEs.

Acknowledgements

The first version of this integration scheme has been developed by one of us, for the one-dimensional CGLE, mainly following Peter Grassberger's ideas and in collaboration with Helge Frauenkron [16,17]. We wish also to thank Markus Bär and Roberto Genesio for useful discussions. A.T. acknowledges the hospitality of family Frese in Wuppertal (Germany) during the final write up of this text and Leonardo Torcini for providing him a lap-top.

References

1. E. Bettelheim and B. Lehmann, Ann. Rev. of Comp. Phys., **7** to appear in 1999 (preprint cond-mat/9904205).
2. M.C. Cross and P.C. Hohenberg, Rev. Mod. Phys. **65** 851 (1993).
3. P. Grinrod, *Patterns and Waves. The theory and application of reaction-diffusion Equations*, (Clarendon Press, Oxford,1991).
4. W. Van Sarloos, Phys. Rep. **301** 9 (1998).
5. J. Tyson and J.P. Keener, Physica D **32** 327 (1988).
6. A.T. Winfree. Chaos **1** 303 (1991).
7. Y. Kuramoto, *Chemical Oscillations, Waves and Turbulence*, (Springer Verlag, Berlin, 1984).
8. H. Sakaguchi, Prog. Theor. Phys. **84** 792 (1990); B.J. Shraiman *et al.*, Physica D **57** 241 (1992); H. Chate, Nonlinearity **7** 185 (1994); D.A. Egolf and H.S. Greenside, Phys. Rev. Lett. **74** 1751 (1995); P. Manneville and H. Chate, Physica D **96** 30 (1996); R. Montagne *et al.*, Phys. Rev. Lett. **77** 267 (1997); A. Torcini, Phys. Rev. Lett. **77** 1047 (1997).
9. J.D. Murray, *Mathematical Biology*, (Springer Verlag, Berlin 1993).
10. L. Verlet, Phys. Rev. **159** 98 (1967).
11. M.P. Allen and D.J. Tildesley, *Computer simulations liquids*, (Clarendon Press, Oxford, 1987).
12. H. Yoshida, Phys. Lett. A **150** 262 (1990); E. Forest and R.D. Ruth, Physica D **43** 105 (1990); J. Candy and W. Rozmus, J. Comp. Phys. **92** 230 (1991); J.M. Sanz-Serna, Physica D **60** 293 (1992); M. Suzuki, Int. J. Mod. Phys. C **7** 355 (1996).
13. R.I. McLAchlan and P. Atela, Nonlinearity **5** 541 (1992).
14. W.H. Press et al., *Numerical Recipes*, (Cambridge University Press, Cambridge, 1986).
15. H. Frauenkron and P. Grassberger, Int. J. Mod. Phys. **C5** 37 (1994).
16. A. Torcini, H. Frauenkron, and P. Grassberger, Physica D **103** 605 (1997).
17. A. Torcini, H. Frauenkron, and P. Grassberger, Phys. Rev. E **55** 5073 (1997).

18. D. Goldman and L. Sirovich, Quart. Appl. Math. **53** 315 (1995).

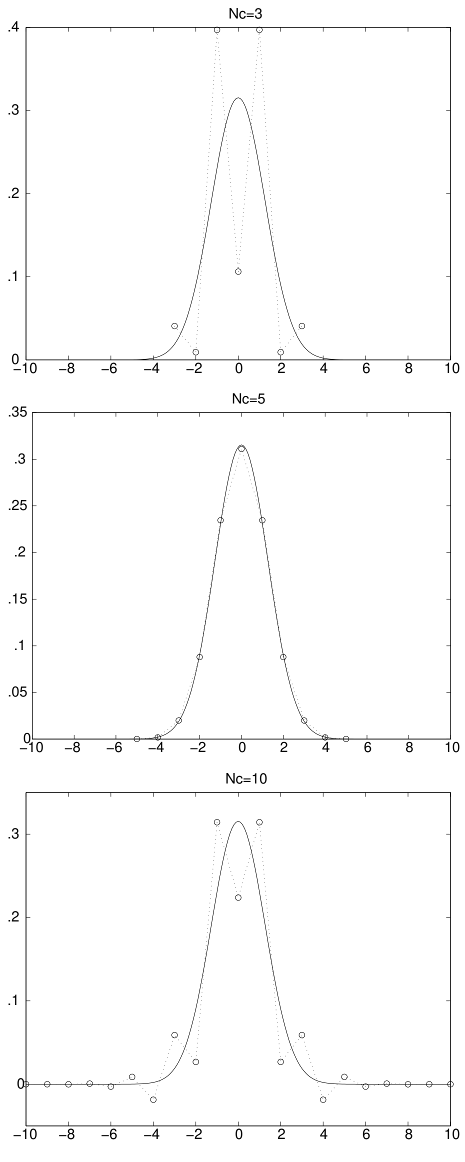

Figure 2: Comparison of the true kernel with the optimized one for three different values of $N_c = 3, 5, 10$ for the one-dimensional FHNE, obtained employing the following parameters $\tau = 0.05,\ \Delta_x = 0.25$ and $\alpha = 0.10$.