New resolution strategy for multi-scale reaction waves using time operator splitting, space adaptive multiresolution and dedicated high order implicit/explicit time integrators

Max Duarte, Marc Massot, Stéphane Descombes, Christian Tenaud, Thierry Dumont, Violaine Louvet, Frédérique Laurent

NEW RESOLUTION STRATEGY FOR MULTI-SCALE REACTION WAVES USING TIME OPERATOR SPLITTING, SPACE ADAPTIVE MULTIRESOLUTION AND DEDICATED HIGH ORDER IMPLICIT/EXPLICIT TIME INTEGRATORS*

MAX DUARTE[†‡], MARC MASSOT[†], STÉPHANE DESCOMBES[§], CHRISTIAN TENAUD[¶], THIERRY DUMONT[‖], VIOLAINE LOUVET[‖], AND FRÉDÉRIQUE LAURENT[†]

Abstract. We tackle the numerical simulation of reaction-diffusion equations modeling multi-scale reaction waves. This type of problems induces peculiar difficulties and potentially large stiffness which stem from the broad spectrum of temporal scales in the nonlinear chemical source term as well as from the presence of steep spatial gradients in the reaction fronts, spatially very localized. In this paper, we introduce a new resolution strategy based on time operator splitting and space adaptive multiresolution in the context of very localized and stiff reaction fronts. It considers a high order implicit time integration of the reaction and an explicit one for the diffusion term in order to build a time operator splitting scheme that exploits efficiently the special features of each problem. Based on recent theoretical studies of numerical analysis, such a strategy leads to a splitting time step which is not restricted neither by fast scales in the source term nor by restrictive diffusive step stability limits, but only by the physics of the phenomenon. Thus, we aim at solving complete models including all time and space scales within prescribed and controlled accuracy, considering large simulation domains with conventional computing resources. The efficiency is evaluated through the numerical simulation of configurations which were so far out of reach of standard methods in the field of nonlinear chemical dynamics for 2D spiral waves and 3D scroll waves as an illustration. Future extensions of the proposed strategy to more complex configurations involving other physical phenomena as well as optimization capability on new computer architectures are finally discussed.

Key words. Reaction-diffusion equations, multi-scale reaction waves, operator splitting, adaptive multiresolution

AMS subject classifications. 33K57, 35A18, 65M50, 65M08

1. Introduction.

Numerical simulations of multi-scale phenomena are commonly used for modeling purposes in many applications such as combustion, chemical vapor deposition, or air pollution modeling. In general, all these models raise several difficulties created by the high number of unknowns, the wide range of temporal scales due to large and detailed chemical kinetic mechanisms, as well as steep spatial gradients associated with very localized fronts of high chemical activity. Furthermore, a natural stumbling block to perform 3D simulations with all scales resolution is either the unreasonably small time step due to stability requirements or the unreasonable memory requirements for implicit methods. In this context, one can consider various numerical strategies in order to treat the induced stiffness for time dependent problems.

*This research was supported by a fundamental project grant from ANR (French National Research Agency - ANR Blancs) *Séchelles* (project leader S. Descombes - 2009-2013) and by a CNRS PEPS INSMI-INSIS project *MIPAC* (project leader V. Louvet - 2009-2010).

[†]Laboratoire EM2C - UPR CNRS 288, Ecole Centrale Paris, Grande Voie des Vignes, 92295 Chatenay-Malabry Cedex, France ({max.duarte,marc.massot,frederique.laurent}@em2c.ecp.fr).

[‡]Supported by a Ph.D. grant from Mathematics (INSMI) and Engineering (INSIS) Institutes of CNRS. Supported by INCA project (National Initiative for Advanced Combustion).

[§]Laboratoire J. A. Dieudonné - UMR CNRS 6621, Université de Nice - Sophia Antipolis, Parc Valrose, 06108 Nice Cedex 02, France (sdescomb@unice.fr).

[¶]LIMSI - CNRS, B.P. 133, Campus d'Orsay, 91403 Orsay Cedex, France (tenaud@limsi.fr).

[‖]Institut Camille Jordan - UMR CNRS 5208, Université de Lyon, Université Lyon 1, INSA de Lyon 69621, Ecole Centrale de Lyon, 43 Boulevard du 11 novembre 1918, 69622 Villeurbanne Cedex, France ({tdumont,louvet}@math.univ-lyon1.fr).

The most natural idea is to use dedicated numerical methods and to solve the complete models where diffusion, reaction and eventually convection are coupled together. One aims at solving strongly coupled nonlinear systems with either a fully implicit method, or yet semi-implicit or linearized implicit methods instead (see [7] and references therein). However, the strong stability restrictions for the latter when dealing with very fast temporal scales, as well as the computing cost and the huge memory requirements of these methods, even if adaptive grids are used, make these strategies difficult to be handled.

An alternative numerical strategy is then to combine implicit and explicit schemes to discretize nonlinear evolution problems in time. Further studies settled the appropriate numerical background for these methods called IMEX, which in particular might be conceived to solve stiff nonlinear problems [29, 23]. These methods are usually very efficient. Nevertheless, on the one hand, the feasibility of utilizing dedicated implicit solvers over a discretized domain becomes soon critical when treating large computational domains. And on the other hand, the time steps globally imposed over partial regions or the entire domain are strongly limited by either the stability restrictions of the explicit solver or by the fastest scales treated by the implicit scheme.

If one takes into account that in many multi-scale problems, the fastest time scales do not play a leading role in the global physics of the phenomenon, one might consider the possibility of using reduced models where these chemical scales have been previously relaxed. These simplified models provide reasonable predictions and the associated computing costs are significantly reduced in comparison with comprehensive chemical models. Nevertheless, these reduced models are usually accessible when the system is well-partitioned and the fast scales can be identified or isolated [24], a process that in realistic configurations, relies on sensitivity analysis which is most of the time difficult to conduct and justify.

It is then natural to envision a compromise, since the resolution of the fully coupled problem is most of the time out of reach and the appropriate definition of reduced models is normally difficult to establish. In this context, time operator splitting methods have been used for a long time and there exists a large literature showing the efficiency of such methods for evolution problems. A splitting procedure allows then to consider dedicated solvers for the reaction part which is decoupled from the other physical phenomena like convection, diffusion or both, for which there also exist dedicated numerical methods. Hence, a completely independent optimization of the resolution of each subsystem might be pursued.

When considering multi-scale waves, the dedicated methods chosen for each subsystem are then responsible for dealing with the fast scales associated with each one of them, in a separate manner, while the reconstruction of the global solution by the splitting scheme should guarantee an accurate description with error control of the global physical coupling, without being related to the stability constraints of the numerical resolution of each subsystem. A rigorous numerical analysis is therefore required to better establish the conditions for which the latter fundamental constraint is verified. As a matter of fact, several works [30, 25, 7] proved that the standard numerical analysis of splitting schemes fails in presence of scales much faster than the splitting time step and motivated more rigorous studies for these stiff configurations [12, 10] and in the case where spatial multi-scale phenomena arise as a consequence of steep spatial gradients [9].

We therefore introduce a new operator splitting strategy, based on these theoretical results, that considers on the one hand, a high order method like Radau5 [17],

based on implicit Runge-Kutta schemes for stiff ODEs, to solve the reaction term; and on the other hand, another high order method like ROCK4 [1], based on explicit stabilized Runge-Kutta schemes, to solve the diffusion problem. Finally, the proposed numerical strategy is complemented by a mesh refinement technique based on Harten's pioneering work on adaptive multiresolution methods [18], being aware of the interest of adaptive mesh techniques for propagating waves exhibiting spatial multi-scale phenomena due to locally steep spatial gradients. The main goal is then to perform computationally very efficient as well as accurate in time and space simulations of the complete dynamics of multi-scale phenomena under study with splitting time steps purely dictated by the physics of the phenomenon and not by any stability constraints associated with mesh size or source time scales.

The paper is organized as follows. In section 2, we first recall the standard time operator splitting schemes; then, we describe the new operator splitting strategy proposed for multi-scale problems, and its coupling with a suitable grid adaptation strategy, the space adaptive multiresolution technique [5, 19], which is briefly presented. The implementation of the numerical strategy is detailed in section 3, as well as the algorithm scheme. In section 4, we present 2D and 3D simulations of a three species reaction-diffusion system modeling the Belousov-Zhabotinsky reaction, and we illustrate the potential and performance of the method by conducting 3D numerical simulations of very stiff reaction waves on a 512^3 mesh size within a reasonable time on a workstation, a simulation out of reach of any standard method. We end in section 5 with some concluding remarks and prospects on future applications including other phenomena such as convection, and numerical developments on new parallel architecture, where we can envision very large scale simulations.

2. Construction of the Numerical Strategy. In this section, we first recall standard operator splitting schemes to then introduce a new splitting strategy for multi-scale waves modeled by stiff reaction-diffusion systems. In the last part, we detail briefly the adaptive multiresolution method that we have implemented as mesh refinement technique for this new resolution technique.

2.1. Time Operator Splitting. Let us first set the general mathematical framework of this work. A class of multi-scale phenomena can be modeled by general reaction-diffusion systems of type:

$$\begin{aligned} \partial_t \mathbf{u} - \partial_{\mathbf{x}} \left(\mathbf{D}(\mathbf{u}) \partial_{\mathbf{x}} \mathbf{u} \right) = \mathbf{f}\left(\mathbf{u}\right), \quad \mathbf{x} \in \mathbb{R}^d, \ t > 0, \\ \mathbf{u}(0, \mathbf{x}) = \mathbf{u}_0(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^d, \end{aligned} \tag{2.1}$$

where $\mathbf{f} : \mathbb{R}^m \to \mathbb{R}^m$ and $\mathbf{u} : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^m$, with the diffusion matrix $\mathbf{D}(\mathbf{u})$, which is a tensor of order $d \times d \times m$.

The proposed numerical strategy normally deals with general problem (2.1). However, to simplify the presentation, we shall consider problem (2.1) with linear diagonal diffusion, in which case the elements of the diffusion matrix are written as $D_{ijk}(\mathbf{u}) = D_k \delta_{ij}$, so that the diffusion operator reduces to the heat operator with scalar diffusion coefficient D_k for component u_k of $\mathbf{u}$, $k = 1, \ldots, m$. Performing a fine spatial discretization, we obtain the semi-discretized initial value problem

$$\begin{aligned} \frac{d\mathbf{U}}{dt} - \mathbf{B}\,\mathbf{U} = \mathbf{F}\left(\mathbf{U}\right), \quad t > 0, \\ \mathbf{U}(0) = \mathbf{U}^0, \end{aligned} \tag{2.2}$$

where $\mathbf{B}$ corresponds to the discretization of the Laplacian operator with the coefficients D_k within; $\mathbf{U}$ and $\mathbf{F}(\mathbf{U})$ are arranged component-wise all over the discretized spatial domain. Considering a standard decoupling of the diffusion and reaction parts of (2.2), we denote $\mathcal{X}^{\Delta t}(\mathbf{U}^0)$ as the numerical solution of the diffusion equation

$$\frac{d\mathbf{U}_D}{dt} - \mathbf{B}\,\mathbf{U}_D = 0, \quad t > 0, \tag{2.3}$$

with initial data $\mathbf{U}_D(0) = \mathbf{U}^0$ after an integration time step Δt. We also denote by $\mathcal{Y}^{\Delta t}(\mathbf{U}^0)$ the numerical solution of the reaction part,

$$\frac{d\mathbf{U}_R}{dt} = \mathbf{F}(\mathbf{U}_R), \quad t > 0, \tag{2.4}$$

with initial data $\mathbf{U}_R(0) = \mathbf{U}^0$.

The two Lie approximation formulae of the solution of system (2.2) are then defined by

$$\mathcal{L}_1^{\Delta t}(\mathbf{U}^0) = \mathcal{X}^{\Delta t}\mathcal{Y}^{\Delta t}(\mathbf{U}^0), \quad \mathcal{L}_2^{\Delta t}(\mathbf{U}^0) = \mathcal{Y}^{\Delta t}\mathcal{X}^{\Delta t}(\mathbf{U}^0), \tag{2.5}$$

whereas the two Strang approximation formulae [26, 27] are given by

$$\mathcal{S}_1^{\Delta t}(\mathbf{U}^0) = \mathcal{X}^{\Delta t/2}\mathcal{Y}^{\Delta t}\mathcal{X}^{\Delta t/2}(\mathbf{U}^0), \quad \mathcal{S}_2^{\Delta t}(\mathbf{U}^0) = \mathcal{Y}^{\Delta t/2}\mathcal{X}^{\Delta t}\mathcal{Y}^{\Delta t/2}(\mathbf{U}^0), \tag{2.6}$$

where Δt is now the splitting time step. It is well known that Lie formulae (2.5) (resp. Strang formulae (2.6)) are approximations of order 1 (resp. 2) of the exact solution of (2.2) in the case where $\mathcal{X}^{\Delta t}$ and $\mathcal{Y}^{\Delta t}$ are the exact solutions $X^{\Delta t}$ and $Y^{\Delta t}$ of problems (2.3) and (2.4). Then, appropriate numerical approximations of $X^{\Delta t}$ and $Y^{\Delta t}$ are required in order to compute Lie and Strang formulae with the prescribed order.

Higher order splitting schemes are also possible. Nevertheless, the order conditions for such composition methods state that either negative time substeps or complex coefficients are necessary (see [16]). The formers imply normally important stability restrictions and more sophisticated numerical implementations. In the particular case of negative time steps, they are completely undesirable for PDEs that are ill-posed for negative time progression.

2.2. Time Integration Strategy. The standard orders achieved with a Lie or Strang scheme are no longer valid when we consider very stiff reactive or diffusive terms (see [12]). Furthermore, if the fastest time scales play a leading role in the global physics of the phenomenon, then the solution obtained by means of a operator splitting scheme will surely fail to capture the global dynamics of the phenomenon, unless we consider splitting time steps which resolve such scales.

In the opposite case when these fast scales are not directly related to the physical evolution of the phenomenon, larger splitting time steps might be considered, but order reductions may then appear due to short-life transients associated to fast variables. This is usually the case for propagating reaction waves where for instance, the speed of propagation is much slower than some of the chemical scales. In this context, it has been proved in [12] that better performances are expected while ending the splitting scheme by the time integration of the reaction part (2.4) or in a more general case, the part involving the fastest time scales of the phenomenon (see a numerical study with convection and complex chemistry in [10]). In particular, in the

case of stiff reaction-diffusion systems with linear diagonal diffusion, no order loss is expected for the $\mathcal{L}_2^{\Delta t}$ and $\mathcal{S}_2^{\Delta t}$ schemes when faster scales are present in the reactive term. However, one must also take into account possible order reductions coming this time from space multi-scale phenomena due to steep spatial gradients whenever large splitting time steps are considered, as analyzed in [9].

All these theoretical considerations give us some insight into the numerical behavior of splitting techniques and thus, help us to select among the various splitting alternatives, depending on the nature of the problem. Nevertheless, the choice of suitable time integration methods for each subsystem is mandatory not only to guarantee such theoretical analysis but also to take advantage of the particular features of each independent subproblem in order to solve them very accurately, with reasonable resources, using error control and adaptive time steps, as it is detailed in the following.

2.2.1. Time Integration of the Reaction: Radau5. Radau5 [17] is not only an A-stable method, but also L-stable, so that very stiff systems of ODEs might be solved without any stability problem. It considers also an adapting time step strategy which guarantees a requested accuracy of the numerical integration and at the same time, allows to discriminate stiff zones from regular ones; hence, smaller time steps correspond to stiffer behaviors. It is a high order method (formally of order 5, which at worst might be reduced to 3) and thus, all error coming from the time integration will be bounded by the one due to the splitting procedure itself.

Nevertheless, this high order method is achieved thanks to an implicit Runge-Kutta scheme, this means that in a general case, nonlinear systems must be solved throughout the time integration process. Even if the solving system tools are highly optimized (which are based on modified Newton's methods), these procedures become very expensive for large systems and important memory requirements are needed in order to carry out these computations. As a consequence, the size of the system of equations to be solved is terribly limited by the computing resources. However, in a splitting scheme context, we easily overcome this difficulty because the reactive term of (2.2) is a system of ODEs without spatial coupling. Therefore, a local approach node by node is adopted where the memory requirements are only set by the number of local unknowns, which normally does not exceed conventional memory resources. Even more, this approach allows straightforward computing parallelization where no data exchange is needed among nodes (see numerical implementations in [13]).

Another very important feature of this strategy is that precious computing time is saved because we adapt the time integration step only at nodes where the reaction phenomenon takes place. For multi-scale reaction waves, this happens in a very low percentage of the spatial domain, normally only in the neighborhood of the wavefront. Therefore, larger time steps are considered at nodes with a chemistry at (partial) equilibrium. This would not be possible if we integrated the entire reaction-diffusion system (2.2) at once. In particular, for complex source mechanisms as considered in [13], complementary studies proved that a huge amount of computing time is saved.

2.2.2. Time Integration of the Diffusion: ROCK4. If we now consider ROCK4 [1], we recall that one of the most important advantages of such method is its explicit character, hence the simplicity of its implementation. In fact, no sophisticated Linear Algebra tools are needed (no resolution of linear systems required) and thus, the resolution is based on simple matrix-vector products. Nevertheless, the computation cost relies directly on the requested quantity of such products, that is the number of internal stages s needed over one time integration step Δt. The memory requirements are also reduced as a consequence of its explicit scheme, nevertheless we must keep

in mind that these requirements increase proportionally with the number of nodes considered over the spatial domain.

ROCK4 is formally a *stabilized* explicit Runge-Kutta method and such methods feature extended stability domain along the negative real axis [28]. Therefore, in order to guarantee the stability for a fixed time step Δt, the number of stages s needed is directly related to the spectral radius $\rho(\partial \mathbf{g}/\partial \mathbf{v})$ (considering a general problem such as $\mathbf{v}' = \mathbf{g}(\mathbf{v})$), since it should verify

$$0.35 \cdot s^2 \geq \Delta t \, \rho \left(\frac{\partial \mathbf{g}}{\partial \mathbf{v}}(\mathbf{v}) \right). \tag{2.7}$$

The method is then very appropriate for diffusion problems because of the usual predominance of negative real eigenvalues for which the method is efficiently stable. A very suitable example is the linear diagonal diffusion problem (2.3) with only negative real eigenvalues and constant spectral radius $\rho(\mathbf{B})$. In our particular applications, the diffusive phenomenon has a leading role of propagator of perturbations over the (partial) equilibrium nodes that result on excitation of the reactive schemes and thus, the propagation of the reaction wave. The resulting self-similar character implies that the number of stages needed will remain practically constant throughout the evolution of the phenomenon. The spectral radius must be previously estimated (for example, using the Gershgoring theorem or even numerically, as proposed by the ROCK4 solver by means of a nonlinear power method).

Once again, the implementation of this diffusion solver over the entire reaction-diffusion system (2.2) will not be appropriate under neither theoretical nor practical considerations, and highlights the inherited advantages of operator splitting. ROCK4 is also a high order method (order 4); therefore, the theoretical operator splitting analysis rest valid and the overall time integration errors are mainly due to the splitting scheme, where all the inner reaction and diffusion time scales are properly solved by these high order dedicated solvers.

2.3. Mesh Refinement Technique. We are concerned with the propagation of reacting wavefronts, hence important reactive activity as well as steep spatial gradients are localized phenomena. This implies that if we consider the resolution of reactive problem (2.4), a considerable amount of computing time is spent on nodes that are practically at (partial) equilibrium. Moreover, there is no need to represent these quasi-stationary regions with the same spatial discretization needed to describe the reaction front, so that the diffusion problem (2.3) might also be solved over a smaller number of nodes. An adapted mesh obtained by a multiresolution process which discriminates the various space scales of the phenomenon, turns out to be a very convenient solution to overcome these difficulties; the basis of this strategy is presented in the following. For further details on adaptive multiresolution techniques, we refer to the books of Cohen [5] and Müller [19].

2.3.1. Basis of Multiresolution Representation. Let us consider nested finite volume discretizations of general problem (2.1) with only one component, $m = 1$. For $j = 0, 1, \cdots, J$ from the coarsest to the finest grid, we build regular disjoint partitions (cells) $(\Omega_\gamma)_{\gamma \in S_j}$ of an open subset $\Omega \subset \mathbb{R}^d$, such that each Ω_γ, $\gamma \in S_j$, is the union of a finite number of cells Ω_μ, $\mu \in S_{j+1}$, and thus, S_j and S_{j+1} are consecutive embedded grids. We denote $\mathbf{U}_j := (u_\gamma)_{\gamma \in S_j}$ as the representation of $\mathbf{U}$ on the grid

S_j where u_γ represents the cell-average of $u : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$ in Ω_γ,

$$u_\gamma := |\Omega_\gamma|^{-1} \int_{\Omega_\gamma} u(t,x) d\mathbf{x}. \tag{2.8}$$

Data at different levels of discretization are related by two inter-level transformations which are defined as follows: (1), the projection operator P_{j-1}^j, which maps $\mathbf{U}_j$ to $\mathbf{U}_{j-1}$. It is obtained through exact averages computed at the finer level by

$$u_\gamma = |\Omega_\gamma|^{-1} \sum_{|\mu|=|\gamma|+1, \Omega_\mu \subset \Omega_\gamma} |\Omega_\mu| u_\mu, \tag{2.9}$$

where $|\gamma| := j$ if $\gamma \in S_j$. As far as grids are nested, this projection operator is *exact* and *unique* [5]. And (2), the prediction operator P_j^{j-1}, which maps $\mathbf{U}_{j-1}$ to an approximation $\hat{\mathbf{U}}_j$ of $\mathbf{U}_j$. There is an infinite number of choices to define P_j^{j-1}, but we impose at least two basic constraints:

1. The prediction is local, *i.e.*, $\hat{u}_\mu$ depends on the values u_γ on a finite stencil R_μ surrounding Ω_μ, where $|\mu| = |\gamma| + 1$.
2. The prediction is consistent with the projection in the sense that

$$|\Omega_\gamma| u_\gamma = \sum_{|\mu|=|\gamma|+1, \Omega_\mu \subset \Omega_\gamma} |\Omega_\mu| \hat{u}_\mu; \tag{2.10}$$

i.e., $P_{j-1}^j \circ P_j^{j-1} = Id$.

In the case where P_j^{j-1} is linear, we have

$$\hat{u}_\mu := \sum_\gamma c_{\mu,\gamma} u_\gamma, \tag{2.11}$$

and if the prediction has some prescribed order $r > 0$ of accuracy, then it is exact for polynomials of degree $r-1$, i.e., if $u \in \prod_{r-1}$, then $u_\gamma = \hat{u}_\gamma$ for all γ [18, 6].

With these operators, we define for each cell Ω_μ the prediction error or *detail* as the difference between the exact and predicted values:

$$d_\mu := u_\mu - \hat{u}_\mu. \tag{2.12}$$

The consistency assumption (2.10), the definitions of the projection operator (2.9) and of the *detail* (2.12) yield

$$\sum_{|\mu|=|\gamma|+1, \Omega_\mu \subset \Omega_\gamma} |\Omega_\mu| d_\mu = 0. \tag{2.13}$$

Thus, we define the *detail vector* as $\mathbf{D}_j = (d_\mu)_{\mu \in \nabla_j}$, where the set $\nabla_j \subset S_j$ is obtained by removing for each $\gamma \in S_{j-1}$ one $\mu \in S_j$ such that $\Omega_\mu \subset \Omega_\gamma$ in order to avoid redundancy from expressions (2.12) and (2.10), and to get a one-to-one correspondence:

$$\mathbf{U}_j \longleftrightarrow (\mathbf{U}_{j-1}, \mathbf{D}_j), \tag{2.14}$$

issued by operators P_{j-1}^j and P_j^{j-1}. By iteration of this decomposition, we finally obtain a multi-scale representation of $\mathbf{U}_J$ in terms of $\mathbf{M}_J = (\mathbf{U}_0, \mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_J)$:

$$\mathcal{M} : \mathbf{U}_J \longmapsto \mathbf{M}_J. \tag{2.15}$$

2.3.2. Compression and Time Evolution on Graded Tree-structured Data. One of the main interests of carrying on such a multi-scale decomposition is that this new representation defines a whole set of regularity estimators all over the spatial domain and thus, a data compression might be achieved as follows.

Given a set of index $\Lambda \subset \nabla^J$ where $\nabla^J := \bigcup_{j=0}^{J} \nabla_j$, we define a truncation operator $\mathcal{T}_\Lambda$, that leaves unchanged the component d_λ if $\lambda \in \Lambda$ and replaces it by 0, otherwise. In practice, we are interested in sets Λ obtained by thresholding:

$$\lambda \in \Lambda \text{ if } |d_\lambda| \geq \varepsilon_j, \quad j = |\lambda|, \tag{2.16}$$

with the level-dependent threshold values:

$$\varepsilon_j = 2^{\frac{d}{2}(j-J)}\varepsilon, \quad j = |\lambda|, \quad j \in [0, J], \tag{2.17}$$

where ε is the threshold value for the finest level J. A data compression is then obtained by discarding the cells whose *details* are not into Λ according to (2.16).

Nevertheless, allegedly useless *details* can not be deliberately deleted because a certain data structure must be respected in order to perform the different computations associated to the multi-scale transformation itself, mainly the prediction operator. The set Λ must then exhibit a *graded tree* structure in order to guarantee the availability of cell values within the local prediction stencil (see [6, 21] for more details on the definition and construction of such structures). In this paper, we will not conduct the analysis of such data structures, but we present the following terminology associated to a tree representation that we will adopt throughout this paper:

- If $\Omega_\mu \subset \Omega_\lambda$ with $|\lambda| = |\mu| - 1$, we say that Ω_μ is a *child* of Ω_γ and that Ω_γ is the *parent* of Ω_μ.
- Moreover, we define the *leaves* $\mathrm{L}(\Lambda)$ of a *tree* Λ as the set of Ω_λ with $\lambda \in \mathrm{L}(\Lambda)$ such that Ω_λ has no children in Λ.
- Finally, we define Ω_λ as a *root* when it belongs to the coarsest grid, that is $\lambda \in S_0$ or $|\lambda| = 0$, in which case, we denote λ as λ_0.

An effective data compression is accomplished because $\mathbf{U}$ is not represented on the finest grid S_J as $\mathbf{U}_J$ anymore, but on Λ_ε, where Λ_ε is the smallest *graded tree* containing Λ defined by (2.16). More precisely, the numerical solution $\mathbf{U}^n$ at time $n\Delta t$ can be represented on an adapted grid by the set $(u_\lambda^n)_{\lambda \in \mathrm{L}(\Lambda_\varepsilon^n)}$.

The time evolution is then performed only on the *leaves* of a fixed adapted grid. A refinement operator $\mathcal{R}$ is therefore defined in order to generate a set $\tilde{\Lambda}_\varepsilon^{n+1}$ containing Λ_ε^n, on which the time integration is computed, such that $\tilde{\Lambda}_\varepsilon^{n+1}$ is adapted for describing the solution at both $n\Delta t$ and $(n+1)\Delta t$. In our numerical implementation, operator $\mathcal{R}$ refines the adapted grid based on the values of the *details*: by creating *children* for all $\lambda \in \mathrm{L}(\Lambda_\varepsilon^n)$ such that $|d_\lambda| \geq \varepsilon_{|\lambda|}$, which adds new levels to Λ_ε^n; and according to Harten's heuristics (see [18]), enlarging uniformly the refined region of Λ_ε^n in order to predict the propagation of the solution. The u_λ^n with $\lambda \in \tilde{\Lambda}_\varepsilon^{n+1} \backslash \Lambda_\varepsilon^n$ can be constructed applying $\mathcal{M}^{-1}$. These criteria are rather conservative, nevertheless they completely avoid unrefined resolution taking into account the propagating nature of reaction waves at finite speed.

An important theoretical result is that if we denote by $\mathbf{V}_J^n := (v_\lambda^n)_{\lambda \in S_J}$, the solution fully computed on the finest grid, and denote by $\mathbf{U}_J^n$, the solution reconstructed on the finest grid that used adaptive multiresolution (keeping in mind that the time integration was really performed on the *leaves* $\mathrm{L}(\tilde{\Lambda}_\varepsilon^n)$ of a compressed representation of $\mathbf{U}^n$); then, for a fixed time $T = n\Delta t$, it can be shown that [18, 6]:

$$\|\mathbf{U}_J^n - \mathbf{V}_J^n\|_{L^2} \propto n\varepsilon. \tag{2.18}$$

Numerical experiments, based on this error estimate and on the splitting ones, allow one to properly choose the various simulation parameters used to predict the expected level of accuracy of the resolution.

2.4. Summary of the Numerical Strategy. The numerical resolution strategy can be summarized as follows:

$$(u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde\Lambda_\varepsilon^n)} \xrightarrow{\mathcal{M}} (u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde\Lambda_\varepsilon^n} \tag{2.19}$$

$$(u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde\Lambda_\varepsilon^n} \xrightarrow{\mathcal{T}_{\Lambda_\varepsilon^n}} (u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\Lambda_\varepsilon^n} \tag{2.20}$$

$$(u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\Lambda_\varepsilon^n} \xrightarrow{\mathcal{R}} (u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde\Lambda_\varepsilon^{n+1}} \tag{2.21}$$

$$(u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde\Lambda_\varepsilon^{n+1}} \xrightarrow{\mathcal{M}^{-1}} (u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde\Lambda_\varepsilon^{n+1})} \tag{2.22}$$

$$(u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde\Lambda_\varepsilon^{n+1})} \xrightarrow{\mathcal{S}^{\Delta t}} (u_\lambda^{n+1})_{\lambda\in \mathrm{L}(\tilde\Lambda_\varepsilon^{n+1})} \tag{2.23}$$

The set $(u_{\lambda_0}^n)$ is defined as the set of roots of $\tilde\Lambda_\varepsilon^n$, that is all $\lambda\in\tilde\Lambda_\varepsilon^n$ such that $|\lambda|=0$ or $\lambda\in S_0$.

For $n=0$, the initial condition should be represented on $\mathrm{L}(\tilde\Lambda_\varepsilon^0)$ in step (2.19), which can normally be the finest grid, that is all Ω_λ such that $|\lambda|=J$ or $\lambda\in S_J$. Nevertheless, this is not possible for large domains simulations, in which case, the initial condition is computed on an intermediate grid level j: all Ω_λ such that $|\lambda|=j$ or $\lambda\in S_j$, and then refined after an initial thresholding process.

The algorithm can schematically be summarized by

$$\mathbf{U}^{n+1} = \mathcal{S}^{\Delta t}(\mathcal{M}^{-1}\mathcal{R}\mathcal{T}_{\Lambda_\varepsilon^n}\mathcal{M}\mathbf{U}^n), \tag{2.24}$$

with the compressed representations of $\mathbf{U}^{n+1}$ and $\mathbf{U}^n$ given by $(u_\lambda^{n+1})_{\lambda\in\mathrm{L}(\tilde\Lambda_\varepsilon^{n+1})}$ and $(u_\lambda^n)_{\lambda\in\mathrm{L}(\tilde\Lambda_\varepsilon^n)}$ respectively, and the Strang operator splitting $\mathcal{S}^{\Delta t}$ given by one of the formulae (2.6) as time integration operator. One might add a last thresholding step in order to obtain the solution on $\mathrm{L}(\Lambda_\varepsilon^{n+1})$ instead of $\mathrm{L}(\tilde\Lambda_\varepsilon^{n+1})$ in order to obtain slightly higher data compression. In what follows, the key aspects of the implemented algorithm are detailed.

3. Algorithm Implementation. A dynamic *graded tree* structure is used in this implementation to represent data in the computer memory. The adapted grid corresponds to a set of nested dyadic grids generated by refining recursively a given cell, depending on the local regularity of the solution. The chosen data structure can handle 1D, 2D and 3D Cartesian geometries; the basic element of the structure is the cell itself, which consists of a set of geometric and physical values, plus pointers to its *parent*, their *children* and the contiguous cells in each dimension, the *neighbors*. Figure 3.1 shows an example of a *graded tree* structure in 1D.

The *roots* correspond to the basis of the tree, Ω_{λ_0}, whereas the *leaves* are the upper elements with no *children* in the tree. In d dimensions, a *parent*-cell at a level j has at most 2^d *children* cells at level $j+1$. When there is only one *root* in the tree, the maximal number of *leaves* N on which the solution might be represented is given by $N=2^{dJ}$, which is exactly the number of cells on the finest grid. The maximal number of cells M in the tree is given by $M=(2^{d(J+1)}-1)/(2^d-1)$. Additionally, in order to guarantee conservativity at the cell interfaces, virtual cells called *phantoms*

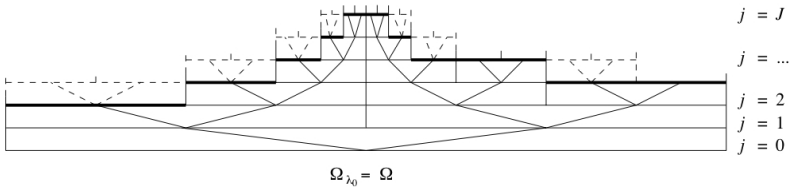

FIG. 3.1. *Example of 1D graded tree structure. Nodes and links to their corresponding children are indicated (solid lines) as well as the leaves (solid bold lines) and the phantoms (dashed lines).*

are added to the tree in order to always compute the numerical fluxes of diffusion and convection operators at the highest grid level between two neighboring cells, following the procedure introduced in [21] for finite volume discretizations.

The implemented code represents the tree-structured data as a set of cells linked by pointers. In a FORTRAN 90/95 environment, a pointer is just an alias to the target; nevertheless, we take advantage of the fact that each pointer has a different state, depending on whether it is associated or not to another object. We work then with cells that are at different grids and that are not necessarily arranged in a contiguous way. Hence, we must conceive the mechanisms to navigate through the tree structure. In this implementation, we adopt a recursive strategy in which one moves from one cell to another passing by the *child* of the first, and by the consecutive *children*, until one gets to the desired cell. At each step, the state of the pointers tells us whether the target exists or not. In this recursive way, we are able to locate *leaves* or any cell and the same kind of procedure is conducted in the opposite direction, from *leaves* towards *roots* when necessary. Pointers to *neighbors* as well as other flags or indicators are not strictly necessary but eases considerably the searching process for certain routines.

3.1. Prediction Operator. To obtain the approximated values $\hat{u}_\mu$ by P_j^{j-1} according to (2.11), we consider centered linear polynomial interpolations of order $2l$, *i.e.* accuracy order $r = 2l+1$, computed with the l nearest *neighbors* in each direction. For instance, in 1D configuration for $l = 1$, the prediction is explicitly given by:

$$\begin{aligned} \hat{u}_{j+1,2k} &= u_{j,k} + \frac{1}{8}(u_{j,k-1} - u_{j,k+1}), \\ \hat{u}_{j+1,2k+1} &= u_{j,k} + \frac{1}{8}(u_{j,k+1} - u_{j,k-1}), \end{aligned} \tag{3.1}$$

where the first index denotes the grid level and the second, the indexation of the cell into the tree. As Cartesian mesh is used, extension to multidimensional polynomial interpolations is easily obtained by a tensorial product of the 1D operator [2, 21]. In the numerical illustrations we will restrict the different analysis to the case $l = 1$.

3.2. Choice of Splitting Time Step. The splitting time step is set by the desired level of accuracy in the resolution of the wave speed, the wave profile, both, or any other parameter, depending on the problem and considering that each subsystem if perfectly resolved. It is thus only depending on the phenomenon we want to describe and therefore, on the degree of decoupling we can achieve between the various subsystems within a prescribed error tolerance. For instance, if a reference wave solution $\mathbf{u}$ of problem (2.1), or the corresponding wavefront speed v, can be computed either numerically or based on theoretical/analytical estimates, then the

approximated solution $\mathbf{u}_{split}$ of speed v_{split}, computed with splitting time step Δt and an operator splitting technique with exact integration of the subsystems, must verify:

$$E_p = \|\mathbf{u} - \mathbf{u}_{split}\|_{L^2} \leq \eta_p, \quad E_v = \frac{|v - v_{split}|}{v} \leq \eta_v, \tag{3.2}$$

where η_p and η_v are accuracy tolerances for the profile and velocity errors: E_p and E_v, respectively. The profile error E_p should be evaluated superposing both $\mathbf{u}$ and $\mathbf{u}_{split}$; however, a simpler and more practical strategy would just evaluate the L^2-error at a fixed time $t^\star$:

$$E = \|\mathbf{u}(t^\star) - \mathbf{u}_{split}(t^\star)\|_{L^2} \leq \eta, \tag{3.3}$$

in which case, both profile and velocity errors are simultaneously considered. Notice that in order to remain coherent with the previous constraints and also to guarantee an accurate resolution of the reaction and diffusion problems, the corresponding accuracy tolerances η_{Radau5} and η_{ROCK4} of these solvers must verify:

$$\eta_{Radau5}, \eta_{ROCK4} < \min\{\eta_p, \eta_v, \eta\}. \tag{3.4}$$

Finally, taking into account that the time evolution is performed on an adapted grid, fixed during each time step, the resulting splitting time step should verify a CFL-like condition:

$$\Delta t \leq \frac{n\Delta x}{v_{split}}, \tag{3.5}$$

where Δx corresponds to the spatial discretization at the finest grid and $n \geq 2$ considers the refinement criterion that enlarges uniformly the refined region (see §2.3.2). This criterion is used to verify that the locally refined spatial gradients rest into the finest regions during a time step evolution; this is required not because of stability issues as for time integration of hyperbolic problems, but to guarantee the spatial accuracy of the approximation.

In the case of self-similar progression of wavefronts, the selection of the time step is simplified by the fact that usually it does not need to be computed more than once. Let us underline that the proposed procedure has been designed in such a way that, in case one is able to estimate such a splitting time step at any computed time based on error control at a given tolerance, then the numerical strategy can be used exactly as it is provided. We will come back on this issue in the concluding remarks.

3.3. Algorithm Scheme. The global algorithm can be summarized as:

1. **INITIALIZATION:**
   - *Initialization of parameters: e.g.* maximum and minimum grid level, domain size, number of *roots*.
   - *Initialization of the mesh structure:*
     - creation of the different grids;
     - initialization of parameters of each cell from the *roots*, *e.g.* position, coordinates, level threshold value ε_j;
     - assignation of *children* and *neighbors* from the *roots*.
   - *Computation of initial solution* at an intermediary grid level.
2. **LOOP IN TIME:**
   - *Computation of cell values:* projection operator P_{j-1}^{j} from *leaves* towards *roots*.

- *Computation of details:* operator $\mathcal{M}$ from *roots* towards *leaves*.
- *Thresholding and graduation:* operator $\mathcal{T}_{\Lambda_\varepsilon}$ throughout the *tree*.
- *Refinement of the tree:* operator $\mathcal{R}$ throughout the *tree*.
- *Computation of cell values from details:* operator $\mathcal{M}^{-1}$ from *roots* towards *leaves*.
- *Creation of phantom cells:* needed for diffusion time step.
- *Time integration:* Strang operator splitting $\mathcal{S}_2^{\Delta t}$ applied only on *leaves*:
  - reaction half time step, time integration by Radau5 cell by cell;
  - diffusion time step, time integration by ROCK4. At each time internal stage, virtual cells are updated through projection/prediction operations;
  - reaction half time step, time integration by Radau5 cell by cell.

3. **OUTPUT:**
   Save adapted grid with the corresponding cell values represented on it.

4. Numerical Simulations. In this last section, we present some numerical illustrations of the proposed strategy. A problem coming from nonlinear chemical dynamics is described and treated. The performance of the method is discussed in the context of 2D and 3D simulations.

4.1. Mathematical Model of Study. We are concerned with the numerical approximation of a model of the Belousov-Zhabotinski reaction, a catalyzed oxidation of an organic species by acid bromated ion (see [14] for more details and illustrations). We thus consider the model introduced in [15] and coming from the classic work of Field, Koros and Noyes (FKN) (1972), which takes into account three species: hypobromous acid $HBrO_2$, bromide ions Br^- and cerium(IV). Denoting by $a = [\text{Ce(IV)}]$, $b = [HBrO_2]$ and $c = [Br^-]$, we obtain a very stiff system of three PDEs:

$$\begin{aligned} \frac{\partial a}{\partial \tau} - D_a \Delta a &= \frac{1}{\mu}(-qa - ab + fc), \\ \frac{\partial b}{\partial \tau} - D_b \Delta b &= \frac{1}{\epsilon}\left(qa - ab + b(1-b)\right), \\ \frac{\partial c}{\partial \tau} - D_c \Delta c &= b - c, \end{aligned} \tag{4.1}$$

with diffusion coefficients D_a, D_b and D_c, and some real positive parameters f, small q, and small ϵ, μ, such that $\mu \ll \epsilon$.

The dynamical system associated with this system models reactive excitable media with a large time scale spectrum (see [15] for more details). Moreover, the spatial configuration with addition of diffusion generates propagating wavefronts with steep spatial gradients. Hence, this model presents all the difficulties associated with a stiff multi-scale configuration. The advantages of applying a splitting strategy to these models have already been studied and presented in [11]. In what follows, we will consider 2D and 3D configurations of problem (4.1).

4.2. 2D BZ Equation. We first consider the 2D application of problem (4.1) with homogeneous Neumann boundary conditions and the following parameters, taken from a preliminary study [11]: $\epsilon = 10^{-2}$, $\mu = 10^{-5}$, $f = 1.6$ and $q = 2 \times 10^{-3}$, with diffusion coefficients $D_a = 2.5 \times 10^{-3}$, $D_b = 2.5 \times 10^{-3}$ and $D_c = 1.5 \times 10^{-3}$. The phenomenon is studied over a time domain of $[0, 4]$ and a space region of $[0, 1] \times [0, 1]$. We define the reference or *quasi-exact* solution as the resolution of the coupled reaction-diffusion problem (4.1) on an uniform mesh of 256×256 performed by Radau5 with

very fine tolerances, $\eta_{Radau5} = 10^{-10}$. The main limitation to perform such computation on finer grids comes from the important memory requirements of Radau5.

Let us consider an application of the proposed MR/Splitting numerical strategy with 8 nested dyadic grids with $N = 2^{2\times 8} = 65536 = 256 \times 256$ cells on the finest grid $J = 8$. The time integration method uses the RDR Strang $\mathcal{S}_2^{\Delta t}$ scheme with Radau5 for the time integration of the reaction term and ROCK4 for the diffusive part, $\eta_{Radau5} = \eta_{ROCK4} = 10^{-5}$. The spiral waves simulated can be seen into Figure 4.1, where colors represent the grid levels at $t = 2$, when the wave is fully developed and at $t = 4$, after a complete rotation period: the adapted grids are tightened around the stiff regions and clearly propagate along the waves.

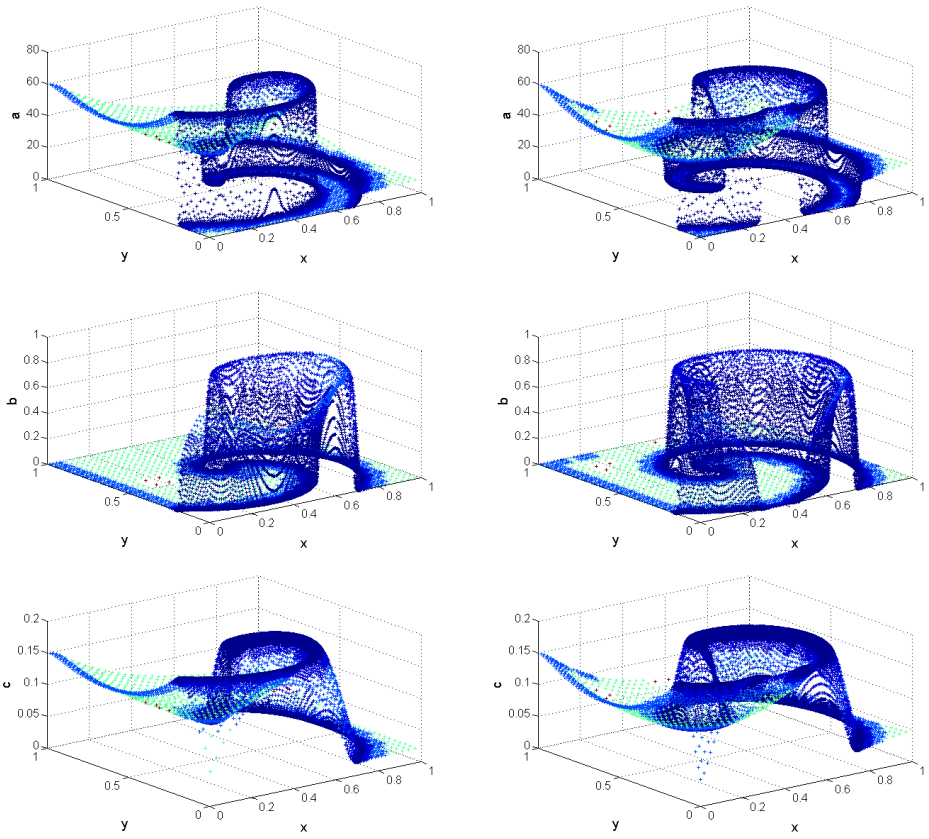

FIG. 4.1. *2D BZ spiral waves. MR/Splitting solutions on adapted grids with $\varepsilon = 10^{-2}$ at $t = 2$ (left) and $t = 4$ (right). Variable a (top), b (center) and c (bottom), where colors represent grid levels. Finest grid:* 256^2.

To illustrate the performance of the method, we have set an accuracy tolerance of $\eta = 10^{-2}$, considering for simplicity the normalized L^2-errors (3.3). Nevertheless, in order to remain consistent with the physics of the phenomenon, we aim also at wave speed resolutions with a maximum relative error of 1%, $\eta_v = 10^{-2}$ according to (3.2). After numerical experiments that considered the splitting solver with different Δt on an uniform grid of 256×256, the order 2 of the numerical scheme according to §2.2, and the reference *quasi-exact* solution previously defined, we have chosen a rounded value of splitting time step with a L^2-error close to η for all three variables and times

$t^\star \in [0,4]$ into (3.3): $\Delta t = 4/1024 \approx 3.9 \times 10^{-3}$.

Verification of η_v with the chosen Δt was conducted in the following way: we studied the time evolution of each variable contained in a 2D slice of the 3D representation into Figure 4.1, which results into a 1D configuration as shown in Figure 4.2 for variable a along y-axis. Then we calculated the propagating speed of each variable given by the *quasi-exact* and splitting solutions, and finally, the relative errors E_v following (3.2), which remain practically lower than 0.2%, and thus, much lower than η_v. Similar results were obtained while considering slices along x-axis. The profile differences of the waves for a fixed $t^\star \in [0,4]$ show that the relative higher L^2-errors computed by (3.3), comes from the spatial shift of the steep gradients, even for a slight speed discrepancy of both resolutions. This is a simple strategy to measure and approximate wave speeds, nevertheless we might think of developing more sophisticated techniques based on more detailed theoretical studies of wave dynamics (see for example [22]), which are out of the scope of this work.

For this particular problem and with the selected Δt, the CFL-like condition (3.5) is verified for all $\Delta x \geq (4 \times 0.7)/(1024 \times 3) \approx 9.11 \times 10^{-4}$, that is up to ~ 1024 points in each dimension, considering $v_{split} \approx 0.7$ and $n = 3$; therefore, this choice of the splitting time step is really consistent and avoids any kind of refinement problem for a wide range of spatial discretizations. Anyway, this is a rather conservative estimate because the numerical tests show that the refined zones during a time step imply normally more than 3 cells, mainly due to the extended stencils obtained by graduation of the tree structure in multidimensional configurations.

The proposed MR/Splitting strategy represents and computes solutions only on adapted grids, the *leaves* of the tree structure, throughout the time domain. Therefore, we define the data compression (DC) as one minus the ratio between the number of cells on the adapted grid (AG) and those on the finest uniform grid (FG), expressing the whole as a percentage:

$$DC = \left(1 - \frac{AG}{FG}\right) \times 100. \tag{4.2}$$

Table 4.1 shows different data compression rates for several threshold values. Smaller threshold values ε imply more refinement and thus, compressions are less important.

TABLE 4.1
2D BZ. Data compression (DC) and number of cells on the adapted grid (AG) for different threshold values ε. Finest grid: 256^2.

	$t=2$		$t=4$	
ε	AG	DC	AG	DC
10^{-1}	10594	83.97	14035	78.58
10^{-2}	15244	76.74	18874	71.20
10^{-3}	27544	57.97	29182	55.47
10^{-4}	55207	15.76	51934	20.76

Considering now the L^2-error of these results, $\mathbf{u}_{MR}$, with respect to the reference *quasi-exact* solution, $\mathbf{u}^J_{qe}$, we decompose it into two parts, the error coming from the splitting process and that of the multiresolution decomposition:

$$\|\mathbf{u}^J_{qe}(t) - \mathbf{u}_{MR}(t)\|_{L^2} \leq \|\mathbf{u}^J_{qe}(t) - \mathbf{u}^J_{split}(t)\|_{L^2} + \|\mathbf{u}^J_{split}(t) - \mathbf{u}_{MR}(t)\|_{L^2}. \tag{4.3}$$

Therefore, we also consider the splitting solution $\mathbf{u}^J_{split}$ obtained without grid adaptation on the uniform finest grid $J = 8$. Figure 4.3 shows these errors for variables

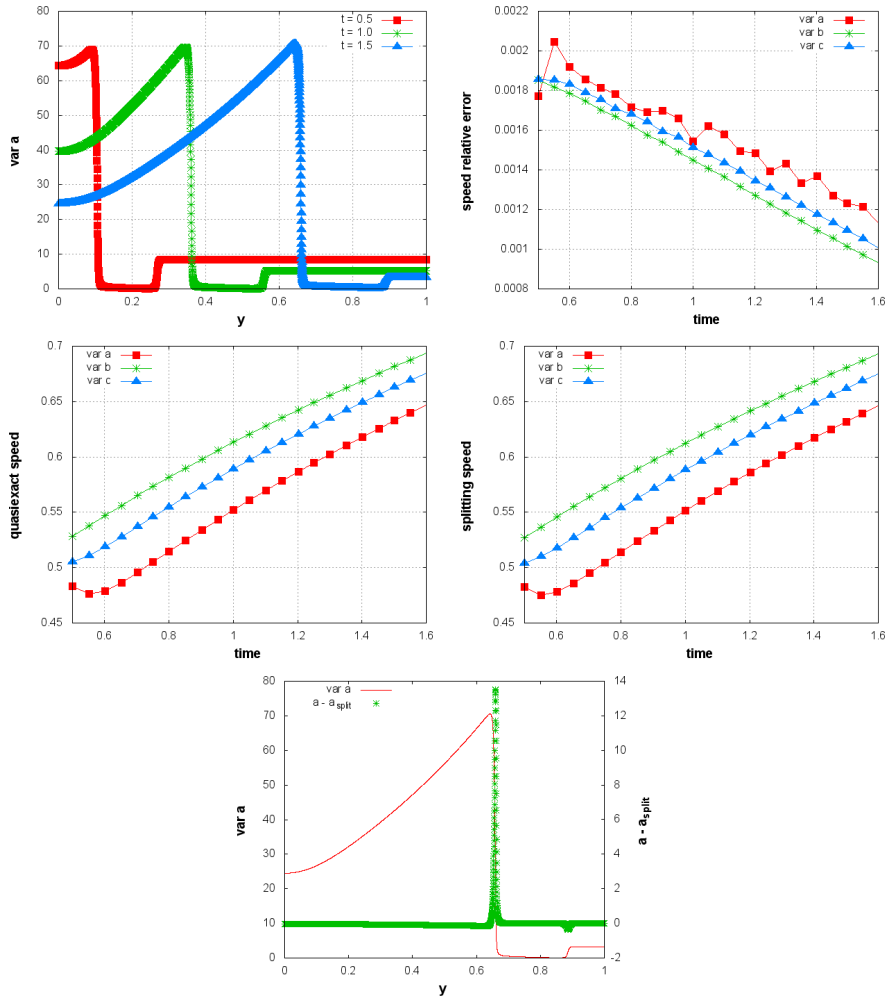

FIG. 4.2. *BZ wave speed for splitting time step* $\Delta t = 4/1024$. *Top: left, time evolution of variable* a *considering a slice along* y*-axis (see Figure 4.1); and right, speed relative errors* E_v *for all three variables according to (3.2). Center: time evolution of wave speeds computed by quasi-exact (left) and splitting (right) solvers. Bottom: Profile difference between variable* a *and* a_{split} *at* $t^\star = 1.5$.

a, b and c, and for different threshold values ε at final time $t = 4$. Notice that for $\varepsilon < 10^{-4}$ one should consider $\eta_{Radau5} = \eta_{ROCK4} < 10^{-5}$, in order to remain coherent throughout the error analysis and to guarantee an accurate resolution of reactive and diffusive time scales.

The splitting error $\|\mathbf{u}^J_{qe}(t) - \mathbf{u}^J_{split}(t)\|_{L^2}$ is computed on the finest grid and does not depend on the grid adaptation process: it is fixed by the splitting time step $\Delta t = 4/1024 \approx 3.9 \times 10^{-3}$. The error coming from the multiresolution process is given by $\|\mathbf{u}^J_{split}(t) - \mathbf{u}_{MR}(t)\|_{L^2} \propto \varepsilon$ according to (2.18), considering that both solutions use the same time integration strategy. In this case, these error estimates show that for $\varepsilon \leq 10^{-2}$, the multiresolution errors become negligible compared to the operator splitting ones. All of these numerical results are then used to set the various

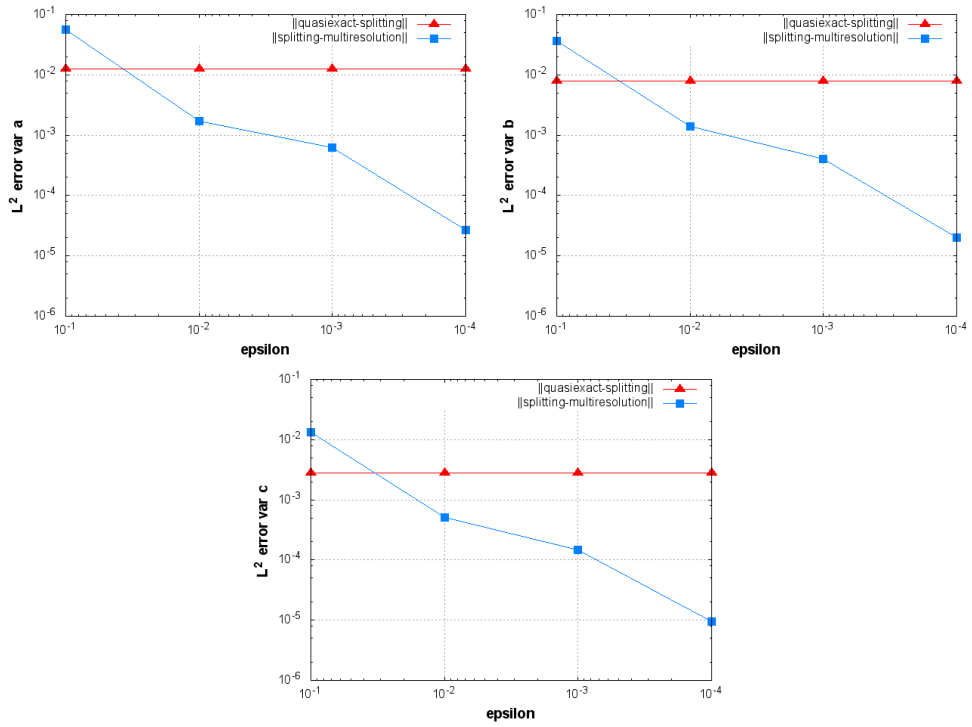

FIG. 4.3. *2D BZ. L^2 error at $t = 4$. Variables a (top left), b (top right) and c (bottom). Finest grid:* 256^2.

parameters for larger and more accurate simulations, as presented in the following.

We therefore consider the same problem but with finer spatial discretization on the finest grid J. Table 4.2 summarizes these results for a threshold value of $\varepsilon = 10^{-2}$. Data compression increases with the number of levels as the space scales present in the problem are better discriminated by finer spatial resolutions.

TABLE 4.2
2D BZ. Data compression (DC) and number of cells on the adapted grid (AG) for $\varepsilon = 10^{-2}$ and different finest grids (FG) and levels of refinement (J).

		$t = 2$		$t = 4$	
FG	*J*	*AG*	*DC*	*AG*	*DC*
128×128	7	6718	59.00	8008	51.12
256×256	8	15244	76.74	18874	71.20
512×512	9	33712	87.14	44917	82.87
1024×1024	10	80425	92.33	110428	89.47

In order to take into account the memory requirements of each resolution strategy for a fine spatial resolution of 1024×1024, we estimate the array size of the working space needed by Radau5 and ROCK4:

1. Radau5: $L_1 = 4 \times W_1 \times W_1 + 12 \times W_1 + 20$ (from [17]);
2. ROCK4: $L_2 = 8 \times W_2$ (from [1]);

where W_1 and W_2 are the number of unknowns solved by Radau5 and ROCK4. In the case of an uniform mesh, the total number of unknowns is $W = 3 \times 1024 \times 1024 \approx$

3.15×10^6 and thus, the global size L required for each solver is:

1. Quasi-exact: $W_1 = W \approx 3.15 \times 10^6$ and $L = L_1 \approx 4 \times 10^{13}$.
2. Splitting: $W_1 = 3$, $W_2 = W \approx 3.15 \times 10^6$ and $L = L_1 + L_2 \approx 2.5 \times 10^7$.
3. MR/Splitting with $\varepsilon = 10^{-2}$: $W_1 = 3$, $W_2 = 0.09 \times W \approx 2.9 \times 10^5$ and $L = L_1 + L_2 \approx 2.3 \times 10^6$; with an average data compression of 91%.

Considering a standard platform on which each double precision value is represented by 64 bits, we shall require 2.3 Pb, 23.8 Mb and 2.2 Mb respectively, for each solver.

Therefore, on the one hand, it is hopeless trying to solve problem (4.1) with the *quasi-exact* strategy for these very fine discretizations, at least with standard computing resources. For instance, for these 2D simulations we have used an Intel(R) Core(TM)2 processor of 2 GHz with memory capacity of 16 Gb. And on the other hand, also a splitting strategy becomes more difficult to implement since the diffusion term is solved considering the entire spatial domain at once. A major advantage of the proposed numerical strategy is the possibility of representing results in a highly compressed way. As an example, Figure 4.4 shows the adapted grids obtained for the 1024×1024 configuration with threshold value of $\varepsilon = 10^{-2}$.

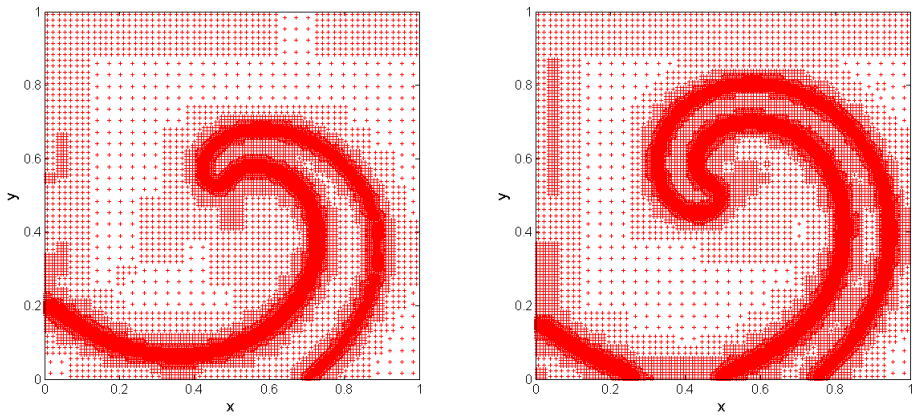

FIG. 4.4. *2D BZ. Adapted grid at $t = 2$ (left) and $t = 4$ (right), $\varepsilon = 10^{-2}$. Finest grid:* 1024^2.

4.3. 3D BZ Equation. We consider now problem (4.1) in a 3D configuration with the same parameters considered in the 2D case for a time domain of $[0, 2]$ and in a space region of $[0, 1] \times [0, 1] \times [0, 1]$. First, we take into account 8 nested dyadic grids with $\mathrm{N} = 2^{3\times 8} = 16777216 = 256 \times 256 \times 256$ cells on the finest grid $\mathrm{J} = 8$. Then, with a threshold value of $\varepsilon = 10^{-2}$ and a splitting time step $\Delta t = 4/1024 \approx 3.9 \times 10^{-3}$, the proposed numerical strategy features data compressions of 83.06% for the initial condition, 84.40% at $t = 1$ when the scroll waves are fully developed and 79.61% at final time $t = 2$. Figure 4.5 shows the evolution of the finest grid of the adapted grid during the period of study. We distinguish clearly the development of the scroll wave where colors represent the values of variable a: the finest regions correspond to the neighborhood of the wavefront. All of the 3D simulations were performed on an AMD-Shanghai processor of 2.7 GHz with memory capacity of 32 Gb.

In order to explore the feasibility and potential advantages of the method, let us consider 9 nested dyadic grids with $\mathrm{N} = 2^{3\times 9} = 134217728 = 512 \times 512 \times 512$ cells on the finest grid $\mathrm{J} = 9$. The initialization must take place on a intermediary grid,

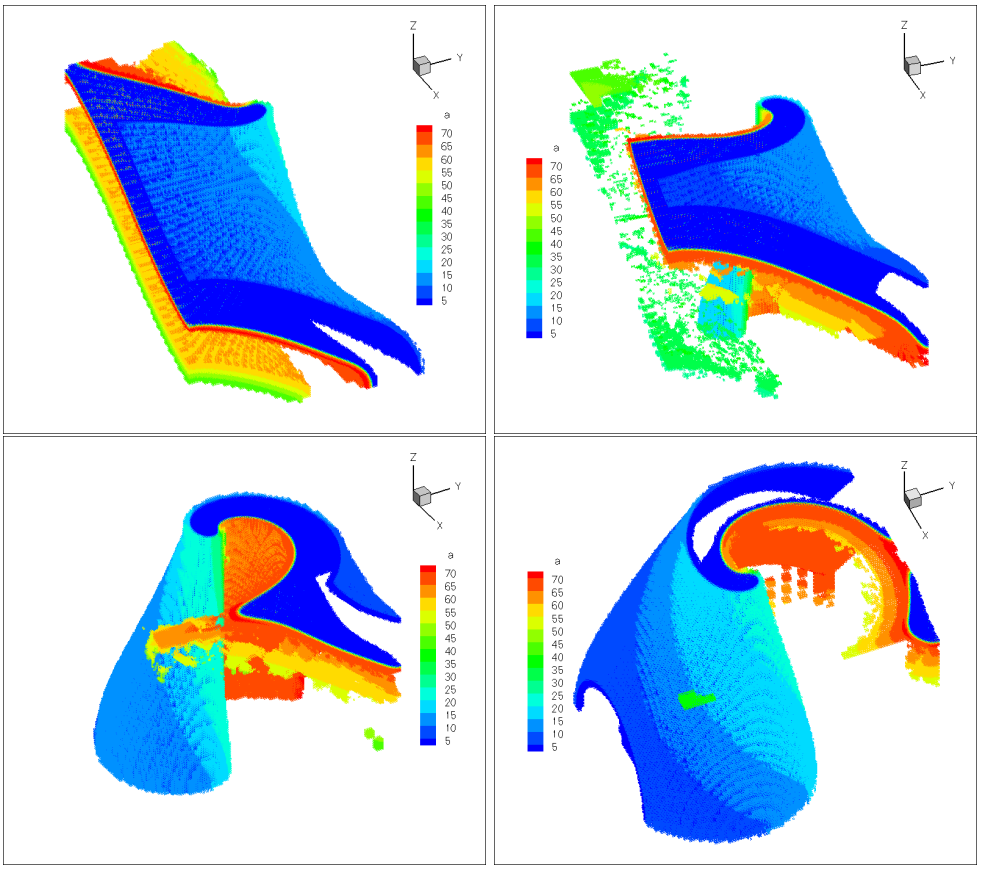

FIG. 4.5. *3D BZ scroll wave. Finest grid of the adapted grids at* $t = 0.5$ *(top left),* $t = 1$ *(top right),* $t = 1.5$ *(bottom left) and* $t = 2$ *(bottom right),* $\varepsilon = 10^{-2}$*, where colors represent values of variable* a*. Finest grid:* 256^3*.*

$j = 8$ in this example. For this configuration, a two times larger splitting time step of $\Delta t = 4/512 \approx 7.8 \times 10^{-3}$, and a threshold value of $\varepsilon = 10^{-1}$ were chosen in order to have splitting and multiresolution errors potentially of the same order. Smaller threshold values yield larger simulation domains which are not longer feasible with the considered computing resource and the current state of development of the code. Figure 4.6 shows the corresponding finest grids. Data compressions are now of 95.54% for the initial condition, 89.62% at $t = 1$ and 87.05% for final time $t = 2$.

Performing the same comparison concerning memory requirements, the total number of unknowns for this case is $W = 3 \times 512 \times 512 \times 512 \approx 4.03 \times 10^8$ and the global size of L required by each solver is:

1. Quasi-exact: $W_1 = W \approx 4.03 \times 10^8$ and $L = L_1 \approx 6.5 \times 10^{17}$.
2. Splitting: $W_1 = 3$, $W_2 = W \approx 4.03 \times 10^8$ and $L = L_1 + L_2 \approx 3.2 \times 10^9$.
3. MR/Splitting with $\varepsilon = 10^{-1}$: $W_1 = 3$, $W_2 = 0.13 \times W \approx 5.3 \times 10^7$ and $L = L_1 + L_2 \approx 4.2 \times 10^8$; with a data compression of 87%.

Therefore, we shall require at least 36.1 Eb, 190.7 Gb and 25.0 Gb of memory capacity, respectively for each solver.

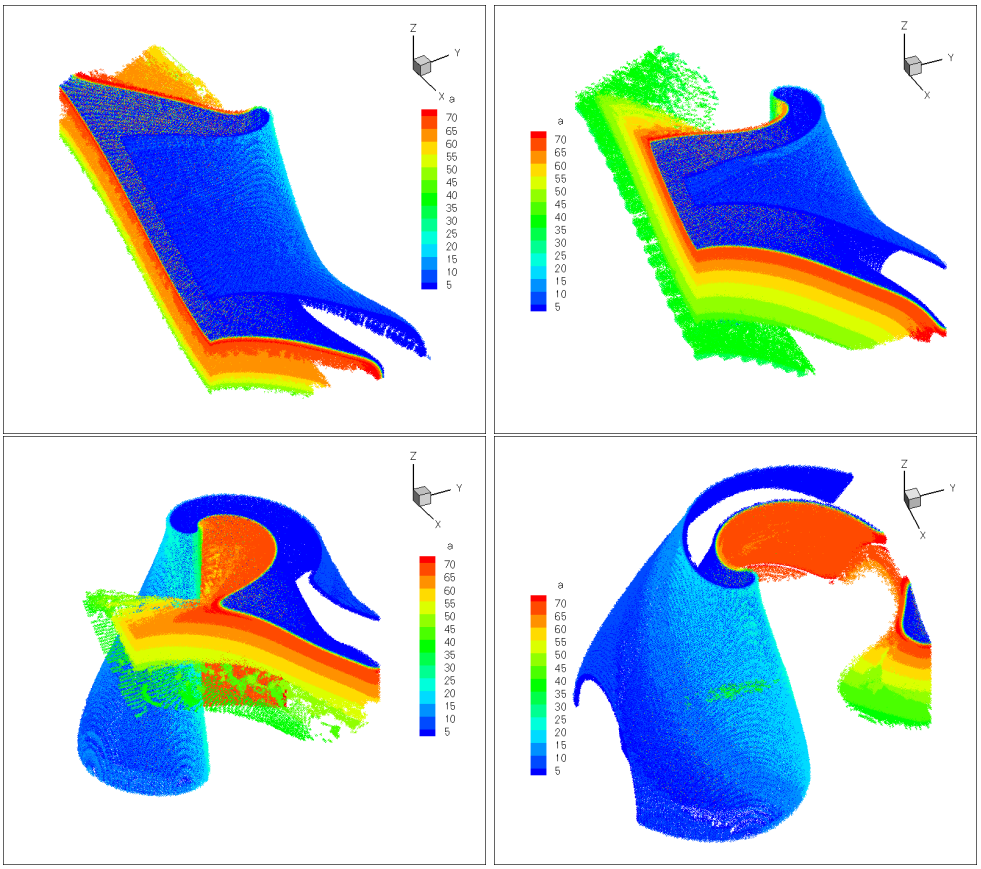

FIG. 4.6. *3D BZ scroll wave. Finest grid of the adapted grids at* $t = 0.5$ *(top left),* $t = 1$ *(top right),* $t = 1.5$ *(bottom left) and* $t = 2$ *(bottom right),* $\varepsilon = 10^{-1}$*, where colors represent values of variable* a*. Finest grid:* 512^3.

5. Concluding Remarks and Outlook. The present work proposes a new numerical approach which is shown to be computationally efficient. It couples adaptive multiresolution techniques with a new operator splitting strategy for multi-scale reactions waves modeled by stiff reaction-diffusion systems. The splitting time step is chosen on the sole basis of the structure of the continuous system and its decoupling capabilities, but not related to any stability requirement of the numerical methods involved in order to integrate each subsystem, even if strong stiffness is present. The technique considers, on the one hand, dedicated high order time integration methods to properly solve the entire spectrum of temporal scales of both the reaction and diffusion parts; and on the other hand, an adaptive multiresolution technique to represent and treat more accurately local spatial gradients associated to the wave front. The global accuracy of the simulation is evaluated based on theoretical and numerical results in the context of self-similar propagating waves. As a consequence, the resulting highly compressed data representations as well as the accurate and feasible resolution of these stiff phenomena prove that large computational domains previously out of reach can be successfully simulated with conventional computing resources (typically of the order of a couple of days of simulation on a single workstation). We claim that

we thus have reached some sort of optimality of the splitting strategy.

For the moment, we have focused our attention on reaction-diffusion systems in order to settle the foundations for simulation of more complex phenomena with fully convection-reaction-diffusion systems and more detailed models such as combustion with complex chemistry. The main difference of the method proposed here compared to what has been done in the combustion community where other numerical techniques have already been proposed (see for instance [8, 20]), is related to the fact that we provide an error control in both space and time of the solution, once the splitting time step for the continuous system of PDEs is defined in order to respect a given tolerance compared to the coupled solution. So far, since we have investigated reaction waves, this time step is evaluated once for all and does not need to be re-evaluated dynamically during the simulation.

However, an important amount of work is still in progress concerning on the one hand, programming features such as data structures, optimized routines and parallelization strategies. For instance, some interesting investigations have recently addressed this issue very nicely [3, 4]. And on the other hand, numerical analysis of theoretical aspects, which may surely lead to better error estimates as well as the ability to dynamically evaluate the splitting time step for more general unsteady solutions of reaction-diffusion and convection-reaction-diffusion systems of equation, to extend and further improve the proposed numerical strategy. Finally, when dealing with more complex systems such as complex chemistry or stroke modeling in the brain, the source term involves many species (typically 50) and many reactions (typically several hundreds) or complex mechanisms. In such a case the integration of the source term leads to a heavy computational cost, even if it is embarrassingly parallel in the framework of operator splitting and if the multiresolution allows to optimize its resolution as shown in [13]. Therefore, this field requires some further studies in order to obtain high efficiency in terms of load balancing on parallel architectures. These issues constitute particular topics of our current research.

REFERENCES

[1] A. Abdulle, *Fourth order Chebyshev methods with recurrence relation*, Society for Industrial and Applied Mathematics J. Sci. Comput., 23 (2002), pp. 2041–2054.

[2] B. L. Bihari and A. Harten, *Multiresolution schemes for the numerical solution of 2-D conservation laws I*, SIAM J. Sci. Comput., 18 (1997), pp. 315–354.

[3] K. Brix, R. Massjung, and A. Voss, *A hash data structure for adaptive PDE-solvers based on discontinuous Galerkin discretizations*, IGPM-Rep. 302, RWTH Aachen, (2009).

[4] K. Brix, S. Melian, S. Müller, and G. Schieffer, *Parallelisation of multiscale-based grid adaptation using space-filling curves*, ESAIM: Proc., 29 (2009), pp. 108–129.

[5] A. Cohen, *Wavelet methods in numerical analysis*, vol. 7, Elsevier, Amsterdam, 2000.

[6] A. Cohen, S.M. Kaber, S. Müller, and M. Postel, *Fully adaptive multiresolution finite volume schemes for conservation laws*, Mathematics of Comp., 72 (2003), pp. 183–225.

[7] Y. D'Angelo, *Analyse et Simulation Numérique de Phénomènes liés à la Combustion Supersonique*, PhD thesis, Ecole Nationale des Ponts et Chaussées, 1994.

[8] M. S. Day and J. B. Bell, *Numerical simulation of laminar reacting flows with complex chemistry*, Combust. Theory Modelling, 4 (2000), pp. 535–556.

[9] S. Descombes, T. Dumont, V. Louvet, and M. Massot, *On the local and global errors of splitting approximations of reaction-diffusion equations with high spatial gradients*, Int. J. of Computer Mathematics, 84 (2007), pp. 749–765.

[10] S. Descombes, T. Dumont, V. Louvet, M. Massot, F. Laurent, and J. Beaulaurier, *Operator splitting techniques for multi-scale reacting waves and application to low Mach number flames with complex chemistry: Theoretical and numerical aspects*, Submitted to SIAM, available on HAL, (2010).

[11] S. Descombes, T. Dumont, and M. Massot, *Operator splitting for stiff nonlinear reaction-*

diffusion systems: Order reduction and application to spiral waves, in Patterns and waves (Saint Petersburg, 2002), AkademPrint, St. Petersburg, 2003, pp. 386–482.

[12] S. Descombes and M. Massot, *Operator splitting for nonlinear reaction-diffusion systems with an entropic structure: Singular perturbation and order reduction*, Numer. Math., 97 (2004), pp. 667–698.

[13] T. Dumont, M. Duarte, S. Descombes, M.A. Dronne, M. Massot, and V. Louvet, *Simulation of human ischemic stroke in realistic 3D geometry: A numerical strategy*, Submitted to Bulletin of Mathematical Biology, available on HAL, (2010).

[14] I.R. Epstein and J.A. Pojman, *An Introduction to Nonlinear Chemical Dynamics*, Oxford University Press, 1998. Oscillations, Waves, Patterns and Chaos.

[15] P. Gray and S. K. Scott, *Chemical oscillations and instabilites*, Oxford Univ. Press, 1994.

[16] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, Springer-Verlag, Berlin, 2nd ed., 2006. Structure-Preserving Algorithms for Odinary Differential Equations.

[17] E. Hairer and G. Wanner, *Solving ordinary differential equations II*, Springer-Verlag, Berlin, second ed., 1996. Stiff and differential-algebraic problems.

[18] A. Harten, *Multiresolution algorithms for the numerical solution of hyperbolic conservation laws*, Comm. Pure and Applied Math., 48 (1995), pp. 1305–1342.

[19] S. Müller, *Adaptive multiscale schemes for conservation laws*, vol. 27, Springer-Verlag, 2003.

[20] H. N. Najm and O. M. Knio, *Modeling low Mach number reacting flow with detailed chemistry and transport*, Journal of Scientific Computing, 25 (2005), pp. 263–287.

[21] O. Roussel, K. Schneider, A. Tsigulin, and H. Bockhorn, *A conservative fully adaptive multiresolution algorithm for parabolic PDEs*, J. Comput. Phys., 188 (2003), pp. 493–523.

[22] B. Sandstede, A. Scheel, and C. Wulff, *Bifurcations and dynamics of spiral waves*, Journal of Nonlinear Science, 9 (1999), pp. 439–478.

[23] L. F. Shampine, B. P. Sommeijer, and J. G. Verwer, *IRKC: An IMEX solver for stiff diffusion-reaction PDEs*, J. Comput. Appl. Math., 196 (2006), pp. 485–497.

[24] B. Sportisse, *Contribution à la modélisation des écoulements réactifs: Réduction des modèles de cinétique chimique et simulation de la pollution atmosphérique*, PhD thesis, Ecole Polytechnique, 1999.

[25] ———, *An analysis of operator splitting techniques in the stiff case*, J. Comput. Phys., 161 (2000), pp. 140–168.

[26] G. Strang, *Accurate partial difference methods. I. Linear Cauchy problems*, Arch. Ration. Mech. Anal., 12 (1963), pp. 392–402.

[27] ———, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal., 5 (1968), pp. 506–517.

[28] J. G. Verwer, *Explicit Runge-Kutta methods for parabolic partial differential equations*, Appl. Numer. Math., 22 (1996), pp. 359–379.

[29] J. G. Verwer, B. P. Sommeijer, and W. Hundsdorfer, *RKC time-stepping for advection-diffusion-reaction problems*, J. Comput. Phys., 201 (2004), pp. 61–79.

[30] J. G. Verwer and B. Sportisse, *Note on operator splitting in a stiff linear case*, Rep. MAS-R9830, (1998).