HAL
open science

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

To cite this version:

Max Duarte, Marc Massot, Stéphane Descombes, Christian Tenaud, Thierry Dumont, et al.. New resolution strategy for multi-scale reaction waves using time operator splitting, space adaptive multiresolution and dedicated high order implicit/explicit time integrators. SIAM Journal on Scientific Computing, 2011, pp.1-29. hal-00457731v3

HAL Id: hal-00457731

https://hal.science/hal-00457731v3

Submitted on 31 Jul 2011 (v3), last revised 30 Sep 2011 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

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 the fastest scales in the source term nor by stability contraints of the diffusive steps, but only by the physics of the phenomenon. We aim thus at solving complete models including all time and space scales within a prescribed 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

*This research was supported by a fundamental project grant from ANR (French National Research Agency - ANR Blancs) *Séchelles* (project leader S. Descombes - 2009-2013), by a CNRS PEPS Maths-ST2I project *MIPAC* (project leader V. Louvet - 2009-2010), and by a DIGITEO RTRA project *MUSE* (project leader M. Massot - 2010-2014).

[†]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 and by INCA project (CNRS/ONERA/SAFRAN).

[§]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).

problems. 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 [8] 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 high 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 [34, 28]. These methods are usually very efficient but the feasibility of utilizing dedicated implicit solvers over a discretized domain becomes soon critical when treating large computational domains. 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. We know though that these fastest time scales do not always play a leading role in the global physics of many multi-scale problems and therefore 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 provide only approximate solutions and are usually accessible when the system is well-partitioned and the fastest scales can be identified or isolated [29], 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 usually 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 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. In order to guarantee the accuracy of the solution obtained by a splitting scheme, the splitting time steps used for the independent resolution of each subproblem are usually taken of the order of the fastest scales included in the problem. As a matter of fact, several works [35, 30, 8] showed that the standard numerical analysis of splitting schemes fails in presence of scales much faster than the splitting time step. Nevertheless, more rigorous studies for these stiff configurations [14, 12] and in the case where spatial multi-scale phenomena arise as a consequence of steep spatial gradients [11], allow to characterize the behavior of splitting schemes with splitting time steps much larger than the fastest scales of the problem.

As a consequence, we introduce in this work a new time operator splitting approach for which the dedicated methods chosen for each subsystem are responsible for dealing with the fast scales associated with each one of them, in a separate manner. The global solution is then reconstructed by the splitting scheme with splitting time steps dictated by the global physical coupling, possibly much larger than the fastest

time scales. In this way, the splitting time step is chosen based only on error estimates of the numerical simulation in order to guarantee the description of the physics of the phenomenon within a prescribed accuracy, without being related to the stability constraints of the numerical resolution of each subsystem and with an important improvement of efficiency whenever a broad decoupling of the time scale spectrum is possible.

The operator splitting strategy proposed in this article considers then a high order method like Radau5 [22], 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. In this way, the global accuracy of the time integration scheme is mainly set by the splitting scheme through the choice of the splitting time step. This numerical strategy is then complemented by a mesh refinement technique based on Harten's pioneering work on adaptive multiresolution methods [23], being aware of the interest of adaptive mesh techniques for problems exhibiting locally steep spatial gradients. Since a multiresolution technique allows to better control the accuracy of the adapted and compressed spatial representation, both the space and time errors can be regulated for a given semi-discretized problem. 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. In particular, in the case of propagating wavefronts, we show that a constant splitting time step is good enough to capture the dynamics of the phenomenon even though an adaptive splitting time step technique as the one developed in [10] can be used to compute the appropriate splitting time step whenever the determination of the latter becomes difficult to conduct.

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 [6, 25], which is briefly presented. The implementation of the numerical strategy is detailed in section 3, as well as the algorithm scheme and the choice of the appropriate splitting time step. 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, 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 some 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 $d \times d \times m$.

Even though the proposed numerical strategy handles general problem (2.1), in order to simplify the presentation we shall consider problem (2.1) with linear diagonal diffusion. In this case the elements of the diffusion matrix are written as $D_{i_1 i_2 i_3}(\mathbf{u}) = D_{i_3} \delta_{i_1 i_2}$ for some positive indices i_1, i_2, i_3, so that the diffusion operator reduces to the heat operator with scalar diffusion coefficient D_{i_3} for component u_{i_3} of $\mathbf{u}$, $i_3 = 1, \ldots, m$. Performing a fine spatial discretization, we obtain the semi-discretized initial value problem

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

where $\mathbf{B}$ corresponds to the discretization of the Laplacian operator with the coefficients D_{i_3} within. $\mathbf{U}$ and $\mathbf{F}\left(\mathbf{U}\right)$ 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

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

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,

$$\mathrm{d}_t \mathbf{U}_R = \mathbf{F}\left(\mathbf{U}_R\right), \quad t > 0, \tag{2.4}$$

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 [31, 32] 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 [21]). The last ones imply usually 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 [14]). 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 small enough to solve 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 [14] 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 [12]). 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 fast 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 [11].

These theoretical studies allow to depict more precisely the numerical behavior of the 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 estimates but also to take advantage of the particular features of each independent subproblem in order to solve them with reasonable resources as accurately as possible. In particular, our splitting technique considers high order dedicated integration methods for each subproblem in order to properly solve the fastest time scales associated with each one of them, and in such a way that the main source of error is led by the operator splitting error. Then, the control of the accuracy of the time integration is ruled by the splitting scheme by means of the splitting time step needed to describe the global physical phenomenon within a required level of accuracy.

2.2.1. Time Integration of the Reaction: Radau5. Radau5 [22] is a fifth order implicit Runge-Kutta method for which order conditions proven by Butcher [5] are satisfied up to the order 5, whereas its stability function is generated by a collocation procedure with Radau quadrature formulae [17] that guarantees A- and L-stability properties, so that very stiff systems of ODEs might be solved without any stability problem.

Nevertheless, nonlinear systems must be solved throughout the time integration process because of this implicit character. Even if the system resolution 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 strongly 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 usually does not exceed conventional memory resources. Even more, this approach is posed as an embarrassingly parallel

problem where no data exchange is needed among nodes, that therefore yields optimal load balancing on shared memory architectures (see for example the numerical implementations achieved in [16]).

A very important feature of the Radau5 solver is that precious computing time is saved because it considers an adaptive time stepping strategy which guarantees a requested accuracy of the numerical integration and at the same time, allows to discriminate stiff zones from regular ones so that small time steps are only considered for stiff behaviors. In a splitting context, the reaction time integration step Δt_R will be adapted only at nodes where the reaction phenomenon takes place, yielding local reaction time steps much smaller than the splitting time step so that the global time advancement of the resolution given by the splitting time step Δt will not be limited by these fast time scales. For multi-scale reaction waves, this adaptation happens in a very low percentage of the spatial domain, usually only in the neighborhood of the wavefront. On the other hand, larger time steps are considered at nodes with a chemistry at (partial) equilibrium. This local time stepping without data exchange nor reconstruction of intermediary values would not be possible if we integrated the entire reaction-diffusion system (2.2) at once.

2.2.2. Time Integration of the Diffusion: ROCK4. One of the most important advantages of ROCK4 [1] 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 within one time integration step of the diffusion problem, Δt_D, inside each splitting time 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.

The ROCK solver [1] features also dynamic time step adaptation so that Δt_D is chosen in order to guarantee a prescribed accuracy of computations. ROCK4 is formally a fourth order *stabilized* explicit Runge-Kutta method and such methods feature extended stability domain along the negative real axis [33]. Therefore, considering a general problem such as $\mathbf{v}' = \mathbf{g}(\mathbf{v})$, in order to guarantee the stability of computations for a given time step Δt_D, the number of stages s needed is directly related to the spectral radius $\rho(\partial \mathbf{g}/\partial \mathbf{v})$, as long as the latter is dominated by real negative eigenvalues. Hence, for a given Δt_D needed to guarantee the accuracy of the integration, the minimum number of stages s needed to guarantee the stability is computed by the ROCK4 solver through

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

which extends quadratically on s the stability domain of the method along the negative axis. For an explicit Runge-Kutta method, the order 4 implies at least $s = 5$ internal stages.

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 $\rho(\mathbf{B})$. An important gain of efficiency is obtained in this case because the discretized diffusion operator has a sparse matrix

structure that yields more performing matrix-vector products. 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 the time operator splitting. In particular, solving within a prescribed tolerance the diffusion problem may also yield Δt_D much smaller than the splitting time step Δt, and thus, the global time advancement of the solution given by Δt will not be necessarily limited by the diffusive time scales but by the global features of the coupled problem.

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 the reactive problem (2.4), a considerable amount of computing time is spent on nodes that are practically at (partial) equilibrium. We show for instance in [16] that for a numerical simulation with complex source mechanisms on a uniform grid, 60 % of the computing time is spent on nodes with very reduced chemical activity. 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 [6] and Müller [25].

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,

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

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* [6]. 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 at least two basic constraints are usually imposed [7]:

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

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

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 [23, 7].

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

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

or in terms of inter-level operations:

$$d_\mu = u_\mu - P_{|\mu|}^{|\mu|-1} \circ P_{|\mu|-1}^{|\mu|} u_\mu. \tag{2.13}$$

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

$$\sum_{|\mu|=|\gamma|+1, \Omega_\mu \subset \Omega_\gamma} |\Omega_\mu| d_\mu = 0. \tag{2.14}$$

We can then construct as shown in [7] a *detail vector* defined 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:

$$\mathbf{U}_j \longleftrightarrow (\mathbf{U}_{j-1}, \mathbf{D}_j), \tag{2.15}$$

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 $\mathbf{M}_J = (\mathbf{U}_0, \mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_J)$:

$$\mathcal{M} : \mathbf{U}_J \longmapsto \mathbf{M}_J. \tag{2.16}$$

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:

$$\lambda \in \Lambda \text{ if } |d_\lambda| \geq \varepsilon_j, \quad j = |\lambda|, \tag{2.17}$$

with the level-dependent threshold values:

$$\varepsilon_j = 2^{\frac{d}{2}(j-J)}\varepsilon, \quad j = |\lambda|, \quad j \in [0, J], \tag{2.18}$$

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 (2.17), whereas the conservativity of the multiresolution scheme remains guaranteed by expression (2.14) as it can be shown in [25].

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 with 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 [7, 27] 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 related to a tree representation that we will adopt throughout this paper:

- If $\Omega_\mu \subset \Omega_\lambda$ with $|\mu| = |\lambda| + 1$, 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 (2.17). More precisely, the numerical solution $\mathbf{U}^n$ at time $n\Delta t$ can be represented on an adapted grid by the set $(u^n_\lambda)_{\lambda \in \mathrm{L}(\Lambda^n_\varepsilon)}$.

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 $\tilde{\Lambda}^{n+1}_\varepsilon$ containing Λ^n_ε, on which the time integration is computed, such that $\tilde{\Lambda}^{n+1}_\varepsilon$ 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 $\lambda \in \mathrm{L}(\Lambda^n_\varepsilon)$ such that $|d_\lambda| \geq \varepsilon_{|\lambda|}$, which adds one more level all over Λ^n_ε so that the finest cells do not verify the threshold criterion (2.17); and according to Harten's heuristics (see [23]), by enlarging uniformly with $\bar{k}$ cells in each direction the refined region of Λ^n_ε in order to predict the propagation of the solution. The u^n_λ with $\lambda \in \tilde{\Lambda}^{n+1}_\varepsilon \backslash \Lambda^n_\varepsilon$ can be constructed by applying the inverse operation of (2.16): $\mathcal{M}^{-1}$. These criteria are rather conservative, nevertheless they completely avoid unrefined resolution taking into account the propagating nature of reaction waves at finite speed. For more general cases, a more sophisticated refinement criteria might be required, as the one introduced in [7], that allow to add more than one refined level during one time step in order to properly describe the forthcoming solution. On the other hand, a dynamic time step adaptation strategy for highly unsteady problems as developed in [15], shows that a high frequency of remeshing given by the adaptive time step allows also to rapidly capture the spatial dynamics of the problem within a prescribed tolerance of the adapted spatial representation.

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

the compressed spatial representation is given by:

$$\|\mathbf{U}_J^n - \mathbf{V}_J^n\| \propto n\varepsilon. \tag{2.19}$$

This result was first stated by Harten in [23] for hyperbolic problems in a L_1-norm and then, it has been mathematically proven in [7] under more rigorous constraints for refinement criteria and computation of the fluxes. In the same spirit of these works, in this study we will consider (2.19) but with a L^2-norm, more suited for parabolic problems, as it was also previously considered in [27]. Even though a fully mathematical proof is still required for this case, a numerical demonstration will be provided. Finally, the latest error estimate of the compressed spatial representation as well as the numerical behavior of the time operator splitting schemes, allow one to properly choose the simulation parameters in order to predict the expected level of accuracy of the resolution.

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

$$(u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde{\Lambda}_\varepsilon^n)} \xrightarrow{\mathcal{M}} (u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde{\Lambda}_\varepsilon^n} \tag{2.20}$$

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

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

$$(u_{\lambda_0}^n, d_\lambda^n)_{\lambda\in\tilde{\Lambda}_\varepsilon^{n+1}} \xrightarrow{\mathcal{M}^{-1}} (u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde{\Lambda}_\varepsilon^{n+1})} \tag{2.23}$$

$$(u_\lambda^n)_{\lambda\in \mathrm{L}(\tilde{\Lambda}_\varepsilon^{n+1})} \xrightarrow{\mathcal{S}^{\Delta t}} (u_\lambda^{n+1})_{\lambda\in \mathrm{L}(\tilde{\Lambda}_\varepsilon^{n+1})} \tag{2.24}$$

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 (2.20), which can usually 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 $\lambda \in S_j$, then the solution is refined and recomputed over the next finer level after a thresholding process. This procedure is recursively applied until the pre-established finest level J is reached; the general procedure is explained in details in [25].

The algorithm can schematically be represented as

$$\mathbf{U}^{n+1} = \mathcal{S}^{\Delta t}(\mathcal{M}^{-1}\mathcal{R}\mathcal{T}_{\Lambda_\varepsilon^n}\mathcal{M}\mathbf{U}^n), \tag{2.25}$$

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 evolution operator. One might add a last thresholding step to represent 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. This kind of data structure has been used in other multiresolution applications [27] and other dedicated

data structures have also been developed [3]. 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, whereas 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.

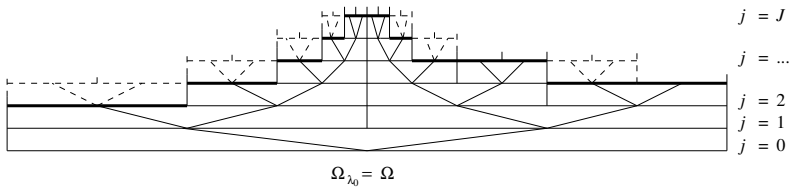

FIG. 3.1. *Example of 1D graded tree structure. Nodes and links to their corresponding children are indicated (solid lines) as well as the leaves (solid bold lines) and the phantoms (dashed lines).*

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, although there is no rigorous mathematical analysis, it was shown in [7, 25] that a much more efficient alternative to fully evaluate the fluxes at the finest level considers the flux evaluations directly on the adapted mesh with essentially the same accuracy. Nevertheless, we consider in this work the intermediary case, also detailed in [25] and [27], that takes into account virtual cells called *phantoms* that are locally 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 [27] 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 with 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 a 1D configuration for $l = 1$, the prediction operator 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. Higher order formula can be found in [25]. Since a Cartesian mesh is used, extension to multidimensional polynomial interpolations is easily obtained by a tensorial product of the 1D operator [2, 27]. More accurate interpolations (higher r) yield better predicted values and thus, lower values of *details* following (2.12). Therefore, the thresholding process should also yield higher data compression according to (2.17). Nevertheless, this is not necessarily the case because larger stencils (higher l) and consequently more cells are needed to perform these interpolations so that the optimal choice of these parameters depends finally on the application. In the numerical illustrations we will restrict the different analysis to $l = 1$, unless otherwise noted.

3.2. Choice of the 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 is perfectly solved. It is thus only depending on the global physics of 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. Considering the semi-discretized problem (2.2) with a sufficiently fine spatial discretization, if an accurate reference wave solution $\mathbf{U}$ 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 a splitting time step Δt and an operator splitting technique with exact integration of the subsystems, must verify:

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

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 $t^\star$:

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

in which case, both profile and velocity errors are simultaneously considered. All these norms are normalized. 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:

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

In this way, we can isolate the various integration errors and guarantee that the overall time integration error is practically given by the splitting scheme approximation. In particular, an evaluation of the sole splitting error allows to decouple the time scale spectrum of the problem whenever this is possible, contrary to classical applications of time operator splitting methods for which the splitting time step is directly settled by

the fastest time scale of the phenomenon. In our strategy, we therefore extend the use of these methods to splitting time steps given by the global coupling scales, potentially larger than the fastest physical or numerical scales. This is a direct consequence of previous mathematical studies [14, 11] that demonstrate that even though there will possibly be a loss of order of these methods for time or space stiff problems and large splitting time steps, the splitting schemes will still consistently approximate the coupled resolution with an error piloted by the splitting time step.

We have established so far the criteria to handle time integration errors given by the splitting procedure. Nevertheless, the proposed strategy combines this splitting approach with a space multiresolution adaptive technique so that the approximation error introduced by the latter must be also taken into account. According to (2.19) and even though a rigorous mathematical demonstration is not yet available for parabolic problems, we consider as in previous studies [27] the following error bound for a fixed time $T = n\Delta t$:

$$\left\|\mathbf{U}^{MR}_{split} - \mathbf{U}^{J}_{split}\right\|_{L^2} \leq Cn\varepsilon, \tag{3.5}$$

for some positive C, where $\mathbf{U}^{MR}_{split}$ is the MR/splitting solution at $n\Delta t$ reconstructed on the finest grid J, which corresponds to the spatial discretization of the semi-discretized problem (2.2). A basic constraint to assume the validity of (3.5), as in the hyperbolic case, is that the propagating locally refined spatial gradients remain into the corresponding fine regions during each time step evolution Δt. In order to guarantee this for a given spatial discretization $(\Delta x, \Delta y, \Delta z)$ corresponding to the maximum J level, the splitting time step must be bounded by a maximum splitting time step $\Delta t_{\max}$ computed by:

$$\Delta t \leq \Delta t_{\max} = \min\left\{\frac{\hat{k}\Delta x}{v_x}, \frac{\hat{k}\Delta y}{v_y}, \frac{\hat{k}\Delta z}{v_z}\right\}, \tag{3.6}$$

for the directional components (v_x, v_y, v_z) of the wavefront speed and where $\hat{k} \geq 2 + \bar{k}$ stands for the refined region obtained with the refinement criteria detailed in § 2.3.2 for which one finer level is added everywhere (2 cells at J in each direction) and $\bar{k}$ cells on the same level.

In the case of propagating wavefronts, a constant splitting time step based on a prescribed accuracy of the global time integration process as described in this section is more than reasonable, whereas the bound (3.6) guarantees a proper coupling of the space and time numerical methods. If no theoretical hints of the wave profile or velocity exist, the computation of a reference solution is usually very expensive but still feasible for one-dimensional or relatively small computational domains, which might give some insights in the behavior of the numerical methods in order to extrapolate these results to larger or multi-dimensional problems according to a standard numerical procedure. On the other hand, the speed of the wavefront needed to compute (3.6) can be always approximated by one-dimensional measurements taken either from fully coupled 1D configurations or for instance from feasible multi-dimensional MR/splitting resolutions for which the accuracy tolerances are tightened.

However, either if a more precise information on the accuracy of the time integration to choose the splitting time step is required, or if we are faced with highly unsteady problems, an adaptive splitting scheme introduced in [10] allows to properly estimate the local error of the splitting time integration without any need to compute a reference solution. Hence, based on these local error estimates, the splitting time

steps are dynamically computed within a prescribed accuracy tolerance $\eta_{\Delta t}$. Nevertheless, this procedure introduces naturally an overhead which might not be justified in the simplified case of propagating waves so that a hybrid strategy that allows to consider a constant splitting time step issued from the local error estimates of the adaptive splitting procedure can be seen as the most convenient solution. In this work we are focused on the propagation of reaction waves, therefore the previously mentioned adaptive splitting scheme [10] is partially used only to justify and validate the chosen constant splitting time step and the proposed numerical strategy for this kind of problems.

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

1. **INITIALIZATION:**
   - *Initialization of parameters: e.g.* maximum and minimum grid levels, 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;
     - definition of *children* and *neighbors* from the *roots*.
   - *Computation of initial solution* at an intermediary grid level and recursive refinement and computation up to the maximum 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:* operator $\mathcal{T}_{\Lambda_\varepsilon}$ throughout the *tree*.
   - *Refinement and graduation 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 considering *phantoms* cells at grid level boundaries, computed by prediction operator;
     - 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. All simulations were performed on an AMD-Shanghai processor of 2.7 GHz with memory capacity of 32 GB.

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 [18] for more details and illustrations). We thus consider the model detailed in [20] and coming from the classic work of [19] 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:

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

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 [20] 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 this problem have already been studied and presented in [13]. 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 [24] and a preliminary study [13]: $\epsilon = 10^{-2}$, $\mu = 10^{-5}$, $f = 1.6$ 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 $[0, 1]^2$. The initialization of the problem is based on [24] for the two-variable model of (4.1) with b and c, and it is given by

$$
a = \frac{fc}{q+b}, \tag{4.2}
$$

$$
\begin{aligned}
b &= 0.8, \quad \text{if } 0 < \theta < \arctan(0.3), \\
b &= b_{ss}, \quad \text{elsewhere},
\end{aligned}
\tag{4.3}
$$

$$
c = c_{ss} + \frac{\theta}{8\pi f}, \tag{4.4}
$$

where θ is a polar coordinate angle considering as origin $(0.5, 0.5)$ into $[0, 1]^2$.

$$
b_{ss} = c_{ss} = q\,\frac{f+1}{f-1} \tag{4.5}
$$

is an approximation of the steady state values of the dynamical system associated with the two-variable problem obtained by taking (4.2) into the evolution equations of b and c. (4.2) is obtained by taking $\mu \to 0$ into the evolution equation of a.

In the following, we will first consider a 2D computational domain with a uniform mesh of 256^2 for which the coupled and split resolutions of the semi-discretized problem derived from (4.1) are rather expensive but still feasible. The idea is to compare these solutions with the ones computed by the proposed MR/splitting procedure in order to analyze the splitting and multiresolution errors regarding the corresponding accuracy tolerances detailed in § 3.2 and to evaluate the computational performance of each approach. In a second step and based on these results we will study larger computational domains and 3D problems.

We thus consider 8 nested dyadic grids with $N = 2^{2\times 8} = 65536 = 256^2$ cells on the finest grid $J = 8$ and define a reference or *quasi-exact* solution $\mathbf{U}^J_{qe}$ as the

solution of the semi-discretized coupled reaction-diffusion problem (4.1) on the finest mesh J performed by ROCK4 with very fine tolerances, $\eta_{ROCK4} = 10^{-14}$. For this problem and with the previous parameters, the spectral radius of the Jacobian of the reaction term into (4.1) is usually dominated by the negative real parts of the associated eigenvalues even though the imaginary parts are also present. Therefore only fine tolerances that yield sufficiently small time steps allow to fully guarantee the stability of the ROCK4 scheme whenever the imaginary part appears, considering the reduced stability domains of these methods along the imaginary axis. The main limitation to directly perform such computation with the Radau5 solver comes from its important memory requirements.

The split solution $\mathbf{U}^J_{split}$ uses the RDR Strang $\mathcal{S}_2^{\Delta t}$ scheme as time integration method of the semi-discretized problem (4.1) with Radau5 for the time integration of the reaction term and ROCK4 for the diffusive part, $\eta_{Radau5} = \eta_{ROCK4} = 10^{-5}$. In order to choose the appropriate splitting time step Δt, we set an accuracy tolerance of $\eta = 10^{-2}$ considering for simplicity the normalized L^2-errors (3.3) as explained in § 3.2. Figure 4.1 shows these errors evaluated at final time $t^\star = 4$ for all three variables. A rounded value of splitting time step of $\Delta t = 4/1024 \approx 3.91 \times 10^{-3}$ is finally chosen for which L^2-errors are close to η for all three variables and times $t^\star \in [0, 4]$ into (3.3). In this work, we have computed several split solutions with different time steps for the whole time domain in order to analyze the different aspects of the scheme. Nevertheless, in a practical situation and for the simulation of propagating waves, we can consider a much less expensive procedure that evaluates the local errors after one splitting time step starting from an intermediary solution for which the waves are fully developed.

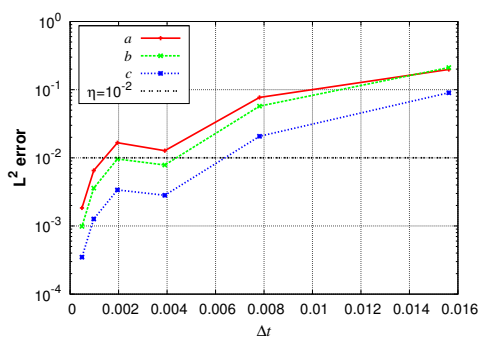

FIG. 4.1. *2D BZ spiral waves. Normalized L^2-errors for several splitting time steps Δt at final time $t^\star = 4$ according to (3.3) and prefixed accuracy tolerance of $\eta = 10^{-2}$. Uniform grid of 256^2.*

A rather large accuracy tolerance η was considered in order to show the decoupling of time steps for reaction, diffusion and the time operator splitting needed to solve the problem within the prescribed tolerance. The imposed tolerances for the reaction resolution imply time steps varying from 8.88×10^{-5} to $\Delta t/2 \approx 1.95 \times 10^{-3}$ for points located respectively in the neighborhood of the reactive front and the reduced chemical activity regions. On the other hand, the selected tolerances for ROCK4 yield time steps Δt_D relatively constant of about $6.5 - 8 \times 10^{-4}$, that is $5 - 6$ diffusion time steps within each splitting time step Δt. For the spatial discretization of 256^2, the spectral radius $\rho(\mathbf{B})$ estimated by ROCK4 is about 1400, so that no more than the minimum number of stages $s = 5$ is required according to (2.7). As a consequence, a CPU time of 1029 s is needed compared with the coupled resolution with ROCK4 that

takes 23967 s. The latter considers time steps of about 2.4×10^{-6} given by the fine tolerances and 5 internal stages for a larger spectral radius of 95000, that considers both reaction- and diffusion-associated eigenvalues. Even though this coupled resolution should be more accurate than a splitting technique, it will be no longer feasible for larger computational domains and moreover not appropriate for more complex chemical terms[1].

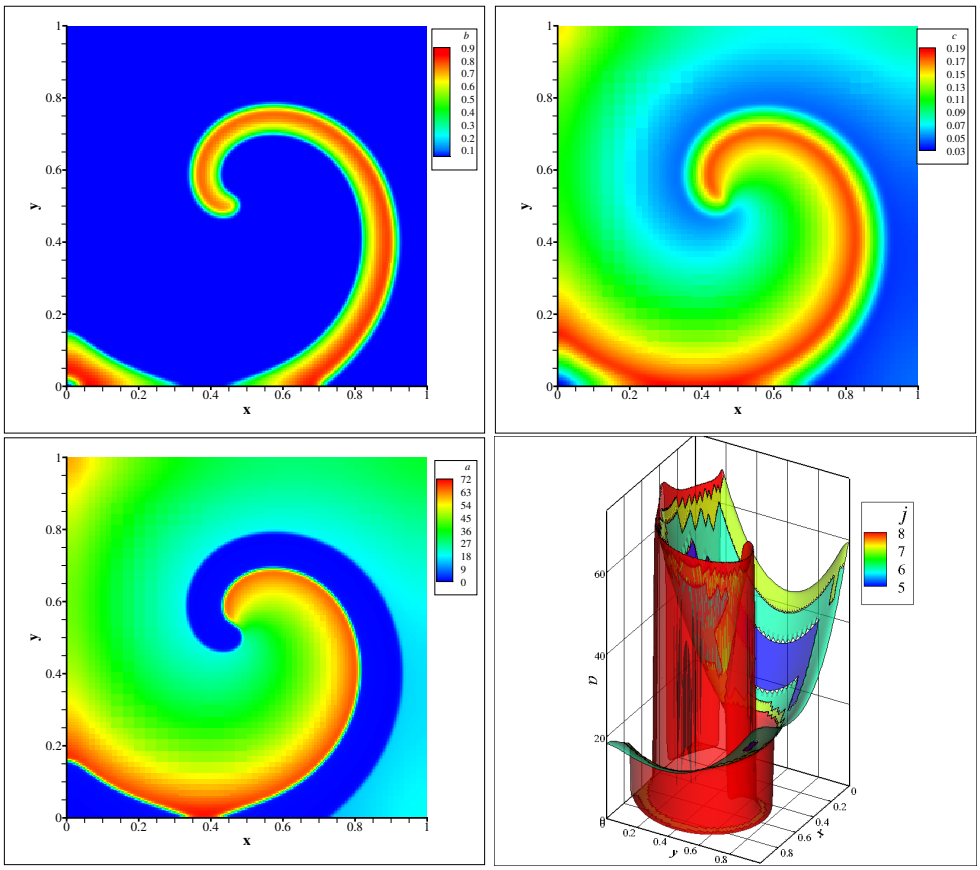

FIG. 4.2. *2D BZ spiral waves. Top: variable b (left) and c (right) at $t = 4$. Bottom: variable a (left) and its representation with four levels of mesh discretization with $\varepsilon = 10^{-2}$ (right). Finest grid:* 256^2.

We consider now the proposed strategy that combines the previous splitting solver with the multiresolution adaptive technique. Figure 4.2 shows the spiral waves and the four different levels of spatial discretization on which they have been simulated with $\varepsilon = 10^{-2}$ for the stiffest variable a. Whenever we consider grid adaptation the bound (3.6) on the splitting time step Δt must be taken into account. We need then to estimate the speed of propagation v of the wavefront. This can be done by computing the propagating speed of each variable along each direction as it is shown in Figure 4.3 for variable a along the y-axis. For the BZ waves, we have estimated a maximum speed of $v_x = v_y \approx 0.7$, which yields a maximum splitting time step

[1]For instance, for the coupled problem (4.1) ROCK4 starts showing stability problems for $\eta_{ROCK4} < 10^{-5}$.

$\Delta t_{\max} = 1.6 \times 10^{-2}$ for $\Delta x = \Delta y = 1/256$ and $\hat{k} = 3$, considering one enlarging cell in each direction $\bar{k} = 1$ in the refinement criteria. In this particular case, we can also compute the speed relative error E_v following (3.2) between the *quasi-exact* and a splitting solution with $\Delta t = 4/1024$. These errors remain practically lower than 0.2% as seen in Figure 4.3, which imply an accuracy tolerance of $\eta_v = 2 \times 10^{-3}$ considering the speed resolution.

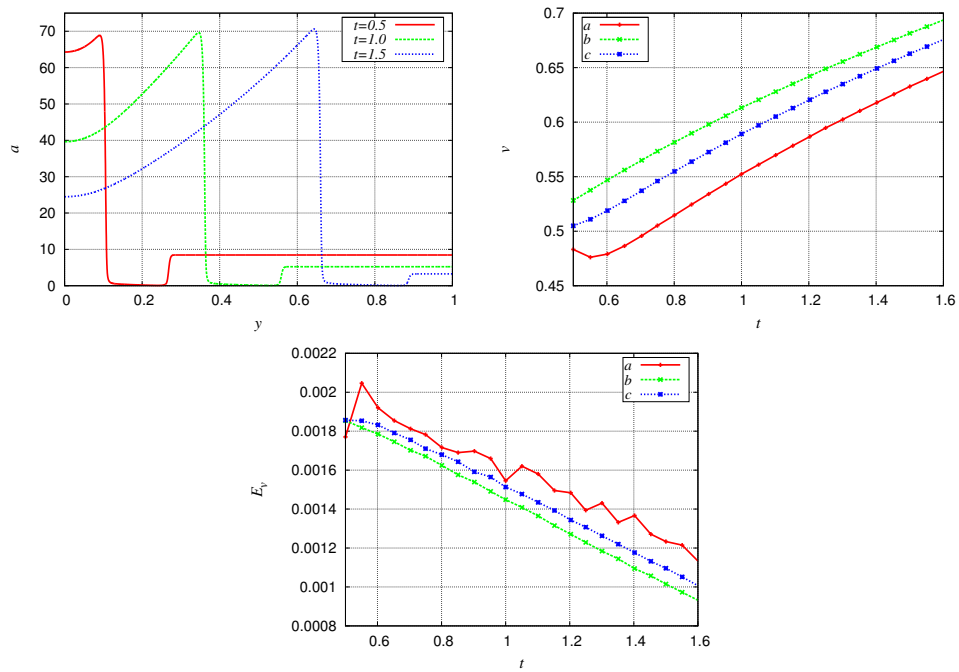

FIG. 4.3. *BZ wave speed for splitting time step* $\Delta t = 4/1024$. *Top left: time evolution of variable a along y-axis (see Figure 4.2); and right, estimated wavefront speed v. Bottom: speed relative errors* E_v *for all three variables according to (3.2).*

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:

$$DC = \left(1 - \frac{AG}{FG}\right) \times 100. \tag{4.6}$$

Figure 4.4 shows different data compression rates for several threshold values. Smaller threshold values ε imply more refinement and thus, compressions are less important. The whole finest grid is necessary for $\varepsilon < 10^{-5}$. The corresponding CPU times for each one of these computations are included in Table (4.1) along with the *quasi-exact* and the splitting solution without any grid adaptation.

A more precise analysis of the CPU time consumption summarized in Table (4.2) shows that, as it was expected, an adapted grid allows to significatively reduce the time cost of the reaction integration as a consequence of the important reduction of the number of points without any chemical activity. On the other hand even

TABLE 4.1
2D BZ. CPU time in seconds for quasi-exact, splitting and MR/splitting solutions with different threshold values ε. Finest grid: 256^2.

MR/splitting $\varepsilon =$	10^{-1}	10^{-2}	10^{-3}	10^{-4}	splitting	*quasi-exact*
CPU time (s)	536	886	1233	2402	1029	23967

though for $\varepsilon = 10^{-2}$ we consider only 25 % of the 256^2 points, an important overhead is introduced in the time integration of the diffusion because the introduction of *phantoms* cells yield denser matrix representations of the discretized diffusion operator and thus more expensive matrix-vector products. For these stiff problems, the MR operations represent less than 15 % of the total time, whereas the construction of the diffusion matrix takes over 6 % since it has to be recomputed at each splitting time step contrary to a uniform grid representation for which this matrix is constant. From a practical point of view, we can see that a more efficient strategy will directly consider the adaptive grid for the flux evaluations as detailed in [25] without the introduction of any *phantoms* cells. Nevertheless, in this work we will keep the previous approach for being more accurate and taking into account that better performances are expected for larger computational domains for which grid adaptation is mandatory for the feasibility of simulations. An overhead is introduced in the CPU times in Table (4.2) coming from the profiling of the codes.

TABLE 4.2
2D BZ. CPU time in seconds for the reaction and diffusion time integrations for a splitting and a MR/splitting resolution with $\varepsilon = 10^{-2}$. *Finest grid:* 256^2.

	splitting		MR/splitting $\varepsilon = 10^{-2}$	
	CPU time (s)	%	CPU time (s)	%
Reaction	963	65.4	486	44.0
Diffusion	481	32.7	348	31.5
Total	1472	100.0	1104	100.0

We consider now the numerical accuracy of the MR/splitting strategy $\mathbf{U}^{MR}_{split}$, discussed in § 3.2, with respect to the reference solution, $\mathbf{U}^{J}_{qe}$, for the semi-discretized problem (4.1) on a uniform mesh J given by 256^2 points. The accuracy of the splitting scheme $\mathbf{U}^{J}_{split}$ applied to (4.1) discretized on the same uniform mesh J, is given by an accuracy tolerance η according to (3.3) through the proper choice of the splitting time step Δt (see Figure 4.1) regardless the possible loss of order for the Strang $\mathcal{S}_2^{\Delta t}$ scheme [14, 11]. On the other hand, a multiresolution decomposition yields a compressed spatial representation whose accuracy to approximate the corresponding uniform mesh representation is related to the threshold value ε through (3.5). At some fixed time $t^\star$ the overall numerical accuracy of the MR/splitting resolution is then set by the previous splitting and multiresolution errors:

$$\left\|\mathbf{U}^{J}_{qe} - \mathbf{U}^{MR}_{split}\right\|_{L^2} \leq \left\|\mathbf{U}^{J}_{qe} - \mathbf{U}^{J}_{split}\right\|_{L^2} + \left\|\mathbf{U}^{J}_{split} - \mathbf{U}^{MR}_{split}\right\|_{L^2}. \tag{4.7}$$

Figure (4.4) shows the corresponding normalized L^2 errors at $t^\star = 4$ for several threshold values ε and $\eta = 10^{-2}$ for a splitting time step of $\Delta t = 4/1024$. Multiresolution errors are evaluated at the finest grid J after reconstruction from the adapted mesh solution and depend proportionally on the imposed threshold value ε according to (3.5). For this time multi-scale phenomenon, the accuracy of the MR/splitting

strategy should be fixed by the time integration process in order to guarantee an appropriate resolution of the time scale spectrum of the stiff problem, whereas the multiresolution procedure allows to compress the spatial representation by retaining the desired level of refinement only wherever it is necessary, taking into account the space multi-scale features of the physical problem. In this case, these error estimates show that for $\varepsilon \leq 10^{-2}$, the multiresolution errors become negligible compared with the operator splitting ones so that the overall accuracy is indeed given by η.

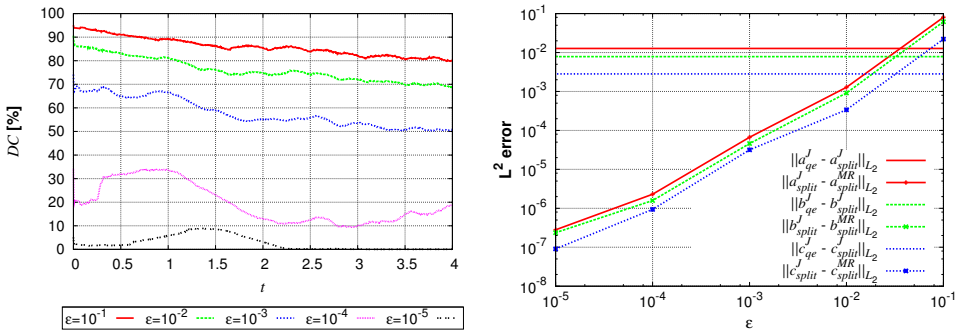

FIG. 4.4. *2D BZ spiral waves. Left: time evolution of the data compression DC given by (4.6) in percentage. Right: normalized L^2 errors at $t^\star = 4$ given by the splitting technique on a uniform grid according to (3.3), and by the MR procedure according to (3.5), for several threshold values ε and splitting time step of $\Delta t = 4/1024$. Finest grid: 256^2.*

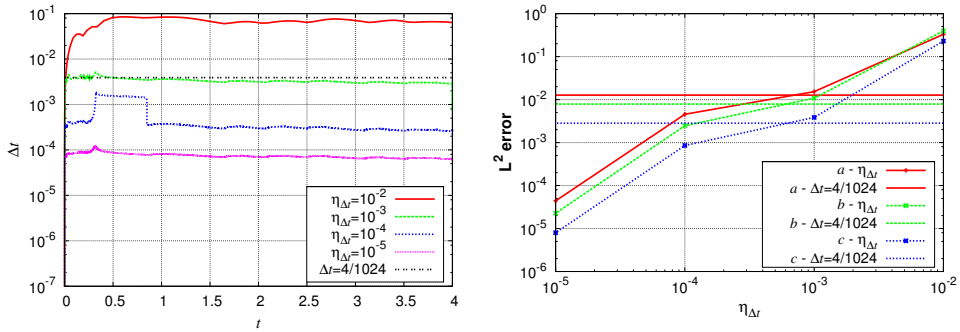

FIG. 4.5. *2D BZ spiral waves. Time evolution of splitting time steps (left) and normalized L^2 errors at $t^\star = 4$ (right) given by the splitting technique on a uniform grid of 256^2 according to (3.3) for different accuracy tolerances $\eta_{\Delta t}$ for the time adaptive strategy [10] and with $\Delta t = 4/1024$.*

Figure (4.5) shows the adaptive splitting time steps corresponding to different accuracy tolerances $\eta_{\Delta t}$ for the local error of the splitting time integration, according to the time adaptive procedure introduced in [10]. In all cases, the splitting time step is adapted from a chosen initial value of $\Delta t = 10^{-7}$ to a roughly constant value that depends on the prescribed accuracy due to the self-similar character of the wave. The global error is indeed controlled by the local error accuracy tolerance $\eta_{\Delta t}$ as it is shown by the normalized L^2 errors at $t^\star = 4$ according to (3.3). The same time integrations errors for the various tolerances $\eta_{\Delta t}$ are found for different adapted meshes as long as $\varepsilon \leq 10^{-2}$. We see thus that for this kind of propagating phenomena a constant splitting time step computed based on an accuracy criterion as detailed in § 3.2,

is appropriate to describe accurately the multi-scale phenomenon. Furthermore, a splitting time step of $\Delta t = 4/1024$ yields practically the same results as the adaptive splitting strategy with $\eta_{\Delta t} = 10^{-3}$, so that the overhead of estimating the local errors of the adaptive scheme can be saved, even though this overhead implies no more than 25 % of additional CPU time considering the embedded procedure developed in [10].

We have analyzed so far the numerical behavior of the proposed numerical strategy in terms of the splitting time integration method, the computational costs and the numerical errors for a computational domain of 256^2 points, that allows to represents accurately enough the multi-scale phenomenon and furthermore, it allows to conduct several computations with reasonable computational resources. Let us now consider a more challenging configuration with larger computational domain in order to complete the present study. We therefore consider the semi-discretized problem (4.1) discretized this time over 10 nested dyadic grids with $N = 2^{2\times 10} = 1048576 = 1024^2$ cells on the finest grid $J = 10$.

In order to take into account the memory requirements of each resolution strategy, 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 [22]);
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 a uniform mesh, the total number of unknowns is $W = 3 \times 1024^2 \approx 3.15 \times 10^6$ and thus, the global size L required for each solver is:

1. Quasi-exact with Radau5: $W_1 = W \approx 3.15 \times 10^6$ 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, 1.5 Gb and 140.4 Mb respectively, for each solver. For standard computational resources, an implicit resolution with Radau5 is completely out of reach. These expensive memory requirements are strongly reduced with a splitting strategy but further reductions are achieved by adding a multiresolution adaptive procedure.

Figure 4.6 (top) shows the spatial representation of variable a on the finest level corresponding to a 1024^2 spatial discretization of problem (4.1) at an intermediary time $t = 2$ and after one revolution at final time $t = 4$. The corresponding data compressions DC are respectively of 92.3 % and 89.9 %, while the steepest spatial gradients of the front are always solved within the finest region taking into account that the splitting time step $\Delta t = 4/1024$ remains bounded by $\Delta t_{\max} = 4.2 \times 10^{-3}$ for $\Delta x = \Delta y = 1/1024$ according to (3.6). For this case, six levels of grid discretization were used[2] from 5 to the finest grid 10.

Bottom of Figure 4.6 shows the corresponding reaction time steps averaged within $\Delta t/2$ for each point. We have the same distribution as in the previous 256^2 case with reaction time steps going from $\Delta t/2$ to time steps almost 22 times smaller depending of the local chemical activity. On the other hand, the spectral radius $\rho(\mathbf{B})$ estimated by ROCK4 is larger because of the finer spatial discretization of the Laplacian operator and it is of the order of 23000, so if we consider a diffusion time step Δt_D equal to the splitting time step, $s = 16$ stages will be needed according to (2.7)[3]. However,

[2]Same performance is obtained by considering 16 *roots* in each direction and $J = 6$ levels of discretization.

[3]It is important to notice that ROCK4 needs to save only 8 arrays of the size of the number of

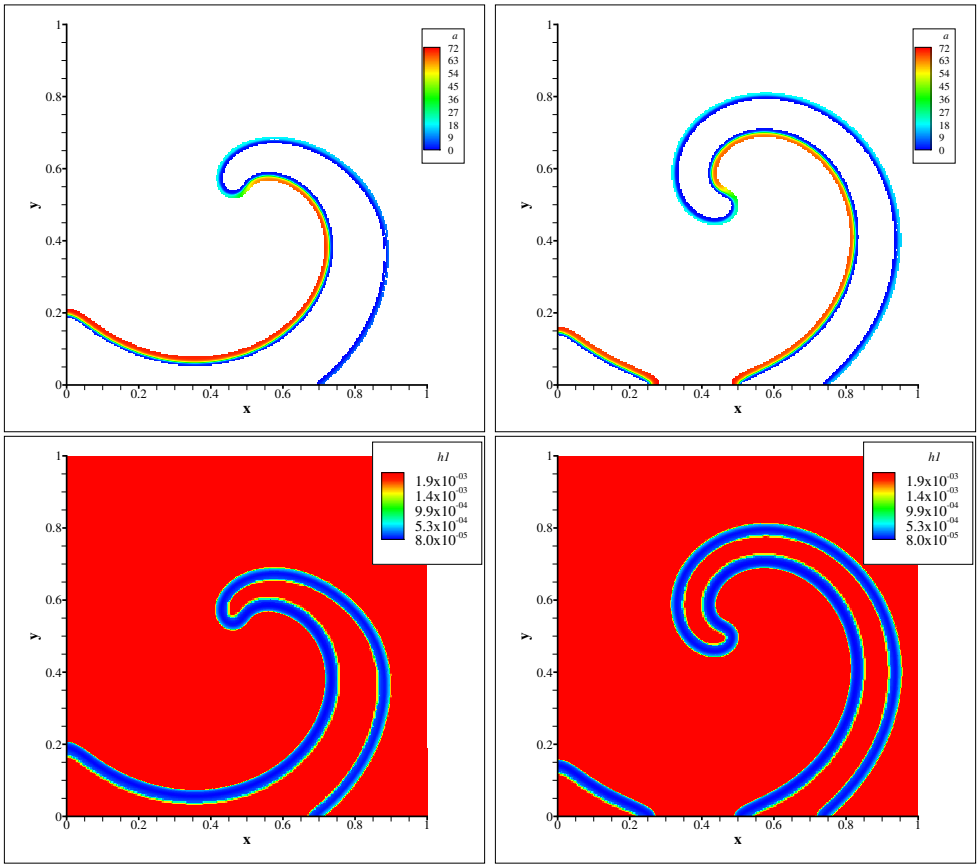

FIG. 4.6. *2D BZ spiral waves. Variable a on the finest grid (top) and local reaction time steps $h1$ (bottom) at $t = 2$ (left) and $t = 4$ (right) with $\Delta t = 4/1024$ and $\varepsilon = 10^{-2}$. Finest grid:* 1024^2.

for a given tolerance of $\eta_{ROCK4} = 10^{-5}$, an initial time step given by $\Delta t_D = \Delta t$ is rejected to finally reach a relatively constant value of $2.5 - 3.5 \times 10^{-4}$ for which 5 stages instead of 16 are enough to guarantee the stability of the method. Finally, $11 - 15$ diffusion time steps Δt_D are computed inside each splitting time step Δt.

Figure 4.7 (top left) shows the adaptive splitting time steps obtained by the local error estimate procedure introduced in [10] with an accuracy tolerance of $\eta_{\Delta t} = 10^{-3}$. This is shown to be almost equivalent to the constant splitting time step $\Delta t = 4/1024$ as in the previous 256^2 spatial discretization case, where this splitting error is measured for a given semi-discretized problem. In general and for this kind of propagating phenomena, the adaptive scheme can be used to initially compute the corresponding constant splitting time step for a given accuracy.

In all these illustrations, the finest grid of computation is previously settled and it is basically limited by the computational resources so that the multiresolution error is indicating the numerical approximation of the compressed spatial representation with respect to the semi-discretized problem regardless its spatial discretization as

unknowns regardless the number of stages. One of these arrays contains the approximate solution used to estimate the local error in order to adapt the time step of integration within the prescribed tolerance.

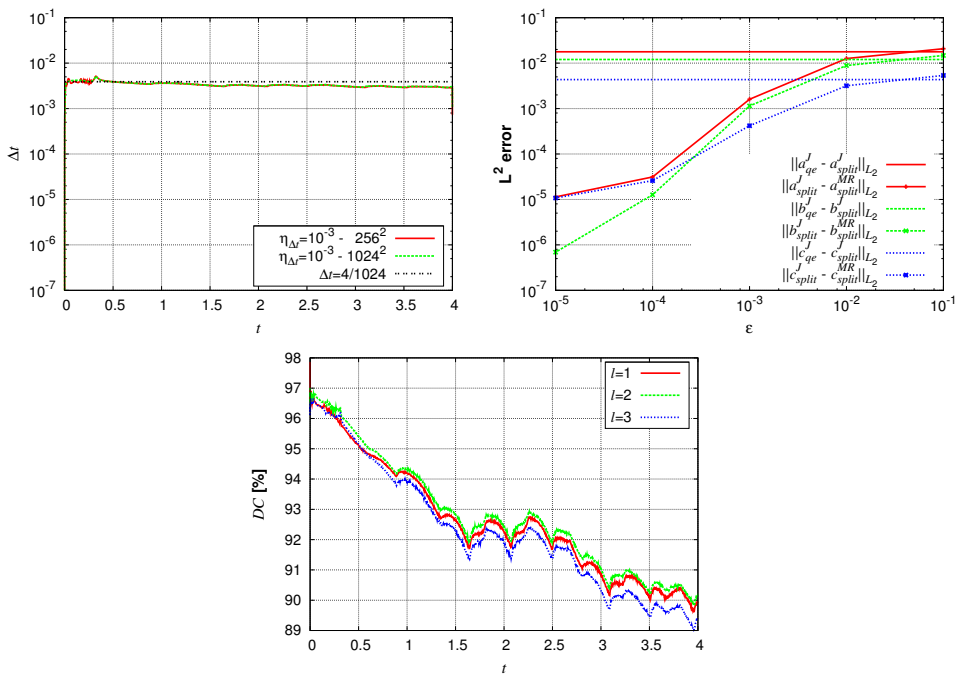

FIG. 4.7. *2D BZ spiral waves. Top left: time evolution of splitting time steps with* $\eta_{\Delta t} = 10^{-3}$ *and finest grids of* 256^2 *and* 1024^2*, with* $\Delta t = 4/1024$*. Top right: normalized* L^2 *errors at* $t^\star = 4$ *given by splitting technique on a uniform grid according to (3.3), and MR procedure according to (3.5) for several threshold values* ε*, splitting time step* $\Delta t = 4/1024$ *and a finest grid of* 1024^2*. Bottom: time evolution of data compression DC for prediction operators given by polynomial interpolations of different orders* $2l$ *and a finest grid of* 1024^2*.*

it is shown in Figure 4.7 (top right). The *quasi-exact* solution was computed with ROCK4 and $\eta_{ROCK4} = 10^{-10}$ and took over 65072 s compared with 13943 s and 9529 s respectively for the splitting solution and the MR/splitting solution with $\varepsilon = 10^{-2}$.

Figure 4.7 (bottom) shows the dependence of the data compression on the definition of the prediction operator (2.11) according to § 3.1. For higher order polynomial interpolations that yield more accurate and thus more compressed multiresolution representations according to (2.12), larger stencils are also needed so that the resulting data compression results of both opposite actions. In a general case, this is a problem dependent feature that can be a useful parameter to improve the performance of a multiresolution technique. A more detailed study of this dependence is out of the scope of the present work.

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 $[0, 1]^3$. Initialization is made in the same way but with the coordinate angle θ given by

$$\theta = \arctan\left(\frac{(x-0.5)\sin(\pi z) + (y-0.5)\cos(\pi z)}{(x-0.5)\cos(\pi z) - (y-0.5)\sin(\pi z)}\right), \tag{4.8}$$

z equal to zero corresponds to the previous 2D case.

First, we take into account 8 nested dyadic grids with $N = 2^{3\times 8} = 16777216 = 256^3$ cells on the finest grid $J = 8$. Then, with a threshold value of $\varepsilon = 10^{-2}$ and

a splitting time step $\Delta t = 4/1024$, the proposed numerical strategy features data compressions of 92.61 % for the initial condition, 85.64 % at $t = 1$ when the scroll waves are fully developed and 81.42 % at final time $t = 2$. Figure 4.8 shows the scroll waves for variable a at two different times and the adapted grid at $t = 2$. The finest regions correspond to the neighborhood of the wavefront. The adaptive splitting time technique yields also a roughly constant splitting time step after a short initial transient whereas the choice $\Delta t = 4/1024$ corresponds approximatively to $\eta_{\Delta t} = 10^{-3}$. The CPU computation time was of about 41.94 hours with one processor.

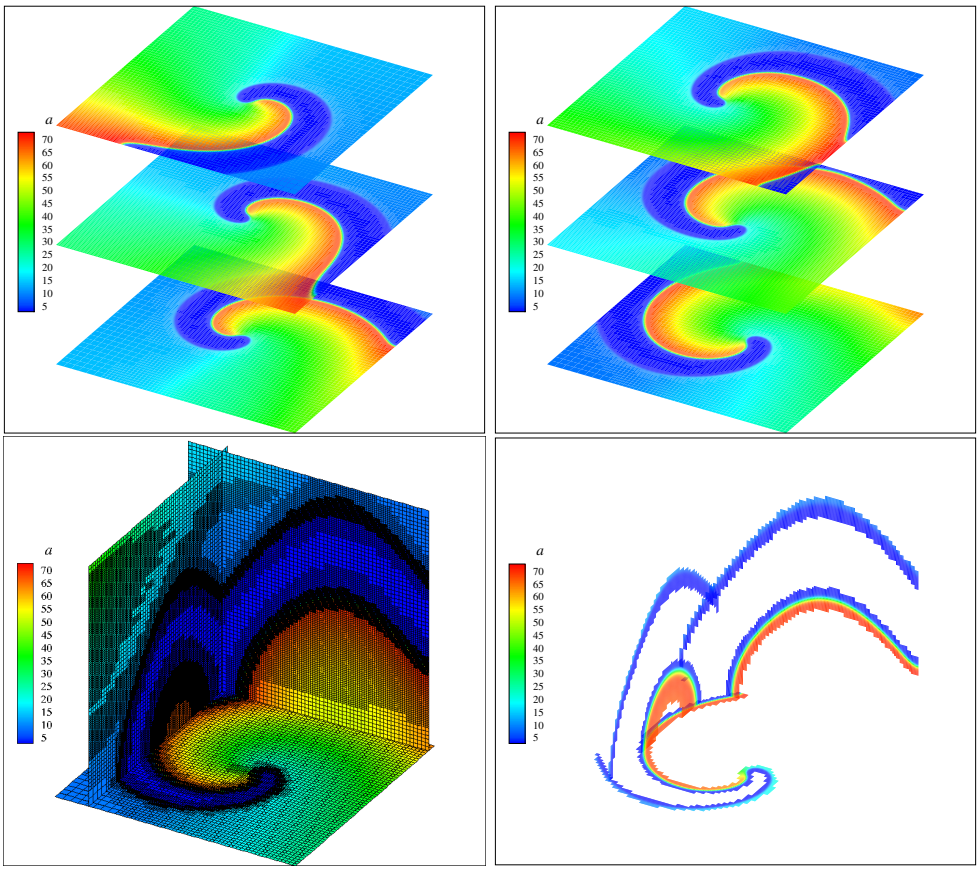

FIG. 4.8. *3D BZ scroll wave. Top: evolution of variable a at $t = 1$ (left) and $t = 2$ (right). Bottom: Adapted grid (left) and finest grid (right) at $t = 2$ for $\varepsilon = 10^{-2}$. Finest grid: 256^3.*

In order to explore the feasibility and potential advantages of the method, let us consider 9 nested dyadic grids with $N = 2^{3\times 9} = 134217728 = 512^3$ cells on the finest grid $J = 9$. The initialization must take place on a intermediary grid, $j = 8$ 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 4.9 shows the adapted grid at $t = 2$ and the corresponding finest regions. Compared with the 256^3 case, finer regions are added at the steepest spatial gradients of

the front. On the other hand, in order to globally guarantee (3.5), more refinement is needed at the lower levels according to (2.18) for a given threshold ε, where the multiresolution representation error (3.5) is always measured with respect to the corresponding uniform semi-discretized problem at the finest level. The latter is mainly limited by the computational resources and the desired level of accuracy of the spatial resolution. Data compressions are now of 95.79 % for the initial condition, 91.56 % at $t = 1$ and 91.20 % for final time $t = 2$, with a CPU computation time of 159.4 hours.

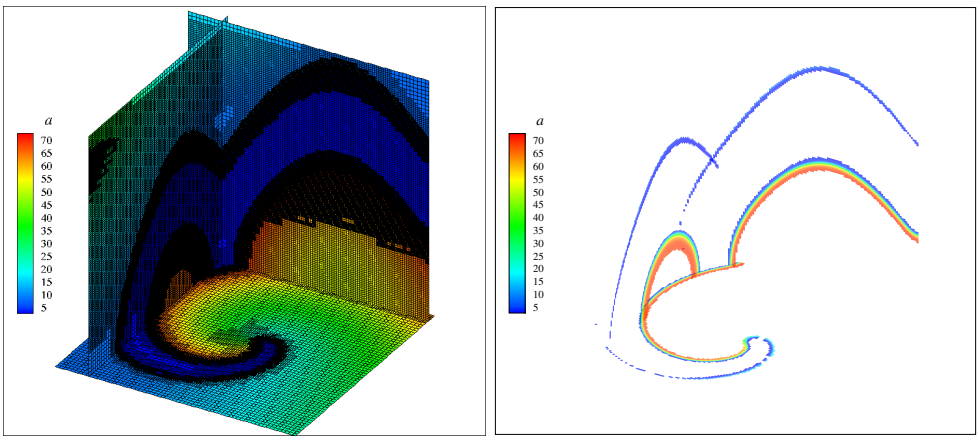

FIG. 4.9. *3D BZ scroll wave. Adapted grid (left) and finest grid (right) at $t = 2$ for $\varepsilon = 10^{-1}$. Finest grid:* 512^3.

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 with Radau5: $W_1 = W \approx 4.03 \times 10^8$ 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.

5. Concluding Remarks and Outlook. Based on recent mathematical studies [14, 11] that allow to better characterize the behavior of splitting techniques for time-space stiff PDEs and in particular for splitting time steps much larger than the fastest scales present in the problem, this work introduces a new time operator splitting approach that exploits these results with an important gain of efficiency. Contrary to classical splitting strategies that consider splitting time steps set by the fastest time scales, in our strategy the splitting time step is chosen on the sole basis of the physics of the global phenomenon and its decoupling capabilities, but not related to any stability requirement of the numerical methods used to integrate each subsystem, even if strong stiffness is present. Dedicated high order time integration methods are then chosen for the reaction and diffusion problems to properly solve the entire spectrum of temporal scales associated with each independent problem. In this way, an effective decoupling of the time scale spectrum is achieved whenever this is allowed by the physics of the problem, so that different physical or numerical time scales associated with each problem can be isolated and treated independently by each numerical method.

We have shown that in the context of self-similar propagating waves a constant splitting time step is enough to capture the dynamics of the phenomenon. This was confirmed by an adaptive splitting strategy introduced in [10] that yields the same conclusion. Considering the adequate choice of higher order numerical methods with adaptive time stepping based on accuracy criteria such as Radau5 and ROCK4, the main error of the time integration is piloted by the splitting scheme and it is thus settled by the splitting time step even for stiff cases [14, 11]. The latter is then computed based on error estimates of some physical feature such as the profile of the wavefront or its propagation speed. In a standard numerical strategy, this kind of preliminary study can be easily performed in 1D or smaller multidimensional configurations to then extrapolate to larger and more expensive computational domains taking into account that these splitting errors do not consider the spatial discretization error. However, from a practical point of view an adaptive splitting strategy can be used to initialize the splitting time step within the prescribed accuracy.

In order to improve the performance of the time integration strategy an adaptive mesh refinement technique based on spatial multiresolution was also implemented. One of the main advantage of such method is that for a given semi-discretized problem, the error introduced by the corresponding compressed spatial representation can be controlled. Even though a rigorous mathematical proof of multiresolution errors for parabolic problems is not yet available, the numerical results confirm the error estimates. In this way, for a problem represented with a spatial discretization limited mainly by the computational resources, the proposed MR/splitting strategy allows to track the corresponding numerical error of the simulation introduced by the numerical methods of resolution. 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.

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 [9, 26]. Therefore, an important amount of work is still in progress concerning programming features such as data structures, optimized routines and parallelization strategies for the time integration technique as well as for the multiresolution environment, even though the global CPU time is largely dominated by the time resolution for these stiff problems. For instance, some dedicated and efficient implementations have been recently developed for multiresolution applications [3, 4]. Finally, when dealing with more complex systems such as complex or more detailed 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 data compression issued from the multiresolution allows to improve its resolution as shown in [16]. Therefore, this field also 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*, SIAM 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] J. C. Butcher, *Implicit Runge-Kutta processes*, Math. Comp., 18 (1964), pp. 50–64.

[6] A. Cohen, *Wavelet Methods in Numerical Analysis*, vol. 7, Elsevier, Amsterdam, 2000.

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

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

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

[10] S. Descombes, M. Duarte, T. Dumont, V. Louvet, and M. Massot, *Adaptive time splitting method for multi-scale evolutionary PDEs.*, Confluentes Mathematici, (2011). Accepted for publication, available on HAL (http://hal.archives-ouvertes.fr/hal-00587036).

[11] 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.

[12] 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*, In preparation, (2011).

[13] 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.

[14] 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.

[15] M. Duarte, Z. Bonaventura, M. Massot, A. Bourdon, S. Descombes, and T. Dumont, *A new numerical strategy with space-time adaptivity and error control for multi-scale streamer discharge simulations.*, J. Comput. Phys., (2011). Accepted for publication, available on HAL (http://hal.archives-ouvertes.fr/hal-00573043).

[16] 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 Math. Biology, (2011). Available on HAL (http://hal.archives-ouvertes.fr/hal-00546223).

[17] B. Ehle, *A-stable methods and Padé approximations to the exponential*, SIAM J. Math. Anal., 4 (1973), pp. 671–680.

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

[19] R. J. Field, E. Koros, and R. M. Noyes, *Oscillations in chemical systems. II. Thorough analysis of temporal oscillation in the bromate-cerium-malonic acid system*, J. Amer. Chem. Soc., 94 (1972), pp. 8649–8664.

[20] P. Gray and S. K. Scott, *Chemical Oscillations and Instabilites*, Oxford Univ. Press, 1994.

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

[22] E. Hairer and G. Wanner, *Solving 0rdinary Differential Equations II*, Springer-Verlag, Berlin, 2nd ed., 1996. Stiff and differential-algebraic problems.

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

[24] W. Jahnke, W. E. Skaggs, and A. T. Winfree, *Chemical vortex dynamics in the Belousov-Zhabotinsky reaction and in the two-variable Oregonator model*, J. Phys. Chem., 93 (1989), pp. 740–749.

[25] S. Müller, *Adaptive Multiscale Schemes for Conservation Laws*, vol. 27, Springer-Verlag, 2003.

[26] 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.

[27] 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.

[28] 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.

[29] 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.

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

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

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

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

[34] 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.

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