Revista Facultad de Ciencias Universidad Nacional de Colombia, Sede Medellín
V 3 N°2 julio-diciembre 2014 • ISSN 0121-747X
Páginas 81 a 93

SOME NUMERICAL STUDIES OF OSCILLATING CHEMICAL REACTIONS USING DISCONTINUOUS FINITE ELEMENTS

ALGUNOS ESTUDIOS NUMÉRICOS DE REACCIONES QUÍMICAS OSCILANTES USANDO ELEMENTOS FINITOS DISCONTINUOS

FELIPE PONCE VANEGAS [a], JUAN GALVIS [a], JUAN MIGUEL MANTILLA [b]

Recibido 04-09-14, aceptado 15-12-14, versión final 16-12-14.
Artículo Investigación

RESUMEN: La modelación numérica de la interacción entre reacciones químicas oscilantes y procesos difusivos requiere métodos numéricos sofisticados. La rigidez de estos modelos es eludida usando un método de Runge-Kutta diagonal para incrementar la estabilidad, y el tiempo de paso es controlado con un esquema incrustado de orden inferior. Dado que los patrones espaciales involucran grandes cambios de concentración, conviene realizar la discretización espacial con el método de Galerkin discontinuo (dG). Adicionalmente, mejoramos el método por medio de un precondicionamiento de multimalla. La eficiencia es evaluada con una reacción de un solo paso con solución exacta y se muestran algunos resultados con un modelo típico de reacción oscilante, el Brusselador.

PALABRAS CLAVE: Difusión, método de Galerkin discontinuo, reacción química oscilante.

ABSTRACT: Modeling the interplay between oscillating chemical reactions and diffusive processes requires sophisticated numerical methods. The stiffness of these models is worked around using a singly diagonally Runge-Kutta method to enlarge the stability and the time step is controlled with an embedded scheme of lower order. Since spatial patterns involve large changes of concentration, spatial discretization is carried out by discontinuous Galerkin method (dG). In addition, we improve the method by means of multigrid preconditioning. The efficiency of the method is tested with a one-step reaction exactly solvable and some results are shown for a prototypical model of oscillating reaction, the Brusselator.

KEYWORDS: Diffusion, discontinuous Galerkin method, oscillating chemical reaction.

1. INTRODUCCIÓN

The discovery of oscillating chemical reactions by Boris Belousov around 1950 while experimenting with bromate, citric acid, and ceric ions, and improved by Anatol Zhabotinsky changing citric acid

[a]Departamento de Matemáticas, Universidad Nacional de Colombia, Bogotá, Carrera 45 N 26-85.
[b]Departamento de Ingeniería Mecánica, Universidad Nacional de Colombia, Bogotá, Carrera 45 N 26-85.

for malonic acid (Epstein and Pojman, 1998), astonished the chemical community and challenged the understanding about chemical reactions and thermodynamics. A chemical reaction often proceeds monotonically from reactants to products. The amount of reactant decreases continuously up to some equilibrium point and the amount of product increases likewise. Oscillating reactions are quite different, since they go forward and backward, switching between reactants and products during some time, until the system attains the equilibrium. These reactions can be classified in two groups: Belousov-Zhabotinsky (BZ) reactions and CIMA (Chlorite-Iodide-Malonic Acid) reactions, (De Kepper et al., 1982). BZ reactions can be further divided in two models with oscillatory behavior: the Brusselator, (Prigogine and Lefever, 1969), and the Oregonator, (Field and Noyes, 1974). CIMA reactions are distributed into Brandeisator, (Lengyel and Epstein, 1991), and Schnakenberg models, (Schnakenberg, 1979), among others. CIMA reactions play an important role in present days in the study of chemical oscillation during Turing pattern formation (Turing, 1952; Garzón et al., 2011).

Now we focus our discussion on oscillating chemical reactions. For detailed discussion on oscillating chemical reactions, see Epstein and Pojman (1998). A chemical equation is a short way of writing a chemical reaction, involving only reactants and products. For instance

$$r_1 S_1 + r_2 S_2 + \cdots + r_n S_n \to p_1 S_1 + p_2 S_2 + \cdots + p_n S_n, \tag{1}$$

where S_i are the substances involved in the reaction, and r_i and p_i are the stoichiometric coefficients. A chemical equation does not provide information about how the chemical reaction proceeds, and in general rather complex experiments are conducted to determine the *reaction mechanism*, which is the set of elementary steps a reaction goes through. If the reaction (1) is an elementary step, then the change of concentration of a substance S_i is given by

$$\frac{1}{p_i - r_i}\dot{u}_i = k \prod_{i=1}^{n} u_i^{r_i},$$

where k is the rate constant of the elementary step and u_i is the concentration of the substance S_i; in general we write $\boldsymbol{u} = (u_1, \ldots, u_n)$. Note that we assume that reactions proceed only in forward direction.

We consider a chemical reaction with n substances and N elementary steps of the form (1), and in each step ℓ we have the coefficients $r_{\ell 1}, \ldots, r_{\ell n}$ for the corresponding substances. We collect these coefficients into the multi-index $\boldsymbol{r}_\ell = (r_{\ell 1}, \ldots, r_{\ell n})$ and introduce, for a given vector $\boldsymbol{u} \in \mathbb{R}^n$, the notation

$$\boldsymbol{u}^{\boldsymbol{r}_\ell} := \prod_{i=1}^{n} u^{r_{\ell i}};$$

with this notation we can write the reaction term as

$$\boldsymbol{R} = (R_1, \ldots, R_n), \qquad \text{where} \qquad R_i(\boldsymbol{u}) = \sum_{\ell=1}^{N} (p_{\ell i} - r_{\ell i}) \kappa_\ell \boldsymbol{u}^{\boldsymbol{r}_\ell},$$

where κ_ℓ is the rate constant of the ℓ-th step.

Oscillating reactions are particularly nice when they are conducted in a Petri dish, avoiding convection due to airstream or movement of the dish, so that the movement of molecules is due only to diffusion. When a chemical reaction is coupled with diffusion, the system of partial differential equations to solve is

$$\dot{u}_i - \operatorname{div}(D_i \nabla u_i) - R_i(\boldsymbol{u}) = 0 \qquad \text{for every } i = 1, \ldots, n,$$

where D_i is the diffusion coefficient of S_i. If the reaction occurs in a closed system Ω, then the flux in the boundary is zero. We impose this boundary condition (Neumann condition) for every substance as follows

$$\partial_{\boldsymbol{n}} u_i|_{\partial\Omega} = 0.$$

Because of the sharp spatial changes of concentration originated by oscillations, we prefer to use discontinuous Galerkin methods (dG). Discontinuous elements were introduced by Reed and Hill (1973) and have been successfully applied in many diffusion-reaction systems (Michoski et al., 2010; Zhu et al., 2009; Kanevsky et al., 2007). Additionally, dG methods feature more stability than continuous elements in case we want to add an advective term. Detailed discussion can be found in Di Pietro and Ern (2012).

Chemical reactions could be strongly stiff, as the oregonator, so explicit schemes are unsuitable because of their poor stability; we implement thus a singly diagonal Runge-Kutta scheme to overcome the stiffness of the oscillating systems. On the other hand, it could well happen that the system remains almost linear and changes suddenly, so that we use an embedded scheme and set the step length according to the behavior of the solution.

In Section 2 we outline the dG method for diffusion-reaction systems. In section 3 we explain the time discretization and the control over the time. In section 4 we test the numerical method with a one-step reaction, whose exact solution is available. Furthermore, we show some results for the Brusselator.

2. SPACE SEMI-DISCRETIZATION

Let n be the number of substances and u_i the concentration of the i–th substance. We define $a : H^1(\Omega)^n \times H^1(\Omega)^n \to \mathbb{R}$ by

$$a(\boldsymbol{u}, \boldsymbol{v}) = \sum_{i=1}^{n} D_i \int_\Omega \nabla u_i \cdot \nabla v_i.$$

Additionally, for $\boldsymbol{u}$ and $\boldsymbol{v}$ in $H^1(\Omega)^n$ define $(\boldsymbol{u},\boldsymbol{v}) = \sum_{i=1}^n \int_\Omega u_i v_i$. We consider the problem of finding a function $\boldsymbol{u}$ of x and t such that

$$(\dot{\boldsymbol{u}}(t),\boldsymbol{v}) + a(\boldsymbol{u}(t),\boldsymbol{v}) - (\boldsymbol{R}(\boldsymbol{u}(t)),\boldsymbol{v}) = 0 \quad \text{for all } t>0 \text{ and } \boldsymbol{v}\in\boldsymbol{V}, \tag{2}$$

where $\boldsymbol{V}$ is an appropriate test function space and $\boldsymbol{u}(t) = \boldsymbol{u}(\cdot,t)$. Note that this weak formulation corresponds to the case of homogeneous Neumann boundary condition for each concentration. The function $\boldsymbol{R}$ is the reaction term, depending on the chemical set of elementary reactions.

We now introduce a quasi-uniform and shape regular triangulation $\mathcal{T}$ of Ω, consisting of quadrilateral elements. A generic element of the triangulation is denoted by τ. When the elements τ are rectangles, we define the discontinuous finite element space as follows

$$Q^p = \left\{ v\in L^2(\Omega), \quad \begin{array}{l}\text{such that } v|_\tau \text{ is a polynomial of degree } p\\ \text{in } \tau.\end{array} \right\}.$$

For a general description when the elements τ are quadrilaterals, see Ciarlet (1979). The jump of a function $u\in Q^p$ through a face $e = \partial\tau^+\cap\partial\tau^-$ is defined as $[u] = u^+ - u^-$ where $u^\pm$ is the value of $u|_{\tau_\pm}$ on e. The average of a function in a face e is denoted and defined by $\{\!\{u\}\!\} = \frac{1}{2}(u^+ + u^-)$.

Let us define the space $\boldsymbol{V}^h = (Q^p)^n$, in which we introduce the mesh-depending norm

$$\|\boldsymbol{u}\|_h^2 = \sum_{i=1}^n \int_\Omega u_i^2 + \sum_{i=1}^n \int_\Omega |\nabla u_i|^2 + \sum_{e\in F}\sum_{i=1}^n \frac{\eta}{|e|}\int_e |[u_i]|^2,$$

where η is the penalty term and F is the set of all inner faces in the triangulation. We also introduce in $\boldsymbol{V}^h$ the discrete bilinear form

$$\begin{aligned} a^h(\boldsymbol{u},\boldsymbol{v}) \;=\; & \sum_{\tau\in\mathcal{T}}\sum_{i=1}^n D_i\int_\tau \nabla u_i\cdot\nabla v_i \\ & - \sum_{e\in F}\sum_{i=1}^n D_i\int_e \{\!\{\nabla u_i\}\!\}\cdot\boldsymbol{n}[v_i] + \{\!\{\nabla v_i\}\!\}\cdot\boldsymbol{n}[u_i] \\ & + \sum_{e\in F}\sum_{i=1}^n D_i\frac{\eta}{|e|}\int_e [u_i][v_i], \end{aligned} \tag{3}$$

where $\boldsymbol{n}$ is the outward normal vector to the τ^+ element. For details see Di Pietro and Ern (2012). Throughout calculations the penalty term is $\eta = 10$. The semi-discrete form of the diffusion-reaction equation is: For $t>0$, find $\boldsymbol{u}(t)\in\boldsymbol{V}^h$ such that

$$(\dot{\boldsymbol{u}}(t),\boldsymbol{v}) + a^h(\boldsymbol{u}(t),\boldsymbol{v}) - (\boldsymbol{R}(\boldsymbol{u}(t)),\boldsymbol{v}) = 0 \quad \text{for all } \boldsymbol{v}\in\boldsymbol{V}^h,$$

with initial condition $\boldsymbol{u}(0) = \boldsymbol{u}(0,x) = \boldsymbol{u}_0(x)$.

3. TIME DISCRETIZATION

The system is fully discretized using a singly diagonally implicit Runge-Kutta scheme with an embedded pair 4(3). Let us define the operator $\mathcal{F}^h$ that solves

$$(\mathcal{F}^h(\boldsymbol{u},t),\boldsymbol{v}) = -a^h(\boldsymbol{u}(t),\boldsymbol{v}) + (\boldsymbol{R}(\boldsymbol{u}(t)),\boldsymbol{v}) \quad \text{for all } \boldsymbol{v}\in\boldsymbol{V}^h.$$

We then write the Runge-Kutta method as follows. The solution $\boldsymbol{u}_{n+1}$ of the main scheme at the time $t_{n+1} = t_n + h$ is computed as

$$\boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \Delta t \sum_{\ell=1}^{N} b_\ell \mathcal{F}^h(\boldsymbol{U}^\ell, t_n + c_\ell \Delta t),$$

where each term $\boldsymbol{U}^i$ is the solution at time $t_n + c_\ell h$ and they are given by

$$\begin{aligned}
\boldsymbol{U}^1 &= \boldsymbol{u}_n^h \\
\boldsymbol{U}^2 &= \boldsymbol{U}^1 + \Delta t[a_{21}\mathcal{F}^h(\boldsymbol{U}^1, t_n) + a_{22}\mathcal{F}^h(\boldsymbol{U}^2, t_n + c_2\Delta t)] \\
&\vdots \\
\boldsymbol{U}^N &= \boldsymbol{U}^1 + \Delta t \sum_{\ell=1}^{N} a_{N\ell}\mathcal{F}^h(\boldsymbol{U}^\ell, t_n + c_\ell \Delta t).
\end{aligned}$$

The solution $\tilde{\boldsymbol{u}}_{n+1}^h$ of the embedded scheme is calculated likewise, but using weights $\tilde{b}_i$ instead of b_i. The coefficients $a_{s\ell}$, weights b_i and nodes c_i of the Runge-Kutta method are arranged in a Butcher tableau as follows,

0				
c_2	a_{21}	a_{22}		
$\vdots$	$\vdots$		$\ddots$	
c_N	a_{N1}	a_{N2}	$\dots$	a_{NN}
	b_1	b_2	$\dots$	b_N
	$\tilde{b}_1$	$\tilde{b}_2$	$\dots$	$\tilde{b}_N$

The Butcher tableau of the implemented Runge-Kutta scheme can be found in Appendix A of Kanevsky et al. (2007) and for completeness we present it here in Appendix A

As time controller we use the I-based controller

$$(\Delta t)^{n+1} = \kappa_{sf}(\Delta t)^n \Big(\frac{\varepsilon}{er + a\varepsilon}\Big)^{1/\gamma},$$

where γ is the order of the embedded scheme, ε is the time tolerance, er is the error of the time scheme and κ_{sf} is the safety factor that we set as 0.9. The term $a\varepsilon$ is to avoid very long steps, so we set $a = 5 \times 10^{-3}$. If the time step error is greater than twice the tolerance, then the step is repeated with a half of the current time step.

The error of the time scheme is estimated as the difference between the main and the embedded scheme $\boldsymbol{w}_m = \boldsymbol{u}_m - \tilde{\boldsymbol{u}}_m$. The error used in the time controller is

$$er = \sup_{i=1,\dots,n} \frac{\|w_{i,m}\|_h}{\|u_{i,m}\|_h}, \tag{4}$$

where $\|\cdot\|_h$ refers to the mesh norm. Note that the error is the worst relative error among the components of $\boldsymbol{u}$.

3.1. Newton method and preconditioning

Since the implicit Runge-Kutta scheme requires to solve nonlinear equations, we use Newton method using always as initial guess $\boldsymbol{u}_m$. As the error of the stopping criteria we use again equation (4), where $\boldsymbol{w}_m$ is replaced by the difference between two successive Newton iterations. The tolerance is fixed as $10^{-2}\varepsilon$, where ε is the time tolerance.

The linear system corresponding to each Newton iteration is solved with the GMRES solver. Multigrid preconditioning is used to alleviate the computational cost of solving linear systems. The preconditioner is calculated once every four or five time steps. We implement a variable V-cycle; details can be found in Kanschat (2004). To track the performance of the preconditioning, we calculate the average of the number of GMRES iterations for the first iteration in the Newton method.

4. NUMERICAL STUDIES

All calculations were carried out on a processor intel5 2.40 GHz×4, 3.7 GB RAM Ubuntu PC. The code was written using a C++ FEM library called deal.II (Bangerth et al., 2007).

4.1. One-step Reaction

We test the numerical method with the one-step reaction $2X \to P$ with a source term f. The resulting differential equation is solved on a square domain of unit side-length. If the concentration of X is u we get

$$\dot{u} - a\Delta u + u^2 = f,$$

where $a = 10^{-2}$ and

$$f(x,y,t) = -3\pi^2 a e^{-2\pi^2 at} + (\sin(\pi x)\sin(\pi y) + 1{,}5)^2 e^{-4\pi^2 at}.$$

We impose homogeneous Neumann conditions and compare against the exact solution $w(x,y,t) = (\sin(\pi x)\sin(\pi y)+1{,}5)e^{-2\pi^2 at}$. Calculations are performed with different time tolerances and uniform refinement of the mesh, halving the length of the sides of the quadrilaterals in the mesh.
Several errors together with the computed rates of converge are displayed in Table 1. As regards the gradient of the approximated solutions, the error due to the space discretization clearly dominates and the improvement by lowering the time tolerance is small. Nevertheless, polynomials of degree two (P^2) are much better than polynomials of degree one (P^1), lowering at least 100 times the error due to the mesh. In general the convergence of the mesh-depending norm and H^1_{semi} semi-norm agrees with the theoretical order of convergence (Ciarlet, 1979; Di Pietro and Ern, 2012).

Table 1: Error of the solution of the one-step reaction in the mesh-depending norm, L^2, H^1 seminorm and L^∞, together with the order of convergence (Ciarlet, 1979; Di Pietro and Ern, 2012). The errors are computed at the final time $t = 10$

cells	steps	H^1_{mesh}/ %		L^2		H^1_{semi}		L^∞	
time tolerance = 1e-3, space P^1									
256	8	4.79	–	2.37e-4	–	1.75e-2	–	9.70e-4	–
1024	8	2.35	1.03	1.72e-4	0.46	8.75e-3	1.00	4.39e-4	1.14
4096	8	1.17	1.01	1.58e-4	0.13	4.38e-3	1.00	2.73e-4	0.69
time tolerance = 1e-4, space P^1									
256	11	4.79	–	1.75e-4	–	1.75e-2	–	8.14e-4	–
1024	11	2.35	1.03	4.65e-5	1.91	8.75e-3	1.00	2.17e-4	1.91
4096	11	1.16	1.01	1.78e-5	1.39	4.37e-3	1.00	6.71e-5	1.69
time tolerance = 1e-5, space P^1									
256	16	4.79	–	1.74e-4	–	1.75e-2	–	7.89e-4	–
1024	15	2.35	1.03	4.34e-5	2.00	8.75e-3	1.00	2.00e-4	1.98
4096	16	1.16	1.01	1.10e-5	1.98	4.37e-3	1.00	5.27e-5	1.92
time tolerance = 1e-3, space P^2									
256	8	2.63e-1	–	1.77e-4	–	5.56e-4	–	2.75e-4	–
1024	8	1.37e-1	0.94	2.03e-4	-0.20	3.63e-4	0.62	3.48e-4	-0.34
4096	8	1.01e-1	0.43	1.96e-4	0.06	2.99e-4	0.28	3.27e-4	0.09
time tolerance = 1e-4, space P^2									
256	10	2.31e-1	–	1.59e-5	–	4.62e-4	–	2.69e-5	–
1024	10	5.87e-2	1.98	1.35e-5	0.23	1.19e-4	1.96	2.00e-5	0.43
4096	10	1.74e-2	1.74	1.49e-5	-0.14	3.76e-5	1.66	2.32e-5	-0.22
time tolerance = 1e-5, space P^2									
256	15	2.29e-1	–	3.26e-6	–	4.57e-4	–	7.34e-6	–
1024	15	5.74e-2	2.00	1.64e-6	0.99	1.15e-4	2.00	2.81e-6	1.38
4096	16	1.46e-2	1.98	1.86e-6	-0.19	2.93e-5	1.97	3.11e-6	-0.14

The error in the L^2 and L^∞ norms is much lower than the error of the gradient. For elements P^1, the order of convergence is closer to the theoretical order as the time tolerance decreases. On the other hand, for P^2 the time error seems to dominate, so the order of convergence is outside the expected. We conclude so that the elements P^2 are more reliable, even for coarse meshes, and thus we can increase exactitude without excessive computational cost.

Low time tolerance requires more steps, however Table 1 shows that the increase in the number of steps is small enough to use low time tolerance. Furthermore the lower the time tolerance, the less the number of GMRES iterations, because of the decrease of the step length.

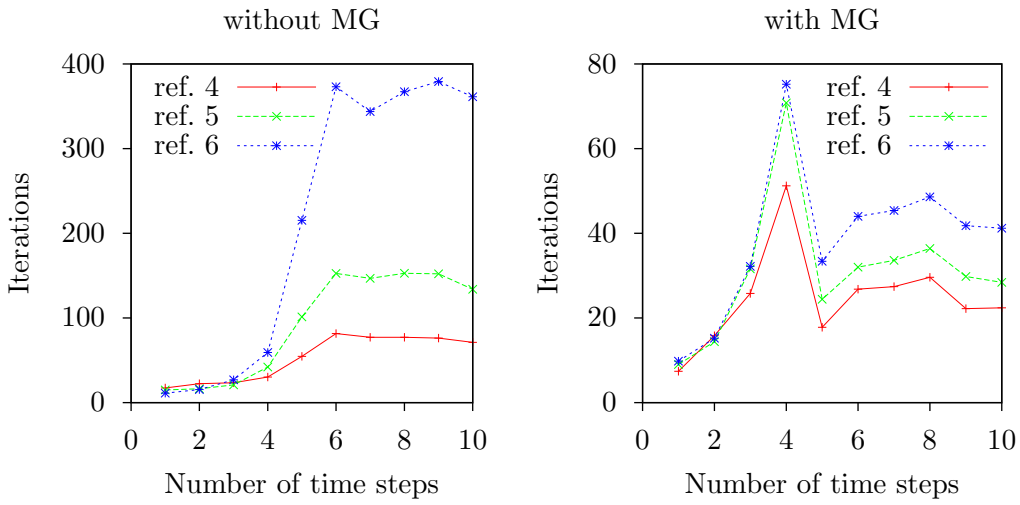


Figure 1: Number of GMRES iterations with and without multigrid preconditioning. The refinement (ref. n) refers to the number of cells 2^n in the triangulation. The space is P^2 and the time tolerance is 10^{-4}

The preconditioning improves significantly the performance of GMRES, see Figure 1. While a fine mesh requires 400 iterations of GMRES, the preconditioning reduces the iterations to 50 with low computational cost. The preconditioning also avoids a great increase in the number of GMRES iterations as the mesh is finer.

4.2. Brusselator

The Brusselator was the first theoretical model for oscillating chemical reactions and it was developed by the Brussels school to show that oscillations and wave propagation can arise in simple models. The elementary steps of the Brusselator are

$$\begin{aligned} A &\to X \\ 2X + Y &\to 3X \\ B + X &\to Y + C \\ X &\to E. \end{aligned}$$

The concentrations of A, B, C and E are held constant. The concentrations of X and Y are denoted by u_1 and u_2 respectively. The total change of concentration of X is

$$\dot{u}_1 = k_1 c_A + k_2 u_1^2 u_2 - k_3 c_B u_1 - k_4 u_1. \tag{5}$$

In the same fashion, the total change of concentration of Y is

$$\dot{u}_2 = -k_2 u_1^2 u_2 + k_3 c_B u_1. \tag{6}$$

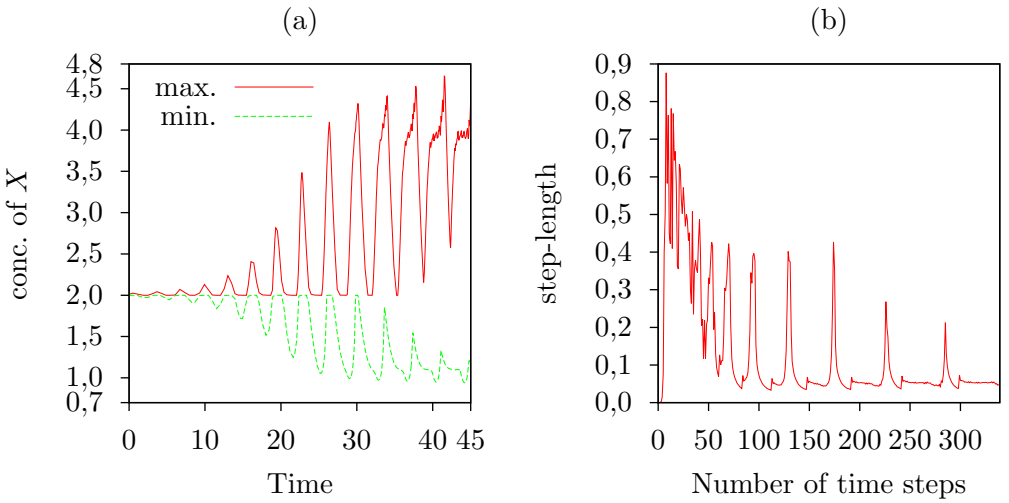


Figure 2: Brusselator with diffusion coefficients $D_1 = 10^{-4}$ and $D_2 = 5 \times 10^{-5}$. The domain is a circle of radius 0.5. (a) Maximum and minimum concentration of X in time. (b) Step-length according to the I-based controller of the time scheme

When coupling the reaction with diffusion, we get the system of equations

$$\begin{aligned} \dot{u_1} - D_1\Delta u_1 - k_1 c_A - k_2 u_1^2 u_2 + k_3 c_B u_1 + k_4 u_1 &= 0 \\ \dot{u_2} - D_2\Delta u_2 + k_2 u_1^2 u_2 - k_3 c_B u_1 &= 0, \end{aligned}$$

with no-flux boundary conditions. The rate constants used in calculations are given in Table 2.

Table 2: Constants used in the calculations.

$k_1 c_A$	k_2	$k_3 c_B$	k_4	D_1/D_2
2	5.45	1	1	2

The system is unstable if $k_2 > 1 + (k_1 c_A)^2$ and the steady state is $u_{1,s} = k_2$ and $u_{2,s} = k_1 c_A / k_2$. To achieve easily oscillations, the initial concentrations are chosen to be a perturbed steady state of the system, for example in the Figure 3 the initial conditions are

$$\begin{aligned} u_1^0 &= u_{1,s} + 10^{-2} e^{-100(x^2+y^2)} \\ u_2^0 &= u_{2,s} - 10^{-3} e^{-100((x-1)^2+(y-0{,}8)^2)}. \end{aligned}$$

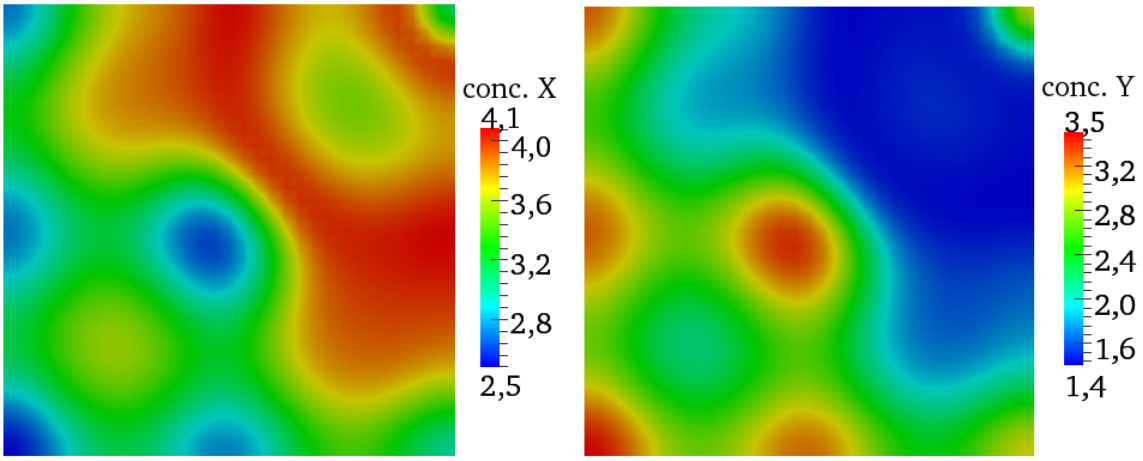
This perturbation of the steady state is implemented at some distance from the boundary in order to match the Neumann boundary condition.

The Figure 2(a) shows the unstable behavior of the system. The concentration begins to oscillate even in early stages of the reaction and the amplitude of the oscillations increases up to a seemly

constant value. The Figure 2(b) shows that the step-length is long at the beginning, while the concentration is approximately homogeneous, but the step-length tends to oscillate along with the concentration. The diffusion coefficients of the system in the Figure 3 are relatively high, therefore diffusive processes dominate and there are poor spatial patterns, as if there was a long wave. When the diffusion coefficients are sufficiently small, the system generates waves as the Figure 4(a) shows for a system perturbed in the center of a circular domain.

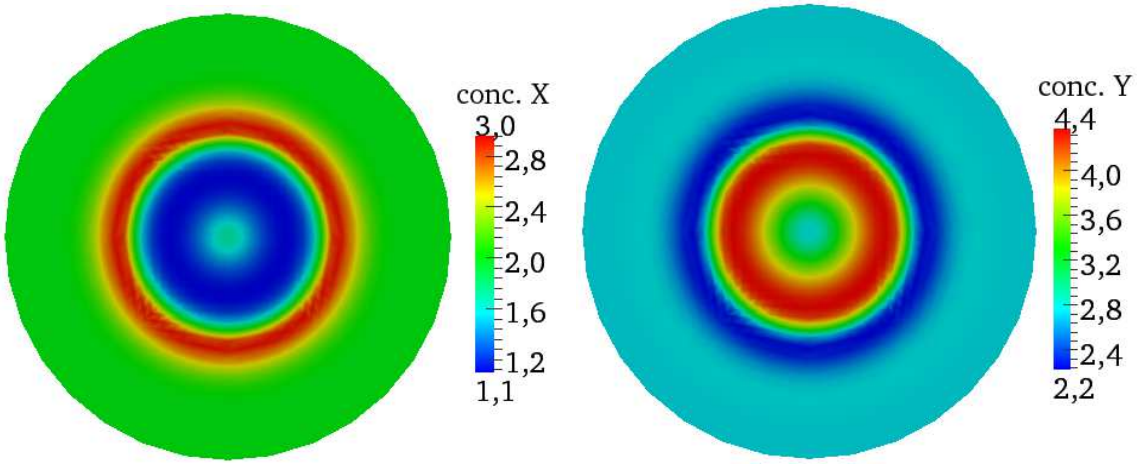
5. CONCLUSIONES

The discontinuous Galerkin method are adequate for the simulation of oscillating systems from the modelling of oscillating chemical reactions. We recall that traditional conforming finite element schemes may not capture oscillations due to instabilities that can be overcome by some special stabilization procedures (e.g., moving mesh, hp–adaptivity) or by using a small time step and a very fine mesh. Our numerical studies show that our implementation is stable. Our experiments also show that singly diagonally Runge-Kutta scheme is sufficient to deal with the stiffness of some oscillating unstable systems as the Brusselator. The multigrid preconditioner reduces greatly the number of GMRES iterations with low computational cost, so improving the performance of the method. The one-step reaction shows the advantage of working with polynomials of degree two and how small error can be achieved in L^2 norm. The embedded time scheme saves computation as it allows to enlarge the time step for almost flat regions of the solution.



(a) Substance X (b) Substance Y

Figure 3: Brusselator with diffusion coefficients $D_1 = 10^{-2}$ and $D_2 = 5 \times 10^{-3}$. The domain is a square with sides of length 3. Time 57.0



(a) Substance X (b) Substance Y

Figure 4: Brusselator with diffusion coefficients $D_1 = 10^{-4}$ and $D_2 = 5 \times 10^{-5}$. The domain is a circle of radius 0.5. Time 39.1

ACKNOWLEDGMENTS

The authors thank the reviewers for the suggestions that helped to improve the paper.

APPENDIX A Butcher tableau

Runge-Kutta scheme ARK4(3)-ESDIRK

0						
$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$				
$\frac{83}{250}$	$\frac{8611}{62500}$	$-\frac{1743}{31250}$	$\frac{1}{4}$			
$\frac{31}{50}$	$\frac{5012029}{34652500}$	$-\frac{654441}{2922500}$	$\frac{174375}{388108}$	$\frac{1}{4}$		
$\frac{17}{20}$	$\frac{15267082809}{155376265600}$	$-\frac{71443401}{120774400}$	$\frac{730878875}{902184768}$	$\frac{2285395}{8070912}$	$\frac{1}{4}$	
1	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$
	$\frac{82889}{524892}$	0	$\frac{15625}{83664}$	$\frac{69875}{102672}$	$-\frac{2260}{8211}$	$\frac{1}{4}$
	$\frac{4586570599}{29645900160}$	0	$\frac{178811875}{945068544}$	$\frac{814220225}{1159782912}$	$-\frac{3700637}{11593932}$	$\frac{61727}{225920}$

Referencias

Bangerth, W.; Hartmann, R. & Kanschat, G. (2007), deal.II – general purpose object–oriented finite element library. *ACM Trans. Math. Softw.*, 33(4), 24/1–24/27. Disponible en: http://doi.acm.org/10.1145/1268776.1268779

Ciarlet, P. G. (1979), *The finite element method for elliptic problems*. North-Holland, 2nd edition.

Epstein, I. R. & Pojman, J. A. (1998), *An introduction to nonlinear chemical dynamics*. Oxford University Press.

Field, R. & Noyes, R. (1974), Oscillations in chemical systems IV. Limit cycle behavior in a model of a real chemical reaction. *J. Chem. Phys.*, 60, 1877-1884. Disponible en: http://dx.doi.org/10.1063/1.1681288

Garzón-Alvarado, D.; Galeano, C. & Mantilla, J. (2011), Turing pattern formation for reaction-convection-diffusion systems in fixed domains submitted to toroidal velocity fields. *Applied Mathematical Modelling*, 35(10), 4913–4925. Disponible en: doi:10.1016/j.apm.2011.03.040

Kanevsky, A.; Carpenter, M. H.; Gottlieb, D. & Hesthaven, J. S. (2007), Application of implicit–explicit high order Runge-Kutta methods to discontinuous-Galerkin schemes. *J. Comput. Phys.*, 225(2), 1753–1781. Disponible en: doi:10.1016/j.jcp.2007.02.021

Kanschat, G. (2004), Multi-level methods for discontinuous Galerkin FEM on locally refined meshes. *Comput. & Struct.*, 82, 2437–2445. Disponible en: doi:10.1016/j.compstruc.2004.04.015

De Kepper, P.; Epstein, I. R.; Orban, M. and Kustin, K. (1982), Batch oscillations and spatial wave patterns in chlorite oscillating systems. *Phys. Chem.*, 86, 170-171. Disponible en: doi:10.1021/j100391a007

Lengyel, I. and Epstein, I. R. (1991), Modeling of turing structures in the chlorite–iodide–malonic Acid–starch reaction system. *Science*, 251(4994), 650–652.

Michoski, C. E.; Evans, J. A. and Schmitz, P. G. (2010), Modeling chemical reactors I: Quiescent reactors. *arXiv:1012.5682v1 [physics.chem-ph]*.

Di Pietro, D. A. and Ern, A. (2012), *Mathematical aspects of discontinuous Galerkin methods*. Springer.

Prigogine, I. and Lefever, R. (1969), Symmetry breaking instabilities in dissipative systems II. *J. Chem. Phys.*, 48, 1695–1700. Diponible en: doi.org/10.1063/1.1668896

Reed, W. H. and Hill, T. R. (1973), Triangular mesh methods for neutron transport equation. Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory.

Schnakenberg, J. (1979), Simple chemical reaction system with limit cycle behavior. *J. Theor. Biol.*, 81, 389–400. Diponible en: doi:10.1016/0022-5193(79)90042-0

Turing, A. (1952), The chemical basis of morphogenesis. *Phil. Trans. R. Soc. B*, 237(641), 37–72. Disponible en: doi:10.1098/rstb.1952.0012

Zhu, J.; Zhang, Y. T.; Newman, S. A. and Alber, M. (2009), Application of discontinuous Galerkin methods for reaction-diffusion systems in developmental biology. *J. Sci. Comput.*, 40, 391–418. Disponible en: doi:10.1007/s10915-008-9218-4