New strategies for multi-scale reaction wave : splitting methods coupled with space adaptative multiresolution and parareal algorithm

S. Descombes [1] M. Duarte[2] T. Dumont [3]
V. Louvet [3] M. Massot [2] C. Tenaud [4]

[1] Laboratoire J. A. Dieudonné - Nice - France

[2] EM2C - Ecole Centrale Paris - France

[3] ICJ - Université Claude Bernard Lyon 1 - France

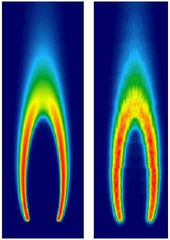
Outline

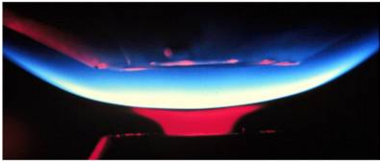
Outline

Application Background

Numerical simulation of unsteady reactive phenomena

- Flames (Instabilities, dynamics, pollutants)





- Chemical “waves” (spiral waves, scroll waves)
- Biochemical Engineering (migraines, Rolando’s region, strokes : clinical anomaly which follows an anatomic lesion of one of many cerebral blood vessels)

Application Background

Numerical simulation of unsteady reactive phenomena

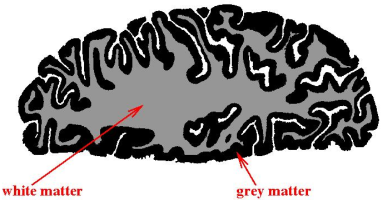
- Flames (Instabilities, dynamics, pollutants)
- Chemical "waves" (spiral waves, scroll waves)

- Biochemical Engineering (migraines, Rolando's region, strokes : clinical anomaly which follows an anatomic lesion of

Application Background

Numerical simulation of unsteady reactive phenomena

- Flames (Instabilities, dynamics, pollutants)
- Chemical “waves” (spiral waves, scroll waves)
- Biochemical Engineering (migraines, Rolando’s region, strokes : clinical anomaly which follows an anatomic lesion of one of many cerebral blood vessels)



To summarize

Dynamics involving many "species" and "reactions"

Multiple scales problems

"Complex Chemistry"

Convection-diffusion coupled to chemistry

$$\partial_t U + \sum \partial_i \big(\Phi_i(U, \partial_{\boldsymbol{x}} U)\big) = \Omega(U)$$

Examples

- KPP or Fischer equation

$$\partial_t \beta - \partial_{xx} \beta = \beta^2 (1 - \beta)$$

- Belousov-Zhabotinsky system of equations
- Compressible flame equations with complex chemistry
- In both cases : Low diffusion ($\varepsilon \Delta$)

Examples

- KPP or Fischer equation
- Belousov-Zhabotinsky system of equations

$$\left\{\begin{array}{rcl}\dfrac{\partial a}{\partial \tau} - D_a \Delta a & = & \dfrac{1}{\mu}(-qa - ab + fc), \\ \dfrac{\partial b}{\partial \tau} - D_b \Delta b & = & \dfrac{1}{\epsilon}\left(qa - ab + b(1-b)\right), \\ \dfrac{\partial c}{\partial \tau} - D_c \Delta c & = & b - c,\end{array}\right.$$

- Compressible flame equations with complex chemistry
- In both cases : Low diffusion ($\varepsilon\Delta$)

Examples

- KPP or Fischer equation
- Belousov-Zhabotinsky system of equations
- Compressible flame equations with complex chemistry
- In both cases : Low diffusion ($\varepsilon\Delta$)

Examples

- KPP or Fischer equation
- Belousov-Zhabotinsky system of equations
- Compressible flame equations with complex chemistry
- In both cases : Low diffusion ($\varepsilon\Delta$)

Outline

Strategies

Resolving the large scale spectrum coupled

- Explicit methods in time (high order in space)
- Fully implicit methods with adaptative time stepping
- Method of lines coupled to a stiff ODE solver
- Semi-implicit methods (IMEX, source/diffusion explicit in time)

The computational cost and memory requirement have suggested the study of alternative methods : decoupling

- Reduction of chemistry (large litterature)
- Operator Splitting techniques

Outline

Operator splitting techniques

Operator splitting : separate convection-diffusion and chemistry

- High order methods exist
- Allow the use of dedicated solver for each step
- Yield lower storage and optimization capability

Basis of operator splitting - I

Reaction-diffusion system to be solved (t : time interval)

$$U(t) = T^t U_0 \quad \begin{cases} \partial_t U - \Delta U = \Omega(U) \\ U(0) = U_0 \end{cases}$$

Two elementary “blocks”.

$$V(t) = X^t V_0 \quad \begin{cases} \partial_t V - \Delta V = 0 \\ V(0) = V_0 \end{cases}$$

$$W(t) = Y^t W_0 \quad \begin{cases} \partial_t W = \Omega(W) \\ W(0) = W_0 \end{cases}$$

Basis of operator splitting II

First order methods :

Lie Formulae.

$$L_1^t\, U_0 = X^t\, Y^t\, U_0 \qquad L_1^t\, U_0 - T^t\, U_0 = O(t^2),$$

$$L_2^t\, U_0 = Y^t\, X^t\, U_0 \qquad L_2^t\, U_0 - T^t\, U_0 = O(t^2),$$

Basis of operator splitting II

First order methods :

Lie Formulae.

$$L_1^t\, U_0 = X^t\, Y^t\, U_0 \qquad L_1^t\, U_0 - T^t\, U_0 = O(t^2),$$

$$L_2^t\, U_0 = Y^t\, X^t\, U_0 \qquad L_2^t\, U_0 - T^t\, U_0 = O(t^2),$$

$\square$

Basis of operator splitting III

Second order methods :

Strang Formulae.

$$S_1^t\, U_0 = Y^{t/2}\, X^t\, Y^{t/2}\, U_0 \qquad S_1^t\, U_0 - T^t\, U_0 = O(t^3),$$

$$S_2^t\, U_0 = X^{t/2}\, Y^t\, X^{t/2}\, U_0 \qquad S_2^t\, U_0 - T^t\, U_0 = O(t^3),$$

□

Higher order...

Basis of operator splitting III

Second order methods :

Strang Formulae.

$$S_1^t\, U_0 = Y^{t/2}\, X^t\, Y^{t/2}\, U_0 \qquad S_1^t\, U_0 - T^t\, U_0 = O(t^3),$$

$$S_2^t\, U_0 = X^{t/2}\, Y^t\, X^{t/2}\, U_0 \qquad S_2^t\, U_0 - T^t\, U_0 = O(t^3),$$

$\square$

Higher order...

Basis of operator splitting III

Second order methods :

Strang Formulae.

$$S_1^t\, U_0 = Y^{t/2}\, X^t\, Y^{t/2}\, U_0 \qquad S_1^t\, U_0 - T^t\, U_0 = O(t^3),$$

$$S_2^t\, U_0 = X^{t/2}\, Y^t\, X^{t/2}\, U_0 \qquad S_2^t\, U_0 - T^t\, U_0 = O(t^3),$$

$\square$

Higher order...

Error estimate

Error estimate -> Lie formalism. For an ODE $\dot{y} = f_1(y)$, we denote by φ_1^t the exact solution, we introduce the differential operator (Lie derivative)

$$D_1 = \sum_j f_{1,j} \frac{\partial}{\partial y_j}.$$

For a smooth function from $\mathbb{R}^n$ to $\mathbb{R}^n$, we have

$$\frac{d}{dt} F\left(\varphi_1^t(y_0)\right) = F'\left(\varphi_1^t(y_0)\right) f_1\left(\varphi_1^t(y_0)\right) = (D_1 F)\left(\varphi_1^t(y_0)\right)$$

Error estimate

By iterations, the Taylor's expansion of $F\left(\varphi_1^t(y_0)\right)$ in $t = 0$ gives (formally)

$$F\left(\varphi_1^t(y_0)\right) = \sum_{k\geq 0} \frac{t^k}{k!}\left(D_1^k F\right)(y_0) = e^{tD_1} F(y_0).$$

With $F =$Id, we obtain

$$\varphi_1^t(y_0) = e^{tD_1}\mathrm{Id}(y_0).$$

Error estimate

Moreover, if we introduce a second flow φ_2^t, we have :

$$\left(\varphi_2^t \varphi_1^t\right)(y_0) = e^{tD_1} e^{tD_2} \mathrm{Id}(y_0).$$

Error estimate

Then if we denote by φ_3^t the exact solution of $\dot{y} = (f_1 + f_2)(y)$, we have the following relation :

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = e^{t(D_1+D_2)}\mathrm{Id}(y_0) - e^{tD_1}e^{tD_2}\mathrm{Id}(y_0),$$

we then work with linear operators ! For example, for two linear operators A et B, we have

$$e^{t(A+B)} - e^{tA}e^{tB} = \frac{t^2}{2}[A, B] + O(t^3),$$

this yields,

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = \frac{t^2}{2}[D_1, D_2]\mathrm{Id}(y_0) + O(t^3),$$

Error estimate

Then if we denote by φ_3^t the exact solution of $\dot{y} = (f_1 + f_2)(y)$, we have the following relation :

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = e^{t(D_1+D_2)}\mathrm{Id}(y_0) - e^{tD_1}e^{tD_2}\mathrm{Id}(y_0),$$

we then work with linear operators ! For example, for two linear operators A et B, we have

$$e^{t(A+B)} - e^{tA}e^{tB} = \frac{t^2}{2}[A, B] + O(t^3),$$

this yields,

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = \frac{t^2}{2}[D_1, D_2]\mathrm{Id}(y_0) + O(t^3),$$

Error estimate

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = \frac{t^2}{2}[D_1, D_2]\mathrm{Id}(y_0) + O(t^3),$$

and $[D_1, D_2]$ is now a Lie bracket...

$$[D_1, D_2] = \sum_i \left(\sum_j \left(\frac{\partial f_{1,i}}{\partial y_j} f_{2,j} - \frac{\partial f_{2,i}}{\partial y_j} f_{1,j} \right) \right) \frac{\partial}{\partial y_i}$$

We are not limited to the finite dimension...

$$\left(D_1 = \sum_j f_{1,j} \frac{\partial}{\partial y_j} \right)$$

Error estimate

$$\varphi_3^t(y_0) - \left(\varphi_2^t \varphi_1^t\right)(y_0) = \frac{t^2}{2}[D_1, D_2]\mathrm{Id}(y_0) + O(t^3),$$

and $[D_1, D_2]$ is now a Lie bracket...

$$[D_1, D_2] = \sum_i \left(\sum_j \left(\frac{\partial f_{1,i}}{\partial y_j} f_{2,j} - \frac{\partial f_{2,i}}{\partial y_j} f_{1,j} \right) \right) \frac{\partial}{\partial y_i}$$

We are not limited to the finite dimension...

$$\left(D_1 = \sum_j f_{1,j} \frac{\partial}{\partial y_j} \right)$$

Error estimate

Application to Lie et Strang formulae denoting by F the reaction term for a scalar equation.

$$T^t u_0 - Y^t X^t u_0 = \frac{t^2}{2} F''(u_0)(\partial_x u_0)^2 + O(t^3),$$

$$T^t u_0 - Y^{t/2} X^t Y^{t/2} u_0 =$$
$$\frac{t^3}{24}\left(2F^{(4)}(u_0)(\partial_x u_0)^4 + 8F^{(3)}(u_0)(\partial_x u_0)^2(\partial_{xx} u_0) + 4F''(u_0)(\partial_{xx} u_0)^2\right)$$
$$-\frac{t^3}{24}\left(\left(F(u_0)F^{(3)}(u_0) + F''(u_0)F'(u_0)\right)(\partial_x u_0)^2\right) + O(t^4)$$

Error estimate

Application to Lie et Strang formulae denoting by F the reaction term for a scalar equation.

$$T^t u_0 - Y^t X^t u_0 = \frac{t^2}{2} F''(u_0)(\partial_x u_0)^2 + O(t^3),$$

$$T^t u_0 - Y^{t/2} X^t Y^{t/2} u_0 =$$
$$\frac{t^3}{24}\left(2F^{(4)}(u_0)(\partial_x u_0)^4 + 8F^{(3)}(u_0)(\partial_x u_0)^2(\partial_{xx} u_0) + 4F''(u_0)(\partial_{xx} u_0)^2\right)$$
$$-\frac{t^3}{24}\left(\left(F(u_0)F^{(3)}(u_0) + F''(u_0)F'(u_0)\right)(\partial_x u_0)^2\right) + O(t^4)$$

Stiffness comes into play

- Detected by the beginning of 90'
  (Hairer Wanner 91, D'Angelo Larrouturou 95)
- Numerical analysis of linear model ODEs
  (Verwer Sportisse 00)

Various origins of stiffness

- Large spectrum of temp. scales in chemical source
- Large spatial gradients of the solutions

Stiffness comes into play

- Detected by the beginning of 90'
  (Hairer Wanner 91, D'Angelo Larrouturou 95)
- Numerical analysis of linear model ODEs
  (Verwer Sportisse 00)

Various origins of stiffness

- Large spectrum of temp. scales in chemical source
- Large spatial gradients of the solutions

Large spectrum of temporal scales

- A “model” problem for the fast scales for $U^\varepsilon = (u^\varepsilon, v^\varepsilon)^t$

$$\begin{cases} \partial_t u^\varepsilon - \partial_x \cdot \left(B^u(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon\right) &= f(u^\varepsilon, v^\varepsilon), \quad x \in \mathbb{R}^d \\ \partial_t v^\varepsilon - \partial_x \cdot \left(B^v(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon\right) &= \dfrac{g(u^\varepsilon, v^\varepsilon)}{\varepsilon}, \quad x \in \mathbb{R}^d \end{cases}$$

- The entropic structure of the RD system of equations $\Rightarrow$ Dynamics on the partial equilibrium manifold

$$\partial_t u - \partial_x \cdot \left(B^u(u, h(u))\, \partial_x \begin{pmatrix} u \\ h(u) \end{pmatrix}\right) = f(u, h(u))$$

- Order reduction due to fast scales
  - Diag. diffusion : Lie RD order 0 fast variable only
  - Diag. diffusion : Strang DRD order 0 fast variable only
  - Non-diag. diffusion : Lie DR and RD order 0

Large spectrum of temporal scales

- A "model" problem for the fast scales for $U^\varepsilon = (u^\varepsilon, v^\varepsilon)^t$

$$\begin{cases} \partial_t u^\varepsilon - \partial_x \cdot \left(B^u(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon\right) = f(u^\varepsilon, v^\varepsilon), & x \in \mathbb{R}^d \\ \partial_t v^\varepsilon - \partial_x \cdot \left(B^v(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon\right) = \dfrac{g(u^\varepsilon, v^\varepsilon)}{\varepsilon}, & x \in \mathbb{R}^d \end{cases}$$

- The entropic structure of the RD system of equations $\Rightarrow$ Dynamics on the partial equilibrium manifold

$$\partial_t u - \partial_x \cdot \left(B^u(u, h(u))\, \partial_x \begin{pmatrix} u \\ h(u) \end{pmatrix} \right) = f(u, h(u))$$

- Order reduction due to fast scales
  - Diag. diffusion : Lie RD order 0 fast variable only
  - Diag. diffusion : Strang DRD order 0 fast variable only
  - Non-diag. diffusion : Lie DR and RD order 0

Large spectrum of temporal scales

- A "model" problem for the fast scales for $U^\varepsilon = (u^\varepsilon, v^\varepsilon)^t$

$$\begin{cases} \partial_t u^\varepsilon - \partial_x \cdot (B^u(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon) & = f(u^\varepsilon, v^\varepsilon), \quad x \in \mathbb{R}^d \\ \partial_t v^\varepsilon - \partial_x \cdot (B^v(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon) & = \dfrac{g(u^\varepsilon, v^\varepsilon)}{\varepsilon}, \quad x \in \mathbb{R}^d \end{cases}$$

- The entropic structure of the RD system of equations $\Rightarrow$ Dynamics on the partial equilibrium manifold

$$\partial_t u - \partial_x \cdot \left(B^u(u, h(u))\, \partial_x \begin{pmatrix} u \\ h(u) \end{pmatrix} \right) = f(u, h(u))$$

- Order reduction due to fast scales
  - Diag. diffusion : Lie RD order 0 fast variable only
  - Diag. diffusion : Strang DRD order 0 fast variable only
  - Non-diag. diffusion : Lie DR and RD order 0

Large spectrum of temporal scales

- A "model" problem for the fast scales for $U^\varepsilon = (u^\varepsilon, v^\varepsilon)^t$

$$\begin{cases} \partial_t u^\varepsilon - \partial_x \cdot (B^u(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon) & = f(u^\varepsilon, v^\varepsilon), \quad x \in \mathbb{R}^d \\ \partial_t v^\varepsilon - \partial_x \cdot (B^v(u^\varepsilon, v^\varepsilon)\, \partial_x U^\varepsilon) & = \dfrac{g(u^\varepsilon, v^\varepsilon)}{\varepsilon}, \quad x \in \mathbb{R}^d \end{cases}$$

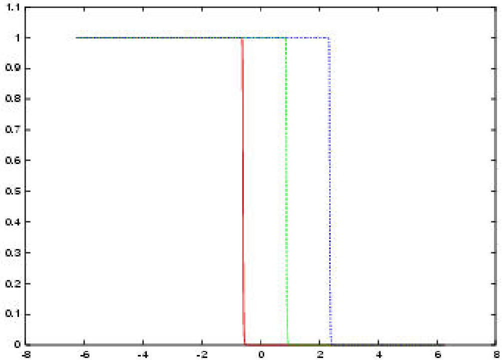
- The entropic structure of the RD system of equations $\Rightarrow$ Dynamics on the partial equilibrium manifold

$$\partial_t u - \partial_x \cdot \left(B^u(u, h(u))\, \partial_x \begin{pmatrix} u \\ h(u) \end{pmatrix} \right) = f(u, h(u))$$

- Order reduction due to fast scales
  - Diag. diffusion : Lie RD order 0 fast variable only
  - Diag. diffusion : Strang DRD order 0 fast variable only
  - Non-diag. diffusion : Lie DR and RD order 0

High spatial gradients

- Initial data with high gradient (L^2 norm)



- High constant in the error estimate $O(t^2) = C(||U_0||_{H^1})\ t^2$
  - Regularizing effect of diffusion

High spatial gradients

- Initial data with high gradient (L^2 norm)
- High constant in the error estimate $O(t^2) = C(||U_0||_{H^1})\ t^2$
  - Regularizing effect of diffusion
  - Example of error estimate for DR

$$|L_1^t - T^t| < C(||U_0||_{L^2}) t^{3/2}$$

  - various asymptotics with a threshold time step
  - Key issue from a numerical point of view

High spatial gradients

- Initial data with high gradient (L^2 norm)
- High constant in the error estimate $O(t^2) = C(||U_0||_{H^1})\ t^2$
  - Regularizing effect of diffusion
  - Example of error estimate for DR

$$|L_1^t - T^t| < C(||U_0||_{L^2})t^{3/2}$$

  - various asymptotics with a threshold time step
  - Key issue from a numerical point of view

Outline

Stability + Accuracy

Considering:

$$y' = \lambda y \quad \Longrightarrow \quad y_{n+1} = R(z) y_n \qquad z = h\lambda$$

We are particularly looking for:

- A-stable methods
- High order methods
- L-stable methods

Implicit Runge Kutta Methods

- Based on Ehle's Methods of type II: (RadauIIA)
- Order: $p = 2s - 1$ (s: stage number)
- A-stable
- L-stable

RADAU5

(Hairer & Wanner Springer-Verlag 91)

- Based on RadauIIA with $s = 3$ and $p = 5$
- Simplified Newton Method $\Longrightarrow$ Linear Algebra tools
- Adaptative time integration step

Explicit Runge-Kutta methods

We want to solve the discrete heat equation

$$\dot{u} = Au,$$

with an explicit s-stage Runge-Kutta method.

Because of the properties of the matrix A, we need to find a stable Runge-Kutta method for the simple problem

$$\dot{u} = \lambda u,$$

with λ real, negative and as big as possible...

ROCK4

(Abdulle SIAM J. Sci. Comput. 02)

- Extended Stability Domain (along $\mathbb{R}^-$) by increasing the number of stages
- Order 4 - Stability $\times s^2$.
- Adaptative time integration step
- Explicit Methods $\Longrightarrow$ NO Linear Algebra problems
- Low Memory Demand

Explicit/Implicit Operator Splitting

Numerical Strategy:

$$\partial_t U - \underbrace{\varepsilon \Delta U}_{\text{ROCK4}} = \underbrace{\Omega(U)}_{\text{RADAU5}}$$

- Reduction in Computational Time
- Reduction in Memory Demand
- Same previous accuracy established by Splitting Scheme
- Highly parallelizable - Diffusion - Reaction

Outline

Background

Consider the general nonlinear system of ODEs:

$$\begin{aligned} \boldsymbol{u}'(t) &= \boldsymbol{f}(\boldsymbol{u}(t)) \\ \boldsymbol{u}(0) &= \boldsymbol{u}^0 \end{aligned}$$

on $t \in (0, T)$ where $\boldsymbol{f} : \mathbb{R}^M \to \mathbb{R}^M$ and $\boldsymbol{u} : \mathbb{R} \to \mathbb{R}^M$.

Decomposition of the Time Direction

We decompose the time domain $\Omega = (0, T)$ into N time subdomains $\Omega_n = (T_n, T_{n+1})$ and consider for $n = 0, 1, \ldots, N-1$:

$$\begin{aligned} \boldsymbol{u}_n'(t) &= \boldsymbol{f}(\boldsymbol{u}_n(t)) \\ \boldsymbol{u}_n(T_n) &= \boldsymbol{U}_n \end{aligned}$$

on $t \in (T_n, T_{n+1})$.

Parareal Algorithm

(Lions *et al.* C. R. Acad. Sci. Paris Sér. I Math. 01)

Combination of two solvers

- Coarse Solver $\Longrightarrow$ fast (sequential calculation)
- Fine Solver $\Longrightarrow$ slow (parallel calculation)

- Convergence from a coarse approximation to the detailed dynamics
- Iterative Method

Parareal Algorithm

The parareal algorithm is based on two propagation operators : $\mathcal{G}^{\Delta T_n}(\mathbf{U})$ and $\mathcal{F}^{\Delta T_n}(\mathbf{U})$, that provide respectively a coarse and an accurate approximation of $\phi^{\Delta T_n}(\mathbf{U})$. In this way, the algorithm starts with an initial approximation $\mathbf{U_n^0}$ given for example by the sequential computation

$$\mathbf{U}_0^0 = \mathbf{u}^0, \quad \mathbf{U}_n^0 = \mathcal{G}^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^0) \; for \; n = 1, \ldots, N,$$

and then performs for $i = 1, \ldots, i_{conv}$ the correction iterations

$$\mathbf{U}_0^i = \mathbf{u}^0, \quad \mathbf{U}_n^i = \mathcal{F}^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^{i-1}) + \mathcal{G}^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^{i}) - \mathcal{G}^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^{i-1})$$

$for n = 1, \ldots, N.$

Parareal Algorithm

Based on the works of Chartier, the time decomposition method can be also interpreted as a multiple shooting method. In fact, considering $\mathbf{U} = (\mathbf{U}_0, \dots, \mathbf{U}_N)^T$ as the unknowns, the system can be written as

$$\mathbf{F}(\mathbf{U}) = \begin{pmatrix} \mathbf{U}_0 - \mathbf{u}^0 \\ \mathbf{U}_1 - \phi^{\Delta T_0}(\mathbf{U}_0) \\ \vdots \\ \mathbf{U}_N - \phi^{\Delta T_{N-1}}(\mathbf{U}_{N-1}) \end{pmatrix} = \mathbf{0}.$$

Parareal Algorithm

Solving this system with Newton's method, leads after a short calculation to

$$\mathbf{U}_0^i = \mathbf{u}^0, \quad \mathbf{U}_n^i = \phi^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^{i-1}) + \frac{\partial \phi^{\Delta T_{n-1}}(\mathbf{U}_{n-1}^{i-1})}{\partial \mathbf{U}_{n-1}^{i-1}} \left(\mathbf{U}_{n-1}^i - \mathbf{U}_{n-1}^{i-1} \right)$$

Outline

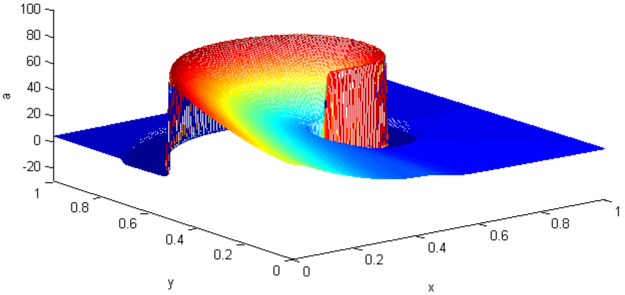
"Toy" Model

Belousov-Zhabotinsky system of equations

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

$$\epsilon = 10^{-2} \quad \mu = 10^{-5} \quad f = 1,6 \quad q = 2.10^{-3}$$
$$D_a = 2,5.10^{-3} \quad D_b = 2,5.10^{-3} \quad D_c = 1,5.10^{-3}$$

"Toy" Model



"Toy" Model - Some results

Grid	129×129		257×257	
Coarse solver	RDR Strang	Rock4	RDR Strang	Rock4
$\mathbf{N_{proc}}$	64			
$\mathbf{N_{proc}/N_{ite}}$	16	32	16	32
$\mathbf{T_{fine}/T_{para}}$	**2.16**	**3.21**	**2.02**	**2.88**

Table: Computation time ratios, 2D BZ

Conclusions

- Convergence rate diminished due to Stiff phenomena
- Parallel speedup is possible, but the speedup is modest
- Appropriate Coarse Solvers $\longrightarrow$ Cheap Stiff Integrators

Outline

Adaptive Multiresolution

(Cohen *et al.* Mathematics of Computation 01)

Principles of the MR:

- Represent a set of function data as values on a coarser grid plus a series of differences at different levels of nested grids.
- The information at consecutive scale levels are related by inter-level transformations: projection and prediction operators.
- The wavelet coefficients are defined as prediction errors, and they retain the detail information when going from a coarse to a finer grid.

Main idea: use the decay of the wavelet coefficients to obtain information on local regularity of the solution.

Multiresolution Transformation

There is a one-to-one correspondence

$$U_j \longleftrightarrow (U_{j-1}, D_j),$$

which can be implemented using the operators P^j_{j-1} and P^{j-1}_j.

By iteration of this decomposition, we obtain a multiscale representation of U_J in terms of $M_J = (U_0, D_1, D_2, \cdots, D_J)$. Using the local structure of the projection and prediction operators, we can implement the multiscale transformation

$$\mathcal{M} : \boldsymbol{U_J} \longmapsto \boldsymbol{M_J}.$$

Compression

Given a set $\Lambda \subset \nabla^J$ of indices λ, 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 typically interested in sets Λ obtained by thresholding: given a set of level-dependent threshold $(\epsilon_0, \epsilon_1, \cdots, \epsilon_J)$, we set

$$\Lambda = \Lambda(\epsilon_0, \epsilon_1, \cdots, \epsilon_J) := \{\lambda s.t. |d_\lambda| \geq \epsilon_{|\lambda|}\}.$$

Applying $\mathcal{T}_\Lambda$ on the multiscale decomposition of U_J amounts to building an approximation $\mathcal{A}_\Lambda U_J$, where the operator $\mathcal{A}_\Lambda$ is given by

$$\mathcal{A}_\Lambda := \mathcal{M}^{-1} \mathcal{T}_\Lambda \mathcal{M}.$$

Outline

1. Context and Motivation
   - Unsteady reactive phenomena
   - Time integration numerical strategies
   - Operator splitting and stiffness
2. Algorithms for multi-scale reaction waves simulation
   - Suitable Stiff Integrators - Parallelization
   - Parallelization of the Time Direction
   - One Illustrating Example
   - Space Adaptive Numerical Method
   - Adaptive Time-Space Numerical Method
   - Some illustrating examples
3. Conclusions

Adaptive Time-Space Numerical Strategy

Refinement Precautionary measure to account for possible translation or presence of finer scales in the solution.

Time Evolution Explicit/Implicit Strang RDR Operator Splitting Method.

Thresholding Wavelet Thresholding Operation:

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

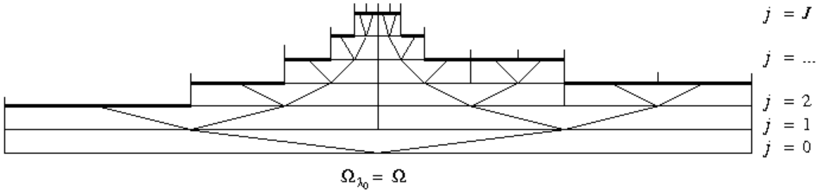
in order to ensure a thresholding error of prescribed order ε ($d =$ spatial dimension).

If $\mathcal{S}^{\Delta t}$ denotes the Strang splitting time integration operator

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

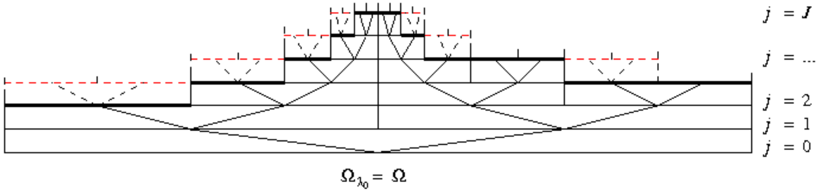
Implementing Configuration

Data Structure : Graded Tree



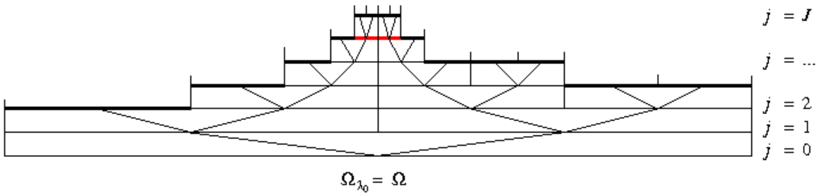
Implementing Configuration

Time Integration : Phantoms



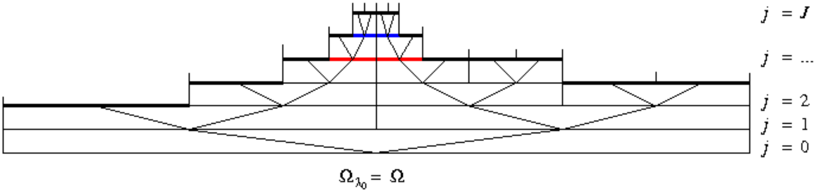
Implementing Configuration

Multiscale Transformation : Projection



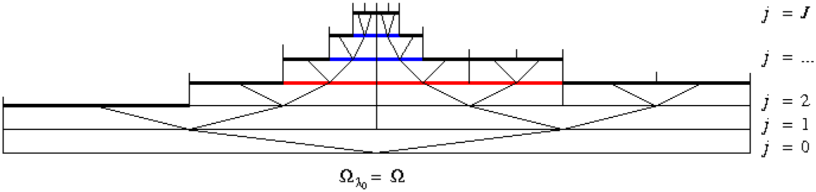
Implementing Configuration

Multiscale Transformation : Projection



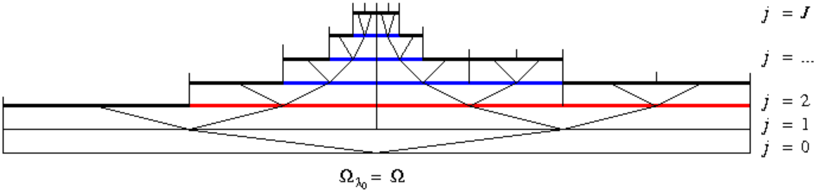
Implementing Configuration

Multiscale Transformation : Projection



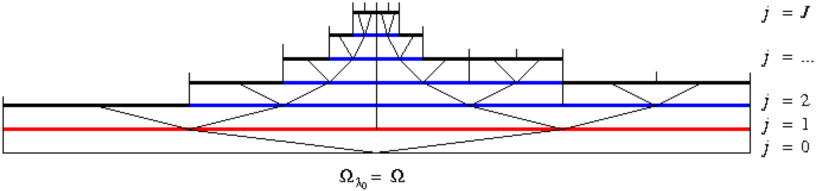
Implementing Configuration

Multiscale Transformation : Projection



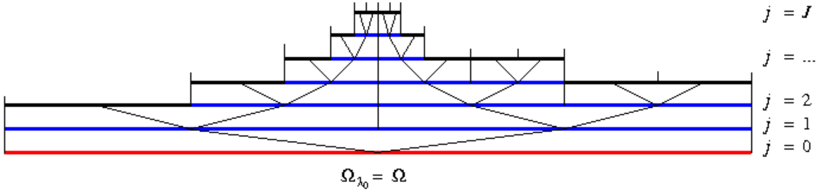
Implementing Configuration

Multiscale Transformation : Projection



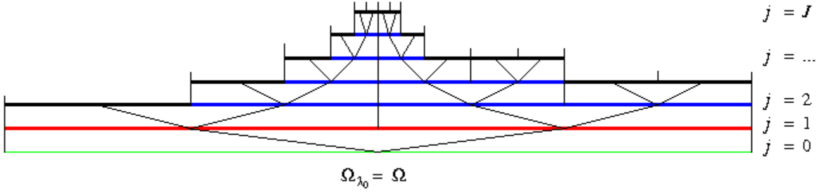
Implementing Configuration

Multiscale Transformation : Projection



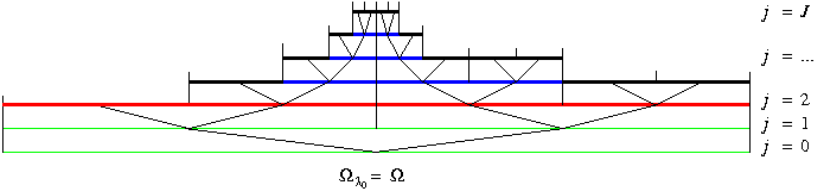
Implementing Configuration

Multiscale Transformation : Prediction



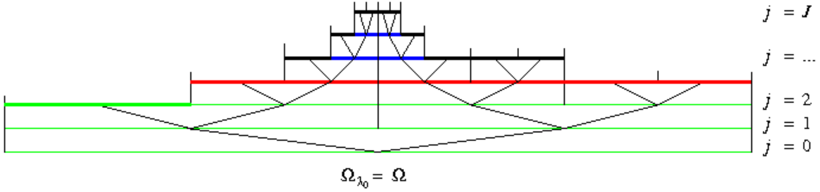
Implementing Configuration

Multiscale Transformation : Prediction



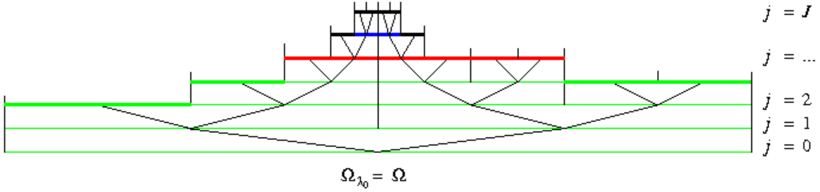
Implementing Configuration

Multiscale Transformation : Prediction



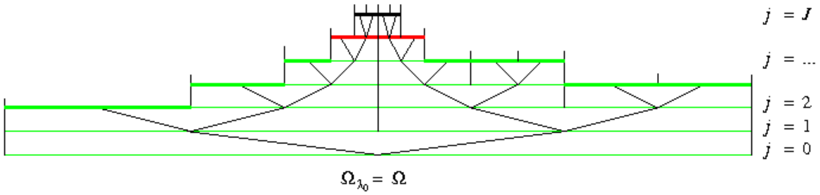
Implementing Configuration

Multiscale Transformation : Prediction



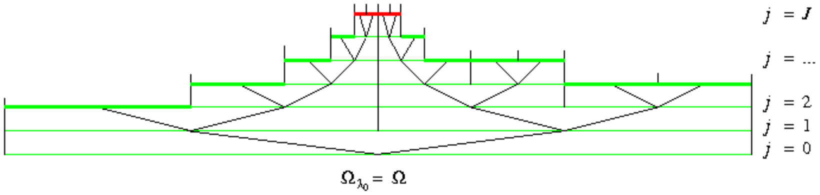
Implementing Configuration

Multiscale Transformation : Prediction



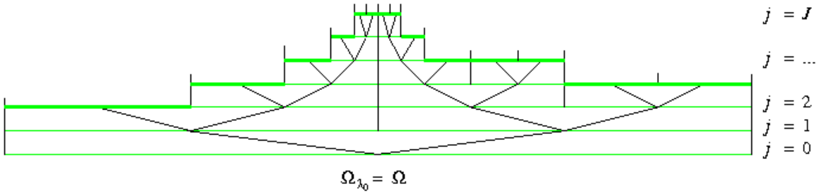
Implementing Configuration

Multiscale Transformation : Prediction



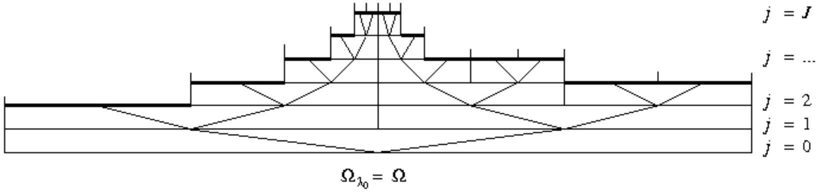
Implementing Configuration

Multiscale Transformation : Prediction



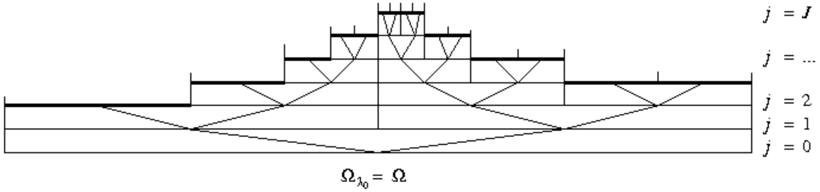
Implementing Configuration

Thresholding & Refinement



Implementing Configuration

Thresholding & Refinement



Outline

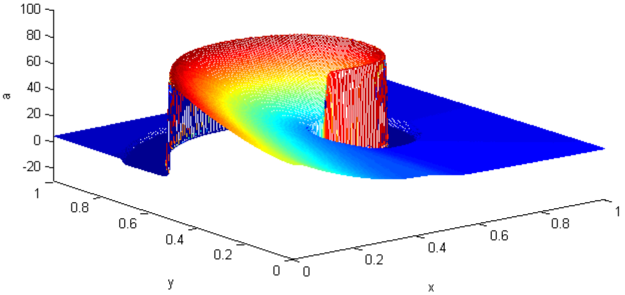
BZ Model

Belousov-Zhabotinsky system of equations

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

$$
\epsilon = 10^{-2} \quad \mu = 10^{-5} \quad f = 1,6 \quad q = 2.10^{-3}
$$
$$
D_a = 2,5.10^{-3} \quad D_b = 2,5.10^{-3} \quad D_c = 1,5.10^{-3}
$$

“Toy” Model

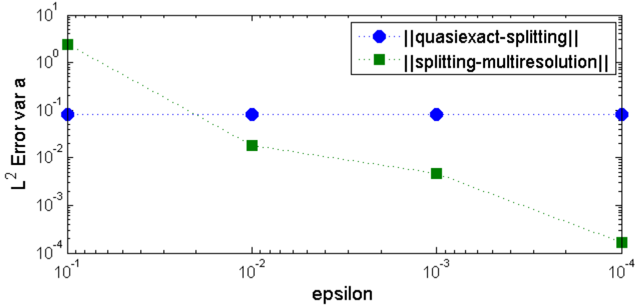


2D Configuration

- Time Domain : $T = [0, 4]$
- Spatial Domain : $\Omega = [0, 1] \times [0, 1]$
- Integration Time Step : $\Delta t = 4/1024$
- Tolerances of Time Integrators: $tol = 1.10^{-5}$
- Finest Grid Level : $J = 10$
- Number of cells at Grid J: $2^{d \times J} = 1048576 = 1024 \times 1024$

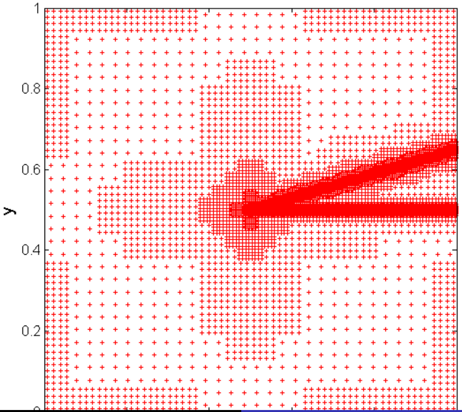
L^2 Error

$$\underbrace{\|u^J_{qe}(t) - u_{MR}(t)\|_{L^2}}_{\textit{Total Error}} \leq \underbrace{\|u^J_{qe}(t) - u^J_{split}(t)\|_{L^2}}_{\textit{Splitting Error}} + \underbrace{\|u^J_{split}(t) - u_{MR}(t)\|_{L^2}}_{\textit{MR Error}}$$



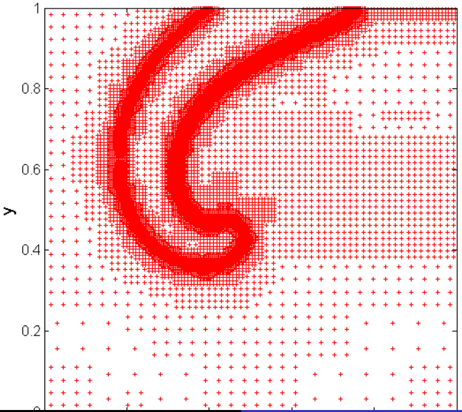
Adaptive Grid

Compression $\longrightarrow$ 1.41%



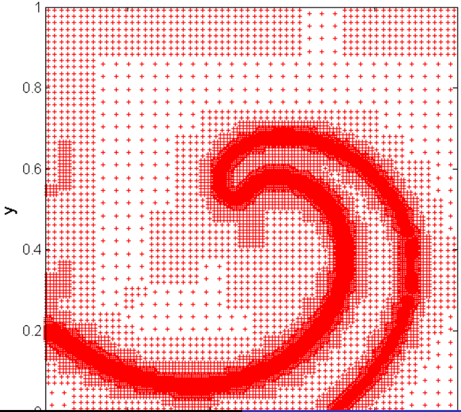
Adaptive Grid

Compression $\longrightarrow$ 2.96%



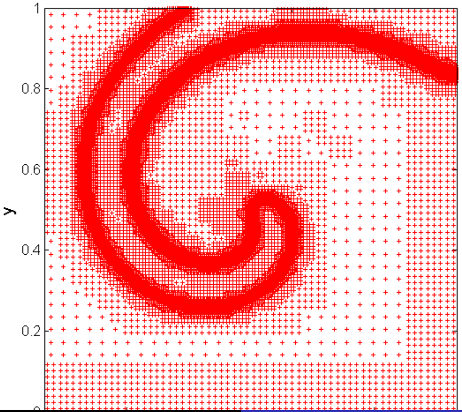
Adaptive Grid

Compression $\longrightarrow$ 3.90%



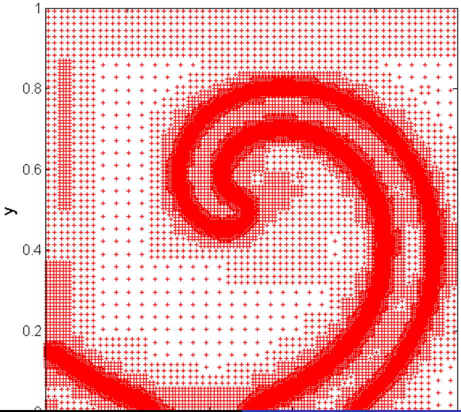
Adaptive Grid

Compression $\longrightarrow$ 4.69%



Adaptive Grid

Compression $\longrightarrow$ 5.77%



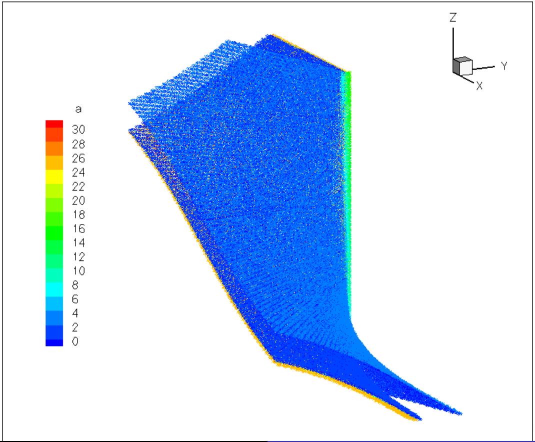
3D Configuration

- Time Domain : $T = [0, 2]$
- Spatial Domain : $\Omega = [0, 1] \times [0, 1] \times [0, 1]$
- Integration Time Step : $\Delta t = 2/256$
- Tolerances of Time Integrators: $tol = 1.10^{-5}$
- Finest Grid Level : $J = 9$
- Number of cells at Grid J:
  $2^{d \times J} = 134217728 = 512 \times 512 \times 512$

3D Configuration

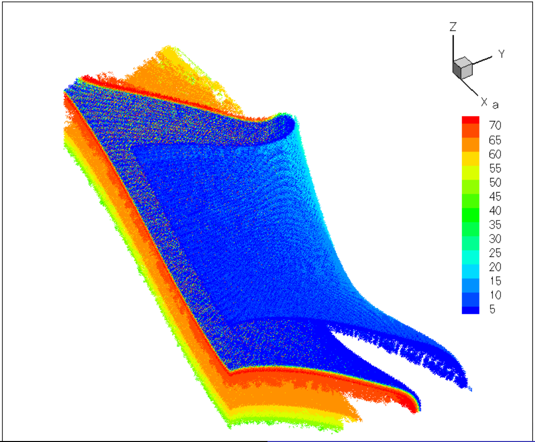
Adaptive Grid

Compression $\longrightarrow$ 4.46%



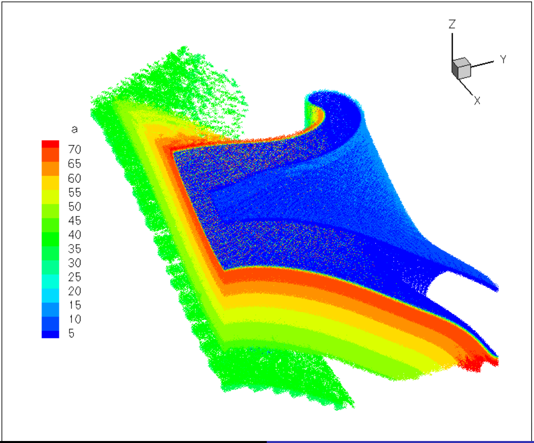
Adaptive Grid

Compression $\longrightarrow$ 10.38%



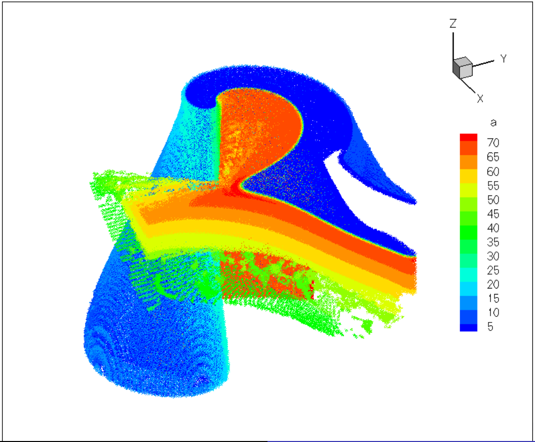
Adaptive Grid

Compression $\longrightarrow$ 17.00%



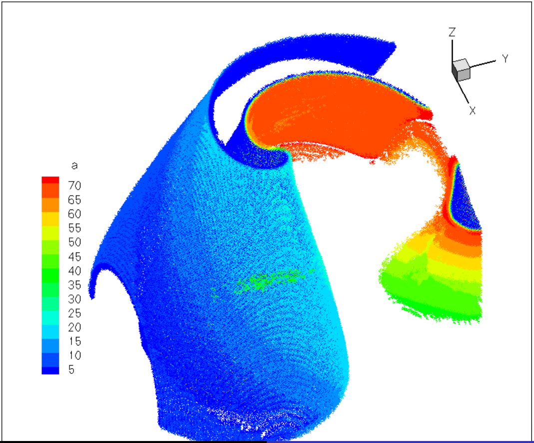
Adaptive Grid

Compression $\longrightarrow$ 12.97%



Adaptive Grid

Compression ⟶ 12.95%



Memory requirements

Radau5 $\longrightarrow$ $L_1 = 4 \times W_1 \times W_1 + 12 \times W_1 + 20$

Rock4 $\longrightarrow$ $L_2 = 8 \times W_2$

$$W = 3 \times 512 \times 512 \times 512 \approx 4.03 \times 10^8 \longrightarrow 24 \text{ Gb}$$

	W_1	W_2	$L = L_1 + L_2$	
Quasi-exact	W	0	6.5×10^{17}	$\longrightarrow$ 36 Eb
Splitting	3	W	3.2×10^9	$\longrightarrow$ 191 Gb
MR/Splitting 1.10^{-1}	3	$0.13W$	4.2×10^8	$\longrightarrow$ 25 Gb

Memory requirements

Radau5 $\longrightarrow$ $L_1 = 4 \times W_1 \times W_1 + 12 \times W_1 + 20$

Rock4 $\longrightarrow$ $L_2 = 8 \times W_2$

$W = 3 \times 512 \times 512 \times 512 \approx 4.03 \times 10^8 \longrightarrow$ 24 Gb

	W_1	W_2	$L = L_1 + L_2$	
Quasi-exact	W	0	6.5×10^{17}	$\longrightarrow$ 36 Eb
Splitting	3	W	3.2×10^9	$\longrightarrow$ 191 Gb
MR/Splitting 1.10^{-1}	3	$0.13W$	4.2×10^8	$\longrightarrow$ 25 Gb

Memory requirements

Radau5 $\longrightarrow$ $L_1 = 4 \times W_1 \times W_1 + 12 \times W_1 + 20$

Rock4 $\longrightarrow$ $L_2 = 8 \times W_2$

$W = 3 \times 512 \times 512 \times 512 \approx 4.03 \times 10^8 \longrightarrow$ 24 Gb

	W_1	W_2	$L = L_1 + L_2$	
Quasi-exact	W	0	6.5×10^{17}	$\longrightarrow$ 36 Eb
Splitting	3	W	3.2×10^9	$\longrightarrow$ 191 Gb
MR/Splitting 1.10^{-1}	3	$0.13W$	4.2×10^8	$\longrightarrow$ 25 Gb

AVC Model - Work in progress - 0-d model

System of 19 ODEs.

Unknowns :

- ions: $(Na^+,\ K^+,\ Ca^{2+},\ Cl^-,\ glu^-)$ inside the neurons, glial cells and extracell space,
- $f_n,\ f_a,$
- V_n et V_a, potentials inside the neurons and glial cells.

In 3D, system of reaction-diffusion

- diffusion of ions in the astrocytes and in the extracell space,
- no diffusion of ions in neurons,
- no diffusion for f_n, f_a, V_n et V_a.

System of reaction-diffusion:

$$\frac{du_i}{dt} - \varepsilon_i \, \Delta \, u_i = F_i(u_1, ..., u_{19}).$$

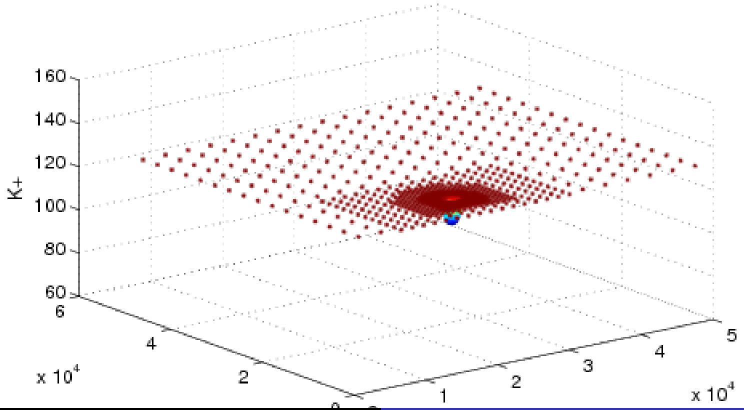
Homogeneous Neumann boundary conditions.

Configuration

- Time Domain : $T = [0, 3600]$
- Spatial Domain : $\Omega = [0, 50000] \times [0, 50000]$
- Integration Time Step : $\Delta t = 3600/360 = 10$
- Tolerances of Time Integrators: $tol = 1.10^{-5}$
- Finest Grid Level : $J = 8$
- Number of cells at Grid J: $2^{d \times J} = 65536 = 256 \times 256$

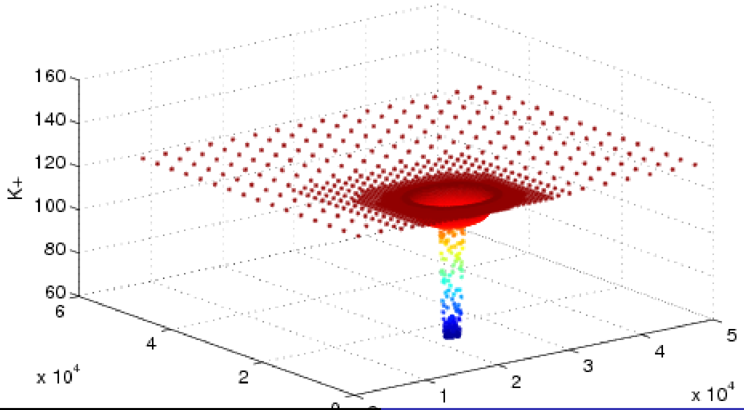
Adaptive Grid

Compression $\longrightarrow$ 1.25%



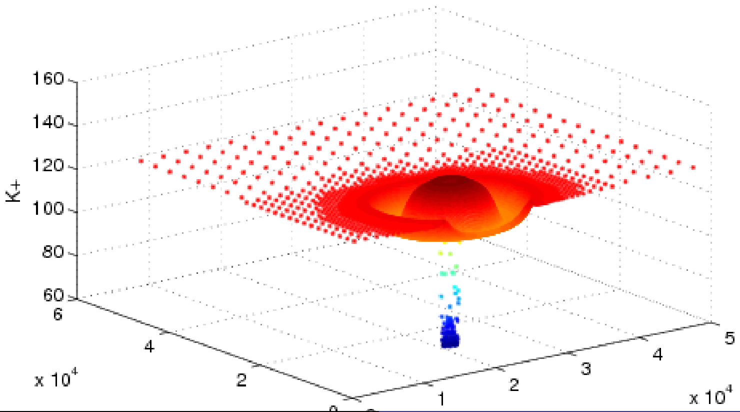
Adaptive Grid

Compression $\longrightarrow$ 7.54%



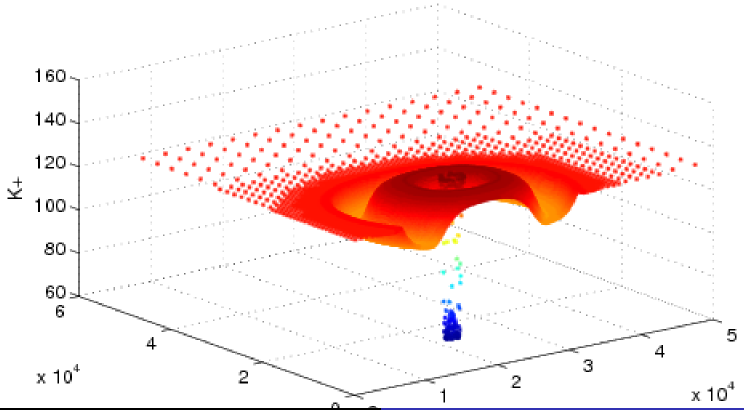
Adaptive Grid

Compression $\longrightarrow$ 16.59%



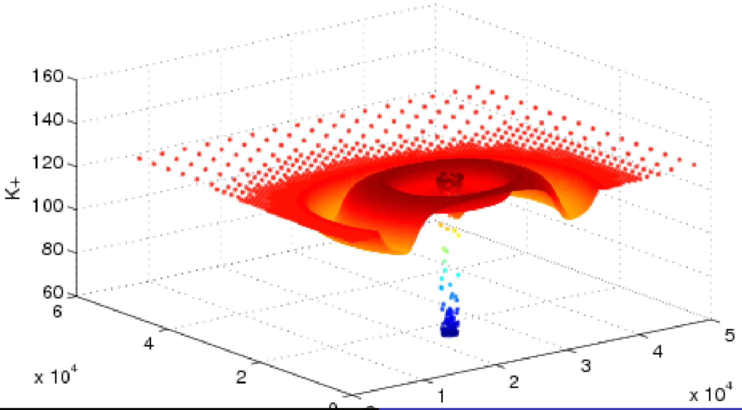
Adaptive Grid

Compression $\longrightarrow$ 28.52%



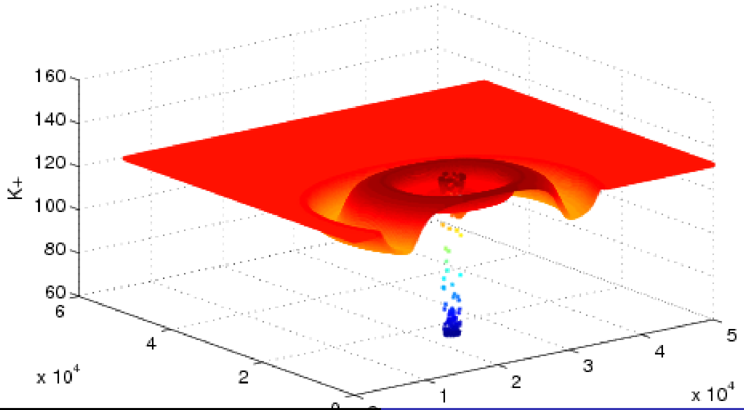
Adaptive Grid

Compression $\longrightarrow$ 35.45%



Fine Grid

Compression $\longrightarrow$ 100%



Conclusions and perspectives

- Convergence rate diminished due to Stiff phenomena
- Parallel speedup is possible, but the speedup is more modest than in space
- Reduction of the size of the system with adaptative muliresolution
- Work in progress
  - Detailed description of “waves”
  - Complex chemistry in a RD configuration
    $\longrightarrow$ detailed analysis of flame propagation
  - Multi-dimensional configurations (Brain in 3D)

Funds

- ANR CIS **PITAC**, 2006-2010. Coordinator Y. Maday
- ANR Blanche **SECHELLES**, 2009-2013. Coordinator S. Descombes
- **PEPS** from CNRS, 2007-2008. Coordinators F. Laurent and A. Bourdon
- **PEPS** from CNRS MIPAC, 2009-2010. Coordinator V. Louvet

References I

Hairer, E. and Wanner, G.
Solving ordinary differential equations II Stiff and differential-algebraic problems
Springer-Verlag, Berlin, 1991.

Abdulle, A.
Fourth order Chebyshev methods with recurrence relation
SIAM J. Sci. Comput., 23, pp. 2041-2054, 2002.

A. Cohen, S.M. Kaber, S. Müller and M. Postel
Fully Adaptive Multiresolution Finite Volume Schemes for Conservation Laws.
Mathematics of Computation, 72, pp. 183-225, 2001.

References II

S. Descombes, T. Dumont and M. Massot
Operator splitting for nonlinear reaction-diffusion systems with an entropic structure : singular perturbation, order reduction and application to spiral waves
Proceeding of the Workshop “Patterns and waves : theory and applications”, Saint-Petersbourg (2003)

S. Descombes and M. Massot
Operator splitting for nonlinear reaction-diffusion systems with an entropic structure : singular perturbation and order reduction
Numerische Mathematik (2004)

References III

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
International Journal of Computer Mathematics (2007)

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

References IV

T. Dumont, M. Duarte, S. Descombes, M.A. Dronne, M. Massot and V. Louvet
Optimal numerical strategy for human stroke simulation in complex 3D geometry of the brain.
Submitted to Bulletin of Mathematical Biology, available on HAL (2010)

M. Duarte, M. Massot, S. Descombes, C. Tenaud, T. Dumont, V. Louvet and F. Laurent
New resolution strategy for multi-scale reaction waves using time operator splitting, space adaptive multiresolution and dedicated high order implicit/explicit time integrators.

References V

Submitted to Journal of Computational Physics, available on HAL (2010)

Wavelet Representation

In the case where P_j^{j-1} is linear, i.e.,

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

using the wavelet terminology, we can write

$$u_\gamma := \langle u, \tilde{\varphi}_\gamma \rangle,$$

where the dual scaling function $\tilde{\varphi}_\gamma$ is simply

$$\tilde{\varphi}_\gamma := |\Omega_\gamma|^{-1} \chi_{\Omega_\gamma},$$

and

$$d_\mu := u_\mu - \hat{u}_\mu = \langle u, \tilde{\varphi}_\mu \rangle - \sum_\gamma c_{\mu,\gamma} \langle u, \tilde{\varphi}_\gamma \rangle = \langle u, \tilde{\psi}_\mu \rangle.$$

Wavelet Representation

The dual wavelet $\tilde{\psi}_\mu$ is given by

$$\tilde{\psi}_\mu := \tilde{\varphi}_\mu - \sum_\gamma c_{\mu,\gamma} \tilde{\varphi}_\gamma.$$

To describe in a simple way the multiresolution vector, we define $\nabla^J := \bigcup_{j=0}^J \nabla_j$ with $\nabla_0 := S_0$ and write

$$\boldsymbol{M_J = (d_\lambda)_{\lambda \in \nabla^J} = (\langle u, \tilde{\psi}_\lambda \rangle)_{\lambda \in \nabla^J}}$$

where we have set $d_\lambda = u_\lambda$ and $\tilde{\psi}_\lambda = \tilde{\varphi}_\lambda$ if $\lambda \in \nabla_0$.