High Order Local Implicit Time Schemes for Wave Equation

Marc Duruflé[1] Mamadou N'Diaye[1]

[1] Project Magique-3D
INRIA Bordeaux Sud-Ouest

July 5, 2018

Bibliography and motivation

Wave equation

$$\begin{cases} \rho\,\partial_t u - \operatorname{div}\vec{v} = 0, \quad \forall (x,t) \in \Omega \times \mathbb{R}^+ \\ \mu^{-1}\partial_t \vec{v} - \nabla u = 0, \quad \forall (x,t) \in \Omega \times \mathbb{R}^+ \\ +\text{ Dirichlet or Absorbing condition} \end{cases}$$

discretized with HDG formulation leads to discrete system

$$\frac{dy}{dt} = Ay(t) + F(t)$$

Explicit schemes too expensive because of restrictive CFL
⇒ Design efficient local implicit schemes for this ODE

Bibliography and motivation

PhD Thesis of Mamadou N'Diaye

- Optimized explicit schemes for HDG wave equation following the procedure proposed in *Optimal stability polynomials for numerical integration of initial value problems*, David. I. Ketcheson and Aron J. Ahmadia
- High-order implicit schemes compared in *High-order Padé and singly diagonally Runge-Kutta schemes for linear ODEs, application to wave propagation problems*, Hélène Barucq, Marc Duruflé and Mamadou N'Diaye
- Coupling of the two families of schemes following the procedure proposed in *Runge-Kutta-based explicit local time-stepping methods for wave propagation*, M. Mehlin, T. Mitkova and M. Grote

Explicit schemes

One-step schemes written in the form

$$y_{n+1} = R(\Delta t\, A) y_n + \widetilde{\phi}_n$$

R : a polynomial approximation of exponential
$\widetilde{\phi}_n$ term due to the source F :

$$\phi_n = \sum_{r=1}^{m} A^{r-1} \Delta t^r \sum_{i=0}^{n_w - 1} \omega_i^r \, F\left(t_n + \Delta t\, c_i\right)$$

c_i are interpolation points, and ω_i^r weights

Explicit schemes

One-step schemes written in the form

$$y_{n+1} = R(\Delta t\, A) y_n + \widetilde{\phi}_n$$

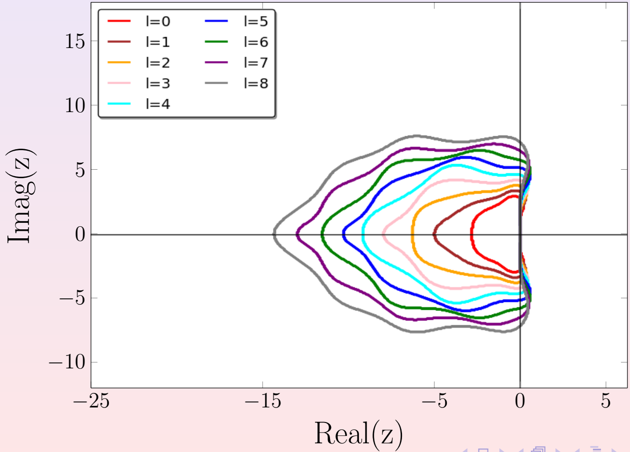
$$R(\Delta t A) = \sum_{j=0}^{m} \alpha_j (\Delta t A)^j$$

$\alpha_j = \frac{1}{j!}$, for $j \leq r$ to ensure a scheme of order r
Others free coefficients α_j are tuned to optimize CFL (with Ketcheson's algorithm)

Explicit schemes

One-step schemes written in the form

$$y_{n+1} = R(\Delta t\, A) y_n + \widetilde{\phi}_n$$



Implicit schemes

One-step schemes written in the form

$$y_{n+1} = R(\Delta t A) y_n + \widetilde{\phi}_n$$

R : a rational approximation of exponential
$\widetilde{\phi}_n$ term due to the source F :

$$\phi_n = \sum_{r=1}^{m} A^{r-1} \Delta t^r \sum_{i=0}^{n_w - 1} \omega_i^r \, F\,(t_n + \Delta t\, c_i)$$

c_i are interpolation points, and ω_i^r weights

Implicit schemes

One-step schemes written in the form

$$y_{n+1} = R(\Delta t A) y_n + \widetilde{\phi}_n$$

- Padé schemes : R chosen as Padé approximant of exponential
- Linear SDIRK schemes : R chosen as $\dfrac{P(z)}{(1-\gamma z)^m}$ providing the smallest error with the constraint of A-stability.

Locally implicit algorithm

Based on paper of Grote and coworkers

$$y = (I - P)y + Py, \quad (P: \text{ projection on fine region })$$

$$y(t_n + \xi\Delta t) = y(t_n) + \underbrace{\int_{t_n}^{t_n+\xi\Delta t} A(I-P)y(t)dt + \overbrace{\int_{t_n}^{t_n+\xi\Delta t} (I-P)F(t)dt}^{\text{source term}}}_{\text{Coarse part}}$$

$$+ \underbrace{\int_{t_n}^{t_n+\xi\Delta t} APy(t)dt + \overbrace{\int_{t_n}^{t_n+\xi\Delta t} PF(t)dt}^{\text{source term}}}_{\text{Fine part}}$$

Locally implicit algorithm

To coincide with explicit time schemes, we obtain

$$y(t_n + \xi \Delta t) \approx y_n + A(I - P) \sum_{j=0}^{m} \alpha_{j+1} (\xi \Delta t)^{j+1} \widetilde{w}_j$$
$$+ (I - P) \left(\hat{Q}(t_n + \xi \Delta t) - \hat{Q}(t_n) \right)$$
$$+ \int_{t_n}^{t_n + \xi \Delta t} APy(t) + PF(t) dt$$

where $\widetilde{w}_j$ is the discrete approximation of $y^{(j)}(t_n)$ by differentiating $j - 1$ times $y' = Ay + F$, and Q the polynomial approximation of F, $\hat{Q}$ its antiderivative

Locally implicit algorithm

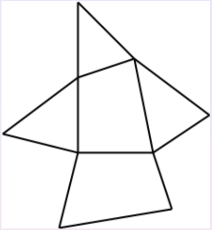
By introducing $\tau = \xi \Delta t$, and differentiating with respect to τ, we get the fine ODE :

$$\frac{d\widetilde{y}(\tau)}{d\tau} = \overbrace{A(I-P)\sum_{j=0}^{m}(j+1)\alpha_{j+1}\,\tau^j\,\widetilde{w}_j + (I-P)Q(t_n+\tau) + PF(t_n+\tau)}^{\text{updated source term}} + AP\widetilde{y}(\tau)$$

Fine ODE is solved implicitly with Padé schemes or Linear SDIRK schemes. It involves only close degrees of freedom (non-null rows of AP).

How to split the mesh

Time step Δt_i computed with adjacent elements of K_i



- We find λ_{max} such that $|R(\lambda \Delta t_{\text{nominal}})|$ is maximal
- Δt_i found by bisection such that $|R(\lambda \Delta t_i)| = 1$
- If $\Delta t_i \leq \Delta t_{ref} \Rightarrow$, element $K_i \in \Omega^{fine}$

Practical algorithm

Algorithm used to compute $\zeta_j = A(I - P)\alpha_{j+1}\widetilde{w}_j$, F_j

$$D_{i,\ell} = \frac{\widetilde{\varphi}_i^{(\ell)}(0)}{(\Delta t)^\ell}$$

for $i = 1 \ldots s$ **do**

  compute $F_i = F(t_n + c_i \Delta t)$

end for

$w = y_n$

for $j = 0 \ldots m$ **do**

  compute $z = A\,(I - P)\,w$ and $z_p = A\,P\,w$

  $\zeta_j = \alpha_{j+1} z$

  compute $Q^{(j)} = \sum_{i=1}^{s} D_{i,j} F_i$

  $w = z + z_p + Q^{(j)}$

end for

Practical algorithm

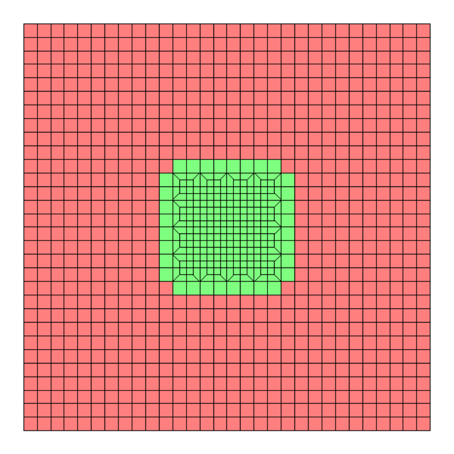
Computation of y_{n+1}:

- Compute vectors F_i and ζ_j with previous algorithm
- Task 1 : compute y_{n+1} for far degrees of freedom with explicit scheme
- Task 2 : compute y_{n+1} for close degrees of freedom by solving the fine ODE with an implicit scheme

Task 1 and 2 can be conducted independently (in parallel)

Time convergence

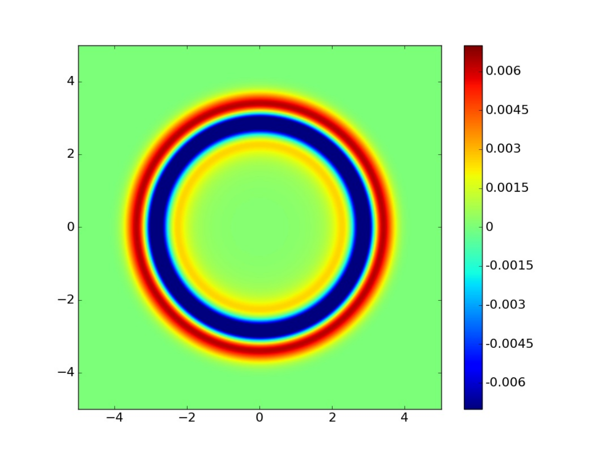
Fixed space discretization with $\mathbb{Q}_{12}$ and given mesh:



Green zone : fine region, Red zone : coarse region

Time convergence

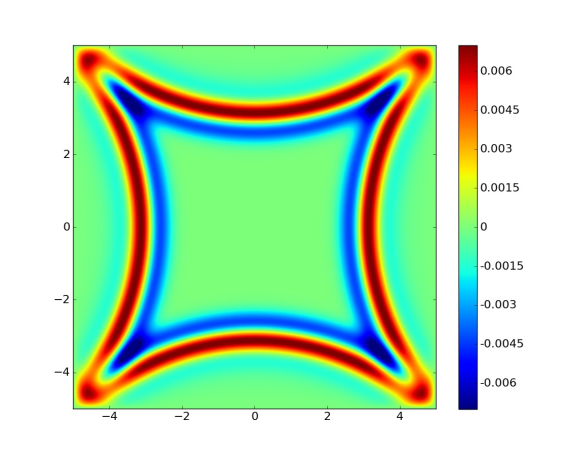
$$\Omega = [-5, 5]^2, \quad c = 1, \text{ Solution at } t = 4$$



Gaussian source in space and Ricker in time ($f_0 = 1 Hz$)

Time convergence

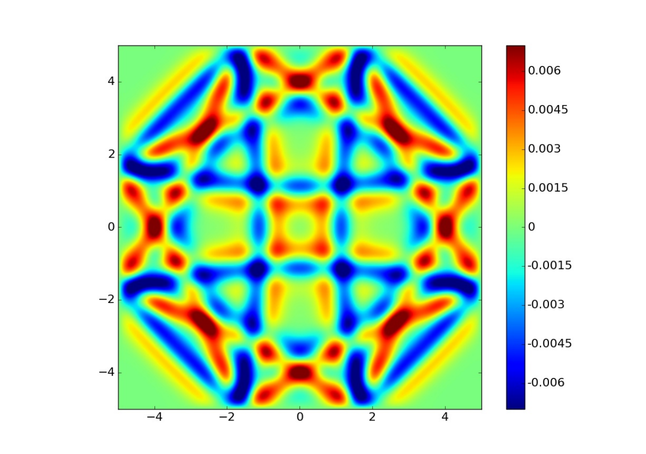
Solution at $t = 8$



Gaussian source in space and Ricker in time ($f_0 = 1$)

Time convergence

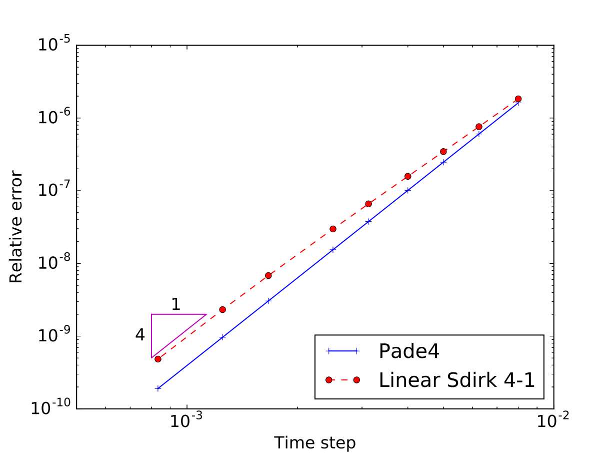
Solution at $t = 20$



Gaussian source in space and Ricker in time ($f_0 = 1$)

Time convergence

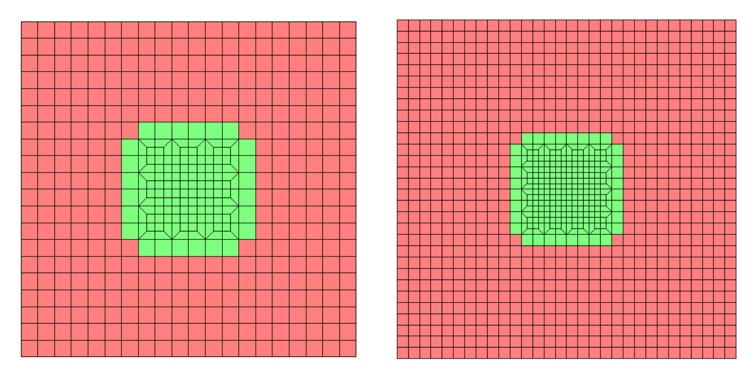
Convergence in time ($\Delta t \to 0$)



for ERK4-2 and Pade4 (or Linear SDIRK 4-1)

Space-time convergence

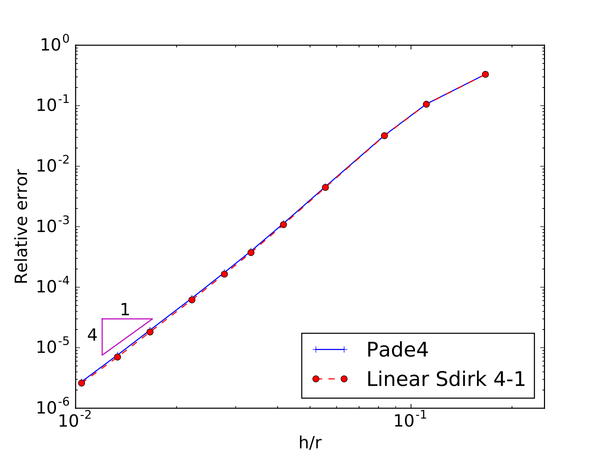
Space-time convergence $\Delta t = \alpha \Delta x$, $\Delta x \to 0$



with fixed coefficient α (close to the CFL of the coarse region)

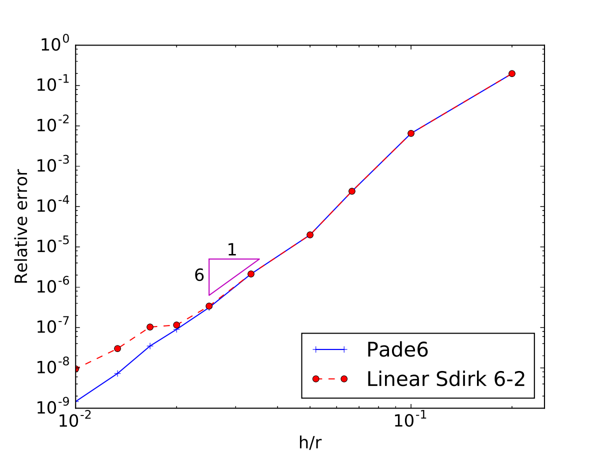
Space-time convergence

Convergence with $\mathbb{Q}_3$, ERK4-2 and Pade 4 (or Linear SDIRK 4-1)



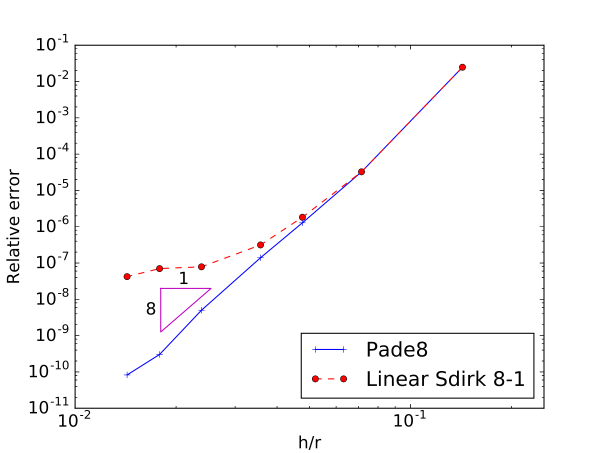
Space-time convergence

Convergence with $\mathbb{Q}_5$, ERK6-2 and Pade 6 (or Linear SDIRK 6-2)



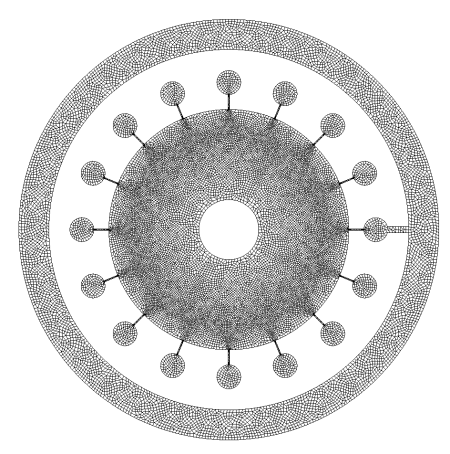
Space-time convergence

Convergence with $\mathbb{Q}_7$, ERK8-2 and Pade 8 (or Linear SDIRK 8-1)



2-D numerical results

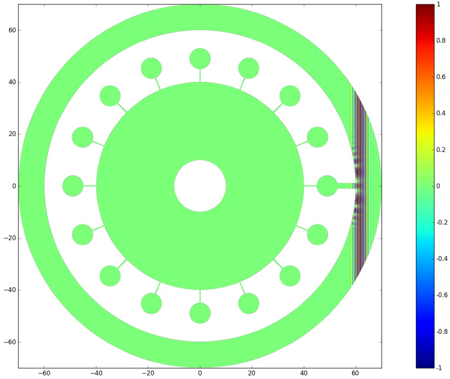
Scattering of a magnetron (diameter=140λ)



16 small circular cavities, Space discretization : $\mathbb{Q}_8$

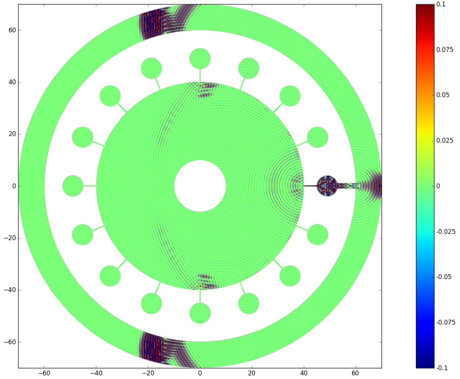
2-D numerical results

Solution for $t = 20$



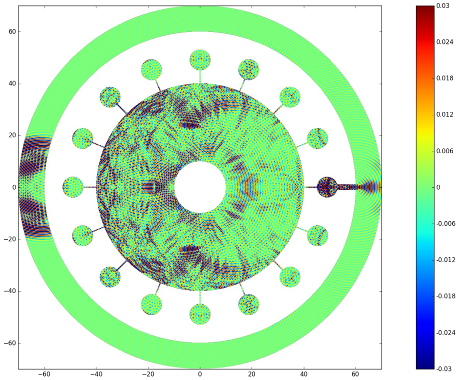
2-D numerical results

Solution for $t = 100$



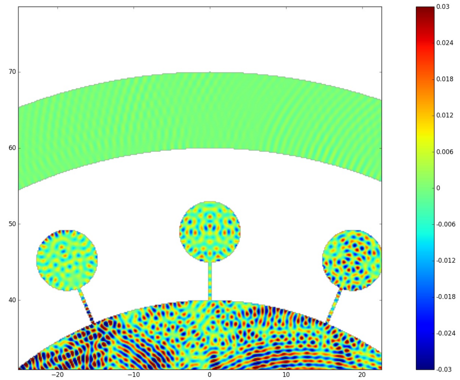
2-D numerical results

Solution for $t = 200$



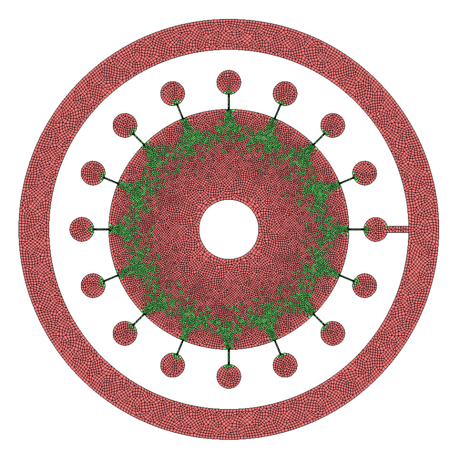
2-D numerical results

Solution for $t = 200$ (zoom on two cavities)



2-D numerical results

Splitting into coarse (red) and fine region (green)



2-D numerical results

Efficiency with ERK 4-2 and LinearSdirk 4-1 (or Padé 4) on 16 cores (error 0.3%)

Method	Time step	Computational Time	Memory
Purely Explicit	$9.09 \cdot 10^{-4}$	8h52min	720 Mo
Local LSDIRK	0.025	54min15s	1.8 Go
LSDIRK implicit	0.04	1h12s	3.2 Go
Local Padé	0.025	39min11s	2.3 Go
Padé implicit	0.033	38min27s	4.8 Go

⇒ Locally implicit scheme is a compromise between computational time and memory usage

2-D numerical results

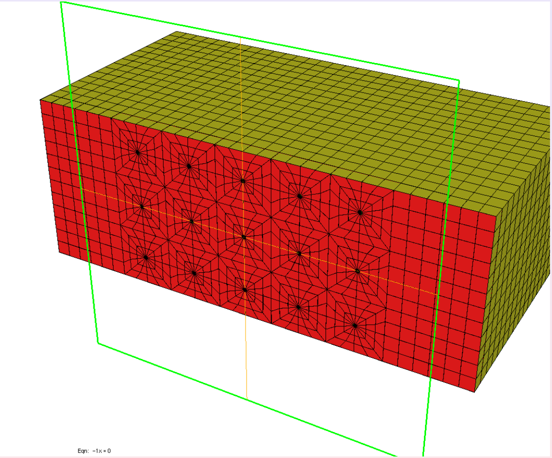
Efficiency with ERK 8-2 and LinearSdirk 8-1 (or Padé 8) on 16 cores

Method	Time step	CPU Time	Memory	Error
Local LSDIRK	0.033	1h23min	2.1Go	$1.9 \cdot 10^{-6}$
LSDIRK implicit	0.167	34min39s	3.2 Go	0.002
Local Padé	0.033	57min21s	3.1 Go	$2.32 \cdot 10^{-9}$
Padé implicit	0.25	11min7s	7.9 Go	0.00202

$\Rightarrow$ Better accuracy with eighth-order schemes

3-D numerical results

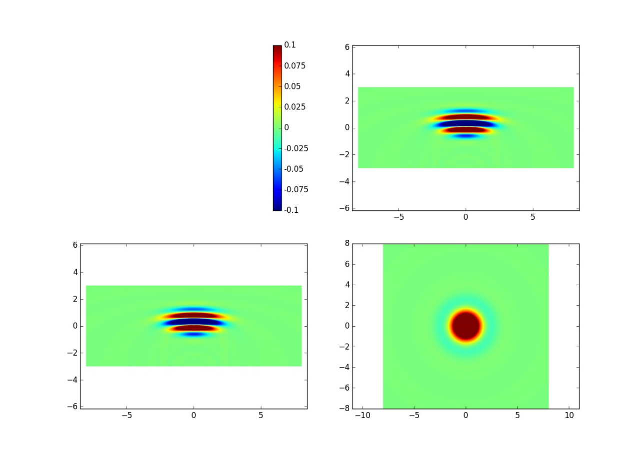
Scattering of a network of small spheres



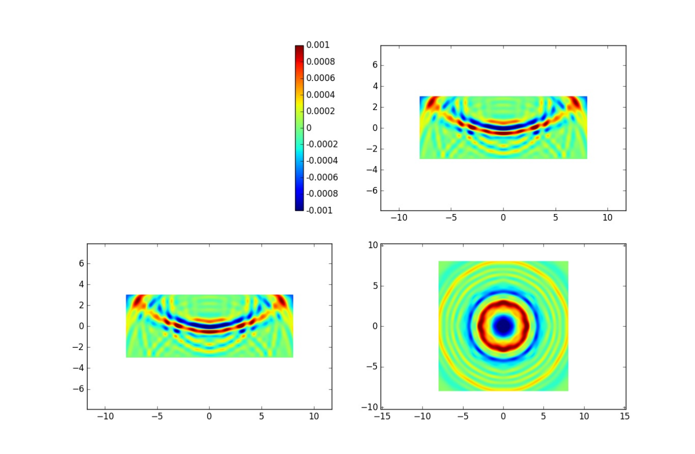
75 small spheres with $\rho = 0.1, \ \mu = 0.8$, Space discretization : $\mathbb{Q}_4$

3-D numerical results

Solution for $t = 6$

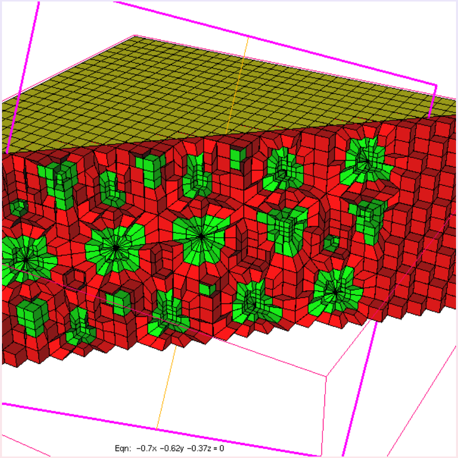


Solution for $t = 12$



3-D numerical results

Splitting into coarse (red) and fine region (green)



3-D numerical results

Efficiency with ERK 4-0 and LinearSdirk 4-1 on 16 cores

Method	Time step	Computational Time	Memory
Local LSDIRK	0.01	2h23	62.8 Go
Purely Explicit	$2.22 \cdot 10^{-4}$	13h40	3.3 Go
LSDIRK implicit	0.05	57min	108 Go

$\Rightarrow$ Locally implicit scheme is a compromise between computational time and memory usage

Prospects

- Improvement of parallelization
- Mix between local time-stepping and locally implicit

Prospects

- Improvement of parallelization
- Mix between local time-stepping and locally implicit

Thanks for your attention