Parareal methods and averaged models for solving stiff differential equations – Applications

Sever Hirstoaga

ALPINES team, Inria Paris & LJLL, Sorbonne Université, CNRS

Strasbourg, August 2024

Joint work with Laura Grigori, Julien Salomon, Pierre-Henri Tournier

Summer school "New trends in computing", Strasbourg

Outline

Outline

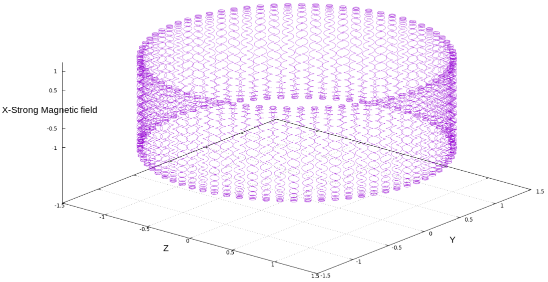
Motivation

Complex dynamics of charged particles (ions and free electrons) in electro-magnetic fields.

$\hookrightarrow$ Plasma confinement under **large** magnetic and/or electric field.

Multiscale dynamics in time.

Example:
fast Larmor gyration $\ll$ parallel motion $\ll$ drift across field lines.



Reduced model's goal:

~~fast Larmor gyration~~

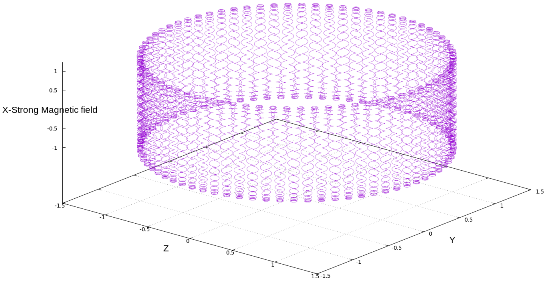
Motivation

Complex dynamics of charged particles (ions and free electrons) in electro-magnetic fields.

$\hookrightarrow$ Plasma confinement under **large** magnetic and/or electric field.

Multiscale dynamics in time.

Example:
fast Larmor gyration $\ll$ parallel motion $\ll$ drift across field lines.



Reduced model's goal:

~~fast Larmor gyration~~

Motivation

Aim : propose a time stepping for solving accurately and rapidly in times $T_{\text{end}} \sim 1/\varepsilon$ stiff equations.

Ingredients:

- Parareal algorithm: A time-stepping scheme for parallel in time computations.
- Reduced models: Zero-order approximations of the multiscale equations.

The parareal strategy

Question: what choice for the coarse solver $\mathcal{G}$?

Standard choices:

- G = approximation scheme of F solver but with a larger time step
- G = different approximation scheme than F's, with lower accuracy

$\hookrightarrow$ Use **reduced (averaged) models** to define the coarse solver.

Reason: Reduced models are not stiff ODEs $\rightsquigarrow$ low computational cost.

Some similar approaches

Maday 2007, Haut, Wingate, ... 2014 – 2022, Ariel, Kim, Tsai 2016

The parareal strategy

Question: what choice for the coarse solver $\mathcal{G}$?

Standard choices:

- G = approximation scheme of F solver but with a larger time step
- G = different approximation scheme than F's, with lower accuracy

$\hookrightarrow$ Use **reduced (averaged) models** to define the coarse solver.

Reason: Reduced models are not stiff ODEs $\rightsquigarrow$ low computational cost.

Some similar approaches

Maday 2007, Haut, Wingate, ... 2014 – 2022, Ariel, Kim, Tsai 2016

The parareal strategy

Question: what choice for the coarse solver $\mathcal{G}$?

Standard choices:

- G = approximation scheme of F solver but with a larger time step
- G = different approximation scheme than F's, with lower accuracy

$\hookrightarrow$ Use **reduced (averaged) models** to define the coarse solver.

Reason: Reduced models are not stiff ODEs $\rightsquigarrow$ low computational cost.

Some similar approaches

Maday 2007, Haut, Wingate, ... 2014 – 2022, Ariel, Kim, Tsai 2016

Outline

Realistic Penning trap: magnetic bottle

Knapp, Kendl, Koskela, Ostermann, 2015. Solve for $0 < \varepsilon \ll 1$

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{x}_\varepsilon}{\mathrm{d}t} = \mathbf{v}_\varepsilon, & \mathbf{x}_\varepsilon(s) = \mathbf{x}, \\ \dfrac{\mathrm{d}\mathbf{v}_\varepsilon}{\mathrm{d}t} = \dfrac{1}{\varepsilon}(\mathbf{v}_\varepsilon)^\perp + \mathbf{v}_\varepsilon \times \mathbf{B}(\mathbf{x}_\varepsilon) + \mathbf{E}(\mathbf{x}_\varepsilon), & \mathbf{v}_\varepsilon(s) = \mathbf{v}, \end{cases}$$

where, for $c > 0$, $k > 0$

$$\mathbf{E}(\mathbf{x}) = c \begin{pmatrix} -x \\ y/2 \\ z/2 \end{pmatrix} \quad \text{and} \quad \mathbf{B}(\mathbf{x}) = k \begin{pmatrix} x^2 - (y^2 + z^2)/2 \\ -xy \\ -xz \end{pmatrix}$$

Device for storing charged particles.

- if $1/\varepsilon > \sqrt{2c}$ then stable periodic trajectory.
- otherwise the particle escapes from the trap.

No analytic solution; oscillations at three time scales: $2\pi\varepsilon$, 1 and $2\pi/\varepsilon$.

Realistic Penning trap: magnetic bottle

Knapp, Kendl, Koskela, Ostermann, 2015. Solve for $0 < \varepsilon \ll 1$

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{x}_\varepsilon}{\mathrm{d}t} = \mathbf{v}_\varepsilon, & \mathbf{x}_\varepsilon(s) = \mathbf{x}, \\ \dfrac{\mathrm{d}\mathbf{v}_\varepsilon}{\mathrm{d}t} = \dfrac{1}{\varepsilon}(\mathbf{v}_\varepsilon)^\perp + \mathbf{v}_\varepsilon \times \mathbf{B}(\mathbf{x}_\varepsilon) + \mathbf{E}(\mathbf{x}_\varepsilon), & \mathbf{v}_\varepsilon(s) = \mathbf{v}, \end{cases}$$

where, for $c > 0$, $k > 0$

$$\mathbf{E}(\mathbf{x}) = c \begin{pmatrix} -x \\ y/2 \\ z/2 \end{pmatrix} \quad \text{and} \quad \mathbf{B}(\mathbf{x}) = k \begin{pmatrix} x^2 - (y^2 + z^2)/2 \\ -xy \\ -xz \end{pmatrix}$$

Device for storing charged particles.

- if $1/\varepsilon > \sqrt{2c}$ then stable periodic trajectory.
- otherwise the particle escapes from the trap.

No analytic solution; oscillations at three time scales: $2\pi\varepsilon$, 1 and $2\pi/\varepsilon$.

Averaged models

from two-scale asymptotic expansion theory.
Sanders, Verhulst, 1985, Frénod, 2006, N'Guetseng 1989, Allaire 1990
We develop the solution

$$\mathcal{X}_\varepsilon(t) = \mathcal{X}^0\Big(t, \frac{t-s}{\varepsilon}\Big) + \varepsilon\, \mathcal{X}^1\Big(t, \frac{t-s}{\varepsilon}\Big) + \varepsilon^2\, \mathcal{X}^2\Big(t, \frac{t-s}{\varepsilon}\Big) + \dots$$

when $\varepsilon \to 0$ and where the functions $\mathcal{X}^i(t, \theta)$ are periodic in θ, $\forall\, i \in \mathbb{N}$.

The limit $\mathcal{Y}^0 = (\mathbf{y}^0, \mathbf{u}^0)$ is solution to the i.v.p.

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{y}^0}{\mathrm{d}t} = \begin{pmatrix} \mathbf{u}^0_x \\ 0 \\ 0 \end{pmatrix}, \quad \dfrac{\mathrm{d}\mathbf{u}^0}{\mathrm{d}t} = \begin{pmatrix} \mathbf{E}_x(\mathbf{y}^0) \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{B}_x(\mathbf{y}^0)\,\mathbf{u}^0_z \\ -\mathbf{B}_x(\mathbf{y}^0)\,\mathbf{u}^0_y \end{pmatrix} \\ \mathbf{y}^0(s) = \mathbf{x},\ \mathbf{u}^0(s) = \mathbf{v}, \end{cases}$$

More complex equations for $\mathcal{Y}^1 = (\mathbf{y}^1, \mathbf{u}^1)$ coupled with $(\mathbf{y}^0, \mathbf{u}^0)$!

Properties of the reduced models

- both reduced models average the fastest rotation motion.
- the first-order model is more accurate than the zero-order one in the approximation of the **bounce motion**.
- the **electric drift** $\mathbf{E} \times \mathbf{e}_1$ is missed by the zero-order model, unlike the first-order one.

The system for $Y = (\mathbf{y}^0, \mathbf{u}^0, \mathbf{y}^1, \mathbf{u}^1)$ is **source-free**

$$\frac{\mathrm{d}Y}{\mathrm{d}t} = F(Y) \quad \text{where } F : \mathbb{R}^{12} \to \mathbb{R}^{12} \text{ satisfies } \nabla \cdot F = 0.$$

$\hookrightarrow$ conserves volumes in the enlarged phase space.

Volume-preserving scheme : splitting method (which is 4th order, time-symmetric).

Feng, Shang, 1995, Hairer, Lubich, Wanner, 2006

Properties of the reduced models

- both reduced models average the fastest rotation motion.
- the first-order model is more accurate than the zero-order one in the approximation of the **bounce motion**.
- the **electric drift** $\mathbf{E} \times \mathbf{e}_1$ is missed by the zero-order model, unlike the first-order one.

The system for $Y = (\mathbf{y}^0, \mathbf{u}^0, \mathbf{y}^1, \mathbf{u}^1)$ is **source-free**

$$\frac{\mathrm{d}Y}{\mathrm{d}t} = F(Y) \quad \text{where } F : \mathbb{R}^{12} \to \mathbb{R}^{12} \text{ satisfies } \nabla \cdot F = 0.$$

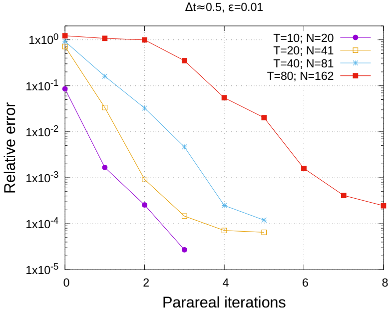
$\hookrightarrow$ conserves volumes in the enlarged phase space.

Volume-preserving scheme : splitting method (which is 4th order,
time-symmetric).

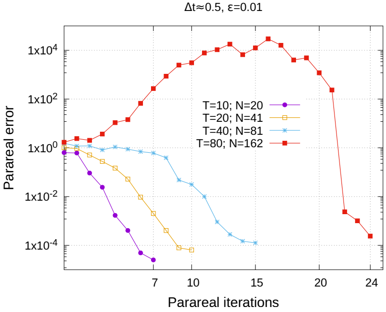
Feng, Shang, 1995, Hairer, Lubich, Wanner, 2006

Parareal numerical results

the **strong** magnetic field is $1/\varepsilon = 100$ and the reduced model timestep is $\Delta t = 0.5 \approx 8$ gyroperiods.



1st order model



limit model

Speedup of Parareal

Computing $(F(T_{n+1}, T_n, U_n^k))_{n=0,\dots,N-1}$ in parallel over N processors.
$N = 162$. $\varepsilon \in \{0.01; 0.001\}$.

Nb. of points in $P = 2\pi\varepsilon$	10	20	40
Error(fine solver) at $T = 80$	$4.2543 \cdot 10^{-3}$	$2.816 \cdot 10^{-4}$	$1.77 \cdot 10^{-5}$
Nb. of Parareal iterations	7	8	8
Speedup	4.8	6.7	10.0

Nb. of points in $P = 2\pi\varepsilon$	10	20	40
Error(fine solver) at $T = 80$	$4.5956 \cdot 10^{-4}$	$2.989 \cdot 10^{-5}$	$1.89 \cdot 10^{-6}$
Nb. of Parareal iterations	2	3	4
Speedup	47.7	43.4	36.0

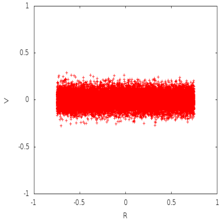
Outline

Vlasov-Poisson equation – beam in a focusing channel

For $\varepsilon \to 0$ solve numerically

$$\begin{cases} \partial_t f^\varepsilon + \frac{v}{\varepsilon}\partial_r f^\varepsilon + \left(E^\varepsilon - \frac{r}{\varepsilon} + rH\left(\frac{t}{\varepsilon}\right)\right)\partial_v f^\varepsilon = 0, \\ \frac{1}{r}\partial_r (r\,E^\varepsilon) = \int f^\varepsilon(t,r,v)\,dv. \\ f^\varepsilon(t=0,r,v) = f_0(r,v). \end{cases}$$

- $f^\varepsilon = f^\varepsilon(t,r,v)$ particles distrib. function
- Time $t \in [0,T]$, Position $r > 0$, Velocity $v \in \mathbb{R}$



- $r \mapsto r/\varepsilon$ **strong external electric field**
- $E^\varepsilon(t,r)$ self-consistent electric field
- H is a periodic external function.

Paraxial approximation: Filbet-Sonnendrücker (2006), Frénod-Salvarani-Sonnendrücker (2009), Mouton (2009), Crouseilles-Lemou-Méhats-Zhao (2013, 2017)

Examples

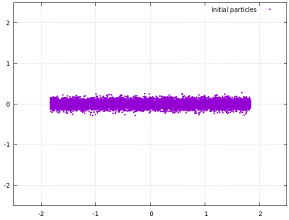
Let the initial distribution

$$f_0(r,v) = \frac{1}{\sqrt{2\pi}\, v_{\text{th}}} \exp\Big(-\frac{v^2}{2v_{\text{th}}^2}\Big)\chi_{[r_{\min},r_{\max}]}(r),$$

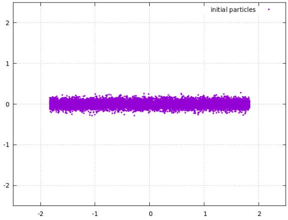
where $v_{\text{th}} = 0.072$, $r_{\max} = 1.83$ and $r_{\min} = -r_{\max}$ and $\chi_{[r_{\min},r_{\max}]}(r) = 1$ if $r \in [r_{\min}, r_{\max}]$ and $\chi_{[r_{\min},r_{\max}]}(r) = 0$ otherwise.

$H(\tau) = \cos^2(\tau)$ $\rightsquigarrow$ focusing effect; $H(\tau) = \cos(2\tau)$ $\rightsquigarrow$ defocusing effect.

$H \equiv 0$



$H = \cos^2(\cdot)$



Two-scale limit model

Frénod, Salvarani, Sonnendrücker (M3AS, 2009).

When $\varepsilon \to 0$, $(f_\varepsilon, E_\varepsilon)$ two-scale converges to $(F, \mathcal{E})$ over $[0, T]$.

$$F(t,\tau,r,v) = G\big(t,\, \cos(\tau)r - \sin(\tau)v,\, \sin(\tau)r + \cos(\tau)v\big),$$

and $(G, \mathcal{E})$ is the solution of the following model

$$\begin{cases} \dfrac{\partial G}{\partial t} + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} -\sin(\tau)\Big[\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big) + H(\tau)\big(\cos(\tau)q+\sin(\tau)u\big)\Big]d\tau\,\dfrac{\partial G}{\partial q} \\ \qquad + \dfrac{1}{2\pi}\displaystyle\int_0^{2\pi} \cos(\tau)\Big[\mathcal{E}\big(t,\tau,\cos(\tau)q+\sin(\tau)u\big) + H(\tau)\big(\cos(\tau)q+\sin(\tau)u\big)\Big]d\tau\,\dfrac{\partial G}{\partial u} = 0, \\ G(0,q,u) = f_0(q,u), \\ \dfrac{1}{r}\dfrac{\partial (r\mathcal{E})}{\partial r} = \Upsilon, \quad \Upsilon(t,\tau,r) = \displaystyle\int_{\mathbb{R}} G\big(t,\, \cos(\tau)r - \sin(\tau)v,\, \sin(\tau)r + \cos(\tau)v\big)\,dv. \end{cases}$$

- When $\varepsilon \to 0$, f_ε is approximated by $f_\varepsilon(t,r,v) \approx F\Big(t, \dfrac{t}{\varepsilon}, r, v\Big)$.
- The transport equation of G is free of high oscillations.

Numerical approximation

• particle in cell algorithm for both models (ε-dependent and the limit).
Raviart (1985), Birdsall-Langdon (1985), Hockney-Eastwood (1988), ...

Dirac sum approximation for f_0

$$f_0^{N_p}(r,v) = \sum_{k=1}^{N_p} \omega_k \, \delta(r - r_0) \, \delta(v - v_0)$$

implies a Dirac sum for the solution f^ε:

$$f_\varepsilon^{N_p}(t,r,v) = \sum_{k=1}^{N_p} \omega_k \, \delta(r - R_k(t)) \, \delta(v - V_k(t))$$

where N_p is the number of **macroparticles** and $\big(R_k(t), V_k(t)\big)$ is the macroparticle k moving along a characteristic curve of Vlasov eq.

$$R'(t) = \frac{1}{\varepsilon} V(t), \qquad R(0) = r_0$$

$$V'(t) = -\frac{1}{\varepsilon} R(t) + E(t, R(t)) + R(t) H\Big(\frac{t}{\varepsilon}\Big), \qquad V(0) = v_0$$

Numerical approximation

• particle in cell algorithm for both models (ε-dependent and the limit).
Raviart (1985), Birdsall-Langdon (1985), Hockney-Eastwood (1988), ...

Dirac sum approximation for f_0

$$f_0^{N_p}(r,v) = \sum_{k=1}^{N_p} \omega_k\, \delta(r - r_0)\, \delta(v - v_0)$$

implies a Dirac sum for the solution f^ε:

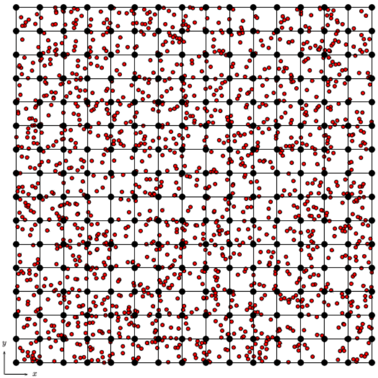
$$f_\varepsilon^{N_p}(t,r,v) = \sum_{k=1}^{N_p} \omega_k\, \delta(r - R_k(t))\, \delta(v - V_k(t))$$

where N_p is the number of **macroparticles** and $\big(R_k(t), V_k(t)\big)$ is the macroparticle k moving along a characteristic curve of Vlasov eq.

$$R'(t) = \frac{1}{\varepsilon} V(t), \qquad R(0) = r_0$$

$$V'(t) = -\frac{1}{\varepsilon} R(t) + E(t, R(t)) + R(t) H\Big(\frac{t}{\varepsilon}\Big), \qquad V(0) = v_0$$

Particle in Cell method



The main time loop :

- deposit particles on the grid $\Rightarrow$ the grid density ρ (RHS of Poisson eq.)
- solve Poisson equation on the grid $\Rightarrow$ the grid electric field E
- interpolate E in each particle
- push particles with this field

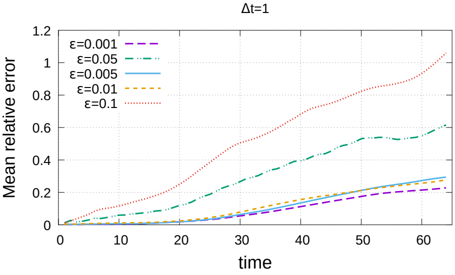
  $\nwarrow$

  ODEs to solve

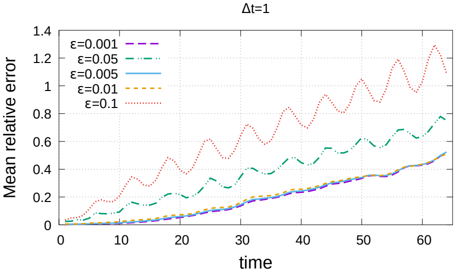
Validity of the reduced model

Runge-Kutta 4 scheme for original and reduced models. Fine $\delta t = 2\pi\varepsilon/100$. 10000 particles and 128 cells.

$$\mathrm{Error}(t_n) = \frac{1}{N_p}\sum_{j=1}^{N_p} \frac{\|(R_j^n, V_j^n) - (\widetilde{R}_j^n, \widetilde{V}_j^n)\|_2}{\|(\widetilde{R}_j^n, \widetilde{V}_j^n)\|_2}.$$



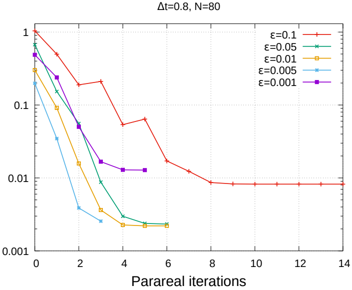
$H \equiv 0$



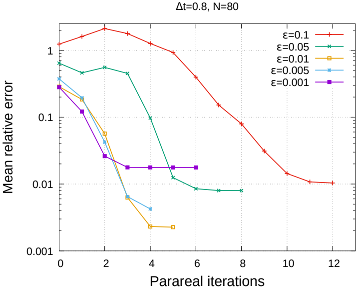
$H = \cos^2(\cdot)$

Convergence of Parareal

Use the two-scale limit model to define $\mathcal{G}$.
K given by the error of the fine solver w.r.t. the very fine solution.



$H \equiv 0$



$H = \cos^2(\cdot)$

Outline

Theoretic Speedup

Computing $(F(T_{n+1}, T_n, U_n^k))_{n=0,\ldots,N-1}$ in parallel over N processors.

The total time of the parareal run is

$$T_{\text{par}} = T_{\text{init}} + K\Big(\frac{T_{\text{fine}}}{N} + T_{\text{coarse}}\Big),$$

where K is the parareal iterations number.

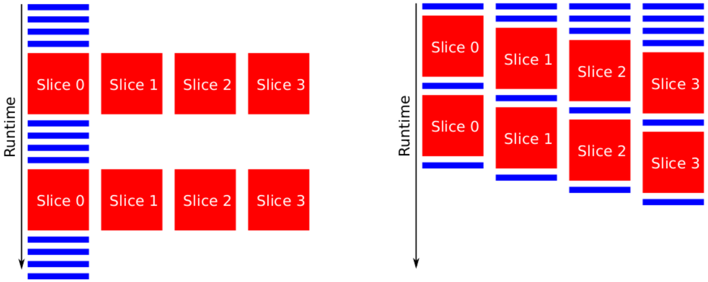
$$\text{Thus} \quad \mathbf{S}(N) = \frac{1}{\Big(1 + K\Big)\dfrac{T_c}{T_f} + \dfrac{K}{N}}$$

where $T_{\text{fine}} = NT_f$, $T_{\text{coarse}} = NT_c$.

Pipelined Parareal

- allows to reduce the time of coarse calculations from NT_c to T_c.

Minion (2010), Aubanel (2011), Ruprecht (2017)

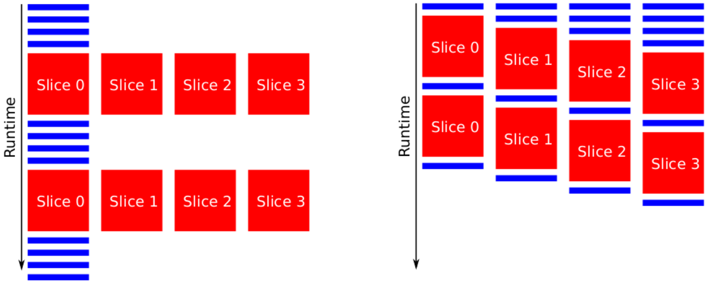


from D. Ruprecht's paper "Shared Memory Pipelined Parareal", Euro-Par 2017.

Thus $\mathbf{S_p}(N) = \dfrac{1}{\left(1 + \frac{K}{N}\right)\frac{T_c}{T_f} + \frac{K}{N}}$

Pipelined Parareal

- allows to reduce the time of coarse calculations from NT_c to T_c.
Minion (2010), Aubanel (2011), Ruprecht (2017)

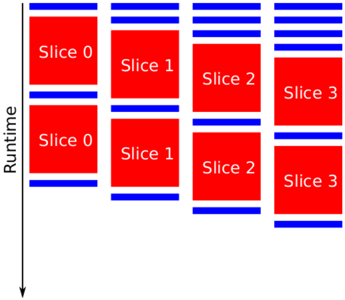


from D. Ruprecht's paper "Shared Memory Pipelined Parareal", Euro-Par 2017.

Thus $\mathbf{S_p}(N) = \dfrac{1}{\left(1+\dfrac{K}{N}\right)\dfrac{T_c}{T_f}+\dfrac{K}{N}} \quad > \quad \mathbf{S}(N)$, since $\frac{K}{N} \ll K$.

Pipelined Parareal

-standard implementation using MPI.

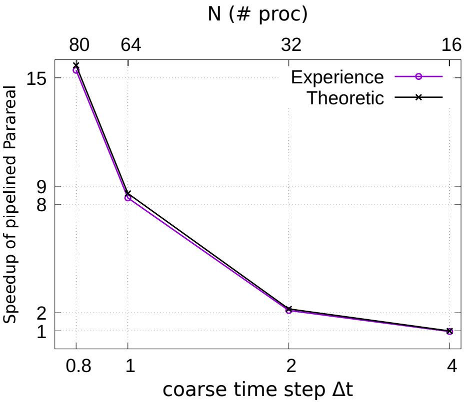


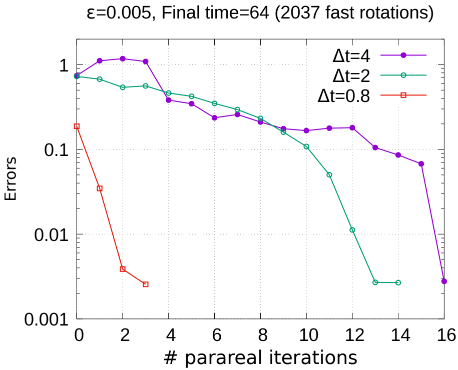
Algorithm 1: Parareal using MPI

```
input: Initial value q_0; number of iterations K
q ← q_0
p = MPI_COMM_RANK()
q ← G_Δt(q, t_p, 0)
q_c ← G_Δt(q, t_{p+1}, t_p)
for k = 1, K do
    q ← F_δt(q, t_{p+1}, t_p)
    δq ← q − q_c
    if Process not first then
        MPI_RECV(q, source = p − 1)
    end
    else
        q ← q_0
    end
    q_c ← G_Δt(q, t_{p+1}, t_p)
    q ← q_c + δq
    if Process not last then
        MPI_SEND(q, target = p + 1)
    end
end
```

Speedup of the pipelined implementation

simulations on Leto.
case $H \equiv 0$ and $\varepsilon = 0.005$ (*i.e.* an accurate coarse solver)

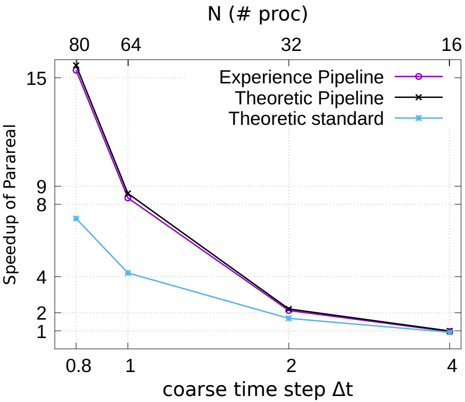


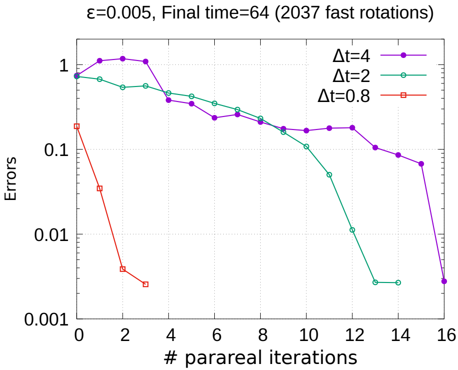


Speedup of the pipelined implementation

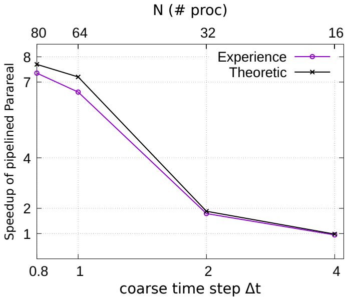
simulations on Leto.

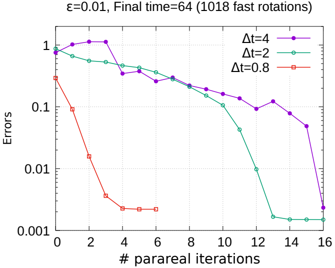
case $H \equiv 0$ and $\varepsilon = 0.005$ (*i.e.* an accurate coarse solver)

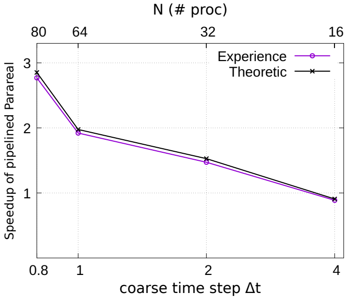


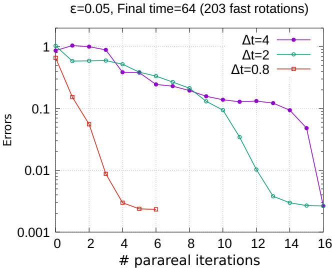


Speedup for $\varepsilon = 0.01$ and $\varepsilon = 0.05$

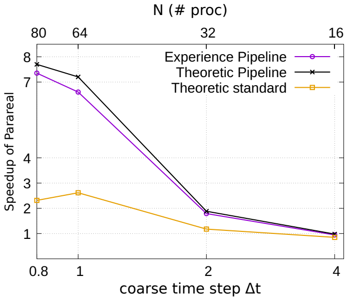


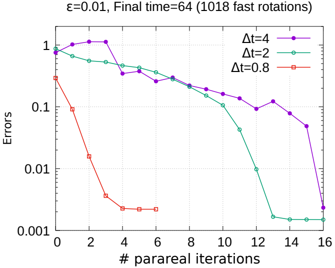


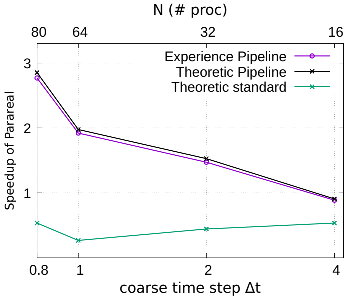


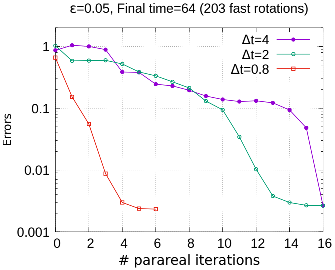


Speedup for $\varepsilon = 0.01$ and $\varepsilon = 0.05$









Summary and Outlook

Conclusion:

Parareal algorithm provides accurate results at any T_{end} and for any ε, with a low computational cost.
The pipelined version speeds up the simulations.

Shared memory parallelism to be added.

`Cons`:

- need for finding the reduced model (not always an easy task).
- derive estimate for the error of the reduced model.
- parareal does not allow to speed up for $\varepsilon \sim 0.1$.

Thank you!

Summary and Outlook

Conclusion:
Parareal algorithm provides accurate results at any T_{end} and for any ε, with a low computational cost.
The pipelined version speeds up the simulations.

Shared memory parallelism to be added.

Cons:

- need for finding the reduced model (not always an easy task).
- derive estimate for the error of the reduced model.
- parareal does not allow to speed up for $\varepsilon \sim 0.1$.

Thank you!