# Time parallelisation for optimal control and data assimilation

Julien Salomon

*Joint work with F. Kwok, S. Reyes-Riffo, L. Perrin*

JLL & INRIA, ANGE project-team (Inria, Paris-Sorbonne Université, CNRS)

PinT22, 14.7.22

**Problem: assimilation on an unbounded interval** $[t_0, +\infty)$

Given a (linear) dynamic

$$\dot{x}(t) = Ax(t) + Bu(t)$$

whose initial condition **is NOT known**, and an output

$$y(t) = Cx(t),$$

which **is known**.

| | |
|---|---|
| **Objective:** | Combine observer approaches with a time-parallelization. |

Given a (linear) dynamic

$$\dot{x}(t) = Ax(t) + Bu(t)$$

whose initial condition **is NOT known**, and an output

$$y(t) = Cx(t),$$

which **is known**: data to be assimilated.
→ **Solver:** Luenberger observer

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L\left(C\hat{x}(t) - y(t)\right).$$

In general:

$$\hat{x}(t_0) \neq x(t_0).$$

**Theoretical result:** Assume the observability condition

$$rank \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix} = n,$$

then **there exists** $L$ such that

$$\rho\,(\exp(A - LC)) \leq 1 \;\Rightarrow \|x(t) - \hat{x}(t)\| \leq \kappa\, e^{-\lambda t} \|x(0) - \hat{x}(0)\|,$$

with $\lambda = \min_{\alpha \in \text{spec}(A-LC)} |\alpha|$, $\kappa = \text{Cond}(A - LC)$.

→ *Standard algorithms to design L: Routh's or Hurwitz criterion, Ackermann's formula, LQ theory... Pole assignment procedures.*

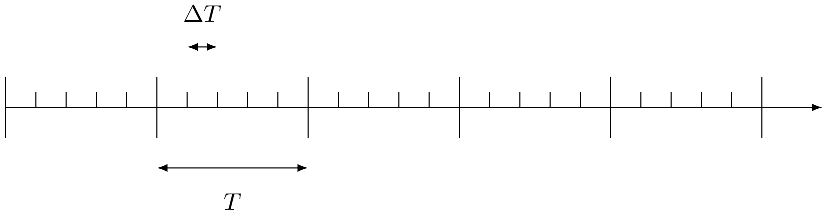
**Important remark:** in

$$\|x(t) - \hat{x}(t)\| \leq \kappa\, e^{-\lambda t} \|x(0) - \hat{x}(0)\|,$$

the factor $\|x(0) - \hat{x}(0)\|$ is unknown...Only the rate of convergence is known.

# A time parallelization strategy

**Idea:** In order to accelerate the assimilation, simulate the observer using time-parallelization on **Windows**.




Consider a time-domain decomposition associated with a time-parallelization procedure and introduce

- **Windows:** interval of length $T$ on which a time-parallelization algorithm is applied.
- **Subintervals:** set of $N$ intervals of length $\Delta T$ that make up the decomposition on which the algorithm are based.

We proceed as follows: Suppose we are on the window $\ell$

$$W_\ell := [t_\ell, t_{\ell+1} = t_\ell + T],$$

1. Consider an approximation $\hat{x}_{\|}(t_\ell)$ of $\hat{x}(t_\ell)$.
2. Compute a time-parallelized approximation $\hat{x}_{\|}$ of $\hat{x}$ on $W_\ell$, with a tolerance $Tol_\ell$ such that **the rate of convergence is preserved**.
3. Let the final state $\hat{x}_{\|}(t_{\ell+1})$ be the initial state for the next window.

# Outline

Suppose we are on the window $\ell$

$$W_\ell := [t_\ell, t_{\ell+1} = t_\ell + T],$$

1. Consider an approximation $\hat{x}_{\|}(t_\ell)$ of $\hat{x}(t_\ell)$.
2. Apply $k_\ell$ **iterations of the Parareal algorithm** to get an approximation of $\hat{x}$ on $W_\ell$.
3. Let the final state $\hat{x}_{\|}(t_{\ell+1})$ be the initial state for the next window.

**Mandatory:**

$$k_\ell << N.$$

**Lemma:** We have

$$\left\|\varepsilon_{\|}(t_n^{\ell-})\right\| \leq \gamma \left( \|x(0) - \hat{x}(0)\| + \sum_{j=1}^{\ell} \mathrm{e}^{\mu j T} \|\mathcal{J}_j^h\| \right) \mathrm{e}^{-\mu \ell T}$$

where $\varepsilon_{\|}(t) = x(t) - \hat{x}_{\|}(t)$ is the error of approximation and $\mathcal{J}_j^h = -\sum_{n=1}^{N-1} \mathrm{e}^{(A-LC)(N-n)\Delta T} J_{j,n}^h$.

**Proof:** Let $\ell \geq 1$. We have

$$\begin{aligned}
\varepsilon_{\|}(t_n^{\ell-}) = x(t_n^{\ell}) - \hat{x}_{\|}(t_n^{\ell-}) &= \mathrm{e}^{(A-LC)\Delta T}\mathrm{e}^{-(A-LC)\Delta T}\left(x(t_n^{\ell}) - \hat{x}_{\|}(t_n^{\ell-})\right) \\
&= \mathrm{e}^{(A-LC)\Delta T}(x(t_{n-1}^{\ell}) - \hat{x}_{\|}(t_{n-1}^{\ell+})) \\
&= \mathrm{e}^{(A-LC)\Delta T}(x(t_{n-1}^{\ell}) - \hat{X}_{\ell,n-1}^{h})) \\
&= \mathrm{e}^{(A-LC)\Delta T}(x(t_{n-1}^{\ell}) - \hat{x}_{\|}(t_{n-1}^{\ell-}) - J_{\ell,n-1}^{h}) \\
&= \mathrm{e}^{(A-LC)\Delta T}(\varepsilon_{\|}(t_{n-1}^{\ell-}) - J_{\ell,n-1}^{h})
\end{aligned}$$

so that

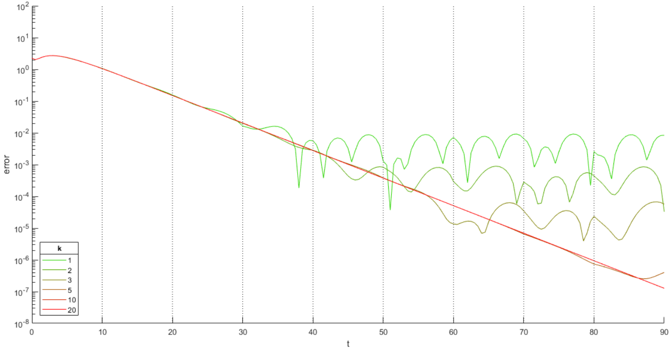
$$\varepsilon_{\|}(t_N^{\ell}) = \varepsilon_{\|}(t_0^{\ell+1}) = \mathrm{e}^{(A-LC)T}\varepsilon_{\|}(t_0^{\ell}) + \mathcal{J}_{\ell}^{h}, \tag{1}$$

where we have used the continuity of $\hat{x}_{\|}$ in $t_N^{\ell}$. In the same way, we obtain

$$\varepsilon_{\|}(T_{\ell}) = \mathrm{e}^{(A-LC)\ell T}\varepsilon_{\|}(0) + \sum_{j=1}^{\ell}\mathrm{e}^{(A-LC)(\ell-j)T}\mathcal{J}_j^{h}.$$

The result follows from the contraction property of $\hat{x}$.

What happens when $k_\ell = k_{\max}$ is fixed for all windows ?




**This is not surprising:** $k_{max}$ parareal iterations introduce a constant error.

**Definition of $k_\ell$:**
Let us assume $h$ is obtained from the stopping criterion in $W_\ell$

$$\gamma \sum_{n=1}^{N-1} \mathrm{e}^{-\mu(N-n)\Delta T} \|J_{\ell,n}^h\| \leq \widetilde{\gamma} \frac{\mathrm{e}^{-\mu\ell T}}{2^\ell}$$

where $\widetilde{\gamma}$ is an arbitrary parameter. Then, the rate of convergence of $\hat{x}_{\|}(t)$ to $x(t)$ is bounded by $\mu$, i.e.

$$\left\|\varepsilon_{\|}(T_\ell)\right\| \leq \gamma \left(\|x(0) - \hat{x}(0)\| + \widetilde{\gamma}\right) \mathrm{e}^{-\mu\ell T}.$$

**Proof:** By definition of $\mathcal{J}_\ell^h$, we have

$$\|\mathcal{J}_\ell^h\| \leq \sum_{n=1}^{N-1} \gamma \mathrm{e}^{-\mu(N-n)\Delta T} \|J_{\ell,n}^h\| \leq \widetilde{\gamma} \frac{\mathrm{e}^{-\mu \ell T}}{2^\ell}.$$

From (1), we deduce that

$$\begin{aligned} \|\varepsilon_\parallel(T_\ell)\| &\leq \gamma \mathrm{e}^{-\mu\ell T} \left( \|x(0) - \hat{x}(0)\| + \sum_{j=1}^{\ell} \mathrm{e}^{\mu j T} \cdot \widetilde{\gamma} \frac{\mathrm{e}^{-\mu j T}}{2^j} \right) \\ &= \gamma \left( \|\varepsilon_\parallel(0)\| + \widetilde{\gamma} \sum_{j=1}^{\ell} \frac{1}{2^j} \right) \mathrm{e}^{-\mu\ell T} \\ &\leq \gamma \left( \|\varepsilon_\parallel(0)\| + \widetilde{\gamma} \right) \mathrm{e}^{-\mu\ell T}. \end{aligned}$$

The result then follows from $\varepsilon_\parallel(0) = x(0) - \hat{x}_\parallel(0) = x(0) - \hat{x}(0)$.

*F. Kwok, S. Reyes-Riffo, J. Salomon, to be submitted ...*

Example: $N = 20$.

$$A = \begin{pmatrix} 0 & 1 \\ -1 & -2 \end{pmatrix}$$

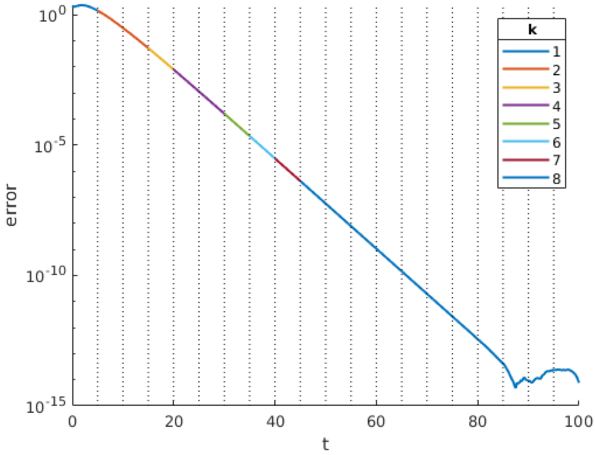
$$B = \begin{pmatrix} 0 \\ 1 \end{pmatrix},\ C = \begin{pmatrix} 0 & 1 \end{pmatrix},\ L = \begin{pmatrix} 0.8 \\ -1.1 \end{pmatrix}$$

$$u(t) = 3 + 0.5 \sin(0.75t)$$

$$T = 5,\ \Delta T = \frac{T}{N} = 0.25.$$

$$\Delta t = \Delta T,\ \delta t = \frac{\Delta t}{25}.$$

Example: $N = 20$.



Efficiency: CPU time to reach $\|\varepsilon_{\|}\| = \|x(t) - \hat{x}_{\|}(t)\| \leq 10^{-12}$.

- $CPU_{\|}$:0.0144 s
- $CPU_{sequential}$: 0.0579 s
- ratio = 0.579
- efficiency = 47%

# Outline

**Setting :**

- Parallel integration of the following linear initial problem :

$$\begin{cases} \dot{\mathbf{x}}(t) = M\mathbf{x}(t) + \mathbf{g}(t), \quad t \in [0,T] \\ \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{2}$$

$M \in \mathcal{M}_{m\times m}(\mathbb{R})$, $\mathbf{x}(t), \mathbf{g}(t) \in \mathbb{C}^m$.

**Setting :**

- Parallel integration of the following linear initial problem :

$$\begin{cases} \dot{\mathbf{x}}(t) = M\mathbf{x}(t) + \mathbf{g}(t), \quad t \in [0, T] \\ \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{2}$$

  $M \in \mathcal{M}_{m\times m}(\mathbb{R})$, $\mathbf{x}(t), \mathbf{g}(t) \in \mathbb{C}^m$.

- We separate the homogeneous IVP and the inhomogenuos with null initial condition:

$$\begin{cases} \dot{\mathbf{w}}(t) = M\mathbf{w}(t), \\ \mathbf{w}(0) = \mathbf{x}_0, \end{cases} \quad \begin{cases} \dot{\mathbf{v}}(t) = M\mathbf{v}(t) + \mathbf{g}(t), \\ \mathbf{v}(0) = \mathbf{0}, \end{cases} , \quad t \in [0, T]$$

**Setting :**

- Parallel integration of the following linear initial problem :

$$\begin{cases} \dot{\mathbf{x}}(t) = M\mathbf{x}(t) + \mathbf{g}(t), & t \in [0,T] \\ \mathbf{x}(0) = \mathbf{x}_0, \end{cases} \tag{2}$$

  $M \in \mathcal{M}_{m\times m}(\mathbb{R})$, $\mathbf{x}(t), \mathbf{g}(t) \in \mathbb{C}^m$.

- We separate the homogeneous IVP and the inhomogenuos with null initial condition:

$$\begin{cases} \dot{\mathbf{w}}(t) = M\mathbf{w}(t), \\ \mathbf{w}(0) = \mathbf{x}_0, \end{cases} \quad \begin{cases} \dot{\mathbf{v}}(t) = M\mathbf{v}(t) + \mathbf{g}(t), \\ \mathbf{v}(0) = \mathbf{0}, \end{cases} , \quad t \in [0,T]$$

- The solution of (2) is given by :

$$\mathbf{x}(t) = \mathbf{v}(t) + \mathbf{w}(t).$$

$\rightarrow$ divide our interval $[0, T]$ in $N$ subintervals ($N$ #processors), solve Type 1 problems in parallel, then Type 2 problems in parallel (matrix exponential).

*PARAEXP: A parallel integrator for linear initial-value problems*, M.J. Gander and S. Güttel, SIAM J. Sci. Comp., Vol. 35 (2), 2013.

Considering the Luenberger observer, we can write it :

$$\dot{\hat{\mathbf{x}}}(t) = M\hat{\mathbf{x}}(t) + \mathbf{g}(t), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0, \quad t \geq 0,$$

with :

$$\begin{cases} M = (A - LC), \\ \mathbf{g}(t) = B\mathbf{u}(t) + L\mathbf{y}(t). \end{cases}$$

Considering the Luenberger observer, we can write it :

$$\dot{\hat{\mathbf{x}}}(t) = M\hat{\mathbf{x}}(t) + \mathbf{g}(t), \quad \hat{\mathbf{x}}(0) = \hat{\mathbf{x}}_0, \quad t \geq 0,$$

with :

$$\begin{cases} M = (A - LC), \\ \mathbf{g}(t) = B\mathbf{u}(t) + L\mathbf{y}(t). \end{cases}$$

- Compute $\hat{\mathbf{x}}$ on a window $W_\ell$ falls under the setting of the ParaExp scheme on each window $W_\ell = (T_{\ell-1}, T_\ell)$.
- $\hat{\mathbf{x}}_{||}(t)$ : approximation of $\hat{\mathbf{x}}(t)$ by the parallel observer

The error only comes from the source term and we use RK4 :

$$\Rightarrow h_{\ell+1} \leq \left((h_\ell)^4 \mathrm{e}^{-\mu T}\right)^{1/4}, \quad \forall \ell \in \mathbb{N}^*.$$

Submarine toy model (*Grimble, van der Molen, Liceaga-Castro, Submarine depth and pitch control, 1993*).

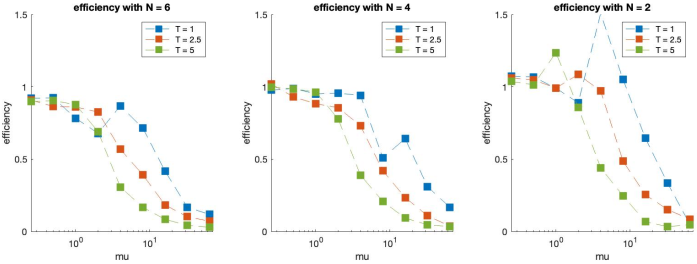



Figure: Results, reminder : $N$ is the number of processors, $T$ the size of the windows, $\mu$ the smallest magnitude of the eigenvalues.

Heat equation on $\Omega = [0,1]^2$ :

$$\left\{ \begin{array}{rcll} \partial_t x(s,t) & = & \Delta x(s,t) + g(s,t), & s \in \Omega, \\ x(s,t) & = & 0, & s \in \partial\Omega, \\ x(s,0) & = & u_0, & s \in \Omega. \end{array} \right.$$

and the observer, $\omega$ being the circle of center $(1/2, 1/2)$ with radius $r$ :

$$\left\{ \begin{array}{rcll} \partial_t \hat{x}(s,t) & = & \Delta \hat{x}(s,t) + g(s,t) + \mathcal{L}(\hat{x}_{|\omega}(s,t) - x_{|\omega}(s,t)), & s \in \Omega, \\ \hat{x}(s,t) & = & 0, & s \in \partial\Omega, \\ \hat{x}(s,0) & = & \hat{x}_0, & s \in \Omega. \end{array} \right.$$

Pole assignement does not work !

$$\mathcal{L} = -\gamma 1_{|\omega}^{\top}.$$

2D Heat equation on $[0,1]^2$, #cpu $= 8$.

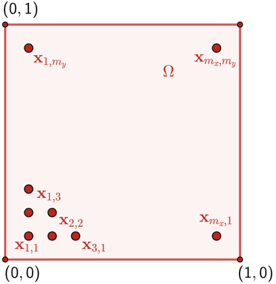
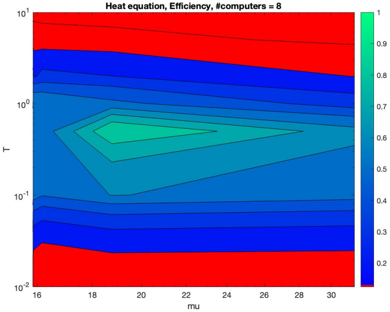



Figure: Setting Heat 2D, A size $(91 \times 91)$.

2D Heat equation on $[0,1]^2$, $r = 0.15$, #cpu $= 8$.



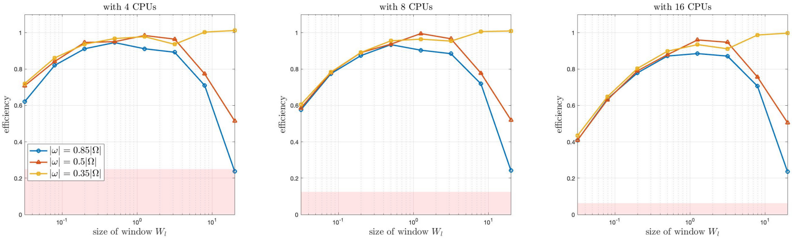
2D Wave equation on $\Omega = [0,1]^2$:

$$\begin{cases} \partial_{tt} x(s,t) &= \Delta x(s,t) + g(s,t), & s \in \Omega, \\ x(s,t) &= 0, & s \in \partial\Omega, \\ x(s,0) &= u_0, & s \in \Omega. \end{cases}$$

and the observer, $\omega$ being the circle of center $(1/2, 1/2)$ with radius $r$ :

$$\begin{cases} \partial_{tt} \hat{x}(s,t) &= \Delta \hat{x}(s,t) + g(s,t) + \mathcal{L}(\hat{x}_{|\omega}(s,t) - x_{|\omega}(s,t)), & s \in \Omega, \\ \hat{x}(s,t) &= 0, & s \in \partial\Omega, \\ \hat{x}(s,0) &= \hat{x}_0, & s \in \Omega. \end{cases}$$

2D Wave equation on $\Omega = [0, 1]^2$.



# Coupling time parallelization with data assimilation

- The **Luenberger observer** is for **unbounded time intervals**,
- **Time parallelization** scheme work on **bounded intervals**.

# Coupling time parallelization with data assimilation

- The **Luenberger observer** is for **unbounded time intervals**,
- **Time parallelization** scheme work on **bounded intervals**.

Strategy to apply time parallelization on unbounded tom intervals :

1. Divide our unbounded time interval $[0, \infty)$ into windows of length $T > 0$ : $W_\ell = (T_{\ell-1}, T_\ell)$, $\ell \geq 1$.

# Coupling time parallelization with data assimilation

- The **Luenberger observer** is for **unbounded time intervals**,
- **Time parallelization** scheme work on **bounded intervals**.

Strategy to apply time parallelization on unbounded tom intervals :

1. Divide our unbounded time interval $[0, \infty)$ into windows of length $T > 0$ : $W_\ell = (T_{\ell-1}, T_\ell)$, $\ell \geq 1$.
2. Apply a time parallelization procedure ($Parareal$,$ParaExp$,etc) on each windows succesively.

# Coupling time parallelization with data assimilation

- The **Luenberger observer** is for **unbounded time intervals**,
- **Time parallelization** scheme work on **bounded intervals**.

Strategy to apply time parallelization on unbounded tom intervals :

1. Divide our unbounded time interval $[0, \infty)$ into windows of length $T > 0$ : $W_\ell = (T_{\ell-1}, T_\ell)$, $\ell \geq 1$.
2. Apply a time parallelization procedure (*Parareal*,*ParaExp*,etc) on each windows succesively.
3. Estimate the error (at the end of the window on $T_\ell$) gives a criterion to switch to the next window.

# Coupling time parallelization with data assimilation

- The **Luenberger observer** is for **unbounded time intervals**,
- **Time parallelization** scheme work on **bounded intervals**.

Strategy to apply time parallelization on unbounded tom intervals :

1. Divide our unbounded time interval $[0, \infty)$ into windows of length $T > 0$ : $W_\ell = (T_{\ell-1}, T_\ell)$, $\ell \geq 1$.
2. Apply a time parallelization procedure (*Parareal*,*ParaExp*,etc) on each windows succesively.
3. Estimate the error (at the end of the window on $T_\ell$) gives a criterion to switch to the next window.
4. Strategy : Time parallelization error decays with the same rate as the observer error.

Merci !