EMR SIM 22

Time parallelisation and data assimilation

Paraexp and Luenberger observer

Lucas Perrin

Julien Salomon

Inria Paris, ANGE Team

30 Mai 2022

Time parallelisation and data assimilation

→ Present a sequential data assimilation : the Luenberger observer

→ Explain the parallel in time scheme used : Paraexp

→ Expose our PinT method for sequential data assimilation

Data assimilation : Luenberger observer & dynamical systems

state dynamical system :

$$\begin{cases} \dot{x} = Ax(t) + Bu(t) \\ y(t) = Cx(t) \\ x(0) = x_0, \quad t \geq 0 \end{cases}$$

- $\rightarrow$ $x(t)$: *state* vector
- $\rightarrow$ $y(t)$: *output* vector
- $\rightarrow$ $A \in \mathcal{M}_{m\times m}(\mathbb{R})$,
  $B \in \mathcal{M}_{m\times p}(\mathbb{R})$,
  $C \in \mathcal{M}_{q\times m}(\mathbb{R})$
- $\rightarrow$ x_0 is unknown

observer system :

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L[y(t) - \hat{y}(t)] \\ \hat{y}(t) = C\hat{x}(t) \\ \hat{x}(0) = \hat{x}_0, \quad t \geq 0 \end{cases}$$

- $\rightarrow$ $\hat{x}(t)$: *observer* vector
- $\rightarrow$ $L \in \mathcal{M}_{m\times q}(\mathbb{R})$
- $\rightarrow$ $\hat{x}_0$ chosen as we want

$$\dot{x}(t) - \dot{\hat{x}}(t) = (A - LC)(x(t) - \hat{x}(t))$$

Data assimilation : Luenberger observer & dynamical systems

state dynamical system :

$$\begin{cases} \dot{x} = Ax(t) + Bu(t) \\ y(t) = Cx(t) \\ x(0) = x_0, \quad t \geq 0 \end{cases}$$

- → $x(t)$: *state* vector
- → $y(t)$: *output* vector
- → $A \in \mathcal{M}_{m\times m}(\mathbb{R})$,
  $B \in \mathcal{M}_{m\times p}(\mathbb{R})$,
  $C \in \mathcal{M}_{q\times m}(\mathbb{R})$
- → x_0 is unknown

observer system :

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L[y(t) - \hat{y}(t)] \\ \hat{y}(t) = C\hat{x}(t) \\ \hat{x}(0) = \hat{x}_0, \quad t \geq 0 \end{cases}$$

- → $\hat{x}(t)$: *observer* vector
- → $L \in \mathcal{M}_{m\times q}(\mathbb{R})$
- → $\hat{x}_0$ chosen as we want

$$\epsilon(t) = \mathrm{e}^{(A-LC)t}(x(0) - \hat{x}(0))$$

Data assimilation : Luenberger observer & dynamical systems

state dynamical system :

$$\begin{cases} \dot{x} = Ax(t) + Bu(t) \\ y(t) = Cx(t) \\ x(0) = x_0, \quad t \geq 0 \end{cases}$$

observer system :

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + L[y(t) - \hat{y}(t)] \\ \hat{y}(t) = C\hat{x}(t) \\ \hat{x}(0) = \hat{x}_0, \quad t \geq 0 \end{cases}$$

How do we choose L ?

so that : $\hat{x}(t) \longrightarrow x(t)$

$\|\epsilon(t)\| \longrightarrow 0$

$\mathfrak{Re}(\Lambda = \sigma(A - LC)) < 0$

→ ODE : $L = \texttt{place}(\Lambda, A, C)$

'pole placement procedure'

→ dsicretized PDE : $L = \gamma C^T \; (\gamma \in \mathbb{R})$

$$\|\epsilon(t)\| = \|e^{(A-LC)t}\epsilon(0)\| \leq \|x(0) - \hat{x}(0)\| \cdot \kappa(X) \cdot e^{-\mu t}$$

$\rightarrow \mu = \min\{|\Lambda|\}$

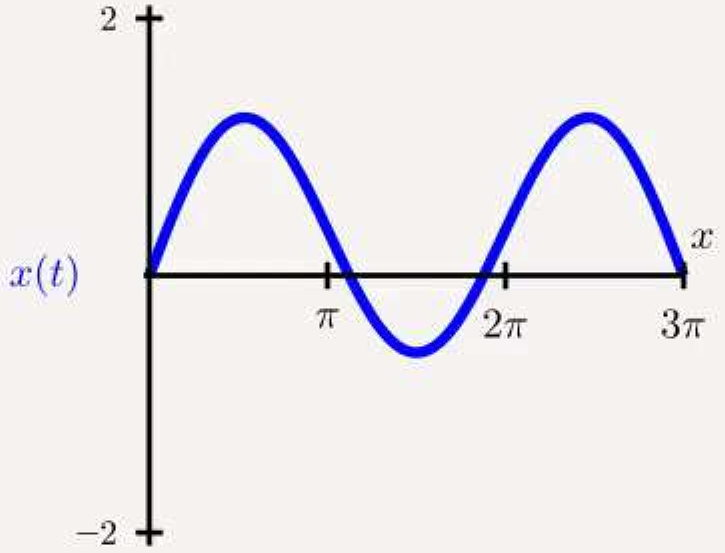
$\rightarrow X =$ e.v. of $A - LC$

[1] Kautsky, Nichols, Van Dooren.'Robust pole assignment in linear state feedback.' (1985)

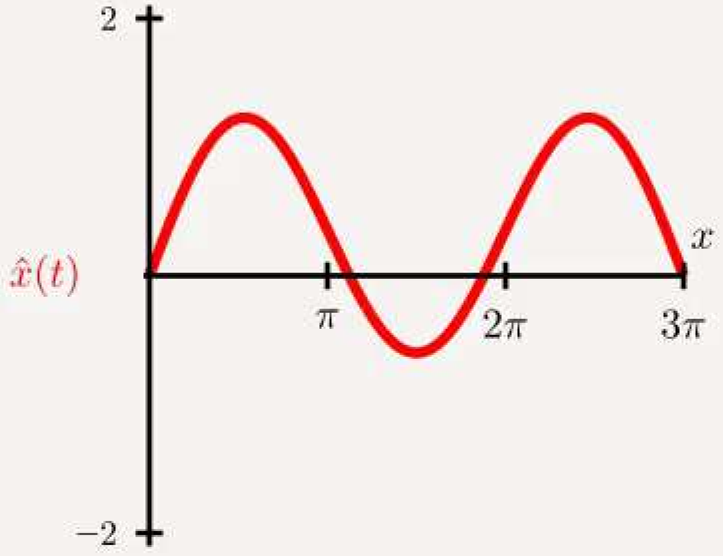
[2] Haine, Ramdani, 'Observateurs itératifs. en horizon fini. Application à la reconstruction de données initiales pour des EDP d'évolution.' (2011)

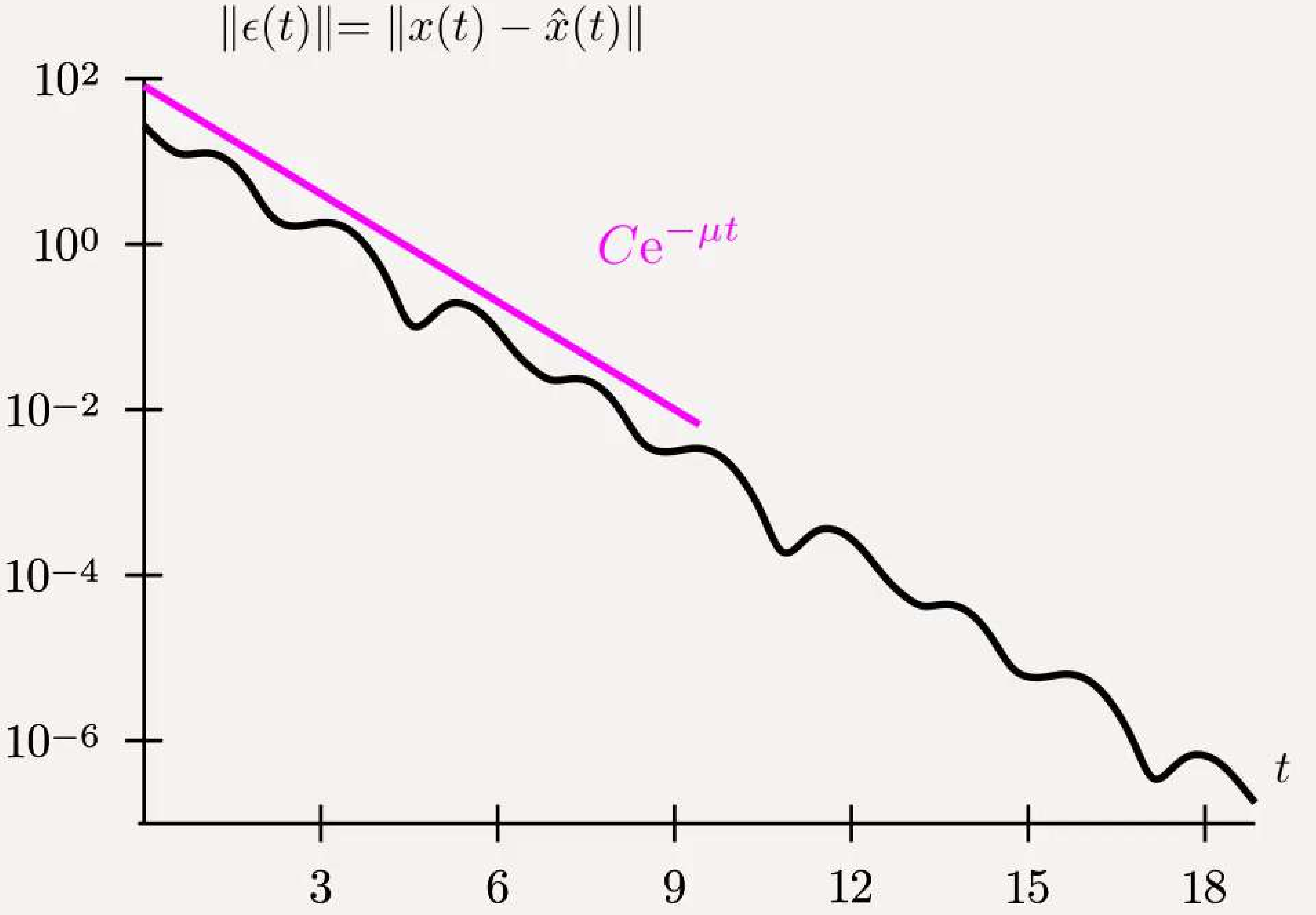
Data assimilation : Luenberger observer & dynamical systems

state dynamical system :



observer system :





$$\|\epsilon(t)\| = \|e^{(A-LC)t}\epsilon(0)\| \leq \|x(0) - \hat{x}(0)\| \cdot \kappa(X) \cdot e^{-\mu t}$$

Time parallelization : the Paraexp algorithm

$$\begin{cases} \dot{x}(t) = Mx(t) + g(t), \quad t \in [0,T] \\ x(0) = x_0 \end{cases} \qquad \begin{array}{l} \to M \in \mathcal{M}_{m\times m}(\mathbb{C}) \\ \to x(t), g(t) \in \mathbb{C}^m \end{array}$$

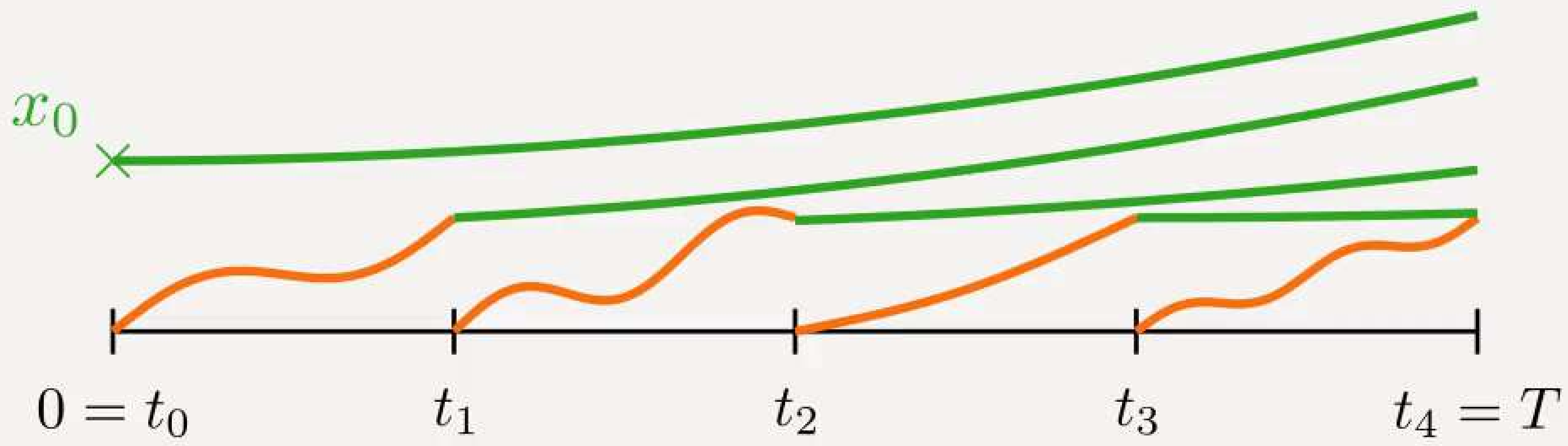
$$x(t) = v(t) + w(t)$$

Euler
Runge-Kutta $\longleftarrow$ $\begin{cases} \dot{v}(t) = M\,v(t) + g(t) \\ v(0) = 0 \end{cases}$ $\qquad$ $\begin{cases} \dot{w}(t) = M\,w(t) \\ w(0) = x_0 \end{cases}$ $\Rightarrow w(t) = \mathrm{e}^{(tM)} w(0)$

'Type 1' on $[t_{j-1}, t_j]$ $\qquad$ 'Type 2' on $[t_{j-1}, T]$

Rational Krylov
Chebyshev polynomials

$p = 4$ computers :



[3] Gander, Güttel. 'Paraexp : a parallel integrator for linear initial-value problems.' (2013)

Coupling PinT & Data assimilation

Objective : PinT(data assimilation)

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval

→ To optimize PinT, we want to start with a coarse approximation and refine it over time

→ We want to preserve the property of the data assimilation scheme : in our case the convergence rate μ

Coupling PinT & Data assimilation

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval

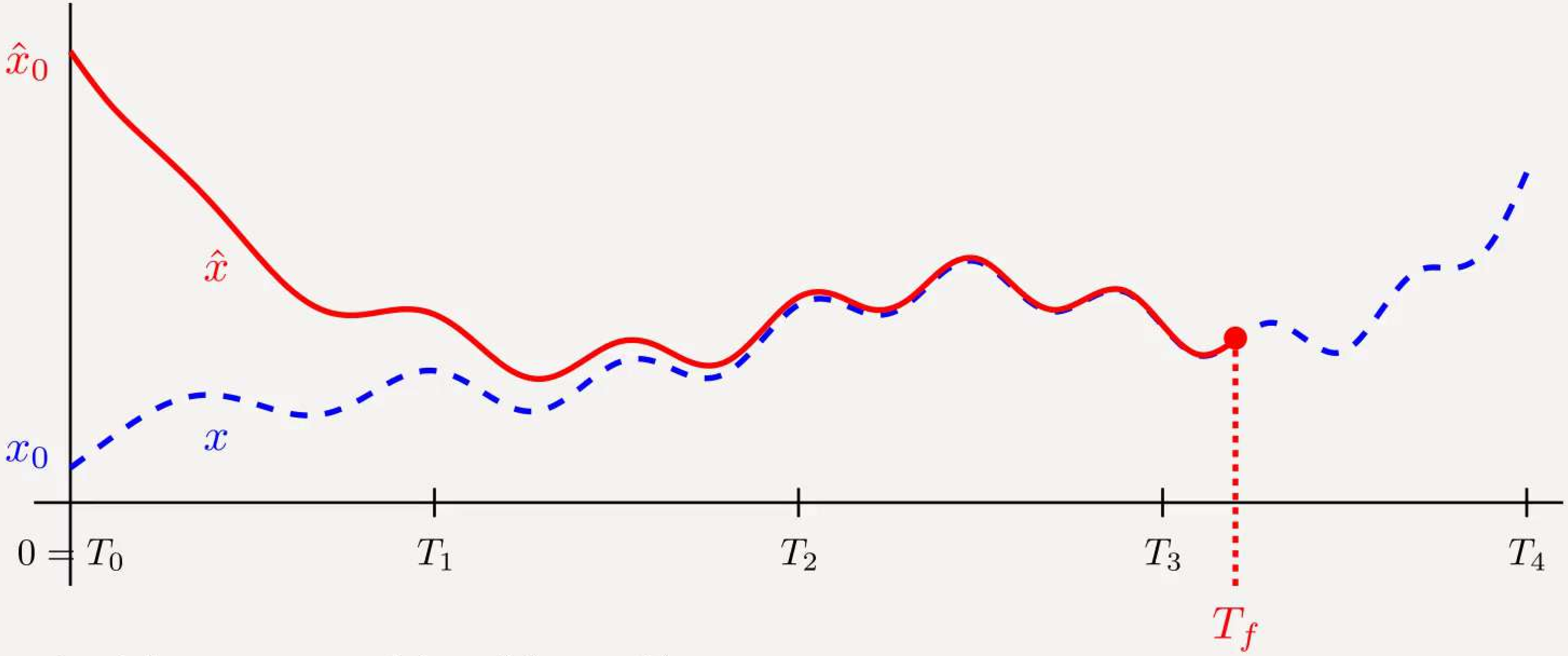
1) Divide the unbonded interval into 'windows' of size T :
$W_\ell = (T_{\ell-1}, T_\ell), \ell \leq 0$

2) Apply time parallelization scheme on each 'window'

3) Estimate the error at the end of each 'window' to go (or not) onto the next one

Coupling PinT & Data assimilation

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval

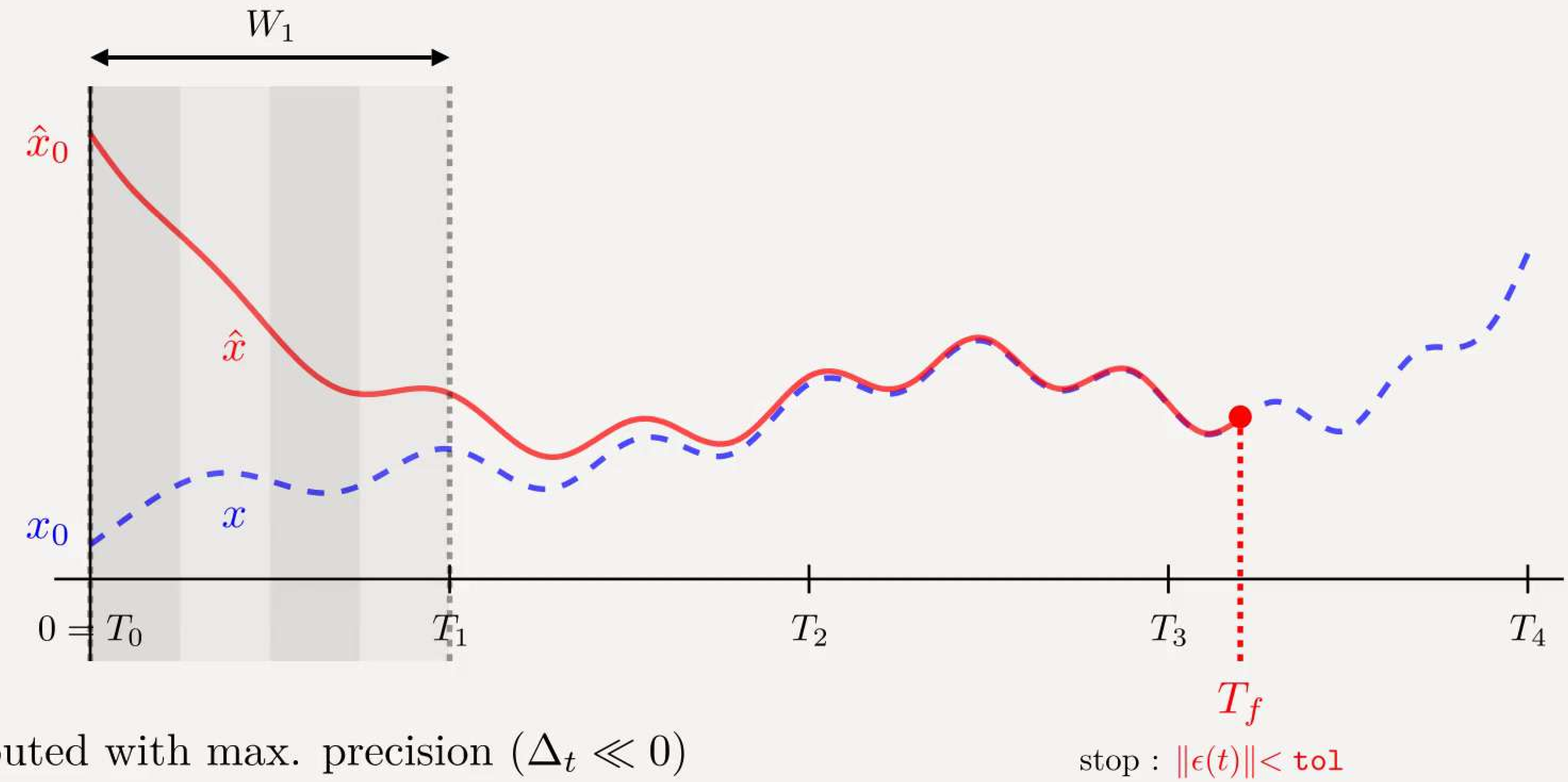


$\hat{x}$ computed with max. precision ($\Delta_t \ll 0$)

stop : $\|\epsilon(t)\| <$ `tol`

Coupling PinT & Data assimilation

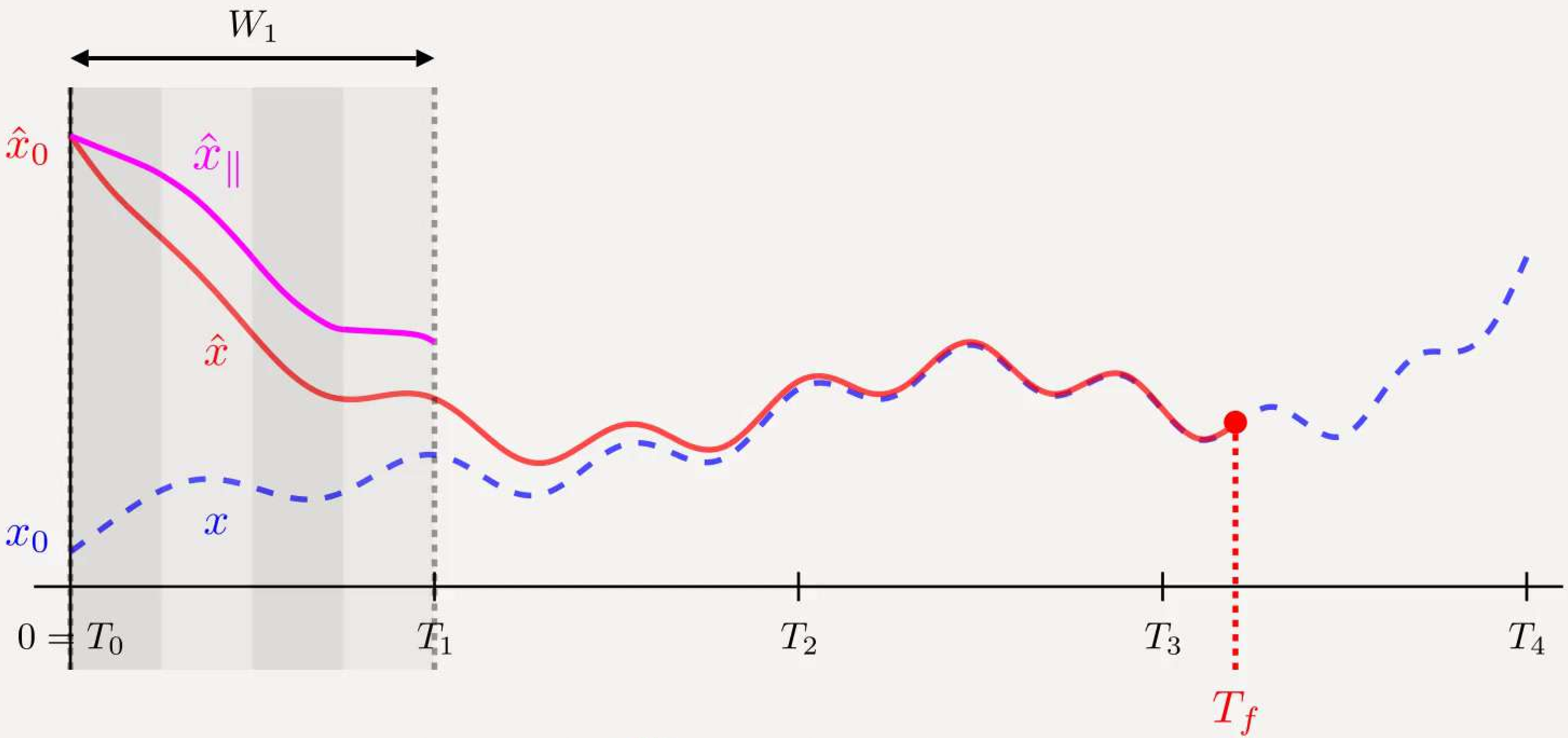
→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval



$\hat{x}$ computed with max. precision ($\Delta_t \ll 0$)

Coupling PinT & Data assimilation

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval

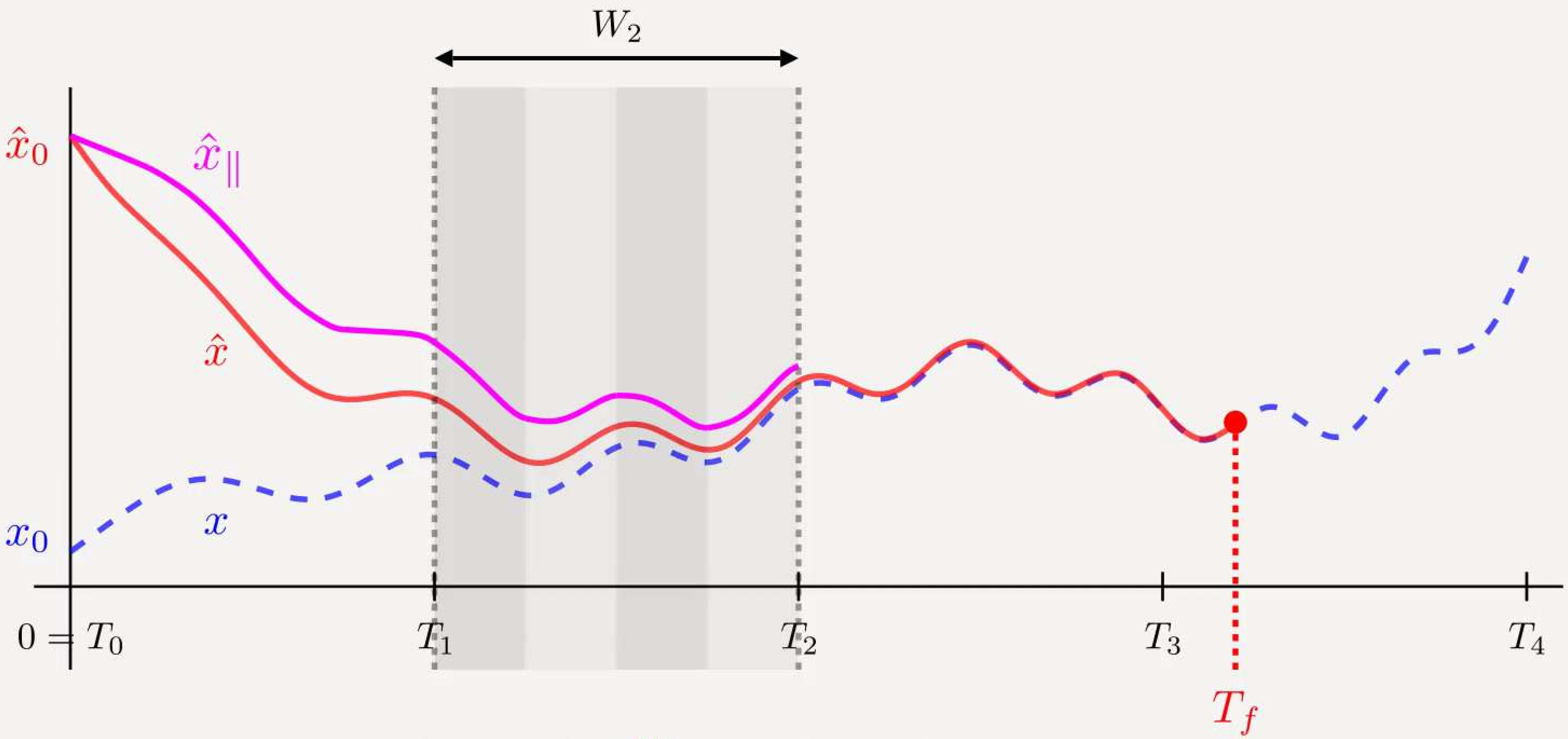


$\hat{x}_{\|}$: $\hat{x}_{\|}^{T1}$: with *some* precision ($\Delta_t \ll 0$), $\hat{x}_{\|}^{T2}$ exact (expm)

stop : $\|\epsilon(t)\| <$ `tol`

Coupling PinT & Data assimilation

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval

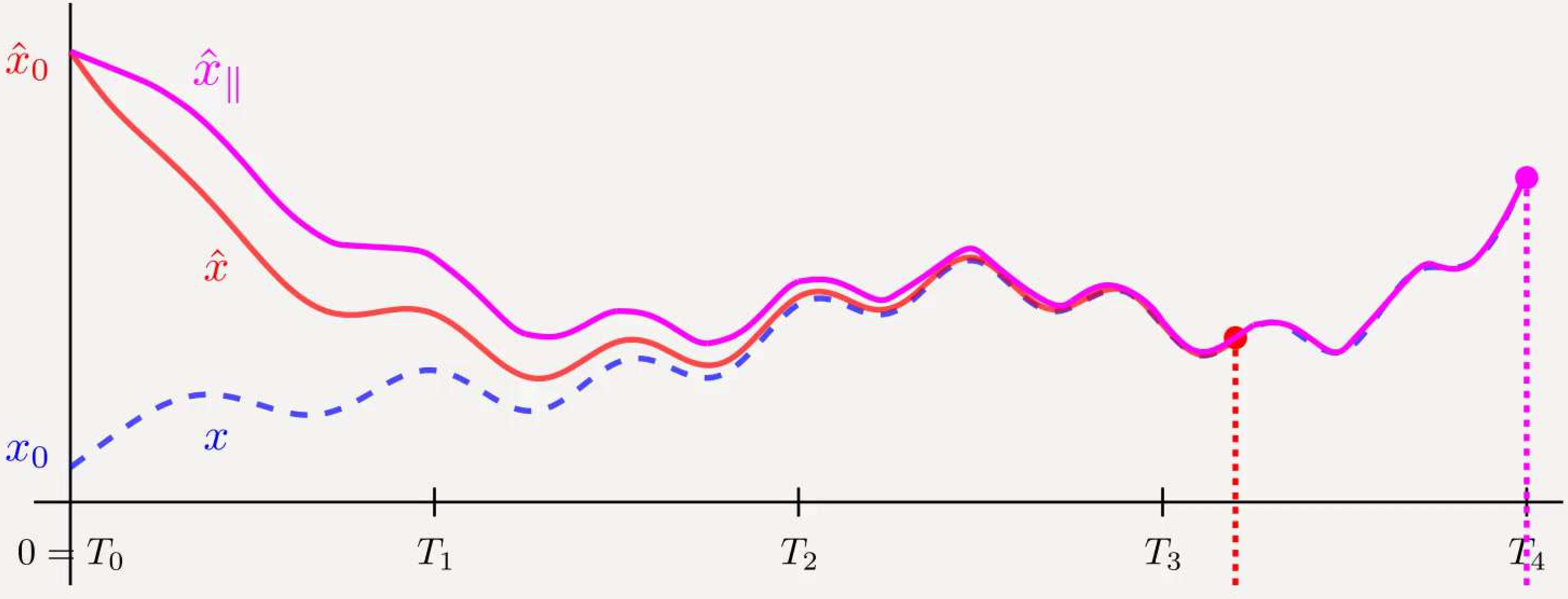


$\hat{x}_\parallel$: $\hat{x}_\parallel^{T1}$: with *some* precision ($\Delta_t \ll 0$), $\hat{x}_\parallel^{T2}$ exact (expm)

stop : $\|\epsilon(t)\| < \texttt{tol}$

Coupling PinT & Data assimilation

→ PinT algorithms are on a bounded time interval, data assimilation is on an unbounded time interval



$\hat{x}_\parallel$: $\hat{x}_\parallel^{T1}$: with *some* precision ($\Delta_t \ll 0$), $\hat{x}_\parallel^{T2}$ exact (expm)

stop : $\|\epsilon(t)\| <$ `tol` $\|\epsilon_\parallel(T_\ell)\| <$ `tol`

Coupling PinT & Data assimilation

→ To optimize PinT, we want to start with a coarse approximation and refine it over time, while conserving the convergence rate μ

$$\boxed{\|\epsilon_{\|}(T_\ell)\| = \|\hat{x}_{\|}(T_\ell) - x(T_\ell)\| \leq \|\epsilon(T_\ell)\| + \|\hat{x}(T_\ell) - \hat{x}_{\|}(T_\ell)\|}$$

$$\begin{aligned}
\epsilon_{\|} &= \hat{x}_{\|} - x \\
&= \hat{x}_{\|}^{T1} + \hat{x}_{\|}^{T2} - x \\
&= \hat{x}_{\|}^{T1} + \hat{x}_{\|}^{T2} - \hat{x}^{T1} + \hat{x}^{T1} - x \\
&= \hat{x}_{\|}^{T1} - \hat{x}^{T1} + \hat{x}_{\|}^{T2} + \hat{x}^{T1} - x \\
&= \hat{x}_{\|}^{T1} - \hat{x}^{T1} + \hat{x} - x
\end{aligned}$$

$$\boxed{\|\hat{x}(T_\ell) - \hat{x}_{\|}(T_\ell)\| = \|\hat{x}_{\|}^{T1}(T_\ell) - \hat{x}^{T1}(T_\ell)\|}$$

$$\boxed{\|\epsilon(T_\ell)\| \approx C\mathrm{e}^{-\mu T_\ell}}$$

$$\begin{cases} \hat{x}(T_\ell) = \hat{x}^{T1}(T_\ell) + \hat{x}^{T2}(T_\ell) \\ \hat{x}_{\|}(T_\ell) = \hat{x}_{\|}^{T1}(T_\ell) + \hat{x}_{\|}^{T2}(T_\ell) \end{cases}$$

Coupling PinT & Data assimilation

→ To optimize PinT, we want to start with a coarse approximation and refine it over time, while conserving the convergence rate μ

$$\|\epsilon(T_\ell)\| \approx C \mathrm{e}^{-\mu T_\ell}$$

$$\|\epsilon_{\|}(T_\ell)\| = \|\hat{x}_{\|}(T_\ell) - x(T_\ell)\| \leq \|\epsilon(T_\ell)\| + \|\hat{x}(T_\ell) - \hat{x}_{\|}(T_\ell)\|$$

$$\|\hat{x}(T_\ell) - \hat{x}_{\|}(T_\ell)\| = \|\hat{x}_{\|}^{T1}(T_\ell) - \hat{x}^{T1}(T_\ell)\|$$

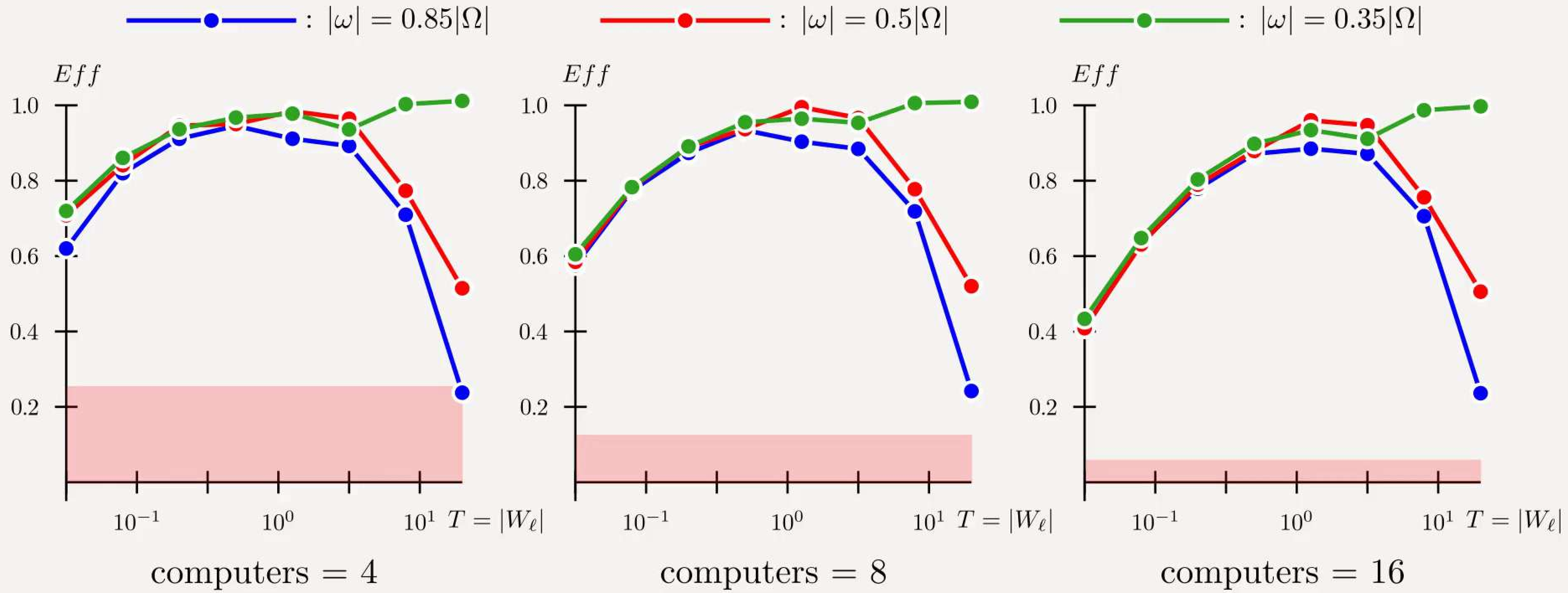
We must have $\|\hat{x}_{\|}^{T1}(T_\ell) - \hat{x}^{T1}(T_\ell)\| \approx C_{\|} \mathrm{e}^{-\mu T_\ell}$

If RK4 : $(\Delta_t)_{\ell+1} \leq \left(((\Delta_t)_\ell)^4 \mathrm{e}^{-\mu T}\right)^{1/4}, \quad \forall \ell \leq 1$

Results : a wave equation

2D Wave eq., on $\Omega = [0, 2\pi]^2$, $N_x = 9$, obs. space : ω

$$\text{Efficiency} = \frac{\textit{cputime(non-parallel)}}{\#\ \textit{computers} \times \textit{cputime(parallel)}}$$



Results : following & leads

$\rightarrow$ Works similarly for heat equation (1D & 2D)

$\rightarrow$ Application to Linear Wave Theory (LWT) : (in progress with N. Desmars)

- Convergence of the observer is not clear ('floor')
- Such 'floor' depends on the waves & obs. domain (complexity, frequencies, amplitudes, etc...)
- Observability / controlability of the eq. system ?
- Choice of L ?

Thanks for your attention !

[1] Kautsky, Nichols, Van Dooren. 'Robust pole assignment in linear state feedback.' (1985).

[2] Haine, Ramdani. 'Observateurs itératifs, en horizon fini. Application à la reconstruction de données initiales pour des EDP d'évolution.' (2011).

[3] Gander, Güttel. 'Paraexp : a parallel integrator for linear initial-value problems.' (2013).

[4] The Manim Community Developers. Manim – Mathematical Animation Framework (Version v0.15.2). https://www.manim.community/. (2022).

Manim