Coupling large and small scale shallow water models with porosity in the presence of anisotropy

Ph.D. defense of
João Guilherme CALDAS STEINSTRAESSER

under the supervision of
Vincent GUINOT and Antoine ROUSSEAU

1 October 2021

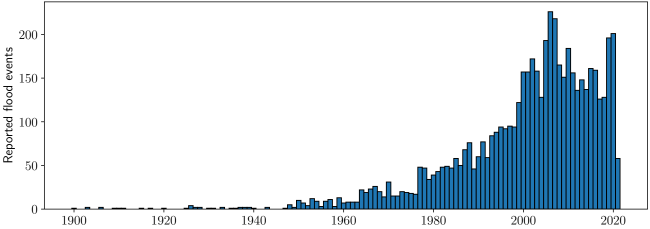
Introduction

Introduction

- Increasing impacts and frequency of urban floods

Introduction

- Increasing impacts and frequency of urban floods



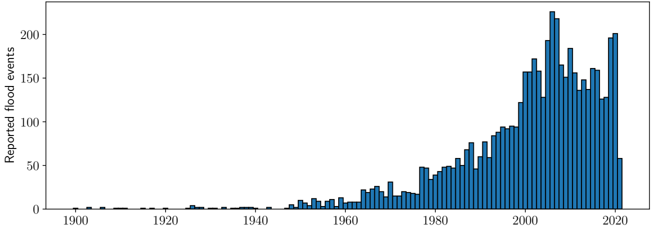
[EM-DAT, CRED]

Introduction

- Increasing impacts and frequency of urban floods

Floods in 1995-2015 (CRED & UNISDR, 2015)

- 47% of weather-related disasters
- 2.3 billion people affected



[EM-DAT, CRED]

Introduction

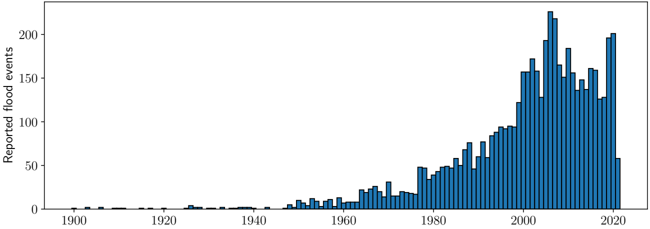
- Increasing impacts and frequency of urban floods

Floods in 1995-2015 (CRED & UNISDR, 2015)

- 47% of weather-related disasters
- 2.3 billion people affected

Urban population (United Nations, 2019)

- Today: 55%
- 2050: 68%



[EM-DAT, CRED]

Introduction

- Increasing impacts and frequency of urban floods
- Numerical simulation using the **shallow water equations (SWE)**

$$\frac{\partial}{\partial t}\boldsymbol{U}(t) + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}(t)) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}(t)) = \boldsymbol{S}(\boldsymbol{U}(t)), \qquad \boldsymbol{U} = \begin{pmatrix} h \\ hu_x \\ hu_y \end{pmatrix}$$

Introduction

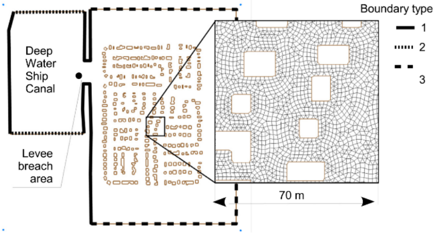
- Increasing impacts and frequency of urban floods
- Numerical simulation using the **shallow water equations (SWE)**
- Accurate results: high computational cost;

$$\frac{\partial}{\partial t}\boldsymbol{U}(t) + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}(t)) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}(t)) = \boldsymbol{S}(\boldsymbol{U}(t)), \qquad \boldsymbol{U} = \begin{pmatrix} h \\ hu_x \\ hu_y \end{pmatrix}$$

Introduction

- Increasing impacts and frequency of urban floods
- Numerical simulation using the **shallow water equations (SWE)**
- Accurate results: high computational cost;

(Guinot et al., 2017)

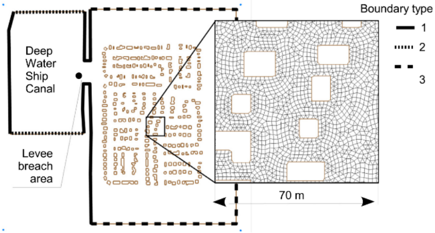


(Guinot et al., 2017)

Introduction

- Increasing impacts and frequency of urban floods
- Numerical simulation using the **shallow water equations (SWE)**
- Accurate results: high computational cost;
- Limited operational application

(Guinot et al., 2017)



(Guinot et al., 2017)

Introduction

- Alternative: **porosity-based SWE**

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.

The classical SWE

$$\frac{\partial}{\partial t}\boldsymbol{U}(t) + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}(t)) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}(t)) = \boldsymbol{S}(\boldsymbol{U}(t))$$

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.

The porosity-based SWE (single porosity model; Guinot and Soares-Frazão, 2006)

$$\frac{\partial}{\partial t}\phi\boldsymbol{U}(t) + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}(t)) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}(t)) = \overline{\boldsymbol{S}}(\boldsymbol{U}(t))$$

Introduction

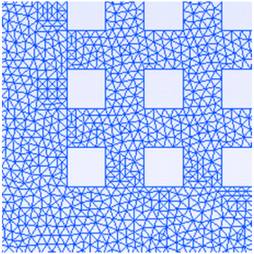
- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.
  - Porosity coefficient $\phi(x,y) \in [0,1]$ for representing the urban geometry;

The porosity-based SWE (single porosity model; Guinot and Soares-Frazão, 2006)

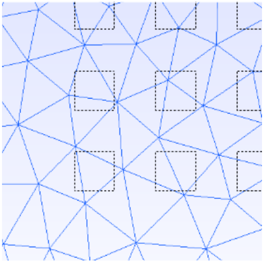
$$\frac{\partial}{\partial t}\phi\boldsymbol{U}(t) + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}(t)) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}(t)) = \overline{\boldsymbol{S}}(\boldsymbol{U}(t))$$

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.
  - Porosity coefficient $\phi(x, y) \in [0, 1]$ for representing the urban geometry;

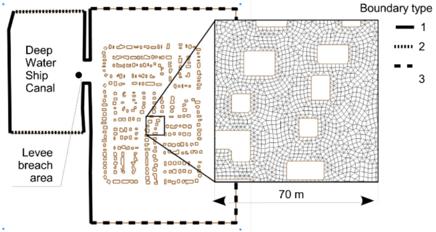


$$\Longleftrightarrow$$

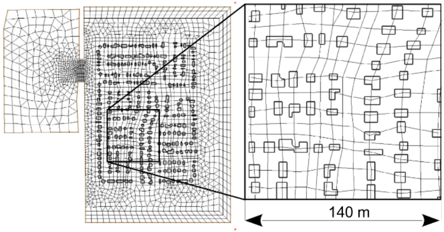


Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.
  - Porosity coefficient $\phi(x,y) \in [0,1]$ for representing the urban geometry;
- Coarser mesh, larger time step $\implies$ smaller computational cost;



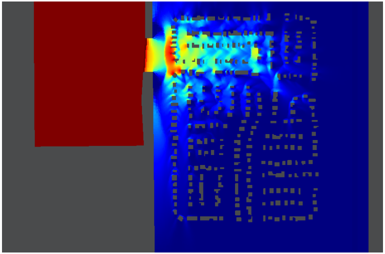
(Guinot et al., 2017)



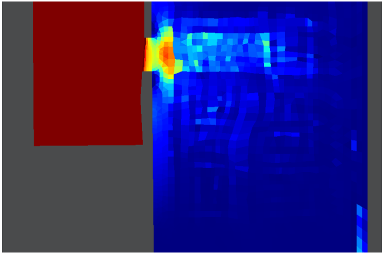
(Guinot et al., 2017)

Introduction

- Alternative: **porosity-based SWE**
  - Defina et al., 1994; Guinot and Soares-Frazão, 2006; Sanders et al., 2008; Guinot, 2012; Guinot et al., 2017; Velickovic et al., 2017;...
  - Spatially averaged, upscaled equations
  - Urban zone: porous media.
  - Porosity coefficient $\phi(x,y) \in [0,1]$ for representing the urban geometry;
- Coarser mesh, larger time step $\implies$ smaller computational cost;
- Good global approximations, but less accurate inside the urban zone.



Computational time: $794s$



Computational time: $14s$

Introduction

- How to improve results provided by the porosity-based SWE?

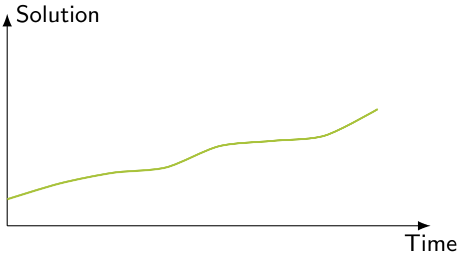
Introduction

- How to improve results provided by the porosity-based SWE?

Introduction

- How to improve results provided by the porosity-based SWE?

Coarse model
low-expensive approximation

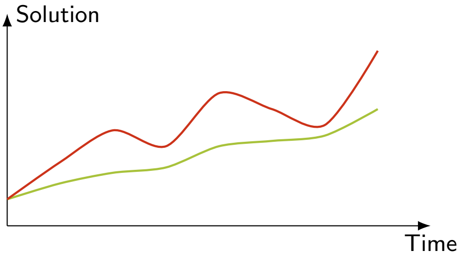


Introduction

- How to improve results provided by the porosity-based SWE?

Coarse model
low-expensive approximation

Fine model
more accurate, more expensive

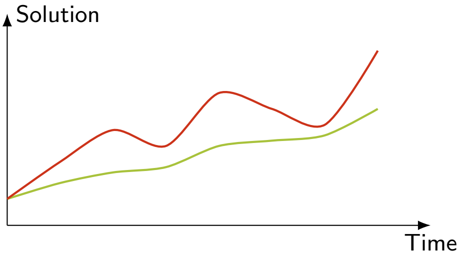


Introduction

- How to improve results provided by the porosity-based SWE?
- **Predictor-corrector iterative parallel-in-time methods**

Coarse model
low-expensive approximation

Fine model
more accurate, more expensive

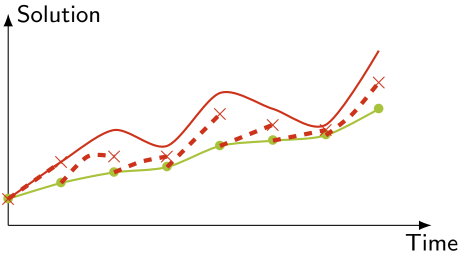


Introduction

- How to improve results provided by the porosity-based SWE?
- **Predictor-corrector iterative parallel-in-time methods**

Prediction
Coarse model
low-expensive approximation

Correction
Fine model
more accurate, more expensive

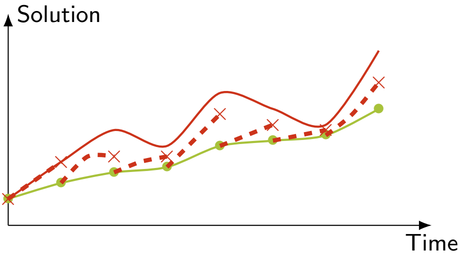


Introduction

- How to improve results provided by the porosity-based SWE?
- **Predictor-corrector iterative parallel-in-time methods**
  - Parareal, PITA, PFASST, MGRIT,...

Prediction
Coarse model
low-expensive approximation

Correction
Fine model
more accurate, more expensive

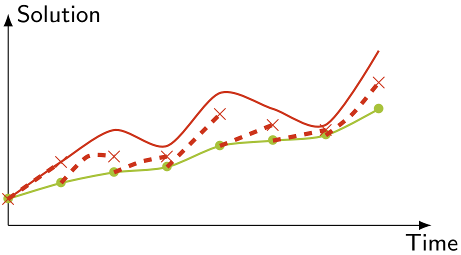


Introduction

- How to improve results provided by the porosity-based SWE?
- **Predictor-corrector iterative parallel-in-time methods**
  - Parareal, PITA, PFASST, MGRIT,...
  - Well-known issues when applied to hyperbolic problems.

Prediction
Coarse model
low-expensive approximation

Correction
Fine model
more accurate, more expensive

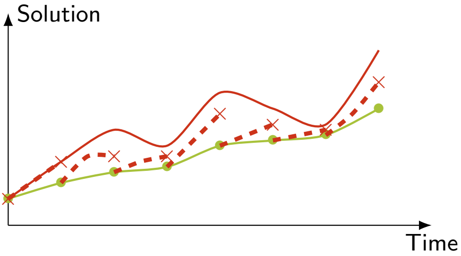


Introduction

- How to improve results provided by the porosity-based SWE?
- **Predictor-corrector iterative parallel-in-time methods**
  - Parareal, PITA, PFASST, MGRIT,...
  - Well-known issues when applied to hyperbolic problems.
  - Alternatives and adaptations

Prediction
Coarse model
low-expensive approximation

Correction
Fine model
more accurate, more expensive



Objectives

Objectives

- Couple classical and porosity-based SWE using **parareal methods**

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**
    - Classical and ROM-based parareal methods

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**
    - Classical and ROM-based parareal methods
    - Compare methods in terms of convergence, numerical cost, stability

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**
    - Classical and ROM-based parareal methods
    - Compare methods in terms of convergence, numerical cost, stability
    - Identify limitations and improve methods

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**
    - Classical and ROM-based parareal methods
    - Compare methods in terms of convergence, numerical cost, stability
    - Identify limitations and improve methods
  - **Application to the simulation of urban floods**
    - Identify additional challenges and improvement opportunities

Objectives

- Couple classical and porosity-based SWE using **parareal methods**
  - **Review and improvements of parareal methods**
    - Classical and ROM-based parareal methods
    - Compare methods in terms of convergence, numerical cost, stability
    - Identify limitations and improve methods
  - **Application to the simulation of urban floods**
    - Identify additional challenges and improvement opportunities

- **Contribution to the SW2D software**
  - Developed by Inria LEMON team
  - Classical and porosity-based SWE
  - Explicit finite volumes discretization;

sw2d.inria.fr

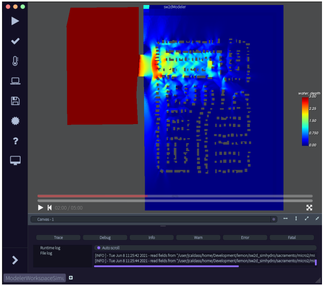


Table of contents

Table of contents

The parareal method

[Lions et al., 2001]

The parareal method

[Lions et al., 2001]

The parareal method

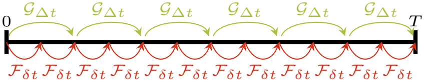
[Lions et al., 2001]

- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive

The parareal method

[Lions et al., 2001]

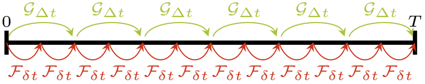
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate



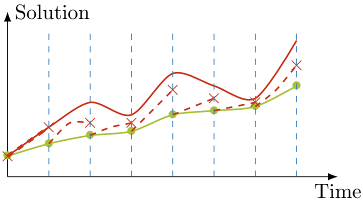
The parareal method

[Lions et al., 2001]

- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate



Predictor-corrector iterative method



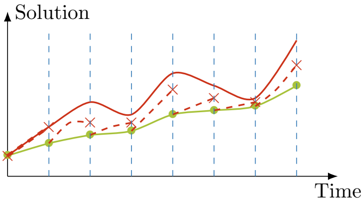
The parareal method

[Lions et al., 2001]

- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate

Predictor-corrector iterative method

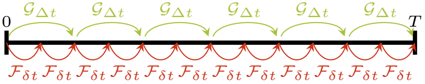
- $N_{\Delta T}$ time slices



The parareal method

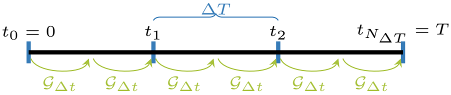
[Lions et al., 2001]

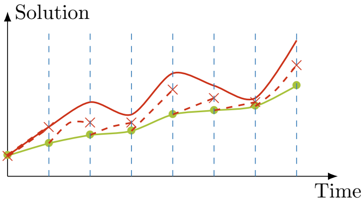
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate



Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions

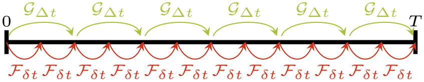




The parareal method

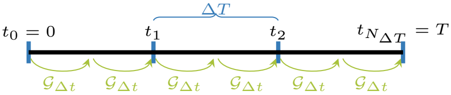
[Lions et al., 2001]

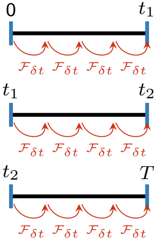
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate

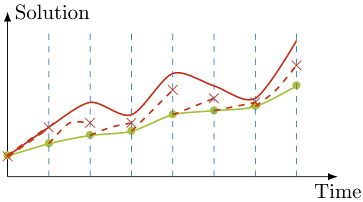


Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions
- **Parallel** fine corrections



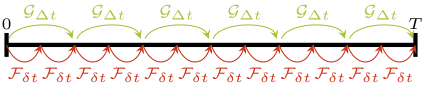




The parareal method

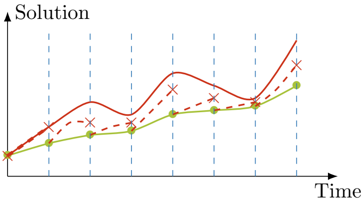
[Lions et al., 2001]

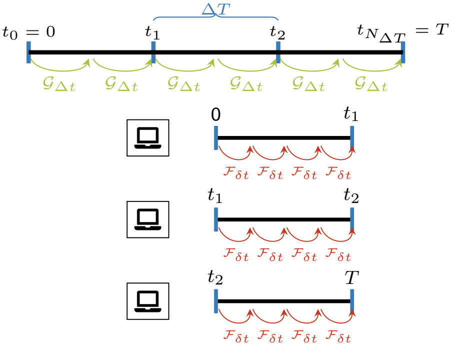
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate



Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions
- **Parallel** fine corrections

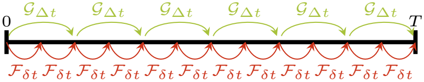




The parareal method

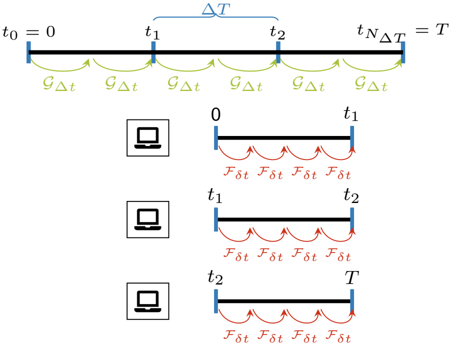
[Lions et al., 2001]

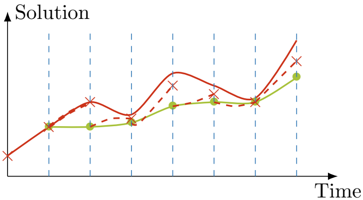
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate



Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions
- **Parallel** fine corrections

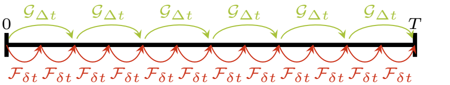




The parareal method

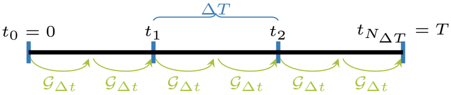
[Lions et al., 2001]

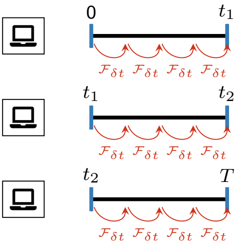
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate

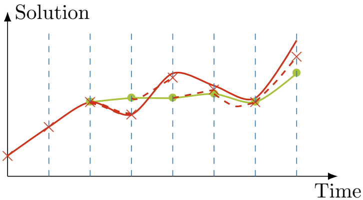


Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions
- **Parallel** fine corrections



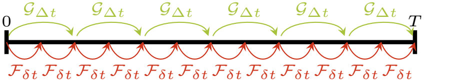




The parareal method

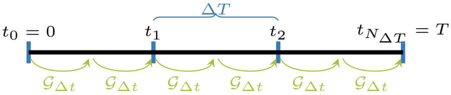
[Lions et al., 2001]

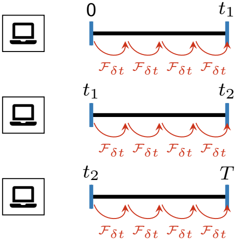
- $\mathcal{F}_{\delta t}$: A **fine discretization**
  - Accurate but too expensive
- $\mathcal{G}_{\Delta t}$: A **coarser discretization**
  - Much cheaper but less accurate

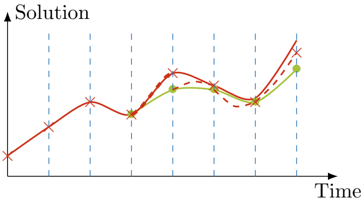


Predictor-corrector iterative method

- $N_{\Delta T}$ time slices
- Sequential coarse predictions
- **Parallel** fine corrections







Performance of parallel-in-time methods

Performance of parallel-in-time methods

- Objective: convergence in few ($\ll N_{\Delta T}$) iterations;

Performance of parallel-in-time methods

- Objective: convergence in few ($\ll N_{\Delta T}$) iterations;
- Fast convergence for parabolic, diffusive problems;

Performance of parallel-in-time methods

- Objective: convergence in few ($\ll N_{\Delta T}$) iterations;
- Fast convergence for parabolic, diffusive problems;
- In the case of hyperbolic problems: slow convergence, instabilities

Performance of parallel-in-time methods

- Objective: convergence in few ($\ll N_{\Delta T}$) iterations;
- Fast convergence for parabolic, diffusive problems;
- In the case of hyperbolic problems: slow convergence, instabilities
- Causes (Ruprecht, 2018):
  - Mismatch of discrete phase speeds between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$
  - Mainly on high wavenumbers (damped in parabolic problems);

The ROM-based parareal method

[Chen et al., 2014]

The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;

$$\frac{dy_1}{dt} = Ay_1 + F(y_1)$$
$$\frac{dy_2}{dt} = Ay_2 + F(y_2)$$
$$\vdots$$
$$\vdots$$
$$\frac{dy_M}{dt} = Ay_M + F(y_M)$$

$$\frac{d\widetilde{y}_1}{dt} = \widetilde{A}\widetilde{y}_1 + \widetilde{F}(\widetilde{y}_1)$$
$$\vdots$$
$$\frac{d\widetilde{y}_m}{dt} = \widetilde{A}\widetilde{y}_m + \widetilde{F}(\widetilde{y}_m)$$

$$m \ll M$$

The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;

$$\begin{aligned} \frac{dy_1}{dt} &= Ay_1 + F(y_1) \\ \frac{dy_2}{dt} &= Ay_2 + F(y_2) \\ &\vdots \\ &\vdots \\ \frac{dy_M}{dt} &= Ay_M + F(y_M) \end{aligned}$$

$$\begin{aligned} \frac{d\widetilde{y}_1}{dt} &= \widetilde{A}\widetilde{y}_1 + \widetilde{F}(\widetilde{y}_1) \\ &\vdots \\ \frac{d\widetilde{y}_m}{dt} &= \widetilde{A}\widetilde{y}_m + \widetilde{F}(\widetilde{y}_m) \end{aligned}$$

$$m \ll M$$

The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;

Fine simulations

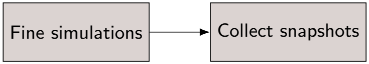
The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;



The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;

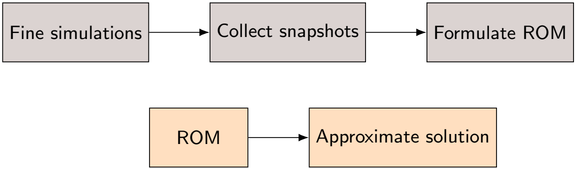
The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;



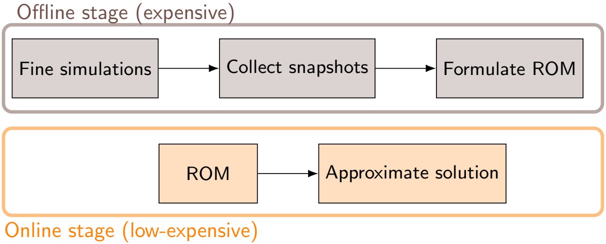
The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;
- Offline-online procedure.



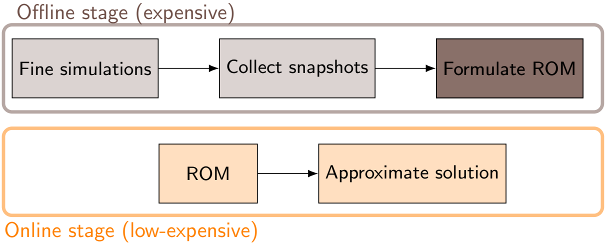
The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;
- Offline-online procedure.



The ROM-based parareal method

[Chen et al., 2014]

- Idea: improve coarse prediction using **reduced-order models (ROMs)**;

Model reduction

- Low-dimensional approximation to an expensive problem;
- Constructed from **snapshots** of the solution;
- Offline-online procedure.

- **Nonlinear problems**: combined POD-EIM
  (Barrault et al., 2004; Chaturantabut & Sorensen, 2010)
  - **POD**: proper orthogonal decomposition
  - **EIM**: empirical interpolation method

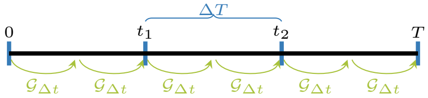
The ROM-based parareal method

[Chen et al., 2014]

The ROM-based parareal method

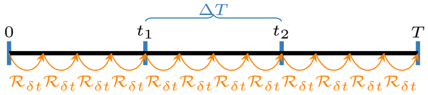
[Chen et al., 2014]

- $\mathcal{G}_{\Delta t}$: only the initial prediction;



The ROM-based parareal method

[Chen et al., 2014]

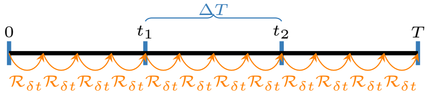


- $\mathcal{G}_{\Delta t}$: only the initial prediction;
- **ROM-based parareal iteration**: coarse model replaced by a ROM
  - Solved using small δt;
  - Reformulated **on-the-fly** at each iteration.

.

The ROM-based parareal method

[Chen et al., 2014]

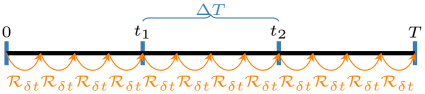


- $\mathcal{G}_{\Delta t}$: only the initial prediction;
- **ROM-based parareal iteration**: coarse model replaced by a ROM
  - Solved using small δt;
  - Reformulated **on-the-fly** at each iteration.

.

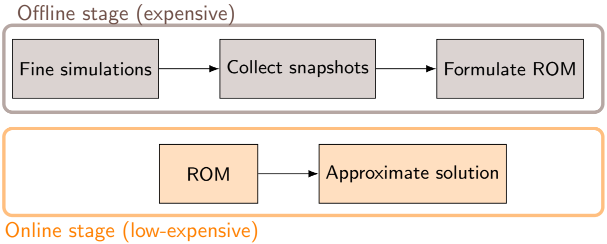
The ROM-based parareal method

[Chen et al., 2014]



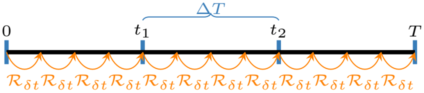
- $\mathcal{G}_{\Delta t}$: only the initial prediction;
- **ROM-based parareal iteration**: coarse model replaced by a ROM
  - Solved using small δt;
  - Reformulated **on-the-fly** at each iteration.

.



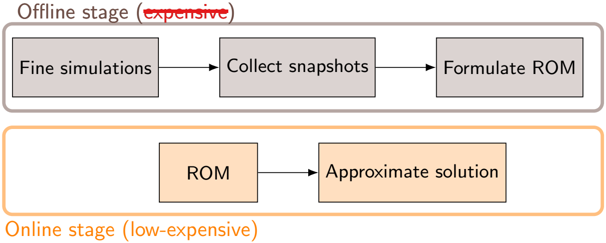
The ROM-based parareal method

[Chen et al., 2014]



- $\mathcal{G}_{\Delta t}$: only the initial prediction;
- **ROM-based parareal iteration**: coarse model replaced by a ROM
  - Solved using small δt;
  - Reformulated **on-the-fly** at each iteration.

.

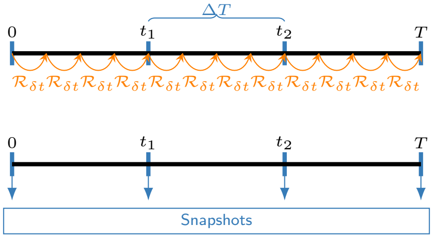


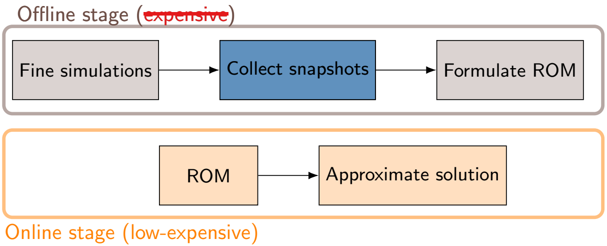
The ROM-based parareal method

[Chen et al., 2014]

- $\mathcal{G}_{\Delta t}$: only the initial prediction;
- **ROM-based parareal iteration**: coarse model replaced by a ROM
  - Solved using small δt;
  - Reformulated **on-the-fly** at each iteration.

.





First comparisons between the classical and ROM-based parareal

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization
- Reference solution:

$$\begin{cases} \boldsymbol{y}_{\text{ref},0} = \boldsymbol{y}_0 \\ \boldsymbol{y}_{\text{ref},n+1} = \mathcal{F}_{\delta t}(\boldsymbol{y}_{\text{ref},n}) \qquad n = 0, \dots, N_{\Delta T} - 1 \end{cases}$$

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization
- Reference solution:

$$\begin{cases} \boldsymbol{y}_{\text{ref},0} = \boldsymbol{y}_0 \\ \boldsymbol{y}_{\text{ref},n+1} = \mathcal{F}_{\delta t}(\boldsymbol{y}_{\text{ref},n}) \qquad n = 0, \dots, N_{\Delta T} - 1 \end{cases}$$

- What to compare?

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization
- Reference solution:

$$\begin{cases} \boldsymbol{y}_{\text{ref},0} = \boldsymbol{y}_0 \\ \boldsymbol{y}_{\text{ref},n+1} = \mathcal{F}_{\delta t}(\boldsymbol{y}_{\text{ref},n}) \qquad n = 0, \dots, N_{\Delta T} - 1 \end{cases}$$

- What to compare?
  - Speed of convergence

First comparisons between the classical and ROM-based parareal

- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization
- Reference solution:

$$\begin{cases} \boldsymbol{y}_{\text{ref},0} = \boldsymbol{y}_0 \\ \boldsymbol{y}_{\text{ref},n+1} = \mathcal{F}_{\delta t}(\boldsymbol{y}_{\text{ref},n}) \qquad n = 0, \dots, N_{\Delta T} - 1 \end{cases}$$

- What to compare?
  - Speed of convergence
  - Numerical speedup;

First comparisons between the classical and ROM-based parareal

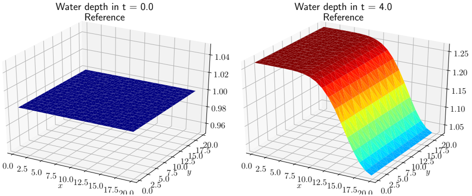
- Simple tests; both $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ are discretizations of the classical SWE
- Explicit FV discretization
- Reference solution:

$$\begin{cases} \boldsymbol{y}_{\text{ref},0} = \boldsymbol{y}_0 \\ \boldsymbol{y}_{\text{ref},n+1} = \mathcal{F}_{\delta t}(\boldsymbol{y}_{\text{ref},n}) \qquad n = 0, \dots, N_{\Delta T} - 1 \end{cases}$$

- What to compare?
  - Speed of convergence
  - Numerical speedup;
  - Stability

First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

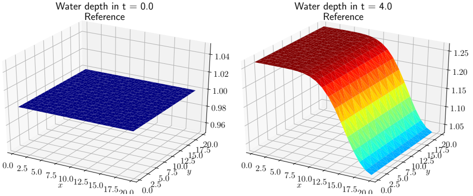


First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors

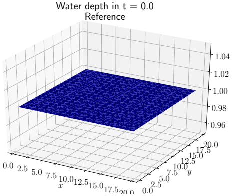


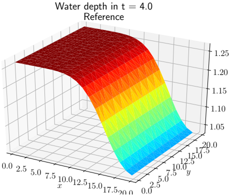
First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

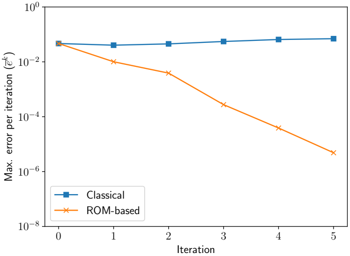
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors





Errors

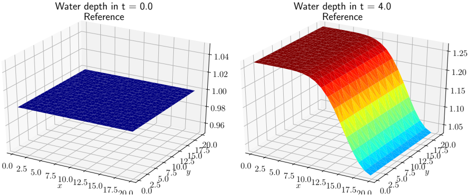


First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

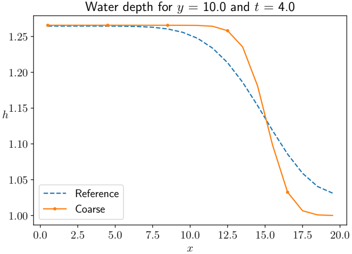
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal

ROM-based parareal

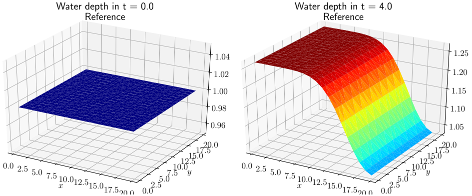


First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

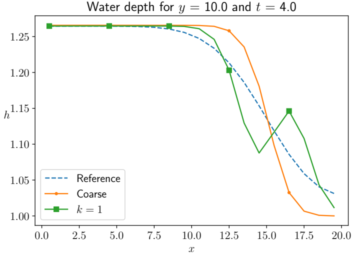
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal

ROM-based parareal

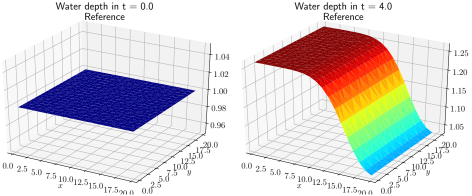


First comparisons between the classical and ROM-based parareal

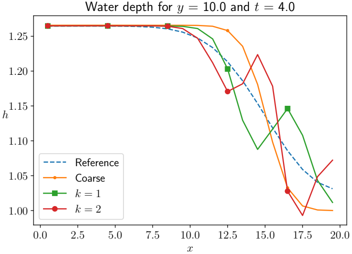
- Pseudo-2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal



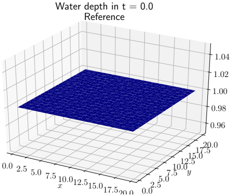
ROM-based parareal

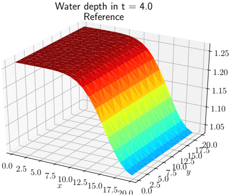
First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

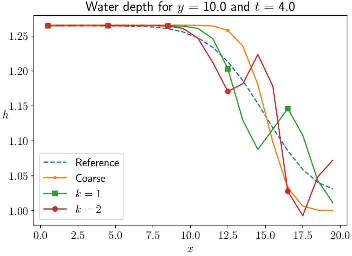
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors

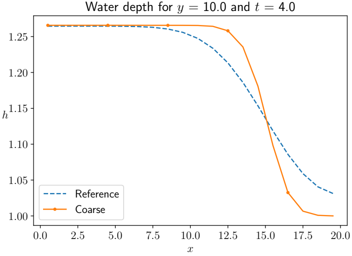




Classical parareal



ROM-based parareal

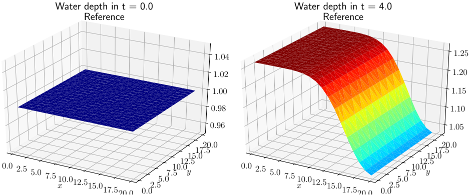


First comparisons between the classical and ROM-based parareal

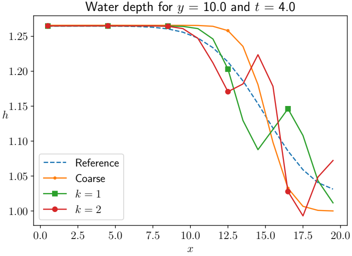
- Pseudo-2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

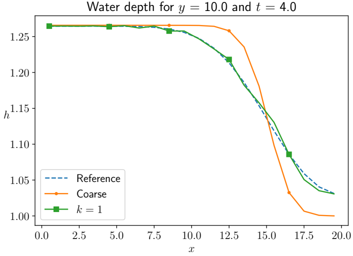
$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal



ROM-based parareal

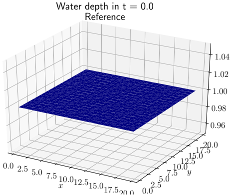


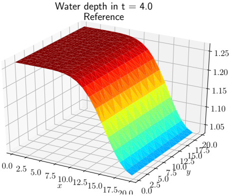
First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

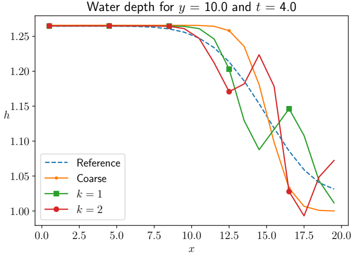
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors

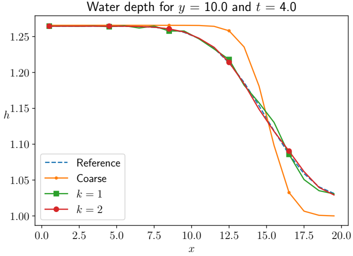




Classical parareal



ROM-based parareal

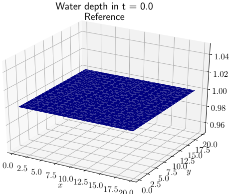


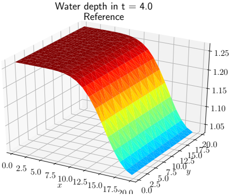
First comparisons between the classical and ROM-based parareal

- Pseudo-2D test case:

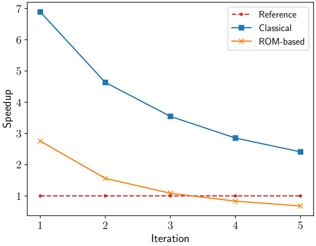
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.2$
$\delta x = 1$	$\Delta x = 1$

$T = 4$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



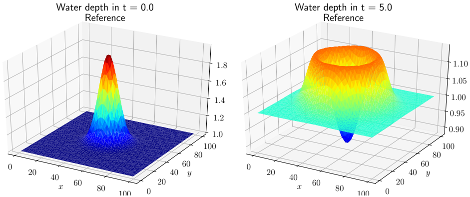


Speedup



First comparisons between the classical and ROM-based parareal

- 2D test case:

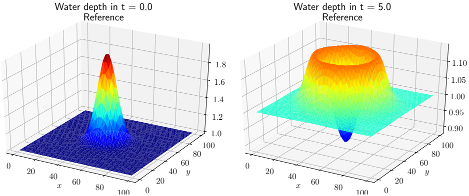


First comparisons between the classical and ROM-based parareal

- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors

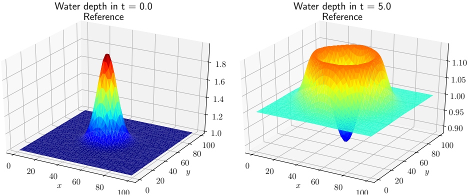


First comparisons between the classical and ROM-based parareal

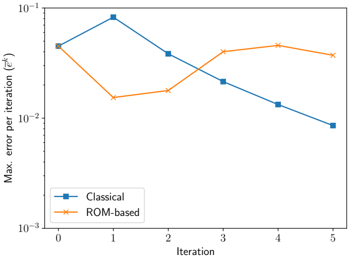
- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Errors per iteration

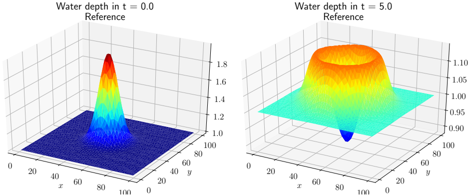


First comparisons between the classical and ROM-based parareal

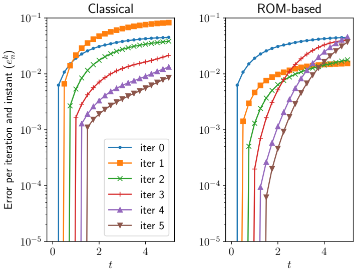
- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Errors per iteration and time

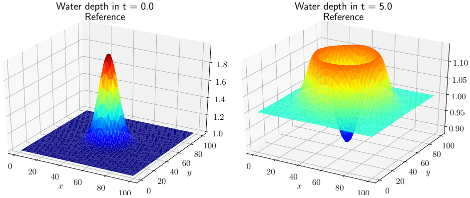


First comparisons between the classical and ROM-based parareal

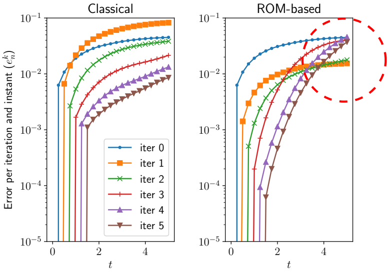
- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Errors per iteration and time

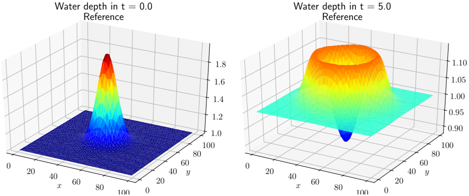


First comparisons between the classical and ROM-based parareal

- 2D test case:

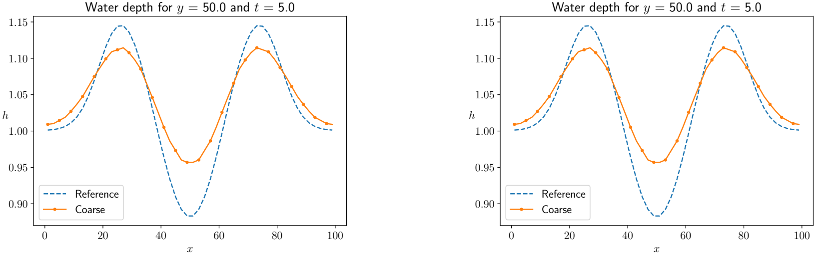
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal

ROM-based parareal

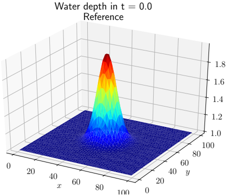


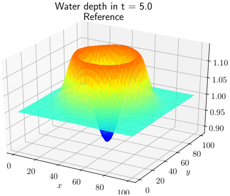
First comparisons between the classical and ROM-based parareal

- 2D test case:

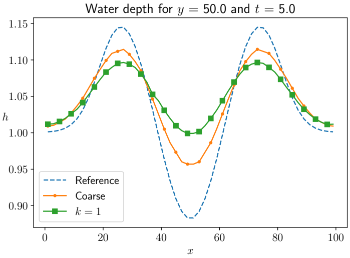
$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors

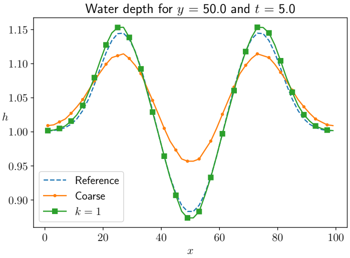




Classical parareal



ROM-based parareal

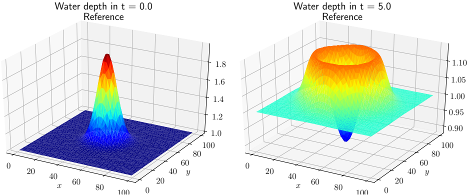


First comparisons between the classical and ROM-based parareal

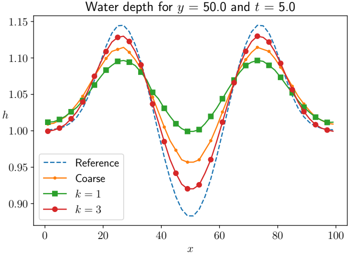
- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

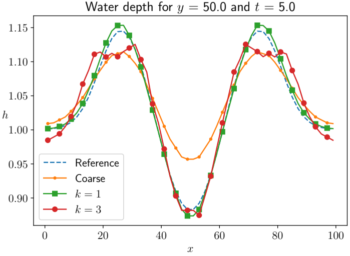
$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal



ROM-based parareal

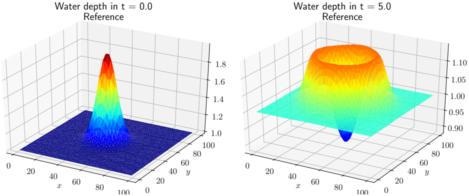


First comparisons between the classical and ROM-based parareal

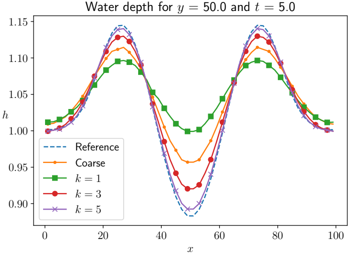
- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

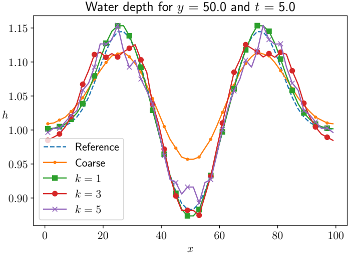
$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Classical parareal



ROM-based parareal

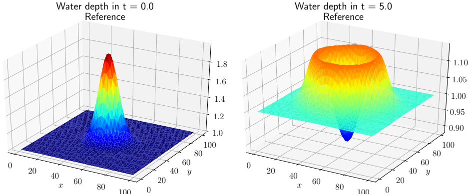


First comparisons between the classical and ROM-based parareal

- 2D test case:

$\mathcal{F}_{\delta t}$	$\mathcal{G}_{\Delta t}$
$\delta t = 0.001$	$\Delta t = 0.25$
$\delta x = 2$	$\Delta x = 5$

$T = 5$, $N_{\Delta T} = 20$ time slices,
$P = 20$ processors



Speedup

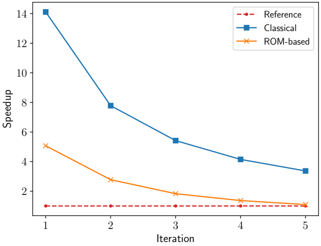


Table of contents

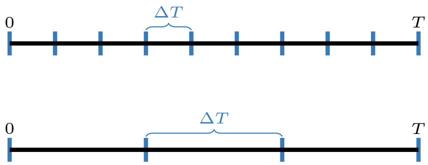
How to improve the parareal performance?

How to improve the parareal performance?

- Increase time slice length (Ruprecht, 2018)

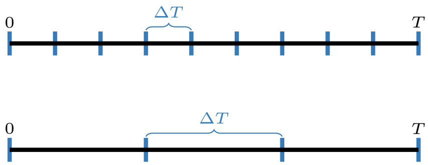
How to improve the parareal performance?

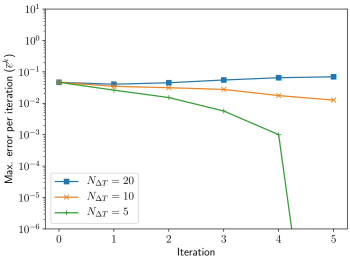
- Increase time slice length (Ruprecht, 2018)



How to improve the parareal performance?

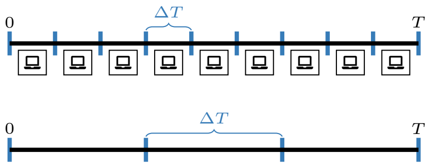
- Increase time slice length (Ruprecht, 2018)

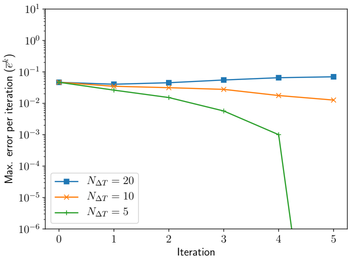




How to improve the parareal performance?

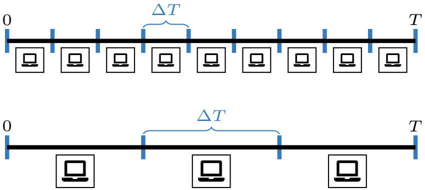
- Increase time slice length (Ruprecht, 2018)

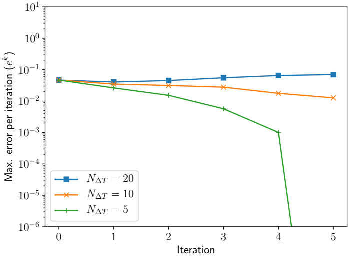




How to improve the parareal performance?

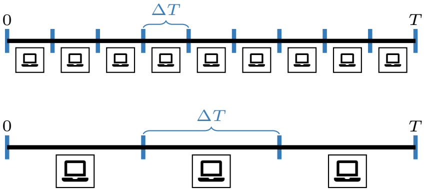
- Increase time slice length (Ruprecht, 2018)

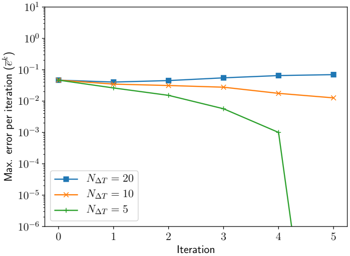




How to improve the parareal performance?

- Increase time slice length (Ruprecht, 2018)





How to improve the parareal performance?

- Increase time slice length (Ruprecht, 2018)
- Improve coarse model

How to improve the parareal performance?

- Increase time slice length (Ruprecht, 2018)
- Improve coarse model
- Increase spatial interpolation order (Ruprecht, 2014; Lunet, 2018)

How to improve the parareal performance?

- Increase time slice length (Ruprecht, 2018)
- Improve coarse model
- Increase spatial interpolation order (Ruprecht, 2014; Lunet, 2018)

Trade-off between convergence, stability and computational cost

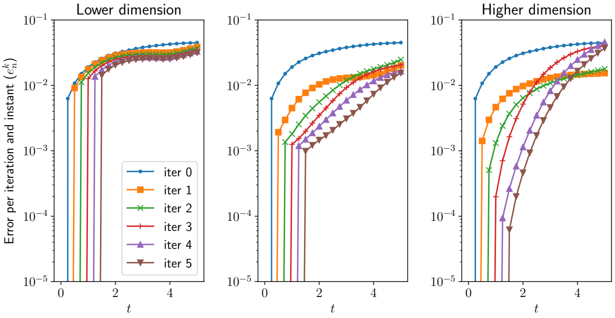
How to improve the ROM-based parareal performance?

How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension

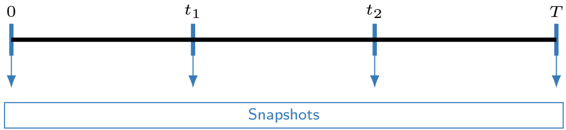
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension



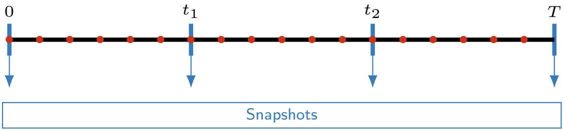
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction



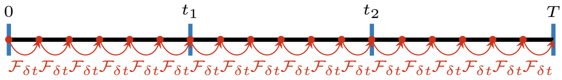
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction



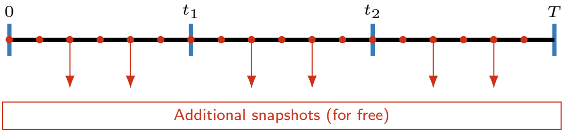
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction



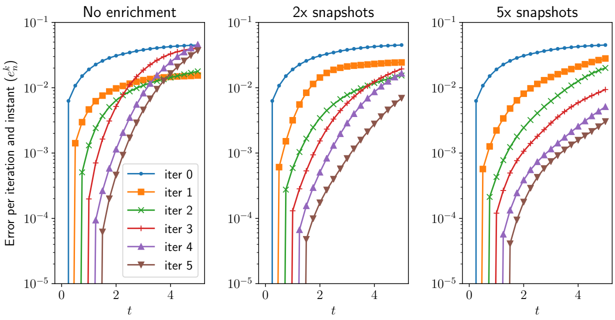
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction



How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction

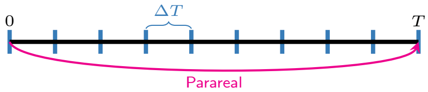


How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;

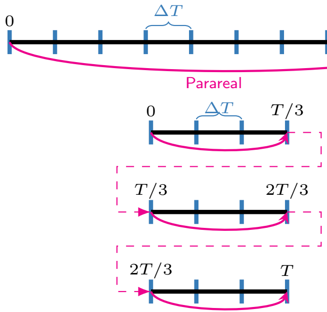
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;



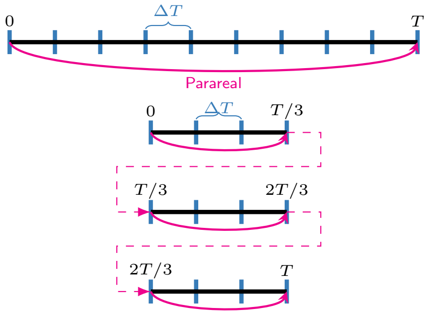
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;



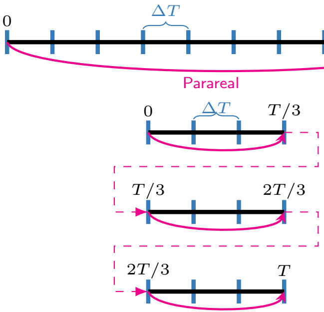
How to improve the ROM-based parareal performance?

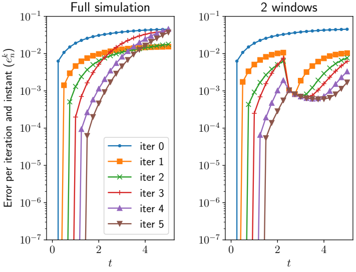
- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;



How to improve the ROM-based parareal performance?

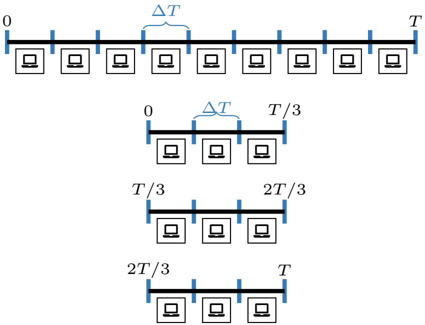
- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;

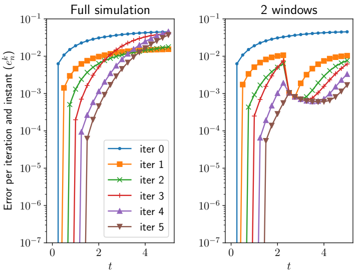




How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;



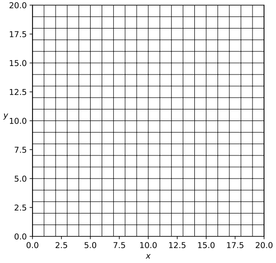


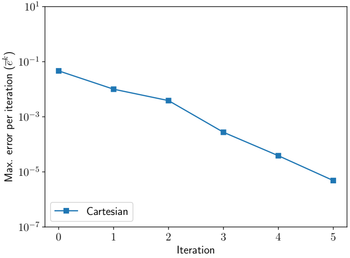
How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;

How to improve the ROM-based parareal performance?

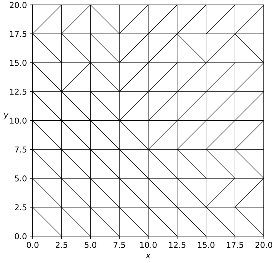
- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;

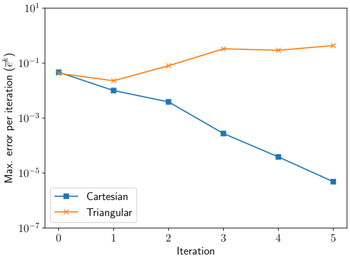




How to improve the ROM-based parareal performance?

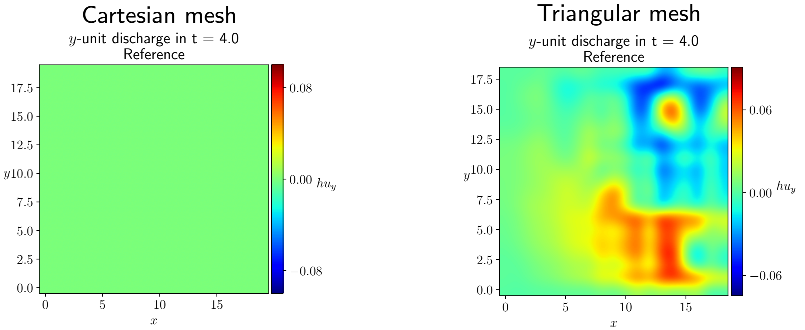
- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;





How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;



How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;

How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;

Trade-off between convergence, stability and computational cost

How to improve the ROM-based parareal performance?

- Properly choose the ROM dimension
- Enrich the model reduction
- Perform local-in-time parareal simulations;
- Properly choose the computational mesh;

Trade-off between convergence, stability and computational cost

ROM quality $\Longrightarrow$ performance of the ROM-based parareal method

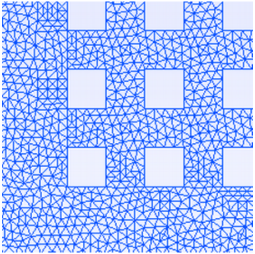
Table of contents

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods

Parareal for coupling the classical and porosity-based SWE

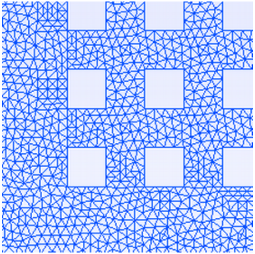
- Application to the simulation of urban floods
  - Fine (reference model): classical SWE



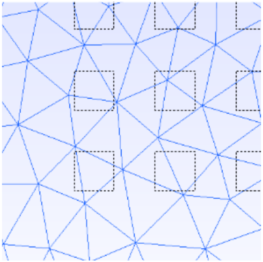
$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods
  - Fine (reference model): classical SWE
  - Coarse model: porosity-based SWE



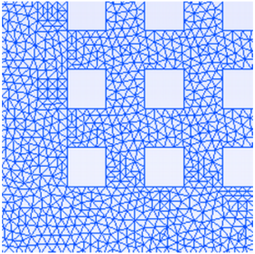
$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$



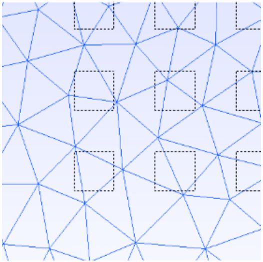
$$\frac{\partial}{\partial t}\phi\boldsymbol{U} + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}) = \overline{\boldsymbol{S}}(\boldsymbol{U})$$

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods
  - Fine (reference model): classical SWE
  - Coarse model: porosity-based SWE
- Challenges:



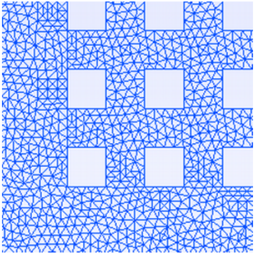
$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$



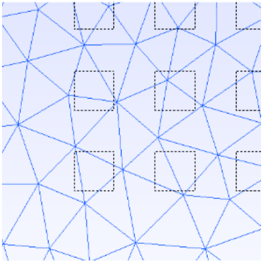
$$\frac{\partial}{\partial t}\phi\boldsymbol{U} + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}) = \overline{\boldsymbol{S}}(\boldsymbol{U})$$

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods
  - Fine (reference model): classical SWE
  - Coarse model: porosity-based SWE
- Challenges:
  - Large spatial and temporal domains



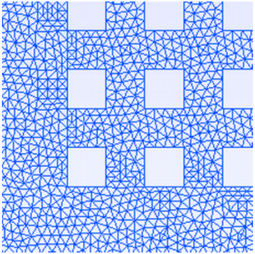
$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$



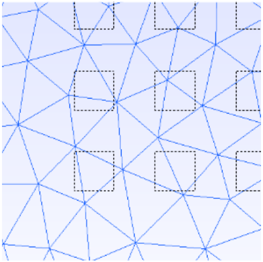
$$\frac{\partial}{\partial t}\phi\boldsymbol{U} + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}) = \overline{\boldsymbol{S}}(\boldsymbol{U})$$

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods
  - Fine (reference model): classical SWE
  - Coarse model: porosity-based SWE
- Challenges:
  - Large spatial and temporal domains
  - Spatial coarsening



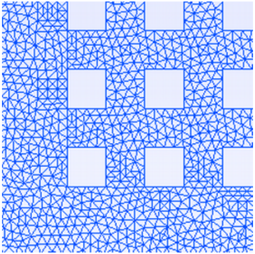
$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$



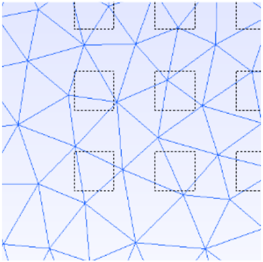
$$\frac{\partial}{\partial t}\phi\boldsymbol{U} + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}) = \overline{\boldsymbol{S}}(\boldsymbol{U})$$

Parareal for coupling the classical and porosity-based SWE

- Application to the simulation of urban floods
  - Fine (reference model): classical SWE
  - Coarse model: porosity-based SWE
- Challenges:
  - Large spatial and temporal domains
  - Spatial coarsening
  - Highly discontinuous solutions



$$\frac{\partial}{\partial t}\boldsymbol{U} + \frac{\partial}{\partial x}\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\boldsymbol{G}(\boldsymbol{U}) = \boldsymbol{S}(\boldsymbol{U})$$

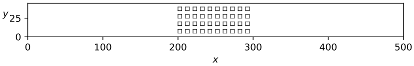


$$\frac{\partial}{\partial t}\phi\boldsymbol{U} + \frac{\partial}{\partial x}\phi\boldsymbol{F}(\boldsymbol{U}) + \frac{\partial}{\partial y}\phi\boldsymbol{G}(\boldsymbol{U}) = \overline{\boldsymbol{S}}(\boldsymbol{U})$$

A more realistic test case

A more realistic test case

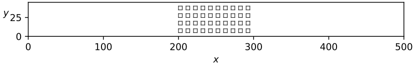
- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;



A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.

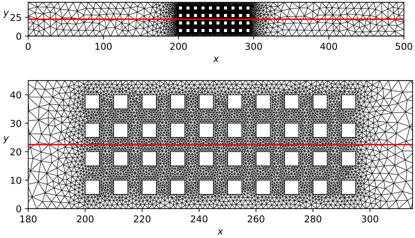
	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

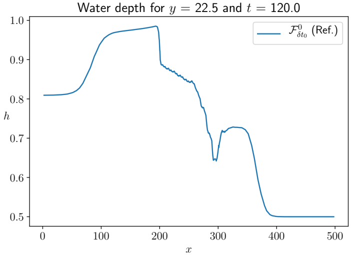


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

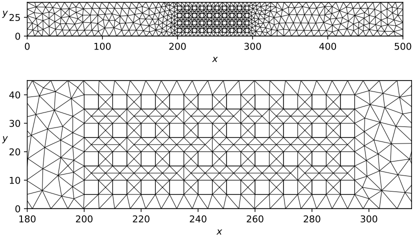


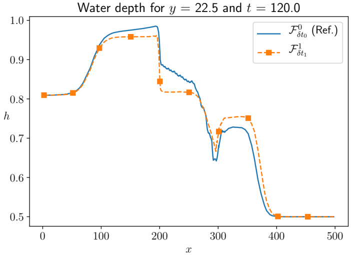


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

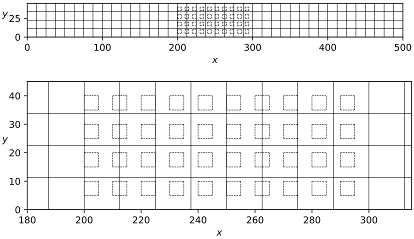


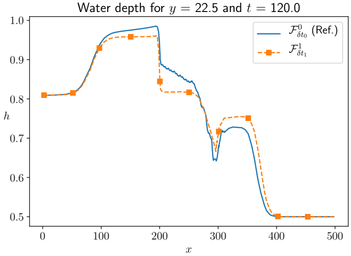


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
    - Various spatial and temporal mesh sizes; near CFL.

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

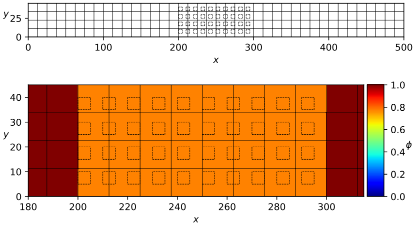


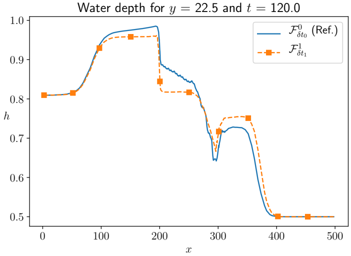


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

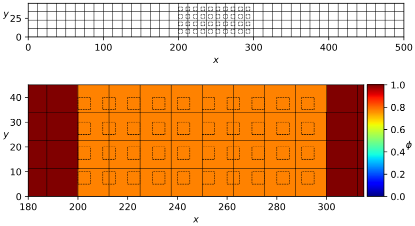


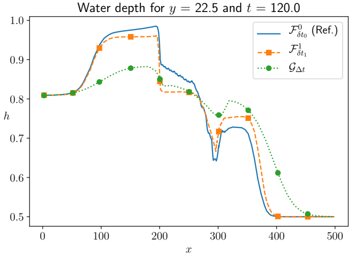


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5

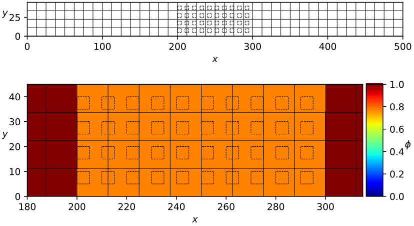


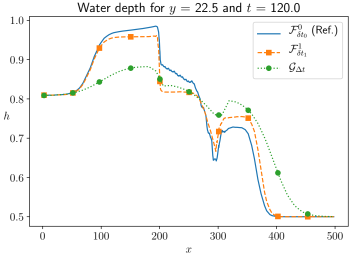


A more realistic test case

- Influence of coarsening between $\mathcal{F}_{\delta t}$ and $\mathcal{G}_{\Delta t}$ on the performance;
  - Various spatial and temporal mesh sizes; near CFL.
- Qualitative evaluation of convergence and speedup;

	Mesh size	Time step
$\mathcal{F}^0_{\delta t_0}$	1.0	0.05
$\mathcal{F}^1_{\delta t_1}$	2.5	0.25
$\mathcal{G}_{\Delta t}$	11.9	0.5





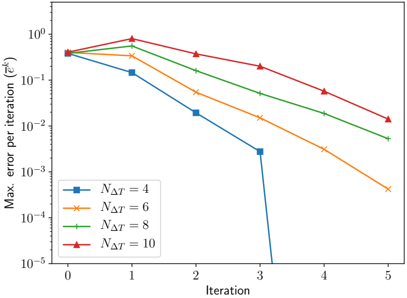
A more realistic test case

Using the classical parareal method

A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$

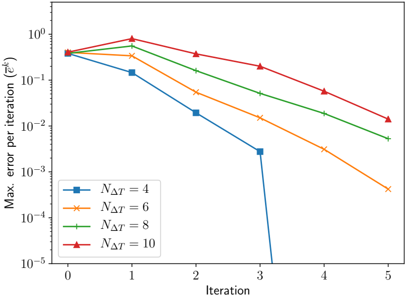


Coarse model: $\mathcal{F}^1_{\delta t_1}$

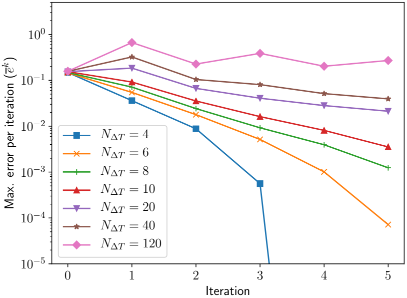
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$



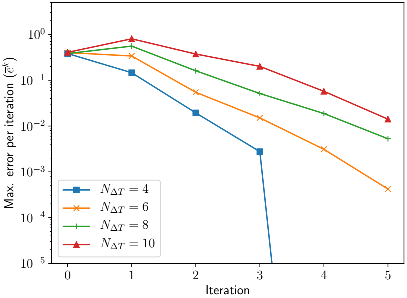
Coarse model: $\mathcal{F}^1_{\delta t_1}$



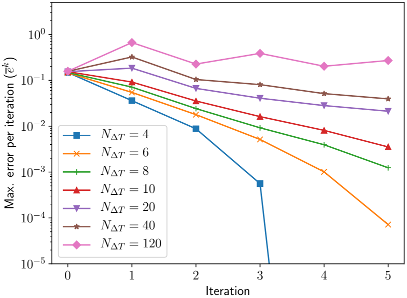
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



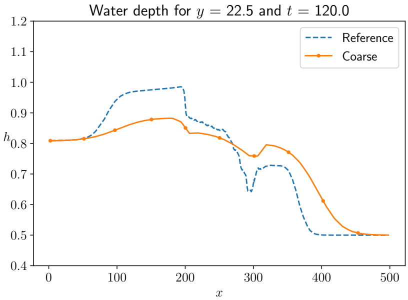
Coarse model: $\mathcal{F}^1_{\delta t_1}$



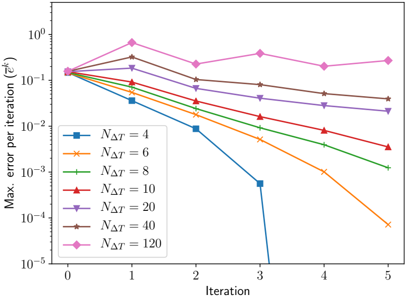
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



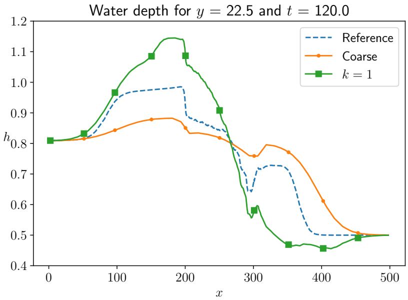
Coarse model: $\mathcal{F}^1_{\delta t_1}$



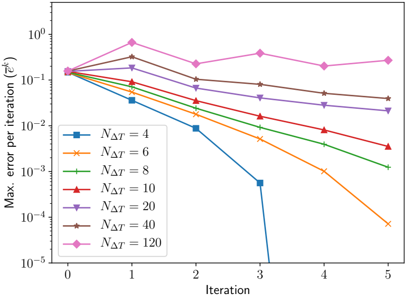
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



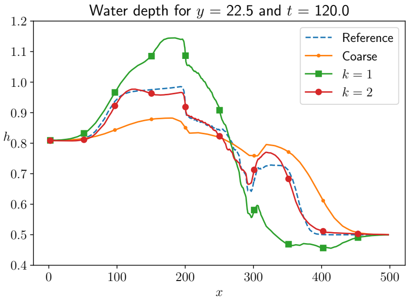
Coarse model: $\mathcal{F}^1_{\delta t_1}$



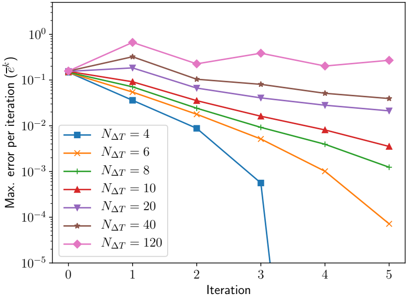
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



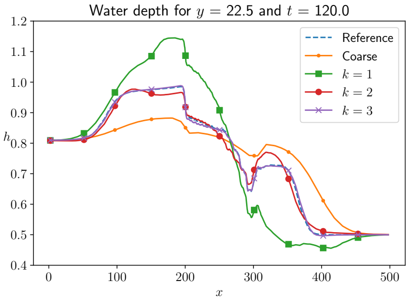
Coarse model: $\mathcal{F}^1_{\delta t_1}$



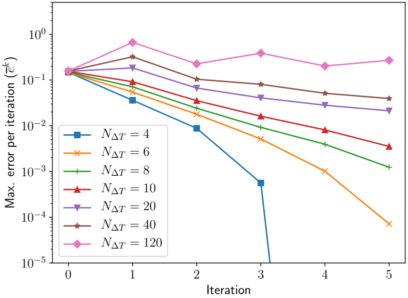
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



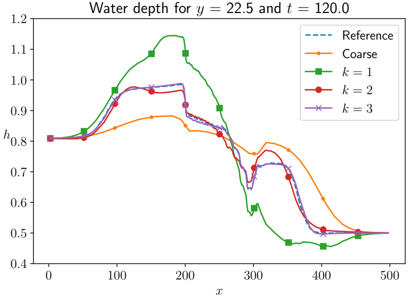
Coarse model: $\mathcal{F}^1_{\delta t_1}$



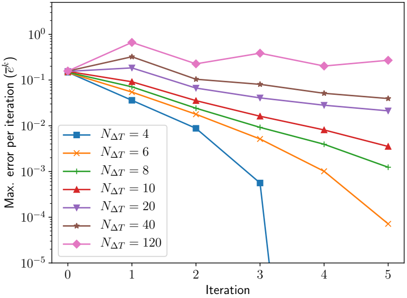
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



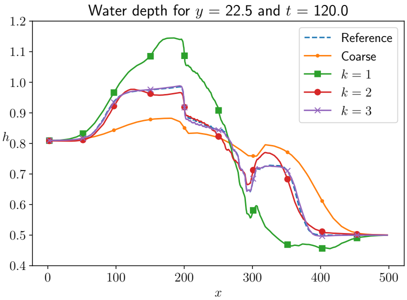
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



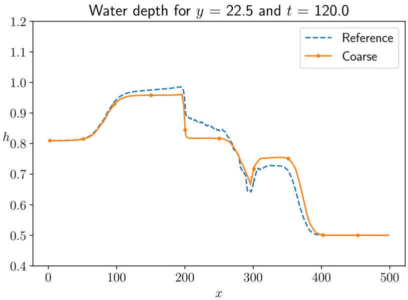
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



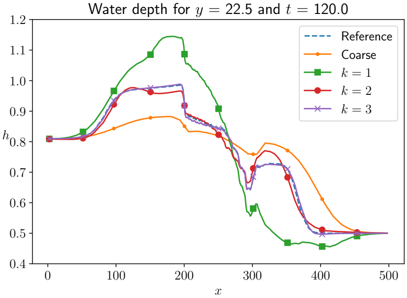
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



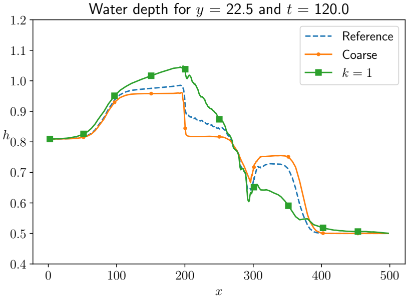
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



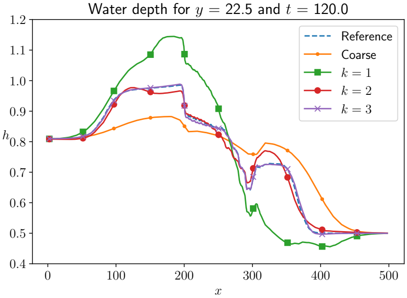
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



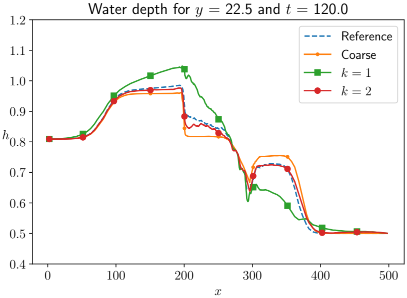
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



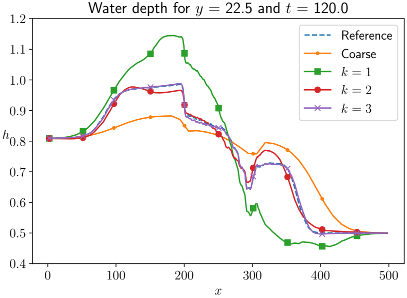
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



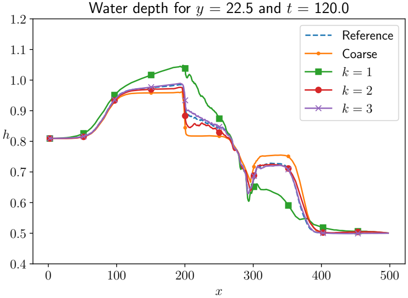
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



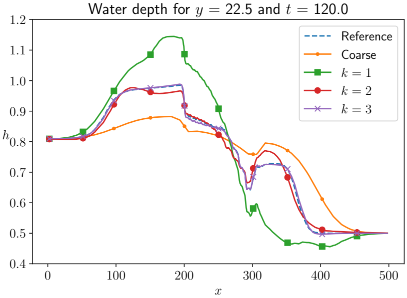
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



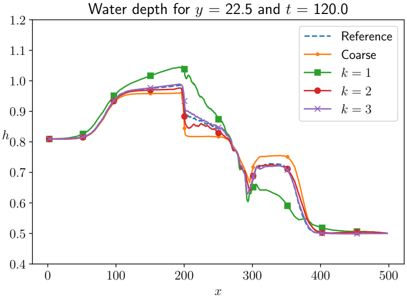
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



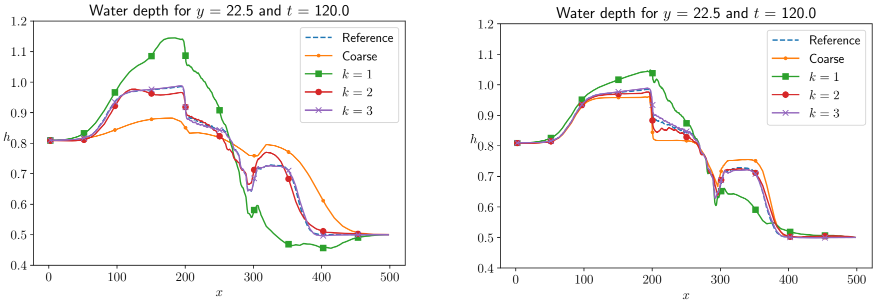
- Speedup after
  - 2 iterations: 2.40
  - 3 iterations: 1.66

A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$

Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



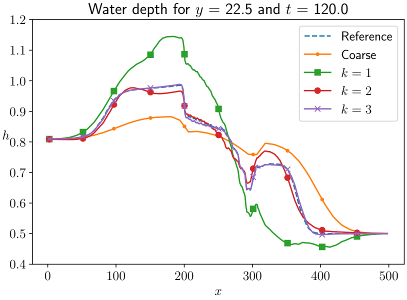
- Speedup after
  - 2 iterations: 2.40
  - 3 iterations: 1.66

- Speedup after
  - 2 iterations: 3.12
  - 3 iterations: 2.35

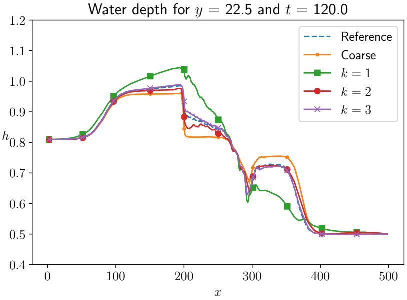
A more realistic test case

Using the classical parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 6$



Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 20$



- Speedup after
  - 2 iterations: 2.40
  - 3 iterations: 1.66

- Speedup after
  - 2 iterations: 3.12
  - 3 iterations: 2.35

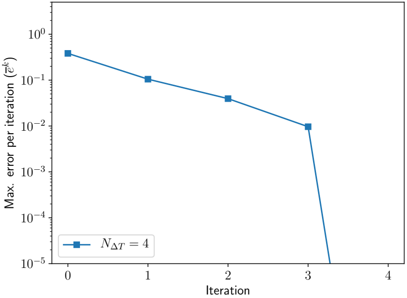
A more realistic test case

Using the ROM-based parareal method

A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$

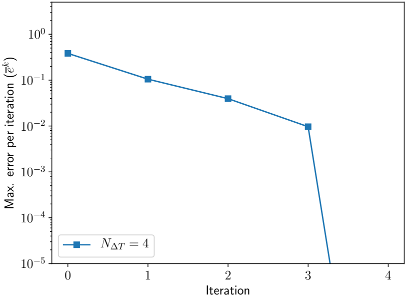


Coarse model: $\mathcal{F}^1_{\delta t_1}$

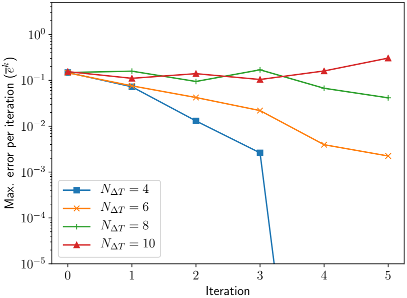
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$



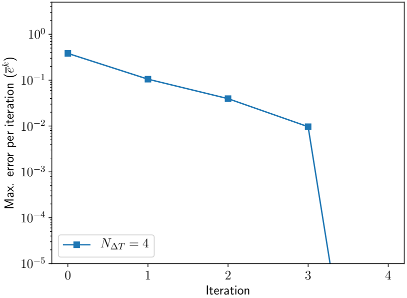
Coarse model: $\mathcal{F}^1_{\delta t_1}$



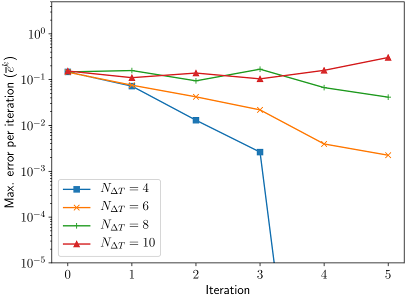
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



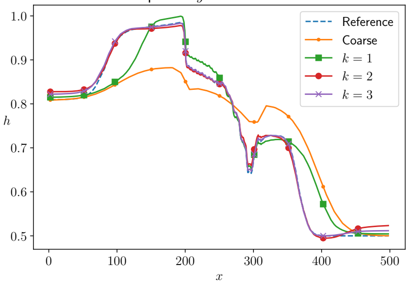
Coarse model: $\mathcal{F}^1_{\delta t_1}$



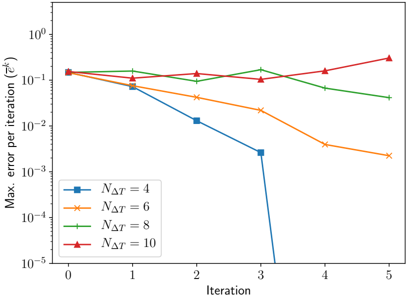
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



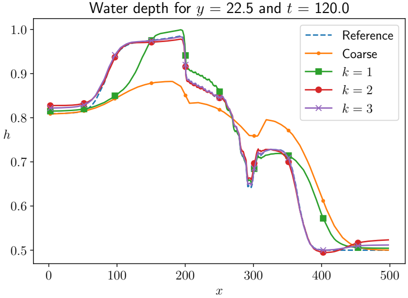
Coarse model: $\mathcal{F}^1_{\delta t_1}$



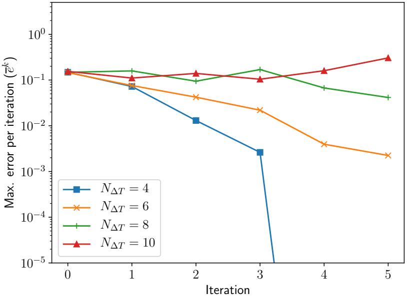
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



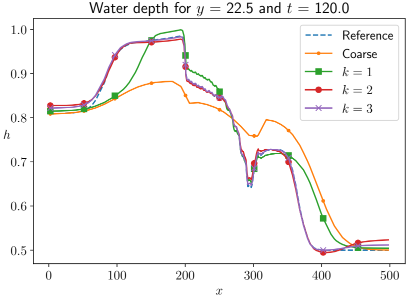
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 4$



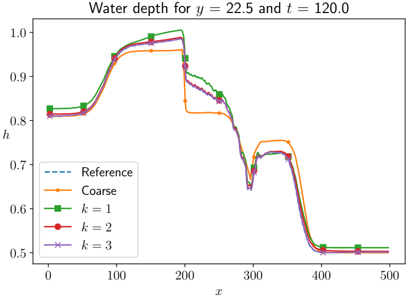
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



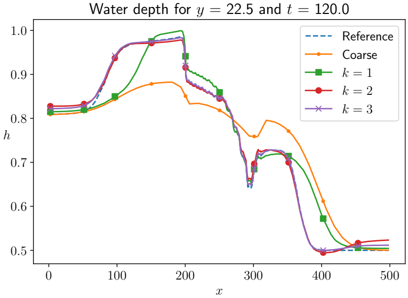
Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 4$



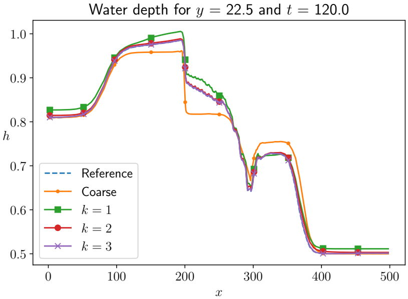
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 4$



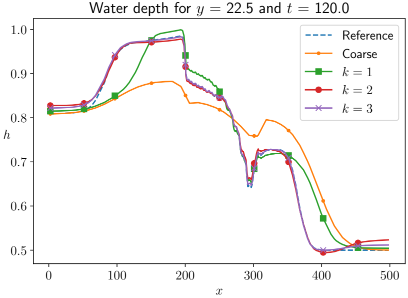
- Speedup after
  - 2 iterations: 1.35
  - 3 iterations: 0.92

- Speedup after
  - 2 iterations: 1.28
  - 3 iterations: 0.89

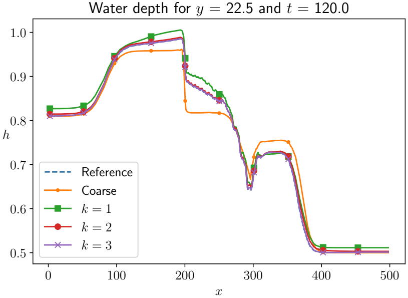
A more realistic test case

Using the ROM-based parareal method

Coarse model: $\mathcal{G}_{\Delta t}$ - $N_{\Delta T} = 4$



Coarse model: $\mathcal{F}^1_{\delta t_1}$ - $N_{\Delta T} = 4$



- Speedup after
  - 2 iterations: 1.35
  - 3 iterations: 0.92

- Speedup after
  - 2 iterations: 1.28
  - 3 iterations: 0.89

Table of contents

Conclusions and perspectives

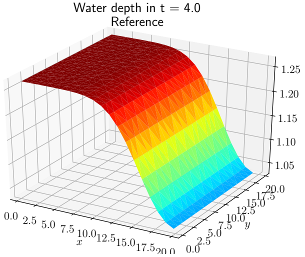
Conclusions

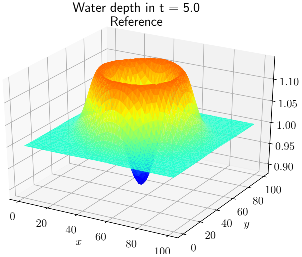
- Detailed investigation of the ROM-based parareal method

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;

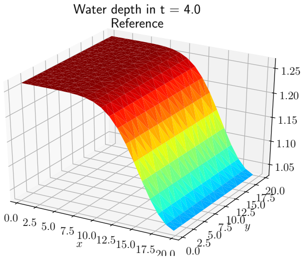


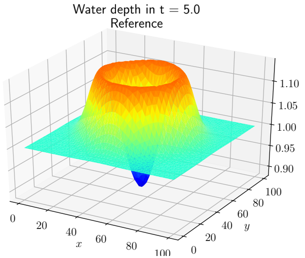


Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;

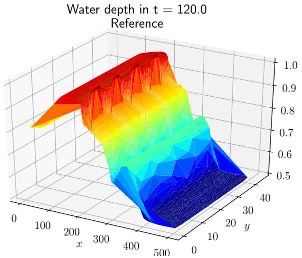


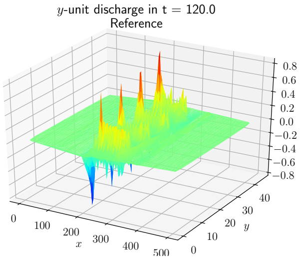


Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;

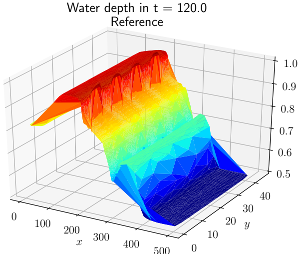


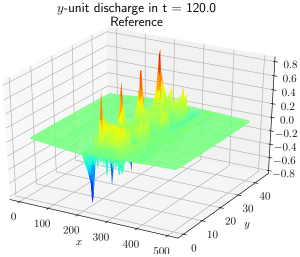


Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations

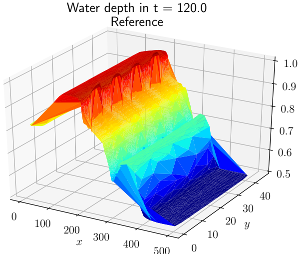


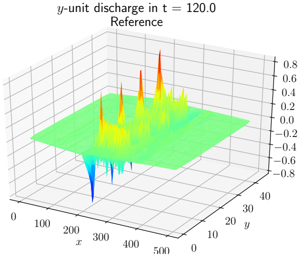


Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;

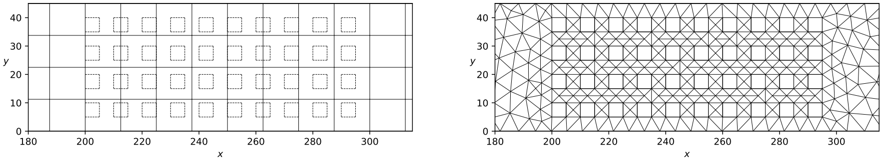




Conclusions and perspectives

Conclusions

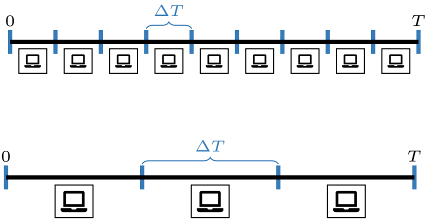
- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;



Conclusions and perspectives

Conclusions

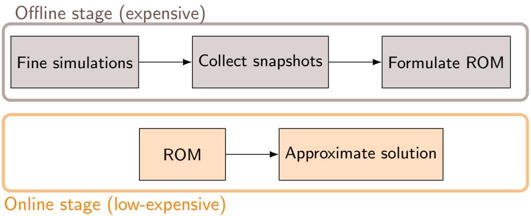
- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\implies$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?



Conclusions and perspectives

Conclusions

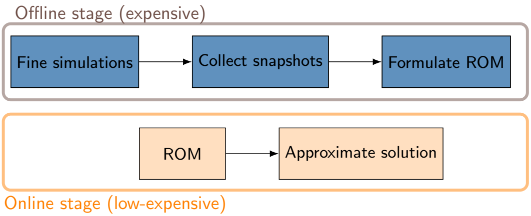
- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;



Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\implies$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;



Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\implies$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;
- More challenging problems;

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;
- More challenging problems;
- Improving the coarse model;

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\Longrightarrow$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;
- More challenging problems;
- Improving the coarse model;
- Dealing with zero water depth;

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\implies$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;
- More challenging problems;
- Improving the coarse model;
- Dealing with zero water depth;
- Other parallel-in-time methods;

Conclusions and perspectives

Conclusions

- Detailed investigation of the ROM-based parareal method
  - Initial promising results;
  - Improvements;
  - Limitations;
- Time parallelization of urban floods simulations
  - Trade-offs $\implies$ restrictive configurations;
  - Choice of coarse model;
  - Large-scale parallelism?
- Model reduction of the SWE;

Perspectives

- Implementation in SW2D;
- More challenging problems;
- Improving the coarse model;
- Dealing with zero water depth;
- Other parallel-in-time methods;
- Coupling with spatial parallelization.

Thank you for your attention

Committee's deliberation

(coming back soon)

Bibliography I

CRED, & UNISDR. (2015, November). The human cost of weather related disasters - 1995 - 2015. https://doi.org/10.13140/RG.2.2.17677.33769

United Nations. (2019). World urbanization prospects: The 2018 revision. United Nations, Department of Economic and Social Affairs, Population Division. https://www.un-ilibrary.org/content/books/9789210043144

Guinot, V., Sanders, B. F., & Schubert, J. E. (2017). Dual integral porosity shallow water model for urban flood modelling. Advances in Water Resources, 103, 16 –31. https://doi.org/https://doi.org/10.1016/j.advwatres.2017.02.009

Defina, A., D'Alpaos, L, & Matticchio, B. (1994). New set of equations for very shallow water and partially dry areas suitable to 2D numerical models. Proceedings of the Specialty Conference on Modelling of Flood Propagation Over Initially Dry Areas, 72–81.

Guinot, V., & Soares-Frazão, S. (2006). Flux and source term discretization in two-dimensional shallow water models with porosity on unstructured grids. International Journal for Numerical Methods in Fluids, 50(3), 309–345. https://doi.org/10.1002/fld.1059

Sanders, B. F., Schubert, J. E., & Gallegos, H. A. (2008). Integral formulation of shallow-water equations with anisotropic porosity for urban flood modeling. Journal of Hydrology, 362(1), 19 –38. https://doi.org/https://doi.org/10.1016/j.jhydrol.2008.08.009

Guinot, V. (2012). Multiple porosity shallow water models for macroscopic modelling of urban floods. Advances in Water Resources, 37, 40 –72. https://doi.org/https://doi.org/10.1016/j.advwatres.2011.11.002

Bibliography II

Velickovic, M., Zech, Y., & Soares-Frazão, S. (2017). Steady-flow experiments in urban areas and anisotropic porosity model. Journal of Hydraulic Research, 55(1), 85–100. https://doi.org/10.1080/00221686.2016.1238013

Lions, J.-L., Maday, Y., & Turinici, G. (2001). Résolution d'edp par un schéma en temps 'pararéel'. Comptes Rendus de l'Académie des Sciences - Series I - Mathematics, 332(7), 661 –668. https://doi.org/https://doi.org/10.1016/S0764-4442(00)01793-6

Ruprecht, D. (2018). Wave propagation characteristics of Parareal. Computing and Visualization in Science, 19, 1–17. https://doi.org/10.1007/s00791-018-0296-z

Chen, F., Hesthaven, J. S., & Zhu, X. (2014). On the use of reduced basis methods to accelerate and stabilize the parareal method. In A. Quarteroni & G. Rozza (Eds.), Reduced order methods for modeling and computational reduction (pp. 187–214). Springer International Publishing. https://doi.org/10.1007/978-3-319-02090-7_7

Barrault, M., Maday, Y., Nguyen, N. C., & Patera, A. T. (2004). An 'empirical interpolation' method: Application to efficient reduced-basis discretization of partial differential equations. Comptes Rendus Mathematique, 339(9), 667 –672. https://doi.org/https://doi.org/10.1016/j.crma.2004.08.006

Chaturantabut, S., & Sorensen, D. C. (2010). Nonlinear model reduction via discrete empirical interpolation. SIAM Journal on Scientific Computing, 32(5), 2737–2764. https://doi.org/10.1137/090766498

Ruprecht, D. (2014). Convergence of parareal with spatial coarsening. PAMM, 14(1), 1031–1034. https://doi.org/https://doi.org/10.1002/pamm.201410490

Bibliography III

Lunet, T. (2018). Stratégies de parallélisation espace-temps pour la simulation numérique des écoulements turbulents (Doctoral dissertation) [Thèse de doctorat dirigée par Gratton, Serge et Bodart, Julien Mathématiques appliquées Toulouse, ISAE 2018]. http://www.theses.fr/2018ESAE0001