EuroHPC project

Time parallelization for eXascale computing

Giovanni Samaey

Katholieke Universiteit Leuven

Martin Schreiber

Tech. Univ. Munich / Univ. Grenoble Alpes

on behalf of the whole Time-X project

2022-03-28, MAELSTROM meeting

Outline

- **What is parallel-in-time?**
- Time-X project: strategy & research goals
- Applications: focus on weather in this talk
- Discussion

Traditional vs. parallel-in-time

Traditional way

- Spatial decomposition
- & Parallelization in space

- Time dimension is treated sequentially
- Beyond spatial scalability limit?

Parallel-in-time approach

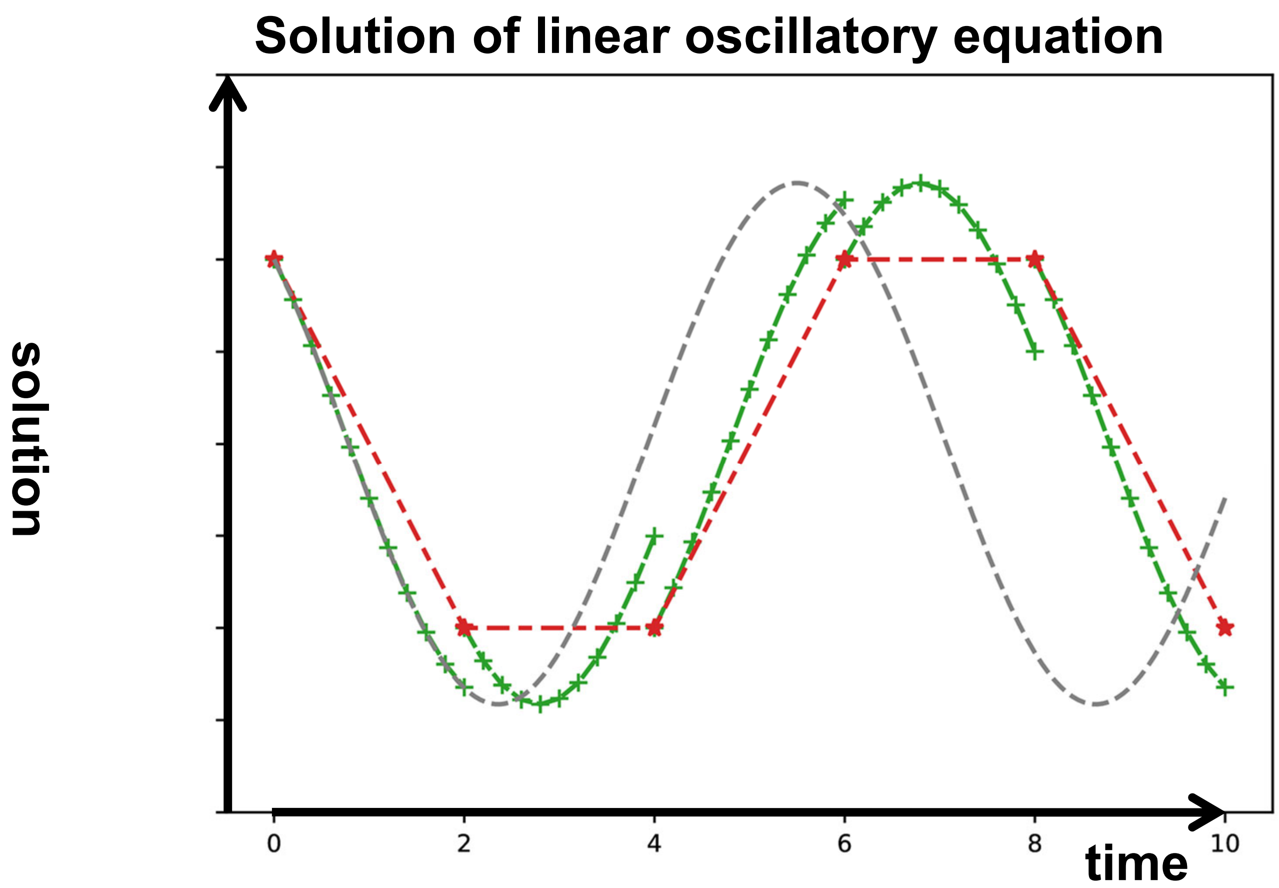
- (In addition), parallelize across the time dimension
- Requires development of new numerical algorithms

Example: Parareal

Oscillatory examples

- Lorenz attractor:

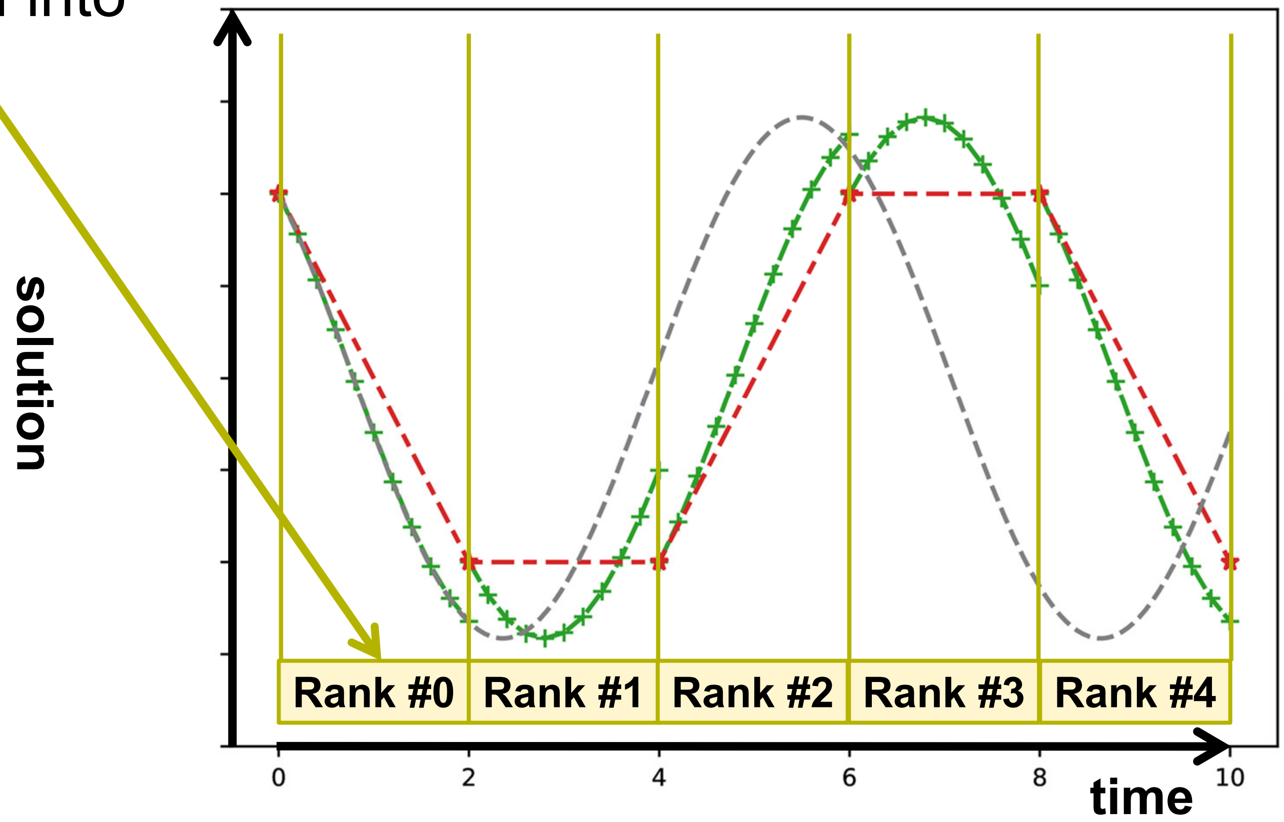
- Right hand side:
Linear oscillatory equation



Example: Parareal

1) **Time parallelization**

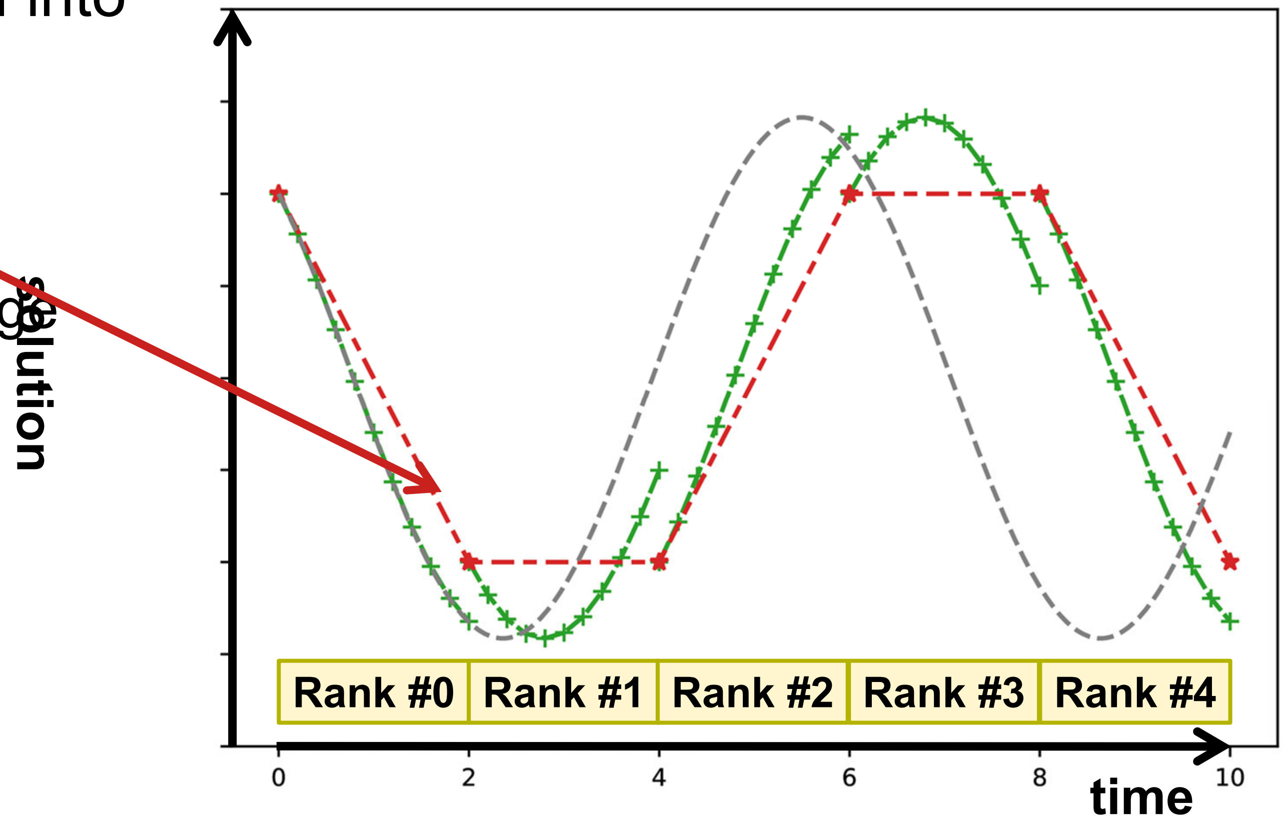
Time dimension is discretized into **coarse time steps**



Example: Parareal

1) Time parallelization
Time dimension is discretized into **coarse time steps**

2) Coarse time integrator
Very cheap one, but allowing large time steps

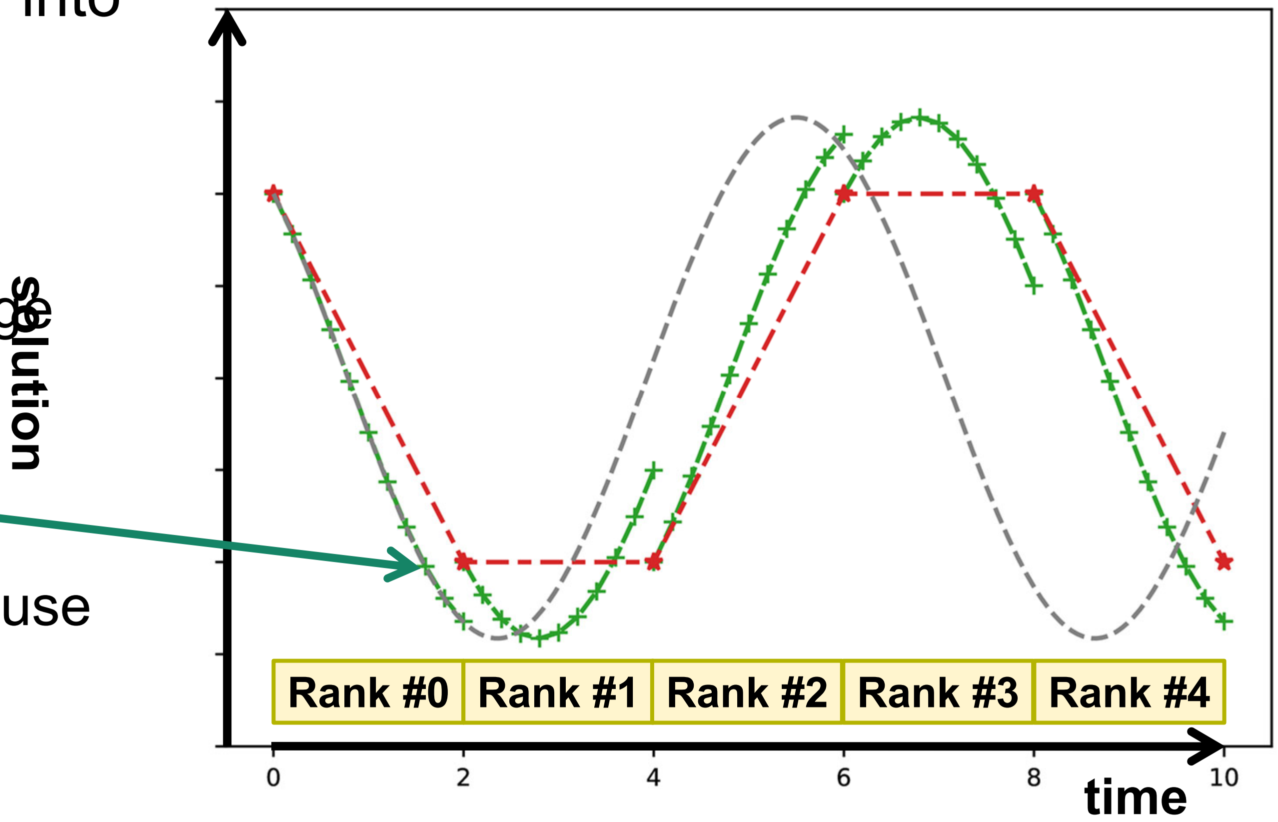


Example: Parareal

1) Time parallelization
Time dimension is discretized into **coarse time steps**

2) Coarse time integrator
Very cheap one, but allowing large time steps

3) Fine time integrator
Regular one you would typically use



4) Iterative corrections
Coarse and fine integrations

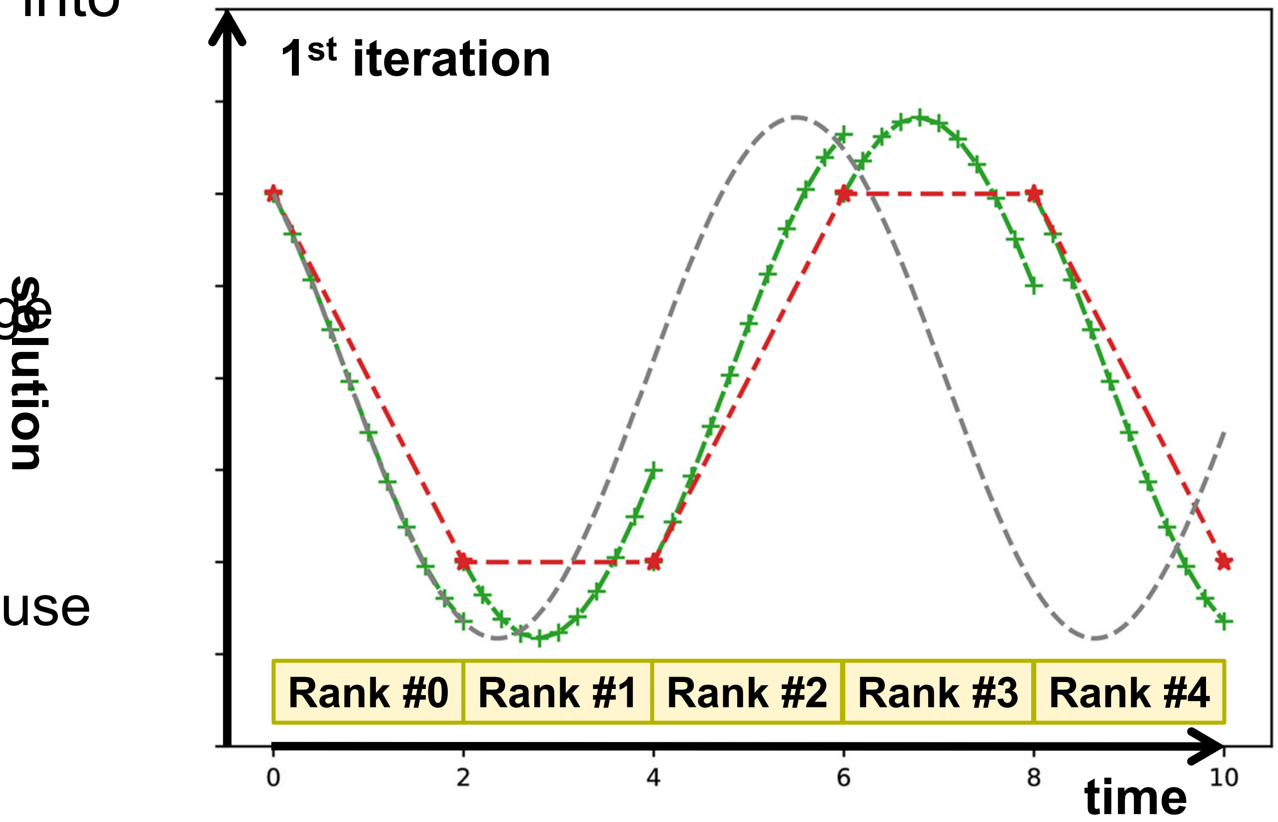
Example: Parareal

1) Time parallelization
Time dimension is discretized into **coarse time steps**

2) Coarse time integrator
Very cheap one, but allowing large time steps

3) Fine time integrator
Regular one you would typically use

4) Iterative corrections
Coarse and fine integrations



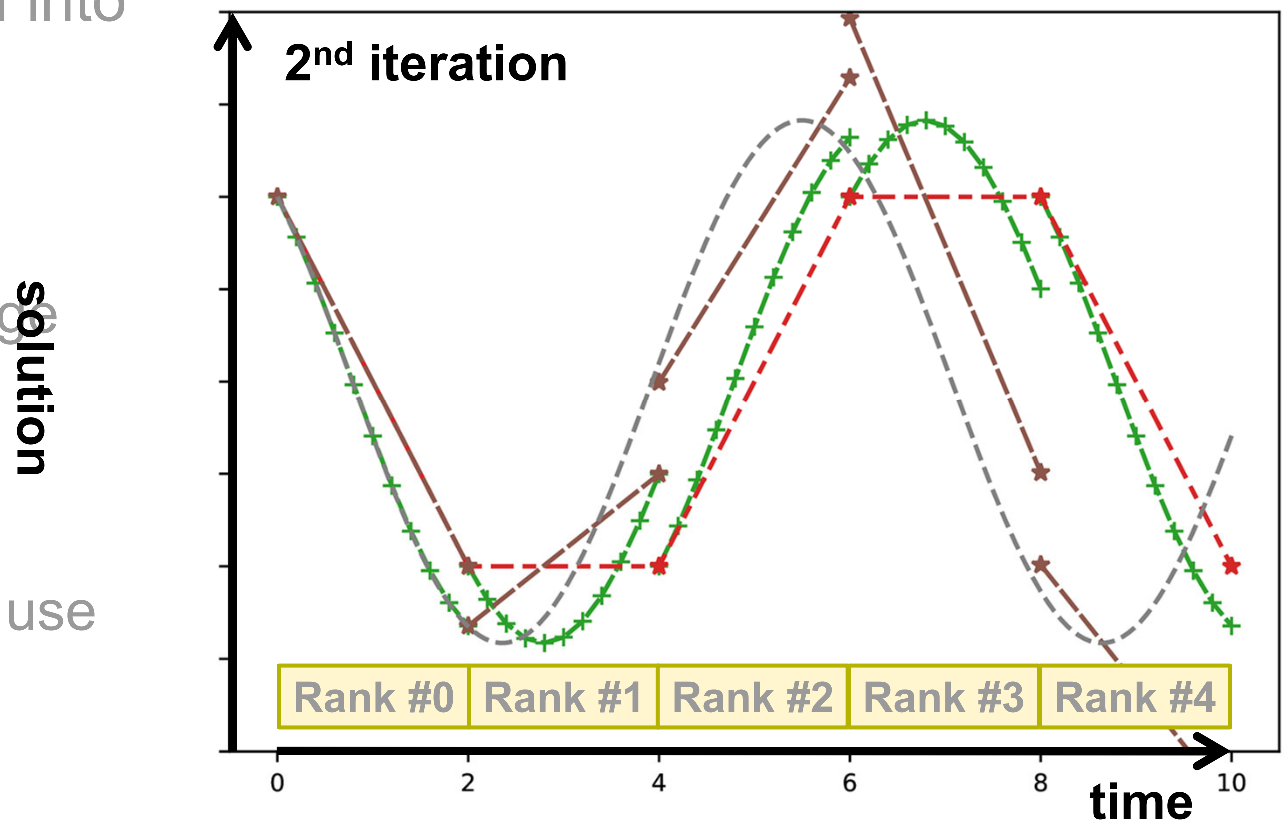
Example: Parareal

1) Time parallelization
Time dimension is discretized into **coarse time steps**

2) Coarse time integrator
Very cheap one, but allowing large time steps

3) Fine time integrator
Regular one you would typically use

4) Iterative corrections
Coarse and fine integrations



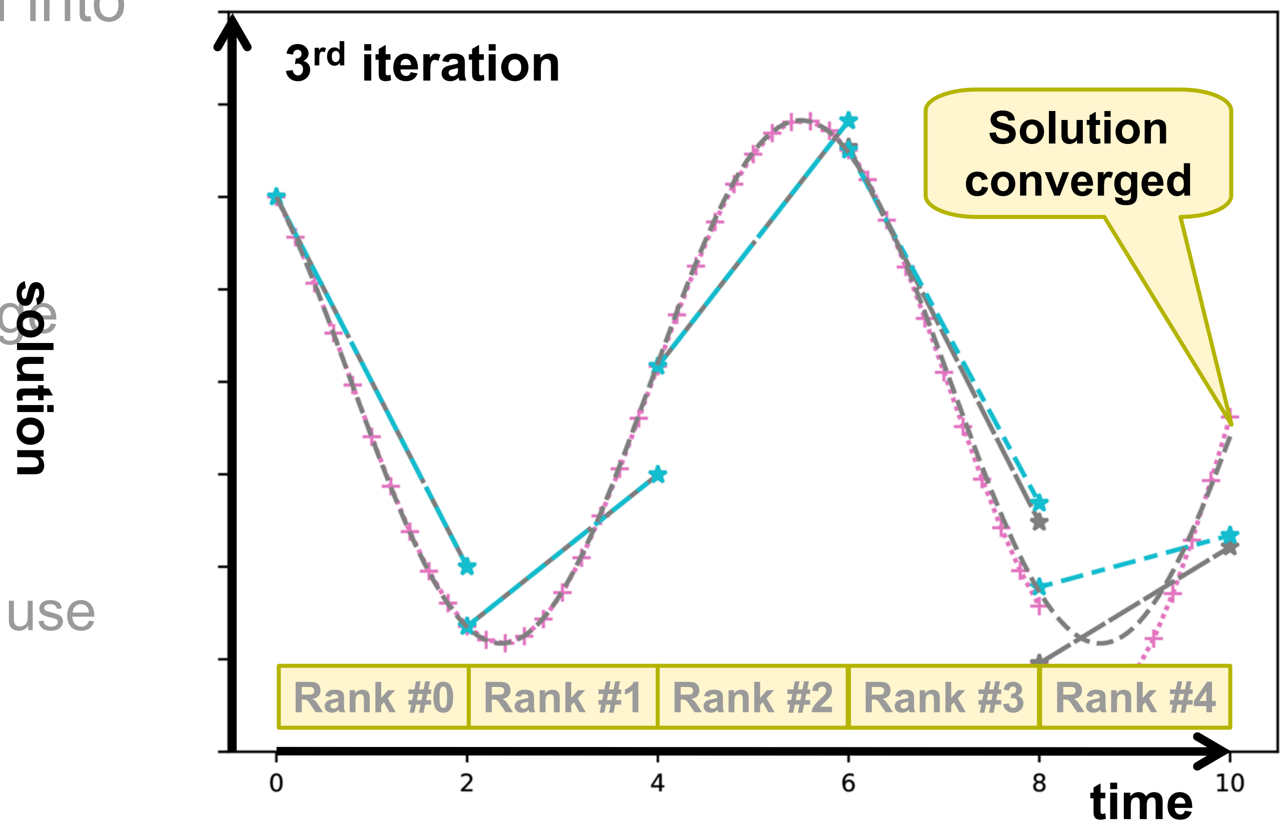
Example: Parareal

1) Time parallelization
Time dimension is discretized into **coarse time steps**

2) Coarse time integrator
Very cheap one, but allowing large time steps

3) Fine time integrator
Regular one you would typically use

4) Iterative corrections
Coarse and fine integrations



Outline

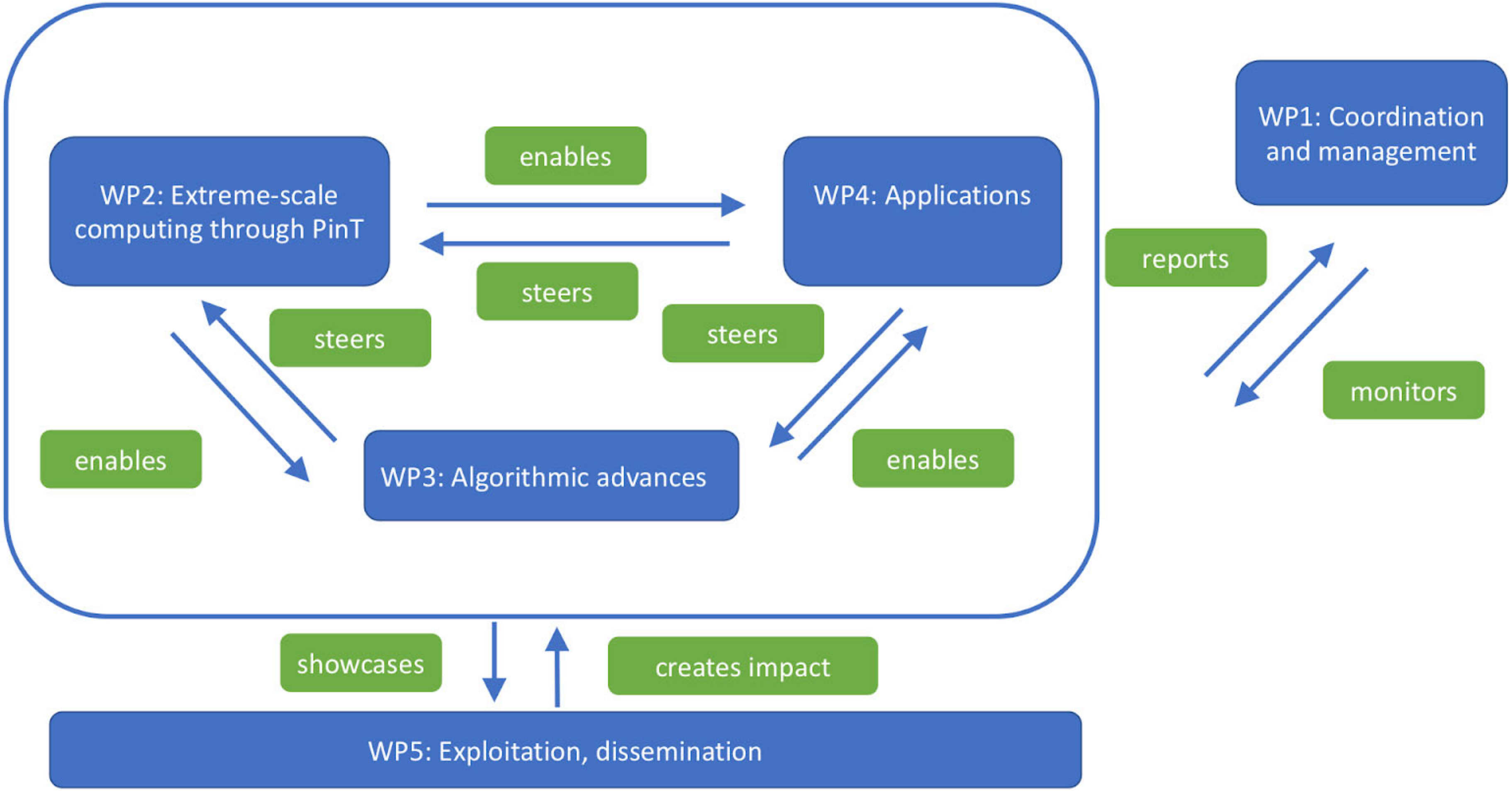
- What is parallel-in-time?
- **Time-X project: strategy & research goals**
- Applications: focus on weather in this talk
- Discussion

Objectives and setup

“**advancing parallel-in-time integration from an academic methodology into a widely available technology**, delivering Exascale performance for a wide range of scientific and industrial applications”

- HPC software development
  - Load balancing
  - Adaptivity
  - Inexactness and robustness (communication)
- Algorithm development
  - Optimization and optimal control
  - Uncertainty quantification and data assimilation
  - Multiscale propagators

Time-X: Structure



Outline

- What is parallel-in-time?
- Time-X project: strategy & research goals
- **Applications: focus on weather in this talk**
- Discussion

Application: Weather/climate simulations

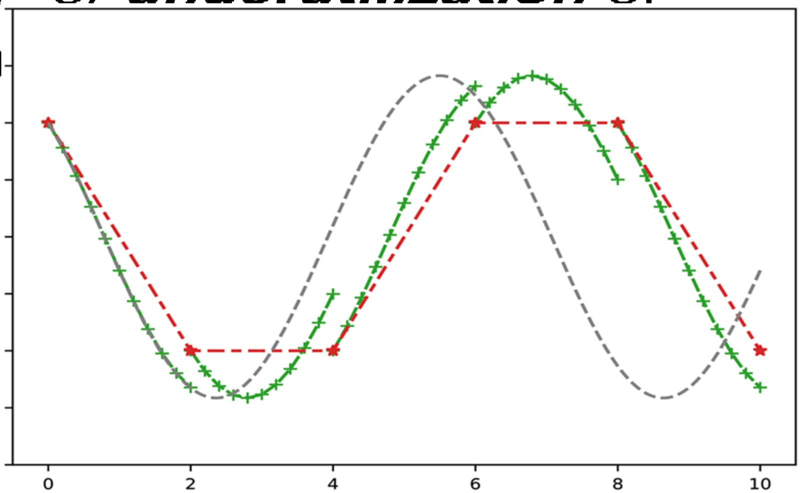
- **Weather simulations**
  - Target: Higher resolution for higher accuracy
  - Higher resolution => more time steps (CFL) + no further increase in per-core performance => longer **simulation runtime**
- **Time-X:**
  - **Single-layer atmosphere simulation** (nonlinear shallow-water equations)

How to improve resource efficiency?

Adopting computing resources

Motivated by two different perspectives

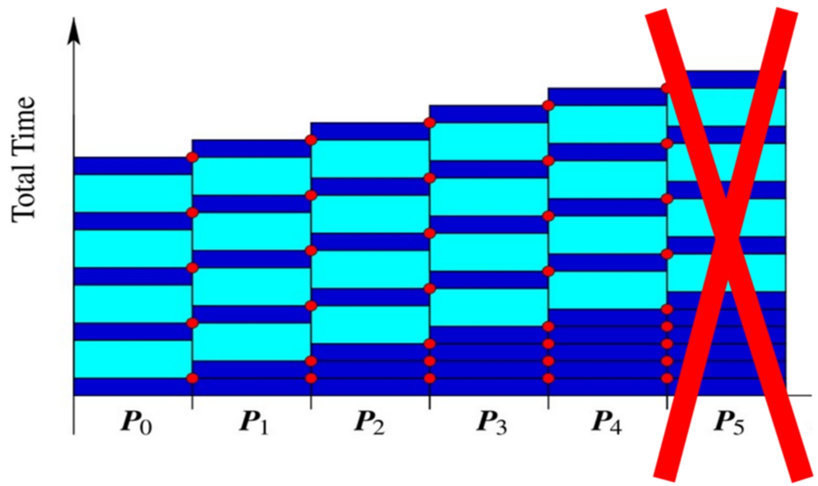
- **Parallel-in-time applications:**
  - **Convergence** often **unknown**
  - Might **change** over runtime
  - **Over-** or **underutilization** of comp

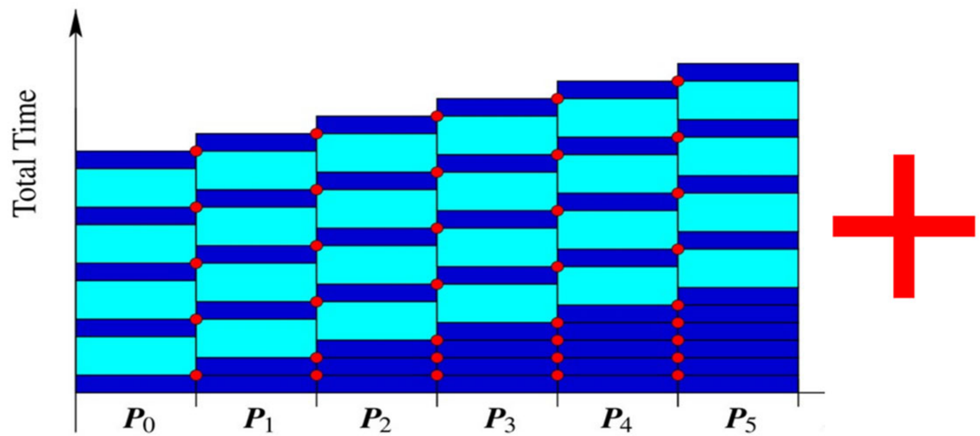


- **Super computing center**:
  - Parallel-in-time algorithms require **significantly more computing resources**
  - => **Allocation** of a large set from the beg **ng**

EuroHPC Time-X: Weather and climate

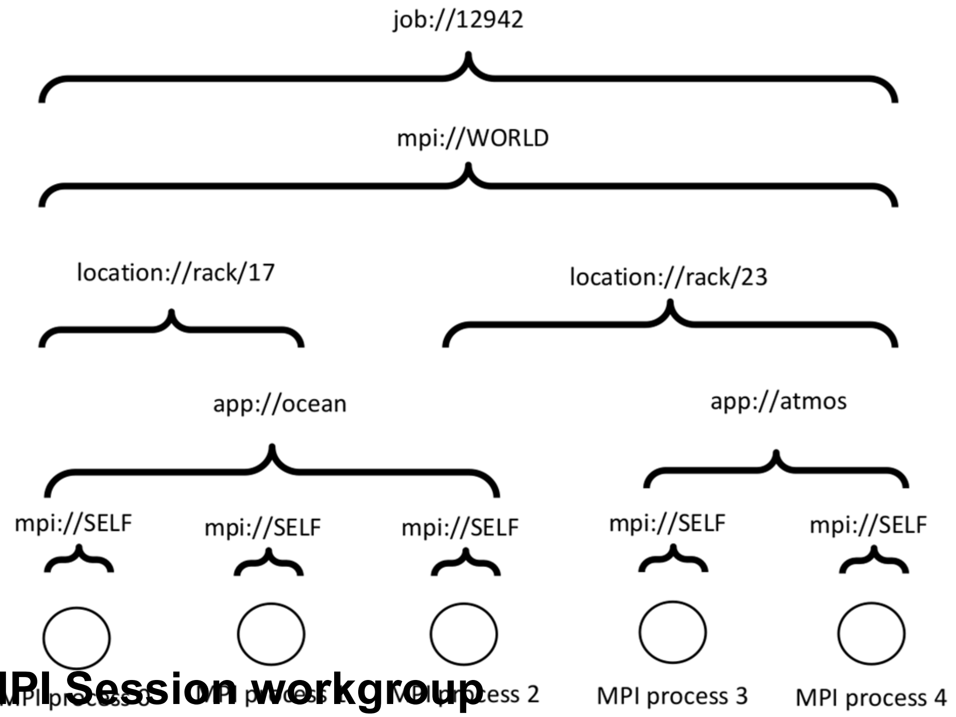
- New strategies in **MPI standard to support varying resources** for parallel-in-time applications
- Based on **application** <=> MPI standard **co-design**
- Support removing time-parallel instances (left) or adding them (right):





MPI Sessions

- New feature to overcome MPI_COMM_WORLD
- Faster communication setup
- Would (in theory) allows adding new resources
- Does not yet support full dynamicity



ion / ideas have been developed in collaboration with the **MPI Session workgroup**

Step-by-step approach

- Step 1) Dynamic MPI **simulation** layer
- Step 2) Extend **applications** with dynamic MPI support

} Part of Time-X proposal

- Step 3) Realize **dynamic MPI sessions** in **MPI implementation**
- Step 4) ...
- Step …) Scheduler

Dynamicity

- But… how to reschedule resources?
- (And what exactly are resources?)
- Which information is it based on?
- …

- **Application or system will provide** (abstract) **information** on **how it will perform (throughput, efficiency, time-to-solution) with resource changes**

=> Some new research for scheduling on the horizon?

Outline

- What is parallel-in-time?
- Time-X project: strategy & research goals
- Applications: focus on weather in this talk
- **Discussion**

Any questions?

Giovanni Samaey <giovanni.samaey@kuleuven.be>
Martin Schreiber <martin.schreiber@univ-grenoble-alpes.fr>