From SDC to PFASST

June 18, 2013 | Robert Speck[1,2], Daniel Ruprecht[2], Matthew Emmett[3]

[1]Jülich Supercomputing Centre (Forschungszentrum Jülich, Germany)
[2]Institute of Computational Science (USI Lugano, Switzerland)
[3]Center for Computational Sciences and Engineering (LBNL, USA)

JÜLICH FORSCHUNGSZENTRUM

Deriving the PFASST algorithm

Evolutionary:

1. Spectral Deferred Corrections (Dutt, Greengard, Rokhlin; 2000)
2. Parareal (Lions, Maday, Turinici; 2001)
3. hybrid Parareal/SDC (Minion and Williams; 2008)
4. hybrid Parareal/SDC + FAS (Emmett and Minion; 2012)

Here: 3 hops via multilevel SDC…

Spectral Deferred Corrections

Dutt, Greengard, Rokhlin (2000)

Consider $y' = f(y,t)$ with $y(t_0) = y_0$ or equivalently

$$y = y_0 + \int_{t_0}^{t} f(y(s))\mathrm{d}s.$$

SDC iteration sweeps over spectral collocation nodes t_m with simple time-steppers, e.g. backward Euler gives

$$y_{m+1}^{k+1} = y_m^{k+1} + \Delta t_m \left[f(y_{m+1}^{k+1}) - f(y_{m+1}^{k}) \right] + S_m^{m+1} f(y^k)$$

with Gaussian quadrature

$$S_m^{m+1} f(y^k) \approx \int_{t_m}^{t_{m+1}} f(y^k(s))\mathrm{d}s.$$

From SDC to MLSDC

SDC solves implicit collocation formula

$$Y = Y_0 + \Delta t Q F(Y)$$

Application of nonlinear multigrid theory gives natural multilevel structure, where...

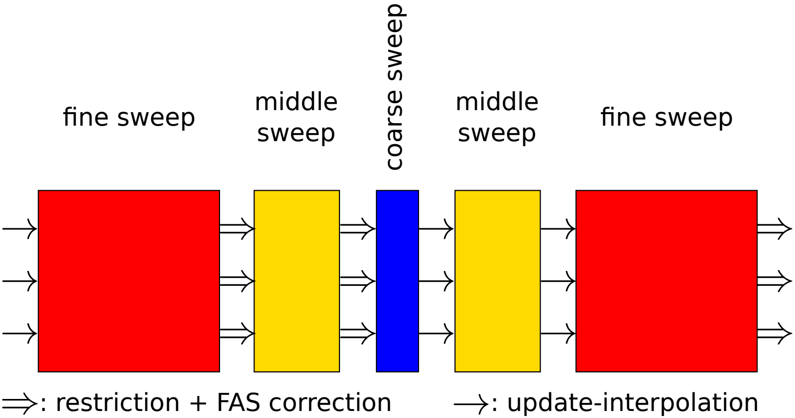
- SDC sweeps can be interpreted as relaxation
- information from level ℓ to $\ell + 1$ is transferred via FAS correction

$$\tau^{\ell+1} = \Delta t \left(R_\ell^{\ell+1} Q^\ell F^\ell(Y^\ell) - Q^{\ell+1} F^{\ell+1}(Y^{\ell+1}) \right)$$

- coarsening actually happens in space and time

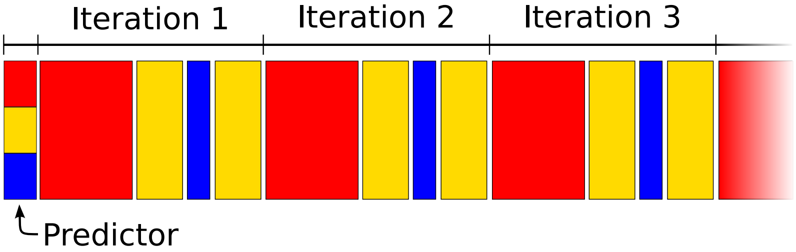
MLSDC: iteration

Three levels, V-cycle scheme



MLSDC: time-step (simplified)

Three levels, V-cycle scheme



Coarsening strategies in space and time

Speck, Ruprecht et al. (in prep.)

Expected benefit from MLSDC (in its own right):

shift work from fine/expensive to coarse/cheap levels

How to make coarse sweeps "cheap"?

1. reduction of temporal SDC nodes
2. reduction of degrees-of-freedom in space
3. reduced order in spatial discretization
4. reduced implicit solve (if implicit integrator used)
5. reduced physical representation

Case study: shear layer instability

We consider the 2D vorticity-velocity equation

$$\omega_t + u \cdot \nabla\omega = \nu\Delta\omega$$

with periodic domain $[0,1]^2$, "slightly disturbed initial conditions" and

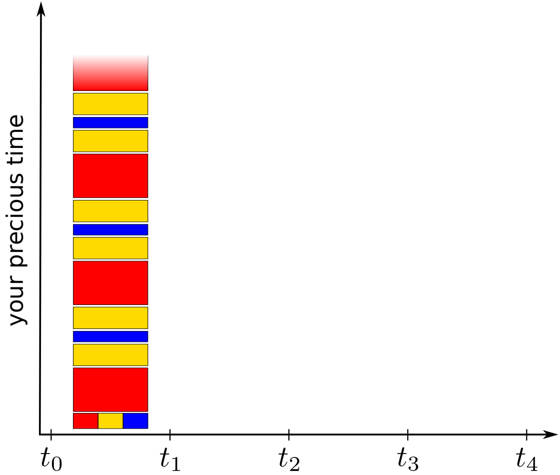
- implicit diffusion (using a linear MG),
- explicit advection (using a streamfunction-formulation and MG for the Poisson problem).

3-level coarsening strategy:

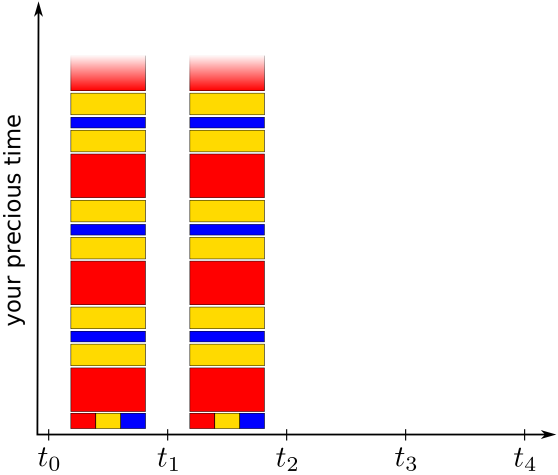
1. fine: 4th-order discretization, 128×128 dofs, 9 SDC nodes
2. middle: 2nd-order discretization, 64×64 dofs, 5 SDC nodes
3. coarse: 2nd-order discretization, 32×32 dofs, 3 SDC nodes

Case study: shear layer instability

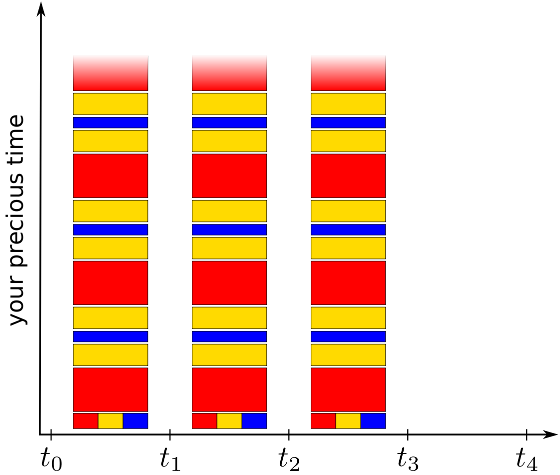
What if. . . ?



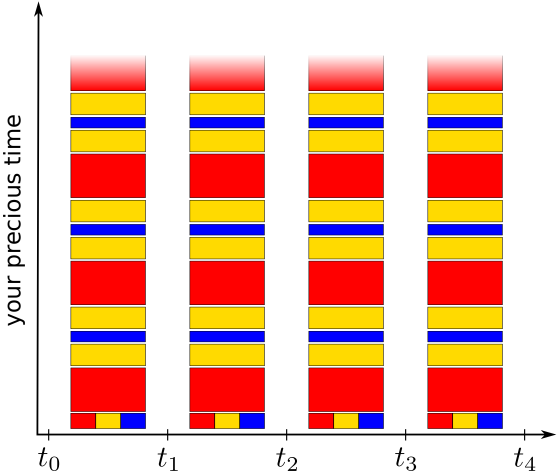
What if. . . ?



What if. . . ?



What if. . . ?



Parallel full approximation scheme in space and time

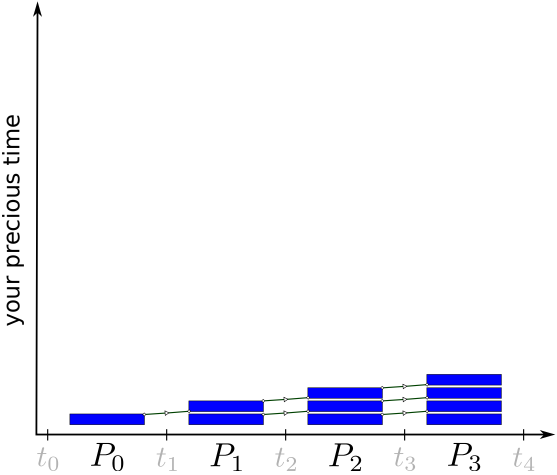
Emmett, Minion (2012)

Challenges of parallel MLSDC & **S**olutions with PFASST

- **C**: provide initial values on all processes
  S: do cascade of serial coarse sweeps on startup
- **C**: update information during iteration
  S: forward updated values on all levels during interpolation
- **C**: avoid idle times during information propagation
  S: blocking communication only on coarsest level
- **C**: augment given spatial parallelization
  S: use intra- or inter-node communication in time

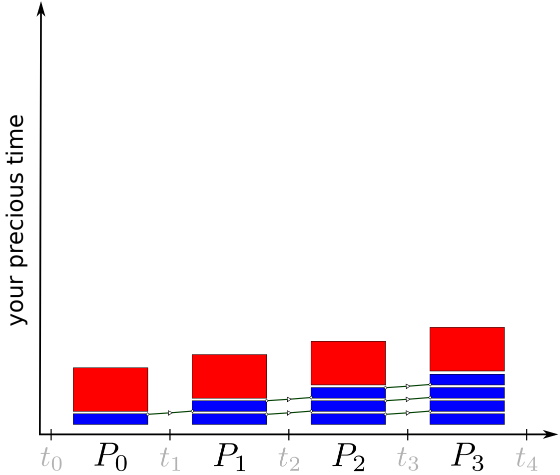
PFASST: time-steps

predictor (serial)



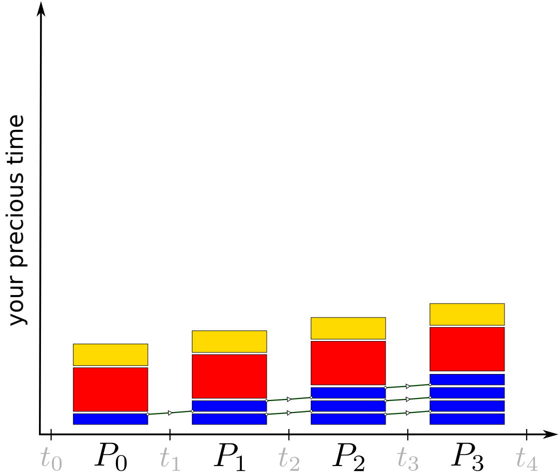
PFASST: time-steps

fine sweep (parallel)



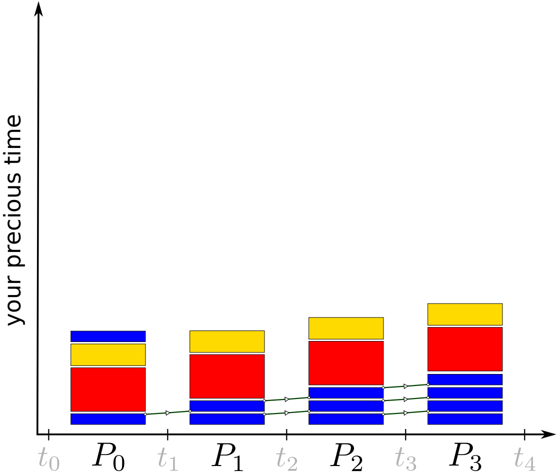
PFASST: time-steps

middle sweep (parallel)



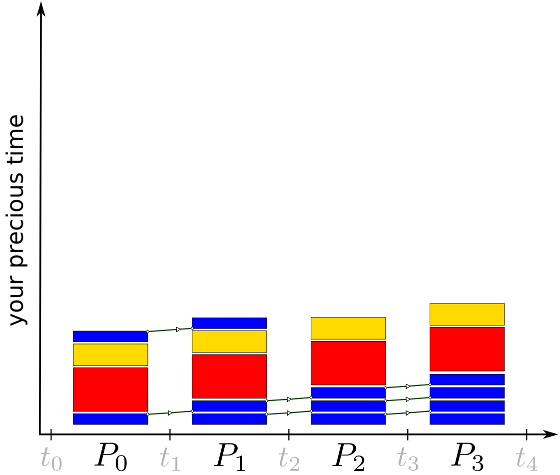
PFASST: time-steps

coarse sweep (serial, blocking)



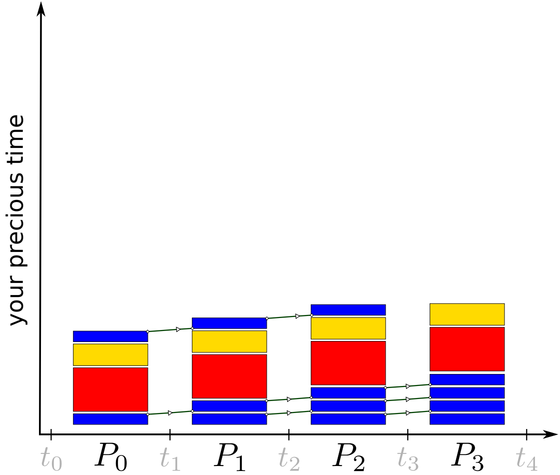
PFASST: time-steps

coarse sweep (serial, blocking)



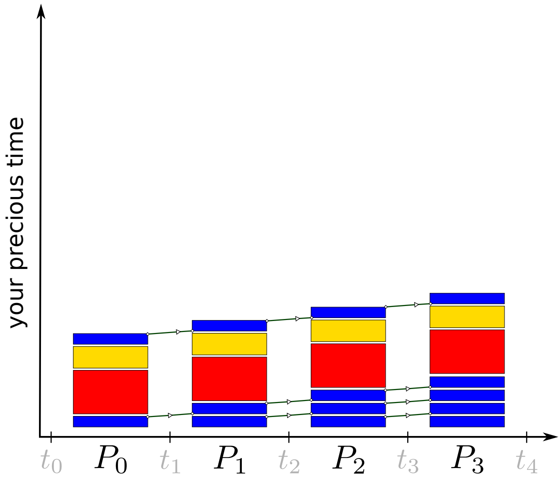
PFASST: time-steps

coarse sweep (serial, blocking)



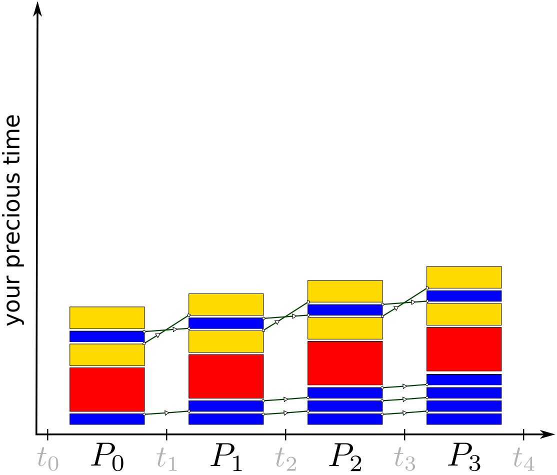
PFASST: time-steps

coarse sweep (serial, blocking)



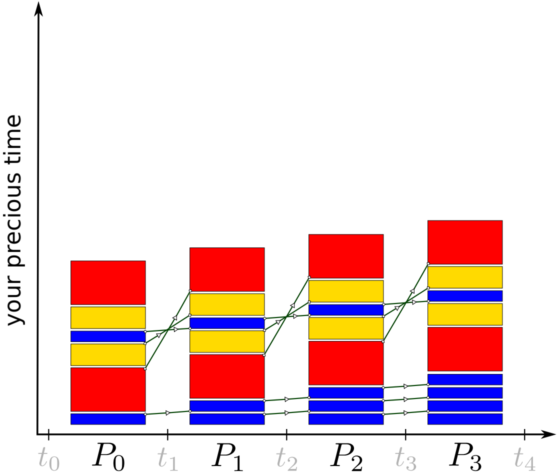
PFASST: time-steps

middle sweep (parallel, non-blocking)



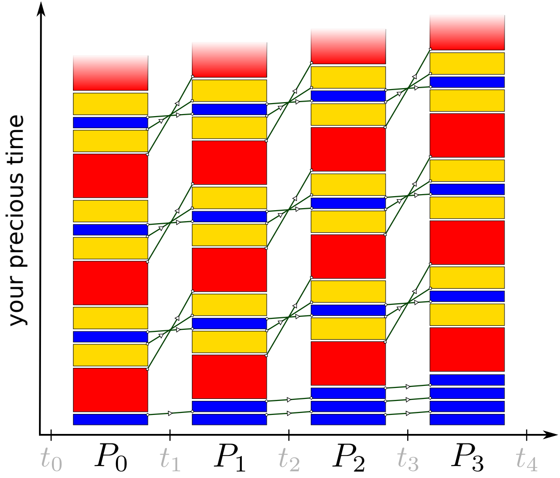
PFASST: time-steps

fine sweep (parallel, non-blocking)



PFASST: time-steps

overview



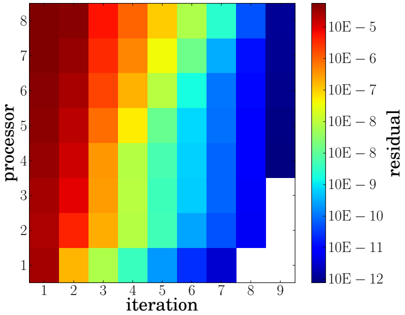
PFASST: visualization of real runs

PFASST: convergence study

8 processors

Scenario:

- 1D viscous Burger's equation
- semi-implicit SDC, 2 levels
- WENO + linear MG in space
- "mildly stiff" case, $\nu = 0.1$
- space-time coarsening in SDC nodes, dofs and discretization
- residual control, $\mathrm{tol} < 5 \cdot 10^{-12}$

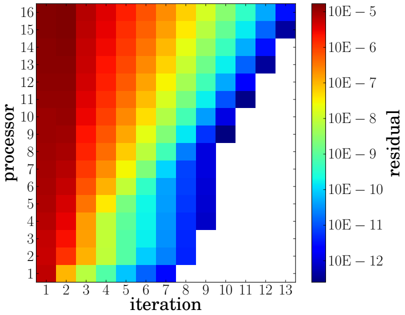


PFASST: convergence study

16 processors

Scenario:

- 1D viscous Burger's equation
- semi-implicit SDC, 2 levels
- WENO + linear MG in space
- "mildly stiff" case, $\nu = 0.1$
- space-time coarsening in SDC nodes, dofs and discretization
- residual control, $\text{tol} < 5 \cdot 10^{-12}$



Construction sites

- Can we characterize problems prone to poor efficiency in time?
  $\rightarrow$ e.g. strong non-linearities, multiple scales/speeds
- Can we do something about these problems?
  $\rightarrow$ Krylov subspace enhanced PFASST?
- What is the link to “classical” space-time multigrid theory?
  $\rightarrow$ SDC as relaxation, MGSDC (see MM’s talk after lunch)
- What are the effects of multiphysics-based coarsening?
  $\rightarrow$ work in progress, see next slide. . .

Outlook: application-tailored space-time coarsening

The parallel *N*-body solver PEPC, PFASST-enhanced

Previous results (SC'12 paper, BG/P system)

- hybrid parallelization of spatial solver saturated at 8,192 cores
- 32 parallel time-steps (262,144 cores) with PFASST led to additional speedup of 7, close to theoretical peak
- spatial coarsening: reduced order of discretization (strategy 3) via less accurate multipole expansion

Upcoming project (see abstract by M. Winkel et al.)

multiphysics-based coarsening for plasma physics applications

Outlook: PFASST for particle simulations

Up to now: only 1st-order ODEs

For MD simulations: 2nd-order ODEs, hamiltonian systems

Key challenges:

- conservation of energy
- very cheap/fast serial time-stepper
- spatial coarsening

See Michael Minion's talk after lunch!

From SDC to PFASST

June 18, 2013 | Robert Speck[1,2], Daniel Ruprecht[2], Matthew Emmett[3]

[1]Jülich Supercomputing Centre (Forschungszentrum Jülich, Germany)
[2]Institute of Computational Science (USI Lugano, Switzerland)
[3]Center for Computational Sciences and Engineering (LBNL, USA)

P.S.

We are currently planning to organize a workshop on

"Space-time multilevel methods"

- at Jülich Supercomputing Centre, Germany
- in 2014, probably also in June
- in collaboration with Università della Svizzera italiana, Lugano