



PDAF - The Parallel Data Assimilation Framework: Experiences with Kalman Filtering

Lars Nerger, Wolfgang Hiller, and Jens Schröter

Alfred Wegener Institute for Polar and Marine Research
Bremerhaven, Germany

Inerger@awi-bremerhaven.de

Overview



PDAF in the context of Kalman filters

Parallel performance of PDAF

Data Assimilation



Estimate system state (atmosphere, ocean, ...) on the basis of a numerical model and measurements by combining both sources of information.

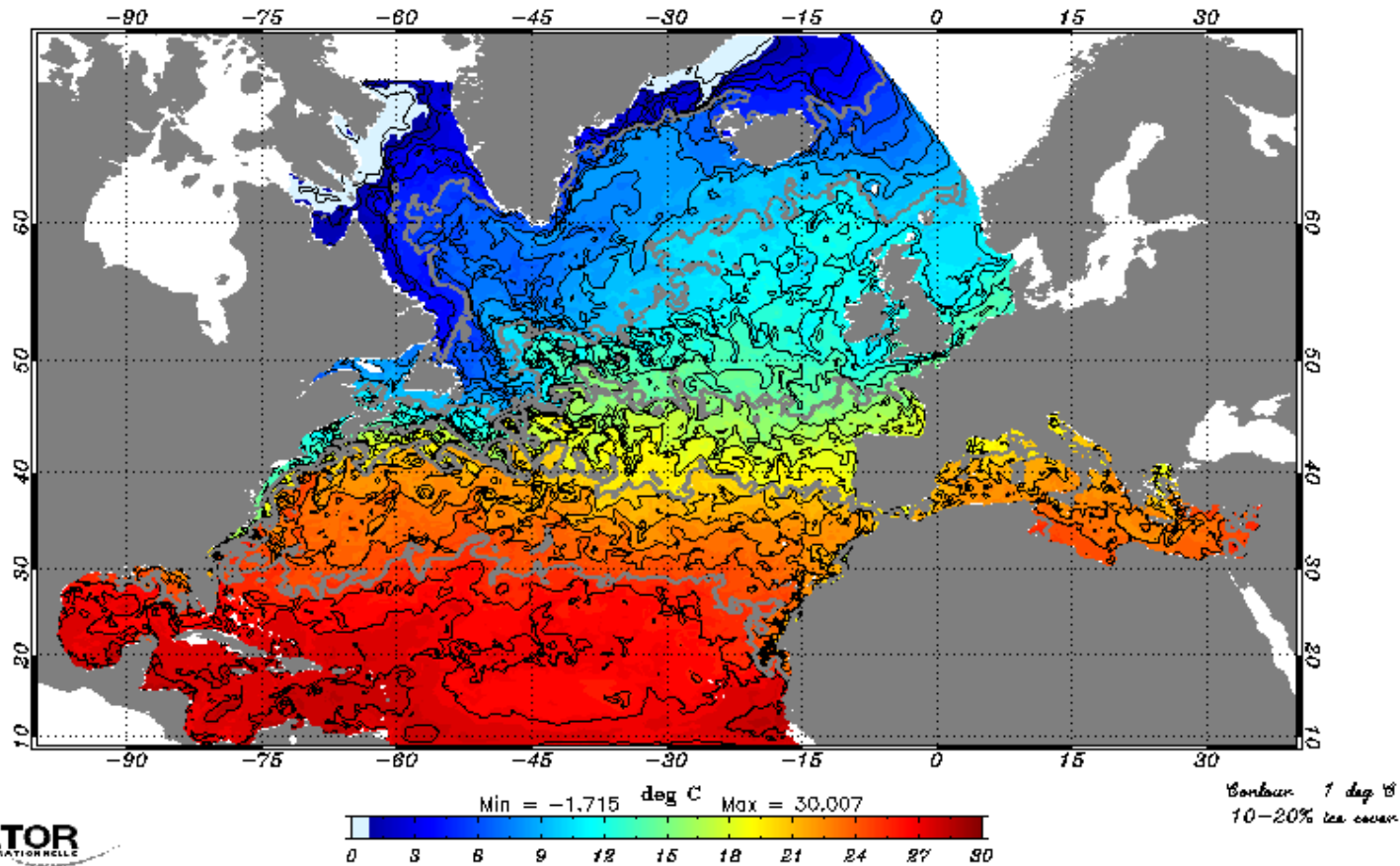
Filter \Leftrightarrow Smoother

Possible applications:
weather/climate forecasts
sensitivity studies

Data Assimilation



14-day forecast of ocean surface temperature



Ensemble-based Kalman filters



Initialization: Sample initial state and its error estimate by an ensemble of model states.

Forecast: Evolve each ensemble member with the non-linear (stochastic) model.

Analysis: Apply update step of the Kalman filter to ensemble mean or all ensemble states. Error estimate given by ensemble statistics.

Re-Initialization: Transform state ensemble to exactly represent updated error statistics.

Computational and Practical Issues



- Huge amount of memory required (model fields and ensemble matrix)
- Huge requirement of computing time (ensemble integrations)
- Natural parallelism of ensemble integration exists - but needs to be implemented
- Existing models often not prepared for data assimilation

PDAF: Considerations for Implementation

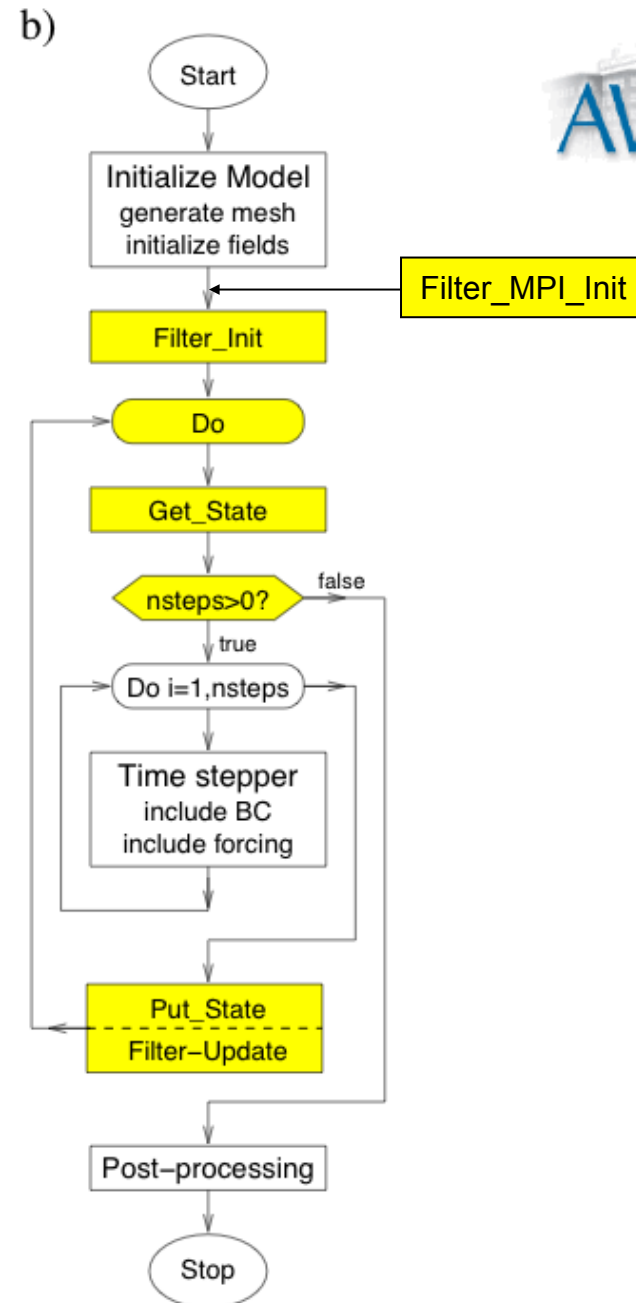
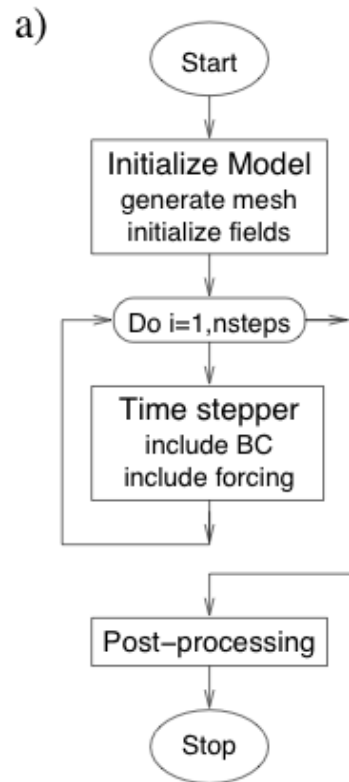


Logical separation of problem



Further considerations

- Combination of filter with model with minimal changes to model code
- Control of assimilation program coming from model
- Simple switching between different filters and data sets
- Complete parallelism in model, filter, and framework

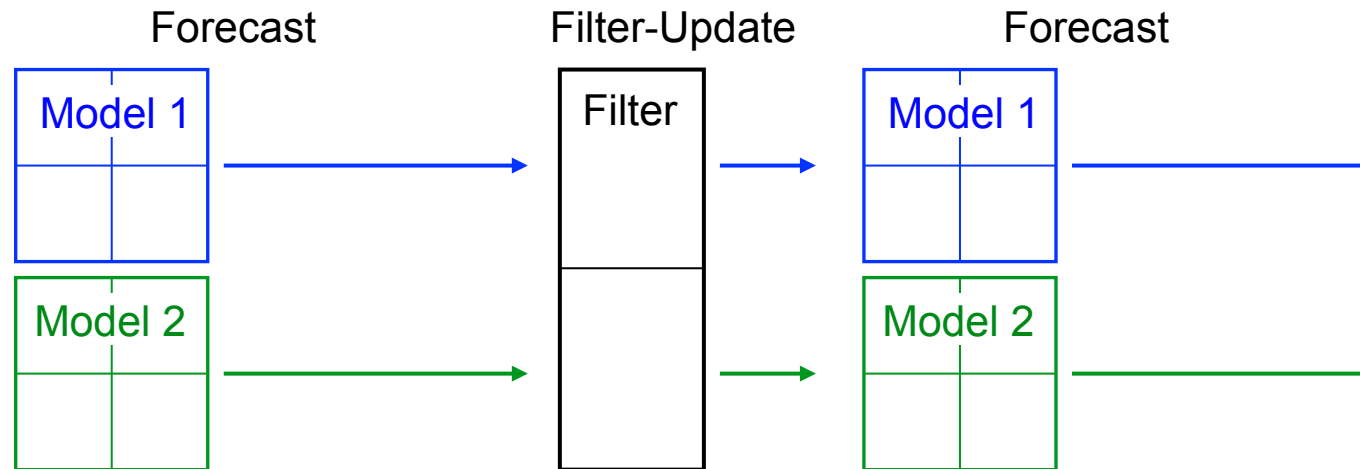


PDAF interface structure



- User-supplied routines for
 - field transformations between model and filter
 - observation-related operations
 - filter post-step
- Defined calling interface for
 - calls of framework routines
 - calls to user-supplied routines
- Interface independent of filter (almost)

2-level Parallelism

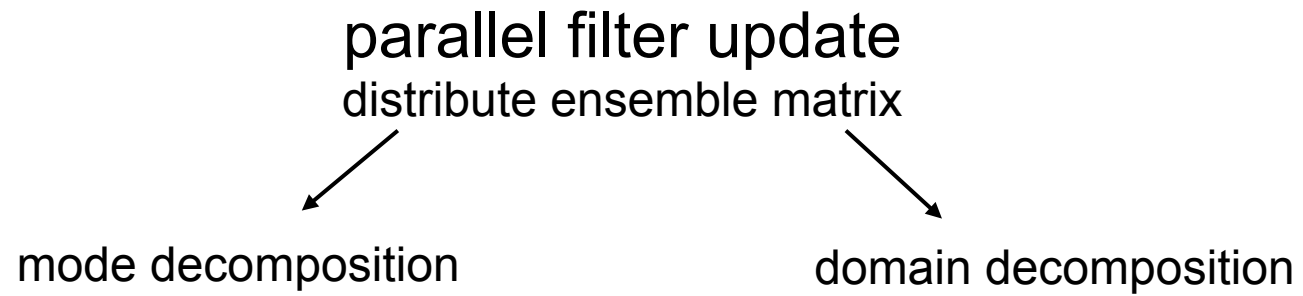


parallelization variants

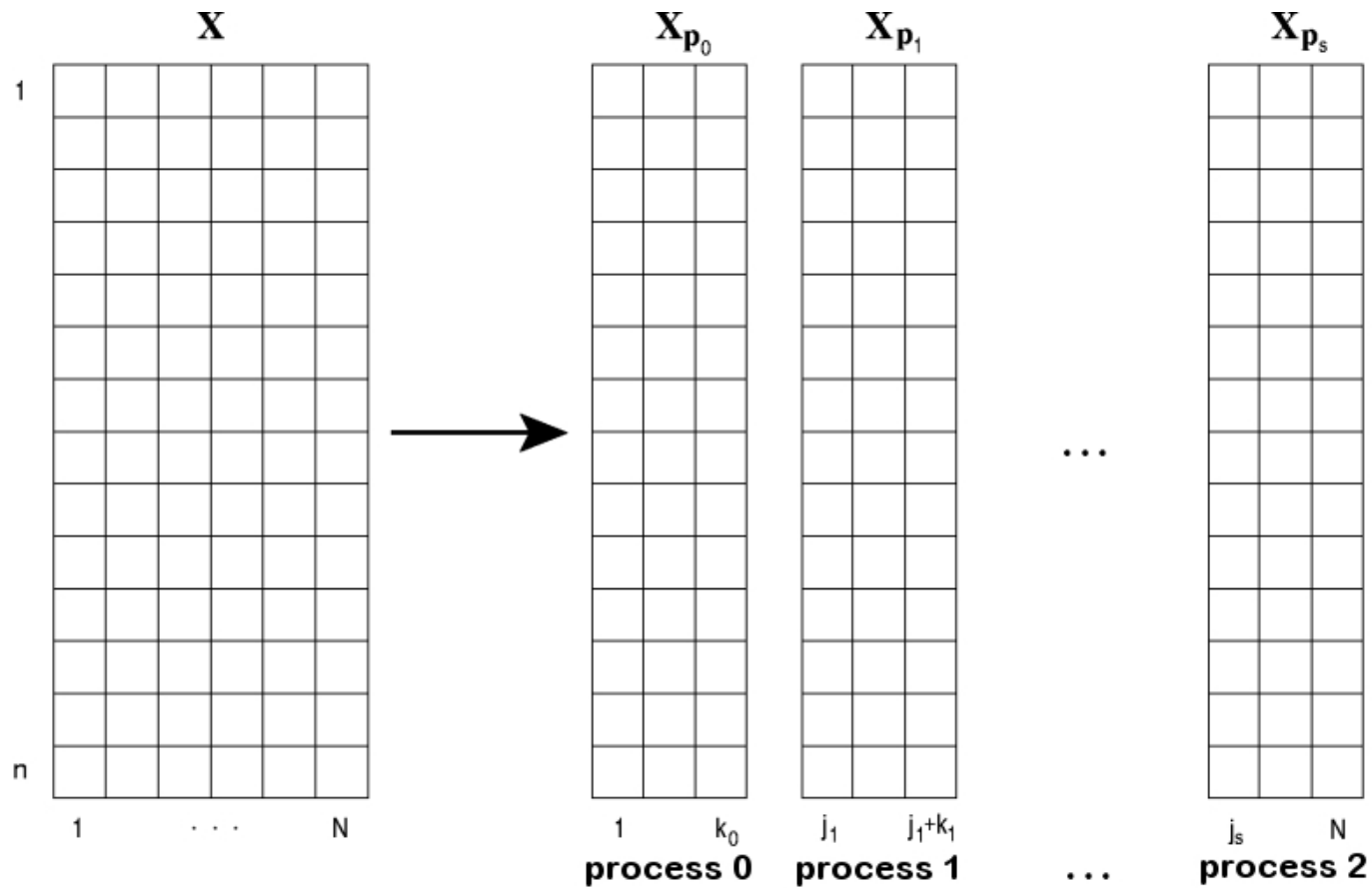
distribute operations

different processes for
model and filter update

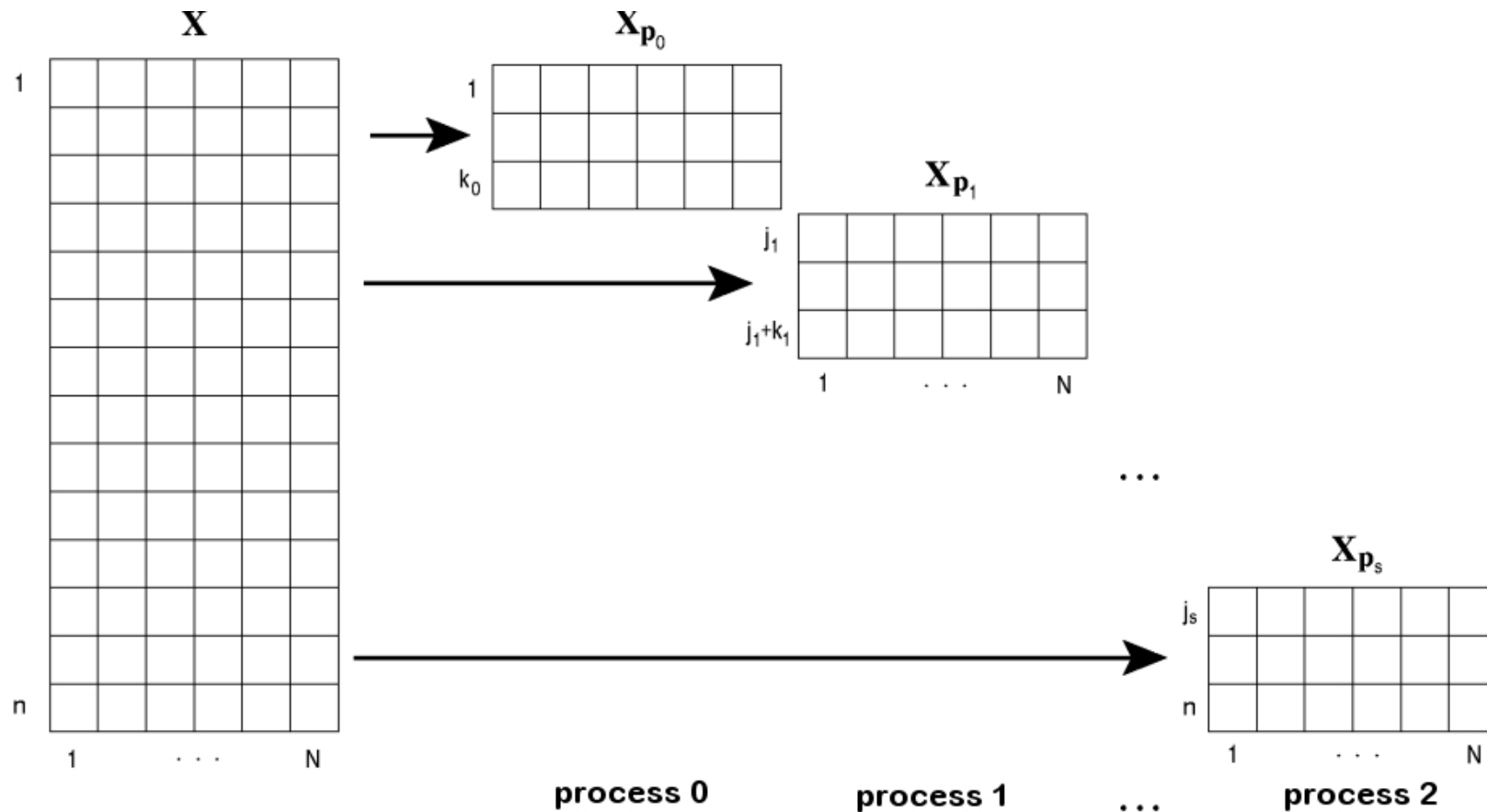
Filter update with
model processes



Mode decomposition



Domain decomposition



MPI parallelization



- Distribute model integrations
- Distribute filter update step

- 3 communicators
 - *Comm_Model*: model tasks
 - *Comm_Filter*: filter processes
 - *Comm_Couple*: communication between model and filter

Current KF algorithms in PDAF



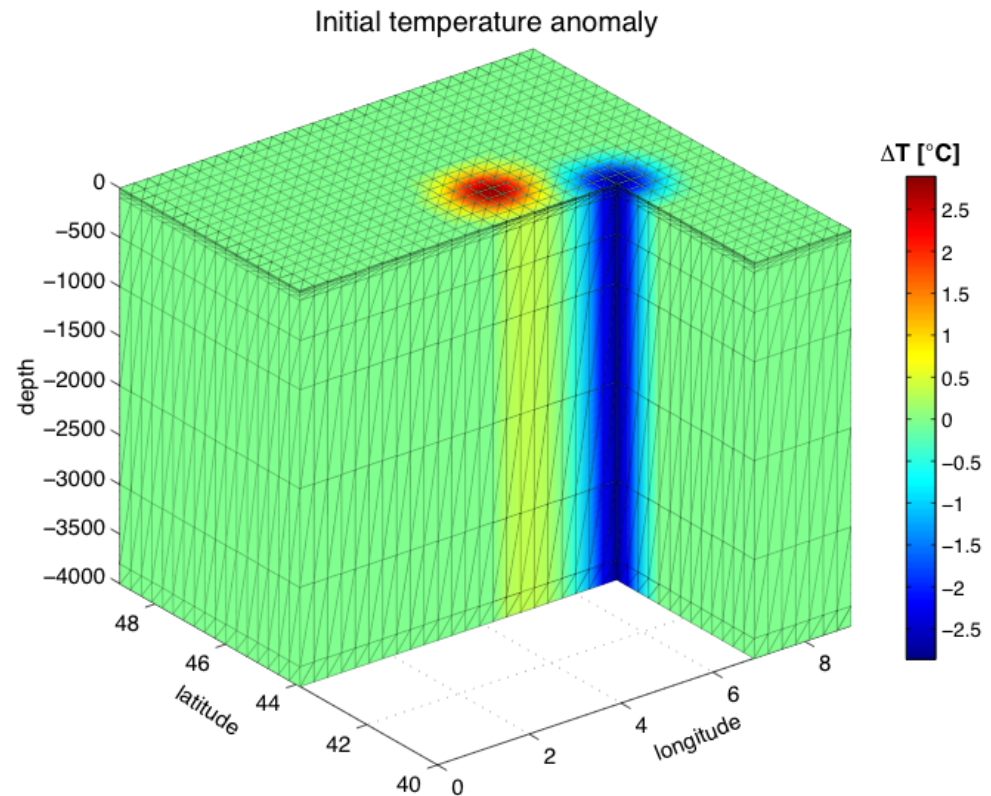
- Ensemble Kalman filter (EnKF, Evensen, 1994)
 - widely used
 - fully nonlinear error forecast
- SEEK filter (Pham et al., 1997)
 - explicit low-rank (error-subspace) formulation
 - linearized error forecast
- SEIK filter (Pham et al., 1997)
 - combination of strengths of EnKF and SEEK

3D box experiment

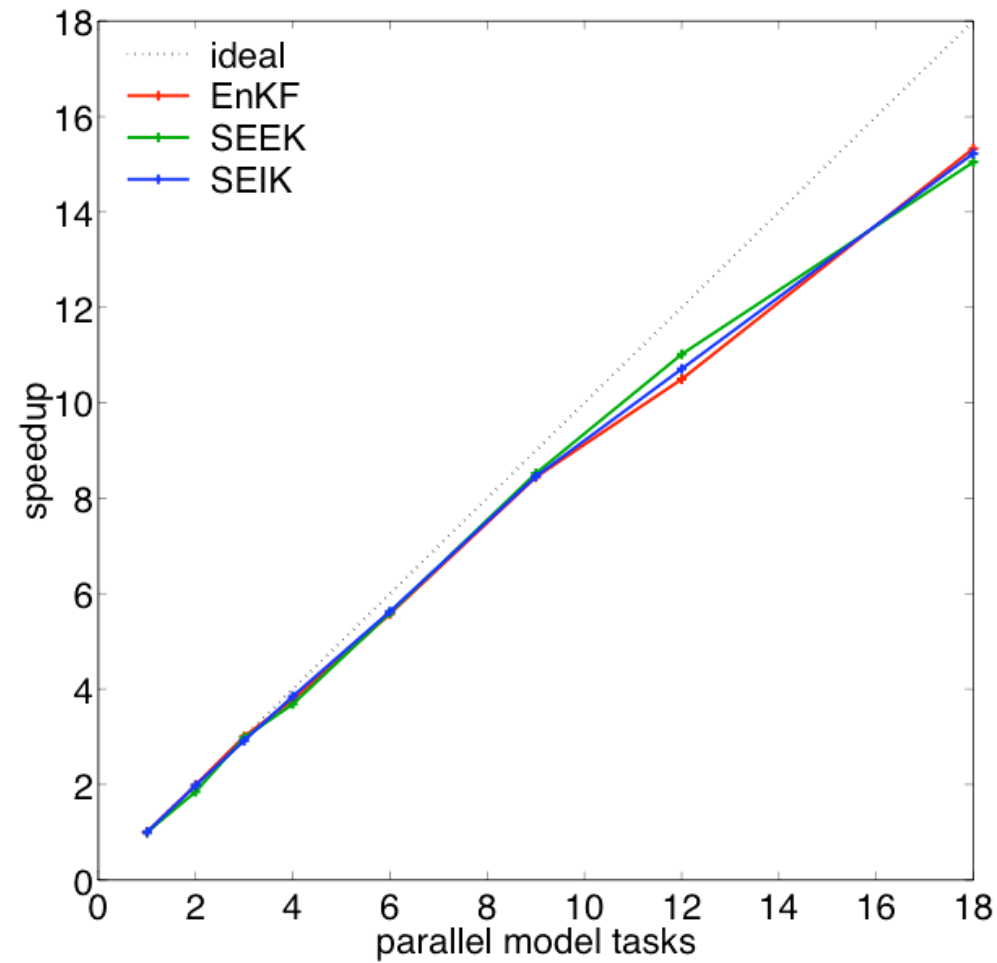


- finite element model FEOM
- 31x31 grid points, 11 layers
- nonlinear problem: interacting baroclinic Rossby waves
- Assimilate sea surface height each 2.5 days over 40 days

(FEOM: Danilov et al., Ocean Modeling, 2004)



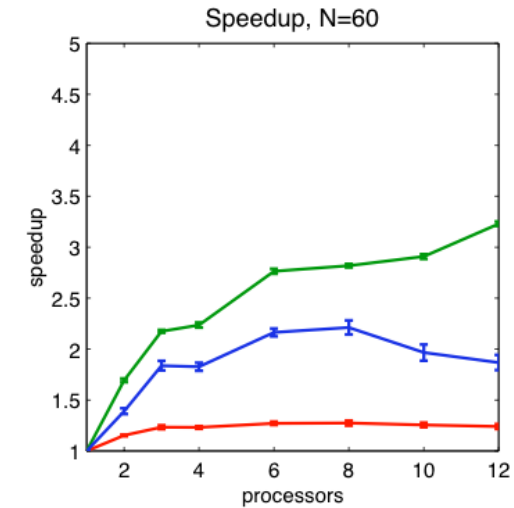
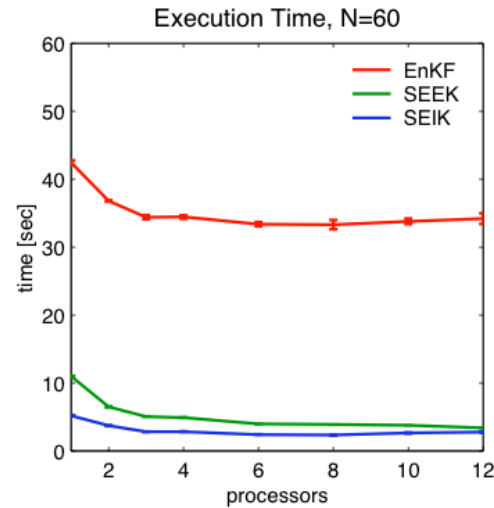
Speedup of PDAF



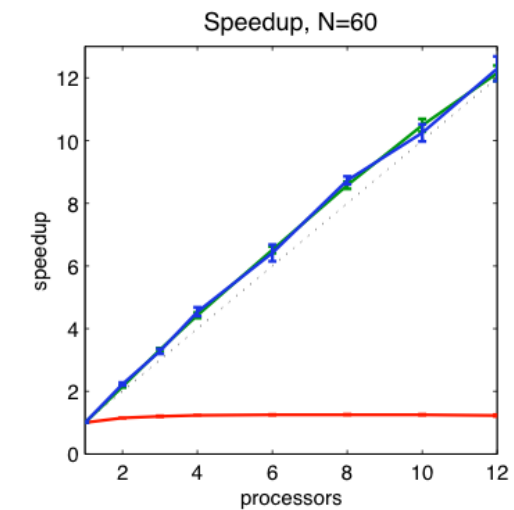
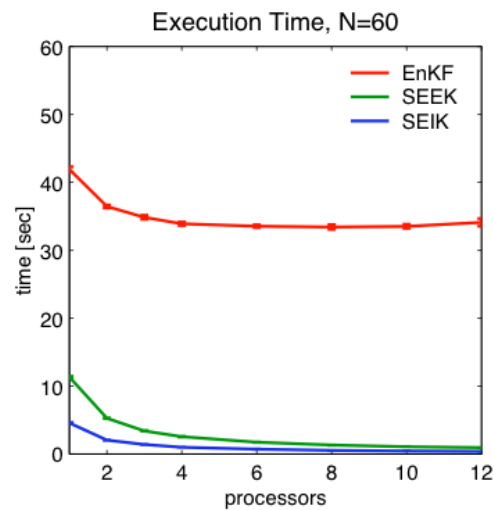
Parallel Efficiency of Filter Update



Mode decomposition



Domain decomposition

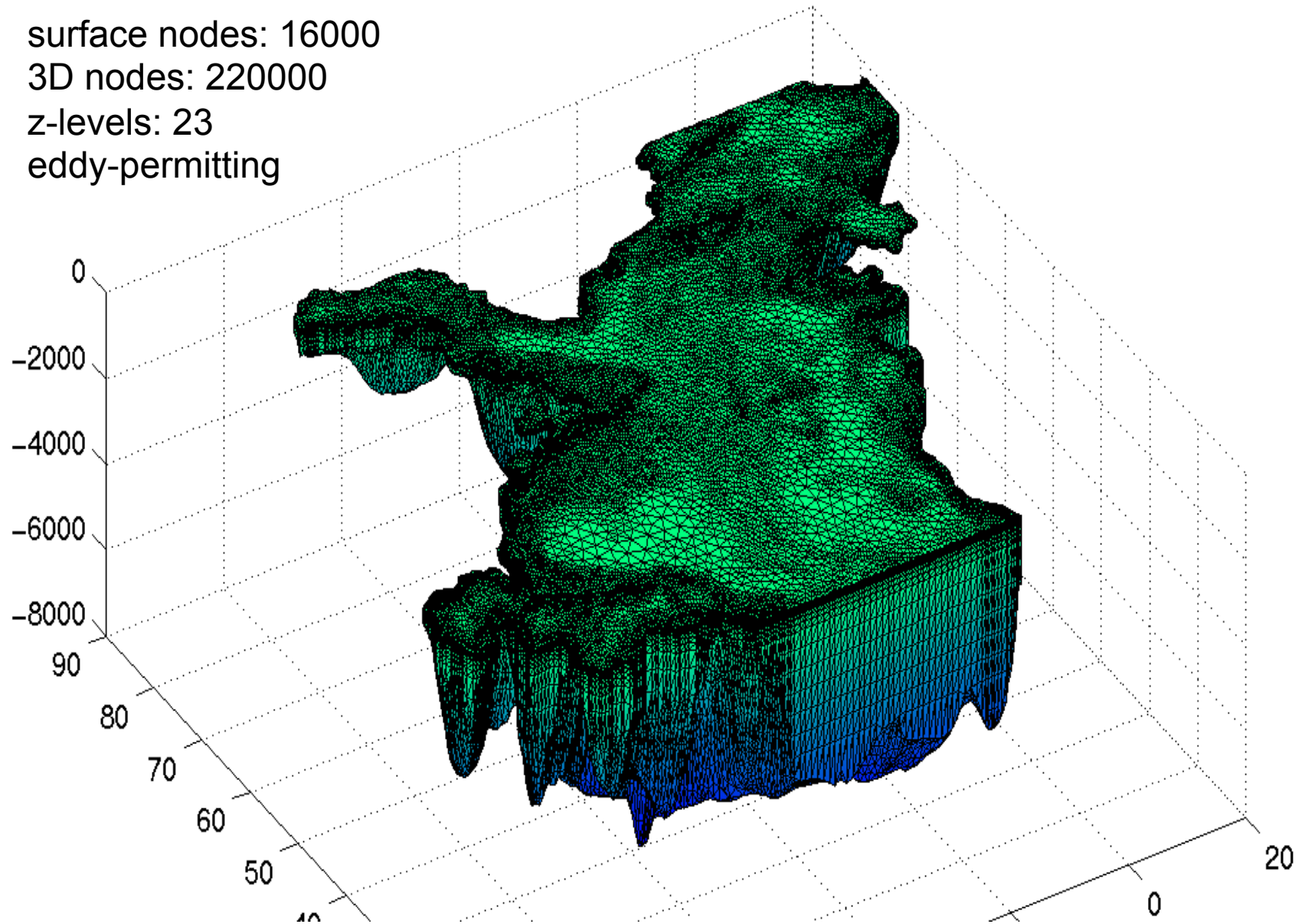


(ens. size = 10)

Further Example: FEOM North Atlantic



surface nodes: 16000
3D nodes: 220000
z-levels: 23
eddy-permitting



Summary



- Parallel Data Assimilation Framework PDAF
 - Simplified implementation of assimilation systems
 - Flexibility: Different assimilation algorithms and data configurations within one executable
 - Full utilization of parallelism
 - High parallel efficiency

Future directions



- Extensions of PDAF
 - more advanced filters (localization, adaptivity)
 - smoother algorithms

- Data assimilation applications (oceanography)
 - FEOM
 - stability of North Atlantic circulation
 - OPA-Model (with C. Böning, IFM-Geomar, Kiel)
 - large-scale circulation interannual to decadal