

Parallel CPR-Preconditioner in OPM

Markus Blatt

2015-03-12 Do

Outline

Parallel Iterative Solvers in DUNE

Parallel DUNE grid interface

Parallel CPR

Outlook / Summary

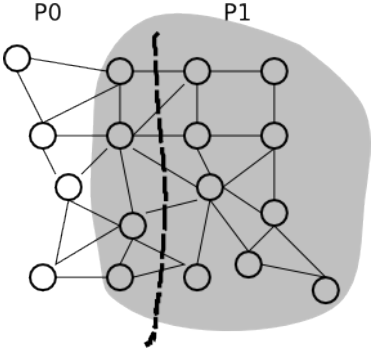
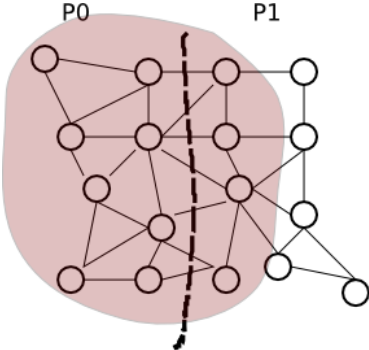
Design Decisions in dune-istl

- ▶ Reuse data structures and algorithms
- ▶ Separate data structures and parallelisation.

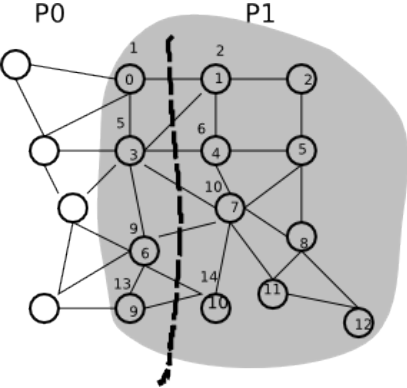
Parallel Index Sets

- ▶ Virtual view on global data structures
- ▶ Used only for communication.
- ▶ Precompute communication patterns
- ▶ Save user from underlying parallelisation paradigm.

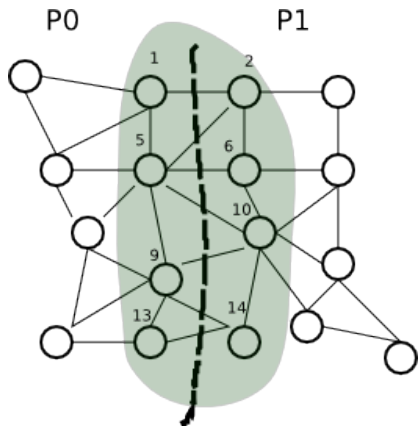
Graph based Partitioning I



Graph Global/Local Numbering



RemoteInformation/Communication



- ▶ Target and source of the communication is chosen by sets of attributes
- ▶ Buffered communication takes place at once.

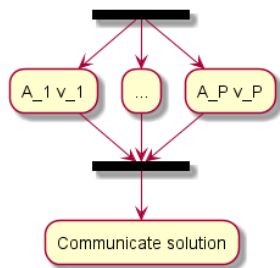
Parallel Linear Systems

- ▶ Uses augmented linear systems
- ▶ Let $\{N_p\}_{p=1}^P$ be a partitioning of our index set I .
- ▶ $I_p, N_p \subset I_p$, such that for all $k \in N_p$ with $|a_{kj}| + |a_{jk}| \neq 0$ also $k \in \tilde{I}_p$ holds.

$$I_p \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right. N_p \left\{ \begin{array}{l} \\ \\ \\ \end{array} \right.$$

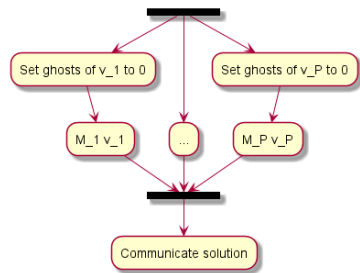
A_{pp}	*
0	I

Global Matrix Vector Product



- ▶ Each process knows the correct values of the global vector v
- ▶ After communication the result is consistent.

Hybrid Smoothers



- ▶ Each process knows the correct values of the global vector v
- ▶ After communication the result is consistent.

Parallelization Agnostic Solvers I

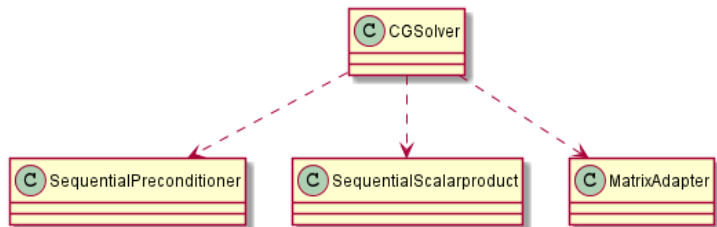


Figure: A Sequential Conjugate Gradient Solver

Parallelization Agnostic Solvers II

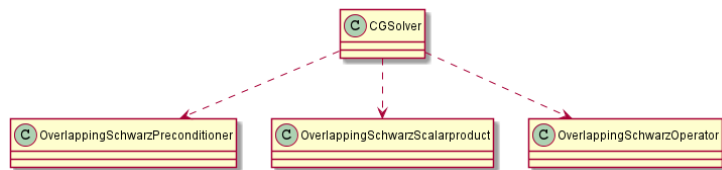


Figure: A Parallel Conjugate Gradient Solver

DUNE's Grid Partitions



$e=0$

$e=1$

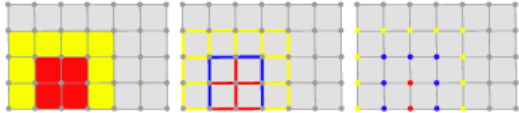
$e=2$



$e=0$

$e=1$

$e=2$



$e=0$

$e=1$

$e=2$

red	interior
green	overlap
yellow	ghost
blue	border
magenta	front
gray	not stored

Parallel Cornerpoint Grid

- ▶ Implements DUNE's parallel grid interface with extensions.
- ▶ Uses parallel index sets.
- ▶ Read eclipse file on every processes.
- ▶ load balance to create a distributed grid (Zoltan).
- ▶ one cell of overlap
- ▶ MPI

Parallel Simulator Properties

- ▶ Created from eclipse file.
- ▶ Uses CPGrid's scatterData to distribute

Propagating Parallel Index Sets to Solvers

- ▶ Uses `boost::any` as additional solver parameter
- ▶ Could be used for other things (PETSc?), too

Outlook / Summary

- ▶ TODO
 - ▶ Parallel well handling
- ▶ Bottlenecks
 - ▶ Sequential eclipse IO