DEISA Training
March 5-6th 2008

Distributed European Infrastructure for Supercomputing Applications

Coupled Computations on the Heterogeneous DEISA Network

HLRS

T. Bönisch

H. Klimach

Requirements of Aero-Acoustics

Distributed European Infrastructure for Supercomputing Applications

- Direct simulation of aero-acoustic phenomena in a flow field is a typical multi-scale problem:
  - Relativly small flow effects are generating noise (within the scale of up to **1 m**)
  - Propagation of sound happens on much larger distance with generally with rather long wavelength. Distances looked at here are rather in the scale of **1 km**.
- Different timescales
  - The flow phenomena which need a high spatial resolution need also a higher temporal resolution.
- Different equations
  - To take all effects in the flow into account, the full Navier-Stokes equations have to be solved
  - For the wave propagation however the linearized Euler equations are sufficient for a physical solution

KOP3D

- Direct simulation of Aero-Acoustic phenomena in flows
- Explicit solvers (compressible flows)
- Mixing different numerical models:
  - Structured meshes far from the body
    - Usually with Finite Volume or Finite Differences
  - Ustructured meshes near the body => time `CPU` consuming ($\Delta x, \Delta t$ small)
    - Usually with Discontinuous Galerkin discretization
- 3 parts:
  - `Hydsol` on unstructured grid
  - `Euler3D` on structured grid
  - `KOP3D` coupling part – space and time interpolations

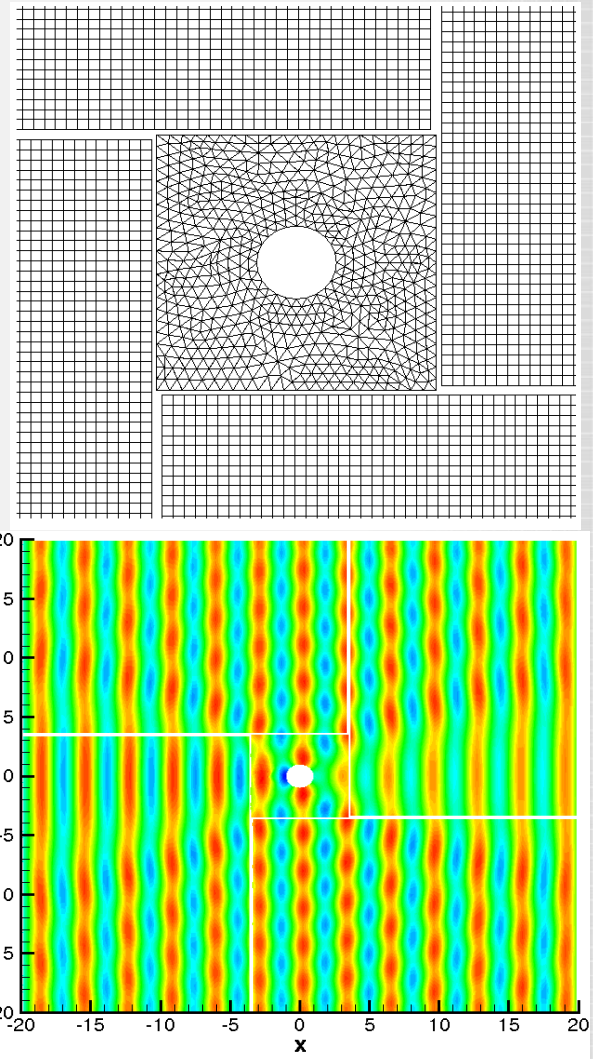


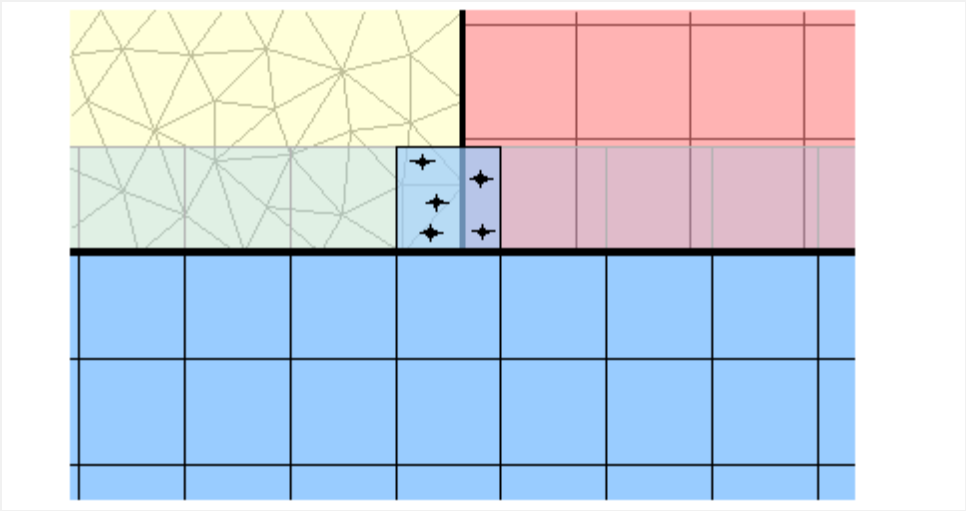
Figure: *Unstructured and structured meshes - Pressure field*

Time Integration: ADER

- High order scheme
- Permit good physical resolution, even on coarse meshes
- Piecewise polynomials as initial conditions for generalized Riemann problem instead of piecewise constants
- Taylor series in time as solution for these problems
- So we can gain **A**rbitrary high orders using **DER**ivations
- This is the time integration scheme used in KOP3D
- Usual order of space and time discretisation is 4 to 6.
- Computation of time derivations is a big computational effort

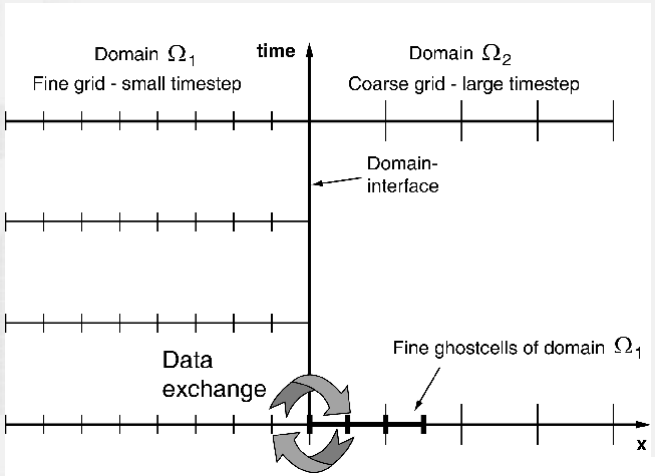
Coupling Heterogeneous Domains

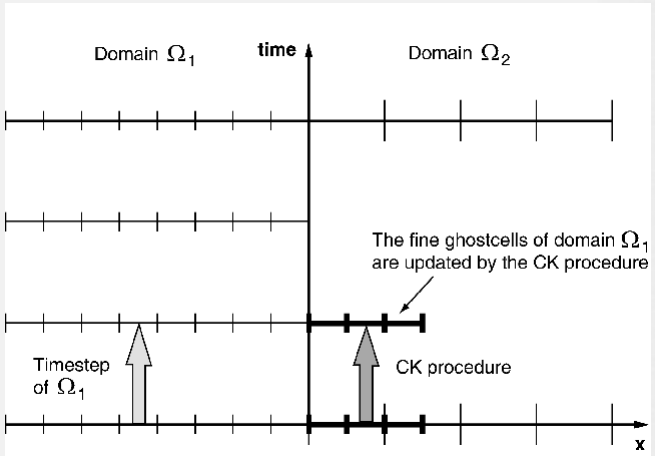
- Coupling is done via Ghost Cells
- Ghost Cells are constructed by discrete points
- Values at points are computed by the corresponding neighbor domain
- Necessary information about neighborhood reduced to minimum

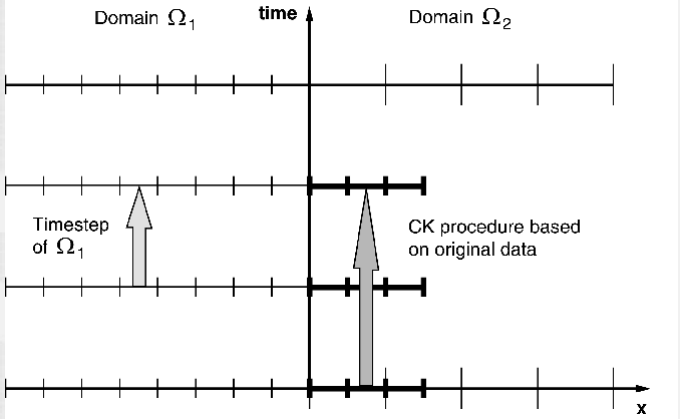


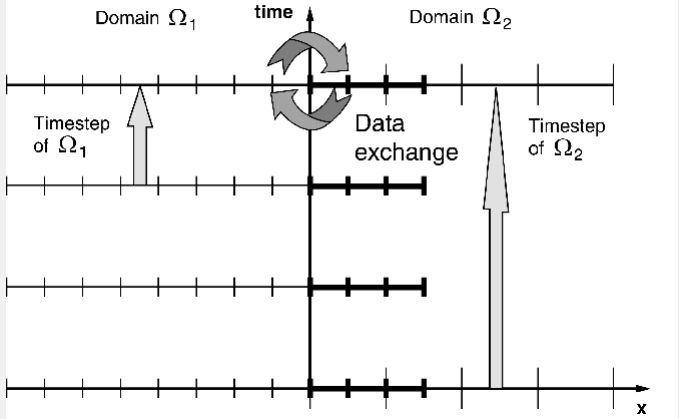
Temporal Coupling

- Subcycling allows different timesteps in neighboring domains









Goals in DEISA

Distributed European Infrastructure for Supercomputing Applications

- Running the different domains in parallel
- Spread the calculation across several sites
  - Enables larger simulations on combined computational strength
  - Involves new requirements on inter-site resource management
  - Computation has to be run simultaneous on each site
  - Communication has to be handled between the involved clusters
- Taking advantage of architectural features on different platforms for the different numerical requirements of the domains
  - For example running the solver for structured meshes with huge areas and so high memory requirements on the NEC SX8 vector machine and the highly local bad vectorized unstructured solver on a cluster of scalar processors.

Parallelization

- Two Levels of parallelism
  - Domains can be computed in parallel
  - Each domain itself may be subdivided into smaller computational partitions itself
- MPI only is used for communication in the parallel program
- Partitioning of domains with unstructured meshes with Metis
- Communication within each domain at each timestep
- Communication between neighboring domains only at matching subcycles
- Distribution of computational resources on the domains requires a-priori estimation of the amount of work needed in each domain
  - This mapping is not trivial and up until now needs to be done manually

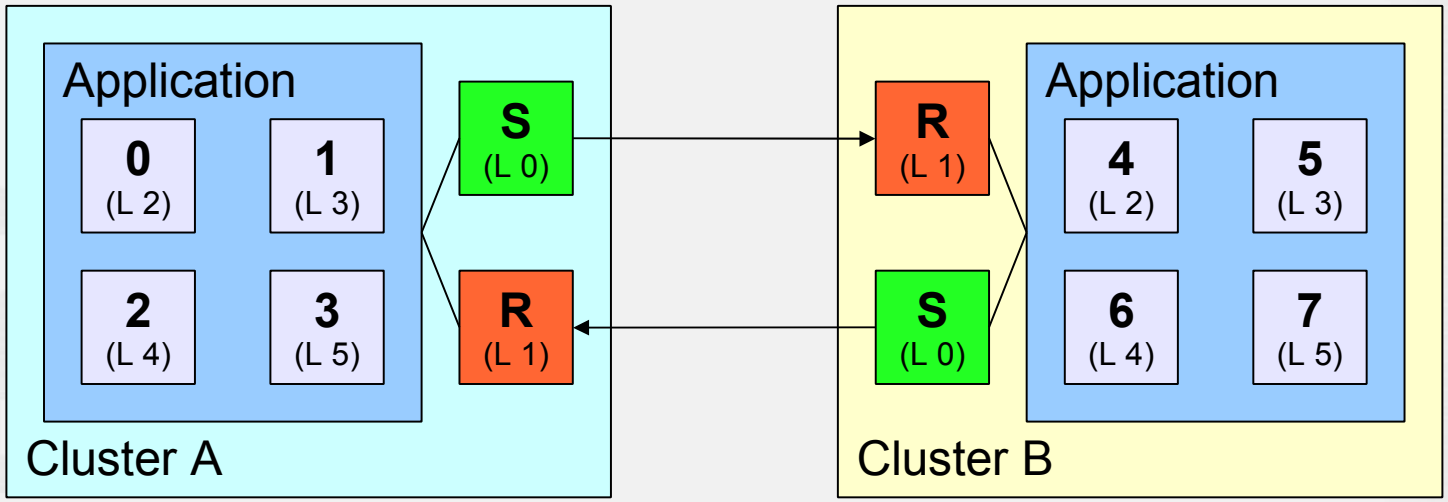
Parallel Coupling

- Within a single domain partitions exchange values using halos
- Between different domains the concept is quite similar, but discrete points are used
  - Avoids problems by non matching meshes, as each point can be uniquely located in a single domain
- A list of points for each neighbor is collected and communicated
- Values at this points are exchanged between neighboring domains by MPI Point-to-Point communication
- Only neighbors on a matching subcycle are involved in communication

PACX-MPI

- Library to enable communication within the MPI context on a heterogeneous network
- Transparent to the application
  - Acts as intermediate layer between the application and the MPI-Library
  - No changes to program needed, relinking is sufficient
  - Usual MPI-Environment for the application
- Can use different protocols between different clusters than within each cluster
- Uses native MPI-Library on each cluster
- Requires 2 daemon processes on each cluster for inter-cluster communication

PACX-MPI Communication



- PACX-Communication setup
  - Local MPI-Ranks get remapped to global ranks
  - First local processes are used for inter-cluster communication
  - Application just sees the global process ranks

KOP3D on PACX-MPI

Distributed European Infrastructure for Supercomputing Applications

- Using PACX-MPI to run the heterogeneous domains of KOP3D on the heterogeneous network of clusters
- No need to leave MPI context in the application
- But it must be known, on which cluster each processor is running
  - Commandline parameter to tell each part, which domains it should serve (structured or unstructured ones)
- Usual MPI startup on each cluster
- Map requirements of different domains to features of different clusters
  - Domains with structured meshes on a vector machine
  - Domains with unstructured meshes on a scalar cluster
- Balancing of loads on different clusters (resource estimation)

Testcase

- Aero-Acoustic 3D Testcase
  - Scattering of soundwaves at a sphere of 1 m radius
  - Unstructured mesh: 9874 DG elements around the sphere
    - 5.5 x 3.5 x 3.5 m
  - Structured mesh: ~42 million FD cells for the far field
    - 102.2 x 57 x 57 m
  - Spatial discretization with 8th order scheme

Performance

Distributed European Infrastructure for Supercomputing Applications

IA64 alone:

	0xSX	1xIA64	total
UNSTRUCT:	0,00	**2993,67**	2993,67
STRUCT:	0,00	23887,32	23887,32
KOP:	0,00	1012,37	1012,37
waiting:	0,00	0,00	0,00
KOP calculating time:			**1012,37**
Total CPU time:			**27893,35**
Total elapsed:			**27924,78**

NEC SX-8 alone:

	1xSX	0xIA64	total
UNSTRUCT:	7745,82	0,00	7745,82
STRUCT:	**2870,66**	0,00	2870,66
KOP:	321,15	0,00	321,15
waiting:	0,00	0,00	0,00
KOP calculating time:			**321,15**
Total CPU time:			**10937,62**
Total elapsed:			**10966,23**

Combined strength of both architectures by coupling them with PACX-MPI:

coupled	1xSX	1xIA64	total
UNSTRUCT:	0,00	3019,24	3019,24
STRUCT:	2869,46	0,00	2869,46
KOP:	368,11	186,10	554,21
waiting:	10,98	153,22	164,20
KOP calculating time:			**390,02**
Total CPU time:			**6278,72**
Total elapsed:			**3207,09**

Equivalent to 8.7 IA64 CPUs or 3.42 NEC-SX8 CPUs with single architecture.

Conclusion

- Heterogeneity of both simulation and computational infrastructure can be beneficial if proper mapping of the attributes is done
- KOP3D performs well in heterogeneous environment running on PACX-MPI
- Using combined computational powers bigger and more realistic simulations become feasible
- DEISA enables such simulations on an european scale
- Still some work needed, inter-cluster management of resources is not yet solved, but co-scheduling is essential for these kind of runs
- New hardware more gets more and more diverse, think of GPUs, FPGAs and special purpose accelerators: You need to use all of the given resources efficiently, to get high performance!

Outlook

- Doing really big Direct Numerical Simulations including aeroacoustics