PEPR NumPEx – Exa-DoST – WP2 with Gysela

Yushan Wang   Laurent Colombet   Gabriel Antoniu

Julien Bigot   Etienne Ndamlabin   Silvina Caino-Lores

Arthur Jaquard   Bruno Raffin

Andrès Bermeo-Marinelli

PROGRAMME
DE RECHERCHE

NUMÉRIQUE
POUR L'EXASCALE

WP2 for Gysela

Identified focus:

- Data locality: GPU data, need to optimise data copy
- Data type: support `std::vector, Kokkos::View, etc`

```cpp
// cpp
Kokkos::View<double> view("v", N);
PDI_expose("my_data", view.data(), PDI_OUT);

// yaml
my_data: type: array, subtype: double, size: $N
```

```cpp
// cpp
Kokkos::View<double> view("v", N);
PDI_expose("my_data", view, PDI_OUT);

// yaml
my_data: kokkos_view
```

WP2 for Gysela

Identified focus:

- Dynamic scheduling between in-process and in-transit diagnostics
  - Gysela diag ≈ local reduction + global reduction
  - Perform local reduction in-process
  - send locally reduced data to Dask workers for global operations.
- Feedback mechanism
  - based on the diagnostic results, send information to simulation for restarting, parameter adjustment, trigger other diagnostics, etc

next steps for Gysela for Ariel

- PDI dev: focus more on netCDF plugin as Gysela has switched from hdf5 to netCDF
    - new features on HDF5 plugin can be easily ported to netCDF
- subfiles with hdf5 (Méline)
- new plugin for vtkhdf output, for visualization
- plugin for metadata/mesh management => conduit, modularity
- compression in netcdf (Yushan and Ariel/Méline)
    - porting what is available for HDF5 to netCDF
- Doreisa, for local reduction on GPU, then transfer to CPU for global reduction (PhD related)
- hdf/netcdf simple precision, but data in double precision, conversion managed by hdf/netcdf
- **kokkos view support**, reduce yml, size definition can be optional for subdata(PDI)
- feedback, close the loop
- plugin catalyst (Jacques and Ariel)

next steps for Gysela w WP2

- test diagnostics with Damaris? perf (PPP) evaluation vs D(or)eisa (Ariel and Etienne)
    - deisa for first attempt of new diagnostics, if validated/needed, move to MPI-based.
- manually reduce using Pycall (on GPU), then transfer to Deisa.
    - Need to care the metadata of the reduced data be coherent.
- testcase construction: 5D, fluid moment computes, checkpointing
- extend regex for other plugins

Confirmed next steps for Gysela WP2

- **kokkos view support**
    - not specially for Gysela, but can offer possibility of std::vector for Coddex
- feedback, close the loop
- plugin catalyst
- test code base:
    - 2X2V special simulation with computation and IO
    - 3X2V only for IO, 5D distribution function f on GPU, with MPI decomp for (Vx and Vy by block), exposed to PDI
        - hdf5 native compression for both hdf5 and netcdf
        - or change the MPI decomposition on X Y and Z to see if it is better for IO
        - insitu, local reduction
        - kokkos view support
        - eg: 256*512*64*128*64 minimal, 2048*2048*128*128*64 for exa