Multibunch PyHEADTAIL simulations

Transverse feedback stabilization of the beam in the FCC-hh

Pickups   Kicker

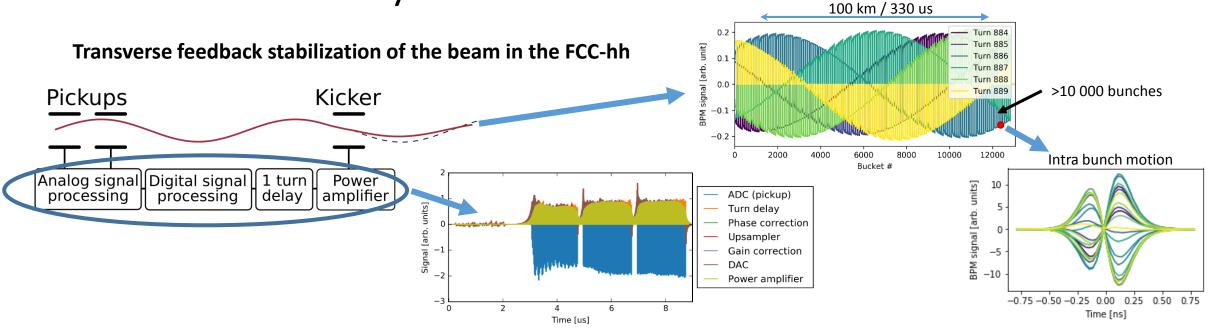| Analog signal processing | Digital signal processing | 1 turn delay | Power amplifier |

100 km / 330 us

>10 000 bunches

Intra bunch motion

- Challenges have been:
  - New code (bugs and optimization)
  - New kind of studies (relevant questions to be answered)
  - Amount of output data (relevant output parameters)
  - How to use the cluster efficiently
- So far the HPC PyHEADTAIL simulations have been focused on two cases
  - FCC-hh CBI damping
    - <20k turns, 10k bunches, 35k mppb, 39 turns wakes, detailed damper model
  - HL-LHC HOM tests
    - 150k turns, 3.6k bunches, 20k mppb, 3 turns wakes, ideal damper

- It is not only computing performance which is relevant but also the effort needed for launching and analyzing simulations
  - Best questions beat the best supercomputers
- Possible large variety of applications in future
  - Required simulation parameters vary a lot between studies
    - Wake field time scale from sub nanosecond to tens of milliseconds
    - Tracking time from hundreds to millions of turns
    - Number of bunches from one to tens of thousands
    - Number of macro particles per bunch from one to million(s)
  - **How to detect the performance bottle necks and optimize the cluster use?**

Some notes from the simulations

- In general the HPC cluster works well
- Some issues with the code and/or the mpi settings on the cluster
  - Issue in the MPI H5py limits the maximum number of processor to one node
    - Was problem multiple times during the last half year, various error messages
  - Timeout crash? when RF matched buckets are used
    - works well with the normal batch service (HTCondor + manually compiled MPI)
  - So far mvapich2 have been used, would the other MPI options work better?
    - *WARNING: Error in initializing MVAPICH2 ptmalloc library.Continuing without InfiniBand registration cache support.*
- Monitoring tools?
  - Challenging to optimize simulation parameters and the number of nodes needed, when requirements vary a lot between the studies
    - N_bunches, N_slices, N_macro_particles_per_bunch, used PyHT modules, etc.
    - Is it possible to develop easy (semi)automatic profiling for the customized simulations and new PyHEADTAIL modules in future?
  - Memory and processor loads and traffic between the nodes