FFTECP

An exascale FFT library for heterogeneous architectures

Alan Ayala

Collaborators: Stan Tomov, Azzam Haidar, Daniel Schultz, Jack Dongarra

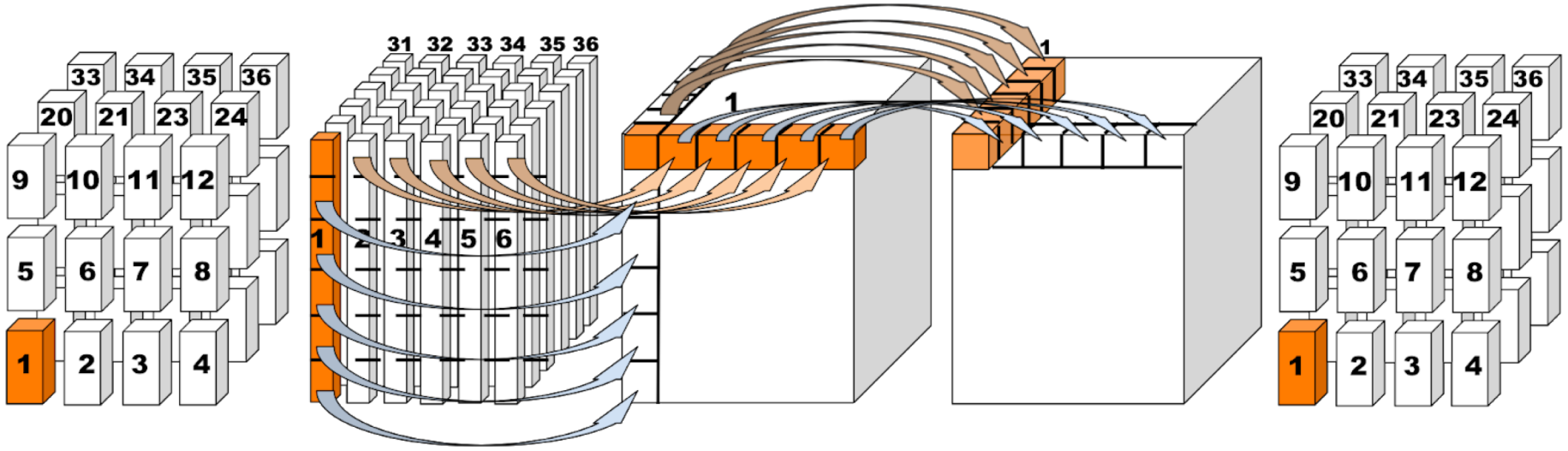
International HPC Summer School 2019
Kobe, Japan

Overview

- The FFT is one of the top 10 algorithms in the 20th century according to *Comp. in Sci. & Eng.*
- FFT takes the operation count for discrete Fourier transform from $\mathcal{O}(N^2)$ to $\mathcal{O}(Nlog(N))$
- Applications in molecular, dynamics, spectrum estimation, fast convolution and correlation, signal modulation, etc
- More than a dozen ECP applications use FFT in their codes, *e.g. LAMPPS, HACC.*
- Over 60 scientific software packages depend on and use FFTs, according on Spack package manager
- State-of-the-art FFT libraries like FFTW are no longer actively developed for emerging platforms
- Current FFT libraries do not scale on heterogenous platforms (*e.g. Several NVIDIA V100 GPUs*)

Approach

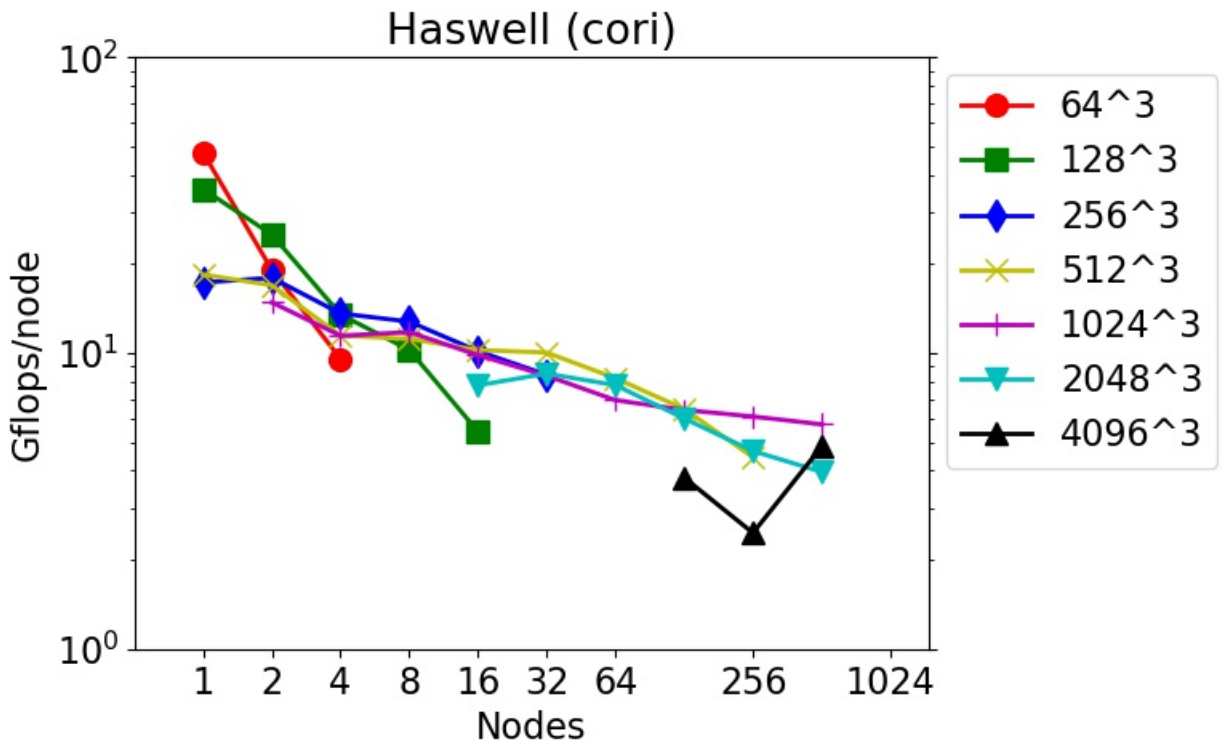
- The distributed 3D FFT is one of the most important kernels needed for particle applications
- 3D FFTs can be found via a combination of 1D/2D FFTs
- FFT-ECP leverages third-party 1-D FFTs from vendors or open-source libraries (FFTW3, CUFFT, MKL)



- **FFTECP features:** Uses **cufft (NVIDIA)** for 1D FFTs, communication overlapped with computation.
  Maximizes data throughput – PCI, NVLINK

State-of-the-art FFT distributed libraries

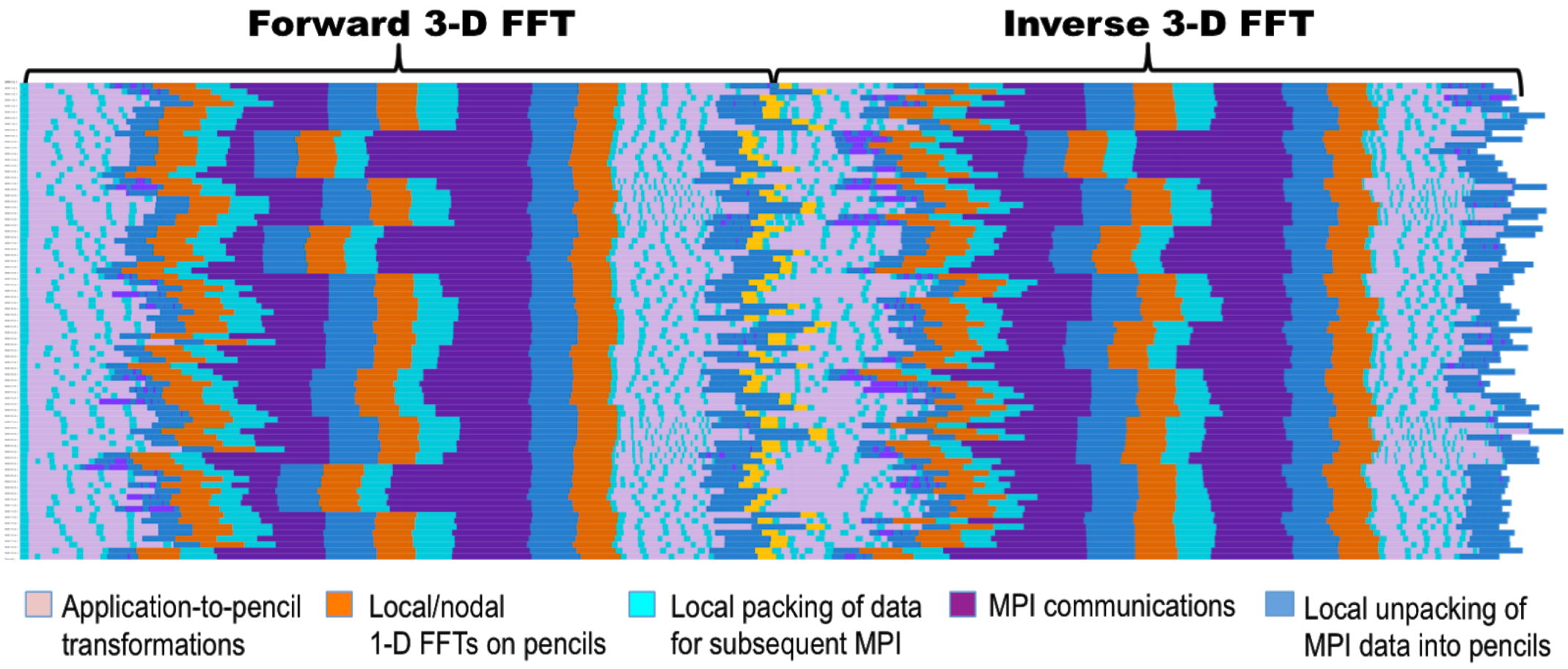
- There exists many parallel FFT libraries, such as P3DFFT, FFTW3, FFTMPI, SWFFT, MKL
- FFTMPI (Plimton et al., Sandia Lab) : focusses on weak scalability
- SWFFT (Pope et al., Argonne Lab) : focusses on strong scalability
- FFT-ECP uses the frameworks of FFTMPI and SWFFT



Performance for CPU parallel FFT codes is about ~20Glops/node

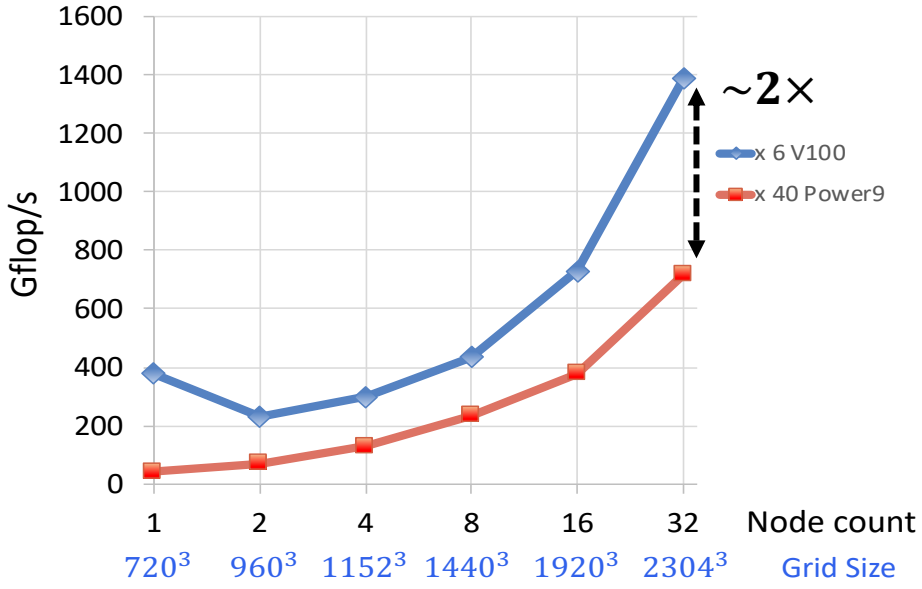
Performance analysis

- Trace for FFT computations using FFTMPI.

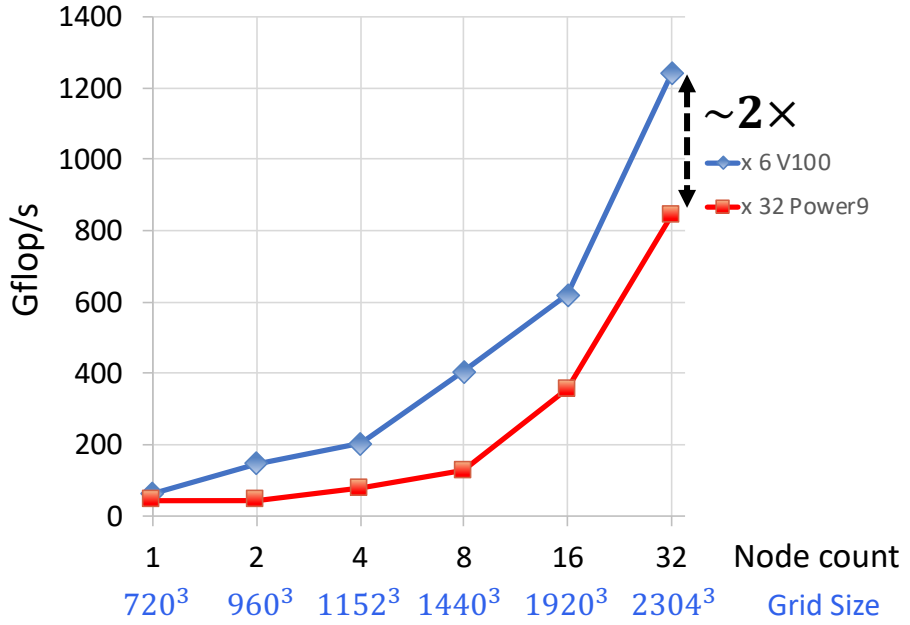


FFTECP weak scalability

FFTECP gets over 40 Gflops/node



Performance comparison FFTECP - FFTMPI



Performance comparison FFTECP - SWFFT

Some of the things ongoing and future work:

- Mixed precision capabilities
- Integration to ECP libraries
- Convolution and general kernels for applications, *e.g. for machine learning software*