

## FAST FOURIER TRANSFORM

The fast Fourier transform (FFT) is used in many domain applications—including molecular dynamics, spectrum estimation, fast convolution and correlation, signal modulation, and wireless multimedia applications. For example, distributed 3-D FFT is one of the most important kernels used in molecular dynamics computations, and its performance can affect an application’s scalability on larger machines. Similarly, the performance of the first principle calculations depends strongly on the performance of the FFT solver. Specifically, for the US Department of Energy (DOE), we found that more than a dozen Exascale Computing Project (ECP) applications use FFT in their codes.

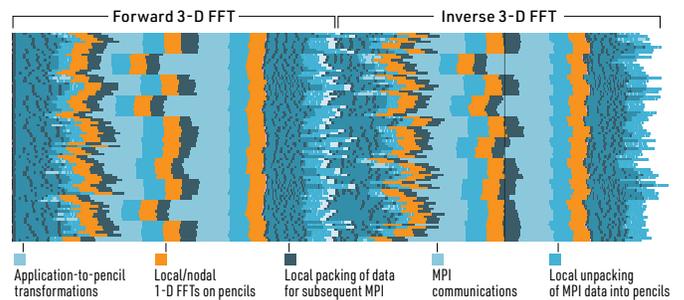
The current state-of-the-art FFT libraries are not scalable on large heterogeneous machines with many nodes or even on one node with multiple high-performance GPUs (e.g., several NVIDIA V100 GPUs). Furthermore, these libraries require large FFTs in order to deliver acceptable performance on one GPU. Efforts to simply enhance classical and existing FFT packages with optimization tools and techniques—like autotuning and code generation—have so far not been able to provide the efficient, high-performance FFT library capable of harnessing the power of supercomputers with heterogeneous GPU-accelerated nodes. In particular, ECP applications that require FFT-based solvers might suffer from the lack of fast and scalable 3-D FFT routines for distributed heterogeneous parallel systems, which is the very type of system that will be used in upcoming exascale machines.

We believe that the design of the existing libraries should be revisited and studied in order to develop a GPU-based, distributed, 3-D FFT library that can deliver high performance on current and future supercomputers. The main objective of the FFT-ECP project is to design and implement a fast and robust 2-D and 3-D FFT library that targets large-scale heterogeneous systems with multi-core processors and hardware accelerators and to do so as a co-design activity with other ECP application developers. The work involves studying and analyzing current FFT software from vendors and open-source developers in order to understand, design, and develop a 3-D FFT-ECP library that could benefit from these existing optimized FFT kernels or rely on new optimized kernels that are developed under this framework. We will also study ECP application needs and define a suitable modular implementation that provides high-performance software.

Reference: S. Tomov, A. Haidar, D. Schultz, and J. Dongarra, **“Evaluation and Design of FFT for Distributed Accelerated Systems,”** *ECP WBS 2.3.3.09 Milestone Report*, no. FFT-ECP ST-MS-10-1216: Innovative Computing Laboratory, University of Tennessee, October 2018.

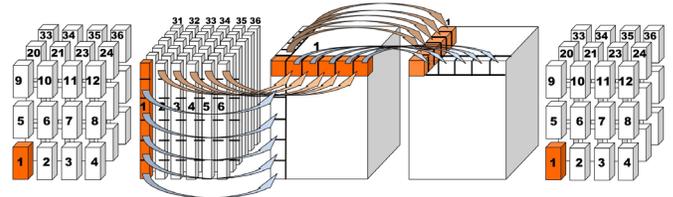
### A 3-D FFT EXECUTION TRACE WITH MAIN FFT COMPONENTS

(80 MPI processes on Intel Xeon E5-2650 v3 cluster, 1K × 1K × 1K grid)



### FFT-ECP FRAMEWORK DESIGN WITH FLEXIBLE API

(Need flexible FFT API for application-specific input and output)



- Main objective is the design and implementation of a sustainable FFT library for exascale platforms;
- FFT-ECP’s goal is to help key ECP applications and provide an efficient and flexible FFT API to take application-specific input and outputs;
- Approach is to: (1) collect and leverage existing FFT capabilities to build a sustainable FFT library (instead of creating a new and independent software stack); (2) optimize data movements and overlap computations with communications; (3) autotuning.

FIND OUT MORE AT



<http://icl.utk.edu/fft>

SPONSORED BY



This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.