

Jefferson Lab LQCD Computing

April 2022 All Hands Meeting

Robert Edwards



NP/LQCD & JLab/LQCD programs

- **Computing** - Computational Sciences & Technology Division (CST) [Amber Boehnlein]
- **Theory** - Computations and Nuclear Theory
 - LQCD@JLab started in ~FY01 - all funds directed through JLab/Theory
 - Theory and CST group have worked closely over 20 years
 - CST division very responsive to needs of LQCD Theory
- **Systems** - based on computational requirements for LQCD Theory and LQCD community
 - Funding directly from NP
 - Program is coordinated with USQCD Exec.Comm and Scientific Program Comm.
 - E.g., allocations taken from SPC
- New NPPLC Initiative - started FY18
 - Reporting annually to Research @ DOE/NP - switched away from Facilities @ DOE/NP
 - Anticipate a significant review in FY23, possibly FY24?

Nuclear and Particle Physics Computing Initiative

- NPPLCI:
 - Robert Edwards, PI
 - Graham Heyes, Deputy
 - Amitoj Singh, JLab LQCD site manager
 - Hardware: Edwards & Singh
- NPPLCI reporting structure
 - Report annually to Research Division of DOE/NP
 - First report (in Feb. 2020) included highlights of scientific program and future plans
 - DOE responds to the reports; however, not under an annual panel review process
 - Discussion of panel review in FY23
- Operations
 - Taking part in annual surveys & twice-weekly facility coordination
- FYI, HEP Program now reports to Research - not facilities
 - No annual review this year

Nuclear and Particle Physics Computing Initiative

- Model: dedicated facility - initiative purchases systems and operates them
 - Allows detailed tailoring of hardware & software development
 - Benefits compared to two-lab IC-based model (less constraints)
 - Amortize overhead costs with large system acquisitions
- \$1M per year - about 50:50 labor:hardware
- FY2018: upgraded Jlab's KNL resources (added to system from FY2016)
- FY2019: upgraded Jlab's GPU resources (gamer-card system)
- FY2021: new AMD CPU + MI-100 GPU system (generation before Frontier)

More details in talk by Amitoj Singh

JLab & software development

- Heavy involvement since 2001 & SciDAC-1, then SciDAC-2 & 3
- Currently, lead institution for ASCR/NP SciDAC-4 project
- Also direct NP portion under Exascale Computing Project

- Leverage software devel & local/commodity resources to efficiently utilize national resources
 - Our community well positioned for leadership systems
 - E.g., early adopters of NVIDIA GPUs and KNLs -> Titan and Cori-KNL

- Exascale systems - now AMD and Intel GPUs
 - Opportunity/necessity to diversify our codes

- LQCD commodity systems ↔ on-ramp to LCF systems + cycles for community

Exascale

- Leadership + Exascale
 - [Perlmutter](#) - AMD CPU + NVIDIA GPU + Cray network - now
 - [Frontier](#) - AMD CPU + AMD GPU + Cray network - close! [Bronson Messer's talk]
 - [Aurora](#) - Intel CPU + Intel GPU + Cray network - soon
- Exascale Computing Project (LQCD is one of ~20 applications)
 - Performance metric - improve application performance by 50x
- LQCD code bases
 - QUDA refactorized - now “backend” support for NVIDIA+AMD+Intel GPUs
 - Grid support for all, but focused more on Intel
 - CPS & MILC - rely on QUDA
- NP & Chroma - refactorized to support more GPU systems, but prioritized AMD
 - [Production ready on AMD+NVIDIA, not quite ready for Intel](#)
 - [Software available on GitHub](#)

New LQCD system acquisitions

- Alternatives Analysis in 2020 - guide near term purchases
 - Focused on GPU systems - complement to existing KNL systems
 - Range of possible applications - require ECC
 - Considered two main metrics to quantify price-performance
 - Wilson dslash - proxy for inverters
 - ZGEMM based contractions considering possible swapping to main memory
 - ECP & SciDAC allowed for comparing these benchmarks across different systems
- Chose AMD CPU + AMD MI-100 GPU
 - Two front-end systems - one of them an evaluation system with MI-50 GPUs
 - Market delays - took quite a while to deploy 21g system
- Future? All USQCD systems?
 - Challenging time for new acquisitions!