Bringing DCA++ (Dynamical Cluster Approximation) to HPC Leadership Class Systems -- Summit and Beyond

Arghya Chatterjee‡♣, Thomas Maier‡, Giovanni Balduzzi*, Vivek Sarkar♣, Oscar Hernandez‡

‡ Oak Ridge National Lab.
{chatterjeea, maierta, oscar}@ornl.gov

♣ Georgia Institute of Technology.
{arghya,vsarkar}@gatech.edu

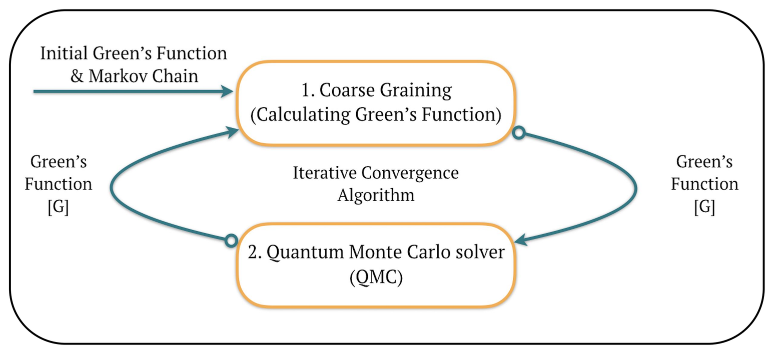
* ETH Zürich
gbalduzz@itp.phys.ethz.ch

1. INTRODUCTION

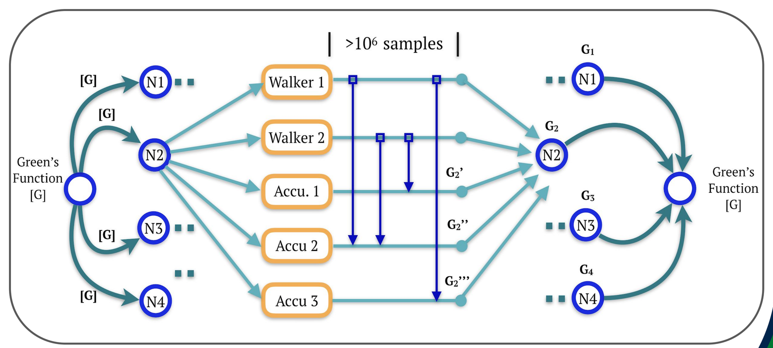
- Pre-exascale and exascale systems :
  — massive amounts of **hierarchical memories**
  — **user-managed caches / DRAM / NVMs**
- Scientific applications — **adapt** to new hardware without compromising **scalability / efficiency**
- Application : **DCA ++** (Dynamical Cluster Approximation)
  — Collaboration between **ORNL** and **ETH Zurich**
  — Recipient of the **Gordon Bell Award** in 2008
- DCA++ today:
  - 16 petaflops of performance on Titan (OLCF)

2. DYNAMICAL CLUSTER APPROXIMATION

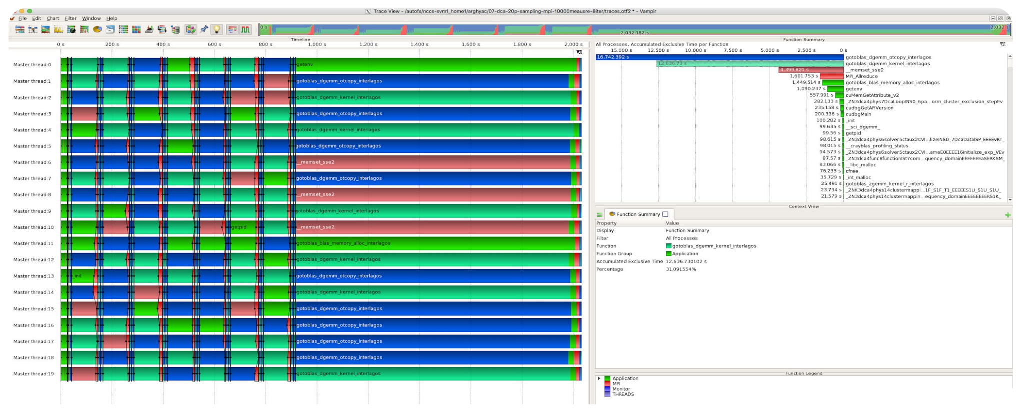
- Numerical simulation tool — predict behaviors of co-related quantum materials **(superconductivity, magnetism)**



- Iterative self consistent algorithm
- Uses 4 programming models:
  - **MPI**
  - **C++ Standard Threads**
  - **NVIDIA CUDA**
  - **BLAS / LAPACK**



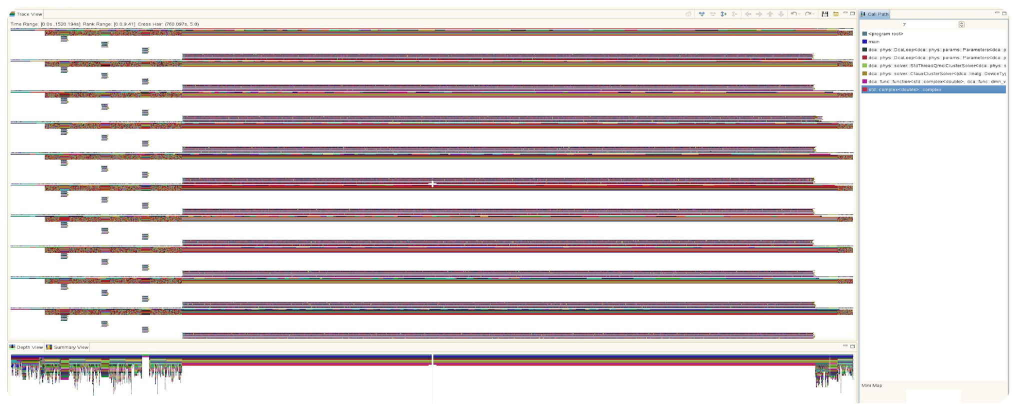
3a. PROFILER: SCORE-P ; VISUALIZER: VAMPIR; MACHINE: TITAN (OLCF)



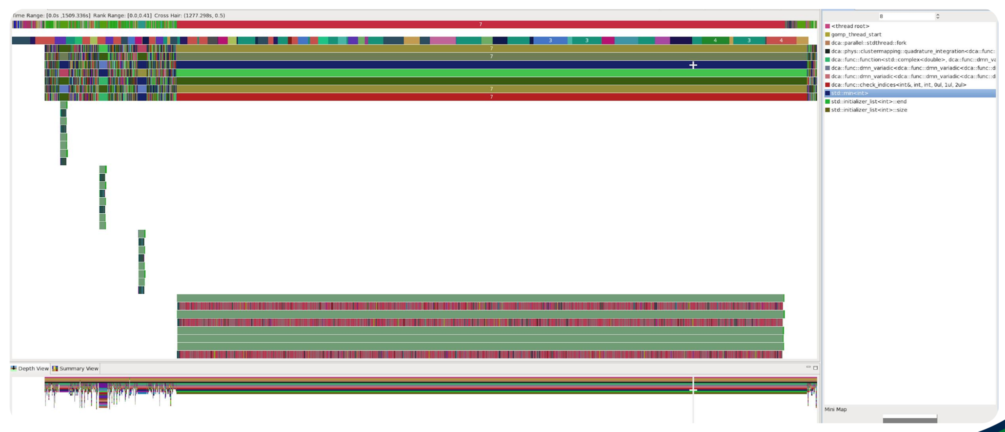
- Full trace (using **synthetic dataset**) - 20 nodes / 20 ranks / 8 iterations
- Final iteration computes **4-point vertex** function
- 41% (dark blue) — **tiling** for DGEMM Kernel
- 31% (teal) — **computing** DGEMM (work performed on GPUs)
- 4-point function needs **massive memory** for storage and computation
  - will be addressed using **DRAM / NVMs** on Summit

- Focussing on **iteration 4** (all iterations have similar workflow)
- **10% (red)** of each iteration — MPI reductions
- **Massive load imbalance** across the nodes (MPI ranks)
  - Task-based programming models might help with imbalance
  - Hardware reductions on Summit

3b. PROFILER: HPCTOOLKIT ; VISUALIZER: HPCTRACEVIEWER; MACHINE: TITAN (OLCF)



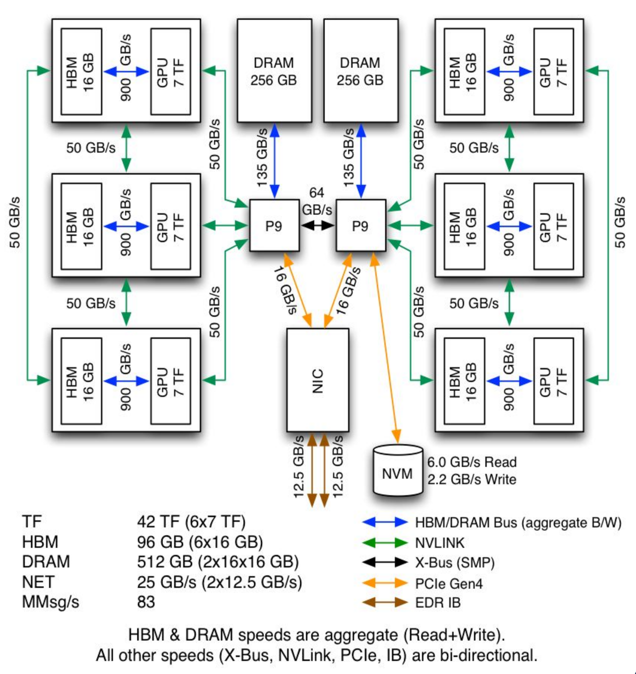
- Full trace (using **synthetic dataset**) - 10 nodes / 10 ranks / 4 iterations
- Y-axis — **MPI ranks** + **OpenMP** + **C++ standard threads**
  X-axis — Execution time **(iterations)**
- Thread level information (**walkers** and **accumulators**)
- Each MPI rank performs **similar** computation
- **OpenMP** — Coarse Graining Function / **Standard threads** -- QMC Solver



- Trace for **1 MPI rank** (shows all 4 iteration steps)
- Each color — procedure call; Each line -- Threads (**OpenMP/C++ std.**)
- White space — C++ std threads **fork / join** (threads **not being reused**)
- 30% of total execution — Mutex locks
  - **Optimal work balance** :: walkers & accumulators
  - **More** than 1 queue for accumulator workers threads

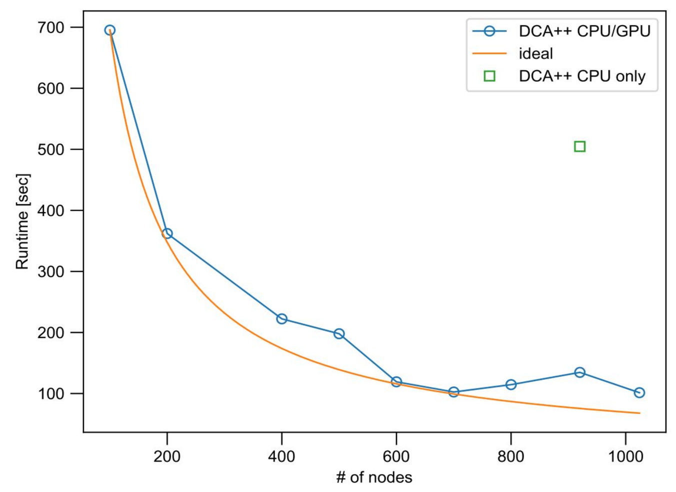
4. SUMMIT AND BEYOND

- World's **fastest** supercomputer
- No of nodes: ~**4600**
- Per node:
  - **2 IBM Power 9** CPUs
  - **6 NVIDIA Tesla V100**s
- Memory:
  - **800 GB** DDR4 [Power 9's]
  - **96 GB** High Bandwidth Memory (HBMs) [GPUs]
  - **1.6 TB** Non-volatile Memory (NVMs) [Burst Memory]
  - **Unified Memory** [eliminates need for explicit data transfer]



5. SCALABILITY ON SUMMIT

- Low temperature simulation of a single-band Hubbard model
  - Coulomb Repulsion $U/t = 4$
  - DCA Cluster Size (N_C) = 4
- Representative of **production** runs:
  - Over **10,000,000** measurements of Green's function
  - **Full 4-point vertex** function



- **1 MPI rank / GPU** ; **12 hardware** threads running on **6 CPU** Cores (hyperthreading; sharing GPU on separate streams)

6. CHALLENGES AND IDEAS

- Improve data movement between host/device:
  - **Unified memory using NVLINK2** (Challenge: prefetching)
- All computation on GPUs; **CPUs** only for communication & I/O (Programming model to **reverse offload** -- device to host)
- Ways to **exploit multiple GPUs** (3 per MPI rank on Summit)
- Reductions on the GPUs / across nodes (using **hardware / non-blocking collectives**)
- **Exploit NVRAM** -- maximize use of on-node memory
- Need for **hybrid asynchronous programming models**:
  - **HPX** [tasking modules reduce load imbalances]
  - **Kokkos** [performance portable / hardware agnostic]

7. ONGOING WORK AND FUTURE DEVELOPMENT

- Exploiting more intra node parallelism on Summit
- Exploring programming models:
  - Support for multiple accelerators
  - Tap into massive hierarchical memories
  - Task based / performance portable
- Improved calculation of dynamical properties:
  - Provide tests of the simplified models to explain real materials
- Develop a Domain Specific Language (DSL) to mimic QMC solver workflow

8. ACKNOWLEDGEMENTS

- *This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.*
- *This work was supported by the Scientific Discovery through Advanced Computing (sciDAC) program funded by U.S. DOE, Office of Science, Advanced Computing Scientific Computing Research (ASCR) and Basic Energy Sciences (BES), Division of Materials Science and Engineering.*
- *We would like to thank -- Ronny Brendel (Score-P); Dr. John Mellor-Crummey and Dr. Laksono Adhianto (HPCToolkit).*