Overview of Research and Developments in Scalable Solvers Group

Sherry Li, xsli@lbl.gov

Computing Sciences Summer Program

6/27/2023

Staff Members

Xiaoye Sherry (Sherry) Li
Senior Scientist & Group Lead
+1 510 486 6684 | XSLi@lbl.gov

Mark F Adams
Staff Scientist
MFAdams@lbl.gov

Pieter Ghysels
Research Scientist
+1 510-486-5594 | PGhysels@lbl.gov

Osni Marques
Staff Scientist
+1 510 486 5290 | OAMarques@lbl.gov

Yang Liu
Research Scientist
510-486-5283 | liuyangzhuan@lbl.gov

Roel Van Beeumen
Research Scientist
+1 (510) 495-2189 | RVanBeeumen@lbl.gov

Chao Yang
Senior Scientist
+1 510 486 6424 | CYang@lbl.gov

Postdoctoral Researchers

Wajih Boukaram
Postdoctoral Fellow
+1 (510) 486-6684 | wajih.boukaram@lbl.gov

Siva Darbha
sdarbha@lbl.gov

Hengrui Luo
Postdoctoral Fellow
hrluo@lbl.gov

Senwei Liang
SenweiLiang@lbl.gov

Osman Malik
Alvarez Postdoctoral Scholar

Mustafa Rahman
Postdoctoral Fellow
+1 (510) 274 1447 | rahman@lbl.gov

Yizhi Shen
Postdoctoral Fellow
+1 510.486.6684 | yizhis@lbl.gov

Tianyi Shi
tianyishi@lbl.gov

Jia Yin
Postdoctoral Scholar
jiayin@lbl.gov

Yuanran Zhu
yzhu4@lbl.gov

Faculty Scientsts

Zhaojun Bai
Faculty Scientist, UC Davis
+1 510 495 2851 | zbai@ucdavis.edu

James Demmel
Faculty Scientist, UC Berkeley
+1 510 495 2851 | demmel@berkeley.edu

John Gilbert
Faculty Scientist, UC Santa Barbara
+1 510 495 2851 | gilbert@cs.ucsb.edu

21 Summer students/visitors!

Mission of Scalable Solvers Group (SSG)

https://crd.lbl.gov/divisions/amcr/applied-mathematics-dept/scalable-solvers/

The group develops fast, parallel algorithms and software for solving the linear and eigenvalue algebraic systems, and deliver the solvers tools to the broad community through libraries and collaboration with domain scientists.

Algebraic solvers are fundamental tools

Black-box solvers

Purely algebraic, matrix input

$Ax = b,\ \ Ax = \lambda x$

Application-specific linear algebra tools

Specialized to accelerator, chemistry, fusion, materials, ML, nuclear physics, quantum comput., transportation, . . .

Improve algorithmic efficiency, parallelism, and solution quality

- Multilevel, multigrid, hierarchical algorithms
- Reduce communication / synchronization
- Increase concurrency
- Improve convergence
- HPC-aware: GPUs, …

Expertise, capabilities

(Most software packages are open source, BSD License)

https://crd.lbl.gov/divisions/amcr/applied-mathematics-dept/scalable-solvers/software/

- **Dense linear algebra** (LAPACK/ScaLAPACK, ButterflyPACK)
- **Sparse linear solvers**
  - Direct solvers (STRUMPACK, SuperLU, symPACK)
  - Multigrid (GAMG in PETSc)
  - Algebraic preconditioner (STRUMPACK)
  - Hybrid solver (PDSLin)
- **Eigenvalue calculations**
  - Lanczos / Arnoldi iterative eigensolver (BLZPACK, PARPACK)
  - Non-Hermitian eigensolver for interior eigenvalues (software: GPLHR)
  - Application-specific structured eigensolvers
    Electronic structure, quantum chemistry, nuclear physics (PEXSI, BSEPAC, SpectrumSlicing)
    Linear, nonlinear, parameterized eigenvalue problems
- **Multi-linear algebra (tensor) (**FunFact**)**
- **High-precision floating-point arithmetic** (QD, ARPREC, XBLAS)
- **Machine learning for sciences** (GAP, GPTune)
- **Quantum computing algorithms** (FABLE, F3C, QCLAB)

Linear Solvers

Research and development in fast solvers

Linear solvers, eigensolvers, preconditioners, …

"Fast" == asymptotically lower arithmetic and/or communication

Research themes:

- Hiding/avoiding communication/synchronization
  - Latency / bandwidth / flops / memory
- Randomized algorithms
  - sampling, sketching, projection, dimension reduction
- Low-rank approximations
  - Exploit localization
- Hierarchical & multilevel methods
  - Multigrid, FFT, FMM, $\mathcal{H}^2$/HSS matrices, Butterfly

Data-sparse approximation via low-rank compression

Same mathematical foundation as FMM [Greengard-Rokhlin'87],

put in matrix form:

- Diagonal block ("near field") exact
- Off-diagonal block ("far field") approximate

FMM

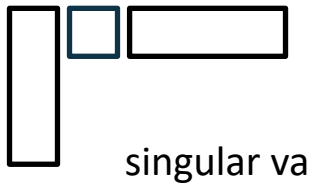
separability of Green's function

$$G(x,y) \approx \sum_{l=1}^{r} f_l(x) g_l(y), \quad x \in X, y \in Y$$

Algebraic

low-rankness off-diagonal

$$A \approx \begin{bmatrix} D_1 & U_1 B_1 V_2^T \\ U_2 B_2 V_1^T & D_2 \end{bmatrix}$$



singular value decomposition (SVD)

Sketching to find good bases
randomized projection

Approximate range of A:

1. Pick random matrix $\Omega_{nx(k+p),}$ k target rank, p small, e.g. 10
2. Sample matrix S = A Ω (tall-skinny)
3. Compute Q = ON-basis(S) via rank-revealing QR

Then, $A \approx QQ^*A$

Benefits: only need matvec, "matrix-free"

Faster sketching

Several choices of Ω rather than dense Gaussian matrix

- Example: Jonhson-Lindenstrauss transform (JLT)

DEFINITION 2.3 (JL Sketching Operator). *Suppose $\mathcal{D}$ is a distribution over matrices of size $d \times n$. We say that a matrix $R \sim \mathcal{D}$ is a $(n, d, \delta, \varepsilon)$-JL sketching operator if for any vector $x \in \mathbb{R}^n$ it satisfies*

$$\Pr_{R \sim \mathcal{D}} \left[\left| \|Rx\|^2 - \|x\|^2 \right| > \varepsilon \|x\|^2 \right] < \delta.$$

- Serve as dimensionality reduction (subspace embedding)
- Sparse JLT: each column in R has $\alpha < d$ nonzeros

  The nonzero entries are drawn independently from a scaled Rademacher distribution, taking values in $\left\{ \frac{1}{\sqrt{\alpha}}, -\frac{1}{\sqrt{\alpha}} \right\}$ with equal probability

Construction of Hierarchically Semi-Separable Matrix Representations Using Adaptive Johnson-Lindenstrauss Sketching

Yotam Yaniv (NSF-MSGI), Osman Malik, Pieter Ghysels, Sherry Li

Scientific Achievement

Implemented adaptive randomized Hierarchically Semi-Separable (HSS) matrix construction using the Sparse Johnson-Lindenstrauss Transform (SJLT), which is significantly faster than Gaussian sketching, and we provide theoretical justifications for this extension.

Significance and Impact

HSS and other hierarchical or data-sparse matrix representations are widely used to represent large dense matrices from various applications such as boundary element methods, ML kernel methods, etc.

Research Details

- **Extended concentration bounds to all JL sketching operators: Gaussian, Subsampled Randomized Hadamard Transform (SRHT) and SJLT**
- **Implemented of SJLT in C++ STRUMPACK library, up to 2.5x speedup for HSS construction over Gaussian sampling, with comparable accuracy and rank pattern**

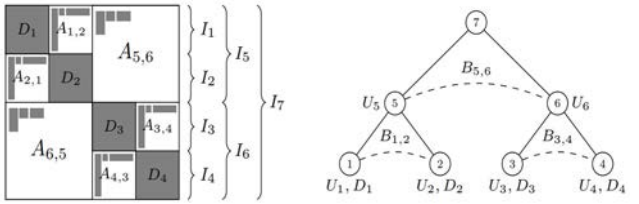
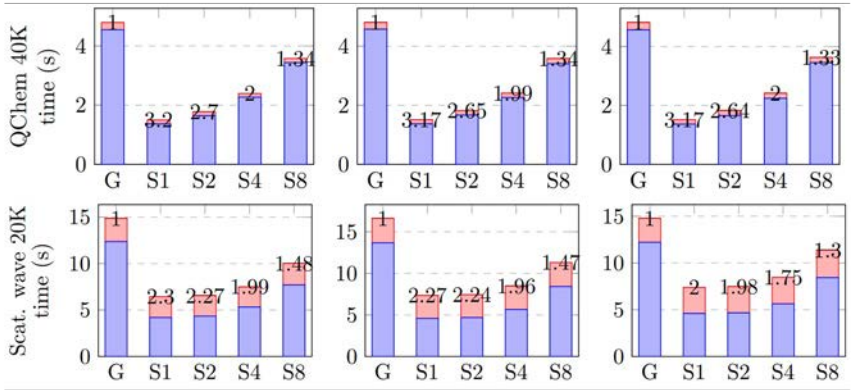


Illustration of a symmetric HSS matrix with 3 levels. Diagonal blocks are partitioned recursively. Gray blocks denote the basis matrices. Right: Tree representation of the HSS matrix.



HSS construction time and random sketching time for a 1D kinetic energy quantum chemistry problem, and for an impedance matrix describing a scattering wave. G refers to Gaussian, S(α) to SJLT with α nonzeros per row. Overall speedup over Gaussian sketching is shown at the top of each bar.

Y. Yaniv, O.A. Malik, P. Ghysels, X.S. Li. "Construction of Hierarchically Semi-Separable Matrix Representation using Adaptive Johnson-Lindenstrauss Sketching". arxiv.org:2302.01977

U.S. DEPARTMENT OF ENERGY | Office of Science
UCLA
BERKELEY LAB

Advances in numerical linear algebra improves ML

Example: Kernel Ridge Regression in ML

Ridge regression + Kernel trick

- Training:

  Minimize cost function: $argmin_w\, C(w) = \sum_i (y_i - w^T x_i)^2 + \lambda \|w\|^2$

  x_i data points, y_i labels

  w is a vector normal to the target hyperplane

  Optimal weights:

  $$w = X^T(\lambda\, I + XX^T)^{-1}\mathrm{y}\,, \qquad X^{n\times d} \text{ a matrix of training data}$$

- Prediction:

  Given a test vector x_1, compute:

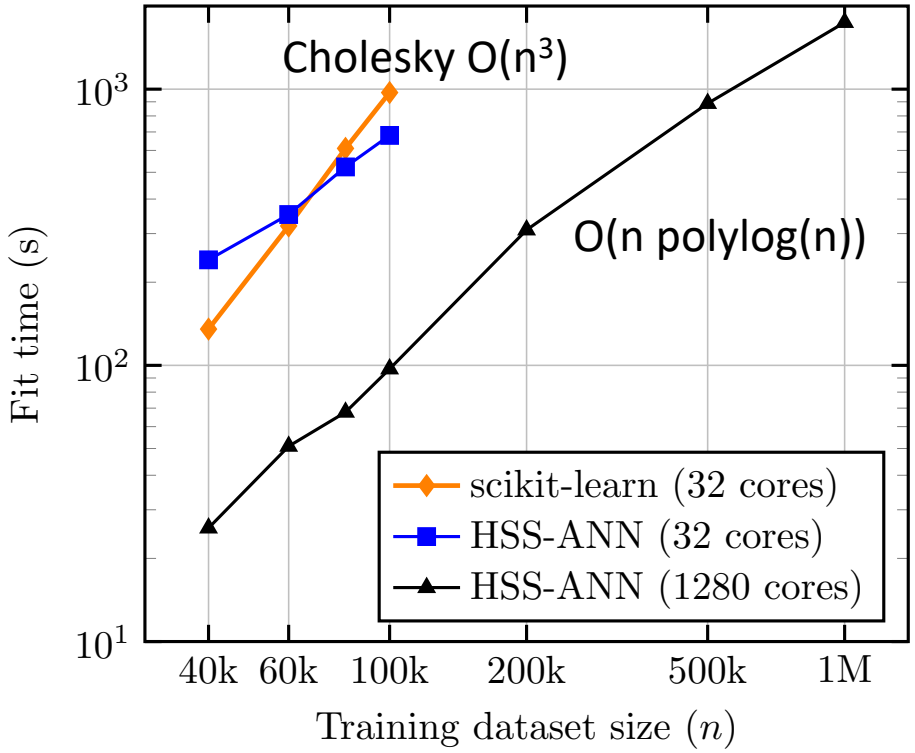
  $$y_1 \coloneqq w^T x_1 = [(\lambda\, I + XX^T)^{-1}\mathrm{y}]^T X x_1$$
  $$\approx [(\lambda\, I + \mathcal{K}(X, X)\,)^{-1}\mathrm{y}]^T \cdot \mathcal{K}(X, x_1) \quad \longleftarrow \text{kernel trick}$$

- Binary classifier: class label predicted by the sign of y_1

Comparison between scikit-learn and HSS-ANN

Scikit-learn only provides shared-memory parallelism

HSS in STRUMPACK works on distributed memory



SUSY dataset fom UCI

STRUMPACK to scikit-learn Python interface

- Scikit-learn: ML in Python, http://scikit-learn.org/stable/
  - classifiers and regressors

- STRUMPACK Python interface class: STRUMPACKKernel
  - derives from scikit-learn base classes BaseEstimator and ClassifierMixin
  - implements member functions: fit, predict and decision_function
  - can be used for multi-class classification through scikit-learn One-Vs-One or One-Vs-All estimators

Fast GPU Solvers for Many Small Systems with PETSc

HBPS FES Partnership

Contact: Mark Adams

Scientific Achievement

Many applications have many small linear systems of equations to be solved concurrently. Modern hardware can consume the mass parallelism available in these applications, but new techniques are required to exploit this parallelism. This project has solved this problem for one application in plasma physics; the solver has been deployed in the PETSc numerical library.

Significance and Impact

Fast small system solvers will allow high dimensional problems with tensor structure (eg, 3D ⊗ X), as well as other applications such as UQ and domain decomposition smoothers for global multigrid solvers, to continue to use emerging hardware effectively.

Technical Approach

A custom implementation of standard Krylov solvers was developed that is designed to use accelerators effectively with vectorization; it was written in Kokkos for performance portability. Solvers were developed in the PETSc numerical framework and deployed in PETSc for dissemination to the broader scientific computing community

N=647K	Total (sec)	Solver	Vec ops
Batch	5.9	1.6	0.2
Ensemble	14.1	7.0	(5.9)

Timing for all-GPU fully implicit evolution of Fokker-Planck collision operator on benchmark problem3. Total solve time (sec) includes cost of GPU Jacobian matrix creation within a Newton nonlinear solver, that uses the new batch linear solvers for each species on each "vertex" in a harness code (example in PETSc release). New batch solvers are comparted with ensemble solvers where all systems are stacked into a single large linear system

PI: Mark F. Adams, Berkeley Lab
Collaborating Institutions: University of Buffalo
ASCR Program: FES Partnership and FASTMath
ASCR PM: FES John Mandrekas; ASCR: Randall Laviolette
Publication: M. F. Adams, P. Wang, and M. G. Knepley, "A performance portable, fully implicit landau collision operator with batched linear solvers" IPDPS 2022, https://doi.org/10.48550/arXiv.2209.03228
Code Developed: Released in PETSc v3.19

PETSc TAO

BERKELEY LAB

U.S. DEPARTMENT OF ENERGY | Office of Science

Eigen Solvers

Scaling Eigenvalue Solvers on GPU-Based Supercomputers

Contact: Osni Marques

Scientific Achievement

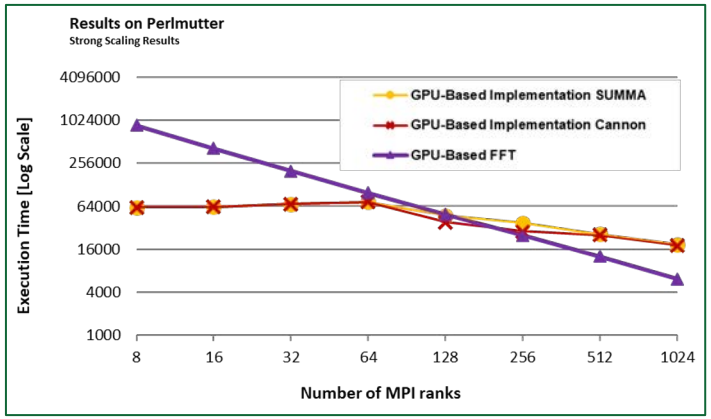
The pursuit of better parallel scaling of iterative eigenvalue solvers on GPU-based supercomputers has exposed a need for counterintuitive rearrangements of data layouts and linear algebra computations.

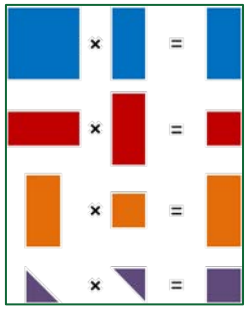
Significance and Impact

- Eigenvalue solvers are an essential part of many scientific applications, including materials and chemistry (Schrödinger equation), often taking a significant share of the workload in DOE computer facilities.
- The system sizes and physical phenomena that can be studied are often constrained by the parallel scaling of the solvers and codes.
- The scaling of iterative eigenvalue solvers is usually limited by a reorthogonalization step that needs to be applied to the iterate vectors.

Technical Approach

- This research revisited unconstrained minimizations techniques – more complex than standard techniques but without a reorthogonalization step – as an alternative for eigenvalue solvers.
- Recent experiments have revealed a need for going beyond kernel implementations provided by well-established libraries .





Linear algebra computations in eigenvalue solvers, e.g., Jacobi Davidson, PPCG, Conjugate Gradient Minimization, and Unconstrained Conjugate Gradient Minimization. While computations involving a square matrix and FFTs scale well, others may need more specialized implementations.

PI(s)/Facility Lead(s): Osni Marques, Berkeley Lab. Collaborators: D. T. Popovici, M. del Ben, and A. Canning.
ASCR Program: SciDAC
ASCR PM: Ceren Susut-Bennett
Publication(s) for this work: M. Del Ben, O. Marques, and A. Canning, "Improved Unconstrained Energy Functional Method for Eigensolvers in Electronic Structure Calculations", *ICPP2019*. Recent results presented at SIAM CSE 2023.

Diffusion Map for Collective Variable Identification

With the BES SciDAC Partnership "A chemistry based, data science enabled and high performance powered predictive framework to control the decomposition of polymer mixtures"

Contact: Chao Yang

Scientific Achievement

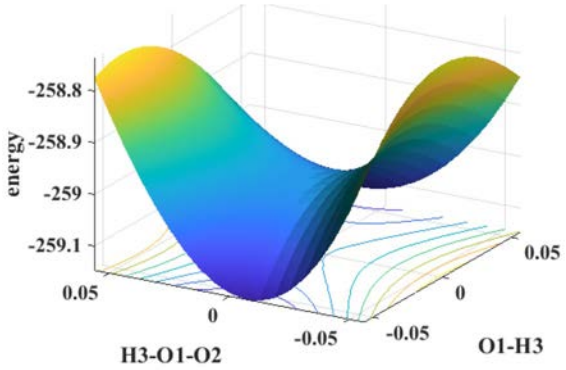
Diffusion coordinates obtained from a local diffusion map (DM), constructed from samples along an *ab initio* molecular dynamics (AIMD) trajectory within a meta-stable region of the potential energy surface of a molecular system, can identify good collection variables (CVs) that can be used to describe the main reaction mechanism. In addition, a global diffusion map can be used to perform a committor analysis of the free energy surface.

Significance and Impact

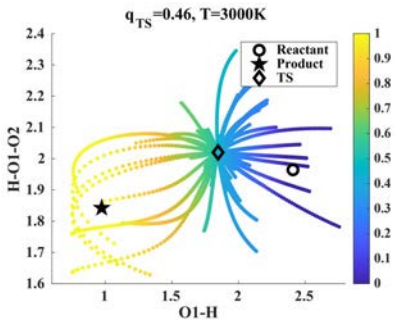
The proposed methodology enables scientists to identify effective CVs for hydrogen combustion systems that are paramount for characterizing the primary reaction pathway in a reduced dimensional space or manifold.

Technical Approach

- Construct a diffusion kernel from samples along an AIMD trajectory and obtain diffusion coordinates by computing the eigenpairs of the DM matrix.
- Compute correlation coefficients between diffusion coordinates and CV candidates(e.g., internal coordinates, principal components).
- Use DM to solve a backward Kolmogorov equation to obtain a committor function.



The potential energy surface in two internal coordinate based CVs near the transition state of the substitution reaction $H_2O_2 + H \rightarrow H_2O + OH$



The committor function obtained from the diffusion map constructed from AIMD trajectories of the reaction $H + O2 \rightarrow HO_2$

PI(s): Chao Yang, LBNL
Collaborating Institutions: UC Berkeley (Teresa Head-Gordon)
ASCR Program: BES SciDAC Partnership
ASCR PM: Lali Chatterjee
Publication: T. Ko, J. Heindel, X.Guan,T. Head-Gordon, D. Williams-Young and C. Yang, "Using Diffusion Maps to Analyze Reaction Dynamics for a Hydrogen Combustion Benchmark Dataset," arXiv:2304.09296

Quantum Computing Algorithms

Quantum Fourier Transform Revisited

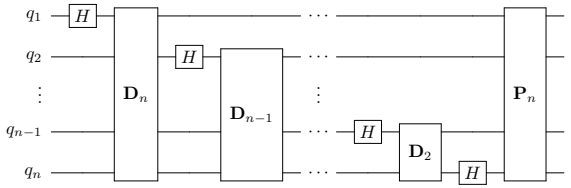
2^n input length
Classical FFT: $O(2^n n)$
Quantum gates: $O(n^2)$

Scientific Achievement

Deriving the quantum Fourier transform (QFT) from the fast Fourier transform (FFT)

Significance and Impact

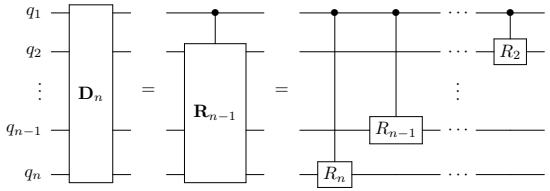
Proves the linear algebra relation between FFT and QFT with little knowledge of quantum computing and by only using elementary properties of Kronecker products of matrices.



Quantum circuit of the DFT matrix decomposition

Research Details

- FFT algorithm can be derived as a particular matrix decomposition of the discrete Fourier transformation (DFT) matrix
- QFT algorithm can be derived by further decomposing the diagonal factors in the FFT decomposition into products of matrices with Kronecker product structure
- QFT decomposition of the DFT matrix and the corresponding quantum circuit is not unique
- Extended the radix-2 QFT decomposition to a radix-*d* QFT decomposition



Quantum circuit representation of the decomposition of the diagonal matrices $\mathbf{D}_n$

D. Camps, R. Van Beeumen, and C. Yang
Quantum Fourier Transform Revisited
Numerical Linear Algebra Appl., 2021.

LDRD

PI: Roel Van Beeumen (LBNL)

QCLAB++: Simulating Quantum Circuits on GPUs

Scientific Achievement

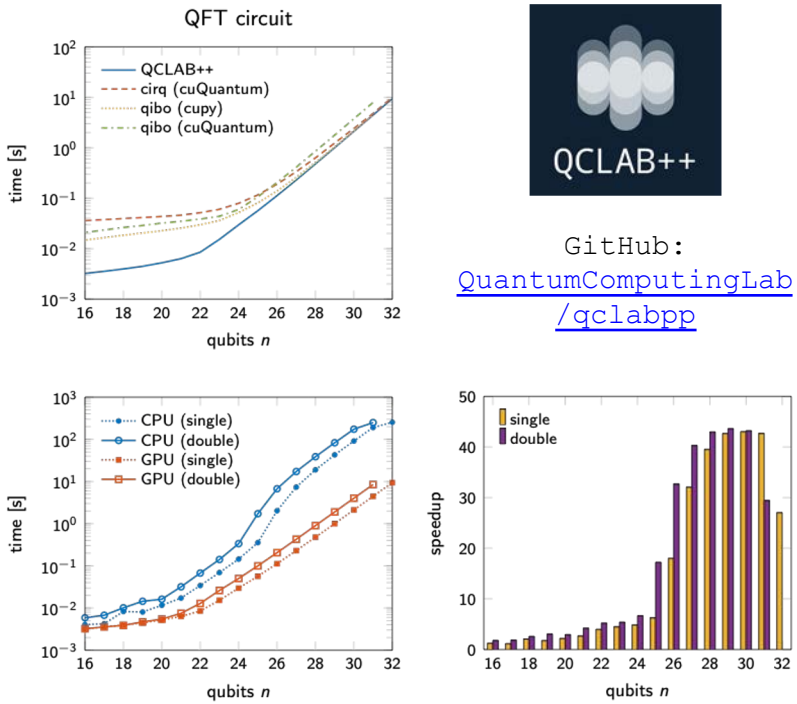
QCLAB++ is a light-weight, fully templated C++ package for GPU-accelerated quantum circuit simulations. The code offers a high degree of portability, as it has no external dependencies and the GPU kernels are generated through OpenMP offloading.

Significance and Impact

QCLAB++ is designed for performance and numerical stability through highly optimized gate simulation. The GPU kernels generated by OpenMP can yield speedup factors of more than 40x, hence enabling efficient quantum circuit simulations up to 32 qubits.

Technical Approach

- Efficient gate simulation algorithms for 1-qubit and 2-qubit gates: a single for loop combined with bit operations for index calculations.
- Portable state vector simulator with GPU kernels generated by OpenMP.
- Benchmarks conducted on NERSC's Perlmutter system illustrate its competitiveness to other circuit simulation packages.



QCLAB++

GitHub: QuantumComputingLab/qclabpp

QCLAB++: CPU versus GPU for QFT circuit (Perlmutter - NVIDIA A100 GPU): GPU kernels exhibit a perfect linear scaling on the loglog plot for systems with more than 22 qubits. The CPU simulation exhibits less regular scaling in the timings due to memory access effects, yielding speedup factors of more than 40x.

PI: Roel Van Beeumen (Berkeley Lab)
ASCR Program: NERSC QIS@Perlmutter
ASCR PM: Dr. Thomas Wong
Publication for this work: R. Van Beeumen, D. Camps, N. Mehta, "QCLAB++: Simulating quantum circuits on GPUs," *arXiv:2303.00123* (2023), doi:10.48550/arXiv.2303.00123.
Code Developed: https://github.com/QuantumComputingLab/qclabpp

AI/ML Methods

Machine Learning for Space Weather Mitigation

An ASCR – Office of Electricity Pilot Project

Contact: Pieter Ghysels

Scientific Achievement

A heterogeneous graph neural network (GNN) was developed to predict optimal load in the maximum loadability problem; the team is also working toward optimal blocker placement.

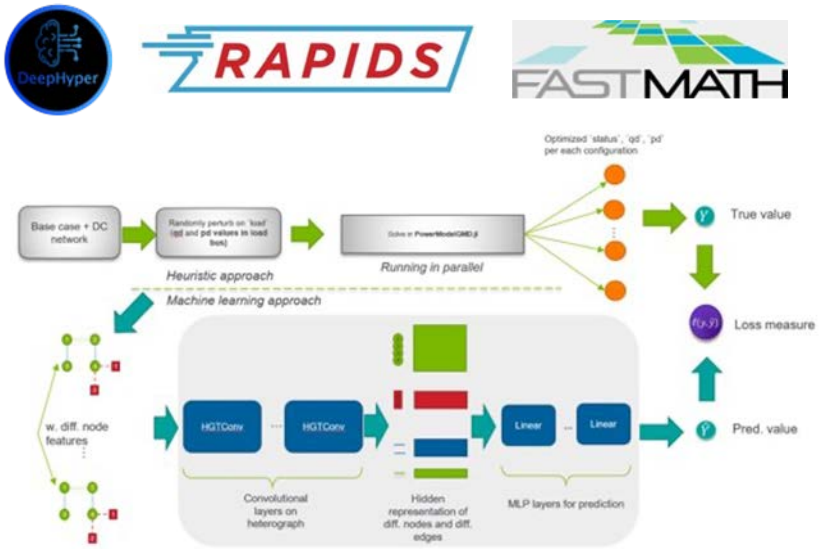
Significance and Impact

Geomagnetic disturbances resulting from intense solar activity caused by coronal mass ejections pose risks to the electrical grid by generating geomagnetically induced currents (GICs). Optimal blocker placement will significantly reduce the impact of this problem. This GNN-based machine learning prediction approach results in high-quality solutions found in shorter computation time than traditional solvers (e.g., Julia optimizers such as PowerModelsGMD.jl).

Technical Approach

- Modeled AC and DC networks with various node and edge features.
- GNN predictive models for maximum loadability regression and for classification for GIC blocker placement.
- Found that training for the loadability problem on 300 perturbed input graphs is faster than heuristic solver on a single graph.
- Used DeepHyper to optimize hyperparameters and improve accuracy.

PI: Pieter Ghysels, Berkeley Lab
Collaborating Institutions: Office of Electricity, LANL, ORNL, UC Berkeley, ANL
ASCR Program: RAPIDS/FASTMath
ASCR PM: Randall Laviolette, Ceren Susut-Bennet, Lali Chatterjee



- *The heuristic approach involves generating random perturbations on a given power grid and feeding those perturbations into one of the optimizers in the PowerModelsGMD.jl package, which finds the optimal real ("pd") and reactive ("qd") power demands.*
- *The machine learning approach on the bottom involves feeding the same power grid data into the heterogeneous GNN, passing through several convolutional and Multi-Layer Perception (MLP) layers.*
- *The true values from the optimizer and the predicted values from the machine learning model are then used to compute the loss to evaluate the model's performance.*

GPTune autotuner: Bayesian optimization with Gaussian Process surrogate modeling

Younghyun Cho, Jim Demmel, Sherry Li, Yang Liu, Henrui Luo

Scientific Achievement

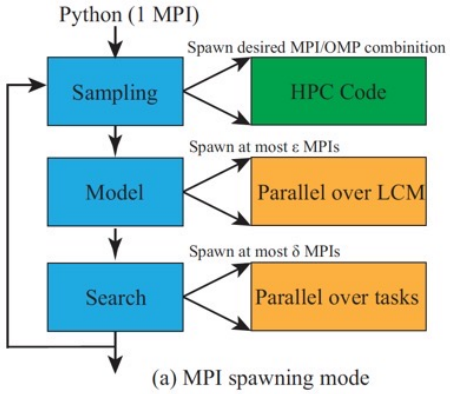
- **Optimization :** $\min_x y(t,x)$, x : parameter configuration
- **Applicable to any black-box software**

Significance and Impact

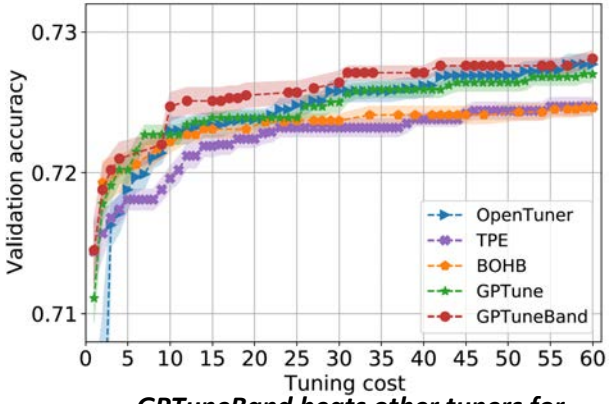
Gaussian process (GP) models can act as surrogates for code performance or first-principle physics for many expensive SciDAC and ECP applications. Our work leverages multi-task and multi-fidelity GP models to allow accurate surrogates.

Research Details

- Features: multi-task, multi-objective, and multi-fidelity
- Added multi-objective tuning features to allow memory/time tradeoff
- Supported multi-task and transfer learning features to leverage correlation between tuning tasks to improve model accuracy
- History database for crowd-tuning
- GPTune has been applied to Hypre, MFEM, STRUMPACK, SuperLU_DIST, PLASMA, SLATE, ScaLAPACK, NIMROD, M3D-C1, IMPACT-Z, CNN, GCN, kernel ridge regression, sketching-based linear square solvers.



(a) MPI spawning mode



GPTuneBand beats other tuners for tuning GCN on the Citeseer dataset

Y. Cho, J. W. Demmel, X. S. Li, Y. Liu, and H. Luo, IEEE MCSoC, 2021
X. Zhu, Y. Liu, P. Ghysels, D. Bindal, and X. S. Li, SIAM PP, 2022
H. Luo, J.W. Demmel, Y. Cho, X. S. Li, and Y. Liu, JMLR, submitted

U.S. DEPARTMENT OF ENERGY | Office of Science | FASTMATH | ECP EXASCALE COMPUTING PROJECT | BERKELEY LAB

THANK YOU

xkcd.com


THIS IS YOUR MACHINE LEARNING SYSTEM?
YUP! YOU POUR THE DATA INTO THIS BIG PILE OF LINEAR ALGEBRA, THEN COLLECT THE ANSWERS ON THE OTHER SIDE.
WHAT IF THE ANSWERS ARE WRONG?
JUST STIR THE PILE UNTIL THEY START LOOKING RIGHT.
DATA
ANSWERS