DOE EXPRESS Quantum Computing Application Teams

Software Stack and Algorithms for Automating Quantum-Classical Computing (QCOR)
2020 Project Report

PI: Eugene Dumitrescu (ORNL)

Introduction

The QCOR project consists of 4 main research tasks: 1) scientific applications and algorithm development for machine learning (ML) and quantum theories of condensed matter systems and lattice gauge theories; 2) high-level quantum-classical (HLQC) language and domain specific language (DSL) development; 3) compiler and interfaces development for the HLQC language; and 4) hardware-level control, optimization, and algorithms libraries. In the succeeding sections we report on progress towards our research tasks during the 2020 calendar year.

Task 1: Quantum Algorithms and Applications

Quantum State Preparation. Fault-tolerant computers will have a hierarchy of faster and slower operations, just as large classical computers do now. In many schemes, Clifford operations are much faster than non-Clifford operations. At the same time, non-Clifford operations have an exponential cost in classical simulations. The non-Clifford resource in comes under the broad heading of magic (T-gates) and from that definition we derive a related quantity called mana. The mana of a state estimates the non-Clifford operations required to prepare that state, i.e. the state's computational complexity (quantum or classical).

We first computed mana for the ground state of the $\mathbb{Z}_3$ Potts model, a generic interacting field theory displaying a second-order phase transition. We found that the mana is peaked at the critical point. Therefore the computational difficulty will be worst near the phase transition, where perturbative approaches break down. This behavior results from the underlying correlation structure of the state which is generic across field theories. We therefore expect it to generalize to essentially all field theories with critical points. At the same time, our results constrain the physics giving rise to the state's generic correlation structure; this illustrates the interplay of quantum information, condensed-matter physics, and computational physics.

[0]This manuscript has been authored by UT-Battelle, LLC, under Contract No. DE-AC0500OR22725 with the U.S. Department of Energy.

We also computed mana for Haar-random states and found that the average mana is extensive in the number of qudits in multi-qudit system. Haar-random states are a key ingredient in simple minimal models for many physical processes involving thermalization. Quantifying the computational difficulty of preparing Haar-random states therefore sheds light on the requirements for simulating those models. This calculation also serves as an important starting point for more detailed investigations of thermalization, quantum hydrodynamics, black holes, and other systems heuristically modeled by random states.

Quantum-inspired Classical Algorithms. We have devised a quantum Monte Carlo algorithm for the simulation of general quantum and classical many-body models within a single unifying framework. The algorithm utilizes a power series expansion of the quantum partition function in terms of its off-diagonal components. As such, the expansion is both parameter-free and Trotter error-free. In our approach, the quantum dimension consists of the number of products of elements of a permutation group. As such, it allows for the study of a very wide variety of models on an equal footing. To demonstrate the utility of our technique, we use it to clarify the emergence of the sign problem in the simulations of non-stoquastic physical models. This new algorithm simultaneously extends the capabilities for classical simulations and sheds light on the classical-quantum complexity divide.

Quantum Simulation of Many-body Systems. We proposed an efficient quantum algorithm for simulating the dynamics of general Hamiltonian systems. Our technique is based on a power series expansion of the time-evolution operator in its off-diagonal terms. The expansion decouples the dynamics due to the diagonal component of the Hamiltonian from the dynamics generated by its off-diagonal part, which we encode using the linear combination of unitaries technique. Our method has an optimal dependence on the desired precision and, as we illustrate, generally requires considerably fewer resources than the current state-of-the-art.

Quantum Machine Learning Algorithms. In order to characterize the representational power and error resilience of quantum machine learning (QML) we have studied the use of QML for forecasting time-series data. This project focused on reproducing a highly non-linear and memory intensive signal that cannot be expressed by any known deterministic equation. In terms of novel machine learning techniques, the work investigated the use of a NISQ reservoir as a computation model. Our work showed how to experimentally evaluate the memory capacity of various reservoir topologies (using IBM-Q's Rochester device) to identify the configuration with maximum memory capacity. Once the optimal design was selected, the time-series forecast is produced by a linear combination of the average spin of a 6-qubit quantum register trained using real world data. Our results show a remarkable ability to predict the highly non-linear and complex functions using today's NISQ processors.

Digital quantum-simulation verification for lattice gauge theories: Based on the loop-string-hadron (LSH) framework for strongly-coupled gauge field theories, we analytically showed how to exactly fix gauge invariance (or Gauss's law). Following up in 2020 we performed the digitization of this particular framework along with systematic truncation of infinite bosonic Hilbert space and presented its detailed qubit-cost analysis. In doing so we have constructed the first digital "physicality oracle" for systems of arbitrary dimensionality. This result constitutes a significant advance toward error detection and mitigation in gauge-theory simulations.

Identification of optimal Hamiltonian simulation frameworks for non-Abelian gauge theories: Given the wealth of formulations of lattice gauge theories in the Hamiltonian language and their relative pros and cons, our team took an initiative to study and carefully analyze each frame-

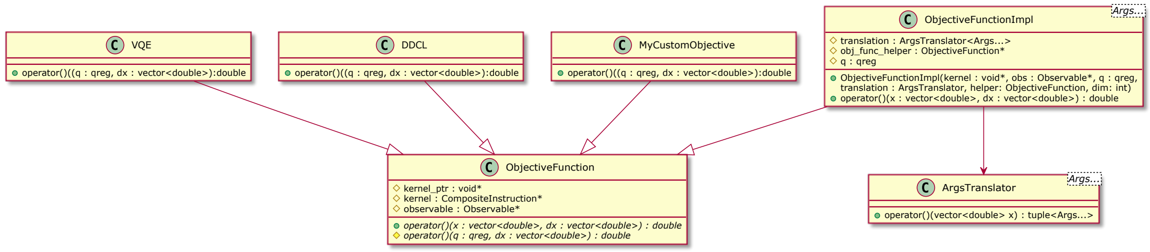

Figure 1: UML diagram outlining class structure for *ObjectiveFunction* abstract data type and related *Optimizer* type. These abstract data types allow the user to implement concrete classes following the pattern in the abstract data type.

work in a non-Abelian theory consisting of a SU(2) gauge field coupled to matter in 1+1 dimensions. In particular, we derived the computational cost of Hamiltonian simulation which consists of Hilbert-space construction, Hamiltonian matrix generation, and observable computation using conventional classical algorithms. Our work establishes that the fully-gauge invariant formulation introduced above, i.e. the LSH formulation, outperforms the angular-momentum formulation and the fully bosonic formulation. We further showed that the LSH formulation is superior to the fully fermionic formulation given its generalizability to higher dimensions. We are now establishing the extent to which the benefits of the LSH formulation transfer to quantum algorithms.

Task 2: HLQC and Domain Specific Languages

The languages team has undertaken several efforts to formalize the QCOR language extension which expresses domain-specific problem instances in a single-source programming context. In 2020, the languages team 1) refined QCOR's memory and execution models (Task 2.1) as well as its data types (Task 2.3), 2) defined expressions and statement design structures for the HLQC language enabling the span of variational quantum computing (Task 2.2), and 3) wrote a specification detailing the semantics of the QCOR language extension (Task 2.3 deliverable) which also contains numerous examples and use cases.

QCOR Data Type and Structure Definitions. We have clarified the definitions for several data structures with which one may concisely express quantum-classical algorithms. Note that the *Observable* and *ObservableTransform* data structures has been removed. As explained below the *Operator* combines the salient features of the former *Observable* type with low-level operators useful in constructing problem instances and operators containing domain-specific information such as algebraic structures, symmetries, etc. Likewise, the functionality of the *ObservableTransform* data-structures is now more naturally called through the *Operator* data-type methods. An up-to-date list and detailed description of data-types appears at https://github.com/ORNL-QCI/qcor_spec.

QCOR Memory Model. We have defined QCOR to employ a sequential consistency model, where memory operations on the controller and device memory within a host thread happen in program order. We are exploring the possibility of utilizing a relaxed memory model introduced in C++11 (e.g., the release-acquire ordering and the relaxed ordering) if it is beneficial for the users. An explicit QCOR qalloc – i.e. allocation of memory for "qubits" and "quantum results" – also implicitly requires corresponding memory for "qubits" and "quantum results" to be accessible on the quantum device. Additionally, the host explicitly allocates memory for instances of objective functions, operators, etc. The size of the memory allocation differs across the host, quantum controller, and quantum device. For example, an explicit allocation of "qubits" by the host allocates $(24N+C)*E$ bits of memory on the host (to store bitstrings and counts), $N*E$ bits of memory on the controller to store output bitstrings, and N qubits on the quantum device, where N is the number of qubits, C is the bit size of an integer on a classical system, and E is the number of experiments to be performed on the quantum device.

Expressions and Statements Design in the HLQC language and DSLs. QCOR exposes two primary library calls to the user that enable a wide variety of hybrid quantum-classical use cases: the taskInitiate and sync calls. The taskInitiate call asynchronously orchestrates and composes the workflow necessary to initialize the *ObjectiveFunction* and begin classical optimization of it through the provided *Optimizer*. The sync call synchronizes execution of host and quantum device and returns the measure values from the quantum device.

QCOR Specification. We have documented the QCOR language in a specification document which is publicly available online at https://github.com/ORNL-QCI/qcor_spec until formally published. The documentation contains sections overviewing the Programming Model, Memory Model, Execution Model, Datatypes, QCOR Library APIs, and QCOR examples. Usable expressions and statements are detailed and explained in the Library API section. These include the taskInitiate and sync calls as well as memory management routines (for creating operators), algebraic routines (for manipulating and transforming operators), and accessor routines for data retrieval. A guide to writing, compiling, and running QCOR code is provided in the specification appendix. This document will continue to be maintained and updated throughout the remained of the QCOR research program.

Task 3: Interfaces and Compiler Development

In FY 2020 the QCAT Compiler team focused on extending the capabilities of the QCOR C++ language extension and associated compiler infrastructure. Building on last FY's prototype compiler we have improved the implementation to utilize more advanced functionalities in the Clang and LLVM compiler infrastructure. Our work, tightly collaborating across the DOE ARQC program (namely the AIDE-QC project), has resulted in a compiler platform that is truly one-of-a-kind. The QCOR compiler is retargetable with respect to distinct hardware backends, contains optimization capabilities, and is extensible with regards to a variety of quantum languages. Close coordination with the Task 2 Programming Languages Team has ensured that the programming model put forward by our compiler implementation is specification. A major effort coming out of the Compiler team this year has been the implementation of specified data structures for variational quantum computing tasks. Finally, we have also extended key XACC interfaces to enable pulse-level programming and quantum optimal control strategies. The low-level programming interfaces provide a platform for the protocols developed by the Task 4 Hardware-Libraries thrust to be realized. Our optimal control work further extends and leverages the specification data model, thereby providing an excellent first use-case demonstration of the utility of the work coming out of the Task 2 Languages team.

The QCOR compiler platform builds upon the XACC framework, previously developed at ORNL. XACC exposes a robust set of APIs that expose service interface implementations for the various aspects of the quantum programming workflow. The qcor compiler builds upon this API and leverages it to provide a single-source programming model, compiler, and execution engine for quantum-classical computing.

Our compiler development team has closely collaborated with the Languages team to ensure that our compiler implementation is specification adherent. We have implemented the QCOR extensible data structures within the QCOR compiler platform (and in the lower-level XACC compiling framework) that promote the high-level programmability of common variational quantum computing tasks. We provide an *ObjectiveFunction* interface upon which variational quantum eigensolver algorithms, and other variational tasks returning the fidelity of an output probability distribution with respect to some target (using the familiar Kullback-Leibler or Jansen-Shannon divergence metrics). We have provided an *Optimizer* extension point and implemented it for common optimizer strategies included in MLPack and NLOpt We have provided an *Operator* interface that enables our platform to provide data structures for modeling spin and fermion operators, and provide measurement functionality on incoming state-preparation quantum kernels. We have provided an *OperatorTransformation* service interface that enables the injection of strategies for modifying or transforming input Operators (useful for symmetry reductions, Jordan-Wigner, etc.). These developments, in tandem with our collaborative work with AIDE-QC on quantum kernel expression and definition, have enabled an API which efficiently expresses quantum-classical variational tasks.

We have extended core interfaces in XACC to enable pulse-level programming, compilation, and execution. Our work this year has extended the XACC IR with Pulse Instruction data-types that model the channel, pulse shape, execution time, and duration of a given custom pulse. The Pulse Instruction is fully IBM OpenPulse adherent and has been used to submit cloud instances successfully. We have defined a new Scheduler interface for the injection of custom pulse scheduling routines. In doing so, we he have adopted and extended the Optimizer concept coming out of the Language thrust for quantum optimal control. Our Optimizer implementation for optimal control enables one to inject target unitaries as matrices (or alternative representations), define custom backend Hamiltonian models, and specify the control strategy for computing an optimal set of pulses with respect to that backend. We expose GOAT and GRAPE algorithms by default. In addition we detail how to extend this in both C++ and Python - enabling the injection of common libraries developed by third parties for optimal control.

In collaboration with the DOE ARQC AIDE-QC project, we have enabled the utility of the QCOR platform from within the Python programming language. QCOR now extends both C++ and Python with support for a single-source programming model for quantum-classical computing. The QCAT Compiler Team work focused on building upon this Pythonic language extension and leveraging it to provide the variational data structures put forward by the QCOR specification. Moreover, since we are now able to operate in Python, we also have integrated popular third-party libraries within the platform for quantum-classical modeling and simulation tasks, specifically OpenFermion, PySCF, Psi4, Numpy, and Scipy.

All of our work continues to be open to the broader community and transparent. The work described above is fully open source and available at https://github.com/ornl-qci/qcor. Interested users are able to clone this repository and build the code to produce the functional QCOR compiler executable. We have provided robust build instructions and demonstrative example source codes that help users understand how to use this compiler. We have focused on various software engineering best-practices, including continuous integration, version control, and test-driven development. In addition to this deliverable, our team has also had a strong influence on the definition of the language specification and participated in the delivery of the initial QCOR specification publication.

Task 4: Hardware-level control and optimization libraries.

Optimal Control for Quantum Detectors: We have developed novel quantum control protocols to optimally discriminate the presence or absence of a stochastic signal from underlying noise. These control protocols were identified using a combination of analytical (filter function formalism) and numerical (optimal control theory) techniques. Our results show that commonly used sensing protocols such as spin-locking or CPMG sequences are nearly optimal for detection. Having identified the optimal quantum algorithm for detecting stochastic signals in this work, we are subsequently testing these algorithms on NISQ hardware. Leveraging our recently developed platform agnostic signal/noise injection techniques, we have enabled currently available quantum processors to be used for an alternative application: quantum sensing simulations. We are currently utilizing NISQ hardware to verify our theoretical findings and develop new approaches for robust sensing protocols that are robust against gate errors in addition to the native noise in the system.

A model-based approach for time-correlated noise in quantum circuits: Temporal noise correlations are ubiquitous in quantum systems, yet often neglected in the quantum circuit analysis due to the complexity required to accurately characterize and model the correlations. Autoregressive moving average (ARMA) models are a fundamental time series analysis technique that models temporal correlations in data. In our recent work, we developed a novel approach, denoted as Schrodinger Wave ARMA (SchWARMA), which provides a natural path for generalization of classic techniques from signal processing, control theory, and system identification for which ARMA models and linear systems are essential. Based on these results we showed that our SchWARMA technique enables the broad theory of classical signal processing to be applied for quantum system simulation, characterization, and noise mitigation.

Non-equilibrium criticality in quench dynamics of long-range spin models: One of the challenges in characterizing Hamiltonian dynamics on a quantum computer is to quantitatively predict the dynamics of correlation functions. We recently studied long-range spin models relevant to certain trapped-ion quantum simulators and, using a combination of analytical and numerical approaches, obtained accurate dynamics of correlations and fluctuations following a quantum quench. In addition we also show that critical features of the non-equilibrium dynamics can be extracted from the dynamics of the correlations and fluctuations in this system.

Toward practical proposals for analog simulations of non-Abelian gauge theories: Exploiting the novel features of LSH framework, this year we have developed an analog quantum simulator for simulating the real-time dynamics of a 1+1 dimensional non-Abelian gauge theory. This is the first manifestly gauge invariant and large-scale analog simulation proposal that is well within the existing capacity of ultracold atom experiments, and hence should be immediately demonstrable. The scheme is shown to be particularly effective in simulating the theory in weak-coupling regime, and also in the bulk limit of the theory in the strong-coupling regime of gauge theory that are particularly expensive to compute on a classical computer.

Digital-Analog Quantum Simulations Using The Cross-Resonance Effect: Digital-analog quantum computation aims to reduce the currently infeasible resource requirements needed for near-term quantum information processing by replacing a series of two-qubit gates with a unitary transformation generated by the systems underlying Hamiltonian. Inspired by this paradigm, we have analyzed superconducting architectures in order to extend the cross-resonance effect, up to first order in perturbation theory, from a pair of qubits to 1D chains and 2D square lattices. In an appropriate reference frame, a purely two-local Hamiltonian comprised of non-commuting interactions arises from the multi-qubit cross-resonant interaction. By augmenting the analog Hamiltonian dynamics with single-qubit gates we generate new families of locally transformed analog Hamiltonians. Toggling between these Hamiltonians we design unitary sequences simulating the dynamics of Ising, XY, and Heisenberg spin models. Our dynamics simulations are Trotter error-free for the Ising and XY models in 1D. We also show that the Trotter errors for 2D XY and 1D Heisenberg chains are reduced, with respect to a digital decomposition, by a constant factor. Our Hamiltonian toggling techniques could be extended to derive new analog Hamiltonians which may be of use in more complex digital-analog quantum simulations for various models of interacting spins.

for public release. Distribution is unlimited.

Preprints in 2020

1. Dasgupta, R., & Raychowdhury, I. (2020). Cold Atom Quantum Simulator for String and Hadron Dynamics in Non-Abelian Lattice Gauge Theory. arXiv:2009.13969

2. Dasgupta, S., Hamilton, K. E., & Banerjee, A. (2020). Designing a NISQ reservoir with maximal memory capacity for volatility forecasting. arXiv:2004.08240

3. Davoudi, Z., Raychowdhury, I., & Shaw, A. (2020). Search for Efficient Formulations for Hamiltonian Simulation of non-Abelian Lattice Gauge Theories. arXiv:2009.11802

4. Dumitrescu, E. F., Hsu, C.-H., Kharazi, T., Lougovski, P., McCaskey, A., Mintz, T. M., Moore, S. V., Powers, S., & Webb, Z. (2020) QCOR: Application Programming Interface Version 1.0., https://github.com/ORNL-QCI/qcor_spec

5. Gonzalez-Raya, T., Asensio-Perea, R., Martin, A., Celeri, L. C., Sanz, M., Lougovski, P., & Dumitrescu, E. F. (2020). Digital-Analog Quantum Simulations Using The Cross-Resonance Effect. arXiv:2011.10507

6. Kalev, A., & Hen, I. (2020). Simulating Hamiltonian Dynamics with an Off-diagonal Series Expansion. arXiv:2006.02539

7. Nguyen, T., Santana, A., Kharazi, T., Claudino, D., Finkel, H., & McCaskey, A. (2020). Extending C++ for Heterogeneous Quantum-Classical Computing. arXiv:2010.03935

8. Nguyen, T., & McCaskey, A. (2020). Enabling pulse-level programming, compilation, and execution in XACC. arXiv:2003.11971

9. Schultz, K., Quiroz, G., Titum, P., & Clader, B. D. (2020). SchWARMA: A model-based approach for time-correlated noise in quantum circuits. arXiv:2010.04580

10. White, C. D., Cao, C. J., & Swingle, B. (2020). Conformal field theories are magical. arXiv:2007.01303

11. White, C. D., Heußen, S., & Refael, G. (2020). Extracting many body localization lengths with an imaginary vector potential. arXiv:2003.09430

12. White, C. D., & Wilson, J. H. (2020). Mana in Haar-random states. arXiv:2011.13937

13. Xu, S., Susskind, L., Su, Y., & Swingle, B. (2020). A sparse model of quantum holography. arXiv:2008.02303

Publications in 2020

1. Nguyen, T., Santana, A., & McCaskey, A. (2020). Extending XACC for Quantum Optimal Control. 2020 IEEE International Conference on Quantum Computing and Engineering (QCE), 391-401.

2. Gupta, L., Barash, L., & Hen, I. (2020). Calculating the divided differences of the exponential function by addition and removal of inputs. Computer Physics Communications, 254, 107385.

3. Xu, L., Lee, J. T., & Freericks, J. K. (2020). Test of the unitary coupled-cluster variational quantum eigensolver for a simple strongly correlated condensed-matter system. Modern Physics Letters B, 34(19-20).

4. Raychowdhury, I., & Stryker, J. R. (2020). Solving Gauss's law on digital quantum computers with loop-string-hadron digitization. Physical Review Research, 2(3).

5. Shaw, A. F., Lougovski, P., Stryker, J. R., & Wiebe, N. (2020). Quantum algorithms for simulating the lattice schwinger model. Quantum, 4, 306.

6. Raychowdhury, I., & Stryker, J. R. (2020). Loop, string, and hadron dynamics in SU(2) Hamiltonian lattice gauge theories. Physical Review D, 101(11).

7. Dumitrescu, E. F., & Lougovski, P. (2020). Hamiltonian assignment for open quantum systems. Physical Review Research, 2(3), 033251.

8. Keen, T., Maier, T., Johnston, S., & Lougovski, P. (2020). Quantum-classical simulation of two-site dynamical mean-field theory on noisy quantum hardware. Quantum Science and Technology, 5(3), 035001.

9. Titum, P., & Maghrebi, M. F. (2020). Nonequilibrium Criticality in Quench Dynamics of Long-Range Spin Models. Physical Review Letters, 125(4).

10. Mintz, T. M., McCaskey, A. J., Dumitrescu, E. F., Moore, S. V., Powers, S., & Lougovski, P. (2020). QCOR: A language extension specification for the heterogeneous quantum-classical model of computation. ACM Journal on Emerging Technologies in Computing Systems, 16(2), 1-17.

11. Davoudi, Z., Hafezi, M., Monroe, C., Pagano, G., Seif, A., & Shaw, A. (2020). Towards analog quantum simulations of lattice gauge theories with trapped ions. Physical Review Research, 2(2).

Invited Talks

ORNL

- E. Dumitrescu "A Holistic Software Stack for Domain Science Applications", DOE-ARQC Quantum Seminar Series, Nov 4 2020.

- S. Dasgupta “Designing a NISQ reservoir with maximal memory capacity for volatility forecasting” Symposia on reservoir computing, International Conference on Neuromorphic Systems 2020.
- Alex McCaskey, Eugene Dumitrescu, Pavel Lougovski, Sarah Powers, Tiffany Mintz, Shirley Moore. Extending Modern C++ for Heterogeneous Quantum-Classical Computing. Programming Languages for Quantum Computing (PLanQC) POPL. Jan 2020.
- Thien Nguyen and Alex McCaskey. Scalable Pulse-level Software Infrastructure for the XACC Framework. OLCF Quantum Computing User Forum 2020.

UMD

- C. White, “Conformal field theories are magical”, Caltech Institute for Quantum Information and Matter, February 2020
- C. White, “Conformal field theories are magical”, Harvard-Smithsonian Institute for Theoretical Atomic, Molecular, and Optical Physics seminar, March 2020 (cancelled due to covid)
- C. White, “Conformal field theories are magical”, Summer Seminar Series on Quantum Computing and High-Energy Physics, June 2020
- B. Swingle, “Conformal field theories are magical”, Quantum Aspects of Space-Time and Matter, July 2020
- B. Swingle, “Conformal field theories are magical”, Complexity from Quantum Information to Black Holes, June 2020
- B. Swingle, “A Sparse Model of Quantum Holography”, MIT Seminar, October 2020
- Z. Davoudi, “Toward analog quantum simulation of lattice gauge theories with trapped ions”, Theory Seminar, Fermi National Laboratory, Department of Theoretical Physics, April 2020.
- Z. Davoudi, “Advances in quantum simulation for nuclear physics”, the 2020 Fall meeting of the APS division of nuclear physics, October 2020.
- Z. Davoudi, “Nuclear physics entering a quantum-simulation era: Lessons from the past, vision for the future”, virtual ECT* workshop on “Advances in many-body theories: from first principle methods to quantum computing and machine learning”, Trento, Italy, November 2020.
- Z. Davoudi, “Quantum simulators for nuclear physics: Progress, challenges, and future”, Physics Division Seminar, Argonne National Laboratory, November 2020.
- Z. Davoudi, “Quantum simulators for nuclear and particle physics: Progress, challenges, and future”, physics colloquium, Institute for Quantum Computing (IQC), University of Waterloo, November 2020.

- Z. Davoudi, "Toward quantum simulation of systems of relevance to nuclear and particle physics", the annual IFT Xmas workshop, the Instituto de Fisica Teorica (IFT), Madrid, Spain, December 2020.
- I. Raychowdhury, "Towards quantum simulating non-Abelian lattice gauge theories", JLab theory seminar, Jefferson National Laboratory, Newport News, VA, January 2020.
- I. Raychowdhury, "Towards quantum simulating non-Abelian lattice gauge theories", Nuclear and Particle Theory Seminar, Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA, October 2020.

USC

- I. Hen, "Future of Analog Quantum Computing: Hopes and Hinderances", LBNL Advanced Quantum Testbed Colloquium, Berkeley CA, February 2020.

APL

- Paraj Titum, "Non-equilibrium criticality in quench dynamics: Universal dynamics of critical quenches in long-range spin models", Condensed Matter Seminar at Michigan State (Oct. 2020)

Outreach

- A. McCaskey co-organized the 2020 OLCF Quantum Computing User Forum.
- Z. Davoudi, invited panelist of the Institute for Nuclear Theory (INT) virtual panel on "Quantum directions for nuclear and particle physics", Seattle, WA, November 2020.
- Z. Davoudi, session chair of the mini-symposium on "Quantum Information Science and Technology for Nuclear Physics" at the 2020 Fall meeting of the APS Division of Nuclear Physics, October 2020.

Software

- Mezze: Quantum Dynamics Simulation Toolbox (APL)
- QCOR compiler and libraries
- QCOR specification
- https://github.com/eclipse/xacc
- https://github.com/ornl-qci/libcmaes
- https://github.com/ornl-qci/QuaC/tree/xacc-integration
- Implemented DMRG noise term for the widely used open source tensor network library ITensors.jl. (A DMRG noise term is a technical improvement to DMRG—the "density matrix renormalization group", a core matrix product state method for finding ground states—that improves convergence in the presence of symmetries.)
- Many bug reports and fixes for ITensors.jl.

Awards and Recognitions

- Eugene Dumitrescu (ORNL) received the Computational Sciences and Engineering Division's 2020 best paper award.

- Austin Adams, Jeffrey Young, Tom Conte, Creston Harold, "Designing a Modular, Full-Stack Compilation Path for Ion Trap-Based Quantum Computers". 2020. Graduate fellowship award from the Georgia Tech Quantum Alliance for Spring 2021 to investigate the use of QCOR with ion trap platforms.