My primary research interests are in **full-stack quantum computation**. Noisy, Intermediate Scale Quantum (NISQ) devices have flaws—but they also hold tremendous power to test the limits of quantum algorithms and address near-term problems in simulation and optimization. My two primary interests center around how we can better optimize across multiple layers of the stack to maximize the value of NISQ devices:

- *Co-designing quantum algorithms with hardware:* How can we best exploit device architectures with tailored algorithms?
- *Engineering device-aware quantum circuits*: How can we compile algorithms to specific devices, for noise resilience and performance? What bounds can be theorized under noisy regimes?

These interest areas—and full-stack as a subfield—excite me because I want to build bridges across disciplines. My professional and personal experiences taught me the value of communication: being mentored by physicists, theoretical chemists, and computer scientists requires active effort, but can yield incredible results. Serving UW CSE in the LGBTQ+ and Diversity orgs requires active empathy to build a common path forward. Through graduate school, I hope to formulate a pragmatic, shared vision for NISQ devices and skill set to design and implement near-term quantum algorithms.

Motivation for Full-Stack Quantum Computation I have observed that mapping real-world problems to device-level instructions occurs in three phases: *targeting* key computational challenges from broader problem scopes, *mapping* these problems to algorithms, and *optimizing* the actual device-level implementation. These phases often occur in isolation, with computer scientists and physicists agreeing to broad abstractions of how quantum devices should operate, enabling computer scientists to focus on algorithm design while physicists achieve agreed abstractions. However, maintaining high-level abstractions is totally unrealistic on NISQ devices: all present platforms are constrained by decoherence and imperfect gates. Furthermore, erecting abstraction barriers prevents computer scientists from designing device-specific algorithms which exploit unique control mechanisms and primitive gate sets.

To yield practical, near-term results, we should re-evaluate inaccurate abstractions. I focus on two thrusts: *co-design* and *compilation*. Early in the algorithm design process, computer scientists need to understand how a platform's quirks translate into algorithmic advantages for specific problem scopes. These scientists need to *co-design* algorithms with hardware by flexibly changing algorithm designs and hardware implementation to realize results. Then, after tailoring device-specific algorithms, an optimized quantum circuit can be *compiled*, taking into account the empirical performance of devices.

Co-Designing Quantum Algorithms with Hardware On the algorithm side, I believe in a pragmatic approach to algorithm design. This entails traditional, theory-focused algorithm design methods, but also empirical simulations using classical computers to help tailor algorithms.

This pragmatic approach was shaped by my research. Under Professor Nathan Wiebe, I was challenged to design algorithms on a hybrid boson-qubit architecture. I realized that, while the qubit paradigm is prevalent, it is not unitary. Boson-qubit, qudits—different computational paradigms exist and require new algorithms designed around the architecture. I leveraged the boson-qubit device's quirks via Hamiltonian simulation techniques, producing a control scheme relevant for physical simulation. This algorithm produces a closed form expression via recursive applications of the Baker-Campbell-Hausdorff and Trotter-Suzuki product formulas, whereas previously scientists would need to rely upon Optimal Control Theory [1]. By designing algorithms around architecture, we could effectively tackle physical simulation problems where other devices running general algorithms would struggle.

Additionally, while hardware matures, classical emulations of quantum algorithms can facilitate algorithm design and testing, even under constrained problem domains. I realized this by designing and implementing QPESIM, a classical emulator of quantum Hamiltonian simulation algorithms [2]. QPESIM emulates quantum phase-estimation based (QPE) Hamiltonian simulation for fermionic systems with superpolynomially less memory than general emulators by exploiting properties of electronic configuration. This introduces two paths forward: first, QPESIM allows empirical tests of QPE based simulation. Second, we can design and test new quantum algorithms using QPESIM. For example, we can now empirically compare deterministic and stochastic simulation algorithms, enabling us to better analyze product formulas and the role of stochasticity in simulation [3].

Engineering Device-Aware Quantum Circuits Given tailored algorithms, we also need effective, device-level implementations. I want to design device-specific compilers while simultaneously contextualizing application discovery with theoretical bounds.

Compilation is critical to overcome device limitations in qubit connectivity and noise [4]. In my sophomore summer, I designed a pipeline to compile optimized simulation circuits under the supervision of Dr. Sriram Krishnamoorthy at the Pacific Northwest National Laboratory (PNNL). By using device-aware compilers, we could significantly reduce circuit depth, thereby making real-world implementation easier. In essence, effective compilation can only occur when actual devices are considered—abstractions must be updated to effectively implement low-level instructions.

Simultaneously, we should theorize how device constraints impact theoretical performance. For example, recent work by Bouland et al. validates that quantum random circuit sampling is classically hard, even under noisy regimes [5]. Incorporating device limitations in theory allows us to identify relevant problem domains where NISQ devices may yield results.

At the University of Chicago, I am interested in **obtaining a PhD in Computer Science** to strengthen my ability to isolate and map key computational challenges to target devices. In particular, I seek to work with Professor Fred Chong and am interested in his work in designing applications for NISQ architectures. I am also interested in Professor Bill Fefferman's work in leveraging theory to assess the computational power of noisy devices. Finally, I'm excited to collaborate with other interdisciplinary scientists through EPiQC and the Chicago Quantum Exchange.

Service and Diversity At UW, I served on the Diversity Committee for more than two years, worked in the K-12 Outreach program for more than a year, and am currently serving in a year-long appointment as the Special Assistant for Undergraduate Research. I hold positions in two student groups, representing both LGBTQ+ individuals and undergraduates broadly. I will have TA'd three times, including for the undergrad quantum course, where I wrote and presented the curriculum on Hamiltonian simulation and received the highest TA review, "Truly Exceptional." I value teaching and service, which align with my goal of becoming a professor.

[1] Childs, Andrew M., and Nathan Wiebe. "Product formulas for exponentials of commutators." Journal of Mathematical Physics 54.6 (2013): 062202.
[2] **Kang, Christopher**, et al. "Extending the Scale of Exact Molecular Simulations via Memory-Efficient Quantum Phase Estimation" *NQN/UW Workshop on Quantum Programming Theory, Experiment, and in the Classroom* (2020).
[3] Faehrmann, Paul K., et al. "Randomizing multi-product formulas for improved Hamiltonian simulation." arXiv preprint arXiv:2101.07808 (2021).
[4] Murali, Prakash, et al. "Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers." Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems. 2019.
[5] Bouland, Adam, et al. "On the complexity and verification of quantum random circuit sampling." Nature Physics 15.2 (2019): 159-163.