Research Statement

Jie Li, Ph.D.

My research asks how large computing systems can use their own operational data to make better decisions about performance, energy, and security. High-performance computing (HPC) centers now support scientific simulations, data-intensive workflows, and AI services on increasingly heterogeneous hardware. Their software must coordinate processors, GPUs, memory, storage, and power while protecting shared resources. I build the measurement tools, resource-management methods, and architectural prototypes needed to make these systems **observable, efficient, and trustworthy**.

My work connects systems research to operational practice. As a Research Assistant Professor at Texas Tech University, I help operate the NSF-funded, $12.25 million REPACSS system, which I co-designed and built, and use it as a production research testbed. Earlier, at Lawrence Berkeley National Laboratory, I studied workload behavior on NERSC systems and developed scheduling methods for disaggregated memory. My software has been used in research on monitoring, scheduling, and global-address-space architectures; MonSTer was also adopted by Dell's Omnia project. Recent work extends this foundation into power-centric observability on REPACSS, the energy cost of LLM inference, CXL memory emulation, and the security of AI agents operating in HPC environments. These results support a research agenda in which systems can **measure their state, reason about tradeoffs, and take bounded actions**. Figure 1 maps this foundation to my future directions.

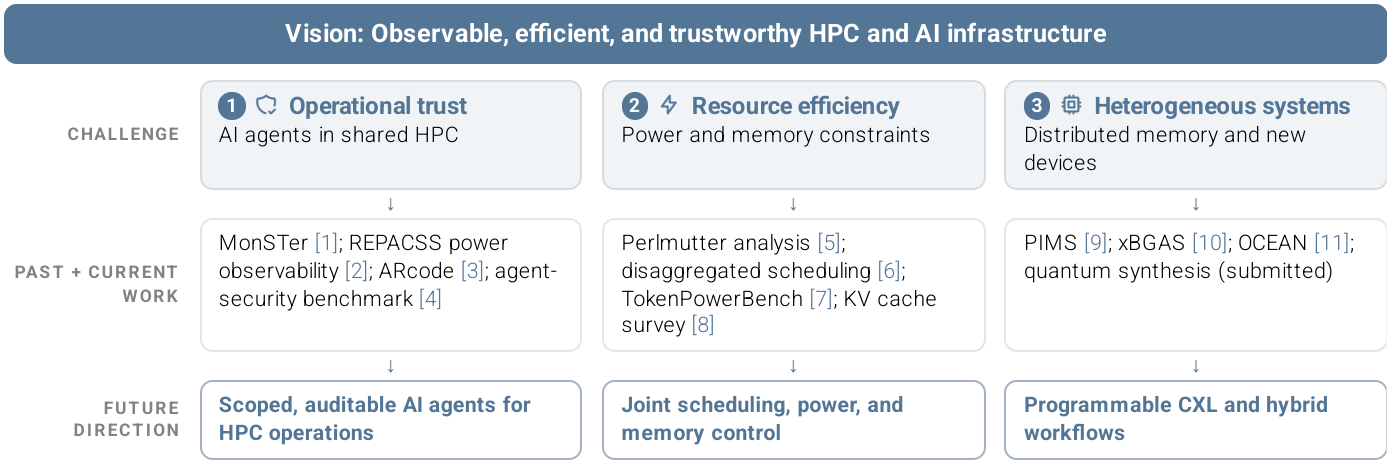

Figure 1. Research roadmap linking current results to three future directions.

Research contributions

Observability as a foundation for reliable operation

An HPC center cannot manage what it cannot see. I developed MonSTer, an out-of-the-box monitoring framework that collects system data with low operational overhead [1]. Its design helped make monitoring practical on production clusters, and its adoption by Dell's Omnia project showed that the approach could transfer beyond one research deployment. During internships at Lawrence Berkeley National Laboratory, I integrated LDMS, DCGM, and

Slurm telemetry from Cori and Perlmutter to study jobs at scale; this work led to the Perlmutter resource-utilization analysis described below [5]. I also developed methods for workload failure prediction in data centers [12], and ARcode represented monitoring data as images for application recognition [3], offering a way to identify unexpected workload behavior.

My recent work moves observability from general system health toward power-aware operation. With collaborators, I developed and evaluated power-centric observability on REPACSS [2]. This work provides an empirical foundation for asking when and where a workload consumes power, and for connecting those observations to scheduling and control. The system itself is an important part of my research method: having helped build REPACSS, I can test whether a proposed mechanism survives the practical constraints of instrumentation, deployment, and user workloads.

Resource management across memory and energy constraints

Resource allocation becomes harder when capacity is distributed or costly to use. My analysis of NERSC's Perlmutter system documented how real applications consume compute and memory resources [5]. Building on those measurements, I designed and evaluated scheduling and allocation methods for disaggregated memory [6], [13], and I led development of the open-source Disaggregation-Aware Scheduler for exploring policy tradeoffs. This line of work treats memory placement, access cost, and job scheduling as a coupled problem rather than independent layers. It also provides a basis for studying newer CXL-enabled systems; I contributed to OCEAN, an open-source CXL emulation effort accepted at SC 2026 [11].

Energy is now a similarly concrete systems constraint. I coauthored TokenPowerBench, a benchmark for the power consumption of LLM inference [7]. This work brings power measurement into AI infrastructure, where serving choices must be evaluated against both user-facing performance and electricity use. A recent survey I coauthored maps KV cache management across the memory hierarchy of LLM serving [8]. Ongoing collaborative work, with manuscripts under submission or revision, examines online CPU-frequency control for HPC and the power effects of model distillation. Together these projects ask which system decisions save energy while maintaining performance.

Hardware-software co-design for emerging systems

My earlier architecture research gives this systems agenda a deeper view of data movement. I designed PIMS, a processing-in-memory accelerator for stencil computations, to reduce transfers between compute and memory [9]. I also contributed to management techniques for 3D-stacked memory, including a memory access coalescer and a hotspot-aware manager [14], [15]. In the xBGAS project, I developed cycle-accurate simulation and runtime support for global addressing in RISC-V-based systems [10]. These prototypes make proposed architectural ideas testable before they are broadly available in hardware. OCEAN extends this empirical approach to CXL memory systems [11].

I am also beginning to explore where HPC methods can support quantum computing. A collaborative manuscript submitted to AAAI 2027 studies neural-native quantum arithmetic and polynomial synthesis. This emerging direction connects to my broader interest in the software and resource-management requirements of hybrid classical-quantum workflows.

Future research agenda

My next goal is to close the loop between measurement and action while keeping the resulting system auditable and under human control. I will pursue three connected directions, using REPACSS and open-source research tools for realistic evaluation.

1 Trustworthy AI agents for HPC operations

HPC operations involve repetitive but consequential tasks: configuring nodes, maintaining software stacks, diagnosing failures, and managing jobs. I have initiated work on agent-assisted REPACSS operations, including student-led automation around Warewulf, Ansible, Spack, and Slurm. My 2026 preprint on LLM-agent security in HPC [4] motivates a central question: **how can an agent be useful when it has legitimate credentials but may take unsafe actions?**

I will build an operational agent architecture with scoped permissions, explicit plans, action logs, and approval gates for high-impact changes. The research challenge is to translate natural-language requests into verifiable system actions while accounting for changing cluster state, incomplete telemetry, and tool failures. I will develop benchmarks that test both task completion and safety, including unauthorized configuration changes, misuse of credentials, and recovery from incorrect actions. Initial studies can run in replicated or sandboxed environments; carefully bounded deployments on REPACSS will then measure administrator time, task success, and incident rates. My previous work on MonSTer, ARcode, and failure prediction supplies the sensing layer, while the SHIELD proposal to NSF CICI, on which I was Co-PI, and my current student mentoring provide a starting point for the security layer.

2 Power- and memory-aware infrastructure for HPC and AI

HPC and AI workloads increasingly compete for power, accelerators, and memory capacity. I will connect REPACSS power telemetry [2], TokenPowerBench [7], and workload traces [5] to models that predict the energy and performance effects of placement, frequency, and memory policy. The aim is to optimize *energy to solution* or *energy per served token* subject to throughput, latency, fairness, and reliability constraints.

One project will study online control of CPU and GPU operating points alongside queue-level scheduling, using measured power rather than static device specifications. Another will treat LLM KV cache placement and disaggregated memory as a shared systems problem: which data should stay near an accelerator, spill to host or pooled memory, or be recomputed? I will evaluate these choices with representative HPC jobs and AI serving workloads, reporting both benefits and overheads. REPACSS's renewable-energy mission also creates an opportunity to study when flexible jobs can shift in time without harming users. This direction builds directly on my scheduling research [6], CXL emulation work [11], and current energy-control collaborations.

3 Programmable heterogeneous and hybrid systems

As memory and accelerators become more distributed, programmers need abstractions that expose useful locality without requiring application-specific management of every device. I will extend my xBGAS and CXL work [10], [11] to investigate how runtimes and schedulers coordinate global addressing, pooled memory, and data movement.

Experiments will compare programmer effort, access latency, system throughput, and power against conventional node-local designs. My PIMS work [9] provides a complementary path: move selected operations toward data when doing so is more effective than moving data toward compute.

For hybrid classical-quantum workflows, I will first focus on tractable software questions: how to synthesize useful quantum arithmetic, represent workflow dependencies, and schedule classical and quantum stages when quantum resources are scarce. The submitted polynomial-synthesis work is an initial step. I will use simulation and available experimental platforms to establish measurable baselines before proposing tighter integration with HPC infrastructure.

Closing perspective

The through line of my research is a progression from **observing** complex systems to **allocating** resources intelligently and then **acting** safely. My experience building REPACSS, collaborating with national laboratories and industry, serving as assistant director of the Texas Tech site of the NSF Cloud and Autonomic Computing Center, and mentoring students gives me a practical setting for this agenda. I aim to develop open methods that make scientific and AI infrastructure more efficient, dependable, and secure, while training students to work across architecture, systems software, and real operations.

Selected references

[1] **J. Li**, G. Ali, N. Nguyen, J. Hass, A. Sill, T. Dang, and Y. Chen, "MonSTer: An Out-of-the-Box Monitoring Tool for High Performance Computing Systems," in *2020 IEEE International Conference on Cluster Computing (CLUSTER'20)*, IEEE, 2020, pp. 119–129. doi: 10.1109/CLUSTER49012.2020.00022. (Acceptance Rate: 27/132=20.5%)

[2] Y. Zhao, **J. Li**, C. Niu, A. Sill, and Y. Chen, "Power-Centric Observability for HPC Systems: Design, Deployment, and Evaluation on REPACSS," in *Practice and Experience in Advanced Research Computing (PEARC'26)*, 2026, pp. 1–8.

[3] **J. Li**, B. Cook, and Y. Chen, "ARcode: HPC Application Recognition Through Image-encoded Monitoring Data," *arXiv preprint arXiv:2301.08612*, 2023, doi: 10.48550/arXiv.2301.08612.

[4] **J. Li**, "Trusted Credentials, Untrusted Behavior: Benchmarking LLM-Agent Security in High-Performance Computing," *arXiv preprint arXiv:2607.18485*, 2026.

[5] **J. Li**, G. Michelogiannakis, B. Cook, D. Cooray, and Y. Chen, "Analyzing Resource Utilization in an HPC System: A Case Study of NERSC's Perlmutter," in *International Conference on High Performance Computing (ISC'23)*, Springer, 2023, pp. 297–316. doi: 10.1007/978-3-031-32041-5_16. (Acceptance Rate: 21/78=26.9%)

[6] **J. Li**, G. Michelogiannakis, S. Maloney, B. Cook, E. Suarez, J. Shalf, and Y. Chen, "Job Scheduling in High Performance Computing Systems with Disaggregated Memory Resources," in *2024 IEEE International Conference on Cluster Computing (CLUSTER'24)*, IEEE, 2024, pp. 297–309. doi: 10.1109/CLUSTER59578.2024.00033.

[7] C. Niu, W. Zhang, **J. Li**, Y. Zhao, T. Wang, X. Wang, and Y. Chen, “TokenPowerBench: Benchmarking the Power Consumption of LLM Inference,” in *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI’26)*, 2026, pp. 32582–32590. doi: 10.1609/aaai.v40i38.40535. (Acceptance Rate: 4,167/23,680=17.6%)

[8] **J. Li**, T. Wang, and Y. Chen, “From Tensor Buffer to Distributed Memory Hierarchy: A Survey of KV Cache Management for LLM Serving,” *arXiv preprint arXiv:2607.02574*, 2026.

[9] **J. Li**, X. Wang, A. Tumeo, B. Williams, J. D. Leidel, and Y. Chen, “PIMS: A Lightweight Processing-in-Memory Accelerator for Stencil Computations,” in *Proceedings of the International Symposium on Memory Systems (MemSys’19)*, 2019, pp. 41–52. doi: 10.1145/3357526.3357550.

[10] **J. Li**, J. D. Leidel, B. Page, and Y. Chen, “Towards Cycle-accurate Simulation of xBGAS,” in *2024 International Conference on Computing, Networking and Communications (ICNC’24)*, IEEE, 2024, pp. 468–472. doi: 10.1109/ICNC59896.2024.10556078.

[11] Y. Yang, M. A. Rafi, J. S. Firoz, Z. Peng, X. Wang, K. Bouasker, S. Ghosh, **J. Li**, D. Wang, D. Li, H. Jeon, K. Barker, N. Tallent, and L. Guo, “OCEAN: Open-Source CXL Emulation for Hyperscale Architecture and Networking,” in *The International Conference for High Performance Computing, Networking, Storage, and Analysis (SC’26)*, 2026. Accepted

[12] **J. Li**, R. Wang, G. Ali, T. Dang, A. Sill, and Y. Chen, “Workload Failure Prediction for Data Centers,” in *2023 IEEE 16th International Conference on Cloud Computing (CLOUD’23)*, 2023, pp. 479–485. doi: 10.1109/CLOUD60044.2023.00064.

[13] **J. Li**, G. Michelogiannakis, B. Cook, J. Shalf, and Y. Chen, “Scheduling and Allocation of Disaggregated Memory Resources in HPC Systems,” in *2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW’24)*, IEEE, 2024, pp. 1202–1203. doi: 10.1109/IPDPSW63119.2024.00206.

[14] X. Wang, A. Tumeo, J. D. Leidel, **J. Li**, and Y. Chen, “MAC: Memory Access Coalescer for 3D-Stacked Memory,” in *Proceedings of the 48th International Conference on Parallel Processing (ICPP’19)*, 2019, pp. 1–10. doi: 10.1145/3337821.3337867. (Acceptance Rate: 106/405=26.2%)

[15] X. Wang, A. Tumeo, J. D. Leidel, **J. Li**, and Y. Chen, “HAM: Hotspot-Aware Manager for Improving Communications With 3D-Stacked Memory,” *IEEE Transactions on Computers (IEEE Trans Comput)*, vol. 70, no. 6, pp. 833–848, 2021, doi: 10.1109/TC.2021.3066982.