Research Statement

Rethinking Reliable and Sustainable Large-Scale Systems

Li (Lilly) Wu

Today's distributed systems have evolved into a vast continuum, from hyperscale cloud data centers spanning continents to thousands of regional edge sites operating closer to users and devices. These systems are powering transformative workloads, such as AI-driven and microservices-based applications, which are reshaping domains ranging from scientific discovery and smart manufacturing to autonomous transportation, healthcare, and immersive real-time experiences. However, this evolution introduces two pressing challenges: (1) the tight interdependencies and operational complexity of these large-scale systems make them increasingly fragile and prone to cascading failures, rendering manual operation infeasible, especially for latency-critical applications; and (2) the escalating computational demands of modern applications have driven a significant rise in energy use and environmental costs, raising urgent concerns about the sustainability of distributed computing. Addressing this dual challenge—ensuring reliable operation under failure while minimizing environmental impact—requires a foundational paradigm shift: **Reliability and Sustainability must become first-class design principles in distributed systems.**

My research tackles this challenge by developing systems and algorithms that make distributed computing inherently reliable and sustainable. While I have broad interests in large-scale distributed systems, my current work is centered at the intersection of edge/cloud computing, AI systems, and sustainability. Specifically, I develop: (1) fault-tolerant edge AI systems that enable resilient inference in resource-constrained environments (§1); (2) reliable cloud microservices that automatically diagnose and mitigate cascading failures (§2); and (3) sustainable data centers that dynamically adapts resource management decisions to reduce energy consumption and carbon emissions while guaranteeing service-level objectives (SLOs) (§3). My research is grounded in principled techniques, such as optimization, machine learning, causal inference, and analytical modeling, to understand and optimize the systems I have built. Looking ahead, I plan to extend this work to support AI systems that train and serve foundation models (e.g., large language models) across the Cloud-Edge-IoT continuum, where distributed training and inference demand new abstractions for fault tolerance, automated diagnosis and recovery, energy-efficient resource management, and cross-tier optimization. Together, these efforts aim to advance the design of distributed systems that make AI systems not only performant, but also inherently reliable and sustainable by design (§4).

1 Resilient Edge AI

Edge AI brings machine learning inference closer to the data source–on device or edge servers–rather than relying on distant cloud data centers. This shift enables millisecond-level decision-making for applications such as real-time video analytics, AR/VR, robotics, and autonomous driving. However, edge deployments are inherently fragile: hardware failures, resource contention, software bugs, and wireless network disruptions can all interrupt edge applications. Since edge applications are typically latency-sensitive and often mission-critical, even brief outages can lead to severe safety risks and economic costs. To prevent such consequences, edge AI systems must be inherently resilient to failure, ensuring the stringent SLOs can be met at scale. *How can we design edge AI systems that are inherently failure-resilient?*

1.1 Rethinking Fault Tolerance Under Resource Constraints A conventional approach for fault-tolerance is to provide replication for failover. However, unlike cloud data centers with abundant resources and energy, edge sites are severely resource-constrained by power, space, and thermal limitations. Under these constraints, cloud-style full replication, where the backup model is identical to the primary, does not scale at the edge. *How can we rethink replication to enable resilient edge AI in resource-constrained environments?*

To address this, my recent work, **FailLite** [SoCC '25], introduces a novel mechanism: *heterogeneous replication.* Rather than replicating full-size models, FailLite intelligently selects and deploys smaller model variants as backup replicas. These variants are carefully optimized to balance the accuracy-resource trade-off inherent in modern machine learning models. FailLite also incorporates the application's criticality: critical applications use warm replicas for near-zero recovery time, while less critical ones rely on cold replicas with *progressive failover*. This adaptive policy minimizes both mean time to recovery (MTTR) and accuracy degradation under resource constraints. Evaluations on real-world edge testbeds and large-scale simulations demonstrate that FailLite achieves significantly higher recovery rates, reducing MTTR by 2× while incurring only 0.6% accuracy reduction compared to full replication baselines.

1.2 Towards Resilient ML Pipelines Building on FailLite's single-model resilience, my broader research [MILCOM '24] extends resilience to entire machine learning pipelines. Modern edge AI applications often span multiple processing stages, are distributed across heterogeneous nodes, and incorporate data from multiple sensors, each introducing additional points of failure. To enable resilient ML pipelines, we propose a holistic framework comprising four complementary mechanisms: (1) sensing redundancy, using multi-modal or multi-vantage sensing to tolerate sensor failures; (2) structural resilience, designing loosely coupled pipelines to prevent cascading failures; (3) heterogeneous replication, extending FailLite's principle to distributed inference; (4) pipeline reconfiguration, leveraging queuing theory to reconfigure model variants and resources at runtime. Together, these mechanisms enable resilient edge AI systems that sustain performance under diverse failures and resource conditions.

2 Reliable Cloud Microservices

Cloud computing has transformed distributed systems by offering elastic compute, storage, and AI services through hyperscale data centers. In the cloud, microservice architecture has emerged as the dominant paradigm for building large-scale applications because of its unmatched flexibility, scalability, and agility. However, cloud microservice systems introduce a new kind of operational complexity. A single application may comprise hundreds or even thousands of loosely coupled services, co-located on shared infrastructure. These services are frequently updated to meet evolving customer demands, creating a dynamic environment where performance issues–such as slow application responses–are almost inevitable. The consequences of such anomalies are nontrivial: degraded user experience, revenue loss, and substantial operational overhead from manual troubleshooting. Additionally, traditional approaches that rely on human intervention are no longer practical at this scale. This reality calls for self-healing systems that not only detect performance anomalies in real time but also uncover their root causes and recommend actionable recovery strategies. *How can we design reliable cloud microservices that accurately diagnose root causes and recommend effective recovery actions?*

2.1 Performance Diagnosis Performance issues in cloud microservices often trigger cascading failures, resulting in widespread anomalies throughout the system. The highly dynamic, heterogeneous, and interdependent nature of microservices makes it exceptionally difficult to localize the root cause–determining both where performance anomalies originate and why they occur–among large-scale anomalies. *How can we accurately identify where performance anomalies originate and why they occur?*

My research addresses this challenge with MicroRCA [NOMS '20], a lightweight, graph-based approach for root cause localization in cloud microservice systems. MicroRCA introduces two key innovations : (1) it models anomaly propagation through both service-call dependencies and co-located effects using an attributed graph, and (2) it leverages rich metrics from both the application and system layers to precisely locate root causes. Experimental results demonstrate that MicroRCA achieves 89% precision, outperforming baselines by at least 13%.

While MicroRCA effectively answers where anomalies originate, it does not explain why they occur. To address this, my research explores deep learning [ICSOC '21, CCGrid '22] and causal inference [ACSOS '21, AIOps '21] to identify the culprit metrics that indicate the root causes. MicroRCA+ [ICSOC '21] extends MicroRCA with deep learning to identify the culprit metric responsible for performance degradation. MicroDiag [AIOps '21] employs causal inference to derive a metrics causality graph to infer root causes. Experimental results show that MicroDiag ranks 97% of root causes in the top 3, outperforming baselines by at least 31.1%.

2.2 Automatic Recovery Once the root cause of a performance issue is identified, the next challenge is selecting an effective recovery action that restores service performance rapidly without introducing side effects on other system components. *How can we choose recovery actions that quickly restore SLOs while minimizing side effects?*

My research addresses this problem through MicroRAS [UCC '20], a model-driven approach that selects recovery action by balancing effectiveness and recovery time. MicroRAS defines effectiveness as a function of two factors: the benefit of recovering the faulty service and the risk of affecting other services, and estimates it via a graphical model that captures the action-effect propagation in the system. Evaluation results show that MicroRAS successfully recovers 94.7% of service performance degradations and completes recovery at least 4× faster than baselines.

Impact: My research on self-healing cloud microservices led to the **MicroX** series, which has been widely cited. This work has been open-sourced, and has also drawn strong industry interest, including invited talks at companies across the US and Europe, and has attracted more than $600,000 in external research funding since 2021.

3 Sustainable Data Centers

The explosive growth of AI and other data-intensive workloads is pushing data centers to unprecedented levels of energy demand. By 2030, their consumption is expected to exceed 1000 TWh annually–more than double today's levels–driving a dramatic increase in their carbon footprint. Meeting this challenge requires a holistic strategy that spans improving energy efficiency with advances in power management [CCGrid '24], expanding renewable energy in the supply mix, and improving carbon efficiency with resource management. Carbon-aware resource management leverages the spatiotemporal variability of grid carbon intensity–measured as emissions per unit of electricity (g·CO_2eq/kWh)–by strategically shifting workloads across space and time. *How can we design carbon-aware resource management that reduces data center emissions while preserving SLOs?*

3.1 Edge Placement Edge applications typically have strict end-to-end latency requirements (often ≤100 ms), making temporal workload shifting infeasible. Meanwhile, prior work shows spatial workload shifting works in the cloud that has large geographic distances, but not at the edge, where long-distance moves risk unacceptable latency. *Is spatial workload shifting feasible for edge data centers? If yes, how can we reduce emissions at the edge while meeting the low-latency requirements?*

My research [HPDC '25] answered the question for the first time. Our empirical study of carbon intensity variations at mesoscale distances, spanning tens to hundreds of kilometers, demonstrates significant variations (up to 10.8×) in grid carbon intensity within mesoscale regions, and such mesoscale variations are prevalent worldwide. Based on this, we designed **CarbonEdge**, a carbon-aware framework for reducing carbon emissions from mesoscale edge sites. It combines spatial variations of grid carbon intensity with energy efficiency of heterogeneous edge devices to jointly optimize the application placement and server activation, thereby minimizing the overall emissions of edge data centers. Experimental results show that CarbonEdge achieves up to 78.7% reduction in regional deployments and 49.5–67.8% at CDN scale, with one-way latency increases kept under 5.5 ms.

3.2 Cloud Scheduling Unlike the edge's many small sites, the cloud has fewer, hyperscale data centers, creating opportunities to schedule workloads within each site. In the cloud, a significant fraction of computing demand comes from batch jobs (model training and scientific computing), which are often delay-tolerant and elastic, enabling schedulers to reduce carbon by suspending/resuming jobs and scaling their resources down/up when carbon is high/low. However, resources and workloads vary over time, and job lengths are often unknown a priori. *How can we design carbon-aware schedulers that minimize carbon emissions for batch jobs with uncertain job length?*

My research [Submitted' 25] integrates the scheduling of parallel batch workloads with the cluster-level resource provisioning to complete jobs in the time windows with low carbon intensity. By incorporating a learning-based approach, the scheduler mitigates uncertainties and dynamics in resource capacity, job length, and arrival rates, achieving a 57% reduction of emissions compared to carbon-agnostic baselines.

4 Future Plans

While my current research has taken important first steps towards designing reliable and sustainable distributed systems for emerging workloads, such as AI and microservices, many important research challenges remain. In the near term, I will work on two complementary research directions. The first is to build reliable AI systems (§4.1) that can sustain the training and inference of foundation models (e.g., large language models) in the presence of failures or performance degradation. The second direction is to advance sustainability across the Cloud-Edge-IoT continuum (§4.2), extending beyond single-layer or workload-centric optimization to cross-layer coordination and sustainable control-plane design.

4.1 Reliable AI Systems AI systems are undergoing a transition from task-specific deep learning models to general-purpose foundation models (FMs), such as large language models (LLMs) like GPTs and vision models like SAM. These models are reshaping domains from code generation and scientific discovery to real-time decision-making across the Cloud-Edge-IoT continuum. Training FMs now spans thousands of GPUs, while inference must serve millions of concurrent queries under strict SLOs. At such scales, failures are the norm: faulty GPUs, HBM errors, NVLink/NIC glitches, and thermal throttling are all common. Even without hard faults, model performance (e.g., accuracy or throughput) can degrade due to distribution drift or partial failures. Moreover, modern optimizations (e.g., speculative decoding, Mixture of Experts, pipeline parallelism) introduce stateful dependencies that make traditional failover insufficient. Thus, reliable AI systems require both rapid fault tolerance and automated performance diagnosis and recovery to maintain SLOs. My prior research in edge AI and cloud microservices has explored several of these challenges. In the future, I plan to expand this research toward developing scalable methods for fault tolerance, online diagnosis, and automated recovery tailored to FM training and inference across large-scale, distributed infrastructure. *How can we guarantee reliable foundation models training and inference across distributed systems without violating strict SLOs?*

4.2 Sustainable Cloud-Edge-IoT Modern applications increasingly span the Cloud-Edge-IoT continuum, from hyperscale cloud data centers, to mesoscale edge sites, and down to IoT sensors [SEC '25]. This layered architecture enables emerging workloads such as collaborative inference across distributed compute resources, including cloud GPUs, edge accelerators, and IoT devices. However, each layer in this continuum operates under distinct resource constraints and energy profiles. These workloads are often data-intensive, requiring frequent movement of intermediate or aggregated results across layers, making network and storage critical contributors to overall energy consumption. Meanwhile, the control plane that orchestrates these heterogeneous resources is becoming increasingly complex and energy-intensive, further amplifying the sustainability challenge. My current research focuses on energy- and carbon-efficient resource management within individual layers (e.g., cloud or edge). In the future, I plan to extend this work to the full continuum, focusing on (1) modeling fine-grained energy consumption of AI workloads accounting for compute, storage, and communication; (2) jointly optimizing compute placement, data movement, and storage to reduce energy and carbon costs; and (3) designing a sustainable control plane that orchestrates these layers with minimal overhead. In parallel, I am excited to explore the interplay between reliability and sustainability to design energy-efficient reliability techniques. *How can we enable an energy sustainable Cloud-Edge-IoT continuum for AI workloads?*

My long-term research agenda centers on developing end-to-end system architectures and abstractions that make reliability and sustainability the defaults in large-scale distributed computing. As both hardware (e.g., specialized accelerators and high-bandwidth interconnects) and applications (e.g., foundation models and high-performance computing) continue to evolve, they introduce novel fault modes, complex multi-objective trade-offs in SLOs, and new operational constraints that existing systems fall short of handling. Addressing these challenges will require rethinking the system design principles, combined with close collaboration across academia and industry. I plan to establish partnerships with edge and cloud providers to access real-world failure traces, telemetry, and at-scale deployments. In parallel, I plan to build interdisciplinary collaboration with researchers in areas such as AI, databases, sensing systems, and application domains such as healthcare, scientific discovery, and transportation. The overarching vision is to design and build a large-scale distributed computing fabric that is not only fast and intelligent, but also inherently reliable and sustainable.

References

[SoCC '25] **Li Wu**, Walid A. Hanafy, Tarek Abdelzaher, David Irwin, Jesse Milzman, and Prashant Shenoy. "FailLite: Failure-Resilient Model Serving for Resource-Constrained Edge Environments". In: *ACM Symposium on Cloud Computing (SoCC)*. 2025.

[MILCOM '24] **Li Wu**, Walid A Hanafy, Abel Souza, Tarek Abdelzaher, Gunjan Verma, and Prashant Shenoy. "Enhancing Resilience in Distributed ML Inference Pipelines for Edge Computing". In: *IEEE Military Communications Conference (MILCOM)*. IEEE. 2024, pp. 1–6.

[NOMS '20] **Li Wu**, Johan Tordsson, Erik Elmroth, and Odej Kao. "MicroRCA: Root Cause Localization of Performance Issues in Microservices". In: *IEEE/IFIP Network Operations and Management Symposium (NOMS)*. IEEE, 2020, pp. 1–9.

[ICSOC '21] **Li Wu**, Jasmin Bogatinovski, Sasho Nedelkoski, Johan Tordsson, and Odej Kao. "Performance diagnosis in cloud microservices using deep learning". In: *International Conference on Service-Oriented Computing*. Springer. 2020, pp. 85–96.

[CCGrid '22] Jasmin Bogatinovski, Sasho Nedelkoski, **Li Wu**, Jorge Cardoso, and Odej Kao. "Failure identification from unstable log data using deep learning". In: *22nd IEEE International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. IEEE. 2022, pp. 346–355.

[ACSOS '21] **Li Wu**, Johan Tordsson, Erik Elmroth, and Odej Kao. "Causal inference techniques for microservice performance diagnosis: Evaluation and guiding recommendations". In: *IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*. IEEE. 2021, pp. 21–30.

[AIOps '21] **Li Wu**, Johan Tordsson, Jasmin Bogatinovski, Erik Elmroth, and Odej Kao. "Microdiag: Fine-grained performance diagnosis for microservice systems". In: *IEEE/ACM International Workshop on Cloud Intelligence (AIOps)*. IEEE. 2021, pp. 31–36.

[UCC '20] **Li Wu**, Johan Tordsson, Alexander Acker, and Odej Kao. "MicroRAS: Automatic recovery in the absence of historical failure data for microservice systems". In: *IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC)*. IEEE. 2020, pp. 227–236.

[CCGrid '24] Mehmet Savasci, Abel Souza, **Li Wu**, David Irwin, Ahmed Ali-Eldin, and Prashant Shenoy. "SLO-Power: SLO and Power-aware Elastic Scaling for Web Services". In: *2024 IEEE 24th International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. 2024, pp. 136–147.

[HPDC '25] **Li Wu**, Walid A. Hanafy, Abel Souza, Khai Nguyen, Jan Harkes, David Irwin, Mahadev Satyanarayanan, and Prashant Shenoy. "CarbonEdge: Leveraging mesoscale spatial carbon-intensity variations for low-carbon edge computing". In: *Proceedings of the 34th ACM International Symposium on High-Performance Parallel and Distributed Computing (HPDC)*. ACM, 2025, pp. 1–13.

[Submitted' 25] Walid A Hanafy, **Li Wu**, David Irwin, and Prashant Shenoy. "CarbonFlex: Enabling Carbon-aware Provisioning and Scheduling for Cloud Clusters". In: *arXiv:2505.18357* (2025).

[SEC '25] Hetvi Shastri, Walid A Hanafy, **Li Wu**, David Irwin, Mani Srivastava, and Prashant Shenoy. "LLM-Driven Auto Configuration for Transient IoT Device Collaboration". In: *2025 ACM/IEEE Symposium on Edge Computing (SEC)*. 2025.