Sankalp Naik

M.S. Artificial Intelligence Engineering (ECE), Carnegie Mellon University
Pittsburgh, PA · sgnaik@andrew.cmu.edu · (412) 608-7192
sankalpnaik.me · github.com/Sankalp22863 · linkedin.com/in/sankalp-naik-ml

Research Interests

Heterogeneous memory systems and 3D-stacked HBM; hardware–software co-design for ML workloads; accelerator design-space exploration; compiler-directed memory placement; MLIR and domain-specific languages for heterogeneous hardware; performance modeling and parallel kernel optimization.

Research Vision

Modern AI systems are increasingly limited not by compute but by how well software can exploit heterogeneous memory, reconfigurable fabric, and accelerator design spaces too large to search by hand. I want to make these hardware decisions **first-class, analyzable objects in the compiler**: a modular, MLIR-based stack in which memory placement, accelerator configuration, and code transformation are expressed, searched, and verified within one IR, with learned models guiding the search without replacing the compiler's reasoning. My work on compiler-visible memory (XBM), design-space search with guarantees (PROSE), and learned program representations (.ksh + TIER) builds toward this goal.

Education

Carnegie Mellon University — Pittsburgh, PA
M.S. Artificial Intelligence Engineering (ECE) — Expected Dec 2026
GPA: 4.0/4.0.
Selected Coursework: How to Write Fast Code II (OpenMP/CUDA); Foundations of Computer Systems; Machine Learning; Deep Learning

Visvesvaraya National Institute of Technology (VNIT) — Nagpur, India
B.Tech in Electronics & Communication Engineering — May 2022
GPA: 9.78/10.0, Rank 1/140, Visvesvaraya Medal of Excellence
Selected Coursework: Computer Architecture; Operating Systems; Numerical Methods & Probability Theory

Research Experience

Graduate Research Assistant — NEXUS Research Group
Carnegie Mellon University — Aug 2025 – Present
Advisor: Prof. Tathagata Srimani

XBM: Heterogeneous 3D-stacked HBM with non-volatile base dies
(first author; manuscript in preparation; patent pending)

- Designed a segmented HBM stack with FeRAM-based NVDRAM in the bottom dies and DRAM above; used the read-dominated weight and write-active KV-cache access patterns in LLM inference to motivate heterogeneous memory placement.
- Profiled dense and mixture-of-experts LLM workloads and used PACT thermal simulation to evaluate the design. Relative to an STCO baseline, simulations showed 150% higher bandwidth on bandwidth-bound ML workloads, approximately 98% lower refresh power, and approximately 8.5 °C lower compute-die temperature.
- Developing the architecture manuscript and an MLIR dialect for compiler-directed placement across heterogeneous memory domains; the compiler extension is ongoing.

PROSE: Region-based probabilistic design-space exploration for HLS
(first author; manuscript in preparation)

- Built a design-space exploration framework around the observation that Pareto-optimal HLS designs concentrate in a few axis-aligned regions; uses concentration-inequality bounds to certify regions of configurations.
- Evaluated PolyBench kernels on three resource-constrained FPGAs against ScaleHLS-HIDA and AutoDSE; pruned approximately 90% of the configuration space while preserving near-optimal designs.
- Extended the framework to multi-kernel workloads by composing per-kernel region sets; implemented parallel exploration and cost-model-guided autotuning to accelerate the search.

.ksh + TIER: Multi-level MLIR representations for code generation
(ongoing)

- Designing Kernel Shorthand (.ksh), a compact DSL prototyped on xDSL, for LLM-generated programs that lower to optimized GPU kernels.
- Built a C-to-MLIR analysis pipeline through Polygeist, extracting typed dataflow and control-flow graphs at multiple dialect levels; training contrastive RGCN and GraphSAGE embeddings per level with a learned merge.
- Evaluating the embeddings as dense reinforcement-learning rewards for LLM code generation in place of binary pass/fail feedback; experiments remain ongoing.

Peer-Reviewed Publications

- **Naik, S.**, et al. "ARMPC: ARIMA-based Prediction Model for Adaptive Bitrate Scheme in Streaming." PCEMS 2022. **Best Paper Award.**
- **Naik, S.**, Katre, A. "YouTube Universe of Comments" (title abbreviated). WCONF 2023.

Patent

- "XBM: Heterogeneous 3D-stacked high-bandwidth memory with non-volatile base dies." Patent pending.

Teaching

Teaching Assistant, 18-645 How to Write Fast Code I — Aug 2026 – Present
Carnegie Mellon University

- Guide students on code optimization, profiling, and benchmarking; diagnose performance bottlenecks, review assignments, and run labs and office hours.

Teaching Assistant, 18-738 Sports Technology — Jan 2026 – May 2026
Carnegie Mellon University; guided AI and embedded-systems student projects

Industry Experience

Associate Applications Developer — India
Oracle Financial Services Software (OFSS) — Jul 2022 – Jun 2025

- Optimized API latency and designed sharded databases and fault-tolerant microservices with asynchronous messaging across multi-cluster deployments.

Selected Projects

- **CUDA linear algebra library** (C++/CUDA): GEMM, LU, TRSM, and TRSV kernels with register/shared-memory tiling and warp-level primitives; Nsight profiling to isolate memory-bound regimes.
- **Fault-tolerant replicated server**: active/passive replication with checkpointing and heartbeat-based failure detection.

Honors & Awards

- Visvesvaraya Medal of Excellence (Rank 1/140), VNIT, 2022
- Best Paper Award, PCEMS 2022

Technical Skills

Compilers & IR: MLIR, LLVM IR, Polygeist, dialect design, analysis passes, pass pipelines, IR lowering, static and dynamic analysis, cost modeling
Hardware & Architecture: Vitis HLS, AMD Versal HBM, memory-hierarchy design, HBM and 3D DRAM, roofline analysis, performance modeling, thermal simulation (PACT)
Parallel Programming: CUDA, OpenMP, SIMD, tiling, warp-level primitives, Nsight
ML: PyTorch, graph neural networks (RGCN, GraphSAGE), contrastive learning, RL for code generation, transformer inference
Languages: C++, C, CUDA C, Python, Java, SQL

References

- **Prof. Tathagata Srimani**, Electrical & Computer Engineering, Carnegie Mellon University (research advisor)