Haowei (Howard) Zhu

+1-617-459-9622 | howardchu95@gmail.com

LinkedIn | GitHub | X | sberm.cn

San Jose, California - 95131, United States

EDUCATION

• **Northeastern University** — *09/2024 - 05/2026*
Master of Science in Computer Science — San Jose, United States

• **Shenzhen University** — *09/2020 - 07/2024*
Computer Science — Shenzhen, China

EXPERIENCE

• **Google Summer of Code: Linux Kernel [G]** — *04/2024 - 09/2024*
System Software Developer — Remote, United States

  ◦ Enabled real-time analysis of CPU idle time for perf record, achieving nanosecond-level granularity.

  ◦ Implemented system call tracing with eBPF and BTF, dereferencing structure, buffer, and string pointers to visualize previously hidden pointer data, collecting 130% more system call data in perf trace.

  ◦ Contributed over 1,200 lines of patches merged into Linux kernel.

  ◦ Engaged in kernel development and maintenance, contributing over 200 patches, delivering more than 50 Reviewed-by tags, 10 Acked-by tags, and 1 Suggested-by tag.

PROJECTS

• **LLVM libc: Direct I/O from GPU using io_uring** — *08/2025 - 11/2025*
LLVM, Linux, io_uring, GPU, Libc, C++17, HSA Runtime

  ◦ Enabled efficient and direct I/O for GPUs using Linux io_uring, replacing the legacy RPC module.

  ◦ Implemented pread and pwrite in LLVM-libc using C++17, io_uring, and HSA Runtime.

  ◦ Enabled printf by forwarding snprintf into pwrite.

• **Sberf: eBPF-powered Profiler** — *06/2023 - 03/2024*
eBPF, Linux, C, Rust, Flamegraph — [O]

  ◦ Enabled stack trace and CPU counter collection with eBPF, boosting speed by 37% compared to perf while using only 1% of the disk space required by perf.

  ◦ Implemented Flamegraph visualization, 178% faster than Flamegraph.pl.

  ◦ Integrated symbol resolution via Rust FFI bindings to optimize lookup efficiency.

  ◦ Implemented an all-in-one solution for stack collection, collapsing, symbol resolution, and visualization.

PUBLICATIONS

[J.1] M. Saqib Nawaz, Shoaib Nawaz, Haowei Zhu, Unil Yun. **SPM4GAC: SPM based approach for genome analysis and classification of macromolecules**. *International Journal of Biological Macromolecules.*

SKILLS

• **Programming Languages:** C++, Rust, C, Python, Shell Script, Awk.

• **Tools and Frameworks:** GDB, Valgrind, CMake, Bazel, Perf, Linux, Nginx, Git, Systemd.