

An LLVM-based DSL Framework for LQCD

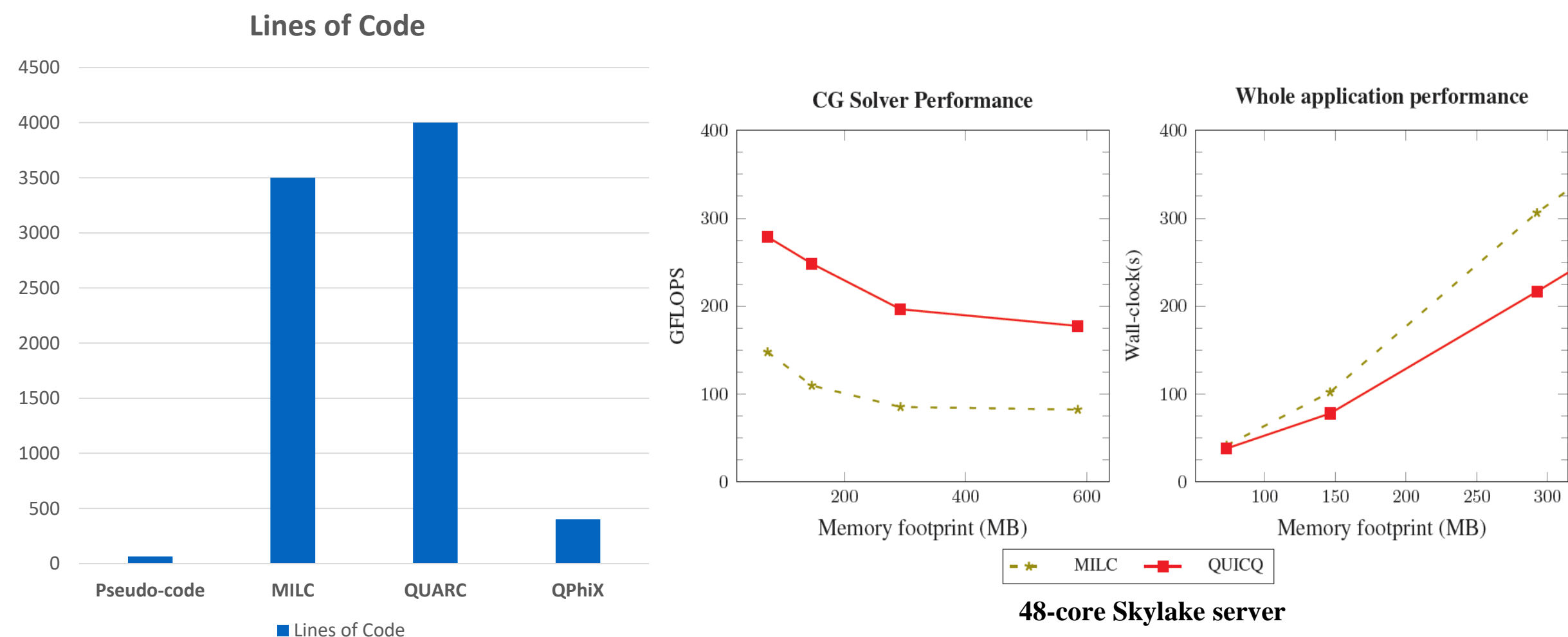
Objectives

- Design and develop a high-productive embedded DSL framework for LQCD to match the best known hand-optimized performance on x86_64 processors
- New compiler techniques for high-performance DSLs using LLVM and C++
- Improved SIMD vectorization and automation of data-layout and data distribution considerations for LQCD algorithms

Impact

- Close to QPhiX performance using an automated DSL using one-tenth lines of code for Wilson and Staggered Dslash kernels
- Complete iterative solvers in MILC rewritten using drop-in DSL code, fully automating SIMD vectorization and MPI code generation
- Improvements demonstrate a way forward for combining high programmer productivity with high performance inside LQCD

QUARC v/s MILC



Accomplishments

- Developed QUARC an array-based data-parallel DSL framework on top of LLVM and C++14
- Developed a data-placement algebra and transformations for improved SIMD vectorization and automated MPI code-generation
- Developed a prototype LQCD DSL, QUICQ, based on QUARC that is an order of magnitude faster than QDP++ and within 20% of QPhiX on Xeon and Xeon Phi servers
- QUICQ code acts as a drop-in replacement for the CG solver in MILC