**Vivek Kale**

217-369-7996.[vivek@illinois.edu](mailto:vivek@illinois.edu). <http://vivek112.googlepages.com>.

**U.S. Citizen**

**General Areas of Interest:** Multi-core Architectures, High-Performance Scientific Computing

**Education:**

Ph.D. in Computer Science at University of Illinois at Urbana-Champaign (May 2015 Graduation)

Lawrence Scholar at Lawrence Livermore National Laboratory

GPA: 3.96/4.00

**Advisor:** ProfessorWilliam D. Gropp

**Research Focus:** Improving scientific application performance through dynamic scheduling

***Research Overview:*** I have worked on low-overhead dynamic scheduling strategies for performance tuning MPI+OpenMP codes on multi-core processors. The techniques have been applied to dense matrix factorization codes, specifically Communication-avoiding LU and Communication-avoiding QR. I have applied the strategies to regular mesh computations and Lattice-Boltzmann simulations, and most recently, n-body simulations. The scheduling strategies developed can be beneficial to mitigate the amplification problem, a problem shown to cause serious performance bottlenecks for bulk-synchronous and loosely synchronous MPI applications running on next-generation supercomputers or a cloud-based infrastructure.

**Publications:**

1. Vivek Kale, Simplice Donfack, Laura Grigori, William D. Gropp. ***Balancing the Tradeoff Between Load Balancing and Locality to Improve Performance of Scientific Applications.*** SC' 14. New Orleans, LA.

1. Vivek Kale, Amanda Randles, William D. Gropp. ***Locality-Optimized Mixed Static/Dynamic Scheduling for Load Balancing on SMPs****.* EuroMPI/ASIA 2014. Kyoto, Japan.
2. Vivek Kale, Todd Gamblin, Torsten Hoefler, Bronis R. de Supinski, William D. Gropp. ***Slack-conscious Lightweight Loop Scheduling for Scaling Past the Noise Amplification Problem.*** SC ‘12 Poster. Salt Lake City, Utah.
3. Simplice Donfack, Vivek Kale, Laura Grigori, William D. Gropp. ***Hybrid Static/Dynamic Scheduling for Already Optimized Dense Matrix Factorizations*.** IPDPS 2011. Shanghai, China.
4. Vivek Kale, Abhinav Bhatele, William D. Gropp. ***Weighted Locality-Sensitive scheduling for Noise Mitigation on Multi-core Clusters.*** HiPC 2011. Bangalore, India.
5. Vivek Kale, William D. Gropp. ***Load Balancing for Regular Meshes on a Cluster of SMPs with MPI.*** EuroMPI 2010. Stuttgart, Germany. ***(Selected as a Best Paper).***
6. Torsten Hoefler, James Dinan, Darius Buntinas, Pavan Balaji, Brian Barrett, Ron Brightwell, William Gropp, Vivek Kale, Rajeev Thakur*.* ***MPI+MPI: A New Hybrid Approach to Parallel Programming with MPI Plus Shared Memory*.** EuroMPI 2012. Madrid, Spain.
7. Amanda Randles, Vivek Kale, Jeff Hammond, William D. Gropp, Efthimios Kaxiras. ***Performance Analysis of the Lattice Boltzmann Model Beyond Navier-Stokes.*** IPDPS 2013. Boston, MA.

**Technical Skills Highlights:**

*Languages*: C, C++, python, Fortran, python, bash, csh, VHDL, Matlab, Java

*Tools:* LaTeX, gnuplot, emacs, autoconf, cmake, svn, git

*Libraries for Parallelism:* POSIX threads (Pthreads), MPI (mpich3), OpenMP (gomp), UPC (Berkeley UPC)

*Performance Profiling Tools:*OpenSpeedShop, hpcToolkit, PMPI, Intel VTune

*Platforms*: clusters of NUMA multi-core nodes, clusters of SMP nodes, GPUs, desktop multi-core processors.

**Experience:**

**Lawrence Livermore Nat’l Lab   Lawrence Scholar                     Feb ‘12 – Jun ‘14**

* Obtained measurements for within-node imbalances on laboratory supercomputers, and developed a cost model for these imbalances.
* Created a software system for automated performance optimization and application programmer usability of low-overhead scheduling strategies.
* Developed a theoretical analysis for obtaining best performing scheduling parameters for a particular application and architecture.
* Developed a ROSE-based custom compiler for automatically transforming MPI+OpenMP applications to use my low-overhead scheduling technique and runtime.
* Assessed further opportunities for performance improvement of low-overhead schedulers, including improvement of spatial locality of low-overhead schedulers.

**Lawrence Livermore and Berkeley Nat’l Lab       Scholar  Jun ‘10 - Sept ‘10 / Jun ‘11 - Sept ‘11**

* Experimented with different OpenMP parameters of implemented MPI+OpenMP application code to understand performance optimizations on LLNL supercomputers.
* Analyzed results for the performance tests developed on NERSC machines, and compared with collectives in reference to MPI (mpich2) runtime system.
* Modified OpenMP gomp runtime system in order to integrate low-overhead schedulers within it.

**Goldman-Sachs     Summer Analyst Jun. ‘09 – Sept. ‘09**

* Wrote code for testing trading system infrastructure functions under extreme market conditions.
* Analyzed performance bottlenecks of system infrastructure functions.

**MIT Research Intern Jun. ‘05 – Sept. ‘05**

* Worked under Professor Alex Pentland to write C code to collect data from bio-sensor and audio data.
* Used spectral methods and statistical methods from Matlab to do simple data analysis of obtained data.