Vivek L. Kale, PhD

Phone: +01 1-217-369-7996 . *Email*: vivek.lkale@gmail.com . *Web*: `http://vlkale.github.io`
U.S. Citizen

Education

Bachelor of Science, Computer Science
University of Illinois at Urbana-Champaign, May 2007

Master of Science, Computer Science
University of Illinois at Urbana-Champaign, December 2010

Doctor of Philosophy, Computer Science
University of Illinois at Urbana-Champaign, May 2015
Advisor: William D. Gropp

Honors

- SC 2017 Early Career Program Invitee and Participant
- 2015 Heidelberg Laureate Forum Invitee and Participant.
- Fellow at Lawrence Livermore National Laboratory through Lawrence Scholar Program.

Courses Taken

Designing and Building Scientific Applications, Parallel Software Patterns, Program Optimization, Parallel Computer Architecture, Algorithms, Artificial Intelligence, Software Engineering

Technical Skills

Languages: C, C++, python, Fortran, bash, csh, CUDA

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

Libraries for Parallelism: POSIX threads (Pthreads), MPI (mpich3), OpenMP (gomp, llvm), OpenACC (pgi)

Performance Profiling Tools and Libraries: hpcToolkit, PAPI, nvprof, gprof

Experience

Charmworks, Inc. Software Developer Jun. '18 - Apr. '19

- Collaborating with LLNL (Harshitha Menon) on an LDRD-type proposal for a synergistic loop scheduling and load balancing strategy.
- Worked on making User-defined Loop Scheduling portable across different parallel programming library, done with ORNL (Oscar Hernandez) through DoE Exascale Computing Program.
- Added examples of loop scheduling in OpenMP in Examples section of OpenMP Specification.
- Working on a NSF startup SBIR proposal for loop scheduling for desktop computers.

- Collaborating on developing a proposal to add an OpenMP User-defined Schedule to the OpenMP specification based on an OpenMPCon 2017 paper, presenting a proposal at the OpenMP F2F in Santa Clara and the upcoming F2F in Toronto.
- Currently working on two papers for User-defined Loop Scheduling for publication.
- Assisting with slides for pitch and marketing of Charm++ software, and providing feedback for tutorials on Charm++.
- Integrating a shared memory library for sophisticated loop scheduling strategies, including some based on my dissertation, into the current version of Charm++.

University of Southern California Computer Scientist Dec. '16 - Jun '18

- Working with postdoc from LLNL on a proposal to study techniques that combine loop scheduling and load balancing to improve performance of scientific applications.
- Working with OpenMP Language Committee to support user-defined loop schedules in OpenMP.
- Translating an x-ray tomography code written in Matlab code to C code and then parallelizing it to run on a supercomputer having nodes with GPGPUs.
- Working on modifications to LLVM compiler to support new OpenMP loop schedules.
- Worked on ensuring external network infrastructure to support transfer of application code's input data files were adequate for an application code's efficient execution using the Globus Toolkit.
- Working in team to manage computational performance aspects of running an application program involving Fast Fourier Transformation and image reconstruction algorithms.

Charmworks, Inc. Developer Jan. '16 - Nov. '16

- Implemented mixed static/dynamic loop scheduling strategies within Charm++'s thread scheduling library.
- Helped to improve portability of Charm++ to a variety of platforms.
- Assisted with business aspects of a high-tech startup.

University of Illinois Postdoctoral Associate Jul. '15 – Dec. '15

- Developed library that allows application programmers to use strategies from dissertation.
- Adapted a plasma physics application code to work on a GPGPU processor and Intel Xeon Phi.
- Incorporated over-decomposition and locality-aware scheduling into strategies from dissertation.

Lawrence Livermore Nat'l Lab Lawrence Scholar Feb. '12 – Jun. '14

- Measured MPI communication delays for micro-benchmarks codes run on supercomputers and worked to find tools to measure dequeue overheads of OpenMP loop schedulers.
- Created a software system for automated performance optimization and application programmer usability of low-overhead hybrid scheduling strategies.
- Developed a ROSE-based custom compiler for automatically transforming MPI+OpenMP applications to use low-overhead scheduling techniques and runtime.
- Assessed further opportunities for performance improvement of low-overhead schedulers, including improvement of spatial locality of low-overhead schedulers.

Lawrence Livermore Nat'l Lab Scholar Jun. '11 - Sep. '11

- Experimented with different OpenMP parameters of implemented MPI+OpenMP application code to understand performance optimizations on LLNL supercomputers.

- Developed software design for low-overhead loop scheduling library based on libgomp software design.

Lawrence Berkeley Nat'l Lab Summer Scholar Aug. '10 - Sep. '10
- Analyzed results for the performance tests developed on NERSC machines.
- Compared with collectives in reference to MPI (mpich2) runtime system.

Lawrence Livermore Nat'l Lab Scholar May. '10 - Aug. '10
- Modified libgomp runtime system in order to integrate low-overhead schedulers within it.
- Developed an algorithm multi-stage low-overhead loop scheduler with each stage associated with a level in the memory hierarchy, allowing for MPI-shared memory extensions to be used in conjunction with the low-overhead loop scheduling strategies.

Goldman-Sachs Summer Analyst Jun. '09 – Sep. '09
- Wrote code for testing trading system infrastructure functions under extreme market conditions.
- Analyzed performance bottlenecks of system infrastructure functions.

Proteus Technologies, LLC Software Developer Aug. '07 – Apr. '08
- Primarily responsible for developing, testing and documenting a service-oriented software application for health and status monitoring of large-scale parallel and distributed networked systems.
- Developed company standards for software development (System Requirements Specifications, Design Documentation).
- Designed and implemented algorithms for cost optimization applications. Used dynamic programming, discrete optimization heuristics, and APIs.

List of Publications

Papers

1. Vivek Kale and William D. Gropp. ***Composing Low-Overhead Scheduling Strategies for Improving Performance of Scientific Applications***. IWOMP 2015. October 2015. Aachen, Germany.
2. Simplice Donfack, Vivek Kale, Laura Grigori and William D. Gropp. ***Hybrid Static/Dynamic Scheduling for Already Optimized Dense Matrix Factorizations***. IPDPS 2012. May 2012. Shanghai, China.
3. Vivek Kale, Abhinav Bhatele and William D. Gropp. ***Weighted Locality-sensitive Scheduling for Noise Mitigation on Multicore Clusters***. HiPC 2011. December 2011. Bangalore, India.
4. Vivek Kale and William D. Gropp. ***Load Balancing for Regular Meshes on a Cluster of SMPs with MPI***. EuroMPI 2010. September 2010. Stuttgart, Germany. *(Selected as a Best Paper)*
5. Torsten Hoefler, James Dinan, Darius Buntinas, Pavan Balaji, Brian Barrett, Ron Brightwell, William Gropp, Vivek Kale and Rajeev Thakur. ***MPI+MPI: A New Hybrid Approach to Parallel Programming with MPI Plus Shared Memory***. EuroMPI 2012. September 2012. Madrid, Spain.
6. Amanda Randles, Vivek Kale, Jeff Hammond, William D. Gropp and Efthimios Kaxiras. ***Performance Analysis of the Lattice Boltzmann Model Beyond Navier-Stokes***. IPDPS 2013. May 2013. Boston, USA.

7. Vivek Kale. ***Towards Using and Improving the NAS Parallel Benchmarks: A Parallel Patterns Approach***. ParaPLoP 2010. April 2010. Carefree, USA.
8. Vivek Kale and Edgar Solomonik. ***Parallel Sorting Pattern***. ParaPLoP 2010. April 2010. Carefree, USA.
9. Vivek Kale. ***The Correlation between Parallel Patterns and the NAS Parallel Benchmarks***. ICSE 2010. May 2010. Johannesberg, South Africa.

Extended Abstracts

1. Vivek Kale and Martin Kong. ***Enhancing Support in OpenMP to Improve Data Locality in Application Programs Using Task Scheduling***. OpenMPCon 2018. September 2018. Barcelona, Spain.
2. Vivek Kale and William D. Gropp. ***A User-defined Schedule for OpenMP***. Extended Abstract. OpenMPCon 2017. September 2017. New York, USA.
3. Vivek Kale. ***A Pattern Language for Dynamic Scheduling***. ParaPLoP 2011. May 2011. Carefree, USA.

Posters

1. Vivek Kale and Oscar Hernandez. ***Performance Portability of User-defined Loop Schedules***. DOE PPP 2019. April 2019. Denver, USA.
2. Vivek Kale, Harshitha Menon and Karthik Senthil. ***Adaptive Loop Scheduling with Charm++ to Improve Performance of Scientific Applications***. SC 2017. November 2017. Denver, USA. *(Selected as a Candidate for Best Poster)*
3. Vivek Kale, Simplice Donfack, Laura Grigori and William D. Gropp. ***Balancing the Trade-off Between Load Balancing and Locality to Improve Performance of Scientific Applications***. SC 2014. November 2014. New Orleans, USA.
4. Vivek Kale, Amanda Randles and William D. Gropp. ***Locality-Optimized Mixed Static/Dynamic Scheduling for Load Balancing on SMPs***. EuroMPI/ASIA 2014. September 2014. Kyoto, Japan.

Membership of Organizations

- Society for Industrial and Applied Mathematics
- Association for Computing Machinery
- Institute of Electrical and Electronics Engineers
- Association for Computing Machinery's SIGHPC

Teaching Experience

1. Teaching Assistant for Programming Studio (CS 242) at University of Illinois at Urbana-Champaign for Spring 2009.
2. Teaching Assistant for Programming Studio (CS 242) at University of Illinois at Urbana-Champaign for Fall 2009.
3. Teaching Assistant for Programming Studio (CS 242) at University of Illinois at Urbana-Champaign for Spring 2010.

Services

- Reviewer for 2017 Elsevier Parallel Computing Journal
- Member of Organization Committee for SC 2018's Early Career Program
- Member of Selection Committee for SC 2018's Experiencing HPC for Undergraduates Program