Vipul Harsh

Department of computer Science
University of Illinois at Urbana-Champaign
Urbana, IL - 61801

Phone: (217) 751-2907
Email: vharsh2@illinois.edu
URL: vharsh2.web.engr.illinois.edu

Interests Parallel Computing, Systems

Education

Masters, Computer Science
University of Illinois at Urbana-Champaign, May 2017 (Expected)
GPA: 3.89/4

B.Tech. (Honors), Computer Science and Engineering
Indian Institute of Technology, Bombay, May 2015
GPA: 9.17/10

Research Experience

Graduate Research Project, UIUC — *March 2016 - Present*
Parallel Histogram Sort using sampling — Prof. Laxmikant Kale

Designed a parallel sorting algorithm that provably ensures arbitrary good load balance upto a factor of $(1+\epsilon)$, where ϵ is provided as an input parameter. Showed that the sample size in sample sort can be reduced from $O(p \log p/\epsilon^2)$ to $O(p \log p/\epsilon)$ by doing one round of histogramming and can be further brought down to $O(p \sqrt[k]{\log p/\epsilon})$ with k rounds of histogramming. Histogram sort with sampling is theoretically more efficient than sample sort. For experiments, implemented node-level optimisations to the algorithm, taking advantage of shared memory programming to make it more scalable on large clusters.

Graduate Research Project, UIUC — *March 2016 - Present*
Collective Operations in charm — Prof. Laxmikant Kale

Working on improving communication latencies for collective operations (like gather, scatter) in charm. Improving multicast functionalities in charm for MPI like sub-communicator sections.

Research Internship, Georgia Tech — *Summer 2014*
Fast Multipole method for RPY tensor — Prof. Edmond Chow

Developed two methods for doing large scale simulations for polydisperse particle systems involving hydrodynamic interactions and RPY tensor. Extended the 4 call method for polydisperse systems involving 5 calls to the harmonic FMM. Used the Kernel Independent FMM method to run simulations on multiple cores and achieved ∼6x speedup with 24 cores.

Research Internship, LaBRI, France — *Summer 2013*
Revisiting the Karp and Miller Algorithm — Prof. Jerome Leroux, Prof. Gregoire Sutre

Researched on the Karp and Miller algorithm to compute the coverability set of a Petri Net and other improvements namely the MP algorithm and the buggy Finkel algorithm. Built a tool that implements the aforementioned algorithms.

Achievments

- Represented IIT Bombay at the ACM ICPC World Finals 2015. Highest ranked team from India
- All India Rank 49 in IIT-JEE 2011, among 500,000 candidates
- Awarded A+ grades in courses: Algorithms (UIUC), Machine Learning (UIUC), Numerical Analysis (IITB) and Differential Equations (IITB)
- Rank 1 in 3^{rd} International Mathematics Olympiad, 2009 conducted by Science Olympiad Foundation
- Certified as among Top 1% (300 students) in India, to appear for the following Indian National Olympiads: Maths (INMO) 2011; Astronomy (INAO) 2009, 2011

Other Projects

Improving communication latencies in charm++ using RDMA operations *March 2016 - Present*
Guide: Prof. Laxmikant Kale
Worked on reducing message sending times for charm by having an API that avoids copies for large messages using RDMA onesided operations provided by the underlying network. Achieved upto 37% improvement for large messages on BG/Q machines.

Preventing Overfitting in Machine Learning Classifiers *Fall 2015*
Guide: Prof. Dan Roth
Explored the properties of Dropout by modifying the training algorithm and analyzed change in performance. Extended the Dropout method for neural networks to other learning classifiers like Perceptron and Support Vector Machines.

Virtual Memory for Experimental OS *Spring 2014*
Guide: Prof. Dhananjay M. Dhamdhere
Designed and implemented effective data structures and algorithms for handling process memory allocation, swap space management, with process swap in and out for Pranali, a virtual OS built on top of Linux.

Comparison of binary exchange and transpose algorithms for FFT *Spring 2016*
Guide: Prof. Marc Snir
Implemented two algorithms: the binary exchange and the transpose algorithm for performing FFT in parallel. Came up with performance models for both and compared them with experimental results. Also compared performance with fftw library.

Teaching

- Teaching Assistant for the course Discrete Mathematics for Autumn Semester, 2013, IIT Bombay
- Guided over 300 students in a 3 day long hands-on GPU Programming and Applications Workshop (GPA) conducted by NVIDIA in association with CUDA Center of Excellence, IIT Bombay

Technical Skills

Programming: C, C++, Java, Python, MPI, Matlab, Charm++
Scripting: Bash, Slurm, Matplotlib, HTML
Miscellaneous: LaTeX, Scheme, Prolog