Rahulkumar Gayatri

skype : rahulkumar.gayatri
email : rahulgayatri84@gmail.com
phone-number : 19253848354

Summary

- Currently a Postdoc at Lawrence Bekeley National Lab (LBNL). I work in the "Application Readiness for Exascale Architectures" project in the NERSC department. I have been involved with two projects:

  - I write performance portable application kernels using programming models such as OpenMP3.0, OpenMP4.5 (for GPU's), Cuda, Kokkos and RAJA.

  - I work on the SW4 project which is a Seismic code that simulates the effects of an earthquake. My work is to improve the performance of the code on the Intel-knl processor.

- I have worked on the Moose project, a simulation model of neural connections in human brain.

  - My role was to parallelize the ODE solvers used to simulate the electrical and chemical interactions between neurons.

- Experience in the areas of compiler and runtime development for parallel programming models.

  - Introduced new compiler directives and the necessary runtime support in the OMPSs framework to handle synchronization of multiple threads.

- Knowledge and experience in the area of Transactional Memory framework.

  - Worked extensively with the TinySTM library.

- Experience in sequential and parallel algorithm development.

  - Designed and implemented a Breadth First Search (BFS) algorithm that takes advantage of low memory on IBM's Cell B/E. processor.

  - Parallelized Graph500 benchmarks, SPECFEM3D, linear iterative solvers on an SMP machine using the OMPSs programming model.

- Experience in porting applications using parallel programming models such as OpenMP4.5, Kokkos, OMPSs, MPI, Pthreads.
- Experience in exploiting the underlying processor arhitecture to enchance the application performance.
- Experience in working with profiling and analysis tools such as Intel-advisor, Intel-vtune, LIKWID, Intel-SDE, valgrind.

Education

1. **Doctor of Philosophy (PhD) in Computer Science** (2015)
   Barcelona Supercomputing Center
   Thesis Title - "Increasing Parallelism through Speculation in Task-Based Programming Model."

2. **Master of Technology** (2009)
   Specialization in Computer Science, Sri Sathya Sai University (SSSU)

3. **Master of Science (MTech)** (2007)
   Specialization in Computer Science, SSSU
4. **Bachelor of Science** (2005)
   Specialization in Mathematics, SSSU

Technical Skills

Programming Languages:	C,C++ , python
Programming Models:	OpenMP4.5, Cuda, Kokkos, OMPSs, OpenMP3.0, Pthreads, MPI, STM
Scripting:	Shell, LaTeX, Sed, awk, gnuplot
Profiling Tools:	Intel-vtune, Intel-advisor, Intel-SDE, LIKWID, Nvidia Visual Profiler, Valgrind

Publications

1. *Tuomas Koskela, Zakhar Matveev, Rahulkumar Gayatri, et all* "A Novel Multi-Level Integrated Roofline Model Approach for Performance Characterization."
   Preleminarily accepted for publication at ISC2018, Frankfurt, Germany.
2. *Rahulkumar Gayatri, Rosa M. Badia, Eduard Ayguade* "Loop level speculation in a task based programming model."
   20th Annual International Conference on High Performance Computing, Bangalore, 2013, pp. 39-48.
3. *Rahulkumar Gayatri, Rosa M. Badia, Eduard Ayguade* "Transactional access to shared memory in StarSs, a task based programming model."
   Euro-Par 2012 Parallel Processing, pp. 514-525
4. *Rahulkumar Gayatri, Rosa M. Badia, Eduard Ayguade* "Analysis of the overheads incurred due to speculation in a task based programming model."
   MULTIPROG-2015: proceedings of 8th Workshop on Programmability Issues for Heterogeneous Multicores. Amsterdam: 2015, p. 1-12.
5. *Roberto Giorgi et al.* "TERAFLUX: Harnessing dataflow in next generation teradevices."
   Microprocessors and Microsystems, Volume 38, Issue 8, pp. 976-990
6. *Rahulkumar Gayatri, Rosa M. Badia, Eduard Ayguade* "Presented a Poster on the benefits of using CellSs (a programming model for Cell Processor) in the ACACES 2010 summer school of HiPEAC."
7. *Rahulkumar Gayatri,Pallav Baruah* "Parallelizing Breadth First Search Using Cell BE, HiPC, Student Symposium, 2008"

Projects

1. **Berkeley GW** - A material science kernel, that predicts the excited state properties of a wide range of materials.. The aim is to port the kernel using different programming models such as OpenMP3.0, OpenMP4.5 (for GPU's using the target directives), Kokkos, RAJA and OpenCL. We compare the performance of the kernel ported using the above mentioned programming models to the best known implementations of the kernel for the specific architecture. The goal is to test the programmability, performance and portability of these frameworks. We are currently running this kernel on machines such as Cori at LBNL (compsising of haswell and KNL processors) and summitdev at Oakridge National lab comprising of 4 Nvidia Pascal GPU's on a single node. We have a fortran and a C++ version of the kernels to compare the performance between these programming languages. We catalog the results obtained, lessons learnt and the experiences gained from this project so that the future users might benefit from them. http://performanceportability.org/case_studies/gw/. It

is an open source code which can be found here : https://github.com/rahulgayatri23/BGW-Kernels

2. **SW4** - Seismic wave code of 4th order. It simulates the effects of an earthquake, https://github.com/geodynamics/sw4. My role in the project is to optimize the code for the Knights Landing architecture from Intel. For this I use techniques such as vectorization, cache-blocking, reducing OpenMP overhead.
3. **MOOSE** - The simulation environment uses various ODE system solvers to understand the chemical and electrical interactions inside a cell. I worked on parallelizing the ODE solvers that simulate the behavior of the cell over multiple timesteps. I have parallelized the kinetic and stochastic solvers that solve a system of linear equations using the Runge-Kutta method of order 5. Kinetic solver achieved a 2.3X speedup with 4 threads whereas the stochastic solver gained a 3.6X speedup.
4. **Doctoral Thesis** - Focussed in the area of parallel programming models, specifically on providing compiler and runtime support for synchronization of multiple threads in StarSs. The synchronization was achieved using TinySTM, a Software Transactional Memory Library (STM). This approach along with improving the performance and the efficiency also offers an opportunity to exploit higher degree of parallelism from an application. Papers published in this project: [1],[2] and [3].
5. **StarSs** - A task-based programming model to make parallel programming easier. It consists of compiler directives and the required runtime support. My contribution to the project was to maintain the runtime framework and resolve conflicts when new directives and their required implementation were introduced. I also worked on design and implementation of parallel applications using the framework for the application repository.
6. **Teraflux** - It was a project supported and funded by European Union which focused on exploiting dataflow parallelism in a Teracomputing device. My contribution to the project was to introduce STM-based concurrency to handle simultaneous access to shared memory. Papers published in this project: paper [4].
7. **MTech Thesis** - An efficient Breadth First Search (BFS) implementation that exploits memory locality in the IBM's Cell.B.E architecture. Poster[5] presented the results achieved in this project.

Professional Career

- * Postdoc at Lawrence Berkeley National Lab.
- Technical Specialist, High Performance Computing (HPC), Wipro Infotech August, 2015 - September 2016
- Doctoral student at Barcelona Supercomputing Center - September, 2009 - March, 2015

Honors

Received a Pre-Doctoral scholarship, FI AGAUR grant, by Generalitat de Catalunya

References

- Jack Deslippe, Application Performance Specialist, Acting Group Leader, NERSC. JRDeslippe@lbl.gov
- Hans Johansen, Computer Systems Engineer, Computational Research, Berkeley Lab. HJohansen@lbl.gov *Phone: +1 510 495 2472*

- Rosa Maria Badia, WORKFLOWS AND DISTRIBUTED COMPUTING GROUP MANAGER, Barcelona Supercomputing Center. rosa.m.badia@bsc.es
Phone: +34 934134075