WHPC
WOMEN IN HIGH PERFORMANCE COMPUTING

Women in HPC: Changing the Face of HPC

Friday 20 November 2015
Hilton Hotel, 500 E 4th St, Austin, TX 78701, United States

In collaboration with Supercomputing 2015

epcc

EPSRC
Engineering and Physical Sciences Research Council

archer

Welcome to the Women in HPC workshop at SC15

Two years ago when EPCC at the University of Edinburgh attended Supercomputing 2013 to identify the best way to set up a Women in HPC group for the UK we had no idea that this would result in hosting a series of international workshops, bringing together women from around the world to discuss gender in HPC or setting up international chapters. Two years on we are delighted to be hosting our third international workshop, the second to be held in collaboration with Supercomputing, showcasing the talent of women from around the world.

The importance of diversifying any field is now well understood as researchers and organizations have documented improved performance, research output, and innovation in heterogeneous teams. In addition, today's rapidly changing global industry demands a highly skilled technical workforce, which cannot be achieved without the 50% of the population that women comprise. However, gender bias remains a key problem in the workforce: in the UK women make up 55% of university graduates yet earn on average £140,000 less than their male colleagues over their lifetimes. In USA and Australia the pay gap for full-time employed women compared to their male counterparts is between 17 and 18%. In HPC women make up between 5 and 17% of the community depending on what is measured.

Your participation in today's workshop and commitment to broadening participation in HPC will help ensure that women have access to role models, are better prepared to fulfil technical positions in the HPC industry and help us all understand what the HPC community can do to encourage the next generation of HPC professionals.

Now more than ever is the time for us to be bold, work together and change this industry making it better for all.

Welcome to Women in HPC in Austin!

Toni Collis
Women in HPC at SC15 Chair

Contents

Agenda

Friday 20 November 2015, 8.30–12.00
Salon A, Hilton Hotel, Austin, Texas, USA

8.30–8.35	**Welcome** Alison Kennedy & Toni Collis (WHPC network, EPCC, UK)
8.35–9.10	**Measuring Progress Towards Sustained Equality and Diversity in HPC** Toni Collis (WHPC network, EPCC, UK) and Lorna Rivera (I-STEM, University of Illinois at Urbana-Champaign)
9.10–9.45	**Panel 1: Career progression** Chair: Toni Collis, EPCC
9.45–10.00	**Poster presentations**
10.00–10.30	**Coffee and posters**
10.30–10.55	**Hot topic 1** **ExTASY: Accelerating molecular shapes discovery in biomolecular simulation** Ardita Shkurti (School of Pharmacy, Centre for Biomolecular Sciences, The University of Nottingham)
10.55–11.20	**Hot topic 2** **Using parallel simulation in the co-design of HPC networks and distributed data-intensive science facilities** Misbah Mubarak (Mathematics and Computer Science Division (MCS), Argonne National Laboratory)
11.20–12.00	**Panel 2: Improving Diversity at SuperComputing** Chair: Trish Damkroger, Lawrence Livermore National Laboratory
12.00	**Close**

Measuring Progress Towards Sustained Equality and Diversity in HPC

Toni Collis

WHPC network, EPCC, UK
toni.collis@ed.ac.uk

Toni Collis is the coordinator and founding member of the Women in HPC (WHPC) network in the UK, as well as an Applications Consultant in HPC Research and Industry at Edinburgh Parallel Computing Centre (EPCC), UK. Within EPCC her role includes providing technical expertise on a range of research projects using HPC in academic software, from engineering to biology. Prior to working at EPCC Toni gained a PhD in computational condensed matter as well as an MSc in HPC and an MPhys in Mathematical Physics. Toni's current role for WHPC involves running the WHPC network and events, as well as research into diversity in the HPC community, how it can be improved and the unique problems faced by this field. She has been on the organising committee for a variety of workshops and conferences including the first WHPC workshop at SC14.

Lorna Rivera

I-STEM, University of Illinois at Urbana-Champaign
lirivera@illinois.edu

Lorna Rivera serves as an I-STEM Senior Research Specialist working on a nationwide project funded by the National Science Foundation: Extreme Science and Engineering Discovery Environment (XSEDE). XSEDE seeks to help scientists and engineers around the world use their collection of integrated advanced digital resources and services to advance research in order to make us all healthier, safer, and better off.

Lorna Rivera received her Bachelor of Science in Health Education and her Master of Science in Health Education and Behavior from the University of Florida. In 2011, she was certified as a Health Education Specialist by the National Commission for Health Education Credentialing, Inc. Prior to joining I-STEM in Illinois, Lorna worked with various organizations, including the March of Dimes, Shands HealthCare, and the University of Florida College of Medicine. Her research interests include the evaluation of innovative programs and their sustainability, and health education and promotion programs.

Abstract

The U.S. Department of Labor projects that by 2018 nearly 1.4 million computing job openings will exist in the U.S. alone, and yet only 61% of those jobs could be filled by U.S. computing degree-earners. Women naturally comprise approximately 50% of the global population, yet just 18% of Computer Science bachelor degrees in the US were awarded to women in 2014. HPC users typically identify as domain scientists in fields like physics, biology, chemistry, and even humanities. Due to this broad pool of domain specialists the lack of women in Computer Science should not be an impediment to female participation in HPC. As the principle aim of the Women in HPC initiative is to encourage participation of women in the HPC community, it is important to provide a baseline of the current HPC community. Our initial analysis is not promising – between 5% and 17% of the HPC community is female, lower than the proportion of women emerging with Computer Science degrees. Identifying this baseline, however low it may be, is the only way we'll be able to measure our progress over time and ultimately move towards a more diverse and equitable community together.

In addition to measuring the demographic distribution of HPC over time, Women in HPC is working with the University of Illinois at Urbana-Champaign's Illinois Science, Technology, Engineering, and Mathematics Education Initiative (I-STEM) to better understand why so many women do not engage with HPC. There is an extensive body of research addressing the lack of female representation in a number of areas, particularly STEM disciplines. While best practices from this work has increased participation of underrepresented groups including women in STEM, many do not address issues specific to HPC. Women in HPC's work with I-STEM uniquely contributes to this body of work by focusing on the HPC community. Preliminary studies take advantage of Compute Canada, PRACE, RIKEN and XSEDE's collaborative annual International Summer School on HPC Challenges Preliminary findings with selected participants demonstrate the importance of language when assessing self-reported skills in HPC. For example, when I-STEM asked participants to self-rate their level of abilities in HPC tools and techniques on a 5 point likert scale of 'novice' to 'expert,' men often scored their abilities statistically significantly higher than women. In contrast, when participants self-rated their usage frequency of similar HPC tools and techniques on a 5 point likert scale of 'never' to 'almost always,' no gender differences were found. This difference was also noted by application reviewers who perceived application quality to be somewhat dependent on applicant gender. Finally, attendees of the 2015 summer school were directly assessed in general parallelism concepts, during which no significant gender differences were found. If gender differences in self-reported abilities is pervasive across HPC, it has the potential to significantly impact women's success and opportunities towards career progression regardless of actual ability thus impeding innovation and research output in HPC. In this talk we will further discuss these results, the impact they may have on the HPC community and how we can all work to foster equity and diversity in this important field.

ExTASY: Accelerating molecular shapes discovery in biomolecular simulation

Ardita Shkurti

School of Pharmacy, Centre for Biomolecular Sciences,
The University of Nottingham
Ardita.Shkurti@nottingham.ac.uk

Ardita is part of the Molecular Recognition group of the School of Pharmacy at the Centre for Biomolecular Sciences since August 2013 working as a research fellow funded by a joint EPRSC/NSF grant. She has a core expertise concerning Computer Science Engineering (BSc and MSc in Computer Science Engineering at respectively Università degli Studi di Modena e Reggio Emila and Politecnico di Torino, and PhD in Information and Systems Engineering at Politecnico di Torino).

Ardita is interested to investigate and apply methods and algorithms from branches of computer science and mathematics to any “big data” field of biomolecular simulation, maximally exploiting the potential that recent High Performance Computing (HPC) architectures and programming paradigms have to offer.

Abstract

The advances in technology and computation over the last thirty years have been extensively leveraged by the biomolecular simulation community that has produced exceptionally performant molecular simulation engines such as AMBER[1], GROMACS[2], NAMD[3] and even specific supercomputing hardware such as the ANTON[4] supercomputer. Hence, huge quantities of molecular simulation data are being generated daily. This data needs to be subsequently analysed and interpreted properly in order to obtain insights into the molecular dynamics and biological processes that have been simulated. However there is a computational imbalance between the highly optimised molecular simulation engines and the much less developed tools for analysis of the data produced by these engines. This is becoming a bottleneck severely affecting the timescales of new discoveries in computational chemistry and consequently of technological processes such as drug design and development.

The biomolecular simulation community is now making particular efforts to address this imbalance. ExTASY[5], an EPSRC/NSF funded project, has been part of these efforts and has provided the biomolecular simulation community with an extensible software framework for advanced molecular simulation analysis and, in particular, the analysis of molecular shapes. Molecular systems such as proteins change their conformations (three-dimensional shapes) when binding or unbinding to other molecules making the relation between molecular shape and function a crucial aspect to investigate, and something that is uniquely accessible through computational simulation.

The ExTASY framework has specifically been designed to be compatible with high performance computing (HPC) hardware architectures and has been so far successfully tested on international HPC facilities such as ARCHER and XSEDE in addition to the local HPC facilities of the affiliations of the ExTASY research scientists.

CoCo-MD, one of the ExTASY workflows, will be showcased. CoCo-MD automates the alternation of molecular simulations and trajectory data analysis using the CoCo[6] method to steer on-the-fly the molecular simulation process in order to promote the simulation of rare shapes of the biomolecular system. Thus, it provides a faster way than traditional molecular simulation to explore the shapes space of the simulated molecular system.

The implementation of CoCo-MD for supercomputing contexts, including resource allocation, data management and task execution is handled by ExTASY relying on a pilot framework – Radical Pilot[7], to run a large number of tasks concurrently.

Examples of comparison between CoCo-MD and traditional molecular simulation approaches for the exploration of the molecular shapes space will be provided highlighting the performance of CoCo-MD.

References

[1] D.A. Case, J.T. Berryman, R.M. Betz, D.S. Cerutti, T.E. Cheatham, III, T.A. Darden, R.E. Duke, T.J. Giese, H. Gohlke, A.W. Goetz, N. Homeyer, S. Izadi, P. Janowski, J. Kaus, A. Kovalenko, T.S. Lee, S. LeGrand, P. Li, T. Luchko, R. Luo, B. Madej, K.M. Merz, G. Monard, P. Needham, H. Nguyen, H.T. Nguyen, I. Omelyan, A. Onufriev, D.R. Roe, A. Roitberg, R. Salomon-Ferrer, C.L. Simmerling, W. Smith, J. Swails, R.C. Walker, J. Wang, R.M. Wolf, X. Wu, D.M. York and P.A. Kollman (2015), AMBER 2015, University of California, San Francisco.

[2] Szilárd Páll, Mark James Abraham, Carsten Kutzner, Berk Hess, Erik Lindahl. Tackling Exascale Software Challenges in Molecular Dynamics Simulations with GROMACS. Solving Software Challenges for Exascale Lecture Notes in Computer Science Volume 8759, 2015, pp 3-27

[3] James C. Phillips, Rosemary Braun, Wei Wang, James Gumbart, Emad Tajkhorshid, Elizabeth Villa, Christophe Chipot, Robert D. Skeel, Laxmikant Kale, and Klaus Schulten. Scalable molecular dynamics with NAMD. Journal of Computational Chemistry, 26:1781-1802, 2005.

[4] Shaw D.E., Grossman J.P., Bank J.A., Batson B. et al. Anton 2: Raising the Bar for Performance and Programmability in a Special-Purpose Molecular Dynamics Supercomputer. International Conference for High Performance Computing, Networking, Storage and Analysis, SC14. 2014, 41-53, New Orleans, LA. DOI: 10.1109/SC.2014.9

[5] http://www.extasy-project.org/

[6] Laughton C.A., Orozco M. and Vranken W.. COCO: A simple tool to enrich the representation of conformational variability in NMR structures, PROTEINS, 75, 206-216 (2009), DOI: 10.1002/prot.22235

[7] http://radicalpilot.readthedocs.org/en/latest/

Using parallel simulation in the co-design of HPC networks and distributed data-intensive science facilities

Misbah Mubarak

Mathematics and Computer Science Division, Argonne National Laboratory
mmubarak@anl.gov

Misbah Mubarak is a postdoctoral researcher in the Mathematics and Computer Science Division of Argonne National Laboratory. She received her Ph.D. and M.S. from Rensselaer Polytechnic Institute in 2015 and 2011 respectively. She also has experience in developing multitier enterprise applications at CERN, Switzerland and Teradata Corporation. She is the recipient of U.S. Fulbright scholarship and a finalist for Google Anita Borg scholarship. Her research interests are modeling and simulation of HPC and data-intensive systems.

Abstract

Modeling and simulation have become increasingly powerful tools to co-design cyber infrastructure for various fields of science including extreme-scale systems and distributed data-intensive science facilities. Today's high-end HPC systems are complex with tens of thousands of compute nodes and hundreds of thousands of processors. On the distributed data-intensive front, science facilities are processing petabytes to Exabyte of data. The research community is relying on parallel discrete-event simulation (PDES) to accurately simulate such large-scale systems with high-fidelity in a reasonable amount of time. PDES of such large-scale systems and workloads require ensemble runs and generate billions of events, which is infeasible to process sequentially or using traditional conservative scheduling. In this work, we present how we apply our parallel simulation framework, CODES [1], to two important classes of problems: co-design of HPC systems interconnects and distributed data-intensive facilities for High-Energy physics. CODES leverages the optimistic event scheduling ability of ROSS to enable efficient and detailed large-scale simulations [2].

HPC Interconnect Co-design: The performance of extreme-scale systems relies on the effectiveness of the underlying interconnect network topology. Using the CODES simulation framework, we have explored the design space of two classes of HPC networks at projected exascale system sizes (1) the low-diameter, low-latency dragonfly network that has been used in recent Cray HPC offerings such as the XC30 and (2) torus networks that have been widely used in IBM Blue Gene architectures. Our torus and dragonfly models efficiently simulate millions of network nodes with detailed, flit-level fidelity. The simulations were executed at Argonne's Mira Blue Gene/Q and Rensselaer's Amos Blue Gene/Q systems where they attained an event rate of up to 1.33 billion events/seconds [3,4,5]. Such full-resolution network simulations provide network designers a capability to analyze the network performance and its various configurations prior to building the actual system.

Data intensive HPC facilities: We are now working to apply the PDES techniques proven in exascale network co-design to address the design challenges in HEP data facilities, which rely on management, processing and storage of large amounts of data. The Energy Frontier and Intensity Frontier experiments at Fermi National laboratory are generating many petabytes of data per year. These frontiers rely on the performance of their distributed data management and access systems to serve a global community of scientists [6]. Using the CODES simulation framework, we are developing an end-to-end simulation of real HEP workflow execution on Fermilab's petabyte-scale distributed data processing pipeline. To ensure a realistic simulation of the data facility, we will validate our model against Fermilab's real data facility logging information. Through this simulation, we will explore the data movement policies as well as software and hardware upgrade options. Specifically, we will explore the archival storage cache policies that will enable longer cache life times in such large-scale scientific data stores. The same simulation technology will also be used to quantify the value of adding new storage hardware. A validated model of this data center simulation can further be applied to explore data access in HEP workflows from HPC systems.

References

[1] CODES: Enabling co-design of Exascale Storage Architectures and distributed data-intensive science facilities, http://www.mcs.anl.gov/research/projects/codes/

[2] P. D. Barnes, C. D. Carothers, D. R. Jefferson, and J. M. LaPre, "Warp speed: executing time warp on 1,966,080 cores," in Proc. of the 2013 ACM SIGSIM Conf. on Principles of Advanced Discrete Simulation (PADS), May 2013, pp. 327–336.

[3] M. Mubarak, C. D. Carothers, R. B. Ross, and P. Carns, "Modeling a million-node dragonfly network using massively parallel discrete-event simulation," in High Performance Comput., Network- ing, Storage and Anal. (SCC) SC Companion, 2012, pp. 366–376.

[4] M. Mubarak, C. D. Carothers, R. B. Ross, and P. Carns, "Using massively parallel simulation for MPI collective communication modeling in extreme-scale networks," in Proc. of the 2014 Winter Simulation Conf., 2014, pp. 3107–3118.

[5] M. Mubarak, C. D. Carothers, R. B. Ross, and P. Carns, "A case study in using massively parallel simulation for extreme-scale torus network codesign," in Proc. of the 2nd ACM SIGSIM/PADS Conf. on Principles of Advanced Discrete Simulation, 2014, pp. 27–38.

[6] "Fermilab's multi-petabyte scalable mass storage system", Gene Oleynik et al. In proceedings of 22nd IEEE/13th NASA Goddard Conference on Mass Storage Systems and Technologies, 2005.

Synchronization Methods for Distributed Agent Based Models

Christine Harvey

Modeling and Simulation Engineer, MITRE Corporation

Christine Harvey specialises in data analysis and high performance computing in simulation. Her research interests include HPC in simulation, Agent Based Modeling, and Big Data Analysis particularity in the field of Healthcare. She completed her Masters in Computational Science from Stockton University in 2013 and is currently working on her PhD in Computational Science and Informatics at George Mason University. Christine is an advocate for diversity and inclusion in STEM through Student Volunteer and Broader Engagement Programs.

Abstract

Distributed computing has the ability to facilitate very large scale agent-based models by addressing the traditional scaling limitations. However, reliable and efficient communication strategies are needed to synchronize agent states across multiple processors. The protocol used to synchronize agent states across pro- cessors has a significant impact on the efficiency of the tool. Traditional man- agement methods are conservative in nature and perform a complete synchro- nization of all agent state information at every time step of the simulation. An alternative approach is proposed which uses an event-driven technique to syn- chronize agents. This procedure only synchronizes changes to pertinent infor- mation in the model at each time step. This technique requires less information to be broadcast which reduces the run time of the simulation while maintaining consistency in the model.

A Massive-Scale Agent-Based Modeling framework was developed in Python using MPI capabilities to support distributed agents. The framework was de- signed using both synchronization techniques: conservative and event-driven. A sample rumor model was used in this experiment, where each agent had exactly two assigned neighbors which were distributed according to the probability of having a neighbor on a foreign processor. Trials were run with 20 million and 80 million agents total with 4, 8, 16 and 32 processors. The completed timing analysis shows that the event-driven technique is significantly faster than the conservative method. In addition to the overall speed increase, the event-driven technique also scales up in both number of processors and number of agents per processor more efficiently than the conservative method.

Synchronization Methods for Distributed Agent Based Models

Christine Harvey and James E. Gentile, Ph.D.
The MITRE Corporation

Is there a faster way to synchronize agent states in distributed models?

Some methods communicate agent states upon request, at each time step. This can reduce processing efficiency.

Introduction

Distributed computing addresses the traditional scaling limitations of Agent Based Models (ABMs) and allows for the development of massive-scale models. Synchronization of agent states between multiple processors is complex and needs to be reliable and efficient. The protocol used to synchronize agent states across processors has a significant impact on the efficiency of the tool. Repast HPC is one of the current tools available for distributed ABMs [1]. It approaches the problem by performing a complete synchronization of all entities of interest at every time step.

This poster reviews an alternative approach to entity synchronization, a design which manages persistent, pertinent information. This protocol performs an initial synchronization among all entities with relationships to other agents and then only performs updates and synchronization following changes to relevant information. The pertinent data synchronization technique is an event-driven method to manage the communication and synchronization between the processors. The conservative and the event-driven approaches are both described and analyzed in the following sections.

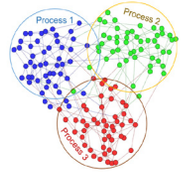


Figure 1: General overview of distributed agents on three processors

Our event-driven approach watches for agent state changes and synchronizes only when those changes occur.

Literature Cited

[1] Collier, N., and M. North. 2013, Oct.. "Parallel agent-based simulation with Repast for High Performance Computing". SIMULATION 89 (10): 1215–1235.

Further Information

For more information on the mabm.py, the Massive Agent Based Modeling Toolkit, please contact Christine Harvey or James Gentile of The MITRE Corporation at ceharvey@mitre.org or jgentile@mitre.org.

Approved for Public Release; Distribution Unlimited. Case Numbers 14-2788 and 14-3634. c 2014 The MITRE Corporation. ALL RIGHTS RESERVED.

Methods

With other Agent Synchronization techniques, local ghost copies are created of agents on foreign processors. These local copies are basic shadow copies of the actual agents, only containing the relevant state information. This process of creating local agent copies can be seen in **Figure 2.**

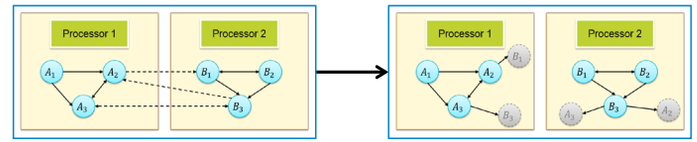


Figure 2: Overview of the local agent copying process. Agents with a relation to agents on foreign processors are recreated as basic local copies.

Agent Requests

The conservative approach to the agent synchronization problem performs a complete synchronization of pertinent information at every time step of the simulation. Each processor cycles through their agents and compiles a list of non-local agents from which information is needed. The root processor aggregates this information and broadcasts the requests to all processors which return the requested state information to the root processor. Once the root processor distributes the information gathered, each processor creates local copies of their non-local agents of interest. These temporary copies only contain necessary state information on the requested entity. Therefore, each processor has information regarding the current state of its own agents as well as basic state information on all agents of interest. This entire process is repeated at the beginning of every time step.

Agent Watches

The alternate approach recognizes that not all pertinent information changes at every time step in the model. Complete synchronization can be achieved by only tracking and reporting the changes to relevant information. Agent watching only synchronizes information when an entity has experienced a change in state.

After the creation of a relationship between two entities, if the agent of interest is not local, it is added to a global list of watched entities. During the synchronization process, the states of newly watched agents are communicated to any processor which has an interest in the agent. Basic, persistent local copies of these watched agents are made on the processors that require the state information of the non-local agent. The processor uses these local copies as a source of information for the updates on their local agents. When watched agents experience a change in state the processor sends the updated state information to the root node to be broadcast to all processors. Then, at the start of the next time step, remote copies of agents are updated to the correct, current state. With this method, fewer and smaller messages are sent at each time step than with the previous techniques.

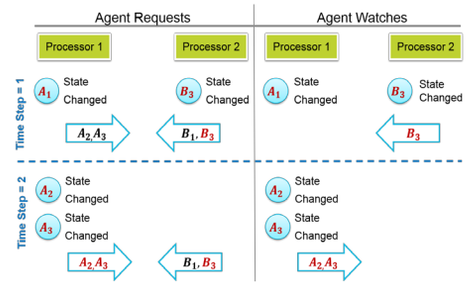


Figure 3: Basic outline of the synchronization process for the Agent Requests and the Agent Watches techniques. This graphic displays the excessive message passing that is part of the Agent Requests technique. Messages are sent for every agent that has a cross-process connection at every time step. The same outcome is accomplished with the Agent Watches technique with many fewer messages passed between the processors.

Nearly 2x Faster!

Results

A complete timing analysis was performed on both techniques using Python's cProfile tool. The sample model used in this experiment was a basic rumor model, where each agent has exactly two assigned neighbors. The connections between agents were distributed according to the probability of having a neighbor on a foreign processor. Trials were run with 20 million and 80 million agents, split across 4, 8, 16 and 32 processors. Three trials were completed at each design point, the average of these three trials can be seen in the charts below. The displayed timing is the actual simulation running time, which does not include the overhead to initialize the model.

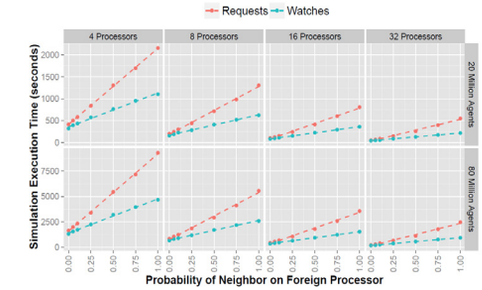


Figure 4: Simulation running time for the test models. The simulation time does not include the time required to initialize the simulation. The information given in these graphs shows that as the number of cross-process connections increases, the impact of the Agent Watches technique becomes more significant. In the most extreme cases, the Agent Watches technique is nearly twice as fast as the Agent Requests technique.

This approach is especially useful for *more static* agents.

Conclusions

The completed timing analysis shows that the watches technique is significantly faster than the requests method while maintaining the same standard of reliability. The run times for the method are quicker overall than with the requests technique. In addition to the overall speed increase, the watches technique also scales up in both number of processors and number of agents per processor more efficiently than the requests method. The use of the agent watches synchronization technique enhances the usability, scalability, and speed for anyone executing massive-scale agent based simulations.

Acknowledgements

Christine Harvey and Dr. James Gentile would like to thank the MITRE Innovation Program for their continued support of this research.

Modelling and Predicting Resource Management for Parallel Implementations of K-means

Janki Bhimani

Electrical and Computer Engineering Dept.,
Northeastern University, Boston
bhimani@ece.neu.edu
bhimani.j@husky.neu.edu

Janki Bhimani is a 2nd-year PhD student. Her current research focuses on performance prediction and capacity planning for parallel computing heterogeneous platforms. She received her M.S. (in 2014) from Northeastern University in Computer Engineering. She received her B.Tech. (in 2013) from Gitam University, India in Electrical and Electronics Engineering. She worked as a lecturer teaching “Robotics Design and Programming” at Gitam University (in 2013). She also led a student activity centre at Gitam University (for 2012 – 2013), mainly working on logic development for programming, simulation and modelling. She was an awardee of the best paper award for “Automatic Billing System Design and Simulation” at Aagama (in 2013).

Abstract

Acceleration using HPC is required for time-consuming applications including those involving big data and unsupervised learning. Currently, researchers perform experiments trying different hardware configurations to find the best performance point. This is time consuming, and moreover usage of more resources does not guarantee performance improvement. To address this problem, we have developed a performance-modeling framework that predicts resources. Our analytical model combines statistical Markov modeling with machine learning techniques and queuing theory to predict resource requirements. It predicts the total calculation time with a thread level Markov model. It learns the hardware behavior during training, such that no instrumentation of hardware details is required during prediction. Our model also predicts the total time spent in network communications through a simulation queuing model. The queuing model provides an abstraction of real world systems in terms of job arrival rate, waiting time and service rate, which helps us to estimate system communication time.

We evaluate effectiveness of our modeling technique by using it to predict performance of the K-means clustering algorithm implemented on a distributed memory platform. K-means is one of the most popular clustering algorithms, which aims to partition n observations into K clusters. Performance of K-means depends mainly on three parameters: the size of the input dataset, the number of clusters to be formed and the convergence criteria to decide the number of iterations. Additionally, in a parallel implementation, the total time to run such an algorithm further depends on the amount of parallel resources. To qualify the accuracy of our prediction technique, we rank resources and compare the predicted and actual ranking lists. Our experimental results show that the correlation between actual and predicted ranking is in the range 0.82 – 0.99 which indicates good accuracy.

There are four major features of our technique which make it unique: 1) hardware independent; 2) capable of predicting resource utilization statistics for calculation and communication separately; 3) independent of algorithm related parameters; 4) different from most analytical models, it can be trained on datasets that need not be similar in size to the actual data. In the future, we plan to evaluate our model on other applications. We also plan to expand its scope to predict the performance of computing platforms with accelerators such as GPUs.

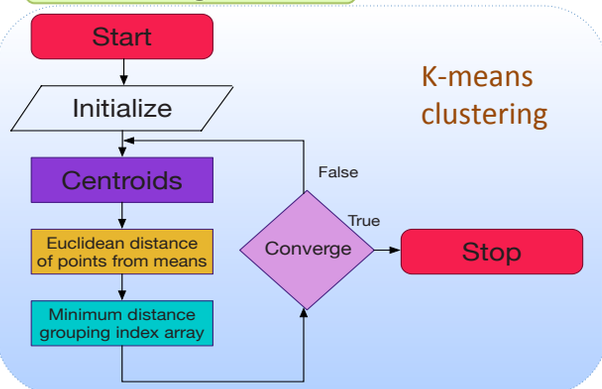
Modeling and Predicting Resource Management for Parallel Implementations of K-means

Janki Bhimani bhimani@ece.neu.edu
Ningfang Mi ningfang@ece.neu.edu
Miriam Leeser mel@coe.neu.edu
Electrical and Computer Engineering Dept., Northeastern University Boston, MA

1. Motivation

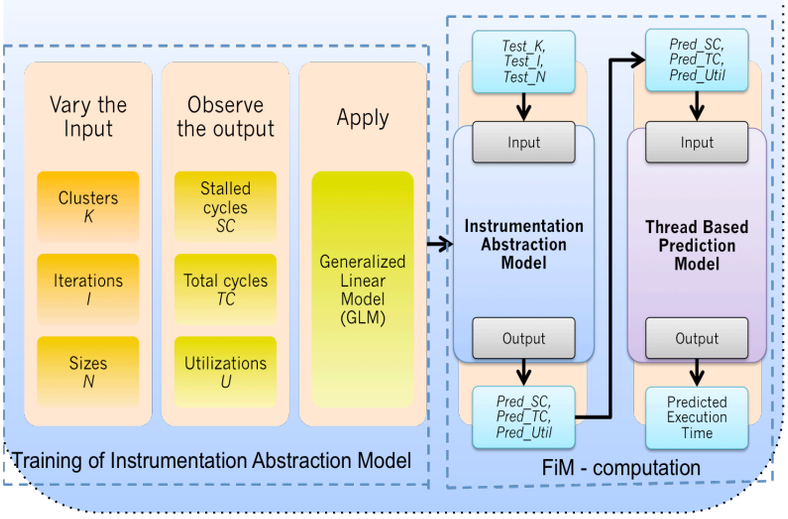
- Researchers perform experiments trying different hardware configurations
- Usage of more resources does not guarantee performance improvement
- **Goal: predict the optimal resources according to application and parallel platform**

2. Background



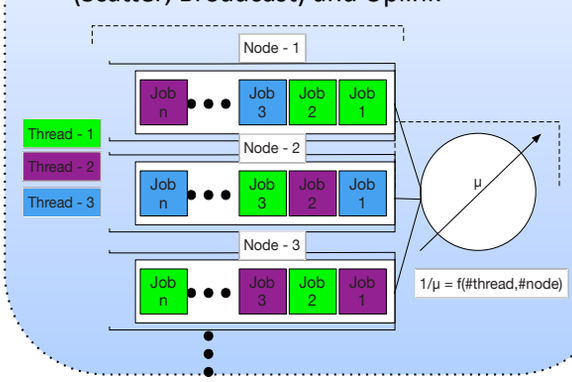
3. FiM- computation

- Combining Markov and machine learning models to predict calculation time of K-means clustering



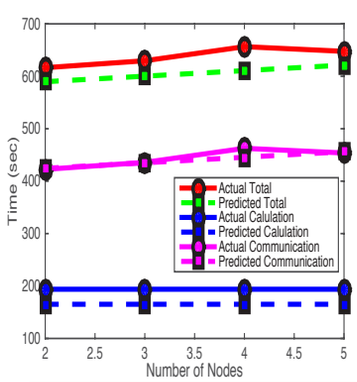
4. FiM- communication

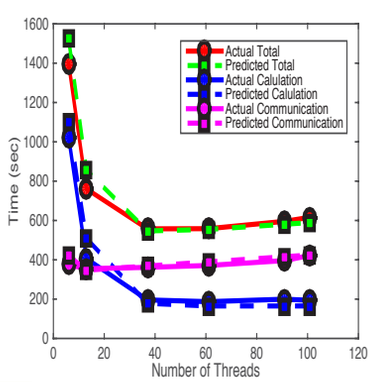
- Building queuing models to predict communication time
- Abstraction of real communication systems in terms of package arrival time, waiting time, transfer rate
- Types of communication: Downlink (Scatter, Broadcast) and Uplink

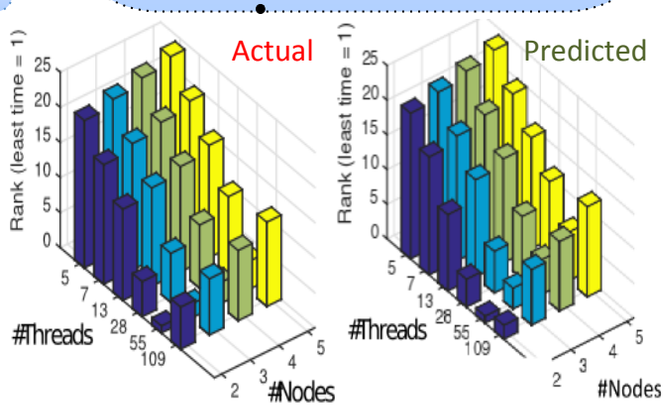


5. Results

- Optimal resources - 55 threads, 2 nodes with correlation of actual and predicted ranking – 0.82 to 0.99







6. Future work

- Test other applications, larger datasets and model overlapping of communication and computation
- Investigate prediction for heterogeneous computing system using GPUs

Supported by:
NSF
U.S. AIR FORCE

Computational characterization of genomic analysis pipelines

Sally Ellingson

Division of Biomedical Informatics, University of Kentucky College of Public Health
sally@kcr.uky.edu

Dr Sally R. Ellingson is an Assistant Professor and she leads the High-Performance Computing (HPC) initiatives at the Cancer Research Informatics (CRI) shared resource facility at the Markey Cancer Center and just this March became a new KL2 scholar funded through the Center for Clinical and Translational Sciences (CCTS) and College of Medicine at UK. Dr. Ellingson received her PhD from the University of Tennessee (UT) where she worked with the joint UT/Oak Ridge National Laboratory Center for Molecular Biophysics. She is a computational scientist that works at the intersection of HPC and computational biology and bioinformatics. As a member of CRI, she leverages the HPC capabilities of the UK Center for Computational Sciences in order to facilitate research that would not otherwise be possible.

Abstract

Since sequencing costs are dropping, improved management of data analysis and storage will be essential for state-of-the-art research and for efficient clinical decision-making based on sequencing data. A common analysis is the identification of variations within sequences that may be the cause of particular traits or diseases; these could be single nucleotide polymorphisms (SNPs), indels (insertion or deletions), or larger structural variations in the genome. It has been shown that using different sequencing technologies results in different SNP calls1 with as many as tens of thousands of SNPs being called only on a specific sequencing platform.2 Different SNP calling pipelines also give drastically different results. Using five different pipelines and fifteen samples from the same sequencing technology, only an average concordance of 57.4% was found3. Even more worrisome, using three indel-calling pipelines only gave an average concordance of 26.8%. These massive differences in results show how important benchmark data will be in testing new pipelines and technologies. Genome in a Bottle (GiaB), initiated by the National Institute of Standards and Technology, provides access to genomic DNA for sequence analysis and high-confidence calls4 developed by integrating calls from several technologies and methods, to help evaluate the entire NGS protocol in a given laboratory. In this study, we use these high-confident calls to evaluate the computational and storage needs of sequence analyses in order to better address future investments in the computational infrastructure of a clinical and research sequencing facility.

1. Rieber, N.; Zapatka, M.; Lasitschka, B.; Jones, D.; Northcott, P.; Hutter, B.; Jäger, N.; Kool, M.; Taylor, M.; Lichter, P., Coverage bias and sensitivity of variant calling for four whole-genome sequencing technologies. PloS one 2013, 8 (6), e66621.

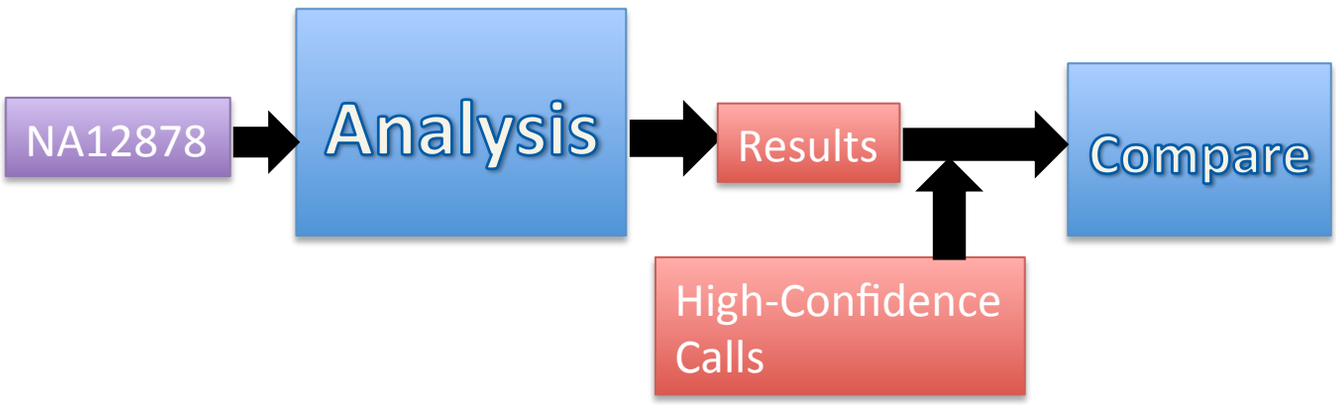
2. Lam, H. Y.; Clark, M. J.; Chen, R.; Chen, R.; Natsoulis, G.; O'Huallachain, M.; Dewey, F. E.; Habegger, L.; Ashley, E. A.; Gerstein, M. B., Performance comparison of whole-genome sequencing platforms. Nature biotechnology 2012, 30 (1), 78-82.

3. O'Rawe, J.; Jiang, T.; Sun, G.; Wu, Y.; Wang, W.; Hu, J.; Bodily, P.; Tian, L.; Hakonarson, H.; Johnson, W. E., Low concordance of multiple variant-calling pipelines: practical implications for exome and genome sequencing. Genome Med 2013, 5 (3), 28.

4. Zook, J. M.; Chapman, B.; Wang, J.; Mittelman, D.; Hofmann, O.; Hide, W.; Salit, M., Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls. Nature biotechnology 2014.

Computational characterization of genomic analysis pipelines

Sally R. Ellingson, PhD, Assistant Professor, Biomedical Informatics, University of Kentucky sally@kcr.uky,edu



Analysis:

1) GATK best practices

High-Confidence Calls:

1) Genome in a Bottle – many technologies and mapping/calling algorithms integrated
2) Illumina Platinum Genome – many individuals used for pedigree consistency check

Compare:

1) GATK 2) Useq 3) VCFeval

ERP001229 4 runs Tranche 99.0	GATK			Useq		
	TPR	PPV	nNotMatch	TPR	PPV	nNotMatch
UG vs NIST2.18	93.98%	99.66%	9301	94.01%	99.71%	8023
HC vs NIST2.18	95.13%	99.72%	7722	95.17%	99.79%	5930
UG vs NIST2.19	93.43%	99.53%	13823	93.47%	99.61%	11679
HC vs NIST2.19	94.83%	99.67%	9986	94.89%	99.77%	7034
UG vs PGv7	88.12%	99.53%	17494	87.75%	99.67%	12324
HC vs PGv07	89.99%	99.36%	24236	89.59%	99.55%	16903

Running time:
~140 h for UG caller, ~280 for HC caller
Multiple runs cost less time (parallel pre-processing), gives more variants.
Total Intermediate File size:
about 10 to 11 times of input file size.
Key immediate file (take ERP001229 as example) for 138GB input:
Total size: about 2 times of input file size
Caller:
UG: Faster but less accurate.
HC: More accurate but slow.
VQSR tranche:
High tranche means high sensitivity, but low precision.
In our test, 99.9 and 99.0 are better.
Comparison tool:
Useq-vcfcomparator: Give VCF file and statistical report, Can compare multi-sample VCF file.
GATK-VariantEval: Only gives statistical report
RTG-vcfeval: Give VCF file and statistical report, But can't compare multi-sample VCF file.

Test data		input	output	runtime
ERP001960 (1 run)	UG	96 GB	960 GB	143 h
ERP001960 (1 run)	HC	96 GB	960 GB	283 h
ERP001229 (4 runs)	UG	138 GB	1210 GB	62 h (156 h)*
ERP001229 (4 runs)	HC	138 GB	1450 GB	118 h (212 h)*

Discussion:
This evaluation pipeline is being developed in order to test new tools and make estimates on high-performance computing and storage needs
Ongoing work:
Optimizing for efficiency on DLX (UKY cluster)
Acknowledgements:
Xiaofei Zhang, Computer Science, UKY
NIH National Center for Advancing Translational Science grant KL2TR000116

Reducing the Barriers to PDES Checkpointing

Elsa Gonsiorowski

Rensselaer Polytechnic Institute, USA
gonsie@rpi.edu

Elsa is an HPC developer and computer scientist at the Center for Computational Innovations, a high performance computing center at Rensselaer Polytechnic Institute. She enjoys bridging the gap between academic research and current HPC resources though teaching, collaboration, and software engineering. Elsa graduated from RPI in 2010 with Bachelor of Science dual degree in Physics and Computer Science. Since then, she has continued her studies at RPI and is now a Doctoral candidate working with Professor Christopher Carothers. Her research includes improving the ROSS framework for parallel discrete-event simulation and developing tools for gate-level circuit simulation models. When she is not at her computer, Elsa enjoys rowing, CrossFit, and exploring new cities.

Abstract

Parallel discrete-event simulation (PDES) is a common application in high performance computing. For any large scale simulation within an HPC system, running time can be a limiting constraint. Through checkpointing, a single simulation can be performed across many independent HPC “jobs.” A simulation checkpoint requires more than saving the state of the model, part of the simulation system itself must also be saved.

This poster presents RIO, a checkpointing API for Rensselaer‚Äôs Optimistic Simulation System (ROSS). Through this API, developers specify the serialization of their model. The remaining tasks of recording version information, global variables, “in-flight” simulation messages, and current simulation time are handled by RIO. RIO enables both the input and output of checkpoint data, in parallel, via MPI I/O collective functions. Through the explicit partitioning of the entire simulation system, RIO enables a change in parallelism (changing the number of MPI ranks) across the disjoined HPC jobs for a single simulation.

By enabling the creation of simulation checkpoints, RIO is able to increase the usefulness of the PDES in the HPC context. Most HPC facilities, including the Center for Computational Innovations at RPI, set a maximum runtime on each job. Without checkpointing, it would be impossible for long-running simulations to make progress. Using PHOLD, an exemplar PDES model, we demonstrate the correctness of the RIO system, and show a minimum overhead in overall application running time. This experiment is performed on multiple configurations of an IBM Blue Gene/Q supercomputer.

Reducing the Barriers to PDES Checkpointing

Elsa Gonsiorowski (gonsie@rpi.edu) and Christopher Carothers (chrisc@cs.rpi.edu)
Rensselaer Polytechnic Institute, Troy, NY

Background

Parallel discrete-event simulation (PDES) is a common application in high performance computing. For any large scale simulation within an HPC system, running time can be a limiting constraint. Through checkpointing, a single simulation can be performed across many independent HPC "jobs." A PDES checkpoint requires saving the state of the model, in-flight events, and the settings of the simulation system itself. Here we use Rensselaer's Optimistic Simulation System, ROSS [2], and its YAWNS-based conservative synchronization algorithm [4].

ROSS Checkpointing: RIO

ROSS I/O, or RIO [1], is an integrated checkpointing system. It is able to take advantage of both the regularity of a PDES system (a model made up of LPs and events partitioned across nodes) and the efficiency of MPI [3] to quickly create compact checkpoints. RIO adds to the ROSS API and allows model developers to create checkpoints at the conclusion of their serial or conservative simulations. These checkpoints can then be used to restart and continue the simulation. RIO is dynamic and can allow for a change in parallel configuration across executions of the same simulation.

A RIO Checkpoint

Read-Me for Humans

To ensure data legacy, it is important to capture details about the software used to create the checkpoint. Both a ROSS and RIO git hash are recorded, as well any command line options and global variables specified during simulation.

Formatted Metadata

A checkpoint of any parallel system will consist of multiple partitions. For a PDES system, a partition will contain both LPs and events. Information about the location, size, and content of each partition's checkpoint data is recorded in a separate metadata file.

Checkpoint Data

Each partition's checkpoint data is stored in binary format. For increased parallelism, multiple files can be used for data within the same checkpoint. Since the metadata is separated from the data, each partition can be read in parallel.

The RIO API

Global Variables

RIO has several variables which the model developer can tune for a simulation. Command line arguments specify the number of partitions in a checkpoint and the number of files used for data. Another variable specifies the way in which a RIO checkpoint is loaded into a simulation. The model developer can also specify a model version string which is written to the checkpoint's read-me file.

LP Serialization

Model developers need to implement a serialize and de-serialize methods for each LP type in the simulation. These methods can be as simple as copying an LP's state into a provided buffer or as complex as needed. To build a compact checkpoint, LP sizing functions must also be defined.

Implementation

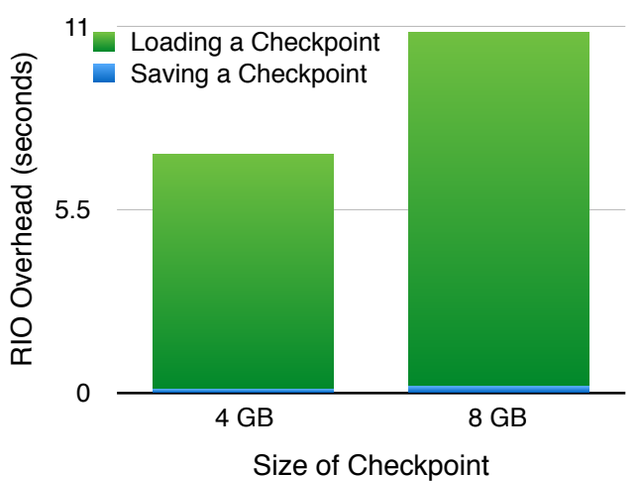
RIO is based on MPI file collective operations including:

- `MPI_File_write_at_all`
- `MPI_File_read_at_all`

These operations are blocking and require an explicit offset into the file [3].

Performance

As with any software reliant on file I/O, RIO performance is linked with file system performance. The graph on the right shows the overhead caused by RIO on 8 nodes of an IBM Blue Gene/Q supercomputer. Overall, RIO is able to take advantage of a fast I/O subsystem. The high overhead of loading a checkpoint is due to the time taken by the initialization of each LP.



[1] RIO: ROSS I/O. http://github.com/gonsie/RIO
[2] ROSS: Rensselaer's Optimistic Simulation System. http://github.com/carothersc/ROSS
[3] MPI: A Message-Passing Interface Standard. Version 3.1 http://www.mpi-forum.org/
[4] D. Nicol. The Cost of Conservative Synchronization in Parallel Discrete Event Simulations. *Journal of the ACM*, 40(2): 304–333, 1993.

Optimising Data Intensive Simulations: Journey from HPC to Cloud

Mariam Kiran

School of Electrical Engineering and Computer Science,University of Bradford, UK

M.Kiran@bradford.ac.uk

Dr. Kiran is a Lecturer of Software Engineering at Department of Computer Science at University of Bradford. Her main research areas are Complex model simulations and Cloud computing. She has been working with various interdisciplinary groups in modelling and simulation, Smart Cities and Internet of Things. She graduated from Sheffield University, where she was one of the developers of FLAME, an open-source agent-based modelling toolkit. Kiran has worked in various EU and UK projects and collaborated with industry regularly. Kiran is regularly experimenting with new technologies and a promoter of open-source software. Her work involves working with other disciplines for reliable software solutions and using new research algorithms. She is a member of BCS, IEEE and ACM and also leads the ACM-women chapter in the UK.

Abstract

Modelling and Simulation (M&S) is heavily influenced by availability of computational power and resources. Some simulations take days or weeks to finish, often stopping in the middle due to memory constraints or processing power. Past modelling techniques have used HPC grids and GPUs to quickly process large complex equations for multi-massive variables to produce emergent solutions [1, 2]. However these platforms are limited, introducing assumptions into models due to their architectures. Cloud deployment can be a way forward for solving multiple processing issues for scientific computing, but raises multiple new challenges for resource optimization and parallelization.

We present a case for using cloud for data intensive simulation techniques as well as a discussion on providing a software engineering approach for writing M&S models such that they can run in an optimal manner regardless of underlying infrastructure. We discuss how the performance can be quantified as well as how to abstract architectural dependency from software codes and how Cloud computing can help M&S modelling in this context. Our work considers true dynamic scalability during simulations for data intensive applications, and the need to process large quantities of ‘in place’ as well has providing effective fault tolerance

References

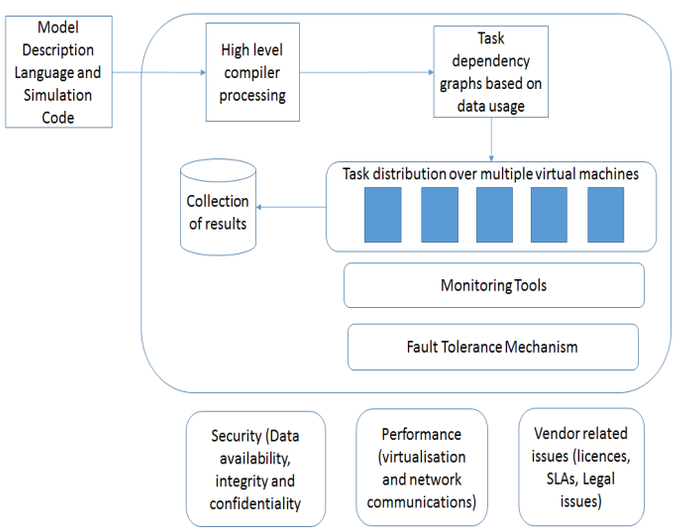
[1] Kiran M., Bicak M., Maleki-Dizaji S., Holcombe M., 2011, FLAME: A Platform for High Performance Computing of Complex Systems, Applied for Three Case Studies, Acta Physica Polonica B, Proceedings Supplement, PACS numbers: 07.05, DOI=10.5506/APhysPolBSupp.4.2011.

[2] Kiran M., Richmond P., Holcombe M., Chin L.S., Worth D. and Greenough C., 2010, FLAME: Simulating Large Populations of Agents on Parallel Hardware Architectures, AAMAS2010: 1633-1636, Toronto, Canada.

Optimising Data Intensive Simulations: Journey from HPC to Cloud

Mariam Kiran
m.kiran@brad.ac.uk

Introduction:
Modelling and Simulation (M&S) is heavily influenced by availability of computational power and resources. We use past research experience, from development of M&S to automatically parallelize, using C/MPI, on HPC grids, now for Cloud computing, such to optimise simulations using cloud advantages but raises multiple architectural challenges.



Research Questions targeted:
Q1) How can a model be written such that it can run in optimum manner regardless of underlying infrastructure? Can performances be quantified?
Q2) How can we abstract architectural dependency from software code, to prevent rewriting of code for newer emergent high performance architectures?
Q3) Can Cloud technologies help, laying a software engineering approach for writing simulation models for non-computing domains.
Q4) What are the verification and validation techniques needed for model checking in M&S for HPC.

Current Challenges:
- Performance concerns for processing, true dynamic scalability and speed during simulations for data intensive applications.
- Enormous amounts of data being produced/processed 'in place'.
- There are three facets of cloud testing such as testing as a service, testing applications installed in the cloud and testing the cloud itself.
- Fault tolerance: If the simulation execution fails, the system should be able to detect these and 'rewind' from an operational point and continue again perhaps using an efficient.
- Additional research challenges such as energy efficiency of virtual machines and economic costs of running multiple data-intensive simulations.

Conclusions:
Using cloud for data intensive simulation techniques, particularly when underlying infrastructure is not available or using public clouds. The research involves working with non-computing scientists who regularly write simulation software. With the evolution of newer exascale and cloud computing architectures, this problem is becoming more apparent, creating a huge divide among the ease of use and information availability for multiple scientific communities to use newer technologies for their work.

References:
Kiran M., et al., 2011, FLAME: A Platform for High Performance Computing of Complex Systems, Applied for Three Case Studies, Acta Physica Polonica B, Proceedings Supplement, PACS numbers: 07.05, DOI=10.5506/APhysPolBSupp.4.2011.
Kiran M., et al., 2010, FLAME: Simulating Large Populations of Agents on Parallel Hardware Architectures, AAMAS2010: 1633-1636, Toronto, Canada.

Programming of Data Centric Applications for Heterogeneous Computing Environments

Nasibeh Nasiri

University of Massachusetts Lowell
Nasibeh_Nasiri@uml.edu

Nasibeh Nasiri achieved her B.Sc and M.Sc in Computer Engineering and has joined Advanced Electronic Technology Center group in Electrical and Computer Engineering Department at University of Massachusetts Lowell. She started her PhD program under the supervision of Prof. Martin Margala, in 2012. She teaches some courses in ECE department and has also gained some industry experience as a HPC Engineer during her graduate studies. Her research interests are low-level and high-level synthesis on FPGAs for HPC and data centric applications, heterogeneous computing, reconfigurable architectures, digital circuit design, low power designs, and VLSI.

Abstract

Considering the power aspects when dealing with large scale data centric applications is becoming vital. Designing a large datacenter now requires access to natural free cooling and cheap electricity in order to be economically viable. Document classification is at the heart of several of the applications that have been driving the proliferation of the internet in our daily lives. The ever growing amounts of data, the need for higher throughput, and more energy efficient document classification solutions motivated us to investigate alternatives to the traditional homogenous CPU based implementation of a document classification system. In this work, we investigate a heterogeneous system where CPUs are combined with GPUs and FPGAs as system accelerators.

OpenCL, the emerging open programming standard, provides a single design environment for heterogeneous systems. In this work, the stream of HTML documents is parsed, compressed and stored on a document-by-document basis. Bigrams and trigrams are created to improve the accuracy of the document classification. We have targeted an HP server, GPU K40C and De5-Net FPGA board with the same OpenCL implementation. However, some minor optimizations are done when targeting each of these platforms. The results show that the best throughput of the document classification application on each platform was 343 MB/s, 432 MB/s and 452 MB/s respectively, when running on CPU, GPU and FPGA. Our experiments show up to 79% performance per watt improvement on FPGA compared to the best results of GPU and CPU. Our results also show up to 32% power efficiency improvement for the FPGA implementation over the GPU and CPU implementations. The results show that heterogeneous implementations offer not only better performance but also better performance per watt.

Programming of Data Centric Applications for Heterogeneous Computing Environments

Nasibeh Nasiri, Martin Margala and Wim Vanderbauwhede
Nasibeh_Nasiri@uml.edu
Electrical and Computer Engineering Department, UMass Lowell

INTRODUCTION

- The need for efficient methods to sift through the massive amounts of information that is generated on a daily basis.
- The need for higher throughput, more energy efficient document classification solutions.
- Alternatives to the traditional homogenous CPU based implementation of a document classification system.
- Ability of heterogeneous systems to use specialized hardware to tackle different types of algorithms
  - ✓ More power efficiency of heterogeneous systems than their homogenous counter parts

HDLs as FPGA Programming Model

Design can be described:

- State machines
- Datapath
- Arbitration
- Interface to external memory
- Buffering
- And etc.

FPGA programmer should take care of:

- Synthesis
- Place and route
- Timing closures
- And etc.

High Level Programming of FPGAs

- Compiler is taking care of synthesis, place and route, timing closure, IO interface and other constraints.
- Consequently without getting involved in RTL details of design the programmer can concentrate on the optimization of the design.

OpenCL SDK for FPGAs

- The hardware design is extracted automatically from the OpenCL kernel
- Extracted hardware design can take the advantage of deeply pipeline parallelism
- Host and device have separate memory spaces
- Data is explicitly moved between them over PCIe bus

Design Flow of OpenCL on FPGA

In OpenCL a program is divided into two parts:

I. Host code

II. Device code

The computational intensive functions are offloaded from the host processor to the accelerator device and are written using a subset of C99 with OpenCL specific keywords.

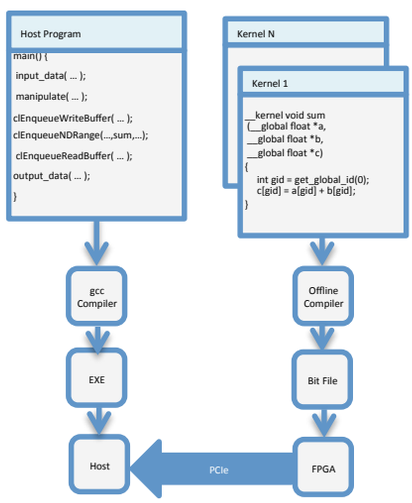


Figure 1. Design flow of OpenCL on FPGA

Data Centric Application

1. The input stream coming from the network will go through the parser and once a word is detected and encoded using compression coding, it will be sent for scoring.

2. The bigram and trigram of each word is generated and were looked up in the profile.

3. N hashing function has been used to evaluate the membership of each unigram, bigram or trigram.

4. If the retuned values of N hashing functions are all one for each unigram, bigram or trigram, this can be a sign of existence of these word(s) in the profile so the next step is looking up in the profile however false positives are possible.

5. When it is required to refer to the profile, the rest bits of the hit word are compared with the rest bits of the profile then if they are matched, the weight bits of this element is accumulated to the score.

6. The final result of classification will be made using naïve Bayesian classifier.

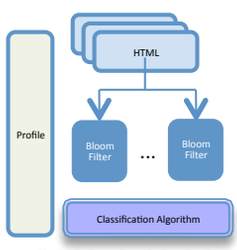


Figure 2. Data Centric Application

Bloom Filter and Data Set

- **Bloom Filter** is a data structure implemented with different hash functions to evaluate the membership in a large set.
- Ideally the performance of the implemented application should be evaluated using real world input data.
- Since our access to real data sets was limited, we relied on synthetic document collections that are statistically similar to real-world collections.
- **TREC Aquaint** provide good coverage on the impact of different document lengths and sizes on filtering time.
- ✓ **Table I. Synthetic data set in our experiments**

Collection	Number of Documents	Average Document Length	Average Unique Terms
TREC Aquaint	1,033,461	437	169

Results

host system: An HP DL180 G6 with dual Intel Xeon L5630 2.13GHz processors and 144GB DDR3, 1333MHz RAM.

GPU: NVIDIA K40C

FPGA board: A Nallatech PCIe-385N A7 FPGA board with 8GB RAM connected through a second generation PCIe connection to the host system.

The results show that the best throughput of the document classification application on each platform was 343 MB/s, 432 MB/s and 452 MB/s respectively, when running on CPU, GPU and FPGA. Our experiments show up to 79% performance per watt improvement on FPGA compared to the best results of GPU and CPU.

Table 1. Throughput and performance per watt on CPU, GPU and FPGA after optimization

	Device	Throughput (MB/s)	Performance per watt(MB/Watts)
No Bloom Filter Bayesian - FSM	CPU	311	1.47
	GPU	432	4.32
	FPGA	353	4.51
Bloom Filter Bayesian-FSM	CPU	343	1.64
	GPU	432	4.59
	FPGA	452	4.92

CONCLUSIONS

Our experiments show that FPGAs are a promising platform for data centric applications. They not only provide high throughput, but also high performance per watt when compared to CPUs and GPUs. High level development of FPGA designs using Altera's OpenCL is making FPGAs an attractive platform for heterogeneous environments.

Spectrum Scale with focus on AFM and how it can help in technical computing and cloud

Trishali Nayar

IBM India Systems Development Lab
ntrishal@in.ibm.com

Trishali Nayar is involved with Spectrum Scale development and focusses on the Active File Management (AFM) functionality. She recently worked on the “Asynchronous Disaster Recovery” feature of AFM. She is also designing solutions around Spectrum Scale, for deploying in the cloud. She has past experience in the area of File Systems development. She has multiple patents in the area of distributed computing. She has co-authored a Redbook and published developerWorks articles as well. She is passionate about various diversity initiatives and is currently part of the Society of Women Engineers (SWE) and involved in their international affiliate group.

Abstract

IBM Spectrum Scale (formerly General Parallel File System) is a high-performance enterprise platform for optimizing data and file management. It is installed in clusters world-wide and supports high-performance computing applications from climate modeling to tornado simulation, with databases such as IBM DB2®, in big data MapReduce analytics, gene sequencing, digital media and across multiple industries, like financial, retail, science and government.

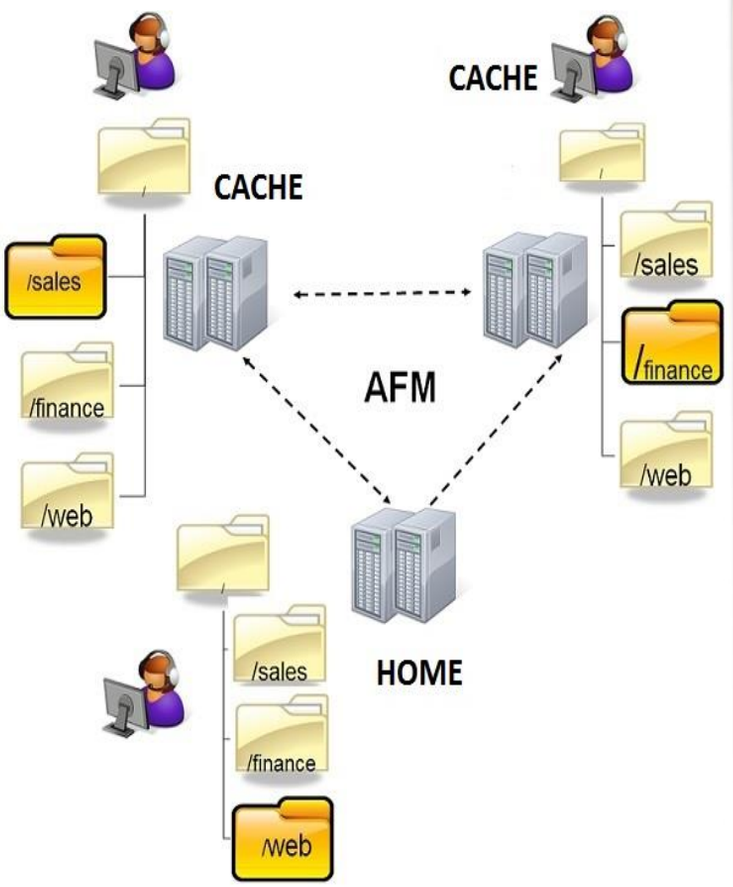
Active File Management (AFM) is a breakthrough technology that allows global sharing in a new and unique way. It is designed to enable efficient data transfers over WAN or high performance transfers over LAN. Data sharing is now extended to asynchronous replication of data and only the changes are sent over the WAN, hence increasing the speed and reducing the costs.

During the workshop, details on what is AFM and how it can help in HPC environments will be discussed. These will include: Facilitating content distribution for global enterprises and “follow-the-sun” engineering teams. How to create a global wide-area file system and allowing data to move transparently, across multiple sites. Scalable Disaster Recovery for Super Scale File Systems, useful in Technical Computing. How it can be used for corporate WANs, hybrid cloud or bursting to cloud providers.

Spectrum Scale with focus on AFM and how it can help in technical computing and cloud

Trishali Nayar

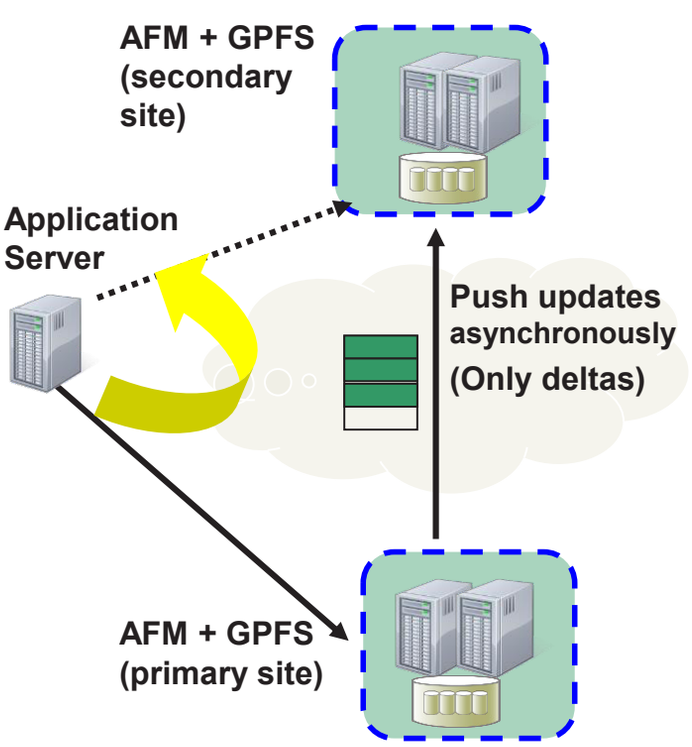
WAN Caching - Global Namespace



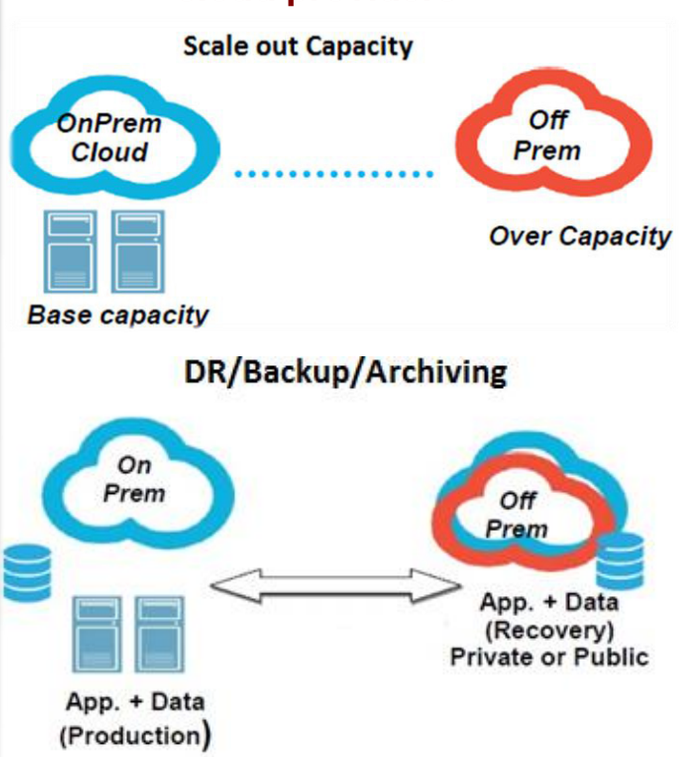
Technology Overview

- **Spectrum Scale** is a fast, simple, scalable and complete storage solution for today's data-intensive enterprise.
- Integrated tools designed to help organizations manage petabytes of data and billions of files.
- **Active File Management** is a high-performance clustered file system cache.
- Easy to deploy, relies on open standards.
- Multiple modes are available.
- NFS & GPFS protocols supported

Disaster Recovery - Asynchronous Replication



Hybrid cloud, Bursting to cloud providers



WHPC
WOMEN IN HIGH PERFORMANCE COMPUTING

Women in High Performance Computing

WHPC works to identify and address gender issues in HPC and how they may affect research quality

The Women in High Performance Computing network aims to encourage female participation in HPC by...

Increasing the visibility of female role models

Providing opportunities for networking

Raising awareness that gender-balancing research groups improves scientific output

Discussing how to balance careers with other commitments

Highlighting best practice by institutions that have achieved gender-balance in HPC

Providing workshops for women in HPC

epcc 25TH ANNIVERSARY

Join us!

www.womeninhpc.org.uk/get-involved

info@womeninhpc.org.uk

@women_in_hpc

www.womeninhpc.org.uk

EPSRC
Engineering and Physical Sciences Research Council

archer