Project Ideas for PCSE 2018 (preliminary list – will be update!)

General
Please investigate suggestions that you are interested in. If you fell confident write the half page abstract. Otherwise talk to Charlie or Lars.

Overview
There are many practical areas of Parallel Computing to explore, ranging from the basics of interprocessor communication to data-partitioning among distributed processors.

Below are some ideas within specific areas, but you are welcome to suggest your own project.

Projects may specifically involve parallel aspects, or supported parallel components. They may involve hardware, standard programming languages, parallel optimization, parallel I/O, etc.

The sections in bold are areas of interest at most HPC conferences. Project ideas are provided in each section. Review all sections to get a feel for the scope of projects, and the variety you might select from.

MPI

Write simple poisson solver in 2-D and evaluate performance. (Undergrads only.) Present scaling study that evaluates data layout and different communications (Use different messaging mechanisms (block/no-blocking/SendRecv, etc.)

Efficient usage of RDMA in MPI communications. Remote Direct Memory Access is a feature that allows a processor (MPI rank) to write directly into the memory of a remote processor. Evaluate the performance w/wo RDM, or discuss the software stack and the features that provide benefits over the non-DMA methods. See http://htor.inf.ethz.ch/publications/img/mpi3-rma-overview-and-model.pdf.

Investigate overlapping communication with computation. An ideal algorithm for testing this is the FOX algorithm for Matrix-Matrix Multiplication. Georgios Goumas, Nikos Anastopoulos, Nikolas Ioannou and Nectarios Koziris. Overlapping Computation and Communication in SMT Clusters with Commodity Interconnects (http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5289174)

Evaluate and extend MPI Benchmarks on TACC Systems (Lonestar5 or stampede2). See Barrett and K. Scoot Hemmert, "An Application Based MPI Message Throughput Benchmark". "This paper introduces the Sandia Message Throughput benchmark which measures message throughput using a communication pattern which is neither best-case nor worst case, but which mimics communication patterns found in real world applications."
(http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=5289198)

Evaluate the communication patterns required for Genome Sequencing. See LCI 2008, "Accelerating MPIBLAST on System X by using RAMdisks".

Discuss and determine the performance of new mechanism/algorithms for collective communications. Rajeev Thakur MATHEMATICS AND COMPUTER SCIENCE DIVISION ARGONNE NATIONAL LABORATORY ARGONNE, IL 60439, USA THAKUR@MCS.ANL.GOV Rolf Rabenseifner RECHENZENTRUM UNIVERSITAT STUTTGART (RUS) HIGH PERFORMANCE COMPUTING CENTER (HLRS) UNIVERSITY OF STUTTGART D-70550 STUTTGART, GERMANY Also, http://www.springerlink.com/content/542207241006p64h/fulltext.pdf

OpenMP, Threading

Implement and evaluate algorithm with OpenMP TASK constructs. E.g. LU decomposition. (http://www.cs.unc.edu/~amos/data/cthpc2014-omp.pdf, see gram-schmidt example: versions range from serial, to worksharing, to using explicit tasks: https://github.com/yohannteston/Parallel-course-Ass3/)

Experiment evaluated Algorithms that exploit task dependences and do-across semantics. (http://www.openmp.org/wp-content/uploads/SC17-Jost-OpenMP-booth-doacross.pdf)

Investigate the use of the PIOMan runtime (scheduler) systems for efficiently handling IO and MPI communications. (http://runtime.bordeaux.inria.fr/Runtime/)

A study to evaluate many-core scaling (with thread count) on Intel's new Xeon Phi Coprocessor (MIC, Many Integrated Core). This could also involve other different numa/node architectures (from single bus to mesh networks—ARM, GPU, FPGA), but working with MIC on Stampede would be convenient. This could include your research, or something as simple as implementing an OpenMP Poisson solver in 2-D and investigating performance relative to the Sandy Bridge CPU on the frontend.

Develop a task parallel code using Charm++ objects instead of OpenMP tasks. Requires C++ knowledge. (http://charm.cs.uiuc.edu/research/charm/)

Algorithms

FFTs, Linear Algebra, Molecular Dynamics, Quantum, QCD, Lattice Boltzmann methods, CFD methods, Genomic Sequencing, …

Fault Tolerance

Write a code using either AMPI Checkpointing (in memory and disk), or use your own MPI-driven code. Can Ma, Zhigang Huo and Dan Meng. DCR: A Fully Transparent Checkpoint/Restart Framework for Distributed Systems

NUMA

Using NUMA to optimize performance (using numa controls and memory affinity)

http://www.hpcadvisorycouncil.com/pdf/Maximizing-Application-Performance-in-a-Multi-Core-NUMA-Aware-Compute-Cluster.pdf

http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6495451

Application Scaling

Use openMP (on node) to eliminate on-node MPI communications and provide better (less congested) scaling for communications between nodes. – Can you beat the MVAPICH library?

Architecture Comparison

IO

Develop toy code that compares all-core IO on node to aggregated IO to fewer cores.

Compare parallel HDF5/NetCDF to home grown approaches or advanced methods controlling them: https://pdfs.semanticscholar.org/18b1/65b7e99ccf3fe216cfcbc3397ad1a79006d5.pdf

Analyze Parallel I/O Performance on TACC SSD system-- https://portal.wrangler.tacc.utexas.edu :

GPUs

Message Passing for GPGPU Clusters. Use cuda on TACC's Maverick system to develop a simple application that uses GPUs for kernel computations, the CPU for other components, and MPI for communication between CPUs. Possibly use cudaMPI and glMPI to send MPI messages directly from the GPU's memory to other graphics cards or CPU memory.

Other programming systems

Program one of the MPI assignments in UPC or X10. Evaluate the difference in performance at difference scales. Discuss the communication costs of each. (You will need to use a system that is not available through TACC.) See slides Programming using the Partitioned Global Address Space (PGAS) Model. Tarek El-Ghazawi - George Washington University; Vijay Saraswat - IBM

Develop a Global-Array algorithm that replaces a message passing algorithm (e.g. finite difference kernel, multi-node matrix multiply, etc.) "Parallel Programming Using the Global Arrays Toolkit" (Bruce Palmer, Manojkumar Krishnan, Sriram Krishnamoorthy - Pacific Northwest National Laboratory; P Sadayappan - The Ohio State University) See http://www.cluster2009.org/tutorial6.pdf. OR Parallel Programming Using the Global Arrays Toolkit Bruce Palmer, Manojkumar Krishnan, Sriram Krishnamoorthy - Pacific Northwest National Laboratory; P Sadayappan - The Ohio State University.

Evaluate the Chapel language (http://chapel.cray.com/, available on Kraken, installable in general). How does it make programming easier? How efficient is it in practice?

Evaluate the Charm++ library (available at TACC) for its programmability and efficiency. What sort of algorithms are easier to express in it?

Evaluate the efficiency and usefulness of MapReduce as a general programming system. Code some algorithms both in MPI and Hadoop (or another MR system) and measure performance.

Tools

One of the most critical problems of benchmarking parallel codes is the ability to use a common clock. Measuring with a "centralize clock" is a complicated problem that requires a special protocol between nodes. Evaluate the use of the recently enchanced PTPd utility to synchronize system times against PTP time, within one microsecond. See "*Hardware Assisted Precision Time Protocol (PTP, IEEE 1588) - Design and Case Study." From LCI 2008 conference.*

Application Software

Experiment with ODE Solver Libraries that have been parallelized with SMP.
Intel: http://software.intel.com/en-us/forums/showthread.php?t=61858&o=d&s=lr

Data Partitioning
Zoltan: http://www.lanl.gov/conferences/salishan/salishan2005/karendevine.pdf

Parallel Lanzcos in Quantum Chemistry calculations (an IO problem?): See http://www.mlgworkshop.org/2016/paper/MLG2016_paper_9.pdf.

Reference:

Google search: "IEEE Cluster 20xx" xx={09, 10, …, 2017}

http://www.nersc.gov/events/hpc-workshops/