Parallel Programming and Computing: Group Project

Christopher D. Carothers
Department of Computer Science
Rensselaer Polytechnic Institute
110 8th Street
Troy, New York U.S.A. 12180-3590

April 3, 2020

HARD DEADLINE: 6 p.m., Tuesday, May 5th, 2020

1 Description

The central key requirement of your group project is that it **must involve** the design, development in software and experimentation of a massively parallel system. **A key factor is that your parallel system must involved a high degree of interaction and not be considered "embarrassingly parallel"**. This means that projects involving "wide-area", "campus-area" or other asynchronous, master-slave approaches are not a viable project for this course.

The end goal is that you are making progress towards a result that is of high enough quality to be published. Beyond that, you are free to explore any particular "non-embarrassingly" parallel application that is relevant to your research. More specifically, use the ACM Master Conference Format which can be found at the following url: `https://www.acm.org/publications/proceedings-template`. Both MS Word and LaTeX example formats are available.

Examples of a project might be:

- Re-implement a current serial algorithm/research code using MPI and CUDA effectively on the AiMOS supercomputer with parallel MPI I/O for input/output and/or checkpointing of data.
- Parallelization of components in an existing HPC code using CUDA threads, MPI and MPI I/O for input/output and/or checkpointing of data.
- Design a performance benchmark that tests how CUDA with different MPI ranks with MPI parallel I/O perform together on the AiMOS supercomputer system. What are the overheads as you change the mix of MPI ranks and threads and how are reading and writing data rates to the parallel I/O system effected ? Here, you'll want to consider a number of point-to-point communications and collective calls.
- Construct a new PARALLEL discrete-event model using a parallel simulator (e.g., ROSS) or re-purpose your Game-of-Life code. For example, create a human "agent" model of the world (8 billion people) being infected by a pandemic virus (e.g., COVID-19).
- **If your project does not have a signficant MPI parallel I/O component to it, your group will need to perform a 4th programming assignment on parallel I/O.**

- Final note: if your project does not involve MPI or CUDA at some level you are probably on the wrong track for this project and should speak with the Prof. Carothers.

The key components of your project submission paper:

1. First and foremost, your project MUST be about High-Performance Computing and performance. So, this excludes any sorts of Cloud Computing projects or embarrassingly parallel "over the web" projects. While those are valid types of computation they are not what this course is about. We are concerned about high-performance computing on supercomputing class systems.
2. **It is critical you pick something you and your team can complete.**
3. List your team members. Teams can have up to 3 members. For each team member describe their contribution to the overall project. That is, what did you do besides attend meetings?.
4. Describe your parallel implementation both code and algorithms used.
5. We have read a number of papers this semester. So, provide a review of related published articles. Do a **Google Scholar** search. In terms of related work, pick key older papers provided they have a very high citation count otherwise pick the newest results possible from key journals and conferences - e.g., look for IEEE, ACM and SIAM conference and journal publications.
6. **Compute Performance Results:** This will include your sequential and parallel results and indicate your overall speedup. You should perform a "strong scaling" study where the problem size remains the same and CPU/GPU/node count increases as well as a "weak scaling" study where the problem size grows with the increase CPU/GPU/node count. Also, understand the performance improvement of just using CPUs via MPI vs. hybrid CPU/GPU with both MPI and CUDA.
7. **I/O Performance Results (done in lieu of Assignment 4):** Here, you will select the collective mode of MPI parallel I/O and measure the time to complete various I/O operations (reads, writes, etc) associated with your particular project. As part of your experiments, make sure that you are able to vary the number of MPI ranks that write to or read from a single file. How does the number files impact parallel I/O performance for your particular project?
8. **Analysis of performance results:** Provide additional information on why your performance turned out they way it did. In particular, you should understand how much communication overhead your program incurred versus doing real computational work. This should be measured and quantified. Use the cycle counters available on the POWER9. The inline machine code for getting the cycle count (with a resolution of 512 MHz) is:

```
typedef unsigned long long ticks;

static __inline__ ticks getticks(void)
   {
   unsigned int tbl, tbu0, tbu1;

   do {
```

```
  __asm__ __volatile__ ("mftbu %0" : "=r"(tbu0));
  __asm__ __volatile__ ("mftb %0" : "=r"(tbl));
  __asm__ __volatile__ ("mftbu %0" : "=r"(tbu1));
      } while (tbu0 != tbu1);

     return ((((unsigned long long)tbu0) << 32) | tbl);
}
```

9. Summary and future work. Provide a summary of what you did and directions of where you think this project could go in the future. That is what problems did you not have time to solve?

10. **You can expect your paper/project write-up to be a similar length regular conference paper with the following provisos: For undergraduate teams your report should be at least 6 pages double column, single spacing, 10 point font with performance graphs and references. For graduate teams, your report should be at least 9 pages double column, single spacing, 10 point font with more performance graphs and more references. As paper format examples, use the papers we discussed in class.**

11. **Submit a final copy of your report in PDF format as well as all source code on Submitty.cs.rpi.edu.**

12. EMAIL Prof. Carothers early if have questions or want any sort of guidance about your project. Waiting until a few days before the deadline to ask how or what you should is not acceptable.

13. The deadline (6 p.m. May 5th, 2020) is hard because it is the end of Final Exam period. So, please plan your use of the supercomputer accordingly and expect the time for 4 node jobs to take potentially hours to get through the system.

14. **Revised Course Grading:** Because of the shift in course programming assignments due to the migration to online instruction as part of the nation wide social distancing efforts to stop the spread of COVID-19, we are making the following adjustments course grading:

    (a) The four course programming assignments are worth 11% each or 44% total. Assignment 4 on MPI parallel I/O can be done separately or included as part of your group project per below.

    (b) The core project (without any optional parallel I/O component) is worth 36%.

    (c) The project with optional I/O component that counts as Assignment 4 is worth 47% total. Here you could earn up to 36% for the project and 11% for Assignment 4.

    (d) All summary point counts remain the same which is 20 course summaries for 1% each or 20% of the total course grade.