Journal of Adrian Gusa for Winter Quarter 2007

Adrian G Gusa

March 9, 2007

1 Week of January 10, 2007

1.1 Installed Python 2.4 on turing

This was the first week and my first assignment was to install Python 2.4 on the SMP machine turing.math.ohiou.edu in the Math Department. This version of python was needed for numpy, scipy and all other libraries to work properly.

2 Week of January 17, 2007

2.1 Read paper with focus on pseudo-code

This week I read the paper "Approximating a Wavefunction as an Unconstrained Sum of Slater Determinants"" and looked in more detail and the pseudocode of the algorithm in the paper.

I installed everything on my laptop and tested some H2 iterations. I also looked briefly at the code, especially the parts that are parallelized using IPython1.

3 Week of January 24, 2007

3.1 Prepared for the workshop next week

I continued to look at the code, run tests on turing.math.ohiou.edu, and on my laptop.

I read some tutorials about parallel architectures, PThreads programming, and MPI programming as part of the requirements for the upcoming workshop.

4 Week of January 31, 2007

4.1 Attended the workshop "Interactive Parallel Computation in Support of Research in Algebra, Geometry and Number Theory"

This week I was at UC Berkeley where I attended the workshop mentioned before.

I also worked on installing the needed packages, the code, and running the application on the Itanium cluster at the Ohio Supercomputer Center. I have read a little bit about how to work on supercomputers: OSC has PBS (Portable Batch System) for submitting computing jobs to a cluster. This means that jobs are described in a job file which is submitted to the PBS system using a qsub command. There are also other commands for investigating the progress of the job, stopping it, etc: qstat, qdel.

It seems that the code is running alright except when using a .dat file generated on a 32-bit machine. When using this saved file as an infile for the next iteration, some exceptions are thrown. This problem needs further investigation.

5 Week of February 7, 2007

5.1 Tests on 64-bit machines

1. I have run a test for 32-bit/64-bit pickle problems (break.py) on 3 machines: my laptop (32 bit), turing.math.ohiou.edu(64 bit), and on the login node of the Itanium cluster at OSC (64 bit). The code works just fine (no error output generated) on my laptop and on the Itanium cluster, but crashes on turing with the exception: "TypeError: list indices must be integers". This result is strange, and it seems we need to further investigate the 64-bit problems.

2. I have written the installation and running instructions for working on the OSC Itanium 2 cluster.

3. I have looked for performance monitoring tools available on the Itanium cluster. There is a package called perfsuite. I haven't used it yet, but I read a little bit about it. My impression is that this package monitors mpi execution and communication. Since we are using the cluster in a rather non-standard way, and only make use of mpi for starting the IPython1 engines, I don't think this tool can help. I have to look more into it, but my feeling is that it will be IPython1's job to provide peformance monitoring tools for communication and execution time.

4. I started working on the parallelization of G_iterP.G_apply_psi"

6 Week of February 14, 2007

I worked on tracing the bug which caused problems when using wave functions saved on 32-bit machines on 64-bit machines. By runnning a serial version of the program, the output of the program was available and I was able to locate the place where the problem appeared.

I also noticed a problem with the parallel program on turing: when starting multiple engines, some of them died and the log of the dead engines said that the problem was due to a failed import.

7 Week of February 21, 2007

After reporting the failed import problem to Fernando Perez, he suggested updating scipy, as this was a bug that has been fixed in newer versions.

I tested the serial and parallel version of the program with a 32-bit saved LiH molecule and they work fine with a small correction for the 64-bit problem that I came up with. I reported the issue to Fernando Perez, sent him a trace of the error appearing without the correction and asked for advice on an appropriate place where to fix the problem: sometime earlier, after loading the file would make better sense.

I continued working on the parallelization of G_apply_psi.

8 Week of February 28, 2007

I updated the libraries (numpy, scipy, mwadap) and the code (pycode) on the supercomputer at OSC. The mwadap library was now fixed in the svn repository in terms of the 64-bit problem. I started running an iteration from a 32-bit wave function, but because of the long-running serial job on the login node, the job was killed. This hasn't happened before so I guess there must have been a change in job policy on the login node of the supercomputer.

Given the failed test, I tried to switch to a different model, and have one of the IPython engines (engine 0) run FSpflat.py and use the remaining nodes as worker nodes. I had problems working with paths in launching FSpflat on one of the engines.

I did some tests to see how to handle paths when working on the supercomputer. I created a small test script which creates a file and writes something to it, and ran it on multiple engines. I was able to use the $HOME environment variable to establish the path to the home directory. This allowed to save the file at the desired location in the home directory. The home directory is mounted through NFS on all the nodes of the cluster, so the nodes can save files in the home directory.

9 Week of March 7, 2007

I continued working on solving the path problems that I had when trying to launch FSpflat on one of the IPython1 engines. I realized that the engines are started by default in the home directory. This is the same directory where the PBS job file was launched. I changed the PBS job file by adding a change directory to the code directory. Now, the engines would start in the code directory, and it is possible to run the FSpflat.py file (the code is located just fine by the python interpreter).

I submitted a long 24-hour job to the PBS system on the supercomputer, which would allow to perform two iterations. One problem that I noticed is that because of the length of the job and the number of nodes required (16), it took about 24 hours for the job to start. At this moment, the job is still running.

The problem with the amount of time it takes to start the engines, might turn out to be more serious in the future. In a distributed model, we might want a large number of nodes running for a long period of time. This might cause the job to be delayed for a long time before actually starting.

10 Week of March 14, 2007

Using the new model for the parallel FSpflat I have run two iterations on the supercomputer at OSC. I have obtained two new 64-bit wave functions starting from a 32-bit wave function.

I started working with the new fsnode.py program which introduces the concept of a data repository.