

Dear EECS 739 Students:

I consulted with a system administrator for the cluster today in regards to what I thought was a hardware issue when I attempted to run my example MPI/C++ code on 8 nodes of the cluster. (This was the issue I mentioned in my email last night.)

As it turns out, the issue was not a hardware issue but an issue with a scheduling policy used by this particular scheduler of which I was not aware. (Not all schedulers use the same policies.) It had to do with how jobs were scheduled on nodes and cores. In this class, students are to run their jobs on nodes only (which represents a distributed memory environment).

Based on the scheduling policy that took place once the current scheduler was employed, another modification is required in the script that you use in order to submit jobs to the cluster. The changes I describe below are necessary in order to run tests in a distributed memory environment and for students to be able to obtain enough nodes on the cluster (in a distributed memory environment) for their experiments.

Please be sure to use the updated script I posted at approximately 10:45pm this evening on the class website which reflects the changes summarized below.

Summary of changes:

1. The following line must now be added to the PBS script in order to achieve a distributed memory environment with this scheduler:

```
#PBS -W x=nmatchpolicy:exactnode
```

In order for all students in the class to be able to obtain enough nodes for their experiments, additional changes are also necessary. (As it turns out, the system administrator I talked to earlier was talking about cores when I thought he had mentioned nodes.)

This involves the following changes:

2. The experiments should now be run on any nodes that are available on the cluster as opposed to only the sled nodes. This means that the hardware nodes employed will be heterogeneous; keep this in mind when explaining your experimental results.

The updated line in the script which pertains to this is as follows:

```
#PBS -l nodes=4:ppn=1,walltime=00:10:0
```

(The reference to the sled has been removed. In addition, the reference to the memory has been removed since different nodes will have different amounts of memory available. The absence of the memory reference means you will use the default amount of memory available on each node.)

3. Students should now be running experiments on at most 16 nodes (and not 40-50 nodes as indicated in the description of Problem 3 in the handout).

Also, the system administrator I talked with indicated that the cluster is pretty highly utilized right now. Students are highly encouraged to run their experiments as soon as possible for Problem 3.

If you wait too long to run your experiments, you might get time on enough nodes. In that case, I recommend that you (1) run your code on fewer nodes or (2) run on cores representing a shared-memory environment in order to obtain partial credit (if I end up grading this problem on the assignment).

In order to pursue option (2), omit the following line:

```
#PBS -W x=nmatchpolicy:exactnode
```

and change ppn to a number greater than 1 on the following line:

```
#PBS -l nodes=4:ppn=1,walltime=00:10:0.
```

(Be sure that your mpirun line matches what you request on the above line; the argument after np is the number of processes as opposed to processors.)

4. Staging has been commented out of the script. It is no longer needed.

Also, here are some more detailed (perhaps clearer) directions in regards to the scaling experiment for Problem 3.

You are to assume that $n = 2^p - 1$ as I indicated in last evening's e-mail. (This simplifies the algorithm.)

I indicated in the Homework #3 handout that I would like you to first show that your code works with 1, 2, 4, and 8 nodes. These values of p correspond to $n = 1, 3, 15,$ and $255,$ respectively.

You are then asked to experiment with $n = 1000$ and $n = 10,000$ (according to the handout). In order to satisfy the above ratio, you should modify this to $n = 1023 = 2^{10} - 1$ and $n = 16383 = 2^{14} - 1$. (These are the values of n that are closest to but larger than the original values of n .)

You are also to perform a scaling experiment. Naturally, the scaling experiment will be atypical given the above relationship between n and p . I recommend using the following modified directions for your scaling tests. Consider the range of matrices from $n = 1 = 2^1 - 1$ to $n = 16383 = 2^{14} - 1$ (and choose the value of p in each case that obeys the relationship between p and n). Consider several values of n (and hence also p) in this range (while respecting the relationship between n and p). Certainly I recommend you use your results from $p = 1, 2, 4,$ and 8 above (in addition to some other points). If you are able to obtain enough nodes for your experiments, please also consider what happens for $n = 32767 = 2^{15} - 1$ and $n = 65535 = 2^{16} - 1$.

Due to the relationship between n and p , such an experiment does not represent speedup. In the spirit of what was asked, please make a table with the following three columns: n , p , and wall clock time in order to report your findings. Those who are feeling more adventurous could make a 3D histogram plot with n and p on the "x" and "y" axes and wall clock time on the "z" axis.

Please let me know if you have any questions on any of these directions.

Thanks, Dr. Shontz