

# September 22nd, 2015 Pre-Class Questions

Elliot Cartee

## Question 0

I have been running behind on the pre-class questions, and started these on September 23rd. I probably spent about half an hour on these questions, although this is after hearing some of the answers in class and I am not quite finished yet.

## Question 1

I'm somewhat confused by how futures work, but that seemed like more of a side note in the slides than a critical concept or anything.

## Question 2

### (2a)

Let's suppose that we run a Monte Carlo simulation with  $N$  the number of trials,  $p$  the number of processors,  $(t_{trial})$  the time per trial, and  $(t_{update})$  the time to update the global counters in the critical section. Let us also suppose that we are using a batch size of  $b$ .

Then the total amount of time spent running the trials will be:

$$\frac{N}{p}t_{trial}$$

While the total amount of time spent updating will be:

$$\frac{N}{b}t_{update}$$

therefore the total execution time will be:

$$\frac{N}{p}t_{trial} + \frac{N}{b}t_{update}$$

This is a rather simplistic model, and doesn't do very well in the limit as the batch size  $b$  approaches the problem size  $N$ , where a lot more than  $N$  trials might be run before the code checks for convergence.

**(2b)**

I couldn't get my code to run correctly on the cluster right now, I will update this later when I get it to work.

**(2c)**

Based on my model, you are best off picking a very large batch size. As mentioned before, my model is rather simplistic and falls apart when considering large  $b$ , as in this case you may be doing a lot more than  $N$  trials. So the best choice of  $b$  should be one that is large, but still small enough compared to  $\frac{N}{p}$  that you will not have to do too many extra trials. In fact, if you know  $N$  and  $p$  beforehand, you could use this to automatically choose a batch size that is large, but still small relative to  $\frac{N}{p}$ .

### **Question 3**

I modified the `workq` files as can be seen in this repository. I ran my code on the cluster and it seemed to work as expected.