CS 495 Assignment 4a Horovod

Parker Joncus A20350578

October 6, 2018

Exercise 1: Horovod on single skylake node

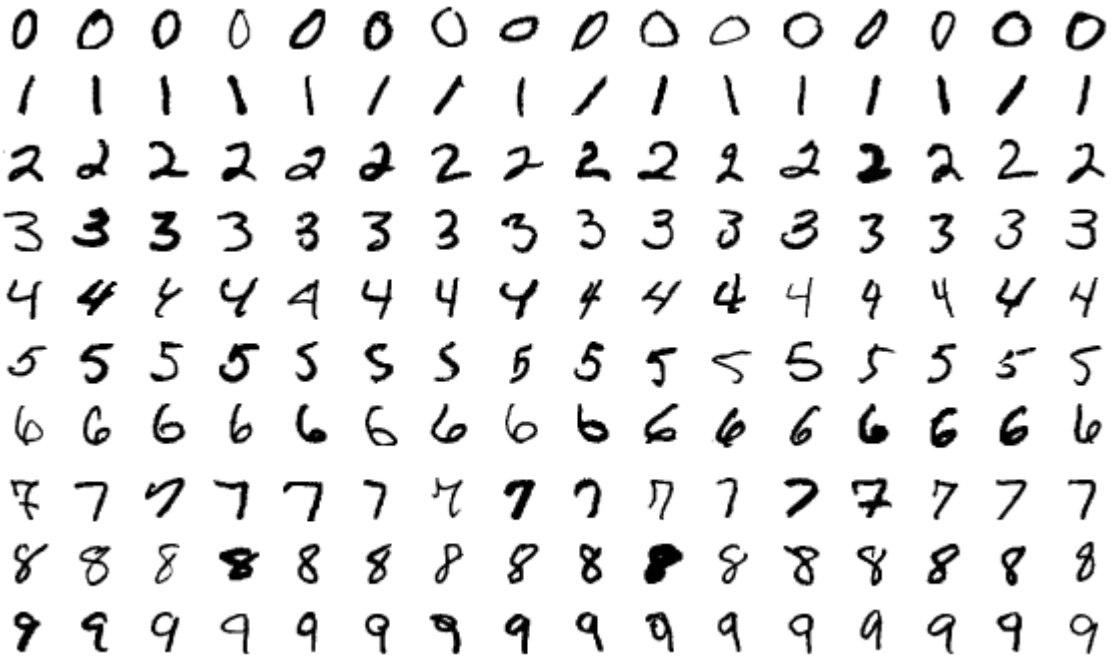
(a) In order to compile Horovod, you can use python pip to install it. There are some prerequisites that you need before you can install it. These dependencies are tensorflow, torch and torchvision with pip. You can also use keras. Once these are installed, you can simply use pip install horovod. You do need to make sure the pip is using the correct mpi however. Once installed, the horovod github files can be cloned. Inside the github files are example data sets that can be used to test horovod.

(b) I ran the tensorflow-mnist.py file to test horovod. I ran this on 1,2,4,8 and 16 nodes to see how the application scaled. I piped the results into the files called 1node, 1node2, 1node4, 1node8 and 1node16 respectfully. It can be seen in these files that with more processes, the application takes less steps, but it takes longer for each step. The loss at each precicion level is hard to compare because they are sporadic and are not constant throughout the runtime.

(c) The power with 1 process is about 100 Watts on average. The power with 2 processes is about 170 Watts on average. The power with 4 processes is about 200 Watts on average. The power with 8 processes is about 215 Watts average. The power with 16 processes is about 225 Watts on average. From this we can see that obviously with fewer processes, we use less power, but we can also see that the power usage increases fast at first and slows down the increase as the number of processes increase.

(d) The input of this file I am running is the mnist data set. This data set is a bunch of pictures of hand written letters and numbers and the application is trying to classify what letter or number is in the picture. The output of the application is time and loss while it is running. Once the application is finished, there is a directory from each process with gunzip files. I am still trying to figure out how to read these files and if there are images or something else.

(e) The input imgaes look somthing like:



(f) I am curious to compare if using different deep learing tools will be faster or better. So far I hve only used tensorflow, but I also want to try keras and pytorch and compare. I also want to see if there is a difference to compile tensorflow myself. Last I also want to find larger and different data sets to test.

Exercise 2: Horovod on 4 skylake nodes

(a) I was able to compile horovod on 4 nodes, but for some reason I was having trouble getting nfs to work, so I was not able to run it across 4 nodes yet. I am going to continue working on it and will definitely have some results before the next class.

Exercise 3: Horovod on GPU

(a) I was also not able to use horovod on a GPU yet, but alex was able to test on the GPUs. He sent me some messages saying that a single node, single gpu took about 5 minutes to completely run. He ran horovod on an older nvidia driver since the one before had some bugs with tensorflow, but the older driver took 7.5 minutes to complete. I am going to work with alex more and figure out how exactly to get horovod to run on the gpus so that I can test it as well. This is still a lot faster than the cpu which takes about 25 minutes.