April 3rd, 2017

Project Proposal

Andrew Showers, Dylan Elliott, Peter Horak

We propose constructing a multi-layer perceptron model on the Blue Gene /Q AMOS system at RPI in order to investigate a strong scaling test at the perceptron level. The model will allow for forward propagation through the network as well as backpropagation to be used for training the network. The multi-layer perceptron model is a great candidate for parallelization as it includes a large network of cells that perform mostly the same exact functions. This problem is certainly not “embarrassingly parallel” as a good deal of effort will need to be taken in order to choreograph the communication between single perceptrons in the model. Additionally, several synchronization levels will have to be employed in order to ensure that forward propagation and backpropagation occurs in proper layer order throughout the network.

We will utilize the MPI library in order to coordinate parallel tasks for designing the perceptron model. The model will consist of the typical fully connected feed forward neural network and we are anticipating that we will use sigmoid activation for each neuron. We will parallelize the model at the neuron level, therefore ranks will be tasked with sequentially calculating any neuron outputs that they are responsible for. Parallelizing the model at the neuron level will allow for us to investigate directly how the network architecture affects the parallel performance of the MPI program. The parameters for our experiments will fall into 2 categories: configurations within the program and configurations on the Blue Gene hardware. The prior will allow the user to specify the number of layers, nodes per layer, and ranks per layer. The latter will allow the user to specify the number of ranks per node on the Blue Gene /Q.

Calculation of the neuron outputs and calculation of the gradient signals will be done as graph computations and these signals will be passed as MPI messages between layers of the model. It is therefore a requirement that the neuron model be converted to a computation graph in which the equations of a forward propagation of a signal and the equations of a backward propagation of the gradient can be derived. The forward pass graph computation will follow the functions that define the neuron and the backward pass will follow the chain rule of derivatives of each node in the computation graph. Each rank will therefore do an identical set of calculations for each neuron that it is responsible for and the passing of signals across each layer will have to be synchronized.

Our first goal will be to create a small neural network to handle a simple test case. We will take two boolean values as input and the neural network will be trained to match the expected output of an XOR gate. This simple test allows us to easily generate and validate training samples. Once this small scale version is working we will attempt to expand to a more difficult problem, such as using images as input data and attempting to classify single characters.