SE 390 - Business Plan

Brain Simulator Parallelization

Prepared by

Team Spike

Greta Cutulenco - gcutulen

Robert Elder - relder

James Hudon - jbhudon

Artem Pasyechnyk - apasyech

Presented to

Paul Ward

University of Waterloo

Software Engineering

December 14, 2012

Introduction

This document outlines the business plan for our project, “Brain Simulator Parallelization”. We will firstly describe the problem at hand and how we plan on solving it. Then, we will discuss project goals and the significance of succeeding at these goals. Finally, we will cover feasibility, risks and legal considerations.

Because our project does not aim to become a product and is instead a research project, this document will emphasize its significance rather than its business value. Terry Stewart, the researcher for which this solution is being built, will greatly benefit from our solution in his research, and so will be referred to as our “customer”.

Problem

The Theoretical Center of Neuroscience at the University of Waterloo has built a system dubbed “Nengo”. It is a neural compiler: a tool for taking a high-level description of the processing algorithms and converting that into a detailed low-level model of how individual realistic neurons can be combined together to implement that algorithm. It has been used to create the world's largest functional brain model, comprising 2.5 million neurons implementing a variety of brain areas.

The issue is that the models only use some very simple Java multithreading to be parallel. As such, to get one second of data requires the models to run for hours. This means that for all the research that depends on empirical data (most research, hopefully), there is a considerable slowdown that ultimately delays the development of research and hinders the quality of the data the research depends on.

Solution

This system could be significantly improved with some decent parallelization across computers and even across implementations (there are Java, Python, Matlab, and CUDA implementations at the moment, and there is an ongoing project to have an FPGA version as well). Thus, our proposed solution is to develop a distributed system with an interface for all these different models. This system would allow any two models to be set up in a network and to communicate with each other.

One way to distribute the work outlined above is to divide all the simulated neurons into distributable groups and establish a lightweight communication protocol between these groups. Then, at every tick of the clock, rather than computing the propagation of neural activity in a sequential fashion, we compute each group's activity in parallel and pass the results along to the next node for the next tick.

Project Goals and Significance

The project's goal is to parallelize the brain simulator, as discussed in the previous section. Furthermore, we expect to see a performance increase of one to two orders of magnitude for the large brain simulations.

The significance of this performance increase will be immediately apparent to any researcher in the Theoretical Center of Neuroscience or any researcher that uses the Nengo brain simulator. In particular, the researchers would have the ability to test their hypotheses faster and thus speed up the development of research in the neuroscience domain. Keeping this in mind, the significance of this project is also correlated with the significance of the original Nengo project. That is, if the 2.5 million neuron brain simulator that we are parallelizing turns out through studies to not be of any research significance, then making it compute faster will also not be very significant in that domain of research. However, if Nengo does prove to be a trustworthy model of the human brain, then allowing the model to compute at a scale closer to real time will be greatly beneficial.

Goal Success Metrics

The directly measurable success metric here is a wall-clock performance metric. The less time the model takes to compute the results for a researcher, the more we have succeeded at our goal. Thus, regular benchmarking should be done to gauge our level of success as we develop our solution to the parallelization problem. Discussions with the customer have suggested that we should be able to achieve a performance increase on the order of 100 times, however he noted that his theory would require further investigation to validate his claims.

Feasibility

Our initial analysis shows that the current models run one tick through all neural groups in a sequential fashion. That is, when the clock ticks, the program computes the result of one neural group, then propagates that result to the next group and calculates that one's result, and so on and so forth. Ultimately, one clock tick will trigger all neural groups to start computing at the same time and pass their result to the next group for the next tick. Therefore, for our project to be feasible, the overhead caused by the cummulative message-passing must be less than the performance hit caused by running the whole process sequentially. Our plan is to assess this as early as possible to know whether our project is feasible or not.

Legal Considerations

The Nengo project is licensed under Mozilla Public License 1.1 (MPL 1.1). In a nutshell, this means that if we are to include Nengo code into our project, whether it is commercial or not, that code must remain licensed under MPL.

We have decided to follow in our parent project's footsteps and also license our code under MPL. This will allow a lot of flexibility for people who wish to modify, redistribute or use our code while still giving credit to the code's authors.