June 3, 2010

Revised: Dec. 20, 2011

Welcome to BrainGrid! This document contains everything you need to get started.

# Step 1: Get a Subversion client

TortoiseSVN is popular on Windows, while MacOS and Linux have several free options.

# Step 2: Enlist

BrainGrid source code is kept on homer.u.washington.edu. The Subversion repository is accessible via SSH. To get access, you must provide Professor Stiber with an RSA public key. To do this, run the following at a Linux or Mac command line:  
   
ssh-keygen -t rsa  
   
Accept the default file name, and the password can be left blank (if you're feeling paranoid, a password can be entered.)  
Two files will be created. The private key, id\_rsa, should be kept on the computers you plan to run a subversion client on. Send the public key, id\_rsa.pub, to Professor Stiber.

After Professor Stiber has granted you access, you must configure your Subversion client for ssh access. The steps vary depending on the client.

From a Linux/Mac command line, svn can be configured by editing ~/.subversion/config to include the following line under [tunnels] section:  
biocomp = /usr/bin/ssh -p 22 -q -l biocomp -i /Users/stiber/.ssh/id\_rsa  
   
This creates a "svn+biocomp" tunnel scheme (where you need to change "/ Users/stiber/.ssh/id\_rsa" to be the full path to your private key).   
For configuring other clients, see <[http://wiki.site5.com/SVN/Subversion\_](http://wiki.site5.com/SVN/Subversion_" \t "_blank)(SVN)\_Setup\_Guide > near the bottom for clues. From the command line, you can try svn --verbose list svn+biocomp://homer.u.washington.edu/BrainGrid/trunk

Once you have a connection set up, you can do a check out do get a local copy of the source code.

# Step 3: Build

After getting a local copy of the source, you can build. Run ‘make’ to make the braingrid project.

Three executable will be created:

growth - Single-threaded version.

growth\_omp - Multi-threaded version (OMP).

growth\_gpu - GPU version (C UDA).

If “PERFORMANCE\_METRICS” is defined, cumulative executable time of each kernel function will be shown.

“-maxrregcount” is a nvcc compiler option, which set the maximum number of registers used within each kernel function.

“include/bgtypes.h” defines the type of floating point (double or single).

To make the Matlab post-processing modules, run ‘make’ from the MATLAB subdirectory.

# Step 4: Running the simulator

To run the simulator, invoke the exe you just built. BrainGrid requires at least one parameter:

./growth -t input.xml.

This input file provides simulation parameters and the name of the output file. If you wish to provide the output file name on the command line, you can use this syntax:

./growth -t input.xml -o output.xml.

“-w” option specifies the name of an output file, which saves the memory image of neurons, synapses, firing rate, and radii of the final state.

“-r” option specifies the name of an input files, which was saved by “-w” option. When the option is added, the simulator restores the previous state and continues simulation.

“-d” option is for the GPU version only, and specifies the device ID of GPU.

In windows similarly without the ‘./’:

growth -t input.xml

or

growth -t input.xml -o output.xml.

# Step 5: Analyze the results

You can run Matlab from any of the UW1-320 Linux machines. The name of the Matlab script is ‘batch’. In order to invoke it correctly, you need to add the path to the matlab post-processing modules:

addpath '/home2/username/BrainGrid/MATLAB' –BEGIN

Substitute the string in quotes for the full path to your MATLAB subdirectory.

Name your output file ‘historyDump.xml’, copy it to ‘/home2/username/BrainGrid’ and run ‘batch’.

If run successfully, 5 plots will be generated and saved as 01 – 05\_historyDump.jpg in the current working directory.

# Step 6: Getting acquainted with the code

The main simulation loop is in the Network class. The advance() method invokes interface implementations of advanceUntilGrowth().

# Misc:

Running XWindows:

#bash shell

#one time

startxwin

#every time

DISPLAY=:0;export DISPLAY

ssh -c arcfour,blowfish-cbc -XC hydra.uwb.edu

DISPLAY=:0;export DISPLAY

ssh -c arcfour,blowfish-cbc -XC uw1-320-17.uwb.edu

To make processes run in the background after logging out:

screen

# Run command

./growth -t input.xml

# Detach screen with control-A followed by ‘d’

^A d