

Here are the guidelines for downloading and running certain tutorials in the catacomb simulations. These guidelines are based on exercises for my course at Boston University - PS530: Neural Models of Memory Function.

DOWNLOADING CATACOMB.

1. Create a directory on your computer for the project work.
2. Open an internet browser.
3. Go to "askja.bu.edu"
4. On the left is a link entitled: PS530: Neural models of Memory Function. Click on this link.
5. On the new page, there is a link entitled "Catacomb 2.034 package". Right click on this link to download it. WARNING: Do NOT choose the JAR file.
6. Choose save file and save it to the directory you created for PS530 work on your computer.
7. Now click the "Back" button to go back to askja.bu.edu.
8. In the askja.bu.edu window, in the larger window on the right, you should click on the link that says "download it now." This opens a window with general instructions for downloading the catacomb program as well as the winzip program and java itself. You are NOT going to download catacomb from this window, since you already downloaded it from a different location, but you need to download winzip and javasoft and can follow instructions for unzipping catacomb. The following steps provide a little more detail about downloading and unzipping catacomb.
9. If you do not have winzip, then click winzip download and follow installation instructions.
10. Once winzip is installed, you will install the catacomb program.
11. Click on the icon which you just downloaded (named something like "catacomb-2.034.zip"), and drag this icon onto the icon for winzip.
12. This should open a winzip window with multiple button choices. Click on "I agree"
13. Click "next"
14. It will ask "do you want to add folder..." click "No."
15. Next window states "Click unzip now to unzip to selected folder. But first you should click "select different folder" and put it in some location that you will be able to find easily (e.g. C:).
16. Click on "unzip now".
17. Next window says "Confirm file overwrite" - Click on "Yes to all."
18. When it is done unzipping, it should open a window showing the location of the folder you just created (catacomb-2.034). Please be sure to note the location of this folder (where it says "Files were unzipped to:).
19. When you click on the folder you just created, you will find the contents to be an icon labeled "ccmb.jar" and a folder labeled "ccmb_help". You are now ready to run catacomb.
20. Double click on "ccmb.jar" and see if it opens up a window. If it does not you may need to go back to the main window in askja.bu.edu and click on the link to "javasoft" to download a version of java that will run catacomb on your computer.

DOWNLOADING ASSOCIATIVE MEMORY EXAMPLE

You also need to download a script demonstrating associative memory function.

21. In your web browser, look at the web page at askja.bu.edu again.
22. On the new page, there is a link entitled associative-learning.ccm. Right click on this link to download it to the same directory where you have the ccmb.jar file.

RUNNING CATACOMB

23. On a windows machine, you can run catacomb by double clicking on "ccmb.jar" file. On a linux machine use the command listed on the web page.
24. Starting catacomb will open a window labeled "catacomb.env.Manager" In that window, click on the button labeled "workbench".
25. This opens a window labeled "catacomb.env.Workbench". Go to the upper right corner of that window and click the "help" button.
26. This opens a window labeled "catacomb.env.HtmlPage". You should click on the link labeled "overview" and read the overview of catacomb.
27. After reading the overview, you should click on the link labeled "Graphs" to find out details about looking at the graphs. As a simple overview, clicking on your left button zooms in, clicking your right button zooms out, holding your left button down and moving the cursor will move the whole graph in the desired direction, holding down the left button and going left then right will create a red rectangle which can be used to select a portion of the graph for closer viewing.
28. After reading the overview, click on the link at the top labeled "examples."
29. Read about the different examples, then click on the link labeled "Integrate and fire cell models" (the 4th in the list).
30. This opens a page labeled "Cell Models". The second item "Current clamp" models passive membrane properties. On the right, in the box starting with the word "Load", click on "cell1a.ccm".
31. This brings back the window labeled "catacomb.lab.Workbench", but now there are icons in the center window. Click on the "display" icon on the right (with white waveform on green background). This brings up a set of menu items on the right which pertain to the display icon. Then go to the "display" label, and click on the button next to this labeled "show." This opens a window labeled "catacomb.standard.DisplayList" This has a blank graph.
32. In the larger window (catacomb.lab.Workbench) go to the left hand menu and click "build" then "run". This runs the simulation.
33. A graph should now appear in the graph window (catacomb.standard.DisplayList). If nothing appears there, try clicking the button on the upper right labeled "resize" - this is always useful for finding graphs in windows.
34. Go back to the larger window (catacomb.lab.Workbench). If the icons do not all appear, then click on the little rectangle in the upper right hand corner of the icon window. This will open a menu, from which you can choose the top option "Reframe".
35. Click on the circle in the center (the cell.integratorCompartment). This will open a new menu on the right. This new menu has components corresponding to the elements of the passive membrane dynamics modeled with the equivalent circuit, including the time constant and the capacitance. Changing the time constant changes the time it takes for the membrane potential to reach 1-1/e of its maximal height. Note that the change in time constant is here being implemented purely by a change in resistance (which can be seen as a change in conductance G in nanoSiemens as shown in the blue box in the right hand menu).
36. Click on the rectangle under the label "timeConstant" and slide it to the right to higher values. Then click on "build" and "run" again to see the new curve. (If you can't find the display window again, you can click on the display icon to get its menu on the right, then click on the "show" button).
37. IMPORTANT NOTE: To set a parameter to a specific value, put the mouse over the parameter and hit the space bar. This will open a text box

with the parameter value. You can type in a new parameter value and then click the "OK" button.

38. SEND ANSWER*****When you increase the time constant (increase resistance), does the asymptotic value of the membrane increase or decrease?
39. SEND ANSWER*****What approximate value of time constant causes the asymptotic (final) value of membrane potential to be approximately -50 mV? (You only need to give this answer to an accuracy of ONE decimal point). REMEMBER THAT EACH TIME YOU CHANGE THE PARAMETERS YOU MUST CLICK "build" and "run" again, then "show" to see the new graph with the changed parameter effect.
40. You should explore the effect of changing timeConstant and other parameters, to get a feel for how they influence the cell's response.

RUNNING SPIKING NEURON WITH AFTERHYPERPOLARIZATION

This is an INTEGRATE-AND-FIRE model, in which the passive membrane properties are combined with a highly simplified version of action potential generation. When the neuron reaches a specific threshold, the cell generates a spike (a preprogrammed change to zero mV, followed by a function which represents the afterhyperpolarization). These models are useful for simulating large networks more rapidly than detailed biophysical simulations.

41. Go back to the window with the integrate-and-fire examples, and scroll down until you can click on cell3.ccm. This model adds a function to represent the afterhyperpolarization in terms of the potassium conductance.
42. Click on the display icon and follow previous commands to show the membrane potential, and do "build" and "run" to run the simulation.
43. SEND ANSWER*****How many spikes is the cell generating with the default cell parameters within the default simulation period of 100 msec?
44. Click on new light green icon (cell.SpikeResponseFunction) and change the fallTime and riseTime to 0.5 msec.
45. SEND ANSWER*****How many spikes does the cell now generate within the 100 msec period?

RUNNING SYNAPTIC POTENTIAL ("Synapses")

46. Go back to the help page window (catacomb.env.HtmlPage) and scroll down until you can click on cell4.ccm.
47. Click on the display icon and click the show button to see the membrane potential, then do "build" and "run" to run the simulation.
48. Click on the icon with three small circles in it (cell.SynapsePopulation). In the menu on the right, change "duration" to any value larger than 100 msec. (this determines how long the duration for each cell is included in calculations - if you leave it at default it will cause very odd shapes in later synaptic potentials)
49. In the menu for the SynapsePopulation, increase G₀ to 2.0 nS.
50. SEND ANSWER*****What is the approximate voltage value of the peak of the first synaptic potential? (in mV - no decimal points necessary)
51. In the menu for the SynapsePopulation, increase the fallTime parameter on the right and see how it affects the membrane potential traces.
52. SEND ANSWER*****Change the fallTime parameter. At what value of fallTime does the cell first generate a spike? (Give this value to just one decimal point).

RUNNING ASSOCIATIVE MEMORY EXAMPLE

This model simulates two different populations of neurons with modifiable excitatory synaptic connections between the two populations. Two different input patterns cause different patterns of spiking in the two populations, and

spike timing dependent Hebbian synaptic modification of excitatory synapses encodes the association between the two patterns. Subsequently, presentation of the pattern in the first population causes activity to spread across the modified connections to cause retrieval of the associated pattern of spiking in the second population.

53. Go to the catacomb.env.Workbench window and click on the "file" menu, then select "open" from the list.
54. You should see the file named associative-learning.ccm that you downloaded from askja.bu.edu to this folder. If you don't see it, go back to askja.bu.edu to download it (as described above). Select associative-learning.ccm and open it.
55. Click on the display icon in the upper right of the network. Then go to the "plot" option and click the button saying "lines". This opens a short menu from which you should choose "grayscale."
56. On the left, click "build" and then "run."
57. Go to the right and next to display click the "show" button.
58. The display shows time plotted horizontally and different neurons plotted vertically. The first population is the bottom ten lines and the second population is the top ten lines. Spikes appear as white rectangles. In the first ten milliseconds, one pattern activates the first population followed by another activating the second. At about 45 msec, input activates the same pattern in the first population, and the activity spreads across modified connections to activate the associated pattern in the second population.
59. Click on the green icon which is lowest on the screen (network.SynapseRecorder).
60. On the right, click the "show" button.
61. In the new window (catacomb.network.SynapseRecorder) click "reset" and then "play". This runs a movie which shows the evolution of weight changes. If you don't see little rows of circles on the bottom and left, click on the rectangle in the upper right corner of the gray/black screen, and choose "reframe" from the menu. This will make the whole image more clearly visible. The rows of circles represent the pre and postsynaptic neurons, and the matrix between shows the change in weights (like an outer product). Circles turn white when cells spike. Weight squares turn white when the change from 0 weight to a nonzero value (there is no border between weights, so four adjacent weights appear as one block).
62. If the movie runs too fast, click on the rectangle in the slider above the "reset" button" and move it slowly to look at different time steps.
63. SEND ANSWER*****At what time does the square matrix of weights change?
64. Click on the population representing the second population (the light blue box with the elements of an integrate and fire neuron that you saw previously). The right side will read network.NeuronSocket, with a button below which says "postsynaptic".
65. Click on the button saying "postsynaptic" which opens a menu. Go to the very bottom of the menu to "edit."
66. This opens a new window (catacomb.network.Neuron) with elements of the neuron. Click on the round "synapse" icon on the left.
67. On the right, click the button next to G_ModRule on the right (the button says "hebbian"). This opens a menu. At the bottom choose "edit"
68. This opens a new window (catacomb.standard.Profile) with a graph showing the synaptic modification function (a step function starting a bit after zero, which is the time of the presynaptic spike).
69. Click on the upper right corner of the step function and move it to the left to shorten the window of synaptic modification. Move it to about 2.5 msec and click "build" and "run", then go back to the display.

70. SEND ANSWER*****What has changed in the display?
71. Move the end of the step function to 5 msec to restore the display.
72. SEND ANSWER*****Write out the vector of presynaptic activity and the vector of postsynaptic activity (write both as row vectors).
73. For studying of for the exam, I would recommend doing the outer product of these two vectors, and then doing the matrix multiplication.