INTRODUCTION TO LAB SIMULATIONS

0.1 Stimulating a Neuron

Purpose: to learn how to use NEURON's graphical interface

The primary aim of this session is to introduce features of NEURON's graphical interface that are used throughout these exercises. These features are presented in the context of an experiment that explores the response of an excitable cell to injected current pulses.

The experimental preparation is a spherical cell with diameter 17.8 µm (surface area = 1000 µm^2). Its membrane contains sodium and potassium channels that are described by the Hodgkin-Huxley equations. It is impaled by two electrodes: one for injecting a stimulus current, and the other for recording membrane potential V_m (Fig. 1).

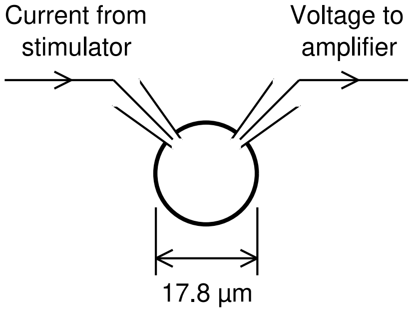


Fig. 1. The experimental preparation

The action potential's amplitude and time course are relatively independent of stimulus amplitude and duration. This independence distinguishes it from responses to subthreshold stimuli. However, stimulus intensity can affect spike timing. A powerful stimulus will trigger a spike very quickly, but a weak stimulus that is barely above threshold may be followed by a very long interval before the spike occurs. A stimulus that is too weak won't trigger a spike at all, regardless of how long it lasts.

The stimulus used in this experiment is a rectangular current pulse with adjustable duration and amplitude. By changing these parameters, you will explore the effects of stimulus intensity on spike amplitude and latency. In the course of the experiment, you will also learn how to

- start a simulation run
- measure results from a trace in a graph

- change the values of parameters
- accumulate several traces in a graph
- erase all traces or individual traces from a graph
- add labels to a graph
- reposition labels in a graph.

Start the simulation in `thresh.hoc`

Launching NEURON

Open a terminal, cd to the directory that contains `thresh.hoc`, and execute the command

```
nrniv thresh.hoc -
```

This will start NEURON, which prints a message to the terminal that includes a version number and copyright notice, and ends with the `oc>` prompt for NEURON's interpreter; the interpreter is discussed briefly in the Appendix to this exercise.

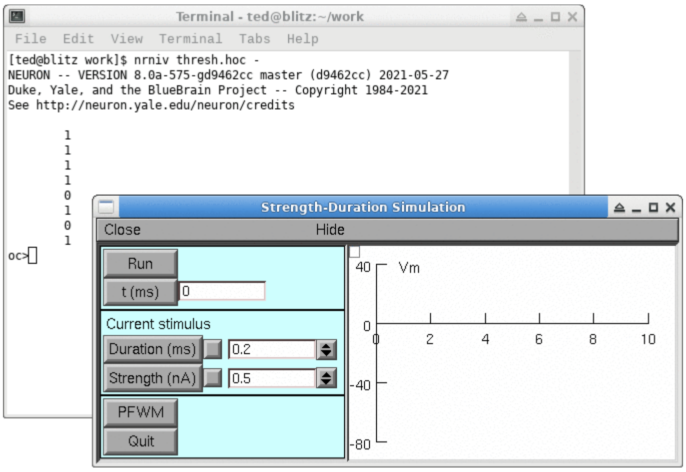


Fig. 2. The "virtual lab rig" for this experiment lies on top of NEURON's interpreter window.

There will also be a new "virtual lab rig" window that we'll be using in this experiment; its title bar is labeled "Strength-Duration Simulation". Just below the title bar is a narrow strip with two buttons labeled Close and Hide. If you click on either of these buttons, the Strength-Duration Simulation window will disappear, and you will have to exit NEURON and start over.

Exiting NEURON

The most convenient way to exit these exercises is to click on the Quit button at the bottom of the "virtual lab rig" window.

Running a simulation

The left panel of the Strength-Duration Simulation window is split into three fields. To start a simulation run, click on the Run button in the top field. NEURON will compute the response of the cell to the stimulus, and plot the results in the graph panel (Fig. 3).

The stimulus current pulse begins after a brief interval. It produces a depolarizing ramp in the V_m plot. When the current pulse stops, V_m sags slightly at first, but it soon sweeps through the typical trajectory of an action potential.

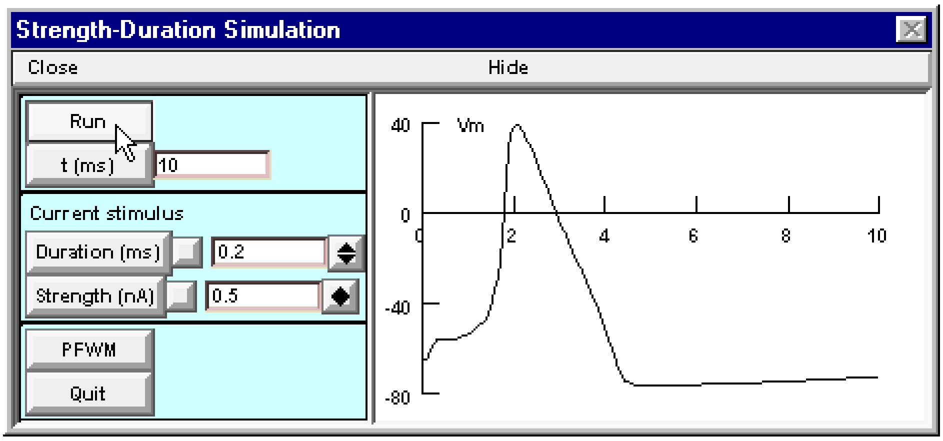


Fig. 3. The Strength-Duration Simulation window. The right panel of this window is a graph that shows the time course of membrane potential V_m. The horizontal axis is calibrated in milliseconds (ms), and the vertical axis is calibrated in millivolts (mV).

Why doesn't the "t (ms)" button do anything?

"Sometimes a button is just a button." This particular button isn't supposed to do anything. It's just a reminder of the meaning of the number in the adjacent box. This readout gives a visual indication of the progress of a simulation.

1. What is the value of V_m at the peak of the spike, and when does the peak occur?

You could try estimating these values by eye, but it is easier and more accurate to use the graph panel's crosshairs.

Condition: stimulus duration = 0.2 ms

Spike peak amplitude (mV) __________

Time of peak (ms) __________

Measuring values from a graph

Place the cursor over the trace in the graph panel and press the left (L) mouse button. Crosshairs will appear on the trace, and under MSWindows the corresponding x and y coordinates will be displayed in the window's title bar (Fig. 4). To make these values appear in NEURON's interpreter window, press the space bar on the keyboard while holding down the L mouse button.

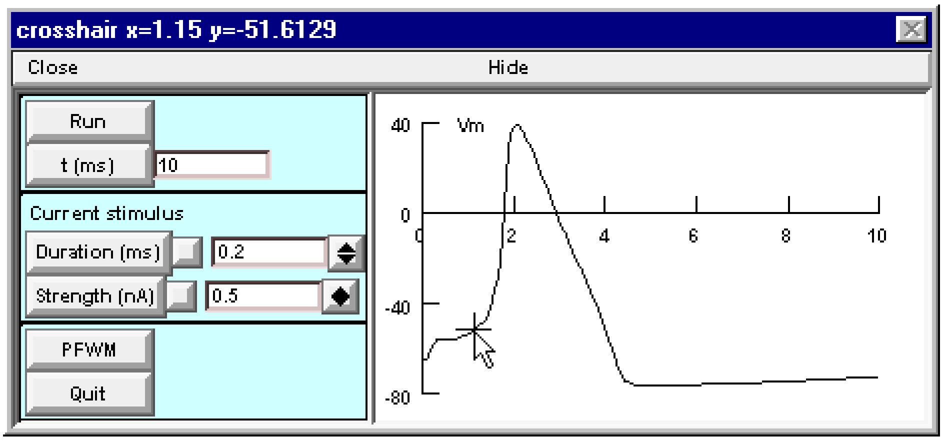
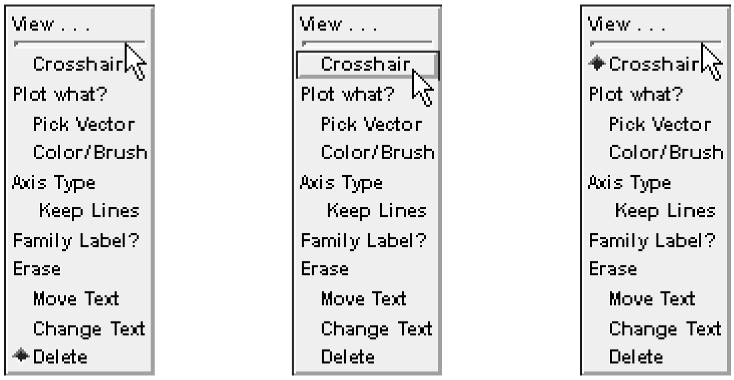


Fig. 4. Using crosshairs.

You may notice that the x and y values do not vary continuously. Instead, they change in a stepwise manner, because NEURON computes simulation results at discrete points in time and space. This is a consequence of using a digital computer to approximate the behavior of a continuous system. The errors that can result from this approximation, and strategies for minimizing them, are topics of ongoing research. For the purpose of these exercises, the approximation is more than good enough!

Making sure the graph is in Crosshair mode

Right (R) click inside the graph panel and hold the mouse button down. This brings up the primary graph menu (Fig. 5A), which lists several modes including Crosshair. In the example shown here, the radio button next to Delete is on. L clicking on a trace while Delete is on would make the trace disappear from the graph. To turn off other modes and restore Crosshair, hold the R mouse button while dragging the cursor down so that the Crosshair button is highlighted (Fig. 5B). Then release the mouse button and the graph menu will disappear. The next time you check the graph menu by R clicking in the graph panel, you will see that it is safe to proceed because Crosshair is on (Fig. 5C).

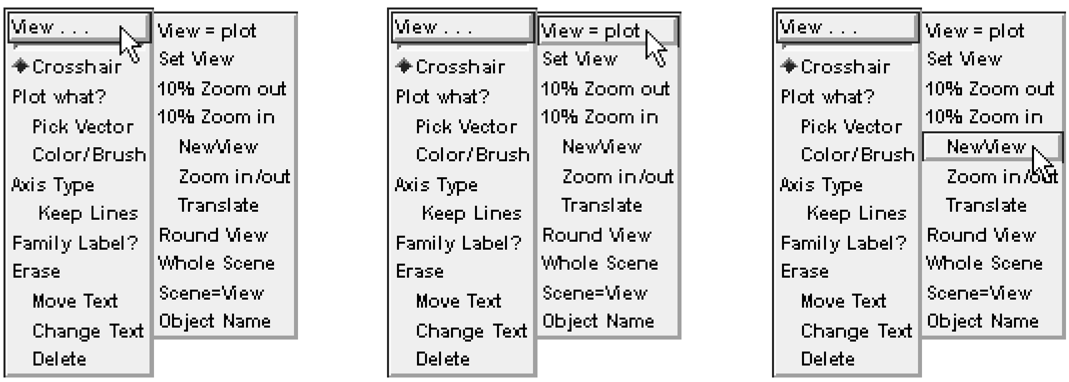


A B C

Fig. 5. If you have been using any of NEURON’s other graph modes (especially Delete, which is described later in this lab session), it is a good idea to make sure that Crosshairs mode is turned on before you click on a trace.

Enlarging part of a graph

Sometimes limited screen resolution makes it hard to see what is happening or to place the crosshairs accurately. For example, suppose we wanted to place the crosshairs as close as possible to the very peak of the spike triggered by the stimulus (Fig. 4). This can be made easier by using NewView to enlarge a selected region of a graph.



A B C

Fig. 6. Turning New View on.

To activate a graph’s NewView mode, bring up its primary graph menu by R clicking in the graph field. Then reveal the second panel of the graph menu (the "secondary graph menu") by holding the R mouse button down and shifting the cursor slightly upward to highlight the View . . . button (Fig. 6A). Still holding the R mouse button down, drag the cursor horizontally over to the second panel (Fig. 6B). A steady hand is helpful, because the second panel will disappear if the cursor moves down prematurely. Finally, drag the cursor down to NewView (Fig. 6C) and then release the mouse button. Now L click inside the graph panel and drag a rectangle around the region of interest (Fig. 7A). This will pop up a small new graph window that contains the region of interest (Fig. 7B). This can be resized by dragging a corner or edge until the graph is big enough to see clearly (Fig. 7C).

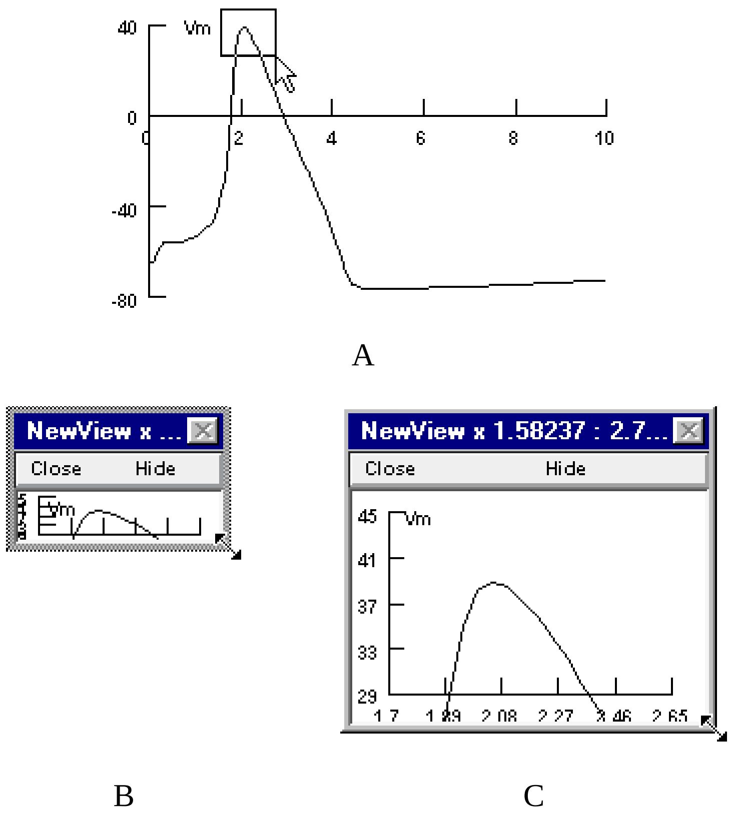


Fig. 7. Selecting a new view.

Return to Crosshair mode!

After creating a new view, you might be tempted to immediately L click on the trace in the new window, expecting that the crosshairs will appear. However, the graph is still in NewView mode, so this would just spawn a tiny new graph window. This won't do any harm, since the useless tiny window can be dismissed by pressing its Close button, but it wastes some time. Before making any measurements in the new window, you should restore Crosshair mode by following the steps shown in Fig. 5.

2. How does changing stimulus duration affect the spike?

Use the spinner to increase stimulus duration to 0.3 ms and run a new simulation. Measure the amplitude and time of the peak of the spike.

Condition: stimulus duration = 0.3 ms

Spike peak amplitude (mV) ____________________

Time of peak (ms) ____________________

Also compare the waveform of the spike with the response to the briefer stimulus. Is there an appreciable difference, or are the shapes basically similar?

Changing parameters with the spinner

In the Strength−Duration Simulation window, the middle field of the left panel has several controls for monitoring and changing the duration and strength of the stimulus current.

L clicking on the spinner's up arrow increases stimulus duration to 0.3 ms. The red check mark that appears on the default box indicates that the stimulus duration has been changed from its initial value. Now the depolarizing ramp lasts longer and reaches a more positive potential (Fig. 8). This triggers the spike at an earlier time, so its peak occurs at a shorter latency (compare with Fig. 4).

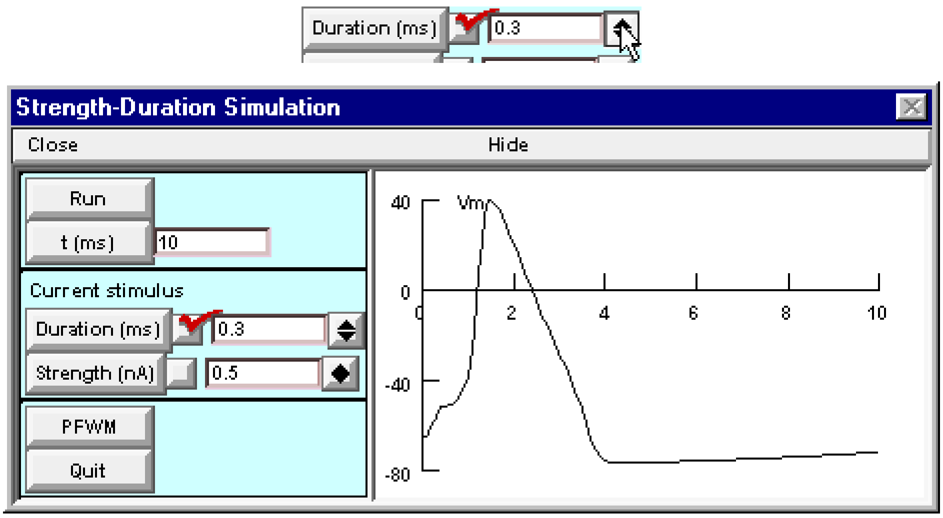


Fig. 8. Using the "spinner."

Toggling between default and new parameter values

A red check over a default box indicates that a parameter has been changed from its initial value (Fig. 10A). L clicking on the default button restores the initial ("default") value (Fig. 10B), and L clicking again returns to the non−default value.

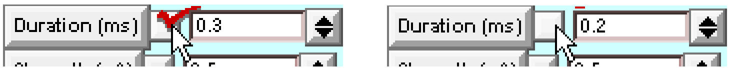


Fig. 9. Click on the default button to switch between default and non-default values.

Is the spike waveform truly resistant to changes in stimulus duration, or is the increase from 0.2 to 0.3 ms too small to have much of an effect? Try increasing the stimulus duration to 1.3 ms and see what happens to spike waveform, timing and amplitude. However, before you run the simulation, make an educated guess at how V_m might respond to the greatly prolonged stimulus current. Then run the simulation with this new stimulus duration and record the spike peak amplitude and time.

To increase duration from 0.3 to 1.3 ms, you could click on the spinner's up arrow 10 times, but it's quicker to change the resolution of the spinner (see below) and then click once.

Condition: stimulus duration = 1.3 ms

Spike peak amplitude (mV) ______________________

Time of peak (ms) ______________________

Did prolonging the stimulus have a larger or smaller effect than you expected? What might account for this observation?

____

____

____

____

Changing a spinner's resolution

L clicking on a spinner can change the value in the adjacent numeric field by adding, subtracting, multiplying, or dividing by a constant. To set the size of this effect, bring up the resolution menu by R clicking on the spinner and holding the mouse button down (Fig. 10A). Drag the cursor along this menu to highlight the desired resolution (Fig. 10B), and then release the mouse button. Now L clicking on the spinner arrow will change the value in the numeric field by the new resolution (Fig 10 C).

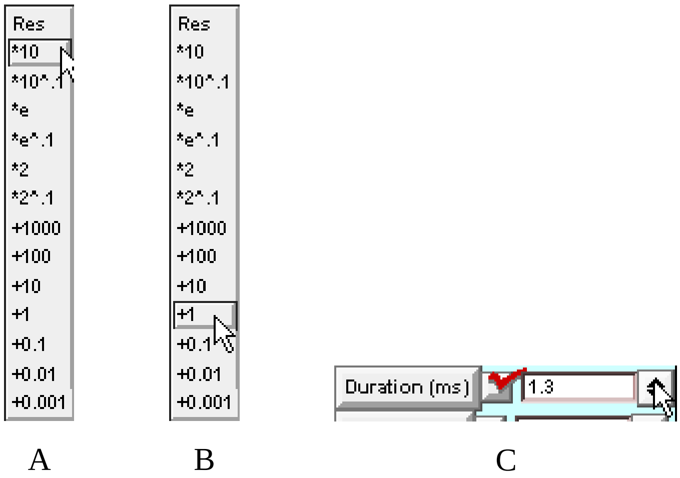


Fig. 10. Changing the resolution of a spinner.

Summarize your observations about the effects of stimulus duration in this table.

	Stimulus duration (ms)		
	0.2	0.3	1.3
Peak Vm (mV)			
Time of peak (ms)			

For these stimulus durations, did the effect on spike waveform seem subtle or gross?

Test your impression by generating a graph that contains several traces so that a side-by-side comparison of waveforms is possible. Label the individual responses with the corresponding stimulus durations. You will also have to move the Vm label from its original location using the graph window's Move Text option, which is described below. Use the Print & File Window Manager (PFWM) to print out your final figure (see the Appendix for instructions). Otherwise, use your computer's system software or some other screen capture utility to get a snapshot (bitmap) of the Strength-Duration Simulation window for later printing.

Keeping more than one trace in a graph

Turning Keep Lines on will preserve any trace that is already in the graph panel. New traces that are generated while Keep Lines is on will also be preserved. Turning Keep Lines off will not affect these traces; they will remain during subsequent runs, but new traces will replace each other. By toggling Keep Lines on and off, you can select just those simulation runs that will become part of your "permanent" record.

Fig. 11. Keep Lines is is controlled by an item in the first panel of the graph menu.

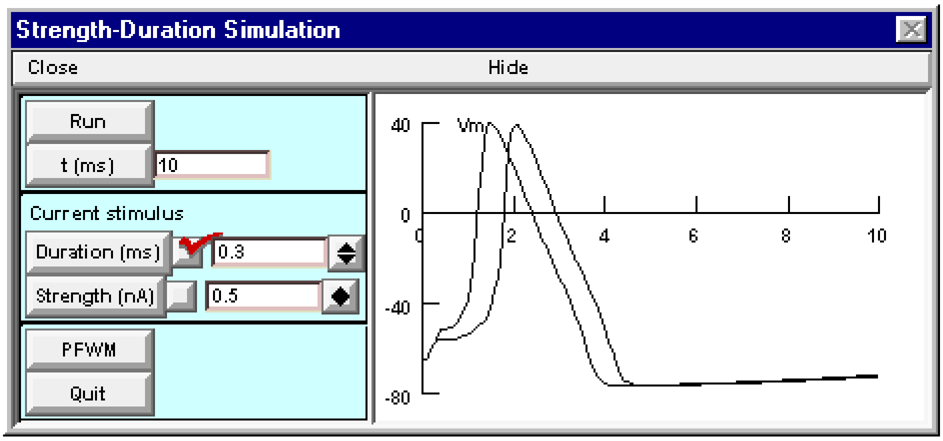
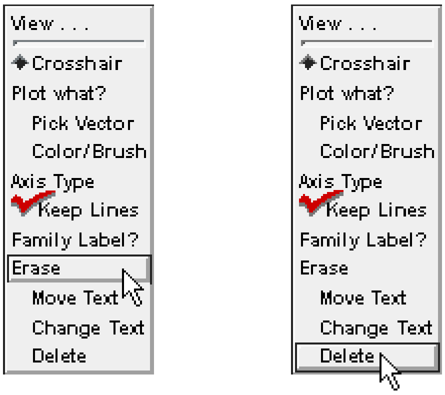


Fig. 12. With Keep Lines on, new simulation runs add new traces to the graph window, and old traces persist. After several runs with different parameters, you can use the cursor to measure from individual traces. If the traces are very close together, it may be helpful to use NewView to focus on the region of interest.

Disposing of unwanted traces

After a series of runs with Keep Lines on, the graph window can become quite cluttered. The quickest way to clear a graph window is just to Erase all traces at once (Fig. 15A). Alternatively, you could enable Delete mode (Fig. 15B), and then eliminate individual traces by clicking on them, one at a time. Although Delete can be more selective, be careful not to delete the name of the variable that is being plotted−−otherwise you will have to exit NEURON and start over again. It is also possible to delete labels created with Change Text (described below), but this is less disastrous because you can always restore labels by using Change Text (a plotted variable can be restored with Plot what? but that is beyond the scope of this exercise). To avoid such accidents, it is always a good idea to return to Crosshair mode after deleting traces.



A B

Fig 13. The Erase and Delete buttons are in the primary graph menu.
If you use Delete, don't forget to restore Crosshair mode!

Adding labels to a graph

To add labels to a graph, select Change Text from the primary graph menu (Fig. 14A). Then L click on a blank area in the graph panel, which pops up a window with a text entry field (Fig. 16B). Type the label you want (Fig. 16C), L click on the Accept button (Fig. 16C), and the label will appear in the graph panel.

MacOS and MS Windows users should note that the mouse cursor (arrow) must be in the text entry field, or else what you type will have no effect. This behavior reflects the origin of NEURON's graphical interface in InterViews.

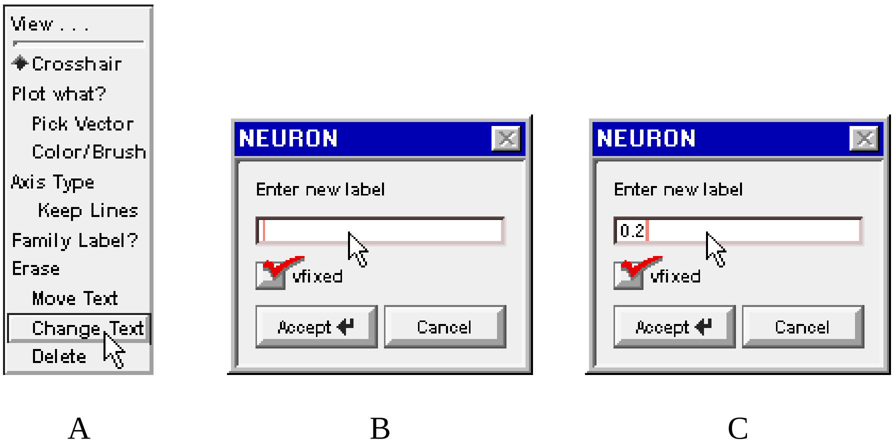


Fig 14. Using Change Text.

Moving a label to a new location

To change the position of a label or variable name on a graph, first select Move Text in the primary graph menu. Then L click on the label to be moved, drag it to its new location, and release the mouse button (Fig. 15).

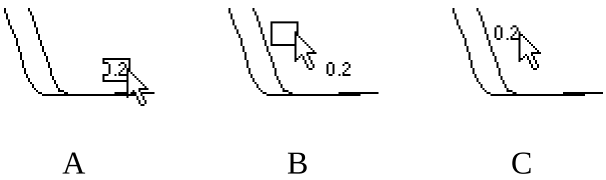


Fig 15. Using Move Text.

3. Find the strength-duration curve for triggering a spike

For each of the following stimulus durations, find the stimulus intensity that is just above spike threshold. Report your results to four significant figures.

Binary search is an efficient way to find the threshold stimulus. Once you find two stimuli that bracket spike threshold, then try a new stimulus halfway between them. For example, if a 0.3 nA stimulus is too low but a 0.4 nA stimulus is too high, try 0.35 nA. After just a few repetitions, you can achieve very fine accuracy.

Stimulus duration (ms)	Threshold stimulus current (nA)
0.125	
0.5	
2.0	
8.0	

Changing a parameter by direct entry into a numeric field

In this example you may have to try many different values for stimulus strength, and you will probably find it most convenient to enter these using the keyboard rather than by repeatedly clicking on the spinner.

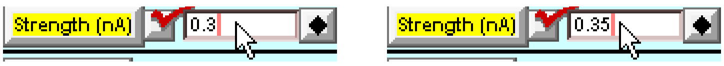


Fig. 16. Entering a parameter value into a numeric field.

If you L click in the numeric field next to the Strength button, a yellow highlight will appear on the Strength button and a vertical line (the field editor cursor) will appear in the field (Fig. 16 left). The cursor can be moved back and forth with the arrow keys on the keyboard, the delete and backspace keys have their usual effects, and you can type numbers into the field (Fig. 16 right). The highlight and cursor will disappear if you click on the Strength button or press the Enter key on the keyboard, but you really don't have to do anything special to make them go away since they will vanish automatically when you run a new simulation.

After you have determined the threshold stimulus current for these four stimulus durations, make a plot of theshold stimulus current in nA vs. stimulus duration in ms. Your plot doesn't have to be a work of art, but it should be legible and accurate. Be sure to label the axes (include units!) and mark data points clearly.

The stimulator delivers a total charge to the cell in picocoulombs (1 pC = 10^{-12} coulomb) that is equal to the product of stimulus duration in ms (milliseconds) and stimulus strength in nA (nanoamps). For very brief stimuli, total charge is a more meaningful indicator of stimulus intensity than either duration or amplitude of the current pulse. Calculate the threshold charge for each of the stimulus durations and enter them in the following table. Then plot threshold charge vs. duration.

Stimulus duration (ms)	Threshold stimulus charge (pC)
0.125	
0.5	
2.0	
8.0	

How does total charge required to reach threshold vary with stimulus duration? What might account for this relationship?

For a given stimulus duration, the threshold potential is V_m at the end of the smallest amplitude current pulse that triggers a spike. What is the threshold potential for a stimulus that lasts 0.125 ms? For a stimulus that lasts 8 ms?

Stimulus duration (ms)	Threshold potential (pC)
0.125	
8.0	

From these observations, do you conclude that prolonged depolarization makes the cell more excitable or less excitable?

What is the experimentally determined total membrane capacitance of the cell, and how does it compare to the actual capacitance of the cell? Measure V_m at the very start and end of a 0.2 ms stimulus of 0.5 nA amplitude (hint: start with a fresh simulation run and use NewView). From the formulas $\Delta q = i \, \Delta t$ and $C = \Delta q / \Delta V$, estimate the total membrane capacitance of the cell. Also calculate the actual membrane capacitance from the surface area of 1000 μm^2 and the specific membrane capacitance of 1 $\mu f/cm^2$. Show the results of these measurements and calculations in the following table.

V_m at end of stimulus (mV)	
V_m at start of stimulus (mV)	
ΔV_m (mV)	
Δq (pC)	
C (nf) = $\Delta q / \Delta V$	
C (nf) from area x 1 uf/cm^2	

Does the experimentally measured capacitance agree with the value calculated from surface area?

Appendix

NEURON's interpreter window

If you type an arithmetic expression after the oc> prompt in the interpreter window, and then press the Return or Enter key, NEURON immediately evaluates the expression and displays the result. As Fig. A1 shows, expressions can assign values to variables, and these variables can then be used in other expressions.

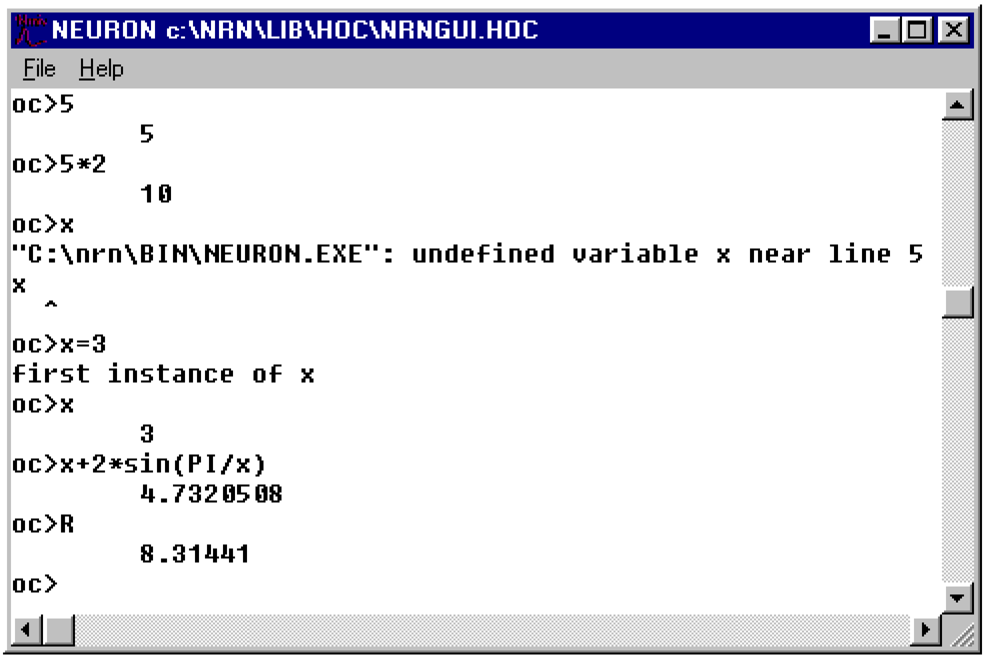
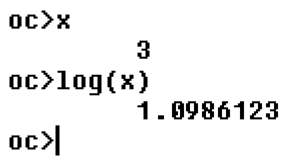


Fig. A1. NEURON's interpreter window

You can also see that NEURON has some built-in constants, including PI and the gas constant R.

The interpreter window has a "history" function that allows you to scroll back and forth through previous expressions by pressing the up and down arrow keys. A recalled expression can be edited and/or re-evaluated. Here's how it works:

x has a value of 3. The interpreter computes the natural log of x, and the oc> prompt indicates that it is ready for the next command.



Pressing the up arrow once recalls the expression that was just evaluated. The vertical line to the right of the closing parenthesis marks the position of the editing cursor. This is the insertion point for new keyboard entries. Pressing the Backspace key would delete the character immediately to the left of this cursor. The keyboard's left and right arrow keys will move the editing cursor back and forth.

```
oc>x
        3
oc>log(x)
        1.0986123
oc>log(x)|
```

Pressing the left arrow key once moves the editing cursor just to the left of the closing parenthesis.

```
oc>x
        3
oc>log(x)
        1.0986123
oc>log(x|)
```

The characters `*2` were typed just before this screen shot was captured.

```
oc>x
        3
oc>log(x)
        1.0986123
oc>log(x*2|)
```

Pressing the Return key makes the interpreter evaluate the new expression.

```
oc>x
        3
oc>log(x)
        1.0986123
oc>log(x*2)
        1.7917595
oc>|
```

The Print & File Window Manager

The Print & File Window Manager (PFWM) has many useful features, but the ones most important to these exercises center on printing hard copy.

Clicking on the button labeled PFWM (see Fig. 4) brings up the Print & File Window Manager. The bottom panel of the PFWM contains two large boxes with red outlines (Fig. A2). The box on the right is a "page composition" or layout area that shows, and allows you to change, where printed windows will appear on the hard copy.

The box on the left is a virtual display of the computer monitor: for each of NEURON's graphical windows, it shows a corresponding blue rectangle. The relative positions and sizes of these rectangles represent the arrangement of NEURON's windows. In this particular example, rectangle 0 is the PFWM itself, and rectangle 1 is the "Strength-Duration Simulation" window.

The toolbar just above the red boxes offers two menu buttons (Print, Session) and three radio buttons (select, move, resize) that help you use the PFWM. The radio buttons set the "mode" of the PFWM, i.e. they determine what happens when you click on the rectangles in the virtual display and page layout area. When the PFWM first comes up, it is in select mode and the radio button next to the word "select" is highlighted.

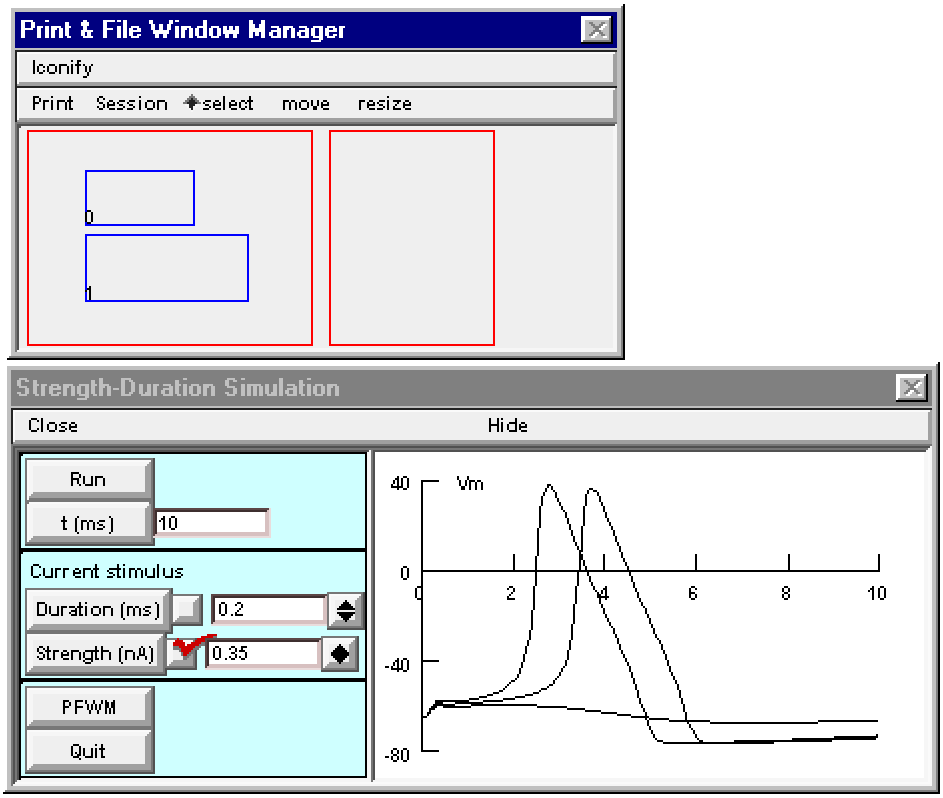


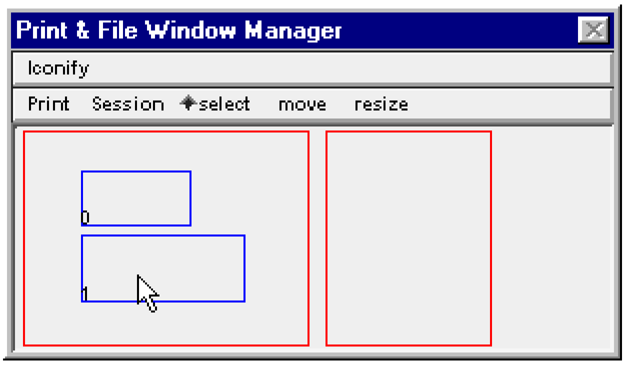
Fig. A2. The Print & File Window Manager (top).

How to choose a window to be printed

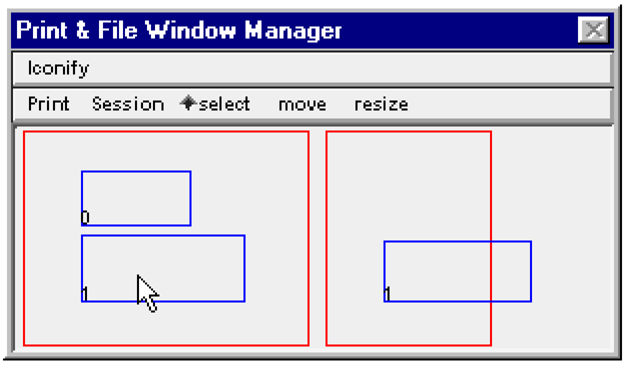
First make sure the PFWM is in select mode. If it isn't, click on "select" in the toolbar.

Decide which blue rectangle in the virtual display corresponds to the window you want to print. If you're not sure, drag the window on your screen to a new location, and notice which of the blue rectangles in the virtual screen moves.

Place the mouse cursor inside the desired blue rectangle in the virtual display . . .



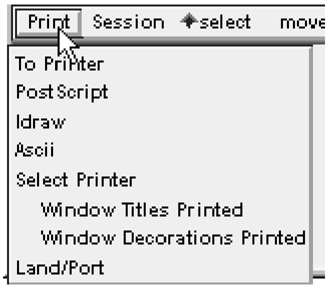
. . . and press the L mouse button. An outline that represents the selected window will appear in the layout area to the right of the virtual screen. More than one window can be selected for printing on a single page, and they can all be resized and positioned as desired (see below).



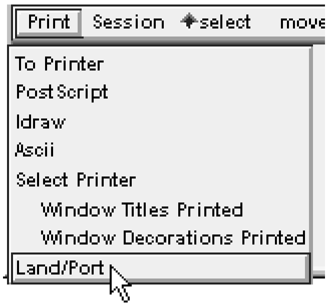
Page orientation

Right now the printout would have portrait orientation.

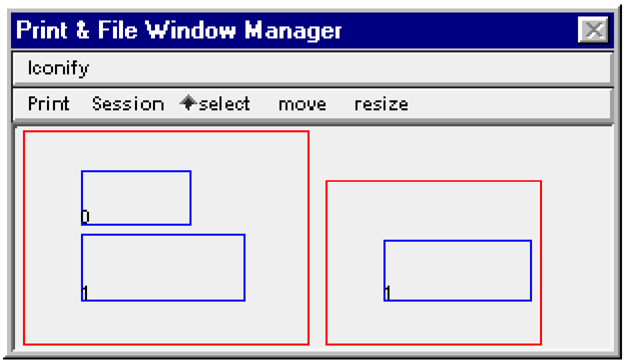
To change to landscape, click on the Print button. This pops up a menu with several options.



Select Land/Port . . .

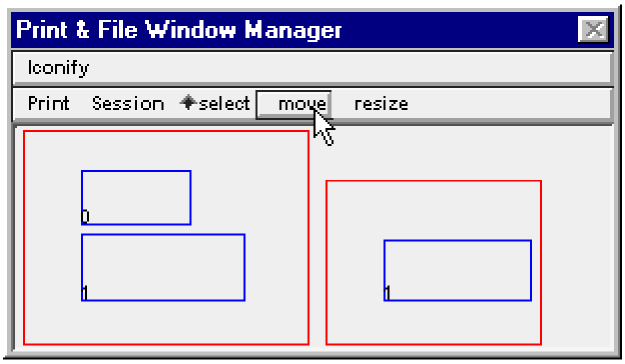


. . . and the page layout toggles to landscape. You can switch back and forth between landscape and portrait orientation by clicking on Land/Port.

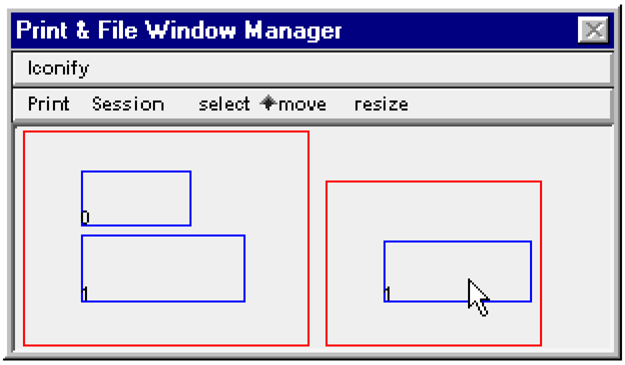


Positioning a window in the layout area

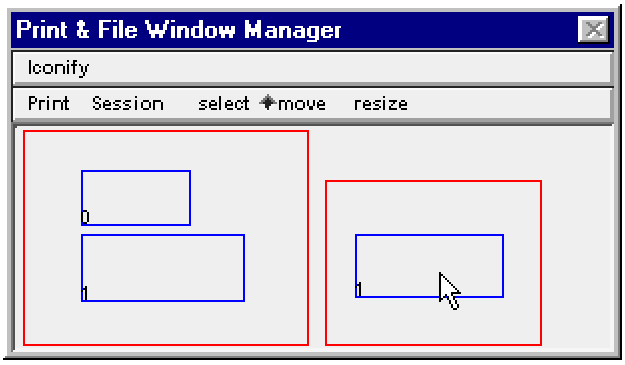
Click on "move" in the toolbar to put the PFWM in move mode.



Then L click inside the window you want to move . . .

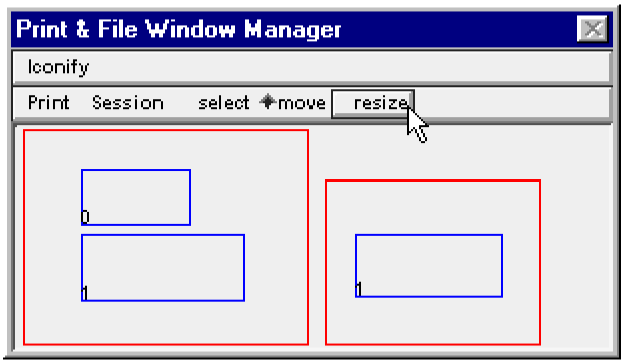


. . . and drag it to the desired location.

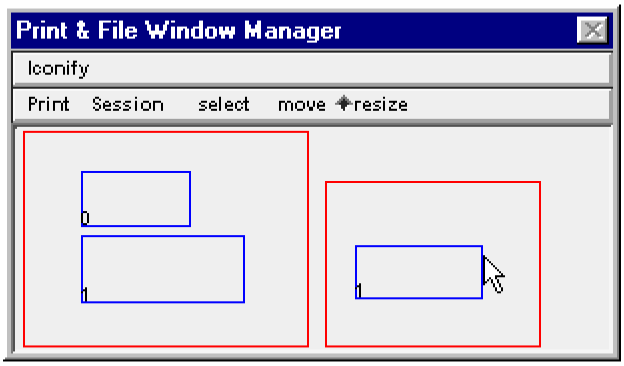


Resizing a window in the layout area

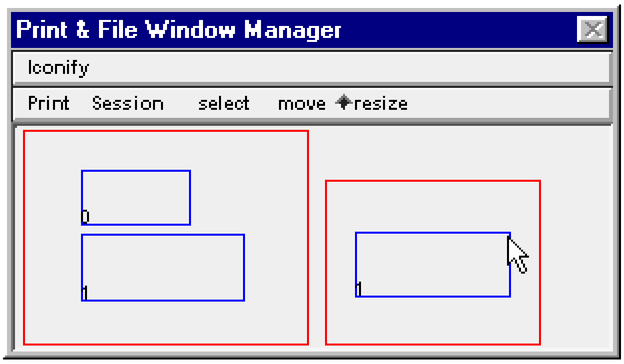
Click on "resize" in the toolbar to put the PFWM in resize mode.



Then L click inside the window you want to resize . . .



. . . and drag the outline to the desired size. Note that the aspect ratio is preserved.

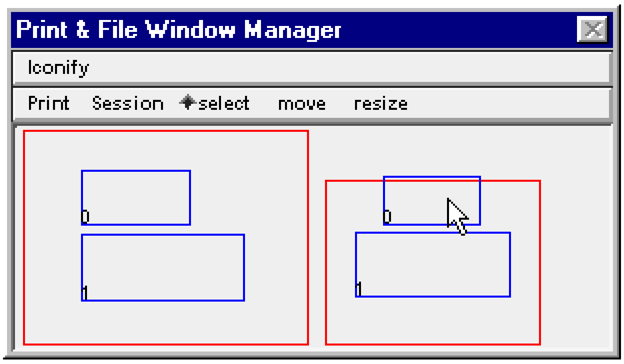


When resized windows are printed to a PostScript file or a PostScript printer, text elements will be scaled proportionally. Under MSWindows, NEURON can also print directly to a non-PostScript printer (i.e. without first passing through a rasterizer such as ghostscript), but if you do this you will find that font size remains unchanged regardless of how much each window was enlarged or reduced.

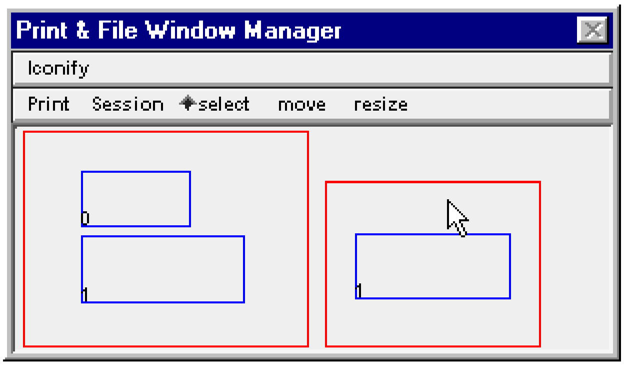
Deleting a window from the layout area

What if you don't really want to print all of the windows you selected? In this example, the undesired window is the PFWM.

To get rid of it, first make sure that "select" is on. Then put the mouse cursor inside its outline in the layout area . . .



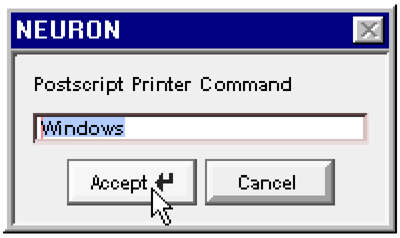
. . . and then press the L mouse button.



Printing the contents of the layout area

Click on the Print button and select the To Printer item from the popup menu.

This brings up a window that gives you the option of changing where the printed output will be sent. Most MSWindows users should just accept the default (Windows), which will send the printout to the local printer; a different choice may be needed in a networked environment. For Unix or Linux the proper choice would generally be `lpr` (with optional command line switch).



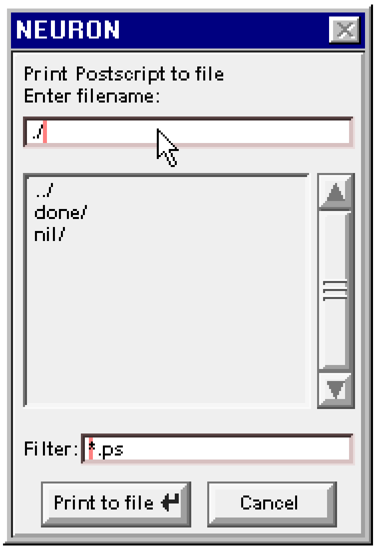
Then click on the Accept button.

Creating a PostScript output file

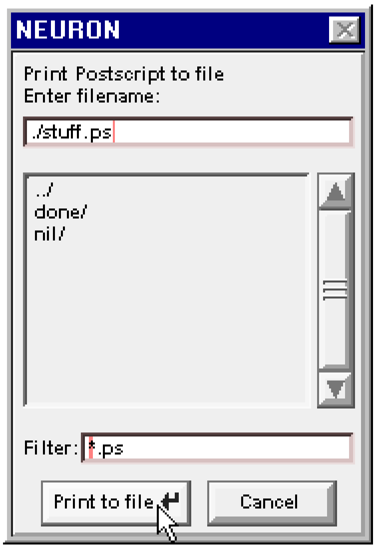
You may want to save the printout in a PostScript file for later printing or conversion to some other graphics format.

Click on the Print button and select the Postscript item from the popup menu.

This brings up a file chooser window in which you can specify the name of the file that will be generated.



Type in the name of the file you want to create. Then press the Print to file button, and you're done.



Creating an ASCII output file

No, this isn't yet another graphics file format. It's just a plain text file that contains a printout of the (x, y) coordinate pairs from the most recently created trace in the selected graph window. This can be a convenient way to export numeric results from NEURON to a file that can be read by other programs.

Creating an ASCII file is very similar to creating a PostScript file. First click on the Print button and select the ASCII item from the popup menu. A file chooser window will appear in which you can type the name of the file that will receive the numeric values in ASCII format.

The output file will contain a brief header that tells the name of the y variable and how many lines of data there are. Next comes a series of lines in which the x and y coordinates of each point are separated by tabs.

```
Graph addvar/addexpr lines
1 201
x Vm
0       -65
0.05    -64.9985
0.1     -64.9971
 .          .
 .          .
 .          .
9.95    -73.841
10      -73.7917
```

Index

Contents