brains, minds media
ISSN: 1861-1680

Virtual Electrophysiology with SPatch

Applications in undergraduate education

Robert C Cannon[a*] and Constance Hammond[b]

[a] Textensor Limited, 37 McDonald Road, Edinburgh, United Kingdom
[b] Institut de Neurobiologie de la Méditerranée, 163 route de Luminy, BP13, 13273 Marseille Cedex 09, France

*Corresponding Author: Robert Cannon, Address: as above, Phone and Fax: +44 131 466 8356, email: robert.c.cannon@gmail.com

urn:nbn:de:0009-3-14036

Abstract. SPatch is an open source virtual laboratory designed to perform simulated electrophysiological experiments without the technical difficulties inherent to laboratory work. It provides the core equipment necessary for recording neuronal activity and allows the user to install the equipment, design their own protocols, prepare solutions to bathe the preparation or to fill the electrodes, and gather data. Assistance is provided for most steps with predefined components that are appropriate to a range of standard procedures. Experiments that can be performed with SPatch at present concern the study of voltage-gated channels in isolated neurons. This allows understanding the ionic mechanisms of Na^+ and Ca^{2+} action potentials, after spike hyperpolarization, pacemaker tonic or bursting activity of neurons, delayed or sustained or adaptive firing of neurons in response to a depolarization, spontaneous depolarization of the membrane following an hyperpolarization, etc. In an educational context, the main interest of SPatch is to allow students to focus on the concepts and thought processes of electrophysiological investigation without the high equipment costs and extensive training required to perform laboratory work. It can be used to acquaint students with the relevant procedures before starting work in a real lab, or to give students an understanding of single neuron behavior and the ways it can be studied without requiring practical work. We illustrate the function and use of SPatch, explore educational issues arising from the inevitable differences between simulated and real laboratory work, and outline possible improvements.

Keywords: Virtual laboratory, electrophysiology, ionic channels, neuron, simulation, modeling.

Citation: Cannon RC & Hammond C (2008). Virtual Electrophysiology with SPatch. Applications in undergraduate education. Brains, Minds & Media, Vol.3, bmm1403, *in: Lorenz S, Egelhaaf M (eds): Interactive Educational Media for the Neural and Cognitive Sciences,* Brains, Minds & Media, 2008.

Published: May 19th, 2008

Introduction

Teaching electrophysiology from real experiments is severely hampered by the need for extensive equipment and technical skills required for performing successful experiments. A group of 20 students, for example, requires at least 6-7 electrophysiological set ups with corresponding biological preparations, well made recording electrodes, special ionic solutions to bathe the preparation or fill the recording electrode, and above all, many hours of training before students can correctly perform electrophysiological recordings. In particular, this involves delicate manual manipulations to attach the electrode to the cell membrane for patch recording or to penetrate the cell for intracellular recording.

To address this problem, we designed a virtual electrophysiology lab, SPatch[1], that can be used in an educational context to replace electrophysiological set ups, neuronal in vitro preparations and manual skills. It contains the minimal necessary equipment to perform electrophysiological recordings: electronic equipment (amplifier, computer, software allowing stimulation and recording, analog-digital converter), a choice of ingredients to prepare intracellular and bath solutions, a choice of recording electrodes and the neuronal preparation (a culture of neurons). The software performs the recording of the activity of a single neuron in the configuration chosen by the student and displays the results obtained on a separate screen.

As well as avoiding technical problems, it enables students to learn how to conduct an electrophysiological experiment without time constraints. They can observe, ask questions, formulate hypotheses, test them, analyze the results and revise their conclusions, as many times as they want. For example, at a very basic level, once action potentials have been observed in recordings from excitable cells, the following questions may arise: do action potentials result from currents through the membrane, which type of ions carry these currents, via which type of ionic channels? To answer these questions students can design a stimulation protocol and analyze the results generated by SPatch. For all these steps students could be guided by their teacher but the advantage of a virtual lab is that they become active participants who control the process rather than passive observers, as would be the case if they were looking at animated pre-generated results or watching an instructor perform the experiment.

The next section describes the software itself, beginning with the underlying computational model of the system and then the user interface through which the student interacts with it. We then present an example lesson and conclude with a discussion of the main differences between learning from a virtual lab as compared with real experiments and suggest how the process may be improved.

Virtual Patch Simulation

The two main concerns in developing SPatch were first that it should provide a virtual system that behaves as much like a real system as possible within well defined and easily understood boundaries, and second that anything going on in this system should be accessible to the user. This leads to a design comprising three main components: an implementation of the underlying mathematical model; a general purpose user interface for operating on the model; and a set of dedicated views that provide a more laboratory-orientated way to interact with the model.

Model structure

The main subjects of study with SPatch are voltage-gated and ligand-gated ion channels. These are represented within the system by kinetic schemes (also known as Markov models) (Strassberg and Defelice 1993) that define the possible states of each channel and the rates of transitions between them. These are a superset of the standard Hodgkin-Huxley model (Hodgkin and Huxley 1952) that are easily extended to include effects such as drug interactions or state-dependent blockers. The model also requires a representation of the environments in which the channels are located and the equipment used to study them. The former is restricted to preparations containing separate compact cells (no dendrites or axons) in a bath where the solution can be changed. The latter comprises the patch pipette, head stage and a computer controlled stimulation and recording system. The

[1] http://www.inmednet.com/education/spatch

connection between a pipette and a cell may be in one of four standard patch clamp (Sakmann and Neher 1995, Hammond 2008) configurations:

- Whole-cell patch in which the pipette is open to the cell and the pipette solution rapidly replaces the cytoplasm because of its much greater volume
- Cell-attached in which the membrane patch separating the two is still intact
- Inside-out in which the patch has been detached from the cell. The rest of the cell can be ignored and only the currents flowing through the channels in the patch are computed. The inside of the membrane is now exposed to the bath solution.
- Outside-out in which the contents of the cell have been taken up into the pipette so that the patch of membrane covering the end is from the far side of the cell such that the inside of the membrane is exposed to the solution in the pipette and the outside is exposed to the bath as before.

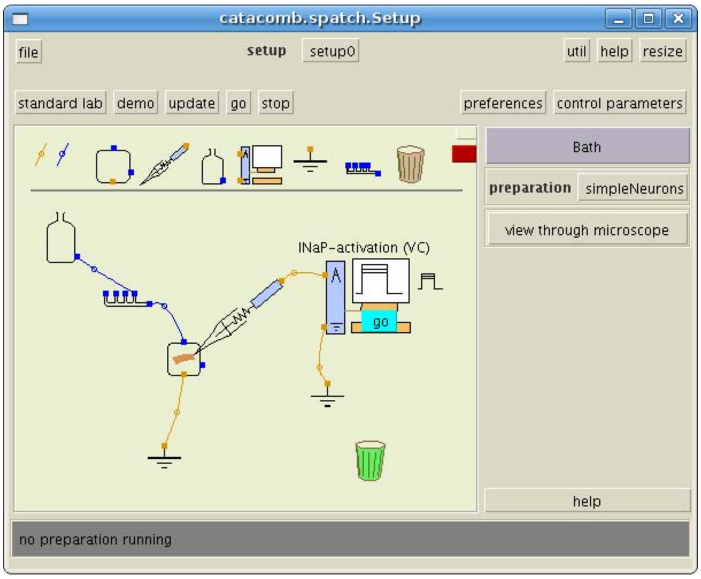
One further configuration is also supported where the pipette is in the bath but not attached to any membrane. In this case the recording system will just register the junction potential between the pipette and the bath. These configurations define the main part of the calculation needed for the model in terms of the ion channels to be included and what solutions they are exposed to on each side. The electrical circuit also comprises one or more patch pipettes and their capacitance to the bath. For each component, the quantities of interest are the potential, any currents that flow and the ions in the solution that it contains or that surrounds it. The ions and molecules in solutions interact with other components through binding reactions to sites on the inside or outside of ion channels, and through electrical potentials created by differences in ionic concentration or mobility.

This model domain is implemented in software components that define the allowed structure, accept parameter values for specific instances and compute their own simulated behavior. The structures are relatively general in the sense that they do not hard-code any particular ions, molecules, ion channel types or cell types. All these quantities can be set from scratch by the user, although in most cases they would be loaded from a library of pre-defined components. The calculations use software adopted from the open source Catacomb simulation system (Cannon et al. 2003).

One difficulty arising from the flexibility of the model specifications is that the user may have access to too many quantities. The same interface as is used to perform virtual experiments can also be used to define new types of ion channels, or even new types of ion. It risks confusing low level modeling tasks, as undertaken by an advanced instructor, with simulated laboratory tasks, as undertaken by the student. To help address this problem, each parameter has an associated "access level" that locates it on a scale between zero and ten according to the stage at which it should be exposed to the user. Only those parameters below a user-selected skill level are shown in the user interface.

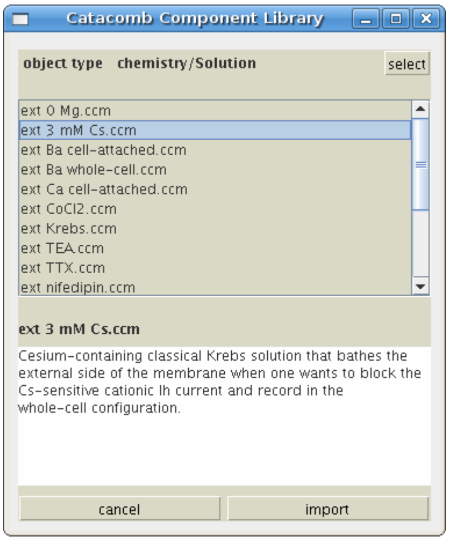
User interface

The main window of the user interface is shown in figure 1. It shows the equipment that is available to the student in the top bar from which items can be dragged onto the main display. Alternatively, a starndard configuration can be pre-loaded as in the figure. Clicking on a component gives access to its properties via the right hand panel. Depending on the the properties to be set this may provide menu selections, sliders or buttons to open new windows.



no preparation running

Figure 1: the main virtual lab window. The lab can be assembled by dragging components from the top bar, or initialized to the default configuration as shown. This image shows the bath with a cellpreparation in it in the model. Pipette and headstage come in from the right and are attached to a schematic computer. The bottle on the left connects via a set of valves to the bath. Clicking on an item in the diagram gives access to its properties in the panel on the right.



object type chemistry/Solution

ext 0 Mg.ccm
ext 3 mM Cs.ccm
ext Ba cell-attached.ccm
ext Ba whole-cell.ccm
ext Ca cell-attached.ccm
ext CoCl2.ccm
ext Krebs.ccm
ext TEA.ccm
ext TTX.ccm
ext nifedipin.ccm

ext 3 mM Cs.ccm

Cesium-containing classical Krebs solution that bathes the external side of the membrane when one wants to block the Cs-sensitive cationic Ih current and record in the whole-cell configuration.

Figure 2: loading a component of a model from the library. The components that are available have been previously saved from SPatch. Additional descriptive text describes what the component does or how it is intended to be used. In this case, the solutions section of the library is displayed. The names are prefixed with 'ex' and 'int' to indicate whether they are intended for external or internal application.

For the novice user, the main starting point is the library (figure 2) from which predefined components can be loaded. Each item in the library is simply a component constructed and saved with SPatch, so an experienced user can set up their own libraries for new experiments. Having selected a preparation from the menu, the patch is made manipulating a pipette on a microscope view of the cells as in figures 3a and 3b.

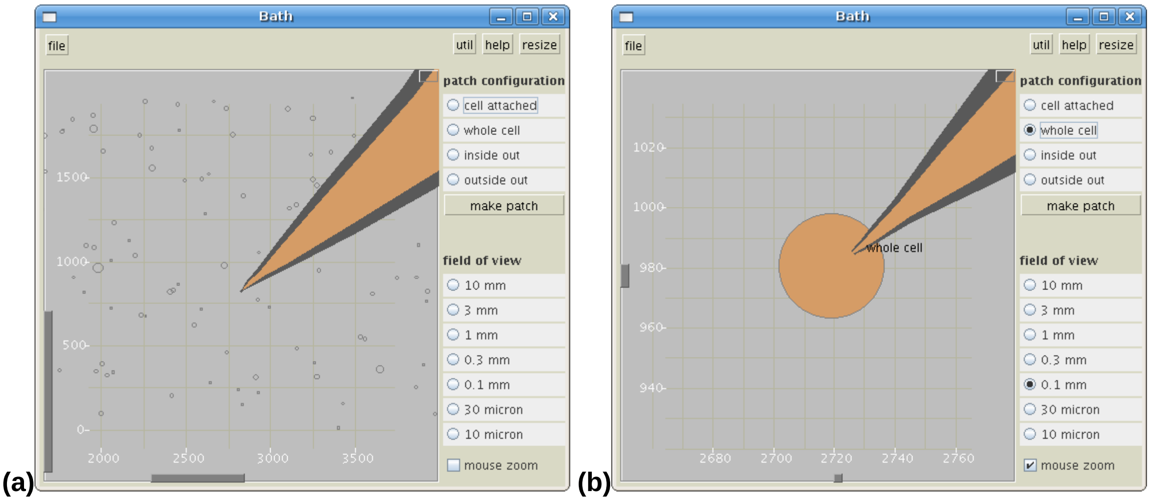
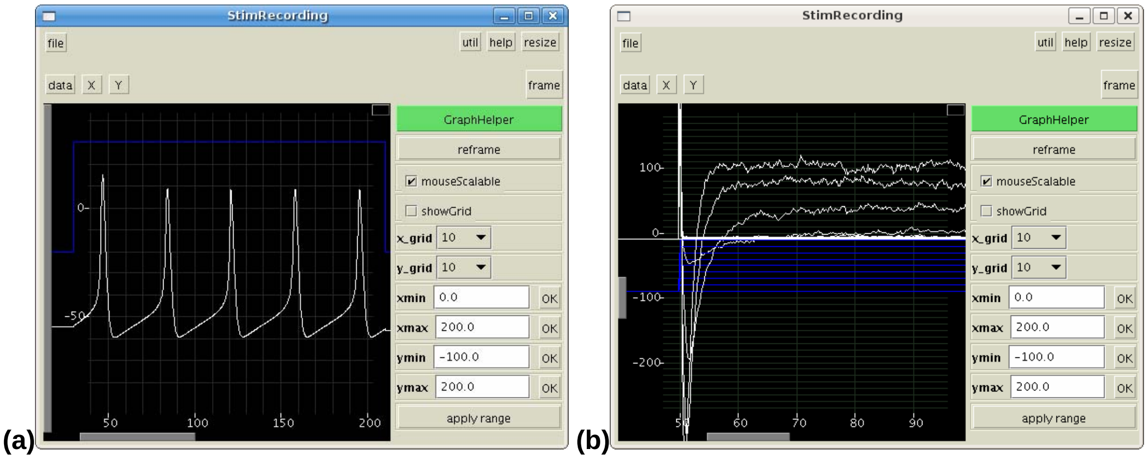


Figure 3: views through the microscope as a cell is patched. As with most graphics displays in SPatch, the scales can be manipulated by zooming and panning with the mouse, or via explicitly setting the ranges with the context menu in the top right of the display. The field of cells in the left image represents a plate of cultured cells. The student can move the pipette around and make a patch with the buttons on the right (a). Image (b) shows an enlarged view of the patched cell. In this case it is a whole cell patch and the color coded solution in the pipette is shown to have filled the cell.

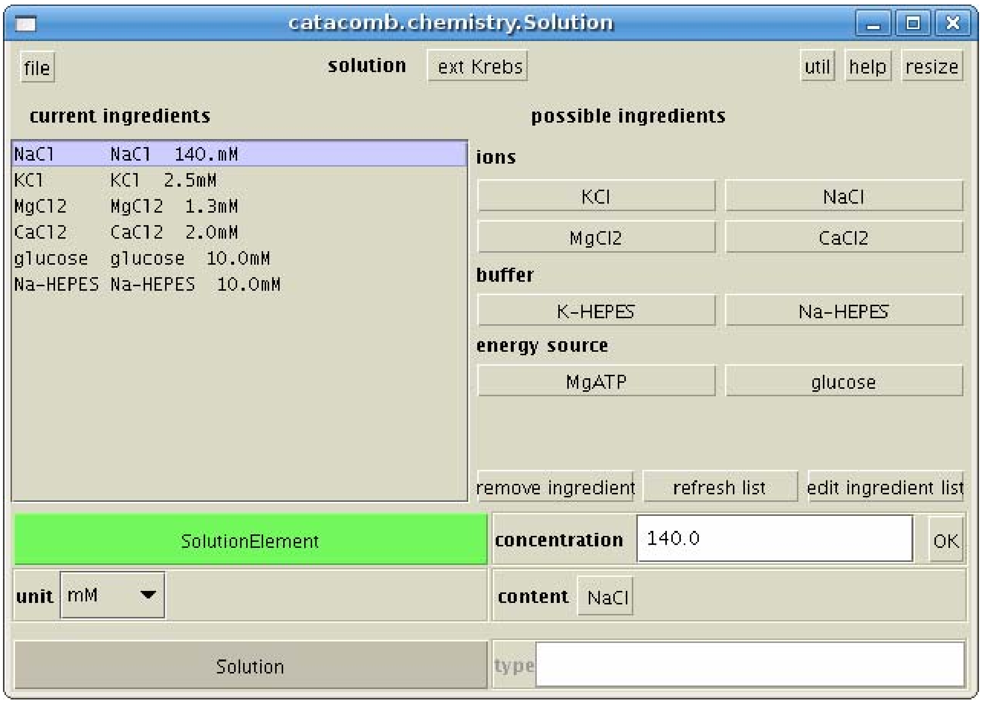


x_grid 10
y_grid 10
xmin 0.0
xmax 200.0
ymin -100.0
ymax 200.0

Figure 4: sample display of results for (a) current and (b) voltage clamp experiments. The noise in figure b comes from the small number of channels in the preparation which are simulated stochastically.

The results are shown in a separate window as in figure 4 where they can be examined in place including options to zoom in and to scroll the displayed portion of the graph on either axis. The results can also be exported for off-line analysis with external tools.

For more advanced users there are facilities for defining ones own solutions (figure 5) and command protocols (figure 6). Indeed, all components of the model are accessible through the user interface if required.

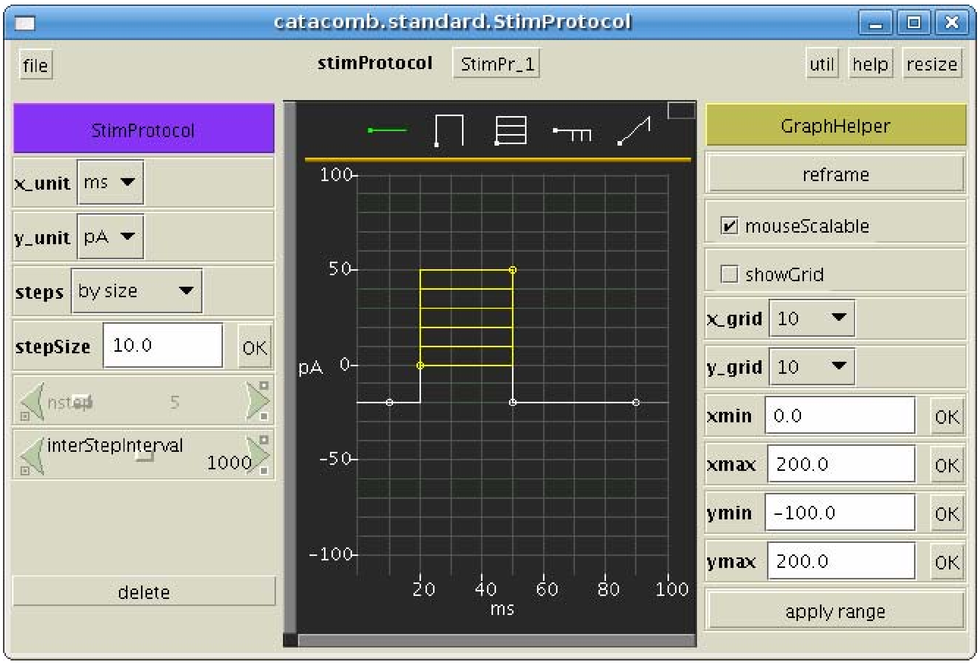


solution ext Krebs

```
NaCl      NaCl  140.mM
KCl       KCl  2.5mM
MgCl2     MgCl2  1.3mM
CaCl2     CaCl2  2.0mM
glucose   glucose  10.0mM
Na-HEPES  Na-HEPES  10.0mM
```

concentration 140.0
unit mM
content NaCl

Figure 5: Editing solutions. A categorized list of the possible ingredients is shown on the right. These are loaded from the library or can be created by the instructor using a similar interface to define the ions present in each type of compound. Clicking on an ingredient allows its concentration to be set with the controls at the bottom. It is also possible to categorize or tag the solutions to help students learn how they are used. In a similar way, the ingredients (also edited via this style of interface) have been tagged here as either ions, buffers or energy sources.


catacomb.standard.StimProtocol

file | stimProtocol StimPr_1 | util | help | resize

StimProtocol

x_unit ms

y_unit pA

steps by size

stepSize 10.0 OK

nstep 5

interStepInterval 1000

delete

GraphHelper

reframe

☑ mouseScalable

□ showGrid

x_grid 10

y_grid 10

xmin 0.0 OK

xmax 200.0 OK

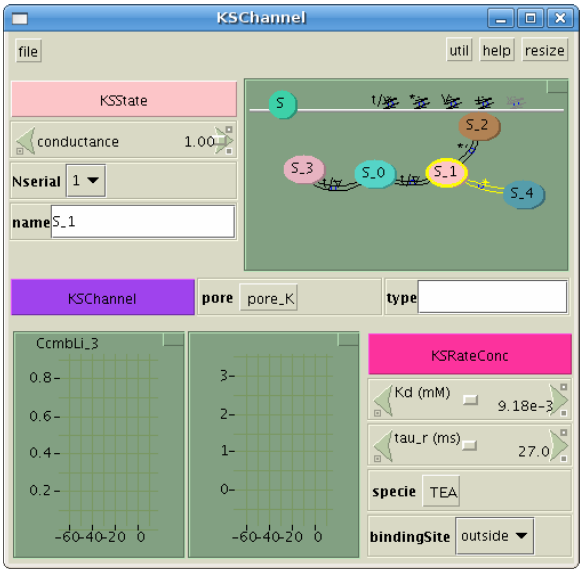
ymin -100.0 OK

ymax 200.0 OK

apply range

Figure 6: Command editing. Like the main lab window, a selection of predefined shapes can be dragged off the top bar and added to the command. Further settings are available on the left for command itself, and on the right to control the display. In particular, the student must decide whether the command controls the current or voltage, and the number of steps if any.

A final example is shown in figure 7 which is more likely to be of interest to model developers than to students, but is included to demonstrate how advanced students can explore the model in whatever depth they choose. It shows one of the ion channels used in the standard library examples. As with other components the model can be constructed via a drag-and-drop interface, in this case involving drawing a state diagram with the components at the top and setting parameter values via the sliders.


KSChannel

file | util | help | resize

KSState

conductance 1.00

Nserial 1

name S_1

KSChannel | pore pore_K | type

KSRateConc

Kd (mM) 9.18e-3

tau_r (ms) 27.0

specie TEA

bindingSite outside

Figure 7: Channel editing. The underlying models that are used in SPatch are accessible through the same type of interface as the lab itself. This image shows the channel editing window with the kinetic scheme governing the channel transitions at the top. The highlighted transition is shown in the lower panels. In this case it is a binding reaction for TEA on the outside of the channel so the two voltage dependence panels are empty but the dissocaition constant, Kd, and off rate tau_r, are shown on the right.

Design and Implementation

SPatch is written in Java, version 1.1 and distributed as an executable jar file that runs on Windows, Linux and Mac OSX. It requires the Java virtual machine (JVM) version 1.4 to be installed in order to run it, but thereafter requires no specific installation and can be run from user space without requiring admin privileges on the destination machine. Ease of installation is increasingly important with the tight restrictions imposed on many machines in computer laboratories that require native programs to be officially authorized as safe and secure before they can be installed. Typically, the JVM will already be installed or will be on a list of pre-authorised applications. Since SPatch uses only facilities provided within the restricted JVM environment no further authorization is required.

The design and implementation are driven by the model structure outlined above. Each class of physical entity allowed in the system (ions, solutions, ion channels, cells, pipettes etc) is represented by a single Java class. Each type of thing corresponding to that class becomes an instance of the java object (eg sodium, delayed rectifier potassioum channel, or a particular solution) that is either read from a library or created by the user. Physically there are also individual ions channels and cells. In the simulation, these only occur implicitly, and as needed within the solution of the system to compute the observed behavior.

Thus the model becomes a collection of objects holding parameters with extensive links between them. They implement a notification system such that any change bubbles up the tree and along the connections so that any component that may depend on the changed quantity is notified. The main destination of these notifications are the user interface components that represent the state of components and allow the user to interact with them. For every object type, a generic user interface can be constructed by reflection (in which the software examines the object and constructs a standard table for the parameters and structures it contains) but for some objects these are overridden with custom interfaces that provide more intuitive access to the object. When any parameter is changed, the notification eventually reaches the top of the tree. Depending on the user settings, this could cause the model to be rerun and new results displayed, or the currently running model to be stopped and restarted with the new parameters. The latter simulates a scenario where the experimenter sets up a repeated stimulation and gradually changes another quantity while watching the response.

Example applications

The aim of electrophysiological experiments performed with SPatch is to record the current through ionic channels present in the membrane. As these are cultured neurons, they need to be bathed with an extracellular solution, specified by the experimenter, that mimics the fluid present outside cells in-vivo. The experimenter also controls the pipette solution that may replace the intracellular fluid, depending on the patch configuration. With the current version, users can record the current through ionic channels present in a small patch of membrane, or the current through the whole membrane of the recorded neuron. The available channels include simple Na^+, K^+ and Ca^{2+} channels involved in generating action potentials, A-type and M-type K^+ channels and H-type cationic channels. To study single channel currents (unitary currents) the electrode or pipette is attached to the membrane (cell-attached patch) or to a piece of membrane excised from the neuron. To study all the channels present in the membrane the electrode is attached to the membrane of the neuron and a little suction is applied to the interior of the pipette. This causes the rupture of the membrane patch under the pipette. Consequently, the patch pipette now records the activity of the whole cell membrane (minus the small ruptured patch of membrane). Rapidly, the intracellular solution equilibrates with that of the pipette, the volume of the latter being many times larger.

The following two study scenarios illustrate how a student can be guided in using SPatch to investigate some basic properties of neurons.

1. After having observed Na^+ action potentials, identify the underlying ionic currents and their properties

1.1. Observation of Na^+ action potentials and of the effect of drugs that block K^+ and Na^+ voltage-dependent channels

Since the Na^+ action potential involves a depolarization of the whole membrane, it is best recorded in current clamp mode (recording of potential change), in the whole cell configuration in response to a depolarizing current step (since action potentials are triggered around -45 mV). To set up the model in SPatch, first install the standard lab, then add three more bottles and connect them to the junction with tubes. Fill one bottle with "ext Krebs" an extracellular solution that mimics the normal extracellular fluid. Add the preparation (simple neurons), fill the recording pipette with "int Kglu", an intracellular solution that mimics the normal intracellular fluid, (the pre-defined solutions are labelled "ext" or "int" to indicate whether they are normally intended for use externally in the bath, or internally, in the pipette) and design a stimulation protocol (StimProtocol) consisting of a step of current of 20-50 pA and 200 ms duration (recording in current clamp mode). Now, on the microscope view move the pipette over a cell and make a whole cell patch. Record and observe the action potential(s). Design a new stimulation protocol with a constant hyperpolarization of the membrane (-40 pA) and the same step. Observe and make hypotheses.

Next, fill one empty bottle with "ext TEA" (an extracellular solution that contains tetraethylammonium ions blockers of K^+ channels of the Na^+ action potentials) and perfuse the preparation. Test the two recording protocols as above, observe and conclude. Fill the last bottle with the TTX containing solution, "ext TTX" (TTX blocks voltage dependent Na^+ channels of the action potential) and perfuse the preparation. Test the two stimulation protocols, observe and conclude. Wash in "ext Krebs" and observe at different times after the end of "ext TTX" perfusion.

1.2. The above results suggest that Na^+ and K^+ currents underlie Na^+ action potentials. Test this hypothesis.

Study the TEA-insensitive current in isolation: perfuse the preparation with the "ext TEA" solution, fill the pipette with "int KGlu", record in whole cell configuration (to record the whole cell current), and apply a series of depolarizing voltage steps from a holding potential VH = -80 mV to +50 mV (recording in voltage clamp mode). Do the same from VH = -50 mV. Observe the time course of the current. Do the same when the preparation is perfused with a solution containing TEA and TTX ("ext TEA TTX"). Conclude.

Study the TTX-insensitive current in isolation: perfuse the preparation with "ext TTX", fill the pipette with "int KGlu", record in whole cell configuration (to record the whole cell current), and apply a series of depolarizing voltage steps from a holding potential VH = -80 mV to +10 mV (recording in voltage clamp mode). Do the same from VH = -50 mV. Observe the time course of the current. Do the same when the preparation is perfused with a solution containing TEA and TTX. Conclude.

2. Which type of current underlies the depolarizing sag observed during hyperpolarizing steps applied to the neuronal membrane?

2.1. Observation of the depolarizing sag

Install a standard lab, add four more bottles, connect them to the junction with tubes and open the four channels of the junction. Fill one with "ext Krebs". Fill the recording pipette with "int KGlu". Choose the preparation "simple-Purkinje" from the library (Purkinje cells are the principal neurons of the cerebellum). Since a change of voltage is to be recorded, the stimulation must be applied as current steps (in pA) and the recording must be in current clamp mode. Design a series of long duration (2s) hyperpolarizing steps from VH = -60 mV. Observe the depolarizing sag during the step and its amplitude as a function of the amplitude of the hyperpolarizing current step and of the membrane potential imposed before the step. In response to negative current step the membrane hyperpolarizes and then slowly repolarizes. The slow repolarization is known as depolarizing sag. Identify the voltage conditions for this sag.

2.2. The ionic mechanisms underlying the depolarizing sag

Once you have obtained a stimulation profile that gives depolarizing sag, perfuse the preparation with extracellular solutions containing blockers of Na^+, K^+ or $Ca2^+$ channels. Do any of them block the depolarizing sag? If not, try the same experiment in "ext 3 mM Cs", an extracellular solution that contains caesium ions, known to block some K^+ and cationic channels. If this blocks the sag then the experiment suggests that the underlying current is carried by K^+ ions and possibly also by other cations such as Na^+ ions (for example the H-type current called H for hyperpolarization-activated is carried by Na^+ and K^+ ions and is sensitive to Cs^+ ions).

To identify the type of ions carrying the current of the depolarizing sag, build a current - voltage relationship (I-V curve): switch to voltage clamp mode and design a stimulation protocol that applies a series of voltage steps from hyperpolarizing to depolarizing. Block voltage dependent Na^+, K^+ and Ca^{2+} currents by perfusing an extracellular solution containing the appropriate blockers and identify the reversion potential of the Cs-sensitive current.

2.3. Functional role of the depolarizing sag

Try to imagine the role of this current in physiological situations. Remember that the activation of synaptic receptors by GABA such as GABAB receptors evokes a transient hyperpolarization of the membrane. Mimic this situation in current clamp mode for different values of resting membrane potential.

Summary

At present, SPatch can replace certain laboratory experiments but does not replace a teacher. For use with high school or university students SPatch must be explained by teachers well trained in neurophysiology, who should be present with students at least initially to explain the process of choosing solutions, designing stimulation and recording schemes, operating the virtual lab, and analysing the results. With the development of further instructional material it should become more acessible for self-guided use. To date it has been used mainly in France for high school students (particularly at the lycée Vauvenargues in Aix en Provence) and will be used next year at the Université de la Méditerranée, in Marseille-Luminy for graduate students in neuroscience.

The principal benefit of the virtual laboratory approach over text book illustrations or computer animations is that it incorporates sophisticated models of the systems being studied which allow the student to pursue a wide range of investigations that go beyond the ones envisaged by the model developers. This raises the question of how far such investigations can be extended and where the simulation most noticeably diverges from a real system.

Apart from the inevitable differences due to the limitations of the domain being modeled and the representation of its contents, the greatest difference between models simulated with SPatch and real laboratory experiments is with the passage of time between eperiments. In SPatch a model is a static tree of objects and parameters that can be run to compute and display its behavior. As with much electrophysiology a typical run is rather short (a few seconds of real time), but whereas the real system carries on behaving and changing when the experimenter is not looking, nothing happens to the SPatch model. Moreover, it can be easily reset to an earlier stage, thereby, for example, effortlessly undoing a mistake that would have rendered a real preparation useless. By construction, it also imposes very few constraints on the sequence of operations used to manipulate a model. In the simulation it is as easy to change the density of an ion channel after the system has been set up as it is to change the stimulation potential, whereas in a laboratory context obtaining cells expressing different ion channels can be many weeks or months of work.

From this perspective, SPatch provides a virtual world that is actually too accessible and too easy to manipulate. For a modeling system, this is clearly desirable, but as a laboratory simulation it could benefit from several further layers of structure to represent the constraints imposed by the environment that surrounds the components that are implemented in the model. A first step would be to keep models running in real time, whatever the student is doing. Further steps might also include a measure of costs in terms of time and materials, or rewards for corect final analysed results.

At present, however, the main limiting factor for the use of SPatch is the availability of well-characterised models for students to work on. Models can be built entirely through the spatch user interface, but the internal format is non-standard, so it is not possible to directly import models from other systems. Until very recently, no suitable standards have existed, but with the emergence of ChannelML (Crook et al. 2005) and MorphML (Qi and Crook 2004) this situation should improve. However, the creation of models in these formats may still take some time, whereas ModelDB (Mirsky et al. 1998) already provides a wide range of cell models with well characterized ion channels. The majority of these models are built for Neuron (Carnevale and Hines 2006) which uses a script-based specification format which is very difficult for other systems to access (Cannon et al. 2007). Recent work on standardising Neuron models to ChanelML/MorphML may eventually make a large fraction of these readily accessible in standardiazed formats which would be of considerable benefit to virtual lab software such as SPatch.

Numerous extensions are also possible in the software itself, such as the inclusion of receptor-channels or synaptic currents. In conjunction with further documentation, these should greatly extend the scope and usefulness of the system to enable more students to gain a solid understanding of electrophysical techniques without the need for costly and time-consuming laboratory practical work.

Acknowledgments

This work was funded by INSERM and a grand from the Ministère de l'Education Nationale.

Reference list

Cannon RC, Hasselmo ME, Koene RA (2003) From biophysics to behavior: Catacomb2 and the design of biologically-plausible models for spatial navigation. Neuroinformatics 1:3-42

Cannon RC, Gewaltig MO, Gleeson P and Cornelis H, Hines ML, Howell FW, Stiles JR, Wills S and de Schutter E (2007) Interoperability of neuroscience modeling software: current status and future directions. Neuroinformatics, 5 (2)

Carnevale NT, Hines ML (2006) The NEURON Book Cambridge: Cambridge University Press. 457 p.

Crook S, Beeman D, Gleeson P, Howell F (2005). XML for Model Specification in Neuroscience - an Introduction and Workshop Summary. Brains, Minds and Media, Vol. 1, bmm228 (urn:nbn:de:0009-3-2282).

Hammond C (2008) Cellular and Molecular Neurophysiology, 3rd edition, Academic Press, London.

Hodgkin AL, Huxley AF (1952) A quantitative description of membrane current and its application to conduction and excitation in nerve. J Physiol (Lond) 117: 500-544.

Mirsky JS, Nadkarni PM, Healy MD, Miller PL, Shepherd GM (1998) Database tools for integrating and searching membrane property data correlated with neuronal morphology. J Neurosci Methods 82: 105-121.

Qi W and Crook SM (2004) Tools for neuroinformatic data exchange: An XML application for neuronal morphology data. Neurocomputing 58-60C, 1091-1095.

Sakmann B and Neher E (1995) Single channel recording. Plenum press. New York.

Strassberg AF and Defelice LJ (1993) Limitations of the Hodgkin-Huxley formalism-Effects of single-channel kinetics on transmembrane voltage dynamics. Neural Comp 5: 843-855.