CELLULAR BIOPHYSICS:
# TEACHING AND LEARNING
WITH COMPUTER SIMULATIONS

By
Thomas Fischer Weiss

with the assistance of

Scott I. Berkenblit, Tanmaya S. Bhatnagar, Elana B. Doering, David Huang, David Koehler, Tommy Ng, Leela Obilichetti, Stephanie Peek, Devang M. Shah, Giancarlo Trevisan

Department of Electrical Engineering and Computer Science
Massachusetts Institute of Technology

FALL 2000
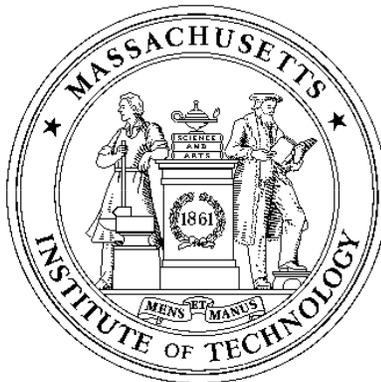
Date of last modification: August 28, 2000

Preface

Historical perspective

During the 1980's the use of computers and information technologies began to have an impact on higher education (Kulik and Kulik, 1986; Balestri, 1988; Wilson and Redish, 1989; Athena, 1990a; Athena, 1990b). As an integral part of this trend, in 1983 MIT in partnership with the Digital Equipment Corporation and the IBM Corporation launched Project Athena which was designed to make computation available to undergraduate students through a network of computers located in public clusters on the MIT campus (Athena, 1990a; Athena, 1990b). A major objective of Project Athena was to improve undergraduate education through the use of computation and information technologies. Faculty were encouraged to participate, and support for faculty software developers was provided on a competitive basis.

I had been involved in teaching cellular biophysics at MIT since the 1960's. The possibility of using software as a pedagogical aid was intriguing. With support from Project Athena, a software package on the Hodgkin-Huxley model for nerve excitation was developed as part of an undergraduate thesis (by David Huang), and was first used to teach cellular biophysics in the Fall 1984 semester. The software was designed to be easy to use so that a student's attention would be focussed on the Hodgkin-Huxley model and not on the computer. Informal discussions with students and a survey of student views showed that the software was an enormous success. During the first semester, the software was used primarily in lecture demonstrations and as the basis for student projects. Both pedagogic methods were found to be effective. The use of the software in lecture was very effective in motivating and engaging students. The student projects were effective in allowing students to pursue a research project of their choice with staff assistance. For many students this was their first experience with a research project. The use of these projects, developed in the first year, was so successful that it has been used ever since.

The initial results with the Hodgkin-Huxley software were so successful educationally, that several other software development projects involving student programmers were launched. During this phase of software development, 5 software packages were developed and, in one form or another, have been used every year to teach the subject. All of these packages were revised extensively in response to suggestions from students and staff. The original software runs on UNIX workstations under MIT's Project Athena and is available to the MIT community on a network of about 1000 UNIX workstations located in public clusters on the MIT campus as well as in some living groups. All this software was written in C and XWindows and was based on a library of graphic user interface subroutines written by one of the students (Giancarlo Trevisan). The software has been used in lectures, in recitations held in an electronic classroom in which each

student uses a workstation, in homework assignments, and in student projects. Various modes of use of the software in teaching were developed and are described briefly elsewhere (Weiss et al., 1992) and more extensively in the last chapter of this text. The software has become an integral part of the subject, and it is difficult to imagine teaching the subject without the software.

Several problems became apparent in the development and utilization of the software. First, it was very expensive, in time and in money, to develop the software with the software tools available in the late 1980's. Much of the time was expended in the development of graphic user interfaces that make the software easy for the user but which are tedious for the programmers to specify. These graphic user interfaces had to be written in a low-level language (XWindows). After 1991, funds from corporate sponsors were no longer available to support the development of new software which slowed considerably. Second, maintenance of the software became a major headache. It became difficult for a single faculty member with research, teaching, and other academic commitments to maintain a library of software in the face of changes in the operating systems. Third, as word spread about the existence of the software, educators and students outside of MIT requested the software. These requests accelerated dramatically after one of the software packages entitled *Hodgkin-Huxley Model* won the 1990 EDUCOM/NCRIPTAL Higher Education Software award for Best Engineering Software. However, almost all of the requests came from students and faculty with access to Macintosh or PC computers and not to UNIX workstations. Thus, when these people were informed that the software ran only on UNIX workstations, they invariably lost interest. At the time the software was written, the computational power of workstations so exceeded that of personal computers (PCs) that it was simply not possible to provide the type of performance on PCs that was achieved on the workstations. Furthermore, MIT's Project Athena was committed to a network of UNIX workstations. Thus, for both software and hardware considerations, it did not make sense to port the existing software to PCs. The high cost of software development and maintenance did not justify further development of educational software on UNIX workstations alone, and the development of new software was terminated in 1991.

By 1995, a number of developments made it feasible to address the problems described above and to develop software for teaching cellular biophysics in a manner that would make it easier to maintain, easier to modify, and widely available. Thus, all the software was rewritten to operate under MATLAB, which is a software package produced by The MathWorks, Inc., for the following reasons:

- MATLAB is a powerful interpretive computational and visualization software package with a large number of higher-level built-in functions. Thus, it is suitable for the development of educational software packages.

- MATLAB is available for most computer platforms. The vendor supports

changes in MATLAB that are required as changes in computer platforms occur. With the use of software built on MATLAB, this major maintenance job is transferred from individual faculty members to the vendor who has both the financial incentive and expertise to maintain the vendor software.

- Large improvements in performance of PCs have made the development of computationally intensive educational software feasible on these platforms.

- MATLAB has provided increasingly sophisticated tools for building graphic user interfaces (GUIs). These GUIs are essential for building user-friendly educational software packages.

- MATLAB has rapidly become the *de facto* leader in supporting educational computational subjects at MIT and elsewhere. Thus, students are exposed to MATLAB in other subjects and the different exposures are mutually reinforcing.

Here, we describe this MATLAB-based version of the software which now constitutes 6 packages. The most recent addition to the library of packages is one devoted to the propagated action potential along an unmyelinated nerve fiber.

Although the software is not linked directly to any textbook, it was developed in parallel with textbooks in cellular biophysics (Weiss, 1996a; Weiss, 1996b).

Acknowledgement

A number of people contributed to the success of the development of this software. We thank Project Athena, especially its two directors Steven Lerman and Earll Murman, for their support. In addition Gerald Wilson, Joel Moses, Richard Adler, Paul Penfield, and Jeffrey Shapiro were unfailingly supportive of this effort. A number of students were involved in this effort. For many students the software project constituted a portion of their undergraduate thesis requirement; others (as norws) used the software projects to satisfy SM thesis requirements. David Huang wrote the first version of the Hodgkin-Huxley model package. David Koehler also contributed to this package. Devang M. Shah wrote the first version of the random-walk model package which was also revised by Elana B. Doering. Chapter 2 is based heavily on Devang's thesis (Shah, 1990). Scott I. Berkenblit wrote the first version of the macroscopic diffusion package. Chapter 3 is based heavily on Scott's Master of Science thesis (Berkenblit, 1990). Stephanie Peek and Leela Obilichetti helped to develop the carrier-mediated transport package. Giancarlo Trevisan was a major contributor to all the packages. He wrote the first version of the voltage-gated ion channel package. He later rewrote the Hodgkin-Huxley package and the carrier-mediated transport package. He wrote all the graphic user interface routines that were ultimately used by all the packages. Generations of students benefited from his efforts. The

Contact information

Information on cellular biophysics texts, errata, changes, etc. can be found in the **study materials section.**

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

1.1   Motivation

I have been engaged in teaching cellular biophysics since 1965. Computer simulations were first used to teach this material in 1984 and have been used ever since. There is no question in my mind that the use of the software has deepened students understanding of the subject material. All faculty who have taught the subject, even those who started as skeptics, have been convinced of the educational efficacy of the software described here. Students have no basis of comparison of the subject before and after use of the software, but generally enjoy using the software and feel that it is helpful, and sometimes crucial, for understanding the concepts. This book provides a complete description of the software including some background material for each topic.

The first chapter introduces the software packages and MATLAB, on which the packages are based. The subsequent 6 chapters each describe one of the software packages in some detail complete with problems and projects. The last chapter is devoted to describing the pedagogical methods we have used to teach cellular biophysics using computer simulation. It is aimed at faculty interested in using the software for teaching but may be useful for students as well.

1.2   Overview Of Software Packages

The software for cellular biophysics consists of 6 software packages. The titles of the software packages, with acronyms in parentheses, are described briefly below.

Random Walk Model of Diffusion (RW) allows users to select parameters of the random walks of particles in a field and to observe the resulting space-time evolution of particle location. This package is intended to link the macroscopic laws of diffusion to its probabilistic, microscopic basis.

Macroscopic Diffusion Processes (MD) allows users to select the initial spatial distribution of solute concentration, the diffusion parameters, and to observe the resulting space-time evolution of solute concentration and flux. This package is intended to give users intuition about macroscopic diffusion processes.

Carrier-Mediated Transport (CMT) allows users to examine simple models of carrier-mediated transport through cellular membranes. For each of three models, the user can change any parameter and can instantly see the effect on the state of the carrier-model. This interactive mode is intended to build intuition about these models. In addition, there is a graphic mode that allows display of steady-state and transient responses to changes in any parameter.

Hodgkin-Huxley Model — Space Clamped (HH) allows users to investigate the space-independent Hodgkin-Huxley model. Users can change parameters of the Hodgkin-Huxley model of the space-clamped giant axon of the squid, and perform simulation experiments in either the voltage-clamp or the current-clamp configuration.

Hodgkin-Huxley Model — Propagated Action Potential (PAP) allows users to investigate the space-dependent Hodgkin-Huxley model. Users can change parameters of the Hodgkin-Huxley model of the giant axon of the squid, and examine propagation of the action potential and of sub-threshold potentials as a function of space and time.

Voltage-Gated Ion Channels (IC) allows users to design a voltage-gated ion channel. The user selects the number of states, the conductance and gating charge associated with each state, the voltage-dependence of state transition rates, and the membrane potential. The user can then perform simulation experiments on the channel model.

These packages are all designed to enhance comprehension of topics in cellular biophysics by providing pedagogic tools that can be used as a basis for lecture demonstration, open-ended problems that lend themselves to classes held in an electronic classroom in which students have access to computers, homework problems, and research projects. Although independent of any textbook, this suite of software packages was developed in parallel with textbooks in cellular biophysics (Weiss, 1996a; Weiss, 1996b).

1.3   Brief Introduction To MATLAB

All the software packages are written in MATLAB which is an interactive programming environment for numerical and symbolic computations and for visualization of computational results. Because MATLAB runs on most of the major computer platforms, the cellular biophysics software operates on all platforms supported by MATLAB. The present version of the software is written on MATLAB version 5.3 which is available under Windows and UNIX but not under MacOS. MATLAB version 5.3 is required to operate this software.[1]

The MATLAB environment is interpretive. That is, commands can be entered at a prompt and interpreted within the scope of a MATLAB session. Thus, computational results generated by a simulation are available to the user for further analysis. The software is designed to perform all the simulations with minimal typed commands. The graphical user interface allows navigation through the

---

[1]Although not tested extensively, the software should run on the Student Version of MATLAB. However, it may be that parameters can be chosen for these packages which will not satisfy the limitations of the Student Version.

software using a sequence of mouse events (e.g., clicking the mouse, pulling down a menu, dragging the mouse).

Users of the software do not need to learn MATLAB to use the software. However, knowledge of MATLAB can enhance user's usage of the software. A number of texts on MATLAB are available (Hanselman and Littlefield, 1997). In addition, MATLAB manuals can be ordered directly from MathWorks. Section 1.3.1 is a glossary of some useful terms in the MATLAB vocabulary. Section 1.3.2 mentions resources that can provide on-line help with MATLAB.

1.3.1   Jargon

The following terms are useful in navigating in the MATLAB environment.

axes The area in a figure containing plots and annotation. See the MATLAB command `axes`.

buttons Rectangular regions allowing a sequence of commands to be executed when they are clicked, selected, or edited. See the MATLAB command `uicontrol`.

command window The window that appears when MATLAB is started. Commands entered at the MATLAB prompt (>>) in this window are evaluated in the workspace.

figure A rectangular window containing graphical objects, such as axes, buttons, and menu items. See the MATLAB command `figure`.

MAT-file A binary data file containing MATLAB variables. Each software package uses and stores a different set of variables.

menubar A bar at the top of a figure (Windows and UNIX implementations) or at the top of the monitor screen (in some implementations) associated with the currently selected figure. When selected, the menu item expands to show its related submenu items. Submenu items marked with a checkmark are currently active selections.

M-file A text file containing a sequence of commands to be evaluated in the MATLAB workspace. The software contains a collection of m-files which can be recognized by the extension .m.

parameters Numeric values that define each model.

popup-menu A rectangular region showing the current popup-menu selection. When clicked, the menu is expanded to show all the options. See the MATLAB command `uicontrol`.

variables Numeric values of independent variables set by the user or of dependent variables calculated from the model.

workspace A collection of variables in the current session of MATLAB. When MATLAB is started, the workspace is empty. Variables appear in the workspace as they are defined.

1.3.2 Help

Help on MATLAB is available through the command window using the following commands at the MATLAB prompt (`>>`)

`who` lists the variables in the current workspace.

`help function` provides some help on the command `function`.

`help help` provides help on getting started using `help`.

`lookfor word` finds functions that involve `word`.

Additional help is available in the descriptions of individual software packages.

1.4 General Operation Of The Software

1.4.1 Directory structure

The software is designed to be run from a directory that includes the following files/directories:

cmt is a directory that contains the carrier-mediated transport software.

hh is a directory that contains the Hodgkin-Huxley model of a space-clamped axon software.

ic is a directory that contains the voltage-gated ion channel software.

md is a directory that contains the macroscopic diffusion processes software.

pap is a directory that contains the Hodgkin-Huxley model of a propagating action potential software.

rw is a directory that contains the random walk model software.

softcell.m is a MATLAB m-file that initializes the software and allows the user to choose software packages from a menu.

startup.m is a MATLAB m-file that may be used to startup the software automatically.
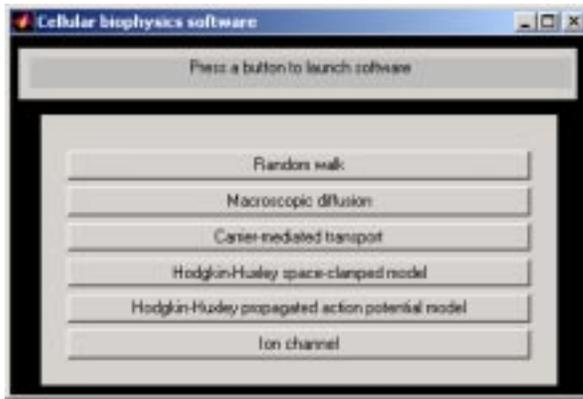
Figure 1.1: This Cellular biophysics software figure is displayed when the software is started and shows the available software packages. The appearance of the menu bar near the top of the figure varies somewhat for the different platforms. This figure was obtained using Windows 2000 and MATLAB version 5.3.

tlib is a directory that contains a library of software routines used by all the packages.

1.4.2   Starting the software

The software is accessible through MATLAB via a menu that allows selection of the software packages and initializes all packages. The startup procedure is given for the different platforms.

Windows, UNIX, and other operating systems that support MATLAB. Initialize MATLAB, and type the following instruction (in the MATLAB window) at the MATLAB prompt (>>)
>> `cd directory`
where `directory` is the name of the directory (folder) that houses the cellular biophysics software.[2] Then type
>> `softcell`
This command displays the *Cellular biophysics software* figure shown in Figure 1.1. Clicking on any package, initializes that package and hides the *Cellular biophysics software* figure.

UNIX workstation on Project Athena at MIT. From the dashboard at the top of the monitor select
`Courseware` $\Rightarrow$
`Electrical Engineering and Computer Science` $\Rightarrow$
`6.021J/6.521J Quantitative Physiology` $\Rightarrow$
`New MATLAB v5 Software`
This procedure displays the *Cellular biophysics software* figure shown in Figure 1.1. Clicking on any package, initializes that package and hides the *Cellular biophysics software* figure.

---

[2]To verify that the directory is the right one, either type `pwd` to indicate the name of the present directory or type `ls` to list the contents of the directory. It should contain the file `softcell.m`. If it does not contain this file then either the selected directory is wrong or the cellular biophysics software is not installed on your computer.

If a software package is selected, a number of figures are displayed including the *XX Control* figure, where XX is the acronym of the software package (RW, MD, CMT, HH, IC). Clicking on the × in the upper right-hand corner of the menubar, deletes any figure. Clicking on the × in the upper right-hand corner of the menubar of the *XX Control* figure exits from the software package and results in the display of the *Cellular biophysics software* figure shown in Figure 1.1.

1.4.3   Quitting the software

Clicking on the × in the upper right-hand corner of the menubar of the *Cellular biophysics software* figure exits the cellular biophysics software. Clicking on the × in the upper right-hand corner of the menubar of the *MATLAB Command* window exits MATLAB.

1.4.4   Printing a figure

Each figure associated with each package contains a *Print* button which if pressed brings up a print figure that allows the user to print on any printer on the network to which the computer is connected or to store a postscript file of the figure.

1.4.5   Reading from and saving to a file

A variety of information about the software can be saved in MAT-files using the standard MATLAB binary file format (see MATLAB's `save` and `load` commands). For example, all the packages allow storage of simulation parameters to allow a simulation to be repeated at a later time. In addition, results of simulations can also be saved in files for later retrieval. However, the information stored varies for different software packages, and the individual descriptions of the packages should be consulted for more detailed information.

A file can be read by pushing the *Open* button in a figure. A file can be stored by pushing the *Save* button. Pushing either button displays a figure (Figure 1.2) that allows the user to navigate to the desired directory to either read or save a file.

Figure 1.2: Save figure opened from the HH package using Windows 2000 and MATLAB 5.3.

Chapter 2

RANDOM WALK MODEL OF DIFFUSION
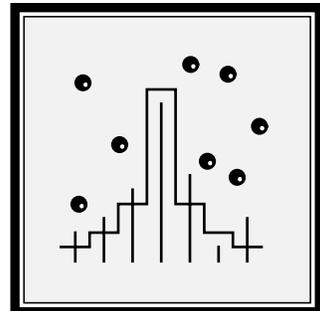
2.1 Introduction

2.1.1 Historical background

A bolus of soluble material will gradually spread out in its solvent until a uniform solution results. This diffusion process must have been familiar to humans in antiquity. However, a mathematical description of these *macroscopic* changes in concentration was not available until the 1850s (Fick, 1855), and a *microscopic* or particle-level model, not until the turn of this century (Einstein, 1906).

Diffusion plays an important role in such a wide range of disciplines, that it is important for students of science and engineering to develop an understanding of the macroscopic laws of diffusion and their microscopic basis. We will review some important characteristics of the macroscopic laws of diffusion and their relation to random-walk models. A fuller treatment is available elsewhere (Weiss, 1996a).

2.1.2 Macroscopic and microscopic models

Macroscopic laws of diffusion

The macroscopic laws of diffusion for the simple case when the particles are not subject to a body force, the medium does not convect the particles, the diffusion coefficient is a constant, and the particles are conserved are summarized in one and three dimensions in Table 2.1. These equations relate the flux of particles

Equation	One Dimension	Three Dimensions
Fick's first law	$\phi = -D\dfrac{\partial c}{\partial x}$	$\bar{\phi} = -D\overline{\nabla}c$
Continuity equation	$\dfrac{\partial \phi}{\partial x} = -\dfrac{\partial c}{\partial t}$	$\nabla \cdot \bar{\phi} = -\dfrac{\partial c}{\partial t}$
Diffusion equation	$\dfrac{\partial c}{\partial t} = D\dfrac{\partial^2 c}{\partial x^2}$	$\dfrac{\partial c}{\partial t} = D\nabla^2 c$

Table 2.1: The macroscopic laws of diffusion for a concentration of particles diffusing in a homogeneous region with a constant diffusion coefficient D, in the absence of a body force on the particles or convection of the medium, and where the particles are conserved.

(ϕ), which is the number of moles of particles transported through a unit area in a unit time, to the concentration of particles (c), which is the number of moles of particles per unit volume. Fick's first law relates flux to particle concentration; it is analogous to other laws that relate a flow to a force such as Ohm's law of electric conduction, Darcy's law of convection, and Fourier's law of heat flow.

Fick's first law implies that a solute concentration gradient causes a solute flux in a direction to reduce the concentration gradient. The continuity equation follows from conservation of particles, and the diffusion equation is obtained by combining Fick's first law with the continuity equation.

Microscopic basis of diffusion

An important notion in understanding diffusion processes is to relate the macroscopic laws of diffusion to microscopic models of diffusing particles. The simplest microscopic model that captures the essence of diffusion is the discrete-time, discrete-space random walk. In a one-dimensional random walk in a homogeneous region of space, we assume a particle moves along the x-axis in a series of statistically independent steps of length $+l$ or $-l$, where the time between steps is τ. In an unbiased walk, positive and negative steps are equally likely, i.e., each has probability $1/2$. It can be shown that statistical averages of properties of a population of particles obey the macroscopic laws of diffusion. In particular, this simple model can be shown (Weiss, 1996a) to yield Fick's first law with a diffusion coefficient,

$$D = \frac{l^2}{2\tau}.$$

Therefore, the connection between the random walk of a particle and the laws of macroscopic diffusion can be made clear if the motion of a number of particles (on the order of 50) can be visualized for a number of steps.

2.1.3   Overview of software

The software described here is intended to allow users to investigate the properties of the simplest microscopic model that captures the essence of diffusion: the discrete-time, discrete-space random-walk model.

   In the discrete-time, discrete space random walk model described here, there is a population of particles which execute statistically-independent, but otherwise identical two-dimensional random walks in a rectangular field. The field can be divided into one, two, or three homogeneous regions whose widths are specifiable, and whose properties may differ. Each particle undergoes a random walk with parameters that include: the probability that the particle takes a step to the left or right, and the step size. These parameters can be set independently in the three regions. The particles can be set to have a specifiable lifetime. One source and one sink of particles can be placed in the field and the initial concentration of particles can be specified in each of the three regions. Characteristics of the boundary conditions between regions can also be specified. With this software package it is possible: to visualize the spatial evolution of particle concentration from a variety of initial distributions selectable by the user; to examine the evolution of particle concentration from a source and in the

1 unit

$j+2$	+	+	+	+ +
$j+1$	+	+	+	+ +
j	+	+	●	+ +
$j-1$	+	+	+	+ +
$j-2$	+	+	+	+ +
	$i-2$	$i-1$	i	$i+1$ $i+2$
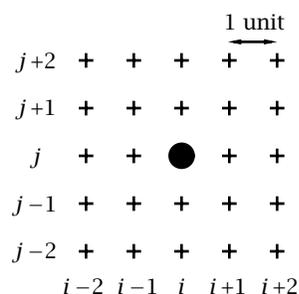
Figure 2.1: Definition of grid of particle locations. A particle is shown at the grid location (i, j). The distance between adjacent grid locations, in both the horizontal and vertical directions, is the unit distance.

presence of a sink; to examine diffusion in regions of differing diffusion coefficients; to simulate diffusion of particles subjected to a body force; to simulate diffusion between two compartments separated by a membrane; to investigate the effects of chemical reactions or recombination which consume particles at a fixed rate; and to investigate the effects of different boundary conditions between regions. Two diffusion regimes can be run and displayed simultaneously to allow direct comparison between the space-time evolution of two different diffusion processes. In addition, a variety of statistics of the spatial distribution of particles can also be displayed.

By watching the particles move and by comparing simulation results to expectations, the user can develop an intuition for the way in which the random motions of particles lead to their diffusive spread.

2.2   Description Of The Random-Walk Model

In this simulation, the discrete-time, discrete-space random walk takes place on a finite two-dimensional grid of locations accessible to the particles and called the *field*. The location of each particle is specified by giving its coordinates on this grid (i, j) where i is the horizontal coordinate and ranges from 0 to 399 and j is the vertical coordinate and ranges from 0 to 99 (Figure 2.1). The horizontal distance between adjacent grid locations is 1 unit of distance and all spatial dimensions of the random walk are expressed as multiples of this unit distance. The location of the particle in the grid can change probabilistically at each step of the random walk. Thus, successive steps represent successive times that are separated by a unit time interval. All times are expressed in terms of the number of steps of the random walk.

The field can be divided into one, two or three homogeneous *regions* (Figure 2.2). Certain parameters of the simulation are defined for the entire field, others at boundaries between regions, and still others are defined independently for each region. The latter parameters will be described first and include: region size, particle step size, directional probabilities, and initial particle distribution.
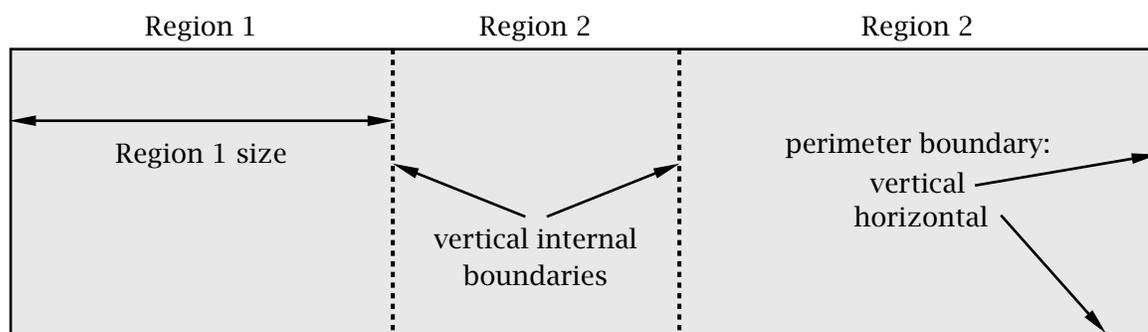
Figure 2.2: Definition of simulation field, regions, and boundaries. Solid lines delimit the perimeter boundaries of the field; dashed lines indicate the internal vertical boundaries that separate the regions.

2.2.1   Particle parameters within a region

The parameters that define the random walk are identical at each location *within* a region — each region is homogeneous. These parameters are described below.

Region size

The width of each region can be specified, but the sum of the widths cannot exceed 400. This allows a variety of diffusion regimes to be defined. For example, if Region 1 has width of 400 then the other two regions must have width 0 and the random walk is defined for one homogeneous region. By specifying two regions with non-zero widths, it is possible to define a diffusion process with different initial conditions in the two regions. This allows a rich variety of initial distributions to be defined. Three non-zero width regions allows simulation of diffusion between two regions separated by a third region with different properties. This might be used to investigate diffusion between two baths separated by a membrane.

Step size

The step size defines the distance, in multiples of unit distances, that particles may move in each step of time. Varying the step size simulates varying the diffusion coefficient. The size of a region is always set to a multiple of the step size in that region; all particles in a region are located at integer multiples of the step size starting from the left boundary of the region. This ensures that particles at a boundary fall on the boundary and simplifies the specification of particle motion at a boundary.
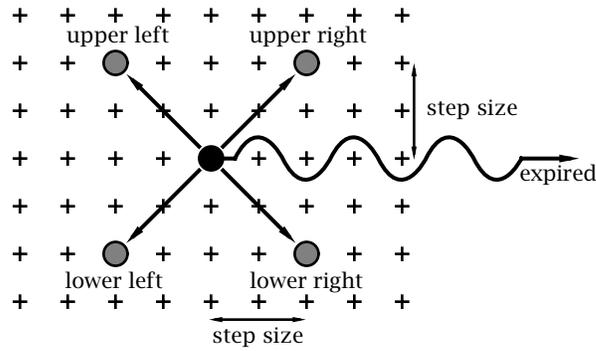
Figure 2.3: Schematic diagram of motion of a particle in a homogeneous region. The grid of possible particle locations, separated by unit distances, are indicated by + symbols. A particle is shown in the center of the figure at one instant in time. One time step later the particle either stays in the same location or moves to one of 4 possible locations (indicated by the shaded particle) or it expires (is removed from the field). If the particle moves, it translates one step size (here shown as 2 units of distance) in both the vertical and the horizontal direction.

Particle motion — directional probabilities

At each instant in time, a particle is at some location in the region. The disposition of the particle at the next instant in time is determined by one of six mutually exclusive and collectively exhaustive possibilities as illustrated in Figure 2.3. The particle can move one step size to the upper left, upper right, lower left, or lower right; stay in the same location (center); or be eliminated (expire). The probabilities for each of the six outcomes is as follows:

$$
\begin{aligned}
P[\text{expired}] &= 1/L \, , \\
P[\text{center}] &= (1 - 1/L)\,(1 - p - q) \, , \\
P[\text{upper left}] &= 0.5\,(1 - 1/L)\,q \, , \\
P[\text{upper right}] &= 0.5\,(1 - 1/L)\,p \, , \\
P[\text{lower left}] &= 0.5\,(1 - 1/L)\,q \, , \\
P[\text{lower right}] &= 0.5\,(1 - 1/L)\,p \, ,
\end{aligned}
\tag{2.1}
$$

where L is the average lifetime of the particle, i.e. the average number of time steps to expiration; p is the conditional probability that the particle moves to the right given that it has not expired; q is the conditional probability that the particle moves to the left given that it has not expired. Note that while the probability of moving to the left and to the right can differ, the probability of moving up or down is always the same. Because the six probabilities define all the possible outcomes at each instant in time, they sum to unity.

Different types of random walks are described by changing the directional probabilities. The random walk defined by assuming $p = q = 1/2$ is the simple,

unbiased random walk described in Section 2.1. In general, if $p = q$ the random walk is unbiased; there is no statistical tendency for particles to move preferentially in either horizontal direction. However, if $p \neq q$, the random walk is biased so that there is a tendency for particles to move in one horizontal direction. For a step size of S, the mean distance $E[m]$ that the particle moves to the right in n units of time is

$$E[m] = Sn(1 - 1/L)(p - q) \, .$$

2.2.2   Boundary conditions

The field contains three different types of boundaries (Figure 2.2) which are, in order of increasing complexity, horizontal perimeter boundaries at the top and bottom of the field, vertical perimeter boundaries at the left and right ends of the field, and vertical internal boundaries that separate regions.

Horizontal perimeter boundaries

Horizontal perimeter boundaries act as perfectly reflecting walls. If a particle is located within one step size of such a boundary and takes a step toward the boundary then the new vertical location of the particle is determined in the following manner: the vertical distance the particle travels to reach the wall plus the vertical distance the particle travels after reflecting from the wall must sum to the step size. This relation determines the new location given the old location and the value of the step size.

Vertical perimeter boundaries

The vertical perimeter boundaries are also reflecting walls. Because the probabilities of stepping to the left and right need not be the same, we found that purely reflecting wall of the type described for the horizontal perimeter boundaries created undesirable artefacts especially when the conditional probabilities of moving to the left and right were not equal ($p \neq q$). Therefore, we modified the boundary condition so that a particle that would have crossed a vertical perimeter boundary at a given step was placed on the boundary and then subject to the following boundary condition which is illustrated for the left boundary in Figure 2.4 and whose directional probabilities are:

$$
\begin{aligned}
P[\text{expired}] &= 1/L \, , \\
P[\text{center}] &= (1 - 1/L)(1 - p) \, , \\
P[\text{upper right}] &= 0.5(1 - 1/L)\, p \, , \\
P[\text{lower right}] &= 0.5(1 - 1/L)\, p \, ,
\end{aligned}
\tag{2.2}
$$

i.e. the particle cannot move to the left.
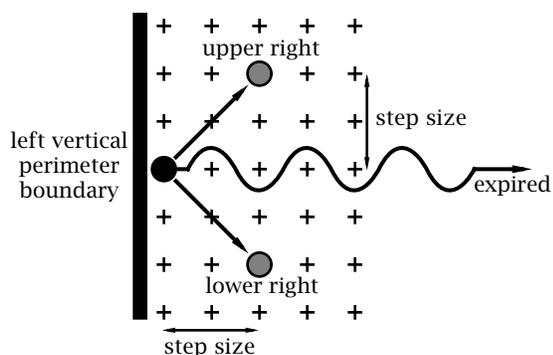
Figure 2.4: Schematic diagram of motion of a particle at a vertical perimeter boundary. For purposes of illustration the particle motion at the left boundary is shown. Conditions at the right boundary are the mirror-image of those shown here.
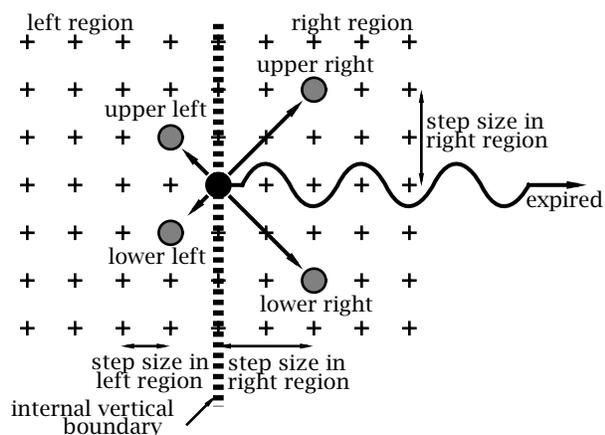
Figure 2.5: Schematic diagram of motion of a particle at an internal vertical boundary between two regions. The step size is 1 to the left of the boundary and 2 to the right of the boundary.

Vertical internal boundaries

The motion of particles at a vertical internal boundary is similar to that within a homogenous region. The differences are that: the step sizes in the two adjacent regions may differ; and special directional probabilities, specified by the user, apply at the boundary. These have been provided to allow users to explore the consequences of a rich variety of boundary conditions. To simplify boundary conditions, the software ensures that particles do not cross this boundary in one time step but rather they land on the boundary. This is guaranteed by forcing the width of boundaries, initial particle locations, locations of sources and sinks to be commensurate with the step size. Given this restriction, the possible outcomes for a particle on a boundary are shown schematically in Figure 2.5. The directional probabilities are identical to those in a homogeneous (region given above) except that the conditional probability of moving to the left and to the right given that the particle did not expire (p and q in the homogeneous region) are independently specified at each internal boundary. The step size to the left is equal to the step size in the region to the left of the boundary; the step size to the right is equal to the step size in the region to the right of the boundary.

Figure 2.6: The RW Control figure.

2.3   User's Guide To The Software

When this software package is selected, 2 figures are displayed (in addition to MATLAB's command window) — RW Control and RW Particle set #1.

2.3.1   RW Control

The RW Control figure controls the random walk software and is shown in Figure 2.6. The part of the RW Control figure below the menubar is divided into two panels. The left panel indicates which of the two independent simulations are to be displayed — the choices are Particle Set #1 and/or Particle Set #2. When RW is selected, Particle Set #1 is selected by default.

The right panel allows the simulation to be started, paused, continued, or reset. If *Reset* is chosen, the step counter is set to 0 and the particle locations are reset to their initial locations. The # steps in the random walk before the simulation pauses can be entered in the text edit box. Clicking *1-step* executes one step of the random walk. A counter for the current number of steps in the random walk is displayed.

2.3.2   RW Particle

The RW Particle figure shows the locations of particles in the field as shown for an initial distribution of particles in Figure 2.7. When the simulation is started by clicking the *Start* button in the RW Control figure, the random walks of the particles are exhibited in the RW Particle figure. A step counter indicates the number of steps taken in the random walks. After 100 steps of the random walks of particles, the RW Particle figure is as shown in Figure 2.8.

Several options are provided in the RW Particle figure. The user can select *Param* to change the parameters of the random walk, *Summ* to obtain a numerical summary of statistics of the random walk, *Hist* to obtain a histogram of particle positions, and *Graphs* to obtain plots of statistics versus step number. These options are described in detail below. The *RW Particle* figure also has options to *Open* a file that contains a particle distribution, *Save* the current particle distribution in a file, and *Print* the current figure. Clicking on *Save* saves the following: the current state of all the particles including the expected locations of particles, the histogram binwidth, the step #, and all the parameters.
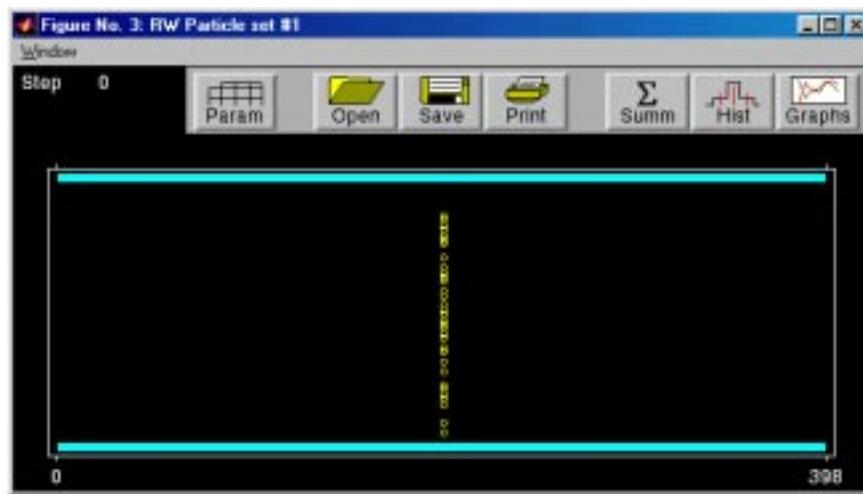
Figure 2.7: RW Particles figure showing the initial locations of particles. Initially there are 50 particles located at position 200 and spread randomly with a uniform distribution in the vertical direction.
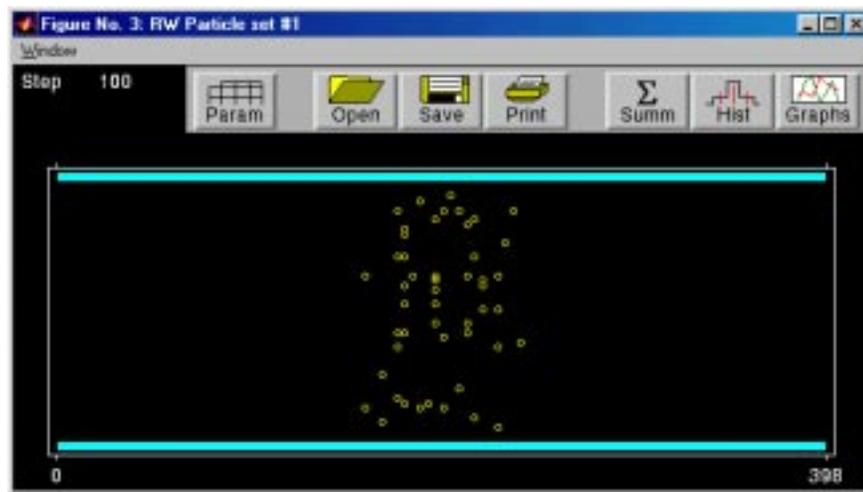
Figure 2.8: RW Particles figure showing the locations of particles after a 100 step random walk for the initial locations of particles shown in Figure 2.7.
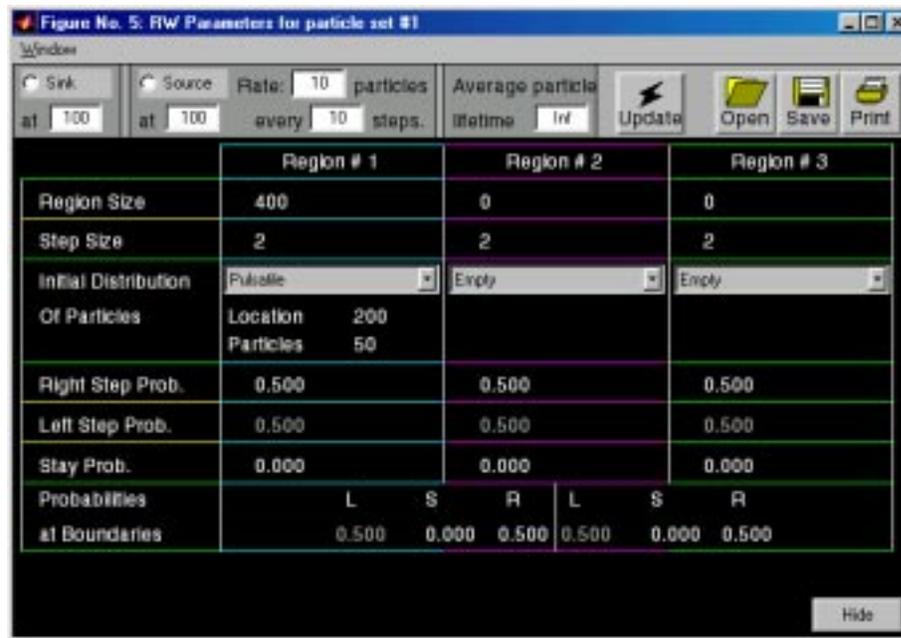
Figure 2.9: RW Parameters figure showing the parameters used to define the simulations whose particle distributions are shown in Figures 2.7 and 2.8.

RW Parameters

Choosing *Param* in the RW Particles figure displays the RW Parameters figure, shown in Figure 2.9, which allows selection of the parameters of the simulation. Region #1 is indicated in blue in both the RW Parameters figure and in the RW Particle figure; magenta is used for region #2 and green for region #3. After new parameters are chosen, clicking on *Update* resets the initial distribution to the newly selected one and activates all the new parameters. The parameters are described below. In addition, the parameters (alone) can be saved in a file (by clicking on *Save*) or read from a file (by clicking on *Open*).

- **Region Size.** The user may define up to three regions, whose combined width may not exceed 400. To eliminate a region, enter 0 for the region size. Regions can also be of size $k(stepsize)$ where $k = 3, 4, 5, ...$ That is, regions of size $stepsize$ and $2(stepsize)$ are not allowed.

- **Step Size.** The step size can be specified independently in each region.

- **Initial Distribution Of Particles.** The initial distribution of particles can be specified independently in the three regions. Particles start at locations that are integral multiples of the step size in each region. Particles are distributed randomly (with a uniform distribution) in the vertical direction and have the selected distribution in the horizontal direction. The initial distribution of particles can be selected to be one of the following.

- **Empty** implies that initially there are no particles in the region.

- **Pulsatile** implies that a specified number of particles are placed at a specified horizontal location in the region and spaced randomly in the vertical direction. Figure 2.7 shows an initial distribution that is pulsatile.

- **Linear** implies that a linear concentration profile is generated whose slope and number of particles are specified. A uniform distribution of particles is obtained if the slope is set to zero. Negative concentrations are not allowed: if the parameters are chosen such that the concentration would become negative at some point in the region, these putative negative concentrations are set to zero.

- **Sinusoidal** implies that the spatial distribution is sinusoidal with a specified period and number of particles.

- **Right Step Prob.** The conditional probability of a step to the right given that the particle has not expired can be specified independently in each region.

- **Left Step Prob.** The conditional probability of a step to the left given that the particle has not expired *cannot* be specified. It is set automatically so that in each region, the sum of the probabilities of all possible outcomes is one.

- **Stay Prob.** The conditional probability that the particle stays in the same location given that the particle has not expired can be specified independently in each region.

- **Probabilities at Boundaries.** The perimeter boundaries of the field act as reflecting boundaries; the boundaries between regions are characterized by user-defined boundary conditions. Boundary conditions are defined as shown in Figure 2.9: left-step-probability L, stay probability S , and right-step-probability R at the boundary between regions. For example, if the Region #1-#2 has the set of probabilities $(L, S, R) = (0, 0, 1)$ then particles at the boundary may only step to the right. Therefore, to a particle from Region #2, this looks like a reflecting boundary, whereas all particles reaching the boundary from Region #1 will cross into Region #2. As with the directional probabilities within a region, the user selects the conditional (on not expiring) probability of a step to the right and the conditional probability of no step; the conditional probability of a step to the left is then computed so that probabilities of all possible outcomes sum to one.

- **Source.** One source can be placed at a horizontal location in the field and covers the entire vertical height of the field. The source must be located at an integer multiple of the step size from the left boundary of each region

Figure 2.10: RW Summary figure showing a numerical summary of statistics after a 100 step random walk for the same parameters as shown in Figure 2.9 and for the initial locations of particles shown in Figure 2.7.

in which it is located. The rate of generation of particles is specified by giving the number of particles (specified as a two-digit number) produced in some number of time steps (specified as a two-digit number). Thus the rate of generation of particles can be a rational number. The particles are generated at the horizontal location of the source and are distributed randomly along the vertical direction using a uniform distribution. When the source is turned on, a small + is placed at the horizontal location of the source just above the field.

- **Sink.** A sink can be placed at a horizontal location in the field and covers the entire vertical height of the field. The sink is located at an integer multiple of the step size from the left boundary of each region in which it is located. Particles that land on the sink are absorbed; hence the number of particles at the sink is always zero. When the sink is turned on, a small − is placed at the horizontal location of the sink just above the field.

- **Average Lifetime.** The user may define the average particle lifetime defined as the number of steps it takes, on average, for the particle to expire.

RW Summary

Choosing *Summ* in the RW Particles figure pauses the simulation and displays numerical values of statistics in the RW Summary figure as shown in Figure 2.10. Two sets of computations are performed simultaneously during a simulation: statistics based on the actual locations of particle and statistics based on the expected locations of particles. Given a particle location at one step and a set of directional probabilities, a random number generator is used to determine which of the possible new locations occurs at the next step. These sequence of locations determines the locations of the particles on the screen and all the statistics labelled *actual*. However, given a particle location at one time and the

same directional probabilities, an *expected* location of the particle in the next step is also computed. Thus, during a simulation both the set of actual and expected locations for the particles are computed and can be displayed.

The available statistics (both *actual* and *expected*) are as follows.

- **Mean** is the mean location of the particles in the entire field.

- **Standard Deviation** is the standard deviation of particle location in the entire field.

- **#Generated** is the cumulative number of particles generated by the source since the beginning of the simulation.

- **#Absorbed** is the cumulative number of particles absorbed by the sink since the beginning of the simulation.

- **#Expired** is the cumulative number of particles lost (due to their finite lifetimes) since the beginning of the simulation.

- **Total #Particles** is the total number of particles in the entire field at each step number.

- **Region 1 #Particles** is the total number of particles in Region 1 at each step number.

- **Region 2 #Particles** is the total number of particles in Region 2 at each step number. For this total only, the particles located at the boundary between Region 1 and Region 2 are counted as belonging to Region 2.

- **Region 3 #Particles** is the total number of particles in Region 3 at each step number. For this total only, the particles located at the boundary between Region 2 and Region 3 are counted as belonging to Region 3.

RW Histogram

Choosing *Hist* in the RW Particles figure pauses the simulation and displays the RW Histogram figure (Figure 2.11). Histograms summarize the spatial distribution of particle locations; both the actual and the expected distribution of particle locations are shown. Each histogram consists of a set of bins that spans the field in the horizontal direction; the number of bins depends upon the specification of the bin size. With a bin size of 16, there are 25 bins that span the entire field of 400 locations. The histogram shows the number of particles in each bin as a function of bin location. Choice of bin size is important. A small bin size depicts the particle distribution with high spatial resolution. However, a small bin size implies that each bin will contain relatively few particles and the number of particles will fluctuate randomly from bin to bin. Conversely, a large
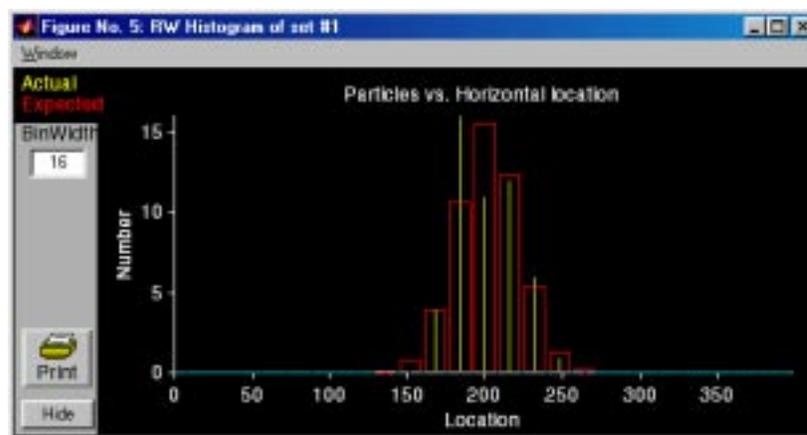
Figure 2.11: RW Histogram figure showing a histogram of particle locations after a 100 step random walk for the same parameters as shown in Figure 2.9 and for the initial locations of particles shown in Figure 2.7. The histogram is for the particle locations shown in Figure 2.8.

Figure 2.12: RW Axis figure showing options for changing axis properties.

bin size gives a histogram with poor spatial resolution but a larger amount of statistical averaging of the spatial distribution of particles.

The shape of the histogram is sensitive to the choice of bin size and can lead to confusing patterns. For example, suppose the bin size is five and the step size is two. Suppose further that particles are located uniformly in the field; one particle per accessible location. However, the step size constrains the possible locations that a particle may occupy to be separated by 2. Therefore, with a bin size of five, successive bins in the histogram alternate between 2 and 3 particles. Thus the histogram will not appear uniform, but oscillatory. This problem is cured if the bin size is an integral multiple of the step size. If the step sizes differ in the three regions, then the bin size should be set equal to the least common multiple of the three step sizes.

Clicking on either the ordinate or the abscissa axis label displays the RW Axis figure (Figure 2.12) that allows the user to change the axis scale. The scale is indicated by a vector [min max] in which min represents the minimum value and max represents the maximum value of the range. Clicking on *Log* makes the axis scale logarithmic.
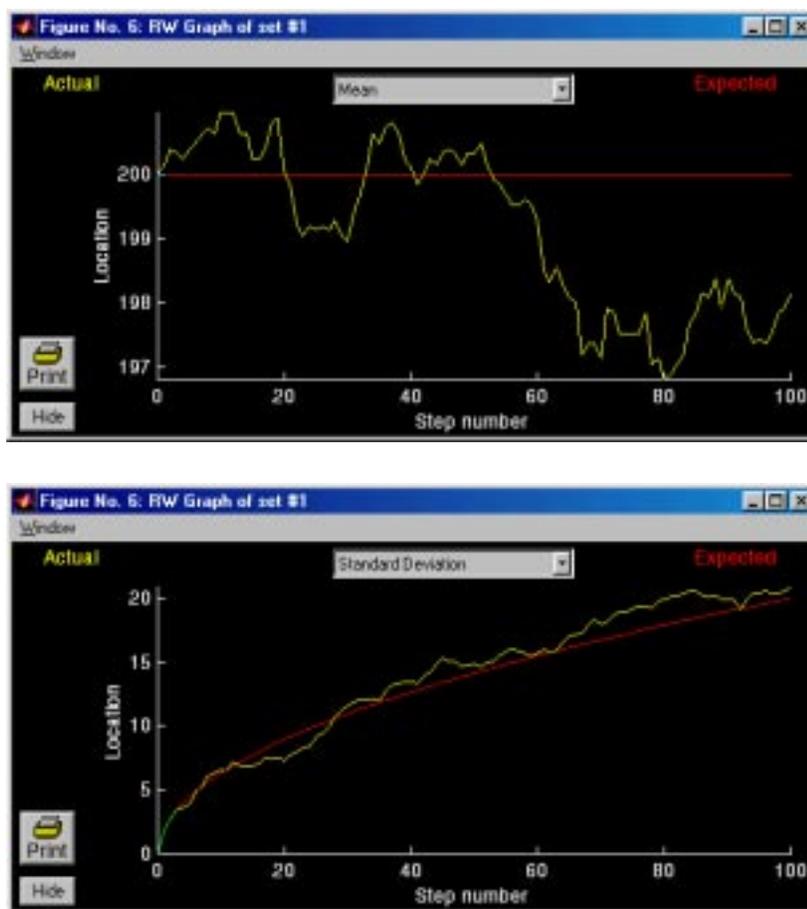
Figure 2.13: RW Graph figures showing the mean (above) and the standard deviation (below) of particle location versus step number for the same parameters as shown in Figure 2.9 and for the initial locations of particles shown in Figure 2.7.

RW Graph

Choosing *Graphs* in the RW Particles figure pauses the simulation and displays the RW Graph figure. The temporal evolution of any of the summary statistics shown in Figure 2.10 can be plotted versus the step number. Figure 2.13 shows an example of both the mean and the standard deviation of particle locations plotted versus step number. Clicking on either the ordinate or the abscissa axis label displays the RW Axis figure that allows the user to change the axis scale (as in Figure 2.12).

These statistics allow a quantitative evaluation of simulation results. We give a number of examples of the use of these statistics. A systematic change in the mean location of the particles as a function of step number demonstrates a drift or migration of the particles as can be achieved by a bias in the directional probabilities. A difference in the standard deviation of two distributions can

be achieved by changing the step size. The dependence on step number of the number of particles in the field, in a region, generated, absorbed, or expired can be used to assess whether a particle distribution has reached steady state. The change in the total number of particles in each region can be used to estimate the rate of transport of particles between regions.

2.4   Problems

Problem 2.1 *The purpose of this problem is to explore random walks from an initial distribution where all particles begin from the same horizontal location.*

   a. Use the default simulation parameters with a pulsatile initial distribution of particles but set the step size to 2 and 4 for Particle Sets #1 and #2, respectively. The parameters of Particle Sets #1 and #2 should differ only in step size. Run the simulation for 100 steps. Examine graphs of the mean and the standard deviation of the horizontal locations of the particles versus step number.

      i. Explain quantitatively the differences in the graphs in terms of the difference in step size of the two simulations.

      ii. Why does the slope of the standard deviation decrease with step number? What type of dependence on step number do you expect? Explain. [Hint — You may wish to replot the graph in different coordinates to check out your expectations.]

   b. Now set the step size to 2 for both simulations and decrease the number of particles in the pulsatile initial distribution for one of the simulations to 25. Rerun the simulation. Describe how the number of particles affects the dependence of mean location and standard deviation on step number? Explain your results.

   c. Set all the parameters at their default values and then set the step size for both Particle Sets #1 and #2 to 2. For Particle Set #1 set the right step probability to 0.75, the left step probability to 0.25, place an impulse of 50 particles at location 100. For Particle Set #2 set the right step probability to 0.25, the left step probability to 0.75, place an impulse of 50 particles at location 300. Run the simulation for at least 100 steps. Examine graphs of the mean and the standard deviation of the horizontal locations of the particles versus step number. Explain your results quantitatively.

Problem 2.2 *The purpose of this problem is to explore the steady-state distribution of particles in the presence of a source and sink for both unbiased and biased random walks.* For all parts of this problem set the parameters for both Particle

Sets #1 and #2 as follows: make the size of Region #1 400; the step size 20; and make the initial distribution of particles empty. Also place a source at location 0 and a sink at location 400 and make the histogram bin size 20.

a. Make the random walks for both Particle Sets #1 and #2 unbiased with a right step probability of 0.5. For Particle Set #1 make a source that generates 1 particle every 2 steps and for Particle Set #2 make a source that generates 1 particle every 1 step. Make the particle lifetime infinite for both simulations. Now run the simulation. In the steady-state, by definition the particle distribution will not depend upon step number.

   i. Before you run the simulation, estimate the form of the steady-state particle distribution. Will it be uniform, exponential, pulsatile, linear, Gaussian, or none of these?

   ii. Using the available statistics, determine a criterion for estimating when the particle distribution is in steady state.

   iii. Now run the simulation until your criterion for steady state is met. Look at the histograms of particle location. What is the steady-state particle distribution? Does it fit with your initial expectation? There are differences between the actual and expected histograms. Explain these differences.

   iv. How can you explain the difference in steady-state distribution for Particle Sets #1 and #2?

   v. Show that the steady-state particle distribution you have found for the random walks is consistent with Fick's laws.

b. Use the same parameters for Particle Set #2 as in part a, but change those of Particle Set #1 so that the source rate is also 1 particle every 1 step; and so that the probability of a right step is 0.55. Run the simulation until it has reached a steady state.

   i. Compare the steady-state distributions for the biased and unbiased random walks. How do they differ?

   ii. Describe the shape of the steady-state distribution for the biased random walk?

   iii. Modify Fick's first law to account for not only diffusion of particles but a steady drift of particles of velocity v. Determine the steady-state distribution of particles predicted from Fick's laws and the continuity equation in the presence of this steady drift of particles. Is this distribution consistent with the simulation results you obtained?

c. Keep the parameters of Particle Sets #1 and #2 the same as in part b, but change the right step probability of Particle Set #2 to 0.48. Explain the

differences in the shapes of the steady-state distributions for the two simulations.

Problem 2.3 *The purpose of this problem is to explore the effect of a finite particle lifetime on the statistics of a random walk.*

   a. Define a simulation with an initial particle distribution that is uniformly distributed in space. Set the average particle lifetime to be finite. Run the simulation until all the particles have expired. Examine a graph of # particles versus step number. Explain the shape of this function.

   b. Define another simulation that is identical with the one in part a, but this time with a source of particles. Do you expect a steady state to be reached between the the source and expirations? Determine a criterion for steady state and check out your intuition.

   c. Design a combination of source rate and lifetime such that the distribution reaches a steady state. This might correspond, for example, to a drug concentration reaching steady state in the body, as it is both infused and eliminated.

   d. Design a combination of source rate and lifetime such that most particles in the region have expired by the time the source injects the next batch. This might correspond to the drug concentration when the dosing interval is longer than the elimination half-life of the drug.

Problem 2.4 *The purpose of this problem is to explore random walks in three regions which simulates diffusion through a membrane that separates two baths.* Set up the parameters as shown in Figure 2.14. For both simulations, the two end regions contain uniform concentrations of particles; but the number of particles differs. There is a concentration difference between the end regions as shown in Figure 2.15 for one of the simulations; the other is similar. The center region can be regarded as a membrane that separates the two end regions and initially contains no particles. The random walk in the membrane region is unbiased in one simulation and biased in the other. Run the simulation for several hundred steps.

   a. Explain the differences between the distributions of particles for the two simulations.

   b. Quantitatively describe the flux of particles between the two end compartments — both the magnitude and sign of the flux.

   c. What physical process might be represented in the simulation with Particle Set #2?

Figure No. 5: RW Parameters for particle set #1

Window

C Sink at 100 C Source at 100 Rate: 10 particles every 10 steps. Average particle lifetime Inf Update Open Save Print

	Region # 1	Region # 2	Region # 3
Region Size	160	32	160
Step Size	4	4	4
Initial Distribution Of Particles	Linear	Empty	Linear
	Slope 0 Particles 150		Slope 0 Particles 50
Right Step Prob.	0.500	0.500	0.500
Left Step Prob.	0.500	0.500	0.500
Stay Prob.	0.000	0.000	0.000
Probabilities at Boundaries	L 0.500 S 0.000 R 0.500	L 0.500 S 0.000 R 0.500	

Hide

Figure No. 8: RW Parameters for particle set #2

Window

C Sink at 100 C Source at 100 Rate: 10 particles every 10 steps. Average particle lifetime Inf Update Open Save Print

	Region # 1	Region # 2	Region # 3
Region Size	160	32	160
Step Size	4	4	4
Initial Distribution	Linear	Empty	Linear
Of Particles	Slope 0 Particles 150		Slope 0 Particles 50
Right Step Prob.	0.500	0.400	0.500
Left Step Prob.	0.500	0.600	0.500
Stay Prob.	0.000	0.000	0.000
Probabilities at Boundaries	L 0.500 S 0.000 R 0.500	L 0.500 S 0.000 R 0.500	

Hide

Figure 2.14: Parameters for a random walk in 3 regions (Problem 2.4).
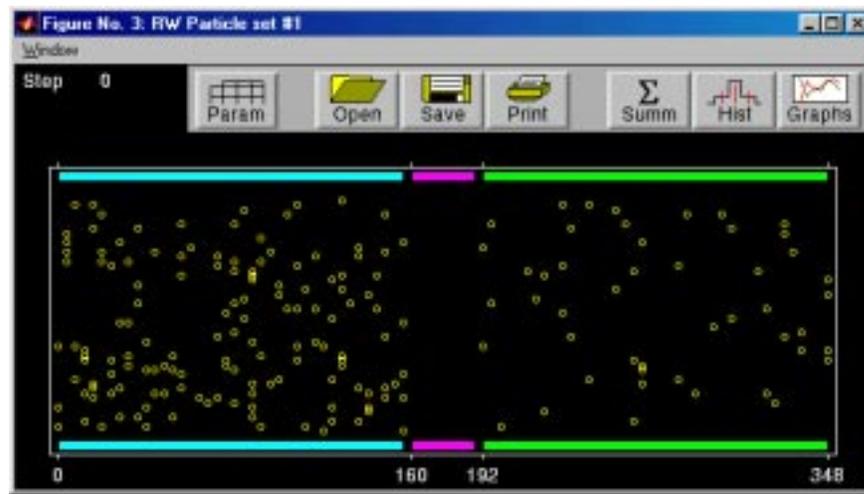
Figure 2.15: Initial locations of particles for a random walk in 3 regions (Problem 2.4).

Problem 2.5 *The purpose of this problem is to explore random walks in three regions which simulates diffusion through a membrane that separates two baths.* Design a three region diffusion regime where the center region is considered to be a membrane. Design membrane characteristics such that the membrane is transparent for particles moving from left to right but purely reflecting for particles moving from right to left. Test your design by running the simulation.

Problem 2.6 *The purpose of this problem is to examine the steady-state particle distribution that results in the presence of sources and sinks.* Start with two identical simulations with Regions #1 and #2 that have: a region size of 400; a step size of 2; unbiased random walks; no sources or sinks; infinite particle lifetimes; and sinusoidal initial particle distributions. Choose one simulation with a period of 200 and the other with a period of 50. Choose an appropriate bin size for the histograms. Run the simulation.

   a. What is the shape of the equilibrium distribution of particles?

   b. Which particle distribution approaches this equilibrium faster?

   c. Experiment with different frequencies and generalize your conclusion from part b.

Problem 2.7 *The purpose of this problem is to examine the factors that influence the equilibrium distribution of particles and the time to reach equilibrium.* This problem deals with a random walk in two regions. Make the size of Regions #1 and #2 each 50 steps wide and set the step size to 5 in each region. Let the initial distribution of particles in Region #1 be 50 particles distributed uniformly and

let the initial distribution of particles in Region #2 be empty. Run the simulation until equilibrium is reached. Plot the number of particles in Regions #1 and #2 as a function of step number.

a. Estimate the initial N_i and final N_f values of the number of particles in each region and explain these values.

b. Define the "equilibrium time" τ as the time taken for the number of particles to change from $0.1|N_f - N_i|$ to $0.9|N_f - N_i|$ and estimate the value of τ for both Regions #1 and #2.

c. The region size affects the time course of particle number in each region. Run simulations with the size of Regions #1 and #2 set to 25 steps and to 100 steps and measure N_i, N_f, and τ for each value of region size. How do N_i, N_f, and τ depend upon regions size for the three sizes used in your simulations? Explain.

d. Assume that the probabilities for steps are unchanged, which other RW parameters will affect the time course of particle number in each region? Indicate how the parameter will affect the time course and test your suggestion by performing appropriate simulations and measuring N_i, N_f, and τ.

Chapter 3

MACROSCOPIC DIFFUSION PROCESSES
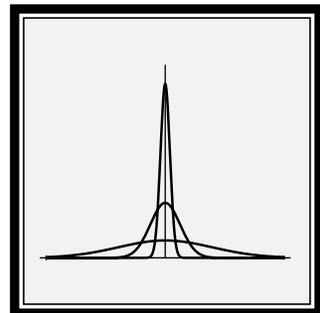
3.1   Introduction

3.1.1   Background

Diffusion is an important transport process in physical, chemical, and biological systems. Two theoretical models of diffusion capture complementary aspects of the phenomenon. The microscopic model describes the probabilistic behavior of a population of individual solute particles which execute statistically independent, but otherwise identical, random walks. Software dealing with such a microscopic model of diffusion is described in Chapter 2. The software described in this chapter involves macroscopic diffusion which describes the aggregate behavior of large populations of solute particles.

3.1.2   Macroscopic model of diffusion

In 1855, Adolph Fick proposed a macroscopic model for passive diffusion (Fick, 1855). By analogy to Fourier's law of heat conduction and Ohm's law for electrical conduction, Fick proposed that the flux of solute at a point in space is proportional to the concentration gradient at that point. Mathematically, this relation is expressed by *Fick's first law,* which, in its one-dimensional form, is as follows:

$$\phi = -D\frac{\partial c}{\partial x},\tag{3.1}$$

where $c(x,t)$ is the solute concentration and $\phi(x,t)$ is the solute flux. The constant of proportionality, D, is known as the *diffusion coefficient.* Conservation of solute (in the absence of sources or sinks or of chemical reactions that create or destroy solute) results in a second relation between concentration and flux, the *continuity equation*:

$$\frac{\partial \phi}{\partial x} = -\frac{\partial c}{\partial t}.\tag{3.2}$$

Differentiating both sides of Equation 3.1 with respect to x and using the continuity equation yields Fick's second law, or the *one-dimensional diffusion equation*:

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2}\tag{3.3}$$

In principle, any one-dimensional diffusion process can be modelled by solving this equation subject to the appropriate initial conditions and boundary conditions.

We next consider the same problem but with two changes: (1) a body force acts uniformly on the solute so that it drifts (migrates) and (2) a chemical reaction occurs that removes solute. The solute flux, $\phi(x,t)$, is now the sum of the diffusive flux, given by Fick's law, and the flux due to drift. Thus, in this case,

$$\phi = -D\frac{\partial c}{\partial x} + vc,\tag{3.4}$$

where v is the *drift velocity* of the solute; i.e., the velocity of the solute in response to the body force in the absence of diffusion. If r is the reaction rate at which solute is removed, then the continuity equation becomes

$$\frac{\partial \phi}{\partial x} = -\frac{\partial c}{\partial t} - rc. \tag{3.5}$$

Combining these expressions yields a modified diffusion equation:

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2} - v\frac{\partial c}{\partial x} - rc. \tag{3.6}$$

Thus, solute drift and a chemical reaction that removes solute contribute additional terms to the diffusion equation.

If the initial concentration, $c(x, 0)$ is known in some region of space and if the relation between the concentration and flux is known at each boundary of this region, then a unique solution exists for $t > 0$. Given an initial concentration profile, $c(x, 0)$, Fick's laws govern the unique evolution of the concentration profile for all later times.

3.1.3 Overview of software

While macroscopic diffusion has been successfully modelled by Fick's equations, studying the equations alone provides only limited insight into the behavior of diffusion processes. The goal of this software is to help users gain intuition about diffusion in one dimension. The solutions to the diffusion equation are functions of two variables: one spatial variable, x, and time, t. These solutions can be plotted either as a function of time at a particular point in space or as a function of the spatial variable at a particular instant in time. Such isolated snapshots, however, do not fully convey the dynamic behavior of diffusion processes. With this software, the user can view the temporal evolution of the solute concentration and solute flux as a function of position. Thus, the software enables the user to gain intuition about the dynamic behavior of macroscopic diffusion processes.

Furthermore, the software provides an opportunity for users to experiment with the macroscopic diffusion processes. For example, the user can explore how changing the diffusion coefficient affects the time-course of the simulation. The user can also specify transparent or reflecting boundaries and can explore how a body force acting on the solute or a chemical reaction that removes solute influence the diffusion process. Thus, the software can be used as a tool for investigating how various parameters and boundary conditions influence diffusion processes.

Five options for initial concentration profiles, corresponding to five characteristic diffusion problems, are implemented in the software (Figure 3.1). Four of these problems involve diffusion from a specified initial concentration profile
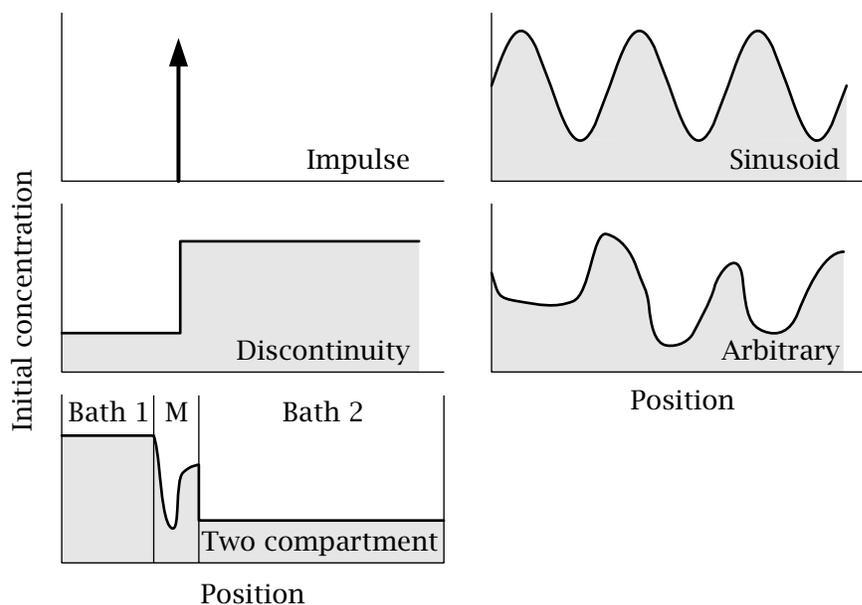
Figure 3.1: Schematic representation of the five classes of initial concentration profiles simulated by the software. M designates the membrane.

in a one-dimensional region of space (which may be infinite in extent), while the fifth involves diffusion between two finite, well-mixed compartments separated by a membrane.

One-dimensional diffusion

In one-dimensional diffusion, the initial concentration can be chosen to be one of four options: impulses, sinusoids, discontinuities, and an arbitrary spatial distributions. An impulse of solute concentration is a mathematical idealization corresponding to a finite amount of solute initially concentrated at a single point in space. In the impulses option, the user can specify the magnitude and position of up to four impulses. This option illustrates diffusion from point sources. The sinusoids option illustrate the spatial frequency response of a system governed by the diffusion equation, specifically, the low-pass spatial filtering effect of diffusion. By superimposing up to four sinusoids, the user can observe the dependence of the space-time evolution of concentration on its spatial frequency content. In the discontinuities option, the initial concentration profile may be piecewise-constant but contain discontinuities. The user can specify the magnitude and location of up to four such discontinuities. For example, this allows simulation of the diffusion that results from an initial rectangular pulse of solute concentration. The user can also explore the response for more complicated initial conditions which can be specified by drawing an arbitrary initial concentration on the monitor screen with a mouse.

For these different initial distributions, the user can specify the boundary type (transparent or reflecting), the value of the drift velocity, and the chemical reaction rate at which solute is removed.

Two-compartment diffusion through a membrane

The baths are assumed to be well-stirred, so that the concentration within each bath varies with time but not with position in the bath. The total amount of solute in the system is assumed to be finite. The user can specify the width of each bath and of the membrane, the initial concentration of each bath, and an arbitrary initial concentration profile in the membrane.

3.2 Methods of Solution

For several of the diffusion problems implemented in the software, analytic solutions of the modified diffusion equation (Equation 3.6) are available in a form that makes computation efficient. Such solutions have been used wherever possible. The remainder of the solutions were obtained by numerical methods. The methods are described in some detail elsewhere (Berkenblit, 1990); only a brief summary of the methods is given here.

The method used to obtain solutions depends upon the type of boundary condition. Two types of boundary conditions for the diffusion problems can be selected by the user: *transparent* or *reflecting* boundary conditions. If transparent boundaries are specified, then the region over which diffusion occurs is infinite; the spatial limits of the region over which diffusion is displayed are set by the user. At a reflecting boundary, on the other hand, the solute flux is constrained to be zero, since no solute particles can cross such a boundary.

3.2.1 Analytic solutions

Analytic solutions were used to compute the response for all one-dimensional diffusion problems with transparent boundaries. These are described in this section.

Impulse response — Green's function

No drift, no chemical reaction. First, we consider the case when the drift velocity is zero and the chemical reaction rate is zero. Let the initial concentration be a unit impulse located at position $x = \xi$ and delivered at time $t = 0$,

$$c(x, 0) = \delta(x - \xi). \tag{3.7}$$

The solution of the diffusion equation (Equation 3.3), called the *Green's function* $G(x, t; \xi, 0)$, is

$$c(x, t) = G(x, t; \xi, 0) = \frac{1}{\sqrt{4\pi D t}} e^{-(x-\xi)^2/4Dt}. \tag{3.8}$$

Thus, the concentration profile is a *Gaussian* function of the spatial variable. The standard deviation of the concentration, $\sqrt{2Dt}$, increases with increasing time, but the total area of the concentration remains constant because solute is conserved.

Drift but no chemical reaction.   We next consider the same problem but with a non-zero drift velocity. In this case, the concentration satisfies the modified diffusion equation, Equation 3.6, with $r = 0$:

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} - v \frac{\partial c}{\partial x}, \tag{3.9}$$

with the initial condition

$$c(x, 0) = \delta(x - \xi). \tag{3.10}$$

The solution can be shown to be

$$c(x, t) = \frac{1}{\sqrt{4\pi D t}} e^{-(x-\xi-vt)^2/4Dt} = G(x - vt, t; \xi, 0). \tag{3.11}$$

Thus, the response in this case is a Gaussian function, as before, but the entire profile drifts in the positive x-direction with drift velocity v.

Drift and chemical reaction.   With a non-zero chemical rate, r, and drift velocity, v, we need to solve Equation 3.6 which has a solution

$$c(x, t) = e^{-rt} G(x - vt, t; \xi, 0). \tag{3.12}$$

In general, if $c'(x, t)$ is the solution to Equation 3.3, with a specified initial concentration profile and transparent boundaries, then the solution to the modified equation, 3.6, satisfying the same initial conditions, is given by

$$c(x, t) = e^{-rt} c'(x - vt, t). \tag{3.13}$$

Sinusoid

First we consider the solution to Equation 3.3 with the initial condition

$$c(x, 0) = A (1 + \sin(\omega x + \theta)) \tag{3.14}$$

for some arbitrary constant amplitude, A, spatial radian frequency, ω, and phase angle, θ. The solution for $t > 0$ is given by

$$c(x,t) = A\left(1 + e^{-D\omega^2 t}\sin(\omega x + \theta)\right),\tag{3.15}$$

which can be verified by substituting this expression into Equation 3.3. Thus, an initially sinusoidal concentration profile remains sinusoidal for all later time, but the amplitude decreases exponentially with time. The rate of attenuation is proportional to the square of the spatial frequency.

With drift and a chemical reaction between solute and solvent, the solution is

$$c(x,t) = Ae^{-rt}\left(1 + e^{-D\omega^2 t}\sin\left(\omega(x - vt) + \theta\right)\right),\tag{3.16}$$

where v is the drift velocity and r is the reaction rate.

Discontinuous initial profile

We next consider the case of an initial profile that is piecewise-constant but contains a jump discontinuity at $x = \xi$. The initial condition is that

$$c(x,0) = \begin{cases} c_L, & x < \xi \\ c_R, & x > \xi \end{cases},\tag{3.17}$$

which can also be written as

$$c(x,0) = c_L + (c_R - c_L)\int_{\xi}^{\infty}\delta(x - x')\,dx'.\tag{3.18}$$

For $t > 0$, we can integrate the Green's function defined by Equation 3.8 to obtain the solution:

$$\begin{aligned} c(x,t) &= c_L + (c_R - c_L)\int_{\xi}^{\infty}G(x,t;x',0)\,dx' \\ &= c_L + (c_R - c_L)\int_{\xi}^{\infty}\frac{1}{\sqrt{4\pi Dt}}e^{-(x-x')^2/4Dt}\,dx' \\ &= c_L + (c_R - c_L)\left(1 - \frac{1}{2}\operatorname{erfc}\left(\frac{x - \xi}{2\sqrt{Dt}}\right)\right), \end{aligned}\tag{3.19}$$

where the *complementary error function*, $\operatorname{erfc}(x)$, is defined by the integral

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_{x}^{\infty}e^{-t^2}\,dt.\tag{3.20}$$

With a drift velocity, v, and reaction rate, r, the solution is

$$c(x,t) = c_L e^{-rt} + (c_R - c_L)e^{-rt}\left(1 - \frac{1}{2}\operatorname{erfc}\left(\frac{x - vt - \xi}{2\sqrt{Dt}}\right)\right).\tag{3.21}$$

3.2.2   Numerical Solutions

One-dimensional problems with reflecting boundaries and the two-compartment problem are solved using numerical methods. To solve a diffusion problem numerically over some specified spatial region and some specified time interval, the position and time variables are discretized. Denoting the time increment by Δt and the position increment by Δx,   we define

$$u_i^j = c(x_i, t_j), \tag{3.22}$$

where

$$
\begin{aligned}
x_i &= x_0 + i\Delta x, \quad i = 1, \dots, N \\
t_j &= t_0 + j\Delta t, \quad\;\; j = 1, \dots, M.
\end{aligned}
\tag{3.23}
$$

A general approach to solving partial differential equations numerically is to replace the partial derivatives by finite-difference approximations. The finite-difference expressions involve the u_i^j terms defined above and result in sets of algebraic equations that can be solved by numerical methods for solving matrix equations. In the software, the equation with no drift term is solved by the implicit *Crank-Nicolson* algorithm, while for problems with a non-zero drift term, this algorithm is combined with the explicit *Two-Step Lax-Wendroff* method (Press et al., 1988; Gerald and Wheatley, 1989). When a non-zero chemical reaction rate, r, is specified, the computed concentration is multiplied by the attenuation factor, e^{-rt}. The Crank-Nicolson and Lax-Wendroff formulas involve the parameters $\alpha = D\Delta t/(\Delta x)^2$ and $\beta = v\Delta t/\Delta x$. The numerical methods give stable solutions provided $\alpha < 0.5$ and $|\beta| \le 1$. For the two compartments option, the boundary conditions at the membrane solution interface is that the concentration in the membrane at the interface is equal to the bath concentration.

3.2.3   Summary

A variety of techniques are used to find analytic solutions to one-dimensional diffusion problems with transparent boundaries. The specific methods used in the software to solve one-dimensional problems depend on the initial conditions and are summarized in Table 3.1. With two reflecting boundaries and in the absence of drift, the analytic solutions consist of infinite series. Thus it is cumbersome to compute these analytic solutions. With two reflecting boundaries and with drift, analytic solutions are in general not available. Therefore, we have chosen to solve these problems numerically using the Crank-Nicolson and Lax-Wendroff methods. The latter methods are also used to compute the solutions to the two-compartment problem.

Option	Boundary Type	Computational Method
Impulse	Transparent	Analytic solution
	Reflecting	Crank-Nicolson
Sinusoid	Transparent	Analytic solution
Discontinuity	Transparent	Analytic solution
	Reflecting	Crank-Nicolson
Arbitrary	Reflecting	Crank-Nicolson
Two-Compartment		Crank-Nicolson

Table 3.1: Summary of computational methods.

Figure 3.2: The MD Control figure.

3.3 User's Guide To The Software

When this software package is selected, 4 figures are displayed (in addition to MATLAB's command window) — *MD Control*, *MD Initial concentration profile*, *MD Plots vs. position*, and *MD Plots vs. time*. A typical session with the software consists of selecting one of five initial concentration profile options, selecting the values of parameters, executing a simulation, and viewing the simulation results.

3.3.1 MD Control

The MD Control figure controls the macroscopic diffusion software and is shown in Figure 3.2. The part of the MD Control figure below the menubar is divided into three panels. The left panel controls the display of the initial concentration and the numerical parameters of the simulation abbreviated as *numerics*. The middle panel controls calculation of the simulation, and allows the simulation to be started, paused, continued, or reset. The user can select the number of steps in time for computing the simulation results. The interpoint time is adjustable in the numerics figures (described later). The *Time* is the elapsed simulation time since the simulation was started. Clicking on *1-step* results in the computation of one step of the simulation. The simulation results and parameters can be saved to a file and read from a file. Clicking on *Save* saves the follow-

ing items in a file: all the initial profile parameters, the numerical parameters, concentration and flux versus position and versus time. Clicking on *Open* reads these same quantities from a file. The right panel (the Simulation results panel) controls the display of simulation results. Clicking on a selection box displays the appropriate figure; clicking again hides it.

3.3.2   MD Initial concentration profile

The *MD Initial concentration profile* figure allows control of the initial concentration and all the parameters of the diffusion process. The four one-dimensional diffusion problems and the two-compartment problem are implemented in the software as five *options* called: Impulses (Figures 3.3), Sinusoids (Figures 3.5), Discontinuities (Figures 3.6), Arbitrary (Figures 3.7), and Two Compartments (Figures 3.8). To change the profile type, the user clicks on the profile push-button in the *MD Initial concentration profile* figure. Each pushbutton has an icon suggesting the initial profile type. Parameters can be viewed by clicking on *Parameters* in the same figure which results in the display of the appropriate parameters figure. Parameters can be changed by typing in the text edit box. To activate the new parameters, the user must click on *Update* either in the *MD Parameters* figure or in the *MD Initial concentration profile* figure; these actions are equivalent. Information about the initial profile and parameters (alone) can be saved in a file by clicking on *Save* or read from a file by clicking on *Open*. These actions can be done either in the *MD Parameters* figure or in the *MD Initial concentration profile* figure; they are equivalent.

Features of initial profiles and parameters common to all options

Variables.   The initial concentration is plotted in the *MD Initial concentration profile* figure (Figures 3.3, 3.5, 3.6, 3.7, and 3.8). Check boxes in the upper left-hand and the upper right-hand corners allow display of the initial concentration and/or the initial flux.

Axes scales.   An axis scale can be changed by clicking on its label to display the *Axis-scale figure* Figure 3.4. The new axis limits can be typed into the edit boxes to change the axis.

Markers.   There are two markers that are indicated by two green triangles on the abscissa of both the *MD Initial concentration profile* figure and the *MD Plots vs. position* figure. These markers indicated positions at which the concentration and flux can be displayed as a function of time in the *MD Plots vs. time* figure. The positions of these markers can be controlled numerically or graphically. Numerical control is achieved by typing into the appropriate text edit box in the parameters figure (Figures 3.3, 3.5, 3.6, 3.7, and 3.8). Graphical control is achieved
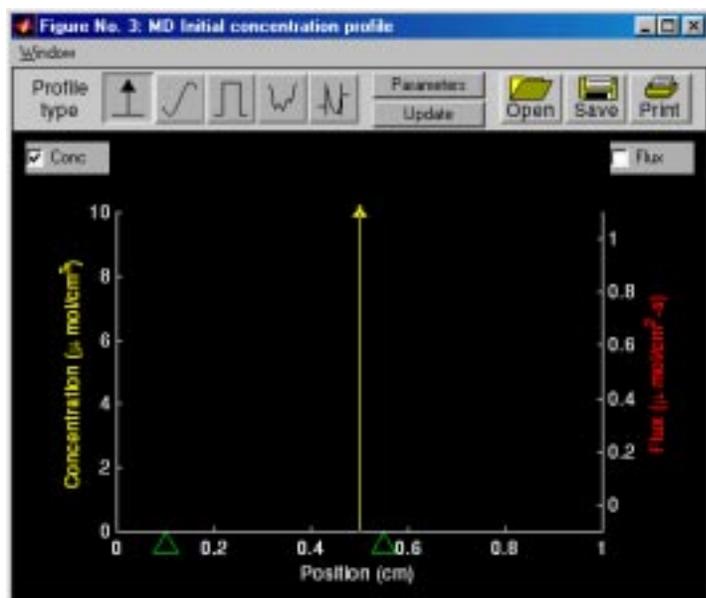
Figure 3.3: Initial concentration profile for impulses; MD Initial concentration profile figure (above), MD Impulse profile parameters figure for impulses (below). A single impulse located at position 0.5 cm of area 10 is specified.
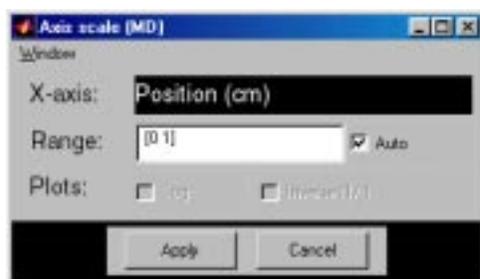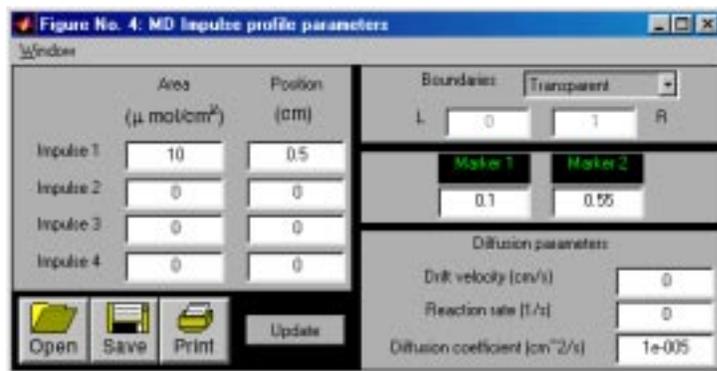
Figure 3.4: MD Axis scale figure when the abscissa is chosen in the MD Initial concentration profile figure.

by clicking just below the abscissa axis in the *MD Initial concentration profile* figure. The left marker is controlled by clicking the left button on a two-button mouse or by clicking the button on a single-button mouse. The right marker is controlled by clicking the right button on a two-button (or three-button) mouse or by shift-clicking the button on a single-button mouse.

Parameters.   In the MD parameters figure for each option, the user can select a parameter by clicking on it and then modify that parameter by typing the new value in the text edit box. This modification is terminated by a carriage return or by selection of another parameter. One panel displays the *Diffusion parameters* — diffusion coefficient, drift velocity, and chemical reaction rate — which are set independently for each option. For the two compartments option these diffusion parameters apply to the membrane and not to the baths.

   After all the parameters in the figure have been changed, the user must press *Update* before the new parameters are reflected in both the *MD Initial concentration profile* figure and the *MD Plots vs. position* figure. All the parameters for each option can be saved in a file and read from a file.

Descriptions of the five options

For each option, the user can modify the values of parameters, change all graph parameters, and run a simulation.

Impulses.   When the software is initiated, *Impulses* is the default initial concentration profile and the *MD Initial concentration profile figure* is shown in Figure 3.3. The areas and positions of 4 impulses can be specified by the user as shown in the parameters figure so that the initial concentration has the form

$$A_1\delta(x - x_1) + A_2\delta(x - x_2) + A_3\delta(x - x_3) + A_4\delta(x - x_4),$$

where $\delta(x)$ is the unit impulse located at $x = 0$, A_j and x_j are the area and position of the jth impulse. The positions of each impulse can be set only if its area is non-zero. Since the initial flux is the spatial derivative of the initial concentration, if the concentration contains an impulse $\delta(x - x_o)$ at $x = x_o$ then the flux contains a doublet $\dot{\delta}(x - x_o)$ at $x = x_o$ which is signified by a two-headed arrow. Either transparent boundaries or reflecting boundaries can be chosen and the locations of right (R) and left (L) reflecting boundaries can be specified.

Sinusoid.   In the *Sinusoid* option, the amplitudes, spatial frequencies, and phase angles of four sinusoids can be specified (Figure 3.5) so that the initial concentration has the form

$$A_1(1 + \sin(2\pi f_1 x + \theta_1)) + A_2(1 + \sin(2\pi f_2 x + \theta_2)) +$$
$$A_3(1 + \sin(2\pi f_3 x + \theta_3)) + A_4(1 + \sin(2\pi f_4 x + \theta_4)),$$
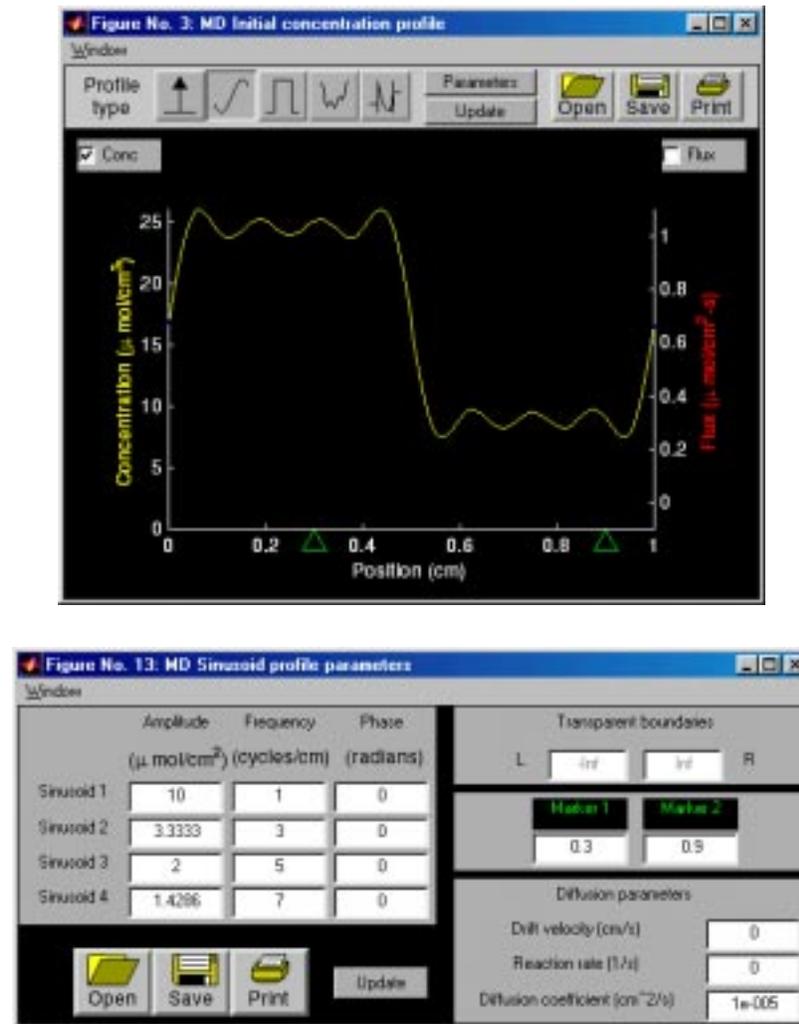
Figure 3.5: Initial concentration profile for sinusoids; MD Initial concentration profile figure (above), MD Sinusoid profile parameters figure (below). The spatial frequencies and amplitudes of four sinusoids were chosen to correspond to the first four terms in the Fourier series of a square wave.

where A_j, f_j, and θ_j are the amplitude, spatial frequency, and phase of the jth sinusoid. The spatial frequency and phase of a sinusoid can be set only if its amplitude is non-zero. Only transparent boundaries are available for the sinusoid option.

Discontinuity. In the *Discontinuity* option, the magnitude and position of four discontinuities can be specified (Figure 3.6) so that the initial concentration has the form

$$A_1 u(x - x_1) + A_2 u(x - x_2) + A_3 u(x - x_3) + A_4 u(x - x_4),$$
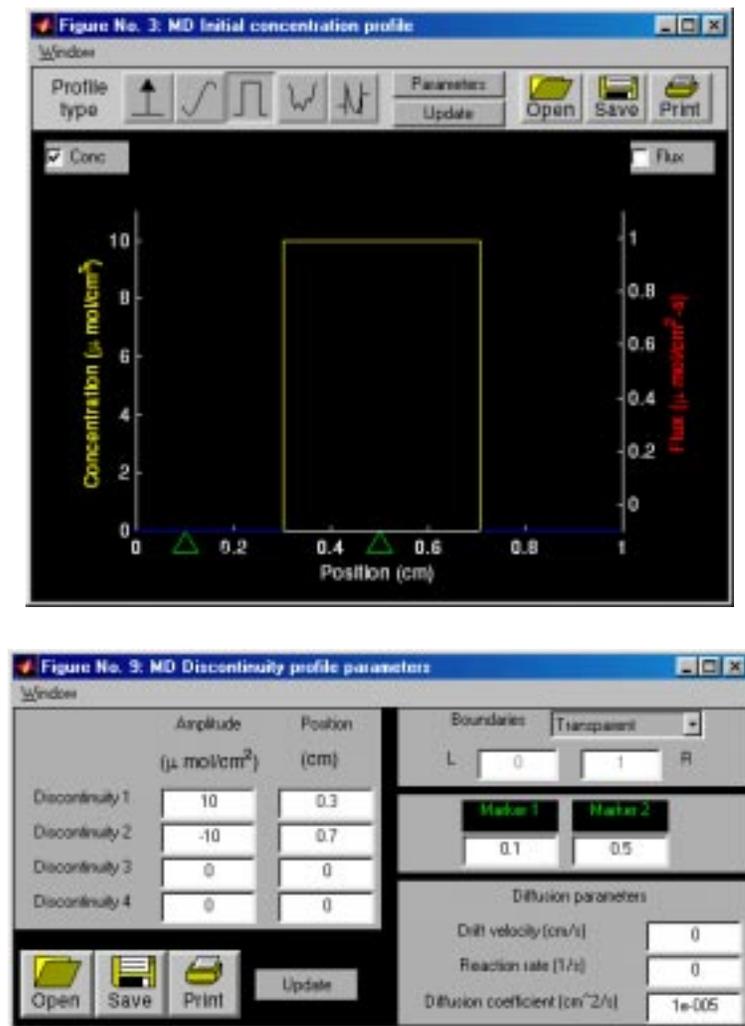
Figure 3.6: Initial concentration profile for discontinuities; MD Initial concentration profile figure (above), MD Discontinuity profile parameters figure (below). Two discontinuities were used to generate an initial concentration profile that is a rectangular pulse.

where $u(x)$ is the unit step function defined as

$$u(x) = \begin{cases} 1 & \text{if } x > 0, \\ 0 & \text{if } x < 0, \end{cases}$$

and A_j and x_j are the amplitude and position of the jth step. The position of a step can be set only if its magnitude is non-zero. The boundary type (transparent or reflecting) is selectable, and, if the boundaries are reflecting, the positions of the two boundaries can be set.

Arbitrary. In the *Arbitrary option*, the initial profile is drawn on the monitor screen by moving the mouse cursor. The details are described in Section 3.3.5. Only reflecting boundary conditions are available for this option, and the positions of the two boundaries are specifiable by the user (Figure 3.7).
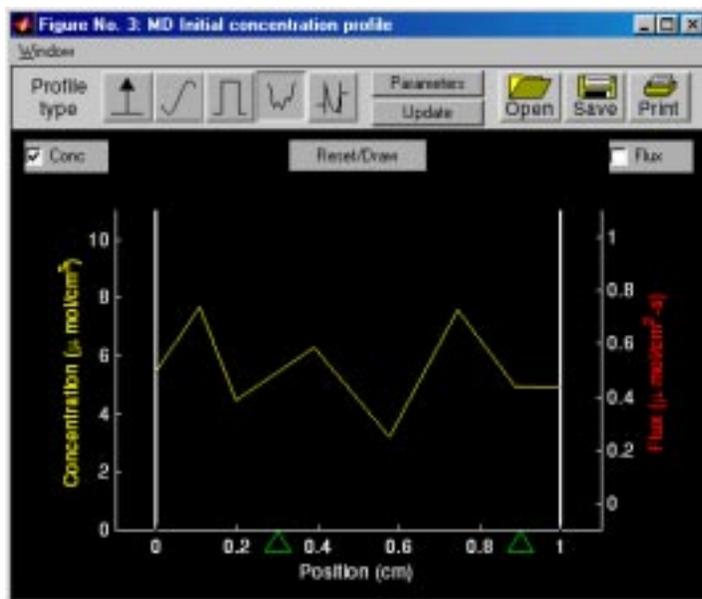
Figure 3.7: Initial concentration using the arbitrary option; MD Initial concentration profile figure (above), MD Arbitrary profile parameters figure (below).

Two-compartment.   In the *Two compartment* option, specification can be made for the width and the initial concentration in each bath, the membrane width, and the initial membrane concentration. The initial membrane concentration profile is drawn with a mouse cursor (as described in Section 3.3.5). The user can also choose how the two baths and the membrane are to be displayed. Selecting the *View bath* option (Figure 3.8) displays the membrane and both baths to scale. The left boundary of the membrane is located at $x = 0$. Hence, the left bath is located for $x < 0$ and if the width of the membrane is W, the right bath is located at $x > W$. Selecting the *View membrane* option displays the membrane and a small portion of each bath on each side.

3.3.3   Numerics

Clicking on *Analytic parameters* or *Numeric parameters* in the *MD Control* figure displays the appropriate numerics figure (Figure 3.9). The numerics figures control the spatial and temporal resolution of the simulation.   The *MD Analytic parameters* figure deals with those computations that make use of analytic solutions of the diffusion equation (i.e., all one-dimensional diffusion problems with transparent boundaries). For this case, the user can select the number of spatial points at which the solution is computed. The larger the number of points, the higher the spatial resolution of the computation and the longer it will take to complete the computation. The user can also specify the temporal resolution of the computation by selecting the time between samples (integration time).

The *MD Numeric parameters* figure deals with those computations that use numerical integration of the diffusion equation (i.e., all one-dimensional diffusion problems with reflecting boundaries and the two-component diffusion problem). The user can choose to select the number of points in space at which the solution is computed. The user has a choice in how the temporal resolution is to be determined. The numerical method is stable provided an inequality that relates the temporal and spatial resolutions is satisfied. If *Link* is selected, the simulation determines the integration time to satisfy the inequality and to guarantee a stable integration method which results in an accurate solution. If this choice is not selected, the user can choose the temporal resolution (integration time) and spatial resolution independently. However, with this choice there is no guarantee that the solution will be accurate.

3.3.4   Simulation results figures

There are two figures that display the solute concentration and flux during a simulation. The *MD Plots vs. position* figure displays the concentration and/or the flux as a function of position at each instant in time during the simulation (Figure 3.10). The current simulation *Time* is displayed in the upper left part of the figure. Axes limits can be changed by clicking on the axes labels to dis-
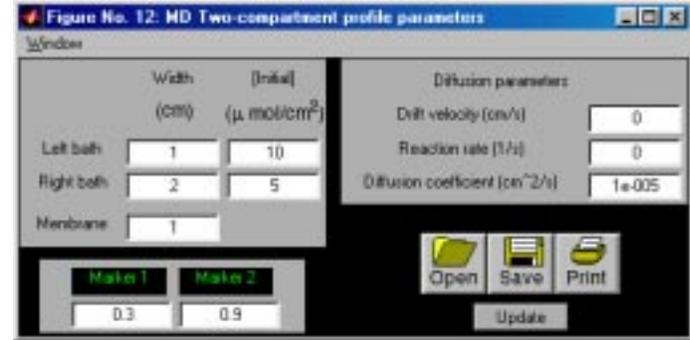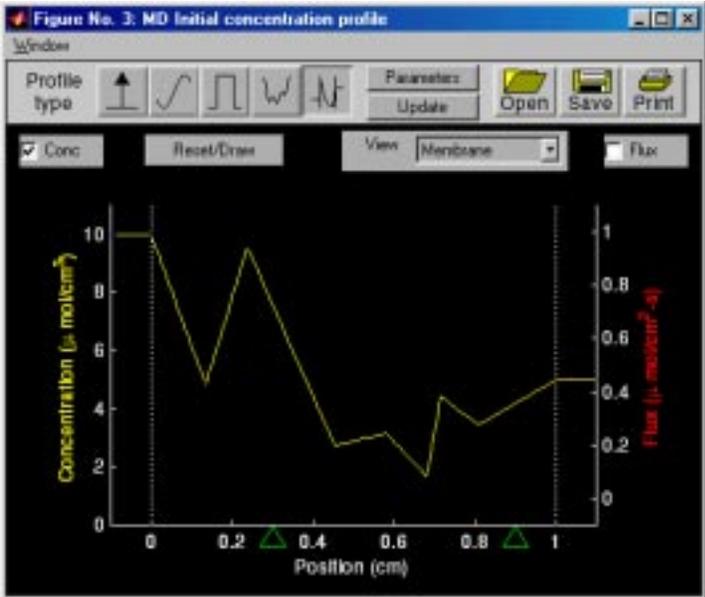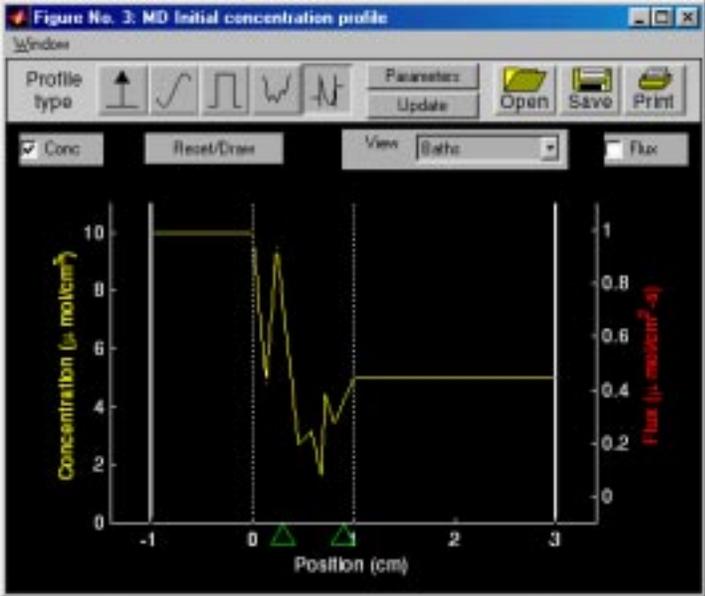
Figure 3.8: Initial concentration for the two compartments option; MD Initial concentration profile figure for the two compartments option for the bath view (top) and the membrane view (center), MD Two-compartment profile parameters figure (bottom). The bath view shows the left bath located between −1 and 0 cm, the membrane located between 0 and 1 cm, and right bath located between 1 and 3 cm.

Figure 3.9: The MD Analytic parameters and MD Numeric parameters figures.
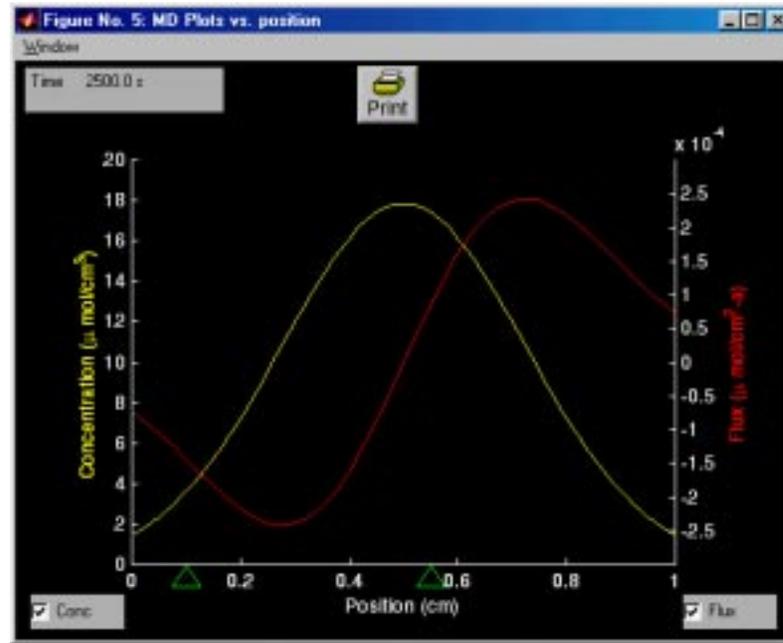
Figure 3.10: MD Plots vs. position figure.
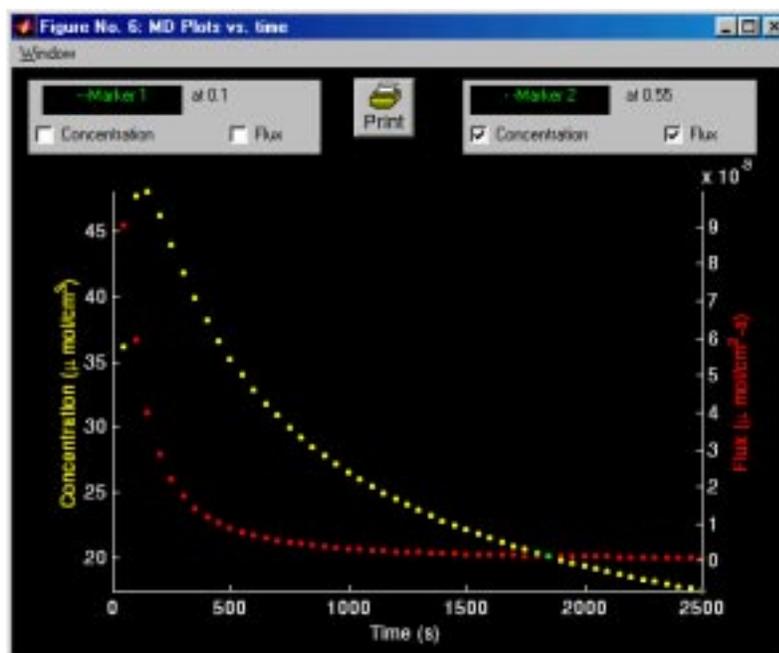
Figure 3.11: MD Plots vs. time figure.

play the *Axis-scale figure.* The two green triangular markers mirror the markers shown in the *MD Initial concentration profile* figure. With the *Two compartments* option, the user can also chose how the two baths and the membrane are to be displayed in the *MD Plots vs. position* figure. Selecting the *View membrane* option displays the membrane and a bit of each bath on each side. Selecting the *View baths* option displays the membrane and both baths to scale.

The *MD Plots vs. time* figure (Figure 3.11) displays the concentration and/or the flux as a function of time during the simulation for two positions selected by the markers found in both the *MD Initial concentration profile* figure and the *MD Plots vs. position* figure (Figure 3.10). The locations of the markers are indicated at the top of the figure. Axes limits can be changed by clicking on the axes labels to display the *Axis-scale figure.*

3.3.5 Arbitrary concentration profiles

For the *MD Arbitrary* and *MD Two compartment* options, the user can draw an arbitrary initial concentration profile by clicking on the *Reset/Draw* button which will also erase an existing profile. Drawing a profile consists of generating a list of concentration values at a corresponding array of positions. The user selects these points by moving the mouse and clicking any mouse button. After the first point is selected by the user, the word *Incomplete* is displayed over the profile indicating that the profile is incomplete and does not span the horizontal

dimension of the plotting field. As the pointer moves within the plotting field in the *MD Initial concentration profile* figure, a line joins the last point which was selected to the current pointer position. When a button is pressed, the position and concentration that are added to the coordinate arrays depend on the pointer location:

1. If a mouse button is pressed within the graph axes and to the right of the previous point, then the coordinates of the pointer location are added to the list.

2. A mouse button pressed at a position that does not lie to the right of the previous point is ignored, in order to ensure that the user-drawn concentration profile is a single-valued function of position.

3. If a mouse button is pressed outside the boundaries of the graph (but still within *MD Initial concentration profile* figure), then the point that is added to the list is the point of intersection between the graph boundary and a line from the previous point to the current position. Thus, while a profile is being drawn, it is constrained to lie entirely within the boundaries of the graph.

4. In the arbitrary initial profile, the concentration at the left-hand point of the profile is set to a default value. The user can change the value of this point by pressing the mouse button for first point at the desired vertical location but to the left of the concentration axis. In the two components option, the concentration at the left-hand point of the profile is set equal to the concentration in the left bath (bath #1), and the concentration at the right-hand point of the profile is set equal to the concentration in the right bath (bath #2).

5. If the mouse button is pressed outside the *MD Initial concentration profile* figure, e.g., on another figure, then the profile is incomplete.

6. An incomplete profile can be completed by pressing *Update*. In the arbitrary initial profile, a horizontal line segment is drawn from the last chosen point to the right boundary. In the two-compartment option, the last chosen point is connected to the concentration in the right bath.

7. An incomplete profile can be completed by clicking the mouse button with the cursor located just to the right of the right boundary of the region. Item 3 (above) gives the rule that indicates the location of the right boundary point.

Clicking on *Update* after completion of the profile, activates the initial profile.

3.4 Problems

Problem 3.1 *The purpose of this problem is to explore the relation between molar solute concentration and molar solute flux.* The diffusion coefficient is $D = 10^{-5}$ cm^2/s.

    a. For an initial, spatial sinusoidal solute concentration with an amplitude of 10 μmol/cm^2 and a frequency of 1 cm^{-1}, determine the initial spatial distribution of the molar solute flux.

    b. Determine an initial spatial distribution of molar solute concentration such that the initial spatial distribution of molar flux is a constant of value 1 μmol/(cm$^2 \cdot$s).

Problem 3.2 *The purpose of this problem is to explore the space-time evolution of solute diffusion from a point source.* For all parts of this problem use a single impulse and transparent boundaries. Set the impulse strength to 100 moles/cm^2, place the impulse at position 0.5 cm, and set the drift velocity and the reaction rate to zero. Set the marker positions to 0.55 and 0.6 cm unless indicated otherwise. Set the diffusion coefficient to 10^{-5} cm^2/s and run the simulation for 1000 seconds. Obtain a plot of the concentration and flux on appropriate ordinate scales. You should obtain the spatial distribution of the concentration and the flux at 1000 seconds and both variables as a function of time at the two marker positions. Repeat these two steps for a diffusion coefficient of 4×10^{-5} cm^2/s.

    a. Briefly describe qualitatively the effect of the change in diffusion coefficient on the spatial distribution of concentration.

    b. For both values of the diffusion coefficient, determine the maximum amplitude of the concentration versus position and the width of the spatial distribution of concentration at an amplitude that is e^{-1} of its peak value. Explain the numerical values of all four measurements. Be brief and precise; state your assumptions explicitly.

    c. For the diffusion coefficient at 10^{-5} cm^2/s, examine the concentration versus time at the two positions 0.55 and 0.6 cm. Briefly describe, in a qualitatively manner, the differences in concentration versus time at the two positions.

    d. For the diffusion coefficient at 10^{-5} cm^2/s, measure the maximum concentration as a function of time at the two positions and the time of occurrence of this maximum. Explain the values of all four measurements. Be brief and precise; state your assumptions explicitly.

    e. Explain the shape of the spatial distribution of flux for a diffusion coefficient of 10^{-5} cm^2/s.

f. Explain the relation between the flux and concentration versus time at location 0.55 cm for a diffusion coefficient of 10^{-5} cm^2/s.

g. Rerun the simulation for a diffusion coefficient of 10^{-5} cm^2/s at positions 0.45 and 0.55 cm. Explain the differences in the flux as a function of time for these two locations.

Problem 3.3 *The purpose of this problem is to explore the space-time evolution of solute diffusion from a point source in the presence of drift.* Set the impulse strength to 100 moles/cm^2; place the impulse at position 0.5 cm; set the diffusion coefficient to 10^{-5} cm^2/s, and the reaction rate to zero. Set the marker positions to 0.75 and 0.9 cm unless indicated otherwise. The parts of this problem that deal with reflecting boundaries are solved numerically. To decrease the time taken for these computations, consider decreasing the spatial resolution of the computation to 10 points.

a. Set the boundary conditions to transparent and the drift velocity to zero and run the simulation for 1000 s. Now set the drift velocity to 10^{-4} cm/s and run the simulation again for 1000 s. Describe the effect of the drift on the spatial distribution of concentration. Account quantitatively for the position of the peak of the spatial distribution at 1000 s.

b. Set the boundary conditions to reflecting and the drift velocity to zero. Run the simulation until the spatial distribution is no longer a function of time. This is the equilibrium distribution. What is the spatial distribution at equilibrium? Explain your answer.

c. Set the boundary conditions to reflecting, the drift velocity to 10^{-4} cm/s, and run the simulation. Describe the effect of drift on the spatial distribution of concentration. Illustrate your description with printed copies of the distribution at characteristic times.

d. With the boundary conditions still set to reflecting and the drift velocity 10^{-4} cm/s, determine the equilibrium spatial distribution of concentration. Account quantitatively for this distribution.

e. It is known that the density of air decreases exponentially with distance above the earth's surface. Explain this phenomenon, taking the above simulation results into account.

Problem 3.4 *The purpose of this problem is to explore some of the assumptions that underlie two-compartment diffusion.* As shown elsewhere (Weiss, 1996a, Chapter 3), two-compartment diffusion is based on 4 assumptions:

1. The two compartments are well-mixed so that the concentrations of solute n are uniform and have values at time t of $c_n^1(t)$ and $c_n^2(t)$.

2. Solute particles are conserved, e.g., there is no chemical reaction present that either creates or destroys particles.

3. The membrane is sufficiently thin and the number of solute particles contained in the membrane at any time is negligibly small.

4. The membrane is sufficiently thin that at each instant in time the concentration profile in the membrane is in steady state.

This problem concerns the conditions for the validity of assumptions 3 and 4. Specifically, you will explore the effect of bath dimensions on two-compartment diffusion without making these two assumptions.

In all parts of the problem, use the *Two compartments* option of the software. Set the membrane width to 0.01 cm, and the concentration of bath #1 to 70 and bath #2 to 10 mol/cm^3. Leave the drift velocity and reaction rate at 0 and the diffusion coefficient at 10^{-5} cm^2/s. Set the parameters of the plot of concentration versus position so that the membrane and a little of each bath are visible on the screen — select membrane view. Note that the left edge of the membrane is at a position of 0 cm. Draw some initial concentration in the membrane. Make sure all ordinate scales on all plots are 0 to 100 mol/cm^3. Keep these parameters fixed throughout this problem. To reduce the time taken for the computation reduce the spatial resolution in the membrane to 10 points in space. The software will determine the time increment (DELTA_T). Initially set the number of steps so that the end time is 5 s.

For each of the pairs of bath widths — Bath #1 = 1 cm, Bath #2 = 1 cm; Bath #1 = 0.1 cm, Bath #2 = 0.1 cm; Bath #1 = 0.01 cm, Bath #2 = 0.01 cm; Bath #1 = 0.01 cm, Bath #2 = 0.05 cm; Bath #1 = 0.001 cm, Bath #2 = 0.001 cm; — answer the following questions.

a. Assess the validity of assumption 4.

   i. Make rough estimates of both the steady-state (τ_{ss}) and equilibrium (τ_{eq}) time constants from the computations.

   ii. Estimate the same two time constants based on theoretical considerations (Weiss, 1996a, Chapter 3).

   iii. What is your conclusion based on your computations and your estimates of time constants?

b. Assess the validity of assumption 3.

   i. Before you do the computation, make an estimate of the final concentration in each bath. Then do the computation, and check your initial estimates against the computed values

   ii. If they differ, explain the basis of the difference.

    iii. How good is the assumption that the quantity of solute in the membrane is negligible? If you decide that the quantity of solute in the membrane is not negligible, design a simulation experiment to test your conclusion.

e. Are the bath concentrations exponential functions of time? Explain.

Problem 3.5 *The purpose of this problem is to explore the successes and pitfalls of the method of images for solving diffusion problems in the presence of a reflecting boundary.* Some problems in diffusion with a reflecting boundary can be solved by using the solution for a transparent boundary and the method of images to match the reflecting boundary condition.

In all parts of this problem, set the drift velocity to zero, the reaction rate to zero, and the diffusion coefficient to 10^{-5} cm^2/s unless state otherwise. The parts of this problem that deal with reflecting boundaries are solved numerically. To decrease the time taken for these computations, consider decreasing the spatial resolution of the computation to 10 points.

a. Use reflecting boundaries, and an initial impulse of concentration of strength 100 moles/cm^2 located at 0.2 cm. Run the simulation for 1000 s and obtain a print of the spatial distribution of concentration at time 1000 s and the concentration versus time at locations 0 and 0.3 cm.

b. At a reflecting boundary, the flux must be zero. Change the boundary condition to transparent and determine the parameters of two impulses: one to match the left boundary condition and another to independently match the right boundary condition. Run the simulation again and compare with the results in part a. Does this "method of images" work exactly? Explain.

Problem 3.6 *The purpose of this problem is to explore the space-time evolution of diffusion from an initial sinusoidal concentration profile.* Diffusion from an initial sinusoidal concentration distribution gives important insights into the space-time evolution of diffusion processes. In all parts of this problem, use a drift velocity of zero, a reaction rate of zero, and a diffusion coefficient of 10^{-5} cm^2/s unless state otherwise.

a. Use an initial concentration that is sinusoidal with amplitude 50 moles/cm^2 and a spatial frequency of 1 cycle/cm. Run the simulation for 1000 s and print the spatial distribution of concentration at 1000 s and the concentration versus time at 0.25 and 0.75 cm. Repeat this procedure for sinusoids with the same amplitudes but the following spatial frequencies: 3, 5, and 7 cycles/cm. Summarize your results both qualitatively (in words) and then quantitatively (with suitable calculations).

Amplitude (μmol/cm^2)	Position (cm)
10	-100
-10	0.4
10	0.6
-10	100

Table 3.2: Concentration well — Problem 3.7.

b. Construct a periodic waveform from four sinusoids with the following amplitudes (moles/cm^2) and spatial frequencies (cycles/cm): 105, 1; 35, 3; 21, 5; 15, 7. Run the simulation and observe the spatial distribution of concentration. Summarize your results and relate them to results of part a.

c. Switch to the arbitrary initial distribution option and set the boundary conditions to reflecting. Using the mouse draw an arbitrary, preferably jagged, initial profile. This part of this problem uses a numerical solution method that is time consuming. To decrease the time taken for these computations, consider decreasing the spatial resolution of the computation to 10 points. Run the simulation and watch the spatial distribution of concentration change. Summarize your results. What is the effect of diffusion on the spatial distribution?

Problem 3.7 *The purpose of this problem is to explore the time it takes for diffusion to take place in a concentration well.* We examine the following two estimates of diffusion time.

1. A crude estimate of the time t_1 it takes for 1/2 the particles to diffuse over a distance x is $t_1 = x^2/D$ (Weiss, 1996a, Sections 3.2 and 3.3).

2. The time for equilibration of a slab of dimensions $2a$ so that the average concentration in the slab is half its final value corresponds to the condition $Dt_2/a^2 = 0.21$ (Weiss, 1996a, Figure 3.18).

Define a concentration well as shown in Table 3.2 and let the diffusion coefficient be $D = 10^{-5}$ cm^2/s.

a. Run the software and measure the concentration in the center of the well as a function of time. Estimate the time t_3 it takes for this concentration to reach 1/2 its final value.

b. Quantitatively compare the estimates t_1, t_2, and t_3.

c. Briefly discuss the bases of the differences you found between these three estimates.

c. Suggest a new method to estimate an equilibration time that will be close to t_2 using the concentration well defined in Table 3.2. Test your suggestion by performing the computations and making the comparison.

d. Why are there discontinuities in the well at ± 100? What effect do they have on the estimates of equilibration time.

Chapter 4

CARRIER-MEDIATED TRANSPORT
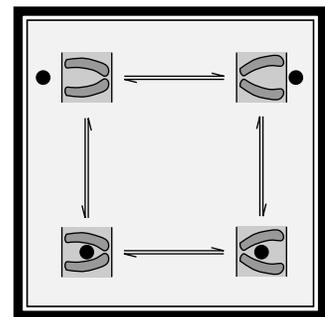
4.1   Introduction

Transport of many solutes including important metabolites (e.g. monosaccharides, amino acids, mononucleotides, phosphates, uric acid, etc.) through cellular membranes is accomplished by membrane-bound carrier molecules that combine with the solute molecule on one face of the membrane, then translocate in the membrane and uncombine at the other face. Thus, transport involves chemical reactions in which the solute binds and unbinds with a carrier. This type of transport is called *carrier-mediated transport*. Carrier-mediated transport has distinctive properties. For example, the relation of flux to concentration saturates, i.e., it does not obey Fick's law for membranes. In addition, different molecules compete for the carrier. For example, glucose and sorbose (two monosaccharides) compete for the sugar carrier. Hence, transport of one sugar can inhibit transport of another simply by occupying a site to which both can bind.

There are canonical models of carrier-mediated transport that capture important properties of the transport of metabolites through cellular membranes. It is important to understand these canonical models in order to understand how metabolites are transported across these membranes. Derivations of predictions of these models are not particularly difficult to follow; the individual steps are simple. However, the models typically result in messy algebraic expressions that relate flux to concentration and transport parameters. Thus, it is easy to get lost in algebraic manipulation as well as in a sea of parameters so that an intuitive grasp of the models becomes illusive. The intent of this software is to allow the user to develop intuition for these models.

4.2   Description Of The Models

Descriptions of carrier-mediated transport as well as models of such transport processes can be found elsewhere (Stein, 1986; Weiss, 1996a). Here we consider three models and list both the assumptions and the important results. First, we consider the steady-state behavior of a simple, four-state carrier that binds one solute Then, we consider the steady-state behavior of a simple, six-state carrier that binds two solutes with different affinities. Finally, we consider both the transient and the steady-state behavior of a general, four-state carrier model that binds one solute.
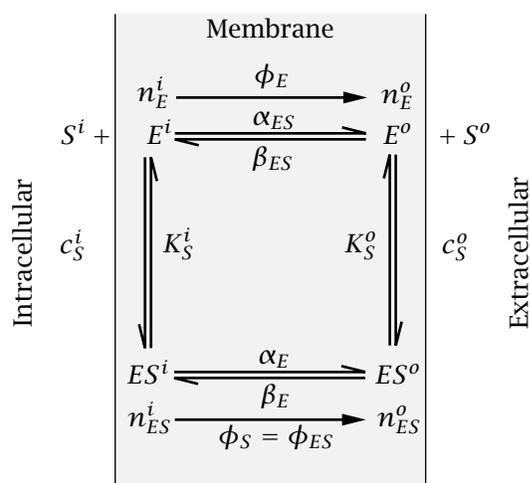
Figure 4.1: Kinetic diagram of the simple, four-state carrier that binds one solute.

4.2.1 Steady-state behavior of a simple, four-state carrier that binds one solute

Assume that the membrane contains N_{ET} moles of carrier per unit area of membrane.[1] Each of these carriers exist in one of four states labeled ES^i, ES^o, E^i, and E^o (Figure 4.1). In the ES states, the solute S is bound to the carrier E; in the E state the carrier is unbound. In the ES^i and E^i states, the carrier, bound and unbound, communicates with the solution on the inner side of the membrane. In the ES^o and E^o states, the carrier, bound and unbound, communicates with the solution on the outer side of the membrane. The densities of carrier in the four states are n_{ES}^i, n_{ES}^o, n_E^i, and n_E^o moles per unit area of membrane. The fluxes of bound and unbound carrier are ϕ_{ES} and ϕ_E and the flux of solute is ϕ_S. The flux is defined as positive when the flux is in the outward direction; the units are in moles per unit area per second. The model is defined by the following assumptions:

- The total amount of carrier, bound and unbound, is constant, i.e. the sum of the carrier density over all of its states equals the total density of carrier in the membrane

$$N_{ET} = n_{ES}^i + n_{ES}^o + n_E^i + n_E^o. \tag{4.1}$$

- Since the carrier resides permanently in the membrane, the total flux of carrier must be zero, i.e.

$$\phi_{ES} + \phi_E = 0. \tag{4.2}$$

- The only time the solute crosses the membrane is when it is bound to the carrier; ES^o is assumed to undergo a reversible change in conformation to

---

[1]Because the kinetic equations for binding of the carrier to the solute are analogous to those of the binding of an enzyme to its substrate, the carrier is denote by E which stands for *enzyme*.

the form ES^i. The unbound carrier is assumed to undergo a similar reaction between the two conformations E^i and E^o. These two reactions are assumed to be first-order reactions with forward and reverse rate constants, α_E, α_{ES}, β_E, and β_{ES}, so that the fluxes are defined as follows:

$$\phi_{ES} = \phi_S \quad = \quad \alpha_{ES} n^i_{ES} - \beta_{ES} n^o_{ES}, \tag{4.3}$$

$$\phi_E \quad = \quad \alpha_E n^i_E - \beta_E n^o_E. \tag{4.4}$$

- In the *simple* model, the binding reactions at the membrane interfaces are assumed to be so rapid, compared to the rate of transport of solute across the membrane, that the membrane interface reactions are assumed to be at equilibrium, i.e.,

$$\frac{c^o_S n^o_E}{n^o_{ES}} = K^o_S \text{ and } \frac{c^i_S n^i_E}{n^i_{ES}} = K^i_S, \tag{4.5}$$

where K^o_S and K^i_S are the dissociation constants on the two membrane interfaces.

These equations can be solved to determine the n's in terms of the concentrations and transport parameters. Equations 4.3 and 4.4 can be used to determine the ϕ's. Combining Equations 4.2, 4.3, and 4.4 yields

$$(\alpha_{ES} n^i_{ES} - \beta_{ES} n^o_{ES}) + (\alpha_E n^i_E - \beta_E n^o_E) = 0. \tag{4.6}$$

Equations 4.1, 4.5, and 4.6 can be expressed as a matrix equation as follows:

$$\begin{bmatrix} K^i_S & 0 & -c^i_S & 0 \\ 0 & K^o_S & 0 & -c^o_S \\ \alpha_{ES} & -\beta_{ES} & \alpha_E & -\beta_E \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} n^i_{ES} \\ n^o_{ES} \\ n^i_E \\ n^o_E \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ N_{ET} \end{bmatrix}. \tag{4.7}$$

The first two rows correspond to the two relations in Equation 4.5. The third row results from Equation 4.6, and the fourth row corresponds to Equation 4.1. This set of simultaneous equations has the following solutions (Weiss, 1996a):

$$n^i_{ES} = N_{ET} \frac{c^i_S (\beta_E K^o_S + \beta_{ES} c^o_S)}{D_1}, \tag{4.8}$$

$$n^o_{ES} = N_{ET} \frac{c^o_S (\alpha_E K^i_S + \alpha_{ES} c^i_S)}{D_1}, \tag{4.9}$$

$$n^i_E = N_{ET} \frac{K^i_S (\beta_E K^o_S + \beta_{ES} c^o_S)}{D_1}, \tag{4.10}$$

$$n^o_E = N_{ET} \frac{K^o_S (\alpha_E K^i_S + \alpha_{ES} c^i_S)}{D_1}, \tag{4.11}$$
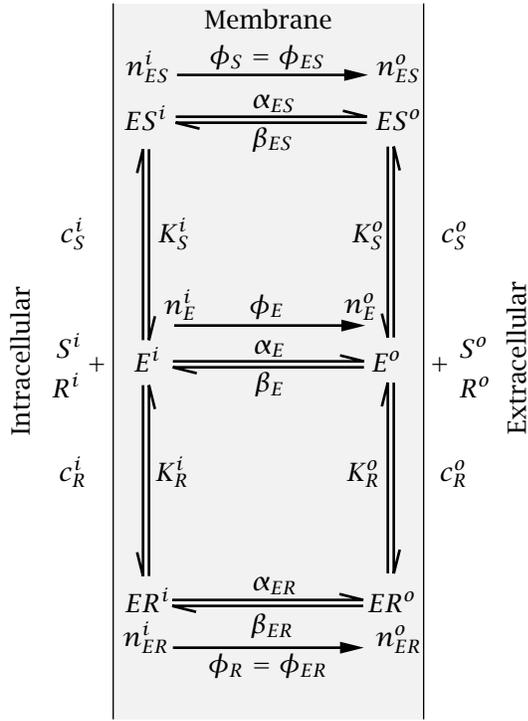
Figure 4.2: Kinetic diagram of a simple, six-state carrier that binds two solutes (S and R).

where

$$D_1 = (\beta_E K_S^o + \beta_{ES} c_S^o)(K_S^i + c_S^i) + (\alpha_E K_S^i + \alpha_{ES} c_S^i)(K_S^o + c_S^o). \tag{4.12}$$

ϕ_S and ϕ_E can be obtained from Equations 4.3 and 4.4.

The flux of solute can also be expressed in terms of the uni-directional fluxes as

$$\phi_S = \overrightarrow{\phi}_S - \overleftarrow{\phi}_S,$$

where

$$\overrightarrow{\phi}_S = \alpha_{ES} n_{ES}^i \text{ and } \overleftarrow{\phi}_S = \beta_{ES} n_{ES}^o.$$

4.2.2 Steady-state behavior of a simple, six-state carrier that binds two solutes

The simple, four-state carrier shown in Figure 4.1 can be extended to account for the binding of two solutes that compete for binding sites on the carrier (Figure 4.2). In this scheme, solutes S and R combine with carrier E but with different affinities. The binding to solute S has dissociation constants K_S^i and K_S^o and the binding to R has dissociation constants K_R^i and K_R^o.

The kinetic equations are analogous to those derived for the simple, four-state carrier except that the carrier now has 6 states:

- The total amount of carrier, bound and unbound, is constant, i.e. the sum of the carrier density over all of its six states equals the total density of carrier in the membrane

$$N_{ET} = n_{ES}^i + n_{ES}^o + n_{ER}^i + n_{ER}^o + n_E^i + n_E^o,$$ (4.13)

  where N_{ET} is the total density of carrier in the membrane.

- Since the carrier remains in the membrane, the net flux of carrier must be zero

$$\phi_{ES} + \phi_{ER} + \phi_E = 0.$$ (4.14)

- The fluxes are related to the carrier densities so that

$$\phi_{ES} = \phi_S = (\alpha_{ES} n_{ES}^i - \beta_{ES} n_{ES}^o).$$ (4.15)
$$\phi_{ER} = \phi_R = (\alpha_{ER} n_{ER}^i - \beta_{ER} n_{ER}^o).$$ (4.16)
$$\phi_E = (\alpha_E n_E^i - \beta_E n_E^o).$$ (4.17)

- In the *simple* model, the reactions at the membrane interfaces are assumed to take place so rapidly, compared to the rate of transport of solute across the membrane, that the membrane interface reactions are assumed to be at equilibrium, i.e.,

$$\frac{c_S^o n_E^o}{n_{ES}^o} = K_S^o, \frac{c_S^i n_E^i}{n_{ES}^i} = K_S^i \text{ and } \frac{c_R^o n_E^o}{n_{ER}^o} = K_R^o, \frac{c_R^i n_E^i}{n_{ER}^i} = K_R^i,$$ (4.18)

  where K_S^o, K_S^i, K_R^o and K_R^i are the dissociation constants for solutes S and R, respectively at the two membrane interfaces.

Combining Equations 4.14 and 4.15, 4.16, and 4.17 yields

$$(\alpha_{ES} n_{ES}^i - \beta_{ES} n_{ES}^o) + (\alpha_{ER} n_{ER}^i - \beta_{ER} n_{ER}^o) + (\alpha_E n_E^i - \beta_E n_E^o) = 0.$$ (4.19)

It is of interest to obtain the flux of solutes S and R as a function of both concentrations and the transport parameters K_S^i, K_S^o, K_R^i, K_R^o, α_{ES}, α_{ER}, α_E, β_{ES}, β_{ER}, β_E, and N_{ET}. Therefore, it is useful to regard the system of algebraic equations given by Equations 4.14 through 4.19 as a set of 6 equations in the 6 unknowns n_{ES}^i, n_{ES}^o, n_{ER}^i, n_{ER}^o, n_E^i, and n_E^o. These can be written in matrix form as follows:

$$\begin{bmatrix} K_S^i & 0 & 0 & 0 & -c_S^i & 0 \\ 0 & K_S^o & 0 & 0 & 0 & -c_S^o \\ 0 & 0 & K_R^i & 0 & -c_R^i & 0 \\ 0 & 0 & 0 & K_R^o & 0 & -c_R^o \\ \alpha_{ES} & -\beta_{ES} & \alpha_{ER} & -\beta_{ER} & \alpha_E & -\beta_E \\ 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} n_{ES}^i \\ n_{ES}^o \\ n_{ER}^i \\ n_{ER}^o \\ n_E^i \\ n_E^o \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ N_{ET} \end{bmatrix}.$$ (4.20)

The first four rows correspond to the four relations in Equation 4.18. The fifth row results from Equation 4.19, and the sixth row corresponds to Equation 4.13. This set of simultaneous equations has solutions (Weiss, 1996a):

$$n_{ES}^i = N_{ET} \frac{c_S^i K_R^i (\beta_E K_S^o K_R^o + \beta_{ES} c_S^o K_R^o + \beta_{ER} K_S^o c_R^o)}{D_2}, \tag{4.21}$$

$$n_{ES}^o = N_{ET} \frac{c_S^o K_R^o (\alpha_E K_S^i K_R^i + \alpha_{ES} c_S^i K_R^i + \alpha_{ER} K_S^i c_R^i)}{D_2}, \tag{4.22}$$

$$n_{ER}^i = N_{ET} \frac{K_S^i c_R^i (\beta_E K_S^o K_R^o + \beta_{ES} c_S^o K_R^o + \beta_{ER} K_S^o c_R^o)}{D_2}, \tag{4.23}$$

$$n_{ER}^o = N_{ET} \frac{c_R^o K_S^o (\alpha_E K_S^i K_R^i + \alpha_{ES} c_S^i K_R^i + \alpha_{ER} K_S^i c_R^i)}{D_2}, \tag{4.24}$$

$$n_E^i = N_{ET} \frac{K_S^i K_R^i (\beta_E K_S^o K_R^o + \beta_{ES} c_S^o K_R^o + \beta_{ER} K_S^o c_R^o)}{D_2}, \tag{4.25}$$

$$n_E^o = N_{ET} \frac{K_S^o K_R^o (\alpha_E K_S^i K_R^i + \alpha_{ES} c_S^i K_R^i + \alpha_{ER} K_S^i c_R^i)}{D_2}, \tag{4.26}$$

where

$$
\begin{aligned}
D_2 \;=\; & K_S^i (c_R^i \alpha_{ER} + K_R^i \alpha_E)(K_S^o K_R^o + c_S^o K_R^o + K_S^o c_R^o) + \\
& K_S^o (c_R^o \beta_{ER} + K_R^o \beta_E)(K_S^i K_R^i + c_S^i K_R^i + K_S^i c_R^i) + \\
& c_S^i K_R^i \alpha_{ES} (K_S^o K_R^o + c_S^o K_R^o + K_S^o c_R^o) + \\
& c_S^o K_R^o \beta_{ES} (K_S^i K_R^i + c_S^i K_R^i + K_S^i c_R^i).
\end{aligned}
\tag{4.27}
$$

The fluxes can be computed using Equations 4.15-4.17.

The solute fluxes can also be expressed in terms of the uni-directional fluxes as

$$
\begin{aligned}
\phi_S &= \vec{\phi}_S - \overleftarrow{\phi}_S, \\
\phi_R &= \vec{\phi}_R - \overleftarrow{\phi}_R,
\end{aligned}
$$

where

$$
\begin{aligned}
\vec{\phi}_S &= \alpha_{ES} n_{ES}^i \text{ and } \overleftarrow{\phi}_S = \beta_{ES} n_{ES}^o, \\
\vec{\phi}_R &= \alpha_{ER} n_{ER}^i \text{ and } \overleftarrow{\phi}_R = \beta_{ER} n_{ER}^o.
\end{aligned}
$$
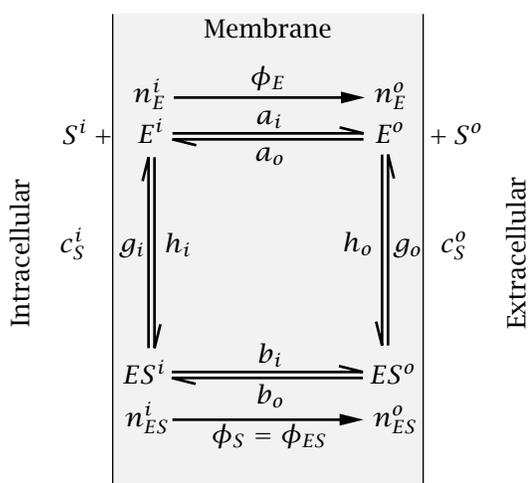
Figure 4.3: Kinetic diagram of a general, four-state carrier model that binds one solute.

4.2.3   Transient and steady-state behavior of a general, four-state carrier that binds one solute

Transient solution

Suppose that in the simple, four-state carrier model shown in Figure 4.1, the assumption that the interfacial binding reactions are fast is dropped. The model which results is called the *general, four-state carrier model* and the kinetic diagram for this model is shown in Figure 4.3.

This model is defined by the following assumptions:

- The total amount of carrier, bound and unbound, is constant, i.e. at each instant in time the sum of the densities of carrier over all of its states equals the total density of carrier

$$N_{ET} = n_{ES}^i(t) + n_{ES}^o(t) + n_E^i(t) + n_E^o(t). \qquad (4.28)$$

  Since both transient and steady-state values will be considered, the notation will be to specify a variable that changes in time as, for example, $n_{ES}^i(t)$ and its steady-state value as $n_{ES}^i(\infty)$.

- Since the carrier resides permanently in the membrane, the total flux of carrier must be zero, i.e.

$$\phi_{ES}(t) + \phi_E(t) = 0. \qquad (4.29)$$

- The only time the solute crosses the membrane is when it is bound to the carrier; ES^o is assumed to undergo a reversible change in conformation to the form ES^i. The unbound carrier is assumed to undergo a similar reaction between the two conformations E^i and E^o. The kinetic equations for the general, four-state carrier model can be written for the rate of change

of carrier in each of the four states by a matrix differential equation which is

$$\frac{d}{dt}\begin{bmatrix} n_E^i(t) \\ n_E^o(t) \\ n_{ES}^i(t) \\ n_{ES}^o(t) \end{bmatrix} =$$

$$\begin{bmatrix} -a_i - h_i c_S^i & a_o & g_i & 0 \\ a_i & -a_o - h_o c_S^o & 0 & g_o \\ h_i c_S^i & 0 & -b_i - g_i & b_o \\ 0 & h_o c_S^o & b_i & -b_o - g_o \end{bmatrix}\begin{bmatrix} n_E^i(t) \\ n_E^o(t) \\ n_{ES}^i(t) \\ n_{ES}^o(t) \end{bmatrix}. \quad (4.30)$$

The binding reactions are not assumed to be arbitrarily fast. Since the binding reactions are assumed to be second-order kinetic equations, the solute concentrations enter the binding reactions.

The four kinetic equations are not independent; e.g., the fourth can be derived from the first three equations. Thus, the fourth equation can be eliminated and $n_{ES}^o(t)$ can be eliminated using the conservation of carrier relation,

$$n_{ES}^o(t) = N_{ET} - n_E^i(t) - n_E^o(t) - n_{ES}^i(t), \quad (4.31)$$

to yield the matrix equation

$$\frac{d}{dt}\begin{bmatrix} n_E^i(t) \\ n_E^o(t) \\ n_{ES}^i(t) \end{bmatrix} =$$

$$\begin{bmatrix} -a_i - h_i c_S^i & a_o & g_i \\ a_i - g_o & -a_o - g_o - h_o c_S^o & -g_o \\ h_i c_S^i - b_o & -b_o & -b_i - g_i - b_o \end{bmatrix}\begin{bmatrix} n_E^i(t) \\ n_E^o(t) \\ n_{ES}^i(t) \end{bmatrix} + \quad (4.32)$$

$$\begin{bmatrix} 0 \\ g_o \\ b_o \end{bmatrix} N_{ET}. \quad (4.33)$$

This matrix equation can be solved for the three carrier densities and Equation 4.31 can be used to solve for the fourth density. The solution for each of the carrier densities has the form

$$n_j(t) = A_{j1}e^{\lambda_1 t} + A_{j2}e^{\lambda_2 t} + A_{j3}e^{\lambda_3 t}, \quad (4.34)$$

where the n_j's are the four carrier densities, the λ's are the three non-zero eigenvalues of the system, and the A's are the residues. This solution is valid when the eigenvalues are distinct. All 4 carrier densities have the same 3 eigenvalues, but in general, the residues differ. Since the rate constants and concentrations are all positive quantities, it can be shown that all the eigenvalues have negative real parts. Both the eigenvalues and residues are functions of the rate constants and the solute concentrations. Hence, the solutions yield the carrier densities $(n_{ES}^i(t), n_{ES}^o(t), n_E^i(t)$, and $n_E^o(t))$ as functions of time for any values of the concentrations of solute S and of the transport parameters a_i, a_o, b_i, b_o, g_i, g_o, h_i, h_o, and N_{ET}.

Steady-state solution

In the steady state, the density of carrier in each of its four states is constant and the derivative terms on the left-hand side of Equation 4.30 are zero. As noted previously, the four equations are not independent. Thus, one of the equations can be eliminated. However, conservation of carrier leads to an independent constraint which can be added to the three independent steady-state equations to yield the following four steady-state equations

$$
\begin{bmatrix}
-(a_i + h_i c_S^i) & a_o & g_i & 0 \\
a_i & -(a_o + h_o c_S^o) & 0 & g_o \\
h_i c_S^i & 0 & -(b_i + g_i) & b_o \\
1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
n_E^i(\infty) \\
n_E^o(\infty) \\
n_{ES}^i(\infty) \\
n_{ES}^o(\infty)
\end{bmatrix}
=
\begin{bmatrix}
0 \\
0 \\
0 \\
N_{ET}
\end{bmatrix}.
\tag{4.35}
$$

The solutions (Weiss, 1996a) are ratios of sums of products of rate constants

$$
\frac{n_E^i(\infty)}{N_{ET}} = \frac{M_E^i}{D}, \frac{n_E^o(\infty)}{N_{ET}} = \frac{M_E^o}{D}, \frac{n_{ES}^i(\infty)}{N_{ET}} = \frac{M_{ES}^i}{D}, \frac{n_{ES}^o(\infty)}{N_{ET}} = \frac{M_{ES}^o}{D},
\tag{4.36}
$$

where

$$
\begin{aligned}
M_E^i &= a_o b_i g_o + a_o g_i g_o + a_o b_o g_i + b_o g_i h_o c_S^o, \\
M_E^o &= a_i b_i g_o + a_i g_i g_o + a_i b_o g_i + b_i g_o h_i c_S^i, \\
M_{ES}^i &= a_i b_o h_o c_S^o + a_o g_o h_i c_S^i + a_o b_o h_i c_S^i + b_o h_i h_o c_S^i c_S^o, \\
M_{ES}^o &= a_i b_i h_o c_S^o + a_i g_i h_o c_S^o + a_o b_i h_i c_S^i + b_i h_i h_o c_S^i c_S^o, \\
D &= M_E^i + M_E^o + M_{ES}^i + M_{ES}^o.
\end{aligned}
\tag{4.37}
$$

The steady-state flux is found elsewhere and equals (Weiss, 1996a)

$$
-\phi_E(\infty) = \phi_S(\infty) = \phi_{ES}(\infty) = \frac{b_i g_o h_i c_S^i n_E^i(\infty) - b_o g_i h_o c_S^o n_E^o(\infty)}{b_i g_o + b_o g_i + g_i g_o}.
\tag{4.38}
$$

These equations yield the values of the $n(\infty)$'s and the $\phi(\infty)$'s in steady state as a function of the concentrations of solute S and of the transport parameters a_i, a_o, b_i, b_o, g_i, g_o, h_i, h_o, and N_{ET}.

The flux of solute can also be expressed in terms of the uni-directional fluxes as

$$
\phi_S(\infty) = \vec{\phi}_S(\infty) - \overleftarrow{\phi}_S(\infty),
$$

where

$$
\begin{aligned}
\vec{\phi}_S(\infty) &= \frac{b_i g_o h_i c_S^i n_E^i(\infty)}{b_i g_o + b_o g_i + g_i g_o}, \\
\overleftarrow{\phi}_S(\infty) &= \frac{b_o g_i h_o c_S^o n_E^o(\infty)}{b_i g_o + b_o g_i + g_i g_o}.
\end{aligned}
$$

4.3 Numerical Methods And Parameters

4.3.1 Numerical methods

The steady-state values of the carrier densities and the carrier fluxes are algebraic functions of the solute concentrations and of the parameters. They could have been computed directly from the steady-state matrix equations for each carrier model. However, it proved to be faster to compute these quantities directly from the solutions of these matrix equations which are: Equations 4.3-4.4 and Equations 4.8-4.11 for the simple, four-state carrier, Equations 4.15-4.17 and Equations 4.21-4.26 for the simple, six-state carrier, and Equations 4.36-4.38 for the general, four-state carrier.

The transient solution for the general, four-state carrier was obtained by solving Equation 4.33 numerically using MATLAB's `ode15s` routine for solving sets of ordinary differential equations. The solution for 3 of the 4 carrier densities were computed; the fourth was computed from Equation 4.31. Eigenvalues of the matrix of rate constants (Equation 4.33) were computed to aid the user in choosing a time scale for displaying transient responses. Eigenvalues were computed using MATLAB's `eig` function.

4.3.2 Choice of numerical parameters

The software enables the user to compute the n's and ϕ's for any values of the parameters, the α's, β's, K's, N_{ET}, and the concentrations of solute, the c's. For the simple models, the software is initiated with default parameters chosen to approximate hexose transport in human erythrocytes (Carruthers, 1984; Stein, 1986) assuming a symmetric transport scheme. The density of carriers was set to $N_{ET} = 10$ pmoles/cm^2. The dissociation constant of solute S was set to approximate that of D-glucose, $K_S^i = K_S^o = 2$ μmoles/cm^2, and that of R was set to that of a solute to which the carrier binds with lower affinity, approximating that of D-xylose, so that $K_R^i = K_R^o = 200$ μmoles/cm^2. All the rate constants were set equal with a value that made the maximum flux 100 pmoles/cm^2-s so that $\alpha_{ES} = \beta_{ES} = \alpha_{ER} = \beta_{ER} = \alpha_E = \beta_E = 20$ s^{-1}. For the general model, the parameters were chosen somewhat arbitrarily except that rate constants of binding reaction were chosen to be much greater than those of translocation reactions. All the parameters were chosen to be approximately consistent with the simple models. The concentrations of solute were chosen arbitrarily as follows: $c_S^i = 2$, $c_S^0 = 1$, $c_R^i = 2$, $c_R^o = 1$ μmoles/cm^3.

4.4 User's Guide

When the program is selected, 3 figures are displayed (in addition to MATLAB's command window): *CMT Control*, *CMT Parameters*, and *CMT State*. The *CMT*
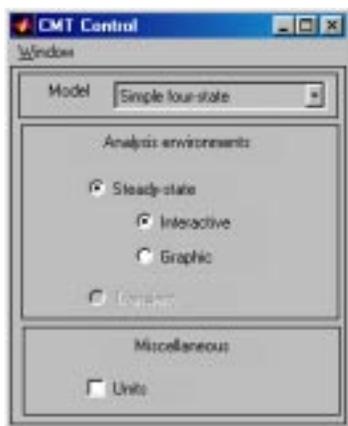
Figure 4.4: CMT Control figure.

Figure 4.5: CMT Units figure showing the units of all the variables in the carrier models.

Control figure controls the software. The remaining figures are for steady-state interactive analysis of the simple, four-state carrier. In the interactive mode, any parameter of the carrier model can be changed and the resulting steady-state values of carrier densities and fluxes are displayed immediately. The same steady-state interactive analysis is available in all three models: the simple symmetric, four-state carrier; the simple symmetric, six-state carrier; and the general, four-state carrier. Additional modes of analysis allow plotting the steady-state values of carrier densities and fluxes as functions of a user-specified model parameter. In addition, for the general, four-state carrier, the user can observe transient changes in carrier densities in response to a change in parameters. All of these modes of analysis are selected using the *CMT Control* figure.

4.4.1   CMT Control

The part of the *CMT Control* figure (Figure 4.4) below the menubar is divided into three panels. The *Model* panel allows the user to choose a particular model for analysis. The *Analysis* panel allows the user to choose among the available modes of analysis in the software by clicking on the appropriate button. The *miscellaneous panel* allows the user to display the units for all variables used by the software (Figure 4.5).

Figure 4.6: The CMT Parameters figure (left) and CMT State figure (right) for the simple, four-state carrier model that binds one solute.

4.4.2 Model

The *CMT Control* figure allows selection of three models of carrier-mediated transport. Selecting a different model will initiate the steady-state, interactive analysis for that model. The models are:

Simple, four-state: refers to the simple, four-state carrier that binds one solute. This is the default model that is displayed when the software is initialized. The parameters and state figures associated with this model are shown in Figure 4.6.

Simple, six state: refers to the simple, six-state carrier that binds two solutes shown in Figure 4.7.

General, four-state: refers to the general, four-state carrier that binds one solute shown in Figure 4.8.

4.4.3 Steady-state interactive analysis

Interactive, steady-state analysis involves two figures — the *CMT Parameters* and the *CMT State* figures. Any parameter (independent variable) can be changed in the *CMT Parameters* figure and the resulting changes in the steady-state values
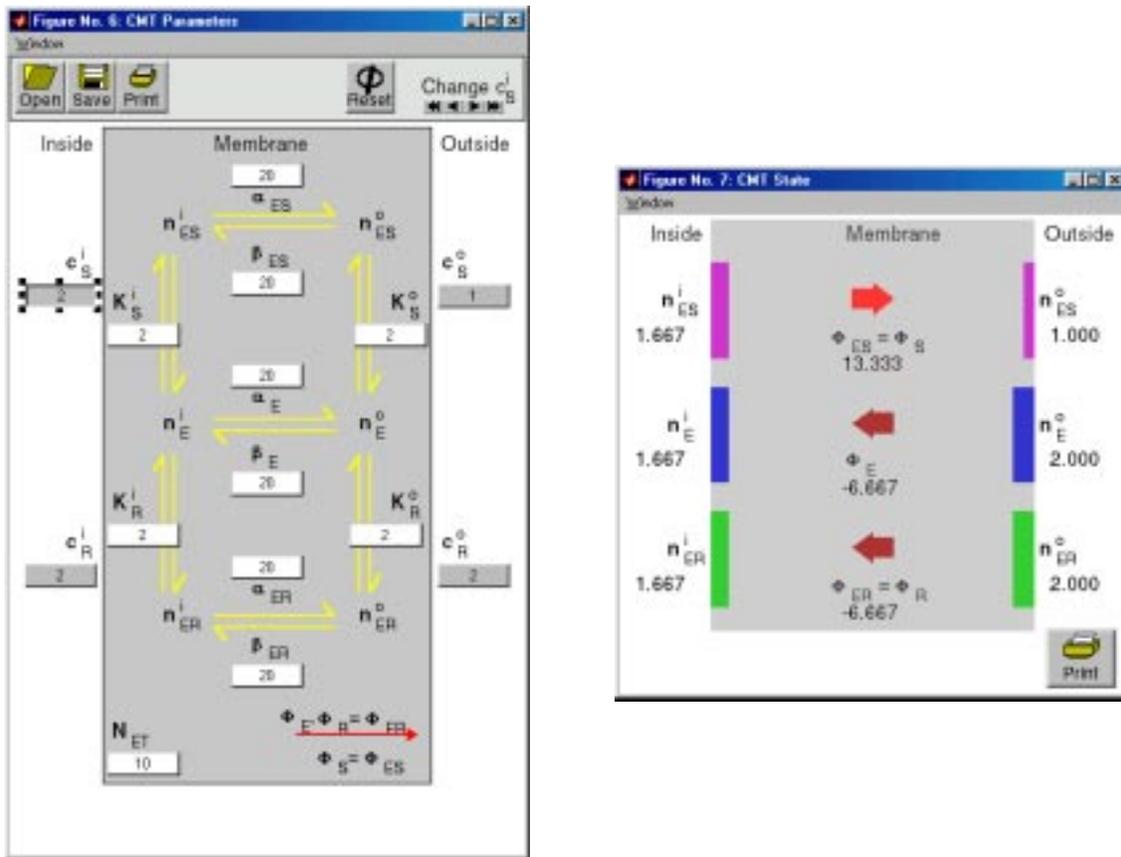
Figure 4.7: The CMT Parameters figure (left) and CMT State figure (right) for the simple, six-state carrier model that binds two solutes.
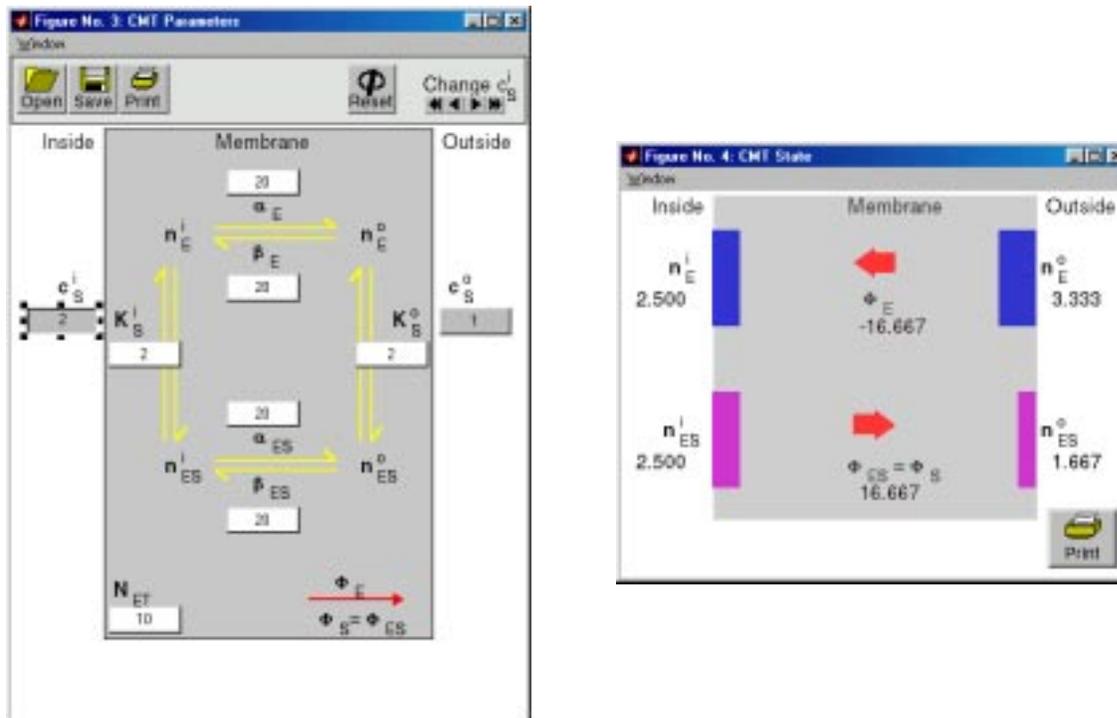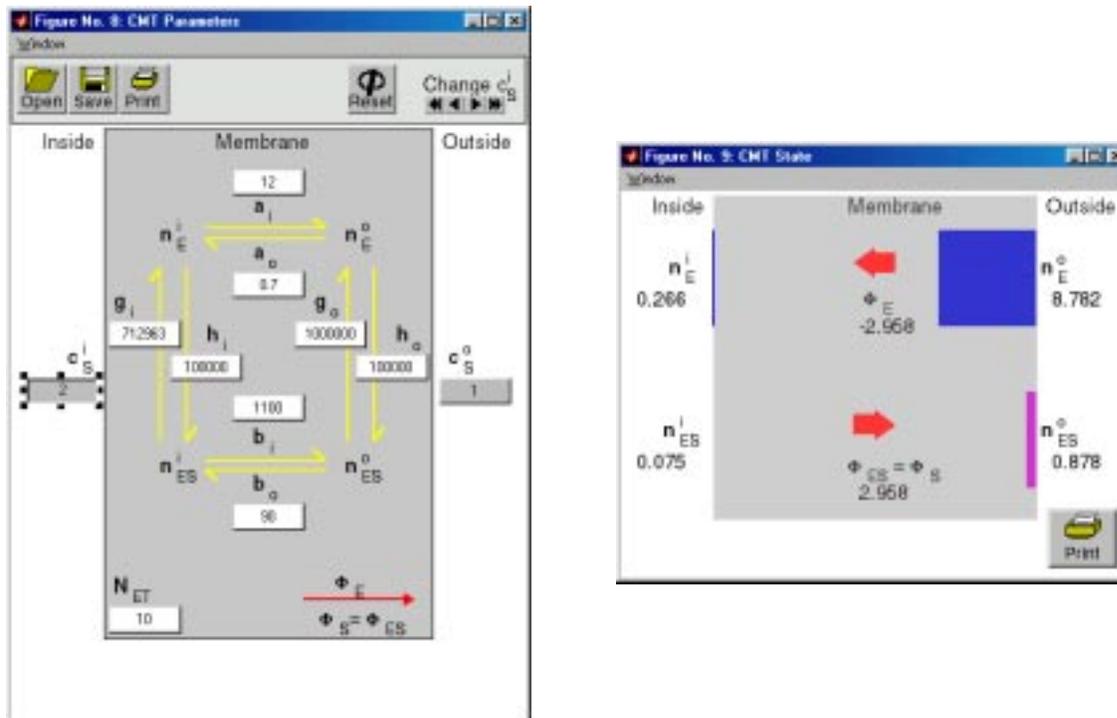
Figure 4.8: The CMT Parameters figure (left) and CMT State figure (right) for the general, four-state carrier model that binds one solute.

of the carrier state densities and carrier fluxes (dependent variables) are displayed in the *CMT State* figure. The parameters for all three carrier models are listed in Table 4.1. In both the *CMT Parameters* and *CMT State* figures, clicking on any variable (parameter, carrier density, or carrier flux) displays information about that variable.

CMT Parameters

The numerical value of each parameter is displayed in the button located either above or below each parameter. Parameter values can be changed *en masse* by clicking the appropriate buttons in the upper part of the parameters figure. Parameters can be *Reset* to their default values or read from a file. Parameters may also be saved to a file.

Parameters may be changed individually in three ways.

- A value may be changed arbitrarily by clicking on the button for each parameter. The button is changed to a text-edit box which allows editing the parameter value. After typing RETURN, the new values of carrier densities and carrier fluxes are computed and displayed in the *CMT State* figure. A non-numeric entry or a negative entry is not accepted and the prior value is restored.

	Simple four state	Simple six state	General four state
Total carrier density	N_{ET}	N_{ET}	N_{ET}
Rate constants	$\alpha_E, \beta_E,$	$\alpha_E, \beta_E, \alpha_{ES}, \beta_{ES},$	a_i, a_o, b_i, b_o
	α_{ES}, β_{ES}	α_{ER}, β_{ER}	g_i, g_o, h_i, h_o
Solute concentrations	c_S^i, c_S^o	$c_S^i, c_S^o, c_R^i, c_R^o$	c_S^i, c_S^o
Dissociation constants	K_S^i, K_S^o	$K_S^i, K_S^o, K_R^i, K_R^o$	

Table 4.1: Modifiable parameters of the carrier models — the independent variables.

- A parameter, selected by clicking on its button, may be incremented or decremented by clicking on ▸ or ◂ in the upper right in the *CMT Parameters* figure. The state figure is updated with each incremental change in the parameter.

- A parameter, selected by clicking on its button, may be incremented or decremented repeatedly by clicking on ▸▸ or to the left of the ◂◂, respectively. The *CMT State* figure is updated with each incremental change in the parameter. Clicking on a repeat button again stops the repetitive process.

In the general, four-state carrier, passive transport of solute (also called the *principle of detailed balance*) requires that (Weiss, 1996a)

$$a_i h_o b_o g_i = a_o h_i b_i g_o.$$

When this condition is violated, a warning messages is displayed in the *CMT Parameters* figure, and the ratio

$$r = \frac{a_o h_i b_i g_o}{a_i h_o b_o g_i}$$

is displayed. To avoid warning messages for small differences in r caused by the limited precision with which parameters are specified, messages are displayed only if r differs from 1 by more than 5%.

CMT State

The steady-state values of carrier densities and carrier fluxes are displayed in the *CMT State* figure below each quantity for the particular set of parameters

shown in the *CMT Parameters* figure. These quantities are also displayed graphically. Carrier densities are displayed with bar graphs that represent the relative densities of carrier in each states. Both four-state carriers display n_E^i, n_E^o, n_{ES}^i, and n_{ES}^o. The six-state carrier displays n_E^i, n_E^o, n_{ES}^i, n_{ES}^o, n_{ER}^i, and n_{ER}^o. The carrier fluxes are displayed with thick arrows in the center of the figure. No arrow is displayed when the flux is zero. The direction of the arrow indicates the direction of the flux. The color of the arrow indicates the relative magnitude of the carrier flux; a small magnitude flux has a black arrow, a large magnitude flux has a bright red arrow. The color coding is irrelevant for the four-state models since there are only two fluxes and they must have the same magnitude, but is useful for the six-state model.

4.4.4 Steady-state graphic analysis

Selection of *Graphic* analysis in the *CMT Control* figure, displays the *CMT Steady state plot* figure (Figure 4.9). This figure allows the user to plot any variable versus any of the parameters of the carrier-mediated transport model. This figure contains a number of panels.

Plot control. The panel in the upper left corner controls plotting in the manner described below.

> **Setup.** Clicking on Setup results in the display of the *CMT Setup steady state plot* figure (Figure 4.10) which allows selection of one independent variable which determines the abscissa of the plot and multiple dependent variables which determine the ordinates.
>
>> **Variables.** Clicking on an independent variable selects it for the abscissa (X) variable and deselects a previously selected variable. Clicking on an unselected dependent (Y) variable adds it to the collection of selected variables. Clicking on a selected dependent variable deselects it. A maximum of four *types* of dependent variables can be selected for plotting. The range over which the independent variable is swept can be selected in the text edit box. Values of parameters of the model (those not being changed) can be selected from a file or taken from the interactive environment by clicking on the appropriate button in the upper left panel of the figure.
>>
>> **Graph.** Clicking on Graph results in a plot of the selected data. The new plot replaces the old plot.
>>
>> **Overlay.** Clicking on Overlay results in the next plot being overlayed over the current plot.
>
> **Open.** Clicking this button allows plotting of results stored in a file.
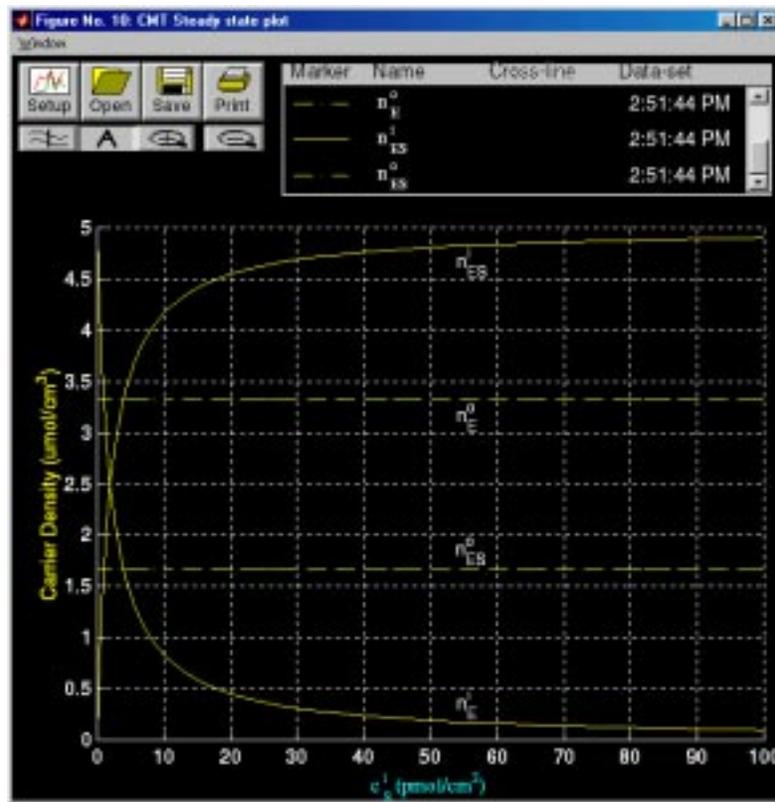
Figure 4.9: The CMT Steady state plot figure showing a plot of carrier densities versus c_S^i on linear ordinate and abscissa scales.
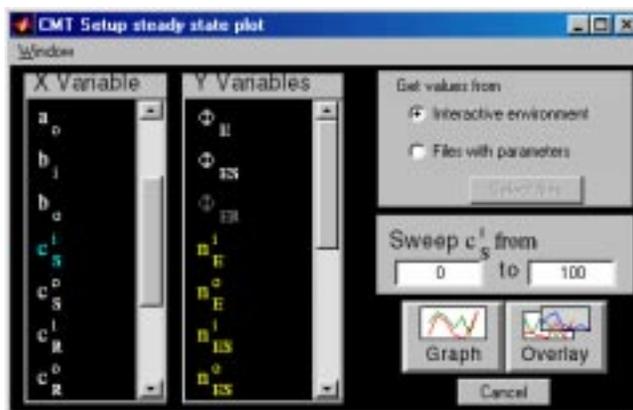
Figure 4.10: CMT Setup steady state plot figure showing a plot of fluxes and carrier densities versus c_S^i on linear ordinate and abscissa scales. Note that c_S^i (which is highlighted in blue) was selected as the independent variable that defines the abscissa (horizontal axis) in the plot. The variables n_E^i, n_E^o, n_{ES}^i, and n_{ES}^o were selected as the dependent variables and plotted as the ordinates (vertical axis).
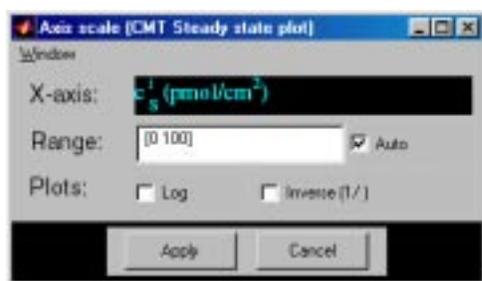
Figure 4.11: The Axis scale figure after the abscissa was selected in Figure 4.9.

Save. Clicking this button allows the information in the plot to be saved in a file. All lines, labels, and annotations are saved.

Print. Clicking on Print prints the figure.

Cross-line. Clicking on ≈ displays a vertical line, the cross line, in the plot field at the location of the pointer cursor. The line follows the cursor as it moves across the plot field. The values of all plotted variables at the intersection with the cross line are displayed in the legend. Clicking on cross-line button again removes the cross-line values from the legend and removes the cross line from the plotting field.

Annotate. Clicking on A and then on a desired location in the plot area displays a text edit box at that location. Click in the text edit box, type the annotation followed by a <RETURN> when the annotation is completed. Clicking on the annotation and dragging the mouse moves the annotation to a desired location in the plot field. The text string can be formatted into mathematical notation by using a LaTeX like notation, e.g., to get n_{ES}^i type n_{ES}^i.

Zoom. Clicking on ⊕ allows the user to magnify a region of the plotting field by clicking on it. Additional magnification can be achieved by zooming again.

Unzoom. Clicking on ⊖ undoes the effects of all prior zooms.

Axes control. The plot axes can be changed by clicking on any of the axis labels — either the abscissa or any of the ordinates — which results in the display of the *CMT axis scale* figure (Figure 4.11). The axis scale can be chosen to be linear, logarithmic, and/or reciprocal. If none of the options is chosen, the scale is linear. If *log scale* is chosen, the scale is the logarithm of the magnitude of the variable. If *invert* is chosen then the reciprocal of the abscissa variable is plotted either on a linear or on a logarithmic scale. Clicking on *Apply* results in a change in the axis. The title of the axis label can also be edited and the plot range can be changed.

Legend. The legend panel records a list of all data plotted. The following are recorded.
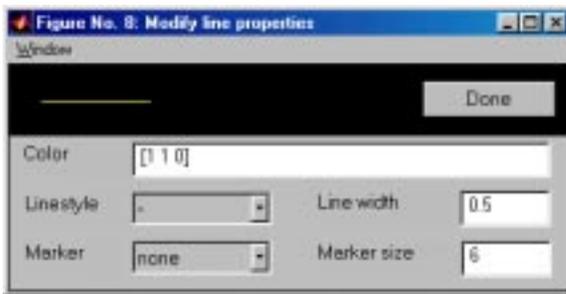
Figure 4.12: Modify line properties figure. The user can select the line color, style, and the marker.

Line properties. The color, line style, and symbol type are used to encode a particular variable; this marker is shown in the legend. Clicking on the marker displays a figure that allows the user to edit all the line and marker properties (Figure 4.12).

Variable name. The name of the variable that was plotted is shown. Clicking on the variable name alternately displays and hides the curve associated with this entry. When variables are added or deleted, the axis is auto-scaled if that option is selected — it is selected by default — in the axis figure.

Cross-line If the cross-line is used the cross-line values are displayed.

Data-set. The time when the curves were generated are indicated. Clicking on the time stamp alternately displays and hides all the curves associated with this legend item.

4.4.5   Transient analysis

Transient analysis is available only for the generalized, four-state carrier by clicking on *Transient* in the *CMT Control* figure. This action displays the *CMT Transient plot* figure (Figure 4.9). The display control features of this figure are similar to those of the *CMT Graphics Figure* described above (Figure 4.9). We describe the differences only.

Setup. Clicking on Setup displays the *CMT Setup transient plot* figure shown in Figure 4.14. The first column of this figure shows all the 11 independent variables for the general, four-state carrier. The initial values are shown in the second column and the final values are shown in the fourth column. The initial values can be changed individually by typing in the new value in the appropriate text edit box. The third column gives several options for changing any particular variable. Several independent variables can be changed at once. The transient parameters can be saved in a file and read from a file. The bottom of the figure gives several additional options for changing variables.

Numerics. Clicking on this item displays the *CMT Transients numerics* figure which gives information to the user and allows for control of the time scale
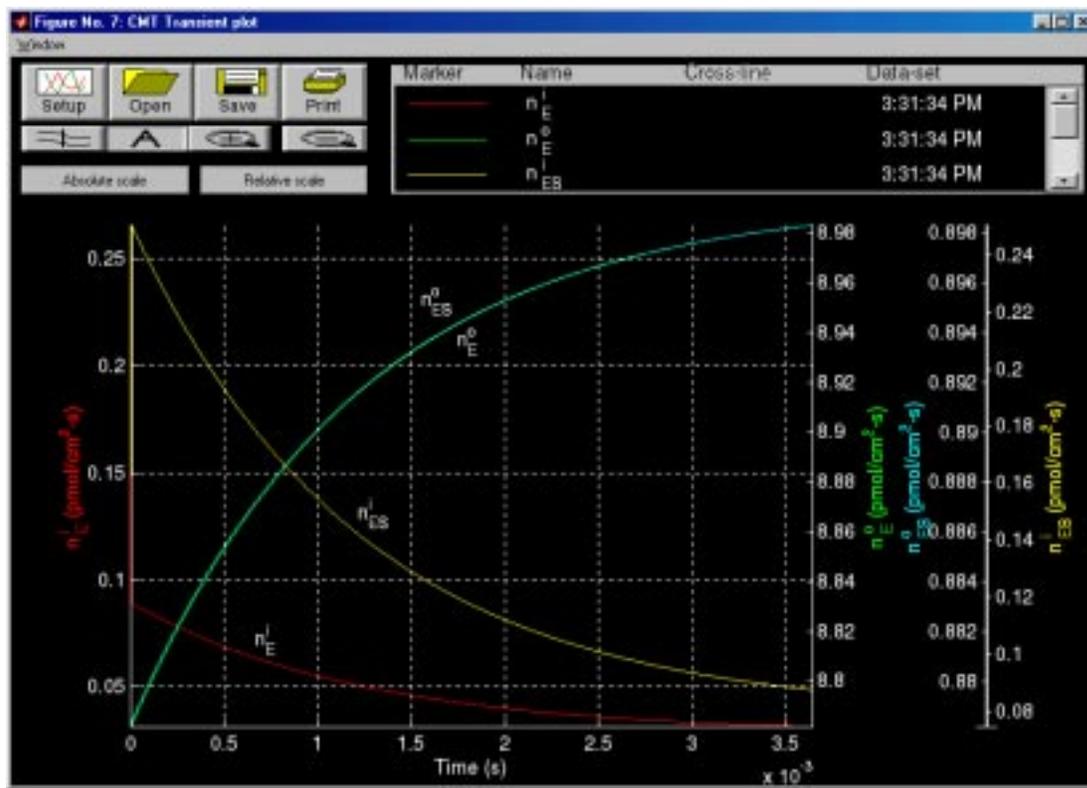
Figure 4.13: The CMT Transient plot figure showing the transient response to the change in parameters shown in Figure 4.14 and using the numerical data shown in Figure 4.15.
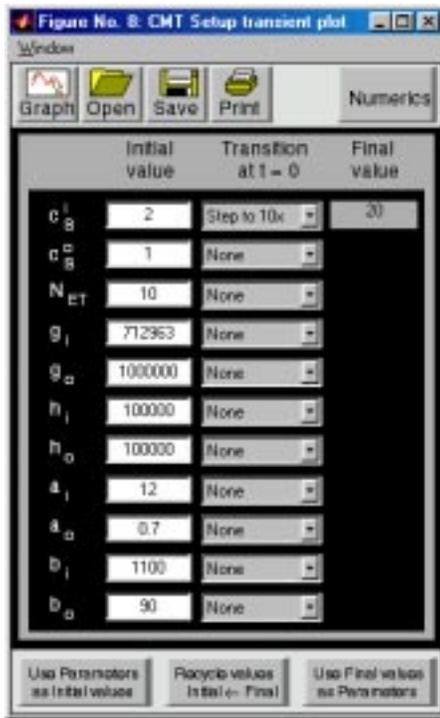
Figure 4.14: The CMT Setup transient plot figure.

of the transient response (Figure 4.15). The *CMT Transients numerics* figure displays the eigenvalues (as rate constants or time constants) of the transient response. These eigenvalues can be computed for the initial or the final values of all parameters by clicking on the appropriate button in this figure. The eigenvalues have the dimensions of rate constants (RC). But the software allows the equivalent time constants (TC = $-1/$RC) to be displayed. In addition, the user can choose to show the absolute value of either the rate constant or the time constant. The final value of the time in the plot is determined by the bottom panel which has several options.
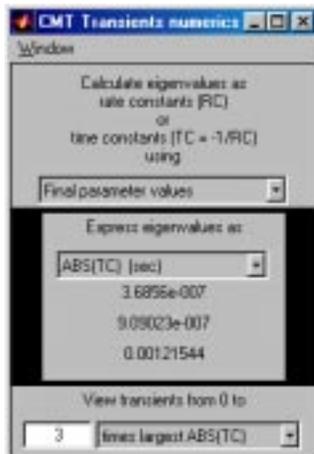
Figure 4.15: The CMT Transients numerics figure.

Absolute scale. Selecting this item displays plots of all carrier densities on the same absolute scale.

Relative scale. Selecting this item displays plots of all carrier densities on individual scales so that all the carrier densities are discernible.

4.5 Problems

Problem 4.1 *The purpose of this problem is to explore the steady-state behavior of the simple, symmetric, four-state carrier model.* You will observe the effects of changes in parameters on both the flux and the carrier densities. Use the simple, four-state carrier model and start with all the parameters at their default values.

a. What is the relation of ϕ_S and ϕ_E? Explain.

b. What is the relation of the direction of ϕ_S to the sign of $c_S^i - c_S^o$? Explain.

c. How does the state of the carrier depend on solute concentration?

    i. Do changes in c_S^i change n_{ES}^i and n_E^i? If so, how? If not, why not?

    ii. Do changes in c_S^i change n_{ES}^o and n_E^o? If so, how? If not, why not?

    iii. Do changes in c_S^o change n_{ES}^i and n_E^i? If so, how? If not, why not?

    iv. Do changes in c_S^o change n_{ES}^o and n_E^o? If so, how? If not, why not?

Problem 4.2 *The purpose of this problem is to explore the effect of asymmetry of the parameters of the simple, four-state carrier model on the relation between flux and concentration difference across the membrane.* Use the simple, four-state carrier model and start with all the parameters at their default values. Note the symmetry of the default parameters, i.e., $\alpha_E = \alpha_{ES}$, $\beta_E = \beta_{ES}$, and $K_S^i = K_S^o$.

a. Using the default parameters, plot ϕ_S versus c_S^i.

    i. Determine the value of c_S^i at which the sign of ϕ_S changes. Explain this sign change.

    ii. On this plot indicate the region for which $c_S^i - c_S^o > 0$. Is $\phi_S > 0$ for this range?

    iii. On this plot indicate the region for which $c_S^i - c_S^o < 0$. Is $\phi_S < 0$ for this range?

    iv. What can you conclude from parts i-iii?

b. Use the interactive environment, and change β_{ES} from its default value of 20 s^{-1} to 60 s^{-1}. At these values what is the relation of the solute concentration gradient to the flux of solute?

  c. Use the default parameters except that $\beta_{ES} = 60$ s^{-1}. Plot ϕ_S versus c_S^i.

     i. Determine the value of c_S^i at which the sign of ϕ_S changes. Explain this sign change.

     ii. On this plot indicate the region for which $c_S^i - c_S^o > 0$. Is $\phi_S > 0$ for this range?

     iii. On this plot indicate the region for which $c_S^i - c_S^o < 0$. Is $\phi_S < 0$ for this range?

     iv. What can you conclude from parts i-iii?

  d. In part c is the flux always down the chemical potential gradient of the solute? If not, from where does the energy for transporting the solute come?

Problem 4.3 *The purpose of this problem is to explore the effect of asymmetry of the parameters of the simple, four-state carrier model on the relation between unidirectional flux and concentration.*

  a. Use the simple, four-state carrier model and start with all the parameters at their default values. Note the symmetry of the default parameters, i.e., $\alpha_E = \alpha_{ES}$, $\beta_E = \beta_{ES}$, and $K_S^i = K_S^o$.

     i. Plot $1/\overrightarrow{\phi}_S$ versus $1/c_S^i$. [The unidirectional efflux $\overrightarrow{\phi}_S$ is called efflux$_S$ in the software.]

     ii. Plot $1/\overleftarrow{\phi}_S$ versus $1/c_S^o$. [The unidirectional influx $\overleftarrow{\phi}_S$ is called influx$_S$ in the software.]

     iii. To make direct comparisons between results you may wish to overlay the plots. What do you conclude from parts i and ii? Explain these results.

  b. Use the simple, four-state carrier model with all the parameters at their default values except change K_S^i from its default value of 2 to 4 μmol/cm^3. Note that the parameters are no longer symmetric.

     i. Plot $1/\overrightarrow{\phi}_S$ versus $1/c_S^i$. [The unidirectional efflux $\overrightarrow{\phi}_S$ is called efflux$_S$ in the software.]

     ii. Plot $1/\overleftarrow{\phi}_S$ versus $1/c_S^o$. [The unidirectional influx $\overleftarrow{\phi}_S$ is called influx$_S$ in the software.]

     iii. To make direct comparisons between results you may wish to overlay the plots. What do you conclude from parts i and ii? Explain these results.

c. Compare the results in part a with those obtained in part b. What can you conclude from these results?

Problem 4.4 *The purpose of this problem is to explore the functional relation between solute flux and solute concentration when expressed in different coordinates.* Use the simple, four-state carrier model and start with all the parameters at their default values, except set $c_S^o = 0$. In parts a through c your job is to estimate the values of $(\phi_S)_{max}$ and K_S^i from the graphs specified, where $(\phi_S)_{max}$ is the maximum flux of S with $c_S^o = 0$.

a. Obtain a graph of ϕ_S versus c_S^i in linear coordinates.

b. Obtain a graph of ϕ_S versus c_S^i in double logarithmic coordinates.

c. Obtain a graph of $1/\phi_S$ versus $1/c_S^i$.

d. Determine the values of $(\phi_S)_{max}$ and K_S^i from the model parameters.

e. Compare the 4 sets of values you have obtained for $(\phi_S)_{max}$ and K_S^i.

Problem 4.5 *The purpose of this problem is to explore the effect of solute concentration on carrier state.* Use the simple, four-state carrier model and start with all the parameters at their default values, except set $c_S^o = 0$. Obtain a graph of all four carrier states as a function of c_S^i. Some of the these n's increase, others decrease, while others remain constant. Summarize and explain the results you found.

Problem 4.6 *The purpose of this problem is to explore some of the conditions for which a plot of flux versus concentration in reciprocal coordinates results in a straight line.* Use the simple, four-state carrier model and start with all the parameters at their default values.

a. Set $c_S^o = 0$ and plot $1/\phi_S$ versus $1/c_S^i$.

b. Set c_S^o to its default value, $c_S^o = 1$ μmol/cm^3, and plot $1/\phi_S$ versus $1/c_S^i$.

c. Compare the results in parts a and b and explain the differences.

d. With $c_S^o = 1$ μmol/cm^3, plot $1/\overrightarrow{\phi}_S$ versus $1/c_S^i$. [The unidirectional efflux $\overrightarrow{\phi}_S$ is called efflux$_S$ in the software.] Compare the results of parts b and d and explain the difference.

Problem 4.7 *The purpose of this problem is to explore the process of exchange diffusion.* Use the simple, four-state carrier model and start with all the parameters at their default values.

a. In the interactive environment, set $\alpha_E = \beta_E = 0$. Explain the values of the variables in the state diagram.

b. In the graphics environment, plot ϕ_S versus c_S^i. What do you conclude?

c. In the graphics environment, plot $1/\overrightarrow{\phi}_S$ and $1/\overleftarrow{\phi}_S$ versus $1/c_S^i$. What do you conclude?

d. Reconcile the results of parts b and c.

e. The process defined in this problem has been called *exchange diffusion*. Critique this choice of name for this process. Why is this name appropriate? Why is this name inappropriate?

Problem 4.8 *The purpose of this problem is to compare properties of simple and general, four-state carriers.*

a. Use the simple, four-state carrier model and start with all the parameters at their default values. For both $c_S^o = 1$ and $c_S^o = 10$ μmol/cm^3, plot $1/\overrightarrow{\phi}_S$ versus $1/c_S^i$. [The unidirectional efflux $\overrightarrow{\phi}_S$ is called efflux$_S$ in the software.] You may wish to overlay the plots to compare the results directly.

b. Use the general, four-state carrier model and start with all the parameters at their default values. For both $c_S^o = 1$ and $c_S^o = 10$ μmol/cm^3, plot $1/\overrightarrow{\phi}_S$ versus $1/c_S^i$. [The unidirectional efflux $\overrightarrow{\phi}_S$ is called efflux$_S$ in the software.] You may wish to overlay the plots to compare the results directly.

c. What do you conclude from parts a and b.

Problem 4.9 *The purpose of this problem is to explore properties of six-state carriers.* Use the simple, six-state carrier model and start with all the parameters at their default values. Set the solute concentrations of S and R to zero on both sides of the membrane.

a. Explain the initial carrier states and flux values.

b. Increase c_S^i and observe both the carrier states and the flux. What is the relation of the flux of S to c_S^i?

c. Now set $c_S^i = 10$ μmol/cm^3 and increase c_R^i. How do the fluxes of S and R depend upon c_R^i?

d. Set $c_S^i = 10$ and $c_S^o = 5$ μmol/cm^3. Now increase c_R^i from an initial value of 0. How do the fluxes of S and R depend upon c_R^i? Pay particular attention to the direction of the flux of S.

Problem 4.10 *The purpose of this problem is to explore properties of the simple, six-state carrier model; in particular, the relation between the solute flux and the parameters of the model.* Use the simple, six-state carrier model and start with all the parameters at their default values. Choose any parameter of the model (a rate constant or a dissociation constant). Try to think of a change in that parameter that will lead to an increase in ϕ_S. Change the value of this parameter accordingly and determine if this change results in an increase in ϕ_S. If ϕ_S does not change the way you predicted, find out why. If it does change as you predicted, then try another parameter and repeat the process. Do this for at least 5 parameters to make sure that you have developed some intuition. Describe what you did and the outcome.

Problem 4.11 *The purpose of this problem is to explore elementary properties of the transient response of the general, four-state carrier.* Use the general, four-state carrier model and start with all the parameters at their default values.

a. In the interactive environment, record the values of all states of the carrier — n_E^i, n_{ES}^i, n_E^o, and n_{ES}^o. Now change c_S^i from its default value of 2 to 20 μmol/cm^3 and record the values of all the states of the carrier.

b. Switch to the transient environment and compute the transient response as c_S^i is stepped from its default value of 2 to 20 μmol/cm^3.

  i. Determine the initial and final values of the transient response and compare these results with the results obtained in part a.

  ii. From plots of the transient response, estimate the time it takes the response to go from its initial value to 0.63 of its final value.

c. Examine the CMT Transients numerics figure.

  i. Explain why there are multiple time constants listed?

  ii. Do the time constants listed in this figure explain the transient response seen in part b? Explain.

Problem 4.12 *The purpose of this problem is to explore eigenvalues of the general, four-state carrier model.* Use the general, four-state carrier model and start with all the parameters at their default values. Switch to the transient response and click on *Setup* in the *CMT Transient plot* figure. This displays the *CMT Setup transient plot* figure. Select *Numerics* in the latter plot to display the *CMT Transient numerics* figure. Make sure that the eigenvalues are calculated from the final parameter values and that rate constants are displayed instead of time constants. In this problem, you will change the final values and calculate the eigenvalues expressed as rate constants.

a. Record the eigenvalues (as rate constants) for the default settings.

b. Change the value of c_S^i from its default value of 2 to 20 μmol/cm^3. Record the eigenvalues (as rate constants) corresponding to $c_S^i = 20$ μmol/cm^3.

c. Compare the eigenvalues (as rate constants) for $c_S^i = 2$ to that for $c_S^i = 20$ μmol/cm^3.

d. Explain why the eigenvalues (as rate constants) depend upon the concentration.

e. Suppose you regard the general, four-state carrier as a system that relates the solute concentrations (as inputs) to the fluxes (as outputs). Is that system linear?

Problem 4.13 *The purpose of this problem is to explore the eigenvalues of the general, four-state carrier model.* Use the general, four-state carrier model and start with all the parameters at their default values. Switch to the transient response and click on *Setup* in the *CMT Transient plot* figure. This displays the *CMT Setup transient plot* figure. Select *Numerics* in the latter plot to display the *CMT Transient numerics* figure. Make sure that the eigenvalues are calculated from the final parameter values and are expressed as time constants. In this problem, you will change the final values and calculate the eigenvalues (as time constants).

a. Record the eigenvalues (as time constants) for the default settings.

b. For each of the values given below, plot the transient response and record the eigenvalues (as time constants).

  i. Change the binding/unbinding rate constants g_i, g_o, h_i, h_o to 0.1 of their default values.

  ii. Change the binding/unbinding rate constants g_i, g_o, h_i, h_o to 0.01 of their default values.

  iii. Change the binding/unbinding rate constants g_i, g_o, h_i, h_o to 0.001 of their default values.

  iv. Change the binding/unbinding rate constants g_i, g_o, h_i, h_o to 0.0001 of their default values.

c. Plot the locations of the eigenvalues (as time constants) versus the factor by which the binding/unbinding rate constants were multiplied in double logarithmic coordinates.

d. Examine the transient responses and the results of part c. Discuss the implication of these results.

Chapter 5

HODGKIN-HUXLEY MODEL — SPACE CLAMPED
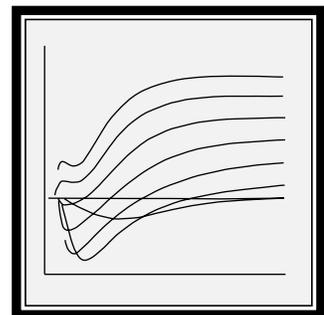
5.1   Introduction

5.1.1   Background

Electrically excitable cells produce action potentials which are important for transmission of information in neurons, for contraction of muscle cells, and for secretion of chemical substances by neurosecretory cells (Aidley, 1989; Hodgkin, 1964; Katz, 1966; Kandel et al., 1991; Keynes and Aidley, 1991; Nicholls et al., 1992; Weiss, 1996b). The *macroscopic* mechanisms of production of action potentials can be understood in terms of the nonlinear electrical characteristics of cellular membranes first described by Hodgkin and Huxley for the membrane of the giant axon of the squid (Hodgkin and Huxley, 1952). Since the nonlinear properties of the membranes of electrically excitable cells are controlled by the membrane potential, it is relatively simple to understand these mechanisms when the membrane potential is constant, but vastly more difficult when the membrane potential is not constant such as occurs during an action potential. Of course, it is the properties of electrically-excitable cells under the latter conditions that are the most important for the function of these cells.

5.1.2   Overview of the software

The purpose of this software package is to allow users to enhance their comprehension of the Hodgkin-Huxley model by providing a complete simulation of the model under space clamped conditions. Thus, this software package allows users to explore the dependence on time of the space-independent Hodgkin-Huxley model. This software package is complementary to the the software package described in Chapter 6 which examines the dependence of the Hodgkin-Huxley model on space and on time. In the software described in the current chapter, the user can specify the mode of stimulation (either voltage or current clamp), model parameters (such as the membrane capacitance, the maximum membrane conductances for membrane ionic channels, kinetic parameters for each of these membrane conductances, the concentrations of all relevant ions, the temperature, etc.), run the simulation, and display the results (such as the membrane potential, membrane-current components, membrane-conductance components, and channel activation and inactivation variables). The results can be saved in files, and the contents of multiple files can be plotted on the same axes. Hence, any of the parameters of the Hodgkin-Huxley model can be changed by the user and simulation experiments can be done to examine the consequences of the parameter changes. Extensive graphics utilities allow the user to plot any variable versus any other variable in the model, to zoom on any part of the plot, to annotate the plot, and to determine the coordinates of any point on a plot using a mouse-driven cursor.
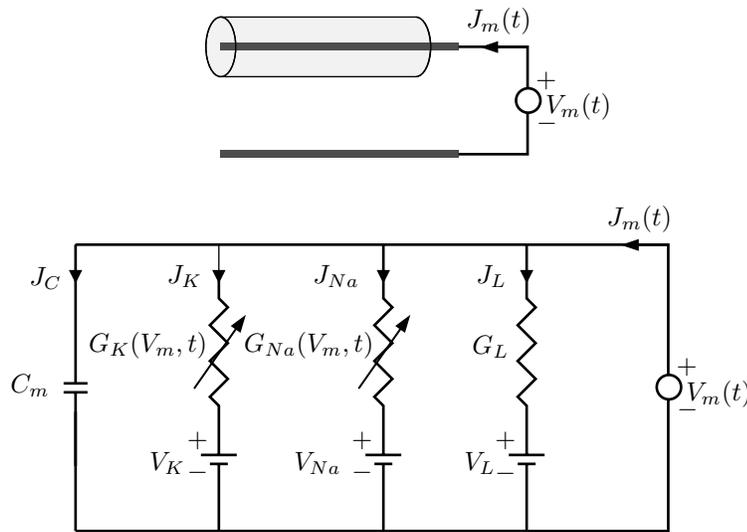
Figure 5.1: Schematic diagram and circuit model of a space-clamped axon in the voltage-clamp configuration.

5.2   Description Of The Model

Descriptions of the Hodgkin-Huxley model are available elsewhere (Aidley, 1989; Hodgkin, 1964; Katz, 1966; Plonsey and Barr, 1988; Johnston and Wu, 1995; Weiss, 1996b). A summary of this model is given in this section. The model that is the basis of this software is a one-compartment model that represents a space-clamped axon stimulated by means of two electrodes so that the membrane current density and membrane potential are uniform along the length of the axon.

5.2.1   Voltage-clamp and current-clamp configurations

Two simple and *mutually exclusive* stimulus configurations can be used to investigate the model: the *voltage-clamp* and *current-clamp* configurations. In the voltage-clamp configuration (Figure 5.1), the simulated space-clamped axon is driven by a voltage source and the membrane current density is computed. In the current-clamp configuration, the simulated space-clamped axon is driven by a current source and the membrane potential is computed.

5.2.2   The membrane current density components

The total membrane current density, J_m, is the sum of the capacitance current density plus the ionic current density

$$J_m = J_C + J_{ion}, \tag{5.1}$$

where the capacitance current density is
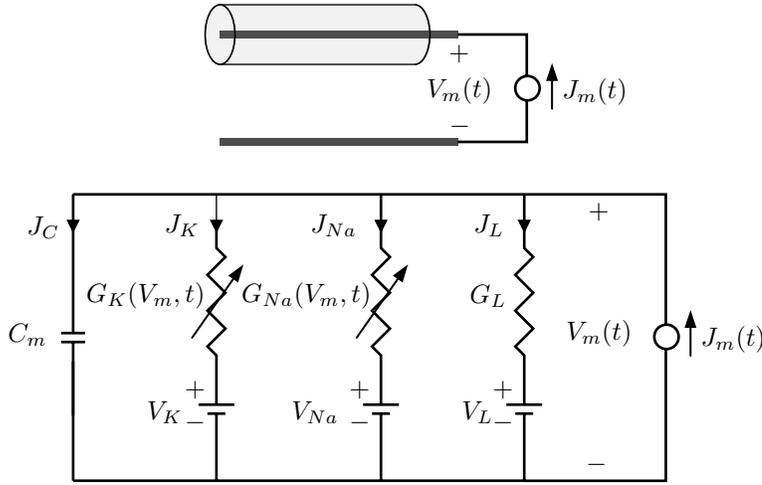
$$J_C = C_m \frac{dV_m(t)}{dt}, \tag{5.2}$$

Figure 5.2: Schematic diagram and circuit model of a space-clamped axon in the current-clamp configuration.

where C_m is the capacitance of a unit area of membrane and V_m is the membrane potential. The ionic current density can be expressed in terms of its components

$$J_{ion} = J_{Na} + J_K + J_L, \tag{5.3}$$

where the ionic current densities due to sodium, potassium and leakage are

$$
\begin{aligned}
J_{Na} &= G_{Na}(V_m,t)(V_m(t) - V_{Na}), &(5.4)\\
J_K &= G_K(V_m,t)(V_m(t) - V_K), &(5.5)\\
J_L &= G_L(V_m(t) - V_L), &(5.6)
\end{aligned}
$$

and where the Nernst equilibrium potential for the nth ion is defined in terms of the concentrations as follows

$$V_n = \frac{RT}{z_n F} \ln\left(\frac{c_n^o}{c_n^i}\right), \tag{5.7}$$

where R is the molar gas constant, T is absolute temperature, F is Faraday's constant, and z_n, c_n^o and c_n^i are the valence, outside, and inside concentrations of ion n, respectively. For a univalent ion, the Nernst equilibrium potential is expressed as

$$V_n = 0.08616\,(T_c + 273.16)\ln\left(\frac{c_n^o}{c_n^i}\right)\ (\text{mV}), \tag{5.8}$$

where T_c is the temperature in Centigrade.

5.2.3   The membrane conductances

The sodium and potassium conductances are defined as

$$
\begin{aligned}
G_{Na}(V_m,t) &= \overline{G}_{Na}m^3(V_m,t)h(V_m,t), &(5.9)\\
G_K(V_m,t) &= \overline{G}_K n^4(V_m,t). &(5.10)
\end{aligned}
$$

5.2.4 The activation and inactivation factors

The first-order kinetic equations for the activation and inactivation factors are written in terms of the rate constants as follows:

$$\frac{dm}{dt} = \alpha_m - m(\alpha_m + \beta_m), \tag{5.11}$$

$$\frac{dh}{dt} = \alpha_h - h(\alpha_h + \beta_h), \tag{5.12}$$

$$\frac{dn}{dt} = \alpha_n - n(\alpha_n + \beta_n), \tag{5.13}$$

where the α's and β's depend upon V_m.

5.2.5 The rate constants

The dependence of the rate constants on membrane potential has been generalized from the original Hodgkin-Huxley model to allow control of the potential dependence of individual rate constants and to represent approximately the effects of changes in calcium concentration and temperature.

The rate constants are

$$\alpha_m = \frac{-0.1(35 + V_m + \Delta V_{Ca} + V_{\alpha m})}{e^{-0.1(35 + V_m + \Delta V_{Ca} + V_{\alpha m})} - 1} K_T K_m, \tag{5.14}$$

$$\beta_m = 4e^{-(V_m + \Delta V_{Ca} + V_{\beta m} + 60)/18} K_T K_m, \tag{5.15}$$

$$\alpha_h = 0.07e^{-0.05(V_m + \Delta V_{Ca} + V_{\alpha h} + 60)} K_T K_h, \tag{5.16}$$

$$\beta_h = \frac{1}{1 + e^{-0.1(V_m + \Delta V_{Ca} + V_{\beta h} + 30)}} K_T K_h, \tag{5.17}$$

$$\alpha_n = \frac{-0.01(V_m + \Delta V_{Ca} + V_{\alpha n} + 50)}{e^{-0.1(V_m + \Delta V_{Ca} + V_{\alpha n} + 50)} - 1} K_T K_n, \tag{5.18}$$

$$\beta_n = 0.125e^{-0.0125(V_m + \Delta V_{Ca} + V_{\beta n} + 60)} K_T K_n, \tag{5.19}$$

where K_T is a temperature factor, that is defined as

$$K_T = 3^{(T_c - 6.3)/10}. \tag{5.20}$$

K_T multiplies all the rate constants; this effect approximates the effect of temperature on the electrical properties of the membrane of the squid giant axon (Huxley, 1959). The factors K_m, K_h and K_n have been added to the original Hodgkin-Huxley model to allow changes to be made in the individual rate constants of m, h, and n. The factor ΔV_{Ca} is used to approximate the dependence of rate constants on calcium concentration and is

$$\Delta V_{Ca} = 0.03335 \, (T_c + 273.16) \left(\ln \left(\frac{c_{Ca}^o}{c_{Ca}^i} \right) - 12.995 \right) \text{ (mV)}. \tag{5.21}$$

At the normal calcium concentration, the potential ΔV_{Ca} has the value 0. For calcium concentration ratios that differ from the normal value, ΔV_{Ca} differs from 0 and the dependence of the rate constants on the membrane potential is shifted by a term that is proportional to the calcium equilibrium potential. This approximates the effect of a change in calcium concentration on the parameters of squid giant axon membrane (Frankenhaeuser and Hodgkin, 1957). The potentials $V_{\alpha m}$, $V_{\beta m}$, $V_{\alpha h}$, $V_{\beta h}$, $V_{\alpha n}$, and $V_{\beta n}$ have the value 0 in the original Hodgkin-Huxley model. These potentials can be used to shift the dependence of individual rate constants on membrane potential.

5.2.6 Time constants and equilibrium values of activation and inactivation factors

Equations 5.11-5.13 are expressed in terms of rate factors, but it is also useful to express these equations in terms of time constants and equilibrium values, as follows

$$
\begin{aligned}
\frac{dm}{dt} &= \frac{m_\infty - m}{\tau_m}, \\
\frac{dh}{dt} &= \frac{h_\infty - h}{\tau_h}, \\
\frac{dn}{dt} &= \frac{n_\infty - n}{\tau_n}.
\end{aligned}
\tag{5.22}
$$

The time constants and equilibrium values can be defined in terms of the rate constants by comparing Equations 5.11-5.13 to Equation 5.22.

$$
\begin{aligned}
\tau_m &= \frac{1}{\alpha_m + \beta_m} \quad \text{and} \quad m_\infty = \frac{\alpha_m}{\alpha_m + \beta_m}, \\
\tau_h &= \frac{1}{\alpha_h + \beta_h} \quad \text{and} \quad h_\infty = \frac{\alpha_h}{\alpha_h + \beta_h}, \\
\tau_n &= \frac{1}{\alpha_n + \beta_n} \quad \text{and} \quad n_\infty = \frac{\alpha_n}{\alpha_n + \beta_n}.
\end{aligned}
\tag{5.23}
$$

5.2.7 Default values of parameters

The default numerical values for the Hodgkin-Huxley model are: $\overline{G}_{Na} = 120$, $\overline{G}_K = 36$, and $G_L = 0.3$ mS/cm^2; $C_m = 1$ μF/cm^2; $c_{Na}^o = 491$, $c_{Na}^i = 50$, $c_K^o = 20.11$, $c_K^i = 400$, $c_{Ca}^o = 44$, $c_{Ca}^i = 0.00011$ mmol/L; $V_L = -49$ mV; temperature is 6.3°C.

5.3 Numerical Methods

5.3.1 Background

In order to compute the membrane potential for a given set of parameters, the equations describing the Hodgkin-Huxley model (listed in Section 5.2) must be approximated in a form that is appropriate for numerical solution on a digital computer. A variety of methods exist for the numerical integration of a set of coupled differential equations (Press et al., 1986). The simplest methods, called the forward and backward Euler methods, is to approximate all derivatives by first differences, and to solve the resulting collection of algebraic equations at a series of discrete times separated by a time increment Δt. Although simple in concept, the Euler methods are not particularly accurate or efficient. Because the Hodgkin-Huxley software is built on MATLAB which includes a variety of methods of numerical integration, the user can choose the method of integration from a number of choices.

5.3.2 Choice of integration step Δt

The accuracy of all numerical methods is dependent on the choice of the integration step Δt. Simulation users *will* need to concern themselves with the choice of Δt. If Δt is decreased, the accuracy of a single step increases but the time it takes to compute the solution increases. Furthermore, if computations are continued for a large number of steps, the accumulation of round-off errors, due to the limited precision for representing numbers in a computer, can become appreciable. Conversely, if Δt is increased, the accuracy of the computation decreases and the solution may even diverge. This divergence can lead to spurious phenomena that may mistakenly be attributed to the Hodgkin-Huxley model when they are really artifacts of the numerical methods. A plausible empirical test of whether or not a solution is sufficiently accurate is to test the sensitivity of the solution to changes in Δt. If increasing the value of Δt does not appreciably change the solution, then the value of Δt used initially is too small and time was wasted in performing the computation. If decreasing the value of Δt does not appreciably change the computational result, then the larger value is preferable. The simulation of the Hodgkin-Huxley model is set initially with *default numerical parameters*. For changes in parameters of the model that do not appreciably change the various time constants of the model, the default numerical parameters are adequate. However, if changing a model parameter (such as the temperature, membrane capacitance, etc.) appreciably decreases some time constant of the system, then a smaller value of Δt should be used. Therefore, it is advisable to explore briefly the effects of varying Δt on simulation results.
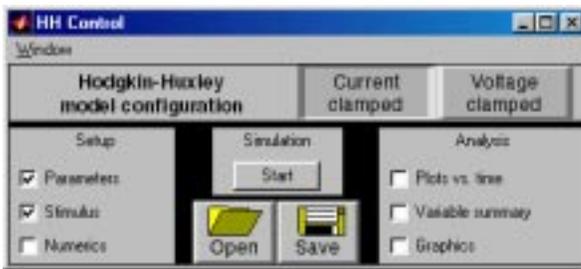
Figure 5.3: The HH Control figure after the software is initiated.

5.3.3   Method for computing solutions

In the *voltage clamp* simulation, the solution is obtained as follows: Suppose the solution is known at time $t_p - \Delta t$ and the solution at time t_p is desired. The value of the membrane potential $V_m(t_p)$ at t_p is determined from specification of the stimulus (see Equations 6.23 and 6.24 on Page 149). This value is substituted into Equations 5.14 and 5.19 to determine the rate constants at t_p. Using these rate constants, and given the values of m, h, and n at time $t_p - \Delta t$, the values of m, h, and n are determined at t_p by solving Equations 5.11-5.13 numerically by an integration method that can be chosen by the user. The values of these factors are used to compute the values of the conductances and current densities from Equations 5.1-5.6. The computation is begun with initial values of m, h, and n that are determined from Equation 5.23 for a membrane potential equal to the source voltage for $t < 0$.

In the *current clamp* simulation, the initial values of m, h, n, and V_m are computed first. This is done by using MATLAB's `fzero` function to solve the set of Equations 5.1, 5.3-5.21 and a specification of the source current (Equation 6.23 on Page 149) with the derivatives of all the variables set to zero. Starting with these initial values, the time-varying equations are solved numerically by an integration method that can be chosen by the user.

5.4   User's Guide To The Software

When this software is selected, 3 figures are displayed (in addition to MATLAB's command window): *HH Control*, *HH Parameters*, *HH Stimulus*. The *HH Control* figure controls the software (Figure 5.3). The *HH Parameters* figure allows the user to change any of the parameters of the model. The *HH Stimulus* figure allows the user to specify the stimulus. Once the parameters and the stimulus are specified, clicking the *Start* pushbutton results in the computation of the response as a function of time. A variable is plotted as a function of time during the computation. Upon completion of the computation, the user can examine the results graphically and numerically.

5.4.1 HH Control

The part of the *HH Control* figure (Figure 5.3) immediately below the menubar allows the user to choose between voltage-clamp and current-clamp configurations. Below this choice of configuration, there are three panels. In the left panel, the user can choose to view the *HH Parameters* figure (which is used to specify the parameters), the *HH Stimulus* figure (which is used to specify the stimulus as a function of time), and/or the *HH Numerics* figure (which control the numerical methods and numerical parameters). In the center panel, the user can start the simulation by clicking on the *Start* button. The user can also save the simulation (including a specification of all the model and numerical parameters, the stimulus, and simulation results) in a file and open a previously stored file (that includes a specification of all the model and numerical parameters, the stimulus, and simulation results). The right (analysis) panel becomes active after the simulation is completed and allows the user to select three options for analysis of the results. The first option allows the user to plot any model variable as a function of time. This figure is displayed automatically as the computation proceeds. The second option displays a numerical summary of simulation variables. The third option allows the user to access a quite general plotting facility that not only plots any of the model variables versus any other model variable, but allows plotting results stored in files.

5.4.2 HH Parameters

Changing parameter values

The *HH Parameters* figure contains three columns for each parameter: the first column identifies the parameter, the second gives its default value, and the third column shows the value of the parameter if it is modified from its default value. Parameters are selected for modification by clicking on the entry in the third column of the parameter field. When a parameter is selected, its third column entry is replaced by an edit box. Clicking any mouse button inside the edit box allows the parameter to be changed. The new value of the parameter is entered from the keyboard; $<$ RETURN $>$ terminates the parameter entry. If the new value differs from the default value, then the new value will appear in the third column. With this method of display, the parameter list can be scanned quickly to indicate which parameters differ from their default values.

There are some restrictions on the numerical values of parameters; G's ≥ 0, $C_m \geq 0$, c's > 0, K's > 0, and the temperature must be above absolute zero. The derived parameters cannot be changed but are derived from the other parameters and are displayed for the convenience of the user. For example, the sodium equilibrium potential V_{Na} cannot be changed directly by the user, but changes automatically when the sodium concentrations or temperature are changed.
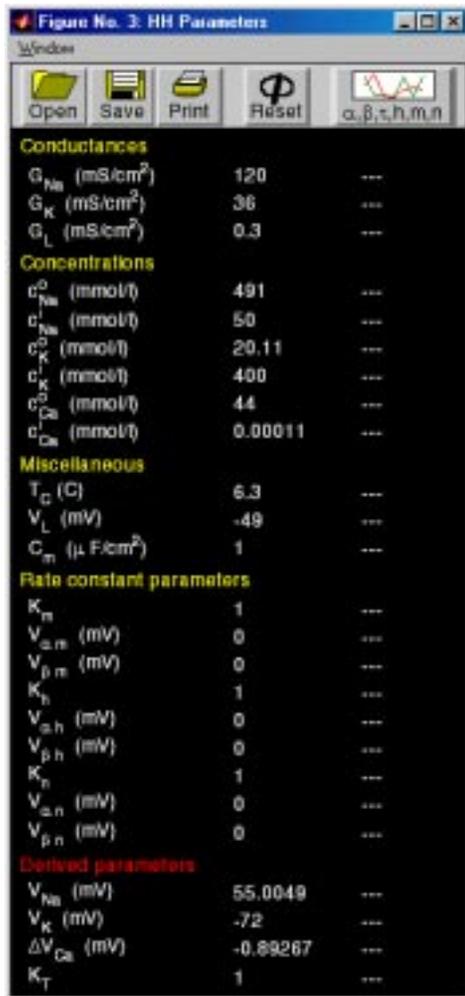
Figure 5.4: The HH Parameters figure after the software is initiated.

Reading, saving, printing, reseting the parameters

Parameters (only) can be read from a file by clicking on *Open* and can be saved by clicking on *Save*. Saving and reading in the *HH Parameters* figure deals only with the model parameters; e.g., simulation results are not read or saved. The *HH Parameters* figure can be printed by clicking the Print button. The parameters can be reset to default values by clicking on Reset.

Viewing the voltage dependent parameters

The Hodgkin-Huxley model contains several parameters that are instantaneous functions of the membrane potential. Plots of these parameters are available as a function of V_m for the following: $\alpha_m(V_m)$, $\beta_m(V_m)$, $m_\infty(V_m)$, $\tau_m(V_m)$, $\alpha_h(V_m)$, $\beta_h(V_m)$, $h_\infty(V_m)$, $\tau_h(V_m)$, $\alpha_n(V_m)$, $\beta_n(V_m)$, $n_\infty(V_m)$, $\tau_n(V_m)$ (see Sections 5.2.5 and 5.2.6). These can be accessed by clicking on the button marked $\alpha, \beta, \tau, h, m, n$ which results in the display of the *HH parameters vs. potential* figure (Figure 5.5). This figure contains a number of panels.

Plot control. The panel in the upper left corner controls plotting in the manner described below.

> **Setup.** Clicking on Setup results in the display of the *HH Setup parameters vs. potential plot* figure (Figure 5.6) which allows selection of the voltage-dependent parameters as a function of the membrane potential.
>
> > **Variables.** The abscissa is the membrane potential. Clicking on any variable adds it to the collection of selected ordinate variables. Clicking on it again deselects it. Values of parameters of the model can be selected from a file or taken from the parameters by clicking on the appropriate button in the upper right panel of the figure.
> >
> > **Graph.** Clicking on Graph results in a plot of the selected data. The new plot replaces the old plot.
> >
> > **Overlay.** Clicking on Overlay results in the next plot being overlayed over the current plot. Since the voltage-dependent parameters depend upon additional parameters and variables, overlaying the plots allows a display of the effect of these additional parameters and variables on the voltage dependence parameters. For example, a plot of m_∞ versus V_m for different calcium concentrations reveals the effects of calcium concentration on the voltage dependence of sodium activation (Figure 5.7).
>
> **Open.** Clicking this button allows plotting of results stored in a file.
>
> **Save.** Clicking this button allows the information in the plot to be saved in a file. All lines, labels, and annotations are saved.
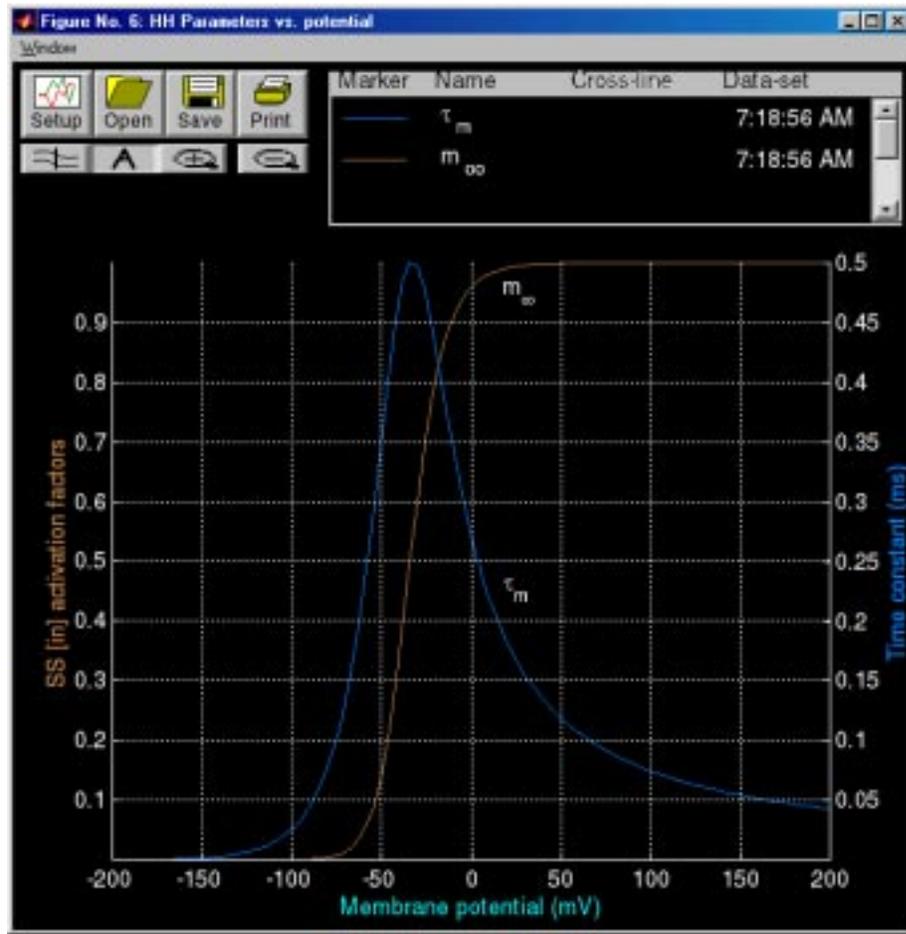
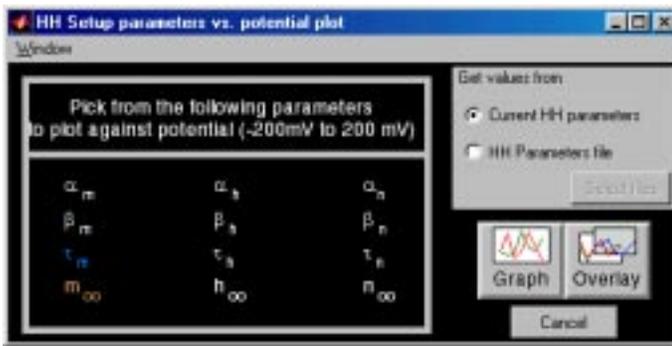Figure 5.5: The HH parameters vs. potential figure after selection of parameters as in Figure 5.6.

Figure 5.6: The HH Setup parameters vs. potential plot figure that was used to produce the plot shown in Figure 5.5.
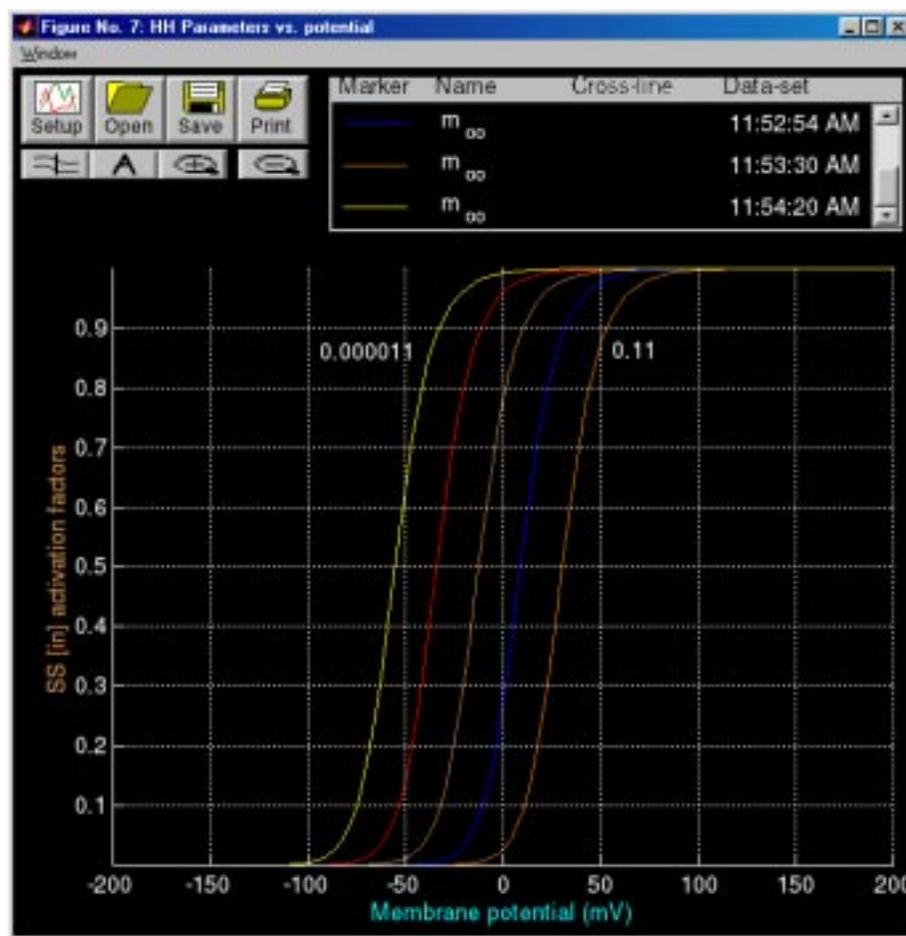
Figure 5.7: The HH parameters vs. potential figure showing the effect of calcium concentration on the voltage dependence of the sodium activation parameter m_∞. The calcium concentration was varied from 0.000011 to 0.11 mmol/L by factors of 10.
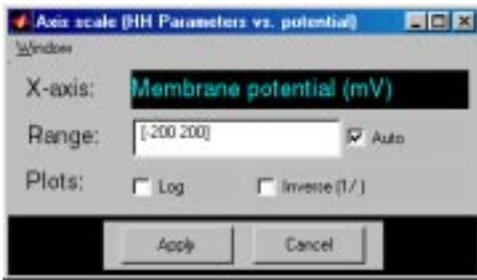
Figure 5.8: The Axis scale figure.

Print. Clicking on Print prints the figure.

Cross-line. Clicking on ≈ displays a vertical line, the cross line, in the plot field at the location of the pointer cursor. The line follows the cursor as it moves across the plot field. The values of all plotted variables at the intersection with the cross line are displayed in the legend. Clicking on cross-line button again removes the cross-line values from the legend and removes the cross line from the plotting field.

Annotate. Clicking on A and then on a desired location in the plot area displays a text edit box at that location. Click in the text edit box, type the annotation followed by a <RETURN> when the annotation is completed. Clicking on the annotation and dragging the mouse moves the annotation to a desired location in the plot field. The text string can be formatted into mathematical notation by using a LaTeX like notation, e.g., to get m_∞ type m_{\infty}.

Zoom. Clicking on ⊕ allows the user to magnify a region of the plotting field by clicking on it. Additional magnification can be achieved by zooming again.

Unzoom. Clicking on ⊖ undoes the effects of all prior zooms.

Axes control. The plot axes can be changed by clicking on any of the axis labels — either the abscissa or any of the ordinates — which results in the display of a figure that allows changing the axis (Figure 5.8). The axis scale can be chosen to be linear, logarithmic, and/or reciprocal. If neither the *Log* nor the *Inverse (1/)* options is chosen, the scale is linear. If *Log* is chosen, the scale is the logarithm of the magnitude of the variable. If *Inverse (1/)* is chosen then the reciprocal of the variable is plotted either on a linear or on a logarithmic scale. Clicking on *Apply* results in a change in the axis. The plot range can be changed, but the axis labels cannot be edited.

Legend. The legend panel records a list of all data plotted. The following are recorded.

Line properties. The color, line style, and symbol type are used to encode a particular variable; this marker is shown in the legend. Clicking on the
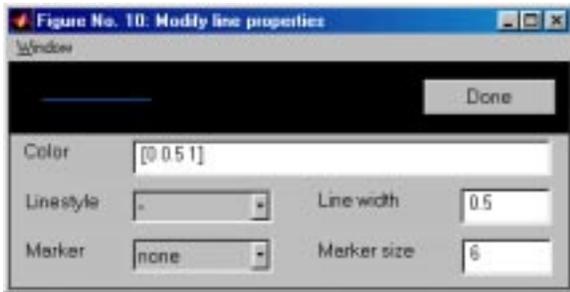
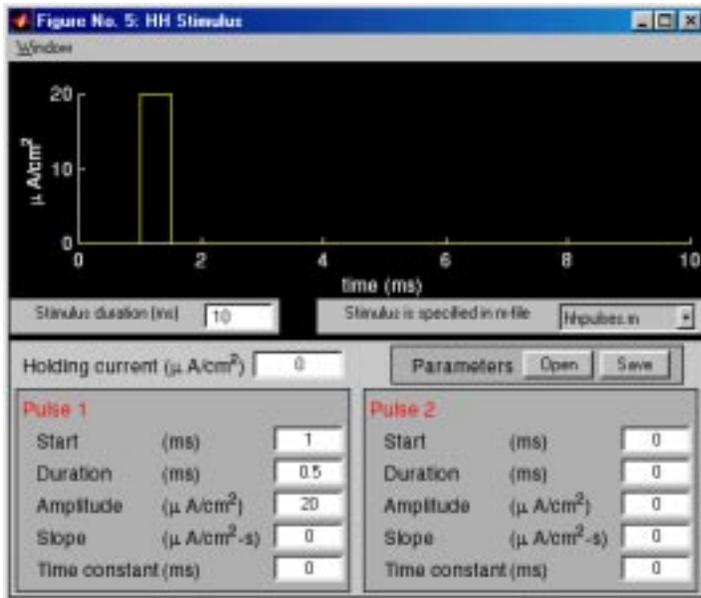Figure 5.9: The Modify line properties figure.

Figure 5.10: HH Stimulus figure with the default stimulus waveform for a current-clamp configuration.

marker allows the user to edit the attributes of the plot line/symbol (Figure 5.9).

Variable name. The name of the variable that was plotted is shown. Clicking on the variable name alternately displays and hides the curve associated with this entry. When variables are added or deleted, the axis is auto-scaled if that option is selected — it is selected by default — in the axis figure.

Cross-line If the cross-line is used the cross-line values are displayed.

Data-set. The time when the curves were generated are indicated. Clicking on the time stamp alternately displays and hides all the curves associated with this legend item.

5.4.3 HH Stimulus

Selecting *Stimulus* in the controls figure displays the *HH Stimulus* figure as shown in Figure 5.10. The stimulus is the membrane potential in the voltage-clamp con-

figuration and the membrane current density in the current-clamp configuration. The duration of the stimulus can be set and, as described below, the user can also select a file that will define the stimulus arbitrarily.

Use of the m-file hhpulse.m

By default, the stimulus is generated by the MATLAB m-file `hhpulse.m` which uses the parameters listed in the lower part of the stimulus figure. The stimulus is the sum of a constant plus two independently specifiable pulses. Each pulse can be a rectangular pulse, a ramp pulse or an exponential pulse or a combination of these. If the slope and the time constant are set to zero, then the remaining parameters specify the amplitude, duration and onset time of a rectangular pulse. If the slope has a non-zero value, a ramp of that slope is added to the rectangular pulse for the duration of the pulse. Similarly, if the time constant has a non-zero value, then an exponential pulse with the selected time constant is added to the waveform. The time constant can be positive or negative.

The stimulus, $s(t)$, is defined as follows

$$s(t) = s^o + s^1(t) + s^2(t), \tag{5.24}$$

where s^o is a constant and $s^1(t)$ and $s^2(t)$ are independently specifiable pulses. Each pulse has the form

$$s^i(t) = \begin{cases} m^i(t - t^i_o) + A^i e^{-(t-t^i_o)/\tau^i} & \text{if } t_o \leq t \leq t_o + t_{dur}, \\ 0 & \text{otherwise,} \end{cases} \tag{5.25}$$

where pulse i ($i = 1$ or 2) starts at t_o and has a duration t_{dur}. The amplitude of the pulse is A^i, the slope is m^i, and the time constant is τ^i. With this specification, a rich variety of stimuli can be produced; an example is shown in Figure 5.11.

The stimulus parameters alone can be read from or saved to a file by selecting either *Open* or *Save* in the *HH Stimulus* figure.

Use of stimulus m-files

This specification of the stimulus by `hhpulse.m` is quite rich. However, a user who wishes to define a stimulus that differs from the one provided can write a MATLAB m-file to generate that stimulus. That file can then be selected in the *HH Stimulus* figure, and the stimulus it generates will be displayed in the figure. However, the parameters will have to be set by the user in the m-file. The script for the available m-files can be used as templates to define new user-specified m-files. For example, the simple m-file shown in Figure 5.12 generates a sinusoidal stimulus.
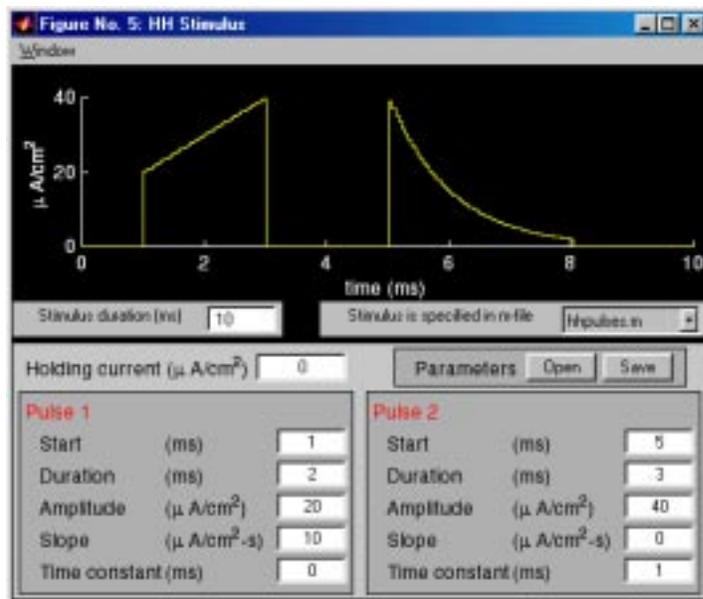
Figure 5.11: HH Stimulus figure showing an example of a stimulus current waveform consisting of a constant plus two pulses: the first pulse is a ramp plus rectangular pulse; the second is an exponential pulse.
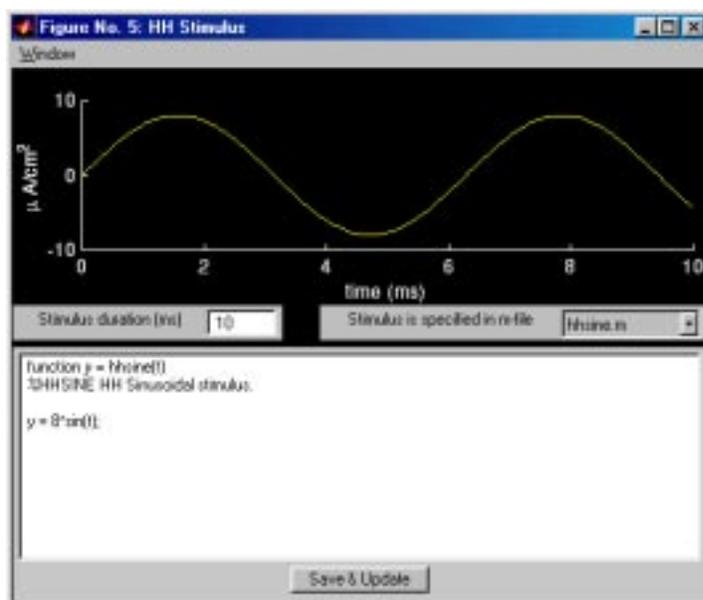

Figure 5.12: HH Stimulus figure showing an example of the use of an m-file to generate a sinusoidal stimulus.

Figure 5.13: HH Numerics figure.

5.4.4 Numerics

Selecting *Numerics* in the controls figure displays the *HH Numerics* figure (Figure 5.13). This figure allows selection of the numerical method used to solve the ordinary differential equations (or ode solver) as well as the numerical parameters of the simulation. The dialog-box that appears at the top allows direct selection of one of seven ode solvers called: ode45, ode23, ode113, ode23t, ode15s, ode23s, and ode23tb. The MATLAB manuals or MATLAB's help facility should be consulted for further information about these methods. For example, to learn about ode15s type `Help ode15s` at MATLAB's command line in the *MATLAB Command Window*. More detailed descriptions are available elsewhere (Shampine and Reichelt, 1997). By default, the software uses the ode15s ode solver. The integration methods require that the following parameters be defined: `Initial step-size, Maximum step-size, Relative Tolerance`. The relative tolerance is an estimate of the accuracy of the solution, i.e., a tolerance of 0.001 implies that the error in the solution is less than 0.1% of the correct value. Decreasing the tolerance will increase the accuracy of the solution, but will increase the computation time.

The numerical parameters alone can be saved in a file or read from a file by clicking on *Save* or *Open*, respectively, in the *HH Numerics* figure.

5.4.5 Analysis

The software provides one numeric and two graphic displays of simulation results. These three displays are selected and deselected in the controls figure. Analysis is available after a simulation has been run if the parameters have not been changed after the simulation was run.

Plots versus time

Selecting *Plots vs. time* in the Analysis panel of the *HH Control* figure displays the *HH Plots vs. time* figure as shown in Figure 5.14. During a computation, this figure displays the following variables: V_m in current clamp configuration and m in voltage-clamp configuration. When the computation is completed, any vari-
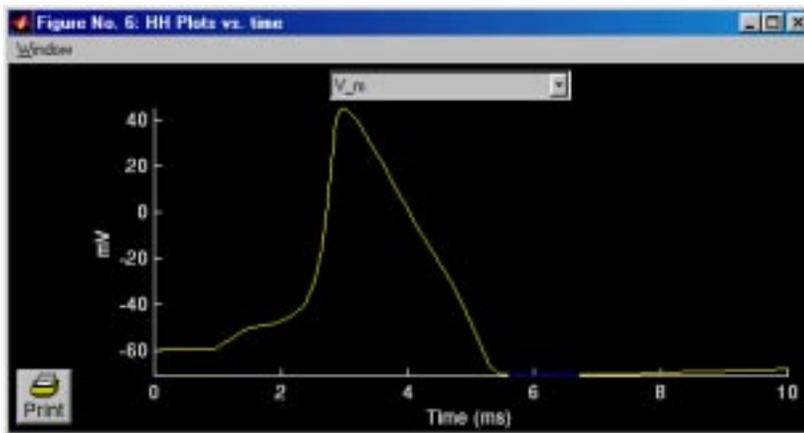
Figure 5.14: The HH Plots vs. time figure with V_m chosen for plotting.

Figure 5.15: The HH Variable summary figure after the simulation has been completed using the default parameters for a current-clamp configuration.

able can be selected by clicking on the dropdown menu at the top of the figure and then dragging the mouse to the variable of interest. This figure provides a simple and rapid means of plotting any of the simulation variables versus time. Clicking on an axis label displays the *Axis scale* figure (see Figure 5.8) which allows the axis to be changed.

Variable summary

Selecting *Variable summary* in the *HH Control* figure displays the *HH Variable summary* figure as shown in Figure 5.15. This figure contains five columns of text. The first column identifies the simulation variable. The second shows the units for the variable. The third contains the initial value of the variable. The fourth and fifth columns contain the minimum and maximum values of the
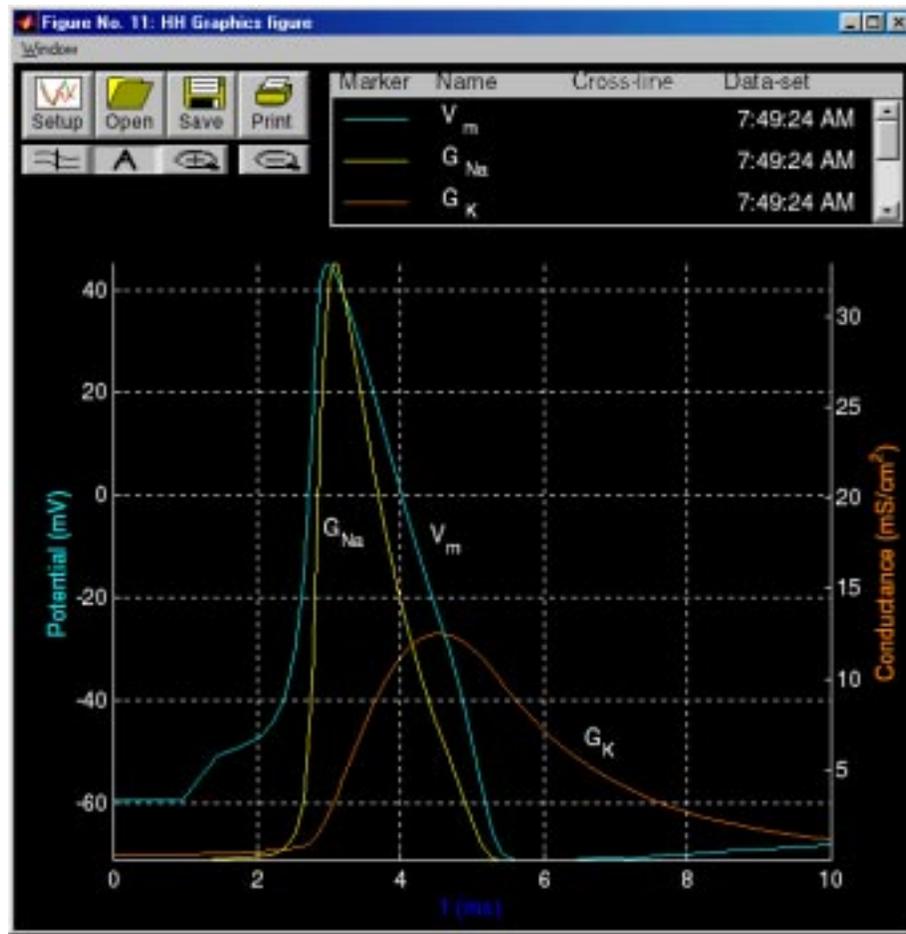
Figure 5.16: HH Graphics figure showing the membrane potential and ionic conductance obtained from a simulation with model parameters and the stimulus defined in Figures 5.4 and 5.10.

variables during the course of the simulation. The dropdown menus located below the menubar allow the user to control the precision and formatting of the numerical values.

Graphics

Selecting *Graphics* in the *HH Control* figure displays the *HH Graphics figure* as shown in Figure 5.16. This figure allows plotting any result of the simulation against any other in coordinates chosen by the user. The control of this figure is virtually identical to that for the *HH Parameters vs. potential* figure (Figure 5.5). Hence, we describe only the differences. The main difference occurs if *Setup* is selected in the *HH Graphics figure* which results in display of the *HH Setup plot* figure (Figure 5.17). Several dependent variables can be plotted simultaneously.
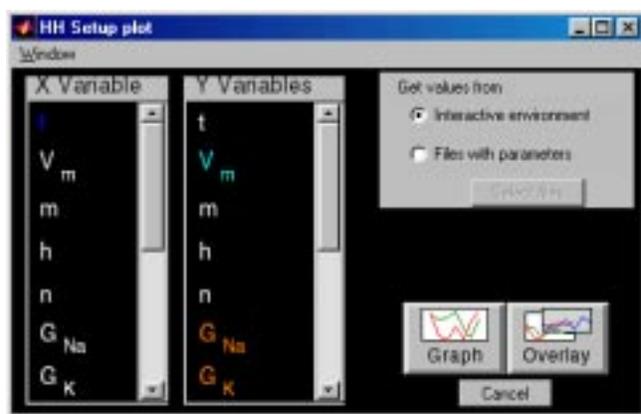
Figure 5.17: HH Setup plot figure showing the variables used to setup Figure 5.16.

In addition, results saved in files can be plotted so that results for different parameters can be compared. In addition, results of consecutive simulations can be overlayed. We give a few examples of the types of graphics that are readily available with the *HH Graphics figure*.

As shown in Figure 5.18, parametric plots of one variable versus another with time as the parameter are easily obtained. Figure 5.19 shows the membrane potential obtained at several different temperatures. This figure was obtained by overlaying successive computations. It could also have been created by saving the results of simulations performed for different temperatures in files and then plotting the results from these files.

5.4.6   Scripts

Scripting commands alleviate the burden of being present while repetitive, time consuming simulations are computed. The set of commands available to the software user can be found by issuing the command `help hhscr`.

Although the set of scripting commands is not exhaustive enough to allow operating all aspects of the software from the command-line, it is sufficient to carry out some of the time-consuming elements of the simulation.

The following is an example of a script used to perform current clamp simulations at different temperatures.

```
%A Hodgkin-Huxley script demonstration
%(contents are in HHDEMO1.M)


%This script can be used to generate simulation data using the
%current-clamped configuration.  The temperature is varied while
%all other parameters remain unchanged.

%For each value of the temperature, the simulation is run
%and the resulting data are saved to data files which can then
```
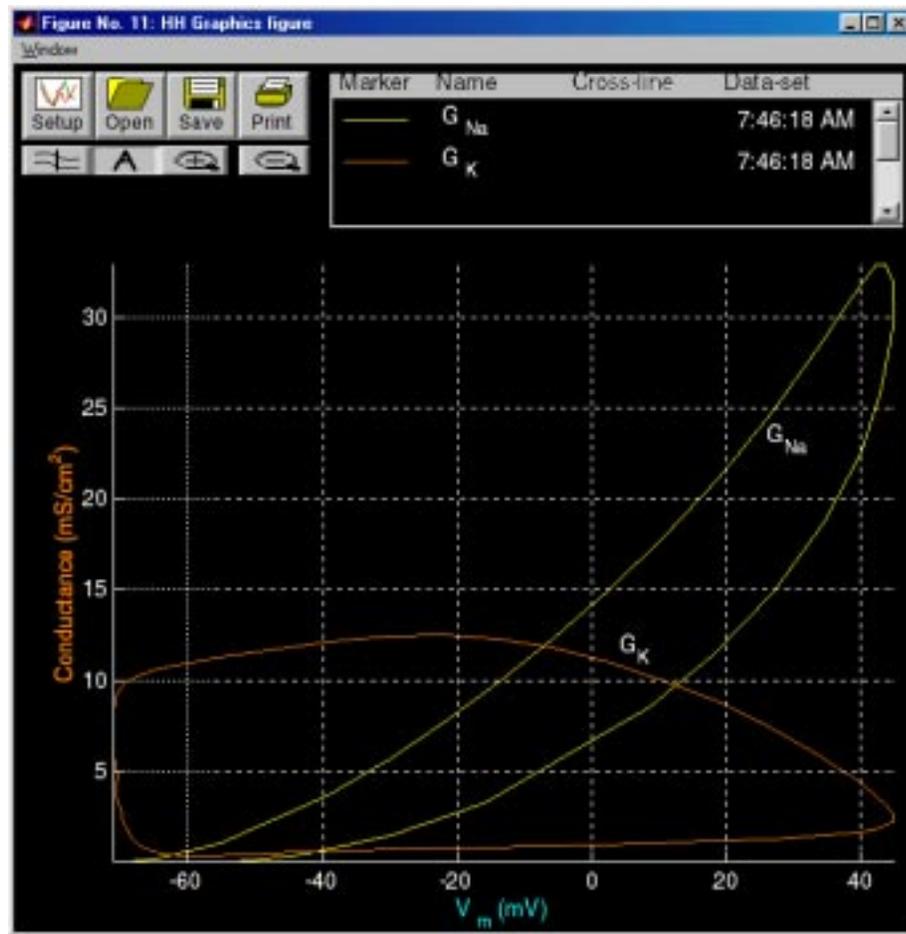
Figure 5.18: HH Graphics figure showing a parametric plot of membrane conductances versus membrane potential during an action potential. All the parameters were the default parameters shown in Figure 5.4 and the stimulus was the same as shown in Figure 5.10 except that the duration was increased to 20 ms.
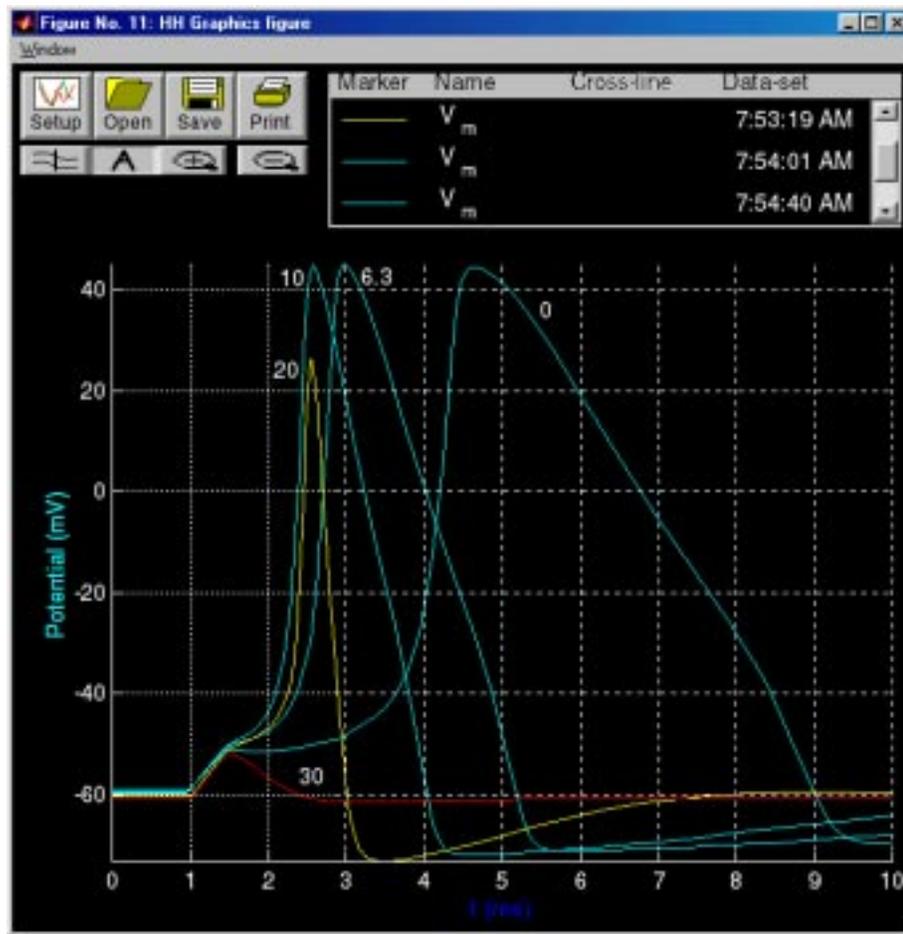
Figure 5.19: HH Graphics figure showing the membrane potential obtained at different temperatures. All the parameters were the default parameters shown in Figure 5.4 except for the temperature and the stimulus was the same as shown in Figure 5.10.

```
%be used to display the results in the Graphics figure.

%Before using the script, make sure that the Hodgkin-Huxley
%software has been started.
%If it hasn't been started, start the software using SOFTCELL.M
%as follows softcell('hh');

%hhscr('set') will list the parameters that can be changed
%The parameter names are case-sensitive, so make sure that
%you type them exactly as shown.

%Insure that the current clamped configuration is being used.
hhscr('set','Configuration','Current clamped');

%Since a lot of files will be created, it is best to store them
%in a separate folder.  Examples of foldername are
%foldername = 'C:\data\temp'; %Windows
%foldername = '/user/tsb/data/temp'; %UNIX
foldername = pwd;

%Set up a loop that increments through a temperature range.
%In this case, we use the range T = [0 10 20 30] Celsius
for temp = 0:10:30

  %set the model's temperature parameter
  newtemp = hhscr('set','T_C',temp);

  %perform simulation
  hhscr('start');

  %For feedback, display the temperature in the console
  disp(['Temperature is set to ' num2str(newtemp)]);

  %create a unique filename that contains the simulation data
  filename = ['temp' num2str(temp) '.mat'];

  %save the simulation data to the file
  hhscr('save','simulation',foldername,filename);
end

%That's it -- all the data have been saved.

%Now, display the data simultaneously by following these steps:
```

```
%1. call up the "Graphics" figure,
%2. using "Setup", select which variable(s) to plot,
%3. still inside "HH Setup plot" figure, select the source of "Get
%   values from" to be the "Files with parameters".
% 3a. using "Select files", add the files to the list "Use these
%     files".
% 3b. select "Overlay" to view the selected variable(s) in all
%     the datafiles.
```

Scripting can also be used to create customized plots of variables. For example, after running a simulation, the time dependance of the sodium conductance can be plotted on a separate figure using the following commands:

```
u_gna = hhscr('get','G\{ Na\}'); %get the sodium conductance vector
u_time = hhscr('get','t'); %get the vector of sampled time values
figure; %create a new figure
plot(u_time,u_gna); %plot sodium conductance vs. time
```

Once the values of these variables are so obtained, they can be manipulated, modified, and saved to files.

5.5 Problems

Problem 5.1 *The purpose of this problem is to explore the Hodgkin-Huxley model in the voltage-clamp configuration.* Use the default parameters of the Hodgkin-Huxley model under voltage-clamp conditions to complete the following problems. You may change only the stimulus parameters.

  a. The potassium conductance

    i. Find the amplitude and duration of a rectangular pulse of membrane potential such that the potassium conductance reaches its final of $G_K(V_m, \infty) = 9 \pm 0.2$ mS/cm^2.

    ii. What is the value of the time constant of n, τ_n, during this pulse?

  b. The sodium conductance

    i. Find the amplitude and duration of a rectangular pulse of membrane potential such that the peak sodium conductance is 15 ± 0.2 mS/cm^2. Make sure that the pulse duration is sufficient so that there is a peak in the sodium conductance before the pulse terminates.

    ii. What are the values of m and h at the time of the peak value of the sodium conductance?

iii. Increase the amplitude of the pulse 20 mV from the value you found above and determine the peak sodium conductance and the values of m and h at the time of the peak value of the sodium conductance.

iv. Decrease the amplitude of the pulse 20 mV from the value you found above and determine the peak sodium conductance and the values of m and h at the time of the peak value of the sodium conductance.

Problem 5.2 *The purpose of this problem is to explore the differences in time dependence of the [in]activation factors in voltage-clamp and current-clamp experiments.*

a. Run the simulation in the voltage-clamp configuration using the default parameters, except delay the onset of the pulse of voltage by 1 ms to make the on and off transients more readily visible. Examine the results of the simulation and answer the following questions.

   i. Describe and explain the dependence on time of τ_m, m_∞, and m.

   ii. Describe and explain the dependence on time of τ_h, h_∞, and h.

   iii. Describe and explain the dependence on time of τ_n, n_∞, and n.

   iv. Plot the voltage dependence of τ_m, τ_h, τ_n, m_∞, h_∞, and n_∞. Check the numerical values of these variables against those obtained in parts i, ii, and iv.

b. Run the simulation in the current-clamp configuration using the default parameters. Examine the results of the simulation and answer the following questions.

   i. Describe and explain the dependence on time of τ_m, m_∞, and m.

   ii. Describe and explain the dependence on time of τ_h, h_∞, and h.

   iii. Describe and explain the dependence on time of τ_n, n_∞, and n.

c. Succinctly summarize the differences in the time dependence of all these variables in voltage-clamp and current-clamp configurations.

Problem 5.3 *The purpose of this problem is to explore a useful aspect of performing simulation experiments on a model; the model can be used to test ideas concerning proposed measurements on cells.*

Assume that the membrane of neuron X has voltage-current characteristics as described by the Hodgkin-Huxley model except that the voltage dependences of the parameters of the factors n, m, and h are unknown. In this problem you are asked to develop methods to estimate the voltage dependence of some of these parameters and to assess the accuracy of your methods. To make the

simulations realistic, in all parts of the problem you may only change the following parameters: the concentrations of sodium, potassium, and calcium either inside or outside the cell, and the temperature. You may use either voltage-clamp experiments or current clamp experiments. You can choose any stimulus waveform that you can construct with the simulation. However, you must base your estimation method on the membrane current in voltage clamp experiments and/or on the membrane potential in current-clamp experiments. For example, you cannot base your estimation method on a direct measurement of $m(t)$.

a. Develop a method to estimate the voltage dependence of m_∞. Use the simulation to assess the accuracy of your method by comparing the results of your estimation of m_∞ with a direct computation of m_∞.

b. Develop a method to estimate the voltage dependence of h_∞. Use the simulation to assess the accuracy of your method as in part a.

Problem 5.4 *The purpose of this problem is to explore the threshold for eliciting an action potential for a model of a space-clamped axon.* By changing the amplitude of the stimulus pulse, while keeping all other parameters at their default values, determine the *threshold* of the action potential to a precision of 4 significant digits.

a. Describe the changes in action potential waveform as a function of current amplitude.

b. Does the change in waveform of the action potential as a function of stimulus amplitude violate the *all-or-none* principal? Explain.

c. Investigate the basis of the change in waveform of the action potential by examining the behavior of variables in addition to the membrane potential as the current amplitude is varied.

Problem 5.5 *The purpose of this problem is to explore the refractory properties of the Hodgkin-Huxley model.* Apply two current pulses, each of amplitude 30 μA/cm^2 and duration 0.5 ms, separated by 12 ms in time. Set the simulation duration to 20 ms. Run the simulation. You should observe two action potentials. Reduce the inter-pulse interval to 10 ms. What happens? Now change the amplitude of the second pulse to 50 μA/cm^2 and observe the response. Change the inter-pulse interval to 5 ms and observe the outcome.

a. Have you observed properties of the *absolute* or the *relative* refractory period?

b. By examining the behavior of the factors m, n, and h during these two-pulse simulations, determine which of these factors are most influential in refractory properties.

Problem 5.6 *The purpose of this problem is to explore the phenomena of repetitive activity and depolarization block of the Hodgkin-Huxley model.* Perform a sequence of simulations with the following parameters. Use a single current pulse of duration 40 ms, a simulation duration of 50 ms, and stimulus amplitudes of 10, 25, 50, 100, and 200 μA/cm^2.

a. Note that both the number of action potentials and their amplitudes vary with the current amplitude. Sketch the relation between the number of action potentials and current amplitude and between action potential amplitude and current amplitude.

b. By examining the behavior of the factors m, n, and h determine which of these factors are most influential in determining repetitive activity. Explain your reasoning.

Problem 5.7 *The purpose of this problem is to explore the phenomenon of anode-break excitation.* The phenomenon can be demonstrated by applying a pulse of current of duration 20 ms and amplitude -20 μA/cm^2. Use a simulation duration of 40 ms.

a. Why is this phenomenon called *anode-break* excitation?

b. By exploring the model variables explain this phenomenon.

Problem 5.8 *The purpose of this problem is to explore the phenomenon of accommodation.* Accommodation can be illustrated by applying a ramp of current of slope 0.4 μA/(cm^2 · ms) starting at an amplitude of 0 for a duration of 6 seconds (also set the simulation duration to 6 seconds). The model does not show accommodation for the default value of $\bar{G}_{Na} = 120$ mS/cm^2 but does for a value of 80 mS/cm^2. When the simulation is completed, plot the sodium, potassium and membrane current versus the membrane potential.

a. Perhaps no action potential has occurred in response to the ramp of current because the reduced value of the sodium conductance prevents action potential from being generated for any stimulus. Test this hypothesis.

b. Explain the underlying basis of accommodation.

c. What does the phenomenon of accommodation suggest about the notion of the *threshold* for eliciting an action potential?

Problem 5.9 *The purpose of this problem is to explore sub-threshold oscillations in the Hodgkin-Huxley model.* Sub-threshold oscillations can be observed by using the default parameters with current pulses of 50 ms duration and simulation durations of 100 ms. Use current amplitudes of .01, .1, 1, 2, and 2.5 μA/cm^2.

a. Describe your observations.

b. Can the response of the model to a step of current be represented by the response of a network consisting of a constant resistance in parallel with a constant capacitance. Explain.

c. If your answer to part b was no, then devise a current stimulus such that the model's response will approximate that of a network consisting of a constant resistance in parallel with a constant capacitance. Explain your method.

Problem 5.10 *The purpose of this problem is to explore the strength-duration relation of the Hodgkin-Huxley model.* Determine the threshold current amplitudes (to three significant figures) for eliciting action potentials for current pulses of duration 0.01, 0.05, 0.1, 0.5, 1, 2, 5, 10, and 20 ms.

a. How does the threshold depend on current duration for low durations? Explain this behavior.

b. How does the threshold depend on current duration for long durations? Explain this behavior.

Problem 5.11 *The purpose of this problem is to explore the interplay of factors that determine the excitation of an action potential in the Hodgkin-Huxley model.* The Hodgkin-Huxley model with the default parameters stimulated by a pulse of membrane current of amplitude 20 μA/cm^2 and of duration 0.5 ms produces an action potential as shown in Figure 5.20. In each of the following computations, one membrane parameter only differs from the default parameters, and for each of these computations no action potential occurs. The parameter changes are:

i. A change in the temperature from 6.3° C to 35° C blocks the action potential as shown in Figure 5.20.

ii. A change in the maximum potassium conductance from 36 to 72 mS/cm^2 blocks the action potential. [Results not shown.]

iii. A change in the internal calcium concentration from 0.0001 to 0.0002 mmol/l blocks the action potential. [Results not shown.]

iv. A change in the membrane capacitance from 1 μF/cm^2 to 20 μF/cm^2 blocks the action potential as shown in Figure 5.20.

v. A change in the external calcium concentration from 44 mmol/l to 22 mmol/l blocks the action potential. [Results not shown.]
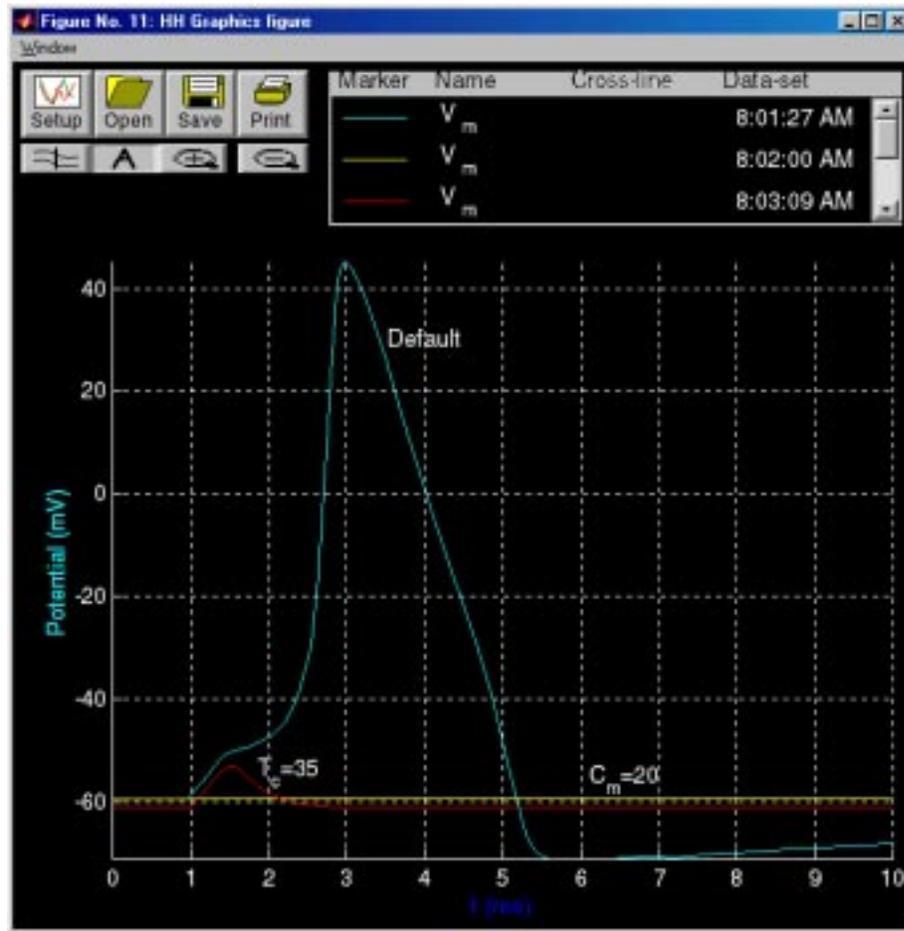
Figure 5.20: Results of simulation with all model parameters at their default values except where indicated (Problem 5.11). The curve labelled *Default* was obtained with the default parameters. The curve labeled $T_c = 35$ was obtained with the default parameters except that the temperature was set to $35°$ C. The curve labeled $C_m = 20$ was obtained with the default parameters except that the membrane capacitance was set to $20 \ \mu F/cm^2$.

Examination of the results for the above computations (e.g., Figure 5.20) reveals that there is no large change in membrane potential, i.e., no action potential has occurred. For each of the computations:

a. Explain why no action potential occurred. You may wish to base your explanation on the results obtained with each of the above computations. For example, you may wish to examine the membrane currents, conductances, and activation variables that occurred for each computation. You may also wish to perform additional computations using different model parameters. For example, to see why no action potential occurs for a large temperature change, it might be instructive to examine the effects produced by smaller changes in temperature. Since for many of the parts of this problem, there are large changes in parameters you may wish to explore the role of the numerical methods in arriving at the solutions.

b. Determine a set of parameters so as to produce an action potential with a waveform that is identical, *except for a change in time scale and a change in resting potential*, to the action potential obtained with the default parameters. However, for each of the above parts, you may not change the parameter that differs from its default value. For example, for computation in part i, you may not change the temperature. Your task is to change some other parameter or group of parameters such that the action potential is restored. For computation in part ii, you may not change the maximum potassium conductance. For computations in parts iii and v, you may not change either the internal or the external calcium concentration, etc. REMEMBER — the objective is not just to produce an action potential, that is too easy, but to obtain an action potential whose waveform is **identical** to that obtained with the default parameters. To determine whether or not the action potential is identical to that obtained with the default parameters, you can superimpose plots and/or look at the variables summary figures from the responses. You should use the computer as a tool to check your ideas and not as a substitute for thinking. You should avoid a strictly trial-and-error approach. There are simply too many parameters in the Hodgkin-Huxley model for you to explore them all randomly. When you have arrived at a satisfactory solution, print a hard copy of the model parameters figure showing the parameters you used to solve the problem and include this with your solution. Explain why your parameter change produces the desired result. [HINT: simulations are not required to solve these problems, but they are helpful for checking ideas quickly.]

5.6   PROJECTS

The simulation software lends itself to more extensive study than that included in the Problems. These more extensive studies are called Projects and these are described here.

5.6.1   Practical considerations in the choice of a topic

Projects can involve almost any of the properties of the Hodgkin-Huxley model. However, to avoid projects whose aims are vague (e.g., "I would like to understand how the Hodgkin-Huxley model works") the proposed project should be in the form of a specific and testable hypothesis. Projects that involve months of computation should obviously be avoided. Experience should indicate the duration required to run simulations of the Hodgkin-Huxley model. These durations should be taken into account explicitly in planning a project. For example, any project that involves measuring the threshold of occurrence of an action potential for many different parameter values is bound to be very time consuming, because determining the threshold for a single set of parameters itself involves many computations. The task is to choose a physiological property of the excitation of the action potential that is of interest, and then to define a specific, feasible project.

5.6.2   Choice of topics

Topics can involve comparing predictions of the Hodgkin-Huxley model with measurements on cells. For example, the effects of changes in sodium ion concentration on the action potential have been measured (Hodgkin and Katz, 1949a; Baker et al., 1961). A project might involve reading the original papers that describe such measurements (some were made before the Hodgkin-Huxley model was formulated), and testing the hypothesis that these measurements are (or are not) consistent with the Hodgkin-Huxley model. Similarly, a project might involve examining the effect of some pharmacological substance on measurements of the action potential and testing the hypothesis that the substance produces its effect by changing one or another parameter of the model. These projects will require some reading of original literature which is often difficult and usually time consuming. However, such a project can lead to a very rewarding educational experience. Alternatively, the project might involve a purely theoretical topic in which some property of the model is explained in terms of the underlying structure of the model. This type of project does not necessarily involve reading the original literature. To make the process of project definition more tangible, we shall discuss one property of the Hodgkin-Huxley model of a space-clamped axon, the shape of the action potential, and suggest some appropriate projects based on this property.
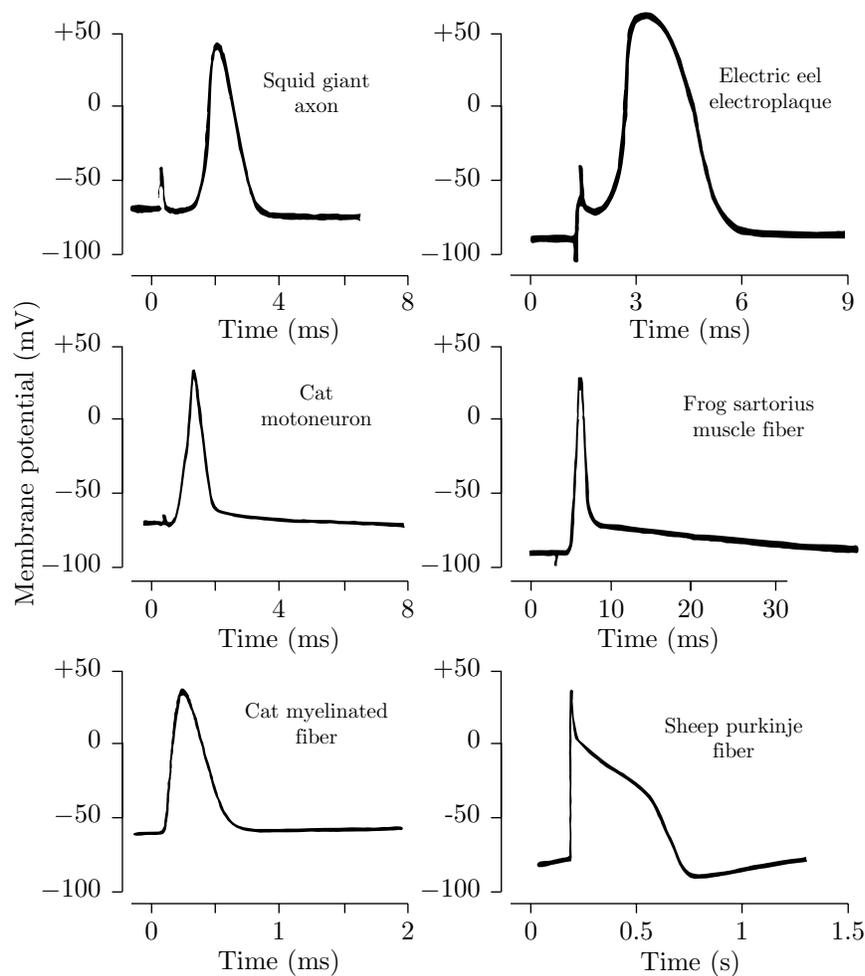
Figure 5.21: Examples of action potentials measured in a variety of cell types (Keynes and Aidley, 1991, adapted from Figure 2.4).

Background

The waveform of the action potential is different in different types of electrically-excitable cells. As demonstrated in Figure 5.21, while the action potential waveform has a peak-to-peak amplitude of about 100 mV, the duration and time course vary greatly. It is also known that the action potential waveform of a cell can be changed by changing a number of experimental variables. For example, increasing the temperature tends to decrease the duration of the action potential (Figure 5.22), and reducing the external sodium concentration reduces the amplitude and increases the duration of the action potential (Figure 5.23). Since action-potential waveforms vary from cell to cell and vary for one cell if environmental variables are changed, this raises a question — *To what extent are changes in action-potential waveforms seen in different cells due to inherent differences in the membrane mechanisms rather than to differences in membrane parameters with identical membrane mechanisms?* For example, the action po-
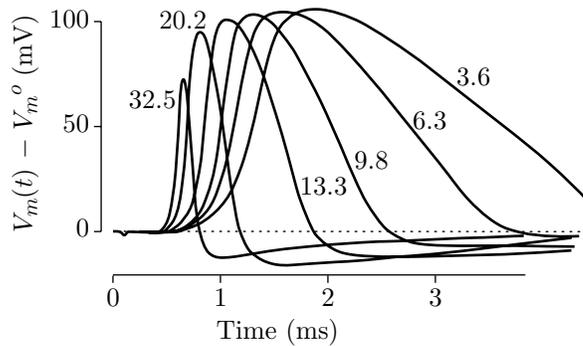
Figure 5.22: Dependence of the action potential on temperature for measurements obtained from a giant axon of the squid (Hodgkin and Katz, 1949b).
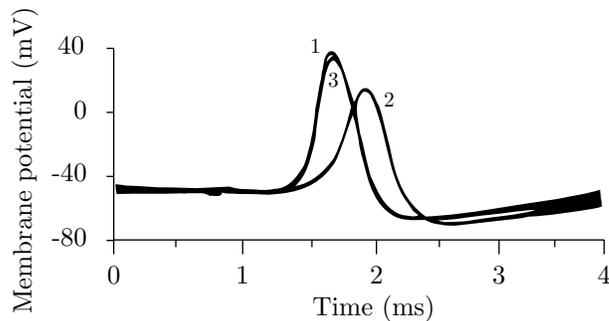
Figure 5.23: Effect of extracellular sodium concentration on the membrane potential of a giant axon of a squid (Hodgkin and Katz, 1949a, adapted from Figure 4). Traces 1 and 3 are the action potentials obtained in normal sea water before and after trace 2 which was obtained in sea water with 50% sodium content. Isotonic solutions were obtained by using mixtures of sea water and a glucose solution.

tentials of the squid axon and the frog and rat nodes of Ranvier shown in Figure 5.21 were measured at different temperatures. In fact, the briefest action potentials were obtained at the highest temperatures. Since Figure 5.22 indicates that the duration of the action potential *does* become shorter at higher temperatures in squid axon, it may be that the temperature differences in the measurements could explain the differences in the waveforms of the action potentials of these three preparations. It is clear that temperature changes alone cannot explain all the variations seen in Figure 5.21 since the long-duration action potential of frog cardiac muscle was obtained at a higher temperature than was the action potential of the squid giant axon. However, perhaps some combination of changes of parameters in addition to temperature (e.g., ion concentrations, ion conductances, membrane capacitance, etc.) might explain the long-duration action potential of cardiac muscle.

Examples of hypotheses

The list of projects given below includes a number that deal with the waveform of the action potential generated by the Hodgkin-Huxley model as well as a few that deal with other topics.

1. Hypothesis — The difference in waveform of the action potential of a frog node of Ranvier and of a squid giant axon (Figure 5.21) can be reproduced by the Hodgkin-Huxley model of a squid giant axon by a change in temperature.

2. Hypothesis — The membrane capacitance determines the time course of the rising phase of the action potential. Increasing the membrane capacitance decreases the rate of increase of the rising phase of the action potential.

3. Hypothesis — The falling phase of the action potential (repolarization) can occur in the absence of a change in potassium conductance.

4. Hypothesis — Increasing the temperature sufficiently blocks the occurrence of the action potential because the membrane time constant limits the rate at which the membrane variables can change and prevents the difference in time course of the sodium and potassium activation which is responsible for initiation of the action potential.

5. Hypothesis — The initiation of the action potential is independent of the potassium conductance.

6. Hypothesis — The prolonged plateau of the cardiac muscle action potential can be accounted for by the Hodgkin-Huxley model with a potassium conductance that has a slow activation.

7. Hypothesis — The effect of tetraethylammonium chloride (TEA) on the action potential of the squid giant axon can be modelled with the Hodgkin-Huxley model by decreasing K_n and increasing K_h. Articles in the literature should be consulted for this project (Armstrong, 1966; Armstrong and Binstock, 1965; Tasaki and Hagiwara, 1957).

8. Hypothesis — The shape of the action potential in the presence of tetraethylammonium chloride (TEA) can be accounted for by the Hodgkin-Huxley model with a reduced maximum value of the potassium conductance. Articles in the literature should be consulted for this project (Armstrong, 1966; Armstrong and Binstock, 1965; Tasaki and Hagiwara, 1957).

9. Hypothesis — Increasing the external calcium concentration will block the occurrence of the action potential because this will reduce the difference in the time constant of sodium and potassium activation which is responsible for the initiation of the action potential.

10. Hypothesis — Increasing the external concentration of potassium will decrease the refractory period; decreasing this concentration will lengthen the refractory period.

11. Hypothesis — Increasing the external concentration of sodium will decrease the refractory period; decreasing this concentration will lengthen the refractory period.

12. Hypothesis — Absolute and relative refractory periods are decreased by increasing the rate constants for sodium inactivation and for potassium activation.

13. Hypothesis — Repolarization cannot occur if the potassium activation rate constant is zero.

14. Hypothesis — The threshold of the action potential to a brief pulse of current decreases as the external potassium current is increased.

15. The Hodgkin-Huxley model with default parameters does not exhibit accommodation (Weiss, 1996b). Hypothesis — Accommodation occurs if the leakage conductance is increased.

16. The Hodgkin-Huxley model with default parameters does not exhibit accommodation (Weiss, 1996b). Hypothesis — Accommodation occurs if the potassium conductance is increased.

17. Hypothesis — Increasing the leakage equilibrium potential will block the action potential.

18. Hypothesis — The effect of the changes in concentration of sodium ions on the action potential of the giant axon of the squid can be accounted for by the Hodgkin-Huxley model. Articles in the literature should be consulted for this project (Hodgkin and Katz, 1949a; Baker et al., 1961).

19. Hypothesis — In response to rectangular pulses of current, the rheobase of the strength-duration relation increases as temperature increases.

20. Hypothesis — An increase in temperature results in a decrease in the duration of the refractory period.

21. Hypothesis — The threshold membrane potential at which the Hodgkin-Huxley model produces an action potential in response to a brief pulse of current is equal to the membrane potential for which the linearized Hodgkin-Huxley equations have unstable eigenvalues.

22. Application of a long-duration constant current to the Hodgkin-Huxley model produces a train of action potentials. Hypothesis — The frequency of the action potentials increases with increasing current amplitude.

23. Application of a long-duration constant current to the Hodgkin-Huxley model produces a train of action potentials. Hypothesis — The frequency of action potential increases as the parameter K_n is increased.

24. Application of a long-duration constant current to the Hodgkin-Huxley model produces a train of action potentials. Hypothesis — The frequency of action potential increases as the temperature is increased.

25. Hypothesis — An increase in the external concentration of potassium increases the threshold potential at which an action potential is elicited.

26. Hypothesis — Increasing K_h will result in an increase in the steepness of the repolarization phase of the action potential.

Caveats

Any of these (or other) hypotheses can be the starting point for a project. Most of the hypotheses given above are simplistic, and a careful investigation will reveal their shortcomings. The Hodgkin-Huxley model is sufficiently complex that investigation of any of the hypotheses will most likely lead to unexpected results. It is informative to pursue these unexpected results and to try to understand their bases. For example, you may find that in pursuing some hypothesis you choose to change some parameter of the model that you expect to result in some change in action potential waveform. The resulting computation might reveal, much to your surprise and chagrin, that no action potential has occurred

or that multiple action potentials occur. Investigation of these unexpected findings will usually be instructive. The aim should be not simply to reject or accept the hypothesis but to delve into the topic in sufficient depth so as to a deepen understanding of the model. One outcome of the project might be to restate the original hypothesis in a new and more sophisticated form.

In pursuing these projects, it is important to keep in mind that it is not nerve membrane that is being investigated but rather a model of nerve membrane; in fact, a very specific model — the Hodgkin-Huxley model for the membrane of the squid giant axon. Explanations of all phenomena seen with the model must involve the primitive concepts of this model — e.g., the ionic conductances, ionic concentrations, ionic currents, the capacitance, and the variables m, n, and h. Explanations in terms of molecular channel mechanisms or electrodiffusion of ions in the membrane or ion pumps are irrelevant in so far as they are not contained in the Hodgkin-Huxley model! Finally, it is important to remember that the software performs a numerical integration of the Hodgkin-Huxley equations. Thus, the user will have to contend with the inherent approximations involved in numerical solutions of equations.

Chapter 6

HODGKIN HUXLEY MODEL — PROPAGATED ACTION POTENTIAL

6.1   Introduction

6.1.1   Background

An electrically excitable cell produces a difference of potential across its cell membrane, called an action potential. The difference in membrane potential is accompanied by a flow of current longitudinally through the intracellular and extracellular media, as well as current flow across the cell membrane along its entire length. The spatial coupling of current flow through neighboring segments of the membrane results in the propagation of the action potential along the cell surface.

In the 1950s, Hodgkin and Huxley derived a partial differential equation that effectively describes the electrical properties of the giant axon of the squid (Hodgkin and Huxley, 1952). This partial differential equation consists of the core conductor equations, that describe the relation of current and voltage for an electrical cable, plus a characterization of the electrical properties of the cell membrane. The cell is assumed to be cylindrical and the inner and outer conductors are assumed to be equipotentials at each longitudinal position. Numerical solutions to the partial differential equation were originally computed tediously by hand (Hodgkin, 1977; Huxley, 1964). With the introduction of the digital computer in the 1950s, solutions to these equations were computed more rapidly (Fitzhugh and Antosiewicz, 1959). By the 1960's computer speeds had increased to allow the first comprehensive study of the solutions to the partial differential equations (Cooley and Dodge, 1966). Further increases in computer speeds and the development of improved numerical methods (Moore et al., 1975; Joyner et al., 1978; Hines, 1984), have allowed solutions to be obtained on personal computers. The software package described here yields numerical solutions of the partial differential equation that describe the propagation of the action potential along a model of the squid giant axon.

6.1.2   Overview of the software

This software deals with the time and space dependent Hodgkin-Huxley model and is complementary to the software described in Chapter 5. The software described in the current chapter is comprised of a number of setup and analysis figures that are launched from a main control figure. Using the setup figures, the user can specify the physical characteristics of the axon (length, radius, etc.) and parameters of the membrane (capacitance, conductances of membrane ionic channels, kinetic parameters, etc.) and bath (concentration of all relevant ions, temperature, etc.). A stimulus source can be customized to the user's specifications, and graphic icons of a pair of stimulus electrodes can be visually placed along a schematic representation of an axon. In addition, graphic icons representing a pair of recording electrodes can be placed along the axon

schematic to compute and display the difference in potential. When a simulation is started, solutions are calculated using one of four user selectable numerical methods, and the time-evolution of the membrane potential is shown color coded to convey the space-time evolution of potentials on the axon schematic. After a simulation is completed, the user can display any of the computed solution variables (such as membrane potential, membrane-current components, membrane-conductance components, channel activation and inactivation variables, etc.) using different interactive analysis figures which include: a figure that displays the space-time evolution of any solution variable, a figure that generates three-dimensional plots to show the dependence on time and on space of any solution variable, and a figure that generates comparison plots between the solution variables at fixed times or at fixed positions.

6.2 Theory

The theory of the propagation of an action potential along an unmyelinated axon was first described for the squid giant axon (Hodgkin and Huxley, 1952). We briefly review this theory here; a more detailed description is available elsewhere (Weiss, 1996b).

6.2.1 Core conductor equations

An incremental portion of the core conductor model of a cylindrical unmyelinated nerve fiber is shown in Figure 6.1. Kirchhoff's laws relate the variables of this incremental model. The core conductor equations result (Weiss, 1996b) if the limit of these equations is taken as the incremental length of the fiber goes to zero, $\Delta z \to 0$,

$$\frac{\partial I_i(z,t)}{\partial z} = K_{ei}(z,t) - K_m(z,t), \tag{6.1}$$

$$\frac{\partial I_o(z,t)}{\partial z} = K_m(z,t) - K_{eo}(z,t), \tag{6.2}$$

$$\frac{\partial V_i(z,t)}{\partial z} = -r_i I_i(z,t), \tag{6.3}$$

$$\frac{\partial V_o(z,t)}{\partial z} = -r_o I_o(z,t), \tag{6.4}$$

$$V_m(z,t) = V_i(z,t) - V_o(z,t), \tag{6.5}$$

$$\frac{\partial V_m(z,t)}{\partial z} = -r_i I_i(z,t) + r_o I_o(z,t). \tag{6.6}$$

These equations can be combined to obtain a relation between the membrane potential V_m and the membrane current density J_m at time t and location z along
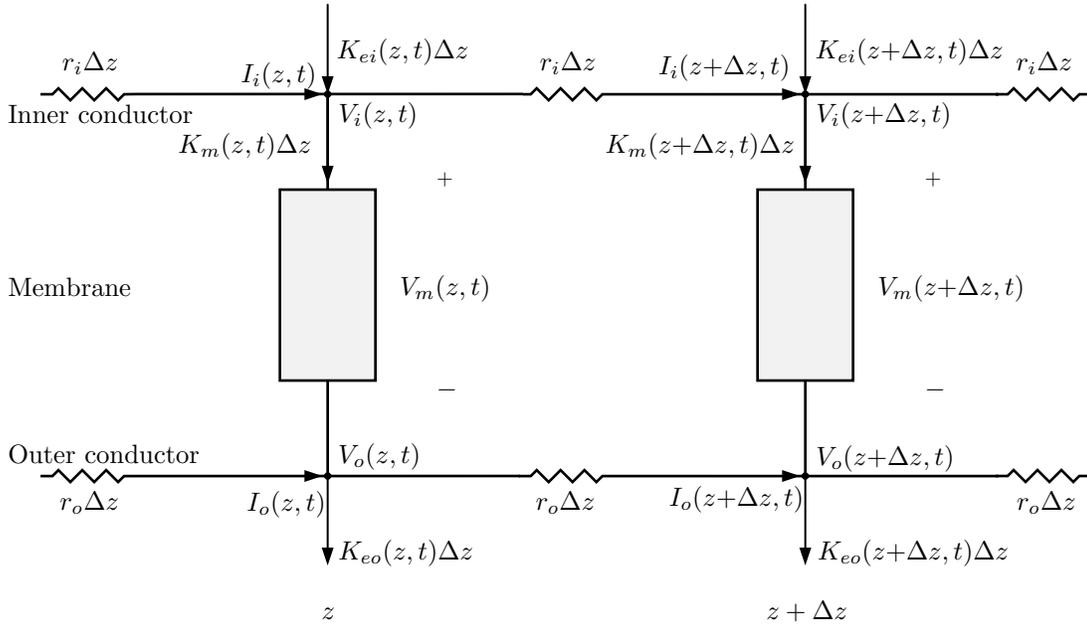
Figure 6.1: Equivalent network model of incremental sections of an unmyelinated nerve fiber showing the reference directions for all the voltage and current variables. The Ks are currents per unit length of the fiber: K_m is the membrane current; K_{eo} and K_{ei} are extrinsic currents applied (by electrodes) externally and internally, respectively. I_i and I_o are the internal and external longitudinal currents. V_m is the potential across the membrane and V_i and V_o are the internal and external potentials, respectively. The internal and external resistance per unit length of the fiber are r_i and r_o.

a cylindrical, unmyelinated nerve fiber, often called the *core-conductor equation*

$$\frac{1}{2\pi a(r_o + r_i)}\frac{\partial^2 V_m(z,t)}{\partial z^2} = J_m(z,t) - \frac{1}{2\pi a(r_o + r_i)}(r_o K_{eo}(z,t) + r_i K_{ei}(z,t)),$$

where a is the radius of the cylindrical cell and $K_m = 2\pi a J_m$, where J_m is the membrane current per unit area. In addition, the resistance per unit length of cytoplasm can be expressed in terms of the resistivity as follows

$$r_i = \frac{\rho_i}{4\pi a^2}$$

where ρ_i is the resistivity of cytoplasm.

6.2.2 Representation of stimulating electrodes

An arrangement of stimulating electrodes is shown in Figure 6.2. From Kirchhoff's current law we have that at any position along the fiber

$$I_i(z,t) + I_o(z,t) + I_s(z,t) = 0, \tag{6.7}$$
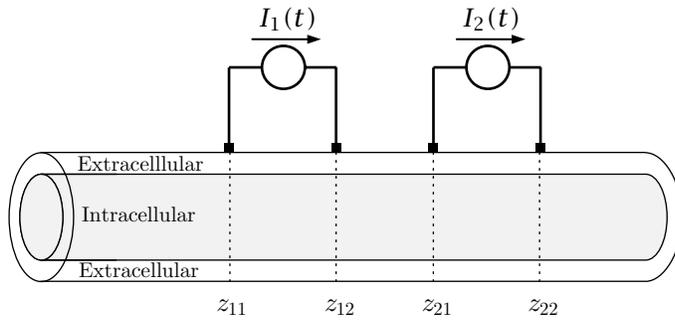
Figure 6.2: The relation of two pairs of stimulus electrodes to a fiber is shown. The stimulus electrodes are shown as extracellular but any of the electrodes can be intracellular or extracellular.

where $I_s(z,t)$ is the stimulus current in the positive z-direction which can be written as

$$I_s(z,t) = I_1(t)\left(u(z-z_{11}) - u(z-z_{12})\right) + I_2(t)\left(u(z-z_{21}) - u(z-z_{22})\right),$$
(6.8)

where $u(z)$ is the unit step function defined as

$$u(z) = \begin{cases} 1 & \text{if } z > 0, \\ 0 & \text{if } z < 0. \end{cases}$$

The electrode configuration shown in Figure 6.2 also imparts current components to Equations 6.1 and 6.2. With all the electrodes external, the currents per unit length are as follows

$$\begin{aligned} K_{eo}(z,t) &= I_1(t)\left(\delta(z-z_{11}) - \delta(z-z_{12})\right) + I_2(t)\left(\delta(z-z_{21}) - \delta(z-z_{22})\right), \\ K_{ei}(z,t) &= 0, \end{aligned}$$

where $\delta(z)$ is the unit impulse function.

If all four of the electrodes were intracellular then the currents per unit length would be

$$\begin{aligned} K_{eo}(z,t) &= 0, \\ K_{ei}(z,t) &= -I_1(t)\left(\delta(z-z_{11}) - \delta(z-z_{12})\right) - I_2(t)\left(\delta(z-z_{21}) - \delta(z-z_{22})\right), \end{aligned}$$

With careful attention to signs, any combination of intracellular and extracellular stimulating electrodes can be accommodated. The algorithm is that extracellular electrodes supply positive impulses at the cathode and negative impulses at the anode; intracellular electrodes supply negative impulses at the cathode and positive impulses at the anode.

6.2.3 Longitudinal currents

By using the core conductor equations (Equation 6.6) and the KCL equation for the current (Equation 6.7), we can solve for the longitudinal currents as follows

$$I_i(z,t) = -\frac{1}{r_i + r_o}\frac{\partial V_m(z,t)}{\partial z} - \frac{r_o}{r_i + r_o}I_s(z,t),$$

$$I_o(z,t) \quad = \quad \frac{1}{r_i + r_o}\frac{\partial V_m(z,t)}{\partial z} - \frac{r_i}{r_i + r_o}I_s(z,t).$$

6.2.4   Intracellular and extracellular potential differences

The expressions for the longitudinal currents can be combined with Equations 6.3 and 6.4 and the result integrated on z to yield the potential differences of a recording electrode at z_{r2} minus that at z_{r1} as follows

$$
\begin{aligned}
V_i(z_{r2},t) - V_i(z_{r1},t) \quad &= \quad \frac{r_i}{r_i + r_o}\left(V_m(z_{r2},t) - V_m(z_{r1},t)\right) \\
&\quad + \frac{r_i r_o}{r_i + r_o}\int_{z_{r1}}^{z_{r2}} I_s(z,t)\,dz, \\
V_o(z_{r2},t) - V_o(z_{r1},t) \quad &= \quad -\frac{r_o}{r_i + r_o}\left(V_m(z_{r2},t) - V_m(z_{r1},t)\right) \\
&\quad + \frac{r_i r_o}{r_i + r_o}\int_{z_{r1}}^{z_{r2}} I_s(z,t)\,dz,
\end{aligned}
$$

where use of Equation 6.8 results in

$$
\begin{aligned}
\int_{z_{r1}}^{z_{r2}} I_s(z,t)\,dz \quad &= \quad I_1(t)\left((z - z_{11})u(z - z_{11}) - (z - z_{12})u(z - z_{12})\right) \\
&\quad + I_2(t)\left((z - z_{21})u(z - z_{21}) - (z - z_{22})u(z - z_{22})\right).
\end{aligned}
$$

6.2.5   The membrane current density

The membrane current density consists of a capacitance current density and an ionic current density J_{ion} as follows,

$$J_m(z,t) = C_m\frac{\partial V_m(z,t)}{\partial t} + J_{ion}(z,t),$$

where C_m is the specific membrane capacitance, i.e., the capacitance of a unit area of membrane. For purposes of numerical solution of the core conductor equation, it is helpful to combine the ionic current density with the external currents so that

$$J(z,t) = J_{ion}(z,t) - \frac{1}{2\pi a(r_o + r_i)}(r_o K_{eo}(z,t) + r_i K_{ei}(z,t)), \qquad (6.9)$$

and, therefore, the core conductor equation can be expressed as

$$\frac{1}{2\pi a(r_o + r_i)}\frac{\partial^2 V_m(z,t)}{\partial z^2} = C_m\frac{\partial V_m(z,t)}{\partial t} + J(z,t). \qquad (6.10)$$

The ionic current density can be expressed in terms of its ionic components

$$J_{ion}(z,t) = G_K(V_m,t)(V_m - V_K) + G_{Na}(V_m,t)(V_m - V_{Na}) + G_L(V_m - V_L), \quad (6.11)$$

where V_n is the Nernst equilibrium potential of ion n

$$V_n = \frac{RT}{z_n F} \ln\left(\frac{c_n^o}{c_n^i}\right),$$

where R is the molar gas constant, T is absolute temperature, F is Faraday's constant, and z_n, c_n^o and c_n^i are the valence, outside, and inside concentrations of ion n, respectively. For a univalent ion, the Nernst equilibrium potential is expressed as

$$V_n = 0.08616\,(T_c + 273.16)\ln\left(\frac{c_n^o}{c_n^i}\right)\;(\text{mV}),$$

where T_c is the temperature in Centigrade.

6.2.6 The membrane conductances

A full description of the Hodgkin-Huxley model of the ionic conductances is given beginning with Section 5.2.3.

6.2.7 Default values of parameters

The default numerical values for the Hodgkin-Huxley model are: $\overline{G}_{Na} = 120$, $\overline{G}_K = 36$, and $G_L = 0.3$ mS/cm^2; $C_m = 1$ μF/cm^2; $c_{Na}^o = 491$, $c_{Na}^i = 50$, $c_K^o = 20.11$, $c_K^i = 400$, $c_{Ca}^o = 44$, $c_{Ca}^i = 0.00011$ mmol/L; $V_L = -49$ mV; external resistance per unit length, $r_o = 0$ Ohm/cm; resistivity of axoplasm, $\rho_i = 35.4$ Ohm·cm; axon radius, $a = 238$ μm; temperature is 6.3°C.

6.3 Numerical Method of Solution

There are several ways to obtain a numerical solution to the equations describing the membrane potential in a cylindrical cell. In the *finite-element method*, the cylindrical axon is segmented into small cylinders over each of which the membrane potential is assumed to be constant. Ordinary differential equations (with time as the independent variable) can be written to describe the membrane potential and membrane current for each segment and the set of ordinary differential equations can be solved numerically with appropriate boundary conditions linking the electrical variables in adjoining segments. In the *finite-difference method*, the partial differential equations are discretized directly by discretizing the membrane potential and membrane current and approximating the spatial and temporal derivatives with finite differences. We use the second method which is described in a number of texts on numerical methods of solution of partial differential equations (Smith, 1985; Press et al., 1986; Gerald and Wheatley, 1989) and in research papers on the solution of the Hodgkin-Huxley model

(Hines, 1984; Hines and Carnevale, 1998; Mascagni and Sherman, 1998; Pearlmutter and Zador, 1999).

The first step is to discretize the membrane potential and current density variables as follows,

$$
\begin{aligned}
V_i^n &= V_m(i\Delta z, n\Delta t) = V_m(z,t)|_{z=i\Delta z, t=n\Delta t}\,, \\
J_i^n &= J(i\Delta z, n\Delta t) = J(z,t)|_{z=i\Delta z, t=n\Delta t}\,,
\end{aligned}
$$

where Δz and Δt are the increments in z and t.   The second spatial derivative is approximated by the centered difference (also known as the method of lines),

$$
\frac{\partial^2 V_m(z,t)}{\partial z^2} \approx \frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{(\Delta z)^2}.
$$

All of the numerical methods implemented use the above centered difference approximation to the second spatial derivative. The methods differ only in the approximation of the temporal derivative. There are two simple ways to approximate the temporal derivative at time t. In the forward Euler approximation,

$$
\frac{\partial V_m(z,t)}{\partial t} \approx \frac{V_i^{n+1} - V_i^n}{\Delta t},
$$

whereas in the backward Euler approximation

$$
\frac{\partial V_m(z,t)}{\partial t} \approx \frac{V_i^n - V_i^{n-1}}{\Delta t}.
$$

The temporal approximation used in the Crank-Nicolson method essentially averages the forward Euler and the backward Euler approximations. The staggered increment Crank-Nicolson method is a variant of the Crank-Nicolson method with desirable numerical characteristics.

6.3.1   Forward Euler method

With the forward Euler approximation, the discretized core conductor equation (Equation 6.10) is

$$
\frac{1}{2\pi a(r_o + r_i)}\left(\frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{(\Delta z)^2}\right) = C_m\left(\frac{V_i^{n+1} - V_i^n}{\Delta t}\right) + J_i^n. \tag{6.12}
$$

By collecting terms, the forward Euler approximation leads to the following difference equations

$$
V_i^{n+1} = \eta V_{i+1}^n + (1 - 2\eta)V_i^n + \eta V_{i-1}^n - \xi J_i^n,
$$

where

$$
\eta = \frac{\Delta t}{2\pi a(r_o + r_i)(\Delta z)^2 C_m} \text{ and } \xi = \frac{\Delta t}{C_m}.
$$

Note that the forward Euler approximation results in an *explicit* method of solution since it gives an explicit formula for computing the V_i^{n+1} given both V_i^n and J_i^n, i.e., the values of variables at time $n + 1$ are computed directly from their values at at time n.

In order to solve the equations numerically, we require initial conditions along the length of the fiber plus boundary conditions at the two ends of the fiber. Initial conditions are simply that the membrane potential is at its resting value and that the membrane current is zero. The method for computing the resting potential is described in Section 5.3.3. A variety of boundary conditions are plausible. We shall use the boundary condition that the fiber is closed at both ends so that the longitudinal current is zero at the two ends of the fiber. Equation 6.12 essentially expresses Kirchhoff's current law which can be modified at the closed end to set the appropriate longitudinal current to zero. Using this method for $i = 1$ and setting the longitudinal current from node 0 to node 1 to zero yields

$$\frac{1}{2\pi a(r_o + r_i)}\left(\frac{V_2^n - V_1^n}{(\Delta z)^2}\right) = C_m\left(\frac{V_1^{n+1} - V_1^n}{\Delta t}\right) + J_1^n,$$

which, after collecting terms, results in the difference equation

$$V_1^{n+1} = \eta V_2^n + (1 - \eta)V_1^n - \xi J_1^n.$$

A similar equation results at the other end of the fiber model so that the matrix equation can be written as

$$
\begin{bmatrix} V_1^{n+1} \\ V_2^{n+1} \\ V_3^{n+1} \\ \vdots \\ V_{I-1}^{n+1} \\ V_I^{n+1} \end{bmatrix}
=
\begin{bmatrix}
1-\eta & \eta & 0 & \cdots & 0 & 0 & 0 \\
\eta & 1-2\eta & \eta & \cdots & 0 & 0 & 0 \\
0 & \eta & 1-2\eta & \cdots & 0 & 0 & 0 \\
0 & 0 & \eta & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1-2\eta & \eta & 0 \\
0 & 0 & 0 & \cdots & \eta & 1-2\eta & \eta \\
0 & 0 & 0 & \cdots & 0 & \eta & 1-\eta
\end{bmatrix}
\begin{bmatrix} V_1^n \\ V_2^n \\ V_3^n \\ \vdots \\ V_{I-1}^n \\ V_I^n \end{bmatrix}
- \xi
\begin{bmatrix} J_1^n \\ J_2^n \\ J_3^n \\ \vdots \\ J_{I-1}^n \\ J_I^n \end{bmatrix}.
$$

The matrix equation can be written compactly as

$$\mathbf{V}^{n+1} = \mathbf{A}_{FE}\mathbf{V}^n - \xi\mathbf{J}^n.$$

To obtain a solution, we need to compute J_i^n. This current is linked to the ionic currents (Equation 6.9) which themselves depend upon factors that are solutions to differential equations (Equations 5.11-5.13). Let x stand for m, n, and h then we have

$$\frac{dx}{dt} = \alpha_m - x(\alpha_m + \beta_m),$$

where α and β are instantaneous functions of membrane potential. We discretize this equation with the forward Euler algorithm as follows

$$\frac{x_i^n - x_i^{n-1}}{\Delta t} = \alpha_x\left(V_i^{n-1}\right) - x_i^{n-1}\left(\alpha_x\left(V_i^{n-1}\right) + \beta_x\left(V_i^{n-1}\right)\right),$$

and collect terms to obtain

$$x_i^n = x_i^{n-1}\left(1 - \Delta t\left(\alpha_x\left(V_i^{n-1}\right) + \beta_x\left(V_i^{n-1}\right)\right)\right) + \Delta t\,\alpha_x\left(V_i^{n-1}\right). \qquad (6.13)$$

With the factors m, n, and h substituted for x, the ionic current can be computed as follows

$$(J_{ion})_i^n = \bar{G}_K(n_i^n)^4(V_i^n - V_K) + \bar{G}_{Na}(m_i^n)^3 h_i^n(V_i^n - V_{Na}) + G_L(V_i^n - V_L),$$

where each of the factors m_i^n, n_i^n, and h_i^n are computed according to Equation 6.13. The total current is

$$J_i^n = (J_{ion})_i^n - \frac{r_o(K_{eo})_i^n + r_i(K_{ei})_i^n}{2\pi a(r_o + r_i)}.$$

Although the Forward Euler method is relatively easy to understand, it's numerical properties are undesirable. The forward Euler method is stable provided $\eta \le 0.5$ which implies that to achieve stable solutions, the discretization in space and in time cannot be chosen independently. Specifically, a stable solution requires $\Delta t \le \pi a(r_o + r_i)(\Delta z)^2 C_m$ (Mascagni and Sherman, 1998). Thus, a choice of Δx limits the choice of Δt required to achieve stable solutions. If Δt is too large, the solution will diverge. Thus, for example, the forward Euler algorithm cannot be used to efficiently compute the response for large values of Δt such as would be desirable to determine steady-state responses. Furthermore, the error resulting from temporal and spatial discretization behaves as $O(\Delta t) + O((\Delta x)^2)$. Thus, although the method of solution using the forward Euler algorithm is simple to understand, the numerical properties of the method are not desirable.

6.3.2   Backward Euler method

With the backward Euler approximation (advanced one point in time), the discretized core conductor equation is

$$\frac{1}{2\pi a(r_o + r_i)}\left(\frac{V_{i+1}^{n+1} - 2V_i^{n+1} + V_{i-1}^{n+1}}{(\Delta z)^2}\right) = C_m\left(\frac{V_i^{n+1} - V_i^n}{\Delta t}\right) + J_i^{n+1}, \qquad (6.14)$$

which, after collecting terms and changing signs, yields the difference equation

$$-\eta V_{i+1}^{n+1} + (1 + 2\eta)V_i^{n+1} - \eta V_{i-1}^{n+1} = V_i^n - \xi J_i^{n+1}.$$

Note that the backward Euler method is an *implicit* method of solution since it gives an implicit formula for computing the V_i^{n+1} given both V_i^n and J_i^n. Thus, the solution can only be obtained by solving a set of coupled algebraic equations.

We use the same initial and boundary conditions as for the forward Euler algorithm. By modifying Equation 6.14 with a closed end condition at $i = 1$ and collecting terms, the difference equation is

$$-\eta V_2^{n+1} + (1 + \eta)V_1^{n+1} = V_1^n - \xi J_1^n.$$

A similar equation results at the other end of the fiber model so that the incorporation of the closed-end boundary conditions results in the following matrix equation

$$
\begin{bmatrix}
1+\eta & -\eta & 0 & \cdots & 0 & 0 & 0 \\
-\eta & 1+2\eta & -\eta & \cdots & 0 & 0 & 0 \\
0 & -\eta & 1+2\eta & \cdots & 0 & 0 & 0 \\
0 & 0 & -\eta & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1+2\eta & -\eta & 0 \\
0 & 0 & 0 & \cdots & -\eta & 1+2\eta & -\eta \\
0 & 0 & 0 & \cdots & 0 & -\eta & 1+\eta
\end{bmatrix}
\begin{bmatrix}
V_1^{n+1} \\
V_2^{n+1} \\
V_3^{n+1} \\
\vdots \\
V_{I-1}^{n+1} \\
V_I^{n+1}
\end{bmatrix}
=
\begin{bmatrix}
V_1^n \\
V_2^n \\
V_3^n \\
\vdots \\
V_{I-1}^n \\
V_I^n
\end{bmatrix}
- \xi
\begin{bmatrix}
J_1^{n+1} \\
J_2^{n+1} \\
J_3^{n+1} \\
\vdots \\
J_{I-1}^{n+1} \\
J_I^{n+1}
\end{bmatrix}.
$$

The matrix equation can be written compactly as

$$\mathbf{A_{BE}} \cdot \mathbf{V}^{n+1} = \mathbf{V}^n - \xi \mathbf{J}^{n+1},$$

which has the solution

$$\mathbf{V}^{n+1} = \mathbf{A_{BE}}^{-1} \cdot \left(\mathbf{V}^n - \xi \mathbf{J}^{n+1}\right). \tag{6.15}$$

Similar to the procedure with the forward Euler algorithm, J_i^{n+1} can be computed from the factors that determine the ionic conductances. Once again, we let x stand for m, n, and h, and we discretize this equation, this time with the backward Euler algorithm as follows

$$\frac{x_i^{n+1} - x_i^n}{\Delta t} = \alpha_x\left(V_i^{n+1}\right) - x_i^{n+1}\left(\alpha_x\left(V_i^{n+1}\right) + \beta_x\left(V_i^{n+1}\right)\right),$$

and collect terms to obtain

$$x_i^{n+1} = \frac{x_i^n + \Delta t\,\alpha_x\left(V_i^{n+1}\right)}{1 + \Delta t\left(\alpha_x\left(V_i^{n+1}\right) + \beta_x\left(V_i^{n+1}\right)\right)}. \tag{6.16}$$

With the factors m, n, and h substituted for x, the ionic current can be computed as follows

$$(J_{ion})_i^{n+1} = \bar{G}_K(n_i^{n+1})^4(V_i^n - V_K) + \bar{G}_{Na}(m_i^{n+1})^3 h_i^{n+1}(V_i^n - V_{Na}) + G_L(V_i^n - V_L),$$

where each of the factors m_i^{n+1}, n_i^{n+1}, and h_i^{n+1} are computed according to Equation 6.16. The total current is

$$J_i^{n+1} = (J_{ion})_i^{n+1} - \frac{r_o(K_{eo})_i^{n+1} + r_i(K_{ei})_i^{n+1}}{2\pi a(r_o + r_i)}.$$

One difference is that the factor x_i^{n+1} depends upon the potential V_i^{n+1} so that the four equations — Equation 6.15 and three equations (one for each of the factors m, n, and h) of the form of Equation 6.16 — need to be solved simultaneously. To obtain accurate solutions, the set of 4 equations are iterated until the 4 variables V^{n+1}, m_i^{n+1}, n_i^{n+1}, and h_i^{n+1} converge to within some specified limit. For example, Equation 6.15 can be solved for the value of V^{n+1} using the values of the factors at time increment n to compute new values of the factors. These are then used to compute the new value of V^{n+1} which is used to compute new values of the factors, etc.

The backward Euler algorithm is stable for all values of Δt and Δx. Thus Δt and Δz can be chosen independently. Large values of Δt can be used to determine approximate steady-state values of the solution. Just as for the forward Euler algorithm, the error resulting from temporal and spatial discretization using the backward Euler algorithm behaves as $O(\Delta t) + O((\Delta x)^2)$.

6.3.3   Crank-Nicolson method

In the Crank-Nicolson method, the forward Euler and backward Euler approximations are averaged to yield the following difference equation

$$\frac{1}{4\pi a(r_o + r_i)} \left(\frac{V_{i+1}^{n+1} - 2V_i^{n+1} + V_{i-1}^{n+1}}{(\Delta z)^2} + \frac{V_{i+1}^n - 2V_i^n + V_{i-1}^n}{(\Delta z)^2} \right) =$$

$$C_m \left(\frac{V_i^{n+1} - V_i^n}{\Delta t} \right) + \frac{1}{2}(J_i^n + J_i^{n+1}). \tag{6.17}$$

Alternatively, this equation can be regarded as satisfying the partial differential equation at the discrete time $n + \frac{1}{2}$ with central differences used for all derivatives. This interpretation is exploited in the next section on the *Staggered increment Crank-Nicolson method*.

Equation 6.17 can be rearranged and simplified so that all the voltage variables evaluated at time step $n + 1$ are on the left and all those at time step n are on the right as follows

$$-\frac{\eta}{2}V_{i+1}^{n+1} + (1 + \eta)V_i^{n+1} - \frac{\eta}{2}V_{i-1}^{n+1} = \frac{\eta}{2}V_{i+1}^n + (1 - \eta)V_i^n + \frac{\eta}{2}V_{i-1}^n - \xi\frac{J_i^{n+1} + J_i^n}{2}.$$

As can be seen from the above equations, the Crank-Nicolson method is also an implicit method.

We use the same initial and boundary conditions as for the forward and back-ward Euler algorithms. By modifying Equation 6.17 at the closed end at $i = 1$ and collecting terms, the difference equation is

$$-\frac{\eta}{2}V_2^{n+1} + (1 + \frac{\eta}{2})V_1^{n+1} = \frac{\eta}{2}V_2^n + (1 - \frac{\eta}{2})V_1^n - \frac{1}{2}\xi(J_1^n + J_1^{n+1}).$$

A similar equation results at the other end of the fiber model so that with the incorporation of the closed-end boundary conditions, the matrix equation for the Crank-Nicolson method is

$$
\begin{bmatrix}
1+\frac{\eta}{2} & -\frac{\eta}{2} & 0 & \cdots & 0 & 0 & 0 \\
-\frac{\eta}{2} & 1+\eta & -\frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
0 & -\frac{\eta}{2} & 1+\eta & \cdots & 0 & 0 & 0 \\
0 & 0 & -\frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1+\eta & -\frac{\eta}{2} & 0 \\
0 & 0 & 0 & \cdots & -\frac{\eta}{2} & 1+\eta & -\frac{\eta}{2} \\
0 & 0 & 0 & \cdots & 0 & -\frac{\eta}{2} & 1+\frac{\eta}{2}
\end{bmatrix}
\begin{bmatrix}
V_1^{n+1} \\
V_2^{n+1} \\
V_3^{n+1} \\
\vdots \\
V_{I-1}^{n+1} \\
V_I^{n+1}
\end{bmatrix}
=
$$

$$
\begin{bmatrix}
1-\frac{\eta}{2} & \frac{\eta}{2} & 0 & \cdots & 0 & 0 & 0 \\
\frac{\eta}{2} & 1-\eta & \frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
0 & \frac{\eta}{2} & 1-\eta & \cdots & 0 & 0 & 0 \\
0 & 0 & \frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1-\eta & \frac{\eta}{2} & 0 \\
0 & 0 & 0 & \cdots & \frac{\eta}{2} & 1-\eta & \frac{\eta}{2} \\
0 & 0 & 0 & \cdots & 0 & \frac{\eta}{2} & 1-\frac{\eta}{2}
\end{bmatrix}
\begin{bmatrix}
V_1^n \\
V_2^n \\
V_3^n \\
\vdots \\
V_{I-1}^n \\
V_I^n
\end{bmatrix}
-\frac{\xi}{2}
\left(
\begin{bmatrix}
J_1^n \\
J_2^n \\
J_3^n \\
\vdots \\
J_{I-1}^n \\
J_I^n
\end{bmatrix}
+
\begin{bmatrix}
J_1^{n+1} \\
J_2^{n+1} \\
J_3^{n+1} \\
\vdots \\
J_{I-1}^{n+1} \\
J_I^{n+1}
\end{bmatrix}
\right).
$$

The matrix equation can be written as

$$\mathbf{A_{CN1}} \cdot \mathbf{V}^{n+1} = \mathbf{A_{CN2}} \cdot \mathbf{V}^n - \frac{\xi}{2}\left(\mathbf{J}^n + \mathbf{J}^{n+1}\right),$$

which has the solution

$$\mathbf{V}^{n+1} = \mathbf{A_{CN1}}^{-1} \cdot \mathbf{A_{CN2}} \cdot \mathbf{V}^n - \frac{\xi}{2}\mathbf{A_{CN1}}^{-1} \cdot \left(\mathbf{J}^n + \mathbf{J}^{n+1}\right).$$

Similar to the procedure with the forward and backward Euler algorithms, we need to compute J_i^n from the factors that determine the ionic conductances. Once again, we let x stand for m, n, and h, and we discretize this equation, this time with the trapezoidal rule, so as to preserve the $O((\Delta t)^2)$ error in time, as follows

$$
\frac{x_i^{n+1} - x_i^n}{\Delta t} = \frac{\alpha_x\left(V_i^{n+\frac{1}{2}}\right) - x_i^{n+1}\left(\alpha_x\left(V_i^{n+\frac{1}{2}}\right) + \beta_x\left(V_i^{n+\frac{1}{2}}\right)\right)}{2}
$$
$$
+ \frac{\alpha_x\left(V_i^{n+\frac{1}{2}}\right) - x_i^n\left(\alpha_x\left(V_i^{n+\frac{1}{2}}\right) + \beta_x\left(V_i^{n+\frac{1}{2}}\right)\right)}{2},
$$

and collect terms to obtain

$$
\begin{aligned}
x_i^{n+1} \;=\;& \frac{\frac{\Delta t}{2}\,\alpha_x\left(V_i^{n+\frac{1}{2}}\right)}{1+\frac{\Delta t}{2}\left(\alpha_x\left(V_i^{n+\frac{1}{2}}\right)+\beta_x\left(V_i^{n+\frac{1}{2}}\right)\right)} \\[2ex]
&+x_i^{n}\,\frac{1-\frac{\Delta t}{2}\left(\alpha_x\left(V_i^{n+\frac{1}{2}}\right)+\beta_x\left(V_i^{n+\frac{1}{2}}\right)\right)}{1+\frac{\Delta t}{2}\left(\alpha_x\left(V_i^{n+\frac{1}{2}}\right)+\beta_x\left(V_i^{n+\frac{1}{2}}\right)\right)}
\end{aligned}
\tag{6.18}
$$

With the factors m, n, and h substituted for x, the ionic current can be computed as follows

$$
(J_{ion})_i^{n+1}=\bar{G}_K(n_i^{n+1})^4(V_i^n-V_K)+\bar{G}_{Na}(m_i^{n+1})^3 h_i^{n+1}(V_i^n-V_{Na})+G_L(V_i^n-V_L),
$$

where each of the factors m_i^{n+1}, n_i^{n+1}, and h_i^{n+1} are computed according to Equation 6.16. The total current is

$$
J_i^{n+1}=(J_{ion})_i^{n+1}-\frac{r_o(K_{eo})_i^{n+1}+r_i(K_{ei})_i^{n+1}}{2\pi a(r_o+r_i)}.
$$

As with the backward Euler algorithm, the factor x_i^{n+1} depends upon the potential $V_i^{n+\frac{1}{2}}$ through the parameters α and β so that iteration must be used to arrive at a solution of the 4 simultaneous equations.

For linear, parabolic partial differential equations, the Crank-Nicolson method yields stable solutions although not as stable as the backward Euler method, i.e., spurious bounded amplitude oscillations may appear for large values of increments. In contrast to both the forward and backward Euler algorithms, the error resulting from temporal and spatial discretization for the Crank-Nicolson method behaves as $O((\Delta t)^2)+O((\Delta x)^2)$ provided appropriate boundary conditions (such as those shown above) are used. Thus, with the Crank-Nicolson method it is possible to achieve more accurate solutions for large time intervals than for the backward Euler method.

6.3.4   Staggered increment Crank-Nicolson method

A computationally efficient method (Hines, 1984) that avoids iteration of the solution is based on the observation that the Crank-Nicolson differencing scheme satisfies the partial differential equation at the discrete time $n+\frac{1}{2}$. This property can be seen by writing out the expression as follows

$$
\frac{1}{2\pi a(r_o+r_i)}\left(\frac{V_{i+1}^{n+\frac{1}{2}}-2V_i^{n+\frac{1}{2}}+V_{i-1}^{n+\frac{1}{2}}}{(\Delta z)^2}\right)=2C_m\left(\frac{V_i^{n+\frac{1}{2}}-V_i^n}{\Delta t}\right)+J_i^{n+\frac{1}{2}},\tag{6.19}
$$

which results in the following difference equation

$$-\frac{\eta}{2}V_{i+1}^{n+\frac{1}{2}} + (1+\eta)V_i^{n+\frac{1}{2}} - \frac{\eta}{2}V_{i-1}^{n+\frac{1}{2}} = V_i^n - \frac{\xi}{2}J_i^{n+\frac{1}{2}}.$$

Note that starting with the membrane potential at discrete time n, the difference equation computes (by an implicit method) the potential at $n + \frac{1}{2}$. Then the potential at discrete time $n + 1$ is computed by the explicit method

$$V_i^{n+1} = 2V_i^{n+\frac{1}{2}} - V_i^n. \tag{6.20}$$

We use the same initial and boundary conditions as for the other algorithms. By modifying Equation 6.19 at the closed end at $i = 1$ and collecting terms, the difference equation is

$$-\frac{\eta}{2}V_2^{n+\frac{1}{2}} + (1+\frac{\eta}{2})V_1^{n+\frac{1}{2}} = V_1^n - \frac{\xi}{2}(J_1^{n+\frac{1}{2}}.$$

A similar equation results at the other end of the fiber model so that the matrix equation is

$$
\begin{bmatrix}
1+\frac{\eta}{2} & -\frac{\eta}{2} & 0 & \cdots & 0 & 0 & 0 \\
-\frac{\eta}{2} & 1+\eta & -\frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
0 & -\frac{\eta}{2} & 1+\eta & \cdots & 0 & 0 & 0 \\
0 & 0 & -\frac{\eta}{2} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1+\eta & -\frac{\eta}{2} & 0 \\
0 & 0 & 0 & \cdots & -\frac{\eta}{2} & 1+\eta & -\frac{\eta}{2} \\
0 & 0 & 0 & \cdots & 0 & -\frac{\eta}{2} & 1+\frac{\eta}{2}
\end{bmatrix}
\begin{bmatrix}
V_1^{n+\frac{1}{2}} \\
V_2^{n+\frac{1}{2}} \\
V_3^{n+\frac{1}{2}} \\
\vdots \\
V_{I-1}^{n+\frac{1}{2}} \\
V_I^{n+\frac{1}{2}}
\end{bmatrix}
=
\begin{bmatrix}
V_1^n \\
V_2^n \\
V_3^n \\
\vdots \\
V_{I-1}^n \\
V_I^n
\end{bmatrix}
- \frac{\xi}{2}
\begin{bmatrix}
J_1^{n+\frac{1}{2}} \\
J_2^{n+\frac{1}{2}} \\
J_3^{n+\frac{1}{2}} \\
\vdots \\
J_{I-1}^{n+\frac{1}{2}} \\
J_I^{n+\frac{1}{2}}
\end{bmatrix}.
$$

The matrix equation can be written as

$$\mathbf{A_{CN}} \cdot \mathbf{V}^{n+\frac{1}{2}} = \mathbf{V}^n - \frac{\xi}{2}\mathbf{J}^{n+\frac{1}{2}},$$

which has the solution

$$\mathbf{V}^{n+\frac{1}{2}} = \mathbf{A_{CN}}^{-1} \cdot \left(\mathbf{V}^n - \frac{\xi}{2}\mathbf{J}^{n+\frac{1}{2}}\right). \tag{6.21}$$

This scheme requires that the current be evaluated at the time $n + \frac{1}{2}$ which can be done by iteration as described previously or by having two computing time grids staggered by $\frac{1}{2}$ units of time. To maintain the error due to time quantization at $O((\Delta t)^2)$, we need to compute the factors with second-order accuracy by a centered difference as follows

$$\frac{x_i^{n+\frac{1}{2}} - x_i^{n-\frac{1}{2}}}{\Delta t} = \alpha_x(V_i^n) - (\alpha_x(V_i^n) + \beta_x(V_i^n))\frac{x_i^{n+\frac{1}{2}} + x_i^{n-\frac{1}{2}}}{2}$$

and collect terms to obtain

$$x_i^{n+\frac{1}{2}} = \frac{\Delta t \, \alpha_x \left(V_i^n\right)}{1 + \frac{\Delta t}{2} \left(\alpha_x \left(V_i^n\right) + \beta_x \left(V_i^n\right)\right)} + x_i^{n-\frac{1}{2}} \frac{1 - \frac{\Delta t}{2} \left(\alpha_x \left(V_i^n\right) + \beta_x \left(V_i^n\right)\right)}{1 + \frac{\Delta t}{2} \left(\alpha_x \left(V_i^n\right) + \beta_x \left(V_i^n\right)\right)} \quad (6.22)$$

With the factors m, n, and h substituted for x, the ionic current can be computed as follows

$$(J_{ion})_i^{n+\frac{1}{2}} = \bar{G}_K (n_i^{n+\frac{1}{2}})^4 (V_i^n - V_K) + \bar{G}_{Na} (m_i^{n+\frac{1}{2}})^3 h_i^{n+\frac{1}{2}} (V_i^n - V_{Na}) + G_L (V_i^n - V_L),$$

where each of the factors $m_i^{n+\frac{1}{2}}$, $n_i^{n+\frac{1}{2}}$, and $h_i^{n+\frac{1}{2}}$ are computed according to Equation 6.22. The total current is

$$J_i^{n+\frac{1}{2}} = (J_{ion})_i^{n+\frac{1}{2}} - \frac{r_o (K_{eo})_i^{n+\frac{1}{2}} + r_i (K_{ei})_i^{n+\frac{1}{2}}}{2\pi a (r_o + r_i)}.$$

The method is outlined as follows:

- Start with the value of V_i^n and the value of $x_i^{n-\frac{1}{2}}$ which allows a computation of $J_i^{n-\frac{1}{2}}$;

- Compute $x_i^{n+\frac{1}{2}}$ using Equation 6.22 which then allows a computation of $J_i^{n+\frac{1}{2}}$;

- Compute $V_i^{n+\frac{1}{2}}$ using Equation 6.21;

- Compute V_i^{n+1} using Equation 6.20;

- The solution has now been advanced to V_i^{n+1} and the value of $x_i^{n+\frac{1}{2}}$.

The process can be initiated by calculating either V_i or x_i for a $\frac{1}{2}$ time step increment to stagger the variables.

6.4   User's Guide to the Software

When the software is selected from the *Softcell* menu, two figures are displayed (in addition to MATLAB's command window): *PAP Control*, and *PAP Workspace*; the *PAP Workspace* is distinct from MATLAB's workspace. The *PAP Control* figure allows the user to display other figures for setup and analysis, and to start a simulation. The *PAP Workspace* figure is (as the name suggests) a workspace where a schematic representation of an axon, a stimulus source, and a voltage recorder are displayed. In the *PAP Workspace* figure, the user can choose to display the stimulus setup figure (*PAP Stimulus*), the parameter setup figure (*PAP*
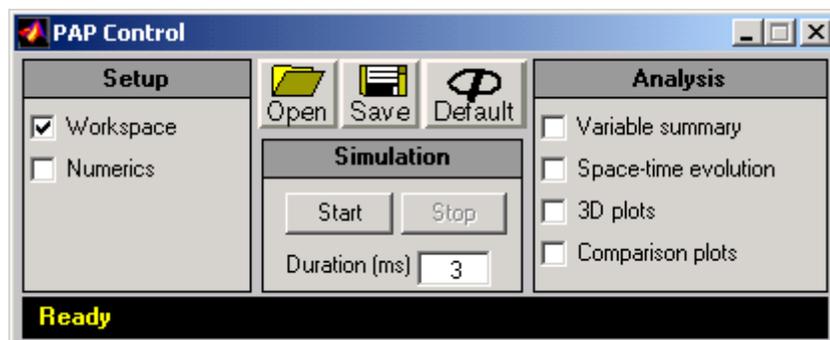
Figure 6.3: The PAP Control figure when the software is initiated.

Parameters), the voltage recorder figure (*PAP Voltage-Recorder*), and place the electrodes of the stimulus source and recorder using the mouse cursor on the axon schematic. Clicking on the *Start* pushbutton in the *PAP Control* figure results in the computation of the response as a function of time and space. Upon completion of the computation, the user can choose to view the space-time evolution of the membrane potential (which is one of the computed solution variables) by means of a color code along the axon schematic in the *PAP Workspace* figure. The user can also examine the space-time evolution of other response solution-variables to gain insight to the model's dynamic behavior or use 3D and comparison plots for static analyses of the model's behavior.

6.4.1 PAP Control

There are three panels in the *PAP Control* figure (Figure 6.3). With the left panel, the user can display the *PAP Workspace* figure (See Section 6.4.2) and the *PAP Numerics* (See Section 6.4.5). In the center (Simulation) panel, the user can specify the simulation duration and start the simulation by clicking on the *Start* pushbutton. The right (Analysis) panel allows the user to do post simulation analysis of the results. The first (Variable summary) option allows the user to display a numerical summary of all the solution variables. The second (Space-time evolution) option allows the user to view the dynamics of a solution variable (such as membrane potential, membrane current-components, etc.) as a function of space with running time. The third (3D plots) option allows the user to view three-dimensional graphs of any solution variable as a function of time and space. The fourth (Comparison plots) option allows the user to compare multiple solution variables as a function of a solution variable of choice (or as a function of time or space) in two different modes: fixed-time or fixed-space.

Starting, pausing, and stopping a simulation

Clicking on the *Start* pushbutton starts the computation of the solution variables. While a computation is in progress, the status-bar located along the bottom of the figure shows the name of the numerical method being used and the percentage of completion. The *Start* pushbutton changes to a *Pause* pushbutton. Clicking on the *Pause* pushbutton pauses the computation and causes the status-bar to display *Paused*. When the simulation is paused, the *Pause* pushbutton changes to a *Continue* pushbutton. Clicking on the *Continue* pushbutton resumes the computation and the two states (pause/continue) toggle with each click of the pushbutton until the computation is completed. Upon completion, the pushbutton changes back to *Start* and the status-bar acknowledges successful completion along with a time stamp. Next to *Start* is the *Stop* pushbutton which allows the user to abort a computation that is currently in progress or paused. Clicking on the *Stop* pushbutton causes the status-bar to display *Simulation Aborted*.

Loading, saving, and re-initializing

The user can save all simulation parameters and results in a file as well as load a previously saved file by clicking on the *Save* and *Open* button, respectively. The data stored includes the parameter values of the axon, the membrane, the bath, the stimulus, the configuration of the electrodes, and solution variables. When a file is loaded, the filename (along with it's directory path) is shown in this figure's status-bar as well as in the status-bar of other setup figures such as *PAP Parameters*, *PAP stimulus*, and *PAP numerics*. This is useful for keeping track of the source of the parameter values used during a session.

When the software is launched, the figure's status-bar displays *Ready*. This indicates that no simulation has been run and that there are no results. This is the initial state of the software and all parameters are set to their default values. After parameters have been changed they can be reset to their default values by clicking on the *Default* pushbutton.

6.4.2   PAP Workspace

Selecting *Workspace* in the *PAP Control* figure displays the *PAP Workspace* figure (Figure 6.4). The *PAP Workspace* figure consists of a window and a control region. The window contains from top to bottom: a ruler, an interactive display area (showing a schematic of an axon, a stimulus-source icon, and a voltage-recorder icon), another ruler, a status-bar, and a horizontal scrollbar. The default length of the axon is three centimeters and fits in the window's display area. The schematic representation of a longer axon model stretches beyond the display area's right edge. The user can use the horizontal scrollbar to scroll
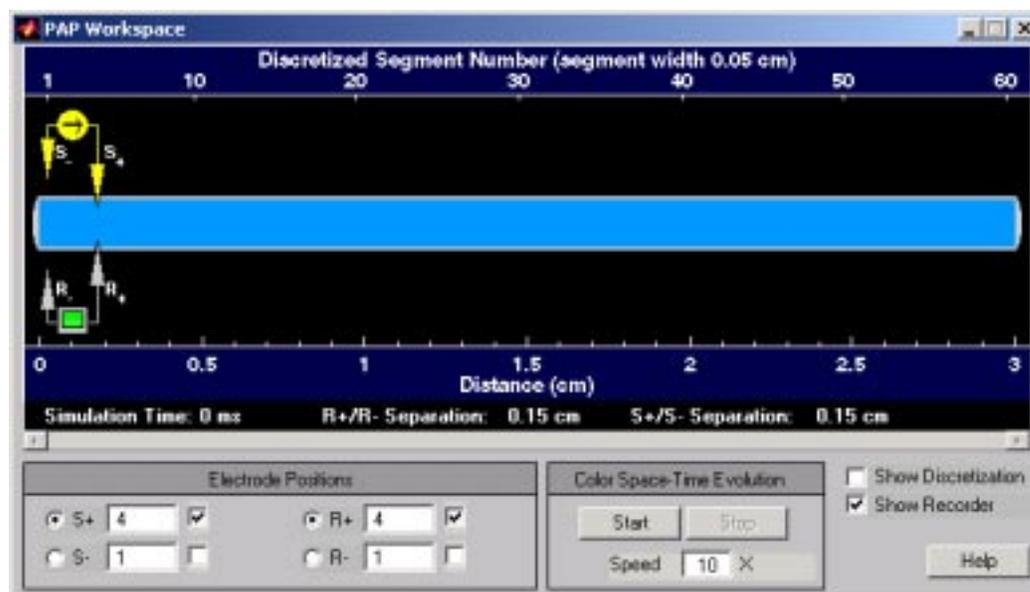
Figure 6.4: The PAP Workspace figure.

to the hidden portion. By default the top ruler is set to show number of discretized segments, and the bottom ruler is set to show distance in centimeters. Clicking on either ruler will toggle its unit of measurement between number of discretized segments and distance. The status-bar shows the simulation time.

Clicking on the *Help* button located in the lower right corner of the *PAP Workspace* figure displays the *PAP Workspace Help* figure. This figure shows a step-by-step guide to setting up a customized simulation as described in Section 6.4.11.

Positioning electrodes

Stimulus electrodes (yellow) are located above the axon schematic and voltage-recorder electrodes (grey) are located below the axon schematic. The user can control the position of an electrode either by pointing and clicking near the electrode or by clicking and typing in the *Electrode Positions* panel.

To change electrode locations by the point-and-click method, first select the electrode by clicking on the appropriate radiobutton in the *Electrode Positions* panel. Note that the positive and negative electrodes of the stimulus ($S+$ and $S-$) are selected independently from the positive and negative electrodes of the recorder ($R+$ and $R-$). After the appropriate radiobutton is selected, clicking anywhere outside of the axon schematic within the interactive display area (e.g., the black region) sets the electrode's longitudinal position to the mouse-click location. Clicking in the black region above the axon schematic moves the stimulus electrodes, and clicking in the black region below the axon schematic moves the

recorder electrodes. Both stimulus and recording electrode positions can also be changed by typing in the relevant text edit boxesin the *Electrode Positions* panel. Clicking on an electrode will toggle the electrode position between intracellular and extracellular. Clicking on the check boxes in the *Electrode Positions* panel also allows control of the intracellular/extracellular locations of electrodes. The check box on the right part of the *Electrode Positions* panel controls the display of the voltage recorder and its electrodes. Distances between stimulus electrodes and between voltage recording electrodes are indicated below the lower ruler.

Displaying other setup figures

Clicking on the stimulus-source icon (yellow) will bring up the *PAP Stimulus* figure where the user can specify the characteristics of the desired stimulus (See Section 6.4.4). Clicking on the voltage-recorder icon (green and grey) will display the *PAP Voltage-Recorder* figure where a real-time potential curve is plotted during a simulation run (See Section 6.4.6). Clicking on the axon schematic will display the *PAP Parameters* figure where the user can change the parameters of the axon, the membrane, and the bath (See Section 6.4.3).

Showing discretization and color-coded space-time evolution

The option (Show discretization) to the right of the *Color Space-Time Evolution* panel allows the user to toggle on/off the spatial discretization display on the axon schematic. In the *Color Space-Time Evolution* panel, the pushbutton *Start* allows the user to playback the space-time evolution of the membrane potential on the axon schematic by using a color code to convey the range of potential levels. The playback can be stopped using the *Stop* button. During playback, the *Start* becomes a *Pause* button which allows the user to pause the playback. The speed of the playback can be adjusted in the *Speed* field located under the pushbutton. For example, if the simulation data contains membrane potential values for every 0.01 ms, changing the field value to ten will playback simulation data of every 0.1 ms.

   Note that a voltage curve is plotted in the *Voltage-Recorder* figure during playback (See Section 6.4.6). Figure 6.5 shows a series of snapshots taken from a time-evolution color code visualization of the membrane potential of a three-centimeters long axon model.

6.4.3   PAP Parameters

Clicking on the axon schematic in the *PAP Workspace* figure displays the *PAP Parameters* figure (Figure 6.6). The *PAP Parameters* figure is divided into three sections: *Membrane Characteristics*, *Bath Characteristics*, and *Axon Characteristics.* In each section, there are three columns for each parameter: the first col-
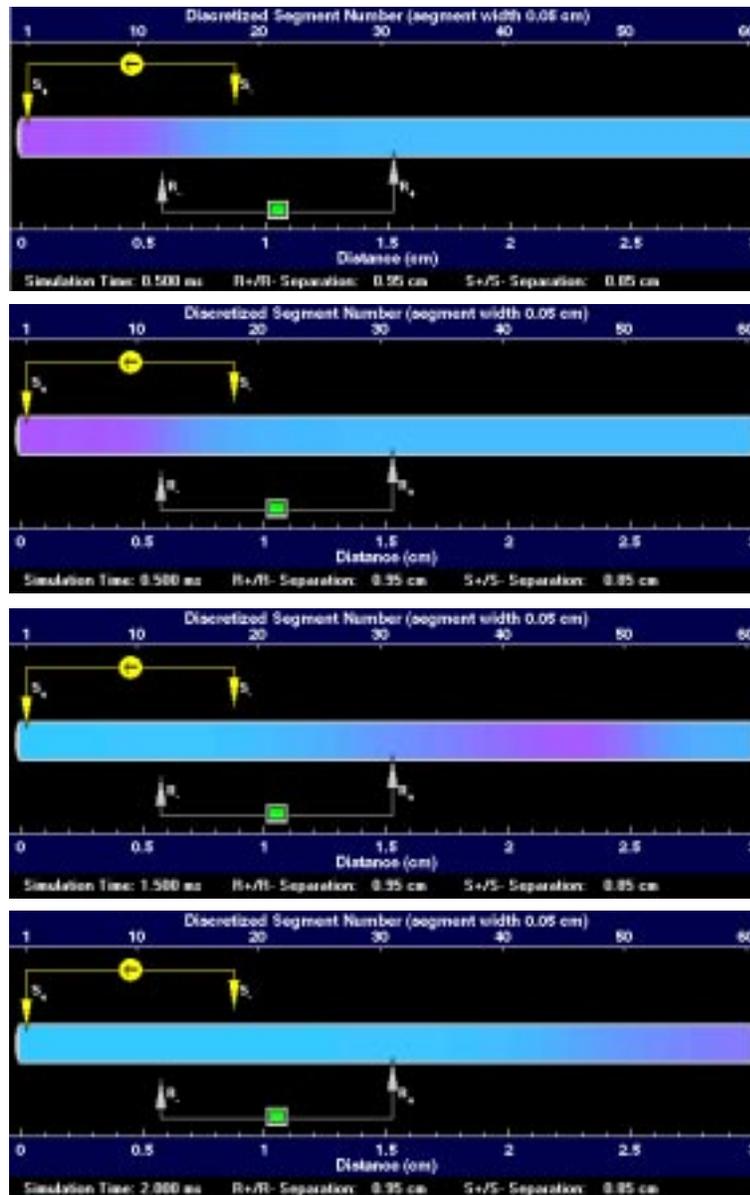
Figure 6.5: Snapshots of a propagating action potential shown using Color Space-Time Evolution in the *PAP Workspace* figure.
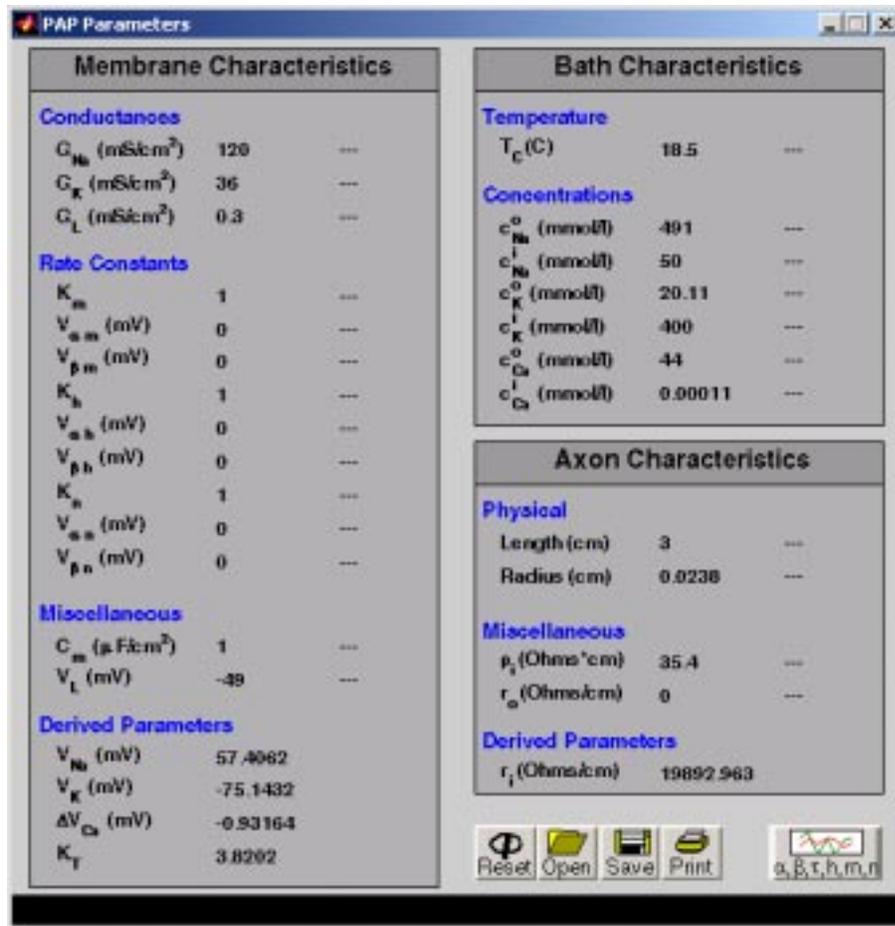
Figure 6.6: The PAP Parameters figure.

umn identifies the parameter, the second gives its default value, and the third column shows the value of the parameter if it is modified from its default value *(Note that the PAP Parameters figure data entry method is based on that of the HH Parameters figure).*

Changing parameter values

Parameters can be modified by clicking on the entry in the third column of the parameter field. When the parameter is selected, its third column entry is replaced by an edit box. Clicking any mouse button inside the edit box allows the parameter to be changed. The new value of the parameter is entered from the keyboard; < RETURN > terminates the parameter entry. If the new value differs from the default value, then the new value will appear in the third column. With this method of display, the parameter list can be scanned quickly to indicate which parameters differ from their default values.

There are some restrictions on the numerical values of parameters; G's ≥ 0, $C_m \geq 0$, c's > 0, K's > 0, $\rho_i \geq 0$, $r_o \geq 0$ (r_o cannot be zero when $\rho_i = 0$), the axon's radius and length must be greater than zero, and the temperature must be above absolute zero. The derived parameters cannot be changed but are derived from the other parameters and are displayed for the convenience of the user. For example, the sodium equilibrium potential V_{Na} cannot be changed directly by the user, but changes automatically when the sodium concentrations or temperature are changed.

Loading, saving, resetting, and printing

The parameters can be loaded from a file by clicking on *Open* and can be saved by clicking on *Save*. Saving and loading in the *PAP Parameters* figure deals only with the model's axon, membrane and bath parameters; e.g., simulation results are not loaded or saved. When a parameters file is loaded, the status-bar of the *PAP Parameters* figure displays the filename (with the directory path) and the parameter values loaded from the file become the default values. For example, if the user changes the temperature value and clicks the *Reset* pushbutton, the temperature value resets to the value stored in the loaded file. To reset the parameter values to their software-default values, use the *Default* pushbutton in the *PAP Control* figure. If no file is loaded, clicking the *Reset* pushbutton resets the parameter values to the software-default values. The *PAP Parameters* figure can be printed using the *Print* button.

Viewing the voltage dependent parameters

The Hodgkin-Huxley model of a propagated action potential as well as the Hodgkin-Huxley model of a membrane action potential of a space-clamped axon contains several parameters that are instantaneous functions of the membrane potential.

Plots of these parameters are available as a function of V_m for the following: $\alpha_m(V_m)$, $\beta_m(V_m)$, $m_\infty(V_m)$, $\tau_m(V_m)$, $\alpha_h(V_m)$, $\beta_h(V_m)$, $h_\infty(V_m)$, $\tau_h(V_m)$, $\alpha_n(V_m)$, $\beta_n(V_m)$, $n_\infty(V_m)$, $\tau_n(V_m)$ (see Section 5.2.3). These can be accessed by clicking on the button marked $\alpha, \beta, \tau, h, m, n$ which results in the display of the *PAP Parameters vs. Potential* figure (Figure 6.7).
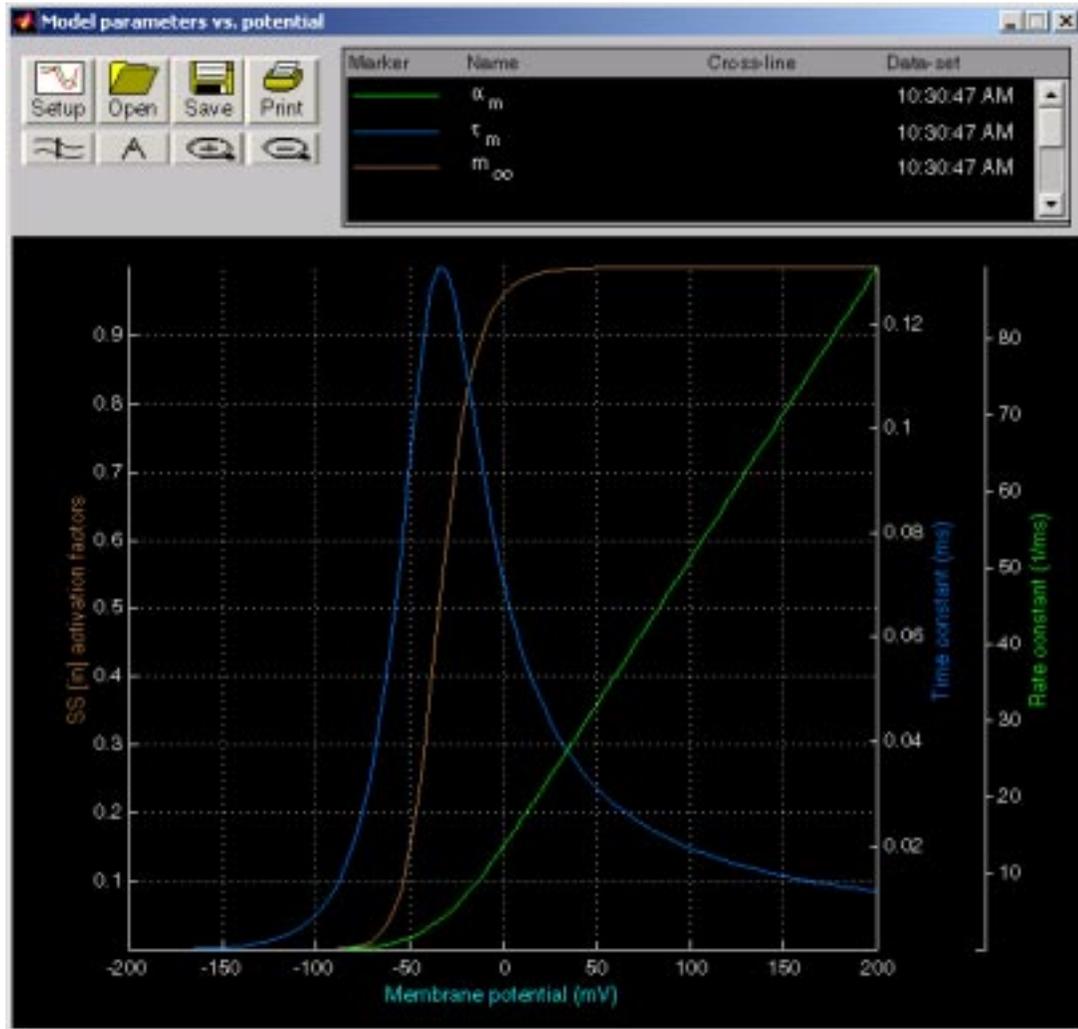
Figure 6.7: The PAP Parameters vs. Potential figure.

Note that the *PAP Parameters vs. Potential* figure is similar to the *PAP Comparison Plots* figure described in Section 6.4.10. Please refer to that section for more information about the available plot manipulation functions. The *PAP Parameters vs. Potential* figure has its own setup figure which differs from the one of the *PAP Comparison Plots* . Clicking on the *Setup* button displays this setup figure. The user can select one or more variables and click on either *Graph* or *Overlay* to make the plot. The *Graph* and *Overlay* buttons are described in Section 6.4.10.
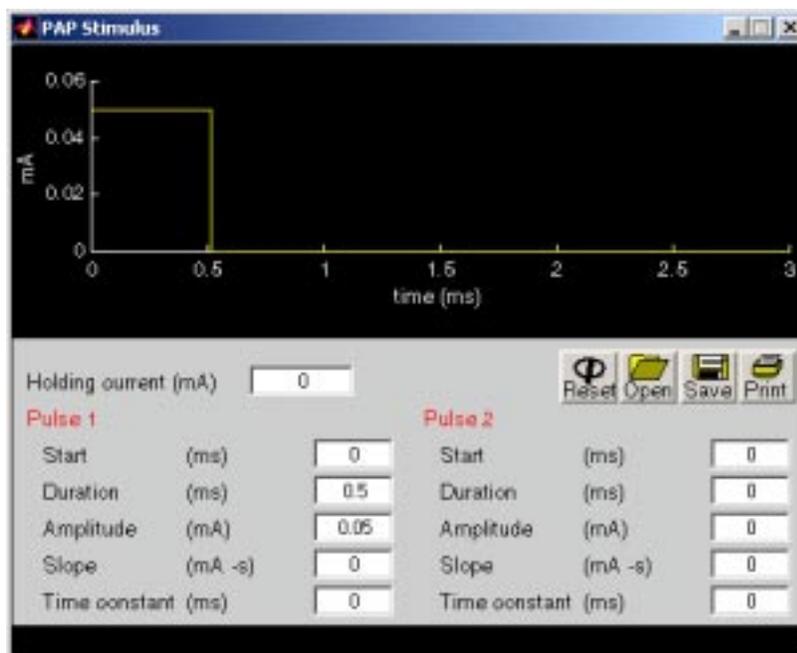
Figure 6.8: The PAP Stimulus figure.

6.4.4 PAP Stimulus

Clicking on the stimulus-source icon (yellow) in the *PAP Workspace* figure displays the *PAP Stimulus* figure (Figure 6.8). Note that the layout of the *PAP Stimulus* figure is similar to that of the HH Stimulus figure. The upper-half of the figure shows a graph of the stimulus pulse(s) specified in the parameter-entry columns: *Pulse 1* and *Pulse 2*. The duration of the stimulus depends on the duration of the simulation specified in the *PAP Control* figure. For example, if the simulation duration is four milliseconds, then the abscissa of the stimulus graph is also of length four milliseconds.

Specifying a pulse

A maximum of two pulses can be defined using a set of parameters for each pulse. In addition, a holding DC current can be specified in the *Holding current* field. Thus, the overall stimulus is the sum of a constant plus two independently specifiable pulses. In mathematical terms, the stimulus, $s(t)$, is defined as follows

$$s(t) = s^o + s^1(t) + s^2(t),$$ (6.23)

where s^o is a constant and $s^1(t)$ and $s^2(t)$ are independently specifiable pulses. Each pulse has the form

$$s^i(t) = \begin{cases} m^i(t - t^i_o) + A^i e^{-(t-t^i_o)/\tau^i} & \text{if } t_o \leq t \leq t_o + t_{dur}, \\ 0 & \text{otherwise}, \end{cases} \qquad (6.24)$$

where pulse i (i = 1 or 2) starts at t_o and has a duration t_{dur}. The amplitude of the pulse is A^i, the slope is m^i, and the time constant is τ^i. If the slope and the time constant are set to zero, then the remaining parameters specify the amplitude, duration and onset time of a rectangular pulse. If the slope has a non-zero value, a ramp of that slope is added to the rectangular pulse for the duration of the pulse. Similarly, if the time constant has a non-zero value, then an exponential pulse with the selected time constant is added to the waveform. The time constant can be positive or negative.

The variables t^i_o, t^i_{dur}, A^i, m^i, and τ^i that defines a pulse in the form of Equation 6.24 are specified in the *PAP Stimulus* figure by the parameter fields *Start*, *Duration*, *Amplitude*, *Slope*, and *Time constant* in the *Pulse i* column, respectively. By adjusting these parameters, the user can generate rectangular pulses, ramp pulses, exponential pulses or a combination of these. An example is shown in Figure 6.9.

Loading, saving, and resetting

The parameters can be loaded from a file by clicking on *Open* and can be saved by clicking on *Save*. Saving and loading in the *PAP Stimulus* figure deals only with the stimulus parameters; e.g., simulation results are not loaded or saved. When a file with stimulus parameters is loaded, the status-bar of the *PAP Stimulus* figure displays the filename (with the directory path) and the parameter values loaded from the file become the default values. For example, if the user changes the holding current value and clicks the *Reset* pushbutton, the holding current value resets to the value stored in the loaded file. To reset the parameter values to their software-default values, use the *Default* pushbutton in the *PAP Control* figure. If no file is loaded, clicking the *Reset* pushbutton resets the parameter values to the software-default values.

6.4.5   PAP Numerics

Selecting *Numerics* in the *PAP Control* figure displays the *PAP Numerics* figure (Figure 6.10). The *PAP Numerics* figure allows the user to select the numerical method used to solve the partial differential equations described in Section 6.2.
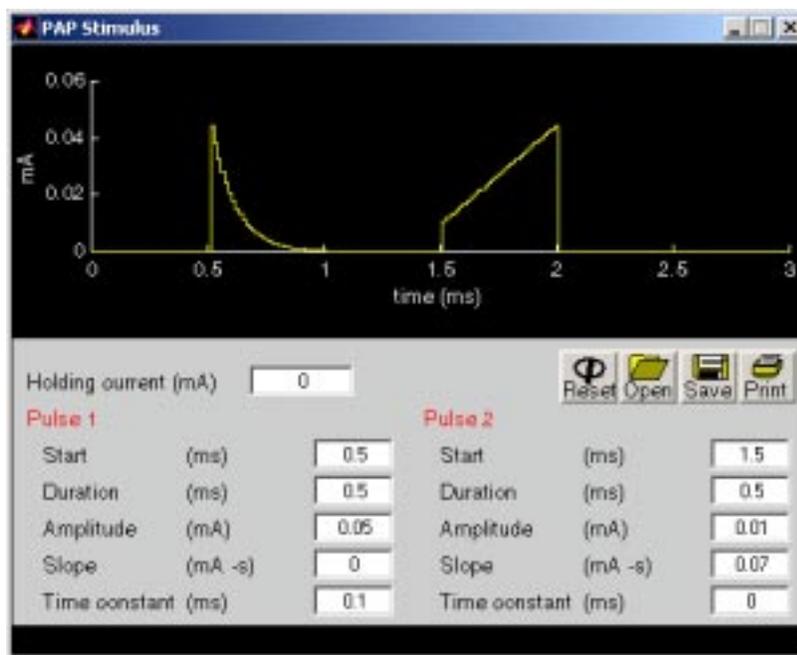
Figure 6.9: The PAP Stimulus figure showing a stimulus that consists of two pulses: the first pulse is an exponential pulse and the second is a ramp plus a rectangular pulse.

Choosing a numerical method and specifying discretization parameters

The list-box located at the top of the figure allows the user to select one of four available numerical methods: forward Euler, backward Euler, Crank-Nicolson, and staggered Crank-Nicholson (staggered C-N). These numerical methods are described in detail in Section 6.3. By default, the software uses the staggered C-N numerical method. All numerical methods employ the finite difference method which requires two discretization parameters: a time step, Δt, and a spatial step, Δz. Decreasing the time step produces a smoother time evolution of the solution variables (for example, a smoother action potential propagation), but computation time and memory usage increases. Decreasing the space step produces an axon schematic with more discrete elements – hence a smoother spatial
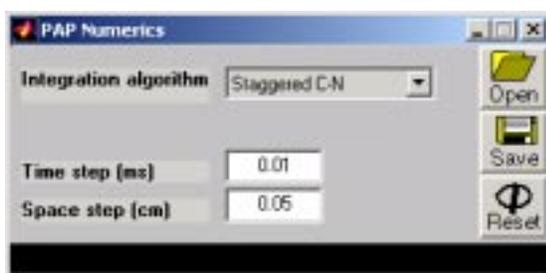
Figure 6.10: The PAP Numerics figure.

evolution of the solution variables (for example, a smoother membrane-potential curve as a function of space when viewed in a snapshot), but computation time and memory usage increases. In order to obtain a reasonably accurate solution using a particular numerical method, the discretization parameters have to be chosen appropriately.

Loading, saving, resetting, and printing

The parameters can be loaded from a file by clicking on *Open* and can be saved by clicking on *Save*. Saving and loading in the *PAP Numerics* figure deals only with the numerics parameters; e.g., simulation results are not loaded or saved. When a numerics parameters file is loaded, the status-bar of the *PAP Numerics* figure displays the filename (with the directory path) and the parameter values loaded from the file become the default values. For example, if the user changes the time step and clicks the *Reset* pushbutton, the time step resets to the value stored in the loaded file. To reset the parameter values to their software-default values, use the *Default* pushbutton in the *PAP Control* figure. If no file is loaded, clicking the *Reset* pushbutton resets the parameter values to the software-default values. The *PAP Numerics* figure can be printed using the *Print* button.

6.4.6   PAP Voltage-Recorder

Clicking on the voltage-recorder icon (grey and green) in the *PAP Workspace* figure displays the *PAP Voltage-Recorder* figure (Figure 6.11). The *PAP Voltage-Recorder* figure shows a graphing region where a voltage curve is plotted when the user clicks on the *Start* button located in the *PAP Workspace* figure. When plotting a curve, the graphing area can display only a portion 3 ms long. If the plot extends beyond 3 ms, the scrollbar under the graphing area becomes active, so the user can scroll to see any section of the curve. Also, a *Show entire plot* option is enabled, so the user can choose to display the whole curve on the graphing area. For example, shrink the abscissa to fit the entire plot in the graphing region.

The voltage plot corresponds to the voltage recorded at the location of the recording electrodes located beneath the axon schematic. The user can reposition the recording electrodes as described in Section 6.4.2 and click on *Start* to obtain the corresponding voltage plot. Clicking on the *Grid* option displays a grid on the graphing region.

Markers can be set to measure the times of occurrence of features in the waveform. Clicking on the t1 marker and clicking on the plot, results in a red, vertical line; clicking again fixes the time of occurrence of that line. The same can be done with the t2 marker. The time of occurrence of each of the markers and the difference in time between the two markers are displayed at the top of the figure.
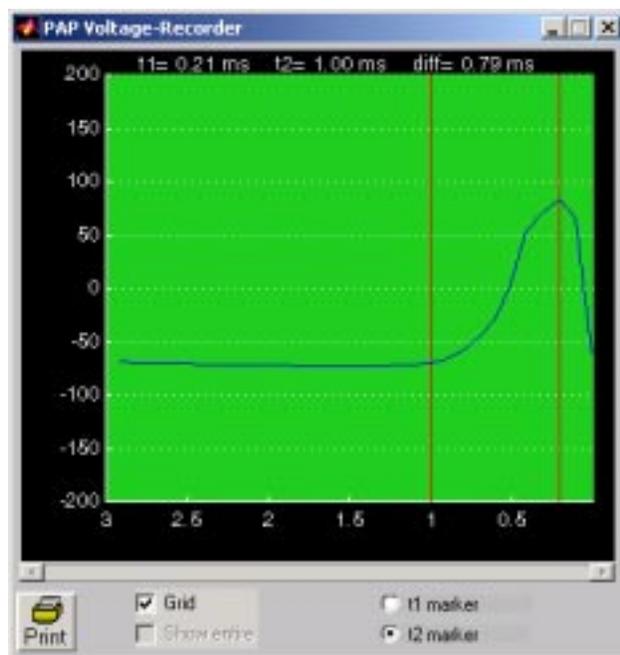
Figure 6.11: The PAP Voltage-Recorder figure.

6.4.7   PAP Variable Summary

Selecting *Variable summary* in the Analysis panel of the *PAP Control* figure displays the *PAP Variable Summary* figure as shown in Figure 6.12. This figure allows the user to display the initial, minimum, and maximum value of any solution variable. The user can either select to view these summary values obtained from the spatial dimension at a specified time, or from the temporal dimension at a specified location. The time and location are specified in units of milliseconds and centimeters, respectively. The largest valid time value is equal to the simulation duration, and the largest valid location value is equal to the length of the axon model.

6.4.8   PAP Space-Time Evolution

Selecting *Space-time evolution* in the Analysis panel of the *PAP Control* figure displays the *PAP Space-Time Evolution* figure as shown in Figure 6.13. This figure only appears if there are simulation results (i.e. after a simulation has been run). The *PAP Space-Time Evolution* figure provides a means of viewing the dynamics of any solution variable along the axon as time changes.

Viewing the space-time evolution of a solution variable

When the figure is first displayed, V_m (membrane potential) is selected in the list-box located at the top of the figure. Use the list-box to view any desired solution variable. After a selection is made, the graph under the list-box shows

Figure 6.12: The PAP Variable Summary figure.
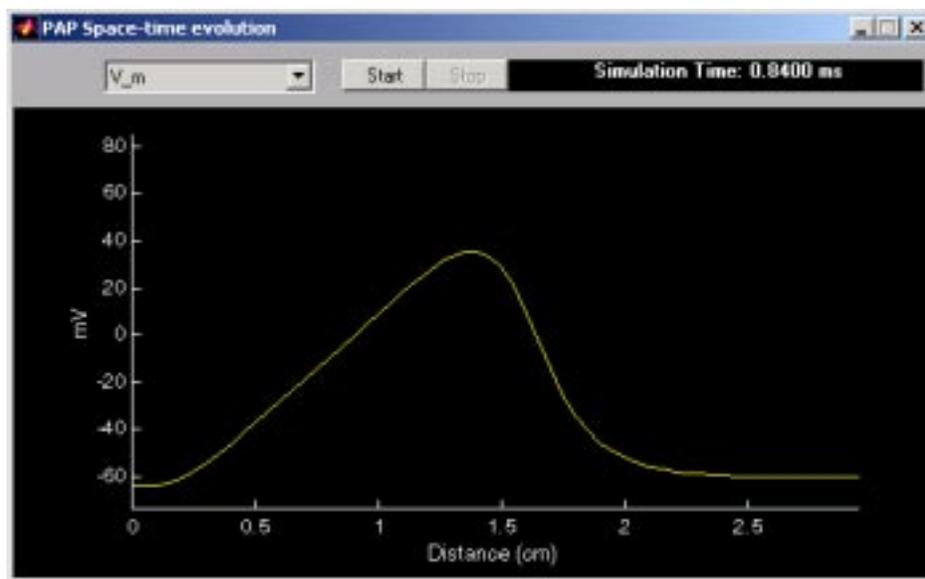
Figure 6.13: The PAP Space-Time Evolution figure showing
a snapshot of a propagating action potential.

the selected solution variable as a function of space at time = 0 ms. To view
the selected solution variable as a function of space at progressing time starting
from 0 ms, click on the *Start* pushbutton. The process can be stopped and
paused using the appropriate buttons. The *Start* button changes to a *Pause*
button when a process is running. Clicking on an axis label displays the *Axis
configuration* figure (Figure 6.18) which allows the user to change the properties
of the axis.

6.4.9   PAP 3D Plots

Selecting *3D plots* in the Analysis panel of the *PAP Control* figure displays the
PAP 3D Plots figure (Figure 6.14). This figure only appears if there are simulation
results (i.e. after a simulation has been run). The *PAP 3D Plots* figure provides a
means of visualizing the space-time dependence of any solution variable.

Viewing a 3D plot of a solution variable

When the figure is first displayed, V_m (membrane potential) is selected in the
list-box located at the top of the figure. Use the list-box to view any desired
solution variable. After a selection is made, the surface plot beneath the list-
box shows the selected solution variable as a function of space and time. The
colorbar on the right indicates the magnitude of the colored surface plot. To
change the view angle use the horizontal scrollbar located at the bottom and the
vertical scrollbar located on the left of the plot. The vertical scrollbar controls

the viewing elevation (i.e. rotates the plot up and down), and the horizontal scrollbar controls the viewing azimuth (i.e. rotates the plot left and right). In addition, the user can switch between mesh and surface plot by toggling the *mesh* option *on/off*. Since the mesh plot samples the surface of the variable as a function of space and time, care must be taken with evaluating 3D mesh plots of discontinuous functions.

Clicking on the *box* option toggles *on/off* an axes-bounding box. Clicking on a solution variable's axis label displays the *Axis configuration* figure (Figure 6.18) which allows the user to change the properties of that axis.

6.4.10   PAP Comparison Plots

Selecting *Comparison plots* in the Analysis panel of the *PAP Control* figure displays the *PAP Comparison Plots* figure as shown in Figure 6.16. The *PAP Comparison Plots* figure allows the user to plot any solution variable against any other. The figure displays a group of pushbuttons on the upper-left corner, a legend panel on the upper-right corner, and a graphing region.

Using the Comparison Plots Setup figure to generate plots

Clicking on the *Setup* button in the *PAP Comparison Plots* figure displays the *Comparison Plots Setup* figure (See Figure 6.15) where the user can choose: fixed-space plots (Figure 6.16) or fixed-time plots (Figure 6.17). Fixed-space plots require the user to specify a longitudinal location of the axon, z_o, in units of centimeters. The user can use this mode to generate *solution-variable-A vs. solution variable-B at location, z_o, plots.* Fixed-time plots require the user to specify a time, t_o, in units of milliseconds. This mode is used to generate *solution-variable-A vs. solution-variable-B at time, t_o, plots.*

Follow these steps to make a plot:

- Choose a plotting mode: fixed-space plots or fixed-time plots.

- For fixed-space plots specify the *Distance* parameter: enter a value in units of centimeters. Note that the maximum allowable value is the length of the axon model. Multiple distance values can also be entered in MATLAB array format (i.e. [1.0 1.5 3.5], [1.0 : 0.05 : 2.0]). Please refer to the MATLAB user's manual for more information.

- For fixed-time plots specify the *Time* parameter: enter a value in units of milliseconds. Note that the maximum allowable value is equal to the simulation duration. Multiple time values can also be entered in MATLAB array format.

- Choose a solution variable in the X-Variable panel: this is the independent variable and it corresponds to the the values along the abscissa.
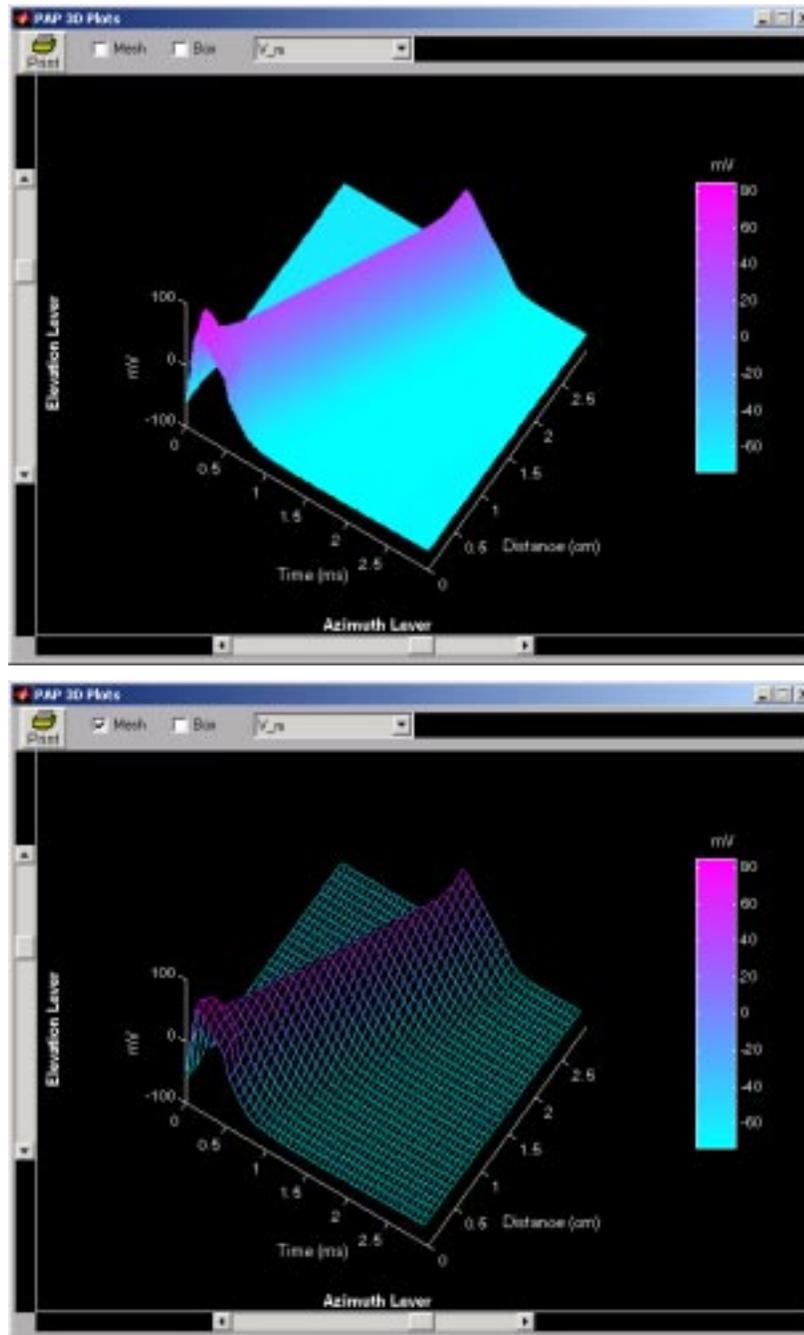
Figure 6.14: The PAP 3D Plots figure showing a propagating action potential. Top: a surface plot. Bottom: a mesh plot.
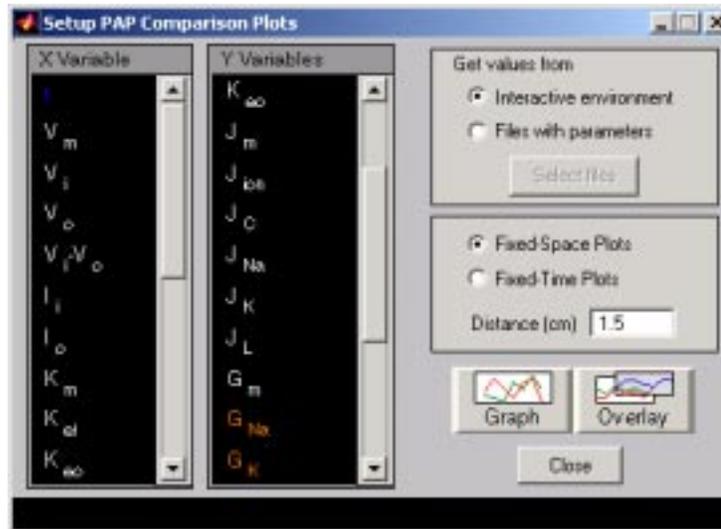
Figure 6.15: The Setup PAP Comparison Plot figure show-
ing the variables used to setup Figure 6.16.).

- Choose one or more solution variables in the Y-Variable panel: multiple dependent variables can be plotted for comparison. They are all plotted as a function of the solution variable selected in the X-Variable panel. Dependent variables that have same unit of measurements share one ordinate axis. Both sides of the plotting area may show distinct ordinate axes to accomodate distinct units of the variables selected. For example, the user can choose to plot the membrane potential, V_m, and the sodium conductance, G_{Na}, both as a function of time, t. In this case, the user would select t in the X-Variable panel, and both V_m and G_{Na} in the Y-Variable panel. Since V_m is in units of millivolts and G_{Na} is in units of mS/cm^2, the plot should show two ordinate axes. The abscissa, in this case, should be in units of milliseconds.

- Click on *Graph* to graph the selections on a cleared plotting region. The *Graph* pushbutton clears all previous plots. Click on *Overlay* to graph the selection on the plotting region without clearing any previous plots. This feature allows the user to overlay the solutions obtained from different simulations. For example, suppose the user wants to plot the membrane potential, V_m, as a function of time, t, at the axon longitudinal location, $z_o = 2$ cm, for temperature values of 6.5°, 18°, and 20° Celsius. The user would need to run a simulation with the temperature parameter set to 6.5, plot V_m vs. t at $z_o = 2$ using the *Graph* button. Then do the subsequent simulation runs with the the other two temperature values and plot using the *Overlay* button so that all three plots are retained in the plotting region.

In addition, the user can make plots using data stored in files. To do this,
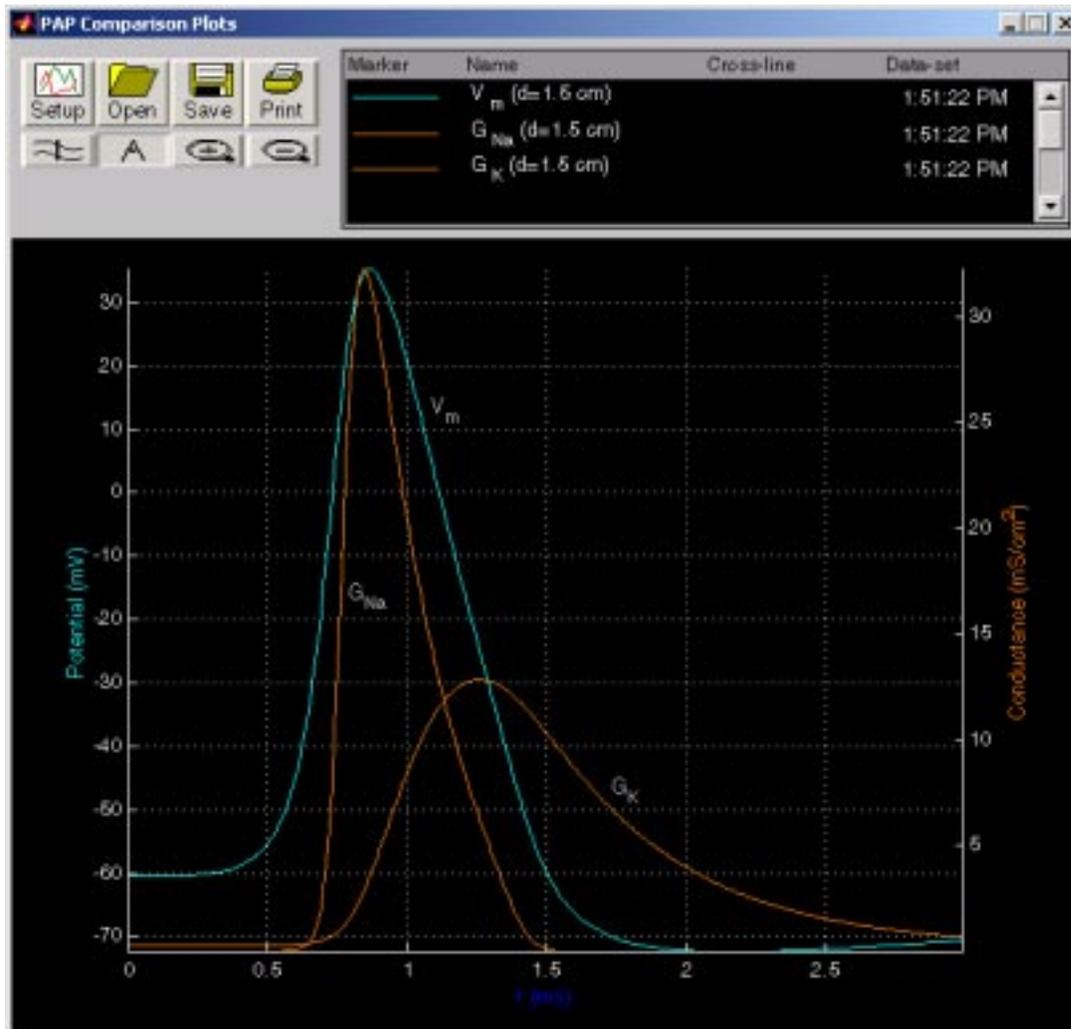
Figure 6.16: The PAP Comparison Plots figure showing a fixed-space plot of the membrane potential, V_m, and the ionic conductances, G_{Na} and G_K, with model parameters and the stimulus defined in Figures 6.6 and 6.8.
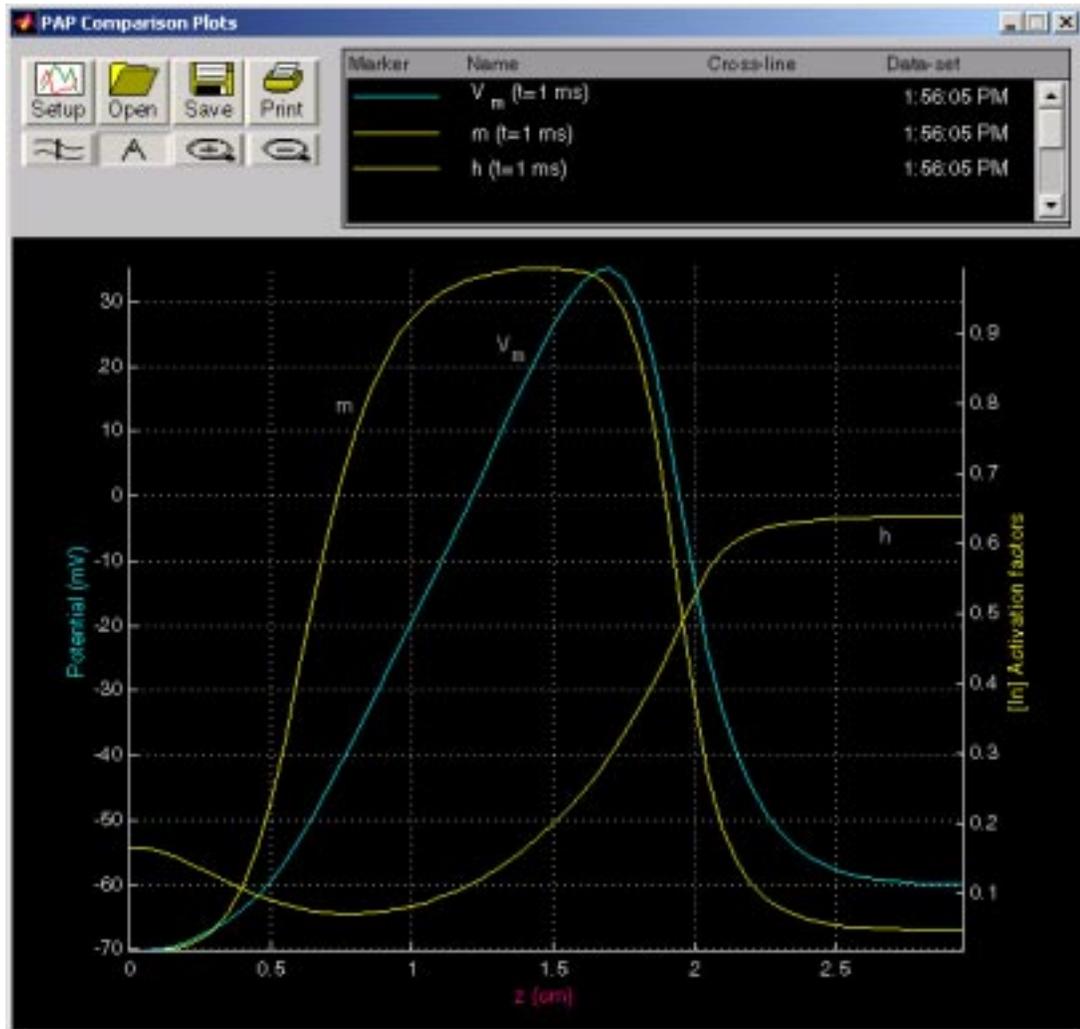
Figure 6.17: The PAP Comparison Plots figure showing a fixed-time plot of the membrane potential, V_m, and the activation factors, m and h, obtained from a simulation with model parameters and the stimulus defined in Figures 6.6 and 6.8.
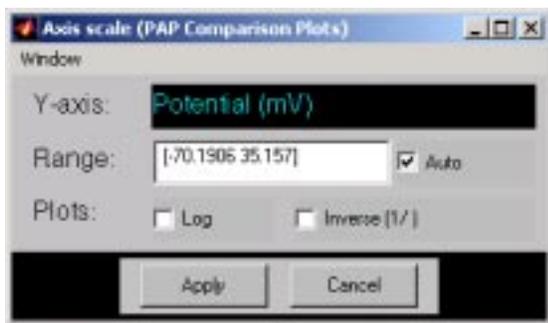
Figure 6.18: The PAP Axis Scale figure.

select the *Files with Parameters* option and click on the *Select files* pushbutton. A file-loading dialog box appears which allows the user to select the data file(s). The user can then plot the loaded solution variables.

Annotating, zooming, and other functions

The *PAP Comparison Plots* figure has a few functions that facilitates the analysis of the plots.

- Clicking on the *Cross-line* pushbutton ≈ displays a vertical line in the plot region at the location of the pointer cursor. The line follows the cursor as it moves across the plot region. The values of all plotted variables at the intersection with the cross-line are displayed in the Legend panel, under the column labeled *Cross-line* . Clicking on the *Cross-line* button again removes the cross-line values from the legend and removes the cross line from the plotting region.

- Clicking on the *A* button allows the user to make annotations on the plot. With this button set to *on*, click on a desired location in the plot to display a text edit box at that location. Click in the text edit box, type the annotation followed by a <RETURN> when the annotation is completed. Clicking on the annotation and dragging the mouse moves the annotation to a desired location in the plot field. The text string can be formatted into mathematical notation by using a LaTeX like notation, e.g., to get m_∞ type m_{\infty}.

- Clicking on the ⊕ button next to the annotation button allows the user to magnify a region of the plotting region by clicking on it. Clicking again causes further magnification. Clicking on the ⊖ button causes the plot to change to its original magnification.

- In the plotting region, clicking on any of the axis labels (i.e. either the abscissa or any of the ordinates) displays an axis scale figure (Figure 6.18) which allows the user to change parameters of the chosen axis. The range of the axis can be specified in MATLAB array format (i.e. [-200 200]) in the
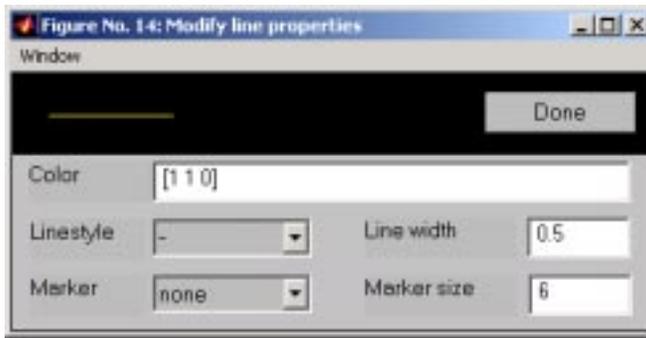
Figure 6.19: The Modify Line Properties figure.

Range field. The range is set to rescale automatically by default. To inhibit auto-scaling, unselect the *Auto* option. Selecting the *Log* option changes the axis scale to a logarithmic scale (i.e. the logarithm of the magnitude of the variable is plotted). Selecting the *Inverse [1/]* option produces a reciprocal scale axis. (i.e. the reciprocal of the magnitude of the variable is plotted). A combination of the two is allowed.

- In the Legend panel, clicking on any entry under the column labeled *Marker* displays a *modify line properties* figure (Figure 6.19) which allows the user to customize the color, line style, line width, marker type, and marker size of the selected plot curve. The color field is specified in RGB values in a MATLAB array (i.e. to specify the primary color, red, use [1 0 0]).

- In the Legend panel, clicking on any entry under the column labeled *Name* causes the curve associated with that entry's variable to alternately display and hide. Hiding and displaying curves causes the axes to rescale automatically unless the auto-scaling function of the particular axis is disabled by means of the axis scale figure described above.

Loading, saving, and printing

The user can save the plots by clicking on the *Save* pushbutton. All lines, axis scale and annotations are saved. The *Open* pushbuttons loads a previously saved file. The plots can be printed using the *Print* button.

6.4.11 Step-by-step Guide To Setting Up A Customized Simulation

Launch the software from the *Softcell* menu. Once the application has started, two figures will be displayed: the *PAP Control* figure, and the *PAP Workspace* figure. All model parameters are started in their default values. To start setting up a simulation with customized parameter values, complete all or any of the following steps:

1. Change the membrane, axon, and bath parameter values: in the *PAP Workspace* figure, click on the axon schematic. The *PAP Parameters* figure is displayed. Adjust any parameter as described in Section 6.4.2. Clicking on the axon schematic again closes the *PAP Parameters* figure.

2. Change the stimulus-pulse parameter values: in the *PAP Workspace* figure, click on the stimulus-source icon (yellow filled-circle with an arrow inside). The *PAP Stimulus* figure is displayed. Adjust any parameter as described in Section 6.4.4. Clicking on the stimulus-source icon again closes the *PAP Stimulus* figure.

3. Position the stimulus electrodes: in the *PAP Workspace* figure, select the appropriate radiobutton in the Electrode Positions panel (i.e. cathode/anode-electrode). To position the selected electrode longitudinally along the axon schematic, click on the black region between the top ruler and the schematic. The desired electrode moves to the position of the mouse cursor. Click on the electrode to toggle its position between intracellular and extracellular. Note that the icon of an intracellular electrode touches the axon schematic, whereas an extracellular electrode does not.

4. Position the voltage-recording electrodes: in the PAP Workspace figure, select the appropriate radiobutton in the Position Electrode panel. Click on the black region between the bottom ruler and the axon schematic to move the desired electrode to the position of the mouse cursor. Click on the electrode to toggle its position between intracellular and extracellular.

5. Change the numerics values: in the *PAP Control* figure, click on the *Numerics* option. The *PAP Numerics* figure is displayed. Adjust any parameter as described in Section 6.4.5.

6. Change the simulation duration: in the *PAP Control* figure, enter the simulation duration in milliseconds.

7. Start simulation: in the *PAP Control* figure, click on the pushbutton *Start*. Note that when a simulation has started, the *Start* pushbutton changes to a *Pause* pushbutton. Clicking on *Pause* pauses the simulation. Click on *Continue* to resume. Click on the *Stop* pushbutton anytime to abort.

6.5   Problems

Problem 6.1 *The purpose of this problem is to estimate the conduction velocity of the action potential.* Using the default parameters of the Hodgkin-Huxley model, devise a method to estimate the conduction velocity of the action potential. Describe your method carefully and give a numerical estimate of the conduction velocity for the default parameters.

Problem 6.2 *The purpose of this problem is to investigate the dependence of the propagated action potential on resistance per unit length of the inner and outer conductors and the relation of these results to the notion of a* space clamp.

   a. Run the simulation using the default parameters. Examine the membrane potential as a function of space and time. You may wish to use the three-dimensional plots or to use the comparison plots to plot the membrane potential as a function of time for locations from 0.5 cm to 2.5 cm every 0.2 cm.

   b. Reduce the resistivity of the intracellular conductor ρ_i to 10 Ohm·cm. Run the simulation again and examine the membrane potential as a function of space and time.

   c. Reduce the resistivity of the intracellular conductor ρ_i to 1 Ohm·cm. Run the simulation again and examine the membrane potential as a function of space and time.

   d. Reduce the resistivity of the intracellular conductor ρ_i to 0.1 Ohm·cm. Run the simulation again and examine the membrane potential as a function of space and time.

   e. Keep the resistivity of the intracellular conductor ρ_i to 0.1 Ohm·cm but increase the resistance per unit length of the outer conductor r_o to 10,000 Ohm/cm. Run the simulation again and examine the membrane potential as a function of space and time.

   f. What do you conclude about the dependence of the membrane potential on space and on time as the internal and external resistance per unit length are decreased?

   g. How are these results related to the notion of a space clamp? How are these results related to the notion of an *electrically large cell* and an *electrically small cell*.

   h. Explain how you could reduce the effective resistivity of the inner conductor ρ_i in an experiment on an axon.

Problem 6.3 *The purpose of this problem is to investigate the step response of the Hodgkin-Huxley model with the nonlinear elements removed.* Determine the stimulus and numerical parameters required to examine the step response of the Hodgkin-Huxley model when stimulated at $z = 1.5$ cm with the nonlinear elements removed, i.e., set $\overline{G}_{Na} = \overline{G}_K = 0$. Perform simulations with $\rho_i = 35.4$ Ohm·cm and for $\rho_i = 0.001$ Ohm·cm. Save the results in files. In the various parts of this problem, you will examine the solutions that you have already computed and saved.

a. For ρ_i = 35.4 Ohm·cm, examine the dependence of the membrane potential on space at various times by plotting V_m versus z at different times. Describe the results as precisely as possible.

b. For ρ_i = 0.001 Ohm·cm, examine the dependence of the membrane potential on space at various times by plotting V_m versus z at different times. Describe the results as precisely as possible. Compare these results with those of part a.

c. For ρ_i = 35.4 Ohm·cm, examine the dependence of the membrane potential on time at various positions by plotting V_m versus t at different positions. Describe the results as precisely as possible.

d. For ρ_i = 0.001 Ohm·cm, examine the dependence of the membrane potential on time at various positions by plotting V_m versus t at different positions. Describe the results as precisely as possible. Compare these results with those of part c.

Problem 6.4 *The purpose of this problem is to investigate the relations among the membrane potential, the intracellular potential, and the extracellular potential.*

a. Run the simulation with the default parameters.

 i. Examine the space-time evolution of V_m, V_i, and V_o. Describe the relation among these three variables. Explain these results.

 ii. Using the comparison plots, plot V_m, V_i, and V_o versus position at the time t = 0.4 ms and at the time t = 1 ms. Describe the relation among these three variables at the two times and explain these results.

b. Set r_o to equal the default value of r_i and repeat part a.

c. Compare the results in parts a and b and explain the differences.

Problem 6.5 *The purpose of this problem in to relate the membrane potential as a function of space to the membrane potential as a function of time during a propagated action potential.*

a. Run the simulation using the default parameters. Compare the membrane potential as a function of time at 2 cm with the membrane potential as a function of position at 1 ms. Determine the relation between these two waveforms.

b. Move the electrode S_+ (keeping it intracellular) to position 2.9 cm and run the simulation again. Once again compare the membrane potential as a function of time at 2 cm with the membrane potential as a function of position at 1 ms. Determine the relation between these two waveforms.

c. The relation between the membrane potential as a function of time and of position for part a differs from that in part b. Explain the difference.

Problem 6.6 *The purpose of this problem in to examine the relation between the membrane potential as a function of time and the longitudinal current as a function of time during a propagated action potential.*

a. Run the simulation using the default parameters.

  i. Compare the membrane potential as a function of time with the external longitudinal current I_o as a function of time at the position 2 cm. Determine the relation between these two waveforms.

  ii. Compare the external longitudinal current I_o as a function of time with the internal longitudinal current I_i as a function of time. Pay particular attention to $t < 0.5$ ms and to $t > 0.5$ ms. What is the relation of these two variables in these two time ranges. Explain these results.

b. Move the intracellular electrode (S_+) to position 2.9 cm and run the simulation again.

  i. Once again compare the membrane potential as a function of time with the external longitudinal current as a function of time at the position 2 cm. Determine the relation between these two waveforms.

  ii. Compare the results of parts a-i and b-i. Explain the differences.

Problem 6.7 *The purpose of this problem in to examine the relation among the membrane potential, the longitudinal currents, the membrane current density and the capacitance current all as functions of time during a propagated action potential.*   Run the simulation using the default parameters.

a. Using the comparison plots, compare V_m and I_i as a function of time at a location of 1.5 cm. What is the relation and why?

b. Using the comparison plots, compare V_m and J_C as a function of time at a location of 1.5 cm. What is the relation and why?

c. Using the comparison plots, compare I_i and J_C as a function of time at a location of 1.5 cm. What is the relation and why?

d. Using the comparison plots, compare I_i and J_m as a function of time at a location of 1.5 cm. What is the relation and why?

e. Using the comparison plots, compare V_m and J_m as a function of time at a location of 1.5 cm. What is the relation and why?

f. Using the comparison plots, compare I_i and I_o as a function of time at a location of 1.5 cm. What is the relation and why?

Problem 6.8 *The purpose of this problem is to examine the waveform of the propagated action potential at various locations along the axon.* Use the default parameters of the model except arrange the stimulating electrodes as follows: S_- is extracellular and located at position 1 (discrete segment 1 located at 0 cm) and S_+ is intracellular and located at position 20 (1 cm). The stimulus current provided by the current source (which can be seen by clicking on the current source) flows from S_- to S_+ and is 0.05 mA with a duration of 0.5 ms. Run the simulation and compare V_m at the locations 1, 1.5, 2, 2.5 cm by plotting all 4 traces on one axis using the Comparison Plots.

a. Does the *all-or-none principal* apply at all the positions along the axon?

b. Explain the differences in waveshape at the different locations along the axon. You may wish to plot additional variables on your plot of V_m.

Problem 6.9 *This problem concerns the direction of a current stimulus that elicits a propagating action potential.* The parameters of the model are the default values as indicated. Throughout this problem, the stimulus electrodes are located as follows: S_- is located at position 1 (discrete segment 1) and S_+ is located at position 30.

a. Simulations with one intracellular and one extracellular electrode.

  i. Set S_- to be extracellular and S_+ to be intracellular. The stimulus current provided by the current source (which can be seen by clicking on the current source) flows from S_- to S_+ and is 0.05 mA with a duration of 0.5 ms. Run the simulation and determine whether an action potential occurs and, if so, at which electrode(s) it arises.

  ii. With all other parameters the same as in part a-i, make S_- intracellular and S_+ extracellular. Run the simulation and determine whether an action potential occurs and, if so, at which electrode(s) it arises.

  iii. Draw a diagram of current flow about the axon and through the current source for the two cases considered in parts a-i and a-ii. Describe the conditions for the occurrence of an action potential and explain the polarity of V_m when an action potential does not occur.

b. Simulations with two extracellular electrodes.

  i. Set the stimulating electrodes and the stimulating current identical to that in part a-i except make both current electrodes extracellular. Run the simulation and determine whether an action potential occurs

and, if so, at which electrode(s) it arises. Examine the two longitudinal currents I_i and I_o and and the membrane current J_m and explain these results.

ii. With the default settings in part b-i, $r_o = 0$ which crudely simulates an axon in a large volume of sea water. This assumption is reasonable for computing whether or not an action potential occurs since only the sum $r_o + r_i$ occurs in these equations and for an axon in a large volume of sea water $r_o \ll r_i$. But assuming that $r_o = 0$ does not give a good approximation for determining longitudinal currents, for example. Therefore, in this part of the problem, increase r_o to 20,000 Ohm/cm so that it is roughly comparable to r_i. This value of r_o can be achieved experimentally by putting an axon in an insulating medium (oil, air, deionized sucrose) so that only a thin layer of sea water clings to the axon. Run the simulation and determine whether an action potential occurs and, if so, at which electrode(s) it arises. Examine I_i, I_o, and J_m. What do you conclude?

iii. Keep all the stimulus parameters the same as in b-ii except make both stimulating electrodes intracellular. Run the simulation and determine whether an action potential occurs and, if so, at which electrode(s) it arises.

iv. Return to the conditions of part b-ii except reverse the polarity of the stimulus current, i.e., make its amplitude -0.05 mA. Run the simulation and determine whether an action potential occurs and, if so, at which electrode(s) it arises.

c. Integrate the results you have obtained in parts a and b to come to a general conclusion about the key properties of an electric current stimulus that elicits an action potential.

Problem 6.10 *The purpose of this problem is to investigate what happens when two action potentials travelling in opposite directions collide.*

a. Start with the default parameters, and adjust the stimulus electrodes so that an action potential is produced both at 0 cm and at 3 cm. You will need to understand the direction of a current stimulus that elicits and action potential. You may wish to explore Problem 6.9 to aid you to understand this matter. [HINT: Note also that the default value for $r_o = 0$ so that little of the stimulus current from an extracellular electrode enters the axon. Therefore, you may wish to adjust the value of r_o.]

b. By plotting the results using the various plotting tools, determine what has happened. Did the action potentials pass through each other or did they annihilate each other? Suggest an explanation for your results.

Problem 6.11 *The purpose of this problem is to explore the bases of the sub-threshold oscillatory response of the model to a step of current.* Start with the default parameters except as follows: set the duration of the current pulse to 10 ms and its amplitude to 0.0001 mA. Set the duration of the simulation to 10 ms in the PAP Control figure. Run the simulation and examine the results. You should find that the membrane potential contains a highly damped slow oscillation at the onset which constitutes the step response. The purpose of this problem is to determine the basis of the oscillation.

    a. Devise an experiment to determine whether the relation of the stimulus current to the membrane potential is a linear relation for current stimulus parameters near those given above. What do you find?

    b. Investigate whether the variation in membrane potential with position is somehow responsible for the shape of the step response, in particular, the presence of the oscillations.

        i. Devise a simulation experiment that will eliminate the spatial variation of the membrane potential and run a simulation.

        ii. Demonstrate that the spatial variation you set out to eliminate is in fact eliminated.

        iii. Are the oscillations eliminated?

        iv. Which aspects of the step response are changed and why?

    c. If oscillations are still present in the step response, devise simulation experiments to determine their origin.

    d. Discuss how the results you have obtained in the previous parts are related to the cable model of an axon?

Problem 6.12 *The purpose of this problem is to explore the variables associated with a propagated action potential near the end of the model axon.* Use the default parameters to compute the propagated action potential.

    a. Using the comparison plots, plot I_i and I_o versus time at positions of 2.9 and 3 cm. Explain the differences in longitudinal currents with position.

    b. Using the comparison plots, plot J_m versus time at positions of 2.9 and 3 cm. Explain the differences in membrane current densities with position.

Problem 6.13 *The purpose of this problem is to investigate the space constant of the Hodgkin-Huxley model of the giant axon of the squid.*

a. Design a simulation experiment in order to estimate the space constant of the Hodgkin-Huxley model of the giant axon of the squid for the default parameters. Determine the value of the space constant.

b. Keeping all other parameters the same, set the radius of the axon to 0.0952 cm and repeat part a.

c. Compare the estimated space constants in parts a and b and explain the difference.

6.6   PROJECTS

This simulation software lends itself to more extensive study than that included in the Problems. These more extensive studies are called Projects. A description of the practical considerations in the choice of a project are contained in Section 5.6. Specific topics for projects with the software described in this Chapter are listed here.

1. Hypothesis — The effect of temperature on the conduction velocity of the squid giant axon can be fit by the Hodgkin-Huxley model. Articles in the literature should be consulted for this project (Chapman, 1967; Easton and Swenberg, 1975).

2. Hypothesis — When two action potentials are elicited, the velocity of the second is slower than the velocity of the first action potential. This phenomenon is predicted by the Hodgkin-Huxley model. Articles in the literature should be consulted for this project (George et al., 1984).

3. Hypothesis — The threshold current for eliciting an action potential with an intracellular electrode is higher for a space-clamped than for an unclamped model of an axon.

4. Hypothesis — Increasing the membrane capacitance will decrease the conduction velocity.

5. Hypothesis — Increasing the membrane conductance (by scaling all the ionic conductances) will increase the conduction velocity.

6. Hypothesis — Increasing the external concentration of sodium will increase the conduction velocity.

7. Hypothesis — Increasing the external concentration of potassium will increase the conduction velocity.

8. Hypothesis — Increasing the external concentration of calcium will increase the conduction velocity.

9. Hypothesis — Increasing the temperature will increase the conduction velocity.

10. Hypothesis — The solution using the forward Euler method is stable provided $\Delta t \leq \pi a(r_o + r_i)(\Delta z)^2 C_m$.

11. Hypothesis — The backward Euler algorithm is stable for all values of Δt and Δx.

12. Hypothesis — The Crank-Nicolson method and the staggered increment Crank-Nicolson method are stable for all values of Δt and Δx.

13. Hypothesis — Using the forward Euler approximation, the error resulting from temporal and spatial discretization behaves as $O(\Delta t) + O((\Delta x)^2)$.

14. Hypothesis — Using the backward Euler approximation, the error resulting from temporal and spatial discretization behaves as $O(\Delta t) + O((\Delta x)^2)$.

15. Hypothesis — Using the Crank-Nicolson method and the staggered increment Crank-Nicolson method, the error resulting from temporal and spatial discretization behaves as $O((\Delta t)^2) + O((\Delta x)^2)$.

Chapter 7

VOLTAGE-GATED ION CHANNELS
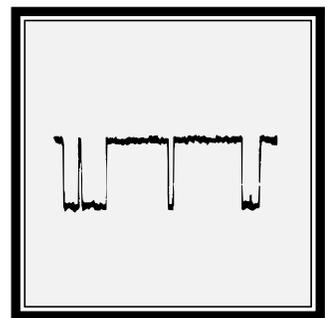
7.1   Introduction

7.1.1   Historical background

Passive transport of ions through cellular membranes is mediated by membrane-bound channel proteins that form selectively permeable ion channels. Although their existence was long suspected, the first direct physiological evidence for these channels became available in 1976 (Neher and Sakmann, 1976). Subsequently, many different types of channels have been identified (Hille, 1992). Channels can be categorized on the basis of the physical/chemical variable that opens or *gates* the channel. For example, the channels that underlie the properties of electrically excitable cells are gated by the membrane potential and are called *voltage-gated channels*. These channels are opened by a change in membrane potential. Other channels are gated by the binding of specific chemical substances, mechanical deformation of the membrane, etc. In all of these cases, the channels behave as if they have a discrete number of states, of which a few (usually only one) are conducting or open states and the others (usually more than one) are non-conducting or closed states. The channel switches rapidly and randomly among its allowable states, and the probability that a transition occurs at a particular time depends on the value of the gating variable.

Models of the gating of single ion channels were developed almost immediately after single-channel data were obtained (Colquhoun and Hawkes, 1977). Descriptions of these models can be found elsewhere (Colquhoun and Hawkes, 1995a; Weiss, 1996b).

7.1.2   Overview of software

This software package enables users to enhance their understanding of models of the gating of single voltage-gated ion channels. The software allows the user to design a channel by choosing the number of states, the voltage dependence of transitions between states, the conductance and gating charge of each state, and the equilibrium potential for the channel. After the channel is designed, the user can analyze the behavior of the channel to a change in membrane potential. Any of the variables associated with the channel can be displayed including the state occupancy probability, ionic conductance, ionic current, gating charge, and gating current. Both random variables and their expected values can be displayed as a function of time. The software is sufficiently powerful and flexible to allow users to synthesize realistic, state-of-the-art models for the gating of voltage-gated ion channels.

7.2 Description Of The Model

A brief description of the channel model is given here; a more complete description is given elsewhere (Weiss, 1996b). We assume that a channel has N states and that at each instant in time t, the channel is found in one of its states (Figure 7.1). The states are enumerated as $\{S_1, S_2, S_3, \ldots S_N\}$. Each state has state at-
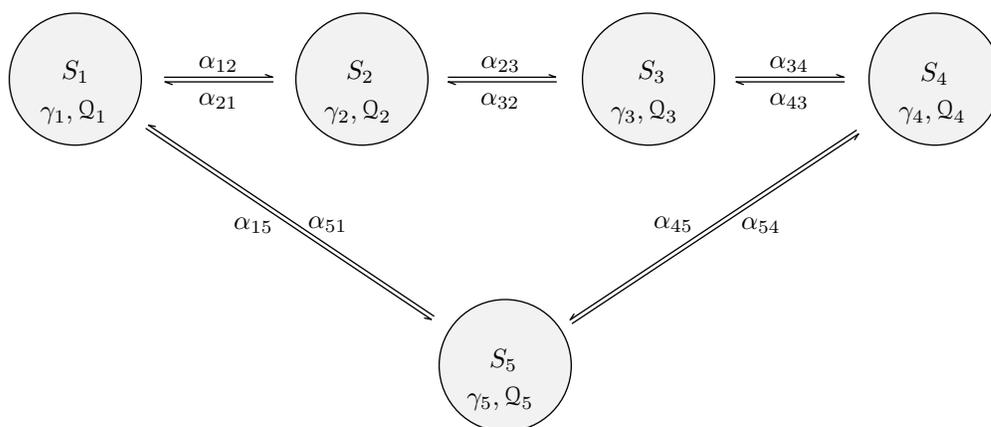
Figure 7.1: Kinetic diagram of a channel that has 5 states. The states are shown linked by rate constants for transitions between states, where α_{ij} is the rate of transition from state i to state j. When the channel is in state j the conductance of the channel is γ_j and the gating charge of the channel is $\mathcal{Q}_j$.

tributes: the state conductance and the state gating charge. The state attributes for these states are denoted as $\{\gamma_1, \gamma_2, \gamma_3, \ldots \gamma_N\}$ for the state conductances and $\{\mathcal{Q}_1, \mathcal{Q}_2, \mathcal{Q}_3, \ldots \mathcal{Q}_N\}$ for the state gating charges. Given the membrane potential V_m and the equilibrium potential V_{eq}, the state ionic currents are $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \ldots \mathcal{I}_N\}$. Thus, when the channel is in state S_j, the state conductance is γ_j, the state ionic current is $\mathcal{I}_j = \gamma_j(V_m - V_{eq})$, and the state gating charge is $\mathcal{Q}_j$. Gating currents occur at the transitions of the gating charge.

The temporal evolution of the electrical properties of a channel depend upon the state occupancy probabilities which we discuss briefly. Define the probability that the channel is in state j at time t as $x_j(t)$, and the transition probability that the channel is in state j at time $t + \Delta t$ given that it was in state i at time t as $y_{ij}(t + \Delta t)$. The fundamental assumption of the model for state transitions is that the transition probabilities depend on the present state of the channel and not on previous states. Such a probabilistic process is called a *Markov process*. Formally, we assume that

$$y_{ij}(t + \Delta t) = \alpha_{ij}\Delta t + o(\Delta t) ,$$

where $o(\Delta t)$ has the property that $\lim_{\Delta t \to 0} o(\Delta t)/\Delta t = 0$ and α_{ij} is a rate constant for transitions from state i to state j. The first term is the probability of

occurrence of a single transition in the interval of duration Δt when this interval is short and the rate of transitions is α_{ij}. The second term takes into account the probability of the occurrence of multiple transitions in the interval. For intervals Δt that are sufficiently small so that no more than one transition is likely to occur in the interval, the $o(\Delta t)$ term is negligible; we will ignore this term in the development that follows. The probability that the channel is in state j at time $t + \Delta t$ and in state i at time t is $y_{ij}(t + \Delta t)x_i(t) \approx \alpha_{ij}x_i\Delta t$.

The increase in probability that the channel is in state j in the time interval $(t, t + \Delta t)$ is $x_j(t + \Delta t) - x_j(t)$. In this time interval, this probability increases as a result of transitions from states i into state j and decreases as a result of transitions from state j into state i. By collecting all such terms we obtain,

$$x_j(t + \Delta t) - x_j(t) \approx \sum_{i=1,i\neq j}^{N} \alpha_{ij}x_i(t)\Delta t - \sum_{i=1,i\neq j}^{N} \alpha_{ji}x_k(t)\Delta t \ ,$$

where the first summation is due to transitions into state j and the second summation is due to transitions out of state j. If we rearrange the terms and take the limit as $\Delta t \to 0$, we obtain the differential equation

$$\frac{dx_j(t)}{dt} = \sum_{i=1,i\neq j}^{N} \alpha_{ij}x_i(t) - \sum_{i=1,i\neq j}^{N} \alpha_{ji}x_j(t). \tag{7.1}$$

This equation can be written in matrix form by defining the matrix of transition probabilities to be

$$\alpha = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1N} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ \alpha_{N1} & \alpha_{N2} & \cdots & \alpha_{NN} \end{bmatrix}.$$

where α_{ij} is the transition rate from state i to state j when $i \neq j$, and α_{ii} is chosen so that $\sum_{i=1}^{N} \alpha_{ij} = 0$. The state occupancy probability satisfies the matrix differential equation

$$\frac{dx(t)}{dt} = x(t)\alpha, \tag{7.2}$$

where $x(t)$ is a row matrix of state occupancy probabilities. In addition, since the channel is in one of its N states at each instant in time,

$$\sum_{j=1}^{N} x_j(t) = 1 \ .$$

The equilibrium state occupancy probability can be obtained directly from Equation 7.2 by setting $dx(t)/dt = 0$ to yield

$$0 = x(\infty)\alpha, \tag{7.3}$$

to obtain the equilibrium state occupancy probability $x(\infty)$.

If the matrix α is constant, then the solution to the matrix equation for the state occupancy probability is (Colquhoun and Hawkes, 1995a)

$$x(t) = x(0)e^{\alpha t},$$

where $x(0)$ is the matrix of the initial value of $x(t)$. Evaluation of the exponential matrix yields the result

$$x(t) = x(0) \sum_{j=1}^{N} S_j e^{\lambda_j t}, \tag{7.4}$$

where λ_j is the jth eigenvalue of and S_j is the jth spectral matrix of α. Therefore, we see that the state occupancy probabilities are weighted sums of exponentials whose exponents are the eigenvalues of the transition rate matrix. Because the α matrix is singular, one eigenvalue has value zero. All other eigenvalues have negative real parts (Cox and Miller, 1965). Thus, the non-zero eigenvalues lead to solution exponentials that decay with time.

For a voltage-gated channel, the transition rates depend explicitly on the membrane potential which in turn may vary with time. If the rate constants are assumed to satisfy the theory of absolute reaction rates (Weiss, 1996a; Weiss, 1996b), then they depend exponentially on the membrane potential,

$$\alpha_{ij}(V_m) = \begin{cases} A_{ij}e^{a_{ij}V_m} & \text{if } i \neq j, \\ 0 & \text{if } i = j, \end{cases}$$

where A_{ij} is a rate factor and a_{ij} is an exponential factor. $\alpha_{ij}(V_m)$ depends on the membrane potential which in turn depends upon time. Therefore, the rate constants will in general depend upon time. A more physical representation follows by linking the rate constants to the state gating charges. In this formulation, the rate constants that link states i and j (where $i \neq j$) can be written as

$$\begin{aligned} \alpha_{ij}(V_m) &= A e^{(0.5+\xi)(Q_j - Q_i)(V_m - V_{ij})/kT}, \\ \alpha_{ji}(V_m) &= A e^{(0.5-\xi)(Q_i - Q_j)(V_m - V_{ji})/kT}, \end{aligned}$$

where k is Boltzmann's constant, T is the absolute temperature, ξ is an asymmetry factor, Q_i and Q_j are the state gating charges in states i and j, V_{ij} and V_{ji} are the membrane potentials at which the rate constants equal A.

During intervals of time when the membrane potential is a constant, Equation 7.2 is a linear, first-order matrix differential equation with constant coefficients. The solution determines the state occupancy probabilities, the $\{x_j(t)\}$s. The expected values of the channel conductance, channel current, and gating charge are computed directly from the state occupancy probabilities. If the conductance of the channel is γ_j when the channel is in state j, then the expected

value of the conductance of the channel, $g(t)$, is

$$g(t) = \sum_{j=1}^{N} \gamma_j x_j(t). \tag{7.5}$$

If we assume that the current through the channel when the channel is in state j is $\gamma_j(V_m - V_{eq})$, the expected value of the single-channel current, $i(t)$, is

$$i(t) = \sum_{j=1}^{N} \gamma_j x_j(t)(V_m - V_{eq}) = g(t)(V_m - V_{eq}). \tag{7.6}$$

Similarly, if the gating charge of the channel when the channel is in state j is Q_j, the expected value of the single-channel gating charge, $q_g(t)$, is

$$q_g(t) = \sum_{j=1}^{N} Q_j x_j(t). \tag{7.7}$$

The expected value of the single-channel gating current, $i_g(t)$, is

$$i_g(t) = \frac{dq_g(t)}{dt}. \tag{7.8}$$

7.3   Numerical Methods

7.3.1   Integration step

The differential equation for the state occupancy probability (Equation 7.2) is solved numerically using methods described elsewhere (Colquhoun and Hawkes, 1995b). The accuracy and efficiency of the numerical solution depends upon the value of the integration step, Δt.   The value of Δt needs to be chosen small enough so that the time rate of change in the occupancy probabilities can be resolved accurately and so that the random variables are represented accurately. The choice of an adequate value of Δt depends upon the eigenvalues of the state transition rate matrix. Typically, Δt should be chosen so that $\Delta t \ll 1/|\lambda_{max}|$, where λ_{max} is the eigenvalue with the largest magnitude. The smaller Δt is chosen the more accurate the integration in any one step, but the longer the computation will take. In addition, long computations can result in the accumulation of round-off errors. Thus, the choice of Δt is a compromise between the speed and accuracy of the simulation. Solution of the matrix equation (Equation 7.4) determines the occupancy probabilities of each state. From these probabilities, the expected values of the conductances, currents and gating charge are computed using Equations 7.5-7.8.
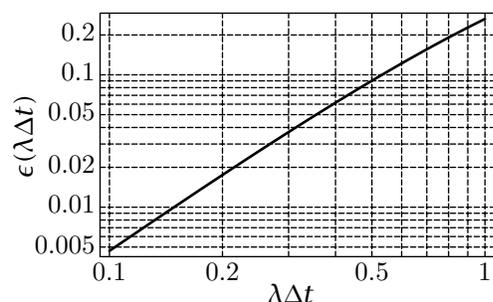
Figure 7.2: The probability of occurrence of more than 1 event in an interval Δt where the rate of events is λ and the events are distributed according to a Poisson distribution.

Random variables are computed from state transition probabilities. Transitions probabilities from state i to state j are computed from the relation $y_{ij}(t + \Delta t) \approx \alpha_{ij}\Delta t$ at each instant in time. The time interval for computing random variables is set equal to the integration step Δt. The occurrence of a transition is determined by a random number generator. The sequence of state transitions determines the sequence of conductance values and gating charges since each state has a unique conductance and gating charge. The conductance values determines the current since this current is determined by specifying the conductance, the membrane potential, and the channel equilibrium potential. The gating current random variable is represented by impulses at the times of transition of the gating charge and whose areas equal the discontinuities in gating charge.

In order to accurately represent the random variables, the time interval must be chosen sufficiently brief that the probability of occurrence of more than 1 change in state in the interval is negligible. The probability of occurrence of k events in an interval Δt with an average rate of events of λ is given by the Poisson distribution

$$\Pr(k, \lambda\Delta t) = \frac{(\lambda\Delta t)^k}{k!}e^{-\lambda\Delta t}. \tag{7.9}$$

Hence, the probability of occurrence of more than 1 event in an interval is

$$\epsilon(\lambda\Delta t) = 1 - \Pr(0, \lambda\Delta t) - \Pr(1, \lambda\Delta t) = 1 - (1 + \lambda\Delta t)e^{-\lambda\Delta t}. \tag{7.10}$$

The probability of more than one event in an interval $\epsilon(\lambda\Delta t)$ is shown plotted versus $\lambda\Delta t$ in Figure 7.2. Note that for $\lambda\Delta t \leq 0.2$, $\epsilon(\lambda\Delta t) \leq 0.018$.

7.3.2 Initial conditions

In order to obtain a solution to Equation 7.2, initial conditions must be specified. There are several plausible choices for the initial conditions, i.e., the initial states of the channel. One choice is simply to specify the initial state deterministically. A second choice is to choose the most probable state as the initial state. The most probable state can be obtained by solving Equation 7.3 and choosing the initial state as the one with the highest probability. A third choice is
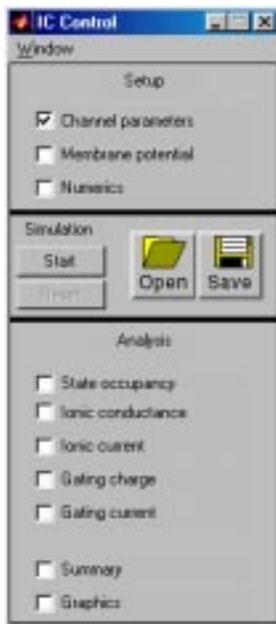
Figure 7.3: The IC Control figure after the software is initiated.

to choose the initial state probabilistically according to the equilibrium probability for state occupancy as determined by Equation 7.3. All three choices are implemented in the software.

7.4   User's Guide To The Software

When the software is initiated, 2 figures are displayed (in addition to MATLAB's command window): the *IC Control* and the *IC Parameters* figures. The *IC Control figure* controls the software (Figure 7.3). The *IC Parameters* figure allows the user to design a single channel. Once the channel is designed, the membrane potential is specified as a function of time, and the numerical variables chosen, the simulation can be run, and the statistical properties of the channel variables can be analyzed.

7.4.1   IC Control

The portion of the *IC Control* figure (Figure 7.3) below the menubar is divided into three panels. The setup panel allows the user to: select *Channel parameters* which is used to design the channel, *Membrane potential* which is used to specify the membrane potential as a function of time, and *Numerics* which is used to specify the initial conditions and the integration of the computation. A panel just below the setup panel controls the simulation. The simulation can be *started*, *stopped*, *paused*, or *continued*. Clicking on *Open* allows results saved in files to be read. Clicking on *Save* allows the user to save the results of the simulation in a file; all parameters of the channel design are saved as well as the
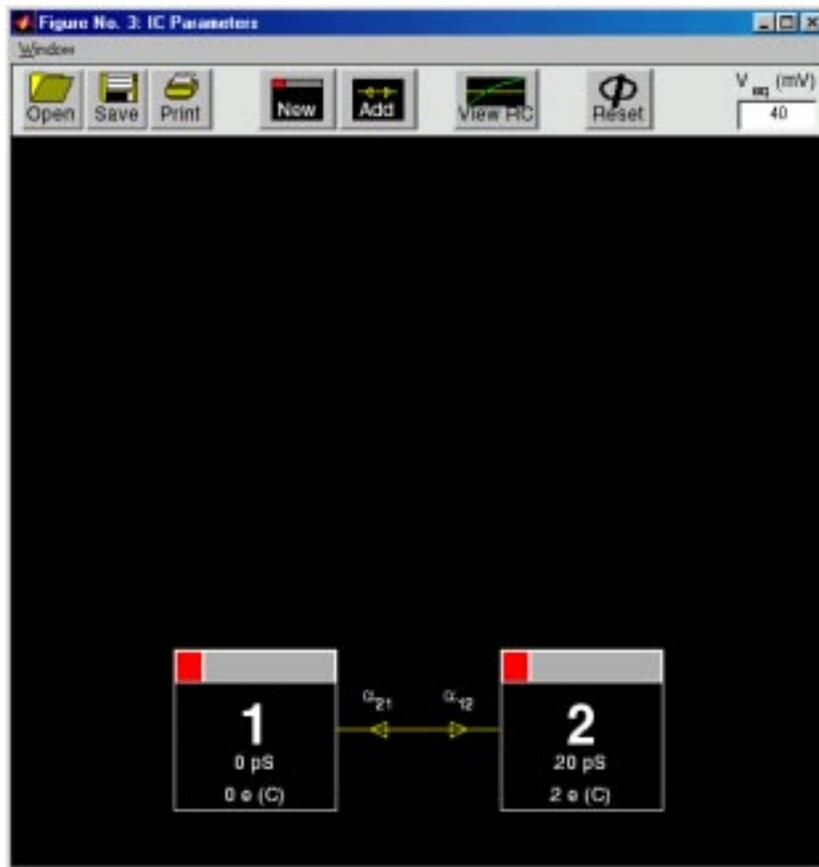
Figure 7.4: The IC Parameters figure.

simulation results. The analysis panel allows display of the results of a simulation.

7.4.2 IC Parameters

The *IC Parameters* figure displays the default channel kinetic diagram as shown in Figure 7.4. The user can either read a previously stored channel model from a file or construct a channel model by editing the default channel model. Editing entails adding new states, adding or modifying rate constants that link the states, assigning ionic conductances and gating charges to states, and assigning an equilibrium potential to the channel model.

Each box in the channel parameters figure (Figure 7.4) represents a particular state of the channel and is given a number from 1 to 8 (a maximum of 8 states is possible). The arrows connecting these states depict the transition rates between states. Each box also contains the ionic conductance of the channel when the channel is in that state, i.e., the state ionic conductance. The ionic conductance can be changed by clicking on it's value and typing the new value into the text

edit box. The ionic conductance is expressed in picosiemens (pS). A value of zero denotes a non-conducting state. Each box also contains the gating charge valence z of the channel when the channel is in that state; the gating charge valence can be changed by clicking on it's value and typing the new value into the edit box. The gating charge valence can be positive or negative and need not be an integer. The gating charge is ze, where $e = 1.602 \times 10^{-19}$ C. The value of the channel equilibrium potential can be changed by clicking in the text edit box in the upper right hand corner of the *IC Parameters* figure and typing the new value in the text edit box. A single click on a rate constants displays its voltage dependence in the *IC Parameters* figure.

Add new state

A new state can be added to the diagram by clicking on *New*. A new state will be created and positioned in the upper left corner of the field. A new state or any existing state can be repositioned in the window by clicking on the grey region at the top of the state boxes and dragging the state box to a new position. To delete a state the user clicks on the red delete box in the upper left-hand portion of the state box.

Add rate constant

To add rate constants linking state i to state j, the user first clicks on *Add* in the *IC Parameters* figure, then clicks in the state box for state i, and then clicks in the state box for state j. During this operation, the cursor becomes a cross hair. A pair of default rate constants is added to link these two states.

IC Specify rate constants

To modify the rate constants that link two states, the user double clicks on the selected rate constants to obtain the *IC Specify rate constants* figure (Figure 7.5). The user selects which rate constant to modify or to create by clicking on the appropriate pushbutton. The voltage dependence of the rate constant, can be specified numerically by typing numbers into the appropriate text edit boxes or graphically by using the mouse. Clicking on two locations in the plot of rate constant versus potential puts an exponential function through those two points. A formula for the dependence of the rate constant on membrane potential is displayed in the lower left-hand corner of the *IC Parameters* figure. This formula is also displayed by clicking on a rate constant in the *IC Parameters* figure.

 Clicking on *Delete* deletes the pair of rate constant from the *IC Parameters* figure. Clicking on the two curved linked arrows exchanges the two rate constants. Clicking on *Revert* resets the rate constant to the previously stored value.
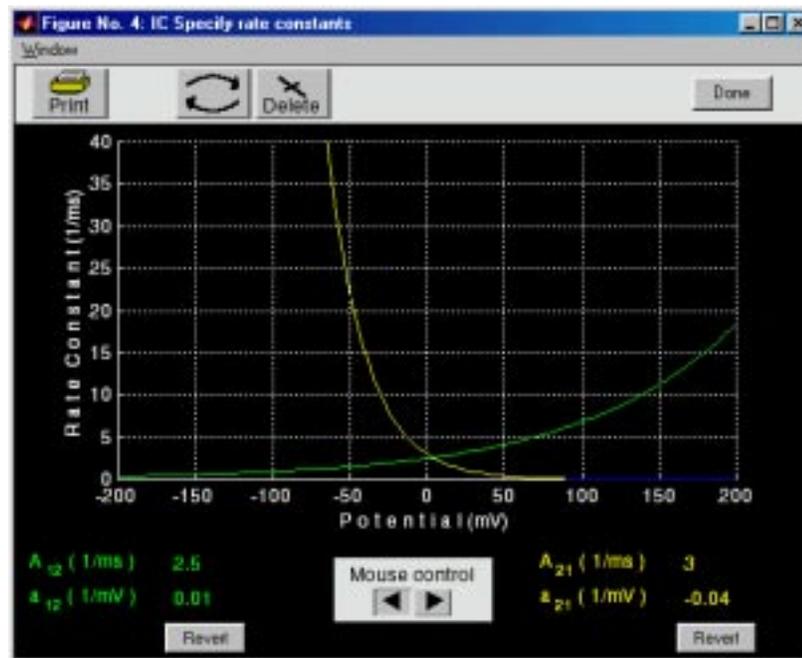
Figure 7.5: The IC Specify rate constants figure after the user has double clicked on the rate constants linking states 1 and 2 in Figure 7.4 and changed the parameters of the the rate constant so that $\alpha_{12} = 2.5e^{0.01V_m}$ (1/ms) and $\alpha_{21} = 3e^{-0.04V_m}$ (1/ms), where V_m is in mV..
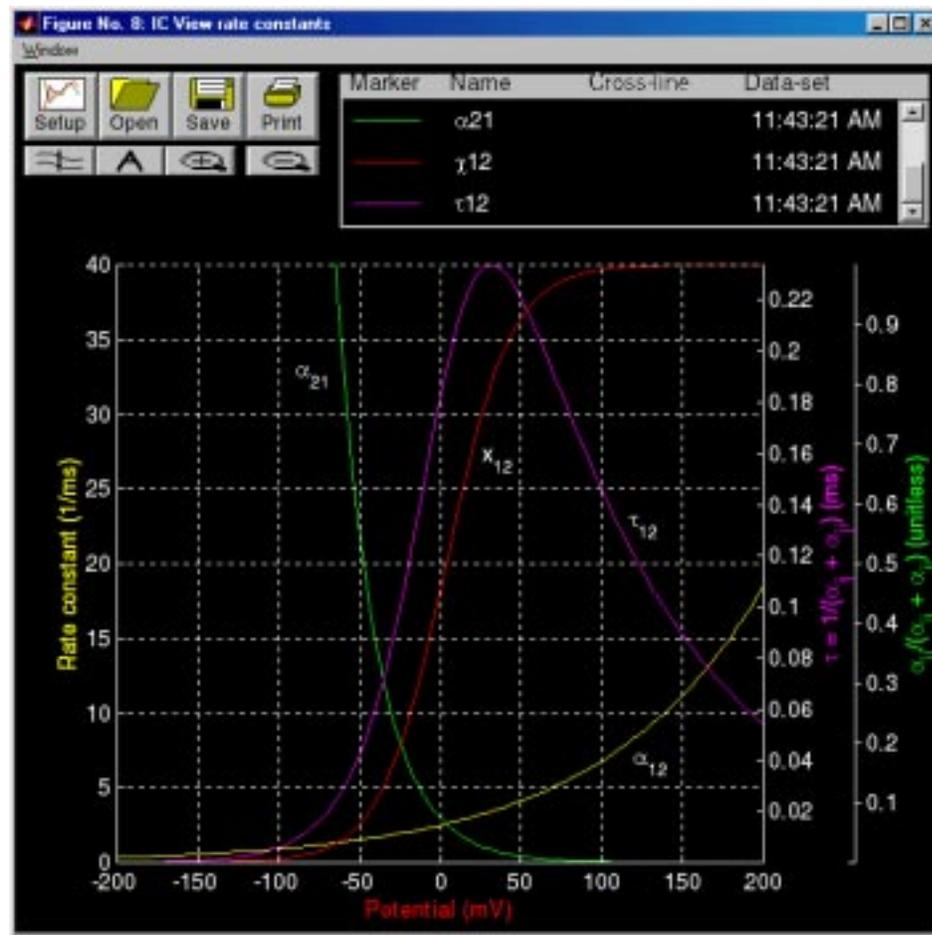
Figure 7.6: IC View rate constant figure after the rate constants have been defined as in Figure 7.5.

Miscellaneous features

If *Reset Diagram* is selected the current channel kinetic diagram is erased and the default channel kinetic diagram is drawn. A previously stored diagram can be retrieved by selecting *Open*. The current diagram can be saved to a file by selecting *Save*. Clicking on *Save* in the *IC Parameters* figure saves all the channels parameters but not the simulation results.

Viewing the voltage dependent parameters

The voltage dependent rate constants can be displayed by clicking on the button marked *ViewRC* in the *IC Parameters* figure which results in the display of the *IC View rate constants* figure (Figure 7.6). This figure contains a number of panels.
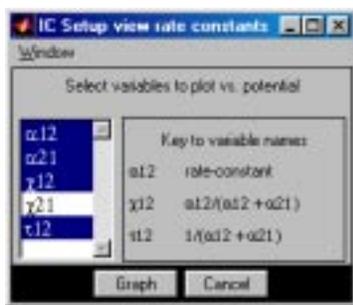
Figure 7.7: The IC Setup view rate constant figure.

Plot control. The panel in the upper left corner controls plotting in the manner described below.

> **Setup.** Clicking on *Setup* in the *IC View rate constants* figure displays the *IC Setup view rate constants* figure (Figure 7.7) which allows selection of the voltage-dependent parameters as a function of the membrane potential.
>
> > **Variables.** The abscissa is the membrane potential. Shift clicking on any variable adds it to the collection of selected ordinate variables. Clicking on any variable, selects that variable and deselects all others. The variables associated with each pair of rate constants are: α_{ij}, α_{ji}, $\tau_{ij} = 1/(\alpha_{ij} + \alpha_{ji})$, $x_{ij} = \alpha_{ij}/(\alpha_{ij} + \alpha_{ji})$, and $x_{ji} = \alpha_{ji}/(\alpha_{ij} + \alpha_{ji})$. τ_{ij} is the time constant for transition between states i and j provided there are only two states; if there are more than two states, then τ_{ij} has no particular significance. Similarly, x_{ij} is the probability that the channel will be in state j at equilibrium provided there are only two states; if there are more than two states, then x_{ij} has no particular significance.
> >
> > **Graph.** Clicking on Graph results in a plot of the selected data. The new plot replaces the old plot.
>
> **Open.** Clicking this button allows plotting of results stored in a file.
>
> **Save.** Clicking this button allows the information in the plot to be saved in a file. All lines, labels, and annotations are saved.
>
> **Print.** Clicking on Print prints the figure.
>
> **Cross-line.** Clicking on ≈ displays a vertical line, the cross line, in the plot field at the location of the pointer cursor. The line follows the cursor as it moves across the plot field. The values of all plotted variables at the intersection with the cross line are displayed in the legend. Clicking on cross-line button again removes the cross-line values from the legend and removes the cross line from the plotting field.
>
> **Annotate.** Clicking on A and then on a desired location in the plot area displays a text edit box at that location. Click in the text edit box, type

the annotation followed by a `<RETURN>` when the annotation is completed. Clicking on the annotation and dragging the mouse moves the annotation to a desired location in the plot field. The text string can be formatted into mathematical notation by using a LaTeX like notation, e.g., to get α_{12} type `\alpha_{12}`.

Zoom. Clicking on ⊕ allows the user to magnify a region of the plotting field by clicking on it. Additional magnification can be achieved by zooming again.

Unzoom. Clicking on ⊖ undoes the effects of all prior zooms.

Axes control. The plot axes can be changed by clicking on any of the axis labels — either the abscissa or any of the ordinates — which results in the display of a figure that allows changing the axis (see Figure 5.8). The axis scale can be chosen to be linear, logarithmic, and/or reciprocal. If neither the *Log* nor the *Inverse (1/)* options is chosen, the scale is linear. If *Log* is chosen, the scale is the logarithm of the magnitude of the variable. If *Inverse (1/)* is chosen then the reciprocal of the variable is plotted either on a linear or on a logarithmic scale. Clicking on *Apply* results in a change in the axis. The plot range can be changed, but the axis labels cannot be edited.

Legend. The legend panel records a list of all data plotted. The following are recorded.

Line properties. The color, line style, and symbol type are used to encode a particular variable; this marker is shown in the legend. Clicking on the marker allows the user to edit the attributes of the plot line/symbol (see Figure 5.9).

Variable name. The name of the variable that was plotted is shown. Clicking on the variable name alternately displays and hides the curve associated with this entry. When variables are added or deleted, the axis is auto-scaled if that option is selected — it is selected by default — in the axis figure.

Cross-line If the cross-line is used the cross-line values are displayed.

Data-set. The time when the curves were generated are indicated. Clicking on the time stamp alternately displays and hides all the curves associated with this legend item.

7.4.3   Membrane potential

Selection of *Membrane Potential* in the controls figure displays the *IC Membrane potential* figure (Figure 7.8). The membrane potential can be a rectangular pulse of potential superimposed on a steady holding potential. The user can select
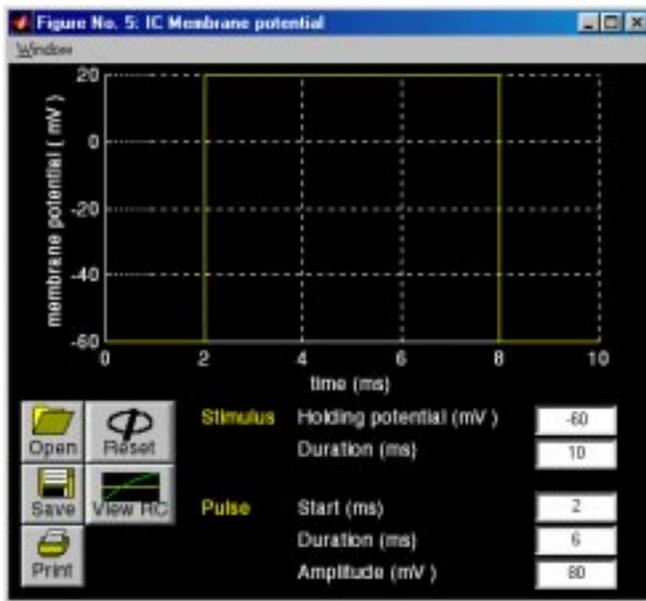
Figure 7.8: IC Membrane potential figure after selection of a pulse of potential.
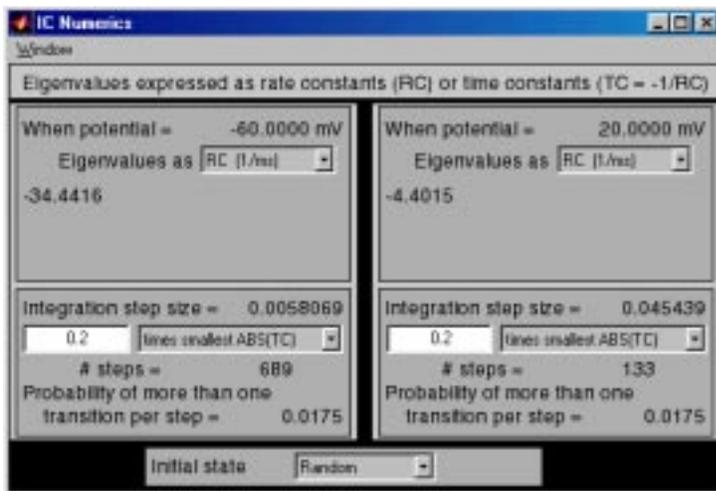
Figure 7.9: IC Numerics figure appropriate for rate constants shown in Figure 7.6 and membrane potential shown in Figure 7.8.

the holding potential, the duration of the stimulus, the onset time of the pulse starts, the pulse duration, and pulse amplitude. The parameters of the membrane potential can be read from and saved to a file.

7.4.4 Numerics

The *IC Numerics* figure (Figure 7.9) controls the choice of the integration step and initial conditions for the simulation. The figure consists of five panels. The two upper panels deal with the eigenvalues of the transition rate matrix. The middle panes deal with the method for choosing the integration step size. The bottom panel deals with the choice of initial conditions (initial states).

Eigenvalues

For a given choice of membrane potential and voltage-dependent rate constants, the software computes the eigenvalues of the transition rate matrix and displays these results to the user. These eigenvalues have the dimensions of rate constants (RC). The user can also choose to display the time constants (TC = −1/RC) corresponding to these eigenvalues. In addition, the absolute value of these rate and time constants can also be displayed.

Integration step

The smallest time constant is computed from the fastest eigenvalue and the user can choose, by editing the text edit box, to make the integration step a fraction (or a multiple) of this smallest time constant. Alternatively, the user can choose a fixed integration time. The integration time that results from the user's choice is displayed as is the probability of more than one transition during an interval of time that equals the integration step.

Initial state

The initial state for a simulation can be chosen deterministically to be any of the allowable states or the most probable state for the given initial values of the membrane potential. By default, the initial state is chosen probabilistically by computing the state occupancy probabilities for the chosen initial value of the membrane potential and choosing the initial state randomly using these probabilities.

7.4.5   Analysis

Clicking on the analysis panel of the *IC Control* figure allows access to a variety of displays of simulation results including such variables as: the state occupancy, the state occupancy probability, the ionic conductance, the ionic current, the gating charge, and the gating current. Both the random variables and their expected values can be displayed. Ten channels with similar characteristics can be examined simultaneously to allow users to relate macroscopic to microscopic variables. Results of multiple simulations can also be plotted on a single set of axes for the sake of direct comparison.

   Three different types of graphs are available. The first type allows plotting a single variable as a function of time in fixed coordinates. An example is the *IC State occupancy* figure shown in Figure 7.10. The top panel shows the simulation for 10 independent channels, the middle panel shows the results for channel 1 alone, and the bottom panel shows the state occupancy probabilities for both states of the channel. A second example of this type of graph is the *IC Ionic current* figure shown in Figure 7.11. Selected for plotting are the ionic
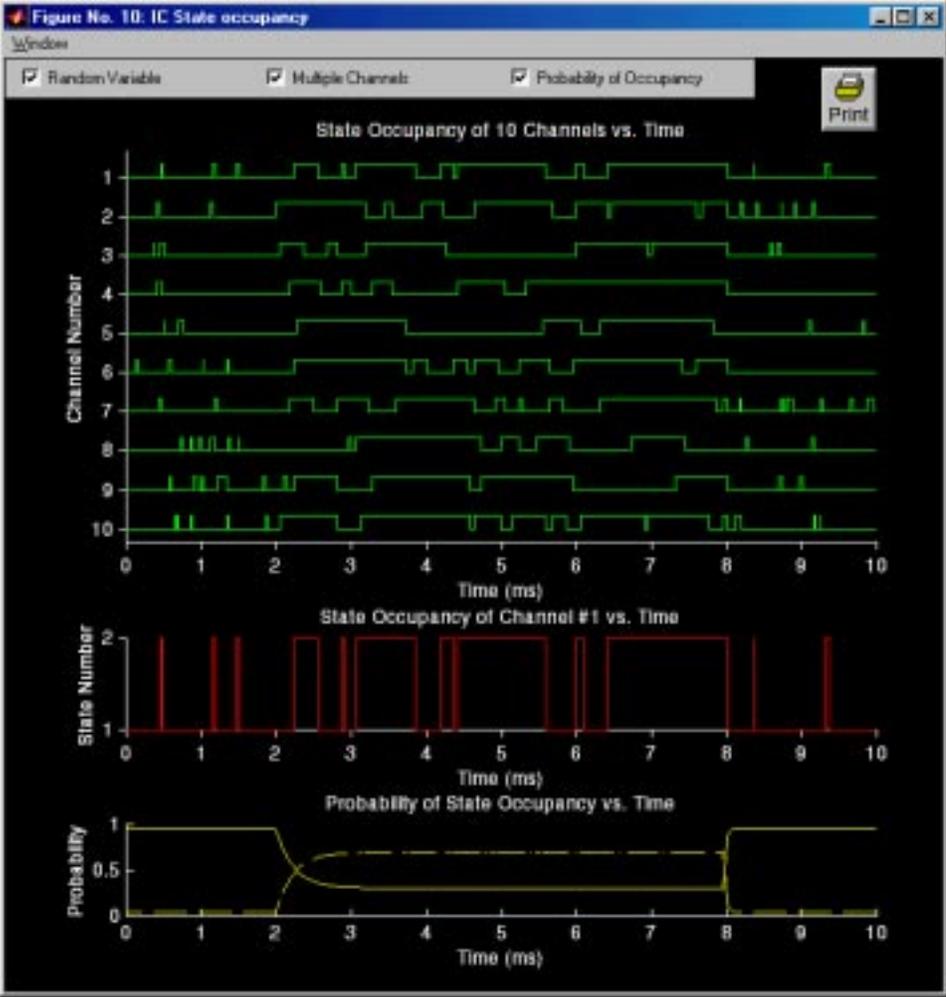
Figure 7.10: IC State occupancy figure with all variables selected for display.

Figure 7.11: IC Ionic current figure showing the relation between the random variable and its expected value of the ionic current.

current random variable, its expected value, and the average ionic current of 10 identically distributed but statistically independent channels. Similar plots are available for the ionic conductance, gating charge, and gating current. These displays are helpful for relating macroscopic to microscopic variables and random variables to their expected values.

A second type of graph is the *IC Summary* figure which allows comparison of the time dependence of channel variables to each other in fixed coordinates (Figure 7.12). Most of the variables are self-evident from the display. However, the representation of the gating current random variable requires some comment. The gating charge random variable is a random rectangular wave. Hence, the gating current random variable, which is defined as the derivative of the gating charge random variable, is a random impulse train (train of Dirac delta functions). We represent these impulses with vertical arrows. The time of occurrence of an arrow equals the time of occurrence of an impulse and the height of the arrow equals the area of the impulse. On the scale of visibility of the gating current random variable, the expected value is often too small to see. Hence, in the plot of the gating current, deselecting the random variable will allow its expected value to be observed. Clicking on the radio button for *Cross line* displays a vertical line that tracks the position of the mouse cursor. The values of the intersection of the line with each variable is displayed.

The above graphs have limited capability for control of scales. The ordinate
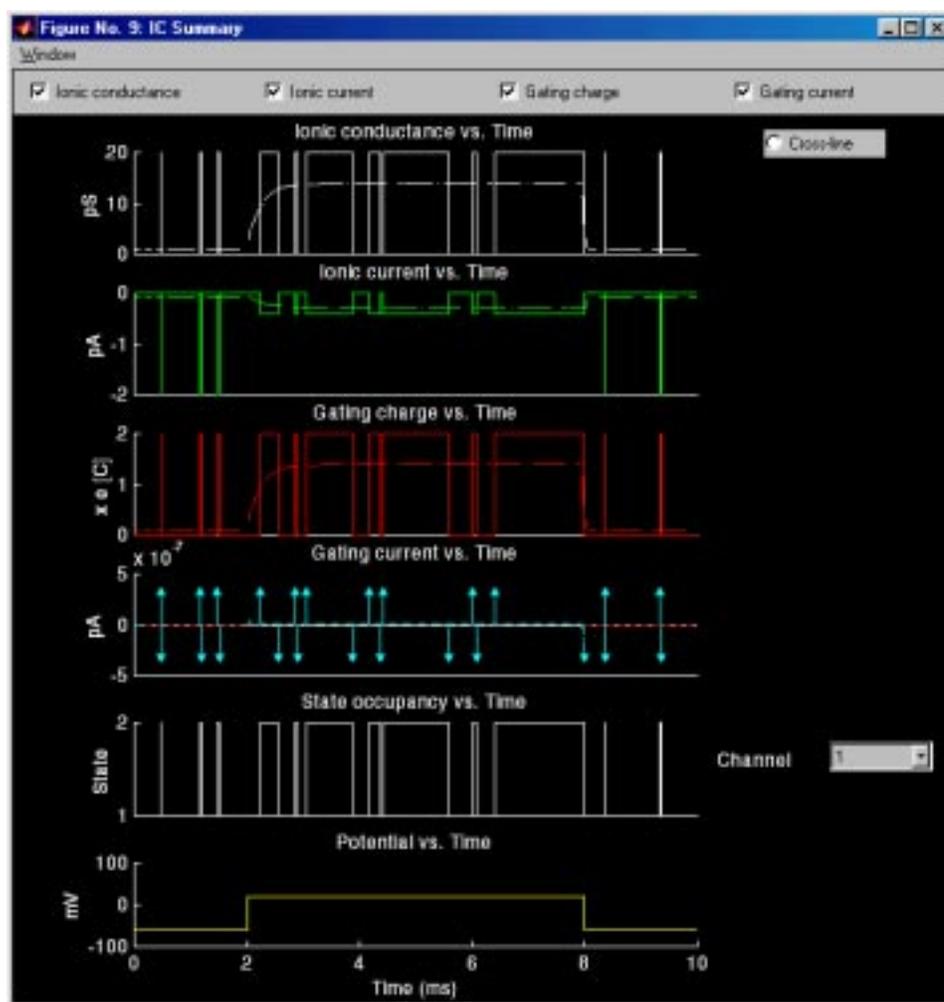
Figure 7.12: IC Summary figure with all variables selected for display.
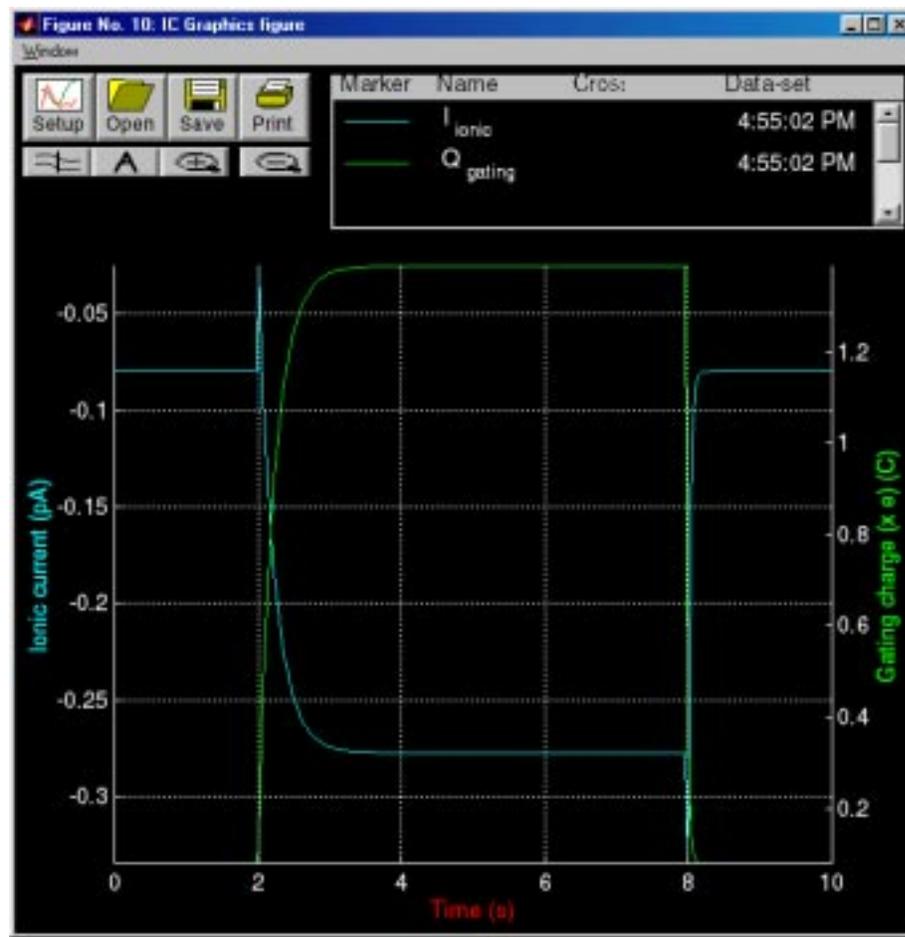
Figure 7.13: IC Graphics figure showing the expected value of the ionic current and of the gating charge for the channel defined in Figures 7.4-7.8.

scale is autoscaled automatically. This is convenient but can be misleading in that errors inherent in numerical calculations become autoscaled and can appear large if the scale is not read. For example, if for some set of parameters some current variable is in principle zero, its numerical value may be limited by the numerical accuracy of the computations to be 10^{-23} pA, a small current. However, if autoscaled in the above graphs, this current may appear large. The type of graph described below yields more control over plotting scales and can help the user to interpret such small currents.

The third type of graph allows comparison of results obtained in different simulations in coordinates defined by the user. Clicking on *Graphics* in the em IC Control figure displays the *IC Graphics* figure (Figure 7.13).

This figure allows plotting most of the results of the simulation; the random variables are not plotted with the *IC Graphics* figure but all the expected values and probabilities of all variables are plotted. For the allowable variables, any
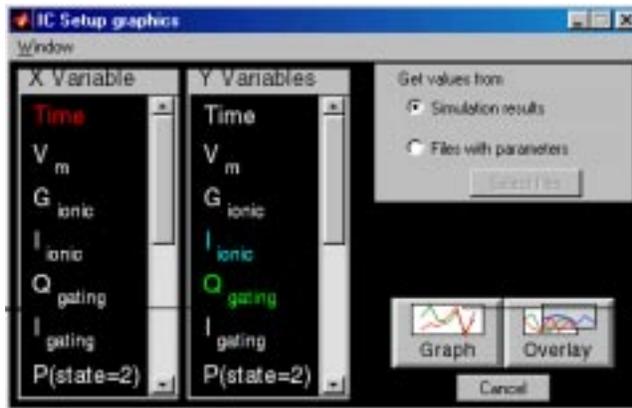
Figure 7.14: The IC Setup graphics figure showing the variables used to setup Figure 7.13.

variable can be plotted against any other in coordinates chosen by the user. The control of this figure is virtually identical to that for the *IC View rate constants* figure (Figure 7.6). Hence, we describe only the differences. The main difference occurs if *Setup* is selected in the *IC Graphics* figure which results in display of the *IC Setup graphics* figure (Figure 7.14). Several dependent variables can be plotted simultaneously. In addition, results saved in files can be plotted so that results for different parameters can be compared. In addition, results of consecutive simulations can be overlayed. We give two examples of the type of graphics that are readily available with the *IC Graphics figure*. Figure 7.15 shows the state occupancy probabilities of all four states as a function of time. Figure 7.16 shows the ionic conductances of three different channel models — a two-state, a three-state, and a four-state model.
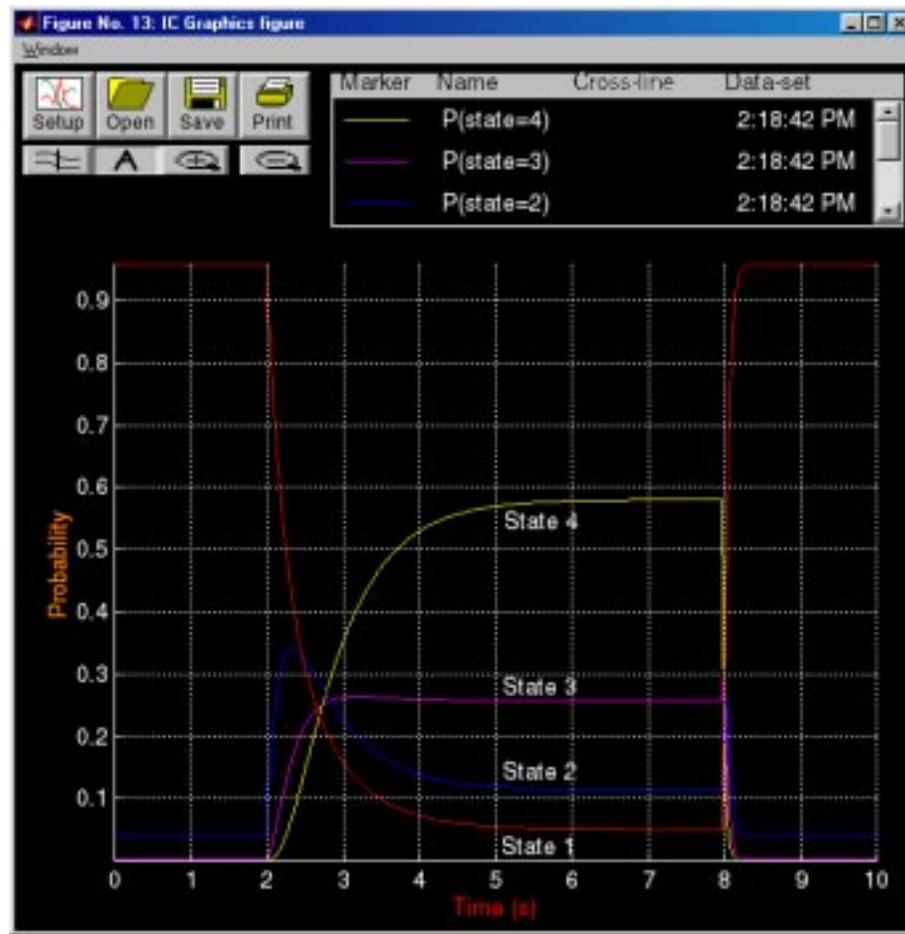
Figure 7.15: IC Graphics figure showing the state occupancy probabilities for a four-state channel whose successive states are linked by rate constant such that $\alpha_{12} = \alpha_{23} = \alpha_{34}$ and $\alpha_{21} = \alpha_{32} = \alpha_{43}$ where α_{12} and α_{21} are identical to those in Figure 7.5.
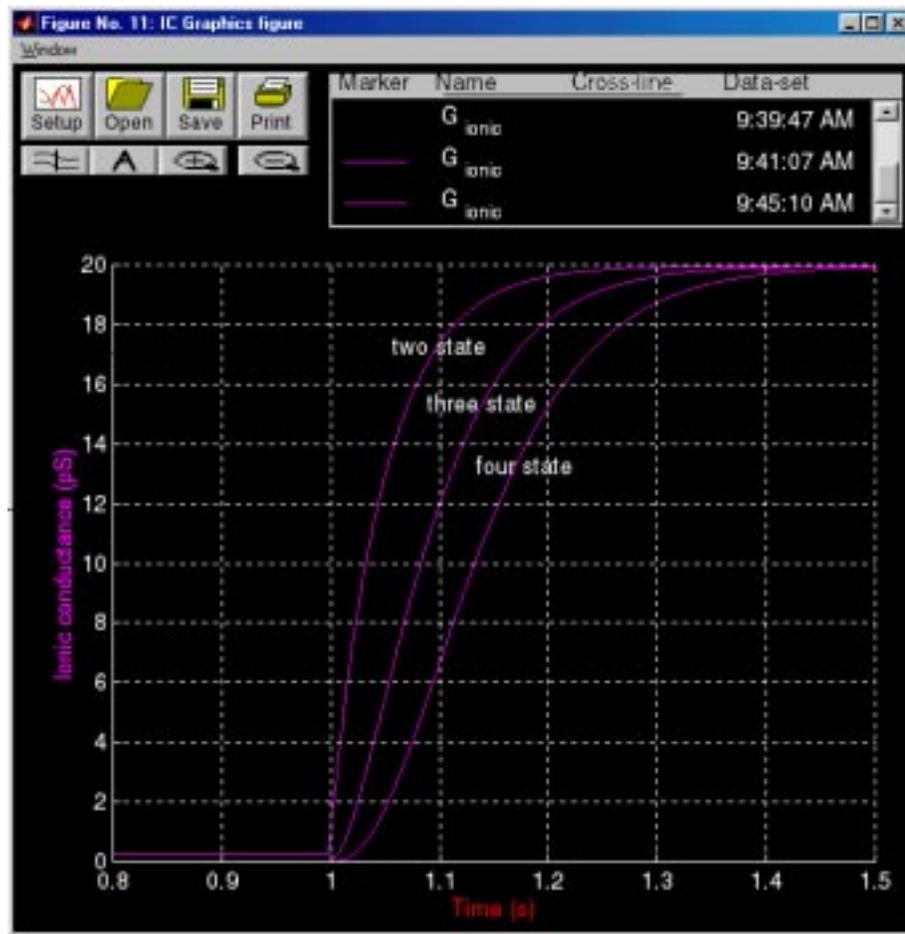
Figure 7.16: IC Graphics figure showing the ionic conductances of a two-state, three-state, and a four-state channel where the last state in each of these three channel models is a conducting channel with a conductance of 20 pS. $\alpha_{12} = \alpha_{23} = \alpha_{34} = 4e^{0.04V_m}$ and $\alpha_{21} = \alpha_{32} = \alpha_{43} = 0.2e^{-0.05V_m}$ (ms^{-1}). The membrane potential had at a holding potential of -80 mV and stepped to $+40$ mV.

7.5 Problems

Problem 7.1 *The purpose of this problem is to explore the properties of a channel that has one two-state gate that is not voltage gated.* This two-state channel has rate constants that are not a function of membrane potential; this channel is *not* voltage-gated. Define a two-state channel for which state 1 is closed (nonconducting) and state 2 is open (conducting),

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \,.$$
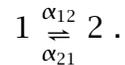
Use rate constants that are constant (independent of membrane potential). In all parts of this problem use rate constants such that $\alpha_{12} + \alpha_{21} = 10 \text{ ms}^{-1}$.

  a. Determine opening and closing rate constants such that the channel is equally likely to be in state 1 or state 2.

  b. Determine opening and closing rate constants such that the channel is three times as likely to be in state 2 than in state 1.

  c. Use the same rate constants as in part b. Set the membrane potential so that it is -50 mV for $t < 0.5$ ms and $+50$ mV for $t > 0.5$ ms. The objective of this part is to design a channel that has a single-channel current when the channel is open (I) of -1.2 pA when the membrane potential is -50 mV and $+0.8$ pA when it is $+50$ mV.

    i. Determine the equilibrium potential and single-channel conductance when the channel is open (γ). Check your answer by running the simulation using the values of the equilibrium potential and single-channel conductance you determined.

    ii. What is the average single-channel current (i) for $t < 0.5$ ms? What is the average single-channel current for $t > 0.5$ ms?

Problem 7.2 *The purpose of this problem is to explore the properties of a channel that has a two-state gate that is voltage gated.* This two-state channel has rate constants that are functions of membrane potential; the channel is voltage-gated. Define a two-state channel for which state 1 is closed (nonconducting) and state 2 is open (conducting),

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \,.$$
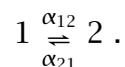
Set the membrane potential to -60 mV for $t < 0.2$ ms and to $+40$ mV for $t > 0.2$ ms.

a. Determine voltage-dependent rate constants such that: the channel is closed (with probability > 0.8) for $t < 0.2$ ms; the channel is open (with probability > 0.8) for $t \gg 0.2$ ms; and the channel opens with a time constant of about 0.2 ms.

b. Determine parameters of the channel such that when the channel is open, the single-channel current (I) is $+0.5$ pA at a membrane potential of $+40$ mV and is zero when the membrane potential is -10 mV.

c. Is this an activation gate or an inactivation gate?

d. Change the equilibrium potential to -70 mV. Repeat the simulation. Could this channel describe the kinetics of the potassium channel in the Hodgkin-Huxley model? In what ways does it resemble the potassium channel? In what ways does it differ from the potassium channel?

e. Change the equilibrium potential to $+50$ mV. Repeat the simulation.

   i. Is the average channel conductance a continuous function of time? Explain!

   ii. Is the average channel current a continuous function of time? Explain!

   iii. Could this channel describe the kinetics of the sodium channel in the Hodgkin-Huxley model? In what ways does it resemble the sodium channel? In what ways does it differ from the sodium channel?

Problem 7.3 *The purpose of this problem is to explore the properties of two two-state voltage-gated channels that differ only in their state conductances.* This problem deals with two channels that each have two states with the same kinetic diagram, namely

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \, ,$$
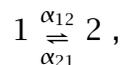
where the rate constants are

$$\alpha_{12} = 4e^{0.04V_m} \text{ and } \alpha_{21} = 2e^{-0.01V_m} \, .$$

Each channel is subject to a membrane potential profile that is -60 mV from $t = 0$ to $t = 1$ ms after which the membrane potential is $+10$ mV for 2 ms. The equilibrium potential for each channel is $+40$ mV. The channels differ only in the conductance assigned to each state. Channel 2 has $\gamma_1 = 0$ and $\gamma_2 = 20$ pS; Channel 2 has $\gamma_1 = 20$ pS and $\gamma_2 = 0$.

a. Which channel opens in response to a depolarization? Explain.

b. Which channel closes in response to a depolarization? Explain.

c. Does the time-dependence of the probability that the channel is in state 1 differ for these two channels? Explain.

Problem 7.4 *The purpose of this problem is to explore the effect of scaling the rate constants on the kinetic properties of a channel.* The kinetic diagram has the form

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 .$$
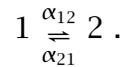
The membrane potential is -60 mV from $t = 0$ to $t = 1$ ms and $+10$ mV for 2 ms. The equilibrium potential is $+40$ mV. The channel conductances are $\gamma_1 = 0$ and $\gamma_2 = 20$ pS.

a. Set the voltage dependence of the rate constants as

$$\alpha_{12} = 4e^{0.04V_m} \text{ and } \alpha_{21} = 2e^{-0.01V_m} ,$$

and run the simulation. Plot the ionic conductance and ionic current. Record the values of eigenvalues.

b. Next double the rate constants so that the voltage dependence of the rate constants becomes

$$\alpha_{12} = 8e^{0.04V_m} \text{ and } \alpha_{21} = 4e^{-0.01V_m} .$$

Run the simulation again. Plot the ionic conductance and ionic current. Record the values of eigenvalues.

c. Next halve the rate constants so that the voltage dependence of the rate constants becomes

$$\alpha_{12} = 2e^{0.04V_m} \text{ and } \alpha_{21} = 1e^{-0.01V_m} .$$

Run the simulation again. Plot the ionic conductance and ionic current. Record the values of eigenvalues.

d. Examine the results and draw conclusions about the effect of scaling the rate constants without otherwise changing their voltage dependence. Focus both on the random variables and on the expected values of both the ionic conductance and the ionic currents. Be as quantitative as possible.

Problem 7.5 *The purpose of this problem is to explore the effect of the choice of initial state on the properties of a channel with a simple, two-state gate.*

a. Start the simulation with the default parameters. Examine the Summary plot and the State occupancy plot.

b. Now display the IC Numerics figure. Note that the default initial state is *Random.* Now change the initial state to be state 1. Start the simulation.

    i. Describe the differences between the results obtained with a random initial state and an initial state of 1. What accounts for this difference.

    ii. Carefully display the time course of the state occupancy probability using the Graphics figure and determine the time constant of the time course of the state occupancy probability.

    iii. Compare the time constant you estimated in part ii with the eigenvalue (expressed as a time constant) from the Numerics figure.

Problem 7.6 *The purpose of this problem is to explore the gating charge of a channel that has one two-state gate.* The rate constants of the channel are $\alpha_{12} = 1$ and $\alpha_{21} = 2$ ms^{-1}. The gating charge of state 1 is 0 and that of state 2 is $2e$C.

a. Run the simulation and determine the expected value of the gating charge. Explain its value.

b. Without changing any other parameter, determine a value of α_{12} such that the expected value of the gating charge is $1e$C.

c. Set both $\alpha_{12} = \alpha_{21} = 2$ ms^{-1} and set the gating charge of both state 1 and 2 to $2e$C. Run the simulation. Explain the expected value of the gating charge. [Hint — you may find it convenient to use the *Graphics* plots (which you can select from the *IC Control* figure. This option allows a great deal of control over axes scales which may be important to interpret the results.]

Problem 7.7 *The purpose of this problem is to explore the relation among ionic conductance, ionic current, gating charge, and gating current for a channel that has one two-state gate.* The rate constants are $\alpha_{12} = 1$ and $\alpha_{21} = 2$ ms^{-1}. The gating charge of state 1 is 0 and that of state 2 is $2e$C. The ionic conductance of state 1 is 0 and that of state 2 is 20 pS. Run the simulation. [Hint — for some parts of this problem, you may find it convenient to use the *Graphics* plots (which you can select from the *IC Control* figure). This option allows a great deal of control over axes scales which may be important to interpret the results. The other plots have autoscaled axes in which small errors, inherent in numerical calculations, are autoscaled to appear large.]

a. Examine the expected value of the ionic conductance.

    i. Describe the ionic conductance.

    ii. Explain its value.

b. Examine the expected value of the ionic current.

    i. Describe the ionic current.

   ii. Explain its values.

  c. Examine the expected value of the gating charge.

    i. Describe the gating charge.

   ii. Explain its value.

  d. Examine the expected value of the ionic current.

    i. Describe the ionic current.

   ii. Explain its value.

  e. Note that the expected value of the ionic conductance and gating charge are non-zero constants, the expected value of the gating current is zero, and the expected value of ionic current is not constant. Explain these results.

Problem 7.8 *The purpose of this problem is to explore the physical properties of the gating charge.* The kinetic diagram for the channel has the form

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2,$$
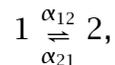
where

$$\alpha_{12} = 4e^{0.04V_m} \text{ and } \alpha_{21} = 2e^{-0.01V_m} ,$$

The membrane potential is -60 mV from $t = 0$ to $t = 1$ ms and $+10$ mV for 2 ms. The stimulus duration is 3 ms. The equilibrium potential is $+40$ mV. The channel conductances are $\gamma_1 = 0$ and $\gamma_2 = 20$ pS.

  a. Perform simulations using the following state gating charges. Plot both the expected value of the gating charge and the expected value of the gating current.

    i. $\mathcal{Q}_1 = 0$ and $\mathcal{Q}_2 = 2e$C.

   ii. $\mathcal{Q}_1 = 0$ and $\mathcal{Q}_2 = -2e$C.

   iii. $\mathcal{Q}_1 = 2e$C and $\mathcal{Q}_2 = 0$.

   iv. $\mathcal{Q}_1 = -2e$C and $\mathcal{Q}_2 = 0$.

  b. Give a quantitative account of the key properties of the gating charge and gating current waveforms — the initial values, final values, and time constants.

  c. Of the results obtained in part a, which are physically plausible results. Explain.

Problem 7.9 *The purpose of this problem is to explore the eigenvalues of channel models.*

a. Consider a two-state channel with the rate constants $\alpha_{12} = 1$ and $\alpha_{21} = 2$ (ms^{-1}) and the default membrane potential shown in Figure 7.8.

   i. What are the eigenvalues for the two values of the membrane potential?

   ii. Why are the eigenvalues at the two membrane potentials the same? Check your explanation by changing the membrane potential and determining the eigenvalues.

   iii. What is the relation of the eigenvalue to the rate constants? Check your relation by changing the rate constants and determining the eigenvalue.

   iv. Design a simulation experiment that computes the temporal change of some variable of the two-state model to reveal the significance of the eigenvalue.

b. Consider a two-state channel with the rate constants $\alpha_{12} = 1.6818e^{0.008664V_m}$ and $\alpha_{21} = 2$ (ms^{-1}) where V_m is in mV and the default membrane potential is shown in Figure 7.8.

   i. What are the eigenvalues for the two values of the membrane potential?

   ii. Why are the eigenvalues at the two membrane potentials different? Check your explanation by changing the membrane potential and determining the eigenvalues.

   iii. What is the relation of the eigenvalues to the rate constants? Check your relation by changing the rate constants and determining the eigenvalue.

   iv. Design a simulation experiment that computes the temporal change of some variable of the two-state model to reveal the significance of the eigenvalue. Which eigenvalue determines the temporal response?

c. Consider a three-state channel with the default rate constants $\alpha_{12} = 1$, $\alpha_{21} = 2$, $\alpha_{23} = 2$, and $\alpha_{32} = 1$ (ms^{-1}) and the default membrane potential shown in Figure 7.8.

   i. What are the eigenvalues for the two values of the membrane potential?

   ii. Why are the eigenvalues at the two membrane potentials the same? Check your explanation by changing the membrane potential and determining the eigenvalues.

   iii. Determine a relation that computes the eigenvalues from the rate con-
        stants? Check your relation by changing the rate constants and deter-
        mining the eigenvalues.

Problem 7.10 *The purpose of this problem is to explore the properties of a three-
state channel whose rate constants are not a function of membrane potential; this
channel is not voltage-gated.*   Define a three-state channel with the following
kinetic scheme

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \underset{\alpha_{32}}{\overset{\alpha_{23}}{\rightleftharpoons}} 3 \ ,$$
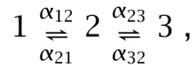
where states 1 and 2 are closed and state 3 is open. In all parts of this prob-
lem, make the rate constants independent of membrane potential, and use rate
constants in the range 0.1-10 ms^{-1}.

   a. Determine a set of rate constants such that the channel is equally likely to
      be in any one of its three states.

   b. Try a set of rate constants such that $\alpha_{ij} = \alpha_{ji}$. What are the probabilities
      of occupying each state? Try another such set having this property. What
      are the probabilities of occupying each state with this new set? What can
      you conclude about this case?

   c. Determine a set of rate constants such that the probabilities that the chan-
      nel is in states 1:2:3 are approximately 0:0.5:0.5.

   d. Determine a set of rate constants such that the probabilities that the chan-
      nel is in states 1:2:3 are approximately 0:0.25:0.75.

   e. Determine a set of rate constants such that the probabilities that the chan-
      nel is in states 1:2:3 are approximately 0.2:0.3:0.5.

Problem 7.11 *The purpose of this problem is to explore the properties of voltage-
gated channels with two, three, and four states.*   Assume that the equilibrium
potential of each channel is 40 mV and that the membrane potential starts at
-60 mV and switches to $+10$ mV at 1 ms and is maintained at this value for 2
ms. The three channels are defined as follows:

  • The two-state voltage-gated channel has a kinetic diagram

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \ ,$$
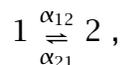
   where the rate constants are

$$\alpha_{12} = 4e^{0.04V_m} \text{ and } \alpha_{21} = 2e^{-0.01V_m} \ .$$

- The three-state voltage-gated channel has a kinetic diagram

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \underset{\alpha_{32}}{\overset{\alpha_{23}}{\rightleftharpoons}} 3 \ ,$$
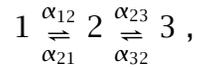
  where the rate constants are

$$\alpha_{12} = \alpha_{23} = 4e^{0.04V_m} \text{ and } \alpha_{21} = \alpha_{32} = 2e^{-0.01V_m} \ .$$

- The four-state voltage-gated channel has a kinetic diagram

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \underset{\alpha_{32}}{\overset{\alpha_{23}}{\rightleftharpoons}} 3 \underset{\alpha_{43}}{\overset{\alpha_{34}}{\rightleftharpoons}} 4 \ ,$$
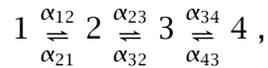
  where the rate constants are

$$\alpha_{12} = \alpha_{23} = \alpha_{34} = 4e^{0.04V_m} \text{ and } \alpha_{21} = \alpha_{32} = \alpha_{43} = 2e^{-0.01V_m} \ .$$

In exploring the following issues, it is instructive to examine graphs that show the membrane potential as well as both the random variables and mean values or probabilities associated with the record of state occupancies, the conductance, and the current through the channel.

a. For each of the channels, make the last state (the state with the highest state number) the conducting state with a conductance of 20 pS and make all other states non-conducting. Compare the mean conductance of the two-state, the three-state, and the four-state channels. What can you say about the effect on channel conductance of increasing the number of closed states that precede the open state?

b. Now consider the four-state channel. Compute the mean conductance when state 1 has a conductance of 20 pS and all other states are non-conducting. Repeat this computation except make state 2 the conducting state with all others nonconducting. Repeat again with state 3 and then state 4. Compare the mean conductance for these four conditions. Which of these conductances most closely resembles the sodium conductance of squid axon? Explain.

Problem 7.12 *The purpose of this problem is to design a three-state channel that activates and then inactivates.* The channel activates at about 0 mV at a rate that is three times as fast as it inactivates.

a. Consider a channel that must open before it inactivates, e.g.

$$1 \underset{\alpha_{21}}{\overset{\alpha_{12}}{\rightleftharpoons}} 2 \underset{\alpha_{32}}{\overset{\alpha_{23}}{\rightleftharpoons}} 3 \ ,$$
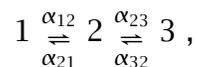
  where both states 1 and 3 are closed and state 2 is open.

b. Consider a channel that need not open before it inactivates.

Chapter 8

PEDAGOGICAL METHODS

8.1   Introduction

This chapter describes the pedagogical methods that were developed to teach cellular biophysics using the software described in this text. The chapter is aimed primarily at instructors who wish to use this material in teaching although students may find the material helpful. The description given here expands that available previously (Weiss et al., 1992).

8.2   Uses Of The Software

The software has become an integral part of teaching cellular biophysics. It is used throughout the subject, wherever it can be didactically effective. The software is used in lecture to motivate, engage, and inform students. The software is helpful to visualize and to intuit dynamic processes. Use of the software in special classes held in an electronic classroom is an effective way to get students away from solving problems by pattern recognition (Mazur, 1997) and forcing them to confront the material in a different way. Use of the software in projects can deepen students understanding of the subject material and can be used to teach students how to go about a research project.

8.2.1   Lectures

We have used a lecture room with a large screen (13.6'×18') for projecting a computer monitor screen. Thus, students are able to see all the interactions of the lecturer with the computer. We have used the software in a variety of ways: to help students to visualize concepts, to build intuition, to motivate, and to engage the students' intellect. We illustrate some of these techniques.

Some concepts are grasped instantly when presented graphically. The intuition that results is complementary to that which results from a mathematical exposition. The combination of an intuitive, visual grasp of a concept and the precise, abstract mathematical description is powerful. Computer simulations can be especially helpful with building intuition.

> **Example 8.1** In the random-walk model of diffusion with an unbiased random walk, each particle moves one step to the left or to the right with equal probability; the particle has no preferred direction of motion. Yet a population of diffusing particles tends to move down the particle concentration gradient. Thus, there is an apparent paradox between the undirected motion of individual particles and the directed motion of a population of particles. This apparent paradox is resolved in an instant by viewing a population of particles undergoing random walks (Figure 8.1). By watching the simulation, the user
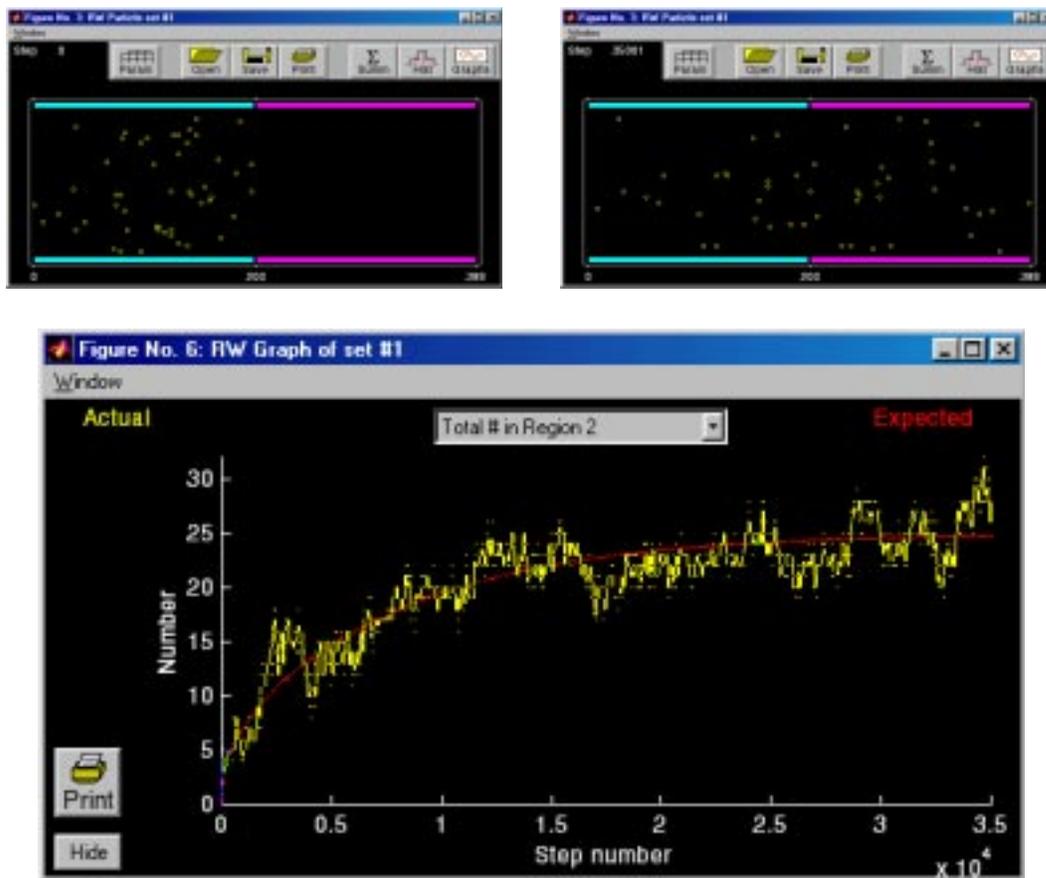
Figure 8.1: Random walk in two regions. The initial distribution of particles (upper left) shows 50 particles in the left region and none in the right region. After 35001 steps, the particles have been dispersed in the two regions (upper right). The number of particles in the right region (lower) zero when the step number is zero and approaches 25 at a large step number. The rate of increase of particles in the right region is a measure of the flux of particles from the left to the right regions. This simulation also demonstrates how long it takes for the two regions to come to equilibrium.

develops an intuition for the way in which the random motions of particles lead to their diffusive spread. Furthermore, this demonstration sets the stage for a precise derivation of Fick's first law (which relates particle flux to concentration gradient) starting from a random-walk model of diffusion (Weiss, 1996a, Section 3.2.1).

Some models of biological phenomena are difficult to teach not because the mathematics is difficult, but because the mathematics is messy with a large number of parameters. Thus, students can get lost in the notation and in the messiness of the description. A software package that allows manipulation of the model parameters can cut through the messy description and build intuition very quickly to expose the inherent simplicity of the underlying model.

> **Example 8.2** Even the simplest models of carrier-mediated transport involve a number of parameters and several physical processes. In our experience in teaching this topic, there is a great tendency for students to get lost. The derivations of the relations of flux to concentration are simple, but take some time and do not give a great deal of insight into the workings of this class of models. We have found that time spent in lecture using the carrier-mediated transport package gives students an immediate intuition into the workings of these models. Students are then motivated to follow the derivations and to appreciate the precise description that results.

Simulations are especially helpful to motivate students to grapple with concepts that may be difficult and to foster a sense of intellectual engagement. Properly used in lecture, the simulations promote an active participation in learning rather than the more passive participation that comes from simply listening to a lecture (Mazur, 1997). Asking students question in lecture gets them to participate intellectually.

> **Example 8.3** The simplest diffusion processes result when both particle flux and concentration are time independent. To motivate students to think about such processes it is helpful to ask them to predict the steady-state spatial distribution of particles when the flux is a constant in space and time. Is the distribution constant? Is it exponential? Is it Gaussian? After the students have made their guesses — and perhaps after a tally of all the guesses has been made — the simulation of the random-walk model can be begun with a constant source of particles at one end of a rectangular field and a sink at the other. The derivation of the steady-state distribution can proceed on the blackboard while the simulation is in progress. The parameters of the simulation can be chosen so that the spatial distribution of particles approaches its steady-state distribution in about the time it
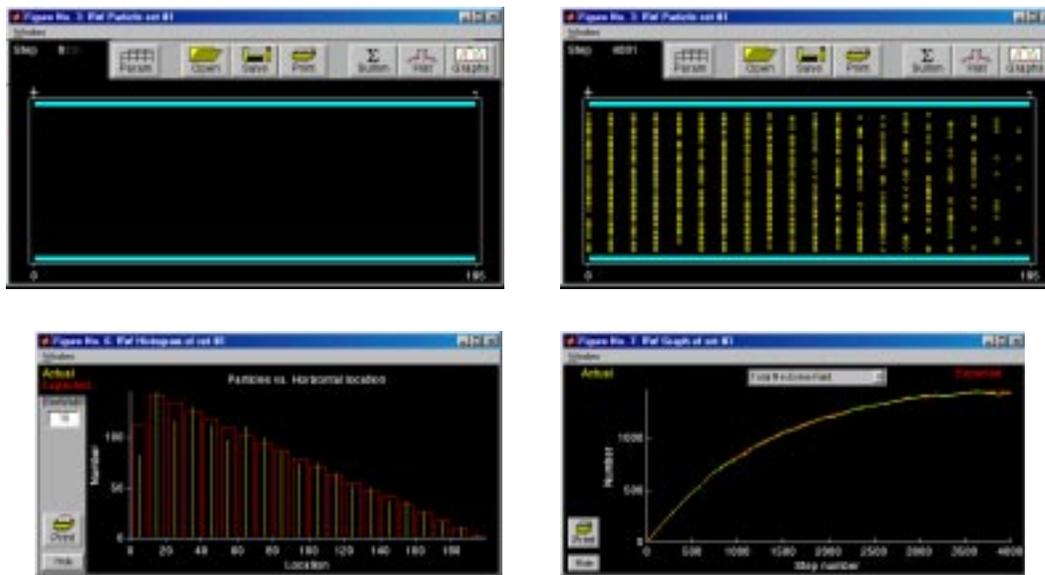
Figure 8.2: Steady-state distribution for random walks in a single compartment. The compartment was initially empty as shown in the upper left panel and the step size was set to 5. There was a source at 0 and a sink at 195. One particle per step was generated by the source. After 4001 steps, the particles were distributed as shown in the upper right panel with the histogram shown in the lower left panel. Except right near the source, the spatial distribution is linear. The number of particles is plotted versus step number in the lower right panel. The results show that the number of particles is approaching a constant for this number of steps indicating that the distribution is near its steady-state distribution.

takes to complete the derivation. From time to time during the derivation, the lecturer can check the progress of the simulation by examining the dependence of the number of particles in the field on time. This number approaches an asymptotic steady-state value. Hence, this plot gives a quantitative estimate of whether the spatial distribution has reached its steady state (Figure 8.2). When the derivation is complete and the simulation is complete the two can be compared. This usage of the computer is fun and motivates students to follow the derivation which solves this problem rigorously.

Example 8.4 Diffusion between two finite compartments through a separating membrane is an important transport process in biological systems. It can model, for example, diffusion between the cytoplasm of a cell and the extracellular fluid, across the plasma membrane. There are two important time scales for two-compartment diffusion through a membrane: the time for the concentration in the membrane to reach its steady-state distribution and the time for equilibration of the two compartments. For thin membranes, the spatial distribution

of concentration in the membrane reaches its steady-state distribution much more rapidly than the time it takes the two compartments to equilibrate. Thus, it is adequate to relate flux and concentration by a steady-state relation even though concentrations in the two compartments are changing in time; a point that can be somewhat tricky for students to understand. This conceptual difficulty is removed if the solution to the two-compartment diffusion problem can be displayed for a membrane of arbitrary thickness. Simulation experiments make these two time scales clear (Figure 8.3). Furthermore, there is a general lesson to be learned about the nature of approximations to be made for a system with many kinetic processes, but where one process is rate-limiting.

Example 8.5 A defining characteristic of electrically excitable cells is their sharp threshold for excitation. Sub-threshold current stimuli produce no action potential; super-threshold currents do. The difference in current level between sub- and super-threshold stimuli can be very small. The Hodgkin-Huxley model possesses such a sharp threshold which can be demonstrated (Figure 8.4) by showing that a small increase in a subthreshold stimulus current can result in an action potential. The mathematical basis for the threshold is subtle. We have found that demonstrating the sharp threshold in the Hodgkin-Huxley model in a dramatic fashion will motivate students to examine the mathematical basis of sharp thresholds in such differential equations (Weiss, 1996b, Section 4.4.1). By preparing in advance, the threshold can be bracketed so that an increase of one unit in the least significant digit changes a sub-threshold stimulus to a supra-threshold stimulus. This raises the student's curiosity about the nature of a threshold in a differential equation and motivates them to follow the lecture devoted to this property.

Another use of computers in lecture is to perform "What if" experiments as a way to build intuition. A typical and obvious use is to ask the class to predict what will happen if some system under study is perturbed in some way. This type of "audience participation" breaks the formality of the lecture, puts students at ease, and engages their intellect. A simulation can be run and the outcome of the experiment can be obtained instantly. There is then an opportunity to discuss the relation between the outcome of the simulation experiment and the students' expectations. Changing almost any parameter of any of the simulations can lead to valuable discussions and deeper insights.

Example 8.6 Suppose we have two solutes that can bind to a transporter in a chemically-mediated transport mechanism. What if ... we change the dissociation constant for the second solute on the inside
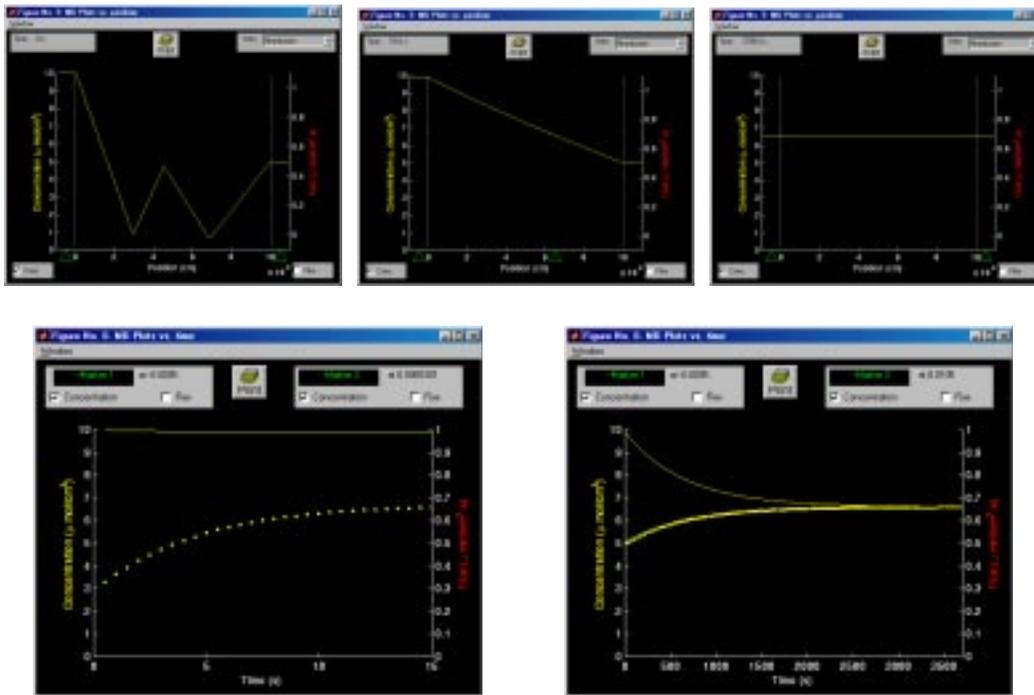
Figure 8.3: Time constants in two-compartment diffusion. The left bath initial concentration was 10 μmol/cm^3 with a width of 0.5 cm; the right bath initial concentration was 5 μmol/cm^3 with a width of 1 cm; the membrane width was 0.01 cm and the initial concentration was drawn as shown in the upper left plot. The upper three plots show the spatial distribution of concentration (membrane view with a bit of each bath on each side) at three times: 0, which shows the initial distribution; 15 seconds, which shows the distribution is linear in the membrane but the two baths are not equilibrated; 3705 seconds, which shows the baths and membrane near equilibration. The lower two plots show the concentration as a function of time. The left plot vs. time shows the concentration as a function of time in the left bath and for a point in the membrane. The concentration in the membrane has not changed appreciably over a time span for which the concentration in the membrane has nearly reached steady-state. The right plot shows the concentration as a function of time in the two baths over a time scale that shows the concentration equilibrating.
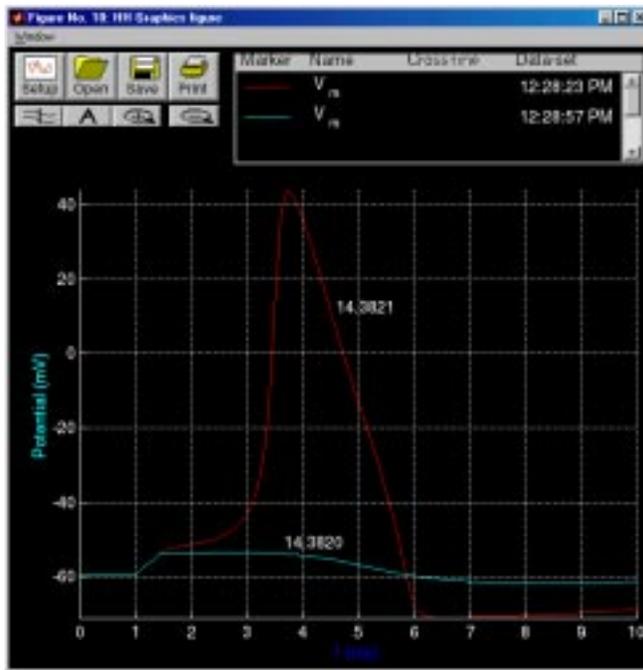
Figure 8.4: The threshold for eliciting an action potential in the Hodgkin-Huxley model. The stimulus is a current pulse of 1 ms duration; the sub-threshold amplitude is 14.3820 and the suprathreshold amplitude is 14.3821 $\mu A/cm^2$.

of the membrane? How will the fluxes of the first and second solutes change? How will the fraction of transporters bound to the first and second solutes on the inside of the membrane change? How will the fraction of transporters bound to the first and second solutes on the outside of the membrane change? Such questions are answered quickly by use of the chemically-mediated transport package and the answers invite further discussion.

As will be described in further detail below, lectures can also be used to perform a research project live with audience participation. Thus, the lecturer and the students are working on the project together. This lends a quite different atmosphere to lectures.

The uses of computation described above have become integral, and now *essential*, parts of lectures and are performed live in the lecture room. It has proved both an effective and an enjoyable part of teaching and learning.

8.2.2 Electronic classroom and homework problems

We hold several classes in an "electronic classroom" in which each student has access to a computer workstation and is free to work either from prepared home-work problems or in a self-directed manner. We have found that most students prefer to work on the homework problems and only a few will work independently. Some students finish that portion of the homework assignment in the session in the electronic classroom. Others start on the problems and finish them at a later date.

Students and staff members work together in this setting. The objective is for everyone to arrive at the solution to the problems, each of which is chosen to illustrate some important issue in the subject matter. This mode of work is noncompetitive, cooperative, and self-paced. It also provides a good opportunity for students and staff to interact in a more informal setting than is possible in a lecture or a recitation.

Some of the problems we have frequently assigned for homework and used successfully in an electronic classroom are described below.

Example 8.7 In connection with two-compartment diffusion, we use Problem 3.4 to make sure students understand the difference between the time constant to reach steady state in a membrane and the time constant to reach equilibrium between the two baths that are separated by the membrane (see Example 8.4).

Example 8.8 We often assign Problem 4.1 to give students some feeling for the relation between flux and concentrations for the simple, symmetric, four-state carrier model. This is the simplest of the models for carrier-mediated transport through membranes and it is a stepping stone for understanding more complex models.

Example 8.9 The Hodgkin-Huxley model is difficult for undergraduate students to comprehend. Therefore, we start students thinking about the model in the voltage-clamp configuration by assigning Problem 5.1. Part a is straightforward but part b requires that students perform simulations iteratively to arrive at a solution. They learn about the interaction of activation and inactivation to yield the sodium conductance in a voltage-clamp configuration.

Example 8.10 We usually assign one part of Problem 5.11 to get students thinking about the Hodgkin-Huxley model in the current-clamp configuration. The assignment is in the form of a puzzle. Students enjoy the challenge of figuring out the puzzle. With a set of default model parameters and stimulus current parameters, the Hodgkin-Huxley model gives rise to an action potential. A change in one of many different parameters of this model leads to no action potential in response to the same current stimulus. Students are given problems in each of which one parameter (e.g., temperature, membrane capacitance, an ion conductance, and ion concentration) has been changed to block the occurrence of the action potential. Students are asked to find a compensatory change in parameters that will restore the action potential to be identical (within certain bounds) to that obtained with the default parameters. This type of problem forces students to examine the interactions of internal model variables in a critical manner.

Example 8.11 Understanding the kinetic behavior of single channels is challenging for students. We usually assign Problem 7.1, which deals with a two-state channel that is not voltage gated, and Problem 7.2, which deals with a two-state channel that is voltage gated. Both of these problems ask students to design a channel that opens at a particular value of the membrane potential and has prescribed kinetic properties. Students learn about the differences between voltage-gated and ungated channels. Also, these problems exercise the students' intellect in a very different manner than do the traditional problem sets that are a routine part of homework assignments. In a traditional homework assignment, students are asked to analyze problems that are completely defined. The software problems can be less well defined so that the definition of the problem is part of the assignment.

8.2.3 Projects

We have found it educationally effective to require that students complete a project of their choice using the Hodgkin-Huxley model software. The exercise affords students an opportunity to formulate a hypothesis about a theoretical topic, to write a proposal, to test the hypothesis, to reach a conclusion, and to communicate their results in writing. Although these projects can be quite time consuming (for both students and faculty), they have been very popular with the vast majority of students. For many of the students, this is the first learning activity that has the flavor of a research project. To emphasize the importance of this educational activity, performance on the project is graded and accounts for 15% of the grade in the subject, a percentage that is comparable to the importance of one of the examinations in the subject.

During one semester when enrollment in the subject was unusually high, it was necessary to pair up students to make the management of projects acceptable.[1] Perhaps not surprisingly, we found that the quality of the work seemed to improve. Thus, we have asked students to work in teams of two ever since although we do not require it.

We also found that as the years went by, reports got longer and longer. Thus, to encourage the students to be concise we placed limits on the lengths of report. We give the following instructions to students about the rules for the project.

"Students may do projects individually or in pairs. If a pair of students collaborate on a project they should submit a single proposal and a single report which identifies both members of the team and gives both email addresses. Proposals will be returned as soon as

---

[1]Interacting with students to help them write acceptable proposals and to conduct these projects takes considerable time. Assessing a single final project report takes over one hour.

possible so that students may revise them. Only the final, accepted proposal will be given a grade.

The demonstration project performed in lecture cannot be the basis of a student project. The report should be at most 10 pages of text and at most 10 figures and tables."

We describe the steps involved in the project in detail.

The proposal

Students are required to write a proposal, which must be approved before they start the project. Sometimes students write as many as 4 proposals before an adequate project is defined. Interestingly, the impetus for rewriting proposals comes as often from the student wanting to *get it right* as it does from the instructor. The main task for the instructor is to help students to define a project that is challenging but still feasible taking into account the student's interests and knowledge. The project involves formulating and testing a hypothesis about some property of the Hodgkin-Huxley model, preferably one that has physiological significance. It is desirable, although not necessary, that students read papers in the literature concerning their topic.

Students are given the following instructions:

"The proposal should contain a brief statement of the hypothesis you propose to test, as well as your proposed method of procedure. Include a list of the computations you will perform and the measurements you plan to make. Indicate how the measurements will be used to help you come to a conclusion. The proposal should be written on one $8\,1/2 \times 11$ sheet of paper. A sample proposal is shown in Figure 8.5."

Demonstration project

We have found it helpful to give a one-hour lecture on how to do a research project, by doing a project live in the lecture room. An acceptable proposal for the project is distributed at the beginning of the lecture (such as the one in Figure 8.5). The lecture involves two members of the instructional staff who act as a student pair working on a project. To make it run smoothly, the staff members rehearse briefly. Simulations are performed in front of the class and the two instructors argue about what is to be done next and on how to interpret the results. They worry out loud about their grade on the project, etc. The project is chosen so that results obtained in an initial range of parameters are plausible and simple to interpret, but something unexpected happens when the parameter is outside this range. One project that has been used successfully is based on

PROPOSAL: HODGKIN-HUXLEY MODEL PROJECT

- **Name:** Hy I.Q. Student

- **Email address:** hiqstudent

- **Hypothesis:** The difference in observed action potential shape between the giant axon of the squid and the node of Ranvier of the rat is due to a difference in temperature.

- **Background:** The rate constants that determine the variables m, n, and h in the Hodgkin-Huxley model increase as the temperature is increased. Hence, I would expect that with increased temperature the time course of the action potential would be briefer. The observed action potential of the rat node of Ranvier measured at 37°C. is briefer than the action potential of the squid giant axon measured at 16°C. Hence, I will determine whether this difference in temperature is adequate to explain the difference in action-potential waveform. In this study, I will assume that the Hodgkin-Huxley model accounts for action potentials in both preparations. Since the intracellular and extracellular ionic concentrations of sodium, potassium and calcium also differ in these two preparations, if time permits I will also investigate the effect of this difference.

- **Procedure:** I will obtain the intracellular and extracellular concentrations of sodium and potassium ions for the rat node of Ranvier from the literature. I will perform a series of simulation studies with a single pulse of membrane current whose amplitude will be set at a constant suprathreshold value. I propose the following series of simulations

#	Temp. °C	Ion Concentrations
1	16	default values for squid giant axon
2	37	default values for squid giant axon
3	16	values found for rat node
4	37	values found for rat node

The waveforms of the action potentials obtained under the various conditions will be compared on the graphics display and the summary statistics for each condition will be examined. If condition 4 yields an action potential that resembles that which is measured on the rat node of Ranvier, then the hypothesis will be supported by these calculations. If they differ, then I will investigate the bases of the differences.

Figure 8.5: Sample proposal for a project.

the hypothesis "Increasing the temperature increases the rate of change of the action potential at its onset." Simulations are performed first for a range of temperature that support this hypothesis. Using the blackboard, one member of the staff plots the maximum rate of change of the potential (obtained conveniently by measuring the peak of the capacitance current) for each temperature after each simulation is performed. However, at a sufficiently high temperature, the action potential is blocked. The staff members argue about whether to ignore this effect or to go on and to confront this unexpected finding. They consider the possibility that they have found a bug in the software or that this phenomenon results from problems of numerical integration. By performing further simulation experiments, it is made clear that these explanations are not plausible. The instructors propose various circular arguments of the type students frequently propose. "The action potential is blocked because the sodium conductance is small." Of course, the sodium conductance depends upon the membrane potential. Hence, the sodium conductance is small because the membrane potential is small. Pursuit of this circular argument (on which some students get stuck each year in their own projects) can have the class in hysterics. In any case, the analysis of the basis of thermal block now becomes the focus of the project. Simulation experiments performed for small changes in temperature below the temperature at which the action potential is blocked, reveals the interplay of kinetic processes that result in thermal block. Since these same processes are involved in the excitation of the action potential, the analysis can give great insight into the mechanism of initiation of an action potential (Weiss, 1996b, Section 4.4.8).

The computations

Students are given the following instructions on how to perform the computations.

> "After carefully defining the computations needed to determine the validity of your hypothesis, you should perform the necessary computations. You should keep careful track of the computations you perform by entering relevant information in a notebook. The notebook should indicate the date of each computational session as well as the identity of the computer you used (make and model). Keep an accurate record of each computation so that you can reconstruct what you did at a later date. Save your results in files. Make composite plots to summarize a series of measurements. For example, to show the effect on the action potential of some parameter, plot the action potentials for selected values of the parameter on one set of axes. Arrange you work so that you have time for some final computations after you begin writing the report. You may find that new questions will arise as you write the report. If you leave adequate

time, you may be able to resolve these questions."

The report

Many undergraduate students have never read a scientific paper and do not know how to write a project report. Hence, we devised the following description of the report.

> "The Project Report is an opportunity for you to work on your ability to express yourself concisely. Do not repeat material that is easily referenced. For example, there is no need to reproduce any figure from readily-available texts: simply refer to it. Technical writing is necessarily directed at some particular intended audience. Write the Project Report as though it were to be published in a journal that is read primarily by students who have taken this subject. Thus, you may assume that your audience has some working knowledge about the subject matter, but no detailed knowledge of your specific project.

> The Project Report should contain the following sections.

> **Cover Page.** On the cover page include the title of the project and the author's/authors' names, and the date.

> **Abstract.** The abstract is a one paragraph (< 100 words) summary of the report including the hypothesis investigated, the methods used, and the principal results and conclusions. This section should be written last.

> **Introduction.** This is a brief section (about 1 page) designed to motivate the reader to read your report. Include background information on the problem, hypotheses to be tested, significance of the work, etc. You may give citations to texts where appropriate. The introduction should be directly relevant to your report and should not discuss neurophysiology or the brain in general terms.

> **Methods.** Explain any methodological or procedural issues that are not obvious. For example, if you measured the time of the peak of the action potential as a function of some parameter, explain how you measured the time of the peak and how you defined when an action potential occurred, i.e., how did you separate the action potential from the local response.

> **Results.** Describe your computational results (whether or not they fit with expectations) in the results section. Generally, results can be communicated more efficiently and accurately with pictures, graphs, and tables than with words alone (e.g., see Figures 5.16, 5.18, 5.19, and 8.4). However, a collection of printed graphs

without a written description of their relevance is unacceptable as a Results section. Students frequently err on the side of including a large number of graphs and little description of their relevance. The Results section should be organized to lead the reader through a logical sequence of results that terminates inexorably in a conclusion. Your objective is to construct an airtight case for some conclusion whether or not it supports or rejects your original hypothesis. This is not the place to interpret or to discuss the results.

Discussion and conclusions. State your conclusions concisely and point out how your results support your conclusions. You might also suggest additional computations to further test your hypothesis. Your conclusions and discussion should be focussed on the relation between your results and your hypothesis. The discussion section can include relevant speculations.

Appendix. The appendix should include a copy of the protocol taken during the computational sessions."

Grade for the project

Grading of the project reports has evolved since its inception in 1984. After using these project for a few years, we noted that many, but still less than a majority, of the projects focussed on a relatively small number of topics. The quality of these reports also improved markedly with time. We felt that the graded reports from previous years were available through student archives. To encourage students to choose new topics and to reward originality we added an assessment for originality. In addition, we found that a hard deadline for accepting reports was hard to enforce so we made the deadline somewhat soft assessing a numerical penalty for lateness by a formula that the students knew ahead of time. The following grading scheme has emerged by trial and error and is satisfactory.

"The grade will not depend on whether the project was an individual or a team project. Both members of the team will get the same grade on the project. The grade for the computer project will be based on the proposal and on the report using the following criteria:

Proposal (10%) The proposal should be concise and clearly define the project in the form of a hypothesis. It should include a brief description of the methods to be employed. A late proposal will get a grade of 0.

Report Structure (15%) All the sections of the report should be present and each should contain the relevant information.

Clarity/Conciseness (20%) A good report is easy to read. The content of each paragraph and each graph should be clear. Everything included in the report should be there for a reason. Points will be deducted for extraneous material such as a large number of graphs without explanation in the text.

Conceptual Correctness (30%) Correctness will be judged by the answers to several questions. Are there clear conceptual errors? Are the results confused? Are the results (i.e. which follow directly from the simulations) confused with the interpretations (which rely on information other than the simulations)?

Originality (25%) This grade will assess the degree of difficulty, novelty, and imagination of the project. The simpler projects, such as many of those suggested in Section 5.6 and used by many students in the past, will receive ≤ 10%, more intellectually challenging, novel projects will receive 25%.

Reports that are not turned in on time will be penalized by the following formula — the grade of the report will be multiplied by a lateness factor L, where $L = 0.3e^{-t/4} + 0.7e^{-t/72}$ and t is the number of hours late."

Electronic interactions

Students are invited to interact with the teaching staff via electronic mail. This has been an increasingly effective means of teaching and learning. For example, some students prefer to submit proposals for projects electronically. The staff responds to these electronically, sometimes within minutes but certainly by the next morning. The rapid turnaround time between submission, critique, and resubmission can make writing an acceptable proposal, which some students find challenging, more efficient. Students use electronic messages to seek help on independent projects as they are conducting them. They send queries electronically and receive responses electronically. This type of interaction, in which the individuals involved may be located at opposite ends of the campus, is an effective way to achieve individualized instruction.

8.3   Assessment

8.3.1   Cost of development

We estimate that it has taken about 5 person-years to develop the software. Although harder to estimate, it has taken perhaps an additional person year of faculty time to design software packages, thoroughly test each software package,

design problems and projects, write documentation, and to investigate pedagogical methods.

8.3.2   Extent of software usage

Our use of the software in teaching has increased over the years. Our current usage is summarized as follows: There are 37 lectures (classes attended by all the students simultaneously) in the subject and the software is used in about 8-10 lectures. There are 24 recitations (classes attended by students in groups of about 25) and the software has not been used in any of these; in part due to the lack of availability of appropriate classrooms with network connections in the past. This has now been rectified and we expect in the future to use the software routinely in recitation classes. Several recitations devoted to using the software are held in electronic classrooms. Typically, there are 13 problem sets assigned during a semester; five or six problem sets include parts that involve the software. There is one laboratory session, which requires a formal laboratory report but does not involve the software. There is one project that involves the software which requires a formal report. There are about 9 short quizzes and two long examinations; none use the software.

This assessment shows that use of the software represents a small but appreciable fraction of the teaching and learning in this subject. Since we have found that use of the software is labor intensive both for the staff and the students, we have been reluctant to increase the use of the software.

8.3.3   Informal assessment of impact on learning

We have not conducted a careful experiment with controlled groups to test objectively the effectiveness of the software in promoting students' comprehension of the material. Subjectively, it was clear at the outset that the students enjoyed using the software and that the staff felt that use of the software was pedagogically effective. Thus, it did not seem fair to exclude some students from the software for the sake of conducting a *scientific* experiment. Surveys of student opinions on the efficacy of the software have been predominantly very positive with opinions ranging from those who thought we should develop software packages for every topic in the subject to a few who found the software confusing. Each year, several students make valuable suggestions on software improvements or on methods for software use. These student inputs have been important for the evolution of the software. This subject matter was taught for 20 years before the software became available and for 15 years after it became available. The staff has little doubt that students learn the material more efficiently and to a greater depth with the use of the software than they had before the software became available.

There are other signs of student interest in the software. Students clearly enjoy working on the software, especially on their own projects, despite the great effort that is required. The impressions of the staff are that while most students get some benefit from using the software, the very best students have an experience that most resembles a small research project. This can be seen in the reports students write to complete their projects. Some of these reports resemble undergraduate theses in scope and in content. Another indication of student interest is that several students have written software on their personal computers to emulate some of the software we have provided. This was done just for their own interest. There is a final and most important indication of student interest. As a result of using the software in this subject, students have been interested to work on the development of educational software. Since 1984, nine undergraduate and three graduate students have been involved in this effort. They have written all the software packages.

8.3.4   Surveys and focus groups

Since 1984, surveys of students who have used the software have been conducted routinely. Sometimes these surveys were done at the end of the semester and sometimes immediately after students completed an assignment that involved the software. Surveys were done by having students fill out forms as well as via email. In recent years, an independent evaluator conducted the surveys and ran focus groups. The feedback we have received has been valuable and has helped us focus on problems students were having with the software. In general, assessments of student have been very positive. Most students find the software pedagogically effective and some find the software critical to understanding complex topics such as the Hodgkin-Huxley model. Students felt they received the most educational benefit from those software packages with which they spent the most time. In particular, the projects were universally praised as educationally effective.

8.3.5   Impact on teaching

It has often been assumed that one *benefit* of the use of computers in teaching is to conserve faculty time, and that a *cost* was isolation of students from each other and from the faculty. In our experience, neither the presumed *benefit* nor the *cost* have materialized. Our use of computers in teaching has increased contact between students and faculty and among students. The main benefit has been more effective teaching and learning. The main cost has been an increased work load for both students and faculty.

8.4 Conclusions

We have used computers to teach biophysics and physiology since 1984. Our use has been exclusively to develop software that simulates some physical, chemical, and biological process that is reasonably well understood. We have invented no new models, but we have explored extant models that underlie our understanding of transport and electrical processes in cells. In teaching this material, we have learned some lessons — both pedagogical and practical — which are summarized below.

8.4.1 Pedagogical matters

Development of educational software must be driven by pedagogy

We have evolved a simple strategy for educational software development. We identify important topics in biophysics and physiology that pose some conceptual difficulties for students and examine whether comprehension could be improved by the use of computational methods. Once we identify such topics, we design and then implement the software.

Software complements other pedagogical vehicles

Software cannot replace a good lecturer or an incisive derivation in a lucid textbook. A simulated experiment cannot replace a *real* laboratory experience. However, software represents an additional pedagogic vehicle that can, if appropriately used, complement other vehicles.

Students should have access to the software

While lecture demonstrations can be pedagogically effective, the potential for learning is greatly enhanced when students use the software themselves. The educational benefits of student usage of the software as opposed to exposure to it in lecture are similar to the benefits derived when students solve problems in homework as opposed to watching a problem solved in class.

The software must be integrated into the subject

The software should be an integral part of lectures, recitations and homework and used only where it enhances learning and never used simply because it is available and because the staff has invested a great effort in its production. Software homework and projects must be assigned and graded with the grade contributing to the student's grade in the subject. If the grades on software assignments do not count toward the student's final grade in the subject, the student

will correctly conclude that the software is secondary in importance to those assignments that do count toward the final grade.

The software should have layers of complexity

Students approach the software with different levels of sophistication in their understanding of the subject matter. It is essential to devise the software so that all students can receive some educational benefit. This requires that the software be simple to use at the outset but contain sufficient depth to challenge the more sophisticated students and even the staff.

8.4.2   Practical matters

Even slightly flawed software should be avoided

The software must aid students to learn the material and should not impede that objective. Software with flaws, even minor flaws, is a distraction. Students will remember even a minor flaw in the software and will forget the point of the demonstration. Undergraduate students, in particular, will lose confidence in an assignment that involves software with even inconsequential flaws. Since they are quite busy, it is easy to rationalize avoiding a difficult assignment by reporting that the software doesn't work even if the flaws should not distract them from the main point of the assignment. The more mature graduate students seem more tolerant of minor flaws and are willing to endure them to derive educational benefit. But bugs, no matter how minor, should be avoided so that students can concentrate fully on the didactic material.

Expect some complications

A computer in the classroom increases the range of pedagogical tools available to the instructor. But it also complicates the instructor's task. If the instructor is prepared and competent and both a blackboard and chalk are available, not much can go wrong with a lecture. With a computer in the classroom, things can go wrong although the number of episodes has actually been extremely low. It is necessary to be prepared for the possibility of a computer problem so that valuable class time is not lost. The availability of overhead transparencies that cover the relevant material that would have been covered with the computer is a good backup strategy.

Design of the user interface is critical

Most of our time spent in software development is devoted to the design of user interfaces that allow students to focus on the subject material and avoid arcane "computerese". A great deal of the students' time can be conserved if the user

interface is graphic, intuitive, and common to all the software. The user interface cannot be too transparent!

Good documentation is important

We have placed a great deal of emphasis on writing comprehensive documentation for all the software. The documentation gives some background information on the subject matter and contains a detailed user's manual for the software plus a list of suggested problems and projects. Remarkably, some students learn to use the software by reading the manual; others experiment with the software. There are different learning styles for using software. To reach a maximum number of students, a variety of modalities are helpful.

Development of good educational software is time-consuming and iterative

All our successful packages have been rewritten many times in response to staff and student input. There is no shortcut to the development of good educational software.

Use of educational software increases contact with students

Early advocates of the use of computers in teaching suggested that one benefit was a saving in faculty time. Effective use of computers could replace faculty in certain tasks. Some educational theorists worried about the isolation of student from faculty resulting from this use of computers. Our experience with our method of use of computers has given the opposite result. It has greatly increased contact with students to the mutual advantage of students and faculty; the interactions have been very rewarding.

Maintenance of software

Changes in operating systems or in higher level languages on which the software is based will continue to require some maintenance. In addition, student suggestions can help to improve the software if the resources for maintenance are available.

Projection systems

Projection systems for use of computers in the classroom have until quite recently been only marginally adequate. However, the technology is improving rapidly and the cost is decreasing rapidly as well. However, this is not a place to skimp. An inadequate display system, where the students cannot really see what is being depicted, defeats all the effort required to develop the software.

8.4.3   Bottom line

Our experience indicates that the development, usage, and maintenance of a successful educational software package is a difficult and time-consuming process — several iterations have been required to bring our most successful packages to their current level of performance. However, we have also found that computers are a powerful vehicle for teaching. They can be used to: engage the student's intellect, to motivate learning, to allow students to test their understanding of a topic, to make learning self-paced and fun. With the aid of software, students take intellectual ownership of a topic rather than experiencing it vicariously. Once effective software has been used in teaching and learning, it becomes an indispensable tool.

Bibliography

Aidley, D. J. (1989). *The Physiology of Excitable Cells.* Cambridge University Press, Cambridge, Great Britain.

Armstrong, C. M. (1966). Time course of TEA$^+$-induced anomalous rectification in squid giant axons. *J. Gen. Physiol.*, 50:491–503.

Armstrong, C. M. and Binstock, L. (1965). Anomalous rectification in the squid giant axon injected with tetraethylammonium chloride. *J. Gen. Physiol.*, 48:859–872.

Athena (1990a). Computation and Educational Community: A Background Paper. Committee on Academic Computation for the 1990s and Beyond, Massachusetts Institute of Technology.

Athena (1990b). Computing for Education at MIT: Final Report. Committee on Academic Computation for the 1990s and Beyond, Massachusetts Institute of Technology.

Baker, P. F., Hodgkin, A. L., and Shaw, T. I. (1961). Replacement of the protoplasm of a giant nerve fibre with artificial solutions. *Nature*, 190:885–887.

Balestri, D. (1988). Ivory towers, silicon basements. *Infor. Tech. Quaterly*, 7:5–17.

Berkenblit, S. I. (1990). Design of a software diffusion simulator and analysis of a problem in two-compartment diffusion. Master's thesis, Massachusetts Institute of Technology, Cambridge, MA.

Carruthers, A. (1984). Sugar transport in animal cells: The passive hexose transfer system. *Prog. Biophys. Mol. Biol.*, 43:33–69.

Chapman, R. A. (1967). Dependence on temperature of the conduction velocity of the action potential of the squid giant axon. *J. Physiol.*, 213:1143–1144.

Colquhoun, D. and Hawkes, A. G. (1977). Relaxation and fluctuations of membrane currents that flow through drug-operated channels. *Proc. R. Soc. London, Ser. B*, 199:231–262.

Colquhoun, D. and Hawkes, A. G. (1995a). The principles of the stochastic interpretation of ion-channel mechanisms. In Sakmann, B. and Neher, E., editors, *Single-Channel Recording*, pages 397–482. Plenum Press, New York, NY.

Colquhoun, D. and Hawkes, A. G. (1995b). A Q-matrix cookbook. How to write only one program to calculate the single-channel and macroscopic predictions for any kinetic mechanism. In Sakmann, B. and Neher, E., editors, *Single-Channel Recording*, pages 589–633. Plenum Press, New York, NY.

Cooley, J. W. and Dodge, F. A. (1966). Digital computer solutions for excitation and propagation of the nerve impulse. *Biophys. J.*, 6:583–599.

Cox, D. R. and Miller, H. D. (1965). *The Theory of Stochastic Processes*. John Wiley and Sons, New York, NY.

Easton, D. M. and Swenberg, C. E. (1975). Temperature and impulse velocity in giant axon of squid *loligo pealei*. *Am. J. Physiol.*, 229:1249–1253.

Einstein, A. (1906). Sur theorie der brownschen bewegung. *Ann. Physik*, 19:371–381. [For translation see Einstein (1956)].

Einstein, A. (1956). *Investigations on the Theory of the Brownian Movement*. Dover Publications. R. Furthe and A. D. Cowper, eds. [Translation of original publications].

Fick, A. (1855). On liquid diffusion. *Philos. Mag.*, 10:30–39.

Fitzhugh, R. and Antosiewicz, H. A. (1959). Automatic computation of nerve excitation — detailed corrections and additions. *J. Soc. Indust. Appl. Math.*, 7:447–458.

Frankenhaeuser, B. and Hodgkin, A. L. (1957). The action of calcium on the electrical properties of squid axons. *J. Physiol.*, 137:218–244.

George, S. A., Mastronarde, D. N., and Dubin, M. W. (1984). Prior activity influences the velocity of impulses in frog and cat optic nerve fibers. *Brain Res*, 304:121–126.

Gerald, C. F. and Wheatley, P. O. (1989). *Applied Numerical Analysis*. Addison-Wesley, Reading, MA.

Hanselman and Littlefield (1997). *Mastering MATLAB 5: A Comprehensive Tutorial and Reference*. Prentice-Hall, Englewood Hills, NJ.

Hille, B. (1992). *Ionic Channels of Excitable Membranes*. Sinauer Associates Inc., Sunderland, MA.

Hines, M. (1984). Efficient computation of branched nerve equations. *Int. J. Bio-Med. Comput.*, 15:69–76.

Hines, M. and Carnevale, N. T. (1998). Computer modeling mehtods for neurons. In Arbib, M. A., editor, *The Handbook of Brain Theory and Neural Networks*, pages 226–230. MIT Press, Cambridge, MA.

Hodgkin, A. L. (1964). *The Conduction of the Nervous Impulse.* Charles C. Thomas, Springfield, IL.

Hodgkin, A. L. (1977). Chance and design in electrophysiology: An informal account of certain experiments on nerve carried out between 1934 and 1952. In *The Pursuit of Nature*, pages 1–21. Cambridge University Press, Cambridge, England.

Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.*, 117:500–544.

Hodgkin, A. L. and Katz, B. (1949a). The effect of sodium ions on the electrical activity of the giant axon of the squid. *J. Physiol.*, 108:37–77.

Hodgkin, A. L. and Katz, B. (1949b). The effect of temperature on the electrical activity of the giant axon of the squid. *J. Physiol.*, 109:240–249.

Huxley, A. F. (1959). Ion movements during nerve activity. *Ann. N.Y. Acad. Sci.*, 81:221–246.

Huxley, A. F. (1964). Excitation and conduction in nerve: Quantitative analysis. *Sci.*, 145:1154–1159.

Johnston, D. and Wu, S. M. S. (1995). *Foundations of Cellular Neurophysiology.* MIT Press, Cambridge, MA.

Joyner, R. W., Westerfield, M., Moore, J. W., and Stockbridge, N. (1978). A numerical method to model excitable cells. *Biophys. J.*, 22:155–170.

Kandel, E. R., Schwartz, J. H., and Jessell, T. M. (1991). *Principles of Neural Science.* Elsevier, New York, NY.

Katz, B. (1966). *Nerve, Muscle and Synapse.* McGraw-Hill Inc., New York, NY.

Keynes, R. D. and Aidley, D. J. (1991). *Nerve and Muscle.* Cambridge University Press, Cambridge, Great Britain.

Kulik, C. C. and Kulik, J. A. (1986). Effectiveness of computer-based education in colleges. *AEDS J.*, pages 81–108.

Mascagni, M. V. and Sherman, A. S. (1998). Numerical methods and neuronal modeling. In Koch, C. and Segev, I., editors, *Methods in Neuronal Modeling*, pages 569–606. MIT Press, Cambridge, MA.

Mazur, E. (1997). *Peer Instruction: A User's Manual*. Prentice Hall, Upper Saddle River, NJ.

Moore, J. W., Ramon, F., and Joyner, R. W. (1975). Axon voltage-clamp simulations. *Biophys. J.*, pages 11–24.

Neher, E. and Sakmann, B. (1976). Single-channel currents recorded from membrane of denervated frog muscle fibres. *Nature*, 260:799–802.

Nicholls, J. G., Martin, A. R., and Wallace, B. G. (1992). *From Neuron to Brain: A Cellular and Molecular Approach to the Function of the Nervous System*. Sinauer Associates, Sunderland, MA.

Pearlmutter, B. A. and Zador, A. (1999). Sparse matrix methods for modeling single neurons. In Koch, C., editor, *Biophysics of Computation*, pages 487–502. Oxford Univ. Press, New York, NY.

Plonsey, R. and Barr, R. C. (1988). *Bioelectricity, A Quantitative Approach*. Plenum Press, New York, NY.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1986). *Numerical Recipes. The Art of Scientific Computing*. Cambridge University Press, Cambridge, Great Britain.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1988). *Numerical Recipes in C. The Art of Scientific Computing*. Cambridge University Press, Cambridge, Massachusetts.

Shah, D. M. (1990). Computer simulation of the random walk model of diffusion. Bachelor's thesis, Massachusetts Institute of Technology, Cambridge, MA.

Shampine, L. F. and Reichelt, M. W. (1997). The MATLAB ODE suite. *SIAM J. Sci. Comput.*, 18:1–22.

Smith, G. D. (1985). *Numerical Solution of Partial Differential Equations: Finite Difference Methods*. Clarendon Press, Oxford, Great Britain.

Stein, W. D. (1986). *Transport and Diffusion Across Cell Membranes*. Academic Press, New York, NY.

Tasaki, I. and Hagiwara, S. (1957). Demonstration of two stable potential states in the squid giant axon under tetraethylammonium chloride. *J. Gen. Physiol.*, 40:859–885.

Weiss, T. F. (1996a). *Cellular Biophysics. Volume 1: Transport.* MIT Press, Cambridge, MA.

Weiss, T. F. (1996b). *Cellular Biophysics. Volume 2: Electrical Properties.* MIT Press, Cambridge, MA.

Weiss, T. F., Trevisan, G., Doering, E. B., Shah, D. M., Huang, D., and Berkenblit, S. I. (1992). Software for teaching physiology and biophysics. *J. Sci. Ed. Tech.*, 1:259–274.

Wilson, J. M. and Redish, E. F. (1989). Using computers in teaching physics. *Physics Today*, pages 34–41.

Index