

“ NEURON ” による神経機能のシミュレーション

藁科 彬¹⁾ 小倉 立也²⁾

新潟大学大学院医歯学総合研究科 細胞生理¹⁾

Colorado State University, Department of Anatomy and Neurobiology²⁾

1. はじめに

研究や教育の目的で、既知あるいは予想される特性を持つイオンチャネル、ポンプ、交換体等を含むモデルを作り、種々の状況下で細胞、神経機能がどのように制御されるかをシミュレーションで調べてみたいという希望は少なからぬ研究者の抱くところと思われる。しかし、数値計算やその結果を表示するプログラムを作製する膨大な労力を払ってまで実行に移す人は稀であろう。ここに紹介する“ NEURON ”はYale大学のM.L. HinesとN.T. Carnevale、それにDuke大学のJ.W. Mooreが中心となり開発を行っている細胞や神経の電気的、化学的信号の発生・伝達に関するシミュレーションを目的とする公開コンピュータプログラムである。すでに“ NEURON ”を利用した200以上の研究が論文として発表されており、多種のチャネル機構、Ca²⁺の細胞内拡散と細胞外への排出機能、伝達物質の放出機構などが既存のモデル要素として公開されている。利用者はそれらの中から必要なものを選択しモデルを組むことができ、また、自分で作製した機能をそれらに加えて使用することも可能である。そのように、NEURONは一般の利用がかなり容易であるように構築されており、マニュアル類も存在するが、予備知識無しに始めてみると、実際に使用できるようになるまでかなりの時間を要するものであることを著者らは体験を通して知った。そこで、筆者らが操作法をメモ書きしてあったものに多少の手を加えたこの小稿により、これからNEURONの利用を試みられる方が手っ取り速く、その概要や運用法を知って頂ければと考えた次第である。

本稿前半では、NEURONに付設されているデモ機能の説明、基本的操作法の要約、および既存機能によるモデル作成法などを述べる。後半では、新たに新規の機構を加えたり、神経回路を含むモデルの作成を行う場合の手順に関して記す。

なお、本文中の<>で囲った用語は、NEURONプログラム中、マウスのクリックにより選択や実行の行われる事項であることを意味する。

2. NEURONのインストールとデモ機能の実行

NEURONのホームページ[1]のDownload and Installからプログラムファイル書庫にアクセスでき、Mac, Windows, UNIX/Linux用の三種が用意されているので、使用OSに合わせインストールを行う。これらは時折改変されるが、基本的構築は常に維持されるよう配慮されている。以下の説明はWindows用NEURON Ver 4.3.1 (Ver5.0.0試用版も可であるが、旧版を完全にuninstallしないと不調になることもある)に基づくものである。ホームページ上にはその他に、About NEURON (システムの概容)、Bibliography (NEURON自身、およびそれを使用した研究論文)、Demonstrations and Examples (シミュレーション結果の例示)、Documentation (NEURONシステム、計算法、プログラミング等の解説)、Course announcements (使用法講習会の案内)、User's Group (使用者グループへの加入申し込みと使用者間のmailによる情報交換)、等々のリンクが存在する。

インストールを終えると、WindowsのスタートメニューのリストにNEURONという項目が追

加され、そのサブメニューに NEURON Demo という項目が登録される。これをクリックしてプログラムを開始すると、インタープリターとの情報交換用の OC ウィンドー、および、Main Menu, Demonstration menu など、複数のパネルがデスクトップに現れる。OC ウィンドーは起動時に組み込まれた機構を表示した後、oc> というプロンプトを出し停止している。ここからパラメータの値やコマンドをインタープリターに送ることも可能であるが、そのような使い方をしない場合は他のパネルに隠れていたり、タスクバーに退避させておいても差し支えない。しかし、シミュレーション中にエラーの発生したときにはその原因がこのウィンドーに表示されるので、再表示して確認する。

NEURON Demonstrations というタイトルのパネルには4個のデモ用プログラムと <No Model> という選択肢を加えた5項目がリストされている。デモ用プログラムの一つをクリックすると、シミュレーションに必要なパラメータ設定用パネルとグラフ表示パネルが更に加わり、シミュレーションの準備が整う。そこで、RunControl というパネルの <Init & Run> というボタンをクリックすると、パラメータの初期値設定に従い直ちにシミュレーションが開始され、結果がグラフ表示される。プログラムを終了するには Main Menu の <Files> <Quit> による。

シミュレーションの過程で、多くのパネルがデスクトップ上に散乱した状態になった場合は、パネルのドラッグにより必要なパネルの配置やサイズを整える。また、パラメータの設定などを行った後、差し当たり不要なパネルはパネルタイトルの下の <hide> という文字をクリックし、退避させる。復帰させるには、NEURON Main Menu の <window> を開き、チェックが消去されているパネルタイトルに再チェックを入れる。

デモ機能のメニュー中に存在する4項目の内容の概略は以下のようである。

< Patch : H-H > Hodgkin-Huxley (H-H) model [2] のシミュレーションプログラム。I/V Clamp Electrode パネルで電流固定 <IClamp> ,

電圧固定 <VClamp> を選択し、電圧あるいは電流パルスを与える遅延時間、持続時間、強度を設定して実行。

< Stylized > 細胞体に有髄軸索と3本の樹状突起を持つニューロンモデルにおける興奮の発生、伝導。各要素はケーブル特性で結合されている。また、細胞体と軸索には H-H type のコンダクタンスが存在する。3本の樹状突起に存在するシナプスの位置や興奮入力タイミングを Synaptic Parameters パネルで設定して実行する。膜電位変化が樹状突起、細胞体、軸索丘、軸索へと伝導する様子を動的なグラフ表示により観察できる。

< Pyramidal > 大脳皮質錐体細胞の3次元トレースデータを取り込み構築したモデルであり、細胞体と8本の樹状突起、およびそれぞれの樹状突起から伸びる数多くの分枝からなる。その形態は PointProcessManager というパネルウィンドー内で右クリックして現れる <3D Rotate> を選択後、左マスの操作で任意の方向への回転表示ができる。また、形態図の任意の部位をクリックし、刺激部位を指定できる。この刺激部位に Select-PointProcess で電流の注入 <IClamp> , あるいは興奮性シナプス入力 <AalphaSynapse> などの刺激条件を設定しシミュレーションを実行する。これらの刺激により発生する活動電位や電気緊張電位が細胞体でどのような膜電位変化を引き起こすかが観察できる。

< Release > 神経筋接合部位のシナプス前神経終末モデルを、 Na^+ 、 K^+ 、 Ca^{2+} チャネル、 $\text{Na}^+/\text{Ca}^{2+}$ 交換体、 Ca^{2+} ポンプ、および細胞内 Ca^{2+} の拡散・緩衝機構の組み合わせにより構築したもの。これら多彩な機能により、活動電位の発生中、発生後の Ca^{2+} 動態を解析する。また、遊離 Ca^{2+} 濃度の上昇によるアセチルコリンの分泌もモデル化されている。

以上のプログラムを使用したシミュレーション結果のモニター画面上での様子は Web サイト [3] で閲覧できる。

3. シミュレーション条件の設定、変更

次に、初期設定から条件を変えてシミュレーション

ョンを行う方法に関して重要と思われる点を幾つか述べてみたい。

機構の挿入と削除：NEURONでは、使用する可能性ある機構（機能）を全て組み込んでおき、ここから個別モデル作成に必要なものを選択できる。例えば、NEURON Demoには11個の機構が組み込まれている。組み込まれている機構は<Tools> <Distributed Mechanisms> <Managers> <Inserter>の順に進むと、Insert/Removeというパネルが現れそこに表示される。このうち、現在使用されている機構には赤いチェックで印しが付付けられており、<Release>では8個の機構がチェックされている。<Stylized> や <Pyramidal>のように、somaに加えて多くのセクション（axonやdendrite）を持つモデルではセクションごとにInserterが表示され、使用する機構を選択できるようになっている。モデルへの機構の挿入や削除はinserterに表示された機構名をクリックすればよく、極めて便利であるが、既に挿入してある機構を削除すると、しばしばシミュレーションが途中で止まったりするトラブルが発生する。このような場合は、あとで述べるパラメーターの変更により、その機構を実質的に無効にすることが推奨される。たとえば、Ca²⁺チャンネル機構（cachan）の場合であれば、その最大透過係数（pcabar_cachan）をゼロにしてしまうといった具合である。

パラメーターの変更：NEURONではモデルに適合した独自のパラメーター表示パネルを作ること可能である。しかし、特別な設定をしなくても、Main Panelから<Tools> <Distributed Mechanisms> <Viewers> <Name Values>で現れるパネルで、<Parameters>にチェックを入れ、somaの欄を2回クリックすると、パラメーター値を表示する標準パネルが現れるようになっている。そのとき、cachan機構が使用されていれば、pcabar_cachan（cm/s）というパラメーターの欄があるので、これを任意の値に変更することができる。なお、この操作で表示されなかったパラメーターや変数は上記<Viewers>から表示される<Mechanisms（Global）>の方の選択により、機

構別に細分化された表示パネル中を検索する。

グラフ表示の設定と変更：Main Menuの<Graph>から、膜電位、膜電流、状態変数などを縦軸とするグラフ表示用パネルを開設することができる。そこで、例えば<Patch：H-H>で、膜電位と膜電流用のグラフに加えて、m、n、hといったチャンネルゲートの状態を表す変数もグラフ表示したい場合には次のような手順を踏む。<Graph> <state axis>で新たなグラフパネルを開き、このパネル上で右クリックして現れる実行メニューから<Plot what?>を選択、出現するパネルの中で、<soma> <accept>の順にクリックすると変数のリストが中央の欄にリストアップされる。この中よりm_hh、n_hh、h_hhをグラフ表示用に選択すればこれらの変数の時間経過が同一グラフ上に描かれる。グラフの色分けは<Color/brush>による。また、グラフ線上でのクリックで横軸、縦軸の詳細値が表示される。また、この例では必要なかったが、横軸、縦軸の数値設定をグラフの表示に適合するよう変えたい場合には<Set/View>で、表示範囲を入力するか、シミュレーションの実行後、<view=plot>で自動変換を行い、適当なレンジを探す。

結果の保存：シミュレーションの結果をプリンターに出力するか、ファイルとして保存する場合はMain Menuから<Window> <Print & File Window Manager>で開かれるパネルの左側の赤枠（デスクトップ領域を示す）中に存在する青四角（現在開かれている各パネルのエリアを示す）のうち、出力したいものを<Select> <Move>により、右側の赤枠に入れた後、<Print>で印刷を行うか、<Session>から、sesという拡張子を付けたファイル名で保存する。このファイルを<File> <load session>で読み込むと、保存したときの条件で次回の実行が開始できる。また、<Print> <Ascii>とするとグラフデータがテキスト形式でファイル化されるので、データを他のグラフ作成ソフトに持ち込むには都合がよい。

4. 組み込まれた機構によるモデルの 作成とシミュレーションの実行

以下に、NEURON Demonstrations 中の <No model> を使用して、ニューロンモデルを作成しシミュレーションを実行した例を紹介する。上記操作法の確認のため読者諸兄も本稿の説明に沿って試して頂ければ幸いである。まず、上述の要領で *Inserter* を開き、Hodgkin-Huxley による Na^+ 、 K^+ チャネル機構 (*hh*)、細胞内 Ca^{2+} の緩衝と拡散およびポンプによる排出機構 (*cadifpmp*)、 Ca^{2+} チャネル機構 (*cachan*)、 $\text{Na}^+/\text{Ca}^{2+}$ 交換機構 (*nacax*) の4個の機構を組み込む。次に、<Name Values> から開いたパラメーターパネル、および <Mechanisms (Global)> <cadifpmp> から開いたパネルで1図の下の説明欄に書き出したパラメーター値を記入する。更に、<Graph> により4つのグラフ表示パネルを設置する。第1は、<Voltage axis> から膜電位表示用のパネル、第2は、<Current axis> から電流表示用のパネルを用意し、パネル上で右クリックし <Plot what?> により開かれるパラメーターリストから、 Na^+ チャネル電流 (*ina*)、 K^+ チャネル電流 (*ik*)、 Ca^{2+} チャネル電流 (*ica_cachan*) を表示するよう設定する。第3も <Current axis> から、 Ca^{2+} ポンプ電流 (*ica_pump_cadifpmp*)、 $\text{Na}^+/\text{Ca}^{2+}$ 交換による Ca^{2+} 電流 (*ica_nacax*) を表示するパネルを設け、縦軸目盛りを - 0.005 から 0.005 (mA/cm²) に変更する。最後の第4は <State axis> からのグラフに細胞内遊離 Ca^{2+} 濃度 *ca_cadifpmp[0]* を表示するよう選択する。なお、*ca_cadifpmp[1]*、*ca_cadifpmp[2]* も同時表示するよう設定するが、これらはパラメーターリストに存在しないので、リストの上部にあるテキストボックスに直接書き込む。後述するように、NEURON の細胞体は球ではなく、円柱状 (直径 *diam*、長さ *L*) で近似されており、その中が同心状の殻層 (現モデルでは10層) に区分されており、その層間を Ca^{2+} が緩衝を受けながら拡散するようになっている。上記の[0]、[1]、[2]の番号は最外殻をゼロとして数えた殻層の番号である。なお、*cadifpmp* では細

胞内 Ca^{2+} バッファの容量は Ca^{2+} 結合比 (*beta_cadifpmp*) というパラメーターで調節される。この量は本来、遊離 Ca^{2+} 濃度およびバッファと Ca^{2+} 間の解離定数の関数であるが [4]、バッファの Ca^{2+} 結合が飽和しないとの仮定を置き、ここでは定数にしてある。

上記でモデル細胞の設定は終了したので、刺激の設定を行う。Main Manu の <Tools> <Point processes> <Managers> <Electrode> で開かれたパネルで <IClamp> とし、*delay*=5 ms、*duration*=0.1 ms、*amplitude*=2nA とする。次に、<Tools> から RunControl パネルを開き、開始膜電位 (*Init (mV)*) を - 65 mV、実行の全時間 (*Tstop*) を 20 ms とし、<Init & Run> よりシミュレーションを開始する。パラメーター類が設定どおりであれば、1図 (*a-d*) のような結果が得られるはずである (図は ASCII 保存したデータを、グラフソフトで再現したもので、モニター上の表示とは多少異なる)。活動電位の発生 (*a*) と各種チャネルによる電流 (*b*)、 Ca^{2+} ポンプや交換体による Ca^{2+} 電流 (*c*)、それらによる細胞内遊離 Ca^{2+} 濃度の上昇と下降 (*d*) が示されている。なお、ここで使用されている $\text{Na}^+/\text{Ca}^{2+}$ は4個の Na^+ と1個の Ca^{2+} が交換する仕組みになっており、 Na^+ 、 Ca^{2+} の平衡電位を *ena*、*eca* とし、膜電位が $2\text{ena}-\text{eca}$ を超えて上昇すると、逆転モードで作動する。このため、活動電位が約 - 30mV 以上になる区間で細胞内への Ca^{2+} の取り込みが見られる (1図 (*c*))。また、ポンプや $\text{Na}^+/\text{Ca}^{2+}$ 交換体により静止状態でも Ca^{2+} の汲み出しがあるが、それでも 50 nM 程度の静止 Ca^{2+} 濃度が保たれるのは、静止膜電位においても僅かな Ca^{2+} チャネルのコンダクタンスが存在するためである。この辺は実際の細胞のシミュレーションを行う場合には、微妙な問題となりそうである。また、15 ms の時間経過では、 Ca^{2+} の緩衝と拡散時間のため内部の Ca^{2+} 濃度上昇は遅れ、ほとんど最外殻の Ca^{2+} 濃度のみが顕著に上昇していることも分かる。

NEURON Demo の <No model> と同様のことは、NEURON をベースに A. Houweling と T. Se-

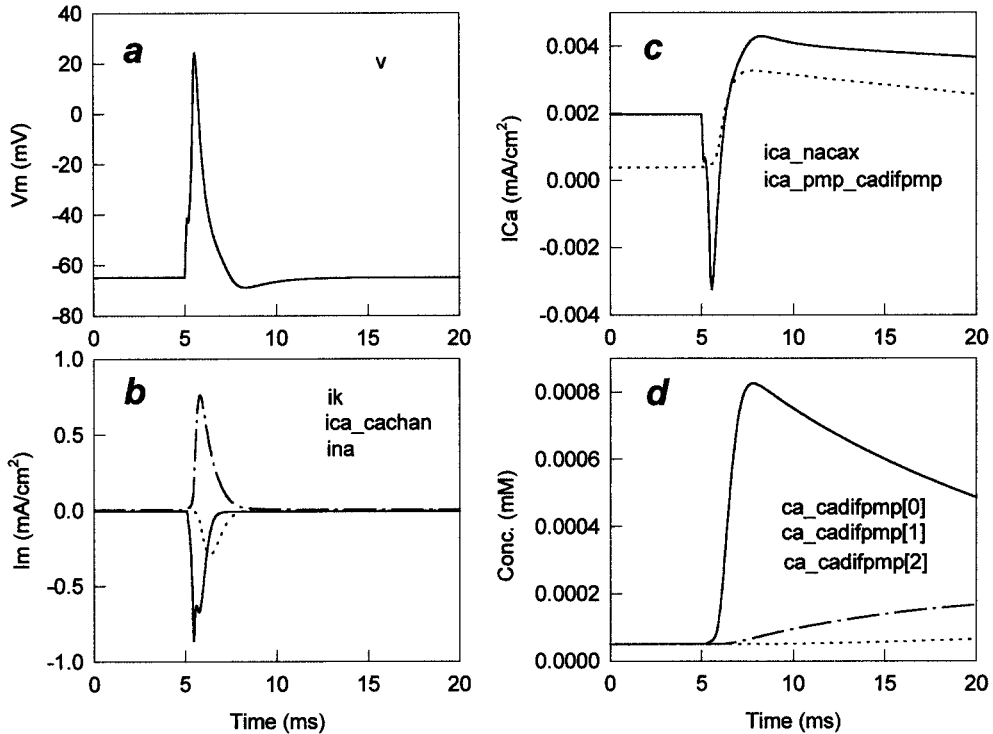


図1. NEURON デモプログラム中に存在する機構を使用したシミュレーションの一例。

(a) 活動電位波形；(b) Na^+ 電流 (実線), K^+ 電流 (鎖線), Ca^{2+} 電流 (点線)；(c) $\text{Na}^+/\text{Ca}^{2+}$ 交換による Ca^{2+} 電流 (実線), Ca^{2+} ポンプによる電流 (点線)；(d) 最外殻層 (実線) とそれに続く二層 (鎖線と点線) の遊離 Ca^{2+} 濃度変化。シミュレーションはパラメーターの以下の値で行われた。温度 celsius (deg C) = 16.3；細胞の直径 diam (um) = 15；長さ L (um) = 15； Na^+ の最大コンダクタンス gnabar (mho/cm2) = 0.12； K^+ の最大コンダクタンス gkbar_hh (mho/cm2) = 0.036；リークコンダクタンス gl_hh (mho/cm2) = 0.001；リーク電流の平衡電位 el_hh (mV) = -65； K^+ の平衡電位 ek (mV) = -77； Na^+ の平衡電位 ena (mV) = 50； Ca^{2+} チャンルの最大透過係数 pcabar_cachan (cm/s) = 0.0004； $\text{Na}^+/\text{Ca}^{2+}$ 交換体のコンダクタンス k_nacax (mho/cm2) = $2e-05$ ；細胞内 Ca^{2+} の拡散定数 DFree_cadifpmp (um2/ms) = 0.6；Ca 結合比 beta_cadifpmp = 50； Ca^{2+} ポンプの細胞内外への Ca^{2+} 輸送の速度定数, k1_cadifpmp (/mM-s) = $5e+08$, k2_cadifpmp (/s) = $2.5e+05$, k3_cadifpmp (/s) = 500, k4_cadifpmp (/mM-s) = 5； Ca^{2+} ポンプ密度 poump0_cadifpmp (mol/cm2) = $3e-14$ 。なお, Ca^{2+} ポンプは $\text{ca}[0] + \text{pump} \rightarrow \text{pumpca} \rightarrow \text{pump} + \text{cao}$ というスキームで作動する (cao は細胞外 Ca^{2+} 濃度)。初めのステップの前進方向の速度定数が k1, 逆方向のそれが k2, 第2ステップの対応する速度定数がそれぞれ k3, k4 である。

jnowski が作製し、公開している MyFirstNEURON [5] でも行える。こちらは 16 の機能要素が組み込まれており、2 個の細胞間のシナプス伝達も取り扱えるように工夫されている。更に、Electrophysiology of the Neuron (J. Huguenard & D. McCormick, Oxford University Press) 中の記載と関連した simulation 問題が含まれており、教育目的にも活用できるよう工夫されている。

5. NEURON の構造

NEURON のデモ機能の使用法、および既存機能の組み合わせによるシミュレーション方法に関して上述した。次に、研究目的などで新たなモデル作成する手順につき記すが、その前に、NEURON システムにつき、もう少し詳しく述べておきたい。既にみてきたように、NEURON では機能別にモデルを作成し、それらを集めて、統合的な環境下でシミュレーションを実施できるよ

うになっている。各機能単位は mod という拡張子を付けたファイルに model description language (NMODL) という様式で記述される。これらの mod ファイルを NEURON に付随するコンパイラに掛け C 言語化したのち、dll (dynamic link library) ファイル形式に変換する。シミュレーションはこの dll ファイルが hoc と呼ばれるインタープリターに呼び込まれて実行されるが、mod ファイルには各機能単位の計算方法が記されているのみなので、統合環境でのシミュレーションの実行に当たっては、細胞の形態、サイズ、実際に使用する機構の選択、mod ファイルで未指定になっているパラメーターの値、刺激条件、グラフやパラメーターの表示用パネルの設定など、各種の情報は別途補う必要がある。これらの情報は、OC ウィンドーや NEURON のモニター画面上から逐一入力することも可能であるが、hoc という拡張子を持つテキストファイルに必要な事項を書き込んでおき、NEURON 起動時に一括してインタープリターへ読み込ませるのが便利である。なお、一つの hoc ファイルの中から別の hoc ファイルを呼び込むことができるので、hoc ファイルについても事項を整理して、小分けに記載することが可能である。通常の NEURON 運用では、起動用の hoc ファイルをクリックすると、自動的に NEURON システム立ち上がり、mod ファイルをコンパイルした dll ファイルがリンクされ、同時に読み込まれる hoc ファイルの情報に基づきシミュレーションの準備が行われる。

6. 個別機能の mod file への記載方法

研究目的では公開されている機構の改変や、新たな機構の製作が必要となる。そこで、既存の mod ファイルに多少の改変を行った例につき説明してみたい。実際には、細胞内の Ca^{2+} 拡散と固定バッファによる Ca^{2+} の緩衝を扱った cadif.mod が nrn/Examples/nrniv/nmodl のフォルダー中に存在するので、これに Ca^{2+} ポンプ機能と、蛍光 Ca^{2+} 指示色素 (可動 Ca^{2+} バッファとしても機能) を付加し、caresp.mod という下記のようなテキストファイルを作成する。紙面の節

約のため、一行にスペース区切りで多くの事項を書き込んだため、読み取り難くなっているの、通常の書式に戻す際には cadif.mod の書式を参照して頂きたい。しかし、説明のために付けた行番号を取り除けば本稿のままの形式でも、後で述べるコンパイル作業に支障はない。

リスト中、斜体字の部分は NMODL 文法で使用が決められた用語である。これらの項目とそれに属する記載のブロックをリスト内に並べる順番は任意であるが、普通には、まず *NEURON* という項目の元に、機構の略称 (*SUFFIX*) やこの機構に関するイオン濃度や電流などの宣言を行ない、関係する単位を *UNITS* の項に列挙し、次に、パラメーターや変数をリストアップする。一つのシミュレーションの間は一定に保たれる変数は *PARAMETER* として、シミュレーションの過程で計算され、それ以後の演算に使用される変数は *ASSIGNED* の項に分類するという目安はあるが、NEURON では変数を *PARAMETER* に登録することが許されるので、両者を厳密に振り分ける基準は無く、実行の際、どの表示パネルに登場させるのが適切であるかによって判断される場合もある。*STATE* 変数として宣言されるものはシミュレーション過程が NEURON 標準グラフ表示パネルを使い、直接グラフ化できる変数である。また *LOCAL* では、この機構内の計算のみに使用するローカル変数を宣言する。*INITIAL* では以後の計算式中で用いる数値ファクターを、*PROCEDURE factors* () に記された内容に沿って計算する。

プログラムの核心は 32 行の *BREAKPOINT* で、states 内に記述された数式を sparse という数値計算法により解けという部位にある。ここで、states は 35 行目より始まる *KINETIC* ブロックを指定するラベル名であり、sparse は backward Euler 法に基づく数値計算が実行されることを意味する。ところで、*KINETIC* ブロックの *COMPARTMENT* (36 行) に関してはやや説明が必要である。NEURON では計算の単純化のため、細胞体は球ではなく、それと同じ直径を持つ円柱

==== caresp.mod の構造 ====

```
1. NEURON { SUFFIX caresp USEION ca READ cai, ica WRITE cai, ica
2. GLOBAL vol, TotalBuffer, TotalDye }
3. DEFINE NANN 4
4. UNITS { (molar)=(1/liter) (mM)=(millimolar) (uM)=(micro/liter)
5. (um)=(micron) (mA)=(milliamp) FARADAY=96480(coul)
6. PI=(pi)(1) }
7. PARAMETER { DCa=0.6(um2/ms) TotalBuffer=0.05(mM)
8. k1buf=100(/mM-ms) CKd=0.001(mM) k2buf=0.1(/ms)
9. Vmax=0.002(mA/cm2) Kca_pmp=0.00083(mM)
10. cai_rest=0.0001 (mM) TotalDye=0.1(mM) Dydif = 0.1(um2/ms)
11. k1dye=500(/mM-ms) DKd=0.00024(mM) k2dye=0.12(/ms) }
12. ASSIGNED { diam(um) ica(mA/cm2) cai(mM) vol[NANN] Ka(/mM)
13. Bf0(mM) ica_pmp(mA/cm2) icaleak (mA/cm2) D0(mM) CaDch(mM) }
14. STATE { ca[NANN](mM) <1e-10> CaBuffer[NANN](mM)
15. Buffer[NANN](mM) CaDye[NANN](mM) Dye[NANN](mM) FDca(1) }
16. LOCAL factors_done
17. INITIAL { if (factors_done == 0) {factors_done=1 factors()}
18. k2buf=k1buf*CKd Bf0=TotalBuffer/(1 + cai_rest/CKd)
19. D0=TotalDye/(1 + cai_rest/DKd) k2dye=k1dye*DKd
20. FDca=1 - D0/TotalDye
21. FROM i=0 TO NANN-1 { ca[i]=cai_rest Buffer[i]=Bf0
22. CaBuffer[i]=TotalBuffer - Bf0 Dye[i]=D0
23. CaDye[i]=TotalDye - D0 }
24. icaleak=Vmax * cai_rest/(cai_rest + Kca_pmp)
25. ica_pmp=0 }
26. LOCAL frat[NANN]
27. PROCEDURE factors() { LOCAL r, dr2
28. r=1/2 dr2=r/(NANN-1)/2 vol[0]=0 frat[0]=2*r
29. FROM i=0 TO NANN-2 { vol[i]=vol[i] + PI*(r-dr2/2)*2*dr2
30. r=r - dr2 frat[i+1]=2*PI*r/(2*dr2) r=r - dr2
31. vol[i+1]=PI*(r+dr2/2)*2*dr2 } }
32. BREAKPOINT SOLVE states METHOD sparse
33. ica=ica_pmp }
34. LOCAL dsq, dsqvol, Tvol
35. KINETIC states {
36. COMPARTMENT i, diam*diam*vol[i] {ca CaBuffer Buffer CaDye Dye}
37. ~ ca[0] << ( - (ica)*(10000)*PI*diam*frat[0]/(2*FARADAY))
38. FROM i=0 TO NANN-2 {
39. ~ ca[i] < - > ca[i+1] (DCa*frat[i+1], DCa*frat[i+1])
40. ~ Dye[i] < - > Dye[i+1] (Dydif*frat[i+1], Dydif*frat[i+1])
41. dsq=diam*diam
42. FROM i=0 TO NANN-1 {
43. dsqvol=dsq*vol[i]
44. ~ ca[i] + Buffer[i] < - > CaBuffer[i] (k1buf*dsqvol, k2buf*dsqvol)
45. ~ ca[i] + Dye[i] < - > CaDye[i] (k1dye*dsqvol, k2dye*dsqvol)}
46. cai=ca[0]
47. : Ca extrusion
48. ica_pmp=Vmax * cai/(cai + Kca_pmp) - icaleak
49. : Ca indicator
50. CaDch=0 Tvol=0
51. FROM i=0 TO NANN-1 {
52. CaDch=CaDch + CaDye[i]*vol[i] Tvol=Tvol + vol[i] }
53. FDca=CaDch/(TotalDye*Tvol) }
```

(細胞膜は側面のみが存在)としてモデル化される。円柱の長さを直径に等しくすると膜面積は球と同じになる。Ca²⁺の拡散や緩衝反応を扱うため、この円柱を2図に示すように同心状の殻層に区分し[6](本モデルでは殻層の数 NANNは3行目の DEFINEで4としている), 単位直径の円柱の最外殻の0から数えi番目の殻層の単位長当たりの体積を vol[i]とする。この数値は31行の式で予め計算される。そこで、直径が diamの円柱の単位長さ当たりの殻層の容積は diam* diam*vol[i]となる。こうして区分した領域間で、Ca²⁺とCa²⁺指示薬の拡散, Ca²⁺とバッファや指示薬との結合・解離反応を扱うことになるが,そのためには、これらに関する物質を示強変数である濃度ではなく、各殻層内での実際の量で与える必要がある。NEURONではこの面倒な操作を COMPARTMENTというstatementが実行してくれる。37行では、細胞の表面積 frat[0]を通り流入するCa²⁺電流(ica)による最外殻層のCa²⁺濃度(ca[0])の変化が計算される。なお、icaはCa²⁺チャネル機構(他のmodファイルに記述)やCa²⁺ポンプ機構によるものがインタープリターで総合され、ここで使用される。Ca²⁺は細胞内に均一濃度で存在する固定バッファ(Buffer, CaBuffer), および可動性Ca²⁺バッファとしても働くCa²⁺指示薬(Dye, CaDye)との間で結合・解離反応をしながら動径方向に拡散する。Ca²⁺の拡散定数(Dca), 殻層の境界の単位長当たり面積(A[i+1]), 殻層の間隔(r)とすると, Ca²⁺のfluxは (ca[i+1] - ca[i]) * Dca * A[i+1]/ rとなる。これは順方向、逆方向の速度定数が共にDca * frat[i+1](ただし frat[i] = A[i]/ r)である反応速度式を解くことに相当する。frat[i]は factors()の計算として INITIALにより数値化されているので、これを使用し NMODLにおいては39行目のような書式でCa²⁺の拡散が、同様に、40行でCa²⁺指示薬の拡散が計算される。更に、Ca²⁺の固定バッファおよび色素への結合・解離反応も速度式として表せ、それぞれ、44行と45行のように記述される。また、Ca²⁺の排出は48行に示すように最大排出速度VmaxのMichaelis-Mentenスキームで

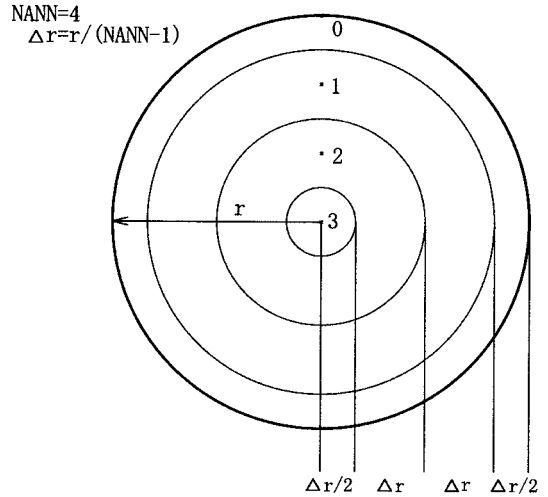


図2. 円柱型で近似されたニューロンの断面図。細胞内をNANN個(この場合は4)の殻層に区分し、各殻層内での化学反応, および殻層間の物質の拡散を扱う。半径rのニューロンでは、殻層の厚さは $r/(NANN-1)$ であるが、最外殻と最内殻の厚さは $r/2$ とする。この区分法により、最外殻と最内殻の容積および物質濃度はDrの2次の精度で計算され、物質の拡散は黒点で表されたr距離にある不連続点への移動として計算される。

作動するポンプ機能を仮定した。なお、静止時のCa²⁺レベル(ca_rest)を安定に保つようにCa²⁺のリーク電流(icaleak)が加えられている。最後に、50から53行で、蛍光色素で計測される[Ca²⁺]_iレスポンスに対応させるため、各殻層に存在するCa²⁺と結合した色素(CaDye[i])の細胞内における総和の全色素量に対する比率(FDca)が計算される。

7. mod ファイルのコンパイル

各機能毎に作製したmod fileは、それぞれに、計算式の運用における単位の整合性, syntaxの適切さなどのチェックを行う必要がある。このために、NEURON基本システム中に modlunit というプログラムが存在する。NEURONスタートメニューの <mknrndll DOS box>を開き、DOSのコマンドラインで、カレントディレクトリーをmod fileの存在するフォルダーに移し、そこで、modlunit carespのように、拡張子を付けずにmod fileの名前を書き、enter keyを押す。この

チェックに合格した mod file を集めてコンパイルを行うために、上記の DOS コマンドラインで、今度は mknrndll と打ち込み、その後、必要 mod file 名を拡張子を付けずに、スペース区切りで並べ、enter key を押すと nrnmech.dll という dll ファイルが作製される。なお、mknrndll の後に、ファイル名を何も書かずに enter key を押すとフォルダー中に存在する全ての mod file がコンパイルされる。今回の試行では、上記の care-sp.mod と、NEURON をインストールしたフォルダー (nrn/Examples/nrniv/nmodl) に存在する Hodgkin-Huxley 型の Na^+ 、 K^+ およびリークチャネル機構 (hh1.mod)、および電位依存性 Ca^{2+} チャネル機構 (Cachan.mod) の 3 機構のファイルを新規フォルダー (例えば C:\Cell_A) 内に集め、上記の要領で mknrndll によるコンパイルを行ない、仮想細胞モデル Cell_A を構成した。コンパイルが無事完了し、フォルダー内に nrnmech.dll が作製されていることを確認する。なお、ここに紹介している NEURON (Ver.4.3.1) は Windows 版であるが、modlunit、mknrndll の部分だけは AT 互換機の DOS モードで実行されるプログラムであるため、NEC の旧 98 系の機種ではうまく作動しない。

8. hoc ファイルの作製

作製された dll ファイルを NEURON のインタープリターに呼び込み、シミュレーションを実行するときには、通常、hoc ファイルのクリックでスタートする。そこで、必要最小限の機能を持つ起動用 hoc ファイルを作成する方法について述べる。まず、NEURON スタートメニューから <nrngui> を立ち上げる。現れる Main Menu の <File> <load dll> で、上記で作製した nrnmech.dll を呼び込む。次に、Main Menu の <Build> から <Cell Builder> を開き、<Topology> の <Basename> を Cell_A に書き換え、Click to にリストされたボタンの中の <Change name> をクリックした後、表示ウインドーの <soma> の文字をクリックすると Cell_A に書き換えられる。次に、<Geometry> の <Cell_A> を選択し、細胞

の長さ <L> と、直径 <diam> をチェックし、<Specify Strategy> のチェックを解除すると、数値記入用のテキストボックスが現れるので、 L (um) = 15, diam (um) = 15 と数値を記入する。更に、<Biophysics> <Cell_A> とし、hh1, cachan, caresp の 3 機構をチェックボタンで選択し、<Specify Strategy> のチェックを解除し、hh1 および cachan の機構のパラメーターを $gnabar_hh1$ (mho/cm²) = 0.25, $gkbar_hh1$ (mho/cm²) = 0.025, gl_hh1 (mho/cm²) = 0.00025, $e1_hh1$ (mV) = - 60 mV, $pcabar_cachan$ (mA/cm) = 0.0002, と設定する。そして、最後にこの設定を、<Management> <Export> から Cell_A.hoc という名前でファイルに保存する。

こうして作製した Cell_A.hoc ファイルをテキストエディターで開くと上記の Cell Builder による設定が一定の書式により書き込まれていることが分かる。このファイルの先頭に xopen ("\$(NEURONHOME)/lib/hoc/Nrngui.hoc") というコマンドを一行書き加えると、NEURON 起動用 hoc ファイルに改造することができる。次回の起動はこの Cell_A.hoc のファイル名をクリックすれば、Main menu と、標準化された表示環境が呼び込まれ NEURON が立ち上がる。

9. Cell_A によるシミュレーションの実行例

それでは、Cell_A を使用したシミュレーションの実行例について述べてみよう。 Ca^{2+} 指示薬の Ca^{2+} 緩衝作用により、膜電位変化や細胞内遊離 Ca^{2+} 濃度変化にどのような影響が出るかを調査してみる。まず、Cell_A.hoc をクリックして NEURON を起動し、Main menu の <Tools> の <Distributed mechanisms> から、本稿の第 3 項で説明した要領で、以下の 4 点につき確認または設定を行う。1) コンパイルした 3 つの機構が inserter に組み込まれていること、2) パラメーターパネルに、Cell Builder で記入した値が初期値として正しく書き込まれていること、3) hh1 機構に関するパラメーターで Na^+ 、 K^+ の平衡電位 (ena) を 50 mV および - 80 mV とする、4) 環境温度

(celsius) を 25 度とする．次に，これも第 3 項で説明した要領で，Main menu の <Graph> から，膜電位 (v)，細胞内遊離 Ca^{2+} 濃度， Ca^{2+} と結合した蛍光指示薬の割合 (FDca_caresp) の 3 つグラフ表示パネルを用意する．このうち，細胞内遊離 Ca^{2+} 濃度表示パネルには最外殻の濃度

(ca[0]_caresp) とその次の層の濃度 (ca[1]_caresp) の二つのグラフを表示する．

上記の準備を済ませ，<Tools> から RunControl パネルを開き，Tstop を 100 ms としてシミュレーションを開始する． Ca^{2+} 指示薬の Ca^{2+} 緩衝作用により差異が顕著になるよう，指示薬濃度

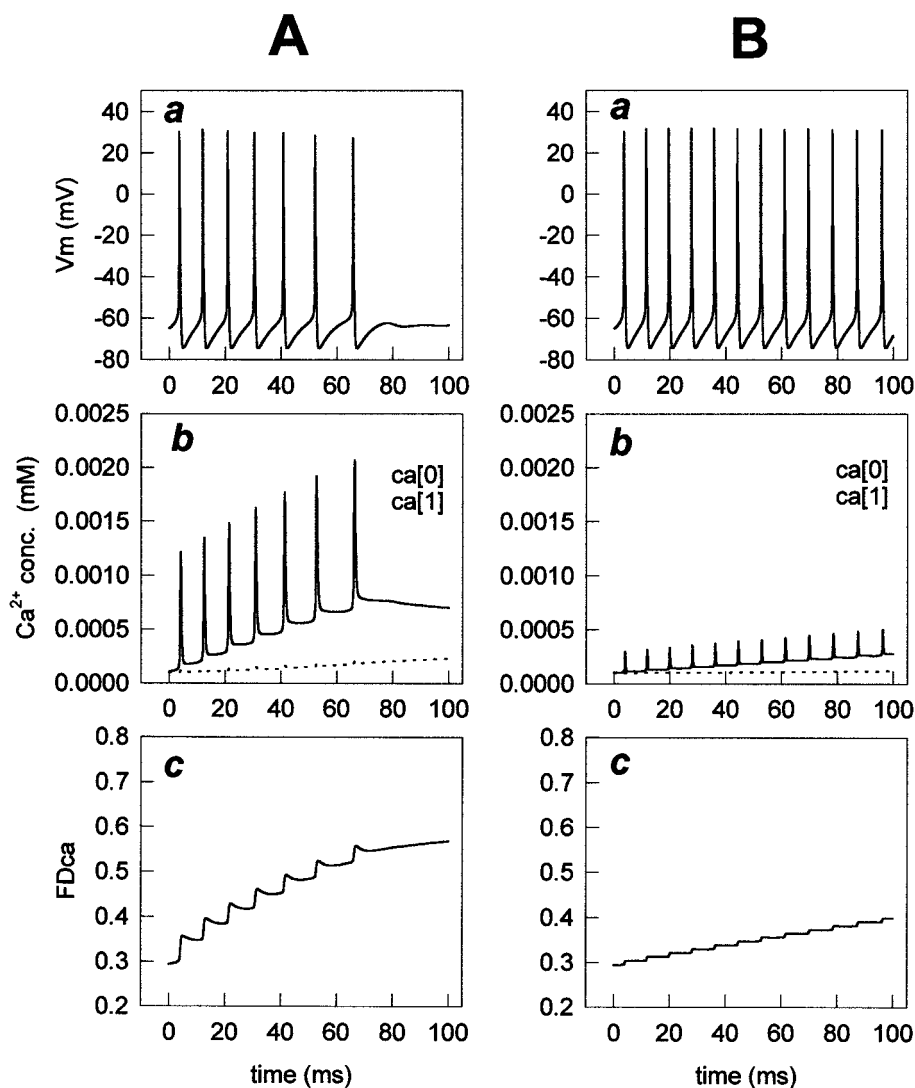


図3．本文中で作成した仮想ニューロンモデル (Cell_A) によるシミュレーション結果の一例．A 図は細胞内 Ca^{2+} 指示薬濃度 (TotalDye) を 0.0001 mM, B 図は 0.1 mM として, 他のパラメーターは固定し, (a) 自発膜電位変化のパターン, (b) 最外殻 (実線) と第 2 層 (点線) における遊離 Ca^{2+} 濃度変化, (c) Ca を結合している指示薬の全指示薬量に対する比率 (FDca) を比較した．パラメーターの設定値: Ca^{2+} 指示薬の拡散定数 Dydif, 指示薬と Ca^{2+} の間の解離定数 Dkd, 結合速度定数 k1dye, ポンプの最大 Ca^{2+} 運搬速度 Vmax と Ca^{2+} の解離定数 Kca_pmp, 細胞内遊離 Ca^{2+} 濃度の静止レベル cai_rest, Ca^{2+} の拡散定数 Dca, 細胞内 Ca^{2+} バッファへの Ca^{2+} 結合速度定数 k1buf, バッファと Ca^{2+} の間の解離定数 CKd は本文中のプログラムリスト参照．他は本文参照．

(TotalDye) を 0.1 mM と 0.0001 mM と極端に変えた場合の様子を比較したのが 3 図である。リークコンダクタンスの平衡電位 (e_{l_HH}) が -60 mV と浅いため、細胞は自発放電モードとなる。A 図の場合、指示薬濃度は 0.0001 mM で、 Ca^{2+} はほとんど細胞内固定 Ca^{2+} バッファのみの緩衝を受ける。 Ca^{2+} チャネルからの Ca^{2+} 流入による最外殻での遊離 Ca^{2+} 濃度の上昇は大きく (3A (b)), これを排出するための Ca^{2+} ポンプ電流が過分極方向に働き自発放電間隔は大きくなり、やがて中断する (3A (a))。これに反し、B 図の指示薬濃度 0.1 mM の場合では、流入 Ca^{2+} は強力に緩衝を受け、最外殻にあっても遊離 Ca^{2+} 濃度の上昇は小さく、自発放電はほとんど影響を受けない。 Ca^{2+} 指示薬による細胞内で平均化された Ca^{2+} シグナルに対応する FDca の値は予想される通り、指示薬濃度が高いと指示薬自身の Ca^{2+} 緩衝能が大きいため上昇が抑えこまれる (3A (c) と 3B (c))。

Cell_A に先に述べた、 Na^+/Ca^{2+} 交換機能 (nacaex.mod) や、MyFirstNEURON [5] で使用されている Ca^{2+} 依存性 K^+ チャネル機構 (ic.mod, iAHP1.mod, iAHP2.mod) などを加えれば、 Ca^{2+} 動態に関するより多彩な事象に対応できると期待される。

10. 神経細胞形態とネットワークの構築

NEURON の真価は、複雑な神経細胞の形態を実態に近い 3 次元モデルとして構築し、そこに興奮性や抑制性のシナプスを分布させ、神経ネットワークを組んでシミュレーションができる点にある。ニューロンの形態については、錐体細胞、顆粒細胞、介在神経などの種々形態を展示する web サイト [7] があり、同じサイトにその形態を編集したり、NEURON に乗せるために hoc ファイル化するエディターも存在する。また、神経ネットワークに関しては NEURON Main Menu の <Build> の中に <NetWork Cell> および <NetWork Builder> というツールが最近のバージョンから登場した。これらはユーザーによるネットワークモデルの作成を容易にし、また多数の神経が関係するモデルの構築を助ける意図で開発された

ものである。NEURON ホームページ [1], Documentation 中の, “A tutorial on the Network Builder” に運動ニューロン発火における Renshaw 抑制のモデルを構築する例などが図入りで詳しく解説されている。ここでの実例演習として、NEURON Demo 中の <Pyramidal> で構築されている細胞へ、別の神経軸索からの興奮性シナプス入力 の付与を扱うが、これも上記 tutorials に則り作業を進めることが可能である。しかし、2 つのニューロン間の単シナプス結合など、単純なモデルは初めから hoc ファイルに書く方が簡単であり、また、複雑な細胞形態のデータがどのような形式でファイル書かれているかを知る上でも参考になるので、敢えて、手作業的な構築を行う。

実際の作業としては、<Pyramidal> の形態を記述する座標が記載された hoc ファイル中に、別の神経の axon とのシナプス結合を付加するための記載を追加する。まず、NEURON Demo のプログラムで、<Pyramidal> を選択し、Main Menu の <Build> から <Cell Builder> を開く。<Management> に入り、大小二つの <import> ボタンをクリックすると、画面に錐体細胞の形態とセクション名が表示される (詳細は NEURON ホームページ [1], Documentation 中の, “A tutorial on the CellBuilder” を参照)。<Topology> に行き、<Basename> を axon に変え、<Make section> を選択し、細胞表示領域でクリックすると、axon という section が付加される。実はここで付加された要素は既に画面上に存在する Pyramidal 細胞の一部となって接続されてしまうため、後で切り離す操作を行う必要があるが、axon という要素をとにかく登録しておく。また、axon の位置も後で数値的に座標で指定するので、挿入場所はどこでもよい。<Geometry> の項では数値計算用の空間区分精度に関係する grid として d_lambda を <all> が選択された状態でクリックする。これにより全要素に空間定数の 1/10 に細分された差分を用いた数値計算が行われるよう自動的に設定される。L や diam は、hoc ファイルに書かれた座標より自動的に計算されるので、ここで指定してはならない。次に、<Biophysics>

でRa (細胞内溶液の比抵抗), cm (膜電気容量), pas (受動的コンダクタンス) を, <all> が選択された状態でクリックする. そして能動機構としてhhを, soma, dendrite_5および, axonに振り付け, パラメーターをgnabar_hh=0.2, gkbar_hh=0.036, gl_hh=0.0003, el_hh= - 65とする. 以上を設定後, <Management>に戻り, <export> から hocファイルをax_pyr.hocとして保存する.

上記のax_pyr.hocをテキストエディターで開き, 細胞形態の座標が多量のデータとして書き込まれていることを確かめ, 以下の作業を行う. 先ず, axonを付加したとき, 例えばdendrite_5(1)部位に接続されたとするとconnect axon(0), dendrite_5(1)との記載があるので, これを見つけ削除する. これで, axonは他の要素から切り離される. 次に, axonの位置を記載した部位

の数値をproc basic_shape(){ axon {pt3dclear() pt3dadd (-240, -180, 0, 2) pt3dadd (60, -180, 0, 2) }}のように書換える. 前のpt3daddの括弧内で, 最初の3つの数字がaxonの起始部のx, y, z座標値, 4つめの数字がaxonの太さを表し, 後ろのpt3daddは終末部の対応する情報を与える. これで太さ2 μm のaxonが左方向から水平に300 μm ほど伸び, pyramidal cellのdendrite_5の中央部位近くで終わるよう配置される(図4A参照). 更に, CellBuilderで設定時の都合で, soma, dendrite_5, axonの3部位には能動機構hhと受動機構pasの両方が挿入されているので, proc biophys()中のこの3者の項目内にそれぞれ, uninsert pasを付け加える. ところで, dendrite_5のpt3daddを眺めると, 太さが先端方向に向けて細くなっており, シナプスを作る予定の中央部位では0.9 μm 程度となっている. 仔細に

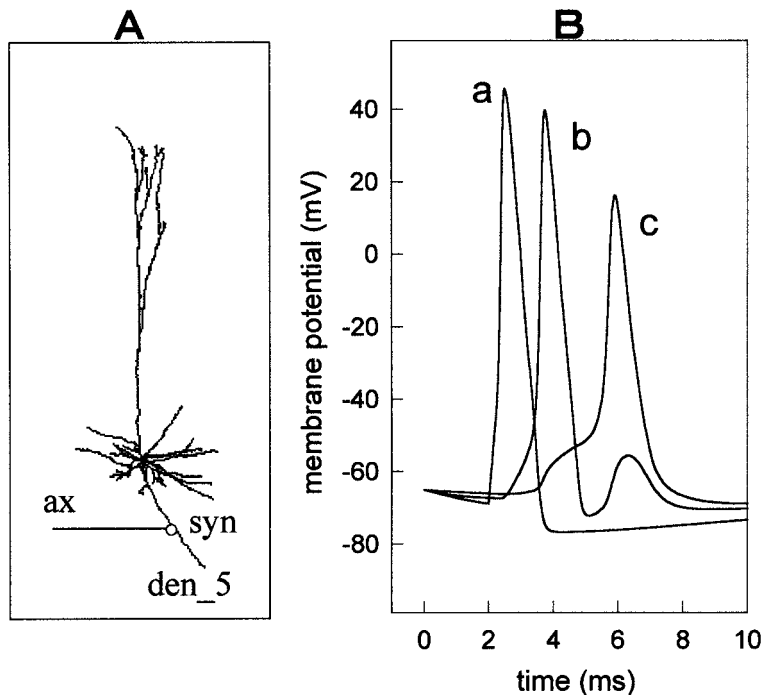


図4. (A図) NEURON Demoの<Pyramidal>に使用されている錐体ニューロンモデルの第5樹状突起 (den_5) に, 外来ニューロンの軸索 (ax) を興奮性シナプス (syn) で結合させた (本文参照). (B図) axの左端を電気刺激したとき, (a) ax部位, (b) den_5部位, (c) pyramidal neuronのsomaに生ずる膜電位変化.

検討した訳ではないが、この細さでは通常の設定のシナプス伝達で活動電位を発生させるのが困難であるようなので、 $2\mu\text{m}$ 以下の部位を全て $2\mu\text{m}$ に書換えた。最後に、synpという機構を用い興奮性シナプスを dendrite_5の中央部に埋め込み、ポインターをセットする。シナプス前要素である axon 終末の膜電位 (axon.v (0)) が閾値膜電位 (vprethresh) を超えると、シナプス電位の平衡電位 (e), 最大コンダクタンス (gmax), 時定数 (tau) によって特徴付けられるシナプス電流が流れる。この仕組みは下記のようなリストを ax_pyr.hoc の最後に書き加えることで実現できる。

```
objref syn
dendrite_5 syn = new synp (.5)
setpointer syn.vpre, axon.v (1)
deadtime_synp = 3
{syn.tau=1 syn.gmax =0.01 syn.e=0 syn.vprethresh=0}
なお、ax_pyr.hocを NEURON 起動用ファイルとする場合にはファイルの先頭に xopen ("$(NEURONHOME)/lib/hoc/Nrngui.hoc") を書き入れる。
```

シミュレーションを実行するには、新たにフォルダーを作り、そこに ax_pyr.hoc と synp 機構を記載した presyn.mod (\$(NEURONHOME)/Examples/Nrniv/Nmodel に存在) を置き、既に述べた方法で presyn.mod のみをコンパイルし、nrnmech.dll を作製する。図 4B は、celsius=15 とし、Tools/Point Processes/Managers/Electrode の IClamp で電流刺激条件を設定後、<Location> で axon の左端を指定し、刺激したときに見られる axon, dendrite_5, それに soma での活動電位波形である。dendrite_5 の二回目の電位変化は、soma の活動電位による再還入である。dendrite_5 の活動電位がシナプス経路であることは、この部位の gnabar_hh をゼロにして、シナプス

電位を観測し、これが synp のパラメーターにより変化することで確かめることができる。

11. おわりに

本稿でシミュレーションを実行したのはいずれも全くの仮想的モデルであり、また初歩的なものであったが、体験的に読んで下さった読者には、NEURON の概要をかなりの程度理解して頂けたのではないかと思う。NEURON プログラムは開発者の努力で今後も進化することが期待され、ユーザーグループもインターネットを介して活発な情報交換が可能である。今後、NEURON を基礎とし、あらゆる種類のチャンネルやレセプターなどの機能要素、それらを組みこんだ細胞単位、さらに、それらをネットワークで繋いだ機能モジュールなどに関する新たな様式のデータベースが構築されて行く可能性もあろう。そのような可能性も視野に入れ、一人でも多くの方が NEURON の活用を試みて下れば、自身も初心者でありながら、このような解説記事を書いた著者らにとり望外の幸せである。

文 献

1. <http://www.neuron.yale.edu>
2. Hodgkin, AL & Huxley AF: A quantitative description of membrane current and its application to conduction and excitation in nerve. J. Physiol. 117: 500-544, 1952.
3. <http://hines.med.yale.edu/neuron/about/what.html>
4. Neher E: The use of fura-2 for estimating Ca buffers and Ca fluxes. Neuropharmacol. 34: 1423-1442, 1995.
5. <http://tesla.salk.edu/arthur/MyFirstNEURON.html>
6. Hines ML & Carnevale NT: The NEURON simulation environment. Neural Computation 9: 1179-1209, 1997. (NEURON ホームページの "Documentations" にプレプリントの掲載)
7. <http://www.cns.soton.ac.uk/jchad/cellArchive/cellArchive.html>