Realistic Single Cell Modeling

WAM-BAMM*05

San Antonio, TX

Dieter Jaeger, Ph.D.

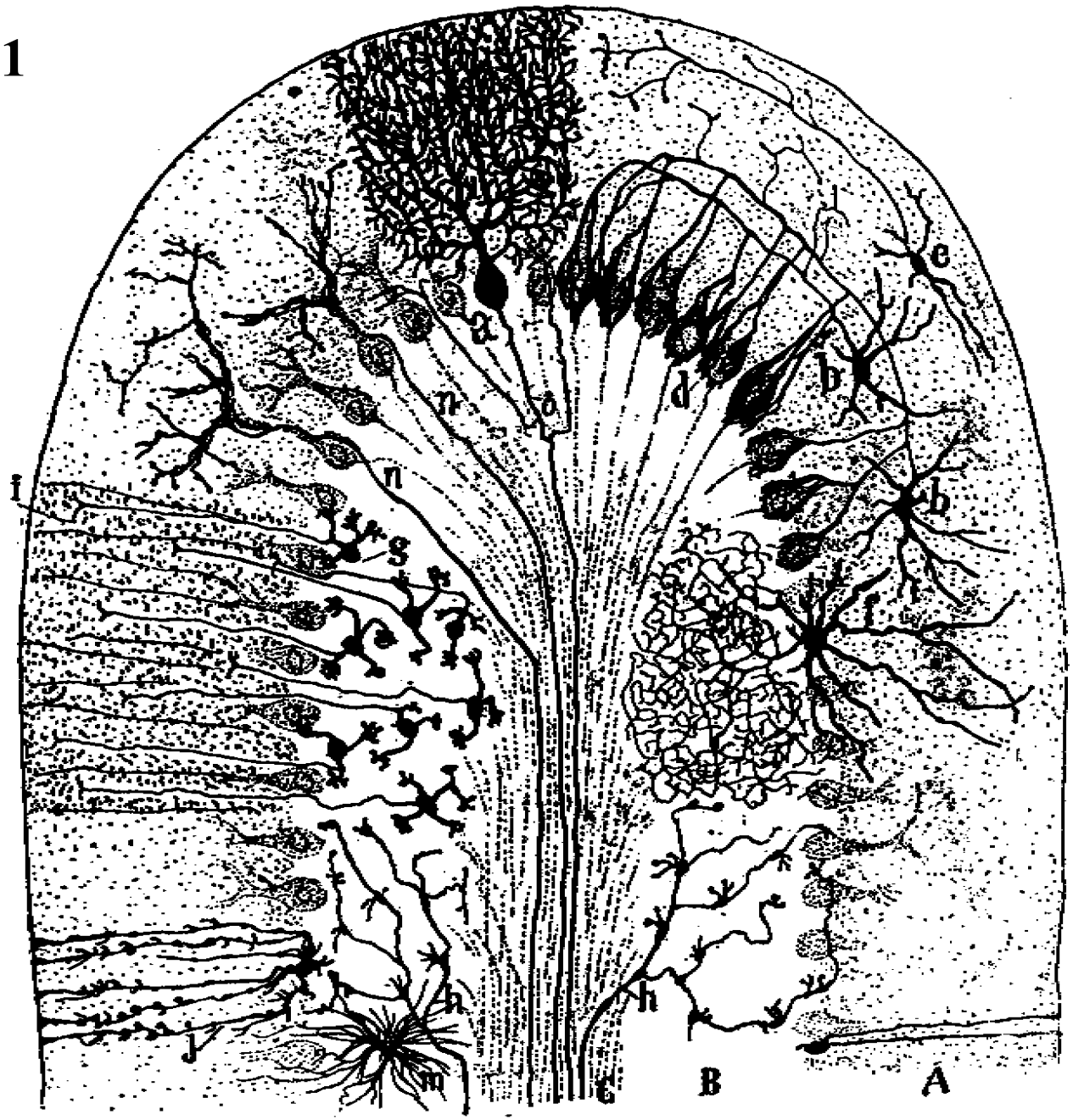
Department of Biology

Emory University

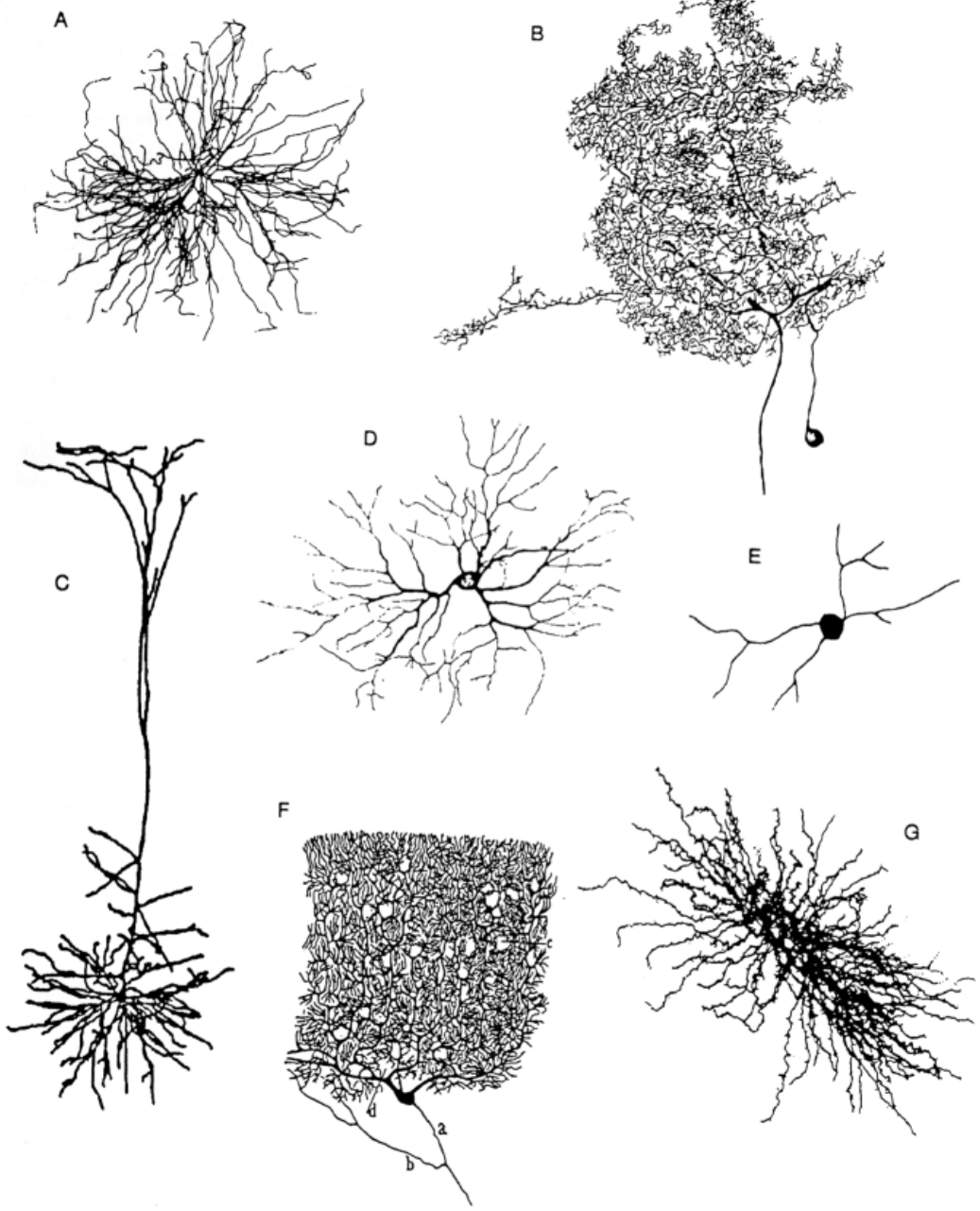
Atlanta, GA

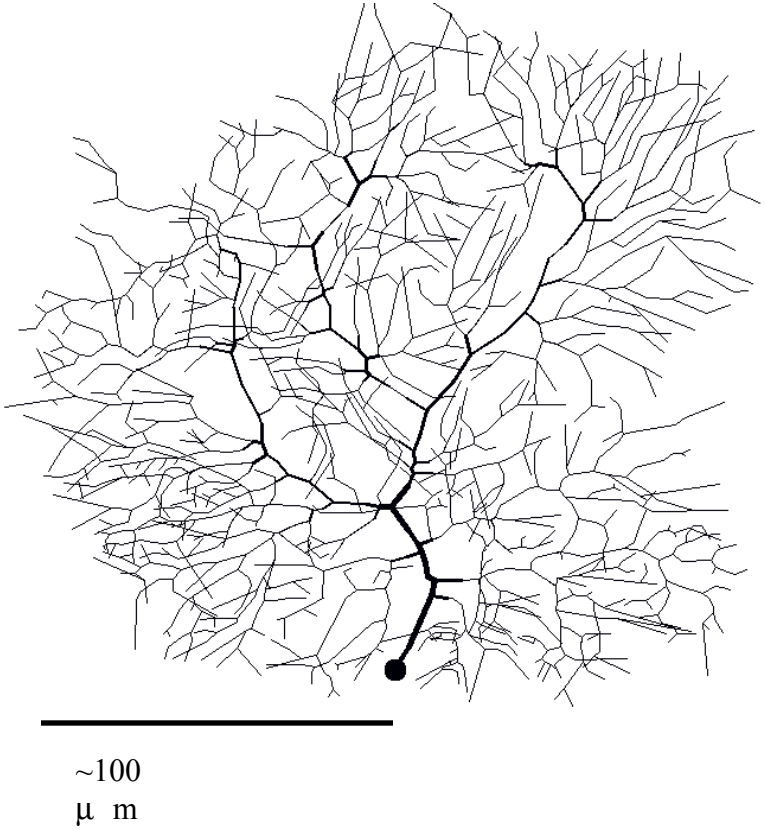
djaeger@emory.edu

Cajal, 1901



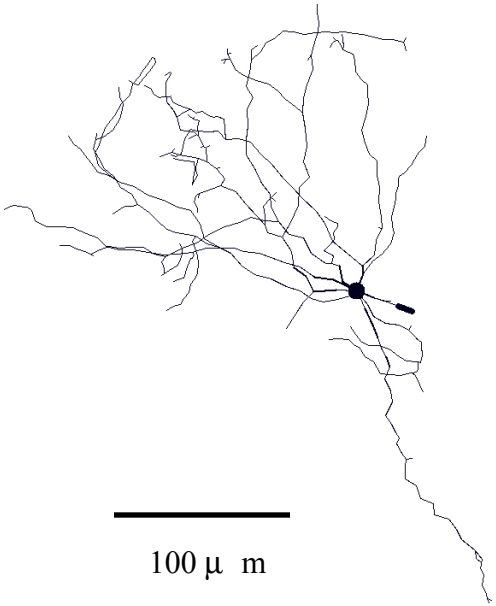
Cajal, 1901





Purkinje cell

- surface area: 261,000 μ m^2
- number of synapses (ex/in): 175,000 / 5,000
- number of inputs / s 350,000 / 10,000



DCN neuron

- surface area: 11,056 μ m^2
- number of synapses (ex/in): 5,000 / 15,000
- number of inputs / s 25,000 / 750,000

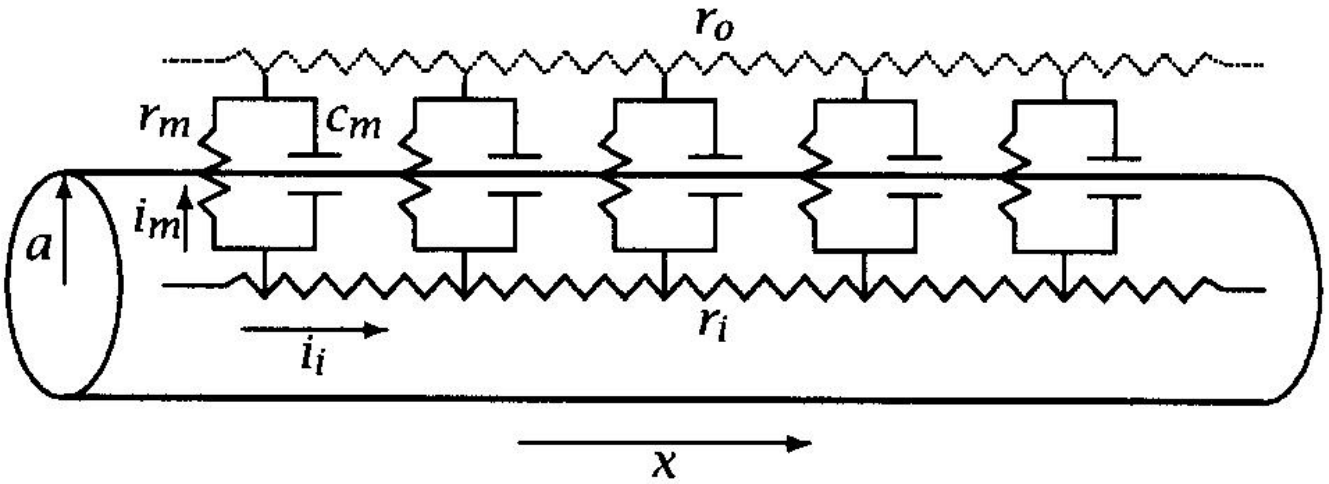


Figure 4.6 Diagram for current flow in a uniform cylinder such as an axon or segment of dendrite.

4.4. Nonisopotential cell (cylinder)

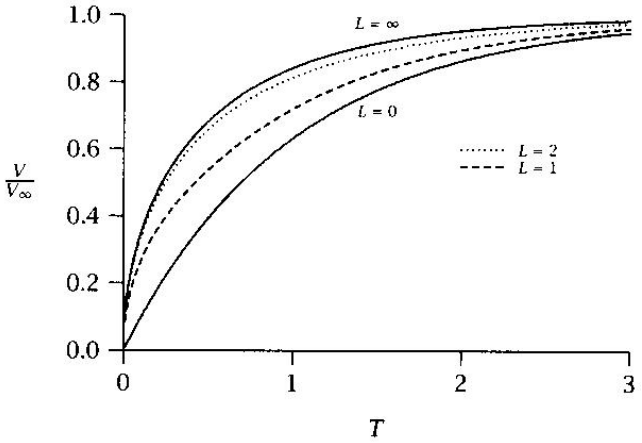


Figure 4.16 Comparison of normalized charging curves for finite cables of different electrotonic lengths. A step of current is injected, and the voltage is measured, at $x = 0$. (After Jack et al. 1975.)

$$V_m(t,x) = C_0 e^{-t/\tau_0} + C_1 e^{-t/\tau_1} + C_2 e^{-t/\tau_2} + \cdots + C_n e^{-t\tau_n}.$$

$$\lambda = \sqrt{\frac{r_m}{r_i}} = \sqrt{\frac{aR_m}{2R_i}}.$$

From Johnston and Wu

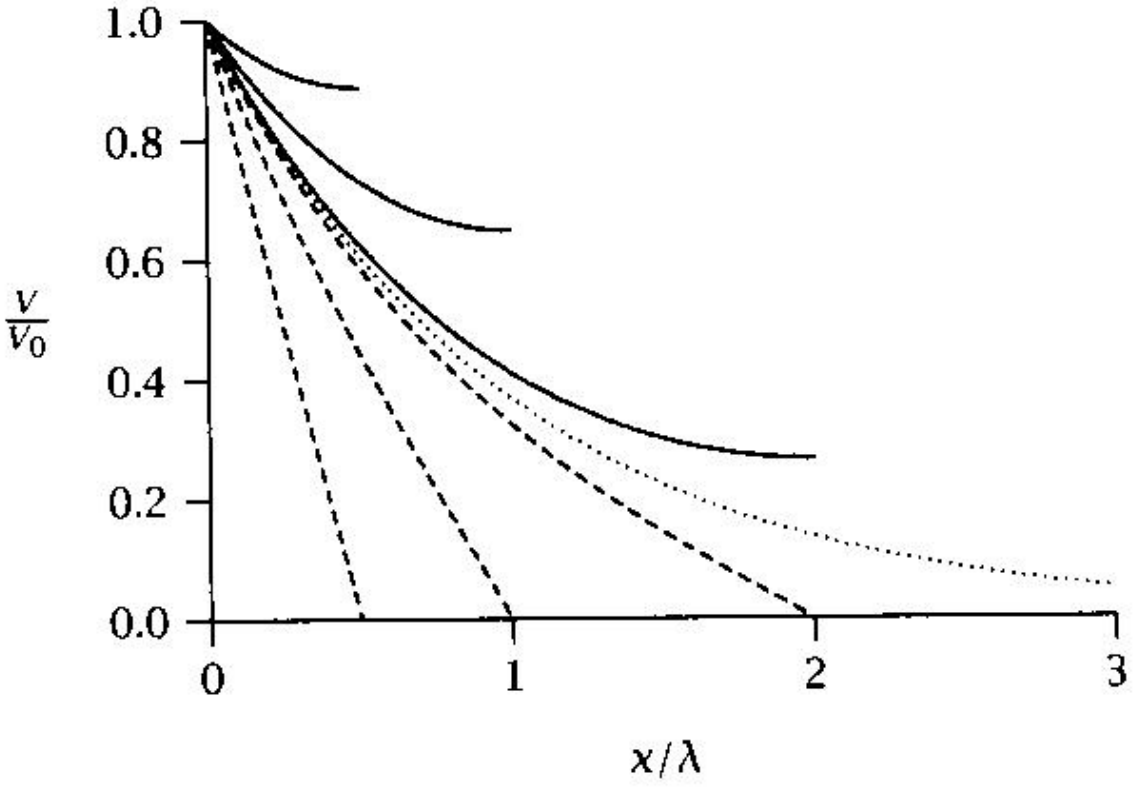


Figure 4.15 Comparison of voltage decays along finite cables of different electrotonic lengths and with different end terminations. Current is injected at $x = 0$. The solid lines are for finite cables with sealed ends, the dashed lines are for finite cables with open ends, and the dotted line is for a semi-infinite cable. (After Rall 1959.)

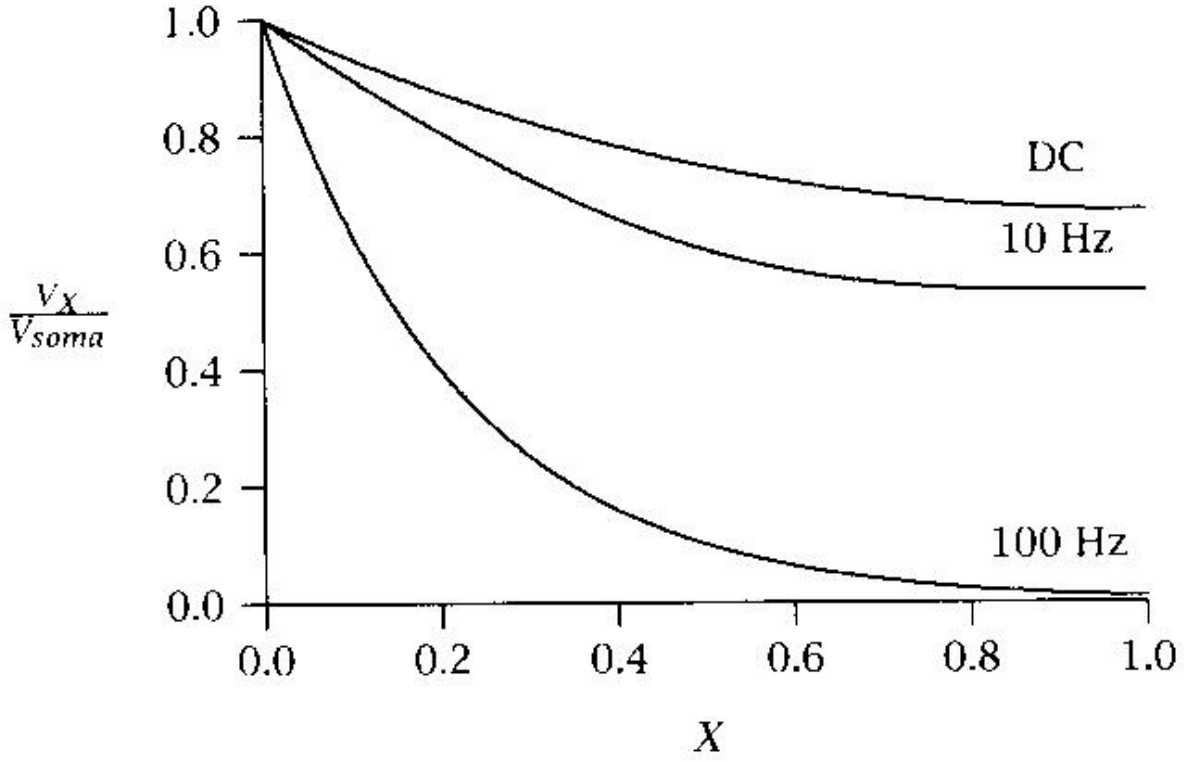


Figure 4.18 Voltage attenuation along a finite-length cable ($L = 1$) for current injections (DC to 100 Hz) at $X = 0$ (i.e., soma) ($R_m = 50,000\ \Omega\text{-cm}^2$).

From Johnston and Wu

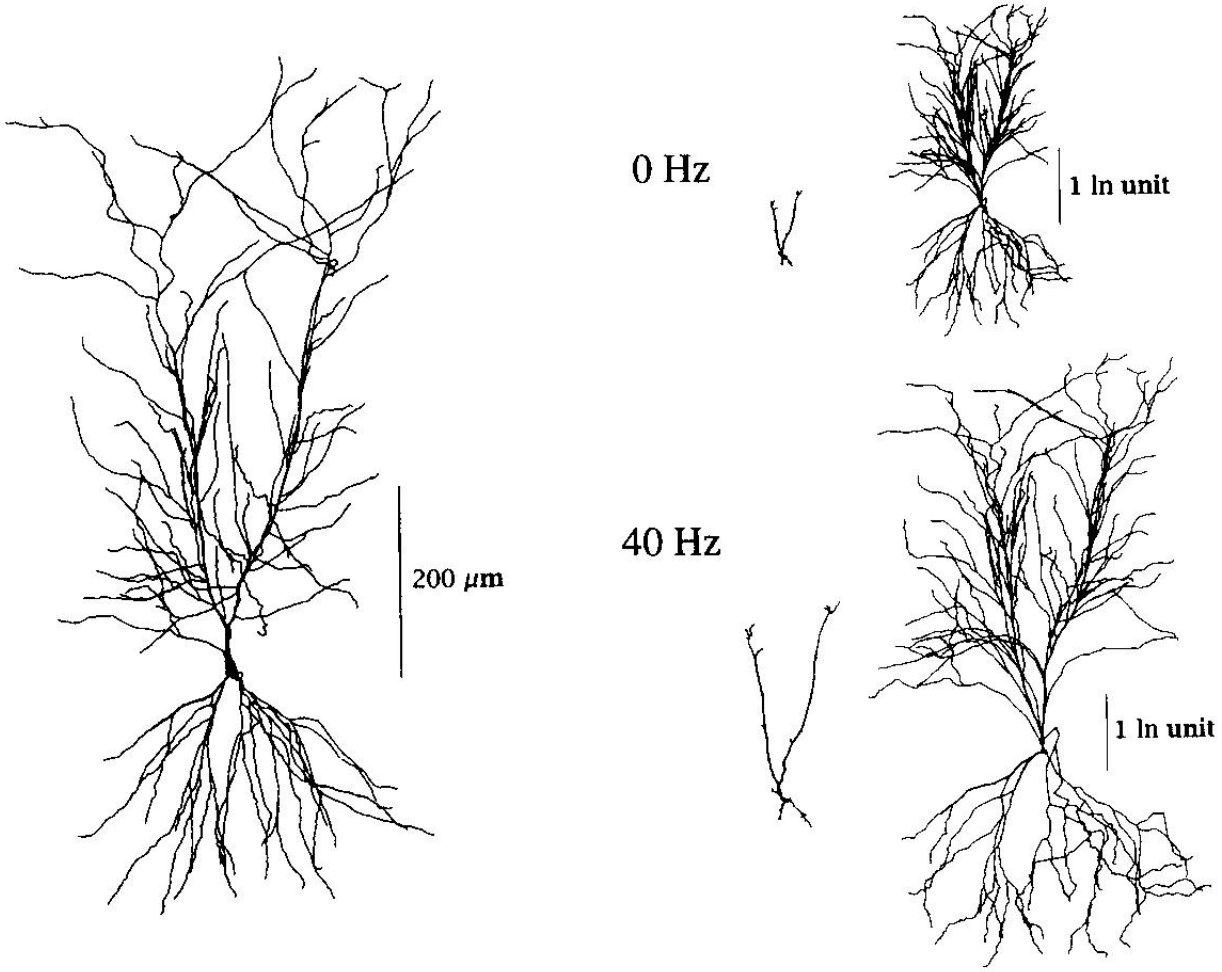
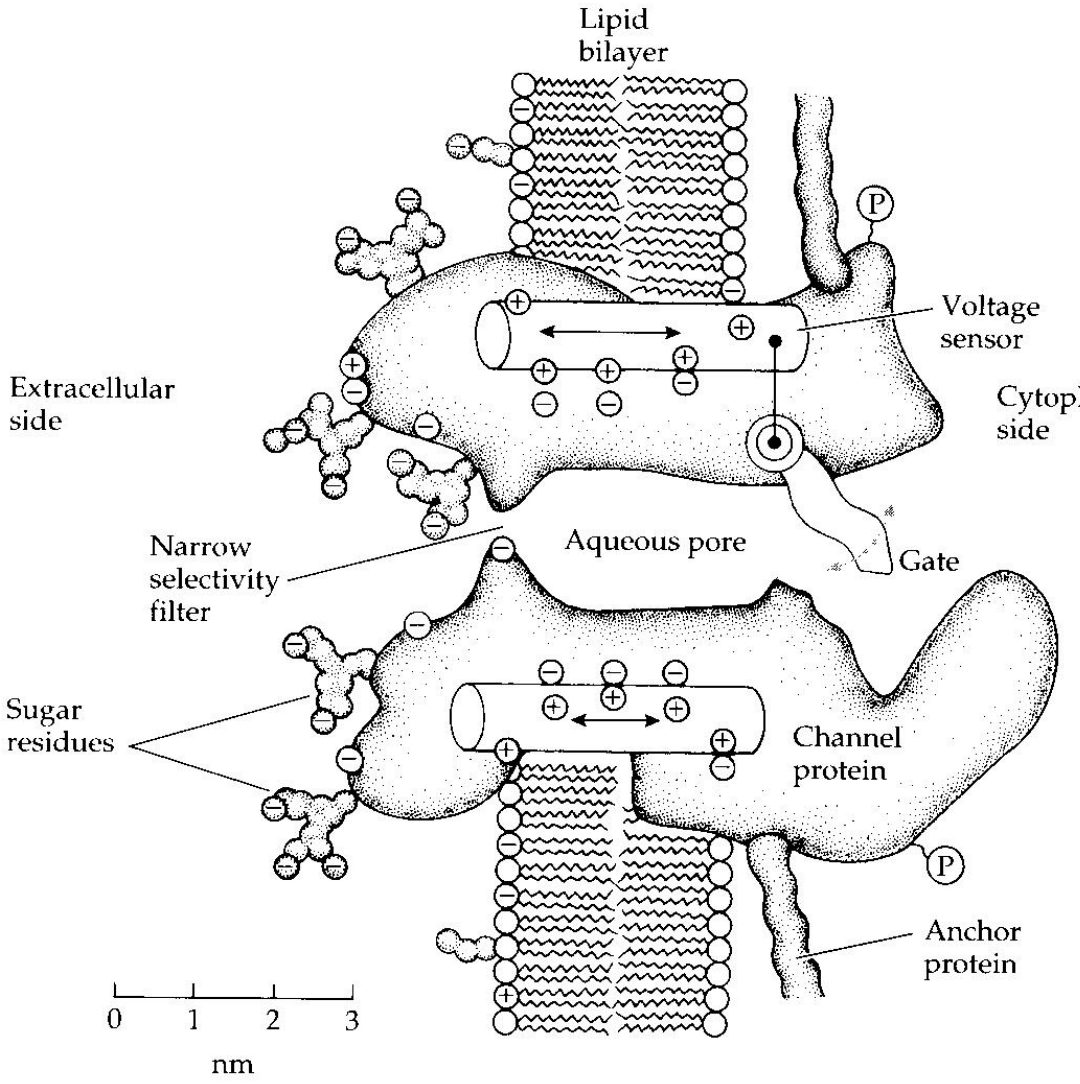


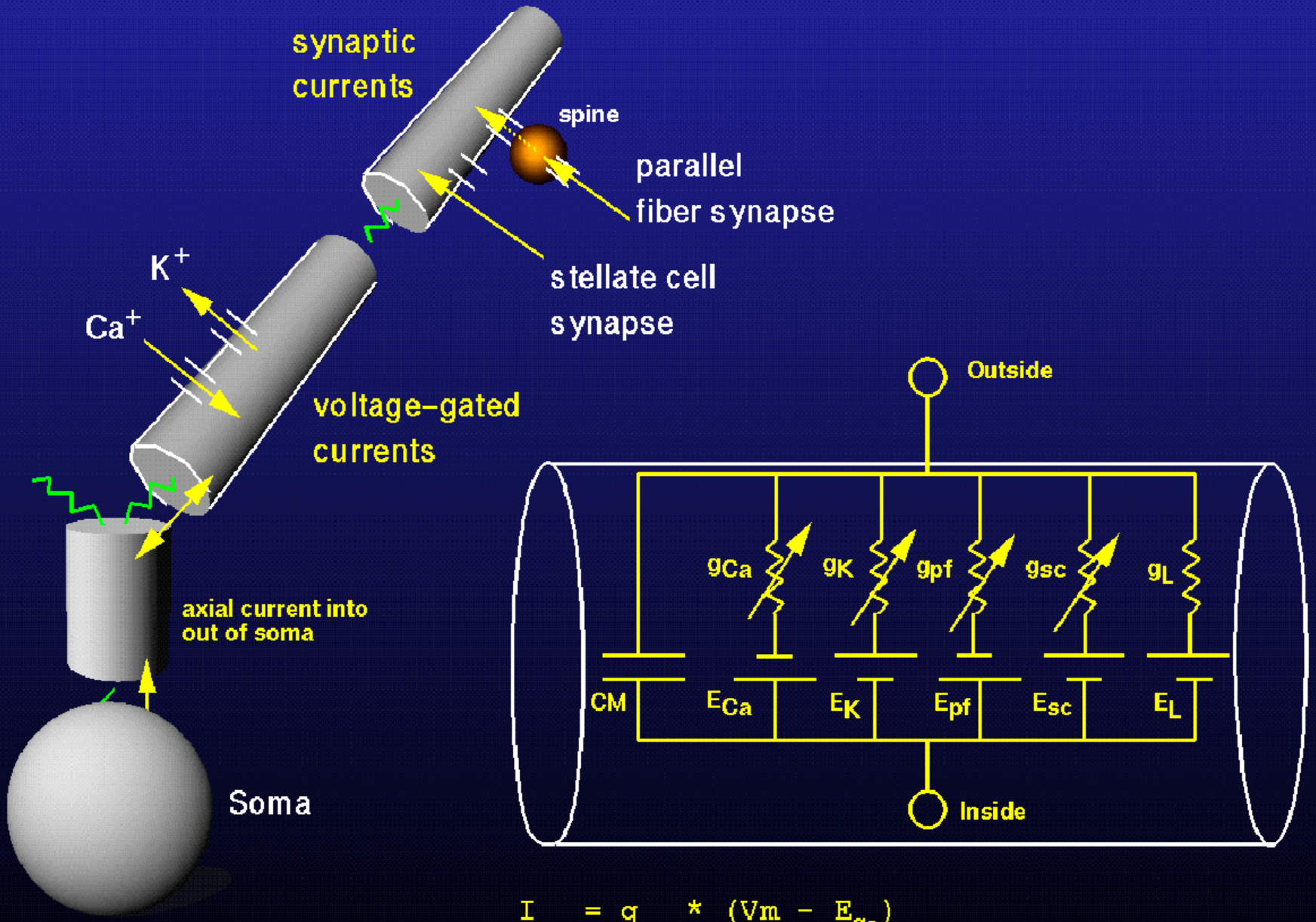
Figure 4.30 Morphoelectrotonic transforms. A reconstructed CA1 neuron is shown on the left, and the morphoelectrotonic transforms for the decay of potential from the soma out the dendrites and from the dendrites toward the soma are illustrated on the right. The neuron diagrams on the right are drawn according to a scale that represents the ln of the voltage attenuation ratios (1 ln unit represents voltage attenuation of $1/e$ from the site of current injection). The top middle transform represents the ln(attenuation) of a DC signal applied to the soma as it decays to different sites in the dendrites. The top right transform represents the ln(attenuation) of a DC signal applied to different sites in the dendrites as it decays toward the soma. The two transforms at the bottom right represent the same measurement as those above but resulting from the application of a 40 Hz sine wave signal instead of 0 Hz. (Kindly provided by Tsai, Carnevale, Claiborne, and Brown.)

From Johnston and Wu



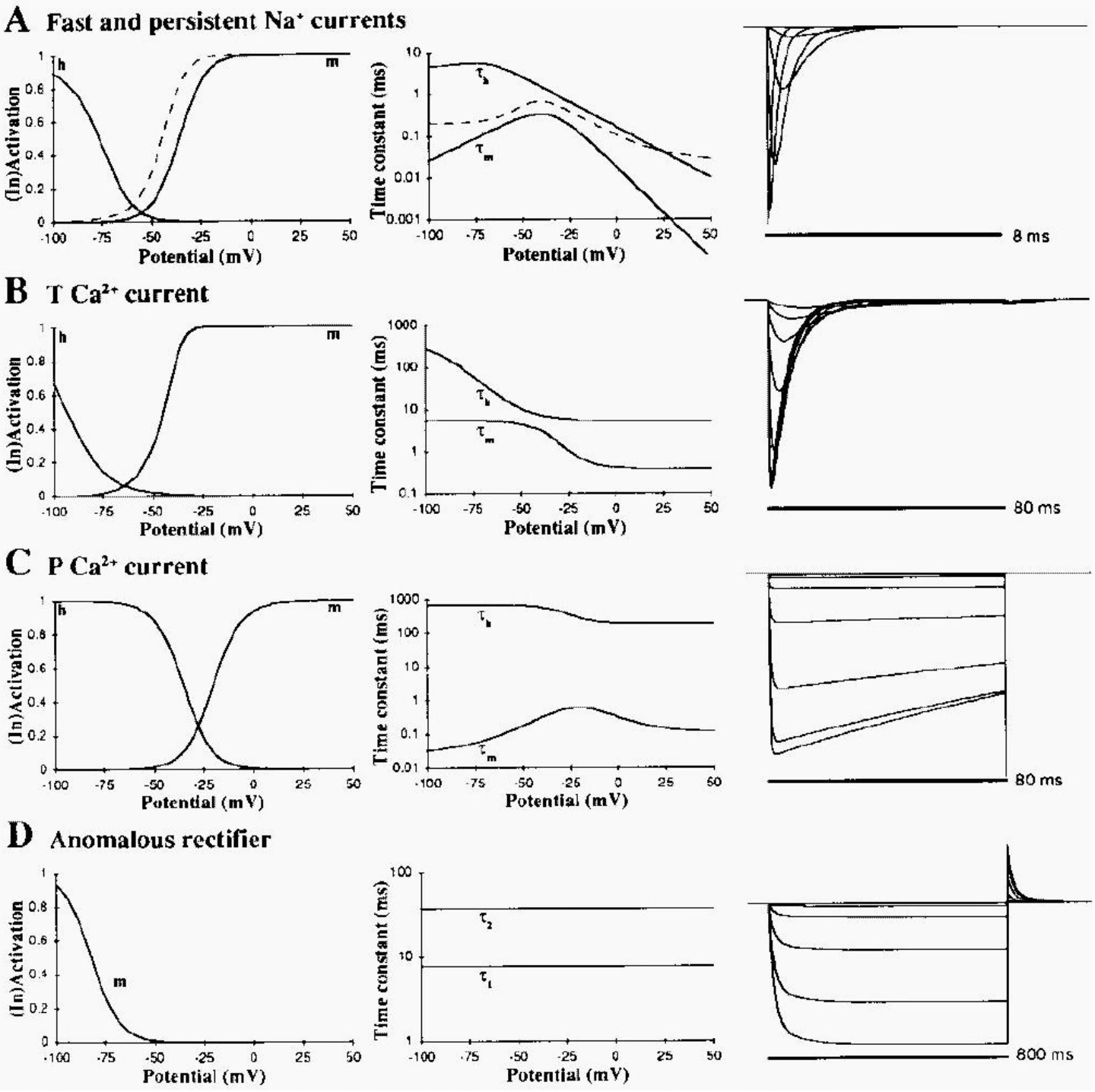
3.14 A 1991 Working Hypothesis for a Channel Fifteen years after Figure 3.6 was drawn, we had learned that there is significant channel mass in the intracellular and extracellular spaces; that there are four α-helical segments with positively charged residues acting as voltage sensors; and that channels are heavily glycosylated and tied to other intracellular proteins. [From Hille 1992.]

From Hille, 3rd edition



$$I_{Ca} = g_{Ca} * (Vm - E_{Ca})$$

$$Vm_{t+\Delta t} = Vm_t + (I_{leak} + I_{chan} + I_{syn}) * \Delta t / CM$$



From

DeSchutter and Bower, 1994

Potential Features of Single Cell Processing

- Spatially inhomogenous processing
  - signal decay
  - signal delay
  - shunting
  - local amplification
  - filtering
- Memory
  - Calcium concentration changes integrative properties
  - Channel phosphorylation as well
  - Time scales: multiple from ms to min to hours/years
- Signal Generation
  - plateau potentials
  - oscillations
  - bursting

Three steps to make a single neuron model

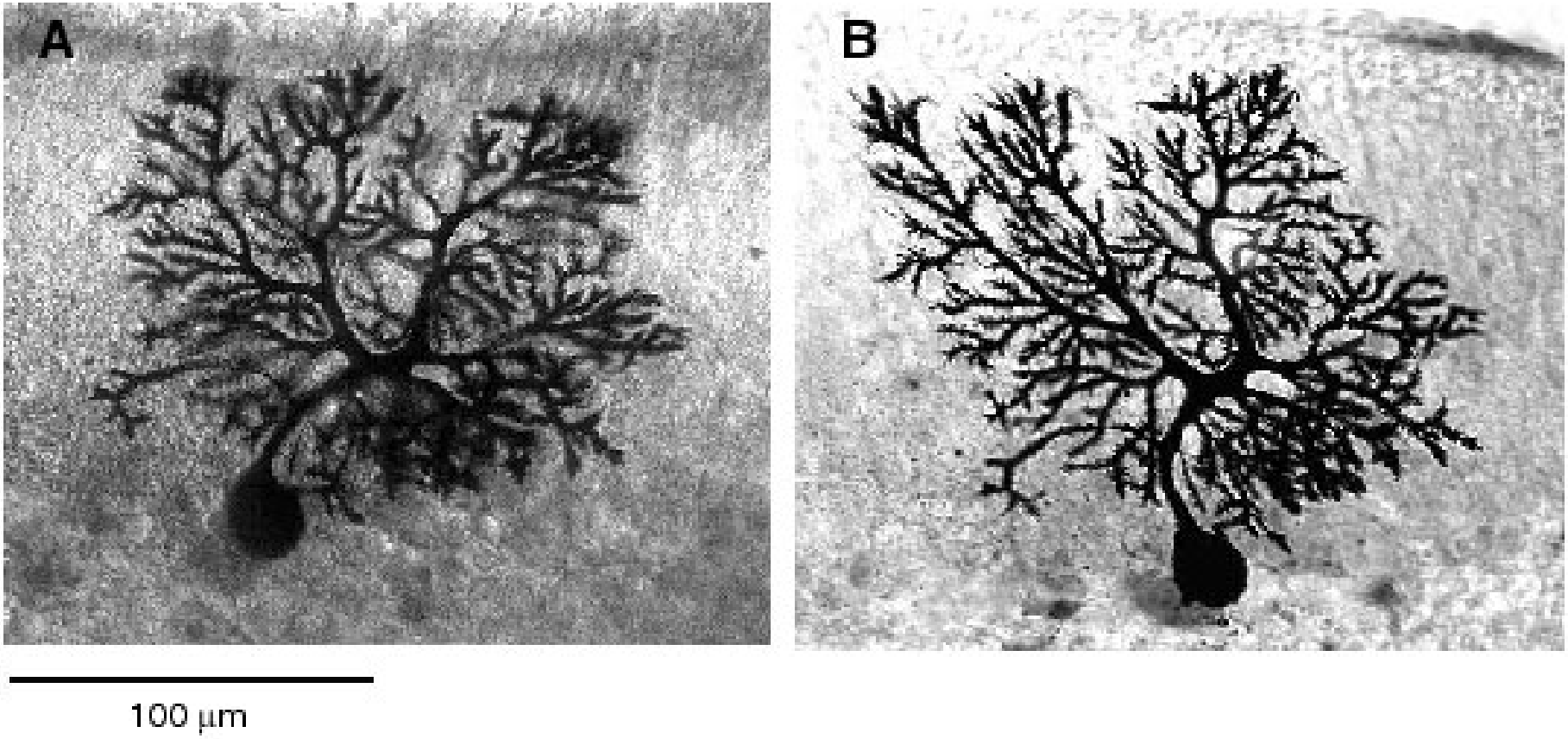
1. **Create an Accurate Morphological Reconstruction**
2. **Create an Accurate Passive Model**
3. **Match active properties with physiological data**

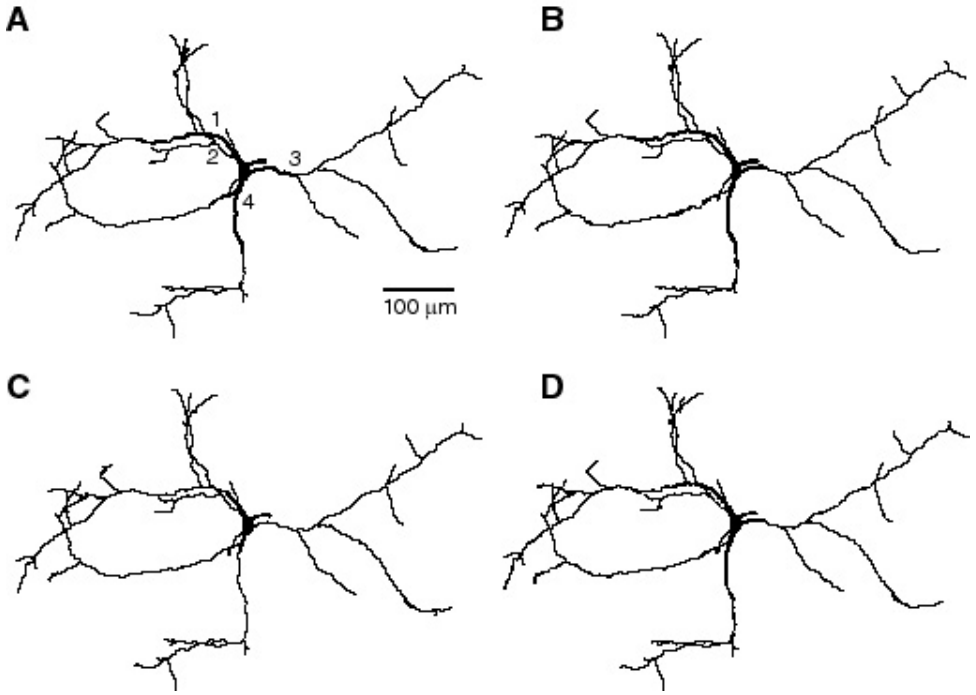
Three steps to make a single neuron model

1. **Create an Accurate Morphological Reconstruction**

1. **Create an Accurate Passive Model**

2. **Match active properties with physiological data**


A
B
100 μm

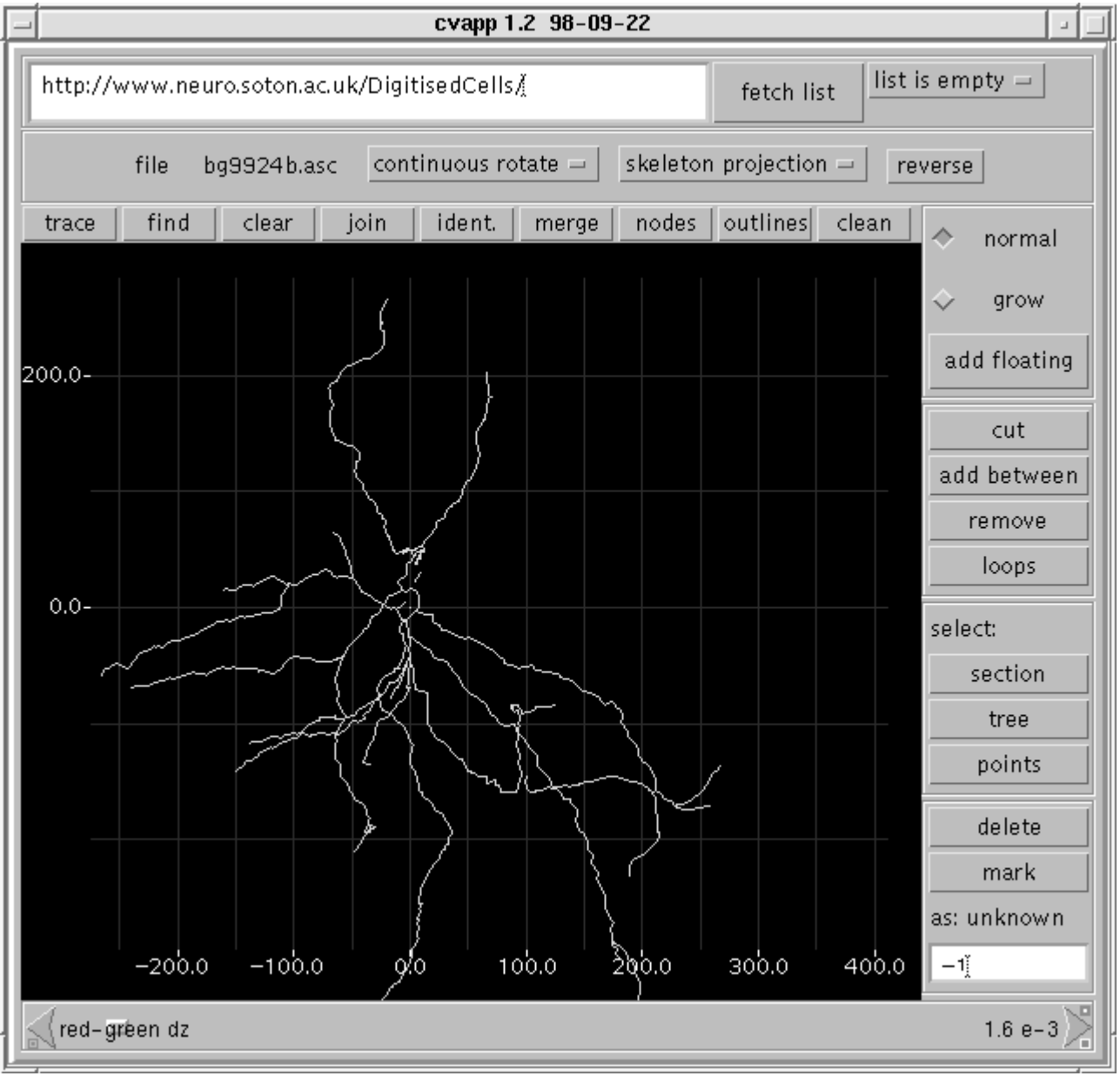


Neurolucida (MicroBrightField, Inc.)

Reconstructions

of GP neuron

	Rec. A	Rec. B	Rec. C	Rec. D
Length Dendrite ١ (µm)	١١٦٠	١٢٢٦.٣	١٣٧٨.١	١٣٠٧.١
Surface Area Dendrite ١ (µm)^{٢}	٤٨٢٩.٠٩	٥٢٣٠.٧٢	٤٩٦٣.٦٢	٤٣٤٤.٦٢
Branch Points Dendrite ١	٨	٨	١١	١٥
Length Dendrite ٢ (µm)	٥٤٠.٦	٥٠٨.٤	٤٨١.٣	٦٥٠.٦
Surface Area Dendrite ٢ (µm)^{٢}	١٨٩٩.٢٣	١٩٣١.٤٧	١٠٥٧.٣٤	٢٣٢١.١٢
Branch Points Dendrite ٢	٤	٣	٥	٦
Length Dendrite ٣ (µm)	١١٠٧.٥	١١٥٨.٤	١١٣٣.١	١٢٦٨.٧
Surface Area Dendrite ٣ (µm)^{٢}	٣٩٨٠.٢٧	٤٢٠٤.٣٦	٢٦٤١.٠٩	٤١٤٧.١٣
Branch Points Dendrite ٣	٧	٧	٨	٩
Length Dendrite ٤ (µm)	١٢٤٩.٦	١٢٥١.٩	١٢٤٢.٧	١٣٨٥
Surface Area Dendrite ٤ (µm)^{٢}	٥٣٥٢.٦٧	٥٥٣٥.٠١	٣٧٢٣.٣٣	٥٠٠٧.٥٣
Branch Points Dendrite ٤	١١	٩	١١	٢٠



CVAPP by

Robert Cannon

```
// Cell morphology file for GENESIS.
// Written by cvapp (http://www.neuro.soton.ac.uk/cells/#software).

*absolute
*asymmetric
*cartesian

// End of cvapp-generated header file.

*origin 0.706  0.756  0

*compt /library/soma
soma  none    0.706  0.756  0  6.067

*compt /library/dendrite
p0[1]  soma  1.28  9.45  0  1.4
p0[2]  p0[1]  2.09  12.56  0  1.4
p0[3]  p0[2]  2.67  15.9  0  1.4
p0b1[0]  p0[3]  10  21.77  0  1.16
p0b1b1[0]  p0b1[0]  20.35  22.12  0  0.93
p0b1b2[0]  p0b1[0]  11.39  29.37  0  0.7
p0b2[0]  p0[3]  -5.93  27.42  0  1.05

p1[1]  soma  22.09  -0.35  0  1.63
p1b1[0]  p1[1]  27.91  9.68  0  1.16
p1b2[0]  p1[1]  35.35  -7.83  0  0.93
```

Three steps to make a single neuron model

1. Create an Accurate Morphological Reconstruction

1. **Create an Accurate Passive Model**

1. Match active properties with physiological data

Task: Set Rm, Cm, and Ri correctly for each compartment

If assumption that parameters are uniform holds, need to fit 3 values.

Strategy: Obtain data from recordings, then optimize Rm, Cm, Ri to fit data.

Message: optimize recordings to get linear (passive) response.

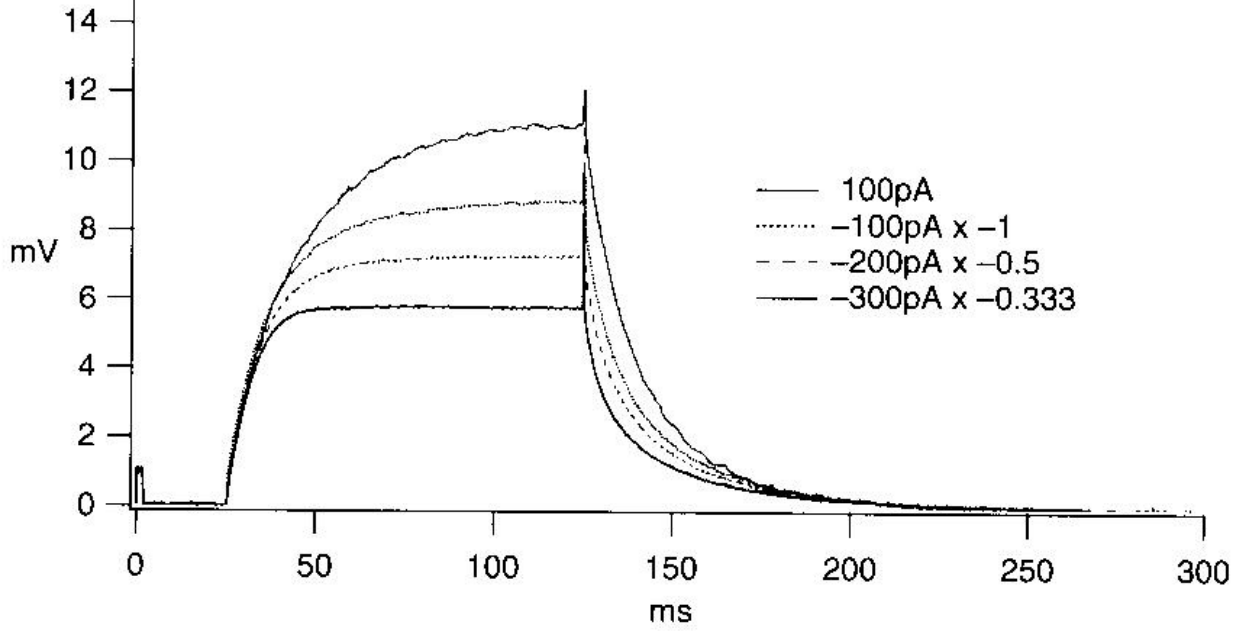
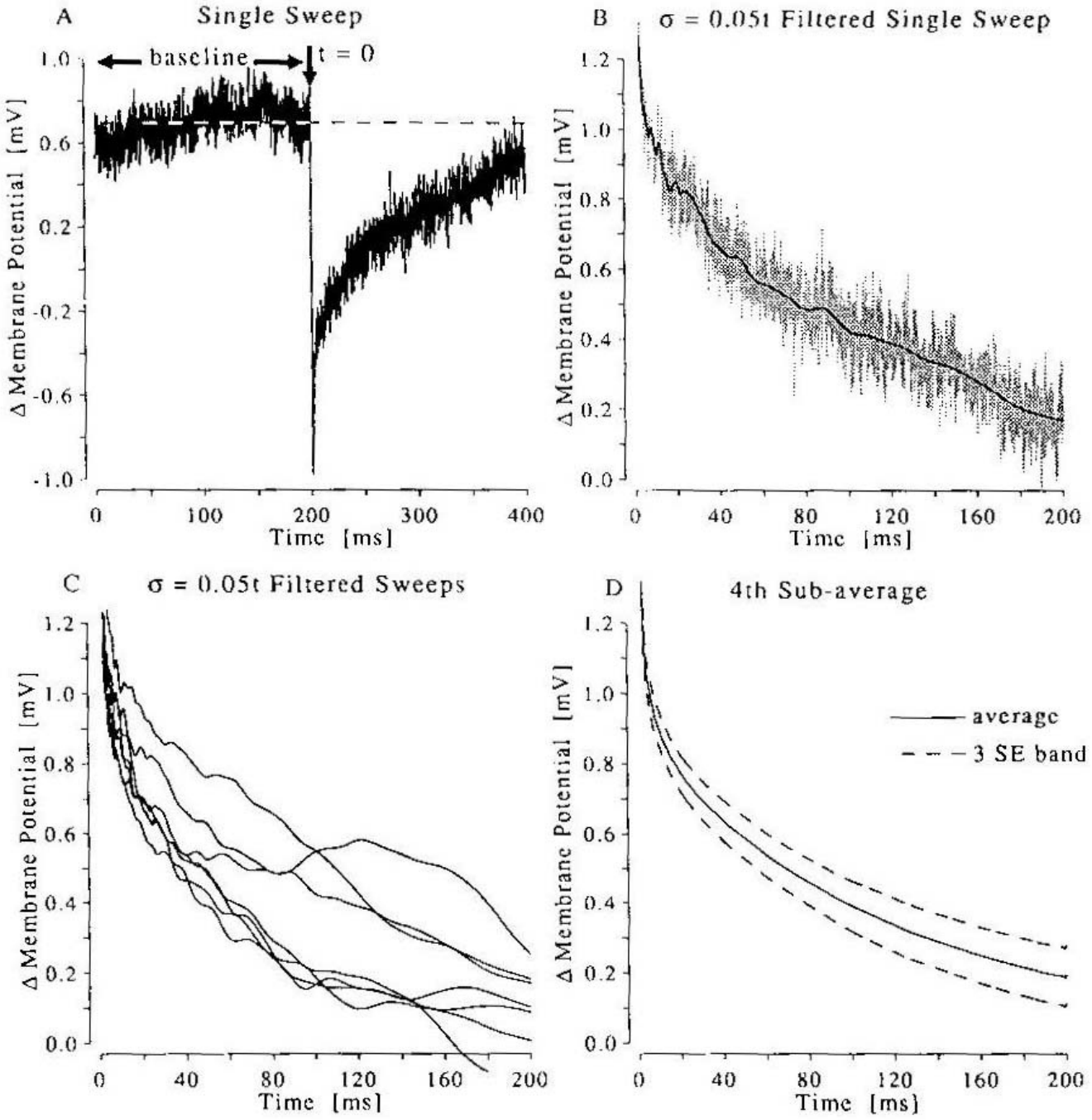


FIGURE 8.6 Long pulse responses in current clamp from same spiny stellate cell as in Color Figure 8.2, scaled by 1/current. Notice how responses show strong inward rectification: time constants and amplitudes increase at more depolarized potentials.

Potential Problems:

- Incomplete Block of Active Properties
- Partial Block of Leak Channels
- Electrode Serial Resistance
- Electrode Shunt



Message:

Short current injection pulses have several advantages, obtain such measurements!

From Major, 1994

Fit Data:

Start with reasonable default parameters, and provide upper and lower bounds. Use automated fitting routine, e.g. with Matlab

Rm: 1 ohm * m^2 (limits: 0.5 …. 10)

Ri: 1 ohm * m (limits: 0.5 .. 3)

Cm: 0.01 F / m^2 (limits: 0.005 … 0.02)

If fitting of uniform parameters fails, try non-uniform ones. 1) Special case for soma 2) Special case for thin dendrites. 3) Possibly try corrections to morphology

Three steps to make a single neuron model

1. Create an Accurate Morphological Reconstruction
2. Create an Accurate Passive Model
1. **Match active properties with physiological data**

Tasks:

1) Construct set voltage-gated conductances with correct kinetics.

2) Incorporate intracellular calcium handling

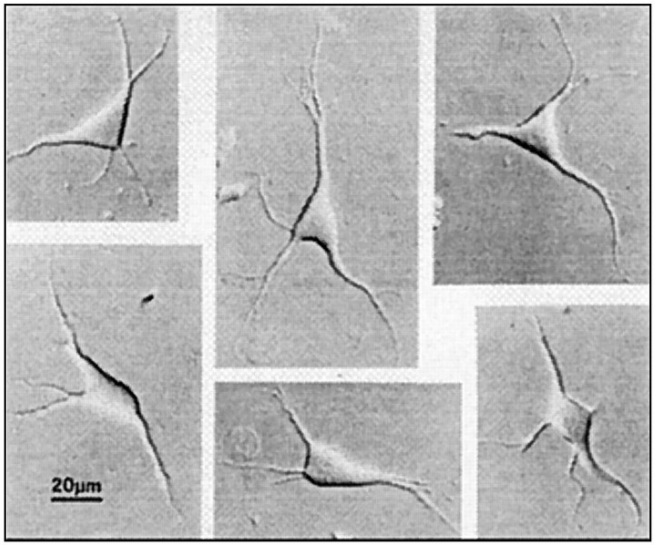
3) Find needed densities of channels in each compartment

4) Add realistic synaptic input conductances

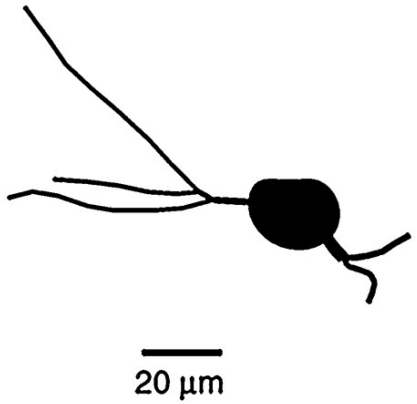
Dissociated cells

Model

A1

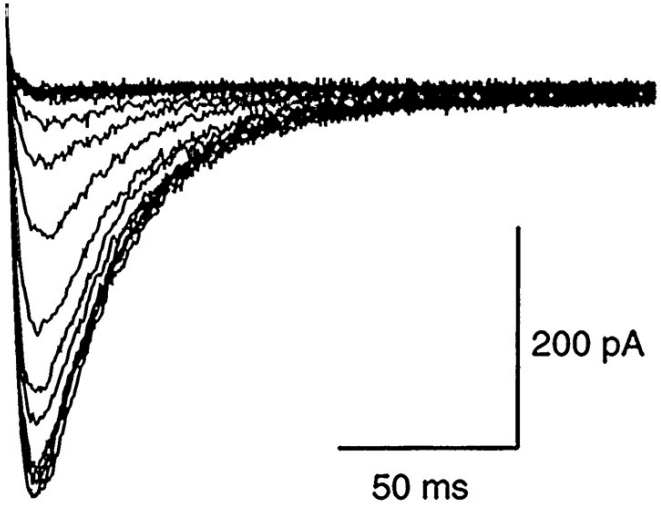


B1

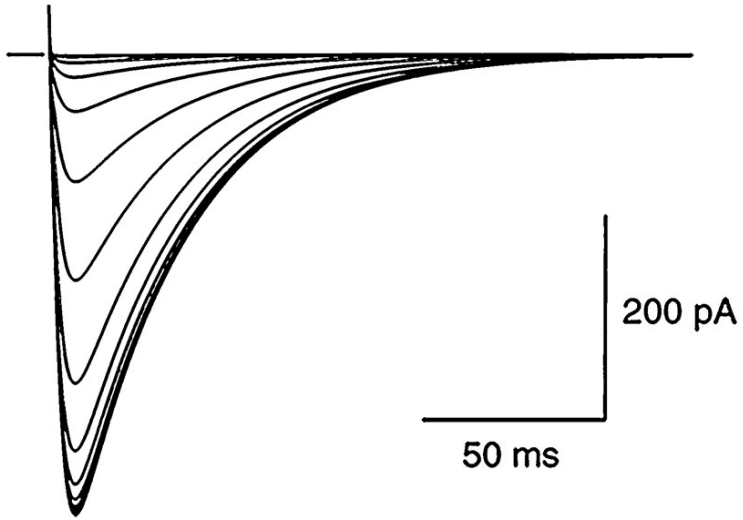


From Destexhe et al., J. Neurosci. (1998). 18:3574-3588

A2



B2



$$I_T = \bar{P}_{Ca}\, m^2 h\, G(V, Ca_o, Ca_i)$$

$$\dot{m} = -\frac{1}{\tau_m(V)}(m - m_\infty(V)) \qquad (1)$$

$$\dot{h} = -\frac{1}{\tau_h(V)}(h - h_\infty(V)),$$

where $\bar{P}_{Ca}$ (in centimeters per second) is the maximum permeability of the membrane to Ca^{2+} ions, and m and h are, respectively, the activation and inactivation variables. $G(V,Ca_o,Ca_i)$ is a nonlinear function of voltage and ionic concentrations:

$$G(V, Ca_o, Ca_i) = Z^2F^2V/RT\,\frac{Ca_i - Ca_o \exp(-ZFV/RT)}{1 - \exp(-ZFV/RT)}, \qquad (2)$$

where $Z = 2$ is the valence of calcium ions, F is the Faraday constant, R is the gas constant, and T is the temperature in Kelvins. Ca_i and Ca_o are the intracellular and extracellular Ca^{2+} molar concentrations, respectively.

From Destexhe et al., 1998

current-clamp simulations of TC cells in the present paper. The optimal functions that accounted for both voltage-clamp and current-clamp data on TC cells were:

$$m_\infty(V) = 1/(1 + \exp[-(V + 56)/6.2])$$

$$h_\infty(V) = 1/(1 + \exp[(V + 80)/4]).$$

The voltage-dependent time constant for activation was:

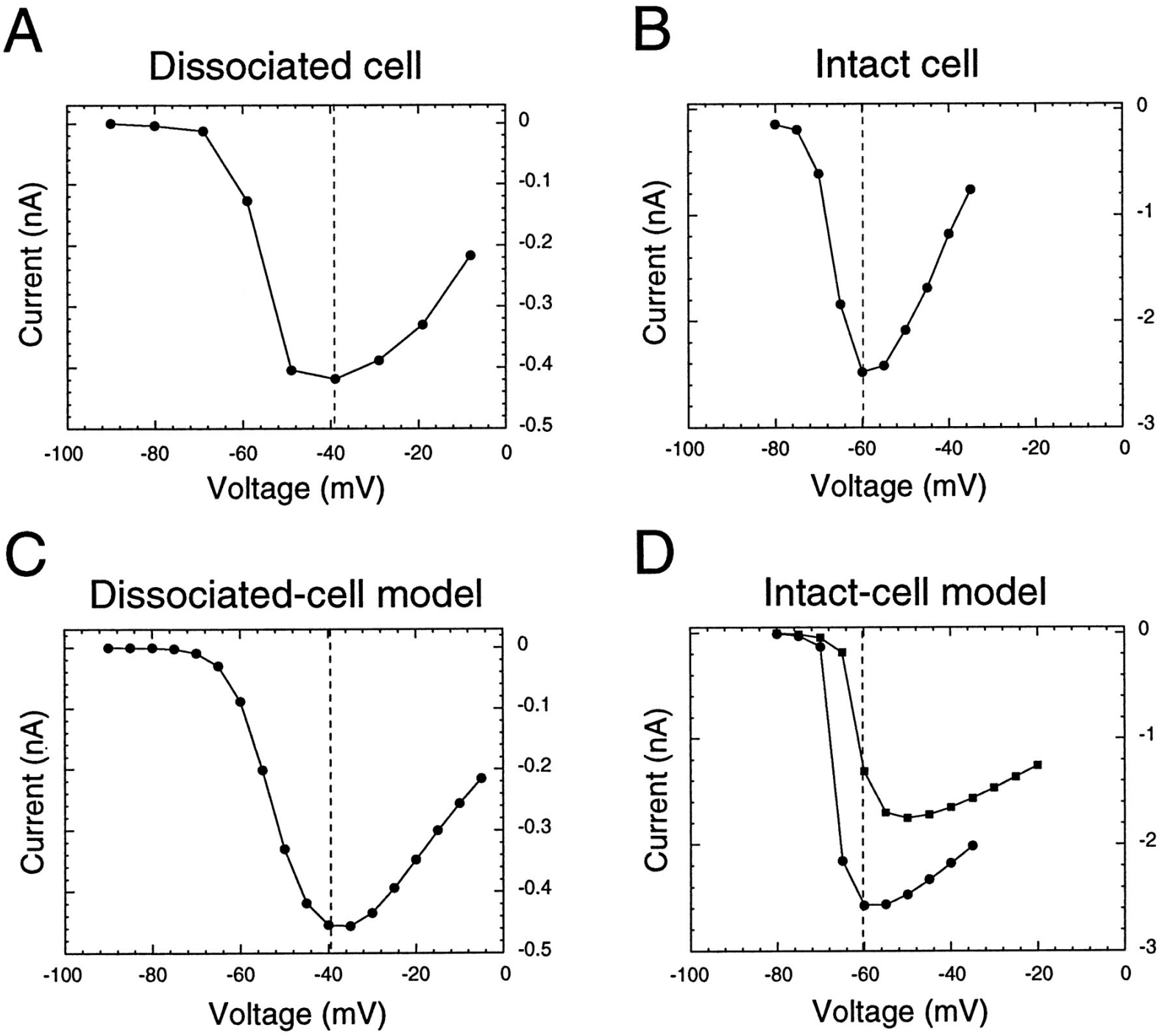
$$\tau_m(V) = 0.204 + 0.333/(\exp[-(V + 131)/16.7] + \exp[(V + 15.8)/18.2]), \quad (3)$$

and for inactivation:

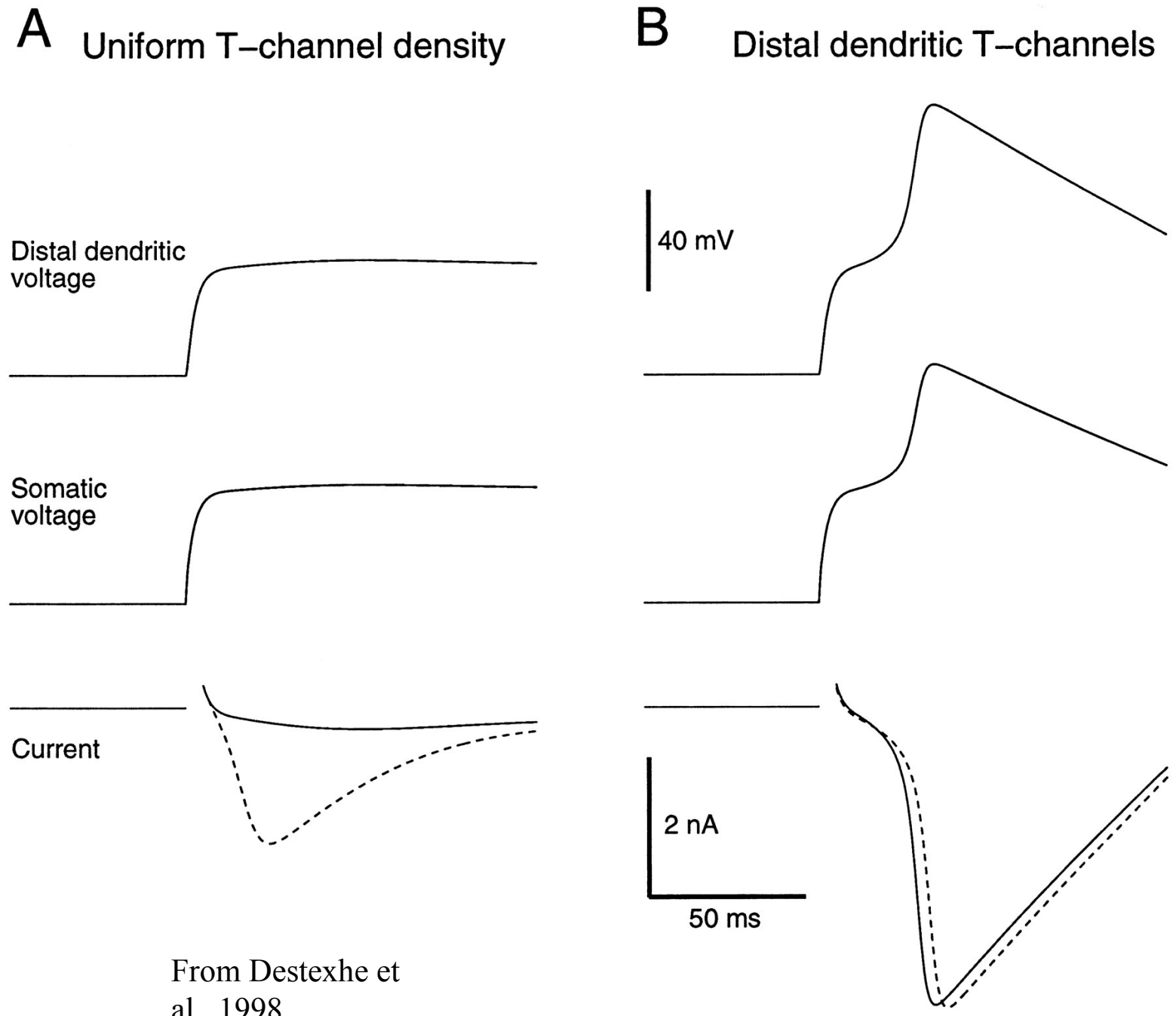
$$\tau_h(V) = \begin{cases} 0.333 \exp[(V + 466)/66.6] & \text{for } V < -81 \text{ mV} \\ 9.32 + 0.333 \exp[-(V + 21)/10.5] & \text{for } V > -81 \text{ mV}. \end{cases} \quad (4)$$

These functions correspond to an external Ca^{2+} concentration of 2 mM and a temperature of 36°C. All voltage-clamp simulations were done at 24°C assuming Q_{10} values of 2.5 for both m and h, whereas current-clamp behavior was simulated at 34°C.

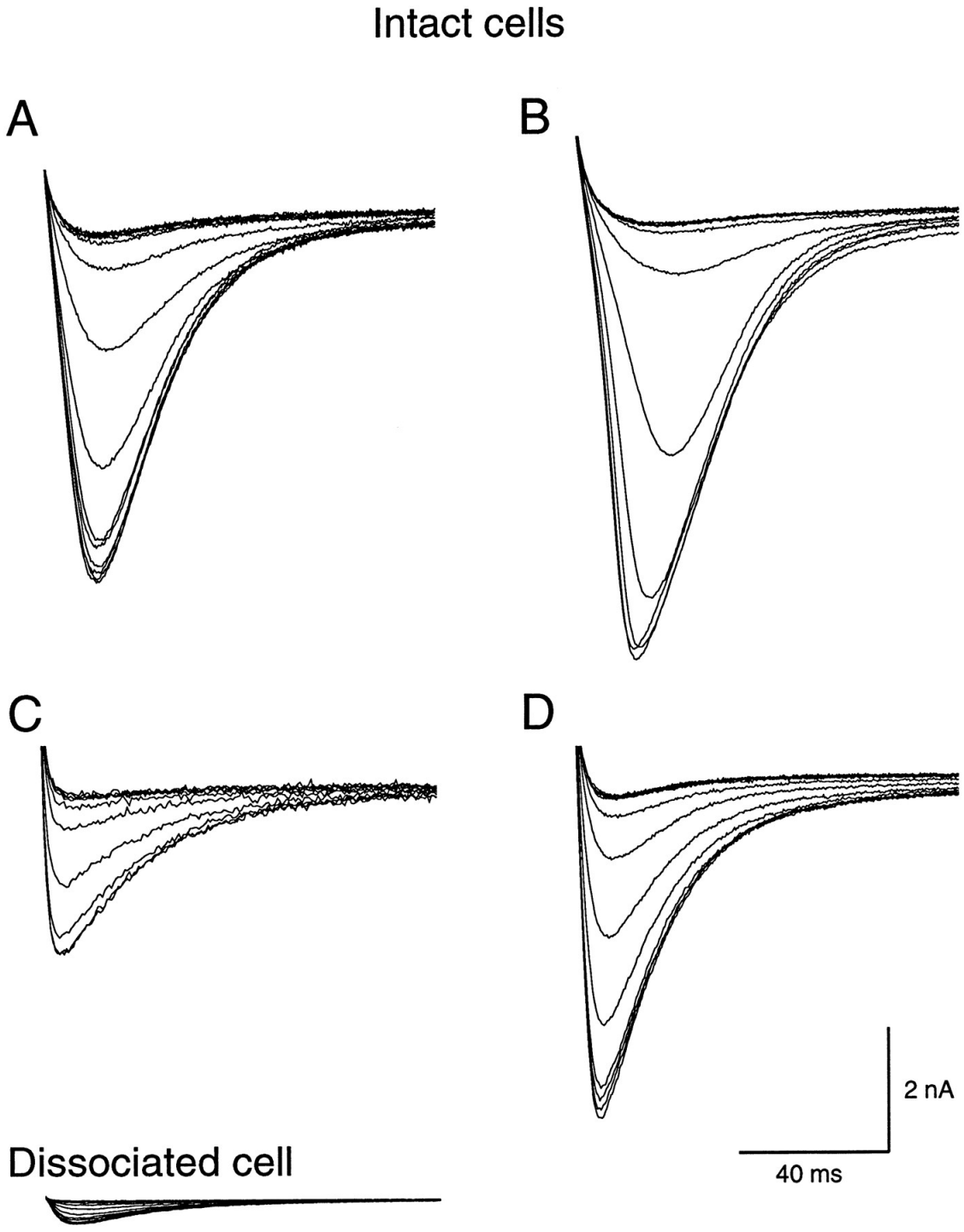
From Destexhe et al., 1998



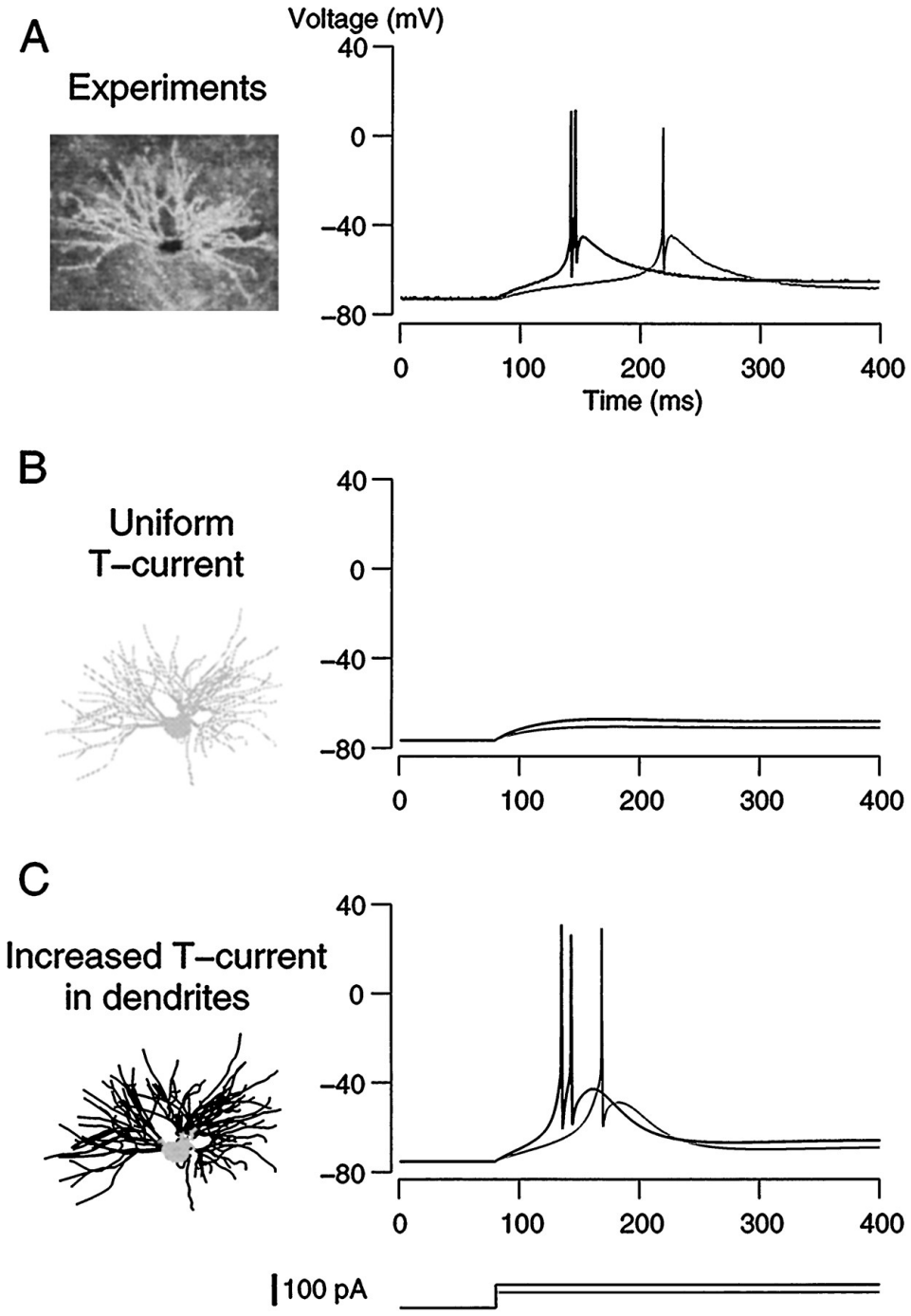
From Destexhe et al., 1998



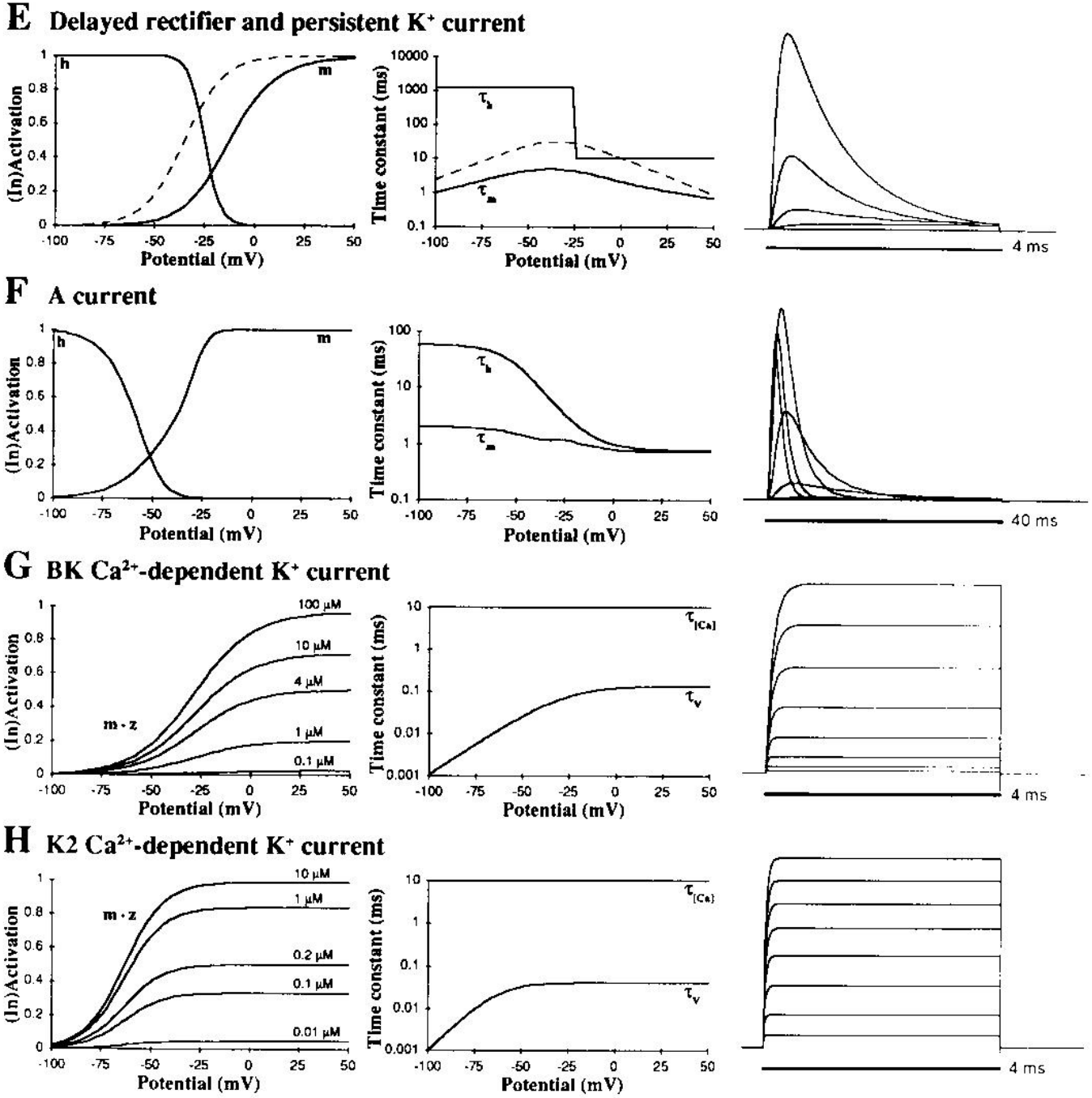
From Destexhe et al., 1998



From Destexhe et al., 1998



From Destexhe et al., 1998



From DeSchutter and Bower, 1994

FIG. 2. (*continued*)

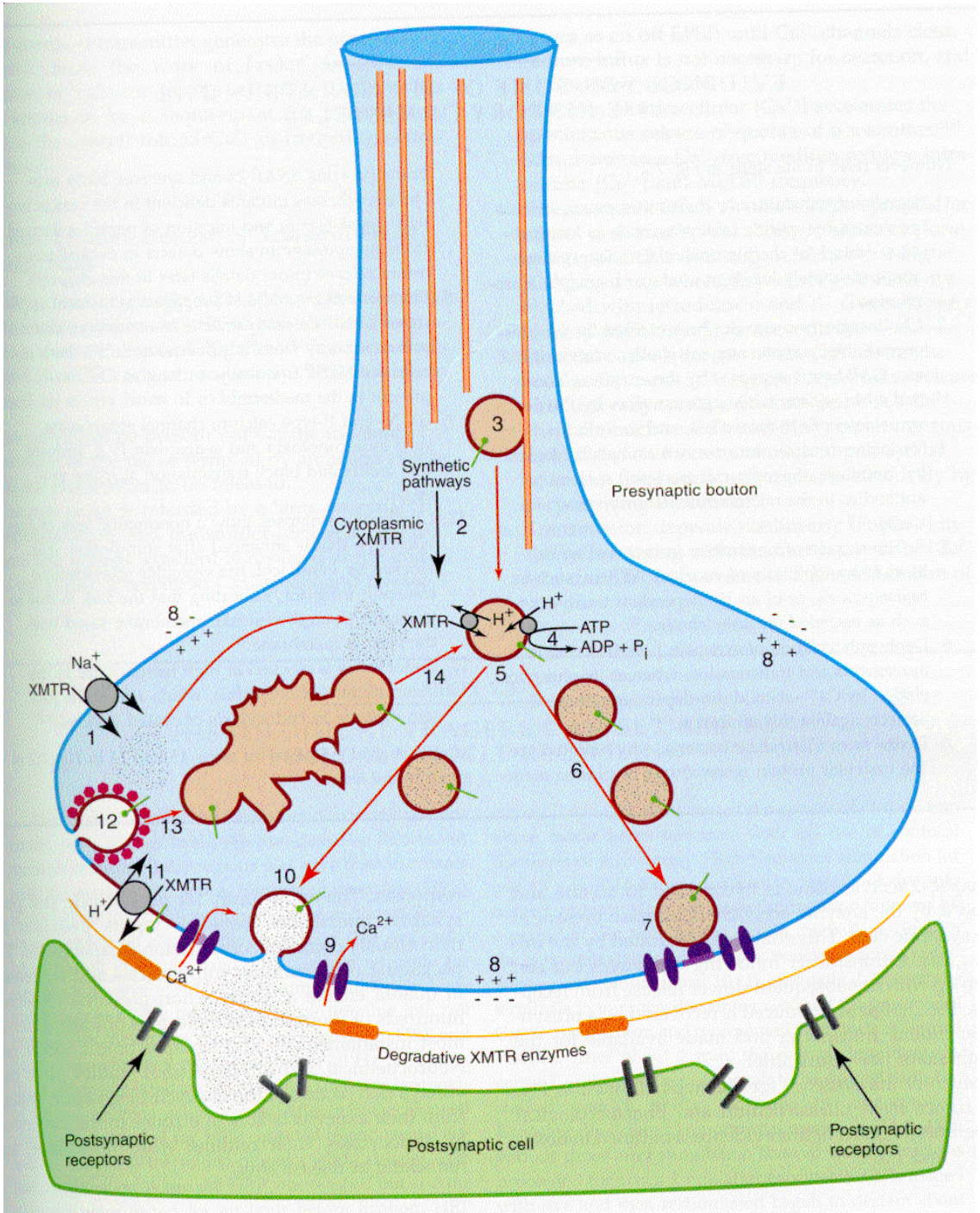
Calcium Handling

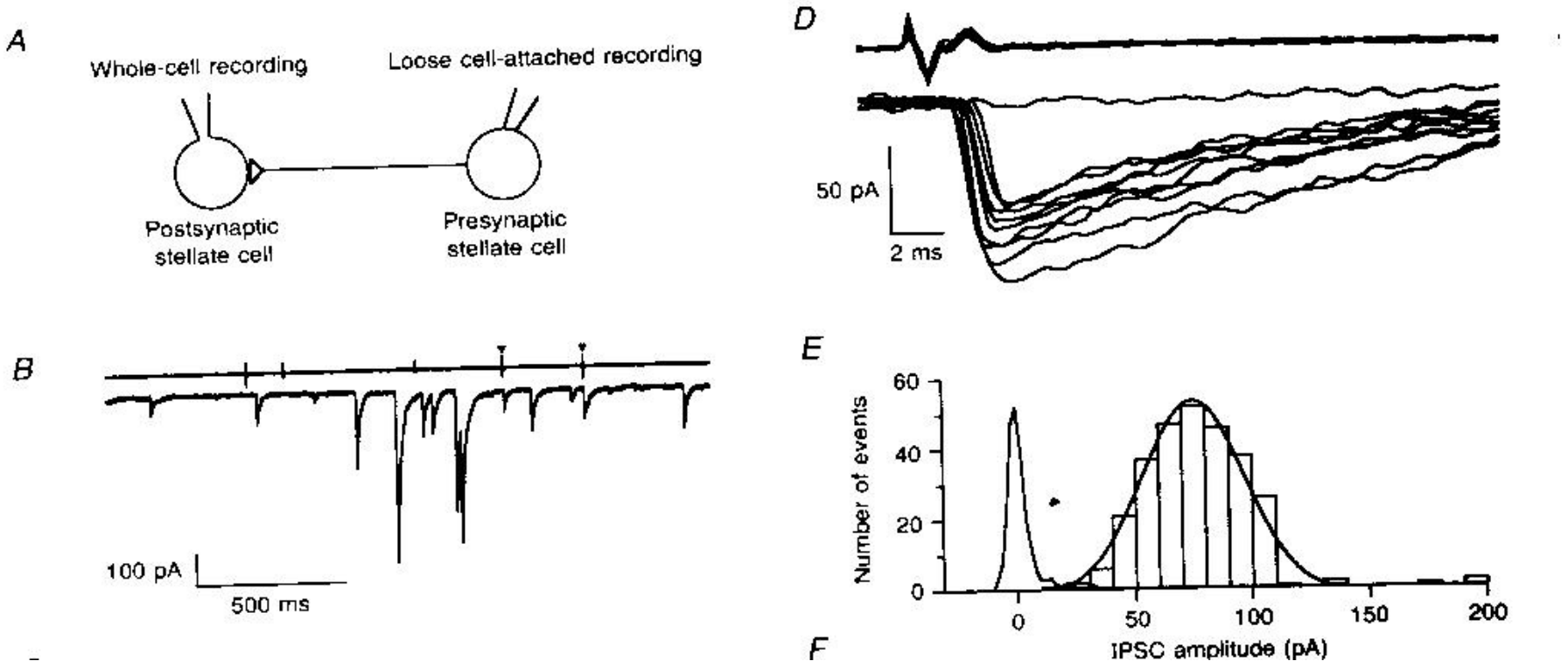
- **A Single diffusion shell is a simple solution that works reasonably well.**

In this method, a thin (e.g. 0.2 micron) shell of volume below the membrane is taken as a reservoir for calcium with an influx proportional to membrane calcium currents, and an exponential decay to baseline. The baseline concentration is taken from physiological data (e.g. 40 nM).

- **More complex solutions include binding to buffers, and intracellular stores. However, it is nearly impossible to get correct kinetic data for these processes.**

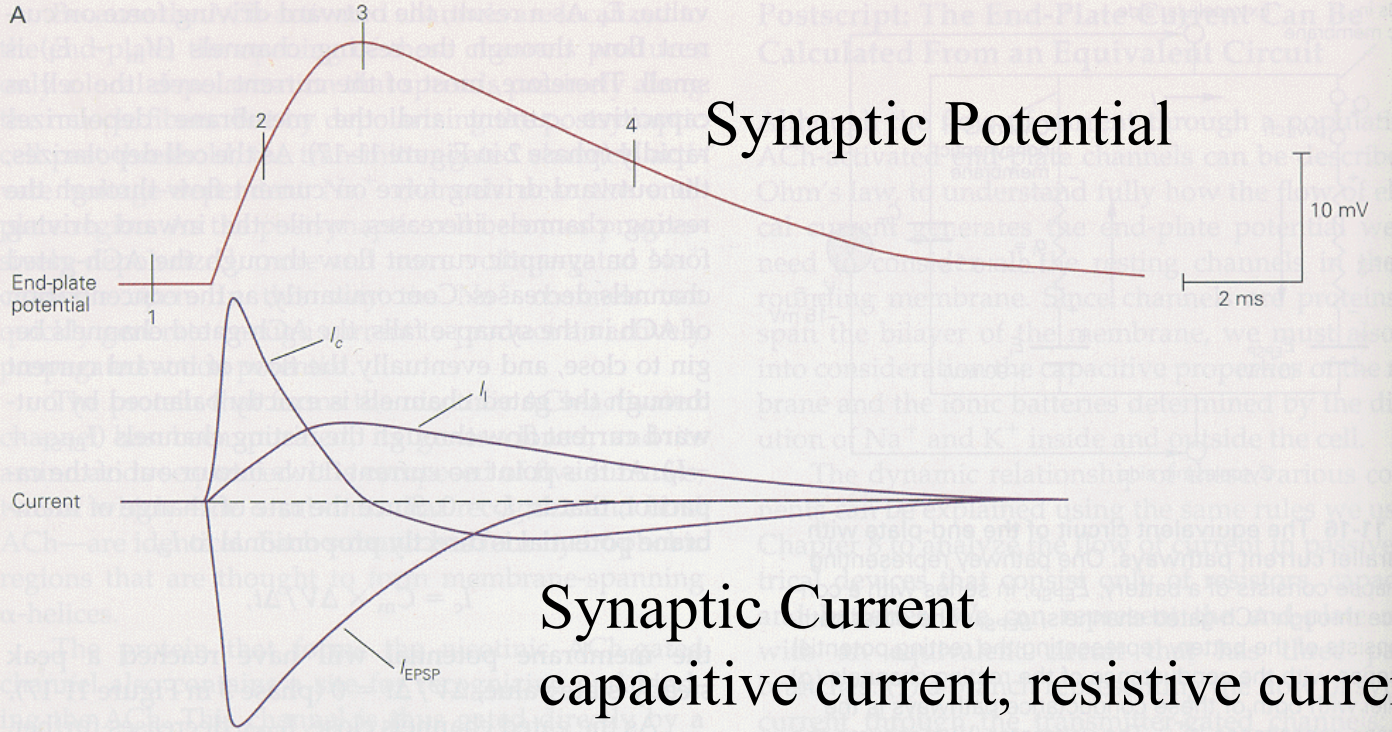
Finally: Synapses!





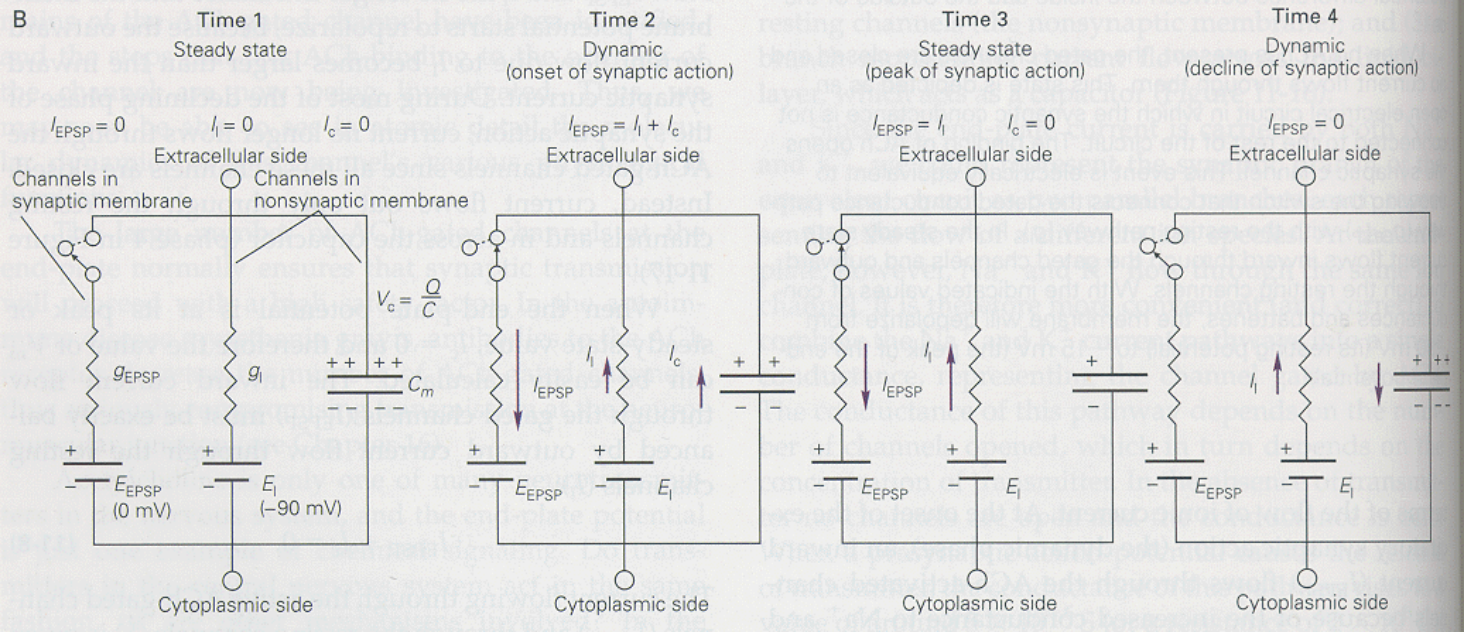
From Kondo and Marty, 1998

$$g_{syn}(t) = \frac{A g_{max}}{\tau_1 - \tau_2}(e^{-t/\tau_1} - e^{-t/\tau_2}), \text{ for } \tau_1 > \tau_2,$$



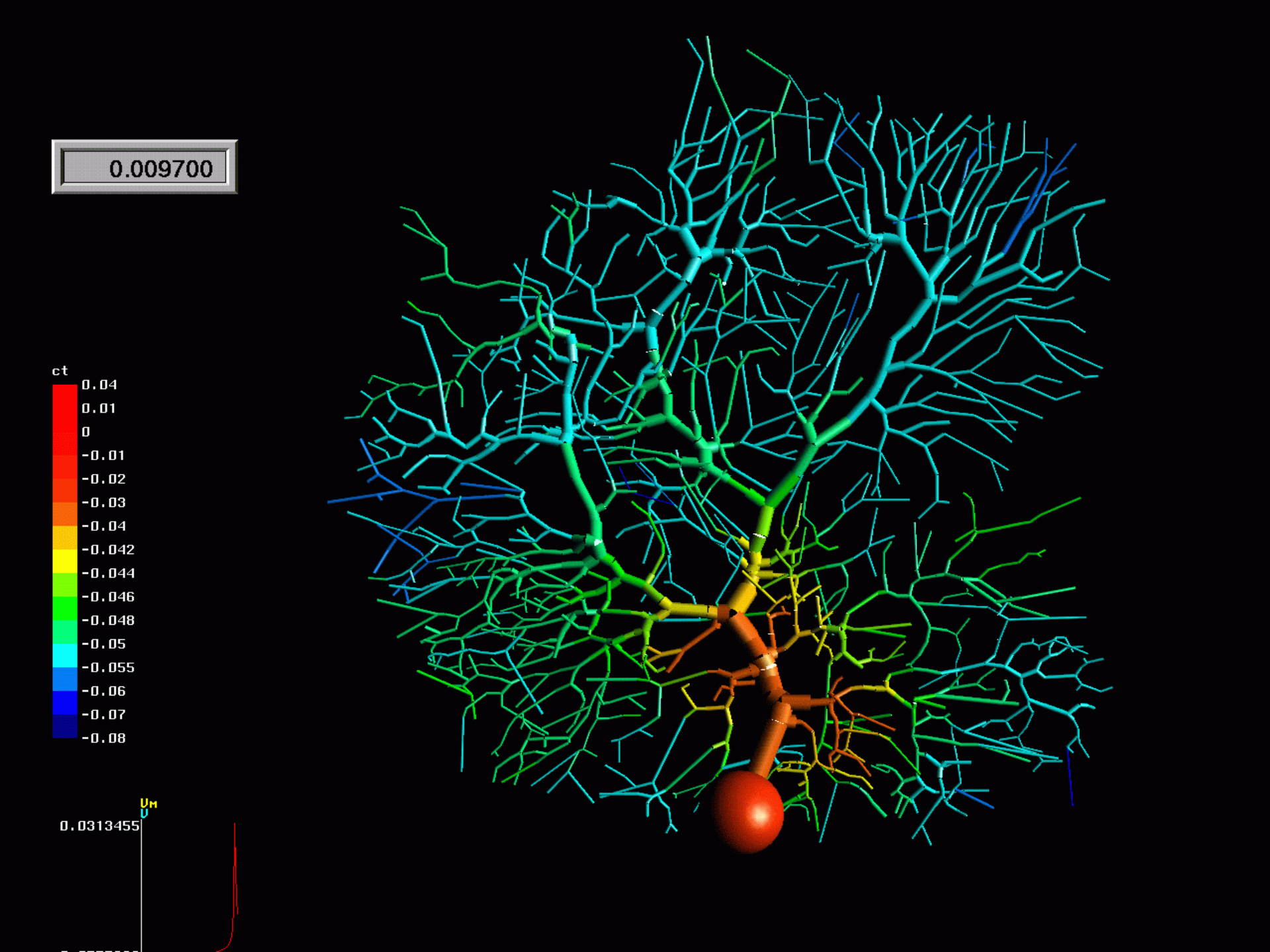
Synaptic Potential

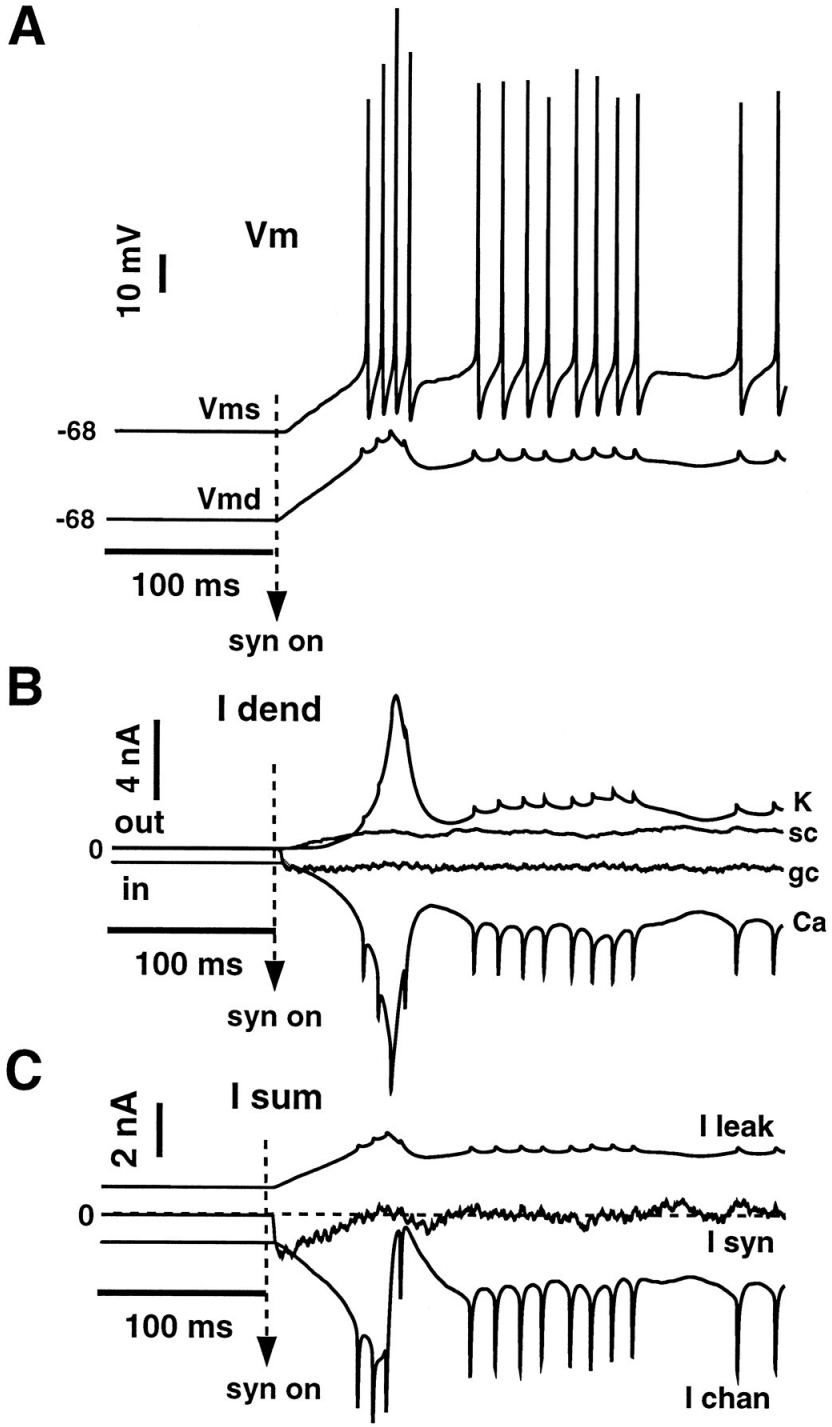
Synaptic Current
capacitive current, resistive current

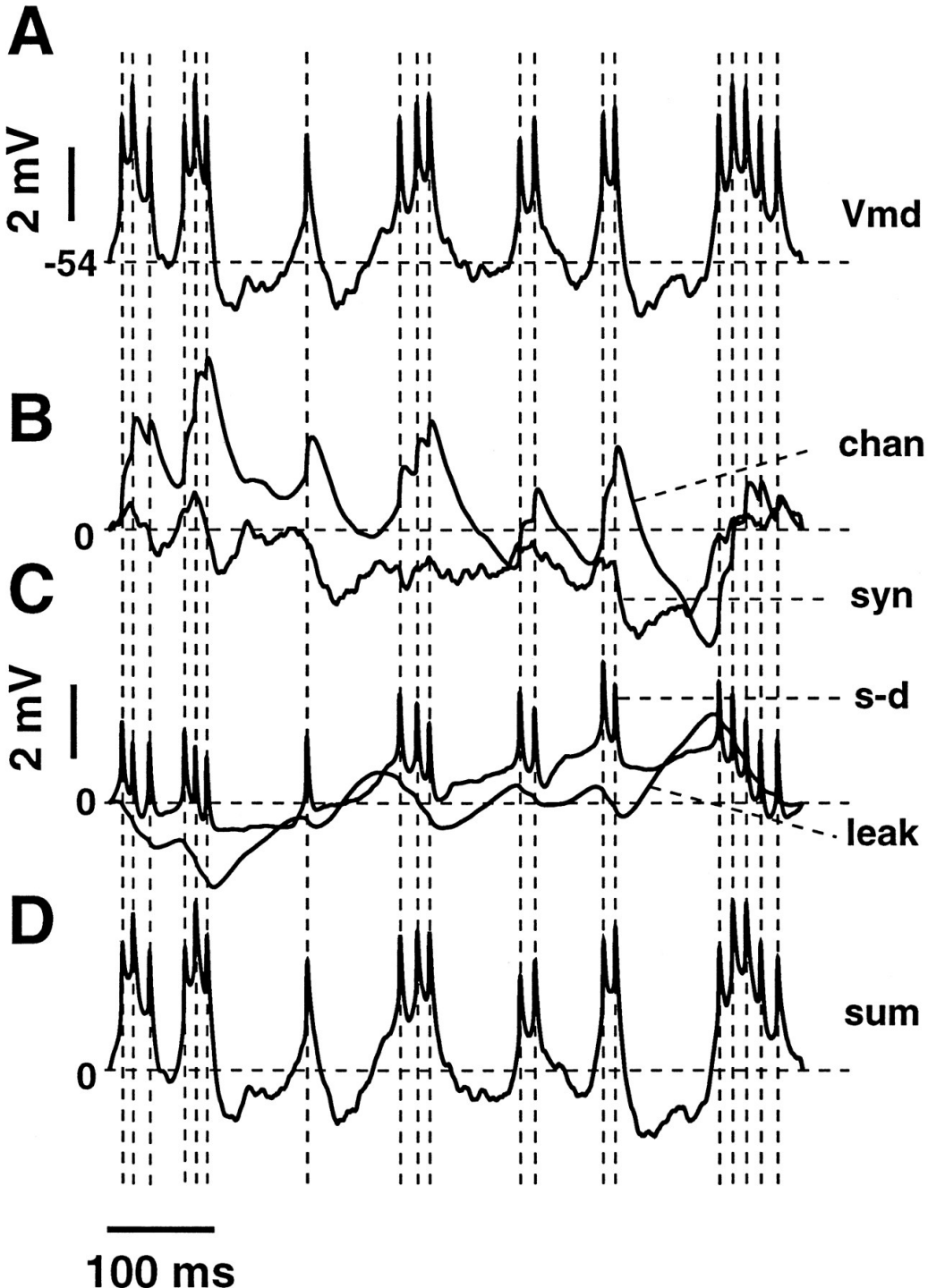


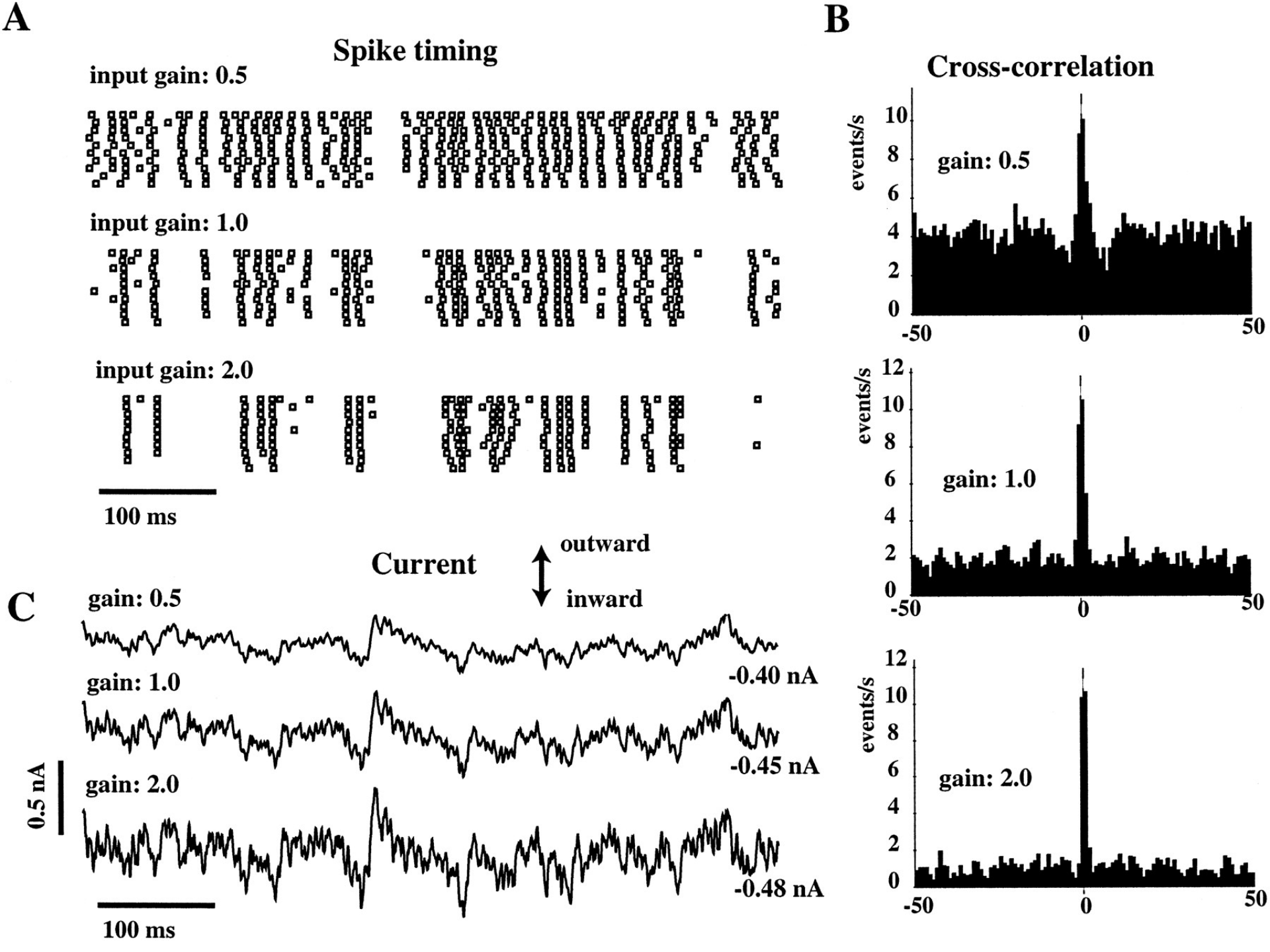
Potential Shortcomings of simplified synaptic modeling:

- Missing all or none release probability
- Missing minis
- Missing short-term plasticity
    - change in release probability, e.g. due to residual Ca
    - extrasynaptic receptor activation
    - presynaptic receptor activation

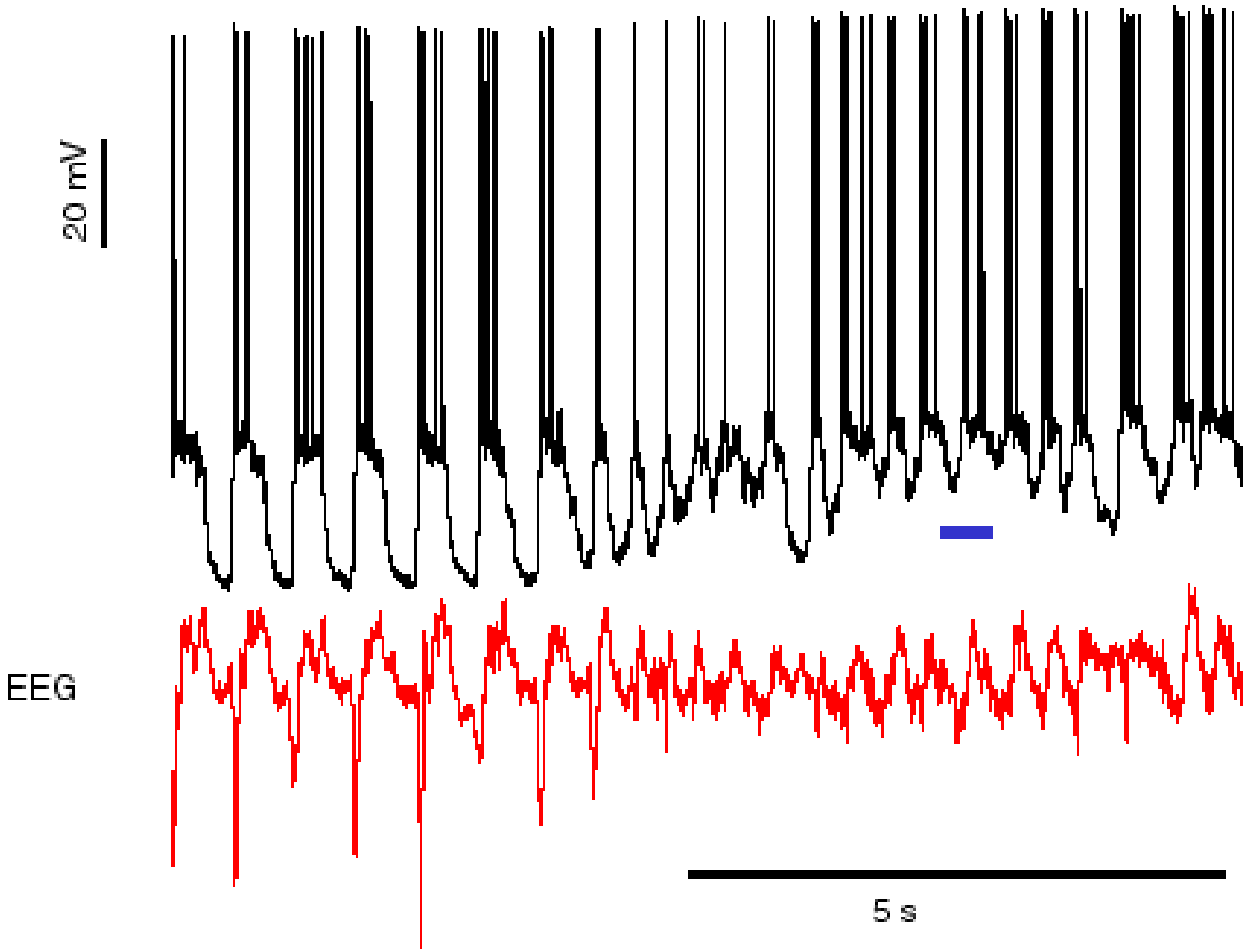

0.009700
ct
0.04
0.01
0
-0.01
-0.02
-0.03
-0.04
-0.042
-0.044
-0.046
-0.048
-0.05
-0.055
-0.06
-0.07
-0.08
Vm
0.0313455


A
Vm
10 mV
Vms
-68
Vmd
-68
100 ms
syn on
B
I dend
4 nA
out
0
in
K
sc
gc
Ca
100 ms
syn on
C
I sum
2 nA
I leak
0
I syn
100 ms
syn on
I chan

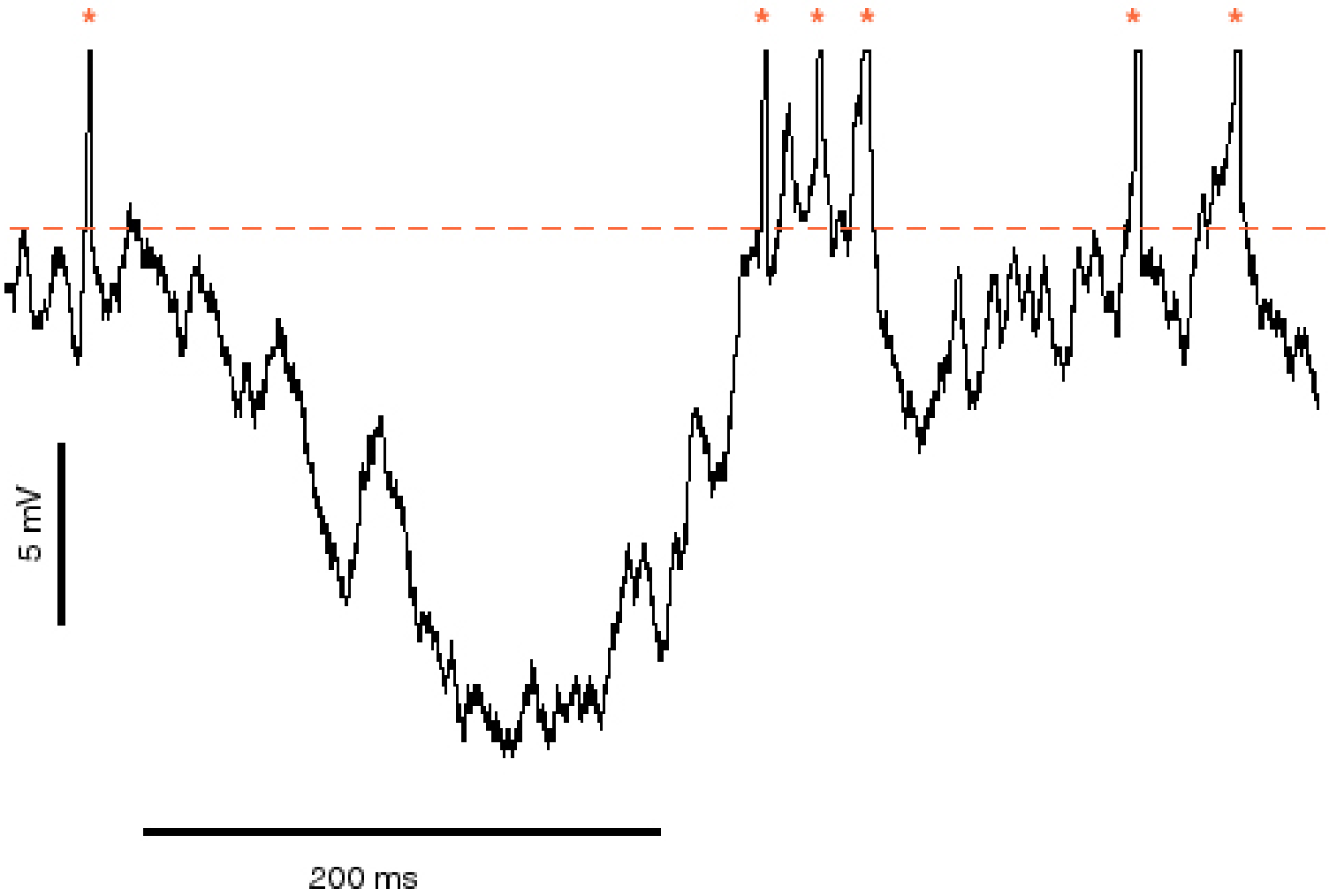

A
2 mV
-54
Vmd
B
chan
0
syn
C
2 mV
s-d
0
leak
D
sum
0
100 ms


A
Spike timing
input gain: 0.5
input gain: 1.0
input gain: 2.0
100 ms
B
Cross-correlation
events/s
gain: 0.5
gain: 1.0
gain: 2.0
-50
0
50
C
Current
outward
inward
gain: 0.5
gain: 1.0
gain: 2.0
-0.40 nA
-0.45 nA
-0.48 nA
0.5 nA
100 ms

In vivo recording from striatal medium spiny neuron



In vivo recording from striatal medium spiny neuron



The big picture

- Realistic single cell models can be constructed, but it is a lot of work
- Even the best model is still not going to completely replicate biological single cell dynamics
- Each type of neuron is a separate modeling job
- Insight into the processing capabilities of neurons can be gained from realistic single cell modeling
- Experimental predictions can be made and tested
- Insights gained are invaluable for realistic network modeling
- Simplified single cell models are likely required for large network models

Vital References

- Johnston and Wu. Foundations of Cellular Neurophysiology, MIT Press, 1995
- Bertil Hille. Ion Channels of Excitable Membranes. Third Edition. Sinauer, 2001
- Erik De Schutter (editor) Computational Neuroscience: Realistic Modeling for Experimentalists. CRC Press, 2001
- Koch and Segev. Methods in Neuronal Modeling. From Ions to Networks. Second Edition. MIT Press, 1998

- For Beginners in Neuroscience: Kandel, Schwarz, Jessell. Principles of Neural Science, Fourth Edition, McGraw Hill, 2000

A brief primer on how to model complex single cells with Genesis

1. Create morphology file
2. Create channel and compartment prototypes
3. Create main simulation script
4. Create parameter search scheme to tune model
5. Include synapses
6. Enjoy

<genhome>/Scripts/Purkinje/Purk_chansave.g

```
/* Make tabchannels */
function make_chan(chan, Ec, Gc, Xp, XAA, XAB, XAC, XAD, XAF, XBA, XBB,  \
   XBC, XBD, XBF, Yp, YAA, YAB, YAC, YAD, YAF, YBA, YBB, YBC, YBD, YBF \
   , Zp, ZAA, ZAB, ZAC, ZAD, ZAF, ZBA, ZBB, ZBC, ZBD, ZBF)
   str chan
   int Xp, Yp, Zp
   float Ec, Gc
   float XAA, XAB, XAC, XAD, XAF, XBA, XBB, XBC, XBD, XBF
   float YAA, YAB, YAC, YAD, YAF, YBA, YBB, YBC, YBD, YBF
   float ZAA, ZAB, ZAC, ZAD, ZAF, ZBA, ZBB, ZBC, ZBD, ZBF
  if ({exists {chan}})
      return
   end
   create tabchannel {chan}
   setfield {chan} Ek {Ec} Gbar {Gc} Ik 0 Gk 0 Xpower {Xp} Ypower {Yp}  \
      Zpower {Zp}
   if (Xp > 0)
      setupalpha {chan} X {XAA} {XAB} {XAC} {XAD} {XAF} {XBA} {XBB} {XBC} \
                  {XBD} {XBF} -size {tab_xfills+1} -range {tab_xmin} {tab_xmax}
   end
   if (Yp > 0)
      setupalpha {chan} Y {YAA} {YAB} {YAC} {YAD} {YAF} {YBA} {YBB} {YBC} \
                  {YBD} {YBF} -size {tab_xfills+1} -range {tab_xmin} {tab_xmax}
      end
   if (Zp != 0)
      setupalpha {chan} Z {ZAA} {ZAB} {ZAC} {ZAD} {ZAF} {ZBA} {ZBB} {ZBC} \
                  {ZBD} {ZBF} -size {tab_xfills+1} -range {tab_xmin} {tab_xmax}
   end
end
```

<genhome>/Scripts/Purkinje/Purk_chansave.g

```
/* non-inactivating (muscarinic) type K current, eq#2
   ** eq#2: corrected typo in Yamada equation for tau: 20 instead of 40
   ** Refs: Yamada (his equations are also for 22C) */
   if (!{exists Purk_KM})
        create tabchannel Purk_KM
        setfield Purk_KM Ek {EK} Gbar {GK} Ik 0 Gk 0 Xpower 1  \
           Ypower 0 Zpower 0

        call Purk_KM TABCREATE X {tab_xdivs} {tab_xmin} {tab_xmax}
           x = {tab_xmin}
           dx = ({tab_xmax} - {tab_xmin})/{tab_xdivs}

        for (i = 0; i <= ({tab_xdivs}); i = i + 1)
             y = 0.2/(3.3*({exp {(x + 0.035)/0.02}}) + {exp {-(x + 0.035)/0.02}})
             setfield Purk_KM X_A->table[{i}] {y}

             y = 1.0/(1.0 + {exp {-(x + 0.035)/0.01}})
             setfield Purk_KM X_B->table[{i}] {y}
             x = x + dx
        end
        tau_tweak_tabchan Purk_KM X
        setfield Purk_KM X_A->calc_mode 0 X_B->calc_mode 0
        call Purk_KM TABFILL X {tab_xfills + 1} 0
   end
   call Purk_KM TABSAVE Purk_KM.tab
```

<genhome>/Scripts/Purkinje/Purk_const.g

```
* cable parameters */
float CM = 0.0164              // *9 relative to passive
float RMs = 1.000              // /3.7 relative to passive comp
float RMd = 3.0
float RA = 2.50
/* preset constants */
float ELEAK = -0.0800          // Ek value used for the leak
conductance
float EREST_ACT = -0.0680      // Vm value used for the RESET
/* concentrations */
float CCaO = 2.4000            //external Ca as in normal slice Ringer
float CCaI = 0.000040          //internal Ca in mM
//diameter of Ca_shells
float Shell_thick = 0.20e-6
float CaTau = 0.00010          // Ca_concen tau
```

```
/* Currents: Reversal potentials in V and max
conductances S/m^2 */
/* Codes: s=soma, m=main dendrite, t=thick
dendrite, d=spiny dendrite */
float ENa = 0.045
float GNaFs = 75000.0
float GNaPs = 10.0
float ECa = 0.0125*{log {CCaO/CCaI}} //
0.135 V
float GCaTs = 5.00
float GCaTm = 5.00
float GCaTt = 5.00
float GCaTd = 5.00
float GCaPm = 45.0
float GCaPt = 45.0
float GCaPd = 45.0
float EK = -0.085
float GKAs = 150.0
float GKAm = 20.0
float GKdrs = 6000.0
float GKdrm = 600.0
float GKMs = 0.400
float GKMm = 0.100
```

<genhome>/Scripts/Purkinje/Purk_cicomp.g

```
if (!({exists Purk_soma})
        create compartment Purk_soma
end
 setfield Purk_soma Cm {{CM}*surf} Ra {8.0*{RA}/
(dia*{PI})}  \
Em {ELEAK} initVm {EREST_ACT} Rm
{{RMs}/surf} inject 0.0  dia {dia} len {len}

copy Purk_NaF Purk_soma/NaF
addmsg Purk_soma Purk_soma/NaF VOLTAGE Vm
addmsg Purk_soma/NaF Purk_soma CHANNEL Gk Ek
setfield Purk_soma/NaF Gbar {{GNaFs}*surf}
 copy Purk_NaP Purk_soma/NaP
 addmsg Purk_soma Purk_soma/NaP VOLTAGE Vm
 addmsg Purk_soma/NaP Purk_soma CHANNEL Gk Ek
 setfield Purk_soma/NaP Gbar {{GNaPs}*surf}
 copy Purk_CaT Purk_soma/CaT
 addmsg Purk_soma Purk_soma/CaT VOLTAGE Vm
 addmsg Purk_soma/CaT Purk_soma CHANNEL Gk Ek
 setfield Purk_soma/CaT Gbar {{GCaTs}*surf}
 copy Purk_KA Purk_soma/KA
 addmsg Purk_soma Purk_soma/KA VOLTAGE Vm
 addmsg Purk_soma/KA Purk_soma CHANNEL Gk Ek
 setfield Purk_soma/KA Gbar {{GKAs}*surf}
copy Purk_Kdr Purk_soma/Kdr
addmsg Purk_soma Purk_soma/Kdr VOLTAGE Vm
addmsg Purk_soma/Kdr Purk_soma CHANNEL Gk Ek
setfield Purk_soma/Kdr Gbar {{GKdrs}*surf}
copy Purk_KM Purk_soma/KM
addmsg Purk_soma Purk_soma/KM VOLTAGE Vm
addmsg Purk_soma/KM Purk_soma CHANNEL Gk Ek
setfield Purk_soma/KM Gbar {{GKMs}*surf}

copy Purk_h1 Purk_soma/h1
addmsg Purk_soma Purk_soma/h1 VOLTAGE Vm
addmsg Purk_soma/h1 Purk_soma CHANNEL Gk Ek
setfield Purk_soma/h1 Gbar {{Ghs}*surf}
copy Purk_h2 Purk_soma/h2
addmsg Purk_soma Purk_soma/h2 VOLTAGE Vm
addmsg Purk_soma/h2 Purk_soma CHANNEL Gk Ek
setfield Purk_soma/h2 Gbar {{Ghs}*surf}
create Ca_concen Purk_soma/Ca_pool
setfield Purk_soma/Ca_pool tau {CaTau}  \
        B {1.0/(2.0*96494*shell_vol)} Ca_base {CCaI}  \
        thick {Shell_thick}
 addmsg Purk_soma/CaT Purk_soma/Ca_pool I_Ca Ik
copy GABA2 Purk_soma/basket
setfield Purk_soma/basket gmax    {{GB_GABA}*surf}
addmsg Purk_soma/basket  Purk_soma  CHANNEL Gk  Ek
addmsg Purk_soma  Purk_soma/basket    VOLTAGE Vm
```

<genhome>/Scripts/Purkinje/Purk_cicomp.g

```
//- include scripts to create the prototypes

include Purk_chanload
include Purk_cicomp
include Purk_syn

include info.g
include bounds.g
include config.g
include control.g
include xgraph.g
include xcell.g


//- ensure that all elements are made in the library

ce /library

//- make prototypes of channels and synapses

make_Purkinje_chans

readcell {cellfile} {cellpath}


echo preparing hines solver {getdate}
ce {cellpath}
create hsolve solve

//- We change to current element solve and then set the fields of
the parent
//- (solve) to get around a bug in the "." parsing of genesis

ce solve

setfield . \
        path "../##[][TYPE=compartment]" \
        comptmode 1 \
        chanmode {iChanMode} \
        calcmode 0

call /Purkinje/solve SETUP

//- Use method to Crank-Nicolson

setmethod 11
```

File Saving System: from DCN simulation by vsteuber

```
include cn_fileout16

outfilev = "data/cn0106c_" @ {simnum} @ "_v_"
outfilei = "data/cn0106c_" @ {simnum} @ "_i_"
outfilevc = "data/cn0106c_" @ {simnum} @ "_vc_"
outfilechan = "data/cn0106c_" @ {simnum} @ "_chan_“


// set up messages to the output files
        write_voltage soma 0
        write_currents_c soma 0
        write_chan_activations soma 0
        //write_voltage p0b2 0
        //write_voltage p0b2b1b2 0
        //write_voltage p0b2b1b2 4
        write_voltage p0 1
        write_voltage p1 1
```

```
function write_voltage(compt, comptnum)
str compt, comptnum
str hinesloc, outfile, outelement, clampstr

outfile = {outfilev @ compt @ comptnum @ "_" @ clampstr @
".dat"}
outelement = "/output/v_" @ {compt} @ {comptnum}

if (!{exists {outelement}})
        create asc_file {outelement}
        echo
        echo creating output element {outelement}
        useclock {outelement} 8
        hinesloc = {findsolvefield {cellpath} {cellpath}/{compt}
Vm}
        addmsg {cellpath} {outelement} SAVE {hinesloc}
    end

    setfield {outelement}      filename {outfile} \
                        initialize 1 \
                        leave_open 1 \
                        append 1


    echo
    echo sending {cellpath}/{compt} Vm to {outfile}.

end
```

Start of Matlab function to read genesis output.

```
function p_cn6c_sc13(cip,n,fignum1,fignum2,eleak)

addpath('../data')


if (cip < 0.0)
    ifile = sprintf('cn0106c_%s_chan_soma0_m%spA.dat',
num2str(n),num2str(-cip));
else
    ifile = sprintf('cn0106c_%s_chan_soma0_%spA.dat',
num2str(n),num2str(cip));
end
temp = load(ifile);
…
```

Matlab output to scan results of parameter combination