Network Simulations with NEURON

Single cell

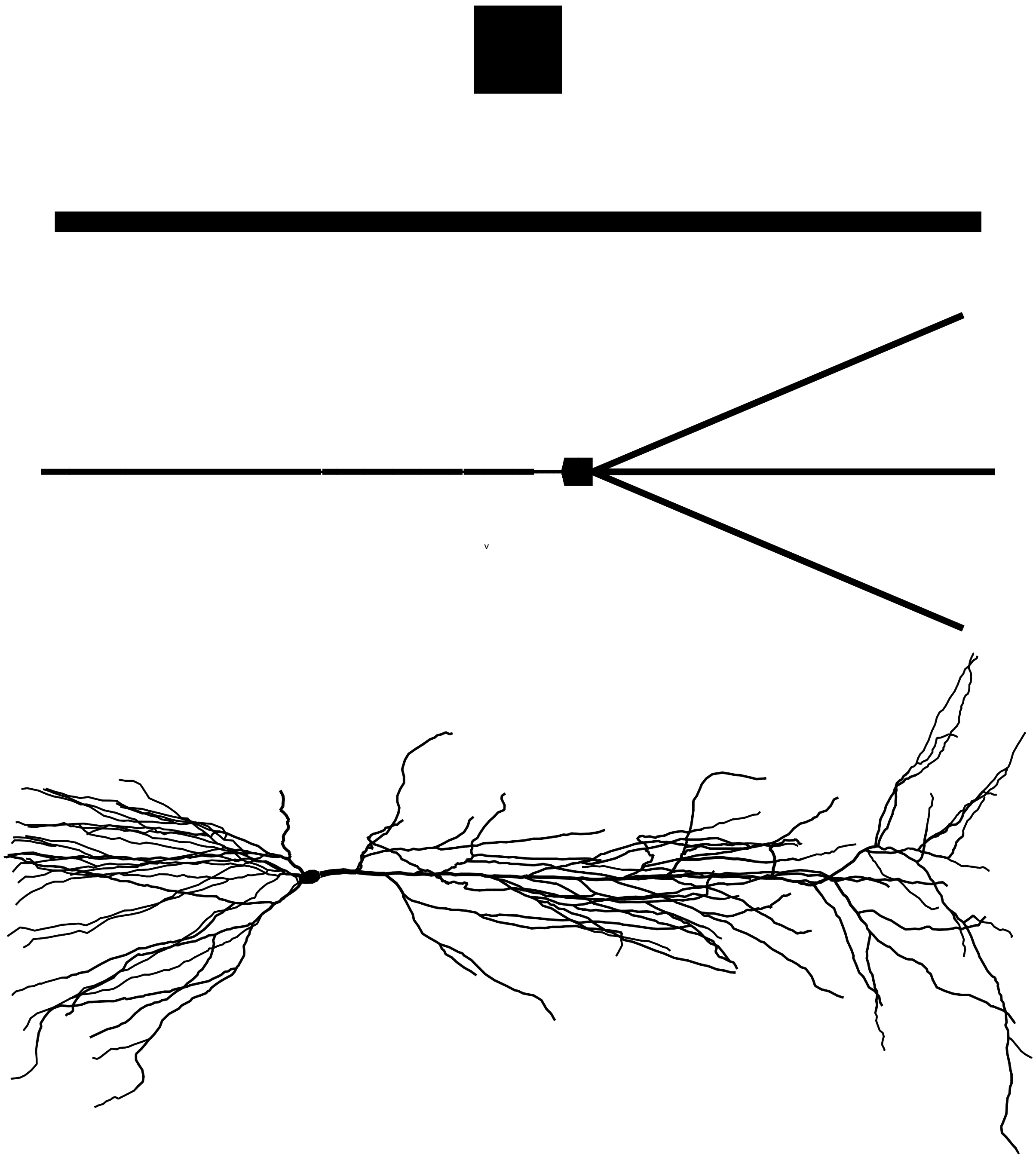
Network

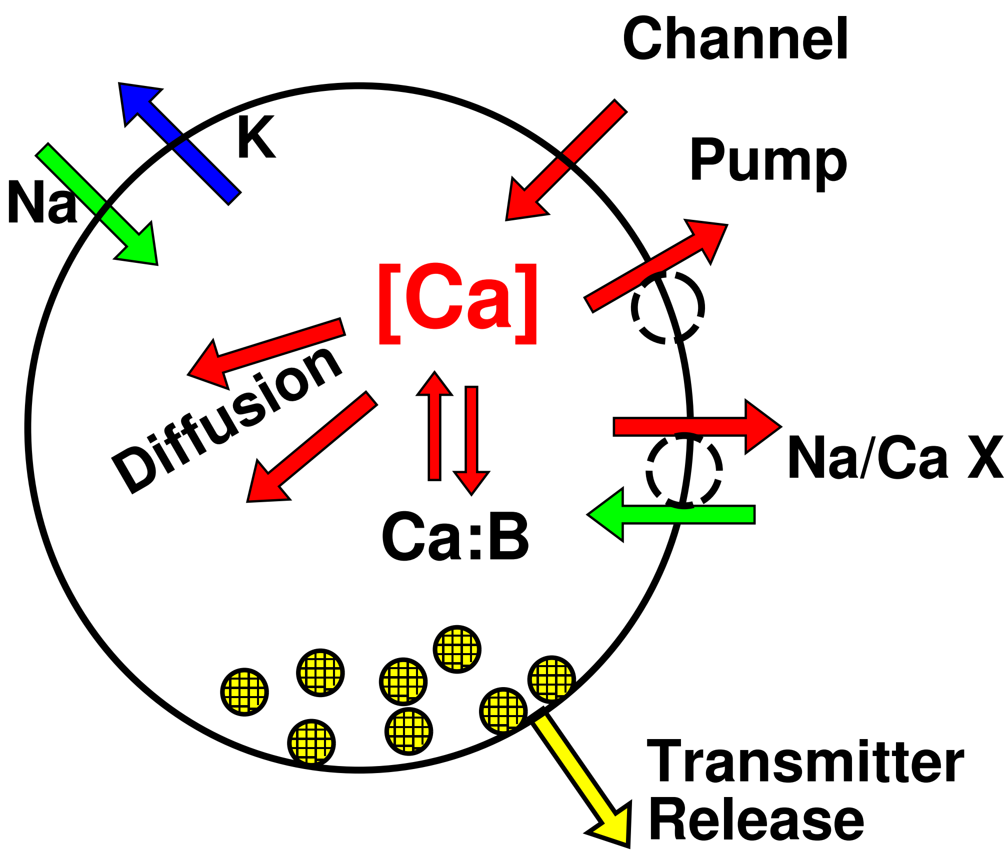
Local variable time steps

Synaptic events

Artificial spiking cells

http://www.neuron.yale.edu/





Single Channels
Extracellular fields
Linear circuits
Synapses
Networks

Physical System

Squid axon

Model

Hodgkin−Huxley cable equations

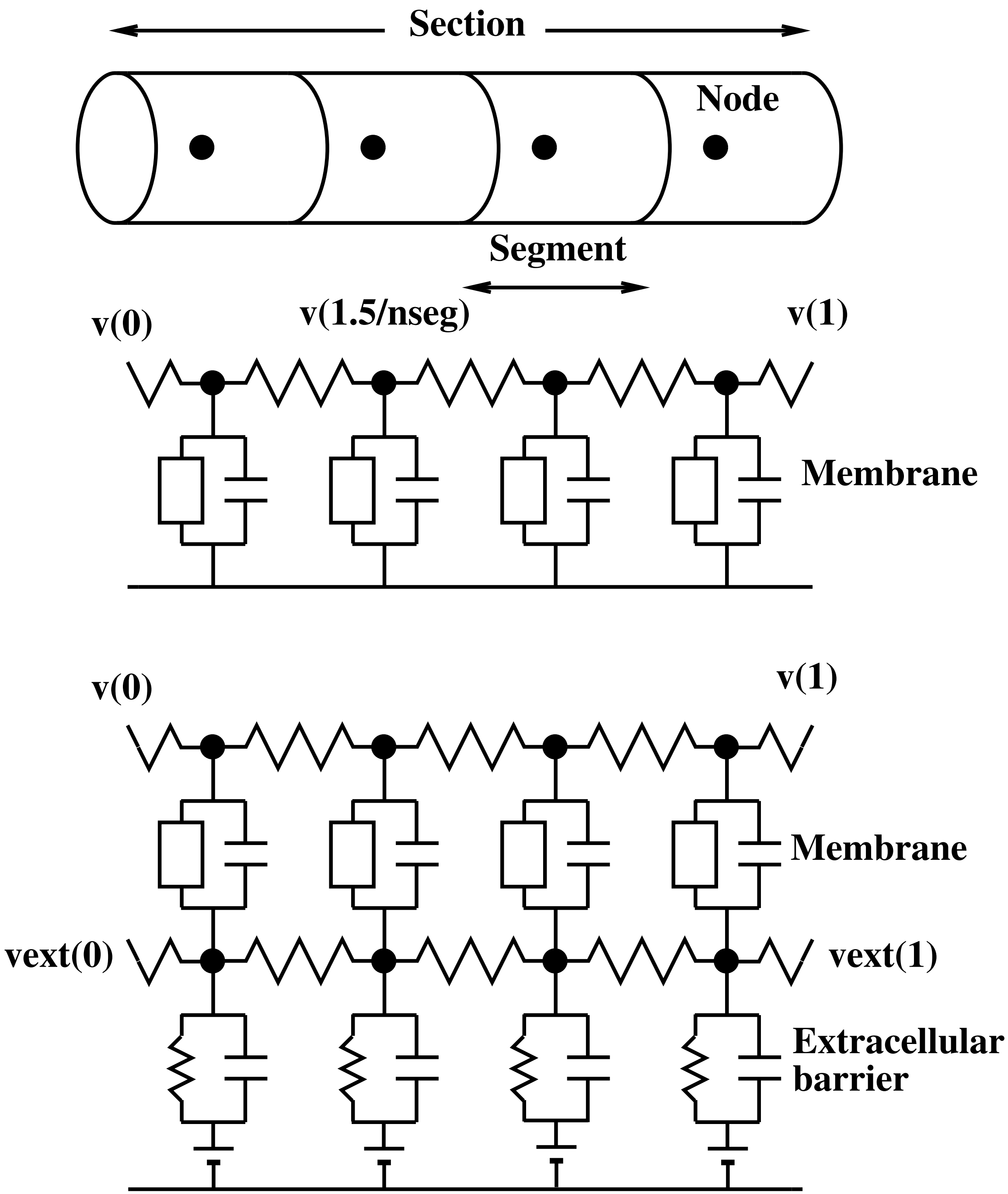
$$\frac{D}{4R_a} \cdot \frac{\partial^2 V}{\partial x^2} = C_m \frac{\partial V}{\partial t} + \overline{g}_{na} m^3 h \cdot (V - E_{na}) + \overline{g}_k n^4 \cdot (V - E_k) + g_l \cdot (V - E_l)$$

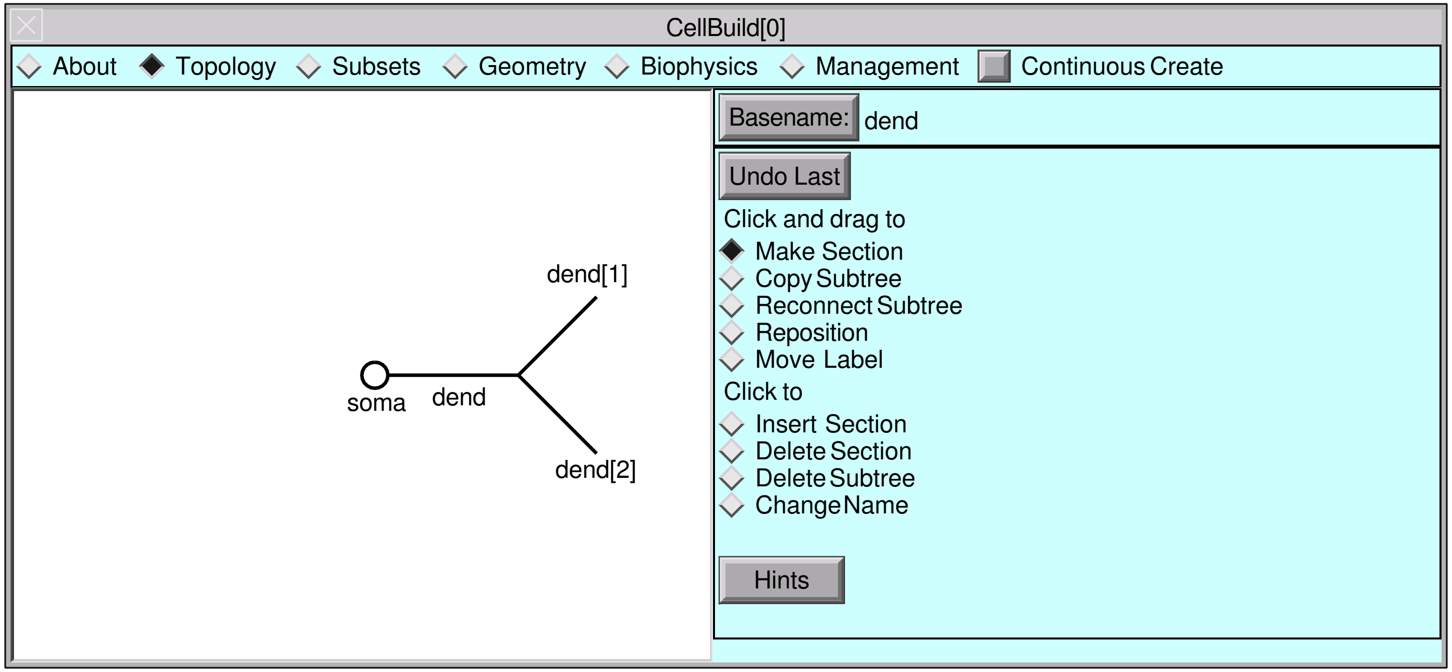
$$\begin{array}{lll} \frac{dm}{dt} = -\alpha_m m + \beta_m \cdot (1 - m) & \alpha_m = \frac{.1(V+40)}{1-e^{-.1(V+40)}} & \beta_m = 4e^{-(V+65)/18} \\ \frac{dh}{dt} = -\alpha_h h + \beta_h \cdot (1 - h) & \alpha_h = .07e^{-.05(V+65)} & \beta_h = \frac{1}{1+e^{-.1(V+35)}} \\ \frac{dn}{dt} = -\alpha_n n + \beta_n \cdot (1 - n) & \alpha_n = \frac{.01(V+55)}{1-e^{-.1(V+55)}} & \beta_n = .125e^{-(V+65)/80} \end{array}$$

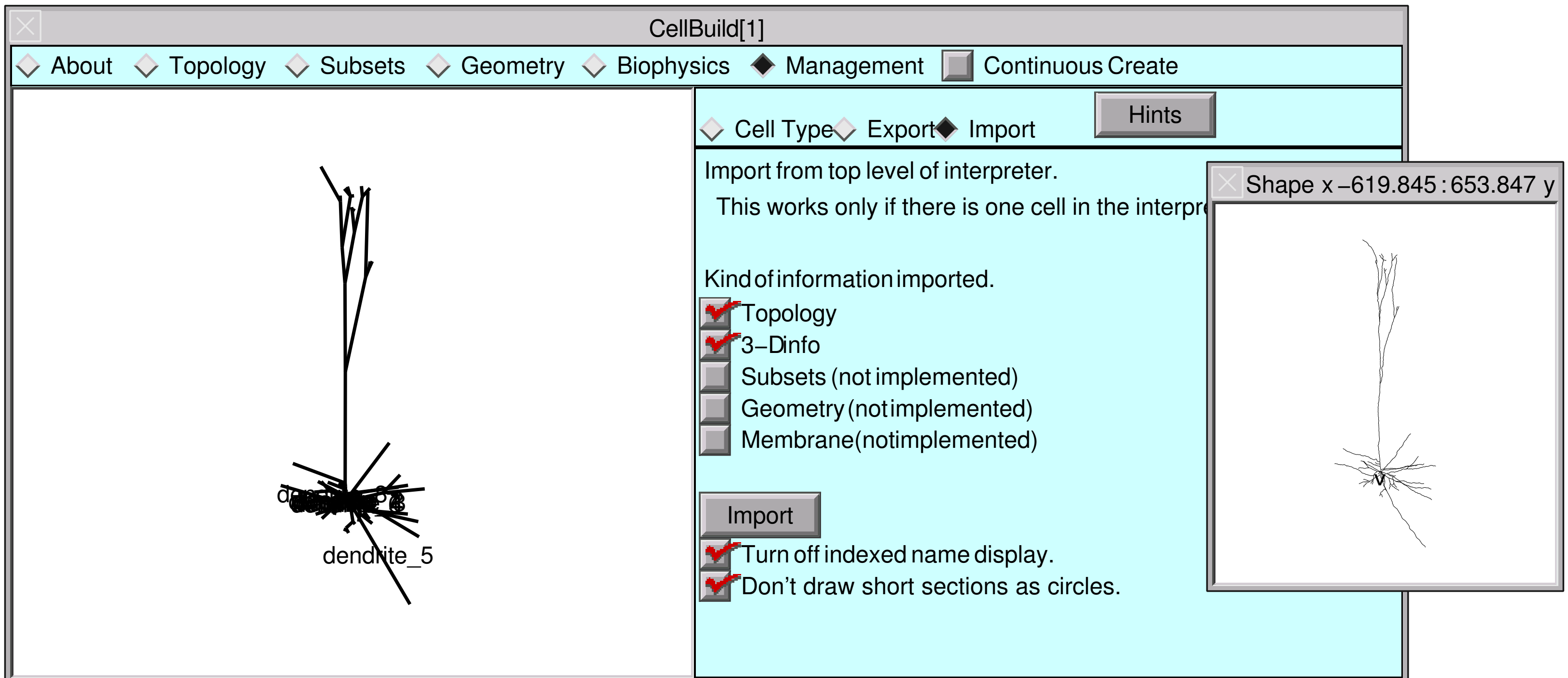
Simulation

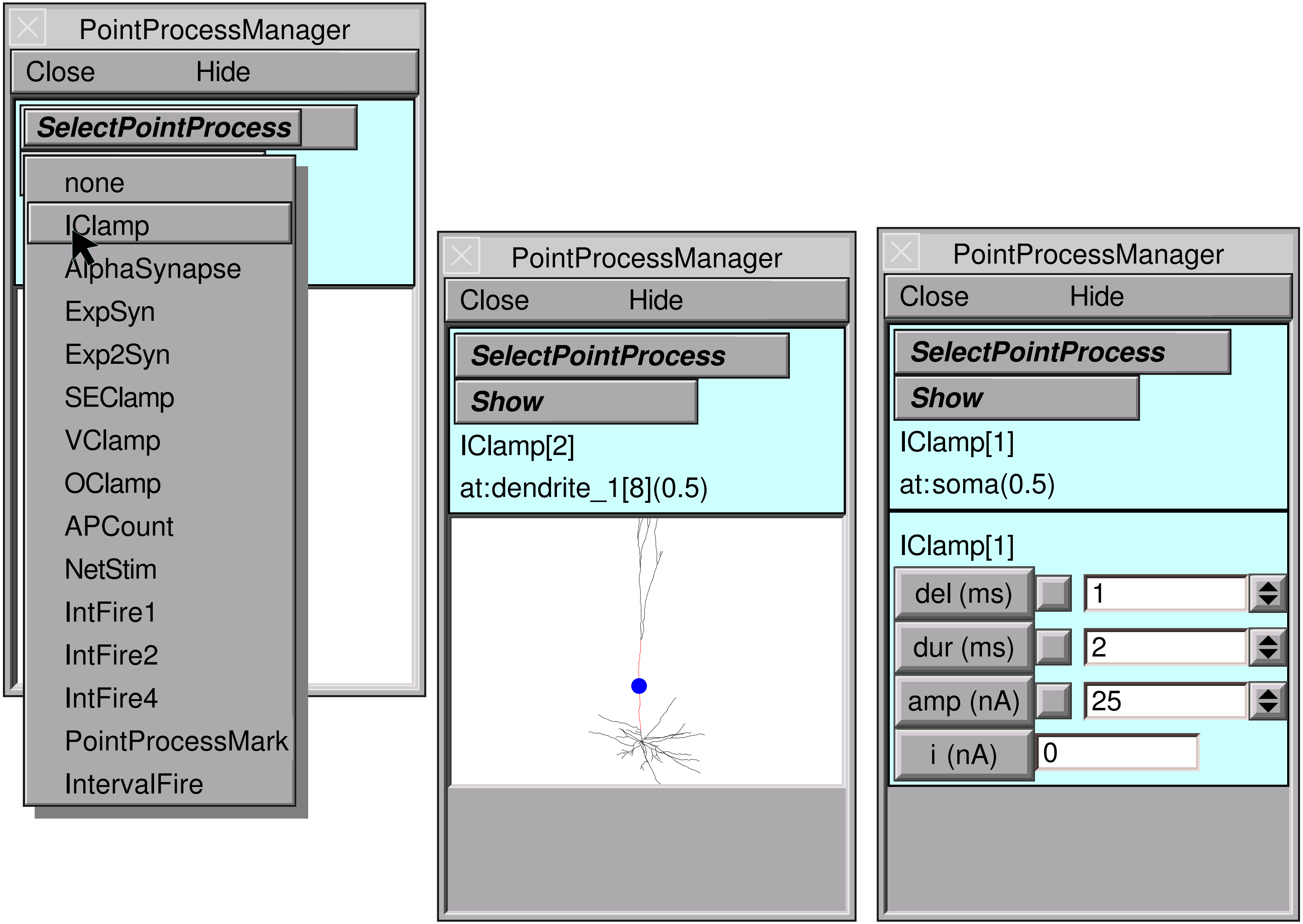
NEURON representation

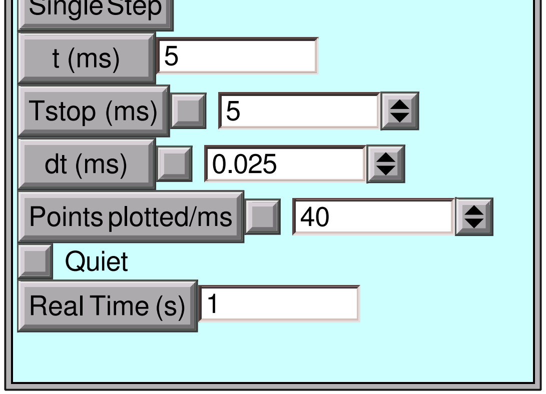
```
create axon
axon {
    nseg = 75
    diam = 100
    L = 20000
    insert hh
}
```

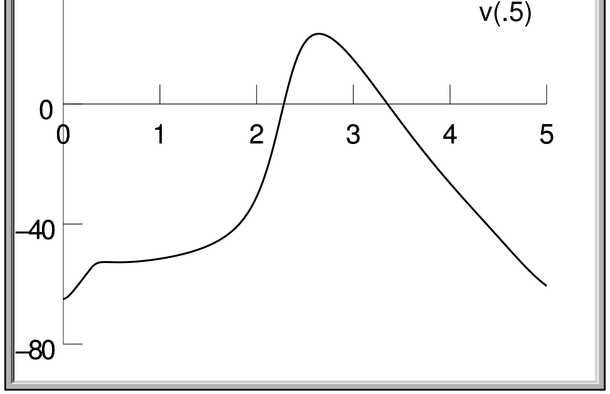

Section
Node
Segment
v(0)
v(1.5/nseg)
v(1)
Membrane
v(0)
v(1)
Membrane
vext(0)
vext(1)
Extracellular
barrier

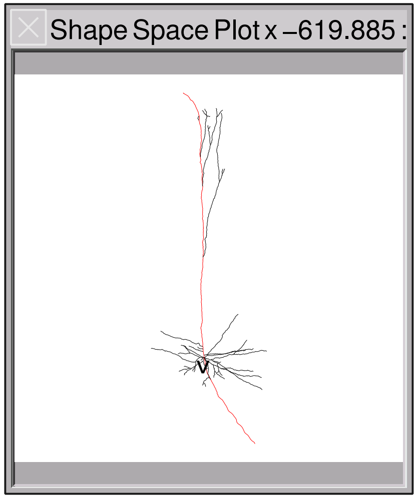

CellBuild[0]
About
Topology
Subsets
Geometry
Biophysics
Management
Continuous Create
Basename: dend
Undo Last
Click and drag to
Make Section
Copy Subtree
Reconnect Subtree
Reposition
Move Label
Click to
Insert Section
Delete Section
Delete Subtree
Change Name
Hints
dend[1]
soma
dend
dend[2]

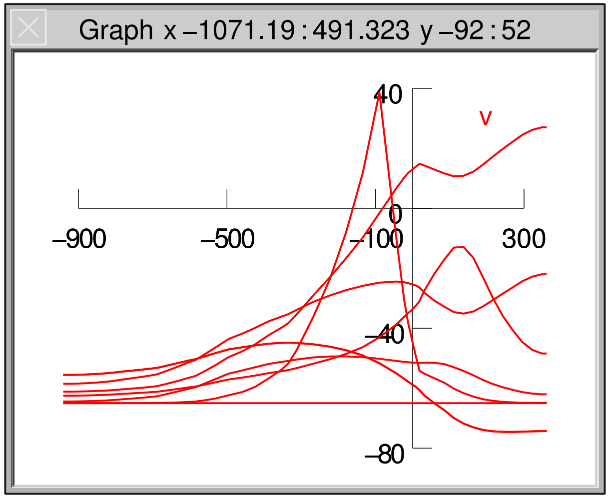

CellBuild[1]
About
Topology
Subsets
Geometry
Biophysics
Management
Continuous Create
Cell Type
Export
Import
Hints
Import from top level of interpreter.
This works only if there is one cell in the interpr
Kind of information imported.
Topology
3-Dinfo
Subsets (not implemented)
Geometry (not implemented)
Membrane (not implemented)
Import
Turn off indexed name display.
Don't draw short sections as circles.
dendrite_5
Shape x -619.845 : 653.847 y


PointProcessManager
Close
Hide
SelectPointProcess
none
IClamp
AlphaSynapse
ExpSyn
Exp2Syn
SEClamp
VClamp
OClamp
APCount
NetStim
IntFire1
IntFire2
IntFire4
PointProcessMark
IntervalFire
PointProcessManager
Close
Hide
SelectPointProcess
Show
IClamp[2]
at:dendrite_1[8](0.5)
PointProcessManager
Close
Hide
SelectPointProcess
Show
IClamp[1]
at:soma(0.5)
IClamp[1]
del (ms)
1
dur (ms)
2
amp (nA)
25
i (nA)
0


Single Step
t (ms)
5
Tstop (ms)
5
dt (ms)
0.025
Points plotted/ms
40
Quiet
Real Time (s)
1


v(.5)
0
–40
–80
0
1
2
3
4
5


Shape Space Plot x –619.885 :
v


Graph x –1071.19 : 491.323 y –92 : 52
40
v
0
–900
–500
–100
300
–40
–80

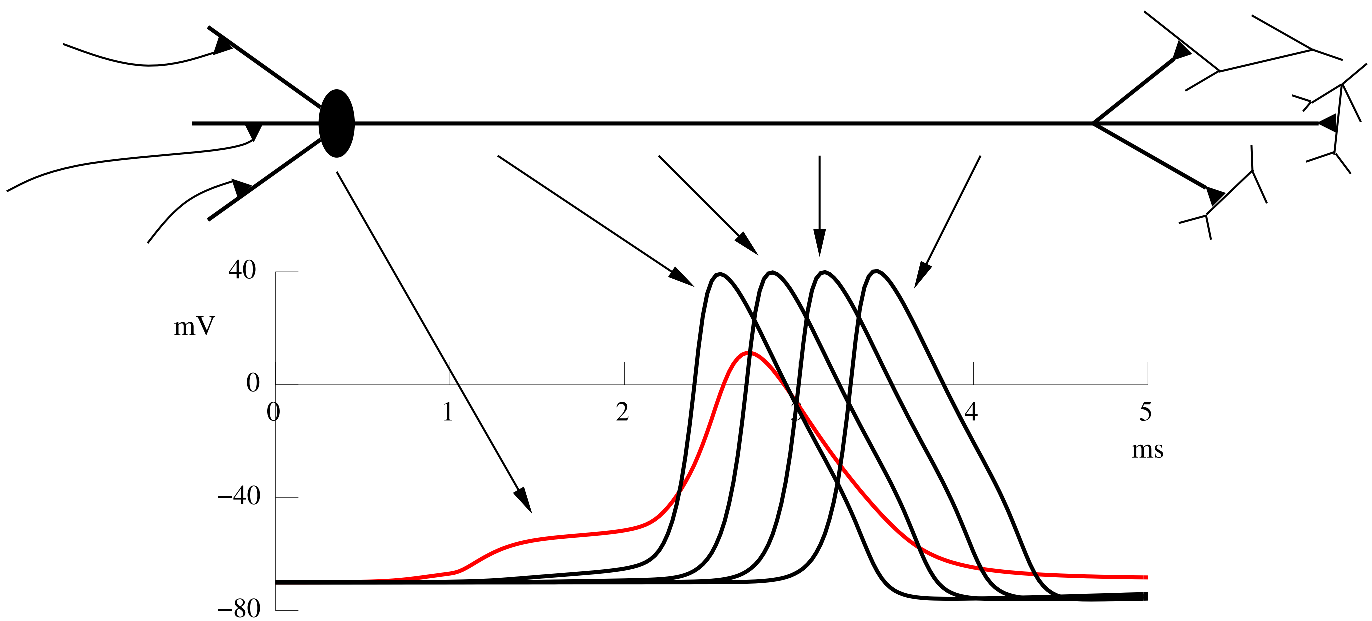
Single cell

- **Network**

Local variable time steps

Synaptic events

Artificial spiking cells



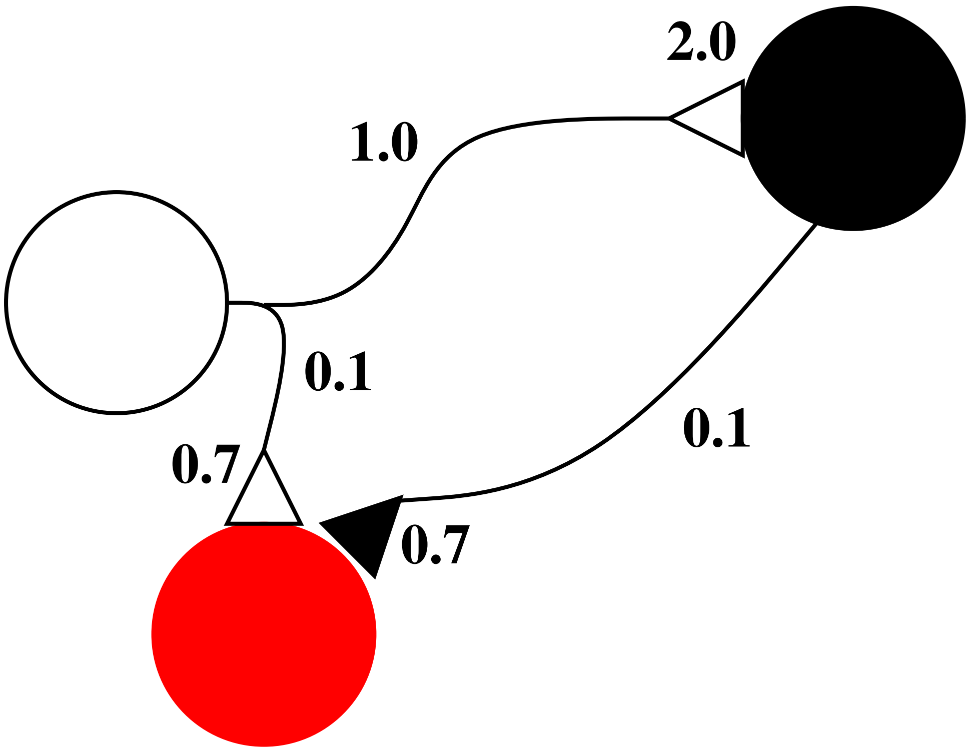
Single cell

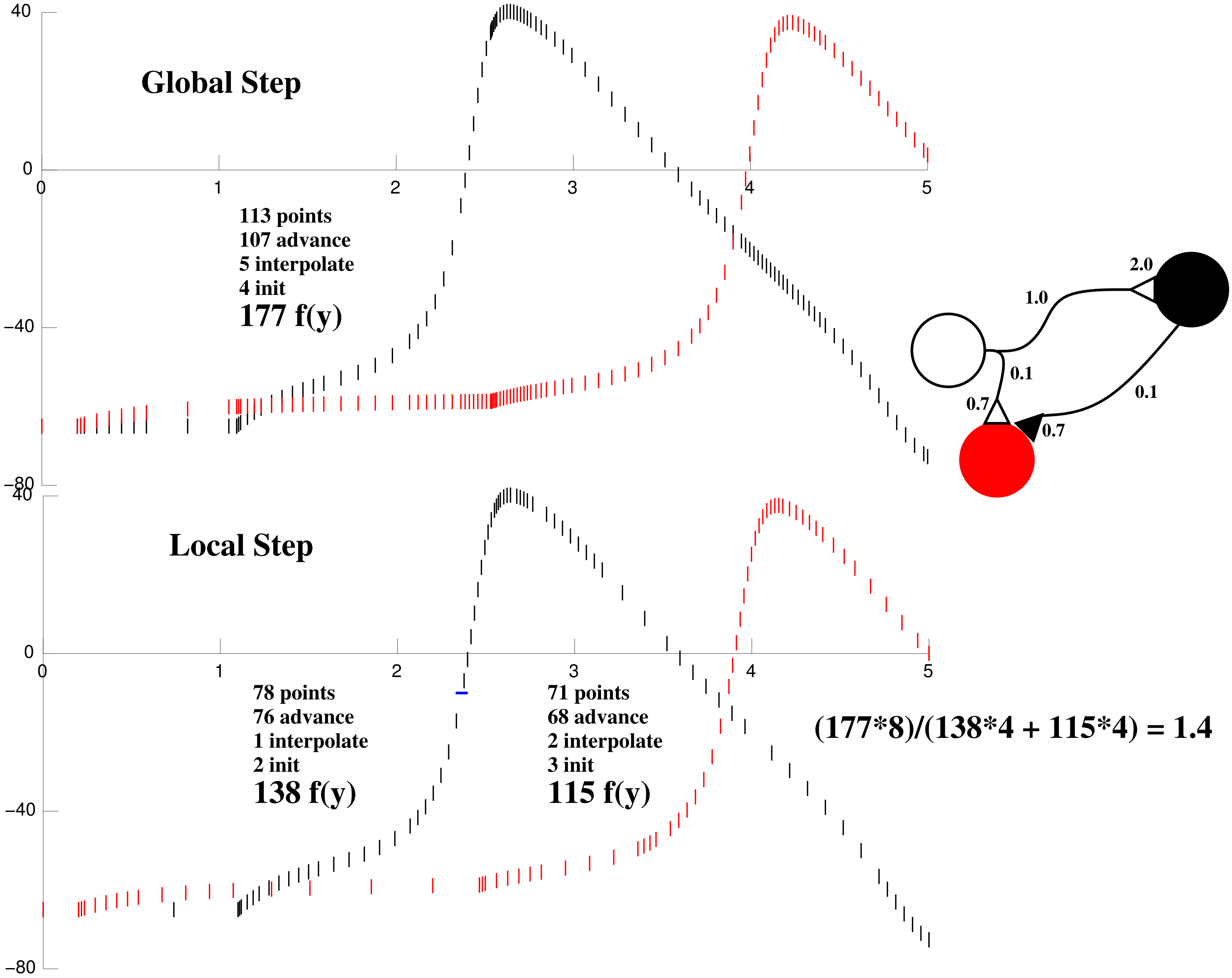
Network

- **Local variable time steps**

  Synaptic events

  Artificial spiking cells

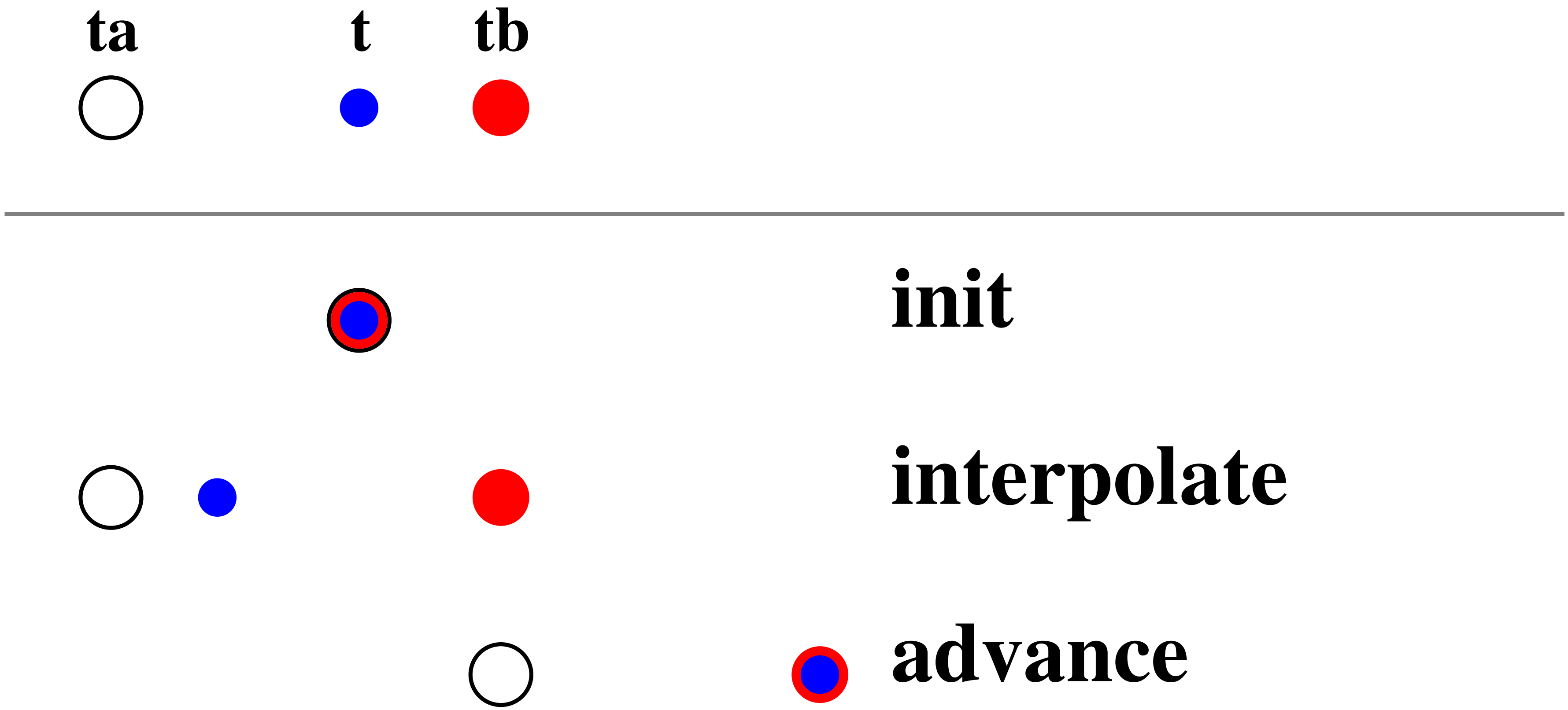


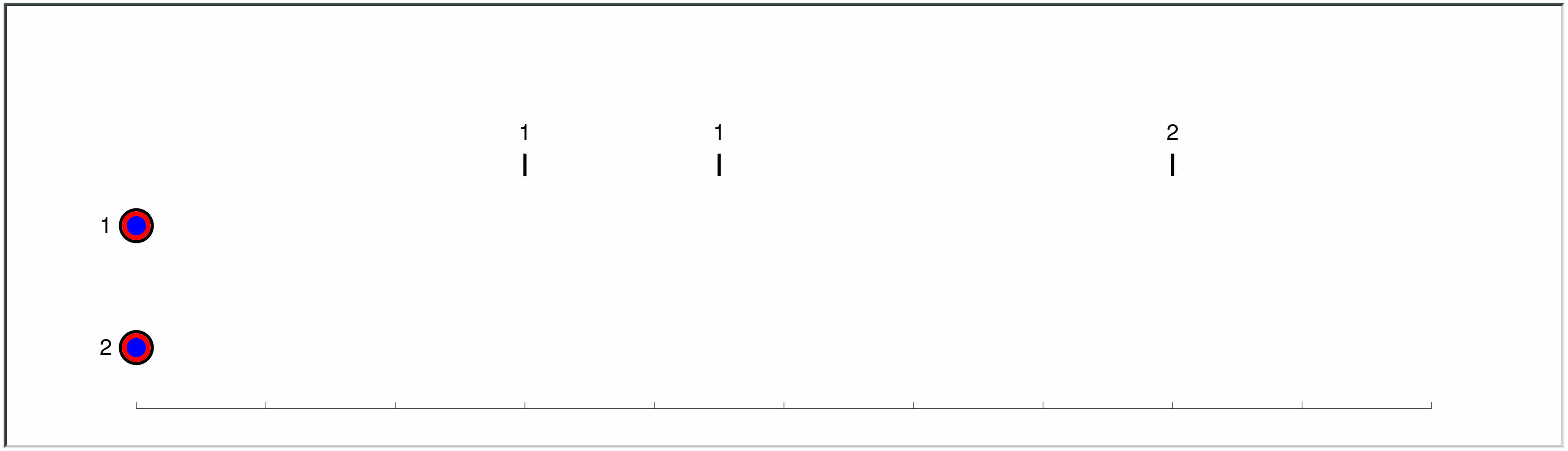


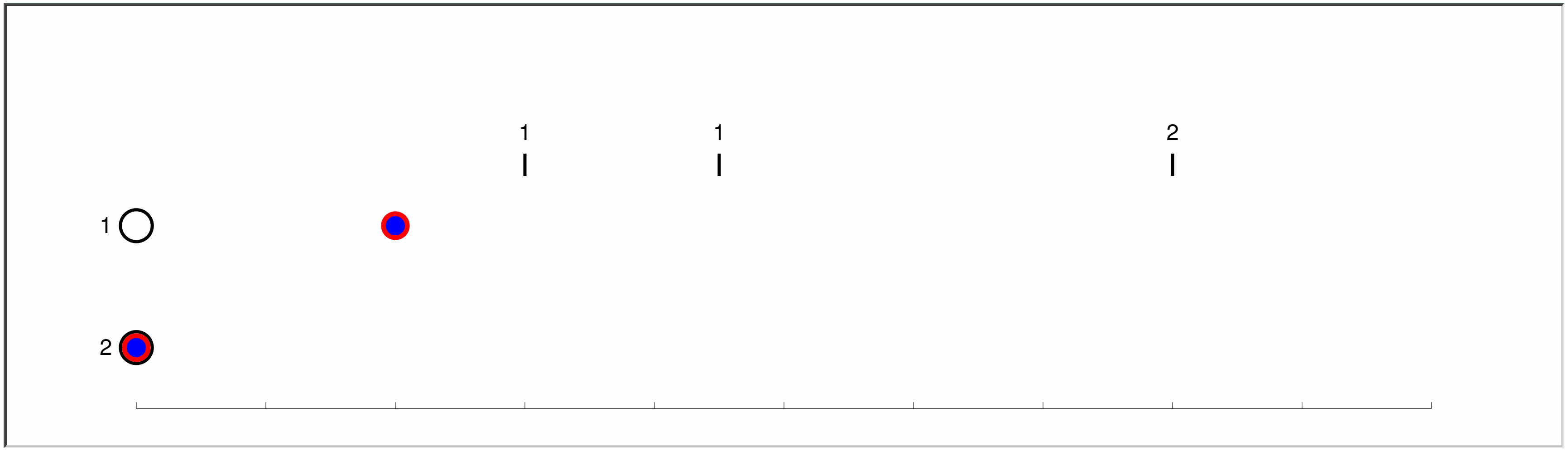
Global Step
113 points
107 advance
5 interpolate
4 init
177 f(y)
Local Step
78 points
76 advance
1 interpolate
2 init
138 f(y)
71 points
68 advance
2 interpolate
3 init
115 f(y)
(177*8)/(138*4 + 115*4) = 1.4
2.0
1.0
0.1
0.1
0.7
0.7
40
0
−40
−80
0
1
2
3
4
5

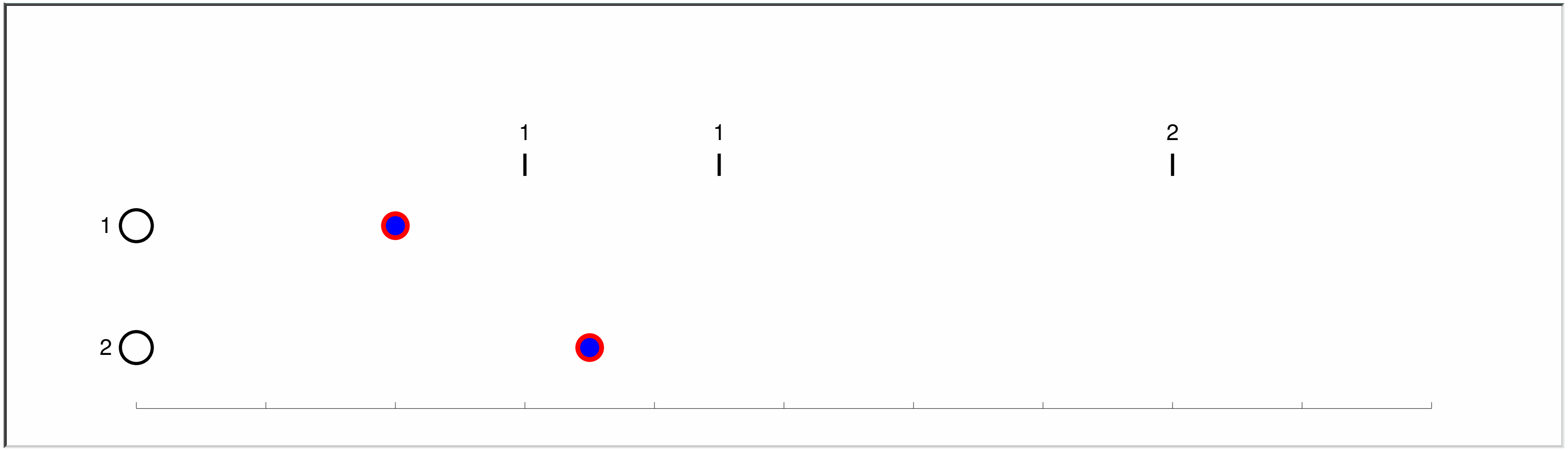
One integrator instance per cell

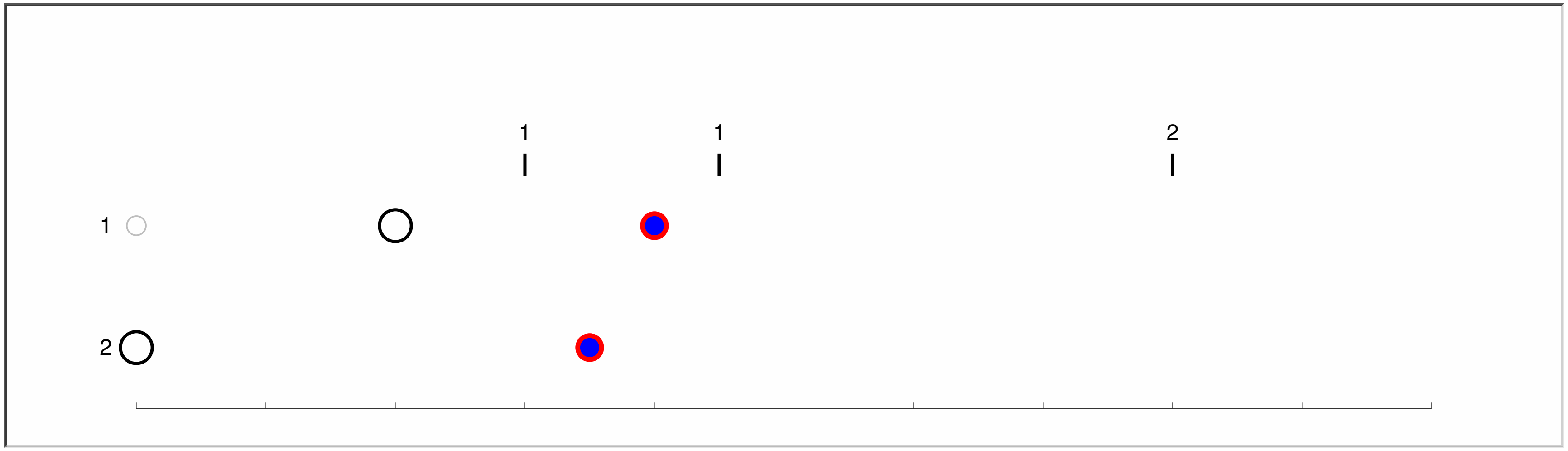
$$\forall\, i, j : \quad ta_i \leq tb_j$$

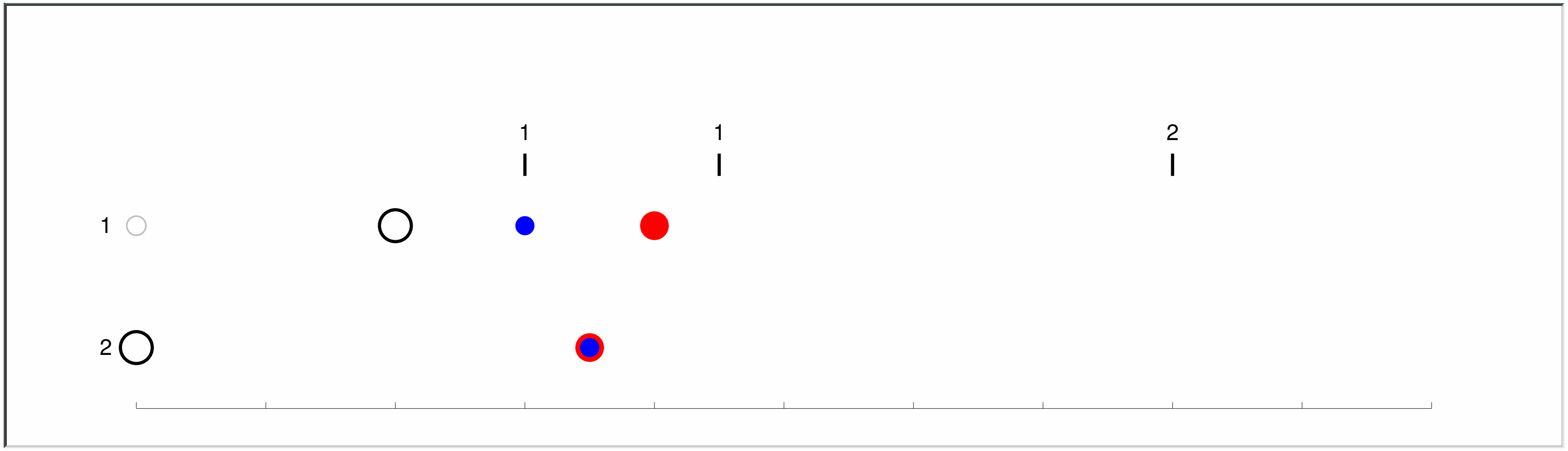



1
1
2
1
2

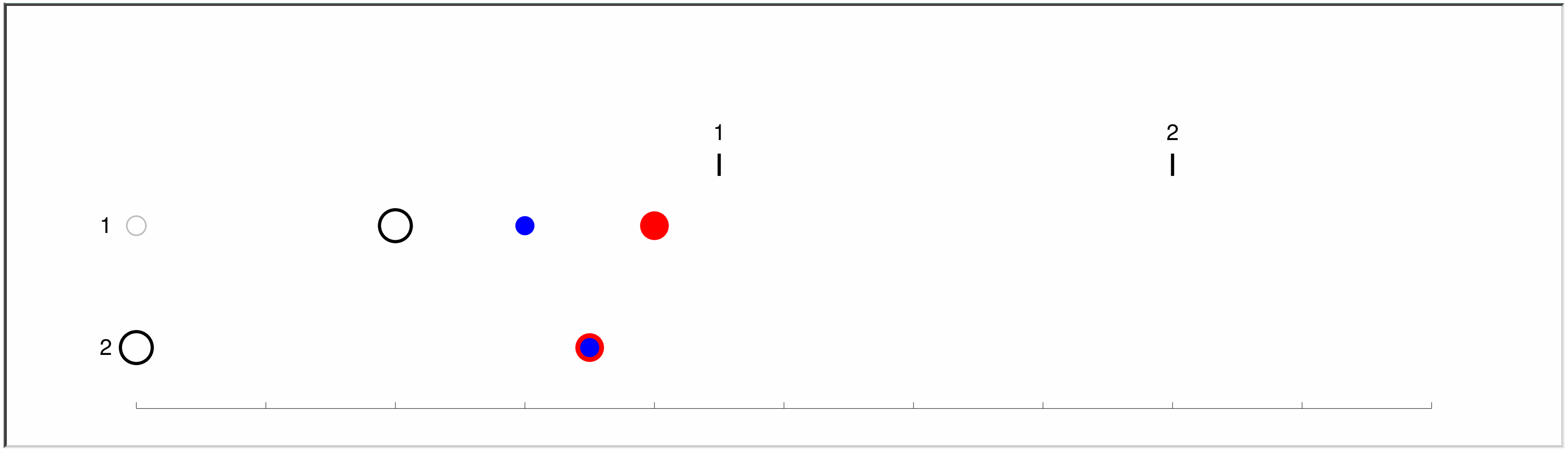

1
1
2
1
2


1
1
2
1
2

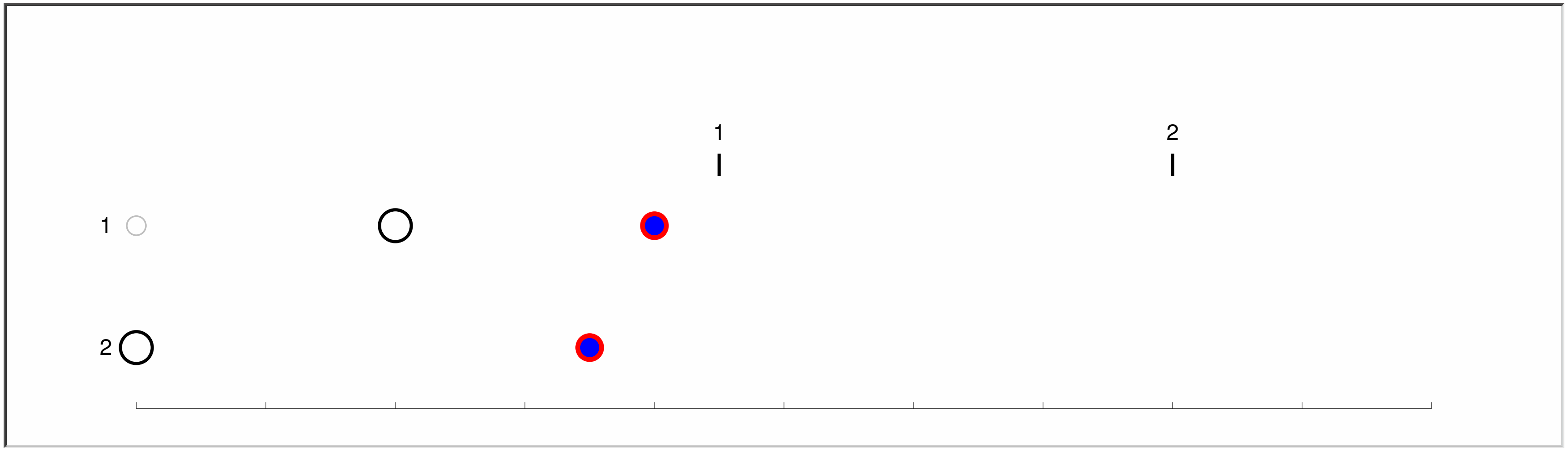

1
1
2
1
2

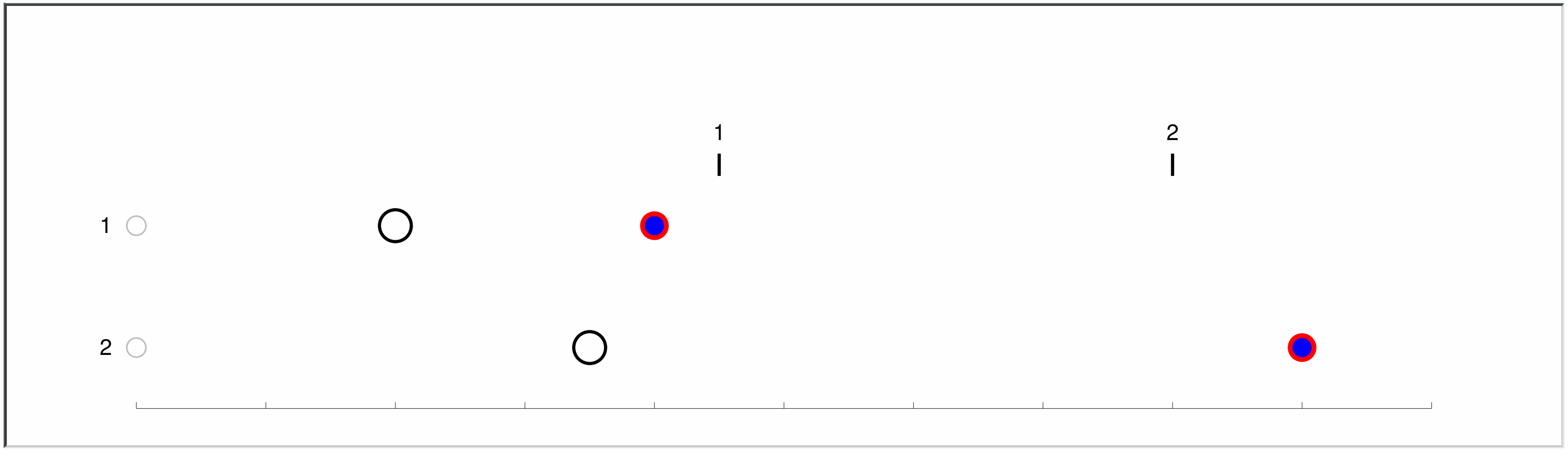


1
1
2
1
2

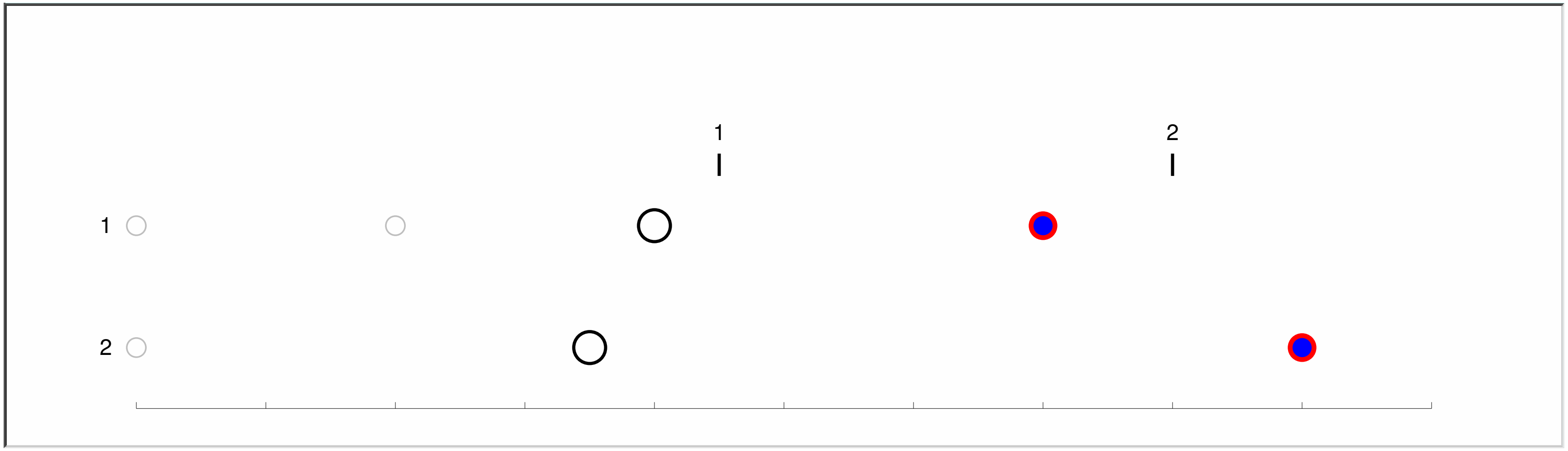


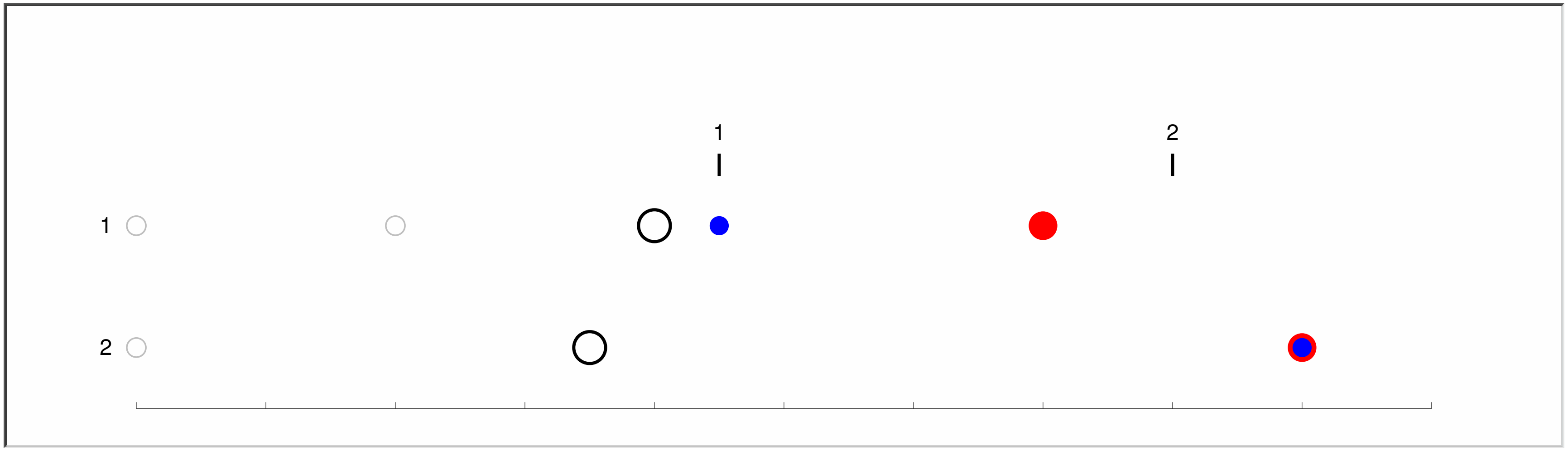
1
2
1
2


1
2
1
2

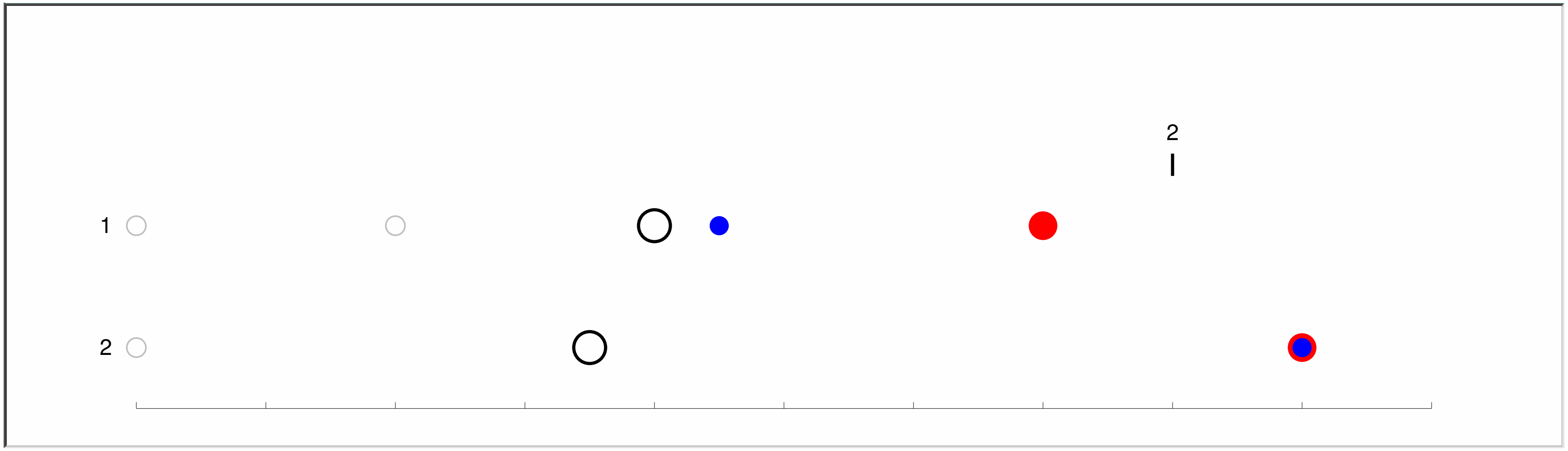


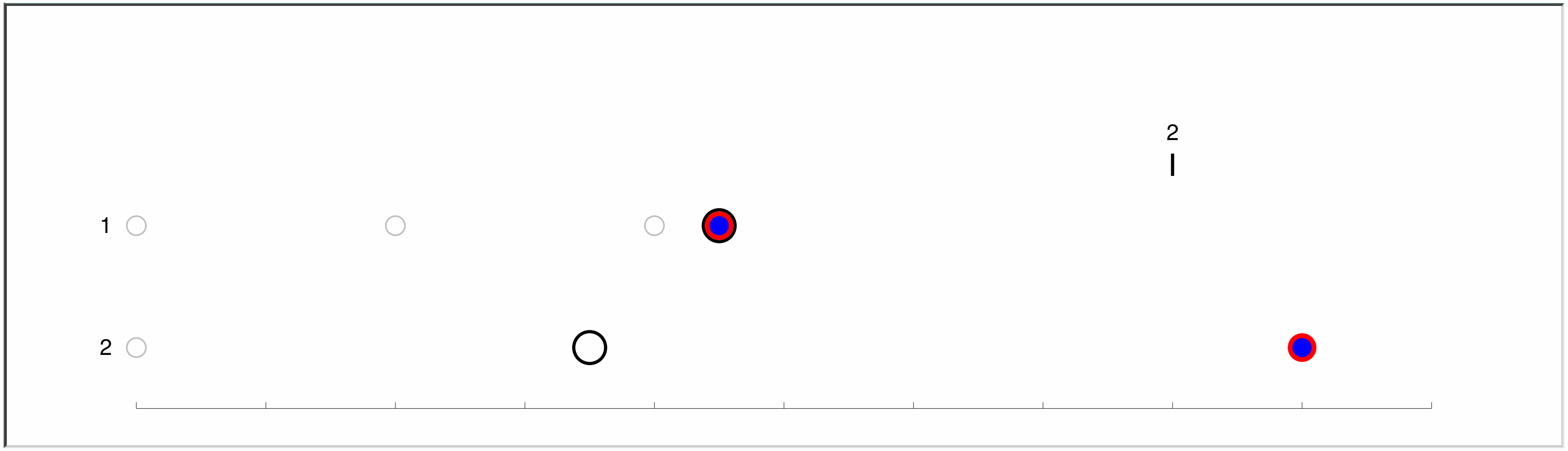
1
2
1
2

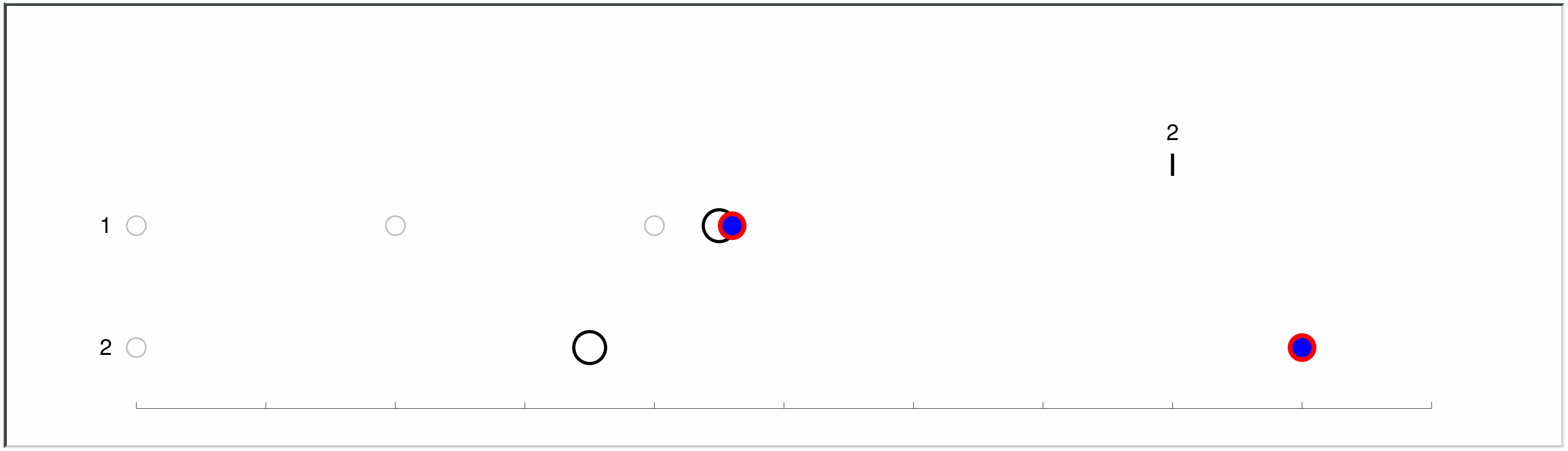

1
2
1
2

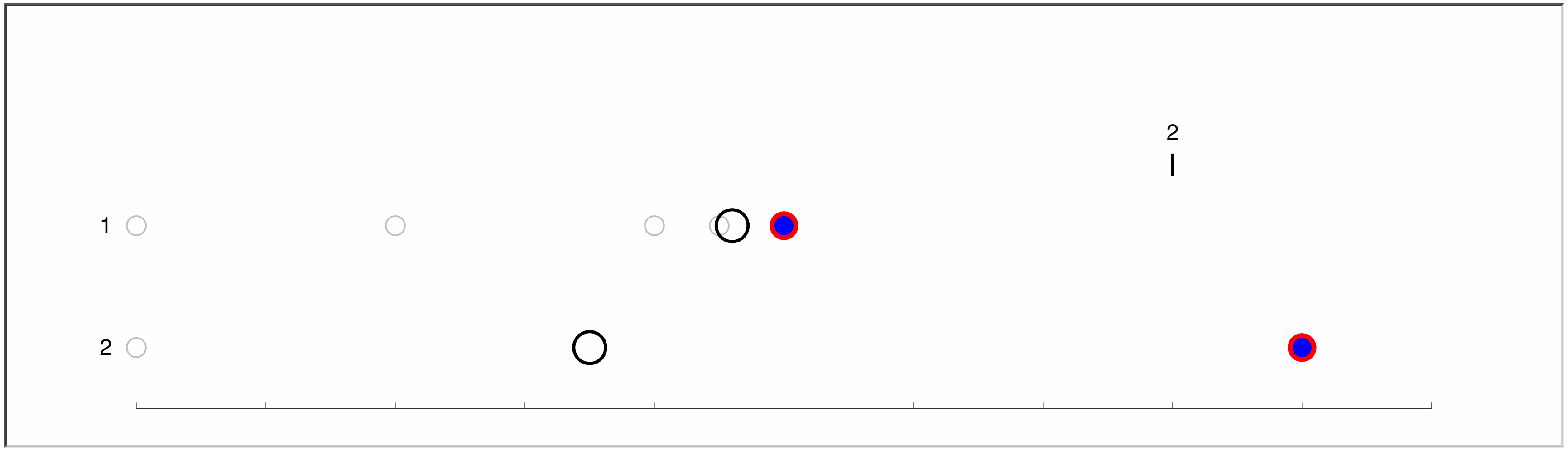


1
2
1
2

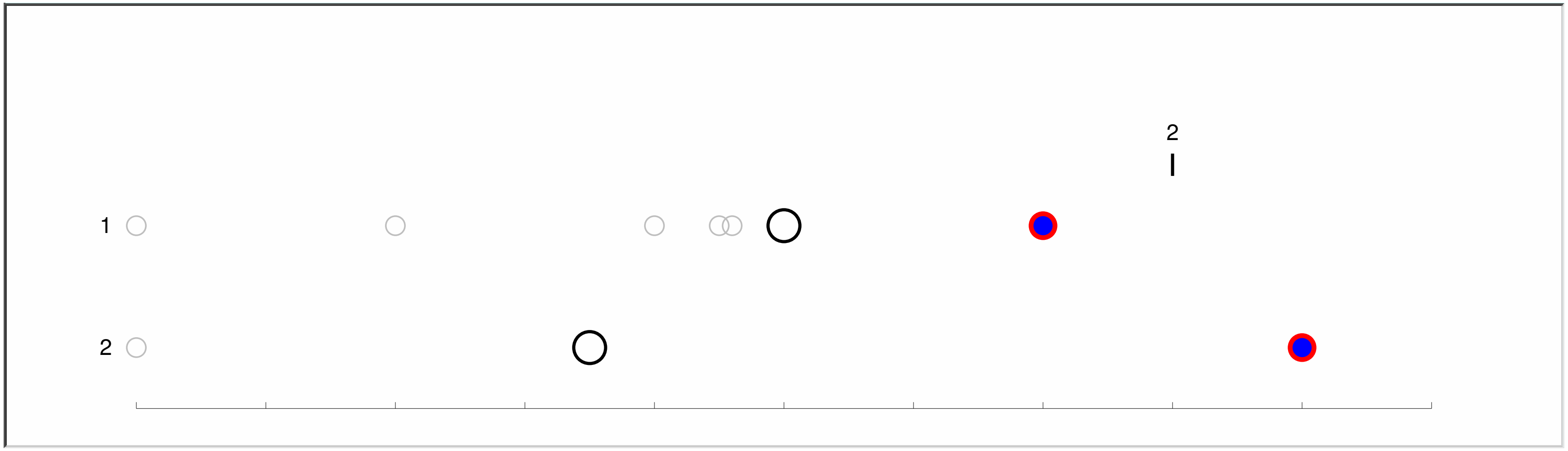

2
1
2

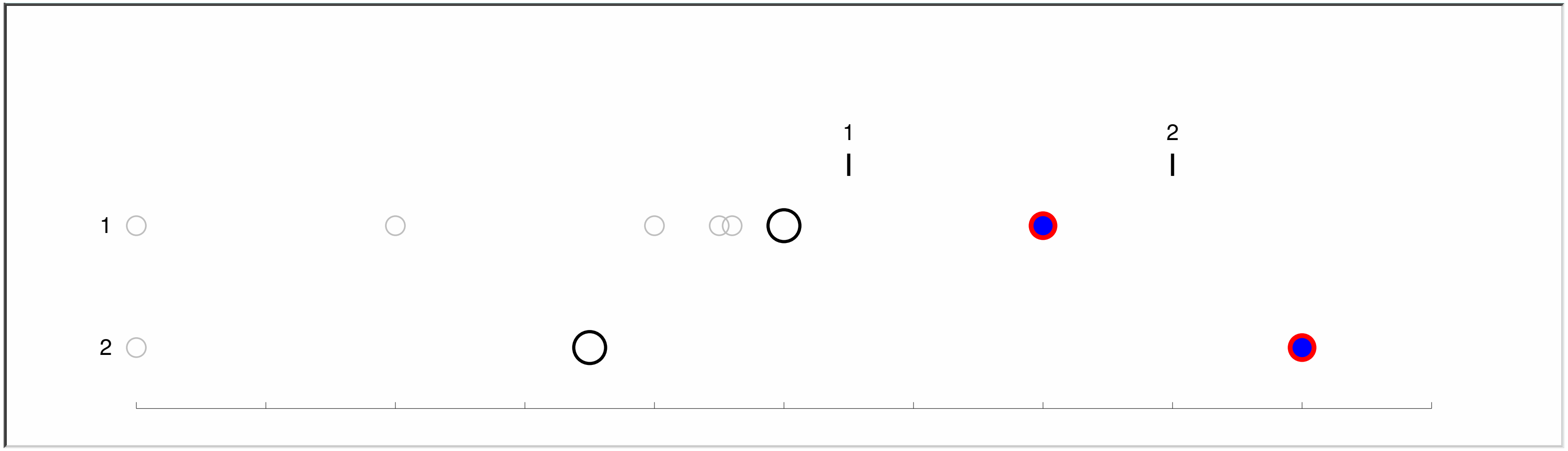

2
1
2


2
1
2

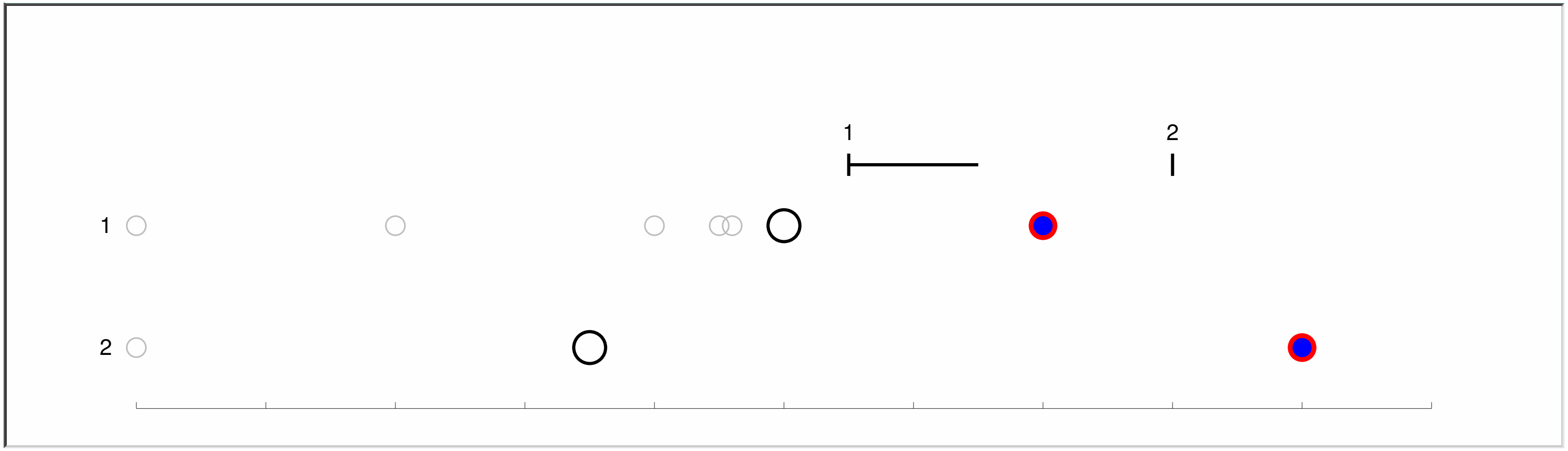


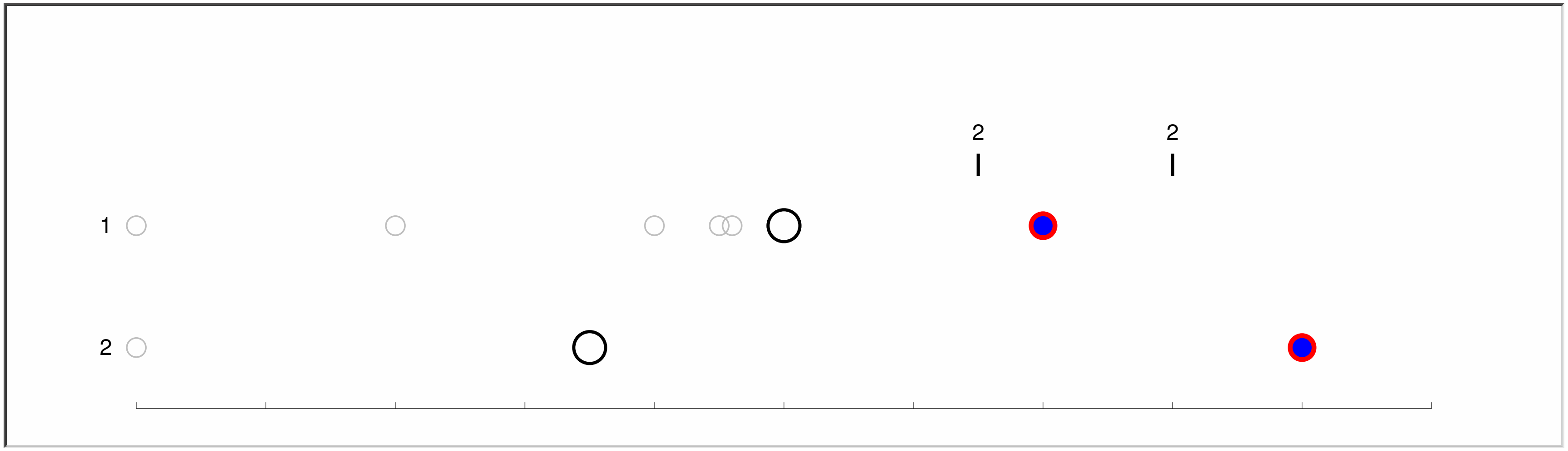
2
1
2


2
1
2

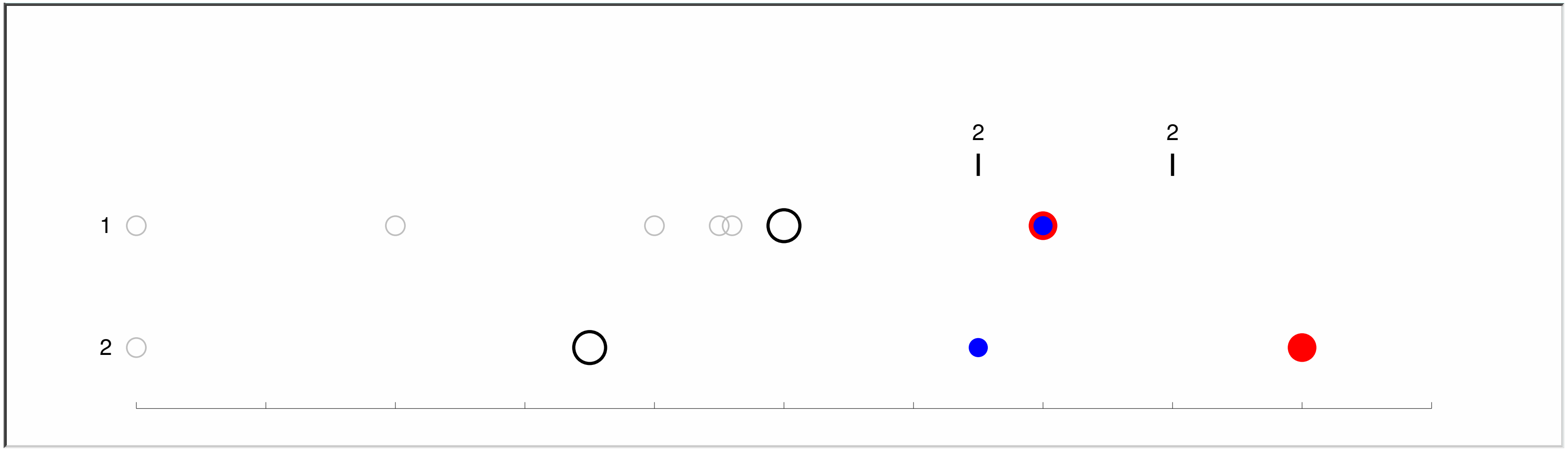


1
2
1
2

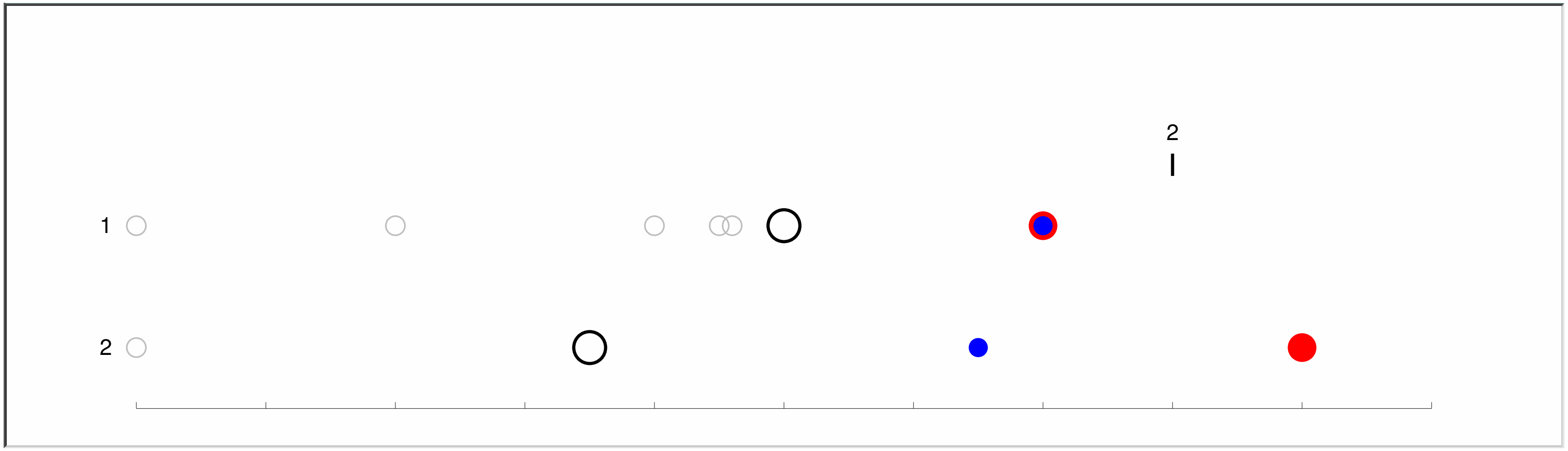

1
2
1
2

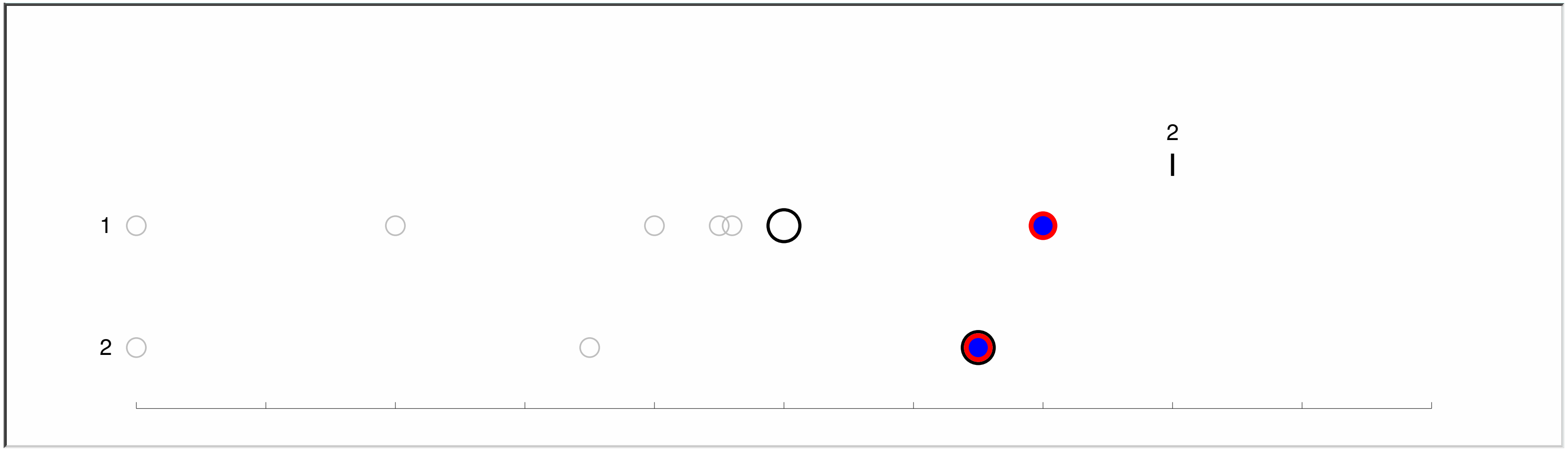


2
2
1
2



2
2
1
2


2
1
2



2
1
2

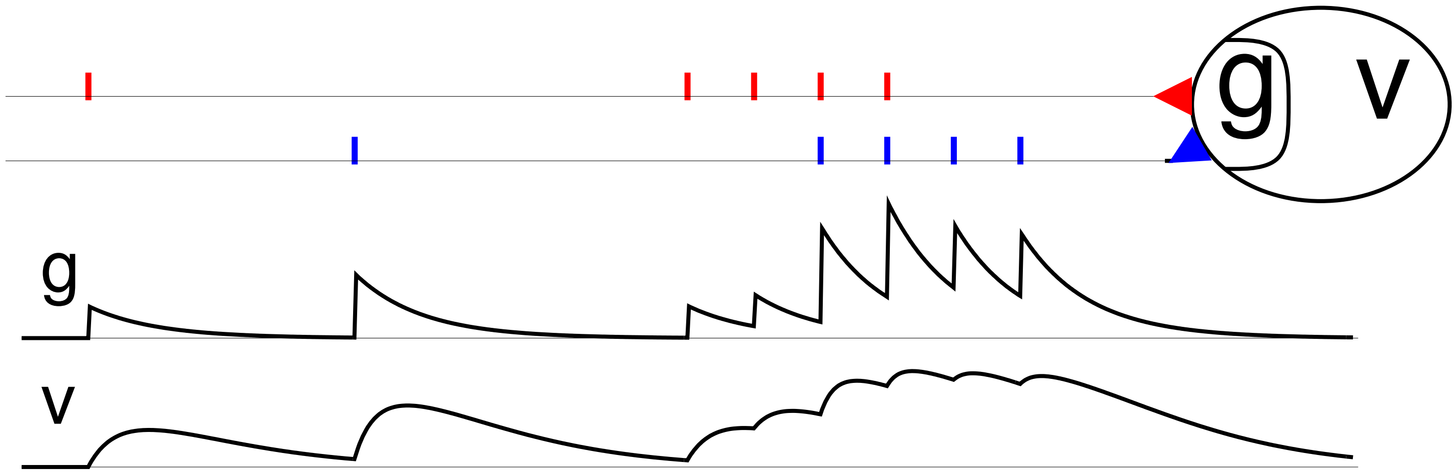
Single cell

Network

Local variable time steps

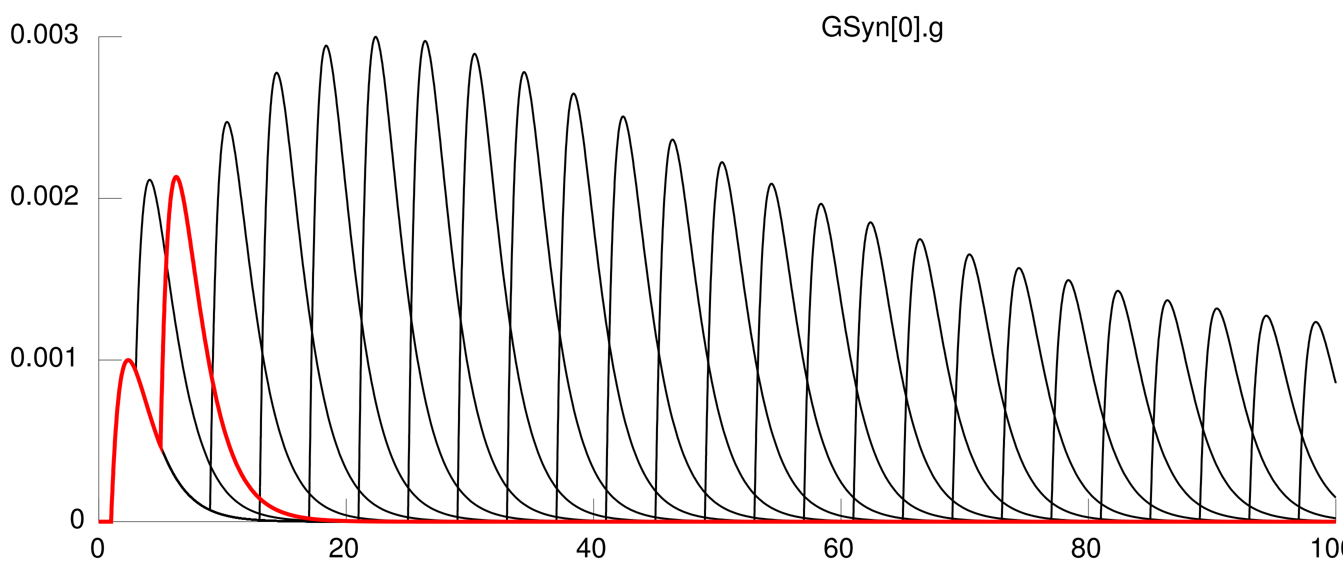
- **Synaptic events in "real" cells**

Artificial spiking cells




```
NEURON {
  POINT_PROCESS ExpSyn
  RANGE tau, e, i
  NONSPECIFIC_CURRENT i
}

...declarations...

INITIAL { g = 0 }
```

```
CHANNEL {
  SOLVE state METHOD cnexp
  i = g*(v - e)
}

DERIVATIVE state {
  g' = -g/tau
}

NET_RECEIVE(w) {
  g = g + w
}
```




```
CHANNEL {
  SOLVE state METHOD cnexp
  g = B - A
  i = g*(v - e)
}

DERIVATIVE state {
  A′ = -A/tau1
  B′ = -B/tau2
}
```

```
NET_RECEIVE(weight, w, G1, G2, t0) {
  INITIAL { w = 0  G1 = 0  G2 = 0  t0 = t }
  G1 = G1*exp(-(t-t0)/Gtau1)
  G2 = G2*exp(-(t-t0)/Gtau2)
  G1 = G1 + Ginc*Gfactor
  G2 = G2 + Ginc*Gfactor
  t0 = t
  w = weight*(1 + G2 - G1)
  A = A + w*factor
  B = B + w*factor
}
```

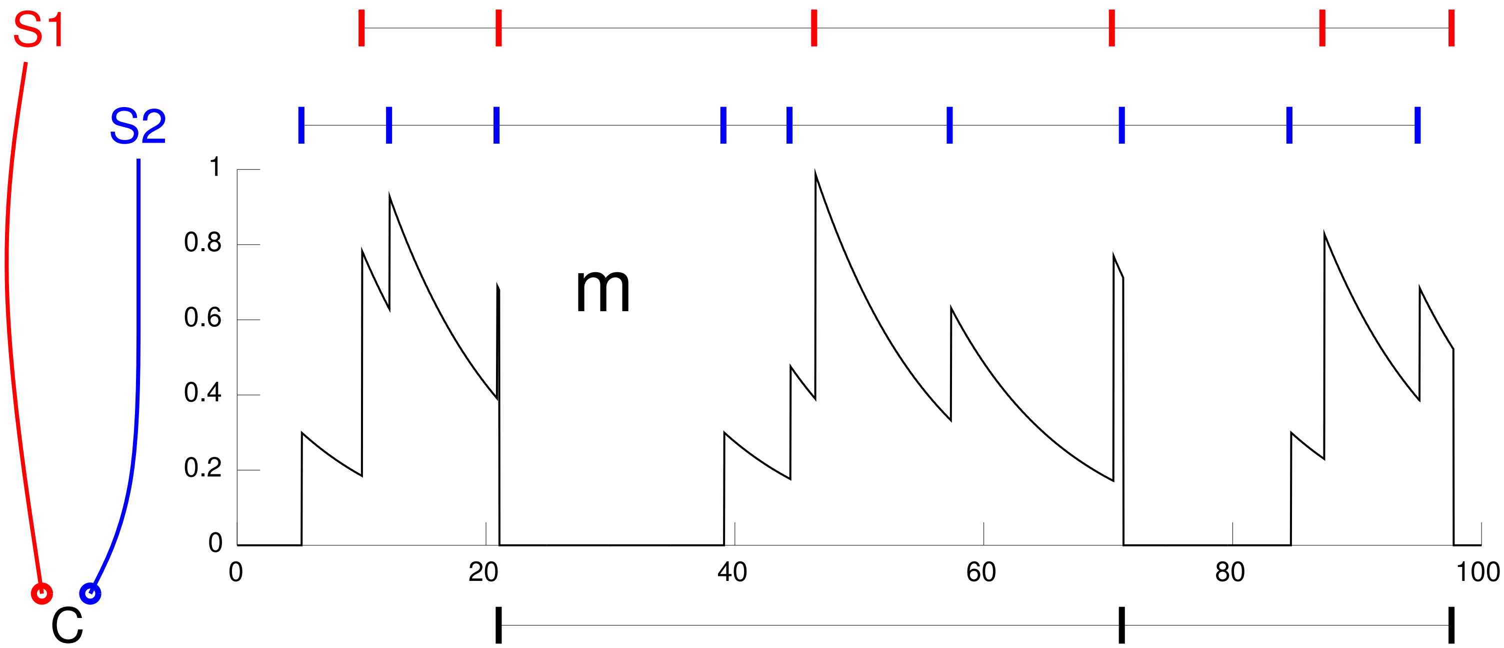
Single cell

Network

Local variable time steps

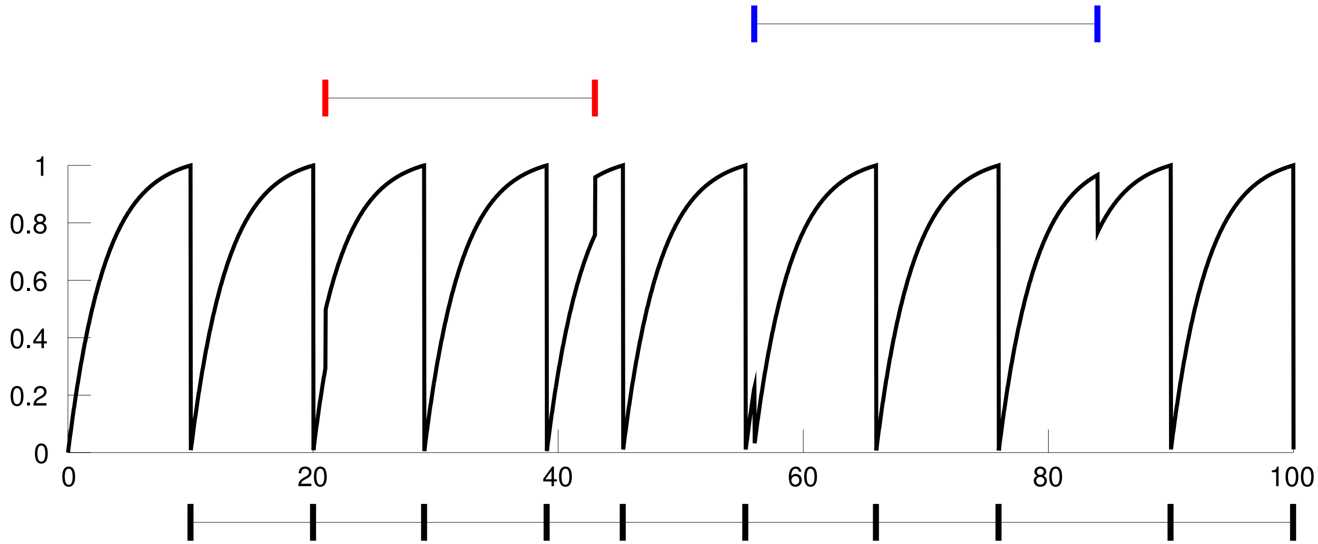
Synaptic events

- **Artificial spiking cells**




```
NEURON {
  ARTIFICIAL_CELL IntFire
  RANGE tau, m
}

...declarations...
```

```
INITIAL { m = 0   t0 = t }

NET_RECEIVE (w) {
  m = m*exp(-(t - t0)/tau)
  t0 = t
  m = m + w
  if (m > 1) {
    net_event(t)
    m = 0
  }
}
```




```
: dm/dt = (minf - m)/tau
: input event adds w to m
: when m = 1, or event
:  makes m >= 1, cell fires
: minf is calculated so
:  that the natural interval
:  between spikes is invl

INITIAL {
   minf = 1/(1 - exp(-invl/tau))
   m = 0
   t0 = t
   net_send(firetime(), 1)
}
```

```
NET_RECEIVE (w) {
   m = minf + (m - minf)*exp(-(t - t0)/tau)
   t0 = t
   if (flag == 0) {
      m = m + w
      if (m > 1) {
         m = 0
         net_event(t)
      }
      net_move(t+firetime())
   }else{
      net_event(t)
      m = 0
      net_send(firetime(), 1)
   }
}

FUNCTION firetime() { : m < 1 < minf
   firetime = tau*log((minf-m)/(minf - 1))
}
```

IntFire4