10-9. Today we continue our discussion of synapses and how to represent the synaptic conductance in models. So far we have discussed the case where we assume that meurotransmitter gets just one chance to bind to receptors in a short period after release from the vesicle and after that brief period it disappears (diffuses away, taken up by neuron of glia, or degraded by enzymes). With this simplification our simple transmitter-receptor binding reaction becomes $A + R \xleftarrow[k_{-1}]{} AR \underset{\alpha}{\overset{\beta}{\rightleftarrows}} AR_{open}$ or $AR \xrightarrow{\alpha_1} AR_{open} \xrightarrow{\alpha_2} A + R$.

In either case the mathematical solution for $AR_{open}(t)$ is either 1) a two exponential solution or 2) an alpha function solution ($\tau_1=\tau_2$).

Then $g_{syn}(t) = AR_{open}(t)$ * single channel conductance and $I_{syn}(t) = g_{syn}(t)(V - V_{syn})$

10-10. Is the synaptic input excitatory or inhibitory? How do we make this distinction? It depends on the relationship of V_{syn} and $V_{threshold}$. Does the synaptic input make it easier or more difficuly for a cell to fire an action potential? So the relationship is not simply whether V_{syn} is positive or negative to rest. A depolarizing conductance can be inhibitory if it tries to clamp the voltage below threshold. For example V_{Cl} may be -64 mV with rest at -70 mV. Then a chloride current will depolarize the cell, but will not allow it to reach threshold.

How do we determine V_{syn}? The way we do this is by voltage clamp. We voltage clamp the cell to various holding potentials, activate the synapse and observe the clamp current, as shown here. What is V_{syn} in this example? The current seems to reverse between +10 and -10 so V_{syn}~0 mV.

What are problems with this measurement? The biggest problem is the lack of space clamp. Mathematically we can estimate the extent of this problem. Consider the solution of the cable equation with BC of voltage clamp at x=0 and sealed end at the other end. What is the voltage at the far end for a clamp of 60 mV positive to rest at the soma end? Perhaps only 40 mV positive to rest! So if the synapse you are activating is at the end of the cable, you are not estimating the V_{syn} correctly because of the inadequate space clamp.

The solution to this problem is to study synapses that are very close to the soma, so that the space clamp problem is minimized.

10-11. NEURON has AlphaSynapse built in. It is contained in the syn.mod file. To use the alpha function representation for a synaptic input, you need to know the tau (tau = 1/alpha) and the maximum conductance.

In your script you would have a line

```
objectvar syn[2]
```

which declares 2 object variables which you will use to create synapses. Recall how you defined stim as an object variable before you could define a current input. Same thing here. The [2] means there are 2 variables created. However they will be referenced as [0] and [1]. The joys of having an indexing scheme starting at 0! These 2 object variables will be associated with point process alpha synapse objects.

If we have a process named dend, we can define the synapses and place them on this process as shown here.

We define `syn[0] = new ALphaSynapse(0.7)`
Which says syn[0] is an alpha synapse object to be placed at relative location 0.7.
If nseg=3, where does the synapse actually get placed? Recall how NEURON divides up sections into segments. It will be placed at location 0.83.

We had $AR_{open}(t)=2AR(0)\ \beta\ t\ \exp(-\alpha t)$ in one of our previous equations. The α here is $1/\tau$ and in our script we need to enter a value for τ. We also enter a value for the maximum conductance. We will see how this gmax value relates to our previous mathematics in the next overhead. The gmax gives the strength of the synapse. Here it is 5 nS. Finally we define the V_{syn} as e. Here $V_{syn}=0$.

In the second synapse our tau=3, gmax=10 nS and $V_{syn} = -85$ (probably a potassium). Note a tau=1 synapse will have a conductance that decays faster than a tau=3 conductance.

10-12. Here is the syn.mod file.

It defines a point process called AlphaSynapse with range variables (values you might want to change or plot) tau, gmax, e, i where i is the synaptic current, which we define as a non-specific current (as compared to a Na or K or Ca current).

In the parameter block we define the characteristics of the synaptic input.

In the Assigned block we give variables that may change

The Breakpoint block does the calculation. This particular implementation will assume the synaptic conductance is 0 before onset and after t is greater than 10 tau ($\exp(-t/\tau)$ is essentially 0 for $t/\tau > 10$).

The calculation here is $g=gmax*t/\tau\ \exp(1-t/\tau)$. This has a maximum at $t=\tau$ and the maximum is g_{max}. (Differentiate wrt t and set equal to 0, solve for t.)

So $g_{max}\exp(1)/\tau = 2\ AR(0)\beta$*single channel conductance from our previous analysis.

10-13. NEURON also has a two exponential conductance representation built in, Exp2Syn and this is also a point process available in the GUI. We can create these synapses by defining object variables as before. Then we create the synapses by
```
syn2[0] = new Exp2Syn(0.7)
```
as before. Here we have 2 time constants to define tau1 and tau2. tau1 < tau2. We also have the reversal potential e to define.

One problem is that as currently written, Exp2Syn requires a netcon event (network connection) where a weight is assigned (taking the place of gmax) and onset is determined by the network

connection, so this is not so useful for single neuron models. However we can write a .mod file to make an Exp2Syn that can be easy to put into single cell models

10-14 and 10-15. Here is the exp2syn.mod file

As before the point process is defined as Exp2Syn and variables tau1, tau2 and e and i are defined as rnage variables, also g

Tau1 and tau2 and e have their values set here.

In the assigned block we also have something called factor.

In the initial block we initialize the factor and total and A and B to starting values. The factor is defined in terms of tp, a local variable which defined the time of the peak of the function as determined by the tau1 and tau2. tp is the maximum of the function $\exp(-t/\tau 2) - \exp(-t/\tau 1)$. To verify this for yourself, take the derivative, set it to 0 and solve for t. The result is tp as defined here. Factor is the value of this function at time tp, and 1/factor is the normalization applied so that the peak is 1. Then the synaptic weight determines the maximum synaptic conductance.

The derivative block defines differential equations to solve for this conductance. These expressions have been optimized and are not what you might normally write. Recall our equations were

$dAR/dt = -\alpha_1 AR$
$dAR_{open}/dt = \alpha_1 AR - \alpha_2 AR_{open}$

if $A = AR$ and $B=AR_{open}$ and $\alpha=1/\tau$ then our equations become
A' = -A/tau1 and B' = A' – B/tau2

They have let B' = -B/tau2 and then have g=B – A

Factor makes the function have a peak at 1 and then the weights are applied to scale the maximum conductance.

10-16. How could we make a 2-exponential synapse that did not require network connections? We could start with the syn.mod file for the AlphaSynapse and modify it to get a 2-exponential synapse. How would we do this?

In the NEURON block we could change the name of the point process to Exp2p for example and replace tau with tau1 and tau2

In the parameter block we would also replace tau with tau1 and tau2

In the assigned block we would add factor

In the breakpoint block we would replace the alpha function with

```
factor*(exp(-t/tau2)-exp(-t/tau1))
```

Then we would need an INITIAL block to define factor with a particular set of tau1 and tau2. Here we borrow code from Exp2Syn

```
INITIAL {
	LOCAL tp
	tp=(tau1*tau2)/(tau2-tau1)*log(tau2/tau1)
	factor=exp(-tp/tau2)-exp(tp/tau1)
	factor=1/factor
}
```

I have this spelled out on **10-16a.**

10-17. What are some issues with using α-functions or 2 exponentials to represent synaptic inputs? We have talked about these already. There are some constraints on the rate constants that may or may not hold, at least with the alpha function, but these are relaxed for the 2-exponential formulation.

First, it may be the case that the kinetic scheme is not the simple one we used to derive the alpha function or the 2 exponential function. What is the actual kinetic scheme? Both acetylcholine and glutamate receptors need 2 molecules of acetylcholine or glutamate to reach the open state as shown in this configuration. Not only this but glutamate has a number of desensitized states. There may be additional closed states that have to be considered. How should we deal with these complications. It is possible to write a .mod file that handles these specific cases, although you might find that for most purposes a 2-exponential form will be quite satisfactory.

Second, is there receptor saturation? For acetylcholine at the NMJ, probably not. For glutamate and AMPA and NMDA receptors where a vesicle has maybe 2000-3000 glutamate molecules and there are 100 AMPA and 30 NMDA receptors, say, there may not be saturation with one input, but with high frequency trains of input, saturation effects may be felt. This may have to be dealt with separately. As we have coded them, conductances sum.

Third, what about voltage dependency of channel opening. We know the NMDA receptors experience voltage dependent block by Mg. If these channel open, they are blocked at voltages near rest, but this block is relieved with depolarization. How do we account for this? We could attach another rate onto our scheme above, or we could apply a voltage block function. If the block function is a sigmoid that resembles our activation curves from before, where block=0 at hyperpolarized voltage and block=1 at depolarized voltages, we could have g=gmax*block*… Such block functions are available and are used for this purpose with NMDA conductances.

10-18. What happens when the number of receptors is very small? In these cases we may have to apply a stochastic approach. A freely available software package for doing stochastic simulations is MCell available from http://www.mcell.cnl.salk.edu This authors are Bartol and Stiles.

With this software you can specify synapse geometry, receptor density, uptake density and release sites as you wish. Each molecule of neurotransmitter is followed as a random walk in 3-D space of the synaptic cleft and it bumps into and binds to receptors (according to a kinetic scheme that you specify).

A few years ago a graduate student in my lab compared stochastic simulations of NMDA channel openings with deterministic simulations and results are as shown here. You see the stochastic simulation results moving in stepwise fashion as a channel opens or closes. Ten stochastic simulations are averaged for the solid noisy line. The deterministic solution is the dashed line. Different stochastic runs will produce different results, but the average of many stochastic runs should match the deterministic result. What this shows is the large variability in channel openings of NMDA channels you might have at a synapse with any one vesicle release event.

10-19. Finally what determines the time course of an EPSP? We discussed much of this last quarter, so I don't want to repeat myself too much!

Some of the factors that affect the time course of an EPSP are
1) the kinetics (or the taus in AlphaSynapse and Exp2Syn),
2) the membrane time constant (note the conductance is independent of the membrane time constant, but the resulting voltage change is affected by it)
3) decay to the soma as influenced by the space constant and cell geometry
4) the amplitude of the synaptic conductance gmax, and also the local input resistance. However for very fast synapses, only Ra and Cm affect the peak of the voltage. For longer synaptic conductance time courses, Rm will begin to play a role in the peak, and of course, the steady-state peak depends on Rm (RN)

Spatial summation, will it be linear or non-linear? Why?
Two inputs on different branches will tend to sum linearly, but two inputs on the same branch will tend to sum non-linearly. The reason has to do with whether or not one input will affect the voltage, and hence the driving force, of the other synapse. Synapses separated spatially will not affect each other's driving force ($V-V_{syn}$) but synapses close to each other will.

Temporal summation—when will it occur? It will occur when inputs occur closely together in time. How close do inputs have to be in time to sum? Depends on the membrane time constant. Inputs will sum better when tau is large.

If there is excitation and inhibition on the same segment, what is the net effect? We just add them $I_{syn} = g_e(V - V_e) + g_i(V - V_i) = (g_e+g_i)V - (g_eV_e+gV_i) = (g_e+g_i)\,[V - (g_eV_e+g_iV_i)/(g_e+g_i)]$
And this last term $(g_eV_e+g_iV_i)/(g_e+g_i)$ is just the new V_{syn}. However you will also have to consider the time courses which will be different for the different types of synapses.