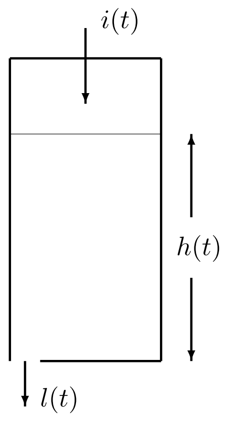
Leaky buckets

These notes describe the modelling of a leaky bucket of water, something that is useful to think about as an analogue of how neurons behaeve.

Buckets of water

In the simplest model of neurons their voltage dynamics is similar to the dynamics of a bucket with a leak and the class of equations that apply in this case will also be applied to synapses, for example.



Consider a bucket with straight sides which is filled to a height h with water. Imagine the water leaks out of a hole in the bottom. The rate the water leaks out depends on h; the larger h is the larger the pressure at the bottom is and hence the faster the water pours out. In other words

$$l(t) \propto h(t) \tag{1}$$

or

$$l(t) = Gh(t) \tag{2}$$

where G is a constant which will depend on the size of the hole and complicated things like the viscosity of water. Of course, we are also ignoring lots of complicated stuff, like turbulence and so forth, but since we are interested in the equation rather than the amount of water in a bucket, this is fine. Imagine water also pours in the top at a rate $i(t)$. This means the total rate of change of the amount of water is $i(t) - Gh(t)$.

Now, $h(t)$ is the height of the water not the volume: the volume is $Ch(t)$ where C is the cross-sectional area of the bucket. The rate of change of the volume is therefore

$$\frac{dCh(t)}{dt} = i(t) - Gh(t) \tag{3}$$

or

$$\frac{dh}{dt} = \frac{1}{C}(i - Gh) \tag{4}$$

Let's solve this equation for constant i before going on to look at neurons. Probably the best way to do this is by using an integrating factor, let $\tau = C/G$ and $\tilde{\imath} = i/G$

$$\tau\frac{dh}{dt} + h = \tilde{\imath} \tag{5}$$

then we multiply across by $\exp t/\tau$ and divide by τ

$$e^{t/\tau}\frac{dh}{dt} + \frac{1}{\tau}e^{t/\tau}h = \frac{\tilde{\imath}e^{t/\tau}}{\tau} \tag{6}$$

Now we can rewrite the left hand side using the product rule

$$\frac{d}{dx}(uv) = u\frac{dv}{dx} + v\frac{du}{dx} \tag{7}$$

to give

$$\frac{d}{dt}\left(e^{t/\tau}h\right) = \frac{\tilde{\imath}e^{t/\tau}}{\tau} \tag{8}$$

Now integrating both sides gives

$$e^{t/\tau}h = \tilde{\imath}e^{t/\tau} + A \tag{9}$$

where A is an integration constant. This gives

$$h = Ae^{-t/\tau} + \tilde{\imath} \tag{10}$$

and putting $t = 0$ shows $A = h(0) - \tilde{\imath}$ so

$$h(t) = [h(0) - \tilde{\imath}]e^{-t/\tau} + \tilde{\imath} \tag{11}$$

so, basically, the value of h decays exponentially until it equilibriates with $\tilde{\imath}$.

These dynamics make good intuitive sense; the more water there is in the bucket, the higher the pressure will be at the leak and the quicker the water will pour out. If there is just the right amount of water the rate the water pours out the leak will precisely match the rate it pours in, this is the equilibrium. If there is more water than required for equilibrium it will pour out faster than the flow coming in, if there is less, it will pour out slower. Either way, as time passes the height of the water will reach the equilibrium. The plot in Fig. 1 illustrates this.

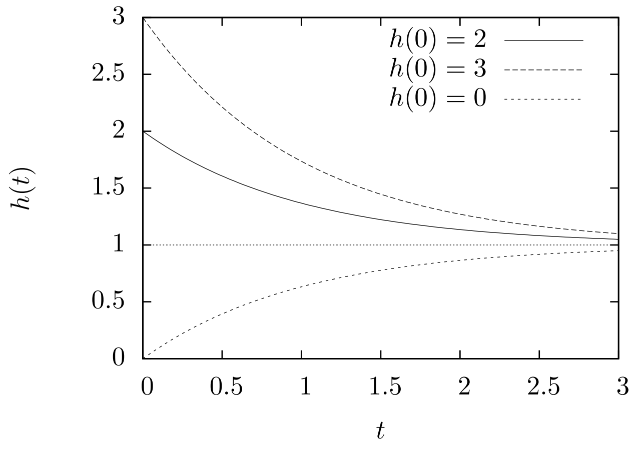


Figure 1: Exponential relaxation. The dynamics described by the 'bucket equation' are very common. Here $h = [h(0) - \tilde{\imath}]\exp(-t/\tau) + \tilde{\imath}$ is plotted with $\tilde{\imath} = 1$, $\tau = 1$ and three different values of $h(0)$. $h(t)$ relaxes towards the equilibrium value $\tilde{\imath} = 1$, the closer it gets, the slower it approaches.

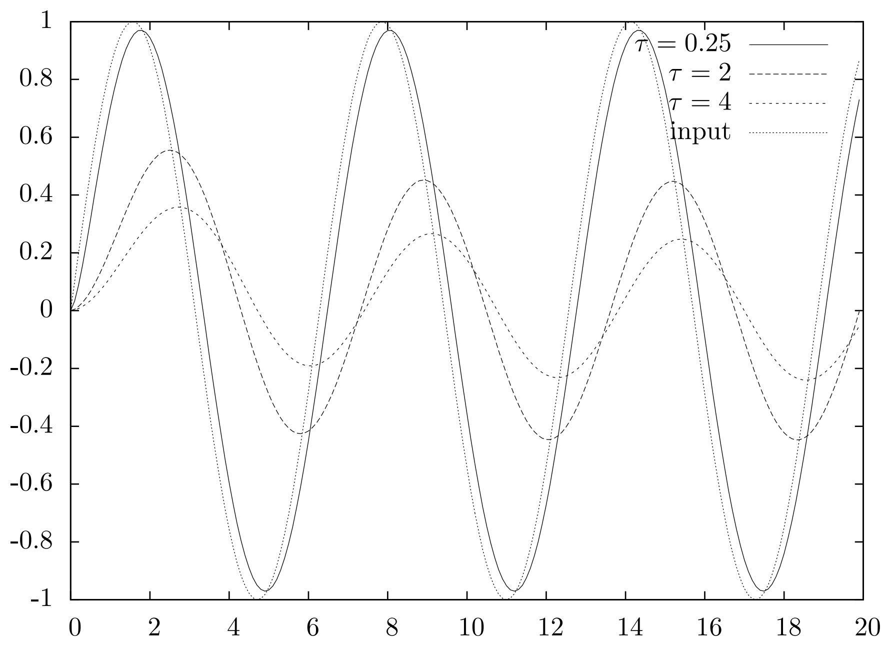


Figure 2: Variable input. Here the input is a sine wave $\tilde{\imath} = \sin t$ and the equation is evolved with $h(0)$ and three different τ values. For $\tau = 0.25$ we see $h(t)$ closely matches the input whereas for larger τ it is smoother and lags behind.

Variable input

We have only discussed constant inputs; the variable input case where i depends on time is harder and although it can sometimes be solved it is often easier just to compute it numerically, we will look briefly at how to do this, but first note that the effect of variable input is that the solution kind of chases the input with a timescale set by τ. For very small τ is chases it quickly, so it is close to the input, but for large τ it lags behind it and smooths it out. This is sometimes described by saying that it *filters* the input. There is an illustration in Fig. 2.