High Frequency Oscillations in Modeled Electrotonically Coupled Inferior Olive Neurons

Jason McInerney
BGGN260 W07

Abstract

The cerebellum, important in learning and movement, receives input from the inferior olive (IO), a nucleus of cells which support subthreshold oscillations and synchronous spiking. Ionic models of these cells have uncovered key kinetic parameters directing the frequency and synchrony within individual IO neurons. Experimentally, electronic coupling between these cells has been shown to affect the synchrony and overall output of individual IO neurons and larger networks within the IO. In an effort to uncover the role in learning that the IO may play, a mathematical model is developed to view the effects of ionic and kinetic parameter control of oscillations in and among electronically coupled IO neurons. The small adjustment of the slow inactivation kinetics of an inwardly rectifying potassium channel, in conjunction with a tuning parameter, offers a powerful mechanism for adjusting the onset of oscillations, their frequency, and that of subsequently connected neurons.

Introduction

A core component in learning and executing coordinated movement is the cerebellum. The inferior olive (IO) nucleus supplies input to the cerebellum via climbing fibers from neurons which have the ability to sustain subthreshold oscillations in membrane potential, both as individual neurons and together in networked groups of cells. The role of IO neurons in learning has yet to be established but may reside in the spatiotemporal dynamics of these oscillations in IO circuit input to the cerebellum.

A small number of IO neuron models have been established showing experimental agreement with both the kinetic parameters leading to subthreshold oscillations and their affect on spiking and stability (1-4). The rhythmic nature of the oscillations provides opportunity for synchronizing or desynchronizing multiple cells if electronically coupled, and thus a mechanism for altering the probability of action potential firing or delivery of information to the cerebellum in general.

Electronic coupling between IO cells has been shown to contribute to the synchronizing of separately asynchronous IO cells (4-7), the blocking of which directly affects the recipient Purkinje cell spiking behavior (6,7). Larger IO ensemble oscillations have been witnessed arising from clusters of 100+ coupled IO neurons (4). It is clear that the status of connectivity when combined with the oscillatory behavior and the ability to synchronize can strongly impact the eventual IO output signal.

In order to understand the role electronic coupling has in modulating IO output, and subsequent possible implications for cerebellar learning, it is necessary to combine the known ionic and dynamical bases for IO neuronal subthreshold oscillations with effects of electrical coupling.

Here a mathematical model of multiple electronically coupled IO neurons is utilized to investigate the effects of key kinetic parameters on oscillation and synchrony among the individual cells and as a group.

Methods

Currents and parameters consistent with experimental data and previous models were used build a MATLAB model of an IO neuron. The membrane potential for each cell is given as:

$$C_m \frac{dV}{dt} = -(I_L + I_{NaP} + I_{Ks} + I_{Na} + I_{Kd} + I_c) + I_{Inp}$$

Where the currents are as follows:

$$I_l = \bar{g}_l (V - E_l)$$

$$I_{Ks} = \bar{g}_{Ks} m\left(\rho h_1 + h_2(1-\rho)\right)(V - E_K) \text{ where}$$

$$m_\infty = \frac{1}{1+e^{\frac{-(V+34)}{6.6}}}$$

$$\tau_{m_\infty} = 50ms$$

$$h_{1_\infty} = \frac{1}{1+e^{\frac{-(V\ +\ -65)}{6.6}}}$$

$$\tau_{h_{1_\infty}} = 200 + 200\left(\frac{1}{1+e^{\frac{-(V\ +\ 71.6)}{6.85}}}\right)$$

$$h_{2_\infty} = \frac{1}{1+e^{\frac{-(V\ +\ -65)}{6.6}}}$$

$$\tau_{h_{2_\infty}} = 200 + 3200\left(\frac{1}{1+e^{\frac{-(V\ +\ 63.6)}{4}}}\right)$$

$$\dot{m} = \frac{m_\infty - m}{\tau_{m_\infty}}$$

$$\dot{h}_1 = \frac{h_{1_\infty} - h_1}{\tau_{h_{1_\infty}}}$$

$$\dot{h}_1 = \frac{h_{2_\infty} - h_2}{\tau_{h_{2_\infty}}}$$

$$I_{NaP} = \bar{g}_{NaP} n_\infty (V - E_{Na}) \text{ where}$$

$$n_\infty = \frac{1}{1+e^{\frac{-(V+51)}{5}}}$$

$$I_{Na} = \bar{g}_{Na} m^3 h (V - E_{Na}) \text{ where}$$

$$\alpha_m = q * 0.1 \frac{V + 30 - \sigma}{1 - e^{\frac{-0.1(V+30-\sigma)}{6.6}}}$$

$$\beta_m = q * 4e^{\frac{-V-55+\sigma}{18}}$$

$$\alpha_h = q * 0.7e^{\frac{-V-44+\sigma}{20}}$$

$$\beta_h = q * \frac{1}{1+e^{-0.1(V+14-\sigma)}}$$

$$\dot{m} = \alpha_m (1-m) - \beta_m m$$

$$\dot{h} = \alpha_h (1-h) - \beta_h h$$

$$I_{Kd} = \bar{g}_{Kd} n^4 (V - E_K) \text{ where}$$

$$\alpha_n = q * -0.1 \frac{V + 34 - \sigma}{e^{\frac{-0.1(V+34-\sigma)}{6.6}} - 1}$$

$$\beta_n = q * 0.125 e^{\frac{-V+44-\sigma}{80}}$$

$$\dot{n} = \alpha_n (1-n) - \beta_n n$$

$$I_c = \bar{g}_c \sum_i \left(V - V_i\right)$$

Table 1

Conductance	**Value (mS/cm^2)**	**Potential**	**Value (mV)**
$\bar{g}_l$	0.1	E_l	-60
$\bar{g}_{Ks}$	14	E_K	55
$\bar{g}_{Na}$	52	E_{Na}	-90
$\bar{g}_{NaP}$	0.1		
$\bar{g}_{Kd}$	40		
$\bar{g}_c$	variable		

Table 1 shows the values for conductances and reversal potentials used in the model.

The full MATLAB code is available as three separate programs in Appendix A.

For this analysis, all neurons modeled were identical in their kinetic parameters. Table 2 provides the conditions for all simulations in this study. When multiple runs were performed sequentially, $\bar{g}_c$ was incrementally increased by 1.

Table 2

Simulation number	**total time (ms)**	**current start time**	**current end time**	**current magnitude (mA)**	$\bar{g}_c$ **initial value**	**number of neurons**	τ_{m_∞}	σ
1	8000	5000	5500	10	0	1	1	1
2	20000	5000	15000	40	0.2	1	1	1
3	8000	5000	5500	10	0.2	4	1	1
4	20000	5000	5500	10	0.2	4	1	1
5	8000	1000	4000	8	0	4	1	2
6	8000	1000	4000	4	0	1	1	1.5
7	8000	1000	4000	4	0	1	10	1
8	8000	1000	4000	4	0	1	5	1
9	8000	1000	4000	4	0	1	8	1
10	8000	1000	4000	4	0	1	1	1
11	8000	1000	4000	8	0	4	1	1

Results

For all the simulations where τ_{m_∞} and σ are held at 1, the neurons displayed fluctuating potentials for the first second, eventually settling to a resting potential (example in figure 1). Each neuron responded to injected current with a spike and the kinetic parameters responded as expected. In cases with more than one neuron, synchronization occurred at the point the membrane voltage reached resting potential. The phases of the neurons during the initial asynchronization changed in response to increasing $\bar{g}_c$, but still settled to a synchronized state. Increasing current increased the response with no other effect.

However, when the timing constant τ_{m_∞} for the K_s channel is altered or σ is altered, the responses of the neurons change drastically. Extremely high frequency oscillations appear in all neurons with small changes in σ or τ_{m_∞} (figure 2). The h_{1_∞} parameter is particularly sensitive to these changes (figure 2b). The synchrony of these oscillations among a group of neurons is dependent on $\bar{g}_c$.

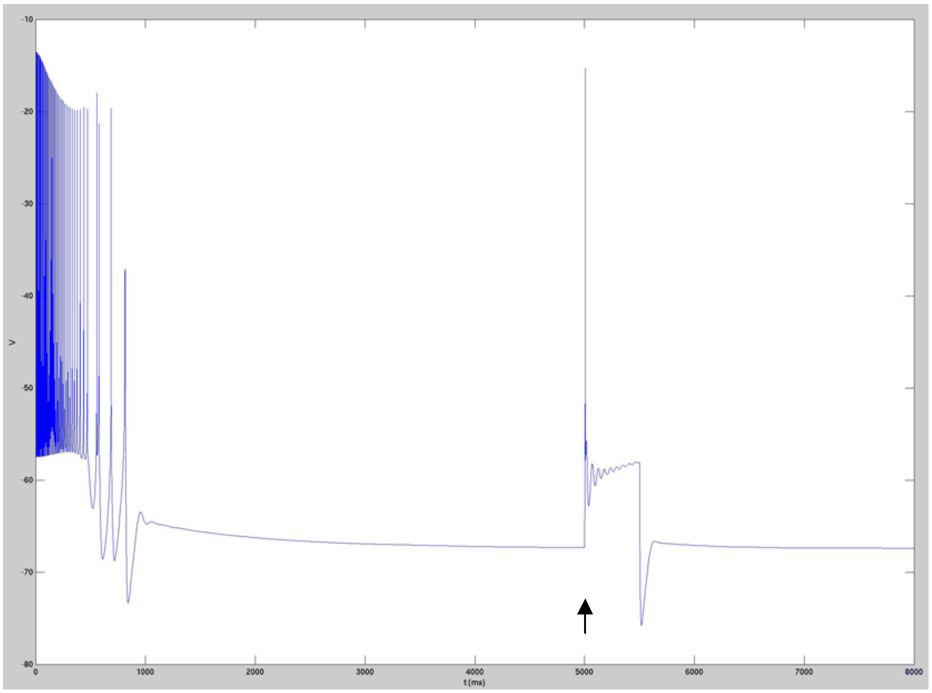

Figure 1
Example neuron voltage over time. Note the initial fluctuation and the oscillation upon current injection (arrow).

Conclusions

The frequency of the oscillations in response to differing values of τ_{m_∞} or σ is orders of magnitude higher than any models or recordings reported in the literature. There are at least two potential reasons for this result. The most likely explanation is that the channels and their parameters need to be modified in some way not clearly reported by other studies. However, it could be that conditions in this model have not been previously tested. There has been little investigation into what the parameters τ_{m_∞} and σ correspond to physically. Oscillations at these frequencies provide an opportunity to deliver massive amounts of information in short bursts and increase the capacity for synchrony and asynchrony to store and retrieve information. Cleary more research is needed to discover whether these oscillations are a result of incorrect modeling or if they might correspond to some physical process not yet evoked experimentally or in other simulations.

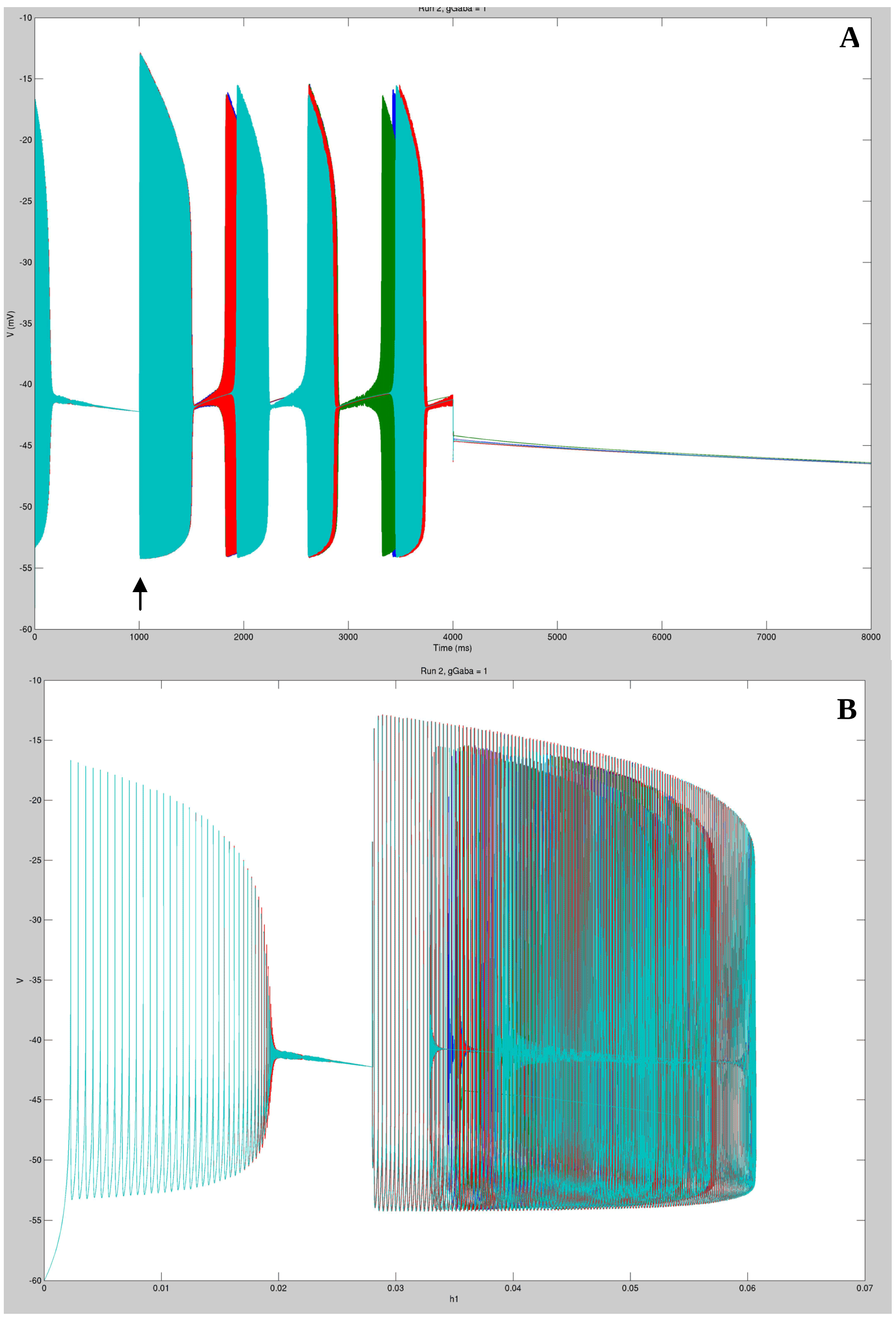

Figure 2
Example of four neurons and their high-frequency oscillations. a) All four, membrane potential over time. b) All four, plotting membrane versus h_{1_∞}. Note the initial synchrony and later asynchrony.

References

1. Manor, Y., Rinzel, J., Segev, I., Yarom, Y. (1997) "Low-Amplitude Oscillations in the Inferior Olive: A Model Based on Electrical Coupling of Neurons with Heterogeneous Channel Densities". *J. Neurophysiol.* **7**:2736-2752.

2. Schweighofer, N., Doya, K., Kawato, M. (1999) "Electrophysiological Properties of Inferior Olive Neurons: A Compartmental Model". *J. Neurophysiol.* **82**:804-817.

3. Velarde, M.G., Nekorkin, V.I., Kazantsev, V.B., Makarenko, V.I., Llinas, R. (2001) "Modeling inferior olive neuron dynamics". *Neural Networks.* **15**:5-10.

4. Leznik, E., Makarenko, V., Llinás, R. (2002) "Electrotonically Mediated Oscillatory Patterns in Neuronal Ensembles: An *In Vitro* Voltage-Dependent Dye-Imaging Study in the Inferior Olive". *J. Neuroscience*. **22**:2804-2815.

5. Devor, A., Yarom, Y. (2002) "Electrotonic Coupling in the Inferior Olivary Nucleus Revealed by Simultaneous Double Patch Recordings". *J. Neurophysiol.* **87**:3048-3058.

6. Leznik, E., Llinás, R. (2005) "Role of Gap Junctions in Synchronized Neuronal Oscillations in the Inferior Olive". *J. Neurophysiol.* **94**:2447-2456.

7. Blenkinsop, T.A., Lang, E.J. (2006) "Block of Inferior Olive Gap Junctional Coupling Decreases Purkinje Cell Complex Spike Synchrony and Rhythmicity". *J. Neuroscience*. **26**:1739-1748.

8. Wang, X.-J. (1993) "Ionic basis for intrinsic 40 Hz neuronal oscillations". *NeuroReport*. **5**:221-224.

APPENDIX A

```
function WM(end_time,I_begin,I_end,I_ext,g_gaba_init,total_runs,n_tot,fig)

% Inferior olive model neurons, multiple
% Jason McInerney, jasonm@salk.edu

% the following are passed via arguments
% total_runs = 1;   % number of runs
% n_tot = 4;        % number of neurons
% fig = 300;        % starting number of figures
% end_time = 20000; % (ms)
% I_begin = 5000;   % (ms)
% I_end = 15000;    % (ms)
% I_ext = 100;      % micA/cm^2

C = 1.0;      % (micF/cm^2)
gNabar = 52;  % mS/cm^2
glNabar = 0.1;
gNaPbar = 0.1;
gKdbar = 40;
gKsbar = 14;

eNa = 55;   % sodium rest potential (mV)
eK = -90;   % potassium
elNa = -60; % leak

g_gaba = g_gaba_init; % GABAa synapse conductance (mS/cm^2)
b_r = 0.18;           % backward rate contstant (ms^-1)

time_vector = 0:0.1:end_time;
tspan= [0 end_time];

%apply the stimulus only at specified time frame
I_vector = ((time_vector > I_begin) & (time_vector < I_end)) * I_ext * rand(1);
% I_vector gives the injected current for each time point

tot = 0;

% loop through the number of runs
for i=0:total_runs-1
    run_cnt = i + 1
    tic
    g_gaba = g_gaba_init + i;

% loop through all neurons
    for j=1:n_tot
        % initial values
        init(j*8-1) = -60; % V
        init(j*8-2) = 0; % mNa
        init(j*8-3) = 0; % hNa
        init(j*8-4) = 0; % nKd
        init(j*8-5) = 0; % mKs
        init(j*8-6) = 0; % h1 (Ks)
        init(j*8-7) = 0; % h2 (Ks)
        init(j*8) = 0; % r
        % parameters put into a single vector, p
        p(j*11-10) = gNabar;
        p(j*11-9) = gNaPbar;
        p(j*11-8) = gKdbar;
        p(j*11-7) = gKsbar;
        p(j*11-6) = glNabar;
        p(j*11-5) = eNa;
        p(j*11-4) = eK;
        p(j*11-3) = elNa;
        p(j*11-2) = C;
        p(j*11-1) = g_gaba;
        p(j*11) = b_r;
    end;
    [t,y]=ode15s(@WM_ode,tspan,init,[],p,time_vector,I_vector,n_tot,curfile);
    rundata(:,:,run_cnt) = y;
    timedata(:,run_cnt) = t;
    toc
    tot = tot + toc
end
% save all data for later manipulation
save rundata.mat rundata;
save timedata.mat timedata;



function y = WM_ode(t,z,p,time_vector,I_vector,n_tot,curfile)

y_sv = zeros(n_tot*8);

for j=1:n_tot
    rho = 0.6;   % for Ks
    sigma = 1;   % for tuning thresholds
    q10 = 200/7; % temperature dependence

    Iinp = interp1(time_vector,I_vector,t);

    V = z(j*8-1);
    mNa = z(j*8-2);
    hNa = z(j*8-3);
    nKd = z(j*8-4);
    mKs = z(j*8-5);
    h1 = z(j*8-6);
    h2 = z(j*8-7);
    r = z(j*8);

    Vpre = V;
    if (j*8 + 1) > n_tot*8
        Vpost = Vpre;
    else
        Vpost = z(j*5 + 1);
    end

    %parameters
    gNabar = p(j*11-10);
    gNaPbar = p(j*11-9);
    gKdbar = p(j*11-8);
    gKsbar = p(j*11-7);
    glNabar = p(j*11-6);
    eNa = p(j*11-5);
    eK = p(j*11-4);
    elNa = p(j*11-3);
    C = p(j*11-2);

    %differential equations

    % GABAa inhibitory synapse
    g_gaba = p(j*11-1); % GABAa synaptic conductance
    b_r = p(j*11);      % backward rate constant (mHz)
    Ecl = 70;           % Cl reversal potential (mV)
    Tmax = 1.5;         % max [neurotransmitter] (mM)

    Kp = 5;  % (mV)
    Vp = 72; % (mV)
    a_r = 5; % (mHz/mM)

   %Na
    amNa = q10*0.1*(V+30-sigma) / (1- exp(-0.1*(V+30-sigma))); % alpha for Na m, fast activation
    bmNa = q10*4*exp((-V-55+sigma) / 18); % beta for Na m
    mNadot = amNa*(1-mNa) - bmNa*mNa;

    ahNa = q10*0.07*exp((-V-44+sigma)/20);     % alpha for Na h, inactivation
    bhNa = q10*1 / (1+exp(-0.1*(V+14-sigma))); % beta for Na h
    hNadot = ahNa*(1-hNa) - bhNa*hNa;
   %NaP
    mNaP = 1 / (1+(exp(-(V + 51) / 5))); %stead-state

   %Kd
    anKd = q10*-0.01*(V+34-sigma) / (exp(-0.1*(V+34-sigma))-1);
    bnKd = q10*0.125*exp(-(V+44-sigma) / 80);
    nKddot = anKd*(1-nKd) - bnKd*nKd;

   %Ks
    mKstau = 1; %may be important in determining oscillation properties
    mKsinf = 1 / (1+(exp(-(V + 34) / 6.6)));
    h1tau = 200+220*(1 / (1+(exp(-(V + 71.6) / 6.85))));
    h1inf = 1 / (1+(exp(-(-V + -65) / 6.6)));
    h2tau = 200+3200*(1 / (1+(exp(-(V + 63.6) / 4))));
    h2inf = 1 / (1+(exp(-(-V + -65) / 6.6)));
    mKsdot = (mKsinf - mKs) / mKstau;
    h1dot = (h1inf - h1) / h1tau;
    h2dot = (h2inf - h2) / h2tau;

    INa = gNabar * mNa^3 * hNa * (V - eNa);
    INaP = gNaPbar * mNaP * (V - eNa);
    IKd = gKdbar * nKd^4 * (V - eK);
    IKs = gKsbar * mKs * ((rho * h1) + (1 - rho) * h2) * (V - eK);
    IL = glNabar * (V - elNa);
    Igaba = g_gaba * r * (Vpost - Ecl);  % GABAa synapse current

    T = Tmax / (1+exp(-(Vpre - Vp) / Kp)); % [neurotransmitter] in cleft
    rdot = a_r * T * (1-r) - b_r * r; % fraction of open receptors

    Vdot = (-IL - INaP - IKs - INa - IKd - Igaba + Iinp) / C;

    y_sv(j*8-1) = Vdot;
    y_sv(j*8-2) = mNadot;
    y_sv(j*8-3) = hNadot;
    y_sv(j*8-4) = nKddot;
    y_sv(j*8-5) = mKsdot;
    y_sv(j*8-6) = h1dot;
    y_sv(j*8-7) = h2dot;
    y_sv(j*8) = rdot;

end;

for j=1:n_tot*8
    if (j == 1)
        y = y_sv(1);
    else
        y = [y; y_sv(j);];
    end
end
```