

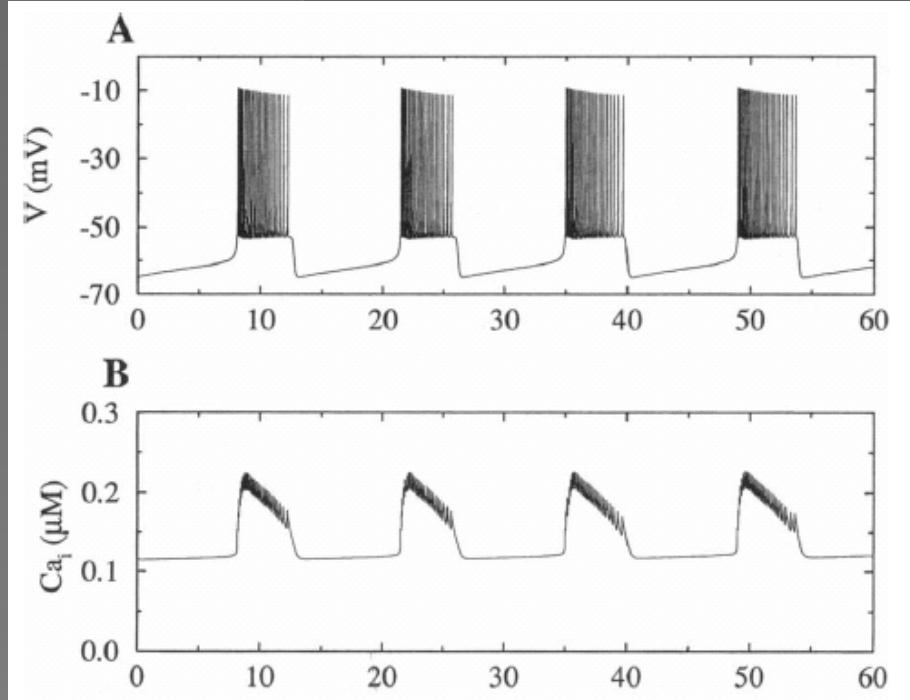
Model curation: the CellML approach

Catherine Lloyd
Auckland Bioengineering Institute

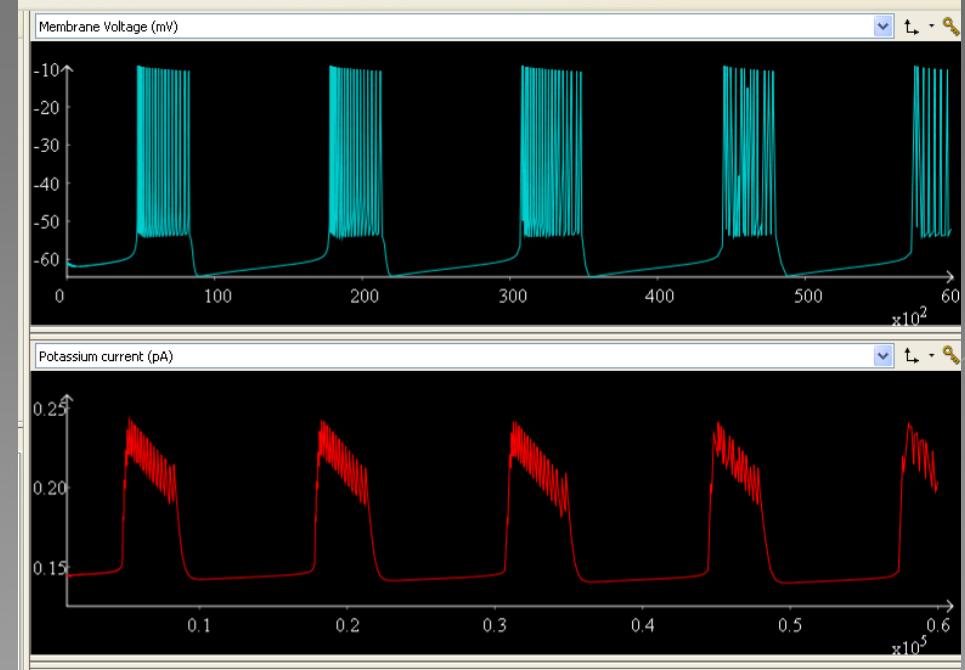




What is curation?



model validation





Publishing a model

model creation

```
current-----  
  
%Ca  
[CfCa,RevPCa]= CalcConstantfield(Cai,Cao,2, Vm);  
[CfK,RevPK] = CalcConstantfield(Ki,Ko,1, Vm); %K  
[CfNa,RevPNa] = CalcConstantfield(Nai,Nao,1,  
Vm); %Na  
if (count ==1 && currenttime == 0)  
    Va = -74.0078;  
else  
    Va = Vm;  
end  
if (count ==0)  
    [mcal, hcal,n] = calcRateConst(1,Va,  
0,Cai,mcal,hcal,count,dt); %Calc m and h  
    ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;  
    ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;  
    %ICaLCa = (PCAL * CfCa*mcal*hcal); %original  
    ICaLCa = (PCAL * CfCa*mcal*hcal);  
    ICaL = ICaLCa + ICaLK+ICaLNa;  
else  
    ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;  
    ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;  
    ICaLCa = (PCAL * CfCa*mcal*hcal);  
    ICaL = ICaLCa + ICaLK+ICaLNa;  
    [mcal, hcal] = calcRateConst(1,Va,  
0,Cai,mcal,hcal,count,dt); %Calc m and h  
end
```

Publishing a model

which cotransporters are to be included in the model. Note that eqs. (3) imply that the total number of Cl^- and HCO_3^- ions and the sum of the number of K^+ and Na^+ ions in the ECS and the astrocyte are conserved at any time. Ion flux through KCC1, NKCC1^[52] and NBC^[45, 53] is modeled in a Nernst-like fashion, i.e.

$$(5) \quad J_{\text{KCC1}} = \frac{g_{\text{KCC1}}}{F} \frac{RT}{F} \ln \left(\frac{[\text{K}^+]_o [\text{Cl}^-]_o}{[\text{K}^+]_i [\text{Cl}^-]_i} \right),$$

$$(6) \quad J_{\text{NBC}} = \frac{g_{\text{NBC}}}{F} [V_m - E_{\text{NBC}}],$$

$$(7) \quad J_{\text{NKCC1}} = \frac{g_{\text{NKCC1}}}{F} \frac{RT}{F} \ln \left(\frac{[\text{Na}^+]_o [\text{K}^+]_o}{[\text{Na}^+]_i [\text{K}^+]_i} \left(\frac{[\text{Cl}^-]_o}{[\text{Cl}^-]_i} \right)^2 \right).$$

Here, g_{NKCC1} , g_{KCC1} and g_{NBC} are the conductances per unit area for the NKCC1, the KCC1 and NBC cotransporter, respectively. The reversal potential of NBC is

$$(8) \quad E_{\text{NBC}} = \frac{RT}{z_{\text{NBC}} F} \ln \left(\frac{[\text{Na}^+]_o [\text{HCO}_3^-]_o^2}{[\text{Na}^+]_i [\text{HCO}_3^-]_i^2} \right),$$

where z_{NBC} is the effective valence of the NBC cotransporter complex, here taken to be -1, setting $z_{\text{NBC}} = -(n - 1) = -1$ where n is the stoichiometry, and adopting $n = 2$.

The assumed electroneutrality condition demands that the algebraic sum of all electric currents into the astrocyte has to be zero at every instant. The astrocytic membrane potential V_m is then given by solving the resulting equation with respect to V_m ;

$$(9) \quad V_m = \frac{g_{\text{Na}} E_{\text{Na}} + g_{\text{K}} E_{\text{K}} + g_{\text{Cl}} E_{\text{Cl}} + \theta_{\text{NBC}} g_{\text{NBC}} E_{\text{NBC}} - J_{\text{NaKATPase}} F}{g_{\text{Na}} + g_{\text{K}} + g_{\text{Cl}} + \theta_{\text{NBC}} g_{\text{NBC}}}.$$

The rate of change of the astrocytic volume relative to its surface area, $\dot{v}_m = v_m / A$, is, by

model creation



translated into text and equations for publication

Publishing a model

The cell membrane is modeled as a capacitor connected in parallel with variable resistances and batteries representing the different ionic currents and pumps. The electrophysiological behavior of a single cell can hence be described with the following differential equation (23)

$$\frac{dV}{dt} = -\frac{I_{ion} + I_{stim}}{C_m} \quad (1)$$

where V is voltage, t is time, I_{ion} is the sum of all transmembrane ionic currents, I_{stim} is the externally applied stimulus current, and C_m is cell capacitance per unit surface area.

Similarly, ignoring the discrete character of microscopic cardiac cell structure, a 2D sheet of cardiac cells can be modeled as a continuous system with the following partial differential equation (23)

$$\frac{\partial V}{\partial t} = -\frac{I_{ion} + I_{stim}}{C_m} + \frac{1}{\rho_x S_x C_m} \frac{\partial^2 V}{\partial x^2} + \frac{1}{\rho_y S_y C_m} \frac{\partial^2 V}{\partial y^2} \quad (2)$$

where ρ_x and ρ_y are the cellular resistivity in the x and y directions, S_x and S_y are the surface-to-volume ratio in the x and y directions, and I_{ion} is the sum of all transmembrane ionic currents given by the following equation

$$I_{ion} = I_{Na} + I_{K1} + I_{Na} + I_{K2} + I_{K3} + I_{Ca} + I_{NaCa} + I_{NaK} + I_{pCa} + I_{pK} + I_{sCa} + I_{sK} \quad (3)$$

where I_{NaCa} is $\text{Na}^+/\text{Ca}^{2+}$ exchanger current, I_{NaK} is Na^+/K^+ pump current, I_{pCa} and I_{pK} are plateau Ca^{2+} and K^+ currents, and I_{sCa} and I_{sK} are background Ca^{2+} and K^+ currents.

model creation



translated into text and equations for publication



reviewed & published



Publishing a model

```
%-----Calc the L-type Ca
current-----
[CfCa,RevPCa]= CalcConstantfield(Cai,Cao,2,
Vm); %Ca
[CfK,RevPK] = CalcConstantfield(Ki,Ko,1, Vm);
%K
[CfNa,RevPNa] = CalcConstantfield(Nai,Nao,
1, Vm); %Na
if (count ==1 & currenttime == 0)
  Va = -74.0078;
else
  Va = Vm;
end
if (count ==0)
  [mcal, hcal,n] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
  ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
  ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
  %ICaLCa = (PCAL * CfCa*mcal*hcal);

%original
  ICaLCa = (PCAL * CfCa*mcal*hcal);
  ICaL = ICaLCa + ICaLK+ICaLNa;
else
  ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
  ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
  ICaLCa = (PCAL * CfCa*mcal*hcal);
  ICaL = ICaLCa + ICaLK+ICaLNa;
  [mcal, hcal] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
end
```

model creation



translated into text and equations for publication



reviewed & published



interpreted & implemented



Publishing a model

```
%-----Calc the L-type Ca
current-----
[CfCa,RevPCa]= CalcConstantfield(Cai,Cao,2,
Vm); %Ca
[CfK,RevPK] = CalcConstantfield(Ki,Ko,1, Vm);
%K
[CfNa,RevPNa] = CalcConstantfield(Nai,Nao,
1, Vm); %Na
if (count ==1 & currenttime == 0)
    Va = -74.0078;
else
    Va = Vm;
end
if (count ==0)
    [mcal, hcal,n] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
    ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
    ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
    %ICaLCa = (PCAL * CfCa*mcal*hcal);

%original
    ICaLCa = (PCAL * CfCa*mcal*hcal);
    ICaL = ICaLCa + ICaLK+ICaLNa;
else
    ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
    ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
    ICaLCa = (PCAL * CfCa*mcal*hcal);
    ICaL = ICaLCa + ICaLK+ICaLNa;
    [mcal, hcal] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
end
```

model creation



error

translated into text and
equations for publication



error

reviewed & published



error

interpreted & implemented



The reality...

- There are >450 models in the repository
- Only a handful been translated straight from the published paper into a working CellML model
- Typographical errors
- Missing parameter values
- Missing initial conditions
- Missing equations
- Lack of unit definitions



Step 1: model translation

Published paper

P_1, P_2, T_0, C
 k_3, k_4, k_d

$$\frac{dP_1}{dt} = V_{1T} \frac{P_1}{K_{1T} + P_1} - V_{2T} \frac{P_1}{K_{2T} + P_1} - k_3 P_1 T_0 + k_4 C - \nu_{dT} \frac{P_1}{K_{dT} + P_1} - k_d P_1 \quad (1d)$$

M_T, C_N, ν_{dT}
 $\frac{dM_T}{dt} = \nu_{dT} \frac{K_{dT}^n}{K_{dT}^n + C_N} - \nu_{mT} \frac{M_T}{K_{mT} + M_T} - k_d M_T \quad (1e)$

T_0, M_T, T_1
 $\frac{dT_0}{dt} = k_{dT} M_T - V_{1T} \frac{T_0}{K_{1T} + T_0} + V_{2T} \frac{T_1}{K_{2T} + T_1} - k_d T_0 \quad (1f)$

T_0, T_1, T_2
 $\frac{dT_1}{dt} = V_{1T} \frac{T_0}{K_{1T} + T_0} - V_{2T} \frac{T_1}{K_{2T} + T_1} - V_{3T} \frac{T_1}{K_{3T} + T_1} + V_{4T} \frac{T_2}{K_{4T} + T_2} - k_d T_1 \quad (1g)$

P_2, C
 k_3, k_4
 $\frac{dT_2}{dt} = V_{3T} \frac{T_1}{K_{3T} + T_1} - V_{4T} \frac{T_2}{K_{4T} + T_2} - k_3 P_2 T_2 + k_4 C - \nu_{dT} \frac{T_2}{K_{dT} + T_2} - k_d T_2 \quad (1h)$

T_2
 k_4
 $\frac{dC}{dt} = k_3 P_2 T_2 - k_4 C - k_1 C + k_2 C_N - k_{dN} C$

$\frac{dC_N}{dt} = k_1 C - k_2 C_N - k_{dN} C_N$

ie total (nonconserved) quantities of PER &

CellML

```

<component xmlns="http://www.cellml.org/cellml/1.0#" name="membrane">
  <variable name="V" units="millivolt" initial_value="-61" public_interface="out"/>
  <variable name="Cm" units="femtoF" initial_value="6158"/>
  <variable name="time" units="millisecond" public_interface="in"/>
  <variable name="i_K" units="picoA" public_interface="in"/>
  <variable name="i_K_Ca" units="picoA" public_interface="in"/>
  <variable name="i_K_ATP" units="picoA" public_interface="in"/>
  <variable name="i_CRAC" units="picoA" public_interface="in"/>
  <variable name="i_Ca" units="picoA" public_interface="in"/>
  <variable name="i_leak" units="picoA" public_interface="in"/>
  <math xmlns="http://www.w3.org/1998/Math/MathML">
    <apply>
      <eq/>
      <apply>
        <diff/>
        <bvar>
          <ci>time</ci>
        </bvar>
        <ci>V</ci>
      </apply>
      <apply>
        <divide/>
        <apply>
          <minus/>
          <apply>
            <plus/>
            <ci>i_Ca</ci>
            <ci>i_K</ci>
            <ci>i_K_ATP</ci>
            <ci>i_K_Ca</ci>
            <ci>i_CRAC</ci>
            <ci>i_leak</ci>
          </apply>
        </apply>
        <ci>Cm</ci>
      </apply>
    </math>
  </component>
<component xmlns="http://www.cellml.org/cellml/1.0#" name="K_current">

```



Step 2: tool-assisted code validation

PhysioME CellML Environment

COR 0.9 [Editorial Mode] - ..\..\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml

Model oyeahaug_model_neuron loaded.

Equation

$$\alpha_m = \frac{1}{\tau_{m_activation}} \times \frac{1}{e^{-(0.143(9m_{n,t}+5.67)+1)}}$$

Dismissible Errors:

- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: Expected a real number, but didn't get one in a valid format
- Error: MathML cn element contains an invalid constant representation
- Warning: MathML equals element has inconsistent units between the sides
- Warning: MathML equals element has inconsistent units between the sides
- Error: MathML cn element contains an invalid constant representation
- Error: MathML cn element contains an invalid constant representation
- Error: MathML cn element contains an invalid constant representation
- Warning: Expected all arguments to MathML apply to have the same units
- Warning: Expected all arguments to MathML apply to have the same units
- Warning: Expected all arguments to MathML apply to have the same units
- Warning: Expected all arguments to MathML apply to have the same units
- Warning: Expected all arguments to MathML apply to have the same units
- Warning: Expected all arguments to MathML apply to have the same units

Equation:

$$\frac{dK_t}{dt} = \frac{i_{fK_t} + i_{Kst_t} + i_{Kl_t} + i_{Kb_t} + i_{NaK_t} - 2 + i_{Kto_t}}{F \cdot Jte_K}$$

pasek_model_2006_flattened:

```

var j_trpn: millimolar_per_second {pub: in;};
var jte_Ma: millimolar_per_second {pub: in;};
var jte_Ca: millimolar_per_second {pub: in;};
var jte_Na: millimolar_per_second {pub: in;};
var Smt_cm2: cm2 {pub: in;};
var Smt: Smt_cm2 {pub: in;};

ode(Na_i, time) = ({_Na_s+i}_Na_Tt+{_Na_b_s+i}_NaB_t+{_Na_Ca_s+i}_NaCa_s+{_Na_K_s+i}_NaK_s+{_Na_Cl_s+i}_NaCl_s)/({F-jte_Na}/Vt);
ode(Na_i, time) = ({_Na_s+i}_Na_Tt+{_Na_b_s+i}_NaB_t+{_Na_Ca_s+i}_NaCa_s+{_Na_K_s+i}_NaK_s+{_Na_Cl_s+i}_NaCl_s)/({F-jte_Na}/Vt);

i_Kext = set
  case ($STIMULATION == 1 (dimensionless)) and (time-time0 < 0.001 (second)):
    $2 (micro_per_cm2) (Sms+Smt);
  otherwise:
    $0 (microA);
  endset;

ode(K_i, time) = (-{_i_Kext+i}_fK_s+i}_fK_Tt+{_i_Kst+i}_Kst_Tt+{_i_Kl+i}_Kl_Tt+{_i_Kb+i}_Kb_Tt+{_i_NaK+i}_NaK_Tt+{_i_Kto+i}_Kto_Tt)/({dimensionless}+{_i_NaK_t+i}_NaK);
ode(K_i, time) = (-{_i_Kext+i}_fK_s+i}_fK_Tt+{_i_Kst+i}_Kst_Tt+{_i_Kl+i}_Kl_Tt+{_i_Kb+i}_Kb_Tt+{_i_NaK+i}_NaK_Tt+{_i_Kto+i}_Kto_Tt)/({dimensionless}+{_i_NaK_t+i}_NaK);
ode(Ca_i, time) = {-(_ca_s+i)_Ca_Tt+{_ca_b_s+i}_CaB_t+{_ca_Ca_s+i}_CaCa_s+{_ca_K_s+i}_CaK_s+{_ca_Cl_s+i}_CaCl_s}/({dimensionless}*(1/mCdN/sqr(_mcMdn*_ca_s)));
ode(Ca_i, time) = {-(_ca_s+i)_Ca_Tt+{_ca_b_s+i}_CaB_t+{_ca_Ca_s+i}_CaCa_s+{_ca_K_s+i}_CaK_s+{_ca_Cl_s+i}_CaCl_s}/({dimensionless}*(1/mCdN/sqr(_mcMdn*_ca_s)));
ode(CaRel, time) = 1/dimensionless/(1/mCdN/sqr(_mcMdn*_ca_s));
ode(CaSrel, time) = 1/dimensionless/(1/mCdN/sqr(_mcMdn*_ca_s));
ode(CaSRup, time) = _j_caSRup-j_trt;
enddef;

def comp_tubular_ion_fluxes as
  var fK_s: millimolar_per_second {pub: out;};
  var fK_Tt: millimolar_per_second {pub: out;};
  var jte_Ca: millimolar_per_second {pub: out;};
  var jte_Na: millimolar_per_second {pub: out;};
  var i_Kext: dimensionless {init: 0.15};
  var tau_u_K: second {init: 0.15};
  var VT_cm2: cm2 {pub: in;};
  var Na_t: millimolar {pub: in;};
  var Ca_t: millimolar {pub: in;};
  var K_t: millimolar {pub: in;};
  var Na_E: millimolar {pub: in;};
  var r_s: dimensionless {init: 1};

```

Warnings:

- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (284): some units used in this equation are not exactly equivalent, but dimensionally equivalent and might therefore have to be converted
- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (425): there is one or several problems with the units used in this equation
- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (437): there is one or several problems with the units used in this equation
- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (439): there is one or several problems with the units used in this equation
- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (461): some units used in this equation are not exactly equivalent, but dimensionally equivalent and might therefore have to be converted
- [Warning] ...:\Documents and Settings\cllo007\Desktop\Pasek_2006\pasek_model_2006_flattened.cellml (783): some units used in this equation are not exactly equivalent, but dimensionally equivalent and might therefore have to be converted

Step 3: simulation comparisons

9110 Cell Biology: Goldbeter

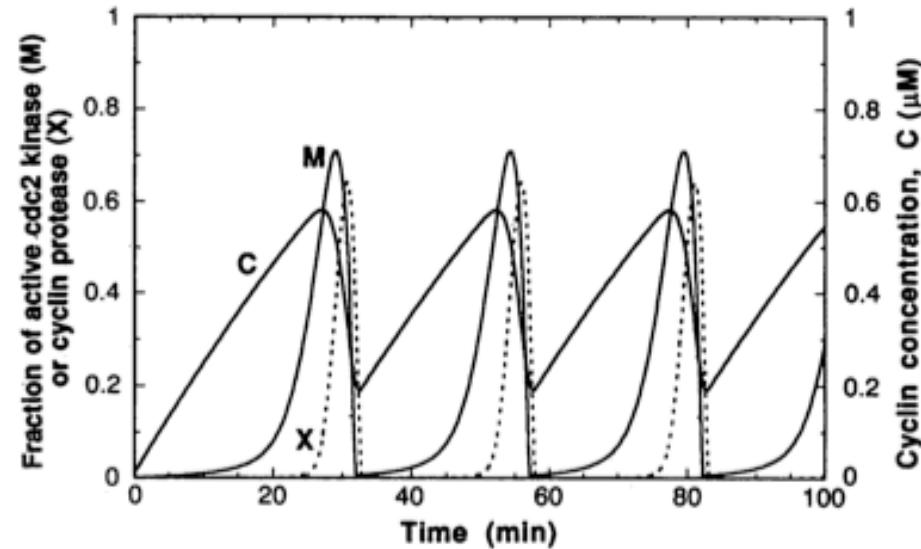
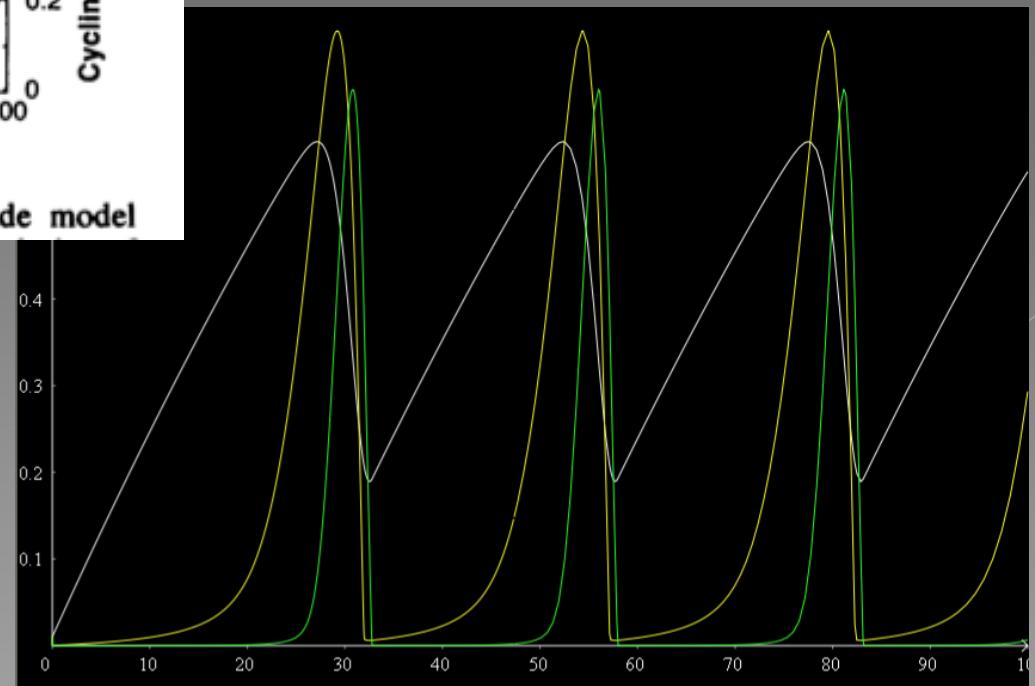


FIG. 3. Sustained oscillations in the minimal cascade model





Step 4: obtain the original code

```
%-----Calc the L-type Ca current-----
[CfCa,RevPCa]= CalcConstantfield(Cai,Cao,2,
Vm); %Ca
[CfK,RevPK] = CalcConstantfield(Ki,Ko,1, Vm);
%K
[CfNa,RevPNa] = CalcConstantfield(Nai,Nao,1,
Vm); %Na
if (count ==1 && currenttime == 0)
    Va = -74.0078;
else
    Va = Vm;
end
if (count ==0)
    [mcal, hcal,n] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
%ICaLCa = (PCAL * CfCa*mcal*hcal); %original
ICaLCa = (PCAL * CfCa*mcal*hcal);
ICaL = ICaLCa + ICaLK+ICaLNa;
else
    ICaLNa = (0.00005*PCAL*CfNa)*mcal*hcal;
    ICaLK = (0.001 * PCAL * CfK)*mcal*hcal;
    ICaLCa = (PCAL * CfCa*mcal*hcal);
    ICaL = ICaLCa + ICaLK+ICaLNa;
    [mcal, hcal] = calcRateConst(1,Va,
0,Cai,mcal,hcal,count,dt); %Calc m and h
end
```

```

582         </apply>
583     </math>
584   </component>
585   <component name="L_type_Ca_channel">
586     <variable units="picoA" public_interface="out" name="i_Ca_L"/>
587     <variable units="nanoS" name="g_Ca_L" initial_value="6.75"/>
588     <variable units="millivolt" name="E_Ca_app" initial_value="60"/>
589     <variable units="dimensionless" name="f_Ca"/>
590     <variable units="millimolar" name="k_Ca" initial_value="0.025"/>
591     <variable units="second" public_interface="in" private_interface="out" name="time"/>
592     <variable units="millivolt" public_interface="in" private_interface="out" name="V"/>
593     <variable units="millimolar" public_interface="in" name="Ca_d"/>
594     <variable units="dimensionless" private_interface="in" name="d_L"/>
595     <variable units="dimensionless" private_interface="in" name="f_L_1"/>
596     <variable units="dimensionless" private_interface="in" name="f_L_2"/>
597     <math xmlns="http://www.w3.org/1998/Math/MathML">
598       <apply>
599         <eq/>
600         <ci>i_Ca_L</ci>
601         <apply>
602           <times/>
603           <ci>g_Ca_L</ci>
604           <ci>d_L</ci>
605           <apply>
606             <plus/>
607             <apply>
608               <times/>
609               <ci>f_Ca</ci>
610               <ci>f_L_1</ci>
611             </apply>
612             <apply>
613               <minus/>
614               <cn cellml:units="dimensionless">1</cn>
615               <ci>f_Ca</ci>
616             </apply>
617             <ci>f_L_2</ci>
618           </apply>
619         </apply>
620       </math>
621     </component>
```



The current situation

- After a long period without tools we now have a legacy of broken models to curate
- Of the 450 models in the repository ~ $\frac{1}{2}$ have been curated to some degree
- We accept there are some models which will never work properly

Future work

- Continuing to validate the models in the repository (existing and new)
- Model annotation
- Automated curation?
- Replace the curation star system with a more meaningful set of curation “flags”





Acknowledgements

