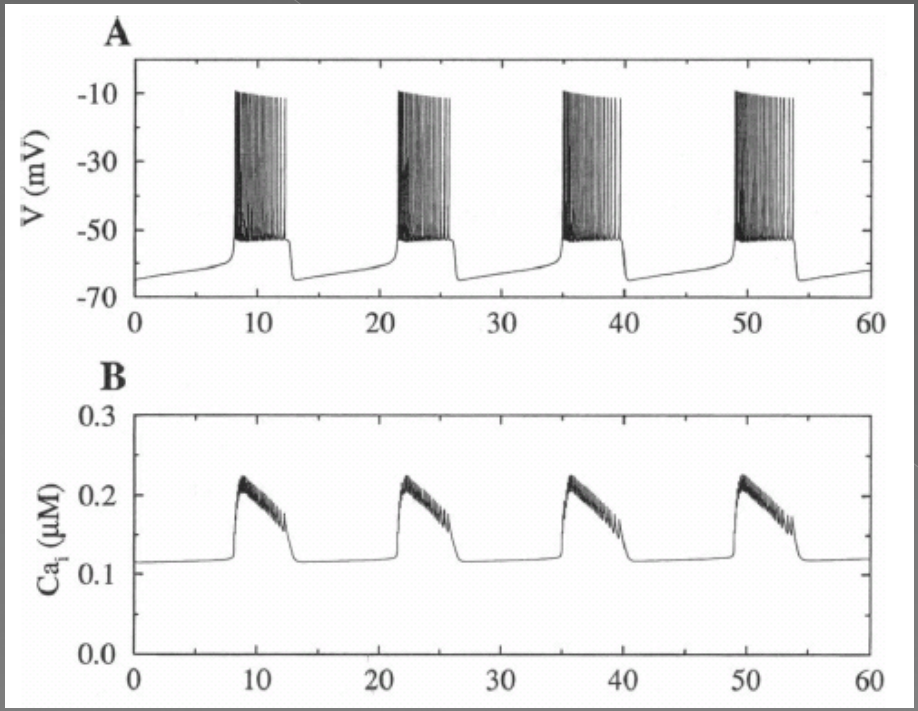
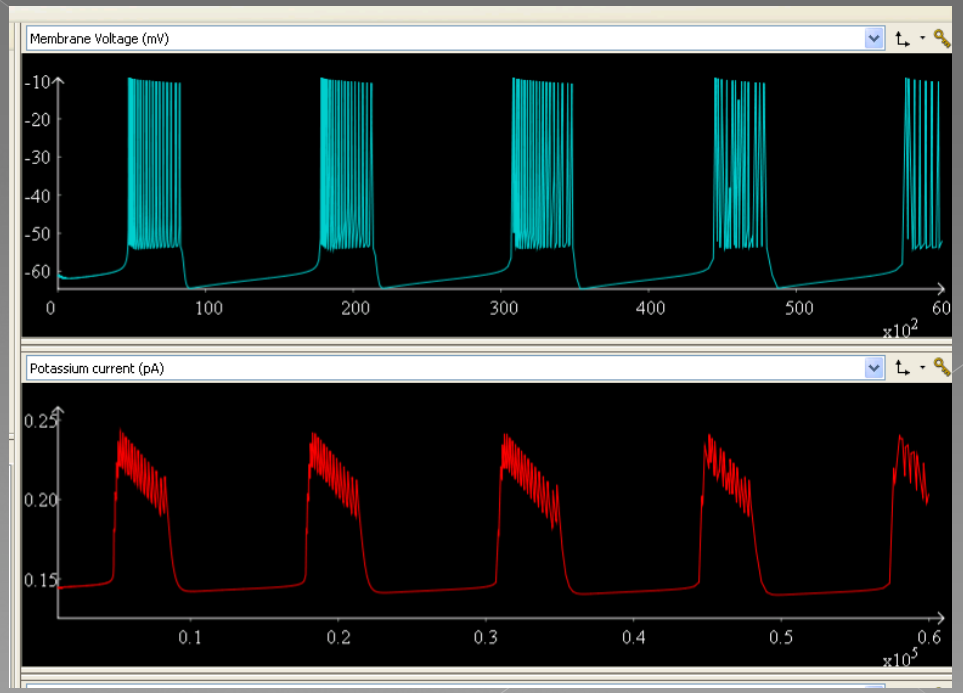
Model curation: the CellML approach

Catherine Lloyd
Auckland Bioengineering Institute

What is curation?



model validation



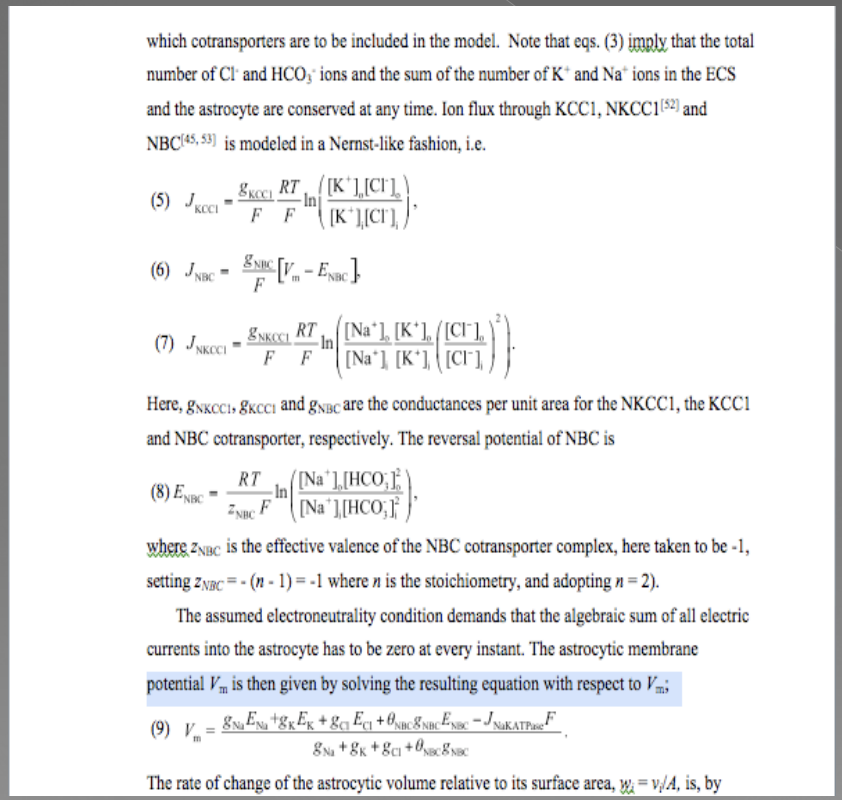
Publishing a model

model creation

```
                                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
```

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.

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

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

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

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

$$(8)\quad E_{NBC} = \frac{RT}{z_{NBC}F}\ln\left(\frac{[Na^+]_o[HCO_3^-]_o^2}{[Na^+]_i[HCO_3^-]_i^2}\right),$$

where z_{NBC} is the effective valence of the NBC cotransporter complex, here taken to be -1, setting $z_{NBC} = -(n-1) = -1$ 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;

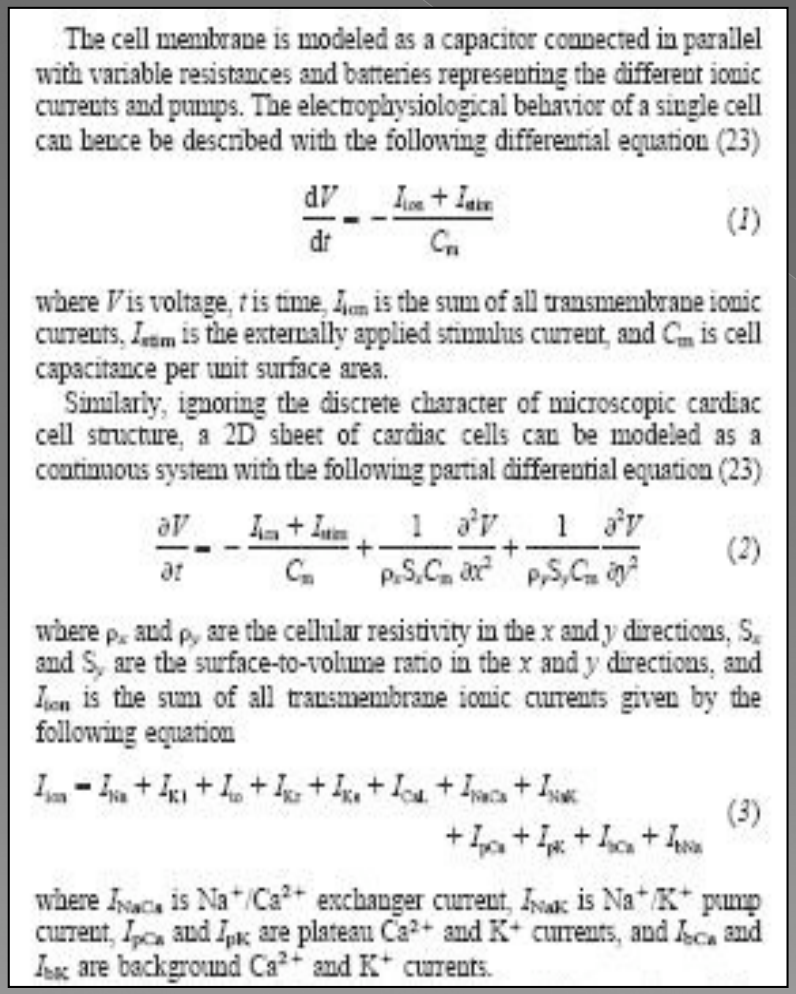
$$(9)\quad V_m = \frac{g_{Na}E_{Na} + g_K E_K + g_{Cl}E_{Cl} + \theta_{NBC}g_{NBC}E_{NBC} - J_{NaKATPase}F}{g_{Na} + g_K + g_{Cl} + \theta_{NBC}g_{NBC}}.$$

The rate of change of the astrocytic volume relative to its surface area, $w_i = v_i/A$, 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

$$I_{ion} = I_{Na} + I_{K1} + I_{to} + I_{Kr} + I_{Ks} + I_{CaL} + I_{NaCa} + I_{NaK} + I_{pCa} + I_{pK} + I_{bCa} + I_{bNa} \quad (3)$$

where I_{NaCa} is Na^+/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_{bCa} and I_{bK} 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

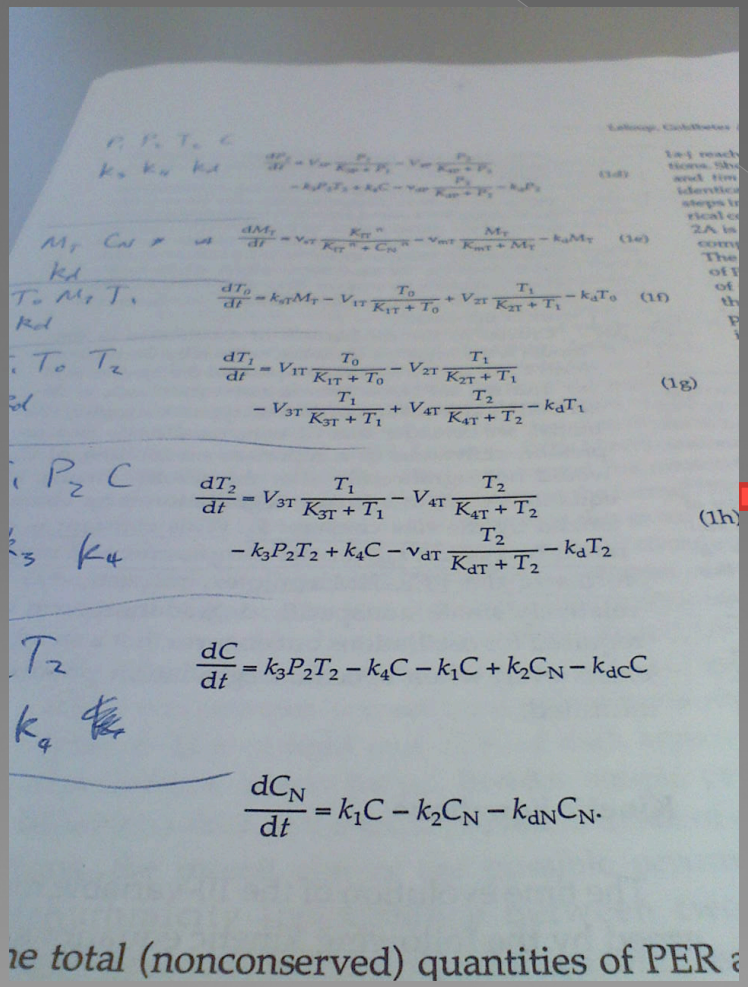
interpreted & implemented

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



$$\frac{dT_0}{dt} = k_{dT}M_T - V_{1T}\frac{T_0}{K_{1T}+T_0} + V_{2T}\frac{T_1}{K_{2T}+T_1} - k_dT_0 \quad (1f)$$

$$\frac{dT_1}{dt} = V_{1T}\frac{T_0}{K_{1T}+T_0} - V_{2T}\frac{T_1}{K_{2T}+T_1} - V_{3T}\frac{T_1}{K_{3T}+T_1} + V_{4T}\frac{T_2}{K_{4T}+T_2} - k_dT_1 \quad (1g)$$

$$\frac{dT_2}{dt} = V_{3T}\frac{T_1}{K_{3T}+T_1} - V_{4T}\frac{T_2}{K_{4T}+T_2} - k_3P_2T_2 + k_4C - v_{dT}\frac{T_2}{K_{dT}+T_2} - k_dT_2 \quad (1h)$$

$$\frac{dC}{dt} = k_3P_2T_2 - k_4C - k_1C + k_2C_N - k_{dC}C$$

$$\frac{dC_N}{dt} = k_1C - k_2C_N - k_{dN}C_N.$$

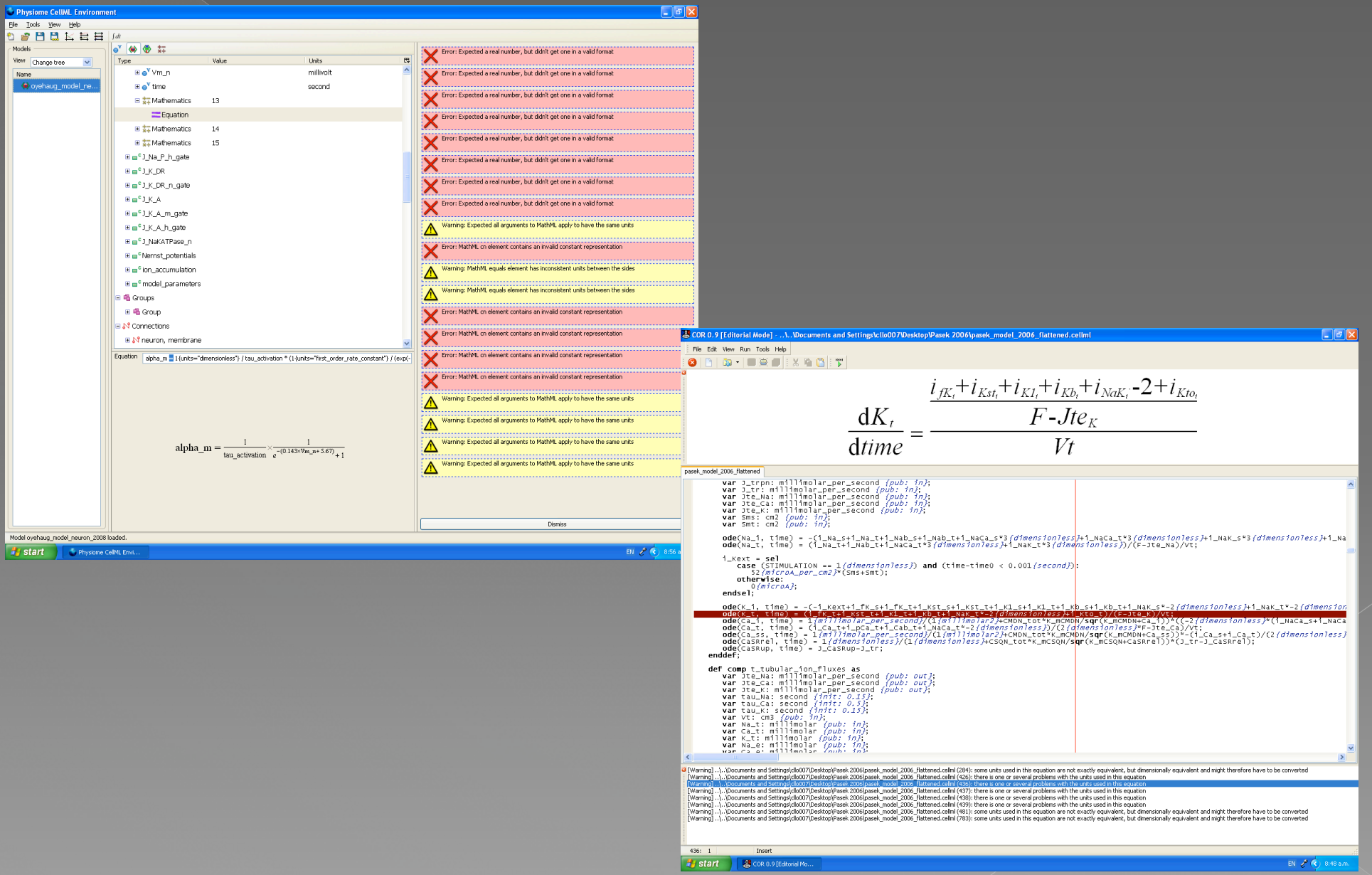
...e total (nonconserved) quantities of PER a

CellML

Bertram_and_Mears_1995.cellml

```
166   <component xmlns="http://www.cellml.org/cellml/1.0#" name="membrane">
167     <variable name="V" units="millivolt" initial_value="-61" public_interface="out"/>
168     <variable name="Cm" units="femtoF" initial_value="6158"/>
169     <variable name="time" units="millisecond" public_interface="in"/>
170     <variable name="i_K" units="picoA" public_interface="in"/>
171     <variable name="i_K_Ca" units="picoA" public_interface="in"/>
172     <variable name="i_K_ATP" units="picoA" public_interface="in"/>
173     <variable name="i_CRAC" units="picoA" public_interface="in"/>
174     <variable name="i_Ca" units="picoA" public_interface="in"/>
175     <variable name="i_leak" units="picoA" public_interface="in"/>
176     <math xmlns="http://www.w3.org/1998/Math/MathML">
177       <apply>
178         <eq/>
179         <apply>
180           <diff/>
181           <bvar>
182             <ci>time</ci>
183           </bvar>
184           <ci>V</ci>
185         </apply>
186         <apply>
187           <divide/>
188           <apply>
189             <minus/>
190             <apply>
191               <plus/>
192               <ci>i_Ca</ci>
193               <ci>i_K</ci>
194               <ci>i_K_ATP</ci>
195               <ci>i_K_Ca</ci>
196               <ci>i_CRAC</ci>
197               <ci>i_leak</ci>
198             </apply>
199           </apply>
200           <ci>Cm</ci>
201         </apply>
202       </apply>
203     </math>
204   </component>
205   <component xmlns="http://www.cellml.org/cellml/1.0#" name="K current">
```

Step 2: tool-assisted code validation



Step 3: simulation comparisons

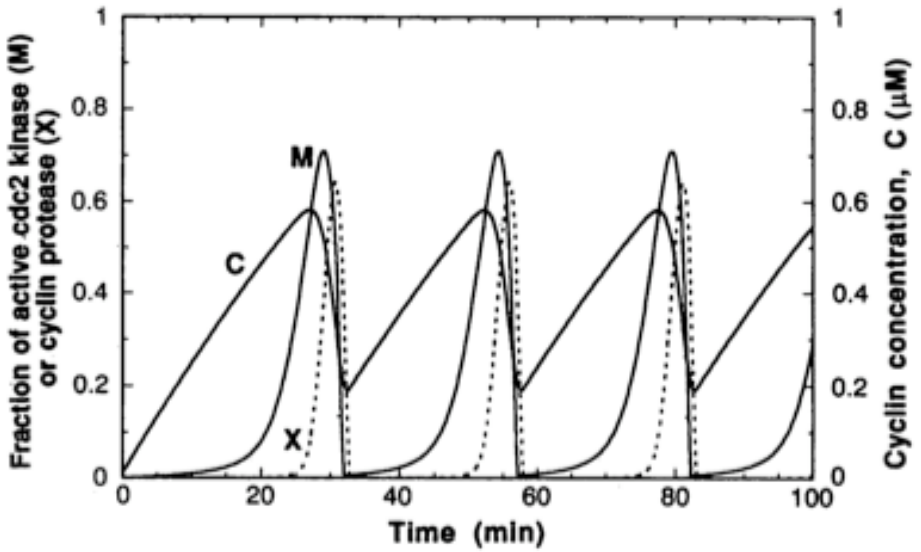
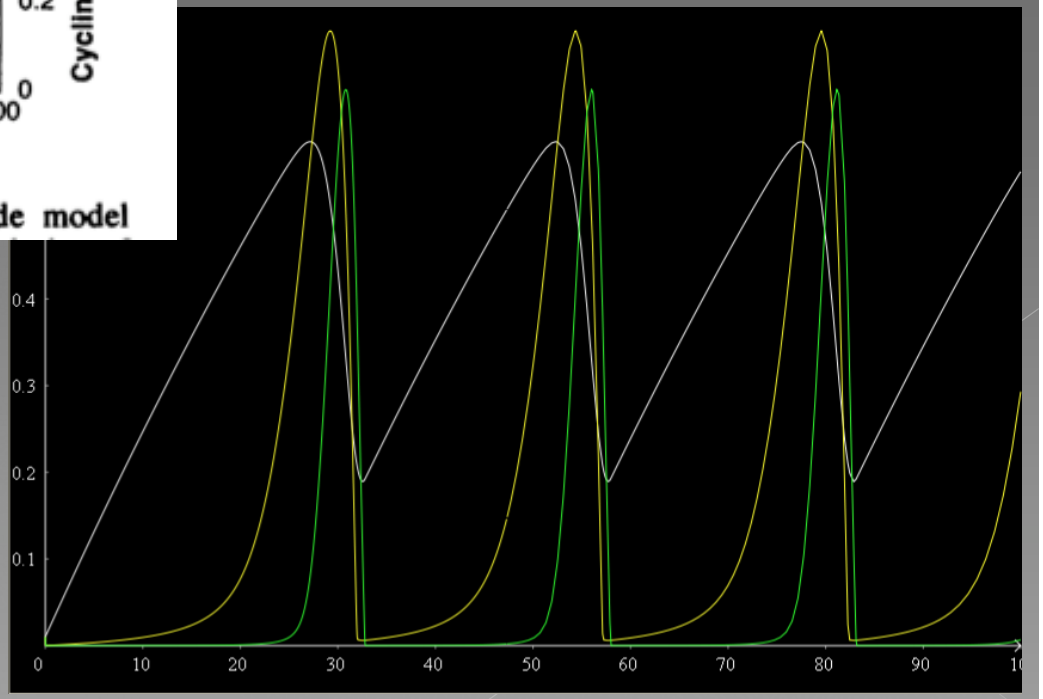


FIG. 3. Sustained oscillations in the minimal cascade model



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

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 ~½ 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