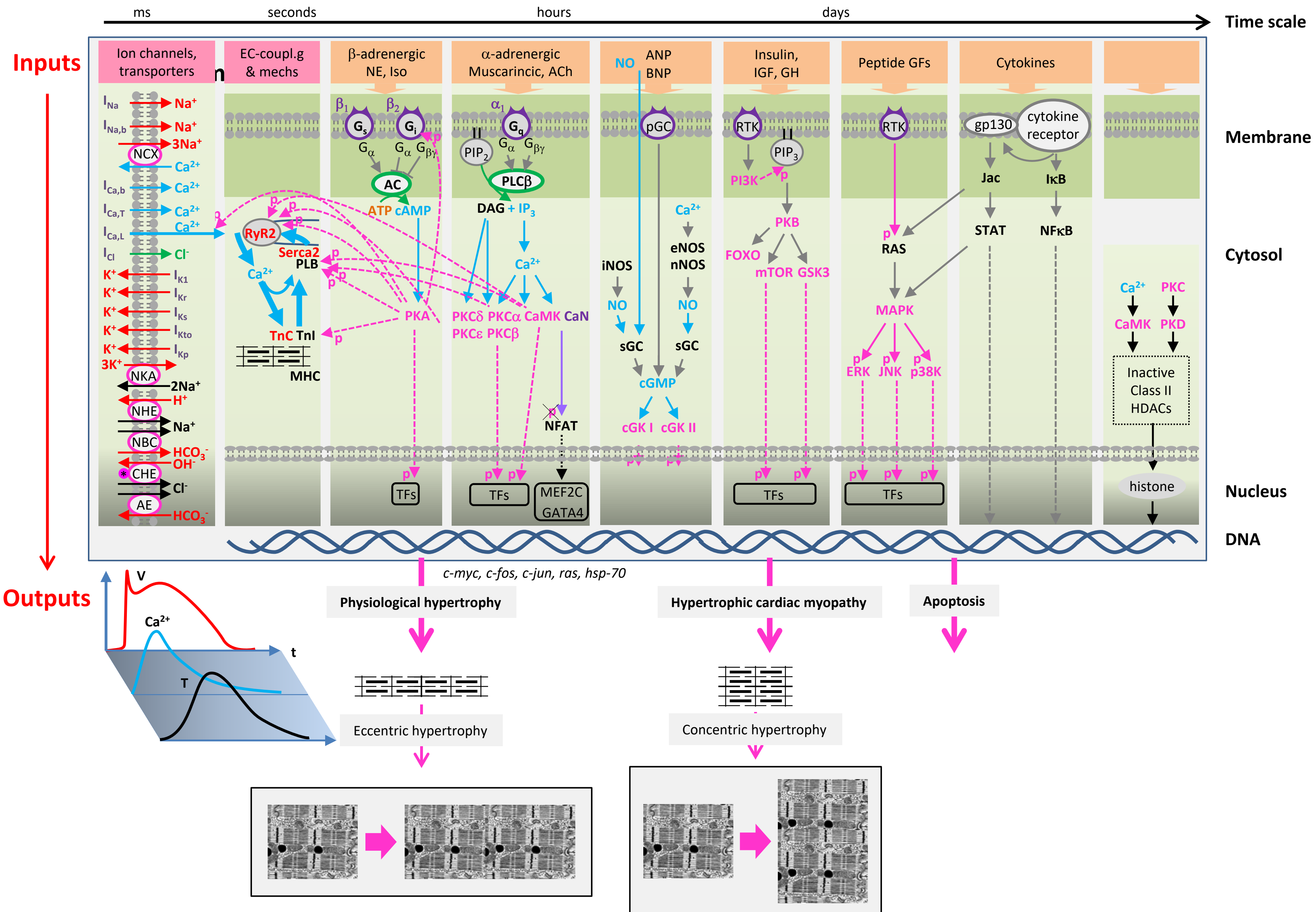
https://doi.org/10.17608/k6.auckland.10080263

CellML, PMR, OpenCOR, CRBM, …

David Nickerson
Auckland Bioengineering Institute
University of Auckland
New Zealand

CellML signalling modules for the cardiac myocyte



cellML

https://cellml.org/

CellML

- XML format for encoding mathematical models
- Reproducibility
  - – Unambiguous description of the mathematical model
- Reusability
  - – Modular, composable
- Comprehensible
  - – Metadata to describe the biological semantics

- Tool support
  - – CellML API library and service
  - – Most tools don’t support model composition

CellML 2.0

- Reactions are gone!
- Only CellML allowed in the XML document
  - No metadata, annotations, cmeta:id
  - No extension elements
- XML syntax simplifications
  - Grouping replaced with only encapsulation
  - No more map_components
- Improved reusability
  - Connections no longer have direction
  - Single interface attribute controlling scope: public, private, public_and_private, none

https://doi.org/10.17608/k6.auckland.10080263

cellML 2.0

- Units clarifications
  - No need to specify base_units explicitly
  - Units with offsets removed
  - “celsius” removed from built-in units
  - Component-scope unit definitions removed
- Reset rules
  - Arbitrary rules to “reset” variables
- New and compulsory MathML subset
  - No more “recommended” subset to support
  - Well defined, no confusion

https://doi.org/10.17608/k6.auckland.10080263

libCellML

- New C++ library to meet the needs of users
- Supporting CellML 2.0 and beyond
- Much more streamlined and maintainable
- Better suited for testing out new features and extensions to the specification
  - Allowing rapid prototyping
  - Exploring alternatives
  - Testing model exchange and reproducibility

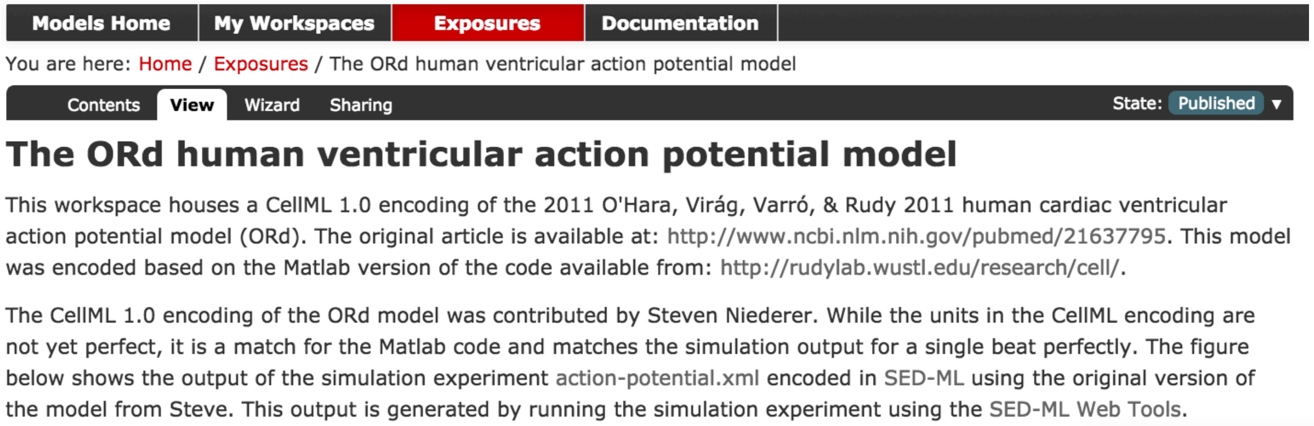
https://doi.org/10.17608/k6.auckland.10080263

The Physiome Model Repository – PMR

https://models.physiomeproject.org/

- Over 800 publicly available workspaces
  - Version control repositories (git)
  - Historically mostly CellML models from the literature
  - Gradually getting more non-CellML data contributed (SED-ML, FE models, code)
- Many more exposures
  - “releases” of workspaces
  - A specific version processed for display and interaction

https://doi.org/10.17608/k6.auckland.10080263



Models Home | My Workspaces | Exposures | Documentation

You are here: Home / Exposures / The ORd human ventricular action potential model

Contents | View | Wizard | Sharing — State: Published ▾

The ORd human ventricular action potential model

This workspace houses a CellML 1.0 encoding of the 2011 O'Hara, Virág, Varró, & Rudy 2011 human cardiac ventricular action potential model (ORd). The original article is available at: http://www.ncbi.nlm.nih.gov/pubmed/21637795. This model was encoded based on the Matlab version of the code available from: http://rudylab.wustl.edu/research/cell/.

The CellML 1.0 encoding of the ORd model was contributed by Steven Niederer. While the units in the CellML encoding are not yet perfect, it is a match for the Matlab code and matches the simulation output for a single beat perfectly. The figure below shows the output of the simulation experiment action-potential.xml encoded in SED-ML using the original version of the model from Steve. This output is generated by running the simulation experiment using the SED-ML Web Tools.

$$\frac{d}{d\,time}(\mathrm{CaMKt}) = \mathrm{aCaMK}\ \mathrm{CaMKb}\ (\mathrm{CaMKb} + \mathrm{CaMKt}) - (\mathrm{bCaMK}\ \mathrm{CaMKt})$$

Component: intracellular_ions

$$\mathrm{cmdnmax} = \begin{cases} \mathrm{cmdnmax\_b}1.3 & \text{if celltype} = 1 \\ \mathrm{cmdnmax\_b} & \text{otherwise} \end{cases}$$

$$\frac{d}{d\,time}(\mathrm{nai}) = \frac{(-(\mathrm{INa}+\mathrm{INaL}+3\mathrm{INaCa\_i}+3\mathrm{INaK}+\mathrm{INab}))\ \mathrm{Acap}\ \mathrm{cm}}{F\ \mathrm{vmyo}} +$$

$$\frac{d}{d\,time}(\mathrm{nass}) = \frac{(-(\mathrm{ICaNa}+3\mathrm{INaCa\_ss}))\ \mathrm{cm}\ \mathrm{Acap}}{F\ \mathrm{vss}} - \mathrm{JdiffNa}$$

$$\frac{d}{d\,time}(\mathrm{ki}) = \frac{(-(\mathrm{Ito}+\mathrm{IKr}+\mathrm{IKs}+\mathrm{IK1}+\mathrm{IKb}+\mathrm{Istim}-(2\ \mathrm{INaK})))\ \mathrm{cm}\ \mathrm{Acap}}{F\ \mathrm{vmyo}}$$

$$\frac{d}{d\,time}(\mathrm{cai}) = \mathrm{Bcai}\ \left(\left(\frac{(-(\mathrm{IpCa}+\mathrm{ICab}-(2\mathrm{INaCa\_i})))\ \mathrm{cm}\ \mathrm{Acap}}{2F\ \mathrm{vmyo}}\right.\right.$$

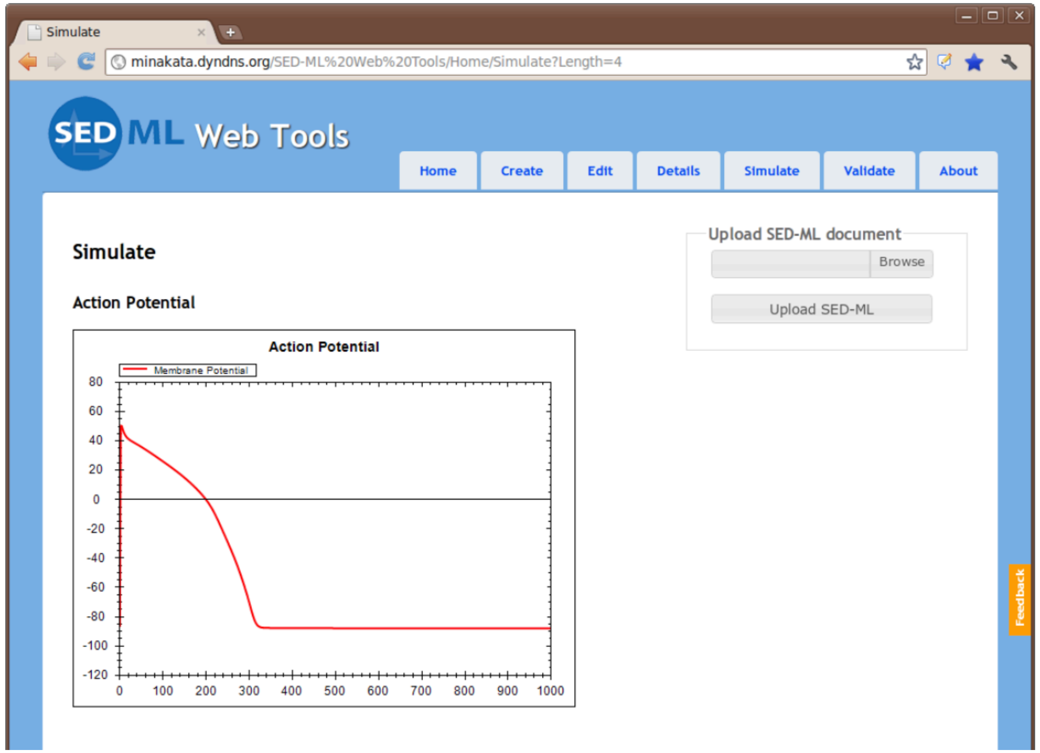
$$\mathrm{Bcass} = \frac{1}{1+\frac{\mathrm{BSRmax}\ \mathrm{KmBSR}}{(\mathrm{KmBSR}+\mathrm{cass})^2}+\frac{\mathrm{BSLmax}\ \mathrm{KmBSL}}{(\mathrm{KmBSL}+\mathrm{cass})^2}}$$

$$\frac{d}{d\,time}(\mathrm{cass}) = \mathrm{Bcass}\ \left(\frac{(-(\mathrm{ICaL}-(2\mathrm{INaCa\_ss})))\ \mathrm{cm}\ \mathrm{Acap}}{2F\ \mathrm{vss}} +\right.$$

$$\frac{d}{d\,time}(\mathrm{cansr}) = \mathrm{Jup} - \left(\frac{\mathrm{Jtr}\ \mathrm{vjsr}}{\mathrm{vnsr}}\right)$$

$$\mathrm{Bcajsr} = \frac{1}{1+\frac{\mathrm{csqnmax}\ \mathrm{kmcsqn}}{(\mathrm{kmcsqn}+\mathrm{cajsr})^2}}$$

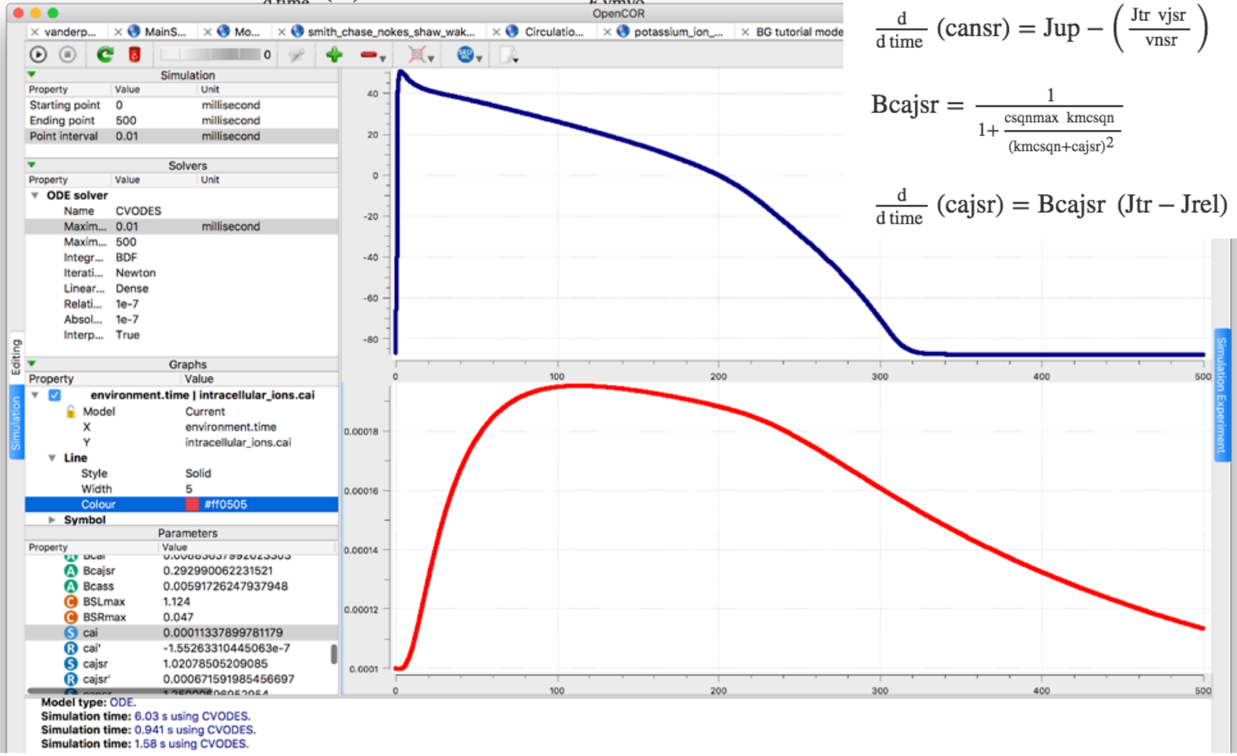
$$\frac{d}{d\,time}(\mathrm{cajsr}) = \mathrm{Bcajsr}\ (\mathrm{Jtr} - \mathrm{Jrel})$$



Simulate

Action Potential

Upload SED-ML document



Simulation

Property	Value	Unit
Starting point	0	millisecond
Ending point	500	millisecond
Point interval	0.01	millisecond

Solvers

Property	Value	Unit
ODE solver		
Name	CVODES	
Maxim...	0.01	millisecond
Maxim...	500	
Integr...	BDF	
Iterati...	Newton	
Linear...	Dense	
Relati...	1e-7	
Absol...	1e-7	
Interp...	True	

Graphs

Property	Value
☑ environment.time | intracellular_ions.cai	
Model	Current
X	environment.time
Y	intracellular_ions.cai
Line	
Style	Solid
Width	5
Colour	🟥 #ff0505
Symbol	

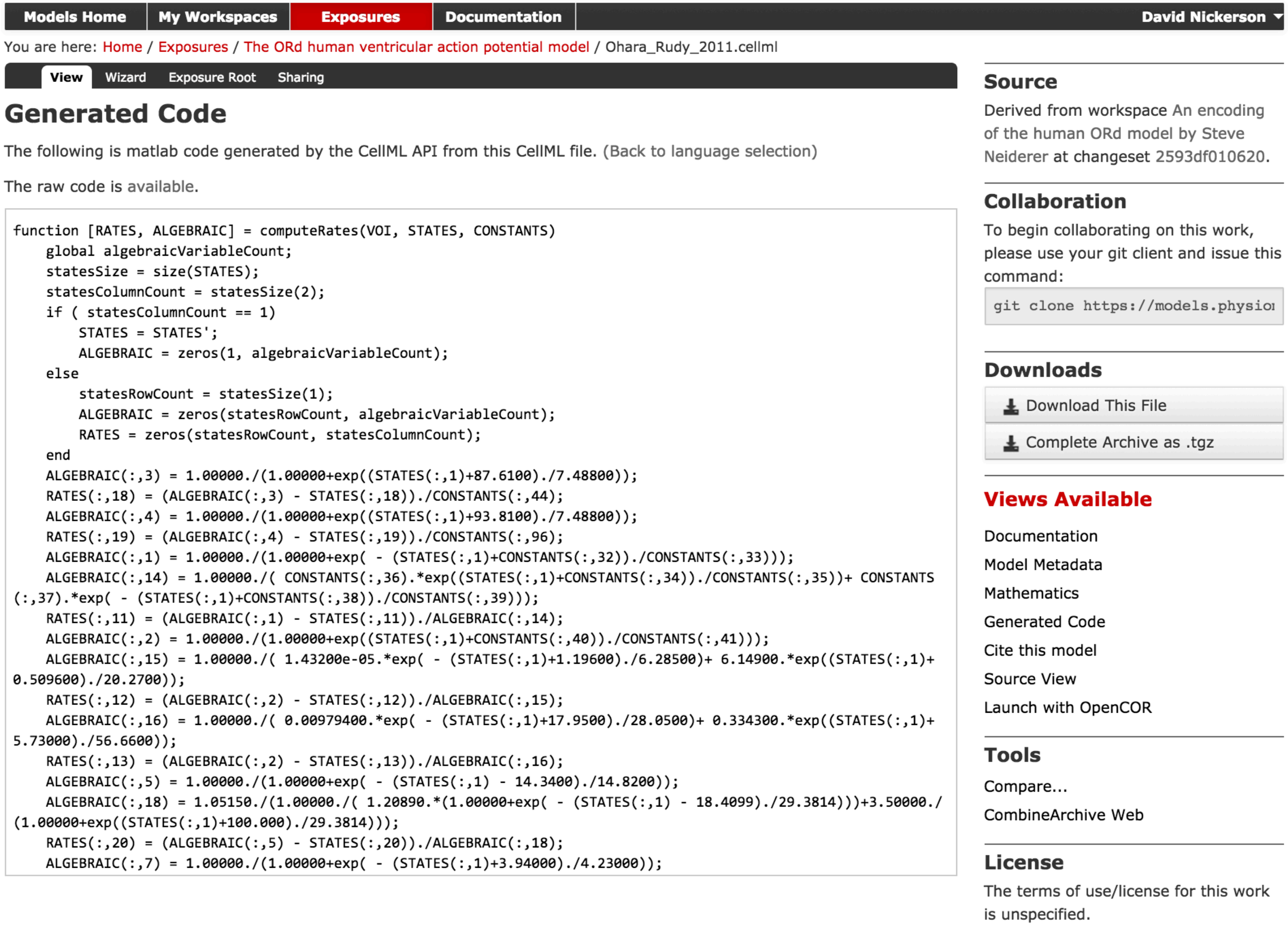
Parameters

Property	Value
Bcajsr	0.292990062231521
Bcass	0.00591726247937948
BSLmax	1.124
BSRmax	0.047
cai	0.00011337899781179
cai'	-1.55263310445063e-7
cajsr	1.02078505209085
cajsr'	0.000671591985456697

Model type: ODE.
Simulation time: 6.03 s using CVODES.
Simulation time: 0.941 s using CVODES.
Simulation time: 1.58 s using CVODES.

https://models.physiomeproject.org/e/71

https://doi.org/10.17608/k6.auckland.10080263



You are here: Home / Exposures / The ORd human ventricular action potential model / Ohara_Rudy_2011.cellml

Generated Code

The following is matlab code generated by the CellML API from this CellML file. (Back to language selection)

The raw code is available.

```
function [RATES, ALGEBRAIC] = computeRates(VOI, STATES, CONSTANTS)
    global algebraicVariableCount;
    statesSize = size(STATES);
    statesColumnCount = statesSize(2);
    if ( statesColumnCount == 1)
        STATES = STATES';
        ALGEBRAIC = zeros(1, algebraicVariableCount);
    else
        statesRowCount = statesSize(1);
        ALGEBRAIC = zeros(statesRowCount, algebraicVariableCount);
        RATES = zeros(statesRowCount, statesColumnCount);
    end
    ALGEBRAIC(:,3) = 1.00000./(1.00000+exp((STATES(:,1)+87.6100)./7.48800));
    RATES(:,18) = (ALGEBRAIC(:,3) - STATES(:,18))./CONSTANTS(:,44);
    ALGEBRAIC(:,4) = 1.00000./(1.00000+exp((STATES(:,1)+93.8100)./7.48800));
    RATES(:,19) = (ALGEBRAIC(:,4) - STATES(:,19))./CONSTANTS(:,96);
    ALGEBRAIC(:,1) = 1.00000./(1.00000+exp( - (STATES(:,1)+CONSTANTS(:,32))./CONSTANTS(:,33)));
    ALGEBRAIC(:,14) = 1.00000./( CONSTANTS(:,36).*exp((STATES(:,1)+CONSTANTS(:,34))./CONSTANTS(:,35))+ CONSTANTS
(:,37).*exp( - (STATES(:,1)+CONSTANTS(:,38))./CONSTANTS(:,39)));
    RATES(:,11) = (ALGEBRAIC(:,1) - STATES(:,11))./ALGEBRAIC(:,14);
    ALGEBRAIC(:,2) = 1.00000./(1.00000+exp((STATES(:,1)+CONSTANTS(:,40))./CONSTANTS(:,41)));
    ALGEBRAIC(:,15) = 1.00000./( 1.43200e-05.*exp( - (STATES(:,1)+1.19600)./6.28500)+ 6.14900.*exp((STATES(:,1)+
0.509600)./20.2700));
    RATES(:,12) = (ALGEBRAIC(:,2) - STATES(:,12))./ALGEBRAIC(:,15);
    ALGEBRAIC(:,16) = 1.00000./( 0.00979400.*exp( - (STATES(:,1)+17.9500)./28.0500)+ 0.334300.*exp((STATES(:,1)+
5.73000)./56.6600));
    RATES(:,13) = (ALGEBRAIC(:,2) - STATES(:,13))./ALGEBRAIC(:,16);
    ALGEBRAIC(:,5) = 1.00000./(1.00000+exp( - (STATES(:,1) - 14.3400)./14.8200));
    ALGEBRAIC(:,18) = 1.05150./(1.00000./( 1.20890.*(1.00000+exp( - (STATES(:,1) - 18.4099)./29.3814)))+3.50000./
(1.00000+exp((STATES(:,1)+100.000)./29.3814)));
    RATES(:,20) = (ALGEBRAIC(:,5) - STATES(:,20))./ALGEBRAIC(:,18);
    ALGEBRAIC(:,7) = 1.00000./(1.00000+exp( - (STATES(:,1)+3.94000)./4.23000));
```

Source

Derived from workspace An encoding of the human ORd model by Steve Neiderer at changeset 2593df010620.

Collaboration

To begin collaborating on this work, please use your git client and issue this command:

```
git clone https://models.physiol
```

Downloads

Views Available

Documentation

Model Metadata

Mathematics

Generated Code

Cite this model

Source View

Launch with OpenCOR

Tools

Compare...

CombineArchive Web

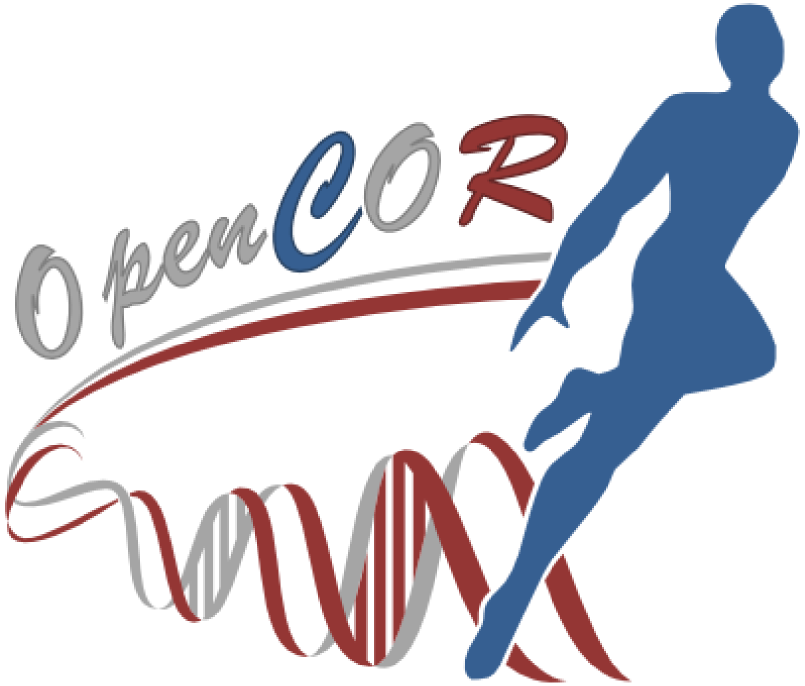
License

The terms of use/license for this work is unspecified.

https://doi.org/10.17608/k6.auckland.10080263

The Physiome Model Repository – PMR

- Consistent browser and tool integration
  - Content type negotiation
  - Same URL
  - REST
- RDF triplestore
  - Indexing versioned annotations
  - Supporting (semantic) querying
- Tools for model composition, parameter estimation, etc.



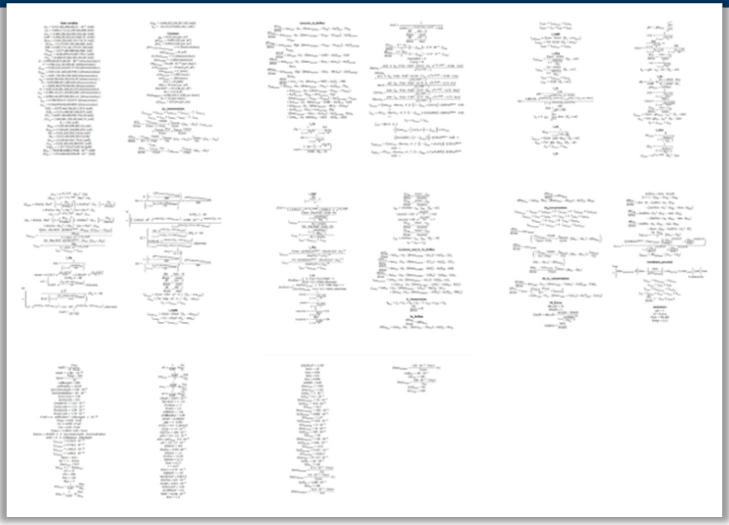
A modelling environment for reproducible science

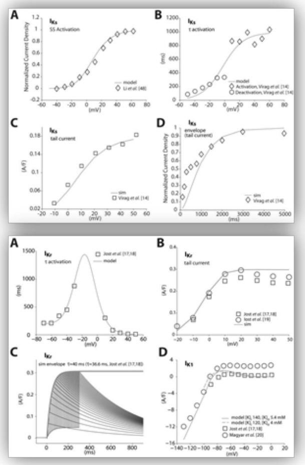
https://opencor.ws

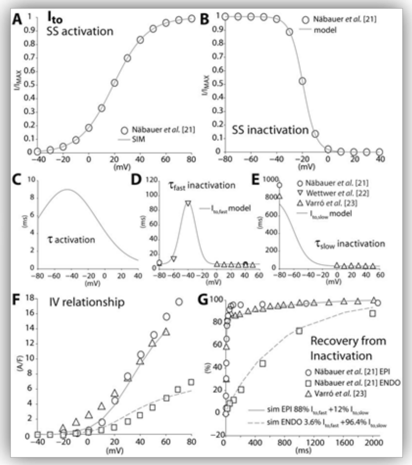
https://doi.org/10.17608/k6.auckland.10080263

THE UNIVERSITY OF AUCKLAND
Te Whare Wānanga o Tāmaki Makaurau
NEW ZEALAND

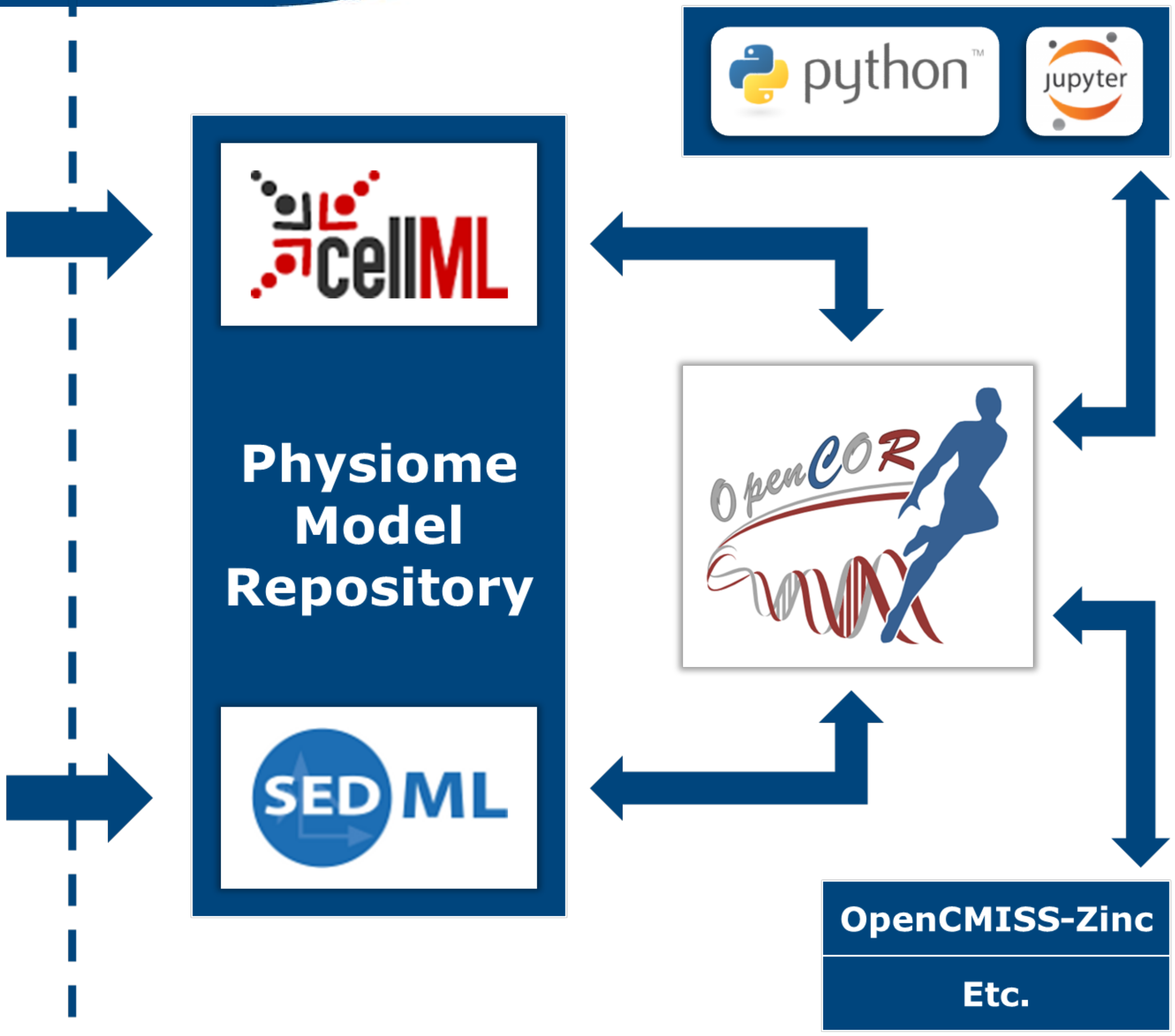
AUCKLAND BIOENGINEERING INSTITUTE







Publication



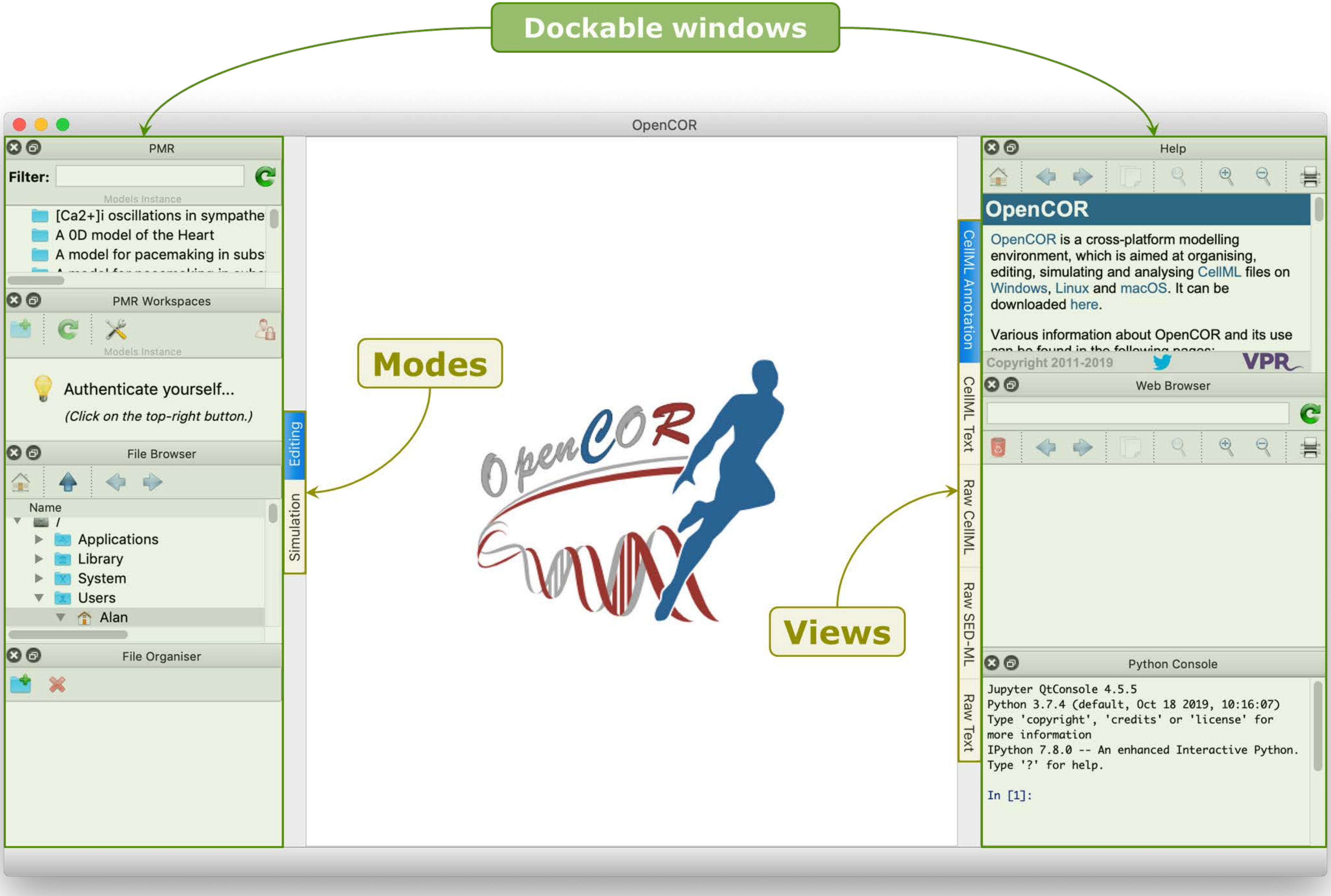
Physiome Model Repository

OpenCMISS-Zinc

Etc.

Software

https://doi.org/10.17608/k6.auckland.10080263



Authenticate yourself...

(Click on the top-right button.)

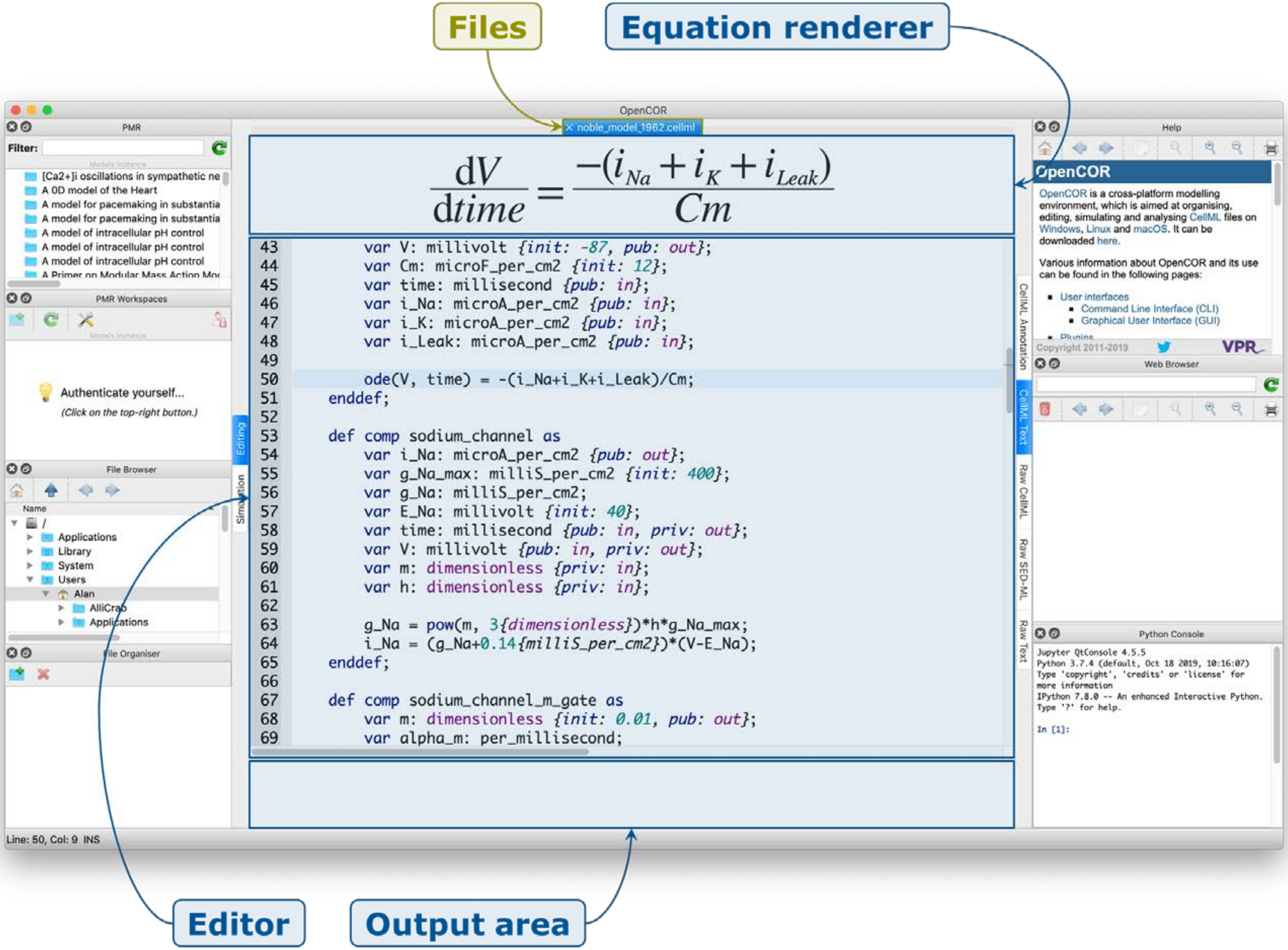
OpenCOR

OpenCOR is a cross-platform modelling environment, which is aimed at organising, editing, simulating and analysing CellML files on Windows, Linux and macOS. It can be downloaded here.

Various information about OpenCOR and its use can be found in the following pages:

```
Jupyter QtConsole 4.5.5
Python 3.7.4 (default, Oct 18 2019, 10:16:07)
Type 'copyright', 'credits' or 'license' for
more information
IPython 7.8.0 -- An enhanced Interactive Python.
Type '?' for help.

In [1]:
```



- [Ca2+]i oscillations in sympathetic ne
- A 0D model of the Heart
- A model for pacemaking in substantia
- A model for pacemaking in substantia
- A model of intracellular pH control
- A model of intracellular pH control
- A model of intracellular pH control
- A Primer on Modular Mass Action Mo

Authenticate yourself...
(Click on the top-right button.)

```
/
  Applications
  Library
  System
  Users
    Alan
      AlliCrab
      Applications
```

$$\frac{dV}{dtime} = \frac{-(i_{Na}+i_K+i_{Leak})}{Cm}$$

```
        var V: millivolt {init: -87, pub: out};
        var Cm: microF_per_cm2 {init: 12};
        var time: millisecond {pub: in};
        var i_Na: microA_per_cm2 {pub: in};
        var i_K: microA_per_cm2 {pub: in};
        var i_Leak: microA_per_cm2 {pub: in};

        ode(V, time) = -(i_Na+i_K+i_Leak)/Cm;
    enddef;

    def comp sodium_channel as
        var i_Na: microA_per_cm2 {pub: out};
        var g_Na_max: milliS_per_cm2 {init: 400};
        var g_Na: milliS_per_cm2;
        var E_Na: millivolt {init: 40};
        var time: millisecond {pub: in, priv: out};
        var V: millivolt {pub: in, priv: out};
        var m: dimensionless {priv: in};
        var h: dimensionless {priv: in};

        g_Na = pow(m, 3{dimensionless})*h*g_Na_max;
        i_Na = (g_Na+0.14{milliS_per_cm2})*(V-E_Na);
    enddef;

    def comp sodium_channel_m_gate as
        var m: dimensionless {init: 0.01, pub: out};
        var alpha_m: per_millisecond;
```

OpenCOR

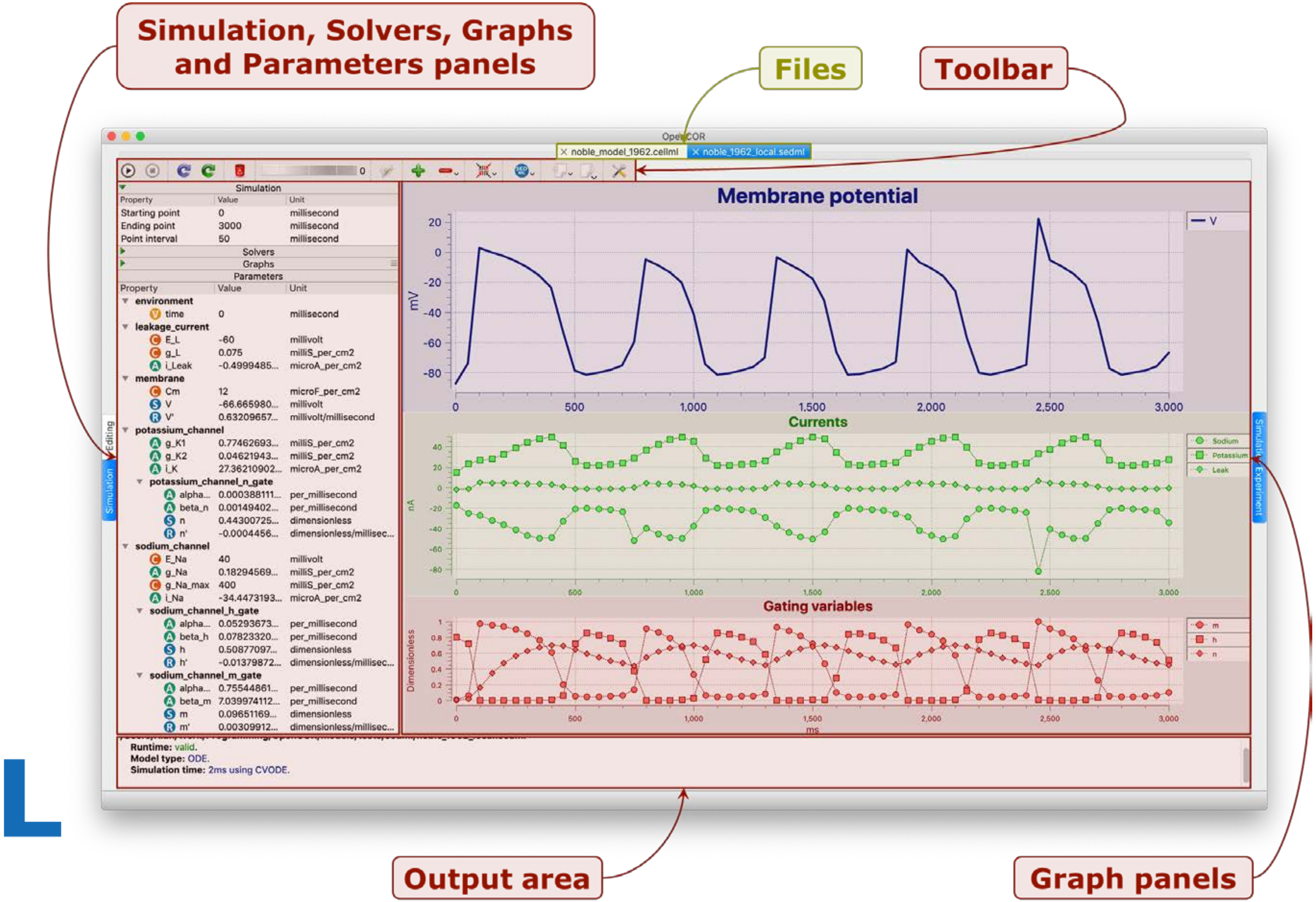
OpenCOR is a cross-platform modelling environment, which is aimed at organising, editing, simulating and analysing CellML files on Windows, Linux and macOS. It can be downloaded here.

Various information about OpenCOR and its use can be found in the following pages:

- User interfaces
  - Command Line Interface (CLI)
  - Graphical User Interface (GUI)
- Plugins

```
Jupyter QtConsole 4.5.5
Python 3.7.4 (default, Oct 18 2019, 10:16:07)
Type 'copyright', 'credits' or 'license' for
more information
IPython 7.8.0 -- An enhanced Interactive Python.
Type '?' for help.

In [1]:
```



Simulation, Solvers, Graphs and Parameters panels

Files

Toolbar

Output area

Graph panels

Simulation

Property	Value	Unit
Starting point	0	millisecond
Ending point	3000	millisecond
Point interval	50	millisecond

Solvers

Graphs

Parameters

Property	Value	Unit
environment		
Ⓥ time	0	millisecond
leakage_current		
Ⓒ E_L	-60	millivolt
Ⓒ g_L	0.075	milliS_per_cm2
Ⓐ i_Leak	-0.4999485...	microA_per_cm2
membrane		
Ⓒ Cm	12	microF_per_cm2
Ⓢ V	-66.665980...	millivolt
Ⓡ V'	0.63209657...	millivolt/millisecond
potassium_channel		
Ⓐ g_K1	0.77462693...	milliS_per_cm2
Ⓐ g_K2	0.04621943...	milliS_per_cm2
Ⓐ i_K	27.36210902...	microA_per_cm2
potassium_channel_n_gate		
Ⓐ alpha...	0.000388111...	per_millisecond
Ⓐ beta_n	0.00149402...	per_millisecond
Ⓢ n	0.44300725...	dimensionless
Ⓡ n'	-0.0004456...	dimensionless/millisec...
sodium_channel		
Ⓒ E_Na	40	millivolt
Ⓐ g_Na	0.18294569...	milliS_per_cm2
Ⓒ g_Na_max	400	milliS_per_cm2
Ⓐ i_Na	-34.4473193...	microA_per_cm2
sodium_channel_h_gate		
Ⓐ alpha...	0.05293673...	per_millisecond
Ⓐ beta_h	0.07823320...	per_millisecond
Ⓢ h	0.50877097...	dimensionless
Ⓡ h'	-0.01379872...	dimensionless/millisec...
sodium_channel_m_gate		
Ⓐ alpha...	0.75544861...	per_millisecond
Ⓐ beta_m	7.039974112...	per_millisecond
Ⓢ m	0.09651169...	dimensionless
Ⓡ m'	0.00309912...	dimensionless/millisec...

Runtime: valid.
Model type: ODE.
Simulation time: 2ms using CVODE.

SED ML

Hands on tutorial

- Using OpenCOR to explore modularity and reuse with CellML models (including SED-ML)
- Making use of PMR as a version controlled workspace to archive and share your work FAIRly
- Python-enabled OpenCOR
- Starting to explore what is possible with machine learning using TensorFlow, CellML, OpenCOR, and Python.

Alan Garny

Gonzalo Maso Talou

CENTER FOR REPRODUCIBLE
BIOMEDICAL MODELING

https://reproduciblebiomodels.org/

Center Team

Herbert Sauro
U Washington
Director

Jonathan Karr
Mount Sinai
TR&D 1

John Gennari
U Washington
TR&D 2

Ion Moraru
UConn Health
TR&D 3

David Nickerson
ABI
Curation Service

Support by NIBIB and NIGMS: NIBIB NIGMS

Goals

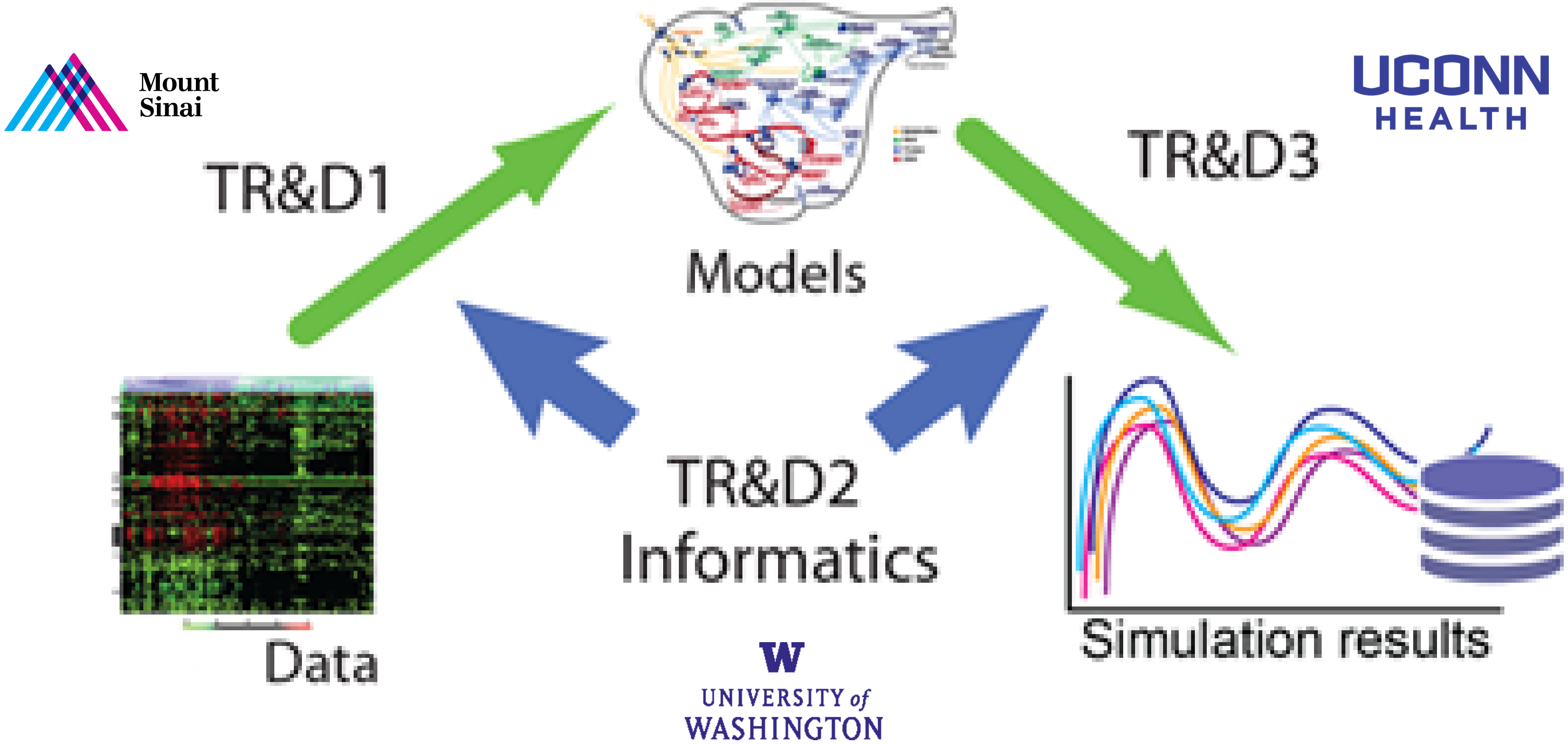
Long-term

- Enable more comprehensive and more predictive models that advance precision medicine and synthetic biology

Short-term

- Make modeling more reproducible, comprehensible, reusable, composable, collaborative, and scalable
- Develop technological solutions to the barriers to modeling
- Integrate the technology into user-friendly solutions
- Push researchers to use these tools
- Partner with journals

TR&Ds



Mount Sinai

TR&D1

Models

UCONN HEALTH

TR&D3

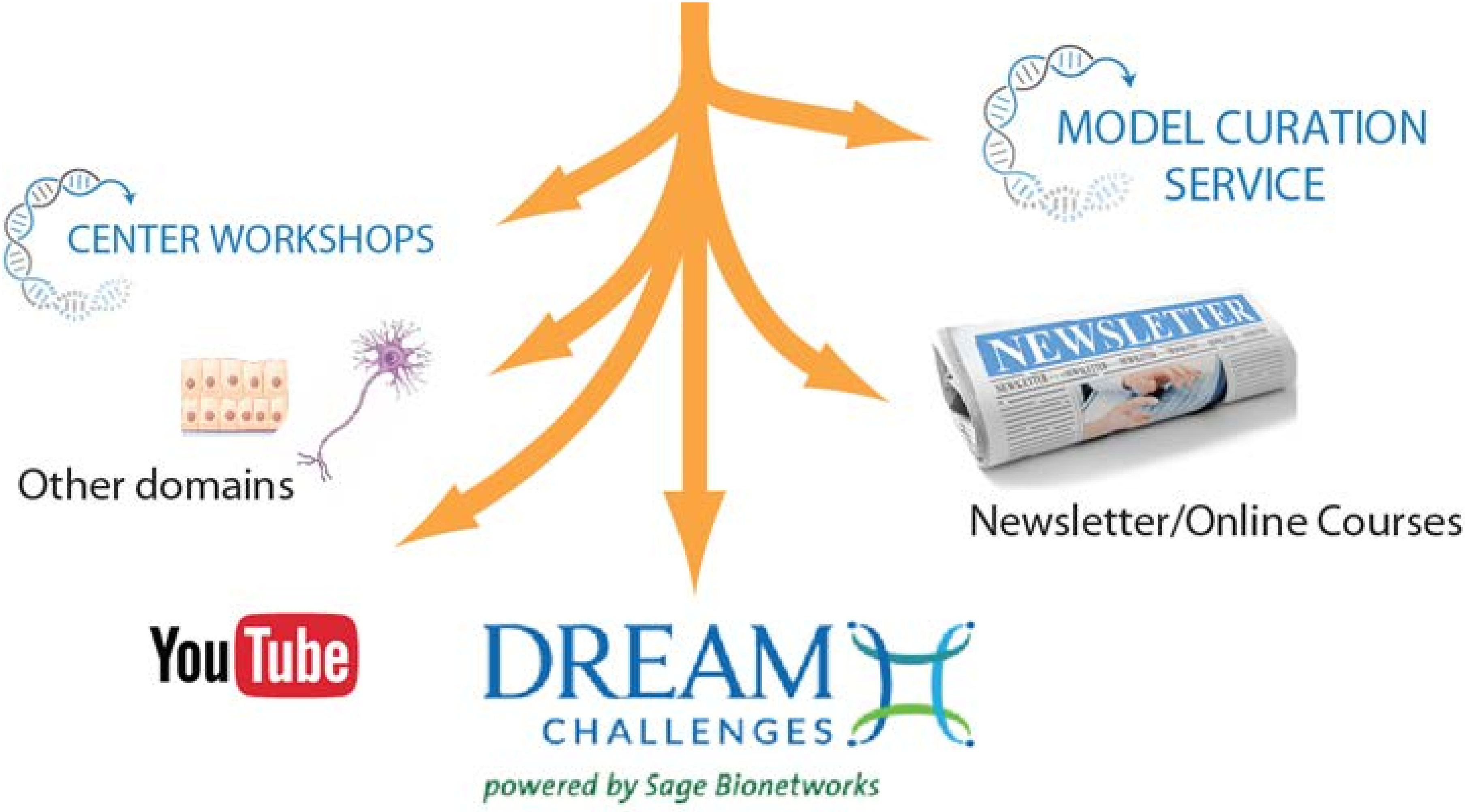
Data

TR&D2 Informatics

Simulation results

UNIVERSITY of WASHINGTON

Training and dissemination



CENTER WORKSHOPS

MODEL CURATION SERVICE

Other domains

Newsletter/Online Courses

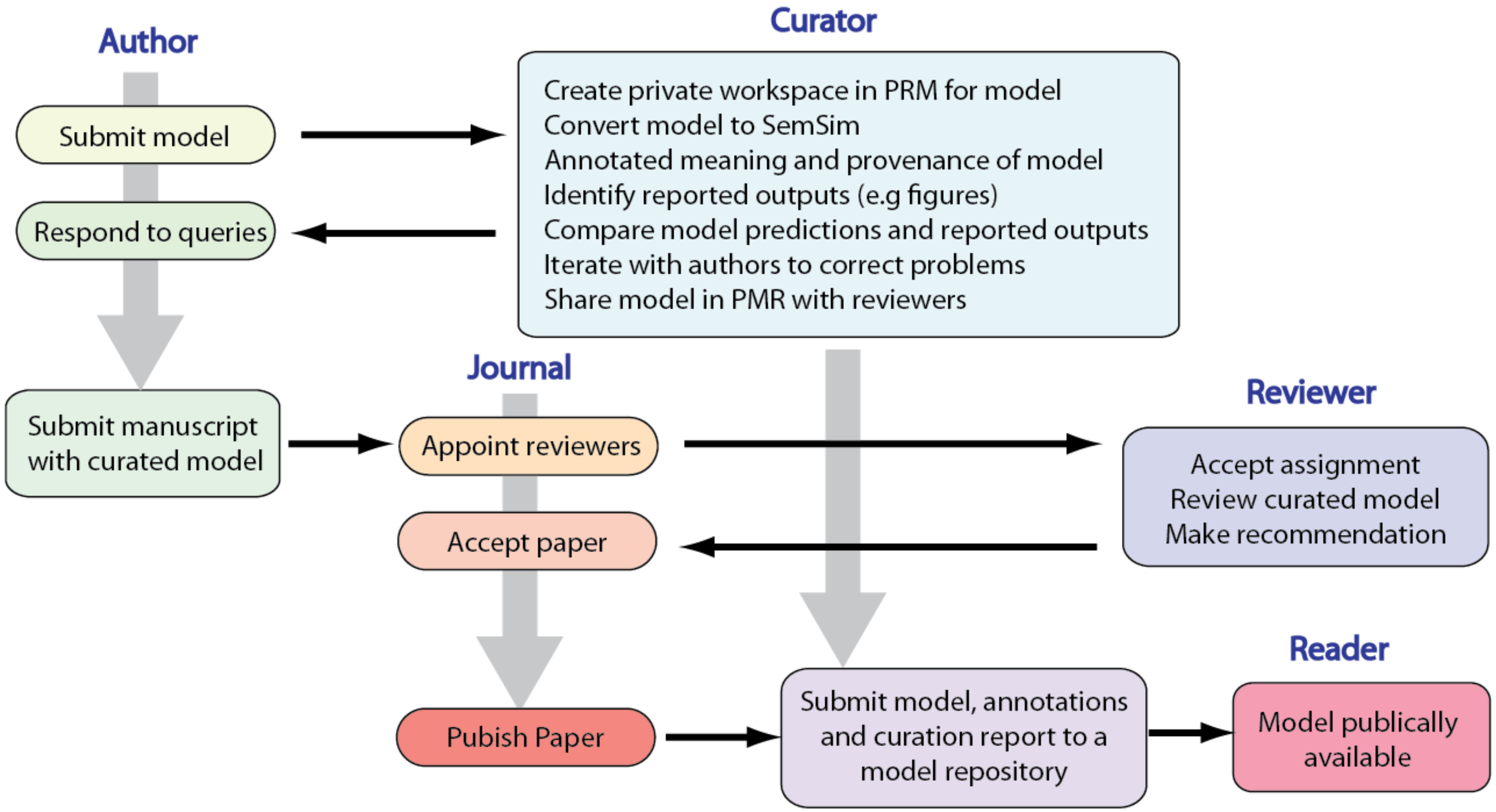
YouTube

DREAM CHALLENGES

powered by Sage Bionetworks

Curation service

Manuscripts received by journals will be curated to make sure that any author supplied code will faithfully reproduce the results presented in the manuscript.



Author
Submit model
Respond to queries
Submit manuscript with curated model
Curator
Create private workspace in PRM for model
Convert model to SemSim
Annotated meaning and provenance of model
Identify reported outputs (e.g figures)
Compare model predictions and reported outputs
Iterate with authors to correct problems
Share model in PMR with reviewers
Journal
Appoint reviewers
Accept paper
Pubish Paper
Reviewer
Accept assignment
Review curated model
Make recommendation
Submit model, annotations and curation report to a model repository
Reader
Model publically available

Acknowledgements

- Gonzalo Maso Talou
- Tommy Yu
- Alan Garny
- Peter Hunter

- ABI Physiome Group

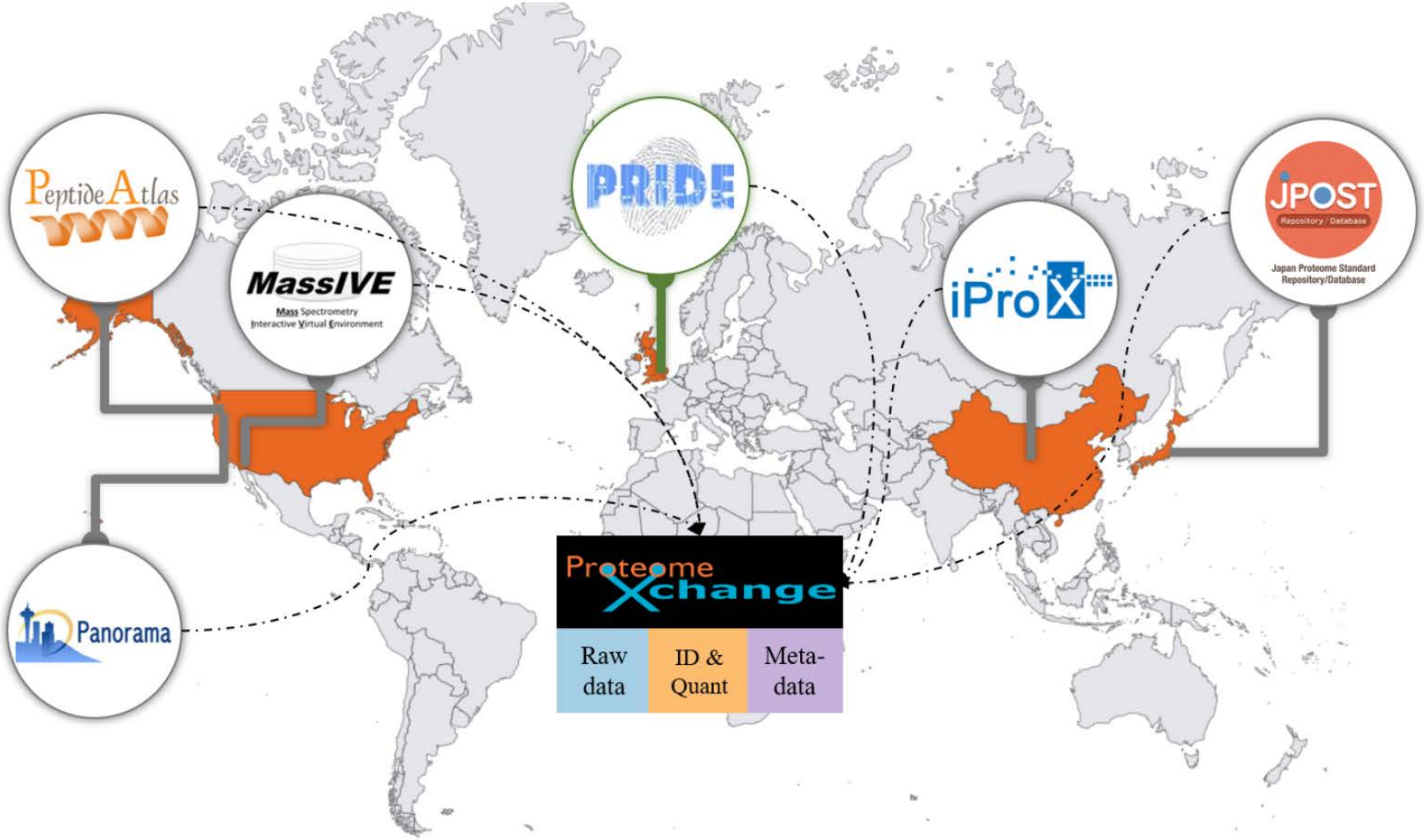
Aotearoa Foundation

SPARC

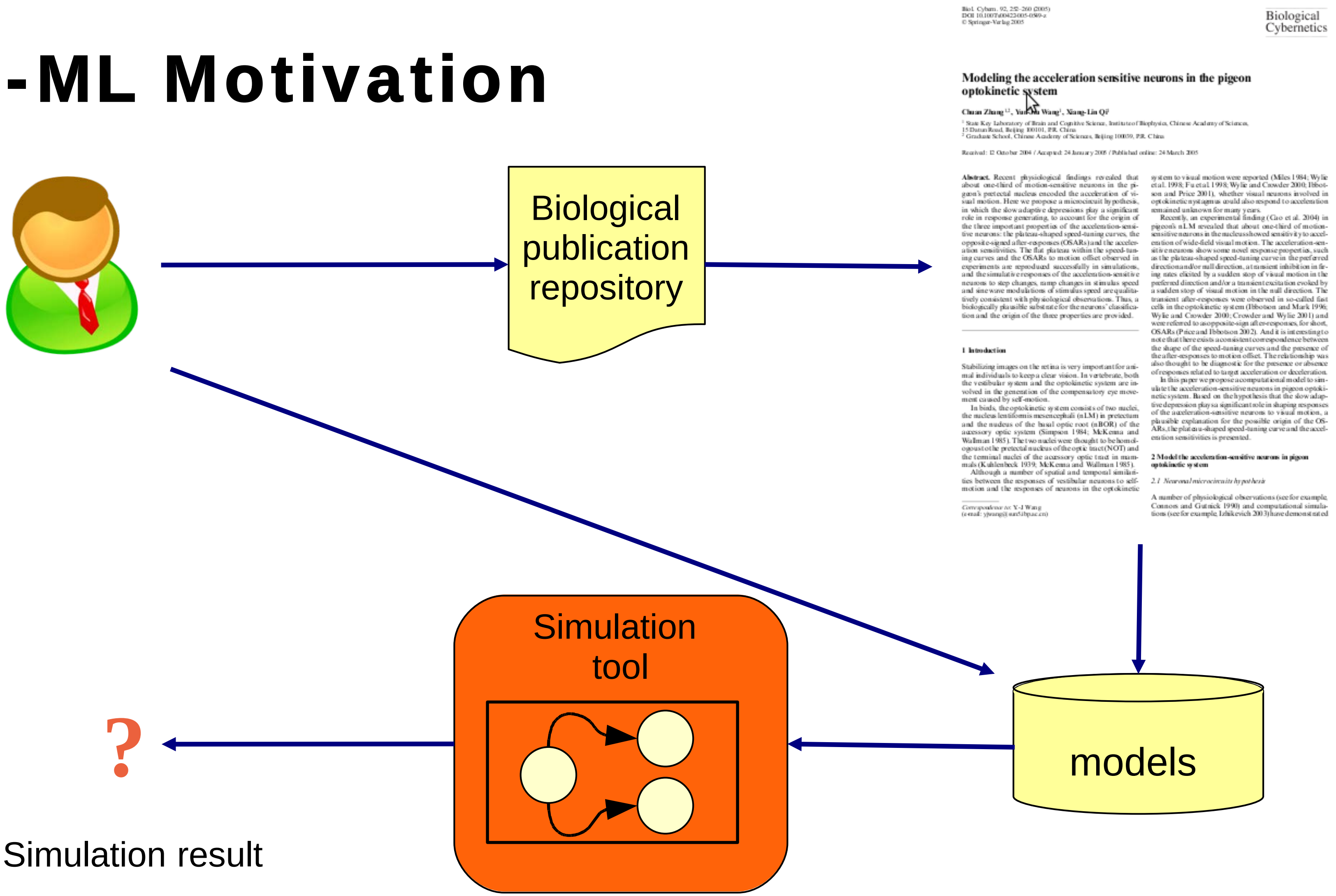
CENTER FOR REPRODUCIBLE BIOMEDICAL MODELING

https://doi.org/10.17608/k6.auckland.10080263

ModelXchange



SED-ML Motivation



Biol. Cybern. 92, 252–260 (2005)
DOI 10.1007/s00422-005-0589-z
© Springer-Verlag 2005

Biological Cybernetics

Modeling the acceleration sensitive neurons in the pigeon optokinetic system

Chuan Zhang[1,2], Yun-Jiu Wang[1], Xiang-Lin Qi[2]

[1] State Key Laboratory of Brain and Cognitive Science, Institute of Biophysics, Chinese Academy of Sciences, 15 Datun Road, Beijing 100101, P.R. China
[2] Graduate School, Chinese Academy of Sciences, Beijing 100039, P.R. China

Received: 12 October 2004 / Accepted: 24 January 2005 / Published online: 24 March 2005

Abstract. Recent physiological findings revealed that about one-third of motion-sensitive neurons in the pigeon's pretectal nucleus encoded the acceleration of visual motion. Here we propose a microcircuit hypothesis, in which the slow adaptive depressions play a significant role in response generating, to account for the origin of the three important properties of the acceleration-sensitive neurons: the plateau-shaped speed-tuning curves, the opposite-signed after-responses (OSARs) and the acceleration sensitivities. The flat plateau within the speed-tuning curves and the OSARs to motion offset observed in experiments are reproduced successfully in simulations, and the simulative responses of the acceleration-sensitive neurons to step changes, ramp changes in stimulus speed and sine wave modulations of stimulus speed are qualitatively consistent with physiological observations. Thus, a biologically plausible substrate for the neurons' classification and the origin of the three properties are provided.

1 Introduction

Stabilizing images on the retina is very important for animal individuals to keep a clear vision. In vertebrate, both the vestibular system and the optokinetic system are involved in the generation of the compensatory eye movement caused by self-motion.

In birds, the optokinetic system consists of two nuclei, the nucleus lentiformis mesencephali (nLM) in pretectum and the nucleus of the basal optic root (nBOR) of the accessory optic system (Simpson 1984; McKenna and Wallman 1985). The two nuclei were thought to be homologous to the pretectal nucleus of the optic tract (NOT) and the terminal nuclei of the accessory optic tract in mammals (Kuhlenbeck 1939; McKenna and Wallman 1985).

Although a number of spatial and temporal similarities between the responses of vestibular neurons to self-motion and the responses of neurons in the optokinetic system to visual motion were reported (Miles 1984; Wylie et al. 1998; Fu et al. 1998; Wylie and Crowder 2000; Ibbotson and Price 2001), whether visual neurons involved in optokinetic nystagmus could also respond to acceleration remained unknown for many years.

Recently, an experimental finding (Cao et al. 2004) in pigeon's nLM revealed that about one-third of motion-sensitive neurons in the nucleus showed sensitivity to acceleration of wide-field visual motion. The acceleration-sensitive neurons show some novel response properties, such as the plateau-shaped speed-tuning curve in the preferred direction and/or null direction, a transient inhibition in firing rates elicited by a sudden stop of visual motion in the preferred direction and/or a transient excitation evoked by a sudden stop of visual motion in the null direction. The transient after-responses were observed in so-called fast cells in the optokinetic system (Ibbotson and Mark 1996; Wylie and Crowder 2000; Crowder and Wylie 2001) and were referred to as opposite-sign after-responses, for short, OSARs (Price and Ibbotson 2002). And it is interesting to note that there exists a consistent correspondence between the shape of the speed-tuning curves and the presence of the after-responses to motion offset. The relationship was also thought to be diagnostic for the presence or absence of responses related to target acceleration or deceleration.

In this paper we propose a computational model to simulate the acceleration-sensitive neurons in pigeon optokinetic system. Based on the hypothesis that the slow adaptive depression plays a significant role in shaping responses of the acceleration-sensitive neurons to visual motion, a plausible explanation for the possible origin of the OSARs, the plateau-shaped speed-tuning curve and the acceleration sensitivities is presented.

2 Model the acceleration-sensitive neurons in pigeon optokinetic system

2.1 Neuronal microcircuits hypothesis

A number of physiological observations (see for example, Connors and Gutnick 1990) and computational simulations (see for example, Izhikevich 2003) have demonstrated

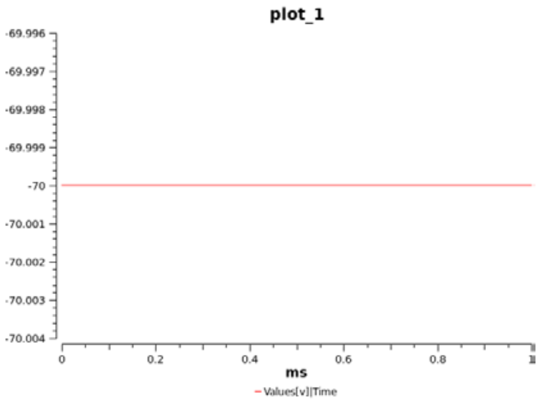
Correspondence to: Y.-J. Wang
(e-mail: yjwang@sun5.ibp.ac.cn)

Simulation result

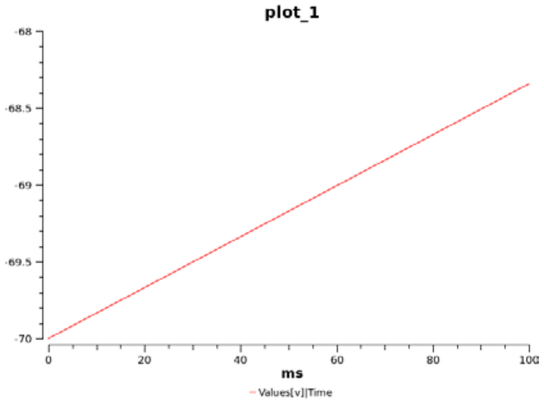
Example

First attempt to run the model, measuring the spiking rate *v* over time

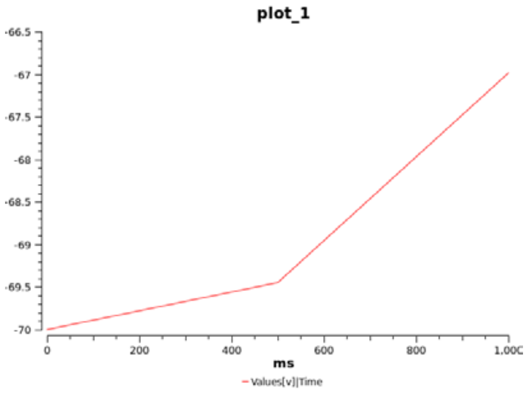
- load SBML into the simulation tool COPASI
- use parametrisation as given in the SBML file
- define output variables (**v**)
- run the time course



1 ms (standard)



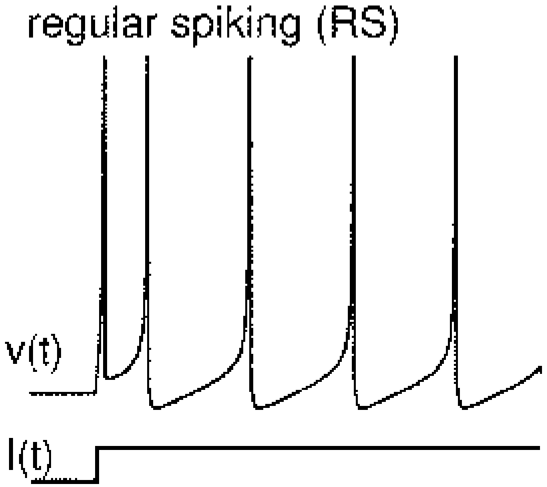
100ms



1000ms

Example

Second attempt to run the model, adjusting simulation step size and duration



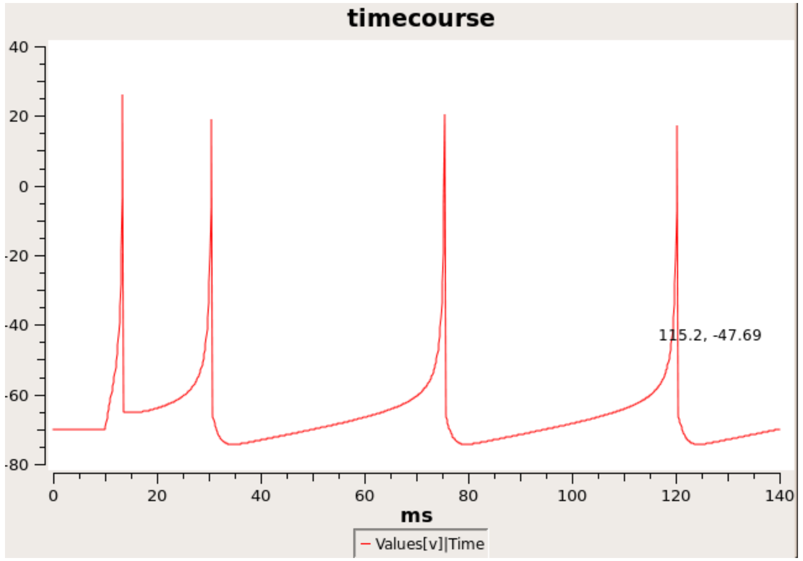
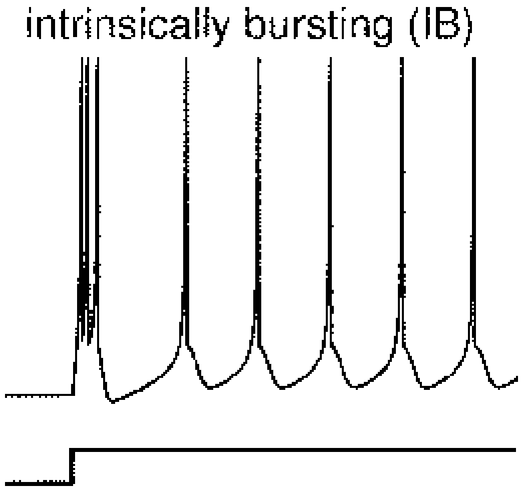


Fig.: COPASI simulation, duration: 140ms, step size: 0.14

Example

Third attempt to run the model, updating initial model parameters



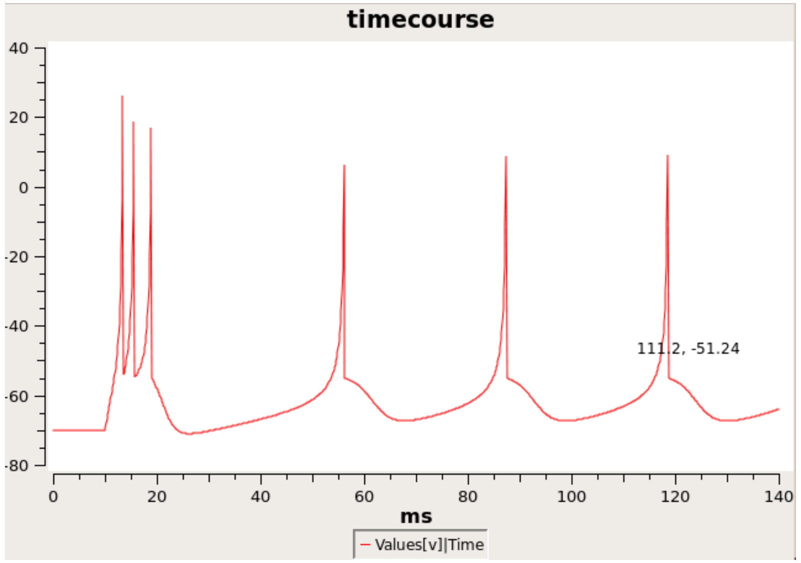


Fig.: COPASI, adjusted parameter values (a=0.02, b=0.2 **c=-55, d=4**)

https://sed-ml.org/

combine

http://co.mbine.org

combine

Core Standards

Standards for Knowledge Representation

BioPAX

SBOL

Standards for Visual Representation

SBGN

SBOL VISUAL

Standards for Models and their Analyses

SBML™

cellML

[NeuroML]

SED ML

Associated Standards

Used by core standards

Projects

Infrastructure

BioModels.net
qualifiers

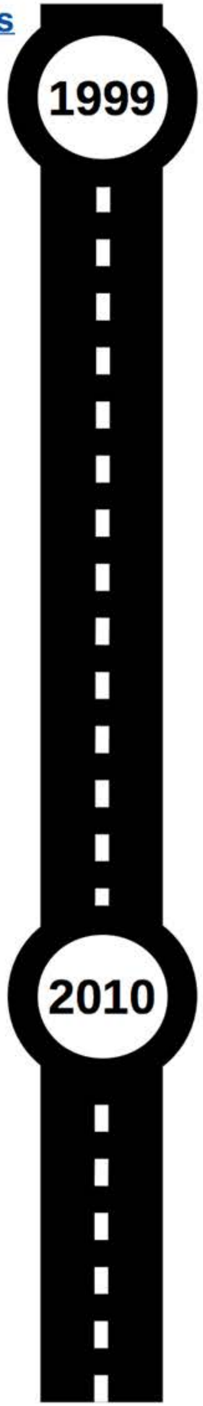
Controlled Vocabularies

SBO

combine

- Coordination board
- Coordinating new efforts, meetings, etc.
  - COMBINE Archive
  - Harmonizing annotation
  - Uncertainty?

- Publications
- Forums/mailing lists
- FAIR and FAIRsharing



Creation of COMBINE

Enabling technologies

2003
libSBML

2005
MIRIAM, SBO, BioModels qualifiers

2006
PaxTools

2007
MIASE, KiSAO

2011
Identifiers.org

September 2014
COMBINE Archive

Representation formats

March 2001
SBML Level 1

August 2001
CellML 1.0, NeuroML

June 2003
SBML Level 2

July 2004
BioPAX Level 1

December 2005
BioPAX Level 2

August 2008
SBGN PD L1

September 2009
SBGN ER L1, SBGN AF L1

March 2010
SED-ML Level 1

July 2010
BioPAX Level 3

October 2010
SBML Level 3

October 2011
SBOL v1

March 2013
SBOLvisual v1

July 2015
SBOL v2

Influential meetings

April 1999
NATO workshop, proposing to create
a language to encode metabolic models

April 2000
Start of SBML at the
1st "ERATO Kitano" workshop,

August 2002
Start of BioPAX project at the
4st Biopathway consortium meeting

July 2003
1st SBML hackathon

October 2005
Start of SBGN project at
the BioPAX face 2 face meeting

2006
Decision to create a language for synth biol designs

January 2008
Okinawa superhackathon
SBGN, BioPAX, SBO, MIRIAM

April 2008
1st SBOL meeting

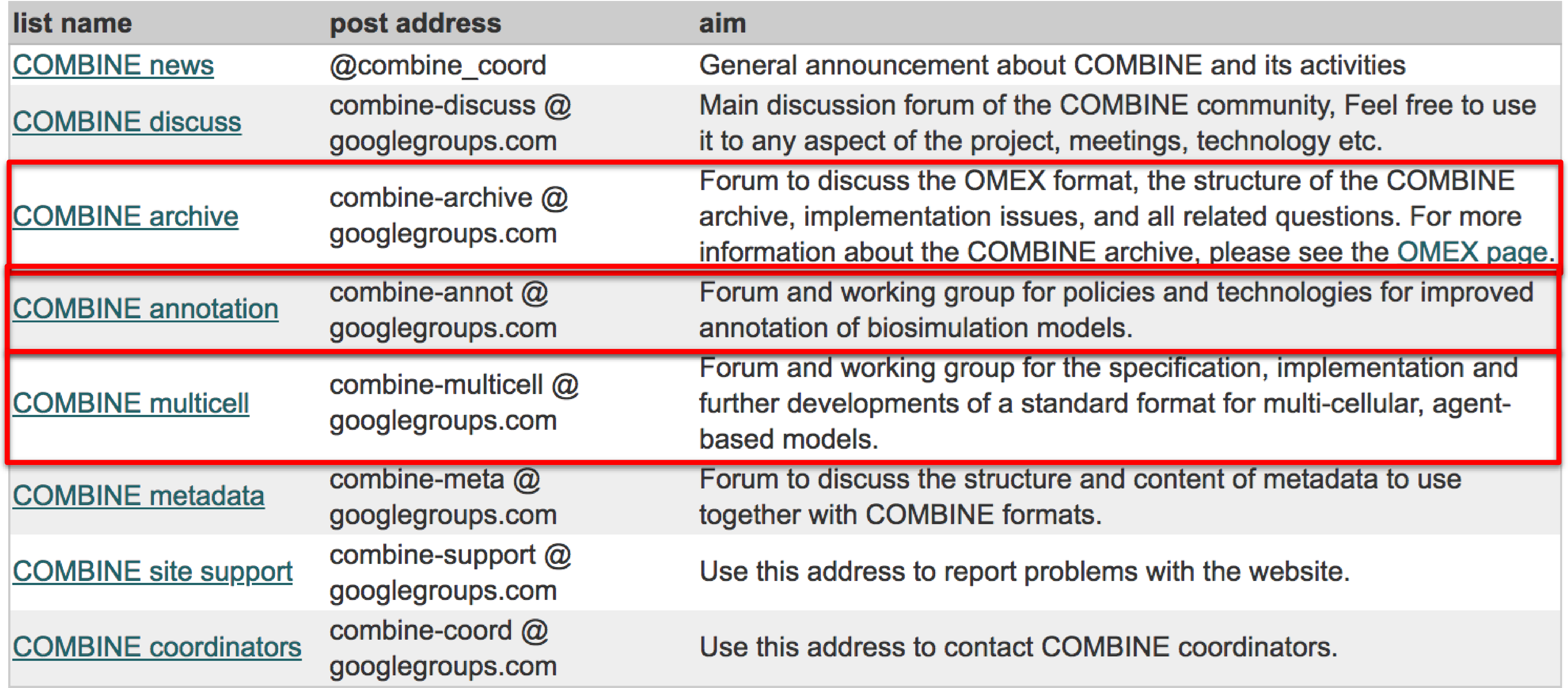
April 2009
Waiheke combined meeting
CellML, SBGN, BioPAX, SBO, MIASE

October 2010
1st COMBINE forum

April 2011
1st HARMONY hackathon

https://doi.org/10.1109/WSC.2017.8247840

Mailing lists and forums of discussion



list name	post address	aim
COMBINE news	@combine_coord	General announcement about COMBINE and its activities
COMBINE discuss	combine-discuss @ googlegroups.com	Main discussion forum of the COMBINE community, Feel free to use it to any aspect of the project, meetings, technology etc.
COMBINE archive	combine-archive @ googlegroups.com	Forum to discuss the OMEX format, the structure of the COMBINE archive, implementation issues, and all related questions. For more information about the COMBINE archive, please see the OMEX page.
COMBINE annotation	combine-annot @ googlegroups.com	Forum and working group for policies and technologies for improved annotation of biosimulation models.
COMBINE multicell	combine-multicell @ googlegroups.com	Forum and working group for the specification, implementation and further developments of a standard format for multi-cellular, agent-based models.
COMBINE metadata	combine-meta @ googlegroups.com	Forum to discuss the structure and content of metadata to use together with COMBINE formats.
COMBINE site support	combine-support @ googlegroups.com	Use this address to report problems with the website.
COMBINE coordinators	combine-coord @ googlegroups.com	Use this address to contact COMBINE coordinators.

http://co.mbine.org/comm

combine

- 10th COMBINE Anniversary
- July 15-19 in Heidelberg
- Registration now open!
- Abstract submission deadline extended to June 15!
- http://co.mbine.org/events/COMBINE_2019