UNDER REVIEW

RESCIENCE C

Reproduction / Systems neurobiology

[Rp] Reproduction of An allosteric model of calmodulin [PNAS 105(31), 10768 (2008)]

Melanie I. Stefan[1]

[1]Centre for Discovery Brain Sciences, University of Edinburgh, UK – [2]ZJU UoE Institute, Zhejiang University School of Medicine, Zhejiang University, Haining, Zhejiang, China

Edited by
(Editor)

Reviewed by
(Reviewer 1)
(Reviewer 2)

Received
–

Published
–

DOI
–

Abstract

Calmodulin is a calcium sensor with an important role in synaptic plasticity. A detailed allosteric model of calmodulin binding to Calcium, conformational transitions, and activation of two downstream targets was published by Stefan et al. 2008. Here, I am attempting to reproduce the code for this study, as part of the Rescience C 10 Years Reproducibility Challenge. This exercise confirms the advantages of the Systems Biology Markup Language (SBML) for robust and reproducible encoding of biochemical models. It also showcases the need to document code information beyond the model itself, including simulation parameters and data analysis scripts.

1 Historical context

This is an attempt to reproduce the code from the paper entitled "An allosteric model of calmodulin explains differential activation of PP2B and CaMKII" by Stefan et al. from 2008 [1] and from a correction to that article[2].
Calmodulin is a small protein that is activated upon binding to up to four Calcium ions (reviewed in [3]). Calmodulin binds to and thereby activates a large number of downstream target and is involved in a variety of Calcium-dependent cellular processes.
For neuroscientists, calmodulin is of interest because of its role in synaptic plasticity: In postsynaptic neurons, calmodulin acts as a switch, inducing either long-term strengthening or weakening of the synapse, depending on the nature of the Calcium signal (reviewed in [4]). How does this switch work? At lower Calcium frequencies, calmodulin preferentially activates the phosphatase PP2B, while at higher Calcium frequencies, it preferentially activates the kinase CaMKII (reviewed in [4]). PP2B and CaMKII activate different pathways, resulting in a decrease or increase in AMPA receptor activity, respectively, along with other synaptic changes. Interestingly, activation of either PP2B or CaMKII is most efficient when calmodulin is fully saturated with calcium, although sub-saturated forms of calmodulin have also been shown to be partially active [5]. The important role of calmodulin in synaptic plasticity has led to an interest in detailed computational modelling of calmodulin regulation [6, 7, 8, 9, 10] and the inclusion of calmodulin into many models of postsynaptic signalling pathways (for a systematic overview, see [11]).
Based on previous observations showing that calmodulin binding to calcium is cooperative [12] and observing conformational changes in calmodulin when bound to Cal-

Correspondence should be addressed to Melanie I Stefan (melanie.stefan@ed.ac.uk)
The authors have declared that no competing interests exists.

cium [13, 14], we proposed an allosteric model of calmodulin [1]. Following the Monod-Wyman-Changeux (MWC) model of allosteric regulation [15], we modelled calmodulin as existing in two possible conformational states (the low-affinity T state and the high-affinity R state). In the absence of calcium binding, a calmodulin molecule is most likely to exist in the T state. With every calcium binding event, the probability of transitioning to the R state increases.
Allosteric models of calmodulin had been proposed before [16], but the model by Stefan et al. [1] went a step further by allowing the different individual Calcium binding sites to have different affinities[17], thus allowing a very nuanced examination of the mechanisms underlying calmodulin activation.
The model was in line with experimental results on calmodulin binding to Calcium, and was also able to reproduce the preferential activation of either PP2B or CaMKII at low or high Calcium concentrations, respectively, thus mechanistically explaining the ability of calmodulin to act as a Calcium-dependent switch between different synaptic plasticity mechanisms[1].
This allosteric model of calmodulin has since served as a basis for other work modelling the mechanisms underlying synaptic plasticity [18, 19], for looking at the behaviour of ligand-receptor systems under biologically typical conditions [20, 21] and for benchmarking new modelling methodologies [22].
While useful, the model by Stefan et al.[1] has since been superseded by another allosteric model of calmodulin[23, 24]. This new model treats the N-lobe and C-lobe of calmodulin as independent parts that can undergo conformational changes independently of each other[23]. This seems plausible given the molecular structure of calmodulin [13, 14, 25] and is in line with other (non-allosteric) computational models of calmodulin regulation that have treated the N-lobe as separate and different from the C-lobe [6, 7]
For the original article, the model was written and simulated in Copasi [26] (version not specified). It was exported as an SBML L2V3 file[27, 28]. This was to ensure platform-independent reproducibility of the model.
At the time the paper was submitted, the model SBML file was deposited on BioModels (https://www.ebi.ac.uk/biomodels/)[29, 30, 31] (accession code: MODEL9885984404). On BioModels database, the model was independently curated. Curation involved reproducing some results from the paper using a different simulator.
Given that the whole purpose of the original SBML encoding and BioModels deposition was to make the model retrievable and reproducible (independent of modelling software used), chances of success in this reproduction exercise seemed fairly high.

2 Retrieval of the software

In order to reproduce the model, I looked for the model entry on BioModels using the accession code MODEL9885984404, as given in the paper[1].
The ”History” tab reveals that there are two versions of this model. The first, original one was submitted in August 2008. This is the one I downloaded. The downloaded file was an omex file archive[32]. Unzipping it reveals two metadata files, and the model file itself (file `BIOMD0000000183.xml.origin`). This is an xml file containing the model SBML. This should run with any software supporting SBML L2V3 [28]. I re-named the file to `BIOMD0000000183.xml` to facilitate import into Copasi.

3 Execution

To reproduce the model, I used Copasi 4.24 (Build 197). The model can be opened using the `File -> Import SBML` command.

The SBML file encodes the model itself, i.e. its component chemical species, their initial concentrations, reactions, and parameters governing those reactions, as well as parameters created to track specific model outcomes (for instance, average moles of Calcium bound per mole of calmodulin).
What is not captured in the SBML file is information relating to specific simulation. This includes initial concentrations or parameters where they are different from the "master" model, information on the simulation time step and total simulation time, and information on what outcomes to plot. Since this information is not provided with the model code, it has to be inferred from the article.
To see whether the code is reproducible, I set out to re-create the three results figures of the original paper.

3.1 Figure 3

(The correct version of this figure is given in [2]): This is a dose-response curve showing Calcium saturation of calmodulin (moles of Ca^{2+} bound per mole of calmodulin). The figure legend specifies that the values plotted are steady-state values and that the calmodulin concentration used in those simulations was 2×10^{-7} M. In the model as imported from BioModels, the total initial amount of calmodulin is 1.00005×10^{-5} M. The bulk of this is made up of calmodulin in the "tense" state (camT, 1×10^{-5} M), which is the prevailing state in the absence of Calcium binding . The rest is calmodulin in the "relaxed" state (camR, 5×10^{-10} M).
The dose-response curve is meant to validate the model against similar experimental dose-response curves that were obtained with mixtures of Calcium and calmodulin, in the absence of other molecules. The protocol for simulating the dose-response curve is not explicitly specified, but the x axis legend suggests that the initial Calcium concentrations tested were between 10^{-8} and 10^{-3} M. In Copasi, this can be done by running a "parameter scan", which will run a number of simulations that differ in initial values of a specified parameter (here: initial Calcium concentration) within a user-specified range. For the parameter scan, the user can select from different "tasks". One option would be to select "Steady-state", to get steady state values of the outcome of interest for each parameter configuration. But the steady-state calculations in Copasi are not always numerically stable, and this can lead to the appearance of errors. An alternative approach is to select "Time course" and, for each parameter configuration, run a time course simulation long enough to reach steady state. Which of these methods was chosen is not specified in the article, but we chose the latter option with time courses of 1000 s.
To reproduce Figure 3, I made the following changes to the model, as downloaded from BioModels database:

- Set initial concentration of camR to 0
- Set initial concentration of camT to 2e-7
- Set initial concentrations of CaMKII and PP2B to 0
- Defined a plot to show initial concentration of Calcium on the x axis (log scale), and moles Calcium bound per mole of calmodulin on the y axis
- Defined a time course to run for $1000\,s$ with a $1\,s$ interval size.
- Defined a "parameter scan" in Copasi that will simulate the system with 100 different initial Calcium concentrations between 1e-8 and 1e-3 (logarithmically distributed) and compute time courses for each initial Calcium concentration

I saved the resulting model with the simulation information as a Copasi file (`Figure3.cps`)
The results of the simulation experiment were stored in a tab-separated text file (`Figure3.txt`).

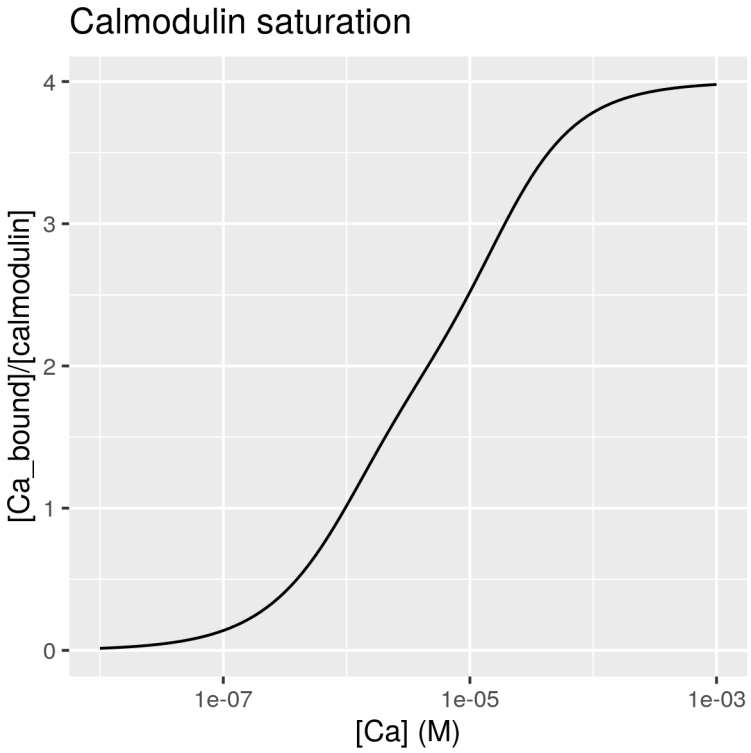

Figure 1. Calmodulin saturation as a function of Calcium concentration (reproduction of figure 3 from [1, 2]

I used R version 3.4.4 to plot the data. The code can be found in file `data_analysis.Rmd` As shown in figure 1, this does indeed produce a dose-response curve similar to that seen in the original paper[2].

3.2 Figure 4

(The correct version of this figure is given in [2]): This figure shows a similar dose-response curve as figure 3, but this time plots fractional occupancy (ratio of occupied binding sites) instead of saturation. The x axis range is wider than for figure 3, from 10^{-8} to 10^{01} M This is plotted for four different situations: Wildtype calmodulin in the absence of targets (similar to figure 3), two "in-silico mutations" where calmodulin exists in the R-state only or the T-state only, respectively, and wildtype calmodulin in the presence one of its targets, CaMKII. According to the figure legend, the total calmodulin concentration was again 2×10^{-7} M. The "R state only" and "T state only" in-silico mutations can be achieved by starting all calmodulin in one state and setting the state transition rate to 0. The concentration of CaMKII (if it is included) is not explicitly stated in the text, but is given both in the model file and in the supplementary material to the article as 7×10^{-5} M. No other target should be present, so PP2B concentration should be set to 0.
To reproduce Figure 4, four separate simulations are needed. These were produced as follows:

Figure 4a (wildtype without targets) — To reproduce the behaviour of wildtype calmodulin without targets, the following changes were made to the original model file

- Set initial concentration of camR to 0

- Set initial concentration of camT to 2e-7
- Set initial concentrations of PP2B and CaMKII to 0
- Defined a plot to show initial concentration of Calcium on the x axis (log scale), and fractional occupancy (represented by the model parameter ybar) on the y axis
- Defined a time course to run for $1000\ s$ with a $1\ s$ interval size.
- Defined a "parameter scan" that will simulate the system with 100 different initial Calcium concentrations between 1e-8 and 1e-1 (logarithmically distributed) and compute time courses for each initial Calcium concentration

This simulation setup was saved as file `Figure4a.cps`. Simulation results were stored in file `Figure4a.txt`.

Figure 4b (R state only) — To reproduce the behaviour of the "R-state only" mutation, the following changes were made to the original model file

- Set initial concentration of camR to 2e-7
- Set initial concentration of camT to 0
- Set initial concentrations of PP2B and CaMKII to 0
- Set kRT to 0
- Defined a plot to show initial concentration of Calcium on the x axis (log scale), and fractional occupancy (represented by the model parameter ybar) on the y axis
- Defined a time course to run for $1000\ s$ with a $1\ s$ interval size.
- Defined a "parameter scan" that will simulate the system with 100 different initial Calcium concentrations between 1e-8 and 1e-1 (logarithmically distributed) and compute time courses for each initial Calcium concentration

This simulation setup was saved as `Figure4b.cps`. Simulation results were stored in file `Figure4b.txt`.

Figure 4c (T state only) — Similarly, the "T-state only" mutation was achieved by making the following changes to the original model file

- Set initial concentration of camR to 0
- Set initial concentration of camT to 2e-7
- Set initial concentrations of PP2B and CaMKII to 0
- Set kTR to 0
- Defined a plot to show initial concentration of Calcium on the x axis (log scale), and fractional occupancy (represented by the model parameter ybar) on the y axis
- Defined a time course to run for $1000\ s$ with a $1\ s$ interval size.
- Defined a "parameter scan" that will simulate the system with 100 different initial Calcium concentrations between 1e-8 and 1e-1 (logarithmically distributed) and compute time courses for each initial Calcium concentration

This simulation setup was save as `Figure4c.cps`. Simulation results were stored in file `Figure4c.txt`.

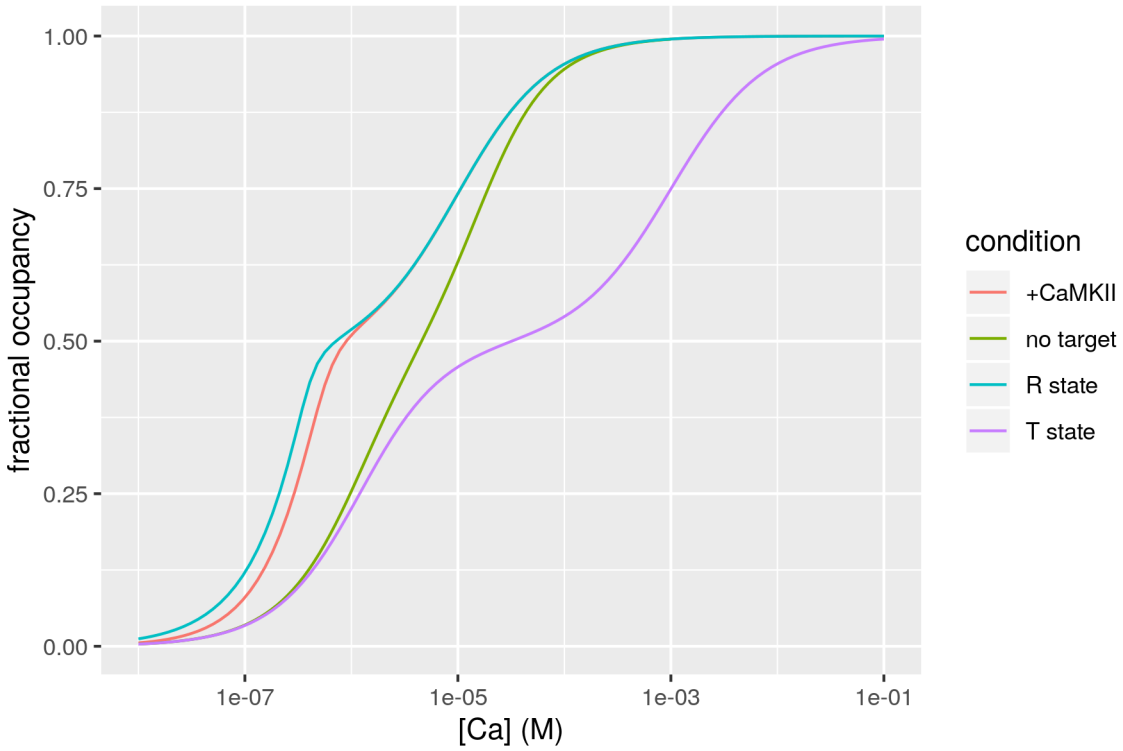

Figure 2. Shift of the binding curve toward the R state in the presence of target (reproduction of figure 4 from [1, 2]

Figure 4d (wildtype in the presence of CaMKII) – Finally, the last plot in figure 4 is wildtype calmodulin in the presence of its target and allosteric activator, CaMKII. The following changes were made to the original model file:

- Set initial concentration of camR to 0
- Set initial concentration of camT to 2e-7
- Set initial concentrations of PP2B to 0
- Defined a plot to show initial concentration of Calcium on the x axis (log scale), and fractional occupancy (represented by the model parameter ybar) on the y axis
- Defined a time course to run for 1000 s with a 1 s interval size.
- Defined a ”parameter scan” that will simulate the system with 100 different initial Calcium concentrations between 1e-8 and 1e-1 (logarithmically distributed) and compute time courses for each initial Calcium concentration

This simulation setup was saved as `Figure4d.cps`. Simulation results were stored in file `Figure4d.txt`.
Finally, data for all parts of figure 4 (a, b, c, d) was imported into R version 3.4.4 and plotted. The code can be found in file `data_analysis.Rmd`
Just like in the original paper[2], the ”R-state only” and ”T-state only” curves are theoretical extreme cases, and the actual binding curve for calmodulin moves between the two, starting in the T-state and moving towards the R state at higher Calcium concentrations. The presence of the target CaMKII shifts the binding curve towards the R state, thus CaMKII acts as an allosteric activator of calmodulin (figure 2).

3.3 Figure 5

Finally, figure 5 from the original publication[1] shows how calmodulin activates two of its downstream targets, the phosphatase PP2B and the kinase CaMKII. Again, we are shown a dose-response curve, with initial Calcium concentration on the x axis (ranging between 10^{-7} and 10^{-3} M).
The y axis shows normalised activity of target (either CaMKII or PP2B), where "activity" is defined as calmodulin binding. For both CaMKII and PP2B, the activity is normalised to the maximal activity, such that both curves have their maximum at 1. (This is to correct for the fact that cellular concentrations of CaMKII and PP2B are very different, so without normalisation, the CaMKII signal would likely drown out the PP2B signal). Though this is not explicitly stated in the paper, this normalisation must have been done after the simulation and before the simulation data was plotted.
The figure legend identified the calmodulin concentration used as 3×10^{-5} M (different from the model as it was deposited). According to the supplementary information, CaMKII concentration is 7×10^{-5} M (same as in the model file that was deposited) and PP2B concentration was 1.6×10^{-6} M (different from the 5×10^{-6} M in the model SBML file).
In order to reproduce figure 5, I therefore made the following changes in the original model SBML file:

- Changed camR to 0
- Changed camT to 3e-5
- Changed PP2B to 1.6e-6
- Defined a plot with initial Calcium concentration on the x axis (log scale), and with calmodulin-bound CaMKII and PP2B on the y axis (these are already encoded in the model variables total_CaMKII_bound and total_PP2B_bound, respectively
- Defined a time course to run for 1000 s with a 1 s interval size.
- Defined a "parameter scan" that will simulate the system with 100 different initial Calcium concentrations between 1e-7 and 1e-3 (logarithmically distributed) and compute time courses for each initial Calcium concentration

This simulation information was saved as `Figure5.cps`, and simulation results were stored in file `Figure5.txt`.
The data was imported into R version 3.4.4, normalised by dividing the PP2B and CaMKII activity data by their respective maximum, and then plotted. The code is provided in file `data_analysis.Rmd`
Unlike figure 3 and 4, I was not able to reproduce figure 5 of the paper[1]: While the model did run, and the CaMKII activation curve looks similar to the one in the paper, the PP2B activation curve is notably much flatter: even at low calcium concentration, almost all of the PP2B that will get activated is activated (see figure 3). I was not able to find the reason for this discrepancy. One possibility is that I misunderstood the "normalisation" procedure, which is not very well explained in the paper. Alternatively, the initial concentrations of CaMKII and PP2B may have been wrong (they were provided in the supplement to the paper, but if different concentrations were used for figure 5, that fact was not stated anywhere.)
All code was run on an HP Elite Book running Ubuntu 18.04.03. Copasi 4.24, R 3.4.4 and RStudio 1.2.5001 were already installed, and no additional changes had to be made to the computational environment.

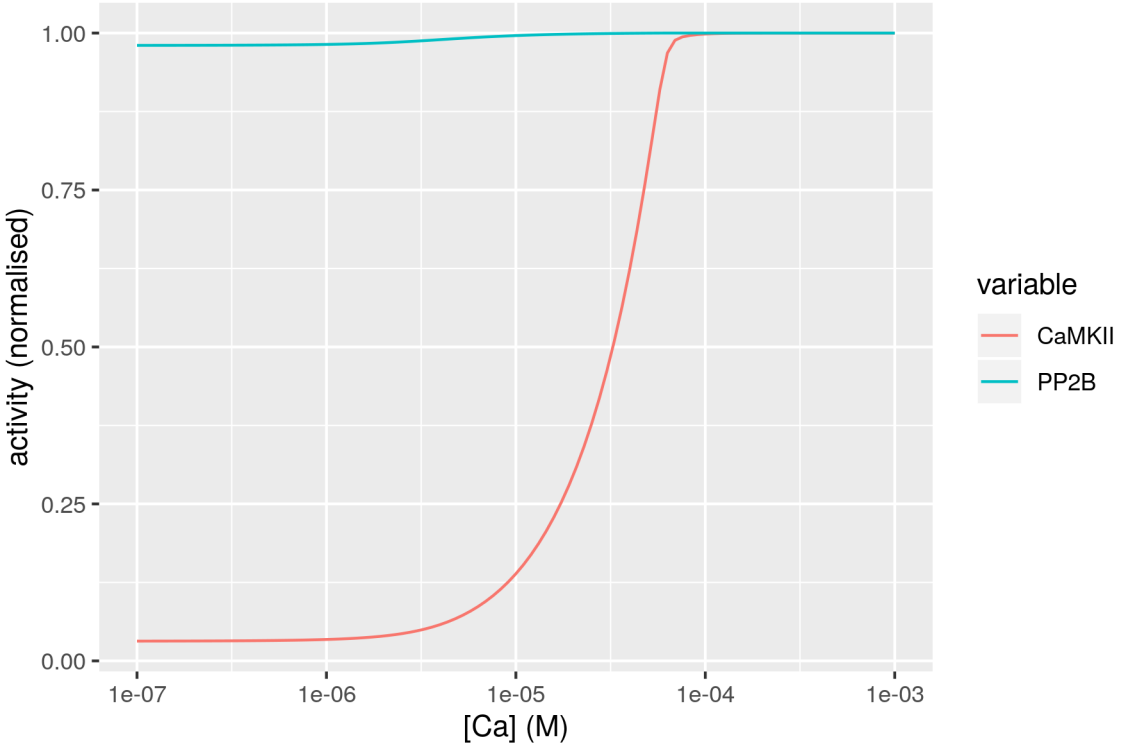

Figure 3. Calmodulin binding to CaMKII and PP2B.

4 Discussion

The Systems Biology Markup Language (SBML) was developed specifically to enable model reproducibility, sharing, and longevity[27]. Similarly, BioModels[29, 30, 31] was designed as a database to share biological models and store them over a long time period. The strength of these tools really becomes apparent now that they are more than a decade old: A user can download any SBML model from BioModels database and work with it in one of the 296 software packages that (according to http://sbml.org/SBML_Software_Guide) currently support SBML. In the present reproduction study, it was incredibly easy to obtain, open, and run the code.

One drawback of SBML is that the SBML code for a model encodes only the model itself, not additional information about specific simulation runs. Thus, while an SBML model will likely run (as it did in this reproduction attempt), it cannot by itself reproduce simulation experiments that were run on that model. For this, a description of the simulation experiment itself is needed, including information about initial conditions, simulation parameters and the output tracked. In the years since the original model was published, the Simulation Experiment Description Markup Language (SED-ML) was developed to achieve exactly this[33]. Together, SBML file, SED-ML files, and any other files pertaining to a modelling project can be stored in a structure called a COMBINE archive[32]. This is to gather all information about a modelling project in one place and thus ensure reproducibility.

For this project, I originally tried to remedy the problem by creating SED-ML files for each individual figure (or part of figure). For this, I used the "Export SED-ML" function in Copasi. It turns out though that (at least in Copasi 2.24), this does not work very well. If a .sedml file is exported and then imported again, some information is lost. For instance, if initial concentrations of a molecule were originally plotted, the re-imported .sedml plots concentrations over time. This makes a difference especially when working

with dose-response curves.
As an alternative approach, I decided to save the exact version of the model that was used to reproduce a figure as a Copasi (.cps) file. Unlike SBML, this format also stores information about plots, simulation parameters etc. The drawback is that this format is not platform-independent, whereas the combination of SBML and SED-ML would be.

Code availability

Code, data files and figures relating to this reproducibility project are available on Github: https://github.com/MelanieIStefan/calmodulin_reproduction

Acknowledgments

Lukas Endler curated the original model for BioModels Database.

References

[1] M. I. Stefan, S. J. Edelstein, and N. Le Novère, "An allosteric model of calmodulin explains differential activation of PP2B and CaMKII.," *Proc Natl Acad Sci USA*, vol. 105, pp. 10768–10773, Aug 2008.

[2] "Correction for Stefan et al., An allosteric model of calmodulin explains differential activation of PP2B and CaMKII," *Proceedings of the National Academy of Sciences*, vol. 105, no. 48, pp. 19024–19024, 2008.

[3] D. Chin and A. R. Means, "Mechanisms for regulation of calmodulin kinase IIalpha by Ca(2+)/calmodulin and autophosphorylation of threonine 286.," *Biochemistry*, vol. 41, pp. 14001–14009, Nov 2002.

[4] Z. Xia and D. R. Storm, "The role of calmodulin as a signal integrator for synaptic plasticity.," *Nat Rev Neurosci*, vol. 6, pp. 267–276, Apr 2005.

[5] J. M. Shifman, M. H. Choi, S. Mihalas, S. L. Mayo, and M. B. Kennedy, "Ca2+/calmodulin-dependent protein kinase II (CaMKII) is activated by calmodulin with two bound calciums.," *Proc Natl Acad Sci USA*, vol. 103, pp. 13968–13973, Sep 2006.

[6] S. Pepke, T. Kinzer-Ursem, S. Mihalas, and M. B. Kennedy, "A dynamic model of interactions of Ca2+, calmodulin, and catalytic subunits of Ca2+/calmodulin-dependent protein kinase II.," *PLoS Comput Biol*, vol. 6, no. 2, p. e1000675, 2010.

[7] G. C. Faas, S. Raghavachari, J. E. Lisman, and I. Mody, "Calmodulin as a direct detector of ca2+ signals.," *Nat Neurosci*, vol. 14, pp. 301–304, Mar 2011.

[8] A. G. Nair, O. Gutierrez-Arenas, O. Eriksson, A. Jauhiainen, K. T. Blackwell, and J. H. Kotaleski, "Modeling intracellular signaling underlying striatal function in health and disease," in *Progress in molecular biology and translational science*, vol. 123, pp. 277–304, Elsevier, 2014.

[9] O. Eriksson, A. Jauhiainen, S. Maad Sasane, A. Kramer, A. G. Nair, C. Sartorius, and J. Hellgren Kotaleski, "Uncertainty quantification, propagation and characterization by bayesian analysis combined with global sensitivity analysis applied to dynamical intracellular pathway models," *Bioinformatics*, vol. 35, no. 2, pp. 284–292, 2019.

[10] D. R. Romano, M. C. Pharris, N. M. Patel, and T. L. Kinzer-Ursem, "Competitive tuning: Competition's role in setting the frequency-dependence of ca2+-dependent proteins," *PLoS computational biology*, vol. 13, no. 11, p. e1005820, 2017.

[11] K. F. Heil, E. M. Wysocka, O. Sorokina, J. H. Kotaleski, T. I. Simpson, J. D. Armstrong, and D. C. Sterratt, "Analysis of proteins in computational models of synaptic plasticity," *bioRxiv*, 2018.

[12] T. H. Crouch and C. B. Klee, "Positive cooperative binding of calcium to bovine brain calmodulin.," *Biochemistry*, vol. 19, pp. 3692–3698, Aug 1980.

[13] H. Kuboniwa, N. Tjandra, S. Grzesiek, H. Ren, C. B. Klee, and A. Bax, "Solution structure of calcium-free calmodulin.," *Nat Struct Biol*, vol. 2, pp. 768–776, Sep 1995.

[14] Y. S. Babu, J. S. Sack, T. J. Greenhough, C. E. Bugg, A. R. Means, and W. J. Cook, "Three-dimensional structure of calmodulin.," *Nature*, vol. 315, no. 6014, pp. 37–40, 1985.

[15] J. Monod, J. Wyman, and J. P. Changeux, "On the nature of allosteric transitions: A plausible model.," *J Mol Biol*, vol. 12, pp. 88–118, May 1965.

[16] G. H. Czerlinski, "Allosteric competition in calmodulin," *Physiol Chem Phys Med NMR*, vol. 16, pp. 437–447, 1984.

[17] M. I. Stefan, S. J. Edelstein, and N. Le Novère, "Computing phenomenologic Adair-Klotz constants from microscopic MWC parameters.," *BMC Syst Biol*, vol. 3, p. 68, 2009.

[18] L. Li, M. I. Stefan, and N. Le Novère, "Calcium input frequency, duration and amplitude differentially modulate the relative activation of calcineurin and CaMKII.," *PLoS One*, vol. 7, no. 9, p. e43810, 2012.

[19] Y. Xie, M. Kazmierczyk, B. P. Graham, M. B. Dutia, and M. I. Stefan, "A multi-scale model reveals cellular and physiological mechanisms underlying hyperpolarisation-gated synaptic plasticity," *Preprint on bioRxiv: 418228*, p. https://doi.org/10.1101/418228, 2019.

[20] S. Edelstein, M. I. Stefan, and N. Le Novère, "Ligand depletion in vivo modulates the dynamic range and cooperativity of signal transduction," *PLoS One*, vol. 5, no. 1, p. e8449, 2010.

[21] R. Dutta Roy, C. Rosenmund, and M. I. Stefan, "Cooperative binding mitigates the high-dose hook effect," *BMC Systems Biology*, vol. 11, p. 74, Aug 2017.

[22] J. F. Ollivier, V. Shahrezaei, and P. S. Swain, "Scalable rule-based modelling of allosteric proteins and biochemical networks.," *PLoS Comput Biol*, vol. 6, no. 11, p. e1000975, 2010.

[23] M. Lai, D. Brun, S. J. Edelstein, and N. Le Novère, "Modulation of calmodulin lobes by different targets: An allosteric model with hemiconcerted conformational transitions," *PLoS Comput Biol*, vol. 11, p. e1004063, Jan 2015.

[24] L. Li, M. Lai, S. Cole, N. Le Novère, and S. J. Edelstein, "Neurogranin stimulates Ca2+/calmodulin-dependent kinase II by suppressing calcineurin activity at specific calcium spike frequencies," *PLoS computational biology*, vol. 16, no. 2, p. e1006991, 2020.

[25] Y. S. Babu, C. E. Bugg, and W. J. Cook, "Structure of calmodulin refined at 2.2 A resolution.," *J Mol Biol*, vol. 204, pp. 191–204, Nov 1988.

[26] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer, "COPASI–a COmplex PAthway SImulator.," *Bioinformatics*, vol. 22, pp. 3067–3074, Dec 2006.

[27] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, J. Doyle, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A. Cuellar, S. Dronov, E. D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J.-H. Hofmeyr, P. J. Hunter, N. S. Juty, J. Kasberger, A. Kremling, U. Kummer, N. Le Novère, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. F.Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D.Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang, "The systems biology markup language (SBML): a medium for representationand exchange of biochemical network models.," *Bioinformatics*, vol. 19, no. 4, pp. 524–531, 2003.

[28] M. Hucka, A. Finney, S. Hoops, S. Keating, and N. Le Novere, "Systems biology markup language (SBML) level 2: structures and facilities for model definitions," *Nature Precedings*, pp. 1–1, 2007.

[29] C. Li, M. Donizelli, N. Rodriguez, H. Dharuri, L. Endler, V. Chelliah, L. Li, E. He, A. Henry, M. I. Stefan, J. L. Snoep, M. Hucka, N. Le Novère, and C. Laibe, "Biomodels database: An enhanced, curated and annotated resource for published quantitative kinetic models.," *BMC Syst Biol*, vol. 4, p. 92, 2010.

[30] M. Glont, T. V. N. Nguyen, M. Graesslin, R. Hälke, R. Ali, J. Schramm, S. M. Wimalaratne, V. B. Kothamachu, N. Rodriguez, M. J. Swat, J. Eils, R. Eils, C. Laibe, R. S. Malik-Sheriff, V. Chelliah, N. Le Novère, and H. Hermjakob, "Biomodels: expanding horizons to include more modelling approaches and formats," *Nucleic Acids Research*, vol. 46, no. D1, p. D1248–D1253, 2018.

[31] R. S. Malik-Sheriff, M. Glont, T. V. N. Nguyen, K. Tiwari, M. G. Roberts, A. Xavier, M. T. Vu, J. Men, M. Maire, S. Kananathan, E. L. Fairbanks, J. P. Meyer, C. Arankalle, T. M. Varusai, V. Knight-Schrijver, L. Li, C. Dueñas-Roca, G. Dass, S. M. Keating, Y. M. Park, N. Buso, N. Rodriguez, M. Hucka, and H. Hermjakob, "BioModels—15 years of sharing computational models in life science," *Nucleic Acids Research*, 11 2019. gkz1055.

[32] F. T. Bergmann, R. Adams, S. Moodie, J. Cooper, M. Glont, M. Golebiewski, M. Hucka, C. Laibe, A. K. Miller, D. P. Nickerson, *et al.*, "Combine archive and omex format: one file to share all information to reproduce a modeling project," *BMC bioinformatics*, vol. 15, no. 1, p. 369, 2014.

[33] D. Waltemath, R. Adams, F. T. Bergmann, M. Hucka, F. Kolpakov, A. K. Miller, I. I. Moraru, D. Nickerson, S. Sahle, J. L. Snoep, and N. Le Novère, "Reproducible computational biology experiments with sed-ml–the simulation experiment description markup language.," *BMC systems biology*, vol. 5, p. 198, Dec. 2011.