OSB Automated Model Validation

Bóris Marin

NPP, University College London, London, UK

OPEN SOURCE BRAIN

![OPEN SOURCE BRAIN]

Status

Note: this table is still in development!

These status values are currently based on a manual curation process, scoring the models on the completeness of the translation of the model from the original format to NeuroML (or PyNN).

For simulators, the score represents how well the model translates to the specific simulator from the simulator independent format, NOT the availability of original scripts in that simulator.

The longer term aim is to have these scores generated automatically from tests based on parsing of the NeuroML and comparison to expected model behaviour.

	Curation	NeuroML v1.x	NeuroML v2.x	PyNN	NEURON	GENESIS 2	MOOSE	PSICS	NEST	Brian	OSB Model Validation
OSB vogelsetal2011	★★		?	★★	★★				★	★	
OSB blender-to-neuroml	★		★★								
OSB bluehive-showcase	🖼		★								
OSB brianshowcase	—		★	★						★★★	
OSB nc_ca1	★	★	?		★	?	?	?			
User nc_superdeep	🖼	?	?		?	?	?				
OSB catmaidshowcase	★		★★								
OSB cnoshowcase	★★										
OSB dbdflymodel	🖼	?	?								
OSB dentategyrus2005	★	★	?		★★	★	★				
OSB dentate	★	★	?		★	?	?				
OSB drosophila-acc-l3-motoneuron-gunay-et-al-2014	★	★	★		★	?	?				
OSB fitzhugh-nagumo-fitzhugh-1969	★		★★		★					★★	build passing
OSB fpgashowcase	🖼	?	?								build passing

Our motivations

- Manual curation / Subjective star rating
- Some form of "quality control" for OSB models
- Concrete example: how accurate are NeuroML translations?
- The danger of blindly using software

Our motivations

- Manual curation / Subjective star rating
- Some form of "quality control" for OSB models
- Concrete example: how accurate are NeuroML translations?
- The danger of blindly using software

Our motivations

- Manual curation / Subjective star rating
- Some form of "quality control" for OSB models
- Concrete example: how accurate are NeuroML translations?
- The danger of blindly using software

Our motivations

- Manual curation / Subjective star rating
- Some form of "quality control" for OSB models
- Concrete example: how accurate are NeuroML translations?
- The danger of blindly using software

Science
AAAS.ORG | FEEDBACK | HELP | LIBRARIANS
All Science Journals
Enter Search Term   SEARCH   ADVANCED
GUEST   ALERTS   ACCESS RIGHTS   MY ACCOUNT   SIGN IN
AAAS   NEWS   SCIENCE JOURNALS   CAREERS   MULTIMEDIA   COLLECTIONS   JOIN / SUBSCRIBE
Science   The World's Leading Journal of Original Scientific Research, Global News, and Commentary.
Science Home | Current Issue | Previous Issues | Science Express | Science Products | My Science | About the Journal

Retraction of Pornillos *et al.*, *Science* 310 (5756) 1950–1953.
Retraction of Reyes and Chang, *Science* 308 (5724) 1028–1031.
Retraction of Chang and Roth, *Science* 293 (5536) 1793–1800.
Science 22 December 2006:
Vol. 314 no. 5807 p. 1875
DOI: 10.1126/science.314.5807.1875b

< Prev | Table of Contents | Next >

LETTERS

Retraction

We wish to retract our research article "structure of MsbA from *E. coli*: A homolog of the multidrug resistance ATP binding cassette (ABC) transporters" and both of our Reports "Structure of the ABC transporter MsbA in complex with ADP-vanadate and lipopolysaccharide" and "X-ray structure of the EmrE multidrug transporter in complex with a substrate" (1–3).

The recently reported structure of Sav1866 (4) indicated that our MsbA structures (1, 2, 5) were incorrect in both the hand of the structure and the topology. Thus, our biological interpretations based on these inverted models for MsbA are invalid.

An in-house data reduction program introduced a change in sign for anomalous differences. This program, which was not part of a conventional data processing package, converted the anomalous pairs (I+ and I−) to (F− and F+), thereby introducing a sign change. As the diffraction data collected for each set of MsbA crystals and for the EmrE crystals were processed with the same program, the structures reported in (1–3, 5, 6) had the wrong hand.

The error in the topology of the original MsbA structure was a consequence of the low resolution of the data as well as breaks in the electron density for the connecting loop regions. Unfortunately, the use of the multicopy refinement procedure still allowed us to obtain reasonable refinement values for the wrong structures.

Validation/Testing: "Quality control"

- OSB projects are dynamic
  - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
  - *tabala rasa* environment for testing
  - (another approach: Andrew Davison's *sumatra*)

Validation/Testing: "Quality control"

- OSB projects are dynamic
  - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
  - *tabula rasa* environment for testing
  - (another approach: Andrew Davison's *sumatra*)

Validation/Testing: "Quality control"

- OSB projects are dynamic
  - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
  - *tabula rasa* environment for testing
  - (another approach: Andrew Davison's *sumatra*)

Validation/Testing: "Quality control"

- OSB projects are dynamic
    - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
    - *tabula rasa* environment for testing
    - (another approach: Andrew Davison's *sumatra*)

Validation/Testing: "Quality control"

- OSB projects are dynamic
  - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
  - *tabula rasa* environment for testing
  - (another approach: Andrew Davison's *sumatra*)

Validation/Testing: "Quality control"

- OSB projects are dynamic
  - Does a new model version behave as "expected"?
- Back our claims: does the code *really* produce figure 7?
- Will the model behave the same in different environments?
  - *tabula rasa* environment for testing
  - (another approach: Andrew Davison's *sumatra*)

Our approach

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?
- Sanity checking
    - activation variables, concentrations, temperatures. . .
- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit / sciunit*

Our approach

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?
- Sanity checking
    - activation variables, concentrations, temperatures...
- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit / sciunit*

Our approach

- *Zeroth order validation*:
  - Is the model doing what we claim it to do?

- Sanity checking
  - activation variables, concentrations, temperatures…

- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
  - Richard Gerkin's *neurounit / sciunit*

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?

- Sanity checking
    - activation variables, concentrations, temperatures…

- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit / sciunit*

Our approach

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?
- Sanity checking
    - activation variables, concentrations, temperatures…
- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit / sciunit*

Our approach

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?
- Sanity checking
    - activation variables, concentrations, temperatures…
- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit* / *sciunit*

Our approach

- *Zeroth order validation*:
    - Is the model doing what we claim it to do?
- Sanity checking
    - activation variables, concentrations, temperatures…
- Are different implementations producing the same results?
- Comparing distinct models is a "higher order" goal
    - Richard Gerkin's *neurounit* / *sciunit*

Guidelines

- Simple to use and setup. Declarative format, no logic
- Automated testing for new commits (*continuous integration*)
- Extensible (adding new tests/backends)

How does it look like

<p style="color:red; text-align:center">proof-of-concept implementation!</p>

<p style="color:red; text-align:center">expect changes!</p>

- https://github.com/borismarin/hh-testing
- https://travis-ci.org/borismarin/hh-testing

How to enable it for a project

1. Dry-run test file (*omt* extension)

```
engine: NEURON
target: NEURON/script.hoc
```

2. Add our `.travis.yaml` file to the repo

OR

1. Install our python package
2. Follow the wizard: `omv_autogen`

Model **E**mergent **P**roperties [*mep*] files

```
system: Classical HH model
experiments:
  experiment 1, autonomous activity:
    expected:
      resting: -64.9
      morphology:
        total area: 157080
        number of cells: 1
      temperature: 6.3
```

- ▶ We need to specify our expectations
- ▶ Information that comes from the model after computations
- ▶ Can be used to generate visual summary

OSB Model **T**est

```
target: hhnostim.hoc
engine: NEURON
implements:
  mep: ../hh.mep
  experiment: experiment 1, autonomous activity
  observables:
    resting:
      file:
        path: /tmp/nrnhhnostim.dat
        columns: [0,1]
        average last: 100
      tolerance: 1e-2
    morphology:
      base section: soma
```

- ▶ Given a *mep* file, specify mappings to implementation
  - ▶ how to "run"
  - ▶ how/where info is stored

Under development

- More tests!
  - BTW, what should we test for?
- More backends!
- Coupling to NeuroML exporters:
  - "download as..." available if test passes
- Unit handling

Under development

- More tests!
  - BTW, what should we test for?
- More backends!
- Coupling to NeuroML exporters:
  - "download as..." available if test passes
- Unit handling

Under development

- More tests!
  - BTW, what should we test for?
- More backends!
- Coupling to NeuroML exporters:
  - "download as…" available if test passes
- Unit handling

Under development

- More tests!
  - BTW, what should we test for?
- More backends!
- Coupling to NeuroML exporters:
  - "download as..." available if test passes
- Unit handling

Under development

- More tests!
  - BTW, what should we test for?
- More backends!
- Coupling to NeuroML exporters:
  - "download as…" available if test passes
- Unit handling

Under development

- ► More tests!
    - ► BTW, what should we test for?
- ► More backends!
- ► Coupling to NeuroML exporters:
    - ► "download as…" available if test passes
- ► Unit handling