Neuroinformatics

ISSN 1539-2791/03/01:141–144/$20.00

Commentary

ModelDB, Neuroinformatics, and Computational Neuroscience

Lyle J. Graham

Unité de Neurosciences Intégrative et Computationnelles, CNRS,
Gif-sur-Yvette, France. E-mail: lyle@cogni.iaf.cnrs-gif.fr

Neuroinformatics and the Organization of Neural Theories

The broad scope of neuroinformatics may be said to encompass not only the ontology or systematic organization of neuroscientific *data*, but also similar issues for neuroscientific *theory*. This latter aspect may be especially important to the neurosciences, primarily because the complexity and diversity of neuroscientific theory rivals that of the empirical phenomena which they strive to explain. By the same token, development of this aspect is certainly necessary in order to elevate the scientific standard of computational neuroscience to that traditionally associated with electrophysiology, imaging, and other more classical investigative methods (Borg-Graham, 2000).

One practical contribution to the organization of neural theory is ModelDB, described in the article in this issue by Migliore and colleagues (2002). ModelDB provides a sophisticated database for models for neurons, their cellular components (channels, synapses, etc.), and networks, allowing cross-referencing according to standard criteria (cell type, system, author, simulation system, etc).

A Consumer's Report of ModelDB

From a user's perspective, ModelDB provides a welcome and promising first step in the systematization of neuronal models. To date, there is no equivalent central resource for computational neuroscience models that emphasize biophysical detail. This has two immediate applications. First, for those interested in building on previously described models, ModelDB is an obvious resource to check if source code is already available. More generally, there are hardly any resolved questions in computational neuroscience, which means that for any topic there are bound to be at least two explanations (that is, models). In these cases, ModelDB will undoubtedly provide a convenient utility, allowing an immediate snapshot of various hypotheses relevant to a given subject query.

There are three basic provenances for the models in ModelDB: 1) model code taken directly from published simulation studies; 2) (unpublished) models that are derived from physiological data papers; and 3) models of models (also unpublished), wherein model parameters taken from published simulation studies (e.g., classics like Hodgkin and Huxley, 1952 and Cooley and Dodge, 1966) have been recoded (e.g., in Neuron code).

These last two cases represent one of the most impressive aspects of this database—such derivations or translations represent a very significant investment of time and effort (apparently done mainly by Michael Hines and Michelle Migliore; email contact references are included in the Readme files), but without the usual credit of a published article—an example, perhaps, of the positive side of public domain software development. Overall, the maintainers of ModelDB state that all models included in the database are from peer-reviewed papers, and furthermore that models have been "verified." While this ambitious and extraordinarily important feature provides some comfort, it remains that verification is a tough challenge. That being said, a welcome addition is that many of the model Readme files include known discrepancies between results from the ModelDB files and those reported in the published reference, which should save the investigator much head-scratching.

ModelDB may also be seen as a case study for exploring the ways such efforts may evolve. Understandably, ModelDB is a work in progress. Although its organization seems fairly straightforward, a simple tutorial and example would help the new user. One might also quibble at some of the design decisions, starting with the very generic name "ModelDB" (obviously, computational neuroscience holds no monopoly on models!). There are also some rough edges in cross-references and the graphical interface.

ModelDB: A Component of SenseLab

Given the presentation and broader design goals of ModelDB, it makes sense to discuss it in the context of its parent project, SenseLab. Here, the presentation, at least with respect to neuron models, is somewhat less successful than when ModelDB is considered alone. SenseLab includes several related databases, three "Neuronal" (CellPropDB, NeuronDB, and ModelDB), and three "Olfactory" (OrDB, OdorDB, and OdorMapDB). Confusion immediately arises from the obvious focus on the olfactory system for the entire project, whereas the Neuronal databases are, supposedly, neuronal. But even then, CellPropDB claims to (eventually) cover all cell types, not solely neurons. It is also not altogether clear where the dividing lines between the databases are drawn. For example, CellPropDB covers data "regarding membrane channels, receptor and neurotransmitters," while NeuronDB covers "three types of neuronal properties: voltage gated conductances, neurotransmitter receptors, and neurotransmitter substances."

There is an implicit hierarchy between the databases: The data in CellPropDB is incorporated in NeuronDB, and models corresponding with many of the neurons in NeuronDB make up ModelDB. SenseLab also includes the "Membrane Properties Resource," which is "provided with an inventory of all major types of membrane properties related to the [six databases mentioned earlier], including ion channels, membrane receptors, and neurotransmitters and neuromodulators." Thus, where one database ends and another starts is somewhat opaque, leading to an uncomfortable feeling that a given query may not be well-posed by the user. It is not surprising, then, that the neophyte quickly encounters labyrinthine loops while exploring the SenseLab family of sites. This might also reflect to some extent the nascent state of neuroscience ontology, a point emphasized by Gardner and colleagues of the BrainML project (Gardner, 2002).

The rather limited sets of categorizations for any given property for the databases (e.g., three "canonical" anatomical forms of neuron anatomies, according to NeuronDB) are understandable, given the fluid nature of the project as well as the field itself. Yet, as presented, the non-expert may be lulled into thinking that these concepts are actually well established, which, in reality, is untrue.

The Limitation of Source Code: Towards High-Level Declarative Neuron Models

Currently, the model form in ModelDB consists of explicit source code, which by definition is coupled to a specific simulation environment (while, in principle, code from any simulation system may be accommodated, at present almost all models are for the Neuron simulation environment). While certainly not imposed by ModelDB, this fact of life for computational neuroscience models provides the practical limitation that a given investigator may not have the appropriate tool at hand to evaluate a given model description. But this touches on a more fundamental point: that of model validation, which will be an increasingly important problem as no formal methods are likely for complex neuronal models. Here, it may be argued that cross-validation of models by comparing results from different analysis tools is the most realistic approach, but the tying of a given model to a particular software implementation only makes this process more difficult.

Thus a major hurdle remains to develop and adapt a transparent layer which isolates model conception from model implementation: One strives for model representations that are more declarative (emphasizing *what* kind of model is desired) rather than imperative (emphasizing *how* to construct a model). Ideally, any specific model implementation should be readily, if not automatically, translatable to some standard representation, which would then be amenable to any simulator package equipped to parse the standard syntax.

In that light, projects that aim to address model representation are promising, in particular the NeuroML project (neuroml.org, Goddard, et al., 2001; *see also* the related Modeler's Workspace project at modelersworkspace.org). NeuroML aims to provide a high-level description language for neuron models that will allow evaluation of any given model on a variety of simulator platforms. Although still at the "vaporware" stage, several simulator projects (e.g., Neuron, GENESIS, and Surf-Hippo) are developing the necessary interfaces for NeuroML. These efforts by simulator developers are necessary since any automatic translation going from the most common model languages (that is, those defined by the various simulation programs) to a standard syntax is much more difficult than the opposite translation.

It is clear that ModelDB will be an excellent platform via which to apply such tools. The basic framework is there, and with that the creators of ModelDB have made an extremely important contribution to the "neuroinformatization" of computational neuroscience.

References

Borg-Graham, L. (2000) Facilitating the science in computational neuroscience. *Nature Neuroscience* 3(supp.):1191.

Cooley, J. and Dodge, F. (1966) Digital computer solutions for excitation and propagation of the nerve impulse. *Biophysics Journal* 6.

Gardner, D. (2002) Open and interoperable design for neuroinformatics. Symposium on Neuroinformatics: bioinformatics for the neurosciences?, Federation of European Neuroscience Societies Meeting, Paris, France.

Goddard, N. H., Hucka, M., Howell, F., Cornelis, H., Shankar, K., and Beeman, D. (2001) Towards NeuroML: model description methods for collaborative modelling in neuroscience. *Philos Trans R Soc Lond B Biol Sci* 356(1412):1209–1228

Hodgkin, A. L. and Huxley, A. F. (1952) A quantitative description of membrane current and its application to conduction and excitation in nerve. *J Physiol* 117:500–544.

Migliore, M., Morse, T. M., Davison, A. P., Marenco, L., Shepard, G. M., and Hines, M. L. (2002) ModelDB: making models publicly accessible to support computational neuroscience *Neuroinformatics* 1:135–140