Quantitative Symbolic Process Models: How a Fair Fraction of Intelligence Could be Abstracted in AI Research

Eric Mjolsness
Departments of Computer Science and Mathematics
University of California, Irvine
emj@uci.edu

Abstract

Quantitative symbolic process models are proposed as a high-level model representation for probabilistic causal knowledge and know-how. These representations have suitable expressive power and mathematical foundation to serve as an abstraction of intelligence in much of AI research.

Extended Abstract

The symposium question “How should intelligence be abstracted in AI research?” is an attractive one for anyone with a favorite Artificial Intelligence representation framework to propagate. The success of graphical models in machine learning provides one example of how influential such paradigms can be. Others include neural networks, Lisp programming, logic, semantic networks, search algorithms, partially observable Markov decision processes, and so on. Even so there is no basis for a single approach to be adopted exclusively. Here we look to some less familiar abstractions developed mainly for quantitative scientific modeling of complex biological systems, that merge symbolic computer algebra with predictive numerical computing. These process-modeling abstractions could provide an encompassing framework that enables model reduction and other model-centric approaches to some of the classic topics of AI.

In AI, a language for representing processes in general could be applied to representing causal knowledge, such as embodied for example in the differential equations of mechanics; generative statistical models for pattern recognition; and procedural knowledge: knowledge of how to do something, including recipes, control laws and game strategies. On the other hand declarative knowledge is not entirely omitted, since the valid inferences of any logic can also be represented as elementary steps in a symbolic logical deduction process. A first desideratum for an AI-adequate process-modeling language is that all these disparate kinds of processes should be concisely representable. A second desideratum, best motivated from scientific applications, is closure under foreseeable multiscale model reductions (discussed in (Mjolsness 2012)). A third desideratum, clearest from the point of view of programming language theory, is a compositional language structure with a correspondingly compositional semantics.

All of these desiderata for a general-purpose quantitative process-modeling language can be met (for example) by starting with “Dynamical Grammars” (Mjolsness and Yosiphon 2006) (Yosiphon 2009), recursively structured grammars that specify hybrid deterministic/stochastic dynamical systems by using discrete rewrite rules modulated by continuous-valued algebraic expressions for rates. The compositional semantics of Dynamical Grammars (DGs) is given in terms of an operator algebra, so that each rule in a grammar determines a linear time-evolution operator on probability distributions over a state space, and if time is modeled continuously then the various operators just sum up. Though a different name was used in (Mjolsness 2012), here I will call the general class of models in such languages “Quantitative symbolic process models” (QSPMs).

The ontology of this kind of modeling language is, coincidentally, grammatical. Just as there are verbs, nouns, and prepositions in natural language, so there are processes, objects, and relationships in DGs or QSPMs. Processes have the semantics of time-evolution operators in a ring of operators. Objects are represented as grammatical terms bearing parameters, starting with integer- and real-valued base types. These operators can define either continuous or discrete changes of state, the former represented in the language by differential equations and the latter by rewrite-rule firings corresponding to discrete events. Variable-binding in rules is an emergent property of summation or more generally integration of operators over object parameter values, in the operator semantics of grammar rules. Relationships between objects can be specified concretely by shared values of discrete-valued parameters. Meta-rules can operate symbolically on rules, allowing symbolic DG rules (including their algebraic rate expressions) to be treated as another data type and enabling metamodels inspired by biological development. These features have all been implemented within a computer algebra system (Yosiphon 2009). Potential generalizations of the allowable set of object types (nouns) are described in (Mjolsness 2010).

For example, a Markov chain on a sparse set of transitions could be represented by the single rule

{state(j), M=transition(i, j, r)→state(i), M **with** r}.

Likewise a fixed Hopfield-style continuous-time analog neural network could be represented by two (unordered) rules:

{in=neuron(u_{in}, v_{in}), out=neuron(u_{out}, v_{out}),
C= connection(in, out, T)
→ in, C, out=neuron (u_{out} +du_{out}, v_{out})
solving $du_{out}/dt = T\, v_{in}$,
neuron(u, v) → neuron(u, v+dv)
solving $dv/dt= k_{fast}\,(g(u)-v)$} .

Similar encodings could be given for genetic algorithms, logical inference, network graph grammars, visual grammars (Mjolsness 1992), and structural graph matching. A more complex scientific application example is the detailed biochemical/biomechanical plant root development model of (Mironova et al. 2010) (Mjolsness 2013).

Alternatively, relationships between DG objects could be formulated more abstractly with graphs and graph grammar rule syntax. DGs themselves could be given an explicitly graphical representation similar to Petri nets: Object terms would be red nodes and process rules are blue nodes in a directed bipartite labelled graph. Object nodes would be labelled with the parameter type signature, and process nodes with the required binding constraints (perhaps represented as a graph) and symbolic-algebra rate expressions. So far these more graphical representations of DGs have not been explicitly implemented.

As the foregoing simple examples show, QSPMs are expressive both for models of the natural world and for other AI frameworks. Maximum likelihood parameter estimation from data is also possible, at least for a much-simplified version of the operator algebra semantics in stochastic chemical kinetics (Wang et al. 2010).

One additional concept is essential to the present vision: model reduction. As practiced eg. in (Johnson 2012), a reduced model can be trained to roughly predict, out-of-sample and under some well-defined conditions, the time course of selected observables in a much larger dynamical stochastic process model. With model reduction, developmentally-inspired metamodels, and subgrammar calls (Yosiphon 2009), an entire ecology of semantically interrelated high-level models could be usefully evolved. Or so we may expect – only future experiment can tell. If so, QSPMs have the potential to translate many external, application-specific problems into internal, AI-general problems whose technical solution will thereby gain wider impact.

References

Mjolsness E., 2012. “Compositional Stochastic Modeling and Probabilistic Programming”, Workshop on Probabilistic Programming, Neural Information Processing Systems Conference Workshops, extended abstract, December 2012. Available as arXiv:1212.0582.

Mjolsness, E., and Yosiphon, G. 2006. Stochastic Process Semantics for Dynamical Grammars. *Annals of Mathematics and Artificial Intelligence*, 47(3-4).

Yosiphon, G. 2009. Stochastic Parameterized Grammars: Formalization, Inference, and Modeling Applications. PhD diss., UC Irvine Computer Science Department, June 2009. Thesis and software: http://computableplant.ics.uci.edu/~guy/Plenum.html.

Mjolsness, E. 2010. Towards Measurable Types for Dynamical Process Modeling Languages. *Electronic Notes in Theoretical Computer Science (ENTCS)*, vol. 265, pp. 123-144, 6 Sept. 2010, Elsevier. DOI 10.1016/j.entcs.2010.08.008.

Mjolsness E., 1992. Visual Grammars and Their Neural Nets. *Advances in Neural Information Processing Systems 4*, editors Moody, Hanson and Lippmann, pp. 428-435, Morgan-Kaufman.

Mironova V.V.; Omelyanchuk, N.A.; Yosiphon, G.; Fadeev, S.I.; Kolchanov, N.A.; Mjolsness, E.; and Likhoshvai, V.A. 2010. A plausible mechanism for auxin patterning along the developing root. *BMC Systems Biology* 4:98.

Mjolsness, E., 2013. Time-Ordered Product Expansions for Computational Stochastic Systems Biology. *Physical Biology*, 035009, June 2013.

Wang, Y.; Christley, S.; Mjolsness, E.; and Xie, X. 2010. Parameter inference for discretely observed stochastic kinetic models using stochastic gradient descent. *BMC Systems Biology* 4:99.

Johnson, G.T., 2012. Dependency Diagrams and Graph-Constrained Correlation Dynamics: New Systems for Probabilistic Graphical Modeling. PhD diss., UC Irvine Computer Science Department UC Irvine, March 2012. (Chapters 5-7.) Available from: http://emj.ics.uci.edu/ → PhD Theses.