Translating BNGL models into Kappa our experience

Kim Quyen Ly

Translating BNGL models into Kappa Our experience[1]

Kim Quyên Lý[2]

Département d'informatique
École normale supérieure,
Inria, École normale supérieure, CNRS, PSL Research University, 75 005 Paris, France

Abstract

So as to test the Kappa development tools on more examples, we translated the models provided with the BNGL distribution, into Kappa. In this talk, we report about our experience. The translation was quite straight-forward except for few interesting issues that we detail here. Firstly the use of static analysis has exposed some glitches in the modelling of some pathways in the models of the BNGL distribution. We explain how static analysis has helped us to detect, locate, and correct these flaws. Secondly, expanding BNGL rules using equivalent sites into rules with uniquely identified sites is not so easy when one wants to preserve faithfully the kinetics of interactions. We recall the semantics of BNGL for equivalent sites, and explain how to perform such translation.

Rule-based languages propose a formal setting to describe and assemble large systems of mechanistic interactions between proteins. Site-graph rewriting languages, such as BNGL [1,10] and Kappa [8,9], focus on the bio-chemical structure of protein compounds, described as graphs, where nodes may denote some proteins and edges may denote some bindings between specific loci in their amino acid chains. Interactions are induced by context-free rewriting rules that model transformations in the biochemical structure of bio-molecular species. Rules offer a local perspective on bio-chemical interactions. There is no need to specify the whole conformation of bio-molecular species. Instead, rules focus only on the information that matters for the interaction to happen (or to define its kinetics).

Although they look very similar, BNGL and Kappa have some differences. From a language point of view, apart from the capability to attach global properties to connected components, each BNGL feature may be encoded in Kappa and conversely. In particular, both Kappa and BNGL allow for giving different rate kinetics depending on whether a rule is applied in a unary or in a binary context (i.e. if

[1] This material is based upon works partially sponsored by the Defense Advanced Research Projects Agency (DARPA) and the U. S. Army Research Office under grant number W911NF-14-1-0367. The views, opinions, and/or findings contained in this article are those of the authors and should not be interpreted as representing the official views or policies, either expressed or implied, of DARPA, or the U. S. Department of Defense.

[2] Email: quyen@di.ens.fr

several connected components in the left hand side of a rule embed into different regions of a same bio-molecular instance or into some regions of different bio-molecular instances). BNGL also allows the end-user to specify equivalent interaction sites within a given agent, which is not possible in Kappa, but can be replaced by the enumeration of a class of rules equal up to permutation of equivalent sites.

We have translated each model from the BNGL distribution [2] into Kappa. We have validated our translation thanks to the following two steps procedure. Firstly, we have run Kappa static analyses [7,14,13] on the so-obtained Kappa models in order to detect potential bugs. Then, we have translated the BNGL and the Kappa models into reaction networks [1,6] and we have compared the underlying differential trajectories. We have corrected the models until getting the same trajectories. For the biggest models, we have not succeeded in integrating the underlying ODEs, instead we have validated the models block-wise (by adjusting the initial state, so as to make each rule applicable).

Static analysis tools infer automatically dead rules [7,14] and separating transitions [13]. A dead rule, is a rule that will be triggered in no trace starting from the initial state. This is usually due to an encoding error: either a misspelling in the name of a protein or of a site, or a site that is used instead of another one in a rule. A separating transition makes a bio-molecular species go from one configuration to another one with no potential coming back (in one or several steps). This is unusual in models of signalling pathways.

Static analysis has discovered dead rules in every model in which the protein `Fyn` was involved. Indeed the protein `Fyn` could never be phosphorylated because its phosphorylation rules were requiring a receptor with a site `b` phosphorylated and bound to the site `U` of a protein `Lyn`. In the models, the protein `Lyn` has two sites `U` and `SH2`. Looking in the invariant inferred by the analyser, we noticed that these sites are mutually exclusive (at most one of them may be bound simultaneously). This is consistent with the modelling assumptions that have been taken in [11] so as to prevent polymers to be former. We also noticed that both sites are dedicated to the binding to the site `b` of the receptor. Yet, the site `U` may be bound to the site `b` of a receptor only whenever this site is not phosphorylated. Whenever the site `b` is phosphorylated, the site `SH2` shall be use instead. We have corrected the models accordingly, both in BNGL and in Kappa, by testing a bond between the site `b` of the phosphorylated receptor and the site `SH2` of `Lyn` instead.

Then, we have compared the trajectories in the differential semantics. We have used the BNGL perl script [1] to compile BNGL models into reaction networks and KaDE [6] to compile Kappa models into reaction networks. Whereas BNGL provides support for user-specified equivalent sites in proteins, KaDE may infer these equivalence automatically [5,4,12,6,3]. Thus, we can obtain, for each model, two systems of ODEs based on the same set of variables. Yet, we have noticed some differences in the trajectories of the models involving trimers. These differences were due to a wrong interpretation of the BNGL semantics of tests on equivalent sites. Let us explain this on an example. We consider a protein `P` with three sites that may be phosphorylated or not. We consider that each site may be phosphorylated at rate kp whenever no other site are phosphorylated in the protein instance and at rate kpp whenever at least one other site is phosphorylated already.

With equivalent sites, we would model this by the following rules:

```
P(l~u,l~u,l~u) -> P(l~p,l~u,l~u) @ kp
P(l~u,l~p) -> P(l~p,l~p) @ kpp
```

With identified sites, we would consider three site names `l1`, `l2`, `l3`. In our first attempt, we model the phosphorylation of the site `l1` the following way:

```
P(l1~u,l2~u,l3~u) -> P(l1~p,l2~u,l3~u) @ kp
P(l1~u,l2~p) -> P(l1~p,l2~p) @ kpp
P(l1~u,l3~p) -> P(l1~p,l3~p) @ kpp
```

The issue is with the following configuration `P(l1~u,l2~p,l3~p)` of the protein. At least one other site, besides the site `l1`, is phosphorylated, thus the site `l1` should be phosphorylatable at rate `kpp`. Yet, with identified sites the second and the third rule may apply, which makes an overall rate of `2kpp`. In order to avoid overestimating the activity of this interaction, we have no other choices than enumerating all the configurations of the protein, which gives the following rules:

```
P(l1~u,l2~u,l3~u) -> P(l1~p,l2~u,l3~u) @ kp
P(l1~u,l2~p,l3~u) -> P(l1~p,l2~p,l3~u) @ kpp
P(l1~u,l2~u,l3~p) -> P(l1~p,l2~u,l3~p) @ kpp
P(l1~u,l2~p,l3~p) -> P(l1~p,l2~p,l3~p) @ kpp
```

Similar groups of rules for the phosphorylation of the sites `l2` and `l3` are required.

References

[1] Blinov, M., J. R. Faeder, B. Goldstein and W. S. Hlavacek, *Bionetgen: software for rule-based modeling of signal transduction based on the interactions of molecular domains.*, Bioinformatics (Oxford, England) **20** (2004), pp. 3289–3291.

[2] *Bionetwiki home - download.*
URL http://bionetgen.org/index.php/Download

[3] Camporesi, F., "Exact and formal reduction of models of signaling pathways written in Kappa," Ph.D. thesis, École normale supérieure, Paris, France (2017).

[4] Camporesi, F. and J. Feret, *Formal reduction for rule-based models*, ENTCS **276** (2011), pp. 29 – 59, proc. MFPS XXVII.

[5] Camporesi, F., J. Feret, H. Koeppl and T. Petrov, *Combining model reductions*, ENTCS **265** (2010), pp. 73 – 96, proc. MFPS XXVI.

[6] Camporesi, F., J. Feret and K. Q. Lý, *Kade: a tool to compile kappa rules into (reduced) odes models*, in: J. Feret and H. Koeppl, editors, *Proc. CMSB'17*, LNCS (2017), to appear.

[7] Danos, V., J. Feret, W. Fontana and J. Krivine, *Abstract interpretation of cellular signalling networks*, in: F. Logozzo, D. A. Peled and L. D. Zuck, editors, *Proc. VMCAI'08*, LNCS **4905** (2008), pp. 83–97.

[8] Danos, V. and C. Laneve, *Core formal molecular biology*, in: P. Degano, editor, *Proc ESOP'03*, LNCS **2618** (2003), pp. 302–318.

[9] Danos, V. and C. Laneve, *Formal molecular biology*, TCS **325** (2004), pp. 69–110.

[10] Faeder, J. R., M. L. Blinov and W. S. Hlavacek, *Rule-based modeling of biochemical systems with bionetgen*, Methods Mol Biol **500** (2009), pp. 113–67.

[11] Faeder, J. R., W. S. Hlavacek, I. Reischl, M. L. Blinov, H. Metzger, A. Redondo, C. Wofsy and B. Goldstein, *Investigation of early events in fcri-mediated signaling using a detailed mathematical model*, The Journal of Immunology **170** (2003), pp. 3769–3781.

[12] Feret, J., *An algebraic approach for inferring and using symmetries in rule-based models*, ENTCS **316** (2015), pp. 45 – 65, proc. SASB'14.

[13] Feret, J. and K. Q. Lý, *Local traces: An over-approximation of the behaviour of the proteins in rule-based models*, in: E. Bartocci, P. Liò and N. Paoletti, editors, *Proc. CMSB'16*, LNCS **9859** (2016), pp. 116–131.

[14] Feret, J. and K. Q. Lý, *Reachability analysis via orthogonal sets of patterns.*, ENTCS (2017), proc. SASB'16.