A Graph-Theoretic Backend Compiler for Concretizing Abstract Chemical Reaction Networks

1st Julien Bienvenu
EPI Lifeware
École polytechnique & Inria Saclay
Palaiseau, France
Julien.Bienvenu@polytechnique.edu

2nd François Fages
EPI Lifeware
Inria Saclay
Palaiseau, France
Francois.Fages@inria.fr

3st Mathieu Hemery
EPI Lifeware
Inria Saclay
Palaiseau, France
Mathieu.Hemery@inria.fr

4st Sylvain Soliman
EPI Lifeware
Inria Saclay
Palaiseau, France
Sylvain.Soliman@inria.fr

***Abstract*—Chemical Reaction Networks (CRN) are a standard formalism used in chemistry and biology to describe, analyze, and now also design, complex molecular interaction networks. The Turing completeness of continuous chemical reaction networks states that any computable real function can be computed by a continuous CRN on a finite set of abstract molecular species. This result was used to design a compilation pipeline which transforms such functions, either defined by mathematical expressions or as solutions of ordinary differential equations, into a finite set of abstract elementary reactions. In this paper, we show that the notion of subgraph epimorphism, previously introduced for detecting CRN model reductions in model repositories, can be used with its implementation using SAT solvers to create a backend compiler for transforming an abstract CRN into a concrete CRN restricted to a given catalog of real enzymatic reactions. Using the Brenda database, we show that our compiler with its concretization backend is able to automatically retrieve concrete CRNs previously designed manually to implement simple logical functions for the diagnosis of different forms of comas.**

***Index Terms*—chemical computation, symbolic computation, graph rewriting, compilation, biochemistry, enzymatic reactions, logical circuits, diagnosis**

Chemical Reaction Networks (CRNs) are a standard formalism used in chemistry and biology to describe complex molecular interaction systems. In the perspective of systems biology, they are a central tool to analyze the high-level functions of the cell in terms of their low-level molecular interactions. In that perspective, the Systems Biology Markup Language (SBML) [18] is a common format to exchange CRN models and build CRN model repositories, such as Biomodels.net [5] which contains thousands of CRN models of a large variety of cell biochemical processes. In the perspective of synthetic biology, they constitute a target programming language to implement in chemistry new functions either *in vitro*, e.g. using DNA polymers [20], or in living cells using plasmids [10] or in DNA-free RNA-free artificial vesicles using enzymes and metabolites [8].

The mathematical theory of CRNs was introduced in the late 70's, on the one hand by Feinberg in [14], by focussing on perfect adaptation properties and multistability analyses [9], and on the other hand, by Érdi and Tóth by characterizing the set of Polynomial Ordinary Differential Equation systems (PODEs) that can be defined by CRNs with mass action law kinetics, using dual-rail encoding for negative variables [11].

More recently, a computational theory of CRNs was investigated by formally relating their Boolean, discrete, stochastic and differential semantics in the framework of abstract interpretation [13], and by studying the computational power of CRNs under those different interpretations [6], [7], [12].

In particular, under the continuous semantics of CRNs interpreted by ODEs, the Turing-completeness result established in [12] states that any computable real function (i.e. computable by a Turing machine with an arbitrary precision given as input) can be computed by a continuous CRN on a finite set of molecular species, using elementary reactions with at most two reactants and mass action law kinetics. This result uses the following notion of analog computation of a non-negative real function computed by a CRN, where the result is given by the concentration of one species, y_1, and the error is controlled by the concentration of one second species, y_2:

A function $f : \mathbb{R}_+ \to \mathbb{R}_+$ is CRN-computable if there exist CRN over some molecular species $\{y_1, ..., y_n\}$, and a polynomial $q \in {\mathbb{R}_+}^n[\mathbb{R}_+]$ defining their initial concentration values, such that for all $x \in \mathbb{R}_+$ there exists some (necessarily unique) function $y : \mathbb{R} \to \mathbb{R}^n$ such that $y(0) = q(x),\ y'(t) = p(y(t))$ and for all $t > 1$

$$|y_1(t) - f(x)| \leq y_2(t),$$

$y_2(t) \geq 0$, $y_2(t)$ is decreasing and $\lim_{t\to\infty} y_2(t) = 0$.

That definition states that a function f is *computed* by a CRN if for any input $x \geq 0$, and initialization of the CRN input species to value $y(0) = q(x)$, the CRN output species converges to the result $y_1(\inf) = f(x)$. The proof of Turing completeness given in [12] for that notion of analog chemical computation with CRNs uses a previous result of Turing completeness for functions defined by polynomial ordinary differential equation initial value problems (PIVP) [1], the dual-rail encoding of real variables by the difference of concentration between two molecular species [15], [19], and a quadratization transformation to restrict to elementary reactions with at most two reactants [2], [4], [16]. This constructive proof immediately gave rise to a compilation pipeline, implemented in BIOCHAM-4[1], to compile any computable real function presented by a PIVP, or directly any elementary function [17], into a finite abstract CRN, following several

[1] http://lifeware.inria.fr/biocham/.

symbolic transformation steps summarized in Fig. 1. A similar approach is undertaken in the CRN++ system [21] and also [3].

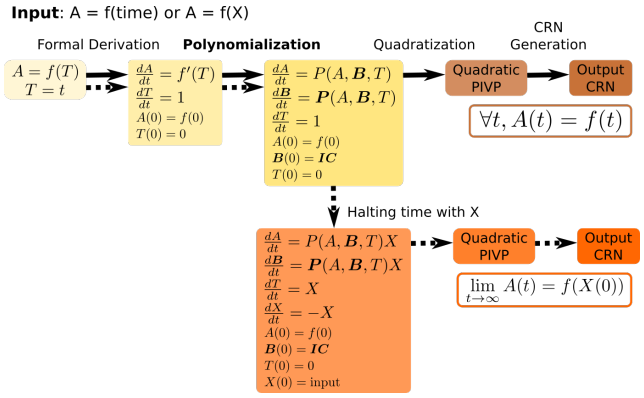

Fig. 1. Symbolic computation steps to compile a formally differentiable function f (termination is proved for elementary functions) into a CRN over a finite set of formal molecular species. The function can be either a function of time (plain arrows) or an input/output function (dashed arrows): P and $\boldsymbol{P}$ are polynomials, and $\boldsymbol{B}$ denotes the set of species introduced by polynomialization given with initial conditions $\boldsymbol{IC}$.

In this communication, we study the CRN concretization problem for mapping an abstract CRN generated by such a compiler to a concrete CRN restricted to the reactions belonging to a given catalog only. We present a graph rewriting algorithm with kinetics constraints to implement a concrete CRN backend compilation step in our compilation pipeline. Using the Brenda database, we show that our algorithm is able to retrieve, among some other solutions, a concrete CRN previously designed, created in articial vesicles, and tested in urine or blood patients for the diagnosis of different forms of comas [8].

Acknowledgment

We are grateful to Patrick Amar and Martin Davy for their help with Brenda database, and with Franck Molina for years of fruitful discussions and collaborative work.

References

[1] Olivier Bournez, Manuel L. Campagnolo, Daniel S. Graça, and Emmanuel Hainry. Polynomial differential equations compute all real computable functions on computable compact intervals. *Journal of Complexity*, 23(3):317–335, 2007.

[2] Andrey Bychkov and Gleb Pogudin. Optimal monomial quadratization for ode systems. In *Proceedings of the IWOCA 2021 - 32nd International Workshop on Combinatorial Algorithms*, July 2021.

[3] Luca Cardelli, Mirco Tribastone, and Max Tschaikowski. From electric circuits to chemical networks. *Natural Computing*, 19, 2020.

[4] David C. Carothers, G. Edgar Parker, James S. Sochacki, and Paul G. Warne. Some properties of solutions to polynomial systems of differential equations. *Electronic Journal of Differential Equations*, 2005(40):1–17, 2005.

[5] Vijayalakshmi Chelliah, Camille Laibe, and Nicolas Novère. Biomodels database: A repository of mathematical models of biological processes. In Maria Victoria Schneider, editor, *In Silico Systems Biology*, volume 1021 of *Methods in Molecular Biology*, pages 189–199. Humana Press, 2013.

[6] Ho-Lin Chen, David Doty, and David Soloveichik. Deterministic function computation with chemical reaction networks. *Natural computing*, 7433:25–42, 2012.

[7] Matthew Cook, David Soloveichik, Erik Winfree, and Jehoshua Bruck. Programmability of chemical reaction networks. In Anne Condon, David Harel, Joost N. Kok, Arto Salomaa, and Erik Winfree, editors, *Algorithmic Bioprocesses*, pages 543–584. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[8] Alexis Courbet, Patrick Amar, François Fages, Eric Renard, and Franck Molina. Computer-aided biochemical programming of synthetic microreactors as diagnostic devices. *Molecular Systems Biology*, 14(4), 2018.

[9] Gheorghe Craciun and Martin Feinberg. Multiple equilibria in complex chemical reaction networks: II. the species-reaction graph. *SIAM Journal on Applied Mathematics*, 66(4):1321–1338, 2006.

[10] X. Duportet, L. Wroblewska, P. Guye, Y. Li, J. Eyquem, J. Rieders, G. Batt, and R. Weiss. A platform for rapid prototyping of synthetic gene networks in mammalian cells. *Nucleic Acids Research*, 42(21), 2014.

[11] Péter Érdi and János Tóth. *Mathematical Models of Chemical Reactions: Theory and Applications of Deterministic and Stochastic Models*. Nonlinear science : theory and applications. Manchester University Press, 1989.

[12] François Fages, Guillaume Le Guludec, Olivier Bournez, and Amaury Pouly. Strong Turing Completeness of Continuous Chemical Reaction Networks and Compilation of Mixed Analog-Digital Programs. In *CMSB'17: Proceedings of the fiveteen international conference on Computational Methods in Systems Biology*, volume 10545 of *Lecture Notes in Computer Science*, pages 108–127. Springer-Verlag, September 2017.

[13] François Fages and Sylvain Soliman. Abstract interpretation and types for systems biology. *Theoretical Computer Science*, 403(1):52–70, 2008.

[14] Martin Feinberg. Mathematical aspects of mass action kinetics. In L. Lapidus and N. R. Amundson, editors, *Chemical Reactor Theory: A Review*, chapter 1, pages 1–78. Prentice-Hall, 1977.

[15] V. Hárs and J. Tóth. On the inverse problem of reaction kinetics. In M. Farkas, editor, *Colloquia Mathematica Societatis János Bolyai*, volume 30 of *Qualitative Theory of Differential Equations*, pages 363–379, 1979.

[16] Mathieu Hemery, François Fages, and Sylvain Soliman. On the complexity of quadratization for polynomial differential equations. In *CMSB'20: Proceedings of the eighteenth international conference on Computational Methods in Systems Biology*, Lecture Notes in BioInformatics. Springer-Verlag, September 2020.

[17] Mathieu Hemery, François Fages, and Sylvain Soliman. Compiling elementary mathematical functions into finite chemical reaction networks via a polynomialization algorithm for ODEs. In *CMSB'21: Proceedings of the nineteenth international conference on Computational Methods in Systems Biology*, Lecture Notes in BioInformatics. Springer-Verlag, September 2021.

[18] Michael Hucka et al. The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003.

[19] K. Oishi and E. Klavins. Biomolecular implementation of linear i/o systems. *IET SYstems Biology*, 5(4):252–260, 2011.

[20] Lulu Qian, David Soloveichik, and Erik Winfree. Efficient turing-universal computation with DNA polymers. In *Proc. DNA Computing and Molecular Programming*, volume 6518 of *LNCS*, pages 123–140. Springer-Verlag, 2011.

[21] Marko Vasic, David Soloveichik, and Sarfraz Khurshid. CRN++: Molecular programming language. In *Proc. DNA Computing and Molecular Programming*, volume 11145 of *LNCS*, pages 1–18. Springer-Verlag, 2018.