



On Design and Analysis of Chemical Reaction Network Algorithms

Anne Condon^(✉)

The Department of Computer Science,
University of British Columbia, Vancouver, Canada
condon@cs.ubc.ca

The fields of DNA computing, molecular programming and DNA nanotechnology offer exciting new possibilities for organizing and manipulating matter at the nanoscale, and prompt us to think about computation in creative new ways. Molecules reacting in a test tube change state, and counts of molecules can in principle be used to simulate counter machines, all in a highly distributed, asynchronous and stochastic manner. In this talk I'll give some background on models of molecular programming, focusing on Stochastic Chemical Reaction Networks, and describe some beautiful results and open problems pertaining to this model of computing.

Stochastic Chemical Reaction Networks (CRNs) have traditionally been used to model the dynamics of interacting molecules in a well-mixed solution [1], particularly when the counts of some molecular species is low, in which case mass action kinetics is not a good model. More recently, stochastic CRNs have become a popular model for describing molecular programs - programs that can be executed in a test tube or other wet environment [2,3]. Reactions are the basic instructions of these programs, acting on molecules in a well-mixed solution. CRNs are closely related to population protocols among resource-limited agents in distributed networks [4], as well as models of gene regulatory networks, infectious diseases and voting processes [5-8].

Stable Function Computation by CRNs. In one model of predicate computation by CRNs proposed by Angluin et al. [9,10], inputs are represented by initial counts of certain molecular species in a well-mixed solution of fixed volume. Chen et al. [11] studied function computation in essentially the same model. For example, if the solution contains n_1 copies of species X_1 and n_2 copies of species X_2 , then the two reactions of Fig. 1(a) eventually produce a number of Y 's equal to $n_1 + n_2$, while the single reaction of Fig. 1(b) produces a number of Y 's equal to the min of the counts of X_1 and X_2 , and the reactions of Fig. 1(c) produce the max of the counts. (Assume that the rate constant associated with each reaction is 1, although these assertions are true regardless of the rate constant.)

The evolution of a "computation" by a CRN can be described as a sequence of configurations, where each configuration is a vector of counts of molecular species and the initial configuration describes initial species counts. There is an underlying probabilistic model, consistent with the principles of chemical reactions (under fixed environmental conditions such as temperature), that determines the rates and relative likelihoods of reactions as a function of molecular

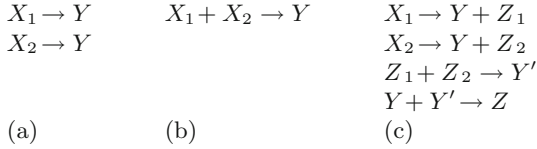


Fig. 1. Simple CRNs for stably computing the (a) sum, (b) min and (c) max of the counts of two molecular species X_1 and X_2 . The output is represented by the number of copies of species Y . The CRN of part (c) integrates the CRN of part (a) with a slight variant of part (b), thereby computing the max as the sum minus the min.

species counts, reaction rate constants, and the volume of the enclosing solution. From this probabilistic model, notions of correctness (perhaps allowing for a small probability of error) as well as efficiency (depending on reaction rates, which in turn depend on rate constants and counts of molecular reactants) can be formulated.

All of the CRNs of Fig. 1 compute their functions *stably* [9–11]: on any computation, a configuration is eventually reached with probability 1 in which the counts of output species are consistent with the function being computed, and once reached, the counts never change subsequently. The class of functions that can be stably computed by CRNs is exactly the semi-linear functions [10, 11].

Stable Function Composition. A basic question is: given two CRNs that stably compute two functions f and g , when is it possible to compose the CRNs in order to compute the composition $g \circ f$? Stable composition is certainly possible when the CRN that computes f is *output-oblivious* - that is, no output species is a reactant in any reaction of the CRN. Not all semi-linear functions are output-oblivious. We will show that the max function and generalizations are not output-oblivious, and present a characterization of which semilinear functions can be stably computed by output-oblivious CRNs.

CRNs with Error: Approximate Majority. The Approximate Majority problem is as follows: in a mixture of two types of species where the gap between the counts of the majority and minority species is above some threshold, which species is in the majority? CRNs that solve Approximate Majority with low error probability have been well studied but have been difficult to analyze [12]. We'll describe a simple way to analyze CRNs for Approximate Majority [13], as well as several variants, e.g., when reaction rates are uncertain or when some molecules are Byzantine. These CRNs are described in Fig. 2 which is from Condon et al. [13]. Key to our approach is to first analyze a very simple CRN for Approximate Majority involving tri-molecular reactions, i.e., reactions with three reactants. We can show that well-studied bi-molecular CRNs for Approximate Majority essentially emulate the tri-molecular protocol, and also that the same analysis principles can be used to prove correctness and efficiency of multi-valued consensus.

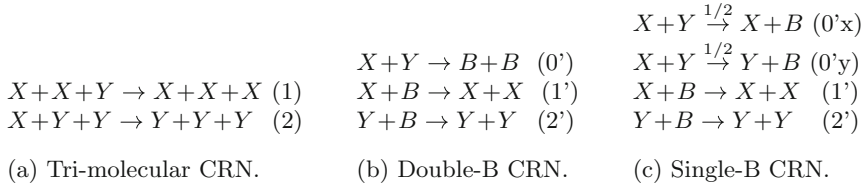


Fig. 2. A tri-molecular and two bi-molecular chemical reaction networks (CRNs) for Approximate Majority. Reactions (0'x) and (1'y) of Single-B have rate constant $1/2$ while all other reactions have rate constant 1.

References

1. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. *J. Phys. Chem.* **81**, 2340–2361 (1977)
2. Cook, M., Soloveichik, D., Winfree, E., Bruck, J.: Programmability of chemical reaction networks. In: Condon, A., Harel, D., Kok, J., Salomaa, A., Winfree, E. (eds.) *Algorithmic Bioprocesses*, pp. 543–584. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-540-88869-7_27
3. Soloveichik, D., Cook, M., Winfree, E., Bruck, J.: Computation with finite stochastic chemical reaction networks. *Nat. Comput.* **7**(4), 615–633 (2008)
4. Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M.J., Peralta, R.: Computation in networks of passively mobile finite-state sensors. *Distrib. Comput.* **18**(4), 235–253 (2006)
5. Bower, J.M., Bolouri, H.: *Computational Modeling of Genetic and Biochemical Networks*. MIT Press, Cambridge (2004)
6. Cruise, J., Ganesh, A.: Probabilistic consensus via polling and majority rules. *Queueing Syst.* **78**(2), 99–120 (2014)
7. Perron, E., Vasudevan, D., Vojnovic, M.: Using three states for binary consensus on complete graphs. In *Proceedings of the 28th IEEE Conference on Computer Communications (INFOCOM)*, pp. 2527–2535. (2009)
8. Moussaïd, M., Kämmer, J.E., Analytis, P.P., Neth, H.: Social influence and the collective dynamics of opinion formation. *PLoS ONE* **8**(11), e78433 (2013)
9. Angluin, D., Aspnes, J., Diamadi, Z., Fischer, M., Peralta, R.: Computation in networks of passively mobile finite-state sensors. *Distrib. Comput.* **18**(4), 235–253 (2006)
10. Angluin, D., Aspnes, J., Eisenstat, D.: Stably computable predicates are semi-linear. In: *Proceedings of the Twenty-Fifth Annual ACM Symposium on Principles of Distributed Computing*, pp. 292–299. ACM (2006)
11. Chen, H., Doty, D., Soloveichik, D.: Deterministic function computation with chemical reaction networks. *Nat. Comput.* **13**(4), 517–534 (2014)
12. Angluin, D., Aspnes, J., Eisenstat, D.: A simple population protocol for fast robust approximate majority. *Distrib. Comput.* **21**(2), 87–102 (2008)
13. Condon, A., Hajiaghayi, M., Kirkpatrick, D., Mañuch, J.: Simplifying analyses of chemical reaction networks for approximate majority. In: Brijder, R., Qian, L. (eds.) *DNA 2017. LNCS*, vol. 10467, pp. 188–209. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66799-7_13