

# The limits of chemical computing

David Doty\*

## Abstract

The model of chemical reaction networks is one of the oldest in science, dating to the work of Guldberg and Waage on the Law of Mass Action in the 1860s. It is well-known, widely-used, and reliable as a model for describing natural well-mixed chemical systems. With recent and ongoing experimental breakthroughs in the synthesis of chemicals that undergo designed reactions—for instance implementing chemical oscillators, Boolean logic, and solving analog computing problems such as majority computation—the model has received scrutiny as a *model of computation*, *i.e.*, a programming language.

One useful interpretation of this model comes from the field of distributed computing, which studies computation among a population of computational “agents”, who must communicate in order to solve some problem, for instance determining which of two opinions distributed among the agents is in the majority across the whole population. In this interpretation, chemical reaction networks model agents (molecules) who *cannot* control which other agents they communicate (collide) with, but who *can* control their response (a reaction). The response changes the state of memory (chemical species) of some agents, possibly destroying or creating agents.

This talk will outline some major questions and what we know about the answers. Most of the questions take the general form: “*what sorts of computational problems can be solved by a chemical reaction network obeying some constraints?*” I will emphasize the importance of modeling choices in formalizing this question, how these choices affect the results, and the assumptions they require from the underlying reality they are attempting to model. For instance,

- What sort of computational problem is being solved? A decision problem with a yes/no answer? A function with more than two possible outputs? How is output encoded?
- Do we require the network to be guaranteed to get the correct result, or do we tolerate a small probability of error?
- Do we employ continuous semantics (the mass-action model with real-valued concentrations) or discrete semantics (the stochastic model with integer-valued molecular counts)?
- What sort of constraints do we place on the structure of the reactions? Do we allow reactions that violate conservation of mass/energy such as  $X \rightarrow 2X$ ?
- What sort of initial conditions do we assume can be prepared? Can it be assumed that a single copy of a certain species is initially present?
- How do we define when the computation is completed? Can the network “know” it has terminated (*i.e.*, signal that it is done)? Does it converge on the correct answer without knowing when convergence occurs? How can we compose it with a downstream computation if the upstream computation cannot signal its termination?
- How many species and reactions do we assume can be synthesized? How does the computational ability improve if more species and reactions are allowed?

---

\*University of California, Davis, CA, USA