

Proposal for the Development of a Software Package for Simulating and Studying Catalytic Reaction Systems and Autocatalytic Sets

Wim Hordijk, Mike Steel and Jotun Hein

February 2011

Introduction

In previous work we have introduced and studied a model of catalytic reaction systems and autocatalytic sets [1,2,3]. A *catalytic reaction system* (CRS) consists of a set of molecule types, a set of possible chemical reactions between these molecule types, and a set of catalyzations indicating which molecule types (if any) catalyze which reactions. An *autocatalytic set* is a subset of molecule types and reactions that is “catalytically closed and self-supportive”: all reactions in the subset have to be catalyzed by at least one molecule type from the same subset, and each molecule type has to be generated from a *food set* (a small set of molecule types that are assumed freely available) by repeated applications of reactions from the same subset. In the original terminology, this is called a *reflexively autocatalytic and food-generated* (RAF) set [1,2]. Furthermore, we introduced a polynomial-time algorithm for determining whether an (arbitrary) CRS has an RAF set embedded in it [2].

Applying our RAF algorithm to a specific CRS model consisting of binary polymers up to a given length n , with ligation and cleavage reactions and randomly assigned catalysts, where each molecule type has a given probability p of catalyzing a reaction [4,5], we showed that only a linear growth rate (with n) in the level of catalysis (i.e., the average number of reactions catalyzed by a molecule) is sufficient for autocatalytic sets to emerge with high probability [2]. This was subsequently confirmed theoretically [6], thus refuting earlier criticism and arguments against the plausibility of autocatalytic sets (see e.g. [7]). Recently, we have investigated these results in more detail, simplified our RAF definition and algorithm slightly, and looked at several extensions and additional questions that can be addressed within our framework [8]. This work is also highly relevant in the larger context of the origin of life [3].

Proposed software package

The experimental results obtained in this project so far were all generated with a rather ad-hoc implementation of our RAF algorithm, one that was developed specifically for applying it to the mentioned binary polymer model. Unfortunately, this leaves little room for generalizations and extensions. Furthermore, given the sizes of the instances of the random CRS model we have used (up to several millions of reactions), we are running into limits in terms of what is computationally feasible. However, there has been a recent increase in the interest in, and questions concerning autocatalytic sets [3]. Therefore, we propose to develop a general software package that will allow us to continue our investigations in this area.

In particular, we propose to develop a software package that consists of several independent but related and interconnected modules:

1. A model-independent and efficient implementation of the RAF algorithm.
2. A generator of CRS instances, depending on a given model and its corresponding parameter values, to be chosen by the user.
3. A visualization tool for displaying and manipulating CRS and RAF sets.
4. A “dynamical systems engine” to simulate the flow of molecules on a CRS or RAF set.
5. A graphical user interface (GUI) that connects all modules together.

1. RAF algorithm

As mentioned above, our current implementation of the RAF algorithm is specifically for applying it to the binary polymer CRS model. However, the algorithm itself is independent of any particular CRS model, and can be applied to any CRS (whether a model or a real one). Thus, it will be very useful to have a model-independent implementation of the RAF algorithm. Furthermore, in our simulations we run into the limits of what is computationally feasible. Our RAF algorithm is *polynomial* in the size of the reaction set of a given CRS. However, in the binary polymer model, for example, the size of the reaction set is *exponential* in n (the size of the largest molecules in the system). Even using a large computer cluster, we have been limited to about $n=20$ in our simulations. So, it would also be useful to try and make our implementation of the RAF algorithm more efficient, by using more clever data structures and “short-cuts”. This module should also calculate statistics such as the number and sizes of (independent) RAF sets found in a given CRS.

2. CRS instance generator

To be able to study various CRS models, it will be useful to have a module for generating instances of a CRS given a particular model and its corresponding parameter values, which will be chosen by the user. Next to the binary polymer model, we can include various types of models such as other (existing) artificial chemistry models, RNA-based models, or extensions such as the template-based catalysis case we studied recently [8]. The main idea is that CRS instances generated by this module can then be fed into the RAF algorithm module to search for RAF sets, or into the visualization module to display them.

3. Visualization tool

CRS and RAF sets can be represented as graphs, and can thus be visualized as such. However, since the CRS instances we are usually working with are rather large (millions of nodes), a sophisticated visualization tool is required. First, we need to use, or perhaps even develop, clever graph lay-out algorithms. Next, we need to be able to zoom in on sub-parts of a CRS or RAF graph, and be able to select subsets of nodes and edges (for example, all molecules smaller or larger than a certain length, or all molecules involved in more than one reaction, or all reactions that are (not) catalyzed, etc.). Furthermore, the user should be able to manipulate the reaction graph, not only in terms of changing the lay-out, but also in terms of actually changing the graph structure, for example by adding or removing nodes and edges, or changing the end points of existing edges, etc. And finally, the visualization tool should be able to show the flow of molecule concentrations on the reaction graph, as described next.

4. Dynamical systems engine

So far, our investigations have been mostly into the structure of the underlying reaction graphs, i.e., whether there are RAF sets embedded in a general CRS, and what levels of catalysis are required to get such RAF sets with high probability. However, one interesting and important direction to take this work into next, is to look at actual molecular dynamics on such reaction graphs. Given an initial condition of molecule concentrations on some or all of the nodes in the reaction graph, how do these concentrations change over time given the possible reactions, presence or absence of catalysts, and a realistic method for simulating such dynamics.

In earlier work [9], we used the Gillespie algorithm [10] to simulate the flow of molecules in a simple model of a CRS (similar to the binary polymer model mentioned above). This algorithm provides a standard and widely used method to perform such molecular dynamics simulations. We wish to re-implement this method as part of the currently proposed software package, thus providing a “dynamical systems engine” to simulate the molecular dynamics on general CRS or RAF sets. This will allow us to take our work even further, and answer many new and interesting questions related to the emergence, stability, and probability of autocatalytic sets.

5. Graphical user interface

Finally, as with any user-friendly software package, we will need a graphical user interface (GUI) to connect all the previously mentioned modules (and possible future ones) together. This GUI should make it easy for the user to select different CRS models and corresponding parameter values, to load, save and manipulate CRS and RAF instances, to apply the RAF algorithm, to provide initial conditions of molecule concentrations on a reaction graph and study the molecular dynamics, and to collect and generate statistics and plots to be used in presentations and (scientific) articles for publication. However, the software package should also have an option to run in batch mode, to collect statistics over many model instances.

Technical issues

The first issue that needs to be worked out is to choose a standard file format for storing CRS instances and exchanging them between the different modules. Several standard formats are available for this, but we will need to find out which one is the most useful and efficient for our purposes, while also trying to maintain compatibility with other, already existing software tools.

The second issue is deciding on a programming language for the implementation. We propose to use Java, as it is a flexible and powerful language, it includes an extensive library for creating graphical tools and interfaces, and it is platform independent.

A final issue is availability. We intend to make our software package completely open source, and release it under the GNU General Public License (GPL). This way it will be freely available for other researchers to use, and to change and add additional models, algorithms, or other modules to it. For this reason, it is important to have a platform independent implementation in a well-know programming language (such as Java).

References

1. M. Steel. The emergence of a self-catalysing structure in abstract origin-of-life models. *Applied Mathematics Letters* 3, 91-95, 2000.
2. W. Hordijk and M. Steel. Detecting autocatalytic, self-sustaining sets in chemical reaction systems. *Journal of Theoretical Biology* 227, 451-461, 2004.
3. W. Hordijk, J. Hein and M. Steel. Autocatalytic sets and the origin of life. *Entropy* 12(7), 1733-1742, 2010.
4. S. A. Kauffman. Autocatalytic sets of proteins. *Journal of Theoretical Biology* 119, 1-24, 1986.
5. S. A. Kauffman. *The Origins of Order*. Oxford University Press, 1993.
6. E. Mossel and M. Steel. Random biochemical networks: The probability of self-sustaining autocatalysis. *Journal of Theoretical Biology* 233, 327-336, 2005.
7. S. Lifson. On the crucial stages in the origin of animate matter. *Journal of Molecular Evolution* 44, 1-8, 1997.
8. W. Hordijk, S. A. Kauffman and M. Steel. Required levels of catalysis for emergence of autocatalytic sets in chemical reaction systems. *International Journal of Molecular Science*. Submitted, 2011.
9. W. Hordijk and J. F. Fontanari. Catalytic reaction sets, decay, and the preservation of information. In *IEEE International Conference on Integration of Knowledge Intensive Multi-Agent Systems*, 133-138, 2003.
10. D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *Journal of Computational Physics* 22, 403-434, 1976.