Paolo Dini, Chrystopher L Nehaniv, Attila Egri-Nagy and Maria J Schilstra

Algebraic analysis of the computation in the Belousov-Zhabotinksy reaction

Book section

Algebraic Analysis of the Computation in the Belousov-Zhabotinksy Reaction

Paolo Dini[1,2], Chrystopher L Nehaniv[1], Attila Egri-Nagy[1,3] and Maria J Schilstra[1]

[1] Royal Society Wolfson BioComputation Research Lab
Centre for Computer Science and Informatics Research
University of Hertfordshire, United Kingdom
[p.dini, c.l.nehaniv, m.j.1.schilstra]@herts.ac.uk

[2] Department of Media and Communications
London School of Economics and Political Science
London, United Kingdom

[3] School of Computing and Mathematics
University of Western Sydney, Parramatta, New South Wales, Australia
a.egri-nagy@uws.edu.au

Abstract. We analyse two very simple Petri nets inspired by the Oregonator model of the Belousov-Zhabotinsky reaction using our stochastic Petri net simulator. We then perform the Krohn-Rhodes holonomy decomposition of the automata derived from the Petri nets. The simplest case shows that the automaton can be expressed as a cascade of permutation-reset cyclic groups, with only 2 out of the 12 levels having only trivial permutations. The second case leads to a 35-level decomposition with 5 different simple non-abelian groups (SNAGs), the largest of which is A_9. Although the precise computational significance of these algebraic structures is not clear, the results suggest a correspondence between simple oscillations and cyclic groups, and the presence of SNAGs indicates that even extremely simple chemical systems may contain functionally complete algebras.

Introduction

In self-organising systems, the "self" or autonomous aspect is provided by the fall towards equilibrium, which serves as the driver or energy source. As a consequence, a system that needs to maintain self-organising behaviour indefinitely must be open since, if it were closed, once it had reached equilibrium it would stop functioning. Therefore, in order to keep going it must be open and connected to a source of (free) energy that can keep it 'far from equilibrium', to use Prigogine's famous phrase [19], even whilst it is continually falling towards it. The Belousov-Zhabotinsky (BZ) reaction has been studied extensively [20] because it was the first reaction to exhibit sustained oscillations even in an isolated system, although they do die down eventually. Before Belousov's discovery in the 1930s and Zhabotinsky's confirmation of the phenomenon in the 1960s,

Dini, P, Nehaniv, CL, Egri-Nagy, A, Schilstra, MJ (2012). Algebraic Analysis of the Computation in the Belousov-Zhabotinksy Reaction, in Lones, M, Smith, SL, Teichmann, S, Naef, F, Walker, JA and Trefzer, MA (Eds) IPCAT2012: 9th International Conference on Information Processing in Cells and Tissues, Springer LNCS 7223, Cambridge, UK, 30 March-2 April, pp 216-224.

species concentrations were believed to vary monotonically unless driven by a periodic forcing function. In a constant-flow reactor the oscillations are periodic and can be sustained indefinitely, as long as the inflow and outflow are kept constant. This qualifies the BZ reaction as a system far from equilibrium. In this paper we analyse a simplified ordinary differential equation (ODE) model of the BZ reaction, the almost equally famous "Oregonator" model, developed by Field and Noyes at the University of Oregon [11].

We compare the structure and behaviour of a very simple system inspired by the Oregonator model of the BZ reaction from the different viewpoints of systems biology and algebraic automata theory. In particular, we focus on its oscillatory behaviour. Although the computational properties of chemical oscillations are not clear, the fact that we are very familiar with them at both an intuitive and a mathematical level makes them a useful reference system when attempting to decipher the computational significance of the algebraic structures uncovered in the corresponding finite state automata, as we discuss below. This was, in fact, the main motivation for selecting the BZ reaction as an object of study. Thus, this work aims to merge two research traditions: dynamical systems theory rooted in physics and informing much of modern-day systems biology, and theoretical computer science rooted in algebraic automata theory [16].

Discretisation

To be able to analyse the (Oregonator model of the) BZ reaction from these two different perspectives we must find a way to discretise it. A good way to achieve this is with a Petri net (PN), although Kauffman's Boolean networks [15,7] and Rhodes's reaction digraphs [22] are also useful possibilities, all amenable to algebraic automata-theoretic methods. The PN notation, invented to describe interaction and transformation in discrete distributed systems [21,6], is highly suitable to depict the structure of biochemical reaction networks at the level of interaction between molecules. In combination with kinetic information, PN models are useful tools in the derivation of the coupled ODE systems that describe the dynamic behaviour of these networks [23]. Once such a PN has been obtained, it is straightforward to derive a finite-state automaton by treating each possible marking of the PN as a different state of the automaton. In order to obtain a finite state automaton the number of tokens is bounded, and the bound is called the "capacity" of the place. This can be viewed as discretising concentration levels or bounding the number of molecules of each type. Since the resulting number of states can be very large, normally the markings-to-states mapping is done for a specific choice of initial conditions, which yields a subset of the global automaton of all possible states.

Krohn-Rhodes Theory

The Krohn-Rhodes prime decomposition theorem for finite automata [16] has been discussed, explained, and applied in a large number of books and articles since the theorem was published in 1965 ([1] and many others). In 1967 Zeiger [26] proved a variant, called holonomy decomposition (HD), according to which

any finite automaton can be decomposed into a cascade of permutation-reset automata arising from a study of how inputs act on certain subsets of the powerset of the state set. In other words, the state transitions of the component automata can only be either permutations (of certain subsets of subsets) of the state set or resets (Cases 1a and 1b in Figure 1, respectively). The permutation-reset automata can then be further decomposed into (finite and discrete) groups and two-state resets (also known as flip-flops). Finally, using the Jordan-Hölder theorem each group can be further subdivided into a sequence of simple groups, known as its composition factors, recovering the Krohn-Rhodes decomposition into irreducible atomic groups (simple groups) and combinatorial semigroups (cascades of banks of flip-flops). The HD has continuously been improved in efficiency over the years (e.g., [10,13,3,18,4], finally leading to computer algebraic realisation ([9], which has more recently been reimplemented in `GAP` [25] as `SgpDec` [8]), making possible the decomposition and analysis of structures previously well beyond human capacity to analyse.

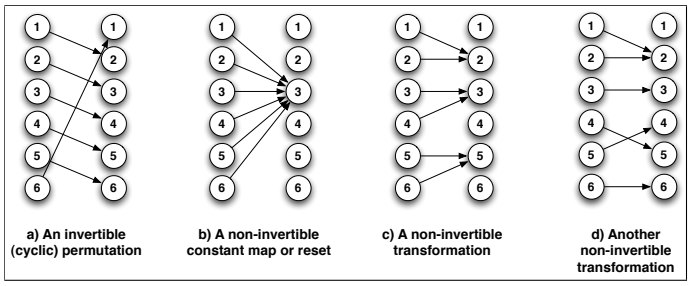

Fig. 1. Different kinds of transformations of 6 states (inspired by [18])

HD diagrams, used to visualise the decomposition, do not show how the groups act explicitly, but indicate the presence of different groups at the various levels of the cascade. These groups indicate the presence of 'local pools of reversibility' [22], but it is still unclear what *algorithmic* significance the transitions these groups induce might have. Further, they show up in the decomposition of automata derived from metabolic and regulatory pathways. Because every group is associated with one or more symmetries and because biological systems exhibit – in fact, depend on – many symmetries in their structure and behaviour, it seems inescapable that the groups embedded in the HD of biological automata have something to do with their self-organising properties. We now look at the BZ system in more detail.

Analysis of BZ Reaction

The BZ reaction has been analysed by many people over the last 50 years. In addition to the Field, Noyes, and Körös works already cited a useful reference is Scott [24], on whom we mostly rely. The size of the HD tends to increase exponentially with automaton size. Thus, although `SgpDec` has made it possible to analyse systems that are immensely greater than what could previously be done only by hand, we still need to keep the systems analysed as small as possible. Therefore, although the Oregonator model is already a significant simplification

relative to the full set of chemical equations of the BZ system, we had to simplify it further in order to bring the corresponding automaton to a size amenable for algebraic analysis. We now provide a brief summary of the original model and corresponding ODE system in order to explain and justify the simplifications effected. The Oregonator chemical equations are the following [24]:

$$A + Y \rightarrow X + P \qquad rate = k_3 AY \tag{1}$$
$$X + Y \rightarrow 2P \qquad rate = k_2 XY \tag{2}$$
$$A + X \rightarrow 2X + 2Z \qquad rate = k_5 AX \tag{3}$$
$$2X \rightarrow A + P \qquad rate = k_4 X^2 \tag{4}$$
$$B + Z \rightarrow (1/2) f Y \qquad rate = k_c BZ \tag{5}$$

X, Y, and Z correspond to the three compounds that undergo periodic oscillations under steady-state boundary conditions, meaning inside the continuous-flow stirred-tank (CFST) reactor. A, B, and P, by contrast, do not vary as a function of time, a consequence of the CFST reactor setup. The three colours Red, Green and Blue shown correspond to the three processes the BZ reaction is conceptually divided into.[4] Table 1 describes the variables in question.

Oregonator Variable	**Chemical Compound**	**Chemical Symbol**	**Association with BZ Reaction Process**
X	Bromous Acid	$HBrO_2$	Process B (reduces X, generates Z)
Y	Bromide Ion	Br^-	Process A (reduces Y, generates X)
Z	Cerum 4	$Ce(IV)$	Process C (reduces Z, generates Y)
A	Bromate	BrO_3^-	All 3 processes
B	Malonic Acid	$CH_2(COOH)_2$	Process C
P	Hypobromous Acid	$HOBr$	Process A and C

Table 1. Summary of Oregonator variables

From Eqs. 1-5 the rate equations are easily derived as a set of three ODEs:

$$\frac{dX}{dt} = k_3 AY - k_2 XY + k_5 AX - 2k_4 X^2 \tag{6}$$
$$\frac{dY}{dt} = -k_3 AY - k_2 XY + 1/2 f k_c BZ \tag{7}$$
$$\frac{dZ}{dt} = 2k_5 AX - k_c BZ, \tag{8}$$

(typo: should be red)

where the colours indicate the contribution to the rates of change of the three variables due to the three processes A, B, and C. Since our objective is to reach an intuitive understanding of the computational significance of the algebraic structure of automata derived from biochemical systems, we now proceed to make a number of radical simplifications to this system, a step which we believe to be necessary at this early stage in the analysis. In fact, since, as we shall see

[4] Using the potentially confusing accepted notation, the names of the processes have nothing to do with the letters A, B, and P assigned to three of the compounds.

below, the HD results can be very difficult to interpret even with very simple systems, it is important to start with the simplest possible system.

In order to see behaviour that is close to a non-equilibrium dynamical system's we needed each place to have a capacity of at least 4 tokens, so we reduced the number of places to the smallest number possible, i.e. 3. The justification for eliminating the three compounds A, B, and P lies in the fact that for a CFST reactor they remain constant. Thus, including them greatly increases the state space of the automaton derived from the PN in spite of their not contributing to the dynamics we are investigating. Furthermore, we eliminated also k_2 and k_4 in Eqs. 6-8. The motivation is that although they are important for reproducing the shape of the BZ oscillations, the oscillations themselves can be generated with a simpler system, which is preferable for now. Similarly, the factor of 2 in front of the first term on the right-hand side of Eq. 8, which comes from the factor of 2 in front of Z in Eq. 3, is also ignored in constructing the PN. This will need to be brought back in at a later stage because it is responsible for the characteristically fast growth of the Z compound (in Process A).

The problem with these modifications is that the resulting system is so different from the original BZ system that it may not even oscillate. This is remedied by artificially introducing inhibition, which is applied cyclically around the three active compounds. Unfortunately once this step is taken it becomes impossible to compare directly the resulting PN to the original ODE system, even if simplified. However, since the average place concentrations obtained with a stochastic PN simulator will converge to the ODE results as the number of runs approaches infinity [12], we can still analyse the resulting system as if we *did* know the governing equations. The result of all these simplifications and modifications is an extremely simple and highly symmetrical PN, shown in Figure 2. A weight n of the inhibition arc between, for example, Place Y and Transition t_1 means that t_1 is inhibited if Y contains n or more tokens.

Figure 2 shows the `GAP` input file prepared for our PN package [5] and the 15-state automaton corresponding to the initial condition $(0, 4, 0)$ (States 5, 8 and 10 are not reachable and are not shown). At the bottom of Figure 2 the output of a stochastic PN simulator coded in *Mathematica* can be seen as a time series of the token values in the three places of this PN. The traces shown are the average of 500 runs. Damped oscillations are clearly visible, as well as the fact that this PN conserves mass ($1.3 \cdot 3 \approx 4$). The states of the automaton shown are the possible markings of the PN from the given initial condition. The rate constants for this example are all 1 ($K = (1, 1, 1)$). Finally, `SgpDec` was used to generate the HD, revealing 12 cascaded permutation-reset levels, whereby '1' indicates either an irreversible component or a trivial group, and the other groups are shown in standard notation. It is not very surprising that in this extremely simple and symmetrical example most of the levels of the decomposition are groups, and they are particularly simple (in the colloquial sense of the term) groups, since they are all cyclic groups.

The abstract concept of algebraic structure is useful for understanding mathematical theorems, but by itself it is not readily applicable. Fortunately, Krohn-

Rhodes theory is closely related to a cognitive tool with which we are intimately familiar in our daily lives: coordinatisation. Namely, the different levels of the Krohn-Rhodes decomposition of an automaton are analogous to the different positions in our positional number systems. The decomposition then becomes an expansion of a given automaton into an "abstract number system" that is defined by the automaton itself: each state is expressed as a different multi-digit "number", where each "digit" corresponds to a level in the decomposition. The significance of this insight, due to Rhodes, is that the coordinatisation perspective gives us at once powerful cognitive and calculational tools for manipulating an automaton in our mind or with possible software support, and also gives us the starting point of a general computer science methodology.

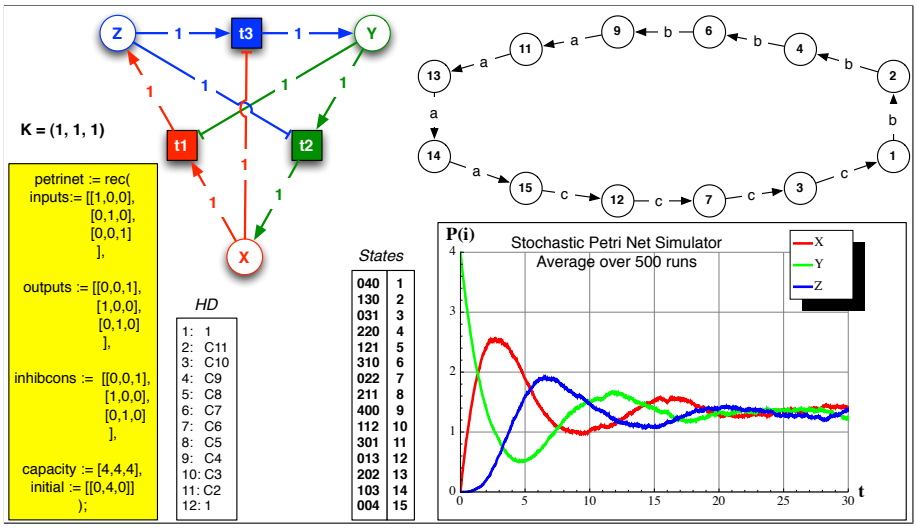

Fig. 2. Stochastic simulation and HD analysis of very simple BZ-like system

Figure 3 shows a rather different story. Here the PN is not symmetrical since one of the inhibitions is missing, and the other two have different values. The stochastic simulation shows strongly damped oscillations, and the automaton (with the same states) is more complex. The most interesting output, however, is by far the HD. The decomposition shows 35 levels, with a fair number of groups. This case is remarkable because of the presence of very large groups (S_9 has $9! = 362,880$ elements acting on subsets of the states shown, at level 14). The groups shown in red all contain SNAGs, i.e. A_9, A_8, A_7, A_6, A_5.

Discussion and Conclusion

The SNAGs are exactly the functionally complete groups [17,14] and are also considered to be related to error-correction [22]. The former property of SNAGs makes them a natural candidate for realizing an analogue of "universal computation" within the finite realm [22,17,14,2]. Therefore, we can conclude that the appearance of SNAGs indicates that even such a simple system is capable of functionally complete computation (i.e. like Boolean algebra). In particular, this

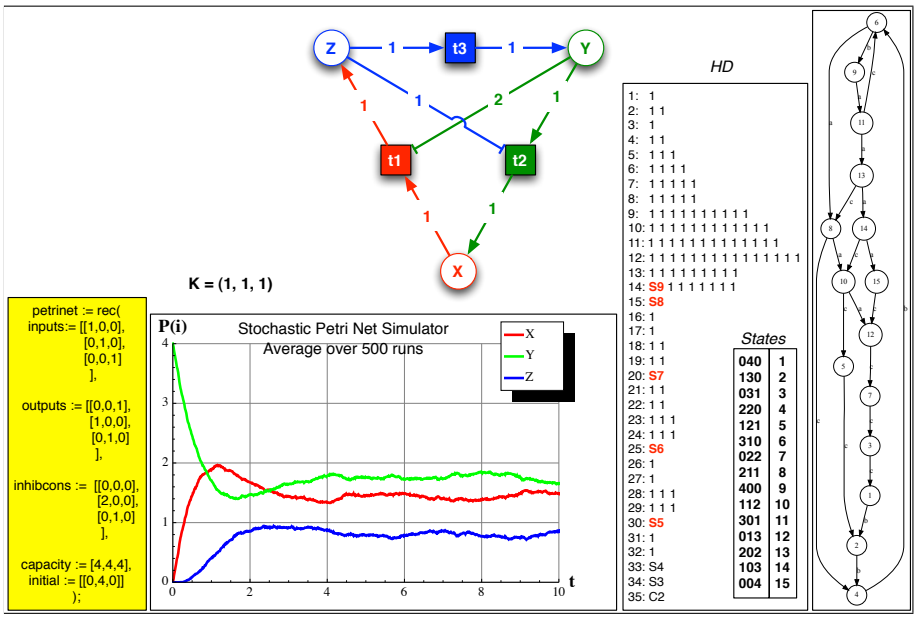

Fig. 3. Analysis of asymmetric variation of very simple BZ-like system

implies that there are reversible subsystems of this system whose dynamics realize this (SNAG) computation by permuting certain "higher-level" (or macro-) states. Natural emerging research questions now are: (1) How, in detail, do oscillatory systems such as the simple BZ-like one analysed here realise this kind of (functionally complete) computation in terms of the dynamics of specific reversible subsystems? And (2) How might the finitary universal computational potential of these BZ subsystems be harnessed?

Acknowledgment. Paolo Dini gratefully acknowledges the partial support of this work by the EU project EINS, Contract number FP7-ICT-288021.

References

1. Arbib, M.A. (ed.): Algebraic Theory of Machines, Languages, and Semigroups. Academic Press (1968)
2. Dini, P., Schreckling, D.: A Research Framework for Interaction Computing. In: Colugnati, F.A.B., Lopes, L.C.R., Barretto, S.F.A. (eds.) Digital Ecosystems: Proceedings of the 3rd International Conference, OPAALS 2010. pp. 224–244. Springer LNICST, Aracaju, Sergipe, Brazil, 22-23 March (2010)
3. Dömösi, P., Nehaniv, C.L.: Algebraic Theory of Automata Networks. SIAM, Philadelphia (2005)
4. Egri-Nagy, A., Mitchell, J.D., Nehaniv, C.L.: Algorithms for the Efficient Calculation of the Holonomy Decomposition. In: Dömösi, P., Iván, S. (eds.) Automata and Formal Languages: Proc. 13th Intern. Conference AFL2011. Institute of Mathematics and Informatics, College of Nyíregyháza, Debrecen, Hungary, 17-22 Aug (2011)

5. Egri-Nagy, A., Nehaniv, C.L.: PN2A: Petri Net Analysis GAP Package, http://sourceforge.net/projects/pn2a/
6. Egri-Nagy, A., Nehaniv, C.L.: Algebraic properties of automata associated to Petri nets and applications to computation in biological systems. BioSystems 94(1-2), 135–144 (2008)
7. Egri-Nagy, A., Nehaniv, C.L.: Hierarchical coordinate systems for understanding complexity and its evolution with applications to genetic regulatory networks. Artificial Life 14(3), 299–312 (2008), (Special Issue on the Evolution of Complexity),
8. Egri-Nagy, A., Nehaniv, C.L.: SgpDec – software package for hierarchical coordinatization of groups and semigroups, implemented in the GAP computer algebra system, Version 0.5.19 (2010), http://sgpdec.sf.net
9. Egri-Nagy, A., Nehaniv, C.L., Rhodes, J.L., Schilstra, M.J.: Automatic analysis of computation in biochemical reactions. BioSystems 94(1-2), 126–134 (2008)
10. Eilenberg, S.: Automata, Languages, and Machines, Vol B. Academic Press, New York (1976)
11. Field, R.J., Noyes, R.M.: Oscillations in chemical systems IV. Limit cycle behavior in a model of a real chemical reaction. Journal of Chemical Physics 60(5), 1877–1884 (1974)
12. Gillespie, D.T.: A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. Journal of Computational Physics 22, 403–434 (1976)
13. Holcombe, W.: Algebraic Automata Theory. Cambridge University Press (1982)
14. Horváth, G.: Functions and Polynomials over Finite Groups from the Computational Perspective. The University of Hertfordshire, PhD Dissertation (2008)
15. Kauffman, S.: The Origins of Order: Self-Organisation and Selection in Evolution. Oxford University Press, Oxford (1993)
16. Krohn, K., Rhodes, J.: Algebraic Theory of Machines. I. Prime Decomposition Theorem for Finite Semigroups and Machines. Transactions of the American Mathematical Society 116, 450–464 (1965)
17. Krohn, K., Maurer, W.D., Rhodes, J.: Realizing complex boolean functions with simple groups. Information and Control 9(2), 190–195 (1966)
18. Maler, O.: On the Krohn-Rhodes Cascaded Decomposition Theorem, http://www-verimag.imag.fr/~maler/, web publication
19. Nicolis, G., Prigogine, I.: Self-Organization in Nonequilibrium Systems. Wiley, New York (1977)
20. Noyes, R.M., Field, R.J., Körös, E.: Oscillations in Chemical Systems I. Detailed Mechanism in a System Showing Temporal Oscillations. Journal of the American Chemical Society 94(4), 1394–1395 (1972)
21. Petri, C.A.: Kommunikation mit Automaten. Schriften des IIM 2 (1962)
22. Rhodes, J.: Applications of Automata Theory and Algebra via the Mathematical Theory of Complexity to Biology, Physics, Psychology, Philosophy, and Games. World Scientific Press (2010)
23. Schilstra, M.J., Martin, S.R.: Simple stochastic simulation. In: Michael, L., Ludwig, B. (eds.) Methods in Enzymology, pp. 381–409. Academic Press (Elsevier) (2009)
24. Scott, S.: Oscillations, Waves, and Chaos in Chemical Kinetics. Oxford University Press, Oxford (1994)
25. The GAP Group: GAP – Groups, Algorithms, and Programming, Version 4.4 (2006), http://www.gap-system.org
26. Zeiger, H.P.: Cascade synthesis of finite-state machines. Information and Control 10(4), 419–433 (1967)