Stability of Metabolic and Balanced Organisations

Pietro Speroni di Fenizio and Wolfgang Banzhaf

Informatik Centrum Dortmund (ICD), Joseph-von-Fraunhofer-Str. 20,
44227 Dortmund, Germany
e-mail: pietro.s@altavista.net, banzhaf@cs.uni-dortmund.de

Abstract. We investigate the possible organisations emerging from an artificial chemistry (AC) of colliding molecules in a well stirred reactor. The molecules are generated from 7 basic components (atoms), each with a different behavior. After discovering two main types of organisations (metabolic o. and balanced o.), we deepen our analysis by studying their behavior over time. The phases they pass through and their stability with respect to an external influx of random information are examined. We notice that no organisation seems to be totally stable over time, yet metabolic organisations pass through a growth phase with a much higher stability. Lastly we observe how the different phases are triggered by the presence or absence of particular atoms.

Introduction

The last years have seen a growing interest for systems where different elements interact to generate new elements, often different from the ones previously present. Those systems, also known as Artificial Chemistries, have been used to model biological, chemical , ecological and social systems. The revolutionary importance of these systems, in comparison to previous linear or non-linear ones, lies in the fact that it is impossible to study them with the method of differential equations. This impossibility and its relevance has been clearly noticed by Fontana and Buss in [Fontana and Buss, 1996].

If we consider S to be the space of all possible elements which can be generated by an Artificial Chemistry , and P(S) the space of all possible subsets of S, we can see that some of those subsets hold two key properties, (1) the property of being a self-maintaining set of elements, and (2) the property of being a closed set of elements. A self-maintaining set of elements is a set where each element can be generated by the interaction of the elements of this set, while a closed set is a set whose elements, as they interact, cannot generate anything external to the set itself. A set which is both self-maintaining and closed is called an *organisation*. Organisations are a central concept in the field of Artificial Chemistries since they represent points of no return in the space of the possible states of the system. When a system reaches an organisation, it can only remain similar to itself, or devolve into a sub-organisation.

All this has already been investigated in [Fontana and Buss, 1993] [Ikegami and Hashimoto, 1996][Dittrich and Banzhaf 1998][Speroni at al, 2000]. Here we wish to address the question what happens if new elements which are not part of the already existing organisation are inserted into the system. Clearly, sometimes the system will

return to the original organisation again. In other cases, it will adapt to a more complex organisation, and sometimes it will pass through a transition phase and reach a different organisation altogether. We are thus studying the stability of the generated organisations respect to an external influx of elements. To study this question we use a system based on combinators [Hindley and Seldin, 1986], [Speroni, 2000].

The System

Our Artificial Chemistry keeps a finite number of basic elements (atoms) which interact to generate combined elements (molecules). The behavior of the system is heavily dependent on the set of atoms which are available at certain moments. Since some organisations tend to consume more and more of the same atoms, they often prepare the right environment for the next organisation to arise. More specifically, in our system we collide algebraic algebraic structures called combinators to generate new combinators. The space of combinators has the following qualities: (1) It is infinite, (2) each combinator directly represents a different operator, (3) every possible operator from the space of strings to the space of strings is present and (4) each combinator (in its normal form) can be expressed using a finite set of basic elements.

Table 1. The seven basic combinators C, R, W, B, I, S, K. s_0 represents the rest of the string. s_0 can be equal to $\varnothing$ or to an arbitrary string.

Atom	Applied Atom	Reduced string	Atom	Applied Atom	Reduced string
C	$C\, x_1 x_2 x_3\, s_0 \rightarrow$	$x_1 x_3 x_2\, s_0$	I	$I\, x_1\, s_0\ \rightarrow$	$x_1\, s_0$
R	$R x_1 x_2\, s_0\ \rightarrow$	$x_1\, s_0, x_2$	S	$S\, x_1 x_2 x_3\, s_0 \rightarrow$	$x_1 x_3 (x_2 x_3)\, s_0$
W	$W\, x_1 x_2\, s_0 \rightarrow$	$x_1 x_2 x_2\, s_0$	K	$K x_1 x_2\, s_0\ \rightarrow$	$x_1\, s_0$
B	$B\, x_1 x_2 x_3\, s_0 \rightarrow$	$x_1 (x_2 x_3)\, s_0$			

Combinators

Each combinator is a string of symbols with balanced parentheses. Each symbol represents an operator that can be applied to the subsequent elements of the string. In our experiments we used 7 basic operators: B, C, K, I, R, S and W. This set of operators contains two disjointed bases of the space of all possible combinators: B, C, K, W, and K, S, I. Each base can generate every possible combinator. Thus, elements of one base might be expressed as combinators of the other base. Some redundancy will therefore be present in the solutions. The seventh combinator, the 'R' has been specifically designed to permit combinators to release independent sub-units (see results section). Each combinator can be seen as a simple Lisp program, which uses the other programs (as data) to generate new programs.
  Each set of parentheses '(,')'encloses a sub-combinator. Each sub-combinator is a combinator in its own right and as such part of the longer combinator. Also each sub-combinator can hold other sub-combinators inside, as a set of parentheses can enclose other sets inside '(( ) ( ))'. In Table 1 all operators, from here on called atoms, and

their effect on the subsequent elements can be seen. Each element x_i can be either an atom or a sub-combinator. If it is a sub-combinator it will be copied altogether (by the atoms W and S), destroyed altogether (by the atom K) or released altogether (by the atom R).

When we write a combinator, some parentheses can be considered useless. These might be eliminated, in particular, any set of parentheses with only one or no element inside (example, '( )' ,'(K)'). Also parentheses which start a combinator or a sub-combinator can be eliminated. For instance, (S(I)((K)K)) can be written as SI(KK).

Reduction Operation and Normal Form Let us consider one combinator. By applying its atoms as operators to the rest of the elements, we move from one combinator to another. This operation is called *reduction*. Each combinator codes for an operator from the space of strings with parenthesis in itself. Each operator can be coded in infinite ways with combinators. When we reduce one combinator to another, through a reduction, the coded operator does not change. An example will clarify this.

If 'a', 'b' and 'x' are combinators, with 'a' and 'b' being one the reduced form of the other ('a'→'b') then 'a'*'x'→'b'*'x'. In our model we consider the list of the operators (the molecules) present at a certain time, regardless of the particular coding with which they were created.

When combinators are reduced, they move from being represented by one string to being represented by another. Sometimes a combinator reaches a form from which no reduction operation is applicable. Those strings are called the *normal form* of the combinator. Not all combinators have a normal form, yet an important theorem in combinator theory declares that *if* a combinator possesses a normal form, this form is unique. It is not essential in which order the atoms are reduced, if a normal form exists, this is unique and it is always possible to reach it [Hindley and Seldin, 1986].

In our system we consider only combinators that possess a normal form and we store combinators in their normal form. When two combinators interact they generate a new combinator and the reduction process starts. A necessary condition for the result to be acceptable is to reach a normal form in less than 'Max_Time' steps.

The Artificial Chemistry

Our system contains a few hundreds of elements in a well stirred reactor, from now on called *the soup*. Those elements interact with each other and generate other combinators, possibly different from both reactants. In our system we do not keep the number of elements fixed. Instead, the number of elements released from the reaction process can vary, depending on the elements involved. What is kept constant, though, is the number of available atoms. Each type of atom exists only with a few copies. Beside the soup we keep a registry (called *pool*) of the available atoms which are supposed to float in the soup. Every time a reaction takes place, the generated combinators consume atoms from the pool, while when the reactants are destroyed their atoms are added to the pool. If there are not enough atoms in the pool or any of the elements does not reach a normal form in a predefined number of steps, the reaction is considered *elastic* in this environment, and the original combinators are retained instead.

When two elements of the soup collide their reaction generates a multiset of new elements. In a formal way we can write the reaction in the following form:

$$A*b \rightarrow a(b) \rightarrow c_1...c_n \qquad (1)$$

with $a,b,c_1,...,c_n$ combinators. c_1 is then the direct result of the normalisation process of $a*b$, while $c_2,...,c_n$ are the other elements that are released in the operation.

We now present a short example for illustration purposes. Let two elements be: WR and (SKI). If we apply the first to the second we reach WR(SKI) which can be reduced to R(SKI)(SKI), then to SKI releasing a separate copy of SKI.

$$WR*SKI \rightarrow WR(SKI) \rightarrow R(SKI)(SKI) \rightarrow SKI, SKI \qquad (2)$$

In our system we measure the time in physical and biological generations. If NMolecule are present at a certain time then, after each interaction, the physical clock will be advanced 1/NMolecule generations, while the biological clock will be advanced by the same quantity, only after non elastic interactions.

The Influx We provide the system with a continuous influx of random information. Each physical generation, with a probability P_{in} depending upon the number of existing molecules, we randomly assemble a new molecule to be inserted into the soup. The molecule is assembled from the elements of the pool, so the total amount of atoms in the soup remains constant. P_{in}(NMolecules) is an exponential function with

$$P_{in}(\text{NMolecules}) = 1 \qquad \text{for} \qquad \text{NMolecule} = \text{MinNMolecules} \qquad (3)$$

and

$$P_{in}(\text{NMolecules}) = 0.5 \qquad \text{for} \qquad \text{NMolecule} = \text{HalfProbMolecules} \qquad (4)$$

The influx function in this contribution is different from the influx function presented in [Speroni, 2000] and this explains some of the differences in the behaviour of the generated organisations. In order to generate a random molecule we pick each atom with a probability of $1/9^{th}$, open a parenthesis with a probability of $1/9^{th}$ and close one with the same probability. If we close a parenthesis that has never been opened the combinator is terminated. If an atom which is not present in the pool is required the combinator is aborted and a new one is tried. After a combinator has been constructed it is reduced to its normal form and the resulting combinator is inserted into the soup provided normalisation is possible.

The K, R Atom and the Outflux A seemingly small difference from the previous model [Speroni, 2000] is the presence of two molecules which eliminate sub-combinators. Both the K and the R eliminate sub-combinator from a molecule. The first totally eliminates it (re-splitting it into atoms), while the second releases it as a separate component. In the previous model only one was present (called K, but working as R here, releasing the sub-element). In that situation, if a molecule was

assembled, the only way to disassemble it was by applying this molecule to other molecules. Yet some molecules just would not normalise into shorter form. And as more and more molecules would be frozen in this state, the system would stop. As a consequence, that system needed a constant outflux.

In the present system, the need to get rid of bulky molecules is not solved for the system. We leave such job as a problem to be solved by the appearing organisations. And (as we shall see) it is a job they happen to solve really well.

Results:

We asked our system few simple questions:
1. What are the possible organisations reachable?
2. Can those organisations be divided into broader categories?
3. Once the system has found an organisation, how stable is such a solution with respect to a random influx of information?
4. Does such stability depend on the type of organisation?

Because of the universal characteristics of the space we are working with, many organisations are possible. Interestingly, many of them also seem to be accessible to the system (an organisation could be present in the space of all the possible ones, but be too complex to be found). We made 150 runs with parameters: operation type A°B→A(B), atoms used all 7. Number of atoms 2000 of each type. MinNMolecules=50, HalfNMolecules 300, no outflux, no mutation, MaxTry=10000. MaxLength=100, MaxDepth=20. Starting number of randomly generated molecules 300. Length of the experiment 10000 generation. We then made 26 esperiments with the same parameters, but length of the experiment 30000.

Each run is then much longer than the runs in the previous work (10000 and 30000 generations, compared to the 1000 of [Speroni, 2000]). The first interesting observation is that *no organisation seems to be totally stable*. Some experiments reached an organisation, kept it for many thousand generations, then unexpectedly switched to a different one. So far, however, we were not able to gather enough data to check whether there is a power law distribution between size and frequency of extinction events of molecules.

Two types of organisation have been observed, *metabolisms*, and *balanced organisations*. Other organisations could be possible as well, but due to the constant influx with which we fed the system, only relatively stable organisations could be noticed in the experiments.

Metabolism

Metabolisms are particular organisations that were recognised and presented for the combinator system in [Speroni, 2000]. Here, we keep the definition of metabolism given in [Bagley and Farmer, 1992]: "A metabolism takes in material from its envi-ronment and reassembles it in order to propagate the form of the host organism...".

 Those organisations gained their name by their ability to 'use' external elements as inputs to grow, while being unable to increase their size without an external influx of elements. As they grow those metabolism preserve their digital form: elements

present in them, and quantitative relations between them. In the earlier work, the influx of elements would stop after 300 elements were present. Here, the influx follows an exponential law and is never totally absent. For this reason the global behavior of the new metabolism is now qualitatively different. Metabolisms in our system have some unique features that makes them easy to recognise. They have, usually, a small number of different elements, sometimes just one or two, and their interaction doesn't lead to the generation of more elements of the same kind.

An example of a simple metabolism would be a single element 'a' which, applied to itself, generates two copies of itself

$$a*a \rightarrow a,a \tag{5}$$

So two elements would be flowing into the reaction and two elements would be flowing out of it. And the elements are the same so the composition of the soup is not changed. Yet if we observe the system after some time we notice an increase of 'a' elements. How can this happen?

The reason lies in the external random elements that we regularly insert. In fact, many times those elements give rise to pathways which end up with the production of more elements of the metabolism. An example will clarify this. Suppose that the molecule S is thrown into the soup.

$$S*a \rightarrow Sa \tag{6}$$

$$Sa*a \rightarrow Saa \tag{7}$$

$$Saa*a \rightarrow aa(aa) \rightarrow a,a,a,a \tag{8}$$

So the whole reaction was

$$S*3a \rightarrow 4a \tag{9}$$

Reactions like this happen very often and the net result is an increase of elements of the metabolic type in the soup. On the other hand, not all molecules inserted generate pathways that lead to the release of a's. Sometimes the molecules just gets bigger and bigger, soon reaching the physical limits imposed on the dimension of a possible molecule (as one of the system parameters). When this happens, the molecule will just remain present in the soup without being able to participate in any further reaction. This second type of reactions is, in general, relatively rarer, and only some atoms gets frozen as long molecules.

The Active Phase and its Stability. A metabolism will, in general, have various possible phases. The first is always the *active phase*. Here the number of molecules of the system tends to increase as new molecules are randomly inserted. Some junk molecules are generated, too, but don't, normally, interfere with the process. During this phase we can notice a constant increase in the number of molecules, a stability in the number of different molecules, a decrease in entropy, nearly no innovativity, and a constant decrease in the free atoms in the pool. The number of atoms will tend to decrease following a precise relation between the various types of atoms. Once one of the atom types is totally consumed this phase ends and the *resting phase* begins.

In an active metabolism no inserted molecule seems to be able to push the system

into another organisation. Each randomly inserted molecule is 'digested' before it can harm the organisation. Digested molecules end up either being molecules of the metabolism or junk long molecules, unable to react.
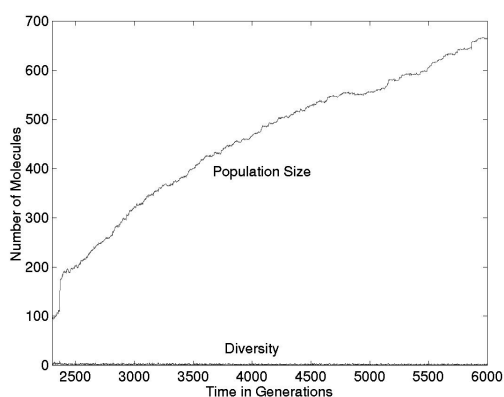

Fig. 1. Example of a metabolism in its active phase.

In some rare case the situation is reverted if a particularly 'harmful' molecule is inserted. Once this happens the metabolism starts loosing elements. In general, elements that revert the metabolism are elements that are able to destroy other molecules without being destroyed (or modified) themselves, for instance of the form 'b' such that b*x$\rightarrow$'b'. Often a metabolism (especially in its first active phase) is too simple to deal with those molecules. If this happens the system shrinks in size. Once the number of elements drops to less than 50 molecules (MinNMolecules), many new molecules start to be inserted into the soup. New reactions begin to happen and the system will enter a transition phase. From here it might revert back to the old metabolism (with the dangerous molecule being removed from the scene by the flood of new molecules), to another metabolism, or to a totally different organisation. Even in this latter case we don't have an abrupt transition to a new organisation, but rather first a shift to a different phase. In a futuristic picture were such organisation is man-made, and performing a particular task, it would be possible to invoke some correction operation before the system gets out of control.

Even if no dangerous molecule is inserted, the general behaviour of a metabolism is directly linked to the molecules of the influx. In our standard example each molecule has double the probability to contain a growing atom (S or W) than to contain a shrinking one (K). In another experiment no W were allowed and metabolisms were very rare. Moreover, those metabolisms did not grow monotonically, but instead behaved as in a random walk in the number of molecules.

Rest-Post Metabolic Cloud. In a limited medium a metabolism cannot grow for ever. Once at least one of the atom types is exhausted, the metabolism passes from the active phase to the resting phase. In this second phase elements are still inserted and the metabolism tries to transform them as it did before. Yet, since there are not enough free atoms in the pool, many transformations stop halfway. Recall that each

interaction to be accepted needs to reach a normal form on every molecule generated. Yet in order to 'digest' a molecule many separate reactions are usually necessary, making it possible for the process to stop halfway. If this happen many new molecules are generated, the diversity and entropy increases and many new reactions are tried out.
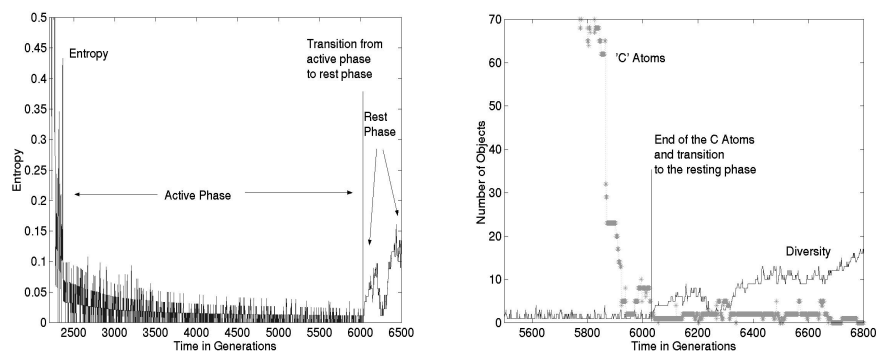
Fig. 2. Transition from active to resting phase. On the left entropy, on the right diversity and Number of C Atoms. Both entropy and diversity increase as one of the atom types is exhausted.

Balanced Organisation

We call *balanced organisations* closed self-maintaining sets which do not need external input to grow. They, too, metabolise external elements, but instead of using their active components to grow (as the metabolism does with the 'S' elements in the
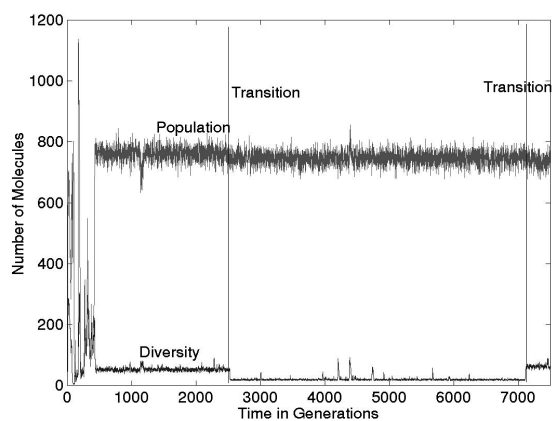
Fig. 3. Transition from a balanced organisation to another.

previous example), they just split them up into their atomic components, then reach back for the elements in the pool to grow. The ability to 'split up' components, unused in metabolic organisations, is new to this version of the system, and presented in 'the K,R atom and the outflux' paragraph. Balancedorganisations often hold two types of molecules, molecules which build and molecules which destroy. The dynamic balance between these two types permits the organistion to build itself continuously and to efficiently destroy the influx elements which could destabilise such balance.

This kind of organisation tends to fluctuate around a particular average value with a corresponding standard deviation. Balanced organisations are not totally stable either, and the system will often percolate to a new organisation being sometimes a balanced organisation, sometimes a metabolism.

The Space of Organisations

A complementary point of view to this dynamic picture is instead to track the movement of the system in the space of all possible organisations. Such a space, spanned by all possible organisations and their relations of intersection $\cap$ and union $\cup$ generates a lattice [Speroni at al, 2000]. Every set of elements 'S' uniquely defines an organisation O_s. Every time we intersect two organisations we uniquely define another organisation, the biggest organisation contained in the intersection. Every time we unite two organisations we, again, uniquely define an organisation, the smallest organisation generated by the union of the two sets. The operation of union is particularly important in our case, since every time we insert a new element into an existing organisation we potentially push it towards a larger organisation. As the system changes in time, its change can be observed as a movement in this lattice of organisations.

Conclusion

As we study specific artificial chemistries more and more properties seem to emerge. We studied a model which used as elements molecules made up of 7 types of basic 'atoms'. We have observed that many organisations are possible, yet none seems totally stable with respect to an external influx of random elements. We discussed how a particular kind of organisation, metabolism, uses such an influx. As long as it is able to transform the elements of the influx, it is able to keep itself relatively stable, while as soon as the influx gets too large or the organisation is unable to transform it any more, it is destabilised. We observed that often the exhaustion of one or more basic atomic types was able to push the system into a different organisation.

The field of artificial chemistries is still a quite young subfield of artificial life. Applications are sporadic at best. Yet the numerous natural examples of systems which contain different interacting elements seem to suggest vast possibilities of application. All too often well planned systems based on few precise components seem unable to cope with the enormous variety of forms that nature provides, and end up being used under very limited work conditions only. The study of Artificial Chemistries, on the other side, of their metabolic organisations and features like

stability could permit to organise vast changes in the world that would otherwise be impossible to obtain. If those tools are developed it will be important to slow down to the point of being able to use them with wisdom, being able to use them for entire humanity more than for a selected group of people, for the whole nature more than only for human beings.

Acknowledgements

References

Bagley J. R. and Farmer J. D. 1992. *Spontaneous Emergence of a Metabolism*. In Proceedings of The Second International Conference on Artificial Life, 94-140. Edited by Langton C. G., Taylor C., Farmer J. D., and Rasmussen S., Redwood City, Addison-Wesley.

Dittrich P. and Banzhaf W. 1998. *Self-Evolution in a constructive Binary String System*. Artificial Life 4(2):203-220.

Fontana W. 1992. *Algorithmic Chemistry*. In Proceedings of The Second International Conference on Artificial Life, 159-209. Edited by Langton C. G., Taylor C., Farmer J. D., and Rasmussen S., Redwood City, Addison-Wesley.

Fontana W. and Buss L. W. 1993. *"The arrival of the Fittest": toward a Theory of Biological Organization*. Bulletin of Mathematical Biology.

Fontana W. and Buss L. W. 1996. *"The barrier of objects: From dynamical systems to bounded organizations"*, In Boundaries and Barriers, J.~Casti and A.~Karlqvist, 56 – 116, Redwood City, Addison-Wesley.

Hindley J. R. and Seldin J. P., 1986. *Introduction to Combinators and λ-Calculus*. Cambridge University Press.

T. Ikegami and T. Hashimoto, Active mutations in self reproducing networks of machine and tapes Artificial Life 2(3):305--318. 1995

Kauffman, S. 1993. *The Origins of Order: Self-Organisation and Selection in Evolution*. Oxford University Press.

Speroni d.F. P. 2000. *A Less Abstract Artificial Chemistry*. In Artificial Life VII: Proceedings of the Seventh International Conference, edited by Bedau M.A., McCaskill J. S., Packard N. H. and Rasmussen S, Cambrdige, MA, MIT Press.

Speroni d.F. P., Dittrich P., Banzhaf W. and Ziegler J. 2000. *Towards a Theory of Organizations*. In Proceedings of the 4th German Workshop on Artificial Life (GWAL 2000).

Suzuki Y., and Tanaka H. 1998. *Order Parameter for Symbolic Chemical System*. In Proceedings of The Sixth International Conference on Artificial Life, 130-139. Edited by Adami C., Belew R., Kitano H., and Taylor C., Cambridge, MA, MIT Press.