Replicators That Make All Their Own Rules

Tim J. Hutton

http://www.sq3.org.uk
tim.hutton@gmail.com

Abstract. We present a novel form of self-replicator in an artificial chemistry, inspired by DNA and designed to facilitate studies of evolution and creativity: an Enzyme Artificial Chemistry (EAC). A string of ‘atoms’ moves in a two-dimensional lattice space, and interacts with other atoms through local reactions. Each atom can act as an enzyme, catalysing a reaction when the correct reactants are nearby. We show how a string of such enzymes is able to self-replicate using free-floating atoms as ‘food’. Mutations cause atoms to be removed from a string, or added with a random enzyme, allowing the very method by which the strings replicate to alter. Evolution in the system is demonstrated.

1 Introduction

The discovery of an efficient computational system that can undergo evolution, and produce the evolutionary growth of complexity [13] is a central goal of artificial life research [3]. Ever since von Neumann presented his self-replicating automaton [22], the search for a self-replicator that is capable of information inheritance and of interactions with its competitors has been ongoing [16]. Various media have been explored for the purpose, including cellular automata, machine-code systems and artificial chemistries.

Arguably the most progress in digital evolution has been made on the Avida system (see for example the recent special issue [2]), in which self-replicating machine-code programs compete for space and other resources. By introducing rewards for certain computational tasks performed (essentially an external fitness function), the evolution of complex features was shown possible [10], against a natural pressure towards shorter programs.

However, it has been suggested that the design of Avida and related systems lacks some of the features necessary for open-ended creative evolution: embededness, rich interactions and materiality [18–20]. Indeed, without an externally imposed sequence of rewards, the organisms in Avida do not exhibit ongoing adaptations to their environment and instead approach the smallest program capable of self-replication. Even with the rewards, the organisms perform the tasks required for them and nothing more - you never get more out than you put in. While recent versions of Avida allow varying levels of interactions through input and output registers and depletable shared resources [15, 5], this has not yet proved sufficient for evolutionary development to continue indefinitely.

Artificial chemistries (ACs) by their nature have the properties of embededness, rich interactions and materiality, thus making them a promising avenue of exploration for evolutionary systems. In this paper we present an AC system in which strings of 'atoms' joined together to form 'molecules' move in a flexible way in a two-dimensional lattice space. By allowing each atom to act as an enzyme, catalysing reactions between nearby reactants, a molecule is capable of self-replication, by making bonds between free-floating atoms it encounters through diffusion and copying over state information. Unlike in previous such systems, the molecules encode all of the rules necessary for self-replication, allowing the very method of replication to evolve.

2 Background

In [6] a system of eight reaction rules was presented that allowed molecules in an artificial chemistry to replicate. The reaction rules could not be modified through evolution and in experiments the smallest molecule always dominated because it could replicate fastest.

In [8] this system was extended to give the molecule some phenotypic properties that could be selected for. With additional system rules, it was shown that longer molecules could dominate in certain environments, and that with partial mixing between heterogeneous environments, the length of the molecule would evolve upwards. However, the evolutionary progress was strictly limited.

In [7] the rules were again extended to allow *enzymes* to be produced, by creating a mapping between the base-sequences in the molecule and the space of possible reactions. This allowed the system rules to be augmented, potentially allowing the molecules to evolve into different forms. However, with naked replicators mingling with their competitors, it was found that no survival advantage could be conferred by the production of enzymes, since parasites would soon appear to take advantage of any useful enzymes that were present.

Thus in [9], a new set of rules was designed that allowed the molecule to replicate inside a semi-permeable cell-wall, and to cause its division and growth. In this system the molecules could now take advantage of the enzyme-production facility, and it was shown that such an enzyme would be conserved against a mutational load. However, no other evolutionary activity was observed, most likely because the cost of replicating the extra bases required to produce additional enzymes was too great relative to the benefits that would be obtained. To reproduce all of the replication rules as enzymes would require a sequence over 700 bases long, and a cell wall large enough to accommodate such a string. In a computational simulation this was found to be too large a structure to usefully study evolution.

In this paper we present a much more compact solution to the issue of evolvability in an AC system. We use the conceptual correspondence between a gene and the enzyme it produces to compress a sequence of bases into a single atom that we give catalytic properties. Every atom has the capacity to behave as an enzyme, and we term this form of AC an *Enzyme Artificial Chemistry* (EAC).

While this design decision abstracts the system further from a simulation of natural biology, the concept of information being inherited and used to control the local environment is retained. In the next sections we give details of the system rules and the experimental conditions, and then the results are discussed.

3 System Description

The EAC is composed of numerous ‘atoms’ which are simple units of matter. Each atom has a *type* $\in \{\mathtt{b}, \mathtt{e}, \mathtt{f}\}$ and a *state* $\in \mathbb{Z}$. The type does not change (thus the proportions of each type are fixed from the beginning of the experiment) but the state can be changed by reactions. Each atom can be bonded to other atoms, these bonds are stored in a list in the atom’s state information. Each atom can also act as an enzyme, catalysing a specific reaction whenever the correct reactants are nearby. The reaction is also stored in the atom’s state information.

The atoms diffuse on a two-dimensional square lattice, each moving to a position within its Moore neighbourhood on each timestep. Constraints apply to the movement, and the final move is chosen at random from the set of permissable moves, if any. These constraints are: a) the position must not contain another atom, b) the position must be within the radius-2 Moore neighbourhood of every atom the atom is bonded to, c) bonds must not become crossed as a result of the move, and d) if an atom has two or more bonds the move cannot cross an existing bond. These physics rules yield movement that is flexible and reasonably efficient, and allows for bonds to change the nature of the local environment, for example by trapping certain components in a closed loop of atoms. While more complex than necessary for the work presented here, these rules allow the creation of enclosed self-reproducing cells on a discrete grid, following the rules and structure given in [9]. The rules for self-replication given below would actually work with a wide range of physics laws, including a continuous space model, and a 3D world as opposed to a 2D one, as long as basic requirements of diffusion and shape flexibility are satisfied.

Bonds between atoms are made and broken by reactions. For example, the reaction:

$$\mathtt{e1} + \mathtt{b1} \rightarrow \mathtt{e2b3}(\checkmark) \tag{1}$$

means that atoms `e1` and `b1` will bond when they encounter each other, and adopt states `2` and `3` respectively. The atoms involved in a reaction may be bonded to other atoms, only their type and state determines whether they match as reactants. Reactions can be written using type variables x and y to save space. The tick (or cross) indicates whether the reaction of the first atom is copied into the second (overwriting any present).

Our ancestor molecule is a string of atoms `e8b1b1b1b1b1b1b1b1f1`, where each `b` atom is an enzyme, carrying one of the reactions shown in Table 1. These rules are the same as in [6] with the addition of the enzyme-copying flag. A soup of atoms in state `0` is required for replication to continue.

	reaction	copy enzyme?
1	e8+e0→ e4e3	✓
2	x4y1 → x2y5	✗
3	x5+x0 → x7x6	✓
4	x3+y6 → x2y3	✗
5	x7y3 → x4y3	✗
6	f4f3 → f8+f8	✗
7	x2y8 → x9y1	✗
8	x9y9 → x8+y8	✗

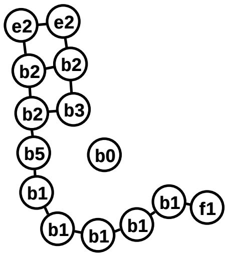
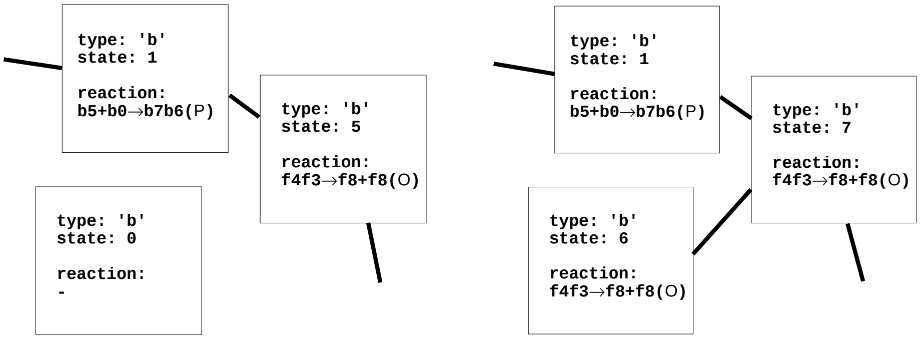


Table 1. *(left)* The hand-designed reaction rules for self-replication used by the ancestor molecule in our enzyme artificial chemistry. The last column specifies whether the reaction stored in the first reactant is copied over to the second reactant (overwriting any previously stored). With a supply of atoms in state 0 these rules are sufficient for a molecule to direct its own replication. *(right)* An example of a molecule in the process of replicating. The next reaction that can be applied is $x5 + x0 \rightarrow x7x6(\checkmark)$ which will bond the free-floating b0 atom to the strand and copy an enzyme to it. This reaction is illustrated in more detail in Fig. 1.



(a) before

(b) after

Fig. 1. An illustration of enzyme action in our enzyme artificial chemistry. The atom b1 has a reaction that matches nearby reactants, b5 and b0. This reaction is therefore catalysed, the atoms become bonded together and have their states changed to 7 and 6 respectively. Additionally, the reaction of the first reactant is copied into the second. With this type of enzyme action, no additional system rules are required for information replication, since the enzyme-carrying molecules direct their own replication.

The enzymes in the strand operate only locally: a search is made to within an extended Moore neighbourhood of radius N (we used $N = 5$ in the experiments below) for reactants that match the specifications for each reaction. If the reactants are found, and no bonds would become crossed, the reaction is applied. To prevent parasitism (strands taking advantage of the enzymes carried by their competitors), we also require that one of the reactants is connected (directly or indirectly) to the atom carrying the enzyme, and that the other is either connected or has no bonds. While this requirement is non-local and thus unsatisfactory, without it we would require a physical mechanism for preventing enzymes assisting their competitors, such as a cell wall, and while such structures are possible [9] they are slower to reproduce and require more space. Figure 1 illustrates the way enzymes work.

To permit replication to continue after the resources of an environment are used up, we use the same mechanism as in previous systems: a *flood*. All the atoms in one section of the world (here we use quarters of the world, and rotate between the four) are assigned a special dissolving state (-2) that breaks all the atom's bonds and leaves it in a food state (typically 0). Additionally, the dissolving state is assigned to all the atoms that were bonded to that atom, causing any of the long tangled chains that tend to form to be removed. Figure 2 illustrates the process of growth and floods, here with a flood period $T = 3000$ iterations.

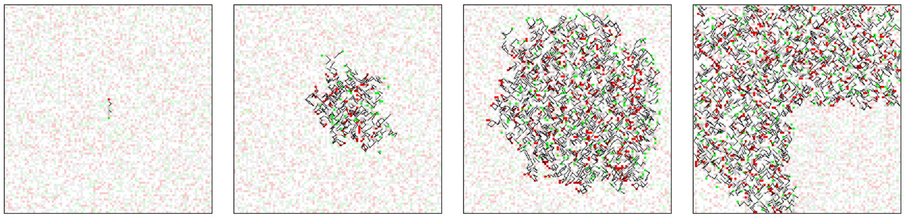


Fig. 2. Snapshots from a simulation run, at iterations 1, 1232, 2160 and 3005. From a single molecule grows a population that fills the available area until periodic floods remove some of them and replenish the supply of atoms that can be used for food.

In the system as presented there is no mechanism for variation to be introduced into the information inheritance, only exact copies of the ancestor will be made. An important consideration for evolution is therefore the mechanisms for introducing mutations. We want to avoid introducing extra atoms, and thus we simply add an extra rule that says that with some low probability (chosen at the start of the experiment) a free-floating b atom will become inserted into a string of atoms in state 1 and given a random enzyme. A similar rule is made for gene deletion (see R15 in [8]). This rule assumes that gene-strings spend time with their atoms in state 1, a property that in theory could change through evolution, but it suffices for now.

For new enzymes, the space of possible reactions is of size $t^2 s^4 2^3$, where t is the number of types available (eg. $t = 3$ for b,e,f) and s is the number of states. The last term, 2^3 simply expresses the fact that there are 3 boolean flags, two for whether the reactants are bonded or unbonded before and after the reaction, and one for whether the enzyme should be copied or not.

The EAC has a rich array of potential behaviours, some of which are problematic for studying evolution. One example is that trivial self-replicators exist: consider a b1 atom carrying the reaction $\texttt{b1} + \texttt{b0} \rightarrow \texttt{b1} + \texttt{b1} (\checkmark)$. The appearance of such an entity would have the effect of rapidly converting all available b0 atoms into copies of itself. Replicators such as the ancestor molecule are vulnerable to this kind of effect and thus for the moment we forbid such trivial replicators by stipulating that atoms with no bonds have no enzymatic effect. Only by having to deal with the problem of bonding and unbonding (to avoid the flood) can a sequence of enzymes cause copies of itself to be perpetuated.

A related problem is caused by the appearance in genomes of enzymes that cause a gradual depletion in the amount of available food, such as $\texttt{b2} + \texttt{b0} \rightarrow \texttt{b2} + \texttt{b9} (\text{✗})$. A molecule that acquires such an enzyme will remove its own food and thus not replicate well but it may also cause a global extinction event, which we want to prevent. The EAC lacks a concept of energy, or metabolism, and instead we impose a simple limit k on the number of times that an enzyme can be applied, before the state of its atom changes, at which point the counter gets reset to zero. $k = 10$ was used in all the experiments below, although the effect of this constraint has not yet been fully explored.

4 Experiments and Results

To demonstrate the robustness of replication and the evolvability of the genome, we provide as food not only atoms b0 but also b10, b11 and b12, although the ancestor molecule (Table 1) is unable to make use of these. The ratio of these components in the random soup is 2:1:1:1. Additionally atoms e0 and f0 are provided at sufficient levels. The acquisition (through mutation) of an enzyme such as $\texttt{b5} + \texttt{b10} \rightarrow \texttt{b7b6} (\checkmark)$ should be adaptive, since it will enable the molecule to replicate more often than its competitors.

To monitor what is happening during an experimental run, we regularly log a profile of the environment. Figure 3 shows a typical plot, for several atom types. Such plots inform the experimental choice of the flood period, since the intention here is to create competition for a limited resource, b0. The plot shows that while e0 and f0 atoms are plentiful, the supply of b0 atoms runs out completely before being replenished by the next flood.

Additionally, we monitor the occurence of reaction 1 (Table 1) and read out the sequence of enzymes in the molecule for logging purposes. While the physical structure of the self-replicator may change through evolution (and thus reaction 1 may not occur), for the moment this is sufficient. We assign each unique sequence an id number, and a plot of these ids against time (Fig. 4

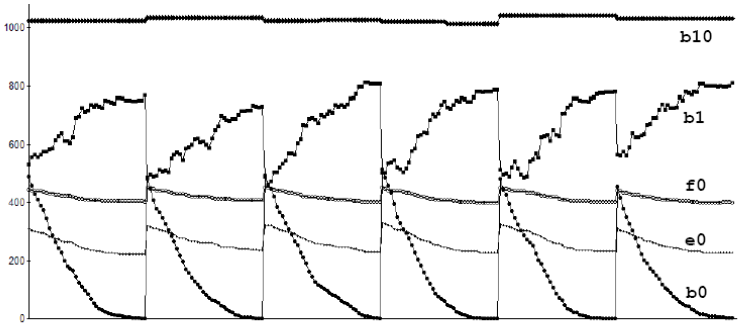


Fig. 3. A plot of the frequency of several atom types over time in a typical run. Atoms that are used as food (e0, b0, f0) get depleted as they are consumed by replicators and replenished by periodic floods, while atoms found in replicators (b1) accumulate and are then removed. The frequency of atoms not currently being used as food (b10) merely fluctuates slightly with each flood.

(top)) reveals the dominant 'species' and those that replicate several times but then disappear.

When useful enzymes appear, the species holding them will most likely (barring the effects of a finite population) come to dominate the world. Figure 5 shows an example of where this has happened, with the atom type b10 becoming used as food. However, in this run the population went extinct shortly afterwards, perhaps due to the mutational load being too high. The novelty of the system means that the parameter space has yet to be fully understood, and much experimental work is required in this direction.

To further demonstrate the effects of different species coming to dominate, we reduce the size of the reaction-space somewhat, making it easier for mutations to hit upon useful enzymes and also reducing the risk of world-killing enzymes appearing. Mutations now select reactions in the pattern $\texttt{b5?b}i \rightarrow \texttt{b7?b}j(?)$, where the question marks indicate unspecified boolean flags and i and j are between 0 and 12 inclusive. Thus there are $13^2 2^3 = 1352$ possible enzymes, of which we expect 3 to be adaptive. (In the full enzyme space there are $3^2 13^4 2^3 = 2$ million possible enzymes.) Figure 6 shows the results of one simulation run using these parameters.

5 Discussion and Conclusions

The simulation runs shown exhibit some evolutionary activity but have not satisfied our goal: the creation of a system that exhibits open-ended creative evolution. We have demonstrated that larger molecules can outperform shorter ones, and that these differences can arise through random mutation. In the simulation runs shown here, the innovations that emerge are precisely and only those that we were expecting and encouraging. However, the EAC has some desirable properties: it is reasonably efficient to implement (the experiments above took

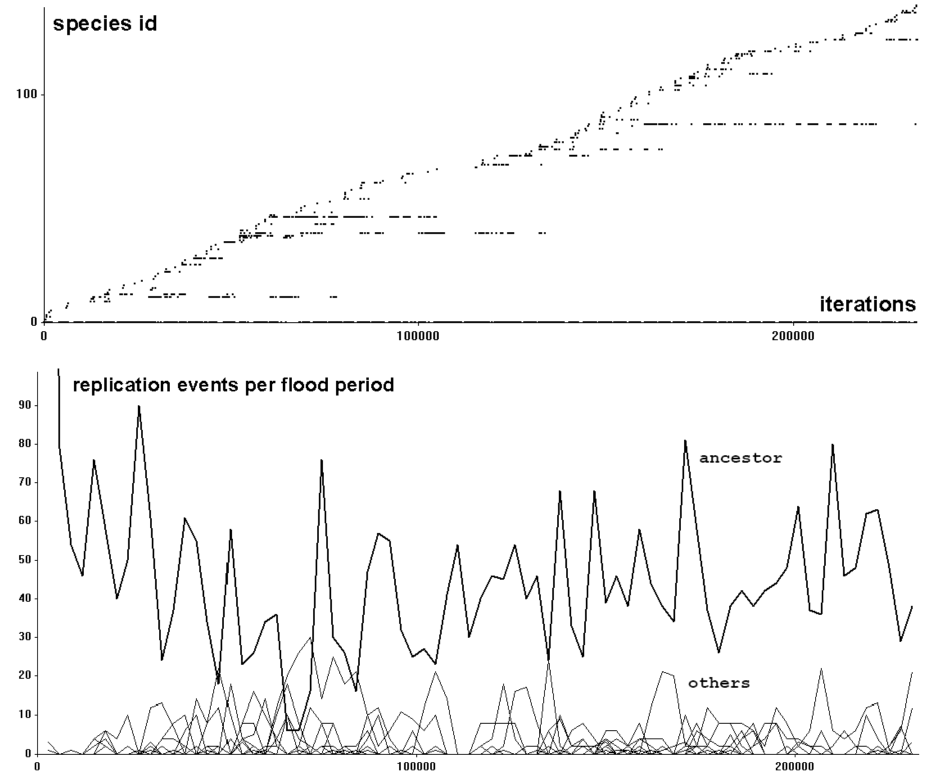
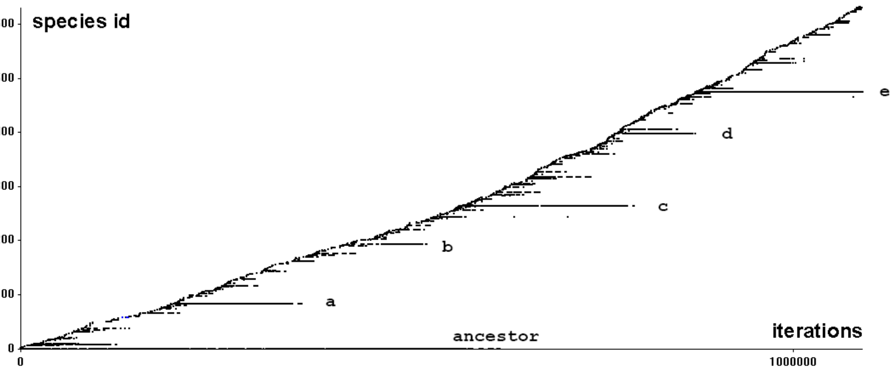


Fig. 4. *(top)* A plot of the occurences of reaction 1 at different timepoints in one run of the simulation, indicating each replication event. The species id label on the vertical axis is plotted against time on the horizontal axis. *(bottom)* The same simulation run shown as replications per flood period against the same time period. The ancestor species (id 0) is shown with a thick line and clearly dominates for the period shown here.



a	b11b6→b9+b6(✓)
b	b8b9→b2+b11(✓)
c	b5+b10→b5+b0(✓)
	b11b11→b5+b4(✓)
d	b6b8→b12+b0(✗)
	b7b10→b0b12(✓)
	b5+b10→b5+b0(✓)
e	b8b6→b12b9(✗)
	b12b9→b10+b8(✗)
	b6b8→b12+b0(✗)
	b7b10→b0b12(✓)
	b5+b10→b5+b0(✓)

Fig. 5. A simulation run where different species became dominant over time. The unique species id is plotted against time on the horizontal axis. Five species are identified and the additional enzymes that they carry are listed in the table. Species a and b carry enzymes that are not useful but also do no harm other than to make the replication time slightly longer, these species lasted a while but were eventually killed off by the ancestor species. Species c has an enzyme that speeds up replication time, by making b10 atoms available as food. This species kills off the ancestor genome at around 600K iterations and dominates the world for a while. Species d and e carry this enzyme but also other non-harmful ones. It is not clear why these species were succesful, more experimentation with the parameters is needed to determine whether this effect is reproducible.

no more than a few hours each on a single PC), it implements full *self*-replication (making the replication method accessible for alteration and improvement), arguably it is fully embedded (in the sense described in [19]), and it is a material system that has rich interactions between the individuals and their environment. The nature of each individual is not well-defined, being instead a construct of the sequence of reactions that it embodies. This permits the nature of the individual to change in dramatic ways, in theory, potentially undergoing the kind of major transitions that were seen in evolution on Earth [11]. We would suggest

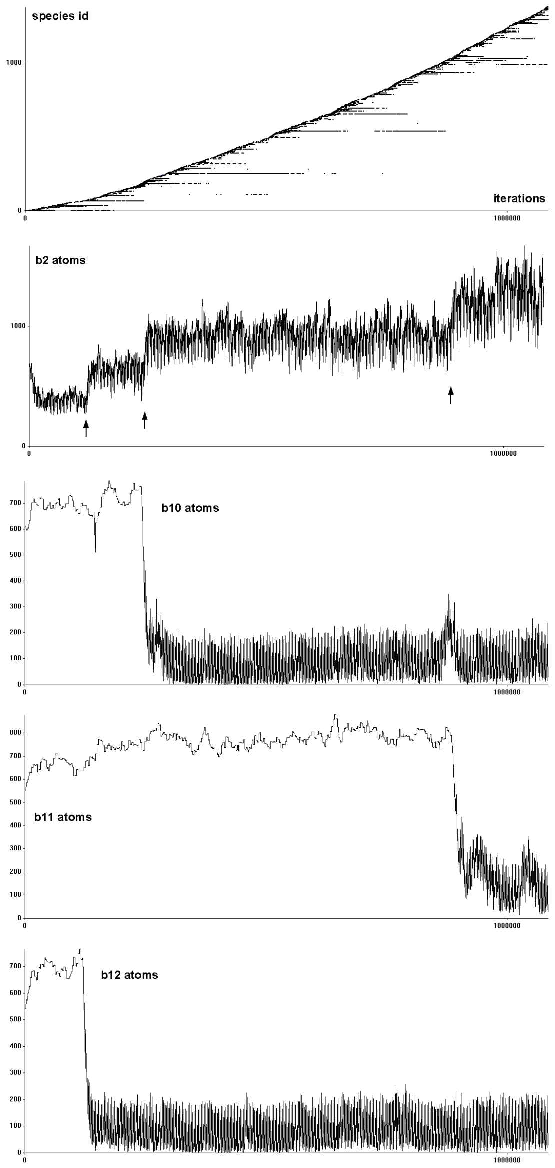


Fig. 6. A simulation run where all the available food types are eventually utilised. At the top is shown a plot of the unique sequence ids against time, showing different species coming to dominate. The second row shows the number of `b2` atoms present in the world, against the same time scale. Where different food types become utilised, the supportable population (and hence the number of `b2` atoms) increases. The next three rows contain plots of the amounts of `b10`, `b11` and `b12` present in the world. As enzymes that make use of these potential foods appear, the amount of each drops suddenly and exhibits the fluctuation seen in Fig. 3 for `b0`. The moment of the appearance of each of the enzymes concerned is marked in the `b2` plot with an arrow, these occur in the order `b12`, `b10`, `b11`.

that the creation of a system with these properties is an important step towards simulating the evolutionary process.

The trick of encoding the system rules into the atoms as enzymes has the great advantage of allowing us to create a small molecule-like object that is truly a *self*-replicator, and that replicates fast enough to enable the evolutionary process to be studied experimentally. However, the disadvantage of this scheme is that the encoding between genotype and phenotype (here the actions of the enzymes) is hard-wired into the system and not accessible to evolutionary change. By contrast, both DNA and von Neumann's automaton [22] use a translation scheme whose mechanisms are themselves specified by the genome. Several authors have highlighted the potential evolvability implications of this feature [4, 12, 14, 17] although these have yet to be proven experimentally.

To achieve open-ended evolutionary activity in an extension of the current system we need to consider many options. Perhaps the physics of the environment needs to be richer, to enable the success of novel methods for sensing different aspects of it. Without this richness it seems likely that a simple form of replicator will outcompete all more complex competitors, since the complexity of the organisms reflects the complexity of the environment [1]. The exploitation of different niches is a key factor in achieving diversity [5] and it seems likely that by making these niches alterable by the organisms (perhaps through waste products or other side-effects [21]) the population as a whole will be forced to continually adapt and innovate. However, these are speculations and it is only when the evolutionary growth of complexity [13] is achieved in a computational system that we will be able to test these hypotheses.

The source code for the system presented here is available at:
`http://www.sq3.org.uk/Evolution/`

References

1. C. Adami, C. Ofria, and T.C. Collier. Evolution of biological complexity. *Proc. Nat. Acad. Sci. USA*, 97(9):4463–4468, 2000.
2. C. Adami and C.O. Wilke. Experiments in digital evolution (Editor's introduction to the special issue). *Artificial Life*, 10(2):117–122, 2004.
3. M.A. Bedau, J.S. McCaskill, N.H. Packard, S. Rasmussen, C. Adami, D.G. Green, T. Ikegami, K. Kaneko, and T.S Ray. Open problems in artificial life. *Artificial Life*, 6(4):363–376, 2000.
4. A.G. Cairns-Smith. *Genetic takeover and the mineral origins of life.* Cambridge University Press, Cambridge, 1982.
5. S.S. Chow, C.O. Wilke, C. Ofria, R.E. Lenski, and C. Adami. Adaptive radiation from resource competition in digital organisms. *Science*, 305:84–86, 2004.
6. T.J. Hutton. Evolvable self-replicating molecules in an artificial chemistry. *Artificial Life*, 8(4):341–356, 2002.
7. T.J. Hutton. Information-replicating molecules with programmable enzymes. In *Proc. Sixth International Conference on Humans and Computers*, pages 170–175, University of Aizu, Aizu-Wakamatsu, Japan, 2003.
8. T.J. Hutton. Simulating evolution's first steps. In W. Banzhaf, T. Christaller, P. Dittrich, J.T. Kim, and J. Ziegler, editors, *Proc. Seventh European Conference on Artificial Life*, pages 51–58, Dortmund, Germany, 2003.

9. T.J. Hutton. A functional self-reproducing cell in a two-dimensional artificial chemistry. In J. Pollack, M. Bedau, P. Husbands, T. Ikegami, and R.A. Watson, editors, *Proc. Ninth International Conference on Artificial Life*, pages 444–449, 2004.
10. R.E. Lenski, C. Ofria, R.T. Pennock, and C. Adami. The evolutionary origin of complex features. *Nature*, 423:139–144, 2003.
11. J. Maynard-Smith and E. Szathmáry. *The major transitions in evolution.* Morgan-Freeman, 1995.
12. B. McMullin. *Artificial Knowledge: An Evolutionary Approach.* PhD thesis, Ollscoil na h'Eireann, The National University of Ireland, University College Dublin, Department of Computer Science, 1992.
13. B. McMullin. John von Neumann and the evolutionary growth of complexity: Looking backwards, looking forwards... *Artificial Life*, 6(4):347–361, 2000.
14. B. McMullin, T. Taylor, and A. von Kamp. Who needs genomes? In *Proc. Atlantic Symposium on Computational Biology, Genome Information Systems & Technology (CBGI)*, 2001.
15. C. Ofria and C.O. Wilke. Avida: a software platform for research in computational evolutionary biology. *Artificial Life*, 10(2):191–229, 2004.
16. M. Sipper. Fifty years of research on self-replication: an overview. *Artificial Life*, 4:237–257, 1998.
17. H. Suzuki, N. Ono, and K. Yuta. Several necessary conditions for the evolution of complex forms of life in an artificial environment. *Artificial Life*, 9(2):537–558, 2003.
18. T. Taylor. *From artificial evolution to artificial life.* PhD thesis, Division of Informatics, University of Edinburgh, 1999.
19. T. Taylor. On self-reproduction and evolvability. In D. Floreano, J-D. Nicoud, and F. Mondala, editors, *Proc. Fifth European Conference on Artificial Life.* Springer, 1999.
20. T. Taylor. Creativity in evolution: Individuals, interactions and environment. In P. Bentley and D. Corne, editors, *Creative Evolutionary Systems*, chapter 1. Morgan Kaufman, 2001.
21. T. Taylor. Niche construction and the evolution of complexity. In J. Pollack, M. Bedau, P. Husbands, T. Ikegami, and R. Watson, editors, *Ninth International Conference on the Simulation and Synthesis of Living Systems*, pages 375–380. MIT Press, 2004.
22. J. von Neumann. *Theory of self-reproducing automata.* Urbana: University of Illinois Press, 1966. Edited and completed by Arthur Burks.