

The Infiniteness of Open Ended Evolution

Peter Andras¹

¹School of Computing and Mathematics
Keele University
p.andras@keele.ac.uk

Abstract

Biological and social systems are considered as a flux of interacting components that transiently participate in interactions with other system components as part of the system. This suggests that any simulated system undergoing open ended evolution should be considered in the context of a variable environment that provides elements for the system and acts as a sink for discarded elements of the system. It is argued that any such system can be seen as a computational system that tries to predict its own environment. Furthermore it is argued that such systems must have an infinite representation equivalent of Turing machines. Consequently the level of infiniteness of such systems must be equal to the infiniteness of the category of pre-orders (i.e. the category that is a smallest representation of Turing machines). Finally, the reconciliation of the practical finiteness and principial infiniteness of these systems and implications of the approach taken here are discussed.

Extended Abstract

Biological and social systems undergo long-term evolution and lead to a large variety of evolved adapted systems. These systems exist in the context of variable environment and they can be seen as a flux of components that originate from the environment, end up in the environment, and transiently participate in interactions with other system components as part of the system (Alberts et al, 2008; Luhmann, 1996).

This suggests that open ended evolution simulations should be considered in the context of a variable environment, where individuals must pick resources from the environment and discard their waste into their environment. The resources are used to maintain the existence of the individual. The environment for any individual is made of the other individuals and possibly other environmental elements, some of which may be used as resources.

An individual in natural systems is made of many components that interact and through patterns of such interactions deliver the behaviors of the individual. For example, proteins and other molecules interact in cells, cells interact in organisms, and organisms interact in social systems. The existence of an individual can be seen as the maintenance of these interactions between components in accordance with individual specific rules about what interactions and patterns of interactions are useful and what are not useful for the individual. For example, a cell may pick up a range of molecules from its environment, some of which

are useful (e.g. sugar) and some of which are not useful (e.g. penicillin for a bacterium). Somehow the cell must be able to select those molecular interactions that are useful for it (e.g. the use of sugar to generate energy storing molecules) and eliminate that ones that are harmful (e.g. by breaking up penicillin molecules before they can block the activity of their target proteins) (Blair et al 2015).

This implies that in order to simulate real-like open ended evolution the simulation of the behavior of individuals in terms of interactions between components of the individuals is important. Furthermore, the existence of individuals should depend on the production of such behaviors, i.e. the existence of individuals should be checked by considering their behaviors and assessing whether these behaviors are consistent with the existence of the individual.

The above reasoning implies that biological and social systems somehow through their behaviors decide which elements of the environment are appropriate and which are not for the maintenance of their existence. For example, cells operate the right kind of channel molecules to pick up the right kind of molecules and ions from their environment. When the environment changes and one nutrient (e.g. an amino-acid) is replaced by another as abundant resource, the cells adapt and change their molecular pick-up behavior in order to use the available resource (Hottes et al 2013).

To be able to adapt to environmental changes these systems must in some way predict their environment. For example, the cell that is ready to pick up one kind of nutrient from its environment experiences that the expected nutrient is not available and it also experiences other molecular interactions that may indicate the presence of an alternative nutrient. In response the cell changes the molecules exposed on its membrane that are in charge of facilitating the pick-up of nutrients. Such changes in the molecular composition of the cell membrane effectively predict what the cell expects to find in its environment.

Thus in a sense the natural systems compute their expectations about their environment. This computation happens through the interactions of their components. This means that in order to maintain their own existence these systems behave as autocatalytic systems that catalyze the reproduction (or recruitment) of the correct components and interactions within themselves and they do this through a self-referential computation aimed to predict their environment (Andras, 2011).

The self-referential computation requires providing references to past interactions and components. This can be satisfied if all patterns of interactions (reference-able computations) can be represented by component (e.g. molecules that formed through corresponding molecular interactions), and if all patterns of components (reference-able data) can be represented by ongoing component interactions (e.g. molecular interactions which can happen only if the referenced pattern of molecules was present earlier). This circular referencing may appear irresolvable; however there is a mathematical formalism that can provide a solution, which is the theory of recursive domain equations (Pierce, 1991).

To put this more abstractly, systems that can produce open ended evolution in nature must provide a practical realization of a solution of the following recursive domain equation:

$$R \cong A + [R \rightarrow R] \quad (1)$$

where R is a domain (e.g. a set or a category), A is a part of R and $[R \rightarrow R]$ are all transformations of R to R , i.e. functions from R to R . In this formalism the patterns of components that are represented by A are not part of the system, but these exist only in the environment of the system.

The simplest non-empty solution of the recursive domain equation is the category of pre-orders, which is also a model of the λ -calculus (Pierce, 1991). This indicates that any natural system that is able to do sustain open ended evolution is such that it constitutes a representation of a solution of equation (1) and consequently it is also equivalent with a representation of the λ -calculus or equivalently of Turing machines. This means that these systems can (at least in principle) compute anything computable and predict their environment as much as it is predictable.

However, given the constraint of finite time available for computations about the environment these systems approximate in practice the prediction of their environment. Their ability to approximate their environment precisely depends on how efficient they are in terms of implementation of the solution of the recursive domain equation.

Thus in principle any system that aims to simulate the open ended evolution that can be observed in nature must be able to produce a representation of a non-empty solution of equation (1). This means that these systems must be at least of the size of the category of pre-orders, which is comparably infinite as the category of sets, which is more infinite than the set of real numbers.

In order to deal with apparent difficulty of representing so infinite systems let us consider first real systems, such as cells or social systems. While the above argument implies that the cell must be an infinite system, the reality is that it is made of a finite set of molecules and molecular interactions at any time. To accommodate the infiniteness requirement, let us consider the life of a bacterial cell. The cell emerges after a division of another bacterial cell and it lasts until its own division into daughter cells. However, if we consider that the cell is in fact the continuation of the parent cell and its daughter cells are continuations of the cell itself, and take the whole life trajectory of the continuations of the cell, both backward and forward, we find that we are dealing with an infinite system. The number of kinds of molecules involved in cells is also similarly infinite, given the possible variations of molecules (e.g. consider the huge DNA molecules). So, while

a given cell at any time provides a finite snapshot of the infinite cell system, considering the cell system in its totality, the system is indeed infinite (Andras, 2011).

This means that in order to be able to simulate a real-like system with open ended evolution the system must be able to extend infinitely in principle and the simulation at any time should provide a finite snapshot of the infinite system, which represents a solution of the recursive domain equation. The key aspect is that the system must be extendable infinitely into a representation of the λ -calculus or equivalently of Turing machines.

Turning this around, if we consider a system that explicitly or implicitly implements λ -calculus then we should be able to use this system to produce a simulation of open ended evolution. For example, consider a set of Turing machines that operate on their own input strings, and an environment made of random input strings of random lengths. Producing certain outputs in their stack string (i.e. patterns of symbols) could be required for the survival of individuals. The individuals could crossover their input strings and pick up an appropriate input string from the environment and merge this according to some pattern with their existing input string (the picked up input string could be a part of the input string of another individual). The appropriate input string from the environment could be determined by a string match with a part of the contents of the stack string. Distinguishing between different variants of the Turing machines might be complicated, but according to the reasoning above the result should be a simulation of open ended evolution if the various rules of the system are appropriately set. An alternative, more visual example could be the consideration of Wolfram's rule 110 for one-dimensional cellular automata (Wolfram, 2002) in combination with random input strings of black and white squares. Again production of certain patterns would be required for survival and strings of black and white squares would constitute the environment from where automata individuals could pick up new parts for their input strings. Individuals could combine their input strings with strings taken from the environment. The generation of offspring would involve crossover of input strings of parents and mutation of the input string of the new individual. Given that rule 110 implements a Turing machine it is expected that this system would also lead to a simulation of open ended evolution (Wolfram, 2002). However, again the determination of species identity of individuals might be complicated (although some form of string matching algorithm similar to BLAST might work well).

References

- Alberts, B. et al (eds.) (2008). *Molecular Biology of the Cell*. Garland Science, 5th ed.
- Andras, P. (2011). Modelling living systems. *Proceedings of ECAL 2009*, Springer.
- Blair, J.M.A. et al, (2015). Molecular mechanisms of antibiotic resistance, *Nature Reviews*, 13:42-51.
- Hottes, A.K. et al (2013). Bacterial Adaptation through Loss of Function. *PLoS Genetics*, 9(7): e1003617.
- Luhmann, N (1996). *Social Systems*. Stanford University Press.
- Pierce, B.J. (1991). *Basic Category Theory for Computer Scientists*. MIT Press.
- Wolfram, S. (2002). *A New Kind of Science*. Wolfram Media.