Neutral Emergence and Coarse Graining

Andrew Weeks, Susan Stepney, Fiona Polack

Department of Computer Science, University of York, UK

Abstract. We introduce the concept of *neutral emergence* (defined by analogy to an information theoretic view of neutral evolution), and discuss how it might be used in the engineering of *robust* emergent systems. We describe preliminary results from an application to coarse graining of cellular automata.

1 Introduction

We introduce the concept of *neutral emergence*, defined by analogy to an information theoretic view of neutral evolution. Our particular interest in neutral emergence is as a potential component of an engineering framework for emergent systems. The key problems for engineering emergence are to be able to implement a system such that a specified property emerges, and to make the property robust, at least to minor unforeseen perturbations. Our initial work is improving our understanding of emergence, and of the "slack", or potential for robustness, between an emergent system and its implementation.

Section 2 summarises background material on neutral evolution, and its formulation in information theoretic terms. Section 3 summarises background material on emergence. In section 4 we cast emergence in information theoretic terms, by analogy to information theoretic evolution. In section 5 we introduce the concept of *neutral emergence*, and show how it is related to system robustness. In section 6 we describe preliminary results from an application to coarse graining of cellular automata, and outline our plans for future work.

2 Background: evolution

Neutral evolution. The mapping from genotype (DNA) to phenotype (organism) is complex, and there is significant redundancy in both. Different genotypes can map to the same phenotype; for example, different codons (DNA nucleotide triplets) can code for the same amino acid. Hence the genotype can change (a nucleotide can mutate) without changing the phenotype. Similarly, the same genotype can result in different phenotypes, due to different environmental conditions during development.

Neutral evolution is a recognised phenomenon in evolutionary biology [15]. It involves a change in the genotype without a significant change in the fitness of the phenotype; that is, it involves a change that is selectively neutral. Such changes allow the phenotypic population to explore their fitness landscape, by drifting

around close to contours of equal fitness, to regions where they may subsequently find fitter solutions not directly accessible from their original position in the landscape.

Information theoretic evolution. Adami [1–3] gives a description of evolution in information-theoretic terms, by considering the information in the genome *in the context of its environment.*

Adami [1] classifies a (digital) genome into 'hot' and 'cold' bits. The cold bits are ones strongly conserved in the population, whereas the hot bits vary between population members, and are therefore (presumably) not significantly contributing to the fitness. So, in general, mutations of hot bits are *neutral*, but occasionally might be beneficial, at which point they become 'frozen in' to the population, and conserved. These fitter organisms are exploiting their environment better, and so must contain more information about those parts of the environment that they are exploiting [1, p115].

In information theoretic evolution, the genome is seen as some kind of representation of the environment: the cold bits are strongly correlated with the environment, whereas the hot bits are not. The more correlated bits there are, the higher the *mutual information* between the organism's genome and the environment. The *mutual information*, or correlation, between the system S and its environment E, $I(S : E)$, is the *entropy* of the system, $H(S)$, less the *conditional entropy* of the system in the context of the environment, $H(S|E)$:

$$I(S : E) = H(S) - H(S|E) \tag{1}$$

The conditional entropy $H(S|E)$ can be thought of as the amount of information in the system that cannot be explained by (correlations with) its environment. Similarly, the conditional entropy $H(E|S)$ can be thought of as the amount of information in the environment that cannot be explained by correlations with the system.

Evolution (increasing fitness) is then *increasing mutual information*: increasing the shared information, or correlations, between the genome and the environment. So "*natural selection can be viewed as a filter . . . that lets information flow into the genome, but prevents it from flowing out*" [2]. See figure 1. This simple description captures the essence; in reality the biological process is much more complicated. For example: bits are correlated within a genome [2], so when some bits go cold it is necessary for other bits to become hot again. Also, the environment co-evolves with the organisms. However, a fuller picture does not undermine the principle of an information-theoretic approach, particularly when the principle is transferred from the biological to an engineering domain.

Neutral evolution in information theoretic terms. The mutual information, I, is independent of the hot bits in S; it depends only on the cold bits correlated with the environment E. Hence, in this context, neutral evolution is a change in the genome that does not affect the mutual information of the genome
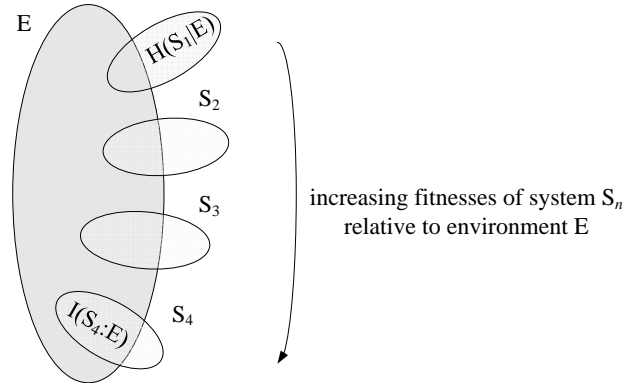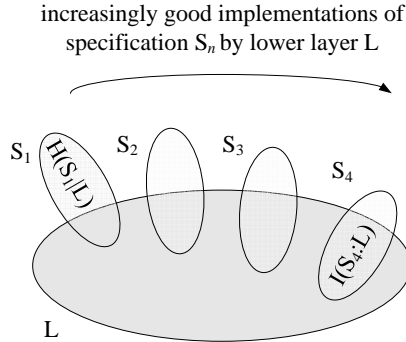
Fig. 1. Evolution increasing the mutual information between the genome S and the environment E (adapted from [1, fig.5.7]). The area of an ellipse represents its information content, and the area of overlap represents the mutual information between an S_n and E. For simplicity, total information is shown as unchanging (the areas of ellipses S_n and E are constant); this implies that conditional information decreases as the mutual information increases. This is not necessarily the case: for example, a neutral evolutionary step could change the amount of conditional information $H(S|E)$ by increasing the size of the uncorrelated part of the genome [2].

and (the fitness relevant part of) the environment, $I(S:E)$. This means that S is *robust* to this kind of genomic change. Similarly, parts of the environment E that are not correlated with S can also change neutrally (from the point of view of S), so S is *robust* to this kind of environmental change, too.

3   Background: Emergence

In [16] we provide a review of opinion on emergence. We start from Ronald *et al*'s definition of emergence: "*The language of design L_1 and the language of observation L_2 are distinct, and the causal link between the elementary interactions programmed in L_1 and the behaviors observed in L_2 is* non-obvious *to the observer—who therefore experiences surprise.*"[13]. We reject the use of 'surprise' as a criterion for emergence: it is subjective, and wears out with repetition. We do, however, follow Ronald *et al*'s use of two languages of description, which we refer to as L for the 'microscopic' *local* level of implementation substrate, and S for the 'macroscopic' global level of the model or *specification* (we are interested in observing and modelling natural emergent systems, but more in specifying and implementing engineered ones). We follow Shalizi [14] in defining emergence in information-theoretic terms, as the greater 'predictive efficiency' of descriptions in S over those in L. Following Bar-Yam [4], we see emergence as a consequence of global constraints on the system (the emergent level) rather than of constraints on components (the implementation level).

Fig. 2. Modelling, or incremental system development, as increasing the mutual information between the system specification S and the implementation L.

In relation to levels, various research identifies 'natural' scales. Israeli and Goldenfeld [7] note that there is an emergent natural length scale for coarse graining (see section 6.1) cellular automata (CA). In the process of "reconstructing the attractor" from time-lagged observations [17] (also a form of coarse-graining), the correct time lag can be found using mutual information [12].

To summarise our view, in an emergent system, the whole is greater than, and different from, the sum of its parts (in terms of behaviour and other qualities); emergence can, in various senses, be equated to novelty; levels are essential: emergence occurs at a higher level or longer time scale than that on which basic processes occur; emergence is a characteristic that is expressed at the high level, caused by the low level. Emergence is not an arbitrary feature; it is not dependent merely on the chosen level of observation. Some levels are 'better' than others (for example, coarse grainings that better describing the underlying dynamics), and form natural levels for discovering, or for designing in, emergent properties.

4 Emergence in information theoretic terms

By direct analogy to the information theoretic description of evolution, we define an information theoretic quantity of emergence.

Definition: The amount of emergence $I(S : L)$ is the mutual information, or correlation, between a specified system S and its implementation in substrate level L.

$$I(S : L) = H(S) - H(S|L) \qquad (2)$$

The conditional entropy $H(S|L)$ can be thought of as the amount of information in the system specification that has not been captured by (correlations with) its implementation.

Modelling, or incremental system development, can be viewed as increasing mutual information: increasing the shared information, or correlations, between the system specification and its implementation (figure 2).

If one were trying to explain (model) an observed system S in terms of L, the conditional information $H(S|L)$ might be the behaviours or properties of S not explained by L. (It might just be noise.) If one were trying to implement a specified system S in an implementation substrate L, this conditional information is the part of the specification that has yet to be captured by the proposed implementation L: more development work is required.

The conditional information $H(L|S)$ (the information in the implementation not correlated with the system specification) might be considered as properties of L that are unnecessary for the realisation of S. Of course, these additional properties might well be *invisible* if the system is subject only to high-level observation [6] that reveals only system-level properties (multiple distinct microstates nevertheless resulting in the same observed macrostate). It is, however, these extra properties that can be exploited to get *robust* implementations of S.

The information theoretic definition of emergence suggests an approach to incremental development of emergent systems: use the mutual information I as a fitness function in a search for good models (system descriptions) S of an existing L (such as a given CA or agent system), or to search for good low level implementations L (such as the required rules and states for a CA or agent system) of a system specification S.

5 Neutral Emergence

Definition: An emergent property exhibits *neutral emergence* when a change in the microstate L does not change the macrostate S, or vice versa. It is a change that does not significantly change the mutual information $I(S : L)$.

Robustness. A system S is robust to changes in either itself, or in L, if the changes do not significantly change the mutual information. In particular, S can be robust to many changes in its implementation, including, possibly, the effect of errors. It is often stated that emergent systems (modelled on natural processes) exhibit robustness: here we see why (and where) this may be the case. The excess information in L (a large $H(L|S)$) is *necessary* for emergent systems to be robust in this manner.

As argued earlier, an engineering development process can be seen as implementing specification S by finding an L with a high mutual information $I(S : L)$. Here we see that, at the same time, the process can seek to maximise robustness, by searching for a system that is insensitive to (uncorrelated with) certain failure modes or other possible changes in L. If a system were stressed during development (exposed to a range of stresses and implementation errors), its implementation could be encouraged towards regions that are insensitive (robust) to such events. (Compare this to the development of formally proven systems: they do not guarantee any level of performance with even the smallest change.) By analogy to evolutionary fitness landscapes, we want to find systems that lie in gently sloping plains and plateaux, rather than on narrow peaks or steep cliffs.
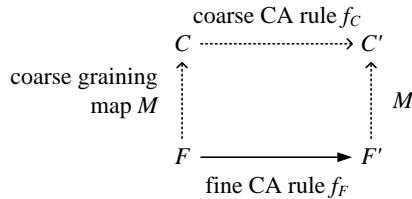
Fig. 3. Exact coarse graining. The fine grained state F evolves under the given CA rule f_F. The coarse graining M maps fine grained state F to coarse grained state C. The coarse grained state C evolves under another CA rule f_C. The task is to find a mapping M and rule f_C such that the square commutes: $M \circ f_F = f_C \circ M$.

Speculation on 'the edge of emergence'. Experiments with CAs [10] and Random Boolean Networks [8, 9] indicate that a phase transition takes place between a 'solid' ordered regime and a 'gaseous' chaotic state, and that a narrow 'liquid' region on the boundary between these regimes is capable of supporting the most complex behaviour, including universal computation. (Note: the detail of Langton's 'edge-of-chaos' results [10] have been questioned [11].)

Compare this with our mutual information measure of emergence: a system exhibits minimal emergence when everything is a surprise (zero mutual information). Clearly such a model, that knows nothing about what it is modelling, is useless, but equally (as argued above) useful emergence does not require maximal mutual information in the system. There should be a level of emergence with the maximum utility: a position at which the model has useful freedom to explore, whilst being held within a constrained region of the search space (maintaining the quantity of mutual information; analogous to the 'fitness contouring' of neutral evolution). While it is tempting to label this the 'emergence liquid region', substantive experimental data is needed to support such a claim.

That said, consideration of the 'edge-of-chaos' analogy raises some interesting questions. Does a phase transition occur between the 'solid' (system specification or model effectively duplicating the implementation) and 'gaseous' (little in common) regimes? Is the region a narrow one? If so, is this why creating effective emergent systems manually has proved difficult?

6   Application

Thus far, our discussion has been conceptual. Now we present preliminary work applying our ideas to CAs.

6.1   Coarse graining cellular automata

Israeli and Goldenfeld [7] coarse grain CAs. Their *exact* coarse graining captures the dynamics of a CA rule with another rule, expressed over a coarser grain of space and time. (See figure 3.) Exact coarse graining preserves the underlying dynamics but loses 'irrelevant' detail.

C_1	1	1	1	1	1	0	...	0	0	0	...	1	1	1
C_2	.	1	.	.	0	.	...	.	1	.	...	.	1	.

F_1	00	00	00	00	00	01	...	01	01	01	...	11	11	11
F_2	.0	00	0.	.0	00	1.	...	.0	00	0.	...	.0	00	0.
F_3	..	00	..	..	01	..	...	..	00	..	...	..	00	..

Fig. 4. Search setup for exact coarse graining.

In terms of emergence, the two languages L and S are *different* CA rules (except in a few cases), and information is lost in the coarse graining (in that low-level fine-grained detail is washed out). Interestingly, the figures illustrating coarse grainings in [7] seem to highlight some of the underlying L structure (for example, various propagating 'signals'), maybe because they have smoothed out other, irrelevant, structure.

The higher level rule S may be a CA rule of lower complexity than the rule at level L: because "*the system (the update rule, not the cell lattice) does not contain enough information to be complex at large scales*" [7]. This complexity-reducing kind of coarse graining (of 'relevant degrees of freedom') removes information that is relevant to the dynamics at L, but is redundant under the coarse graining at S: it does not capture all of the microscopic dynamics.

6.2 Applying neutral emergence to coarse graining

In investigating neutral emergence, we start by recasting Israeli and Goldenfeld's [7] coarse-graining results in the information theoretic emergent framework described above. We then consider an extension of coarse graining that has greater robustness.

Robustness. The exact coarse graining of [7] exhibits robustness in the way described earlier. In particular (i) different low-level CA rules coarse grain to the same high-level rule, so the high level is robust to certain changes in the low-level rule; (ii) even for a fixed low-level rule, different low-level CA states coarse grain to the same high-level state, so the high level is robust to certain changes in the low-level state.

Exact coarse graining. Israeli and Goldenfeld [7] use a brute force approach to discover their coarse grainings. For exact coarse graining, the diagram in figure 3 must commute for all fine states F.

As a control, we implemented this brute force search for 1D CA coarse grainings with $N = 2$ (that is, one high level cell corresponds to 2 low level cells) as follows (figure 4). For $N = 2$, there are 16 candidate mappings M, with fine states states 00, 01, 10, 11 mapping to 0 or 1. 14 of these are non-trivial (target state contains both a 0 and a 1). In figure 4, we start from a specially constructed

$F_1 = 00\ 00\ 00\ \ 00\ 00\ 01\ \dots 11\ 11\ 11$ that ensures all coarse neighbourhoods occur for non-trivial candidate mappings. We apply the fine rule twice, generating state $F_3 = f_F\, f_F\, F_1$. A candidate mapping M is then used to construct coarse states $C_1 = M\, F_1$ and $C_2 = M\, F_3$. Because C_1 is a mapping on the specially-constructed F_1, it contains all possible coarse neighbourhoods.

To determine whether candidate mapping M is a valid coarse graining, we must ensure that whenever a neighbourhood occurs more than once in C_1 it maps consistently to the same central state in C_2. Since C_1 contains all coarse neighbourhoods, the coarse rule f_C can be deduced from the constructed C_1 and C_2 states.

Mutual information as a fitness function. As an alternative to brute-force search, a genetic algorithm (GA) was used to search for 1D CA coarse grainings with $N = 2$. Each member of the GA population has a random candidate mapping M. The coarse graining process described above is followed to generate states C_1 and C_2. We use an approximation to mutual information as the fitness function of the map M: the fitness maximises the number of consistent states between C_1 and C_2.

This GA successfully finds 1D exact coarse grainings (although not on every run), supporting our assertion (section 4) that mutual information would be an appropriate fitness function. The trial on exact coarse graining is used to validate the GA search, but in this case the GA is much less efficient brute force search. However, as the coarse-graining size increases (for example, $N = 5$), brute force search becomes computationally intractable (the number of states to search increases exponentially), and GA search becomes attractive.

Approximate coarse graining. Israeli and Goldenfeld's [7] coarse grainings provide an exact fit over *all possible states*: they lose information, but the systems never differ in their predictions. Exact coarse graining is a form of information *compression*. The mutual information is the same as the entropy of the high level system.

We can search for *approximate coarse grainings*, requiring an exact fit over only *some* CA states, thereby allowing the information content of those states to be exploited, in addition to the information content of the CA rules themselves. (Compare the case in physical emergent systems, where the emergent properties occur only over some restricted set of all possible states, such as a restricted temperature range.)

In approximate coarse graining, we start with a much smaller state F_1 that need not ensure all coarse neighbourhoods occur (figure 5). We build up C_1 and C_2 as before, but because we start from a partial state, C_1 may not cover all possible coarse neighbourhoods.

When we check for consistency of the candidate mapping M, we derive all the neighbourhoods contained in C_1, including the overlapping ones (excluded from the exact coarse graining). Thus, in figure 5, $C_1 = 1\ 0\ 1\ 1\ 1\ 1$ yields neighbourhoods $1\ 0\ 1$, $0\ 1\ 1$, $1\ 1\ 1$, etc. This fully exploits the partial information. If the

C_1	1	0	1	1	1	1
C_2	0	0	1	1	1	1

F_1	00	01	10	11	10	00
F_2	00	10	00	00	01	00
F_3	01	01	00	00	10	10

Fig. 5. Search setup for approximate coarse graining.

mapping is consistent for this partial information, we generate the appropriate set of coarse rules.

The brute force approach was used to search for 1D approximate coarse grainings. Even when starting from very small F_1, we discover good approximate coarse graining rules. Each solution set contains the exact rule discovered by [7]. In many cases, additional rules are found that, superficially at least, look like the exact rule. We are now investigating how the additional rules depend on the chosen F_1.

6.3 Next Steps

To continue our exploration, we propose to use the information theoretic approach to guide our choice of search over interesting states: low complexity patterns in the low level CA (for example, gliders in Conway's Game of Life CA [5]). Thus our coarse grainings will be optimised for such patterns, rather than for 'random' behaviour.

This approach of using patterns to guide the coarse graining rule search requires a way of finding such patterns. We will use a coarse graining approach here, too: coarse graining states over space and time to detect repeating behaviour, and hence stationary and moving patterns. The size of these coarse grained cells limits the size of patterns detectable by this approach.

This combination of object detection and CA prediction [7], when combined, should permit object prediction – an important aspect for reliable engineering.

Our next step is to test our proposals by combine object detection and CA prediction on the Game of Life CA, to investigate how the quality of prediction is related to the quality of rule coarse graining, as a function of the patterns used to guide the GA coarse graining search.

Eventually, we want to examine emergence in the context of an environment (combining our ideas of evolution and emergence), both for defining emergent properties, and for achieving robustness of that emergence.

7 Conclusions

We have cast emergence in information theoretic terms, by direct analogy to evolutionary processes, and have shown how this can be used to derive a fitness function for developing models of emergent systems, or emergent implementations

of system specifications. We have introduced the concept of neutral emergence, and shown its relationship to robust emergence. As an initial demonstration, we have applied these concepts to coarse graining CAs.

Our eventual aim is to incorporate neutral emergence as a component in a robust emergent engineering process.

Acknowledgements

References

1. Christoph Adami. *Introduction to Artificial Life*. Springer, 1998.
2. Christoph Adami. What is complexity? *BioEssays*, 24:1085–1094, 2002.
3. Christoph Adami and N. J. Cerf. Physical complexity of symbolic sequences. *Physica D*, 137:62–69, 2000.
4. Yaneer Bar-Yam. A mathematical theory of strong emergence using multiscale variety. *Complexity*, 9(6):15–24, 2004.
5. E. R. Berlekamp, J. H. Conway, and R. K. Guy. *Winning Ways for your Mathematical Plays*. Academic Press, 1982.
6. J. A. Clark, S. Stepney, and H. Chivers. Breaking the model: finalisation and a taxonomy of security attacks. *ENTCS*, 137(2):225–242, 2005.
7. Navot Israeli and Nigel Goldenfeld. Coarse-graining of cellular automata, emergence, and the predictability of complex systems. *Phys. Rev. E*, 73:026203, 2006.
8. Stuart A. Kauffman. *The Origins of Order: self-organization and selection in evolution*. Oxford University Press, 1993.
9. Stuart A. Kauffman. *At Home in the Universe*. Oxford University Press, 1995.
10. C. G. Langton. *Computation at the Edge of Chaos: Phase-Transitions and Emergent Computation*. PhD thesis, University of Michigan, 1991.
11. M. Mitchell, J. P. Crutchfield, and P. T. Hraber. Dynamics, computation, and the 'edge of chaos': a re-examination. In G. A. Cowan et al., editors, *Complexity: Metaphors, Models, and Reality*, pages 497–513. Addison Wesley, 1994.
12. Asok Ray. Symbolic dynamic analysis of complex systems for anomoly detection. *Signal Processing*, 84:1115–1130, 2004.
13. E. M. A. Ronald, M. Sipper, and M. S. Capcarrère. Testing for emergence in artificial life. In D. Floreano et al., editors, *Advances in Artificial Life: 5th European Conference*, volume 1674 of *LNCS*, pages 13–20. Springer, 1999.
14. C. R. Shalizi. *Causal architecture, complexity and self-organization in time series and cellular automata*. PhD thesis, University of Wisconsin at Madison, 2001.
15. S. C. Stearns and R. F. Hoekstra. *Evolution: an introduction*. OUP, 2000.
16. Susan Stepney, Fiona Polack, and Heather Turner. Engineering emergence. In *ICECCS 2006*. IEEE, 2006.
17. F. Takens. Detecting strange attractors in turbulence. In D. A. Rand and L.-S. Young, editors, *Dynamical Systems and Turbulence*, pages 230–242. Springer, 1981.