

The fallacy of general purpose bio-inspired computing

Seth Bullock

School of Electronics and Computer Science

University of Southampton, Southampton, SO17 1BJ, UK

Tel: +44 (0)2380 595 776

Fax: +44 (0)2380 594 498

sgb@ecs.soton.ac.uk

Abstract

Bio-inspired computing comes in many flavours, inspired by biological systems from which salient features and/or organisational principles have been idealised and abstracted. These bio-inspired schemes have sometimes been demonstrated to be general purpose; able to approximate arbitrary dynamics, encode arbitrary structures, or even carry out universal computation. The generality of these abilities is typically (although often implicitly) reasoned to be an attractive and worthwhile trait. Here, it is argued that such reasoning is fallacious. Natural systems are *nichiversal* rather than universal, and we should expect the computational systems that they inspire to be similarly limited in their *performance*, even if they are ultimately *capable* of generality in their *competence*. Practical and methodological implications of this position for the use of bio-inspired computing within artificial life are outlined.

Within the context of computing, we routinely employ *metaphors* derived from nature: bugs, trees, inheritance, killing, spawning, etc. In terms of novel computational *architectures*, biology has also been a rich source of ideas. Artificial neural networks, swarm intelligence, artificial immune systems, etc., have all arisen as a result of our growing appreciation of the sophisticated “computational” abilities of some biological systems.

The attractiveness of these approaches is compounded by the possibility that they are, in some sense, *general purpose*. For instance, a common class of continuous-time recurrent artificial neural network can be shown to approximate any dynamical system to an arbitrary degree of accuracy (Funahashi & Nakamura, 1993); a swarm of artificial insects can implement a general-purpose optimisation algorithm (Bonabeau, Dorigo, & Théraulaz, 2000a); a cellular automaton is capable of universal computation (Cook, 2004); wasp and termite behaviour might be idealised to deliver general-purpose construction algorithms for self-organising architectures

(Bonabeau, Guérin, Snyers, Kuntz, & Théraulaz, 2000b; Howsman, O’Neil, & Craft, 2004); artificial immune systems could offer the ability to efficiently classify arbitrary classes of patterns (e.g., Tarakanov, Skormin, & Sokolova, 2003).

The generality of these idealised, bio-inspired systems suggests that they might enjoy very wide applicability. However, *real* biological systems are only ever general-purpose *accidentally*. No biological species, organism, organ, trait or mechanism has ever evolved to serve the function of solving a class of problems that is wider than the set of problems actually encountered by its ancestors so far. It is true that some are more or less specialised than others, but natural selection is not in the business of fashioning devices that solve future problems or potential problems, only actual historical ones.

For instance, the behavioural mechanisms that termites use to construct their amazing mounds (Bruinsma, 1979) were not evolved for construction, *per se*, but for constructing termite mounds, specifically (Ladley & Bullock, 2004, 2005). Our immune system has not evolved to classify arbitrary patterns, but to deal with the particular kinds of pathogen to which we have historically been exposed. Even the human brain, indisputably the most awesome problem-solving mechanism that we know of, is not a general purpose cognitive machine. It is specialised to undertake particular cognitive tasks (language learning, face recognition, social cognition, etc.). It is not organised to solve any problem or deal with every cognitive challenge (witness the large literature on our cognitive shortcomings, e.g., Kahnemann, Slovic, & Tversky, 1982). Rather our brain exhibits properties that allow it to successfully tackle the reproductively significant cognitive problems that faced our evolutionary ancestors on the African savannah.

This is not to say that biological mechanisms do not generalise at all. Biological niches are identified with problem classes (digesting a particular *kind* of nut, warning off a particular *species* of predator) rather than just problem instances (digesting a particular nut, warning a particular predator), because evolution cannot tailor mechanisms to individual problem instances. There

must be some exploitable regularity in the instances of a repeatedly encountered problem for them to count as repetitions of the same problem, let alone be solved by a single evolved mechanism. Our immune systems, for instance, have been adapted by evolution such that they are able to cope with many pathogens, some yet to be encountered by our species. We might be tempted to say that our immune system has some predictive ability whenever it copes with a novel pathogen. However, this “predictive ability” is little more than a gamble that the future will resemble the past in particular ways.

Stated more generally, biological devices are shaped by natural selection such that they tend to be well suited to the challenges posed by their Environment of Evolutionary Adaptedness (or EEA, see Foley, 1997). This “environment” is actually the sum total of the selection pressures that have been brought to bear on a device’s lineage (weighted by recency). It is the finite set of reproductive problems that a particular contemporary biological device’s ancestors solved in order that this device (rather than competing forms) currently exists. The EEA is thus similar to the notion of a biological *niche*, in that the character of a biological device can be understood as a reflection (or co-definition) of the demands, pressures, and challenges that characterise its niche. From an alternative perspective, one can expect biological devices to function successfully only under Normal conditions: the conditions that the device’s ancestors tended to find themselves in historically (Millikan, 1984, 1993). Outwith such conditions, the performance of an evolved device may be suboptimal, or even pathological (e.g., some forms of human obesity may result from some of our evolved devices operating in a modern environment featuring many abNormal foodstuffs).

This line of argument implies that the biological systems that inspire novel computational architectures, paradigms, or substrates are likely to be well-suited only to *particular* tasks. Even when (idealised abstractions of) these mechanisms are *capable* of exhibiting a very general class of behaviour, we should not expect them to do so *uniformly*—they will tend to be more suited to some tasks than others.

Universal vs. Nichiversal

The apotheosis of the claim that a class of system exhibits computational generality is the demonstration of *universality*. Work on universal computation has had profound consequences for our understanding of computation and computability (e.g., Turing, 1936). However, demonstrations of general-purposeness, completeness, etc., for CTRNNs, swarm algorithms, genetic encodings, etc., are not typically part of this effort to improve our understanding (some work on cellular automata may be an exception, here). Rather, they are driven by the implicit conviction that the generality of a particular

bio-inspired approach is a point in its favour; general schemes, architectures, or algorithms being intrinsically more preferable than specialised ones. If, say, a swarm of artificial ants is demonstrated to simulate a universal Turing machine, the intention is probably to reveal something about the utility of swarm intelligence rather than the nature of computation.

The meaning of the word *universal* derives from parts meaning “all” and “turned towards or against”. Hence, whereas *universal* might be glossed as meaning “all-facing”, biological devices are “niche-facing” or *nichiversal*. The nichiversality of bio-inspired computational schemes is true, independent of whether they are provably general or not. For example, while continuous-time recurrent neural networks are capable of exhibiting *arbitrary* dynamics (given enough nodes), it is still true that certain dynamics are *characteristic* of such networks, i.e., this class of device does exhibit a *generic* behaviour (Beer, 1995). Attempting to find or construct networks that exhibit dynamics very different from this generic behaviour is difficult.

Similarly, even if termite construction behaviours can be idealised such that they are, in theory, capable of generating arbitrary structures (Howsman et al., 2004), it will remain the case that some classes of structure are more readily buildable by such systems. In order to configure such a system to construct architectures that are *uncharacteristic*, one faces a very difficult reverse engineering task that cannot typically be solved by hand and is often even difficult to solve using some kind of powerful search algorithm (see below).

Comparisons with NFL

In one sense this argument boils down to a well-known fact: every tool is good for some things and not so good for others. However, the argument presented here differs significantly from that of No Free Lunch (NFL) theorems (Wolpert & Macready, 1995, 1997).

First, NFL proofs are formal statements derived from first principles. This paper presents an informal argument predicated on our understanding of evolutionary processes. More importantly, the line of argument presented here cannot be ducked in the way that NFL considerations can sometimes seem to be.

NFL theorems state that no search or optimisation algorithm will outperform any other *over the set of all problems*. It therefore remains possible that some candidate algorithm will outperform others over a *subset* of all problems. Since the set of problems that we care about is such a subset, NFL-style arguments may not bite.

This type of reasoning will not defuse the NFL-like implications of nichiversality: that generality claims for bio-inspired schemes are empty. Consider the distinction between *competence* and *performance* introduced by Chomsky (1965). While a system’s competence cor-

responds to the range of (linguistic) behaviour that it could produce *in principle*, a system's performance corresponds to the range of (linguistic) behaviour that it actually produces *in practice*.¹ Employing these terms: it may be true that the relatively limited set of behaviours that we require from a class of bio-inspired system (its required performance) is a sub-set of the entire (possibly complete) set of behaviours that such a class of system is capable of performing (its competence). Nevertheless, unless we require a bio-inspired system to be merely biomimetic (to simply ape the natural activity that inspired it in the first place), we will undoubtedly require its performance to go *beyond* the range of behaviours that were demanded by its niche, i.e., the performance that we *require* of it will outstrip its *Normal* performance.

Hence, nichiversal thinking (and, for different reasons, NFL thinking) encourages us to *characterise* (rather than merely *quantify*) an algorithm's performance, rather than its competence or scope. This type of thinking implies a reconsideration of the typical working methodology of bio-inspired computing researchers. The rest of this paper attempts to spell out these methodological implications via five guidelines and, subsequently, two brief examples.

Guidelines

1. Embrace the nichiversal nature of bio-inspired computation.

First and foremost, we should adopt the working assumption that characterising the limited, task-specific, generic behaviour of a bio-inspired system is what is important, rather than its potential for generality. Characterising the “niche” of a class of bio-inspired system is a challenging, but critically important goal. It is crucial to our ability to make practical use of bio-inspired computing, but is also important to our ability to make theoretical progress in understanding the nature of adaptive behaviour in general.

2. Accept that multiple idealisations of a biological mechanism/organisation/process can coexist.

Within a particular domain of bio-inspired computing, there often appears to be competition between different flavours of system. The evolutionary algorithms literature offers many clear examples. At one level, genetic programming, genetic algorithms, evolutionary strategies, etc., “compete” (with each other and with alternative search and optimisation algorithms) to demonstrate their ability to solve hard optimisation problems. At a lower level, different genetic encodings, genetic operators, selection schemes, multi-population set-ups, etc., also compete to outperform each other.

¹Thanks to Richard Watson for suggesting the use of this terminology.

The importance of specifying the ways in which these different flavours of algorithm relate to one another, or how one might decide between them when attempting to solve a particular problem, is widely acknowledged, but this aim is rarely achieved.

3. Take note of negative results, carefully examined.

In the context of (2), above, one can see why negative results are unpopular: “I've invented a new type of a swarm intelligence algorithm—here's a number of ways in which it is outperformed by existing swarm intelligence algorithms”. Of course, since we understand that no algorithm will outperform *all* others on *all* classes of problem, it is precisely this type of negative result that can be valuable, when carefully analysed. While merely reporting an *instance* of unsuccessful (or indeed successful) performance can have little value, *per se*, even a thoroughly poor class of algorithm that appears to be outperformed by many others can have theoretical value if in analysing the reasons for its mediocrity, we discover insights into wider classes of scheme or algorithm. Zaera, Cliff, and Bruton (1996), for example, present a failed attempt to evolve realistic flocking behaviour as an indicator of what makes constructing a fitness function hard or easy. Unfortunately, this type of research is rarely undertaken and remains difficult to get published when it is.

4. Attend to the limits of natural biological mechanisms/organisations/processes *in situ*.

Biologists cannot completely and accurately characterise a biological mechanism's EEA or its Normal conditions for functioning. However, they often know *something* about the character of a mechanism's niche. In particular, where a mechanism varies across different populations, there is scope for explaining these differences as resulting from the different selection pressures that these populations have been subjected to. This information can be useful in determining what one might expect a bio-inspired approach to be good for. However, gathering it involves serious engagement with the relevant biological community and their literature, which is time-consuming and difficult work.

5. Choose problems appropriate to particular bio-inspired solutions, and vice versa.

A corollary of the observation that bio-inspired algorithms do not equally suit every problem is that only some of our problems may be suitable for bio-inspired solution. We should give as much consideration to tailoring, adapting, rethinking and refining our problems such that they suit the predispositions of our bio-inspired computing, as we do the converse.

Two Brief Examples

Collective Construction

Paper wasps work collectively to build impressive nest structures. Idealising their behaviour has resulted in a simple class of decentralised construction algorithm capable of generating nest-like structures as well as other interesting architectures (Théraulaz & Bonabeau, 1995). The algorithm involves a swarm of reactive agents moving through a 3-dimensional cubic lattice, depositing different kinds of building material. An agent's building behaviour is determined by a set of production rules, each sensitive to a particular configuration of building material in the 26 cells adjacent to the agent's location (a triggering condition). With two types of building block, there are 3^{26} possible building rules, and thus $3^{3^{26}}$ possible algorithms. Moreover, it is straightforward to demonstrate that, given an arbitrary number of block types, any configuration of contiguous building material can be specified. In some sense, then, the algorithm is general purpose and complete.

However, despite this general *competence* of the algorithm, its *performance* is typically extremely constrained (Ladley & Bullock, in preparation). With small numbers of block types, it is very difficult to find, or hand-design, rule sets that give rise to structures with anything other than very fine-grained spatial scale. Long-range structures are practically impossible to generate. This is because, although there are an astronomical number of different triggering conditions, some (a very small proportion) are more privileged than others. Triggering configurations involving, say, 15 blocks of building material can only arise after 15 blocks have been placed. In placing these blocks, many, many triggering conditions involving only one, two, three, or four blocks will necessarily arise. Any rule-set capable of encouraging the placement of 15 blocks must contain rules that are triggered by configurations of small numbers of blocks. As a result, the inherent reflexivity of the scheme's early behaviour ensures that it is predisposed to undermine the utility of the vast majority of more complicated triggering conditions.

In practice, it turns out that it is very difficult for a rule-set to prevent simple configurations of blocks being constructed repeatedly. This ensures that complicated configurations of blocks, or simple structures that arise only a limited number of times, tend to be precluded. By increasing the number of block types, B , available to the wasp agents, this problem can be somewhat attenuated (e.g., Howsman et al., 2004), but the size of the space of rule-sets scales very badly with B . The inherent stochasticity of the algorithm also works against the construction of particular complex architectures.

In this case, then, knowledge of the generic performance of the algorithm is more important than knowledge of its ultimate competence. The arguments pre-

sented here suggest that this will be true for bio-inspired algorithms in general. Conversely, it is interesting to note that an algorithm subject to such strong limitations is still perfectly capable of generating some (nest-like) repeated structures quite robustly. As such, the discovery of even very serious constraints on performance is still the discovery of knowledge that can be leveraged much more readily than any knowledge of the algorithm's generality.

Neuromodulation

Within neuroscience there is an increasing realisation that the traditional abstraction of neural systems as essentially networks of units interacting via neurotransmission is unsatisfactory since it neglects the role of the chemical substrate within which this electrical activity is embedded (Katz, 1999). The chemicals involved are implicated in numerous kinds of adaptive behaviour, from triggering plasticity and learning, reconfiguring neural circuits, and balancing gross levels of activity, to switching between multiple modes of behaviour. Emerging from this research is a new "liquid brain" perspective on real neural networks (Changeux, 1993).

GasNets are a class of recurrent artificial neural network inspired by this line of neuroscience research (Husbands, Smith, Jakobi, & O'Shea, 1998). In addition to a relatively standard explicit network of idealised neurons communicating via idealised neurotransmission, these ANNs employ an idealised type of chemical signalling in the form of simulated neuromodulators. GasNets have been artificially evolved successfully for a range of tasks including the control of autonomous mobile robots. In fact, they appear to be particularly suited to this kind of application (*op. cit.*).

What is important for the purposes of this paper is not that the GasNet mechanism (or some variant of it) might be demonstrated to be in some sense general purpose. In fact, one might easily imagine that, given an arbitrary number of idealised neurons sharing the same intrinsic timescale, augmented by a similarly arbitrary number of idealised neuromodulators acting over different spatiotemporal scales, the scheme could be shown to share with CTRNNs the ability to approximate any dynamical system to an arbitrary degree of accuracy. By contrast, from a nichiversal perspective, what is of interest is the generic behaviour of GasNets, and the extent to which this generic behaviour matches or suits the control problems facing the autonomous robots that we wish to artificially evolve. It is only by addressing the challenge of characterising this niche that we can determine the circumstances in which GasNets, rather than CTRNNs, say, are the appropriate architecture to employ.

More specifically, answering this question requires more than collecting a large number of examples of one paradigm outperforming another. Rather, a combination of carefully analysed successes (Smith, Husbands,

Philippides, & O’Shea, 2002), basic research into the original biological mechanisms (Philippides, Husbands, & O’Shea, 2000), new conceptual frameworks (Philippides, Husbands, Smith, & O’Shea, 2002), and fundamental modelling work (e.g., Buckley, Bullock, & Cohen, 2004, 2005) is necessary in order to reveal why, for example, ANN schemes that involve analogues of neuromodulatory chemicals are able in some cases to exhibit robust, evolvable, adaptive behaviour over multiple timescales (temporal adaptivity). It is through these parallel, overlapping activities that the GasNet niche might be characterised.

Summary

Only once we accept that, in general, biological devices, processes and organisations are properly viewed as specific to their particular niches, and (in collaboration with biologists) develop theoretical accounts of what it is that individual biological devices, processes or organisations are good at—what it is that they have been “designed” to achieve—will we be in a position to exploit idealisations of them efficiently.

Acknowledgements Thanks to three anonymous reviewers for their comments, and to Chris Buckley, Dan Ladley, Richard Watson, and the Leeds Biosystems group for wide-ranging discussions of these issues. Responsibility for the opinions stated here rests solely with myself.

References

Beer, R. D. (1995). On the dynamics of small continuous-time recurrent neural networks. *Adaptive Behavior*, 3, 469–509.

Bonabeau, E. W., Dorigo, M., & Théraulaz, G. (2000a). Inspiration for optimization from social insect behaviour. *Nature*, 406, 39–42.

Bonabeau, E. W., Guérin, S., Snyers, D., Kuntz, P., & Théraulaz, G. (2000b). Three-dimensional architectures grown by simple ‘stigmergic’ agents. *BioSystems*, 56, 13–32.

Bruinsma, O. H. (1979). *An analysis of building behaviour of the termite Macrotermes subhyalinus (Rambur)*. Ph.D. thesis, Landbouwhogeschool, Wageningen, Netherlands.

Buckley, C., Bullock, S., & Cohen, N. (2004). Toward a dynamical systems analysis of neuromodulation. In Schaal, S., Ijspeert, A. J., Vijayakumar, A. B. S., Hallam, J., & Meyer, J.-A. (Eds.), *Eighth International Conference on Simulation of Adaptive Behavior*, pp. 334–343. MIT Press, Cambridge, MA.

Buckley, C., Bullock, S., & Cohen, N. (2005). Timescale and stability in adaptive behaviour. In Bentley, P., Capcarrere, M., Freitas, A., Johnson, C., & Timmis, J. (Eds.), *Eighth European Conference on Artificial Life*, pp. 292–301. Springer-Verlag, Berlin.

Changeux, J. (1993). Chemical signaling in the brain. *Scientific American*, 268, 58.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.

Cook, M. (2004). Universality in elementary cellular automata. *Complex Systems*, 15, 1–40.

Foley, R. A. (1997). The adaptive legacy of human evolution: A search for the environment of evolutionary adaptedness. *Evolutionary Anthropology*, 4, 194–203.

Funahashi, K., & Nakamura, Y. (1993). Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6, 801–806.

Howsman, T. G., O’Neil, D., & Craft, M. A. (2004). Stigmergic cooperative multi-robot control architecture. In Pollock, J., Bedau, M., Husbands, P., Ikegami, T., & Watson, R. A. (Eds.), *Ninth International Conference on Artificial Life*, pp. 88–93. MIT Press, Cambridge, MA.

Husbands, P., Smith, T. M. C., Jakobi, N., & O’Shea, M. (1998). Better living through chemistry: Evolving GasNets for robot control. *Connection Science*, 10, 185–210.

Kahnemann, D., Slovic, P., & Tversky, A. (1982). *Judgement Under Uncertainty: Heuristics and Biases*. Cambridge University Press, Cambridge.

Katz, P. S. (1999). *Beyond Neurotransmission: Neuromodulation and its Importance for Information Processing*. Oxford University Press, Oxford.

Ladley, D., & Bullock, S. (2004). Logistic constraints on 3D termite construction. In Dorigo, M., Birattari, M., Blum, C., Gambardella, L. M., Mondada, F., & Stützle, T. (Eds.), *Fourth International Workshop on Ant Colony Optimization and Swarm Intelligence*, pp. 178–189. Springer-Verlag, Berlin.

Ladley, D., & Bullock, S. (2005). The role of logistic constraints in termite construction of chambers and tunnels. *Journal of Theoretical Biology*, 234, 551–564.

Ladley, D., & Bullock, S. (in preparation). Constraints on stigmergic structural evolution. Unpublished manuscript.

Millikan, R. G. (1984). *Language, Thought and Other Biological Categories*. MIT Press, Cambridge, MA.

Millikan, R. G. (1993). *White Queen Psychology and Other Essays for Alice*. MIT Press, Cambridge, MA.

Philippides, A. O., Husbands, P., & O’Shea, M. (2000). Four-dimensional neuronal signaling by nitric oxide: A computational analysis. *Journal of Neuroscience*, 20, 1199–1207.

Philippides, A. O., Husbands, P., Smith, T. M. C., & O’Shea, M. (2002). Fast and loose: Biologically inspired couplings. In Standish, R. K., Bedau, M. A., & Abbass, H. A. (Eds.), *Eighth International Conference on Artificial Life*, pp. 292–301. MIT Press, Cambridge, MA.

Smith, T. M. C., Husbands, P., Philippides, A. O., & O'Shea, M. (2002). Neuronal plasticity and temporal adaptivity: GasNet robot control. *Adaptive Behavior*, 10, 161–184.

Tarakanov, A. O., Skormin, V. A., & Sokolova, S. P. (2003). *Immunocomputing: Principles and Applications*. Springer-Verlag, Heidelberg.

Théraulaz, G., & Bonabeau, E. W. (1995). Modelling the collective building of complex architectures in social insects with lattice swarms. *Journal of Theoretical Biology*, 177, 381–400.

Turing, A. M. (1936). On computable numbers, with an application to the entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42, 230–265.

Wolpert, D. H., & Macready, W. G. (1995). No free lunch theorems for search. Technical report SFI-TR-95-02-010, Santa Fe Institute, Sante Fe, NM, USA.

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1, 67–82.

Zaera, N., Cliff, D., & Bruton, J. (1996). (Not) evolving collective behaviors in synthetic fish. In Maes, P., Mataric, M., Meyer, J.-A., Pollack, J., & Wilson, S. (Eds.), *Fourth International Conference on Simulation of Adaptive Behavior*, pp. 635–644. MIT Press, Cambridge, MA.