

Is Evolution by Natural Selection the Algorithm of Biological Evolution?

Richard A. Watson

Natural Systems Group, University of Southampton, U.K.
raw@ecs.soton.ac.uk

Abstract

It is tempting to be confident that we know how biological evolution works. After all, we know a mechanism capable of producing adaptation, and we understand the necessary and sufficient conditions for this to occur, and those conditions are met in natural populations – the rest is surely just details. However, there can be many different algorithms that utilise a given underlying mechanism (sub-algorithm), and in other contexts we cannot assert that we know what algorithm is operating just because we identify a sub-algorithm it contains. Using sorting algorithms based on the mechanism of ‘compare and swap’ (as an analogue of evolutionary algorithms based on natural selection) we discuss three substantial ways in which an algorithm can be based on, and depend on, a mechanism and yet not be that mechanism, each of which has some bearing on natural processes of evolution: 1) unstructured versus structured applications of a mechanism, 2) data-independent versus data-dependent, 3) iterative versus recursive. In the context of computational algorithms more generally, it is easy to see that each of these issues corresponds to different algorithmic classes. We suggest that in natural evolution, it is not obvious that none of these issues apply, nor that the empirical evidence supports the view that an unstructured, data-independent and iterative interpretation of natural selection is sufficient to create biological evolution.

Biological Evolution and Natural Selection

Here we are interested in the adaptive aspects of evolution and the algorithmic principles that produce such adaptation. To begin, we must distinguish adaptive aspects of *biological evolution*, i.e., the phenomenon of adaptation that actually happens in the natural world (by whatever mechanism), from *evolution by natural selection*, ENS, i.e., the specific algorithmic account of biological evolution originating with Darwin. Of course, Darwin did not know a lot of the mechanical details that were filled-in later by neo-Darwinism, the Modern Synthesis and subsequent work. But here we are interested in the general form of the algorithm described by, for example, Lewontin (1970) – i.e. heritable variation in reproductive success. ENS, as used here, therefore refers to the standard model: i.e. an algorithm involving a population of individuals, reproducing at variable rates as determined by heritable characteristics that are susceptible to random variation. These basics have not changed since Darwin and

continue to form the basis of all working models for adaptation in evolutionary biology despite the many complexities of real biology that have become evident since Darwin. To take just one example, although evidence for the neutral theory of evolution (Kimura, 1985) might change our interpretation of evolutionary processes, few would argue that it invalidates ENS as an explanation of adaptive change. Indeed, it almost seems impossible, at least to some, that any such detail could alter the fundamental algorithm of evolution.

We take it as given that biology instantiates ENS. That is, ENS occurs in biological evolution (there is no need to reiterate the evidence for this). However, we wish to separate the conclusion that ENS *occurs in* biological evolution from the conclusion that the algorithm of adaptive biological evolution *is* ENS. Ordinarily, the notions involved are difficult to separate – or at least, care is not taken to separate them. The statement ‘evolution is true’, for example, fails to separate the claims that biological evolution has occurred (species have changed adaptively over time), that ENS has occurred (as described by Darwin), and/or that ENS is the mechanism by which biological evolution has occurred. A lot of emphasis is placed on showing that a biological population instantiates ENS with the implicit assumption that, if we show that it does, then we have shown that we know how biological evolution works. Does that necessarily follow?

Crudely, the issue that we want to discuss is something very simple: that a physical system can (trivially) instantiate more than one algorithm simultaneously; in particular, that an algorithm *B* can contain another algorithm *A*, ($B \neq A$). For example, an algorithm for matrix multiplication contains an algorithm for addition but it is not an algorithm for addition. Thus, biological evolution may *contain* ENS (and it does), but it might not *be* ENS. Accepting this logical possibility immediately and directly leads to the conclusion that no amount of evidence for ENS in biological populations can enable us to conclude that we know the algorithm of biological evolution. Also, despite the fact that addition is simpler than matrix multiplication, addition is not a more parsimonious explanation of matrix multiplication because it is not sufficient for matrix multiplication. Likewise an argument of parsimony does not enable us to conclude that we know the algorithm of biological evolution unless we can show that ENS (alone) is sufficient for biological evolution.

Of course, previous work has discussed at length evidence for the sufficiency of ENS to produce the biological

adaptation we observe in nature, e.g., (Sober, 1984; Neander, 1995; Bedau, 2008). However, such debates are often hampered by the implicit assumption that ENS is effectively a synonym for a natural account of biological evolution (in contrast to a supernatural ‘account’). Because of this assumption, the assertion that there must be a natural algorithm responsible for biological evolution forces the conclusion that ENS is that algorithm – and the result of this circular reasoning leads to the conclusion that ENS must be sufficient for biological adaptation. An inability to see that ENS might not be the algorithm of biological evolution (even though biological evolution surely depends on it) makes it impossible to discuss properly the possibility that ENS might not be sufficient to produce the adaptation we observe¹.

Nonetheless, two *uninteresting* possibilities for how biological evolution might contain ENS but not be ENS might come to mind: a) there exists an altogether different algorithmic process, operating over and above ENS, such that the existence of ENS in biology is sort of a coincidence/ or even a red herring, b) that biological evolution is some small variant of ENS, ‘ENS plus some bells and whistles’, but really, the fundamental nature of the algorithm is still ENS. A constructive discussion will require a carefully considered middle ground that neither depends on fantastical hypothetical alternatives nor on splitting-hairs. The issues we discuss are not merely hypothetical – and, in fact, the existence of relevant features/mechanisms in biological evolution is not in question. The more difficult issue is whether such features are minor details or algorithmically substantive.

The main contribution of this paper thus concerns issues of algorithmic equivalence. We want to address features that change the fundamental nature of an algorithm – but we also want to show that such features can nonetheless be rather subtle. In fact, we restrict ourselves further to cases where *A* and *B* have a particular kind of relationship such that *B* does not merely contain a mechanism (sub-algorithm) *A*, but *B* is *based on A*. Meaning that *B* depends on *A* (*A* is essential for *B*), and in a sense *B* is just *A* arranged in a particular manner, and the difference between *A* and *B* might come ‘for free’ (without design). Using concrete examples from another domain, we then investigate whether it can be the case that even though *B* is based on *A* in this restricted sense, it can nonetheless be the case that, we do not know how *B* works just because we know how *A* works. If so, this potentially prevents us from concluding that we know how biological evolution works even if granted that it is based on ENS.

Natural Selection + $X \approx$ Natural Selection?

The conditions that facilitate the process of evolution by natural selection – i.e. heritable variation in reproductive

¹As scientists we must be careful that we do not fall back on the following argument: 1) either it’s ENS or it’s supernatural, 2) it is not supernatural, 3) therefore it’s ENS. Even though the logic of the argument is correct, the conclusion is false because the first clause is false. Despite the fact that ENS occurs in nature and is capable of producing adaptation, it is not the only logical possibility (even if we restrict ourselves to mechanisms *based on ENS*). The assumption that questioning the sufficiency of ENS implies a willingness to entertain supernatural ‘accounts’ is potentially highly damaging to scientific debate.

success – are common in natural populations. The action of natural selection can be observed in natural populations and under controlled conditions; and, it is evidently capable of producing adaptation. The fossil record and genomic data show that all living things are connected in a tree of incremental (phenotypic and genetic) changes as the theory predicts. There is therefore no doubt that natural selection occurs and that it is fundamental to evolutionary change.

Given these facts, the claim that we know how evolution works seems reasonable. Dawkins, for example, states that “What Darwin achieved was nothing less than a complete explanation for the complexity and diversity of all life” (Dawkins, 2008). Obviously, “complete” is an overstatement. Many details have been filled-in/added-on over the last 150 years – including neutral evolution (Kimura, 1985), kin selection (Hamilton, 1964), niche construction (Odling-Smee et al, 2003), epigenetic inheritance (Jablonka & Lamb, 1995), self-organisation (Kauffman, 1993), symbiogenesis (Margulis & Fester, 1991), exaptation (Gould & Lewontin, 1979), ‘evo-devo’ interactions (Sommer, 2009), lateral gene transfer (Doolittle & Bapteste, 2007), compositional evolution (Watson, 2006), etc. – not to mention the molecular basis of inheritance – and there will surely be more. Moreover, some authors argue that issues such as these have a fundamental bearing on the underlying algorithm of biological evolution (see also Pigliucci, 2007). But others disagree – the hyperbole of Dawkins aside, the sentiment is that all of these ‘add-ons’ are merely contingent implementation details compared to the fundamental mechanism that drives it, ENS.

Underlying this there is perhaps a belief that ENS is the only adaptive algorithm that could *possibly* occur spontaneously in a physical substrate (and that therefore all other details *must* be either derivatives of it or unimportant). This assumption must be dispensed with. After all, prior to Darwin, no one could imagine *any* adaptive algorithm that could possibly occur spontaneously in a physical substrate. But Darwin showed that there exists at least one algorithm in that class. Is ENS really so fundamental that it is impossible for another algorithm to exist that is not simply a derivative of it? Consider a trivial counter-example; the optimization algorithm simulated annealing (Kirkpatrick et al, 1983). This occurs spontaneously in physical systems – in hot lumps of metal and other crystals as they cool – which was the inspiration for the computational algorithm in the first place. But simulated annealing is not natural selection and not a derivative of it. One might object that this example is merely the result of a physical dynamical system just doing what it does naturally – but so is natural selection, of course – there is nothing ‘other worldly’ about ENS. One might also object that simulated annealing is arguably a less sophisticated (weaker) algorithm than ENS, we would agree. Nonetheless, not all natural algorithms are necessarily ENS or derivatives of it.

However, in this paper we deliberately restrict ourselves to algorithms that, like all of the expansions mentioned above, *contain* ENS, and to algorithms that are based on ENS in a fundamental manner such that ENS is essential for their operation – i.e., no adaptation would occur without the inclusion of ENS. This might appear to concede that we are

merely talking about ‘add-ons’ – additions or extensions that do not change the fundamental underlying algorithm. But it is exactly the validity of this conclusion that we wish to discuss.

Algorithmic equivalence and subtle ways an algorithm can be based on a given mechanism

In the context of algorithms in general, it is easy to confirm that algorithm *B* can contain another algorithm *A* and yet not be algorithm *A* (see the matrix multiplication example above, or, in turn, the relationship of matrix multiplication to some signal processing algorithm, for example). Given that *B* contains *A*, we could argue that *A* is not an *incorrect* description of *B* but merely *incomplete*. But this would clearly be disingenuous in some cases (e.g., the missing details between addition and matrix multiplication are clearly fundamental). Likewise, it follows that, *in principle*, many different algorithms could contain natural selection and yet not be natural selection in a fundamental sense. But in the abstract, this point has little biological relevance. We need to restrict ourselves to biologically relevant algorithmic variants. However, rather than, at the other extreme, allowing the specific biological details (neutral evolution, evo-devo etc.) to drive the discussion, in this paper we take a different route.

Instead we discuss some specific but canonical ways in which algorithms can belong to fundamentally different algorithmic classes – despite all being based on the same underlying mechanism – simply by applying that mechanism in different ways. Namely, by applying that mechanism in different structural arrangements, in dynamic arrangements and in recursive arrangements:

- 1) unstructured vs structured applications of a mechanism,
- 2) data-independent vs data-dependent applications,
- 3) iterative vs recursive applications of a mechanism.

To this end we discuss at some length a number of non-evolutionary algorithms; in particular, different types of sorting algorithms all based on the mechanism of ‘compare and swap’. In this context it is clear that many different algorithms can all be based on and dependent on the same underlying mechanism, and yet belong to fundamentally different algorithmic classes. This enables us to discuss the relevant conceptual issues in a domain that is uncontroversial. We suggest that in light of such analogies, some important issues in evolution that presently seem inseparable can be teased apart, enabling us to ask clearer questions about biological evolution, and make clearer claims about our knowledge of it. This approach does, of course, have the weakness that we do not address specific biological mechanisms in detail – one may simply conclude that although our point may be true for sorting, it is not true for evolution. But it is our contention that in the domain we actually care about, biological evolution, the relevant facts (the algorithmic principles and their adaptive consequences) are not known – making it impossible to discuss the relevant issues. In the meantime, we aim to open up the relevant conceptual space to identify the relevant questions.

Our aims in this paper are therefore to discuss concepts such as algorithmic equivalence, and algorithmic classes, and in particular the implications of this for how we understand

evolution by natural selection. This is discussed at a largely conceptual level using analogies with other algorithmic domains. In each of the following sections we discuss the relevant issues with respect to algorithms in general (where the conclusions are uncontroversial), and then indicate their potential relevance to biological evolution. We then recap the implications and draw some general conclusions with respect to what we know about natural evolution.

Unstructured and structured applications of a mechanism

We begin with the subtle ways that the *structural context* of a mechanism can result in different algorithms.

Unstructured and structured sorting algorithms

Consider algorithms for sorting a list of numbers. Many sorting algorithms can be described as multiple applications of a ‘compare and swap’ (C&S) operator:

Compare-and-swap($\langle a, b \rangle$):

If $a \geq b$ return $\langle a, b \rangle$ else return $\langle b, a \rangle$.

Bubble-sort, for example, iterates through a list repeatedly, applying C&S to adjacent numbers in the list. One can visualise the order in which C&S operations are applied as a *sorting network* (Fig. 1) (Knuth, 1973; Cormen et al, 2001). Finding sorting networks with minimal number of comparators (and/or minimum depth) for a given number of inputs is a favourite sport of computer science (including, e.g., Hillis’ (1990) use of coevolutionary methods). Many different sorting algorithms (with different time complexities) can be described as sorting networks (Fig. 1).

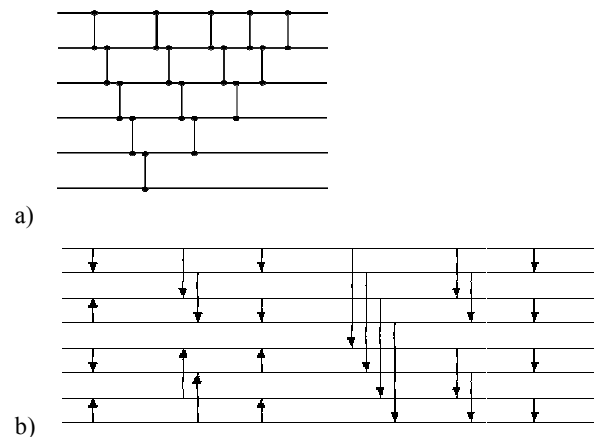


Figure 1: Sorting networks. Horizontal lines carry inputs that flow from left to right, vertical junctions apply C&S operations – a) Bubble-sort (the number of C&S operations can be reduced by 1 each iteration through the list as shown, giving a time complexity of $N(N-1)/2$ rather than N^2). b) Bitonic-sort, time complexity $\Theta(N \log(N)^2)$ (Batcher 1968).

Bubble-sort and Bitonic-sort both involve many C&S operations, and no sorting would occur without C&S; C&S is essential for sorting to occur. But even though the only difference between them is how the C&S operations are arranged, Bubble-sort and Bitonic-sort are not two different

descriptions of the same algorithm. This is evident in the observation that the two algorithms have different time complexities, and therefore, for a given time limit, one can sort lists of a given size correctly that the other cannot – i.e. Bubble sort does *not* sort correctly in time $N \cdot \log(N)^2$.

Note also that compare and swap itself is an algorithm. Both Bubble-sort and Bitonic-sort *contain* the C&S algorithm. But neither of them *are* C&S – there is more to them than that. It is therefore not the case that describing an algorithmic mechanism, *A*, that is essential for and contained in an algorithm, *B*, is the same as describing the algorithm *B* even if *B* is essentially just a particular arrangement of *A*. In these examples, the ‘structural context’ in which C&S occurs is necessary in order to describe Bubble-sort or Bitonic-sort. Note that it would be true to say that, given the right structural context, C&S results in sorting. But this still would not distinguish whether it was Bubble-sort or Bitonic-sort that had been implemented. Note also that some arrangements of C&S operations do not result in correct sorting for all inputs. Thus C&S is not sufficient for sorting.

Of course, Bubble-sort and Bitonic-sort are very special arrangements of C&S operations. However, a random arrangement of comparators with sufficiently many C&S operators would sort correctly so long as all comparators are pointing the same way – no other organisation is required. The expected time complexity (expected number of operators needed to sort correctly) for such a network is no more than N^2 times more than the time complexity of Bubble-sort, for example, (consider the probability of placing a required comparator between a particular pair of lines, and the fact that extra comparators do not hinder sorting). Such a *Random sort* network can reasonably be described merely as ‘lots of C&S operations’ since the structural context is minimal.

It is also instructive to consider the possibility of sorting networks that are structured in only subtle ways – e.g., such that nearby lines have a higher probability of a comparator being placed between them. If we restrict comparators to adjacent lines only, then the time complexity of *Adjacent-only random sort* is no more than N times the time complexity of Bubble-sort, for example – since Bubble-sort uses only this type of comparator (and the probability of placing a particular comparator is now $1/(N-1)$).

Unstructured and structured natural selection

We explore the analogy that ‘C&S is to sorting’ what ‘natural selection is to biological evolution’ (see Box 1). The sorting examples show that the structural context of an algorithm *A* (e.g., C&S) can change the algorithm *B* (containing *A*) in substantive ways. Thus even if biological evolution contains ENS we cannot necessarily conclude that the algorithm of biological evolution is ENS even if the only difference is how ENS is ‘arranged’. This would be reasonable only if (like Random sort) the structural context of natural selection in biological evolution was minimal – in this case biological evolution would be nothing more than ‘lots of natural selection’. But if biological evolution requires natural selection to be applied in a particular structural context then that could constitute a substantially different algorithm. This

To flesh-out the analogy, consider the ‘compare and copy’ (C&C) operator below:

Compare-and-copy($\langle a, b \rangle$):

If $a > b$ return $\langle a, a \rangle$ else return $\langle b, b \rangle$.

We can plug-in this operator in place of C&S into the above sorting algorithms. This will produce algorithms that take a list of N numbers as input and output a list of numbers that has multiple copies of numbers from the input in proportion to the number of two-player tournaments that they win. This is a simple selection algorithm or the reproduction part of an evolutionary process. We could likewise define a probabilistic version of this operator (copying a over b with probability that takes account of the ratio of their fitnesses) if that were desirable. A *compare-and-copy-with-variation* operator would provide all characteristics of heritable variation in reproductive success. To produce multi-generation evolution one would need to repeatedly call the sorter with the output of the previous ‘generation’. (See the *Microbial GA*, (Harvey, 2011), for a genetic algorithm using a steady state strategy with pairwise tournaments and in-situ variation (including sexual recombination) – but no structured context, by default).

Note that the time complexity of the sorting algorithms would then transform into the time complexity required to make N copies of the biggest number (there are easier ways to do that, but that’s not the point here). E.g., Adjacent-only random sort, given the C&C operator instead of C&S, would require less generations on average than Random sort to converge to an output where the biggest number is copied N times. Therefore, given limited time, Adjacent-only random sort could produce convergence in some cases where Random sort could not.

Box 1: From sorting algorithms to evolutionary algorithms

runs counter to the assumption that any algorithm based on natural selection is the same algorithm regardless of context.

In evolutionary theory it is well known that population structure changes the effective unit of selection. That is, *relatedness* in kin selection theory does not measure genetic relatedness in an absolute sense, but rather the genetic relatedness of the individuals that interact compared to the genetic relatedness of the population as a whole (Michod & Hamilton, 1980). Population structure therefore changes relatedness. Kin selection, or inclusive fitness theory, provides an explanation for differing levels of cooperation in a population, for example – i.e., different social outcomes. Multi-level selection theory (Wilson, 1992), including type-1 group selection, extends these principles. In principle, something as simple as the fact that a population is spatially embedded (altering who interacts with whom and who competes with whom) thus alters the structural context in which natural selection applies, e.g. by making proximal individuals more likely to participate in a competitive interaction than distal individuals (compare with Adjacent-only random sort). Then consider gene selection in the context of multi-cellular organisms and how much these ‘vehicles’ (Dawkins 1976) structure the context of genic selection. These

observations are not usually taken to imply a different algorithm for biological evolution. Should we conclude that these issues are minor modifiers on ENS? Or, that structured ENS, like structured sorting, constitutes a different algorithm? We suggest, it is not so obvious that structuring does not change the algorithm of evolution, nor that unstructured natural selection captures what is important about biological evolution any more than unstructured C&S captures sorting.

Data-independent and data-dependent applications of a mechanism

Clearly, the output of an algorithm is sensitive to its input. But the way in which an algorithm operates can also be sensitive to the input, and again, this can be rather subtle.

Data-independent and data-dependent sorting

In a sorting network, the sequence of C&S operations is fixed (part of the interest in them derives from the fact that their fixed arrangement makes them suitable for implementation in hardware). But other sorting algorithms exist that do not have this property. *Merge-sort*, for example, is a data-dependent algorithm. Merge sort depends on a Merge procedure (applied recursively) which takes two sorted lists (length $k/2$) and combines them into a single sorted list (length k). Each Merge requires only order N C&S operations. The recursive application of Merge, in effect, combines lists of length 1 (necessarily already sorted) into successively bigger lists until a complete list is returned. Quick-sort (which has an ‘in situ’ version, using no additional registers) is a sorting algorithm working on similar principles but ‘bottom-up’. Merge-sort and Quick-sort have (optimal) total time complexity $O(N \log N)$.

The Merge procedure compares the items from the tops of the two sorted input lists and transfers the smaller to the output list. This reveals a new top item in one of the lists. By repeating until the lists are empty, a fully-sorted output list is created. Note that there is no fixed order to the comparisons made in Merge – the n^{th} (for $n > 1$) comparison made depends on the outcome of the $(n-1)^{\text{th}}$ comparison and all previous comparisons. E.g. the first comparison is between A1 and B1 (the first elements of each list), then the second comparison is between A1 and B2, if B1 was greater than A1, whereas it is between A2 and B1, otherwise. Thus we cannot describe Merge-sort as any fixed ordering of compare and swap operations. Put another way, to implement Merge-sort in a sorting network would require a network where the result of a comparison at one point in the network influenced the presence or absence of a comparator down-stream.

It is instructive to consider a Random sorting network with a simple kind of data-dependence. For example, suppose that whenever a comparator does not result in a swap, nearby downstream comparators on the same lines are skipped. We can see that this might increase the efficiency of the sort by avoiding redundant comparators in some cases. More sophisticated local rules are also worth contemplating; e.g., if neither of the inputs to a comparator were altered since the last comparator on those lines, skip the comparator. Such a rule could be used in conjunction with a ‘fully-connected

sorting network’ – i.e. where all N^2 comparators are repeated N times. This network sorts correctly (Bubble sort is a subset of this network), and the data-dependence rule cannot prevent it from sorting correctly, and with the data-dependence rule it would use much less than the N^3 comparators present.

Note that even for a data-dependent algorithm there is a trace-back through time such that, at every point in time, we can explain a new list-ordering given the previous list-ordering and the application of the C&S operation applied at that point in time. But that is true for Merge-sort just like it is true for Bubble-sort or Random-sort – i.e., post hoc analysis shows that there is a sequence of C&S operations and, given that they occurred in that order, they explain the correct sorting. But the existence of such a trace (per se) does not distinguish which algorithm we are tracing or explain how they came to be in that order.

Data-independent & data-dependent natural selection

In the examples of contextual structuring we discussed above (e.g., population structure, kin selection, vehicles, multi-level selection) we assumed that these structures were constant or provided by extrinsic factors (e.g., spatial embedding or happenstance contingency). But, of course, they are also influenced by the action of natural selection itself. For example, the evolution of individual traits that affect habitat preference inevitably affect population structure and thus relatedness. Recent work (Powers, 2010; Powers et al, 2011; Snowdon et al, 2009) has begun to investigate the evolution of individual traits that affect the level of selection via *social niche construction* (Powers, 2010). This is a mechanism where (by analogy with *niche construction*, Odling-Smee et al, 2003) an organism alters its social context (who it interacts with and how much) and thereby affects the selective pressures on its social behaviour (e.g. cooperation). This fits directly with well-known theory relating population structure to social evolution (e.g. spatial or grouped population structures promote cooperation; Nowak & May, 1992). But whereas most studies assume that population structure is a given, social niche construction includes individual traits that alter population structure (e.g., via habitat preference, or selective adhesion, or the evolution of vertical transmission mechanisms). One particular study (Powers et al, 2011) investigates the evolution of initial group size in an aggregation and dispersal process and shows that individual natural selection drives group size down to increase cooperation (Szathmary, 2011). We have been investigating analogous mechanisms in various domains, in particular in adaptive networks (Gross & Sayama, 2009) where the topology of the network affects the behaviour on the network, and reflexively, the behaviour on the network affects the network topology (Watson et al, 2010; 2011a; 2011b).

Thus, by straight-forward means, the outcome of natural selection at one point in time can affect the way in which natural selection is applied at a future point in time (see also Neander, 1995). Thus, biological evolution is data-dependent. In principle, this puts it in a fundamentally different algorithmic class from data-independent natural selection. Of course, one might argue that Lewontin’s formulation, for

example, does not categorically exclude data-dependence (since the possibility is not mentioned). But the omission is potentially as substantial as saying that Merge-sort is lots of C&S without mentioning that the ordering depends on input.

Note also that there is a trace back through time such that, at every point, we can explain a new state of a population as a result of ENS acting on individual traits. But that does not distinguish which evolutionary algorithm we are tracing; in particular, whether the population structure that determined the structure of the trace was data-dependent or not. Thus, the fact that evolved organisms fit into a tree of life does not mean we can conclude that evolution is data-independent ENS (both data-independent and data-dependent ENS algorithms would have the property that results fit into such a tree).

Iterative versus recursive applications of a mechanism

Merge-sort, as well as being data-dependent, is also a recursive algorithm. The recursive application of a mechanism can result in a substantially different algorithm from iterative applications. Again, this has interesting analogues in biology.

Iterative versus recursive sorting

Bubble-sort is a simple iterative algorithm. Merge-sort (like Quick-sort) is a recursive algorithm. It sorts a list by dividing it in two, sorting each sub-list *using Merge-sort* (i.e., dividing it in two, sorting each sub-list *using Merge-sort*, and merging the sub-lists back together using the Merge procedure), and merging the sub-lists back together using the Merge procedure. To prove that Merge-sort sorts correctly we can use a proof by induction. First we show that a list of just one number is already sorted. Then we show that the *Merge* procedure, given two sorted input lists, produces one sorted list containing the numbers from both.

Note that the Merge procedure is not in itself a sorting algorithm – it will not produce sorted output from arbitrary inputs, only from two pre-sorted lists. Thus if we describe the Merge operation on its own, i.e. without the context of the recursive structure, it does not describe a process that sorts.

(For interest, it is not too hard to define a sub-network that carries out Merge using only order N C&S comparators, as Merge does, by starting with a full set of N^2 comparators and using data-dependence that turns off downstream comparators based on the outcome of upstream comparators, as mentioned previously. Thus, a Merge-sort network could be constructed using such dynamic sub-networks arranged appropriately).

Suppose we were to jump into a trace of the Merge-sort algorithm at a particular level of recursion; perhaps the last Merge before the sorted output is produced (i.e., two lists of $N/2$ into one list of N). We could explain (using data-dependent C&S operations) how Merge gets Merge-sort from this point in its operation to the final sorted output. However, this would fundamentally fail to explain the sorted result because it fails to explain how the two sub-lists came to be sorted at this stage of operation. We could try to explain this by saying that more Merging was involved; but note that it would not be more Merging at the same level of description.

We have to refer to Merging at multiple levels of organisation – i.e., no one level of Merge explains Merge-sort.

Iterative versus recursive natural selection

Recursion is obviously a very special algorithmic structure. But multiple nested levels of structural organisation are ubiquitous in nature – in both evolved and non-evolved systems (Lenaerts et al, 2005). The major transitions in evolution (Maynard Smith & Szathmari, 1995) describe just such a multi-scale structure, applying natural selection (in a structured and data-dependent manner) at many scales of organisation. Maynard Smith & Szathmari describe a set of transitions that have been fundamental in the evolution of complexity. These events, including for example the transition from self-replicating molecules to protocells and unicellular organisms to multi-cellular organisms, share the property that “entities that replicated independently before the transition can replicate only as part of a larger whole after the transition”. These processes are therefore entangled with issues such as changes in the unit of selection (Okasha, 2006; Michod, 1999; Buss 1987), and the ‘de-Darwinisation’ of lower level units and ‘Darwinisation’ of new higher-level units along various dimensions (Godfrey-Smith, 2009).

Note that jumping in at a particular level within this hierarchy to try and describe how natural selection proceeds at that one level of organisation would not describe the algorithm responsible for the evolutionary outcomes we observe because it would not explain where the inputs to this level of organisation came from. At one of the lower levels of organisation, this is loosely related to Sober’s (1984) position that natural selection can explain why a population exhibits trait *a* in preference to trait *b*, but not how either of those traits originated. It is a little too easy to simply assert that they originated from the prior action of ENS because we may be conflating different descriptive levels when we do this.

Put another way, consider the necessary and sufficient conditions for ENS described by Lewontin – heritable variation in reproductive success. Notice that all these terms require us to define the units we are talking about so that we can define reproduction (and Darwinian fitness/reproductive success), heritability and variation. For example, we could focus on the level of genes (as Dawkins advocates), then we can talk about the heritability of genes given a set of genetic variation operators (mutation and recombination), and selection on genes (either in the context of cells or sexual organisms), and given a physical substrate that defines how well a given genetic sequence survives and replicates. But clearly, a lot of machinery is already assumed here, and not all of it obviously comes ‘for free’ from the biophysical properties of molecules. Sexual recombination, for example, is an evolved mechanism that radically changes the effective unit of selection from genomes to genes (Watson, 2005) – without some mechanism that enables genes to be inherited individually the premise of genic selection is meaningless, and we would be talking only about genome selection.

The point about the Merge procedure is not merely that the inputs (sorted sub-lists) are variable in size or that sub-lists of different sizes are relevant at different stages of the process.

The point is that the Merge procedure (despite containing lots of C&S) is not a sorting algorithm at all, and only when one appreciates that Merge-sort is recursive, and therefore continually redefines the inputs to the Merge procedure, do we understand how Merge-sort produces sorted outputs. Likewise, the point is not merely that the terms of reference in biological evolution are a bit slippery – a bit difficult to define clearly (this point has been made many times, e.g. see Godfrey-Smith, 2009 for many interesting examples). The point is that there is not necessarily *any* one set of terms that satisfies the requirements of the process.

'Self-structuring' & Opaque Consequences of ENS

Recursion involves a process turned upon itself. More generally, the idea that evolution can modify its own operation is discussed in the evolution of evolvability (Kirchner & Gerhart, 1998; Sterelny 2011) and, e.g. evolved exploration distributions (Toussaint & von Seelen, 2007; Parter et al, 2008) that alter the space of phenotypic possibilities on the fly. Likewise, the evolution of new genetic mechanisms (non-random genetic variation mechanisms, e.g. via mobile genetic elements; Shapiro, 2011) can alter the space of genetic possibilities – analogous to 'self-modifying code' in computer science. In the major transitions we contemplate evolution modifying its own operational units.

Such recursive and self-referential notions of evolution present a concept of evolution that continually 'reinvents itself'; changing the level of selection, forming new mechanisms of heredity and creating new evolutionary units at successive scales of biological organisation. Thus evolutionary processes co-create the structural context of selection, effectively re-defining the evolutionary process (Sterelny, 2011; Calcott & Sterelny, 2011; Godfrey-Smith, 2011). We refer to this as '*self-structuring evolution*'.

The analogy suggests that explaining what is going on in some of these biological processes, especially in the major transitions and self-structuring evolution, is not captured by ENS, any more than Merge-sort, for example, is captured by unstructured, data-independent and iterative applications of the compare and swap mechanism. If this is the case, then *plain ENS*, i.e. unstructured, data-independent and iterative ENS, is not necessarily the algorithm responsible for producing biological adaptation even though it is evidently capable of producing some adaptation.

However, the question then becomes, is self-structuring ENS merely plain ENS given the right kind of substrate or materials? That is, *given appropriate conditions*, can plain ENS create structured, data-dependent and recursive ENS? If so, then any failure of plain ENS to explain biological evolution seems to be merely an epistemological issue – i.e., a failure to comprehend or deduce the opaque consequences of the original simple algorithm. We have some sympathy for this position. But plain ENS is not going to create structured ENS in all substrates/ environments – some substrates won't allow data-dependence or recursion for example. We would argue that understanding the conditions for plain ENS to become self-structuring is really a necessary part of describing

the algorithm (Bedau, 2008) – and such understanding is not captured by the plain ENS algorithm per se.

Moreover, the process that transforms plain ENS into structured ENS is not *necessarily* ENS itself. We might use the term 'self-organisation' to cover a multitude of possibilities with respect to order that comes 'for free' in physical systems (although note that here we are talking about self-organisation of the algorithm itself, not merely of the object/material that ENS operates on). We have been investigating a more specific mechanism of *associative induction* (Watson et al, 2010; 2011a; 2011b; submitted) that arises 'for free' in adaptive networks, for example. The hierarchical form of self-structuring evolution that results has a fundamentally different algorithmic capability from plain ENS (Mills, 2010; Watson et al, submitted). This implies that the algorithmic nature of biological evolution is not merely a point of view but can be settled empirically.

Conclusions

Understanding how compare-&-swap or Random sort works is a long way from understanding how Merge sort works. More generally, knowing that an algorithm sorts, and that it contains C&S, does not tell us whether that algorithm is an unstructured, data-independent and iterative algorithm (like Random sort) or, at the opposite extreme, a structured, data-dependent and recursive algorithm (like Merge sort). Thus, it is not the case that we necessarily know how an algorithm *B* works, even if we know that it contains a known algorithm *A*. Moreover, the examples of sorting algorithms show that *B* can belong to fundamentally different classes of algorithm even when the relationship between *A* and *B* is highly restricted such that *B* not only contains *A* but is *based on A*: Specifically, *B* depends on *A*, *A* is essential for the operation of *B*; *B* is, in a sense, just an arrangement of *A* (albeit perhaps a dynamic and/or recursive arrangement) and that arrangement can in some cases be built-up using only local restrictions and/or simple restructuring principles.

This shows that conditions that produce structured, data-dependent and/or recursive applications of a mechanism can result in an algorithm that is in a fundamentally different class from unstructured, data-independent and iterative applications of the same mechanism. Thus, even if we grant that biological evolution not only contains ENS but is *based on* ENS in this restrictive sense, no amount of evidence for the existence of ENS in nature enables us to conclude that we know the algorithm of biological evolution.

Parsimony would preclude the need to consider alternative algorithms for biological evolution if, but only if, it was shown that ENS was a sufficient algorithm to produce biological evolution. Thus, consider the statement: *There are no known examples of extent organisms or adaptations that could not plausibly have been produced by ENS given appropriate conditions/arrangements*. And compare with: *There are no known examples of sorting that could not plausibly have been produced by compare-and-swap given appropriate conditions/ arrangements*. Or for that matter: *There are no known examples of matrix multiplication that could not plausibly have been produced by addition given*

appropriate conditions/ arrangements. Obviously the latter statements can only be true because the ‘appropriate conditions’ clause can bury substantial algorithmic structure. The question is thus whether ‘the conditions’ of biological evolution conceal substantial algorithmic structure. We have discussed how these conditions might include structuring, data-dependence and recursion and that doing so would change the fundamental nature of the algorithm. We cannot therefore accept the first of these three statements as evidence that ENS is algorithmically sufficient for biological evolution.

In conclusion, we suggest that it is not at all clear that biological evolution is unstructured, data-independent and iterative – indeed, we have discussed specific evidence to the contrary. Thus, notwithstanding the fact that biology instantiates ENS, it is certainly not for granted (arguments of parsimony included) that evolution by natural selection is the algorithm of biological evolution.

Acknowledgements: Thanks to Chris Adami for motivating this exploration. Thanks to Hywel Williams, Paul Ryan, Jason Noble, Adam Davies and Miguel Gonzalez Canudas for discussion of the manuscript, and Keyvan mir Mohammad Sadeghi for background research on sorting networks.

- Batcher, K.E. (1968) Sorting Networks and their Applications. *Proc. AFIPS Spring Joint Comput. Conf.*, Vol. 32, 307-314.
- Bedau, M.A. (2008). The evolution of complexity. In Thomas Pradeu, et al., eds., *Mapping the Future of Biology: Evolving Concepts and Theories*.
- Buss, LW, (1987), *The Evolution of Individuality*, Princeton Press, NJ.
- Calcott, B. & Sterelny, K. (2011) "Introduction: A Dynamic View of Evolution" in *The Major Transitions in Evolution Revisited*, MIT Press.
- Cormen, T.H., C.E. Leiserson, R.L. Rivest, C. Stein (2001) *Introduction to Algorithms. 2nd edition*, MIT Press.
- Darwin, C. (1872) *The Origin of Species*, London, John Murray.
- Dawkins, R. (1976), *The Selfish Gene*, Oxford University Press.
- Dawkins, R. (2008) *The Genius of Charles Darwin, Part 1. Life, Darwin & Everything*. Channel 4 television (broadcast August 2008).
- Doolittle, W. Ford, and Eric Baptiste. (2007). Pattern pluralism and the tree of life hypothesis. *PNAS* 104(7): 2043–2049.
- Godfrey-Smith, P. (2009). *Darwinian Populations and Natural Selection*. Oxford.
- Godfrey-Smith, P. (2011) "Darwinian Populations and Transitions in Individuality" in *The Major Transitions in Evolution Revisited*, MIT Press. 65-82.
- Gould S.J. & Lewontin, R.C. (1979) The Spandrels of San Marco and the Panglossian Paradigm. *Procs. Roy. Soc. B* 205:1161, 581-598.
- Gross, T. & Sayama, H. (2009) *Adaptive Networks. Theory, Models and Applications*. Springer-Verlag: Berlin.
- Hamilton W. D. (1964). The Genetical Evolution of Social Behavior. *J. Theor. Biology* 7 1-16.
- Harvey, I. (2011) The Microbial Genetic Algorithm, *Procs. ECAL 2009*, 126-133.
- Hillis, W.D. (1990) "Co-evolving Parasites Improve Simulated Evolution as an Optimization Procedure" *Physica D: Nonlinear Phenomena*. 42(1-3) 228–234.
- Jablonka, E. & Lamb, MJ (1995) *Epigenetic Inheritance and Evolution: the Lamarckian Dimension*, Oxford University Press
- Kauffman S. (1993) *The origins of order: Self organization and selection in evolution*. Oxford.
- Kimura, M. (1985) *The neutral theory of molecular evolution*. Cambridge.
- Kirchner, M. & Gerhart, J. (1998). Evolvability. *PNAS*. 95:8420–8427
- Kirkpatrick, S., Gelatt, C.D. & Vecchi, M.P. (1983) Optimization by Simulated Annealing. *Science*. 220 (4598): 671–680.
- Knuth, D.E. (1973) *The Art of Computer Programming, Vol. 3 - Sorting and Searching*. Addison-Wesley.
- Lenaerts, T., Chu, D., Watson, R.A. (2005) Dynamical hierarchies, *Artificial Life*. 11(4):403-405
- Lewontin, R.C. (1970) "The Units of Selection," *Annual Review of Ecology and Systematics*, 1: 1-18.
- Margulis, L. & Fester, R. (1991) *Symbiosis as a Source of Evolutionary Innovation*. Cambridge, MA: MIT Press
- Maynard Smith, J. & Szathmary, E. (1995) *Major Transitions in Evolution*. W. H. Freeman.
- Michod, R. E. & Hamilton, W. D. (1980) Coefficients of relatedness in sociobiology. *Nature* 288, 694 – 697.
- Michod, R.E., (1999), *Darwinian Dynamics, Evolutionary Transitions in Fitness and Individuality*. Princeton Univ. Press
- Mills, R.M. (2010) *How Micro-Evolution Can Guide Macro-Evolution*, PhD thesis, ECS, Southampton.
- Neander, K. (1995) Pruning the Tree of Life, *British Jnl. for the Philosophy of Sci.* 46(1): 59-80.
- Nowak MA, & May, R.M. (1992) Evolutionary Games and Spatial Chaos, *Nature* 359, 826-829.
- Odling-Smee, F. J., Laland, K. N., and Feldman, M. W. (2003). *Niche construction: the neglected process in evolution*. Monographs in population biology; no. 37. Princeton University Press
- Okasha, S. (2006). *Evolution and the Levels of Selection*. Clarendon.
- Parter, M. et al (2008) Facilitated Variation: How Evolution Learns from Past Environments to Generalize to New Environments. *PLoS Comput Biol* 4(11).
- Pigliucci M (2007) Do we need an extended evolutionary synthesis? *Evolution* 61(12): 2743-2749.
- Powers, S.T. (2010) *Social Niche Construction*, PhD thesis, ECS, Southampton.
- Powers, S.T., Penn, A.S., Watson, R.A. (2011) The Concurrent Evolution of Cooperation and the Population Structures that Support it. *Evolution*. 65(6):1527–1543.
- Shapiro, J. (2011) *Evolution: A View from the 21st Century*. FT Press.
- Snowdon, J., Powers, S. & Watson, R.A. (2009) Moderate contact between sub-populations promotes evolved assortativity enabling group selection. *Procs. 10th Euro. Conf. on Artl. Life (ECAL 2009)*. 2:42-49.
- Sober, E.(1984) *The nature of selection: evolutionary theory in philosophical focus*. Chicago.
- Sommer, R.J. (2009). The future of evo-devo: model systems and evolutionary theory. *Nature Reviews Genetics* 10 (6): 416–422
- Sterelny, K. (2011) "Evolvability Reconsidered" in *The Major Transitions in Evolution Revisited*, MIT Press. pp: 83-100.
- Szathmary, E. (2011) To Group or Not to Group? *Science*. 334. 1648-9.
- Toussaint, M., & von Seelen, W. (2007) Complex adaptation and system structure, *BioSystems* 90: 769–782
- Watson, R.A. (2005) On the Unit of Selection in Sexual Populations. In *Advances in Artificial Life, (ECAL 2005)*.
- Watson, R.A. (2006) *Compositional Evolution: The impact of Sex, Symbiosis and Modularity on the Gradualist Framework of Evolution*. MIT Press.
- Watson, R.A., Buckley, C.L., Mills, R.M. (2010) Optimisation in 'Self-modelling' Complex Adaptive Systems. *Complexity*. 16(5):17-26.
- Watson, R.A., Jackson A., Palmius, N., Mills, R.M., Powers, S.T. (submitted) *The Evolution of Symbiotic Partnerships and Their Adaptive Consequences*.
- Watson, R.A., Mills, R.M., Buckley, C.L. (2011a) Global Adaptation in Networks of Selfish Components. *Artificial Life*. 17(3):147-66.
- Watson, R.A., Mills, R.M., Buckley, C.L. (2011b) Transforming the Scale of Dynamical Behaviour in Nearly Decomposable Systems, *Adaptive Behaviour* 19(4): 227-249.
- Wilson D.S. (1992). Complex interactions in meta-communities, with implications for biodiversity and higher levels of selection. *Ecology*, 73(6):1984-2000.