Critical Focus

Time and Information in Evolution

Winston Ewert,[1*] William A. Dembski,[2] Ann K. Gauger,[3] Robert J. Marks II[1]

[1]Electrical & Computer Engineering, Baylor University, Waco, Texas, USA

[2]Discovery Institute, Seattle, Washington, USA

[3]Biologic Institute, Seattle, Washington, USA

Abstract

Wilf and Ewens argue in a recent paper that there is plenty of time for evolution to occur. They base this claim on a mathematical model in which beneficial mutations accumulate simultaneously and independently, thus allowing changes that require a large number of mutations to evolve over comparatively short time periods. Because changes evolve independently and in parallel rather than sequentially, their model scales logarithmically rather than exponentially. This approach does not accurately reflect biological evolution, however, for two main reasons. First, within their model are implicit information sources, including the equivalent of a highly informed oracle that prophesies when a mutation is "correct," thus accelerating the search by the evolutionary process. Natural selection, in contrast, does not have access to information about future benefits of a particular mutation, or where in the global fitness landscape a particular mutation is relative to a particular target. It can only assess mutations based on their current effect on fitness in the local fitness landscape. Thus the presence of this oracle makes their model radically different from a real biological search through fitness space. Wilf and Ewens also make unrealistic biological assumptions that, in effect, simplify the search. They assume no epistasis between beneficial mutations, no linkage between loci, and an unrealistic population size and base mutation rate, thus increasing the pool of beneficial mutations to be searched. They neglect the effects of genetic drift on the probability of fixation and the negative effects of simultaneously accumulating deleterious mutations. Finally, in their model they represent each genetic locus as a single letter. By doing so, they ignore the enormous sequence complexity of actual genetic loci (typically hundreds or thousands of nucleotides long), and vastly oversimplify the search for functional variants. In similar fashion, they assume that each evolutionary "advance" requires a change to just one locus, despite the clear evidence that most biological functions are the product of multiple gene products working together. Ignoring these biological realities infuses considerable active information into their model and eases the model's evolutionary process.

Cite as: Ewert W, Dembski WA, Gauger AK, Marks II RJ (2012) Time and information in evolution. BIO-Complexity 2012 (4):1–7. **doi:**10.5048/BIO-C.2012.4.

Editor: David Snoke

Received: September 19, 2012; **Accepted:** October 15, 2012; **Published:** December 7, 2012

Notes: A *Critique* of this paper, when available, will be assigned **doi:**10.5048/BIO-C.2012.4.c.

* Email: evoinfo@winstonewert.com

INTRODUCTION

In a paper published in the *Proceedings of the National Academy of Sciences* [1], Wilf and Ewens ask a relatively simple question. Consider a biological change that requires some set of mutations to occur. The evolutionary process necessarily requires time to accumulate these mutations because each individual mutation is typically improbable. The question is whether or not there is enough time to allow these events to occur. Rephrased in algorithmic terms, the question becomes whether the time needed for success has exponential or logarithmic dependence on the number of required mutations.

A number of recent papers have attempted to determine the evolutionary waiting time for complex adaptations, using a variety of mathematical and/or computational methods [2–7], and arriving at widely different answers. At least part of the reason for the disagreement is the complexity of the problem. Numerous biological variables affect the waiting time for new adaptations, including the population dynamics of the organism in question, the shape of the fitness landscape for the trait(s) in question, the number of mutations required, the effect of epistatic interactions, pleiotropy and linkage on the ability to select for a given trait, and the duration and direction of selection in shaping adaptation (for a review of various models see [8,9]). The contribution of Wilf and Ewens suffers first of all from a lack of engagement with the literature on these points.

Description of the model

The evolutionary model that Wilf and Ewens have chosen is similar to the problem of guessing letters in a word or phrase, as on the television game show *Wheel of Fortune*. They specify a phrase 20,000 letters long, with each letter in the phrase corresponding to a gene locus that can be transformed from its initial "primitive" state to a more advanced state. Finding the correct letter for a particular position in the target phrase roughly corresponds to finding a beneficial mutation in the corresponding gene. During each round of mutation all positions in the phrase are subject to mutation, and the results are selected based on whether the individual positions match the final target phrase. Those that match are preserved for the next round.

Wilf and Ewens are not the first to use a language-inspired model. In *The Blind Watchmaker* [10] Richard Dawkins proposes a model where the program evolves a phrase whose distance from the Shakespearean phrase METHINKS IT IS LIKE A WEASEL is measured after each round.[1] Dawkins himself acknowledges in the book that a search that includes information about the long-range target, as does his, is unrealistic, but he still makes the claim that a cumulative evolutionary search, reiterated many times with selection based on fitness after each query, can find the target quickly.

To what biological process does a query (one round of guessing) correspond in Wilf and Ewens's model? How long does one query take? The paper does not explicitly answer these questions. From the construction of the model it appears that a query corresponds to a period in which mutations occur in all individuals in a population, followed by an additional unspecified period to allow the beneficial mutations to reach fixation. Wilf and Ewens specify humans as the model organism, stating that 100–200 new mutations occur in each newborn human, of which only about five can be expected to be in either protein-coding or regulatory regions, and thus have some effect on the individual's phenotype. All other mutations are assumed to occur in junk DNA[2] and thus have no effect. Assuming a million births during each mutation period, they expect approximately five million non-junk mutations per round. Taking the human genome to have 20,000 genes, they expect approximately 250 mutations to occur somewhere in the population in each gene in each round. They assume that 1 in 10,000 of these mutations is beneficial, implying that the probability of a particular gene experiencing a beneficial mutation somewhere in the population during any given round is 1 in 40.

The only number in the above assumptions for which a reference is cited is the number of *de novo* mutations carried by newborn humans, which is itself a matter of debate [14,15]. The other numbers (proportion of junk DNA, number of genes, birth rate (generation size), and rate of beneficial mutations) are ballpark estimates at best. No literature is cited to support them.

Wilf and Ewens set the phrase length for their model at L=20,000 letters and the size of the alphabet at K=40 letters. Choosing a correct letter from that alphabet for a given position in the phrase corresponds to finding a beneficial ("advanced") allele at the corresponding locus. The number of letters in the alphabet was chosen to reflect the supposed 1 in 40 probability of a beneficial allele arising at any given locus during any round (see above). After each round, all "advanced" alleles in the population are treated as fixed, and therefore preserved in the next round. Evolution to the fully "advanced" state is complete when all 20,000 positions match the target phrase.

ANALYSIS

Time required for evolution by the model

Although the title of Wilf and Ewens's paper indicates that they believe they have solved a potential problem with the pace of evolution, they nowhere state an actual amount of time that their model would require. They only indicate that it will take 390 rounds of guessing to find the target phrase for the particular scenario they consider, presumably supposing these rounds to be short enough for this to require less than the available time. However, while they mention the fact that beneficial mutations are not necessarily fixed once they appear, they make no attempt to factor this in. Because of genetic drift, beneficial mutations are often lost before they can become established in the population. In fact, the probability of fixation for a beneficial mutation is roughly equal to $2sN_e$ / N, where N_e is the effective population size, N the census population size, and s is the selection coefficient for that mutation [16]. For humans this translates to a probability of fixation for a strongly beneficial mutation of ≤ 0.01 [5]. Consequently a beneficial mutation may have to arise a hundred times or more in a human population before it becomes established and goes on to fixation.

Furthermore, real biological genes, as opposed to the single-letter loci in Wilf and Ewens's model, are composed of stretches of DNA hundreds or thousands of nucleotides long, and linked linearly into very long strings (chromosomes). Recombination between chromosomes does occur between generations, but genes close to one another tend not to recombine, and end up remaining associated with one another. This genetic linkage slows the process by which sorting and fixing of beneficial mutations occurs (getting all the letters in the population to be the same beneficial type), and can lead to competition between closely linked beneficial mutations that prevents the fixation of one or both [17].

Type of search in the model

The Wilf and Ewens model is, as mentioned above, reminiscent of the game show *Wheel of Fortune.* One technique for finding the hidden phrase in *Wheel of Fortune* is to guess the entire phrase and continue guessing phrases until arriving at the correct phrase. Because all positions in the phrase must be correctly guessed at once, this technique scales exponentially with respect to the length of the phrase, meaning that every additional letter added to a phrase multiplies the search time by some constant factor. For example, obtaining a single letter might take 1 year, whereas obtaining 2 letters would take

[1] We have critiqued this program elsewhere [11].

[2] Much of what was previously described as junk has now been shown to be ubiquitously transcribed and regulated, and is increasingly believed to serve important functions in the cell. [12,13]

10 years, three letters would take 100 years, and so on. One can see that exponential scaling quickly gets out of hand and renders even an intermediate-sized phrase un-guessable. In the case of guessing an English phrase, every English letter added to the phrase increases the time to guess the phrase by 26 times (27 times if the space character is included in the alphabet).

A better technique is to make use of information gleaned during the guesses—this is the procedure actually used on the show. After guessing each of the letters, the oracle tells which (if any) of the guessed letters are correct, and then those letters are retained. The second round of guessing is applied only for the incorrect letters that remain after the first round, and so forth [1]. This kind of search has been dubbed *partitioned search* [18] and has, from Wilf and Ewens, a probability of success, q, in Q queries of

$$q = \left(1 - \left(1 - \frac{1}{K}\right)^{Q}\right)^{L},$$

where K is the number of characters in the alphabet and L is the length of the target message. The game of *Yahtzee* follows essentially the same process, with the player selectively re-rolling any dice that do not have the numbers that are needed. Re-rolling only the incorrect dice rather than all of the dice should reach the goal much more quickly.

A partitioned search, such as the one proposed by Wilf and Ewens, scales logarithmically, meaning that the time required for success is proportional to the logarithm of L. As a result, each additional locus adds less time than the one before it. In terms of Wilf and Ewens's search, finding a beneficial variant at one locus might take one year, at ten loci two years, and at one hundred loci only three years. Clearly, the number of genes to be changed makes relatively little difference according to this model.

In general, more efficient searches require richer sources of active information. In the case of the exponential algorithm, each guess is, in effect, being presented to a so-called *needle in a haystack* oracle that announces either "yes, this is the phrase," or "no, this is not it." The *partitioned search* oracle used by Wilf and Ewens, on the other hand, examines each guess letter by letter and tells us which of the letters are correct. Thus, Wilf and Ewens's partitioned search has access to more information than the exponential algorithm.

Determining the amount of active information used by the model

In order to identify the sources of active information implicitly included in Wilf and Ewens's algorithm, we analyze their model of evolution[3] as an attempt to find a hidden string, remembering that this is intended as an analogy for locating beneficial mutations. Our method of analysis depends on the fact that any technique used to find the phrase is going to have to make a number of guesses or queries. We can measure and compare the efficiency of different searches by determining the number of queries required for success [11,19–22].

[3] The methods used in this analysis are derived from those described previously [11,19–22].

The total active information that must be added to a search in order for it to succeed is:

$$I_+ = -\log \frac{p}{q}\ , \tag{1}$$

where q is the overall probability of that search succeeding and p is the probability of complete success in a single query [18]. That single-query success is calculated simply as:

$$p = K^{-L}. \tag{2}$$

All the searches considered here eventually succeed, which means that q = 1. By substituting our values for p and q into Equation 1, we therefore obtain:

$$I_+ = L \log K\,. \tag{3}$$

However, since some search algorithms will tend to need more queries to find the target than others, we can differentiate their performance by calculating the active information per query. This is calculated by dividing the active information by $E[Q]$, the expected number of queries needed for success.

$$I_\oplus = \frac{L \log K}{E[Q]} \tag{4}$$

We now examine $I_\oplus$ for the two algorithms under consideration.

Exponential Search. First, let us consider the exponential search discussed by Wilf and Ewens, where each query aims to guess the entire phrase at once. This approach will eventually succeed but will take a vast amount of time to do so. The chance of identifying a phrase in Q_x queries sampling with replacement [1,18] is:

$$q_x = 1 - \left(1 - \left(\frac{1}{K}\right)^{L}\right)^{Q_x} \tag{5}$$

The mean of random variable Q_x is:

$$E[Q_x] = K^L. \tag{6}$$

Using Equation 4, this gives us the active information per query as:

$$I_\oplus = \frac{L \log K}{K^L}. \tag{7}$$

This is plotted in Figure 1 as a function of phrase length L for an alphabet of size K = 40. As the plot shows, repeated queries to the *needle in a haystack* oracle give little active information per query and the amount of active information declines rapidly as the phrase gets longer.

The linear nature of Figure 1 can be explained by taking the log of Equation 7,

$$\log I_\oplus = \log \frac{L \log K_2}{K^L} = \log(L \log K) - L \log K \approx O(-L \log K) \tag{8}$$

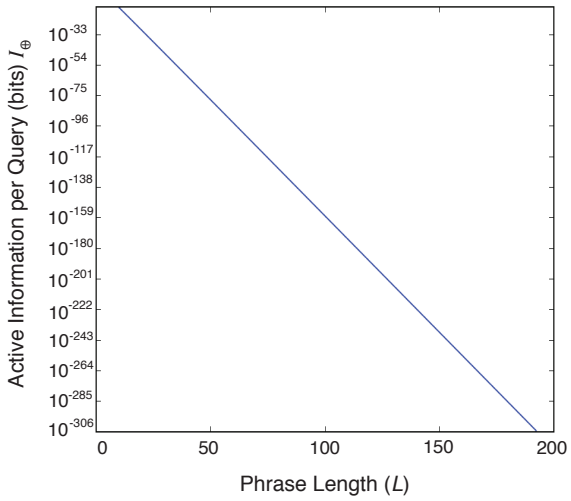

Figure 1: Effect of phrase length on active information per query for the exponential search algorithm. The active information per query has units of bits. The length of the phrase, L, varies from 10 to 200. The number of letters in the alphabet is set at 40. **doi:**10.5048/BIO-C.2012.4.f1

where ≈ signifies an order-of-magnitude approximation to equality. Since K is constant, this confirms the linear dependence of log $I_\oplus$ on L with slope –log K.

Logarithmic Search. The expected number of queries in order to determine the correct phrase using Wilf and Ewens's partitioned search is approximately $K \log L$ [1]. Using Equation 4, we can thus calculate the active information per query as:

$$I_\oplus = \frac{L \log K}{K \log L} \ . \qquad (9)$$

A plot of this function can be seen in Figure 2. It shows much more active information per query than the exponential algorithm. Although visual comparison with the plot in Figure 1 is difficult because Figure 1 uses a log scale while Figure 2 uses a linear scale, the numbers indicate over 32 orders of magnitude difference in the active information per query even for short phrases.

Assuming independence

The basic difference between the exponential and partitioned search is that in the partitioned search each letter in the phrase can be queried and announced independently and simultaneously, preserving the correct answers after each query, while in the exponential search, getting the correct answer is contingent on getting all positions correct at once. In effect, the partitioned search treats each letter independently, while the exponential search treats them as an interdependent set. This difference is what allows the partitioned search to proceed so rapidly.

In terms of evolution, though, is the assumption of independence valid? In natural language phrases, where Wilf and Ewens choose to model the evolutionary process, the assumption is clearly not valid. Suppose it would be beneficial for the phrase

"all_the_world_is_a_stage___"

to evolve into the phrase

"methinks_it_is_like_a_weasel."

What phrase do we get if we simply alternate letters from the two phrases?

"mlt_ihk__otli__siaesaaw_a_e_."

Under the assumptions in the Wilf and Ewens model, the "fitness" of this nonsense phrase ought to be exactly half-way between the fitnesses of "all the world is a stage" and "methinks it is like a weasel." Such a result only makes sense if we are measuring the fitness of the current phrase by its proximity to the target phrase.

This example reveals two biologically unrealistic things about Wilf and Ewens's model. First, evolutionary processes can only depend on the performance of current organisms, not hypothetical target organisms. Only teleological processes have the ability to consider future phrases. Yet Wilf and Ewens's oracle has to have knowledge of the target to assess fitness. Second, the natural language example also highlights the importance of context for assessing fitness, if we think of fitness as the ability to convey meaning in a natural language. As we shall see, this same phenomenon of context dependence is typical of biological systems.

Testing the effect of independence

In order to demonstrate the importance of the assumption of independence for their model, we can modify the model to decrease the degree of independence, simply by requiring that some mutations must occur together. Thus the modified model will only reward complete words rather than individual letters in the phrase. We expect that it will show less active information per query simply because it is harder to guess complete words than individual letters. This modified model is not intended to be biologically realistic. Rather it is intended solely to show the effect that the assumption of independence has on the search.

For simplicity, we assume that every word in the phrase has the same length, W. Because entire words have to be guessed, we can view that as guessing letters from a larger alphabet, one which is K^W in size. The length of the message in terms of this larger alphabet is thus L/W. Using this in Equation 9 we obtain:

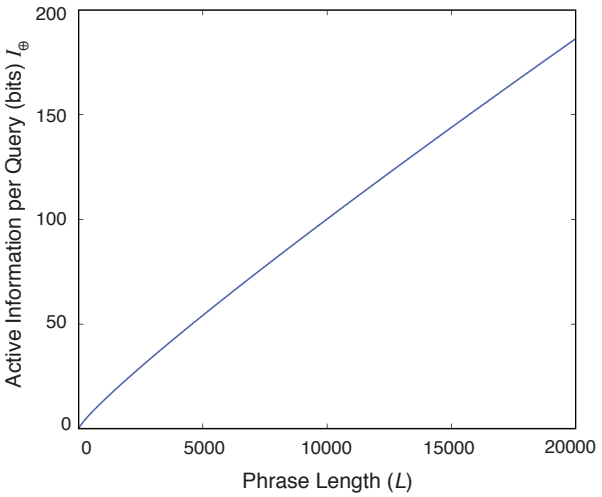

Figure 2: Effect of phrase length on active information per query in partitioned search. The length of the phrase, L, varies from 10 to 20,000. The number of letters in the alphabet is set at $K = 40$. **doi:**10.5048/BIO-C.2012.4.f2

$$I_{\oplus} = \frac{L \log K}{K^W \log \frac{L}{W}} \,. \qquad (10)$$

As seen in Figure 3 (a plot of Equation 10), this equation shows a very rapid decline in active information per query as the amount of independence is decreased. This demonstrates that even a small amount of dependency between the mutations causes a very sharp decline in the information extracted and thus a large increase in the time required to reach the desired phrase.

The Wilf and Ewens model suggests that it will take approximately 390 rounds of guessing to find 20,000 beneficial mutations. However, even the smallest level of dependence, with word lengths W=2, rapidly increases the time required, and that time increases exponentially as the word length increases.

DISCUSSION

If we provisionally grant the analogy proposed by Wilf and Ewens, which likens the evolution of more advanced genomes to a search for a phrase, then the question is whether their search model is fair, in the sense of not being rigged for success. The theory of the conservation of information [23–26], which is often associated with the No Free Lunch Theorems [27], shows that all search algorithms have the same performance when averaged across all search problems. It follows that algorithms that significantly outperform others in specific applications do so because of active information that has been introduced by the programmer. In the case of Wilf and Ewens's model, active information has indeed been introduced by the assumptions built into their model.

Most significant in this regard is their assumption that benefit can be acquired at each locus (letter position) independently. Each letter in the phrase functions and sorts independently, like Scrabble tiles drawn from a bag. With each round, every letter in the phrase that has not yet matched the target receives a new draw from the bag, and is assessed for a match without regard for the surrounding context. In effect, Wilf and Ewens have incorporated an information source in their model by assuming that natural selection operates as an "in parallel" process, with beneficial mutations at each gene locus being selected and retained independently of one another [1].

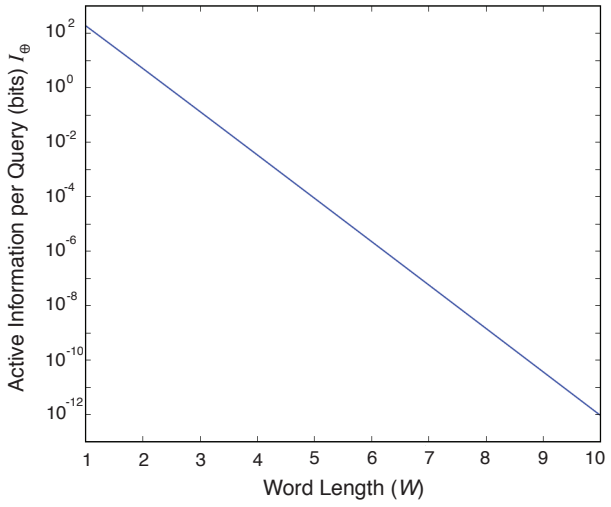

Figure 3: Plot of active information per query for varying phrase lengths given the modified model. The length of the phrase, L, is 20,000. The number of letters in the alphabet is set at 40. W is the length of the words in the phrases being guessed. **doi:**10.5048/BIO-C.2012.4.f3

The overall fitness of individual genomes is assessed after each round, with more matches meaning higher fitness and therefore preservation for the next round. This is equivalent to assuming not just that all loci can be the sites of beneficial change independently, but also that these independent benefits are additive. As we will see, both of these assumptions fail to square with biology.

Unrealistic biology

However, before we consider these problems with the way Wilf and Ewens have constructed their search, we should take a step back to examine the analogy they offer as the context for their search, since a problem here would have the most comprehensive implications for their work. The very suggestion that the state of a particular genetic locus should be likened to the identity of letter at a particular position in a phrase is peculiar. Letters are the building blocks for written communication. They are indivisible elements that come in small sets called alphabets. Some letters are used more than others, but each finds use. Contrast this with what a genetic locus actually is. Each particular locus in a genome is composed of hundreds or thousands of nucleotide bases arranged in sequence. The internal complexity of each sequence, is so great that the number of sequence possibilities is staggeringly large—*far* too large for anything but an infinitesimal fraction ever to have existed. Furthermore, as Axe has shown experimentally [28], only a tiny fraction of those sequences encode proteins capable of folding into the three-dimensional structures necessary for biological functions. To liken the actual complexity of a genetic locus to one letter in an alphabet is to fail to grasp how profoundly different the two things are.

And the difference matters. For evolutionary "advancement" to occur, it must be possible for new functions or structures to arise by a step-wise process entailing very few mutations, or mutations that occur at high frequency in the population. Wilf and Ewens assume 1 out of every 10,000 mutations is beneficial, and that a single mutation is all that is necessary for that beneficial change. This might be the case if genetic loci were composed of single letters as in their model, but since that they are not, we need a realistic estimate of how many changes are required for a new trait or function to arise in a locus.

Converting an enzyme to a new function is the kind of thing that should have occurred thousands of time in the course of evolution, given the vast array of biochemical functions carried out by extant enzymes. Yet recent work has shown that converting an enzyme encoded by a 1,200-nucleotide gene to a genuinely new function[4] is likely to require seven or more coordinated mutations. This is true even though the starting and target enzymes have common three-dimensional protein

[4] A genuinely new enzymatic function would be a new reaction for which the starting enzyme has no activity.

folds and active-site chemistries— just no shared reaction [29].[5] Getting seven specific changes in a gene 1,200 nucleotides long is a 1-in-10^{22} event, not a 1-in-10,000 event. Even then it is by no means clear that significant changes in gene function can be had with just seven base substitutions.

But by far the most important problem with the assumption of independence that Wilf and Ewens make is the fact that genes don't act alone—most biochemical functions in cells involve multi-step pathways. Making biotin requires four dedicated genes, for example. Yet without biotin the cell cannot make membranes. One gene is not enough to make tryptophan, build a ribosome, or transcribe a gene. One gene is not enough to replicate DNA. Getting a genuinely new function or structure is likely to require multiple changes to multiple genes, or the development of whole new genes. Such a thing is many orders of magnitude more difficult than re-engineering an enzyme.

Wilf and Ewens have assumed the pre-existence of a model organism of 20,000 functioning loci, each of which is represented by a single letter. They further stipulate that beneficial mutations arise 250 times in every generation, with each beneficial change requiring just a single mutation to a single locus, and acting additively with the rest. Finally, they assume that all beneficial mutations fix the first time they appear. This kind of model in no way represents biological evolution.

Unrealistic search

In addition to the overwhelming problems mentioned above, the search algorithm they have chosen is unrealistic. Wilf and Ewens assume that the fitness landscape is smooth, with each beneficial mutation trending upward additively. This is not the case in biology. The phenotypic effect of mutations in two genes *individually* may be positive, but the phenotype of the two mutations *combined* may be negative. This epistasis, as it is called, can occur between different genes, or between mutations in a single gene [32–34]. A set of mutations in one context can be beneficial, but in another context deleterious or even lethal. In fact, research suggests that epistasis causes diminishing returns among beneficial mutations [35,36]. Therefore, the genetic context in which new mutations arise matters, and fitness landscapes may be rough and highly constrained.

Indeed, there is much evidence to suggest that real fitness landscapes have many local fitness optima surrounded by fitness deserts [31,37,38]. If it takes more than several mutations to move from one peak to another, adaptation can become stalled on a local peak, with no way to move from one small fitness peak to a higher one. Because natural selection is blind and without foresight, it cannot tell which particular mutations are leading to an unrealized goal of maximal fitness (in this case a target phrase) some distance away in the adaptive landscape. It can only assess the relative local fitness of variants in the population.

It is also important to remember that, in terms of natural selection, "beneficial" means any mutation that leads to greater reproduction, without regard to long-term goals. Under many conditions, that can mean that any mutation that increases efficiency, including mutations that inactivate or delete genes, will be favored [39–41]. Even though genetic information is being lost, such mutations would still be considered "beneficial." That loss of information is a common evolutionary outcome has been shown repeatedly [41]. Wilf and Ewens's model, however, assumes that all beneficial mutations lead to more and more "advanced" forms.

SUMMARY

Wilf and Ewens's evolutionary algorithm works quickly because of the information sources they provide to the search, in the form of a target phrase that is matched to each round of guessing. This search process itself creates no information. Rather, it mines information from the sources available in the search algorithm. Other examples of evolutionary algorithms where this is the case include AVIDA [2,20] and *ev* [3,22].

In addition, Wilf and Ewens present a model of biological evolution that is completely unrealistic. Their model vastly underestimates the number of mutations required to achieve an adaptation at each locus, by compressing to a single-letter change what should be an enormous search through sequence space for functional variants. They also completely ignore the fact that most adaptations are multi-locus features, and neglect to consider variables such as generation time, population size, the time required for fixation of mutations, the confounding effects of epistatic interactions between mutations, and the shape of the fitness landscape [42–44]. Because of these problems, their conclusion that there's plenty of time for evolution is unwarranted.

[5] Others who have tried to engineer enzymes to new functions have reported similar difficulties [30,31].

1. Wilf HS, Ewens WJ (2010) There's plenty of time for evolution. P Natl Acad Sci 107:22454–22456. **doi:**10.1073/pnas.1016207107
2. Lenski RE, Ofria C, Pennock RT, Adami C (2003) The evolutionary origin of complex features. Nature 423:139–144. **doi:**10.1038/nature01568
3. Schneider TD (2000) Evolution of biological information. Nucleic Acids Res 28:2794–2799. **doi:**10.1093/nar/28.14.2794
4. Behe MJ, Snoke DW (2004) Simulating evolution by gene duplication of protein features that require multiple amino acid residues. Prot Sci 13:2651–2664. **doi:**10.1110/ps.04802904
5. Durrett R, Schmidt D (2008) Waiting for two mutations: With applications to regulatory sequence evolution and the limits of Darwinian evolution. Genetics 180:1501–1509. **doi:**10.1534/genetics.107.082610
6. Lynch M, Abegg A (2010) The rate of establishment of complex adaptations. Mol Biol Evol 27:1404–1414. **doi:**10.1093/molbev/msq020
7. Axe DD (2010) The limits of complex adaptation: An analysis based on a simple model of structured bacterial populations. BIO-Complexity 2010(4):1–10. **doi:**10.5048/BIO-C.2010.4

8. Orr HA (2005a) The genetic theory of adaptation: A brief history. Nat Rev Genet 6: 119–127. **doi:**10.1038/nrg1523
9. Orr HA (2005b) Theories of adaptation: What they do and don't say. Genetica 123: 3–13. **doi:**10.1007/1-4020-3836-4_2
10. Dawkins R (1996) The Blind Watchmaker: Why the Evidence of Evolution Reveals a Universe Without Design. Norton (New York).
11. Ewert W, Montañez G, Dembski WA, Marks II RJ (2010) Efficient per query information extraction from a Hamming oracle. 42nd Southeast Symp Syste: 290–297. **doi:**10.1109/SSST.2010.5442816
12. Wells, J (2011) The Myth of Junk DNA. Discovery Institute Press (Seattle, Washington).
13. The ENCODE Project Consortium (2012) An integrated encyclopedia of DNA elements in the human genome. Nature 489:57–74. **doi:**10.1038/nature11247
14. Lynch M (2010) Rate, molecular spectrum, and consequences of human mutation. P Natl Acad Sci 107:961–968. http://www.pnas.org/content/107/3/961.full.pdf
15. Kong A, Frigge ML, Masson G, Besenbacher S, Sulem P, et al. (2012) Rates of *de novo* mutations and the importance of father's age to disease risk. Nature 448:471–475. **doi:**10.1038/nature11396
16. Patwa Z, Wahl LM (2008) The fixation probability of beneficial mutations. JR Soc Interface 5:1279–1289. **doi:**10.1098/rsif.2008.0248
17. Lynch M (2006) The origins of eukaryotic gene structure. Mol Biol Evol 23:450–468. **doi:**10.1093/molbev/msj050
18. Dembski WA, Marks II RJ (2009) Conservation of information in search: Measuring the cost of success. Trans Sys Man Cyber Part A 39(5):1051–1061. **doi:**10.1109/TSMCA.2009.2025027
19. Dembski WA, Marks RJ (2009) Bernoulli's principle of insufficient reason and conservation of information in computer search. IEEE Sys Man Cybern San Antonio Oct 11–14:2647–2652. **doi:**10.1109/ICSMC.2009.5346119
20. Ewert W, Dembski WA, Marks II RJ (2009) Evolutionary synthesis of nand logic: Dissecting a digital organism. IEEE Sys Man Cybern San Antonio Oct 11–14:3047–3053. **doi:**10.1109/ICSMC.2009.5345941
21. Dembski WA, Marks II RJ (2010) The search for a search: Measuring the information cost of higher level search. Journal of Advanced Computational Intelligence and Intelligent Informatics 14(5):475–486.
22. Montañez G, Ewert W, Dembski WA, Marks II RJ (2010) A vivisection of the **ev** computer organism: Identifying sources of active information. BIO-Complexity 2010(3):1–6. **doi:**10.5048/BIO-C.2010.3
23. Dembski WA (2002) No Free Lunch: Why Specified Complexity Cannot Be Purchased without Intelligence. Rowman & Littlefield (Lanham, MD).
24. Dembski WA, Marks II RJ (2010) The Nature of Nature. ISI Books (Wilmington, DE).
25. Mitchell TM (1980) The need for biases in learning generalizations. Technical report, Department of Computer Science, Rutgers University.
26. Schaffer C (1994) A conservation law for generalization performance. In: Cohen WW and Hirsch H, eds. Proceedings of the Eleventh International Machine Learning Conference. Rutgers University (New Brunswick, NJ), 259–265.
27. Wolpert D, Macready W (1997) No free lunch theorems for optimization. IEEE T Evolut Comput 1:67–82. **doi:**10.1109/4235.585893
28. Axe DD (2004) Estimating the prevalence of protein sequences adopting functional folds. J Mol Biol 341:1295–1315. **doi:**10.1016/j.jmb.2004.06.058
29. Gauger AK, Axe DD (2011) The evolutionary accessibility of new enzyme functions: a case study from the biotin pathway. BIO-Complexity 2011(1):1–17. **doi:**10.5048/BIO-C.2011.1
30. Gerlt JA, Babbitt PC (2009) Enzyme (re)design: Lessons from natural evolution and computation. Curr Opin Chem Biol 13(1):10–18. **doi:**10.1016/j.cbpa.2009.01.014
31. Romero and Arnold (2009) Exploring protein fitness landscapes by directed evolution. Nat Rev Mol Cell Biol 10:866–875. **doi:**10.1038/nrm2805
32. Salverda MLM, Dellus E, Gorter FA, Debets AJM, van der Oost J, Hoekstra RF, Tawfik DS, de Visser JAGM (2011) Initial mutations direct alternative pathways of protein evolution. PLoS Genet 7(3):e1001321. **doi:**10.1371/journal.pgen.1001321
33. Lunzer M, Golding GB, Dean AM (2010) Pervasive cryptic epistasis in molecular evolution. PLoS Genet 6(10):e1001162. **doi:**10.1371/journal.pgen.1001162
34. Cooper TF, Remold SK, Lenski RE, Schneider D (2008) Expression profiles reveal parallel evolution of epistatic interactions involving the CRP regulon in Escherichia coli. PLoS Genet 4(2):e35. **doi:**10.1371/journal.pgen.0040035.
35. Khan AI, Dinh DM, Schneider D, Lenski RE, Cooper TF (2011) Negative epistasis between beneficial mutations in an evolving bacterial population. Science 332:1193–1196. **doi:**10.1126/science.1203801.
36. Chou HH, Chiu HC, Delaney NF, Segrè D, Marx CJ (2011) Diminishing returns epistasis among beneficial mutations decelerates adaptation. Science 332:1190–2. **doi:**10.1126/science.1203799
37. Weinreich DM, Watson R, Chao L (2005) Perspective: Sign epistasis and genetic constraint on evolutionary trajectories. Evolution 59:1165–1174.
38. Hayashi Y, Aita T, Toyota H, Husimi Y, Urabi I, et al. (2006) Experimental rugged fitness landscape in protein sequence space. PLoS One **1:**e96. **doi:**10.1371/journal.pone.0000096
39. Gauger AK, Ebnet S, Fahey PF, Seelke R (2010) Reductive evolution can prevent populations from taking simple adaptive paths to high fitness. BIO-Complexity 2010(2):1–9. **doi:**10.5048/BIO-C.2010.2
40. Lang GI, Murray AW, Botstein D (2009) The cost of gene expression underlies a fitness trade-off in yeast. Proc Natl Acad Sci USA 106:5755–5760. **doi:**10.1073/pnas.0901620106
41. Behe M (2010) Experimental evolution, loss-of-function mutations, and "the first rule of adaptive evolution." Q Rev Biol 85: 419–445.
42. Sniegowski PD, Gerrish PJ (2010) Beneficial mutations and the dynamics of adaptation in asexual populations. Philos T Roy Soc B 365:1255–1263. **doi:**10.1098/rstb.2009.0290
43. Takahata N, Satta Y, Klein J (1995) Divergence time and population size in the lineage leading to modern humans. Theor Popul Biol 48:198–221. **doi:**10.1006/tpbi.1995.1026
44. Yang Z (2002) Likelihood and Bayes estimation of ancestral population sizes in hominoids using data from multiple loci. Genetics 162:1811–1823.