Critical Focus

Digital Irreducible Complexity: A Survey of Irreducible Complexity in Computer Simulations

Winston Ewert*

Biologic Institute, Redmond, Washington, USA

Abstract

Irreducible complexity is a concept developed by Michael Behe to describe certain biological systems. Behe claims that irreducible complexity poses a challenge to Darwinian evolution. Irreducibly complex systems, he argues, are highly unlikely to evolve because they have no direct series of selectable intermediates. Various computer models have been published that attempt to demonstrate the evolution of irreducibly complex systems and thus falsify this claim. However, closer inspection of these models shows that they fail to meet the definition of irreducible complexity in a number of ways. In this paper we demonstrate how these models fail. In addition, we present another designed digital system that *does* exhibit designed irreducible complexity, but that has not been shown to be able to evolve. Taken together, these examples indicate that Behe's concept of irreducible complexity has not been falsified by computer models.

Cite as: Ewert W (2014) Digital irreducible complexity: A survey of irreducible complexity in computer simulations. BIO-Complexity 2014 (1):1–10. **doi:**10.5048/BIO-C.2014.1.

Editor: Ola Hössjer

Received: November 16, 2013; **Accepted:** February 18, 2014; **Published:** April 5, 2014

Notes: A *Critique* of this paper, when available, will be assigned **doi:**10.5048/BIO-C.2014.1.c.

*evoinfo@winstonewert.com

INTRODUCTION

The concept of irreducible complexity was introduced by Michael Behe in his book, *Darwin's Black Box* [1]. A system is irreducibly complex if it is:

> a single system composed of several well-matched, interacting parts that contribute to the basic function, wherein the removal of any one of the parts causes the system to effectively cease functioning [1].

Behe illustrated his concept using a snap mousetrap, and then went on to argue that a number of biological systems such as the bacterial flagellum, blood clotting cascade, and mammalian immune system are irreducibly complex [1]. He also argued that irreducible complexity posed a serious challenge to a Darwinian account of evolution, since irreducibly complex systems have no direct series of selectable intermediates.

Irreducible complexity does not mean that irreducibly complex systems are logically impossible to evolve, though many have misunderstood the concept as making that claim [2]. This is not the case, and in fact Behe has said:

> Even if a system is irreducibly complex (and thus cannot have been produced directly), however, one cannot definitively rule out the possibility of an indirect, circuitous route. As the complexity of an interacting system increases, though, the likelihood of such an indirect route drops precipitously [1].

Behe argues that while logically possible, Darwinian explanations of irreducibly complex systems are improbable. Extensive arguments have been written about whether or not Darwinian evolution can plausibly explain irreducibly complex systems [2–8]. Ultimately, this is not a question that can be settled by argument. Instead, it should be settled by experimental verification.

In order to experimentally verify whether or not irreducibly complex systems can be evolved, we need to observe evolution on a very large scale. The *E. coli* experiments of Richard Lenski have begun to meet this criterion [9,10]. Lenski and co-workers have observed the evolution of *E. coli* bacteria in a laboratory under minimal growth conditions over the course of twenty-five years, totaling some 10^{13} bacteria. However, that experiment is dwarfed in size by the natural experiment of HIV evolution, with an estimated 10^{20} viruses over the past few decades, and by the even larger natural experiment of malarial evolution, representing some 10^{20} cells per year [11]. Even at these large scales evolution has not been observed to produce complex novel biological structures [11–13].[1] Certainly nothing of the complexity of a bacterial flagellum or blood clotting cascade has been observed.

Irreducibly complex systems in biological organisms are necessarily very complex, and thus would take more time to evolve than human observers can wait. This observational limitation has led some to turn in a new direction. Instead of attempting to evolve complex structures in biological experiments, they attempt to evolve complex systems or features in computer models.

A number of evolutionary models have been published that claim to demonstrate the evolution of irreducibly complex systems. These models are purported to evolve irreducible complexity in computer code [14], binding sites [15], road networks [16], closed shapes [17], and electronic circuits [18]. It is claimed that these models falsify the claims of Behe and other intelligent design proponents.

However, a closer look at these models reveals that they have failed to falsify the hypothesis because the systems they evolve fail to meet the definition of irreducible complexity. In some cases, the evolved systems fail to pass the knockout test, meaning they continue to function after parts have been removed. In almost all cases, the parts that make up the system are too simple. None of the models attempt to describe the functional roles of the parts. Finally, some of the models are clearly contrived in such a way that they are able to evolve new systems, demonstrating a designer's ability to design a system that can evolve, rather than one that reflects natural processes.

THE MODELS

Computer evolutionary models have existed for some time [19]. It is claimed that some of these models have demonstrated that Darwinian evolution can account for irreducibly complex systems. Sometimes the claim is made directly by the authors of the papers in which these models are presented; sometimes the claim is made not by the authors but by others who have commented on their models. This paper is concerned solely with the claims of evolved irreducible complexity, regardless of whether or not such claims were the primary intention of the creator of the model.

These models can be very complicated; however, for the purposes of determining whether a model can indeed evolve irreducibly complex systems, most of the precise details of the model are not directly relevant. Instead, we can focus on understanding the evolution of a system by looking at the intermediate evolutionary steps. Any complex system will require intermediate evolutionary steps in order to evolve.

Whether or not a complex system *can* evolve hinges on whether the fitness function rewards those intermediate steps. Thus, the key concept for each of these simulations is the fitness function, the rule by which it is decided which digital organisms are fitter than others. The fitness function determines the rewards and penalties assigned to the digital organisms. This rule is often carefully designed in order to help the model achieve its desired outcome.

For each of these models, we are primarily concerned with these two questions. What does the model reward and penalize? What are the intermediate steps leading to the allegedly irreducibly complex system? This work will focus on these questions. Readers interested in more technical details about these models should consult the original works.

Some of these models are biologically inspired. They are necessarily a much simplified version of biological reality. A system modeled after one that is irreducibly complex in biology may no longer fit the definition of irreducible complexity in a simplified form. Therefore, the model itself, not its original biological inspiration, must be evaluated to determine whether its products are truly irreducibly complex.

In the following sub-sections we will introduce each of the models briefly, before turning to a discussion of their flaws in the next section.

Avida

Avida [14] seeks to evolve a sequence of computer instructions in order to perform a calculation known as EQU. EQU is not a commonly known function, but it is in essence very simple. Given two strings of ones and zeros, write down a one for each bit in agreement and a zero for each bit in disagreement:

$$\begin{array}{c} 0\,1\,0\,0\,1\,0\,0\,0\,1\,1\,1\,0\,0\,0\,1\,0\,0\,0\,1\,1\,0\,1\,0\,1\,0\,0\,1\,0\,0\,0\,1\,1 \\ \underline{0\,0\,1\,0\,0\,0\,1\,0\,1\,1\,0\,1\,0\,0\,0\,1\,1\,0\,1\,1\,1\,0\,1\,0\,0\,1\,0\,1\,1\,0\,0\,1} \\ 1\,0\,0\,1\,0\,1\,0\,1\,1\,1\,0\,0\,1\,1\,0\,0\,0\,1\,1\,1\,0\,0\,0\,0\,1\,0\,0\,0\,0\,1\,0\,1 \end{array}$$

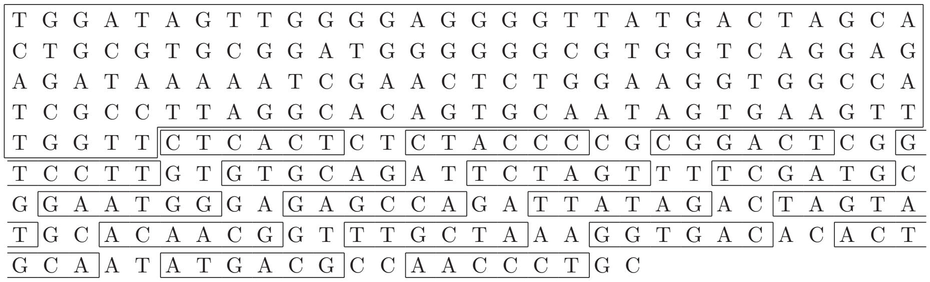
From a human perspective, this seems to be a rather trivial task. However, for the computer model Avida, the EQU function requires nineteen instructions, or separate steps. This is because in a computer each step must be trivially simple, requiring many steps to perform interesting tasks.

Avida begins with simple organisms that can evolve by inserting new instructions into their code. Sometimes those new instructions are able to perform a simple task. Avida rewards organisms that complete increasingly difficult tasks, according to a specified fitness function. The fitness function dictates bonuses that increase exponentially, according to the difficulty of the task accomplished. To evolve EQU requires the prior evolution of a number of these simpler tasks.

A simplified analogy for this process would be to "evolve" the ability of students to successfully calculate the area of a circle (πr^2) by giving an "A" to all students that calculate πr^2, a "B" to all students who calculate πr, and a "C" to all students that write down the value of π. Eventually the correct formula will be found by rewarding students that are closer to the goal with higher grades.

In the case of Avida, if the simpler functions are not rewarded,

[1] Lenski's experiments have produced a strain of *E. coli* that can metabolize citrate under aerobic conditions. This change to the strain is a fascinating innovation. However, because *E. coli* already has the ability to take up and metabolize citrate under anaerobic conditions, this innovation is simply the re-use of existing parts under different conditions. It is not an example of the de novo evolution of an irreducibly complex process. Upon sequencing the new strain it was found that, besides one or two permissive mutations such as the up-regulation of the pathway involved in citrate metabolism, the bacteria had amplified the gene involved in citrate transport, and placed it under the control of a new regulatory element, such that it was now expressed under aerobic conditions. No new complex adaptations or structures had been created; rather pre-existing ones were regulated differently [13].



```
T G G A T A G T T G G G G A G G G G T T A T G A C T A G C A
C T G C G T G C G G A T G G G G G G C G T G G T C A G G A G
A G A T A A A A A T C G A A C T C T G G A A G G T G G C C A
T C G C C T T A G G C A C A G T G C A A T A G T G A A G T T
T G G T T C T C A C T C T C T A C C C C G C G G A C T C G G
T C C T T G T G T G C A G A T T C T A G T T T T C G A T G C
G G A A T G G G A G A G C C A G A T T A T A G A C T A G T A
T G C A C A A C G G T T T G C T A A A G G T G A C A C A C T
G C A A T A T G A C G C C A A C C C T G C
```

Figure 1: A depiction of the Ev Genome. The large box represents the recognizer, which defines the rule for recognizing binding sites. The small boxes represent the binding sites themselves. **doi:**10.5048/BIO-C.2014.1.f1

Avida works less well or not at all [14,20]. The simplest functions require only a few steps to accomplish, with progressively more steps required for more complex functions. A visual depiction of the process of evolving the Avida program is available on the Evolutionary Informatics website.[2]

A paper on Avida did claim to be exploring the "evolutionary origin of complex features" [14]; however, the published research made no claims to have evolved irreducible complexity. Nevertheless, Robert T. Pennock, one of the paper's authors, testified at the Dover Trial:

> We can test to see, remove the parts, does it break? In fact, it does. And we can say here at the end we have an irreducibly complex system, a little organism this [sic] can produce this complex function.[3]

The parts in Avida are the individual steps in the process. If any of the steps in the process are missing, Avida will fail to calculate the EQU function. In this sense Pennock is correct, but we will discuss whether he is correct with respect to the other terms of Behe's definition.

Ev

The purpose of the model Ev [15] is to evolve binding sites. In biological terms, binding sites are the regions or places along the molecular structure of RNA or DNA where they form chemical bonds with other molecules. Ev models the evolution of these binding sites in predefined locations. In Ev the first part of the genome defines a recognizer that specifies a rule for recognizing binding sites; portions of the genome that follow are the binding sites themselves (see Figure 1). The rule is determined by the sequence of bases in the recognizer. In effect, the recognizer encodes a perceptron, which determines the set of patterns that will be recognized as binding sites. Every time that Ev is run, the recognizer evolves to specify a different rule, and requires completely different patterns in the binding sites. The rule, and thus the required bases, may even change during a single run.

Ev starts with a specified set of correct binding site locations and a random genome. A random genome is highly unlikely to contain binding sites in the correct locations. Over time the organisms may evolve binding sites in various locations, some of which match the correct locations, and others of which do not match one of the predefined locations. The organisms are rewarded for each binding site in a correct location and penalized for each binding site in an incorrect location. Thus a digital organism with six correctly located binding sites is considered better than a digital organism with only five correctly located binding sites. The intermediate stages of evolution are various organisms with an increasing number of correct binding sites. This process of evolution can be observed on Schneider's website[4] or the Evolutionary Informatics website.[5]

Schneider, the author of Ev, created Ev to demonstrate the evolution of information as measured by Shannon information theory. This paper is not concerned with that, the primary thesis of Schneider's work, but rather with his claim that Ev has demonstrated the evolution of irreducible complexity:

> This situation fits Behe's definition of 'irreducible complexity' exactly ... yet the molecular evolution of this 'Roman arch' is straightforward and rapid, in direct contradiction to his thesis [15].

Schneider views the recognizer and binding sites as the parts of an irreducibly complex system. Whether or not they are is something we will discuss.

Steiner trees

Dave Thomas presented his model as a genetic algorithm that evolves solutions to the Steiner tree problem [16], a problem that can be viewed as how to connect a number of cities by a road network using as little road as possible. In his model Thomas penalizes excess roads and disconnected cities; the fitness function assesses a small penalty for each length of road and a large penalty for leaving any city disconnected.

Thomas claims that his model can evolve an irreducibly complex system:

> And finally, two pillars of ID theory, "irreducible complexity" and "complex specified information" were shown not to be beyond the capabilities of evolution. [16]

He makes this claim because removal of any roads in Figure 2 disconnects the network, and makes it impossible to travel

[2] http://www.evoinfo.org/minivida/

[3] http://www.aclupa.org/files/5013/1404/6696/Day3AM.pdf

[4] http://schneider.ncifcrf.gov/toms/toms/papers/ev/evj/evjava/index.html

[5] http://evoinfo.org/ev/

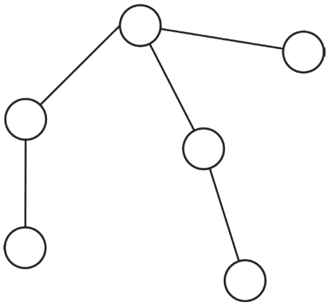


Figure 2: A depiction of a Steiner tree. The circles represent cities, and the lines, roads between the cities. **doi:**10.5048/BIO-C.2014.1.f2

between some of the cities. According to Thomas, the roads are therefore the parts of an irreducibly complex system. It should be noted, however, that obtaining a connected road network is actually trivial—a connected network can be achieved by random chance alone. A depiction of such a network can be seen in Figure 2. The difficulty in the Steiner tree problem is in trying to minimize the amount of road used [21], not in getting a connected network. Therefore we can say that there are no intermediate evolutionary stages in obtaining such a network.

Geometric model

Sadedin presented a geometric model of irreducible complexity [17]. This model operates on a triangular lattice (the shape in Figure 3 here is the same as figure 1A in Sadedin's paper). Each point of the lattice is either on or off. Black circles in the figure represent points that are turned on. If two adjacent points are turned on, they are connected, which is depicted by a solid line between the two points. If the lines form a closed shape, as in the figure, that is taken to be a "functional system."

Sadedin argues that these shapes are examples of irreducible complexity:

> Given these premises, the shape in Figure 1A meets Behe's criteria for an irreducibly complex system [17].

If any of the points were removed, there would be a hole in the side of the figure and it would no longer be closed. It would cease to function. Thus Sadedin argues she has generated an irreducibly complex system.

The model rewards increased area inside closed shapes and penalizes points being turned on. Due to the triangular shape of the lattice, turning on a nearby point will always be able to increase the closed area. The intermediate evolutionary stages increase the shape size by adding additional points, and removing points internal to the shape. The result is shapes enclosing a large area without any active internal points. Without any internal points, removing a single point on the edge will cause the entire shape to be open and thus entirely nonfunctional.

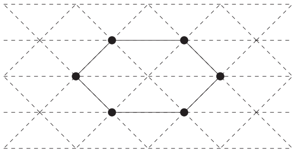


Figure 3: A redrawing of the shape in Figure 1A of Sadedin's model. **doi:**10.5048/BIO-C.2014.1.f3

Digital ears

Adrian Thompson ran a digital evolution experiment to evolve circuits that would distinguish between sounds at different frequencies [18]. It ran on an FPGA, which is a form of programmable hardware. An FPGA allows the use of custom circuits without physically building them. Instead, the FPGA can be configured to act like a large number of different circuits. An FPGA is built out of many cells. Each cell is simple, but can be programmed in a variety of ways. When many such cells are combined, complex behaviors can be exhibited.

Thompson makes no reference to irreducible complexity. Rather, Talk Origins, another website, makes the claim:

> However, it is trivial to show that such a claim is false, as genetic algorithms have produced irreducibly complex systems. For example, the voice-recognition circuit Dr. Adrian Thompson evolved is composed of 37 core logic gates. Five of them are not even connected to the rest of the circuit, yet all 37 are required for the circuit to work; if any of them are disconnected from their power supply, the entire system ceases to function. This fits Behe's definition of an irreducibly complex system and shows that an evolutionary process can produce such things.[6]

The "logic gates" mentioned in the quotation above should be called "cells." The thirty-seven cells represent the parts of the allegedly irreducibly complex system, which we shall discuss.

FLAWS IN CLAIMS OF IRREDUCIBLE COMPLEXITY

The previous section presented the various models that claim to demonstrate the computational evolution of irreducible complexity. This section discusses the various flaws in these models, according to the kind of error they make, and shows why they fail to demonstrate the evolution of irreducible complexity.

The knockout test

Behe's definition of irreducible complexity includes a knockout test. The system must effectively cease to function if essential parts are removed. Some parts of a system may be optional, i.e. their removal will degrade the performance of the system but the system will still continue functioning. Irreducible complexity is concerned with required parts. If these parts are required, the system will stop working when they are removed. A system is considered irreducibly complex if it has several well-matched, interacting parts that must be present for it to function.

Let us now consider the model Ev with respect to the knockout test. Removing one of the binding sites in Ev will leave the system functional—it will still recognize the rest of the binding

[6] http://www.talkorigins.org/faqs/genalg/genalg.html

sites. In fact, this is how the Ev system evolves, one binding site at a time. Schneider's justification for his claim that Ev's evolved genome is irreducibly complex states:

> First, the recognizer gene and its binding sites co-evolve, so they become dependent on each other and destructive mutations in either immediately lead to elimination of the organism[15].

It appears that Schneider has misunderstood the definition of irreducible complexity. Elimination of the organism would appear to refer to being killed by the model's analogue to natural selection. Given destructive mutations, an organism will perform less well than its competitors and "die." However, this is not what irreducible complexity is referring to by "effectively ceasing to function." It is true that in death, an organism certainly ceases to function. However, Behe's requirement is that:

> If one removes a part of a clearly defined, irreducibly complex system, the system itself immediately and necessarily ceases to function [8].

The system must cease to function purely by virtue of the missing part, not by virtue of selection.

Commercial twin jet aircraft are required to be able to fly with only one functional engine. This means that the aircraft would continue to function if one of the engines were lost, albeit at reduced capacity. However, the airline does not fly aircraft with only one functional engine for reasons of safety; if the remaining engine cannot be repaired the aircraft will be dismantled. Does this mean that an aircraft with only one engine ceases to function? No, the aircraft with only one engine still works even if the eventual consequence is the dismantling of the aircraft.

Irreducible complexity is not concerned with the organism's eventual fate but whether the molecular machine still has all necessary components in order to function. The question is whether the machine, in this case the binding site recognition, still retains some function. It is clear that it does. Removing one binding site will allow all of the remaining binding sites to still be recognized.

Now let us consider whether or not the model called digital ears evolved an irreducibly complex system. As noted above, Thompson did not make any claims about irreducible complexity with regard to his experiment. Those claims come from a third-party website, which states that the removal of any part caused the evolved system to cease functioning. It cites a *New Scientist* article [22] that says:

> A further five cells appeared to serve no logical purpose at all—there was no route of connections by which they could influence the output. And yet if he disconnected them, the circuit stopped working.

This would seem to support the claim, but is actually a matter of imprecise wording. If we look at the original paper published by Thompson discussing those same cells, he wrote:

> Clamping some of the cells in the extreme top-left corner produced such a tiny decrement in fitness that the evaluations did not detect it [18].

Rather than the circuit effectively ceasing to function when the parts were removed, the circuit exhibited such a small degradation of performance that it was difficult to detect. This does not represent an example of irreducible complexity.

Trivial parts

According to Behe, irreducibly complex systems are by definition too improbable to be accounted for by *chance alone*, unaided by selection. In addition, Behe argues that irreducibly complex systems are unlikely to be produced by *selectable intermediate steps*. If the system is simple enough to explain without intermediates, Behe's argument against intermediates is irrelevant. It follows that any system that fails to meet Behe's complexity requirement is *not* an example of irreducible complexity.

The requirement for this complexity is captured in the definition that specifies "several" components that are "well-matched." "Several" components indicate that at least three components are required. For a component to be "well-matched" implies that there are other components that would be ill-matched, and that the well-matched component is one of the few possible parts that could fulfill its role. Behe's definition implies both a minimum of three parts *and* individual complexity of the parts.

Although Behe did not explicitly state a requirement for parts to be individually complex in his book, his examples of irreducible complexity make this an implicit requirement. In his examples each individual part is at least a single protein, whose sequence specificity (complexity) is very high. The complexity of a part can be measured by the probability of obtaining that part. Absent selection, how probable would obtaining the part be? For example, proteins are made from 20 standard amino acids. If 3 of those amino acids would function in a given position along a protein chain, the probability of obtaining a working amino acid at that position is $^3/_{20}$. We can estimate the complexity of any part by estimating the probability that random processes could produce a workable part. For proteins the probability of finding that sequence by chance decreases exponentially with each additional amino acid in length. This is one reason why the origin of proteins has proven extremely difficult for Darwinian evolution to explain [23,24].

Although Behe does not argue for the *irreducible* complexity of individual proteins, their complexity is clear. Further, the adaptation of one protein to another is itself complex. If one were to argue that we need not explain the origin of proteins, just the adaptation of existing proteins to work together, that adaptation itself would also require multiple amino acid substitutions, and is itself highly unlikely.

Behe hints at the problem of adapting parts to one another in discussing a hypothetical evolutionary pathway for a mousetrap. He points out the complex aspects of the mousetrap that the pathway skips over:

> The hammer is not a simple object. Rather it contains several bends. The angles of the bends have to be within relatively narrow tolerances for the end of the hammer to be positioned precisely at the edge

> of the platform, otherwise the system doesn't work.[7]

Behe's concept of irreducible complexity thus assumes that the component parts are themselves complex.

From what is said above, it is clear that parts themselves may be constructed of smaller parts. For example a molecular machine is made of proteins, which are made of amino acids. When we consider the complexity of a part, then, we are considering the complexity of the parts that make up the irreducibly complex system, not just the constituent subcomponents of the parts. While an amino acid by itself is too simple to be a component in an irreducibly complex system, a protein made up of many amino acids is sufficiently complex.

How rare or improbable does a component have to be? For computer simulations, this depends on the size of the experiment. The more digital organisms that live in a model, the more complexity can be accounted for by chance alone. For example, suppose that the individual parts in a system each have a probability of one in a hundred. Given a system of three components, the minimum necessary for a system of several components, the probability of obtaining all three components by chance would be one in a million, derived by multiplying the probabilities of the three individual components. Given a million attempts, we would expect to find a system with a probability of one in million once on average. To demonstrate that the irreducibly complex system could not have arisen by chance, the level of complexity must be such that average number of guesses required to find the element is greater than the number of guesses available to the model.

The largest model considered here, Avida, uses approximately fifty million digital organisms [14]. The smallest model considered, Sadedin's geometric model, uses fifty thousand digital organisms [17]. The individual components should be improbable enough that the average guessing time exceeds these numbers. We can determine this probability by taking one over the cube root of the number of digital organisms in the model. We are taking the cube root because we are assuming the minimal number of parts to be three. The actual system may have more parts, but we are interested in the level of complexity that would make it impossible to produce any system of several parts. Making this calculation gives us minimal required levels for complexity of approximately $^1/_{368}$ for Avida and $^1/_{37}$ for Sadedin's model.

Inspection of the models reveals that almost all of them have parts with a complexity less than even the lower limit derived above. Avida has twenty-six possible instructions. That gives a probability of at least $^1/_{26}$: insufficiently complex.

In Ev, there are $4^6 = 4096$ possible binding sites. However, many different binding sites will fit a given rule and be recognized. In order to measure how many patterns would fit, I used the Ev Ware Simulation.[8] I ran the Ev search one thousand times, and each time measured how many possible binding sites fit. On average approximately 235 of the 4096 binding sites fit the pattern. Thus, approximately one in eighteen patterns would be a valid binding site. Ev's parts are also too simple; finding a binding site is too probable by chance alone.

The Steiner and Geometric models are similar in that each part is either on or off, controlled by a single bit. This means that parts are of the smallest possible level of complexity. It is not a matter of being able to build or find the part, but only of being able to turn it on. Neither model can claim to be demonstrating the evolution of irreducible complexity because finding a solution is well within the reach of each model by chance alone.

Almost all of the cases of proposed irreducible complexity consist of parts simple enough that a system of several components could be produced by chance, acting without selection. As such, they fail to demonstrate that their models can evolve irreducibly complex systems, especially on the scale of biological complexity.

Roles of parts

Discussion of irreducible complexity in the literature has typically focused on the second part of the definition, the knockout test. For a system to be irreducibly complex, it must effectively cease to function when essential parts are removed. However, this is only part of the requirements for a system to be irreducibly complex [2]. The first part of the definition is also important. It requires that the system necessarily be "composed of several well-matched, interacting parts that contribute to the basic function"[1,8]. None of the presented models attempt to show that they fit this criterion.

Behe does not argue that systems are irreducibly complex based simply on the fact that the systems fail when a part is removed. Rather, he identifies the *roles* of the parts of the system and argues that all the *roles* are necessary for the system to operate. He states:

> The first step in determining irreducible complexity is to specify both the function of the system and all system components … The second step in determining if a system is irreducibly complex is to ask if all the components are required for the function. [1]

For example, a mousetrap has a hammer. The hammer hits the mouse and kills it. Any trap that kills a mouse by blunt force trauma will require a part that fulfills the role of the hammer. It is not simply that the removal of the hammer causes the trap to fail. Rather, the hammer or something fulfilling the same role must be in place by virtue of the mechanism used to kill the mouse. The other parts of the mousetrap can be argued to be necessary in a similar way.

In most of the models considered, defining the roles of the parts is not possible. In the Steiner trees or geometric model it would be difficult to argue that any sort of mechanism is actually involved. The workings of the digital ear circuit are not understood [18,22,25] and Behe has noted that understanding the mechanism is a prerequisite to identifying irreducible complexity [8]. Ev has a recognizer gene and binding sites; however, that only identifies two roles in the system, not several.

In order to claim that a system is irreducibly complex, it is

[7] http://www.arn.org/docs/behe/mb_mousetrapdefended.htm

[8] http://evoinfo.org/ev/

necessary to establish the roles of the parts and show that all the roles are necessary to the system. However, none of the models attempt to do this. There is no attempt to show that there is a functional system composed of well-matched interacting parts. Without that, the definition of irreducible complexity has not been fulfilled.

Designed evolution

Darwinian evolution is an ateleological process. It does not receive assistance from any sort of teleological process or intelligence. If a model is designed to assist the evolution of an irreducibly complex system, it is not a model of Darwinian evolution. Since it is not a model of Darwinian evolution, it cannot tell us whether or not Darwinian evolution can account for irreducibly complex systems. Any decision in the construction of a model made with an eye towards enabling the evolution of irreducible complexity invalidates the model. In order to demonstrate that a computer simulation can evolve irreducibly complex systems, the simulation must not be intelligently designed to evolve the irreducibly complex system.

Avida deliberately studied a function that could be gradually constructed by first constructing simpler functions. The authors discuss this:

> Some readers might suggest that we 'stacked the deck' by studying the evolution of a complex feature that could be built on simpler functions that were also useful. However, that is precisely what evolutionary theory requires, and indeed, our experiments showed that the complex feature never evolved when simpler functions were not rewarded [14].

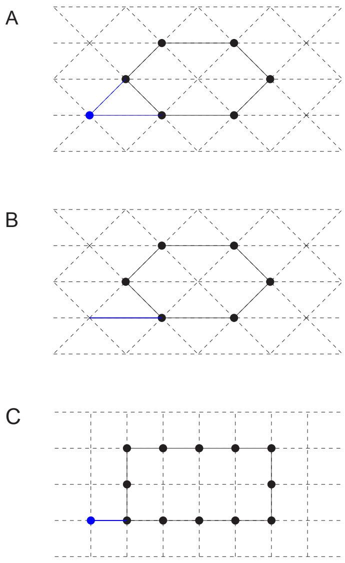


Figure 4: The success of the geometric model depends on several features chosen by the designer. A) As the model is intended to work, when an adjacent blue point is turned on, new area is added to the closed shape. B) When mutations cause a line to be added (not part of the model as designed), no additional area is added to the shape. C) Changing the shape of the grid to rectangular prevents single mutations from adding area to the shape. **doi:**10.5048/BIO-C.2014.1.f4

Out of all the possible features that could be studied, the developers of Avida chose features that would be evolvable. They have deliberately constructed a system where evolution proceeds easily. They justify this by stating that it is required by evolutionary theory. However, the question is whether this requirement will be met in realistic cases, and Avida has simply assumed an answer to that question.

The geometric model was designed to allow irreducible complexity to evolve. The sole intent of the model was to demonstrate the evolution of irreducible complexity. There is no attempt to justify the model in terms of realistic problems.

To see how Sadedin's model is designed to evolve, consider the growth of shapes in her model as compared to two similar models. In Sadedin's model, turning on one point near the existing shape is sufficient to add area to the shape (see Figure 4A). However, if mutations were to add lines rather than points we'd have the situation depicted in Figure 4B. There is no line that can be added to this shape that would expand its area. Figure 4C depicts a similar scenario, except that the shape is on a rectangular grid, rather than a triangular lattice. Here there is no line or point that can be added to the shape to increase its size. The ability of Sadedin's model to easily grow large shapes depends on the specific design of the model. The model was designed to evolve irreducible complexity.

It is possible to evolve irreducibly complex systems using a model carefully constructed toward that end. The irreducible complexity indicates the design of the model itself. It does not show that Darwinian processes can account for irreducible complexity.

MODEL SUMMARY

The models we have described have been presented by others as demonstrating that irreducibly complex systems can evolve. However, we have shown that these models have a number of common flaws. Table 1 shows a summary of these results.

Avida fails by three criteria. The parts are of trivial complexity. There is no attempt to show that the parts are necessary for the working of the system. Furthermore, the system was deliberately chosen as a subject of study because it would be evolvable.

Ev's genes continues to function at a lower capacity if binding sites are removed. The ease of obtaining a working part makes them trivially complex. There is no attempt to assign roles to the parts in the system. However, Ev was not deliberately designed to evolve irreducible complexity.

The parts in Dave Thomas's Steiner tree algorithm consist of only a single on/off switch and thus are as trivial as possible. There is nothing that can be considered a mechanism. There is no attempt to show that the parts are well-matched.

The geometric model has on/off switches for parts. There is no mechanism or an attempt to show how the parts are necessary for the mechanism. Furthermore, the entire system is designed to allow evolution in a particular way.

Thompson's electronic ear circuit fails the knockout test. The mechanism by which the circuit works is unknown, and thus

Table 1: Summary of flaws in models

	Avida	Ev	Steiner	Geometric	Ears
Failed Knockout		X			X
Trivial Parts	X	X	X	X	
Roles of Parts	X	X	X	X	X
Designed to Evolve	X			X	

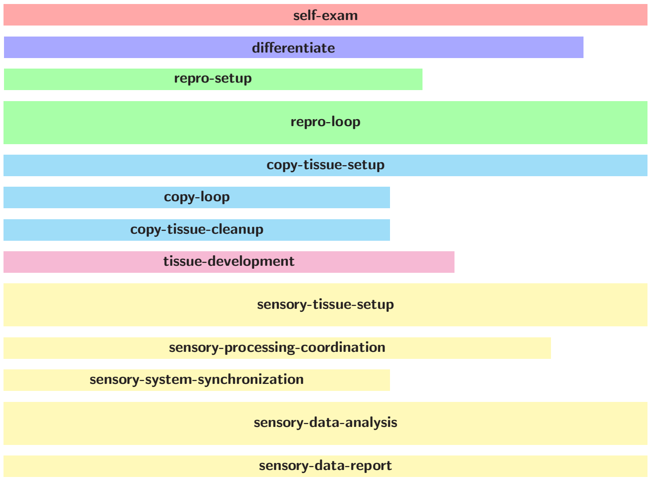


Figure 5: A depiction of the Tierra Network Ancestor. The ancestor cell is composed of genes that carry out particular functions and are shown in a particular color. The subgenes that make up each gene are represented as rectangles, with the size of the rectangle relative to the size of the subgene. Each rectangle is labeled with its sub-function. **doi:**10.5048/BIO-C.2014.1.f5

it would be impossible to attempt to assign roles to the various parts.

Thus, none of the published models has demonstrated the evolution of irreducible complexity because they fail to meet the required definition of irreducible complexity.

AN EXAMPLE OF IRREDUCIBLE COMPLEXITY

Tierra

If none of the presented models demonstrate the evolution of irreducible complexity, what kind of model would demonstrate it? To help answer that question, we present an example taken from Tierra. Thomas Ray developed Tierra in an attempt to produce a digital Cambrian explosion based on self-replicating organisms [26]. However, Tierra has been shown to adapt mostly by loss and rearrangement rather than by acquiring new functionality [27].

In the original version of Tierra, a number of programs, or "cells," were simulated on a single computer and allowed to evolve by competing to become the most efficient self-replicator. Later, Ray developed a network version of Tierra that added the ability for programs to jump between multiple computers running the Tierra simulation [28]. Not all computers were equally desirable from the perspective of the self-replicating programs, however. Some computers ran faster due to faster hardware or lack of other programs running. The Tierra programs were given the ability to read information about other computers before jumping, in order to choose the best one.

Tierra does not attempt to model the origin of life. Rather, it begins with the equivalent of a first self-replicating cell seeded into the simulation. This is called the ancestor program. Figure 5 depicts the ancestor program used in Network Tierra. The distinct colors represent the genes (or subsystems) that carry out a particular function within Tierra, and the individual boxes represent subgenes (sub-subsystems necessary for the overall function). The depiction shown is derived from the network ancestor program available for download from the Tierra website.[9] The yellow gene is the sensory system, which is responsible for collecting the information about other computers on the Tierra network and making the decision about which computer to jump to.

The sensory system present in the network Tierra ancestor is irreducibly complex. To see this, consider the functionality of the three largest subgenes in the sensory system gene. The sensory-tissue-setup subgene collects information about fifteen different computers on the Tierra network. The sensory-data-analysis subgene determines which computers are better. The sensory-data-report subgene acts on that decision, causing the best computer to be selected as the target.

If the sensory-tissue-setup subgene were missing, the information about the other computers would not be collected and the other genes would operate on and produce garbage information. If the sensory-data-analysis subgene were missing, the sensory-data-report gene would act upon decisions that had nothing to do with the data. Effectively, a random computer would end up being chosen. If the sensory-data-report subgene were missing, the decision reached by the analysis gene would be ignored, and again a random computer would be selected. If any of the parts are missing, the system fails to use the available information to select an appropriate target computer; it effectively ceases to function.

Are the parts sufficiently complex? In order to derive the limit, we need to consider the number of guesses or simulated organisms in Tierra. Unfortunately, Ray's paper does not provide an estimate of the number of organisms. The experiment was run for fourteen days on sixty computers. This comes out to approximately three million seconds of computation time. Making the generous assumption of one nanosecond per digital organism, this gives us about three quadrillion digital organisms. Taking the cube root of this number gives us a complexity requirement of approximately one in 144,609.

There are 50 possible instructions in a network Tierra

[9] http://life.ou.edu/tierra/

program. The smallest subgene is twelve instructions long. For the three subgenes considered the core of the irreducibly complex system, the shortest is twenty-two instructions long. It is possible that other, possibly shorter, sequences could fulfill the same role. However, if we assume that at least four of the twenty-two instructions are necessary for the role there are $50^4 = 6{,}250{,}000$ combinations, giving approximately a $^1/_{6250000}$ probability of picking the correct four instructions. This exceeds the complexity limit, and thus we conclude that the parts are sufficiently complex.

Thus the Tierran sensory system is an example of irreducible complexity. It consists of several parts with roles indispensible to the mechanism of the system. It passes the knockout test and the parts and system are not trivial in their complexity.

Could it evolve?

The sensory system did not evolve; it was designed as part of the ancestor used to seed the Tierran simulation. But the real question is whether or not it could have evolved. The observed evolution of the sensory system has been to either lose or simplify that system [28]. There was an opportunity to re-evolve the sensory system after it had been lost, but such an event was not reported and presumably did not happen.

But could it have evolved? Is there a Darwinian way to evolve the complex sensory system that we find in the Tierran ancestor? One might attempt to construct it beginning with a simple sensory system that picks a computer at random. However, even that first step requires the evolution of code of non-trivial complexity. Tierra has shown very limited abilities to evolve new sections of code [27].

But ultimately, the question is not whether humans can find a way to make it happen. The question is whether Darwinian evolution can evolve the system. If Darwinian evolution is capable of developing systems like the Tierran sensory system, then we should see models that demonstrate this evolution.

CONCLUSION

This paper has investigated a number of published models that claim to demonstrate the evolution of irreducibly complex systems, and found that these models have failed on a number of fronts. Two of the models fail to satisfy the knockout test, in that they maintain functionality after parts have been removed. Almost all of the models use parts that are trivially complex, on the order of an amino acid rather than a protein in complexity. None of the models attempt to show why the mechanism used necessarily requires its parts. Finally, some of the models have been carefully designed to evolve. Thus, none of the models presented have demonstrated the ability to evolve an irreducibly complex system.

In contrast, we do find irreducible complexity in the designed sensory system of the Tierran ancestor. This system is an example of what kind of system it would be necessary to evolve in order to *falsify* the claim that irreducible complexity is difficult to evolve. It has not been proven that the sensory system cannot evolve, but neither has it been shown that the sensory system can evolve. The prediction of irreducible complexity in computer simulations is that such systems will not generally evolve apart from intelligent aid.

The prediction that irreducibly complex systems cannot evolve by a Darwinian process has thus far stood the test in computer models. Some have claimed to falsify the prediction, but have failed to follow the definition of irreducible complexity. However, it is always possible that a model will arrive that will falsify the claim. Until then, as a falsifiable prediction the evidence for irreducible complexity grows stronger with each failed attempt.

Acknowledgements

The author thanks George Montañez for insightful comments on previous drafts of this paper.

1. Behe MJ (1996) Darwin's Black Box: The Biochemical Challenge to Evolution. Free Press (New York).
2. Dembski WA (2004) Irreducible Complexity Revisited. Available: http://www.designinference.com/documents/2004.01.Irred_Compl_Revisited.pdf
3. Dembski WA (2004) Still Spinning Just Fine: A Response To Ken Miller: 1–12. Available: http://www.designinference.com/documents/2003.02.Miller_Response.htm. Accessed 6 January 2014.
4. Miller KR (2004) The flagellum unspun: the collapse of "irreducible complexity." In: Dembski WA, Ruse M, eds. Debating Design: From Darwin to DNA. Cambridge University Press, pp 81–97.
5. Miller KR (1999) Finding Darwin's God: A Scientist's Search for Common Ground Between God and Evolution. 1st ed. HarperCollins (New York).
6. Behe MJ (2006) Irreducible Complexity: Obstacle to Darwinian Evolution. In: Dembski WA, Ruse M, eds. Debating Design: From Darwin to DNA. Cambridge University Press, pp 352–370.
7. Aird WC (2003) Hemostasis and irreducible complexity. J Thromb Haemost 1:227–230. doi:10.1046/j.1538-7836.2003.00062.x
8. Behe MJ (2001) Reply to my critics: A response to reviews of Darwin's Black Box: The Biochemical Challenge to Evolution. Biol Philos 16:685–709.
9. Pennisi E (2013) The man who bottled evolution. Science 342:790–793. doi:10.1126/science.342.6160.790.
10. 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.
11. Behe MJ (2007) The Edge of Evolution. Free Press (New York).
12. Behe MJ (2010) Experimental evolution, loss-of-function mutations, and "the first rule of adaptive evolution." Q Rev Biol 85:419–445.
13. Blount ZD, Barrick JE, Davidson CJ, Lenski RE (2012) Genomic analysis of a key innovation in an experimental *Escherichia coli* population. Nature 489: 513–518. doi:10.1038/nature11514.
14. Lenski RE, Ofria C, Pennock RT, Adami C (2003) The evolutionary origin of complex features. Nature 423:139–144. doi:10.1038/nature01568.
15. Schneider TD (2000) Evolution of biological information. Nucleic Acids Res 28:2794–2799. doi:10.1093/nar/28.14.2794.

16. Thomas D (2010) War of the weasels: An evolutionary algorithm beats intelligent design. Skept Inq 43:42–46.
17. Sadedin S (2005) A simple model for the evolution of irreducible complexity. Available: http://www.csse.monash.edu.au/~suzannes/files/Sadedin2006TR.pdf. Accessed 25 July 2013.
18. Thompson A (1997) An evolved circuit, intrinsic in silicon, entwined with physics. In: Evolvable Systems From Biology to Hardware, vol 1259:390–405. **doi:**10.1007/3-540-63173-9_61.
19. Fogel DB (1998) Evolutionary Computation: The Fossil Record. Wiley-IEEE Press.
20. Ewert W, Dembski WA, Marks II RJ (2009) Evolutionary Synthesis of Nand Logic: Dissecting a Digital Organism. 2009 IEEE International Conference on Systems, Man and Cybernetics. IEEE. pp 3047–3053. **doi:**10.1109/ICSMC.2009.5345941.
21. Ewert W, Dembski WA, Marks II RJ (2012) Climbing the Steiner tree—Sources of active information in a genetic algorithm for solving the Euclidean Steiner tree problem. BIO-Complexity 2012(1):1–14. **doi:**10.5048/BIO-C.2012.1.
22. Davidson C (1997) Creatures from primordial silicon. New Sci 2108:30–34.
23. Axe DD (2004) Estimating the prevalence of protein sequences adopting functional enzyme folds. J Mol Biol 341:1295–1315. **doi:**10.1016/j.jmb.2004.06.058.
24. Axe DD (2010) The case against a Darwinian origin of protein folds. BIO-Complexity 2010(1):1–12. **doi:**10.5048/BIO-C.2010.1.
25. Davidson C (1998) The chip that designs itself. Comput Bull 40:18.
26. Ray TS (1991) An approach to the synthesis of life. In: Langton C, Taylor C, Farmer JD, Rasmussen S, eds. Artificial Life II: Proceedings of the Workshop on Artificial Life held February, 1990 in Santa Fe, New Mexico. Addison-Wesley (Redwood City, CA), pp 371–408.
27. Ewert W, Dembski WA, Marks II RJ (2013) Tierra: The character of adaptation. In: Marks RJ II et 'al, eds. Biological Information—A New Perspective. WORLD SCIENTIFIC, pp 105–138. **doi:**10.1142/9789814508728_0005.
28. Ray TS, Hart J (1999) Evolution of differentiated multi-threaded digital organisms. In: Intelligent Robots and Systems, IROS'99. Proceedings. 1999 IEEE/RJS Int Conf, vol 1, pp 1–10.