January 6, 2010 Number Fifteen

Occasional Papers of the BSG

IRREDUCIBLE COMPLEXITY AND RELATIVE IRREDUCIBLE COMPLEXITY: FOUNDATIONS AND APPLICATIONS

Jonathan Bartlett

Occasional Papers of the BSG

Executive Editor: Roger Sanders, Center for Origins Research, Bryan College, 721 Bryan Drive, #7802, Dayton, TN 37321, USA
Email: editor@creationbiology.org

About the BSG: Founded in 1996, BSG: A Creation Biology Study Group is a society of researchers and scholars. The mission of the BSG is to develop and nurture a community of dedicated researchers committed to understanding the life sciences from a young-age creationist perspective through meetings and publications. Membership is open to any who share these goals. For more information, visit the official BSG website, http://www.creationbiology.org, or email secretary@creationbiology.org.

Journal Policies

OPBSG publishes peer-reviewed, original research or review papers of relevance to creationist biology. Acceptable topics include anything related to creationist biology, including philosophy, theology, and biblical studies. *OPBSG* is committed to constructive scientific research in creation biology; manuscripts that are primarily critiques of evolution will be referred to a more appropriate publication. All *OPBSG* issues are published and distributed electronically at http://www.creationbiology.org/opbsg/.

OPBSG seeks original scholarly manuscripts in any field related to creationist biology. Guidelines for manuscript preparation are available at the OPBSG website http://www.creationbiology.org/opbsg. Papers may be submitted electronically to the Executive Editor at editor@creationbiology.org.

All papers and abstracts submitted to *OPBSG* are subject to peer review by an editor assigned by the Executive Editor. Editors are forbidden from participating in the editing of their own papers or papers of their coworkers. Papers and abstracts are evaluated by at least one external peer reviewer. Editors then evaluate the peer review(s) and determine the suitability of the paper for publication in *OPBSG*. Editorial decisions may be appealed to the editorial board after reasonable efforts to resolve differences. Decisions of the editorial board are final.

A detailed editorial manual is available at the *OPBSG* website http://www.creationbiology.org/opbsg.

Occas. Papers of the BSG No. 15, pp. 1-10
www.creationbiology.org

Irreducible Complexity and Relative Irreducible Complexity: Foundations and Applications

JONATHAN BARTLETT[1]

[1] *Independent Scholar, Broken Arrow, OK*

Abstract.
The concept of irreducible complexity is a great intuitive concept which has historically lacked a well-defined theoretical basis. This paper will show how computational theory gives an adequate account for what irreducible complexity is, what irreducible complexity's possible biological implications are, and how irreducible complexity can be useful to molecular biologists and creation biologists.

1. Irreducible Complexity Basics and History

1.1 What is Irreducible Complexity? Many teleologically-oriented biologists have long viewed biology in a holistic manner. That is, they view the whole of a biological system as having priority over any of its parts. That is, the parts are important, but only as contributors to the whole. The concept of irreducible complexity (IC) is an attempt to capture holistic thinking into a framework which is amenable for scientific understanding.

The term "irreducible complexity" was introduced in 1996 by Michael Behe in the book *Darwin's Black Box*. In this book Behe argued that some biological systems are Irreducibly Complex. Behe (1996, p. 39) defined irreducible complexity as follows:

> By irreducibly complex I mean 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. An irreducibly complex system cannot be produced directly (that is, by continuously improving the initial function, which continues to work by the same mechanism) by slight, successive modifications of a precursor system, because any precursor to an irreducibly complex system that is missing a part is by definition nonfunctional.

In Behe's definition of IC, Behe defines a holistic entity as a group of parts which are nonfunctional except in the context of other parts, and only by working together do these parts achieve function. Thus, the empirical test for this type of holism is whether or not removing a part causes the function to cease. This certainly doesn't capture all types of holistic systems, but it is clear to see how IC relates to a holistic view of biology.

1.2 Irreducible Complexity Before Behe. Long before Behe, creationists and other teleologically-oriented biologists made similar claims about biological holism, though in much more general and intuitive forms. The essence of the argument is that many biological systems act holistically, and therefore their development must have taken place in a holistic manner, whether through direct design or some other mechanism that operates holistically. We will refer to all of the arguments which take this form as being instances of irreducible complexity, even though such terminology is anachronistic for the examples discussed.

The claims of irreducible complexity were being made even before Darwin. In *Animal Kingdom*, Cuvier (1834, p. 7) wrote:

> Organisation, then, results from a great variety of arrangements, which are all conditions of life; and it is easy to conceive, that if its effect be to alter either of these conditions, so as to arrest even one of the partial motions of which it is composed, the general movement of life must cease. .... Life, then, in general, presupposes organisation in general; and the life proper to each individual being, presupposes an organisation peculiar to that being, just as the movement of a clock presupposes the clock itself; accordingly we behold life only in beings that are organised and formed to enjoy it, and all the efforts of philosophy have never been able to discover matter in the act of organisation, neither of itself, nor by (pg 8) any external cause.

What Cuvier was saying was that the organization of parts is greater than the motion of any individual piece, and that, without the unit of organization (which, for Cuvier, was the organism), the pieces tend to run counter to the mode needed for life. Cuvier viewed the organization of organisms holistically, and, for him, the organization itself was a species, with the possible variations on that organization being varieties.

Behe's concept of irreducible complexity focuses on much smaller subsystems of organisms, and treats them independently. However, the basic form of the idea remains the same. The parts require the organization of the parts to perform a task. Without the organization (which, for Behe, is a system which performs the basic function), the parts don't help. Likewise, both Behe and Cuvier argue that the organization itself cannot be produced from purely material forces in nature.

After Darwin (1859) published the *On the Origin of Species*, irreducible complexity was used to argue the deficiency of natural selection to explain new structures, or the coordination of changes to structures. Darwin's greatest contemporary critic was George Mivart, whose version of irreducible complexity was set forth in a chapter of his *On the Genesis of Species* titled "The Incompetency of 'Natural Selection' to Account for the Incipient Stages of Useful Structures." Quoting Joseph John Murphy, Mivart (1871, pp. 64-65) said,

> The higher the organization, whether of an entire organism or of a single organ, the greater is the number of the parts that cooperate, and the more perfect is their cooperation; and, consequently, the more necessity there is for corresponding variations to take place in all the cooperating parts at once, and the more useless will be any variation whatever unless it is accompanied by corresponding variations in the cooperating parts; while it is obvious that the greater the number of variations which are needed in order to effect an improvement, the less will be the probability of their occurring at once... [Discussing whether time would allow these improbable events to become probable] the improbability of obtaining an improvement in an organ by means of several spontaneous variations, all occurring together, is an improbability of the same kind [i.e., impossible].

It should be noted that, unlike Cuvier, Mivart and Murphy were not criticizing the ability for new structures or coordinated adaptations to occur. Rather, they were criticizing the mechanism of natural selection to accomplish it. Their point was that new structures, and even many adaptations to existing structures, require coordinated change for their variations to be useful. Any theory of their development would need to include a mechanism which could provide the necessary coordination to produce the changes which were supposed. Thus, natural selection was not a viable mechanism for them because it was attempting to explain development of organized structures without a sufficient coordinating mechanism.

More recently, with the discovery of genetics and the genetic code, most biologists look to the genome as the organizing principle of an organism. Thus, teleological biologists tend to focus their descriptions of irreducible complexity on the development of the genetic code for cellular processes under consideration. Likewise, the modern evolutionary synthesis is also looking to the genetic code, since that is viewed as the locus of organizational development. In the modern synthesis, the development of the code proceeds by copying errors in an original code, where the beneficial copying errors were perpetuated through reproduction and the problematic ones were eliminated through death or sexual selection.

The contention of the teleologists is expressed in this way by Lester and Bohlin (1989, p.86):

> One does not add constructive sentences, paragraphs, or chapters to a complete book by the selective addition of random copying errors. Is it therefore reasonable to expect evolutionary novelty to arise in living creatures by slow accumulation of point mutations?

Thus, as a functional code, the genome is a holistic entity, and purely physical processes are incapable of generating the holism of a functional code.

1.3 The Darwinian Response to Irreducible Complexity. While there are several different terms for it, the basic Darwinian response to irreducible complexity has been to separate form and function. The idea is that while most would agree that the development of complex, coordinated systems by themselves for their own purposes is unlikely or impossible for step-at-a-time evolution, the components can be developed a step at a time for another purpose, and then the components can be co-opted for another use. Thus, the organism maintains *structural continuity* but with a radical functional departure (Gould 1991, p. 143).

This hypothesis, known variously as pre-adaptation, co-option, or exaptation, is intended to limit the number of coordinated steps required for evolution to create complex, multipart mechanisms. The improvements can proceed a step at a time, precisely because it is not currently fulfilling its future role, but rather another one. Wings, for instance, have been thought to be originally for thermoregulation, and only later used for flight. Since a thermoregulation function for wings favors their gradual growth (whereas an aerodynamic function would not favor a gradual growth), it is supposed that wings grew gradually as a thermoregulation mechanism until they were long enough for flight. Once they were long enough to be used for flight they underwent a functional shift, and were able to be used for wings (Gould 1991, p. 150). By postulating alternating functions for the same structure, it is thought that evolution can still operate a step at a time structurally.

The other possible evolutionary scenario for the development of complex, coordinated structures is *coordinated evolution*, where the mechanism for evolution itself manages the coordination of evolution of different parts of the organism. This sort of evolution, however, has been ruled out by most evolutionists as unscientific. The idea of a coordinated evolutionary scenario is, according to Darwin, "to enter into the realms of miracle, and to leave those of Science" (Darwin, quoted in Gould 1991, p. 142). Thus, having ruled out the possibility of an internally-driven, coordinated view of evolutionary processes, the idea of functional shifts tends to be the reigning idea for the origin of complex mechanisms.

The problem with this perspective is that functional shifts often do require many coordinated steps to occur even if the organism maintains gross morphological continuity. Flight, for instance, requires more than just wings. It also requires muscular, motor, balance, visual, and other nervous adaptations to work (Alonso et al. 2004). It requires a certain range of motion, and a certain amount of coordination with the rest of the biology of a bird. Thus, the argument of exaptation tends to only work if the structure under consideration is decontextualized from the rest of the organism.

A similar decontextualization can also occur when examining evolution from a molecular standpoint. When examining the evolution of hormone receptors, Bridgham et al. (2006) used the idea of a functional shift with structural continuity to explain the evolution of hormone receptors. They hypothesized that the AncCR receptor was originally bound to one hormone which had a structural similarity to the modern binding of the descendent receptors. Thus, they thought that the ancestral receptor was able to be refined by natural selection to a point where it underwent two functional shifts, each producing the divergent receptors that exist today (Bridgham et al. 2006). Among other problems with this criticism of irreducible complexity, it presents the same fundamental decontextualization that other proposed functional shifts tend to have. Hormone receptors can only be selected for or against in the context of the wider organism, and no explanation of why such functions might be selected for were presented, nor how the hormones which bind to the receptors came about, much less their own regulation and connection with the larger biochemistry framework.

All of this isn't to say that functional shifts are impossible, only that if they do occur, it does not seem likely that they occur a step at a time, but rather they would need to take place in a coordinated fashion.

Another response to irreducible complexity was the development of the Avida software program to demonstrate empirically that complexity can arise via Darwinian processes (Lenski et al. 2003). Avida has been a serious challenge to IC, though there are many aspects of the Avida program which have been disputed regarding their relevance to IC (Dembski 2004b). The conception of IC which will be developed in this paper, however, will be directly applied to Avida in section 3.4.

1.4 Behe's Irreducibility Concept—Benefits and Limitations. Irreducible complexity prior to Behe was a very vague notion, but had a strong intuitive appeal. The problem with intuitive appeals, however, is that they are not convincing or helpful to people who have different intuitions. Behe's definition of irreducible complexity attempts to move the concept beyond the intuitive and make it more explicit and concrete. Relatively simple tests for irreducible complexity can designed using genetic knock-out experiments, where each gene is disrupted to see what effects that disruption has on the whole system. Whether or not one agrees with the evolutionary implications that Behe assigns to an irreducibly complex system, it is straightforward enough to test and see that some systems are in fact irreducibly complex. As an example, Scott Minnich performed knock-out experiments on the bacterial flagellum and proved that, whether or not the flagellum was evolvable, it was certainly Irreducibly Complex per Behe's definition (Minnich and Meyer, 2004).

However, there are a number of problems with irreducible complexity as it stands today. The first one is that there is little biological use for the concept. Even if everyone agreed with both the concept of IC and its application to evolution, it is unclear exactly how that might be helpful to biologists examining biological phenomena.

The second problem is that IC, as it currently stands, only applies to brittle systems. IC, though it is based on holistic thinking, is not, and was not meant to be, coextensive with the concept of holism. Instead, it operates on a subset of systems which are amenable to the methodology Behe lays out. The problem is that in order for a system to be amenable to Behe's test of IC, it must be brittle. Systems which have backup systems, alternate pathways, and other mechanisms to resist catastrophic failure will not generally be classified as Irreducibly Complex. The current definition of IC may be better at identifying Rube Goldberg devices (functional devices noted for their needless complexity) than it is finding instances of well-engineered systems.

The third problem is that the leap from irreducible complexity to limits on evolvability is still largely intuitive. Dembski (2004a) clarified the lines of argument from irreducible complexity by separating out the logical, empirical, and explanatory sides of the irreducible complexity argument. By Dembski's reasoning, IC rules out direct Darwinian pathways logically, by pointing out that all pieces are necessary for function; IC rules out indirect Darwinian pathways (i.e., functional shifts) empirically by pointing out that, for the systems Behe covers, no plausible indirect pathways have even been suggested; IC then evidences design because, unlike Darwinian evolution, intelligent agents are causally adequate for producing IC systems. The problem with Behe's line of reasoning, though, is that the empirical ruling out of indirect Darwinian pathways relies too much on intuitive measures of plausibility.

Additionally, because Behe's conception of IC is formulated for an Intelligent Design perspective, no real work has been done in relating IC to creation biology. IC has been used apologetically to defend positions as varied as theistic evolution and young-earth creationism. But because it is used as a general defense for a variety of theistic beliefs, very little effort has been put forth to understand it in a specifically creation-oriented context (Purdom, 2006).

A more beneficial definition of IC should accomplish several tasks. First, it should formulate the full argument logically, so that it does not rely on intuitive appeals. Second, it should expand its coverage of holistic systems beyond brittle systems. Third, it should outline a model which is useful to biologists beyond the question of whether a system has been designed. Finally, it should interact more directly with creation biology.

The best way to solve these issues is to strengthen the theoretical foundation of IC. Because biological systems use codes to solve problems, the best place to look for a theoretical understanding of irreducible complexity is computability theory, which explicitly studies the behaviors of systems which are specified by codes.

2. Irreducible Complexity in the Light of Computability Theory

2.1 Turing's Theory of Computation. Before discussing the application of computability theory to biology and IC, a short introduction to some basic, relevant concepts is in order.

Before real computers existed, Alan Turing developed a model for discussing algorithms (an algorithm is a step-by-step set of instructions for solving mathematical problems) (Turing 1936). Turing developed a hypothetical type of machine which can be used to describe the algorithm for performing any computable function. This machine (called a Turing machine) consists of 4 parts (Figure 1):

Tape: The machine has a tape which is infinitely long. The tape is divided into cells, each of which had an integer with a finite range encoded on it. Each cell can be read or modified during the algorithm. The tape holds both the "instructions" for the machine as well as the input values for the mathematical function at the beginning of the computation (these are usually separated logically, but the machine does not place

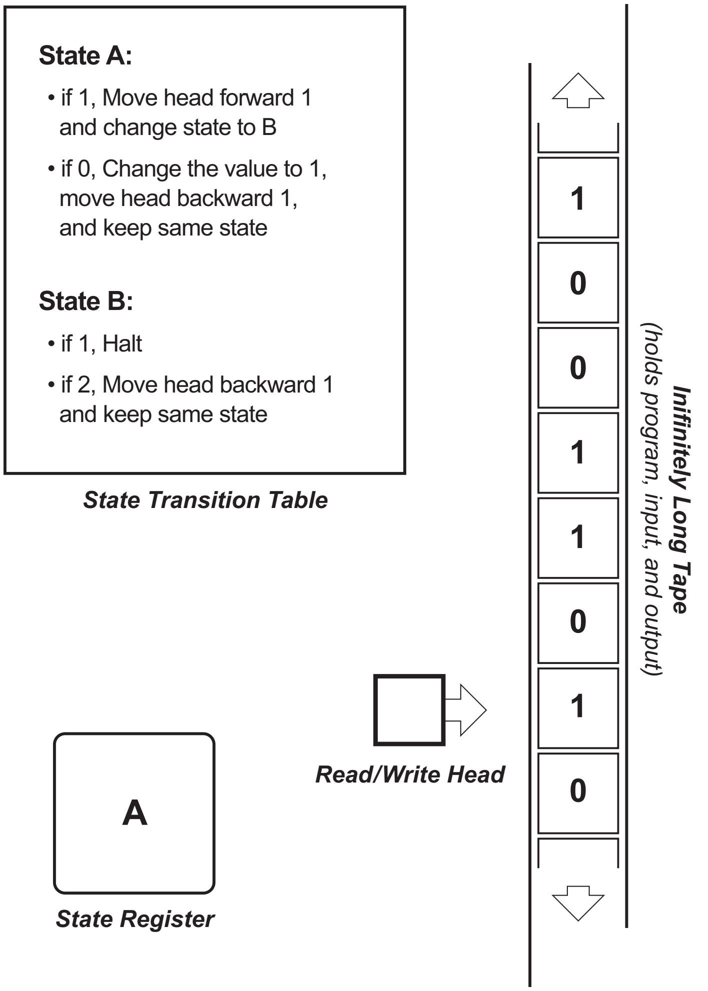

Figure 1. A simple Turing Machine.

any physical restrictions on where they are stored on the tape). At the end of the computation the tape holds the output in a predefined location.

Read/Write Head: The read/write head is an instrument that is located at a certain position along the tape. The read/write head can read the number at the current location on the tape, write a new number at the current location on the tape, and move back and forth along the tape.

State Register: The state register is used to record the current state for the state transition table. It can be exactly one of a finite number of predefined values. The value of the state register is known simply as the "state."

State Transition Table: The state transition table tells the machine how to behave given both the current value of the tape at the position of the read/write head and the value of the state register. For each combination of the state and the current value, the state transition table tells the machine what to do with the read/write head (move it and/or write to its current location), whether to change the state register (and what to change it to), and whether or not to halt the computation.

The algorithm to compute *any computable function* could be described by specifying:

- The contents of the state transition table
- The initial value of the state register
- The initial location of the read/write head
- The location on the tape where the result would be stored
- The initial contents of the tape (this is usually logically divided into the "program" and the "input" but the Turing machine does not physically distinguish these two)

Therefore, a Turing machine could be very simple or very complex. A simple Turing machine might list only one value for the state transition table—to halt no matter what the input. This sort of machine would be very predictable; it would be very easy to predict the final state on the basis of the input state since they would be the same. Alternatively, a Turing machine might have hundreds of state transition rules, and therefore be very flexible, and potentially have a very complex relationship between the input and the output, as we will examine in a later section. In any case, by specifying these components, the algorithm to *any computable function* can be described.

The terminology for the rest of the paper will use that of Turing machines. Because Turing machines can be used to implement any computable function, they can also be used to implement other types of machines (such as cellular automata) used in other research. Therefore, we will use Turing-machine terminology for consistency's sake even if the cited literature is dealing with another, equivalent form of a machine.

2.2 Complexity Classes for Turing Machines. Stephen Wolfram established a set of complexity classes which could be used for classifying the behavior of various machines which include Turing machines (Wolfram 1983; Wolfram 1984; Wolfram 2002). By "complexity," Wolfram is referring to how chaotically the machine behaves. Complexity is determined by:

- The predictability of the state of the machine at any particular iteration
- The number of cells on the tape which might be affected by a small change in the initial values on the tape

This takes into account both the ability of the change in the initial value to propagate as well as the predictability of the effects of that propagation.

Wolfram performed experiments on many types of machines and established a set of four complexity classes:

Class 1: These are machines which tended to converge onto a single result within the tape no matter what the tape's initial values are.

Class 2: These are machines which tend to give relatively simplistic results. Changes to the initial values on the tape tend to affect a limited set of cells in the result.

Class 3: These are machines which tend to have results which are individually unpredictable, but statistically predictable. Small changes to the initial values tend to affect large numbers of cells, but the effects are at least statistically predictable. This class of machines is relatively unstudied, so examples of what they look like biologically are not readily available.

Class 4: These are machines which tend to have results which are not predictable either individually or statistically. Changes to initial values on these machines can affect an arbitrary number of cells in the result, or they may have little effect.

As you can see, while there might be a limited usefulness for class 1 machines, ultimately they are not very expressive. There is very little algorithmic work that can be done on a machine which converges to one or a few values in the result.

Take bee honeycombs, for instance. Camazine et al. (2001, p. 330) have analyzed these structures using cellular automata (a Turing-like machine), and found that the rules of the system will canalize the nest structure into a stable macro-structure in nearly every biologically-realistic scenario. This structure will eventually re-emerge even when elements of it are manually modified by an experimenter. While this is not a biochemical pathway, it demonstrates the type of canalization that class 1 machines tend to produce. While it is obviously quite useful for beehives, no amount of change to the initial conditions of this "machine" will cause the macro-structure to change. The outcome of the process is determinable even if the starting conditions are changed or radical perturbations are introduced.

In order to implement algorithms which solve more arbitrary problems (as opposed to a special-purpose problem such as the bee honeycomb), the machine simply must have more degrees of freedom. As you move through the complexity classes, more degrees of freedom are obtained, but more unpredictability in output is also obtained.

2.3 Universality and Its Implications. A Turing machine itself is a computable function (at least if it terminates), and therefore can be described and implemented by other Turing machines. A Turing machine that can implement any other Turing machine by only changing the instructions on the tape is called a Universal Turing machine or a Universal Computer (Turing 1936). Because any computable function can be described by a Turing machine, this means that, on a Universal Turing machine, any computable function can be described by only modifying the tape.

Please note: One common misconception is that a Universal Turing machine is a machine which does implement every computable function. This is not the case. A Universal Turing machine is simply one which could implement any computable function if it is provided the right tape to do so. This differs from non-Universal machines in that non-Universal machines cannot support all computable functions no matter what the tape's initial values are.

As noted previously, class 1 machines don't have very many degrees of freedom—certainly not enough to be Universal. Therefore, the question becomes, which classes of machines have sufficient degrees of freedom in order to be Universal? It turns out that all known Universal Turing machines exhibit class 4 complexity (Wolfram 1984; Wolfram 2002).

This has several important implications. First of all, it means that in the general case, Universal machines are unpredictably sensitive to change. In other words, it is not possible to predict, in the general case, what the effect of making small changes to the initial values of the tape will be. In addition, small changes to the tape often have wide-ranging effects within the results with unpredictable boundaries, including the possibility of no boundaries whatsoever. Because the effects of changes are not predictable in the general case, they are also not smooth in the general case, or else they would be predictable. However, given the right initial values on the tape, *certain subsets of the tape can be made to behave more predictably*. This follows from the fact that a Universal Turing machine can be used to emulate any other Turing machine on a subset of the tape, which would include Turing machines of lower complexity classes.

2.4 Complexity Classes and Biological Systems. Analyzing biological systems in terms of Turing machines is not new. Kauffman (1993, p. 12) notes that the ability to generate an arbitrary diversity of organisms by means of reprogramming the DNA makes biology analyzable under the same kind of terms as universal computers (i.e., Universal Turing Machines), with the main difference being that biological systems are massively parallel, while most desktop computers are not (though this matters little from a computational perspective).

As we've mentioned earlier, Turing machines can simulate other Turing machines. So, while the overall character of the genetic code is that of a class 4 system, there may be several instances where it can act like other classes of systems by either canalizing the reading of a segment of DNA or canalizing the way in which it can be modified.

For instance, phase-variable genes operate *in vivo* as a class 2 system because (a) the changes to the genes are canalized to certain potentials, and (b) all of these potentials affect a limited scope in the operation of the organism. In some bacteria, for instance, phase-variable genes control attachment of the organism to a surface (Henderson et al. 2002).

2.5 Universality and Irreducible Complexity. From the above discussion we can derive the following principles:

1. In order to be flexible enough so that the solution to an arbitrarily hard problem could be written on its tape, a machine must be Universal.
2. Within a Universal machine, some parts of it may be more chaotic than others.
3. The chaotic portions of the Universal machine are required to implement arbitrarily hard problems, because otherwise they could be implemented on non-Universal machines, thus violating principle 1.
4. Therefore, for a solution to be implemented for which the parameters and constructs needed for the solution are not known ahead of time, the solution will require the chaotic portions of the Universal machine to be used for all but the most trivial problems (if the parameters and needed constructs were known ahead of time then perhaps a non-Universal machine could have been constructed which matched the solution space).
5. Because the chaotic portions of the Universal machine are being used, the solution cannot have been arrived at *incrementally* because it violates the definition of chaotic behavior, which does not display smooth and predictable outcome changes when the initial values of the tape are changed. Therefore, incremental searches will not make the searches find a solution any faster except perhaps on trivial problems.

There are three primary implications of these principles from the point of view of irreducible complexity:

1. Non-Universal machines cannot be programmed to solve arbitrarily hard problems.
2. The implementation of solutions to arbitrarily hard problems on Universal machines requires guidance for how the program should change which originates from outside the machine. This can either be in the form of new information (which would presumably come from an intelligent agent)

or in the form of an outside system which had sufficient heuristics about the problem and the target system in order to implement the solution.

3. The implementation of solutions to hard problems can be achieved on non-Universal systems provided that the solutions are not arbitrarily hard, but instead vary along lines which are taken into account within the system. In other words, the solution space must be parameterized instead of open-ended.

Therefore we will define an Irreducibly Complex program as one which holds the following characteristics:

- The program is non-trivial
- It is implemented on a Universal machine
- It is not implemented within a region of the tape which is emulating a non-Universal machine by another part of the program
- It makes use of the chaotic (Class 4) constructs within the Universal machine
- The problem for which the program is trying to solve *requires the use* of chaotic (Class 4) constructs in order to solve it on the machine (i.e., the problem couldn't be solved using non-chaotic constructs of the same machine).

When those conditions are upheld, because the solution space is chaotic, there is no gradual way to arrive at the solution, and therefore the probability of landing on such a solution by chance decreases exponentially with the minimum number of components required for the solution. Therefore, we will consider any such non-trivial program to be Irreducibly Complex, and therefore require the input of information from outside (whether in the form of an intelligent agent or an outside system containing relevant information) in order to come into existence.

2.6 Relative Irreducible Complexity. While the above definition of irreducible complexity may be useful when we have all of the information about a machine, work within biology (and possibly other fields) will be done in absence of full knowledge about the workings of the machine and the tape. Therefore, it may be, given current knowledge, that a program *appears to be irreducibly complex* when in fact it is not. Therefore, we will use the term *relative irreducible complexity* (RIC) to refer to a system which gives the appearance of irreducible complexity given current knowledge.

Therefore, if a system is properly given the title of relatively irreducibly complex and it is shown to have evolved from an earlier system, then that indicates one of two possibilities:

- The information to evolve was applied to the program from the outside
- One of the IC criteria given above does not hold.

Therefore, the designation of relative irreducible complexity gives biologists specific things to look for if the program is shown to be evolvable. An evolvable RIC system indicates one or more of the following things about the evolved program:

- The program was input into the system from an external source
- The program was regularized by other unknown control factors which made the evolution of its behavior non-chaotic (the differences between this item and the previous item in some cases may simply result from what scale you are viewing the system). In biological terms this would be the canalization of either the phenotype along certain lines or canalization of genotype changes along certain lines. For example, evo-devo, where large sections of complex functionality is enabled, disabled, or mapped out onto a body plan by simple switches, is an example of a complex program being regularized. The switches are a regularized, and therefore potentially evolvable, interface for controlling complex systems, which may themselves be less evolvable.
- The original designation of RIC was incorrect (the system was not using Class 4 constructs, the solution did not require their use, or the overall machine was not Universal)

Therefore, designating features as having RIC tells researches what sorts of mechanisms need to be searched out if those features are shown to be evolvable. An example of this will be given at the end of section 3.1.

3. Applications of IC and RIC

3.1 Initial Applications to Biology. So far, relative irreducible complexity has only been discussed from a theoretical point of view, and its applications to biology may seem unclear. While this area should become more solid with further work, a general sketch of what sorts of features tend to produce Class 4 complexity can be given.

One of the main contributors to chaotic systems in computing are unstructured loops. By "unstructured loop" I mean a loop in which the looping mechanism is not cleanly separated from the computation which is performed within the loop body. As an illustration of the problem, Langdon and Poli (2006) have shown that for Von Neumann architectures (a subset of Turing-Complete machines whose loops are nearly all unstructured), the number of programs that even finish (much less do anything helpful) drops to a statistically insignificant percentage of the possible search space as the program size increases.

In biology, one area where loops are represented is in gene transcriptional regulation. Therefore, to apply this reasoning to a transcriptional network, an unstructured loop would be one in which:

- The function of the downstream pathway of the regulated gene is dependent on the gene being repressed at the appropriate time.
- The products which turn off the regulated gene are manufactured downstream of the gene itself.

There are likely many other instances which would be potentially chaotic (see de Figueiredo et al. 2002 and Andrecut and Kauffman 2007 for examples of how simple a system can be and still exhibit chaotic behavior), but this one is useful for examination because it is apparent how it relates to computer-based unstructured loops. The implementation of this feedback loop requires that the downstream product match the regulator region of the gene in question. It also requires that the gene correctly activate or repress the downstream product. Lack of repression would leave the gene locked in an active mode. As more distance is placed between the gene and its regulator, the ability for it to come together by happenstance decreases dramatically, as all intermediates have to link up as well.

On the other hand, a more directly evolvable loop would be a simple, autoregulated molecular negative feedback loop.

It is more evolvable because there is only one binding site to worry about it. In addition, because there is only one feedback interaction, gradual changes in affinity will produce generally gradual changes in output. As more loop components are added which separate the loop from the control, the interplay between the evolution of the genes and its affect on the regulatory network become more complex.

An example autoregulated feedback loop is the control of arginine production in *Escherichia coli*. The transcription factor, ArgR, is autoregulated. While ArgR controls downstream products, none of those downstream products are involved in the regulation of ArgR (Shen-Orr et al. 2002). Therefore, the autoregulation of ArgR is potentially directly evolvable.

An example of a relatively irreducibly complex mechanism, then, would be the control of the flagellar assembly in the bacterium *Psuedomonas aeruginosa*, which uses a multilevel control system to regulate the formation of the flagellum. FleQ is a transcription factor that regulates a number of other genes used in flagellar assembly. One of the downstream products of the assembly is FleN. FleN interacts with FleQ to deactivate it, preventing multiflagellation (Dasgupta et al. 2003). The regulation of FleQ is done downstream of FleQ itself, making a step-at-a-time evolution of the pathway extremely difficult.

Pallen and Matzke (2006) argue for the exaptational origin of the flagellum. As we've shown, just the FleQ/FleN pathway makes the evolution of this system solely by natural selection unlikely. However, that does not completely nullify the argument of exaptation. Because we lack total knowledge, this system is an RIC system. However, as discussed in section 2.6, this leaves open a few possibilities for its evolution. If it is evolvable, then it means that the traversed sequence space has been somehow regularized. An analogous (though not functionally homologous) way of looking at the possible evolution is to compare it to the V(D)J recombination system in which specific gene regions, designated as either *variable* (V), *diversity* (D), or *joining* (J) regions of the immune system, are randomly selected and assembled. In the V(D)J recombination system, the formation of immunoglobulin genes is facilitated by recombination signal sequences (RSSs), which mark segments of functionality. These, in turn, are then assembled in a regularized way, and the whole process resembles a computer metaprogram—a program which generates other programs (Bartlett 2006). These pathways are not deterministic, but information is the main driving force in their generation. The RSSs provide the information within the genome to guide the recombination towards likely functional paths. The FleQ/FleN pathways (and others) could be evolved through an analogous system which put together pieces of functionality based on templates. Rigoutsos et al. (2006) have claimed to have found gene sequences that match such a description. Whatever the exact mechanism, the RIC concept indicates that although an unguided evolution of the flagellum by exaptation is unlikely, it would be possible if the evolution was regularized in some way.

In absence of a quantitative model, it is qualitatively reasonable to view a gene that is regulated by its downstream products as relatively irreducibly complex. However, a future quantitative model that allows a more precise designation would be desirable.

3.2 Application to Fitness Landscapes. A fitness landscape is a conceptual framework for understanding the way in which natural selection operates to improve the fitness of populations. The idea of a fitness landscape is to imagine a landscape with rolling hills, some steep mountains, some valleys, and some flat plains. Each biological configuration is a point on that landscape, with similar biological configurations occupying nearby physical points to each other, and the elevation of the ground at each point represents that configuration's fitness. The goal is to conceptually model the way in which undirected mutations can achieve improved fitness.

The notion of fitness landscapes was proposed by Sewall Wright (1931). Wright noted the problem of rugged landscapes—that is, if the peaks of a landscape are separated by huge canyons, then evolution is not likely to occur via natural selection, because natural selection would remove organisms in a deep valley, and prevent a gradual move to the peak on the other side (Wright 1932). Wright's conception of fitness landscapes is slightly different than modern conceptions. For Wright, the landscape itself was defined by the traits already available in the population. For Wright, mutations altered the landscapes themselves. In modern conceptions of evolutionary landscapes, the potential mutational range is part of the evolutionary landscape as well, with changes to the environment being the main cause for the landscape to change.

Starting with Wright, the biological issues for fitness landscapes are how fitness landscapes should be characterized (rugged, smooth, etc.) and how and if they get crossed during a population's evolution. Irreducible complexity describes, from an informational standpoint, the difference between rugged and smooth landscapes. Peaks which do not require the usage of class 4 characteristics of machines to reach are relatively smooth, while those which do require such usage are relatively rugged. RIC describes landscapes that are rugged without the assistance of additional mutational mechanisms.

Environmental changes, while they do alter the evolutionary landscapes, do not assist in finding complex solutions. The main issue with complex solutions is not the environment, per se (though that is a part of it), but whether or not they are internally viable and coherent. For instance, in the flagellum example, the FleQ/FleN interaction was not interacting with the environment, but instead providing internal consistency to the flagellar assembly. If the flagellum was misformed, it is difficult to imagine how an environmental change would cause that to be a benefit. Much less likely would be a sequence of malformed flagella, with each change being matched by an environmental change which made it more beneficial. As is evident from this scenario, the steepness of the slopes of IC structures has little to do with the environment, and is instead dependent on the complexity of the interactions of the internal components.

3.3 Application to Creation Systematics. One important aspect of creation systematics is determining the lines of variation that occur within created kinds. Most modern creation research points to most created kinds being exceptionally diverse. Many of the adaptations that are present within the various species within the kind are very complex and coordinated, and many would be classifiable as RIC. One high-level example would be the neck of the giraffe. Bergman (2002), for instance, pointed to the neck of the giraffe in terms that echo irreducible complexity, saying (quoting Gould), "the long neck must be associated with modifications in nearly every part of the body—long legs to accentuate the effect, and a variety of supporting structures (bones, muscles, and ligaments) to hold up the neck." Bergman further points out the further coordination with the giraffe's circulatory system, including complex, unique valves and special tissue below the brain to regulate the sudden rush of blood to and from the brain. Such a necessary coordination of changes to achieve the result makes the giraffe's neck likely RIC. However, the giraffe is probably part of the same created kind as the okapi (a typically short-necked ruminant). If this is true, then, because both the giraffe and the okapi diverged from their common ancestral pair on Noah's Ark, these coordinated features unique to giraffes instead appear to be "evolvable" in the sense used in this paper.

The theoretical framework presented here would indicate that if complex, coordinated features such as these have evolved, that they would require either information from the outside (obtained through many possible mechanisms, including symbiosis, horizontal gene transfer, or by God's direct involvement), or that their genomes are formatted in such a way as to facilitate development along these lines. Creation systematics would then ask questions such as, which types of RIC structures usually arise from outside information? Which ones usually arise from facilitated evolution (e.g., pre-programmed evolutionary potential) within different created kinds? How do the patterns of variation of RIC structures relate to the purposes for which God designed each kind?

RIC helps creationists in identifying which features need design-oriented explanation (as opposed to being explainable in their entirety via historical contingency), and then patterns of RIC structures can then be used to map out the evolutionary directions of each created kind. Williams has pointed out the potential link between evolutionary capacity and God's purposes for each created kind (Williams 2005). RIC can help delineate the important patterns of variation which could then be used to identify the purposes of different created kinds.

Going back to the case of the giraffe, if the giraffe's neck did evolve, then, because it is RIC, this means that it likely had a source of information which guided the change. If so, the next question for creation biology to ask is how does that source of information reflect the purpose for which that kind was created?

3.4 Irreducible Complexity and Avida. The Avida software program has been used as a refutation of the notion of irreducible complexity (Pennock 2005). Therefore, it is useful to analyze both the results of Avida simulations and the Avida code itself according to this framework. Avida works by evolving digital "organisms," where the Avida organisms are programs that perform logic functions and replicate themselves. Avida has been used to show that evolution can evolve new functions as exaptations from previously-existing functions (Lenski et al. 2003). Avida works by taking an initial population of organisms, and subjecting them to mutation and selection, and seeing if they can begin to calculate new functions. Because we can have full knowledge of Avida programs (the instruction set is fully understood, and the programs are small enough to readily comprehend), the concept of relative irreducible complexity is not needed, and we can directly assess IC within Avida's digital organisms.

As we have noted, one of the requirements of IC is that the machine has enough degrees of freedom to solve arbitrarily hard problems—that is, the machine must be Turing-complete. Avida is, in fact, Turing-complete. Another of the requirements for IC is that the IC program must require the chaotic constructs of its machine. In practical terms, we noted that this is usually equivalent with open-ended loops. Avida organisms evolve to accomplish certain tasks. The list of tasks which Avida can test for are a small, finite set (Ofria 2007). All of these tasks are trivial logic functions, and do not require loops at all to compute. Therefore, none of the possible targets within Avida require an IC system to be solved. However, there is a portion of the Avida code which does involve an open-ended loop—the replication function. All Avida organisms have within them the code to replicate themselves, and this is performed via an open-ended loop.

Therefore, if we were to analyze Avida based on the creation systematics outlined above (modified slightly since we are dealing with IC instead of RIC), we would find that the replication loop was Irreducibly Complex, and therefore part of the original creation. The code for the logic functions, since they do not have open-ended loops contributing to the result, can be explained in terms of historical contingency. As supportive of this framework of irreducible complexity, this is precisely the case for Avida-the logic functions develop as a historically contingent process while the replication loop was implanted by Avida itself in the initial population of organisms.

Interestingly, the original Avida paper notes that without the proper environmental setup, even the evolution that did occur would have failed (Lenski et al. 2003). This suggests a question that should be noted for future creation research - to what degree did God use the environment to direct the historically contingent aspects of evolution? This question is far beyond the scope of this paper, but nonetheless is an interesting one.

4. Summary, Limitations, and Further Work

This paper has introduced a modified conception of Irreducible Complexity founded upon computational theory rather than intuitional assumptions about how complex systems respond to change. This conception of IC is based, not on the brittleness of the system, but rather the computational complexity of the components required to implement it. This includes a list of necessary conditions for a system to be IC (see section 2.5).

In addition, because our knowledge of biology is partial and not complete, Relative Irreducible Complexity is introduced as a way of expressing that a system may be IC given certain assumptions. If an RIC system is then shown to be evolvable, then the list of necessary conditions for IC gives a direction for productive research, by pointing out that there is likely a regularizing influence guiding the system's evolution.

For creation biology, RIC can give us information about the purpose of a created kind. By identifying RIC structures, and then showing which ones are evolvable, we can look for mechanisms of evolutionary change which regularize complexity. If such a mechanism of evolutionary change is confined to one or a few created kinds, then it can lead to a better understanding of the purposes of that kind.

So far we have dealt entirely with qualitative descriptions of IC and RIC. It is likely that IC is more than a true/false existence test, but may be quantitatively measurable on a continuum. Therefore, further research will work out quantitatively how different kinds of features contribute to RIC. Specifically, an analysis of the common three- and four-protein network motifs identified by Yeger-Lotem et al. (2004) on their contribution to RIC would be helpful for creation research. Aldana et al. (2006) has made some progress on this by showing how certain generalized network parameters (such as network connectivity and level of gene expression) can contribute to chaos and order in a network system, but little has been done in the way of numerically relating small-scale constructs to the possibility of chaotic results.

In addition to the issue of producing a quantitative version of IC, a more theoretical difficulty deals with the model of IC itself. In addition to digital computers, analog/digital hybrids are another possible model which has not been explored, as well as quantum computers. However, it is not very likely that these would be able to bypass the chaotic factors which have already been presented, though they may change exactly which factors are applicable and to what degree they are chaotic.

REFERENCES

Aldana, M., E. Balleza, S.A. Kauffman, and O. Resendiz. 2007. Robustness and evolvability in genetic regulatory networks. *Journal of Theoretical Biology* 245:433-448.

Alonso, P.D. 2004. The avian nature of the brain and inner ear of *Archaeopteryx*. *Nature* 430:666-669.

Andrecut, M. and S.A. Kauffman. 2007. Chaos in a discrete model of a two-gene system. *Physics Letters A* 367:281-287.

Bartlett, J.L. 2006. Metaprogramming and genomics. *Occasional Papers of the BSG* 8:17-18.

Behe, M. 1996. *Darwin's Black Box*. Free Press, New York.

Bergman, J. 2002. The giraffe's neck: another icon of evolution falls. *TJ* 16(1):120-127.

Camazine, S., J.L. Deneubourg, N.R. Franks, J. Sneyd, G. Theraulaz, and E. Bonabeau. 2001. *Self-Organization in Biological Systems*. Princeton University Press, Princeton.

Cuvier, G. 1834. *Cuvier's Animal Kingdom: Arranged According to Its Organization, Translated from the French, and Abridged for the Use of Students*. Orr and Smith, London.

Darwin, C.R. 1859. *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life*, 1st ed. John Murray, London.

Dasgupta, N., M.C. Wolfgang, A.L. Goodman, S.K. Arora, J. Joyt, S. Lory, and R. Ramphal. 2003. A four-tiered transcriptional regulatory circuit controls flagellar biogenesis in *Pseudomonas aeruginosa*. *Molecular Microbiology* 50: 809-824.

Dembski, W.A. 2004a. Irreducible complexity revisited. Available online at http://www.designinference.com/documents/2004.01.Irred_Compl_Revisited.pdf.

Dembski, W.A. 2004b. The myths of darwinism. In: Dembski, W.A., ed. *Uncommon Descent: Intellectuals Who Find Darwinism Unconvincing*. ISI Books, Wilmington.

de Figueiredo, J.C.B., L. Diambra, L. Glass, and C.P. Malta. 2002. Chaos in two-loop negative feedback systems. *Physical Review E* 65:051905-1 - 051905-8.

Gould, S.J. 1991. *Bully for Brontosaurus: Reflections in Natural History*. W. W. Norton and Company, New York.

Henderson, I.R., P. Owen, and J.P. Nataro. 2002. Molecular switches—the ON and OFF of bacterial phase variation. *Molecular Microbiology* 33:919-932.

Kauffman, S. A. 1993. *The Origins of Order: Self-Organization and Selection in Evolution*. Oxford University Press, New York.

Langdon, W.B. and R. Poli. 2006. The halting probability in Von Neumann architectures. *Lecture Notes in Computer Science* 3905:225-237.

Lenski, R.E., C. Ofria, R.T. Pennock, and C. Adami. 2003. The evolutionary origin of complex features. *Nature* 423:139-144.

Lester, L.P. and R.G. Bohlin. 1989. *The Natural Limits to Biological Change*, 2nd ed. Word Publishing, Dallas.

McFarland, K.A. and C.J. Dorman. 2008. Autoregulated expression of the gene coding for the leucine-responsive protein, Lrp, a global regulator in *Salmonella enterica* serovar *typhimurium*. *Microbiology* 154:2008-2016.

Minnich, S.A. and S.C. Meyer. 2004. Genetic analysis of coordinate flagellar and type III regulatory circuits in pathogenic bacteria. *Proceedings of the Second International Conference on Design & Nature, Rhodes Greece*. Available online at http://www.discovery.org/a/2181.

Mivart, G.J. 1871. *On the Genesis of Species*. Macmillan, London.

Ofria, C., D.M. Bryson, and K. Nanlohy. 2007. The Avida digital life platform. Available online at http://avida.devosoft.org/.

Pallen, M.J. and N.J. Matzke. 2006. From the *Origin of Species* to the origin of bacterial flagella. *Nature Reviews Microbiology* 4:784-790.

Pennock, R.T. 2005. Day 3 (September 28), AM Session. Kitzmiller v. Dover Area School District. Available online at http://www.talkorigins.org/faqs/dover/day3am.html.

Purdom, G. 2006. The Intelligent Design Movement: Does the identity of the Creator really matter? *Answers Magazine* 1(1):18-21.

Rigoutsos, I., T. Huynh, K. Miranda, A. Tsirigos, A. McHardy, and D. Platt. 2006. Short blocks from the noncoding parts of the human genome have instances within nearly all known genes and relate to biological processes. *Proceedings of the National Academy of Science USA* 103:6605-6610.

Shen-Orr, S.S., R. Milo, S. Mangan, and U. Alon. 2002. Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nature Genetics* 31:64-68.

Turing, A.M. 1936. On computable numbers, with an application to the *Entscheidungsproblem*. *Proceedings of the London Mathematical Society* 42:230-265.

Williams, A. 2005. Inheritance of biological information—part III: control of information transfer and change. *TJ* 19:21-28.

Wolfram, S. 1983. Cellular automata. *Los Alamos Science* 9: 2-21.

Wolfram, S. 1984. Universality and complexity in cellular automata. *Physica D* 10:1-35.

Wolfram, S. 2002. *A New Kind of Science*. Wolfram Media, Champaign.

Wright, S. 1931. Evolution in Mendelian populations. *Genetics* 16:97-159.

Wright, S. 1932. The roles of mutation, inbreeding, crossbreeding, and selection in evolution. *Proceedings of the Sixth International Congress of Genetics* 1:356-366.

Yeger-Lotem, E. and 7 others. 2004. Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. *Proceedings of the National Academy of Science USA* 101:5934-5939.

Editor: Roger W. Sanders