War of the Weasels

An Evolutionary Algorithm Beats Intelligent Design

How an intelligent design theorist was bested in a public math competition by a genetic algorithm—a computer simulation of evolution.

DAVE THOMAS

In the summer of 2006, a different kind of war was waged on the Internet—a war between *computer programs* written by both evolutionary scientists and by intelligent design (ID) advocates. The war came to a climax in a public math competition in which dozens of humans stepped forward to compete against each other and against genetic ("evolutionary") computer algorithms. The results were stunning: The official representative of the intelligent design community was outperformed by an evolutionary algorithm, thus learning Orgel's Second Law—"Evolution is smarter than you are"—the hard way. In addition, the same IDer's attempt to make a genetic algorithm that achieved a specific target without "specification" of that target was publicly exposed as a rudimentary sham. And finally, two pillars of ID theory, "irreducible complexity" and "complex specified information" were shown not to be beyond the capabilities of evolution, contrary to official ID dogma.

Genetic Algorithms

"Genetic algorithms" (GAs) are computerized simulations of evolution. They are used to study evolutionary processes and solve difficult (and sometimes intractable) design or analysis problems. Several novel designs generated with genetic algorithms have been patented (Brainz.org 2008). Evolutionary algorithms are currently used in a variety of industries to get effective answers to very difficult problems, including problems whose brute-force solutions would require *centuries*, even on superfast computers. In contrast, GAs can often produce highly useful results for the same problems in just a few minutes.

The basic idea for a genetic algorithm is simple. You start with a randomly generated "herd" of possible solutions to a given difficult problem, where the general structure of any conceivable solution can be represented with a chunk of memory in a computer program. Treat the members of this herd as "organisms," and test every herd member's performance with a *fitness function.* While the fitness function *can* be written in terms of proximity to a distant known "target," it is more often just a straightforward calculation of some parameter of interest, such as the length or cost of some component or feature, or perhaps the gain of a wire antenna. Any candidate organism can have its fitness readily measured, and the performances of any number of candidates can be impartially compared. The fitness test is commonly used to help decide which organisms get to be "parents" for the next generation of organisms. Throwing in some mutations, and letting higher-fitness organisms breed for a few hundred generations, often leads to surprising (and sometimes even astonishing) results.

Creationists and intelligent design proponents vigorously deny the fact that genetic algorithms demonstrate how the evolution of novel and complex "designs" can happen. They claim that GAs cannot generate true novelty and that all such "answers" are surreptitiously introduced into the program via the algorithm's fitness testing functions. The support for this claim stems mainly from a few pages of a book Richard Dawkins wrote nearly twenty-five years ago.

Dawkins and the Weasel

Creationists have been fixated for decades on Richard Dawkins's "Weasel" simulation from his 1986 book *The Blind Watchmaker* (Dawkins 1986). Unlike real genetic algorithms developed for industry or research, Dawkins's Weasel algorithm included a very precise description of the intended target. However, this precise specification was used only for a *tutorial demonstration of the power of cumulative selection* rather than for generation of true novelty. In the Dawkins example, the known target is the phrase from Hamlet, "Methinks it is like a weasel." The organisms are initially random strings of twenty-eight characters each. Every generation is tested, and the string that is closest to the target Weasel phrase is selected to seed the subsequent generation. The exact Shakespearean quote is obtained in just a few dozen generations. Despite Dawkins's explicit disclaimer that, in real life, evolution has no long-distance target, creationists of all varieties have latched on to "Weasel" as a convenient straw version of evolution that is easy to poke holes in.

The main ID theorist dealing with genetic algorithms is William Dembski, who stated the ID/creationist position as of September 2005 with these words:

> And nevertheless, it remains the case that *no genetic algorithm or evolutionary computation has designed a complex, multipart, functionally integrated, irreducibly complex system without stacking the deck by incorporating the very solution that was supposed to be attained from scratch* (Dawkins 1986 and Schneider 2000 are among the worst offenders here). (Dembski 2005)

Stephen Meyer is a top gun in the Discovery Institute's Center for Science and Culture, the Seattle-based center of ID pontification and promotion. In Meyer's "peer-reviewed" ID paper, "The Origin of Biological Information and the Higher Taxonomic Categories," he states:

> Genetic algorithms . . . only succeed by the illicit expedient of providing the computer with a target sequence and then treating relatively greater proximity to future function (i.e., the target sequence), not actual present function, as a selection criterion. (Meyer 2004)

Both Dembski and Meyer cite Weasel in these statements and go on to claim that *all* GAs are similarly targeted. And that is the gist of the formal ID response to genetic algorithms: paint them all with the Weasel brush, and pretend they all need predefined targets to work.

Steiner's Problem

In 2001, as I was preparing a response to an upcoming talk by ID's Phillip Johnson at the University of New Mexico, I decided to address the Weasel problem. I set out to develop a genetic algorithm of my own for solving difficult math problems, without using *any* specified target. I wanted something visual yet simple—a sort of miniature digital playground on the very edge of complexity. I ended up choosing "Steiner's Problem": given a two-dimensional set of points, find the most compact network of straight-line segments that connects the points (Courant and Hilbert 1941).

In Steiner's problem, there can be variable "Steiner points" in addition to the fixed points that are to be connected. If there are four fixed points arranged in a rectangle, the Steiner solution consists of five segments connected in a bowtie shape; each of the points on the rectangle's corners connects to one of two Steiner points in the interior of the rectangle, and a fifth segment connects the two Steiner points (figure 1).

Dave Thomas, a physicist and mathematician, is president of New Mexicans for Science and Reason and a fellow of the Committee for Skeptical Inquiry. He is currently a scientist/programmer at IRIS/PASSCAL in Socorro, New Mexico. E-mail: nmsrdave@swcp.com.

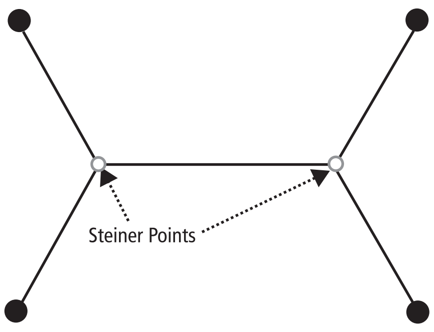

Figure 1: The bowtie, the Steiner solution for four fixed points (solid)

A Genetic Algorithm for Steiner's Problem

In my Steiner genetic algorithm, the organisms are represented by strings of letters and numbers—a kind of primitive "DNA." Two such DNA strands are shown in figure 2. The strands, when read by the transcription routine, supply three types of information about the network represented by each organism: the number of Steiner points, the numerical locations of these points, and a true/false connection map that dictates which points are to be connected by segments.

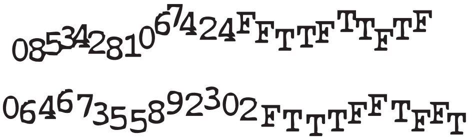

Figure 2: Two organisms represented by alphanumeric DNA

Steiner points can be placed anywhere in the region encompassing the fixed points; for these simulations, the region is a square with 999 units on a side. Length is measured in these units; for example, the length of the horizontal segment joining points (550,600) and (650,600) is 100 units.

Some representative networks for a six-point Steiner problem appear in figure 3. These are the "phenotypes" that correspond to the transcription of DNA (or the "genotype"). The fitness function used tests for two things: Are the fixed points all connected? What is the total length of all "expressed" segments? It's critical to emphasize that the fitness function need not have any descriptions of the actual Steiner solution for any given set of points. Fitness, here, is not based on any specific future function but only on present function. For example, the two organisms of figure 3 are clearly *not* the optimum Steiner solution for six fixed points (solid circles) in a rectangle. Yet, they can both easily be evaluated for *current function*. Here, the organism on the right is considerably shorter than the one on the left, and thus it has a better chance of having its "seed" continue on to the next generation. If an organism fails to connect all the given points, it is given a large "death" length of 100,000 units, making it extremely "unfit."

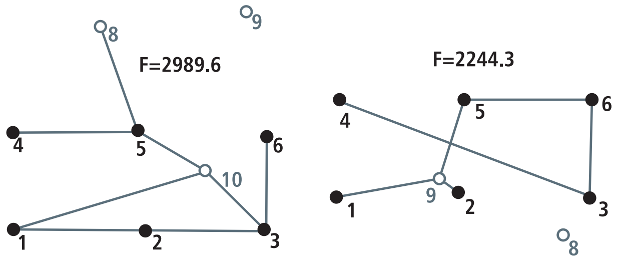

Figure 3: Two organisms shown as phenotypes (candidate networks after DNA transcription). While both organisms connect all fixed points, the right-hand creature is shorter and therefore more likely to be selected for breeding.

The Cyber Battles Begin

I posted a detailed discussion of this work on the Panda's Thumb blog (www.pandasthumb.org) on July 5, 2006. The point of that report was to demonstrate that genetic algorithms can solve difficult problems without knowing anything about the answer(s) in advance. I demonstrated that, while occasionally producing the correct (Steiner) solution, most of the time the algorithm converged on *imperfect* solutions. I called these "MacGyver" solutions, after the television hero who often found clever ways to get out of tough fixes. While the MacGyver solutions are clearly not the optimum Steiner shape, they get the job done efficiently and are often *within one percent of the length* of the formal Steiner solution itself. The GA operates by seeding the next generation with those organisms that are shorter in length in the current generation. This GA does *not*, as Meyer falsely claims, select for *future function* (a precise target) rather than for *present function* (here, the lengths of the digital creatures).

The ID community responded to my article by simply reiterating their claim that the solutions were secretly introduced via the fitness function. IDers are desperate to make Dawkins's Weasel the poster boy for all GAs, and they continue to paint all GAs as similarly "target-driven" or "front-loaded." Some ID theorists have tried to skirt the obvious lack of specific target description in the Steiner genetic algorithm by claiming that its virtual environment—the condition "shorter is better"—is really a description of the "precise target" itself. They say, "After all, you wanted shorter networks, and the Steiner solution is defined as the shortest network, so you are selecting for a specific target!"

This ID argument fails because the specific details of complex solutions are *not* explicitly imbedded in the overall design goals. To use an analogy, simply stating the objective "Build a vehicle that can carry men to the Moon and back" does not result in the spontaneous appearance of the complete plans for an Apollo spacecraft (with separate command, service, and lunar modules), along with a Saturn V launch vehicle.

The Collapse of the Pillars of ID Theory

One reason I chose Steiner's problem was that Steiner solutions possess "irreducible complexity" (IC) and also exhibit "complex specified information" (CSI), two features that intelligent design theorists claim are impossible via evolutionary processes. I contend that the results of the GA—both Steiners and MacGyvers—exhibit IC: if any segment is removed or rerouted, basic function of the system (here, connecting the fixed points) is lost completely. In addition, the Steiner solutions themselves are CSI, by virtue of their being complex (in the sense that the correct answer is rare enough to be improbable) and by virtue of their nature as specified information (as the formal solution to a given math problem).

ID proponents responded by claiming that the Steiner solutions discussed were "not really IC," even though these solutions obviously represent "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," the very definition of IC from Michael Behe's book *Darwin's Black Box* (Behe 1996). Behe goes on to claim that IC structures are impossible in gradual evolution (improvement by slight, successive modifications to precursor systems) "because any precursor to an irreducibly complex system that is missing a part is by definition nonfunctional."

The general ID response to my article was that the Steiner solutions could *not* be IC because they were derived from *ancestors* that were longer but still functional. So, the very existence of functional precursors is now being used to redefine irreducible complexity. IC apparently no longer has anything to do with the existence of critical, precisely interlocking components. This is classic goal-post movement. The concept of IC has become a useless tautology: if it's IC, it can't have evolved, and if it evolved, it can't be IC. Of course, Behe was thinking only about bottom-up evolution initially. In the Steiner GA, however, populations of organisms often become less complex through shedding of *redundant complexity*. This type of pathway to IC structures has been observed numerous times in nature.

The ID Version of a Genetic Algorithm

Bill Dembski's coauthor of his Uncommon Descent blog, software engineer Salvador Cordova, was the most prominent member of the ID community to weigh in on the series of GA articles. Cordova repeatedly misrepresented GAs as necessarily "front-loaded" and dismissed the results as "computational theatrics." On August 15, 2006, Cordova posted *his* code for a genetic algorithm, which he contended could solve for the sum of the first 1,000 integers *without specifying the answer*. He said this program was based on the same "theatrics" I was employing in my Steiner GA. However, I proved that his program was, despite copious amounts of smoke and mirrors, simply a direct method of specifying the answer, or target. Instead of matching the string "Methinks it is like a weasel," Cordova engineered his GA to converge on the specific target sequence 251, 252, 253, . . . 750. Cordova then added these 500 numbers and doubled that sum, inevitably arriving at the sum of the integers from 1 to 1,000, or 500,500. It was easy to prove that his badly written and confusing program was a direct encoding of a fixed target, leading directly to the summation of the first N integers.

The Design Challenge

On August 14, 2006, I posted a public "Design Challenge" on the Panda's Thumb blog in which readers were given one week to submit answers for the tricky six-point Steiner system shown in figure 4. It was an open-book test. Since the ID person responding to this discussion, Salvador Cordova, had been claiming that the answer was "front-loaded" into the fitness test, I challenged him to follow that lead to the answer.

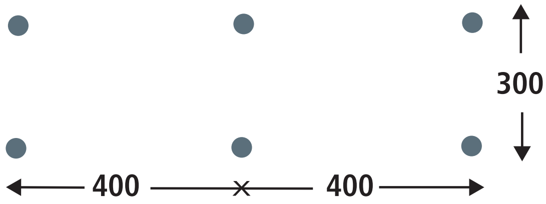

Figure 4: The public Design Challenge: Find the Steiner solution for the six-point grid shown.

I had come up with the six-point problem two days earlier, while trying to design a system that would have the "double bowtie" as its Steiner solution (figure 5). Upon reviewing an overnight batch of three-hundred runs, however, I was surprised to see solutions with lengths much *shorter* than the double bowtie's 1,839 units. And when I checked out the GA's best solution, the odd design shown in figure 6, it was like finding a diamond in the rough. I realized the GA had found the correct Steiner solution, and it *wasn't* what I had been expecting at all. Instead of the double bowtie, the actual Steiner solution twists both bowties a bit, and they become conjoined in a three-segment "dogleg" along the center vertical. There are two possible Steiners, one with the bowties skewed up and the other with them skewed down. The GA found both solutions, along with hundreds of compact MacGyvers.

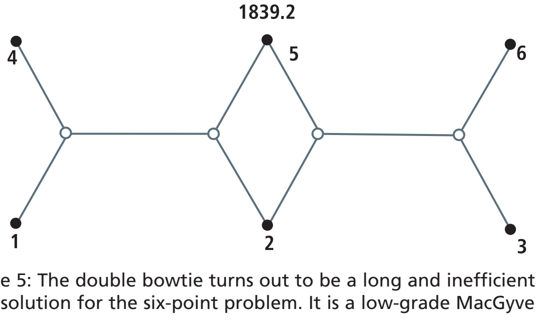

Figure 5: The double bowtie turns out to be a long and inefficient candidate solution for the six-point problem. It is a low-grade MacGyver.

Dozens of Panda's Thumb readers responded to the Design Challenge. Most were pro-science enthusiasts, but ID theorist Cordova submitted several candidate answers as well. Cordova

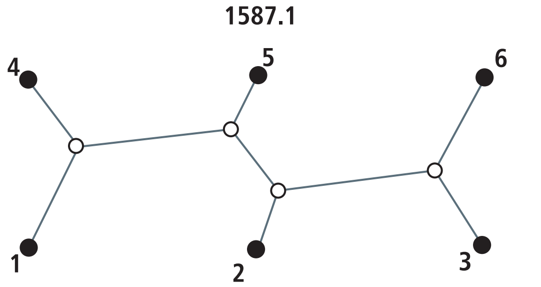

Fig. 6. The actual Steiner solution for the six-point problem has two bowties, twisted and conjoined in the middle, forming a hexagonal grid. It is a far more efficient solution than the double bowtie of figure 5.

had repeatedly compared the Steiner GA's fitness function to a T-shirt with a large bull's-eye emblazoned on it and the Steiner solution itself to the person inside that shirt. He analogized shooting a paintball gun at the bull's-eye symbol and then telling the victim, "Don't be mad, I wasn't aiming at you, I was aiming at the shirt you were wearing." Curiously, Cordova did not reverse-engineer my publicly posted GA (the shirt) to deduce the solution (e.g., the person wearing the shirt). Instead, he went the traditional route and tried to design an answer using Fermat points and trigonometry. Interestingly, Cordova failed to deduce the basic network shape for the six-point solution, finding instead the slightly longer MacGyver solution of figure 7. Fifteen other "intelligent designers" (humans, in other words) were able to derive the *correct* answer—the true Steiner solution. However, all of these humans were pro-science skeptics of intelligent design creationism. Correct solutions were also found by not one but two independent genetic algorithms! An additional fifteen designers derived various MacGyver solutions, thus proving these, too, are complex specified information.

And that's how ID theorist Cordova learned the true meaning of what Daniel Dennett terms Leslie Orgel's Second Law: "Evolution is smarter than you are."

After being bested by an evolutionary algorithm, Cordova changed his tune and moved the goalposts over to computer speed. He said there was no shame in being beaten by the computer because computers are designed to do lots of math very, very fast and are thus superior to humans in that regard. But that argument doesn't wash either. The computer can check out lots of random solutions very quickly (about 8,000 per second), but simply guessing randomly at the answer is a terrible way to solve the problem. After dozens of hours, random guessing couldn't come close to matching even one of the efficient designs the genetic algorithm was pumping out every ninety seconds (figure 8).

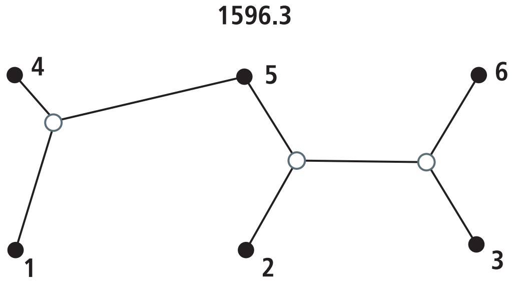

Figure 7: This shape is not the Steiner solution but is the closest second, or "best MacGyver." This solution was the final Design Challenge submission by ID theorist Salvador Cordova.

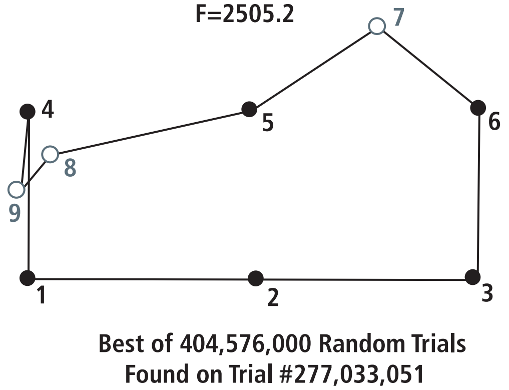
F=2505.2
7
4
5
6
8
9
1
2
3
Best of 404,576,000 Random Trials
Found on Trial #277,033,051

Figure 8: Random guessing is a totally inefficient method of trying to solve the problem. Even at a blazing 8,000 organisms per second, days of frantic computation don't even come close to producing what the GA can produce in just ninety seconds.

Conclusion

The 2006 "War of the Weasels" was, to say the least, not kind to the ID movement. The central dogma of ID regarding genetic algorithms—the Weasel offense—was definitively and publicly shot down. ID theory's two main "evolution stoppers"—irreducible complexity and complex specified information—were shown to be child's play for an evolution-based program that evaluates current function only and is mindless of any specific future optimum. Finally, an ID "theorist" was bested by a program that used evolution to derive solutions. Check out the complete archives of the War of the Weasels on the Panda's Thumb blog, www.pandasthumb.org, in the "Evo Math" category. □

References

Behe, Michael. 1996. *Darwin's Black Box*. New York: The Free Press.

Brainz.org. 2008. 15 real-world uses of genetic algorithms. Available online at http://brainz.org/15-real-world-applications-genetic-algorithms/.

Courant, Richard, and Herbert Robbins. 1941. *What is Mathematics?* London: Oxford University Press.

Dawkins, Richard. 1986. *The Blind Watchmaker*. New York: W.W. Norton and Company.

Dembski, William. 2005. Rebuttal to reports by opposing expert witnesses. Design Inference, May 14. Available online at www.designinference.com/documents/2005.09.Expert_Rebuttal_Dembski.pdf.

Meyer, Stephen. 2004. The origin of biological information and the higher taxonomic categories. *Proceedings of the Biological Society of Washington* 117(2): 213–239.

Online Extra!

For a fascinating look at real-world Steiner solutions, check out the Online Extra at www.csicop.org/SIExtras.