sceptical arguments that all the requisite vegetation could not have been alive on the land surface at the same time.[1] They then went on to propose that there had been large floating ecosystems comprising arboreal lycopods, supporting their contentions with evidences from the morphology of the plant life itself.[2] During the Flood, these *in situ* floating forests would have been washed into basins, which would have supplemented the vegetal material available for coal from forests growing on land.

Many years ago, I examined the same skeptical allegation, and proposed a different solution.[34] The biblioseptics, as it turns out, had actually set up a straw man. Contrary to their assumptions, the vegetation living at the start of the Flood had not been the only source of carbonaceous material which had eventually transformed into coal. A large amount of carbonaceous material must also have accumulated in the 1650 or so years between the Creation and Flood, in the form of peat. I showed that one cubic metre of peat has more organic carbon than a considerable quantity of vegetation. With only a small fraction of earth's land surface underlain with peat, and much of this peat reworked and deposited during the Flood, the requisite quantity of carbon for the earth's coal (and also oil) would have been readily met. And this does not even take into account the inorganic sources of carbon, which are demonstrated to exist.

The evidence for floating forests, presented by Scheven, and Schonknecht and Scherer, is certainly worthwhile to include in any Flood model, if only because of the morphology of the plant structures. However, it is not necessary to include the existence of floating forests as a solution to a nonexistent problem. As discussed in my above-cited study, land-dwelling forests combined with peat accumulation are sufficient to account for the total inventory of organic carbon stored in coal. Thus Schonknecht and Scherer's study actually serves to demolish the biblioseptics' mythological carbon-nonproblem a second time over.

John Woodmorappe
Chicago, Illinois
UNITED STATES OF AMERICA

References

1. Schonknecht, G. and Scherer, S., Too much coal for a young earth? *CEN Tech. J.* ll(3):278-282, 1997.

2. Scheven, J., The Carboniferous floating forest — an extinct pre-Flood ecosytem, *CEN Tech. J.* 10(1):70-81, 1996.

3. Woodmorappe, J., The antediluvian biosphere and its capability of supplying the entire fossil record, *Proceedings of the 1st International Conference on Creationism,* 2:205-218,1986. '

4. Identical article reprinted in Woodmorappe, J., *Studies in Flood Geology,* Institute for Creation Research, El Cajon, CA, 1999.

'Dawkins' weasel revisited'

While I heartily agree with Dr Truman's overall conclusions in the above article,[1] I must point out that there is an error in his understanding of the way Dawkins' algorithm works.

Truman's explanation of the algorithm is as follows:

'Simply envision 28 rings each with every letter of the alphabet and a blank space stamped on each ring, next to each other on a metal cylinder held horizontally. Spin all the rings one after the other or at the same time. Note the rings which show the characters or spaces facing you which match the target sentence. Spin the remaining unsuccessful rings until all the letters match the target.'[2]

The results of one 'run' of this algorithm are shown in Figure 1 as published in Truman's article.[3]

Note that in this instance I have highlighted in bold only letters which differ from those in the same positions in the sequences directly above any particular sequence (except for the starting sequence) and that these may not necessarily relate to the target sequence at all. The number of mutations refers to the number of letters which are different to those at the same positions in sequences directly above any particular sequence, so for example, 22 letters in trial #4 are different from those in trial #3. The reason for this will become clear shortly.

```
GOAL:        METHINKS  IT IS LIKE A WEASEL
STARTING: WDLTMNLT  DTJBKWIRZREZLMQCO P
TRIAL #1:  SEE  SNXD  ETHAIYGSWCWVFCQCQMZ    23 MUTATIONS
TRIAL #2:  FEIQGNIC  ATZTLMMXLTKKGGBVWIL    24 MUTATIONS
TRIAL #3:   ESNWNJE  YTNVQJZKIFUYDYQYUIL    22 MUTATIONS
TRIAL #4:  OEWEVNLO  BTBW ARZDKNYEWRGBYL    22 MUTATIONS
TRIAL #5:  NESBANZF  YTMEHUXGJXXLQWFZGAL    22 MUTATIONS
. . .
TRIAL #40: METHINKS  IT IS PIKE AEWECSEL    22 MUTATIONS
. . .
TRIAL#164:METHINKS  IT IS LIKE A WEASEL     3 MUTATIONS
```

Figure 1.   The results of one run of a Dawkins-type simulation performed by Truman.[1]

The output from the first run Dawkins published is shown in Fig. 2.[4] Highlighting and so forth is as above.

Truman is correct in stating that the initial 28 letter sequence is entirely random, although it is possible, even likely, that at least one letter in this sequence will have a match in the target sequence. However, this is where Dawkins and Truman part company. What Dawkins does is to 'breed' a number of progeny sequences (we are not told how many) from this starting sequence into which mutations are introduced. It *seems* that Dawkins forces every progeny sequence to have one mutation in it although it is possible that he gives every single letter a $^1/_{28}$ chance of mutation and simply breeds progeny upon this basis. The latter would mean that some progeny sequences possess no mutations whilst some would possess more than one. I actually suspect that Dawkins used the former method, but Dawkins does not give a detailed explanation of exactly how his algorithm works, so my conclusions are in a lot of cases educated guess-work.

Whatever the method Dawkins uses to generate mutant progeny, he then selects the best one on the basis of its similarity to the target sequence and uses this to breed the next generation of progeny. This continues until the target sequence is reached. In this respect, Dawkins' method does, in a *very* simplified way, model the process of DNA replication and mutation in real cells (obviously Dawkins uses an incredibly greater mutation rate, amongst other simplifications) whereas Truman's method is basically a 'rigged fruit-machine' affair which appears (although I am happy to be corrected on this point) to generate only one new sequence per generation.

Truman further states that any letters which match those in the target sequence are not subject to further mutation during breeding. Again, this is not something that is clear from Dawkins' book — it could be that he simply generated a large number of progeny from each sequence which obviously massively increases the likelihood of favourable mutations occurring whilst decreasing the likelihood of 'destructive' mutations which mutate letters already matched. Overall this would decrease the number of generations required to reach the target. However, I suspect that Truman is correct given that the number of generations required to reach the target *is* small: only 43, 64 and 41 generations respectively for the three examples Dawkins gives. This is much smaller than the number of generations Truman required: published average 102. Furthermore, the original program only took half an hour to run, even though it was written in 1986 using BASIC — probably the slowest programming language known to man. If Dawkins had bred a huge number of progeny in each generation, his program probably would have taken a lot longer to run.

So we can see that what Truman's algorithm does is actually quite different to what Dawkins' algorithm does, although the final results are comparable.

To be fair, this is a really easy mistake to make: I originally thought that Dawkins' algorithm works the way Truman seems to think it does. This inaccuracy obviously does to some extent call into question the conclusions arrived at by Truman.

Unfortunately my present (lack of) knowledge of statistics means that I do not feel qualified to comment on his mathematical proofs but I would suggest that they rest on an infirm foundation. Since this algorithm appears to be deterministic, it ought to be possible to describe its behaviour using a Boltzmann distribution which would yield the average number of generations required to reach the target for a given mutation rate, sequence length, and number of progeny generated. Surely this would conclusively demonstrate the deterministic nature of the algorithm. However, this is outside of my field of knowledge at the moment, so I will leave it to others to think about.

Bart Read
Nottingham
UNITED KINGDOM

References

1. Truman, R., Dawkins' Weasel Revisited, *CEN Tech. J.* 12(3):358-360, 1998.

2. Truman, Ref. 1, p. 358.

3. These results were also published as part of another article by Truman: Disappointing delusion (Review of Richard Dawkins' book, *Unweaving the Rainbow*), *CEN Tech. J.* 13(1):33-36, 1999.

4. Dawkins, R., *The Blind Watchmaker*, Penguin, UK, pp. 47-48 1991.

5. Note the '?' in the starting sequence which does not occur in the published text: I have added this because of an apparent typographical error in the published work which leads to the introduction of a 'frameshift' mutation meaning that all the letters from that point onwards are misaligned one position to the left.

Royal Truman replies:

In response to my article *Dawkins' Weasel Revisited, CEN Tech. J* 12(3):358-360, both a creationist and an evolutionist have pointed out that the computer simulation and mathematical treatment I published do not fully take into account the number of progeny each generation produces. Both respondents then implied that this casts doubt on the observation that one cannot fail to match *any* sentence selected in a ridiculously short number of iterations!

This claim is incorrect. On the contrary, increasing the number of mutant progeny to arbitrarily choose from serves only to *accelerate* the rate of convergence to any pre-selected goal.

In *The Blind Watchmaker,* Richard Dawkins recognizes that a team of monkeys typing randomly cannot generate Shakespeare's collected works in 20 billion years, so the notion of cumulative change is introduced. A sentence is selected, a random collection of letters is used as a starting

```
GOAL:      METHINKS   IT  IS  LIKE  A  WEASEL
STARTING:  WDL?MNLT   DTJBKWIRZREZ  LMQCOP
TRIAL #1:  WDLTMNLT   DTJBSWIRZREZ    MQCOP     1   MUTATION

TRIAL #10: MDLDMNLS   ITJISWHRZREZ  MECS  P     9   MUTATIONS

TRIAL #20: MELDINLS   IT  ISWPRKE  Z  WECSEL   10   MUTATIONS

TRIAL #30: METHINGS   IT  ISWLIKE  B  WECSEL    6   MUTATIONS

TRIAL #40: METHINKS   IT  IS  LIKE  I  WEASEL    4   MUTATIONS

TRIAL #43: METHINKS   IT  IS  LIKE  A  WEASEL    1   MUTATION
```

Figure 2. *Output from the first run of Dawkins' simulation.*[4,5]

point, then only the letter positions mismatched with the target are reshuffled for every 'generation'. The intermediate most closely matching the goal is flawlessly retained every time as the starting point for a new generation.

Now, Dawkins fails to inform us as to how many progeny permutations are generated to choose from. An astronomically large number would lead to instant success everytime. A smaller number would provide less opportunities to move in the correct direction. **I used the worst possible scenario in the paper** to show convergence is unavoidable: only one surviving mutant is produced per generation, offering a choice between it and an identical copy of the parent. The analysis, based on well-known binomial probability distribution formulas, revealed that any target sentence cannot fail to be matched, irrespective of length and starting letter configuration using Dawkin's constraints.

Clearly, **the only effect of extra mutants per generation is that the sequence converges more rapidly to the goal!** It becomes less likely to reproduce only misaligned letters. Also, occasionally several letters would be concurrently 'corrected' within that generational attempt. The proof lies in the much lower number of iterations that Dawkin's reported simulations required than mine, for which 10,000 simulations were run.

There are significant advantages to the approach published in this journal. It makes transparent that the example is indeed a *'rigged fruit-machine'* affair, which both respondents failed to recognize. In addition to the obvious flaws already discussed in my article, it becomes easy for others to add to this list. For example, if Dawkins' analogy had any validity, several other outcomes are predicted:

- Lengthening the target sentence results in a vanishingly small probability of finding any mutation which does *not* lead closer to the goal starting from the first generation! Consider a random sequence the length of an encyclopedia, to mimic a real genome. A single mutant progeny, to fail to make progress within that generation, would have to fail to match any of the millions of letter positions available. This is virtually impossible statistically. Worse still, Dawkins' example would allow perhaps another 1000 mutants to also try within the same generation period!

- As a long sentence converges to the target, eventually it would be far more likely that one of the letters already correctly lined up would misalign than that the one or two remaining letters should get lucky within that generation. This reflects the real-world constraint that there is a vastly greater number of nucleotide arrangements which can

prevent a gene from functioning than of successful configurations.

- The evolutionary assumption, that a novel, complex function missing in an original population has evolved to produce it in a descendant population has been allegedly demostrated by Dawkins through the following 'proof': a path over time of discrete and viable mutations is *assumed* to exist and a mechanism is *assumed* to exist which cannot fail to converge to any and every outcome by permuting the original genome. As pointed out in the journal article, accepting that kind of argument and the assumptions would make it possible to 'prove' anything since the outcome can never fail to be met.

The objections offered only strengthen the argument that Dawkins' example is irrelevant to the real world.

Royal Truman
Ludwigshafen
GERMANY

Cainan in Luke 3:36: insight from Josephus

Because Luke 3:36 has the extra name Cainan in the genealogy compared with Genesis 10 and 11,