THE GENESIS OF GENETIC PROGRAMMING: A FRONTIERSMAN'S TALE

(Richard S. Forsyth, September 2016)

http://www.richardsandesforsyth.net/

Abstract

This article takes a participant-observer's look back at the genealogy of the computational method now known as Genetic Programming (GP for short). In so doing, it treats GP as a case study for elucidating the process of technical innovation. Working on the assumption that the contrast between sudden Eureka and stepwise improvement is a polarity rather than a sharp dichotomy, it introduces a simple technique for identifying the main steps in the march of GP from margin to mainstream. It is argued that this approach could be applied more widely to other areas of scientific or technological advance -- possibly even offering the prospect of resolution to some of the more belligerent academic-priority disputes.

Phraseological Preamble

Genetic Programming (also GP) will be familiar to most readers of this journal, both as a family of computational techniques and an academic discipline concerned with studying, extending and applying such techniques. This article is an account of the history of GP from genesis to maturity which endeavours to highlight some of the less well known aspects of that history.

This de-familiarization process begins with the name. I suppose that many readers see nothing strange in the fact that GP refers to a topic that gets discussed at conferences attended by computer scientists rather than biologists. Yet the earliest usage of the term "genetic programming" I could find using Google was part of the following extract.

> "The consequence of the ensuing encounter with the impinging multifactored environmental complex may be death, as the inevitable outcome of inadequate **genetic programming**; or it may be survival, with the genes then co-operatively spelling out the individual developmental tendencies." -- Joranson, P.N. *Pulp and Paper Magazine of Canada*, 59, 1958, p. 193.

Here it is clear that the term refers to the programming of an organism's development by its genotype -- viewing **genotypes as programs**. To me this seems a very natural way to interpret the phrase.

Under another plausible interpretation, arguably also more natural than the one that is now widespread among computer scientists, the phrase would refer to the deliberate programming of actual genetic material. This has recently moved from science-fictional speculation to the realm of practical biotechnology, with the likes of Craig Venter creating synthetic microbes from laboratory-assembled components.

Of course, in our field, we consider GP to be a form of computational search applied to structures that can be executed as computer programs, optimized using a simplified analogue of Darwinian evolution. The Wikipedia entry (ranked first when I typed the term into Google) puts it as follows.

> "**genetic programming** (**GP**) is a technique whereby computer programs are encoded as a set of genes that are then modified (evolved) using an evolutionary algorithm." -- Wikipedia, accessed 25 August 2016.

We accept a meaning that considers **programs as genotypes**. Thus we are stuck with what I believe an educated outsider would regard as a third-choice interpretation of the term, but it is too well entrenched now to dislodge.

Having established our field of discourse, I would like to offer my preferred definition of GP, that of Bäck et al. (1997).

> **"Genetic programming** applies **evolutionary search** to the space of **tree structures** which may be **interpreted as computer programs** in a language suitable to modification by **mutation and recombination**." Bäck, T., Hammel, U. & Schwefel, H-P. (1997). *IEEE Transactions on Evolutionary Computation*, 1(1), 3-17. [emphasis mine]

Since 1997, the idea of expressing GP results as tree structures has been generalized, for example, to linear and heterarchic structures (e.g. Miller & Thomson, 2000), while retaining the essential aspects of tree structures, namely variable length and multiple levels.

Who, When and How?

GP in this sense is today an established academic field, with numerous conferences, journals and research groups. It is instructive to chart its history, from emergence to acceptance, which, in my view, illuminates some important issues in the process of scientific and technical innovation.

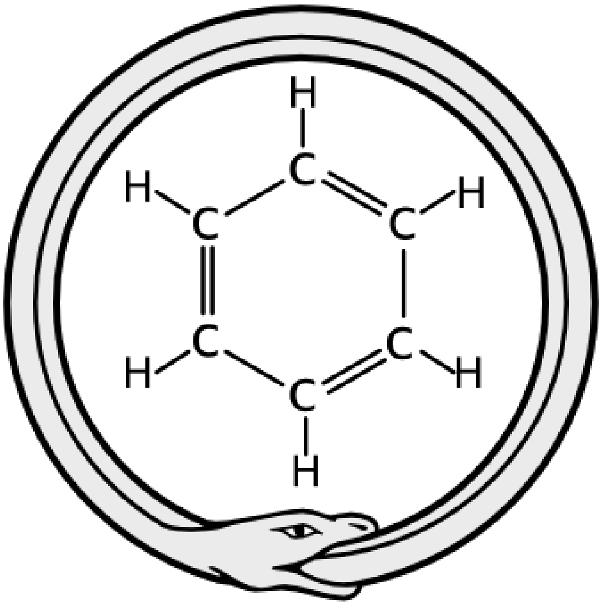
It is no secret that priority is a hotly coveted prize among scientific researchers. To be the first to devise, discover or invent something that becomes widely accepted is a high form of success. It leads to kudos, promotion and respect from colleagues and peers. Therefore priority disputes are common, dating back at least as far as Newton's quarrels with Hooke and Leibniz. Most such disputes are eventually resolved by a creeping consensus that settles on a single party in the debate, at least as far as the textbooks are concerned.

For example, scholars may publish articles in learned journals that examine the diverse formulations by a handful of chemists in the nineteenth century (such as Newlands and Meyer) who devised tabulations exhibiting patterns of relationship among chemical elements; but for the purposes of educating chemistry students or explaining the field to the wider world there was just one originator of "the" periodic table, namely Dmitri Mendeleev. Moreover, its inception can be dated to a single day, 17 February 1869, after apparently coming to Mendeleev in a dream (Scerri, 2011).

This is how we like our scientific history, in legendary style, with a single hero on a single date breaking through from darkness into light. The dream is optional, though it does help to have a quirky detail to make the story memorable.

The classic of this type is the "Eureka Moment" of Archimedes in his bath, or rather jumping out of it once he had worked out how to measure the density of a golden crown, which according to the legend turned out to be an alloy of gold and silver.

Other legendary examples include Galileo dropping balls from the top of the leaning tower of Pisa, Newton watching an apple fall in his mother's orchard and Kekulé realizing the structure of the benzene molecule after a reverie on the upper deck of a horse-drawn bus about a snake eating its tail.



Picture credit: https://commons.wikimedia.org/wiki/File:Benzene_Ouroboros.png

It is quite possible that all four of these famous Eureka Moments are apocryphal, but they do a good job of presenting scientific innovation as a sudden flash of insight. Very much in the same spirit is the following quotation from *Popular Science* (Keats, 2006).

> "In 1987 Koza was on an airplane, returning to California from an AI conference in Italy, when he had the crucial insight ... Koza was 30,000 feet above Greenland when he asked himself why a genetic algorithm, so adept at refining pipelines, couldn't be used to evolve its own software." -- Popular Science, April 2006.
> http://www.popsci.com/scitech/article/2006-04/john-koza-has-built-invention-machine

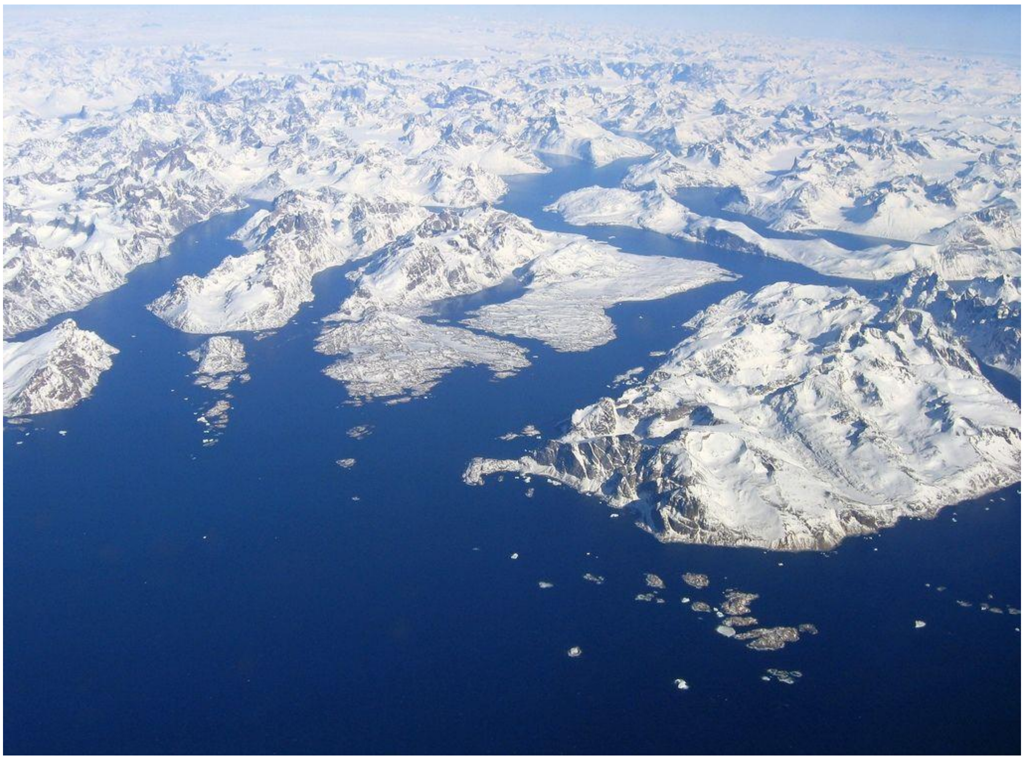


Photo credit: Stig Nygaard, https://www.flickr.com/photos/stignygaard/448189871/sizes/o/

It is easy to imagine that viewing Greenland's rugged terrain below through a porthole might well have set off ideas about algorithms for traversing "fitness landscapes", but does this moment in summer 1987 represent the birthday of what we now call GP? More important, does GP have a single identifiable starting point at all?

This goes to the root of the question of whether our appetite for dramatic moments of inspiration distorts our understanding of scientific advance. Many historians of science and some scientists have argued that scientific innovation is generally a much more cumulative process than depicted in our memorable myths. The prevalence of priority disputes (Merton, 1957) suggests that sudden breakthrough by an individual genius is not in fact the norm. Indeed Merton (1961) has argued that "multiples", i.e. near-simultaneous discoveries by unconnected researchers, are more normal.

Returning to Mendeleev, for example, Eric Scerri (1998) has raised this point in connection with the periodic table (or "system") of chemical elements.

> "The discovery of the periodic system for classifying the elements represents the culmination of a number of scientific developments, rather than a sudden brainstorm on the part of one individual. Yet historians typically consider one event as marking the formal birth of the modern periodic table: on February 17, 1869, a Russian professor of chemistry, Dimitri Ivanovich Mendeleev, completed the first of his numerous periodic charts." -- Scerri, E., 1998, p. 78.

An elegant statement of the polarity from Eureka moment to incremental innovation is made by Gunther Stent (1972), with reference to real genetics rather than simulated genes inside a computer.

> "I believe that if Watson and Crick had not existed, the insights they provided in one single package would have come out much more gradually over a period of many months or years. Dr. B might have seen that DNA is a double-strand helix, and Dr. C might later have recognized the hydrogen bonding between the strands. Dr. D later yet might have proposed a complementary purine-pyrimidine bonding, with Dr. E in a subsequent paper proposing the specific adenine-thymine and guanine-cytosine nucleotide pairs. Finally, we might have had to wait for Dr. G to propose the replication mechanism of DNA based on the complementary nature of the two strands. All the while Drs. H, I, J, K and L would have been confusing the issue by publishing incorrect structures and proposals." Gunther S. Stent (1972). *Scientific American*, 227, p. 90.

In my view, this is a realistic summary of the point at issue (although I would want to bestow the title of Professor on Drs. H, I, J, K and L). Clearly Stent accepts that scientific advance can proceed either by sudden large leaps or in gradual small steps. In the rest of this article, I will take this as given, and examine where GP falls along this polarity, and in so doing develop a simple method which I propose can be employed to elucidate the trajectories of other cases of scientific or technical innovation.

From Idea to Implementation

As might be expected there was a delay between having the concept of GP and embodying that concept in executable computer code. The basic idea of having a computer somehow evolve its own programs is known to have occurred to several people in the years since Turing (1948) wrote of evolutionary search as a route to machine intelligence.

> "There is the genetical or evolutionary search by which a combination of genes is looked for, the criterion being the survival value" -- Turing (1948) p. 16.
> http://www.alanturing.net/turing_archive/archive/l/l32/L32-019.html

Ironically, this report was dismissed as a "schoolboy essay" by none other than Charles Darwin (Copeland, 2012) and remained unpublished until 1969 (Meltzer & Michie, 1969). The Darwin who regarded Turing's ideas on evolutionary search as unworthy of publication was Turing's superior at the National Physical Laboratory, Sir Charles Galton Darwin, grandson of the great naturalist. However, the basic idea behind GP did find its way into print several times before Koza's 1987 flight over Greenland.

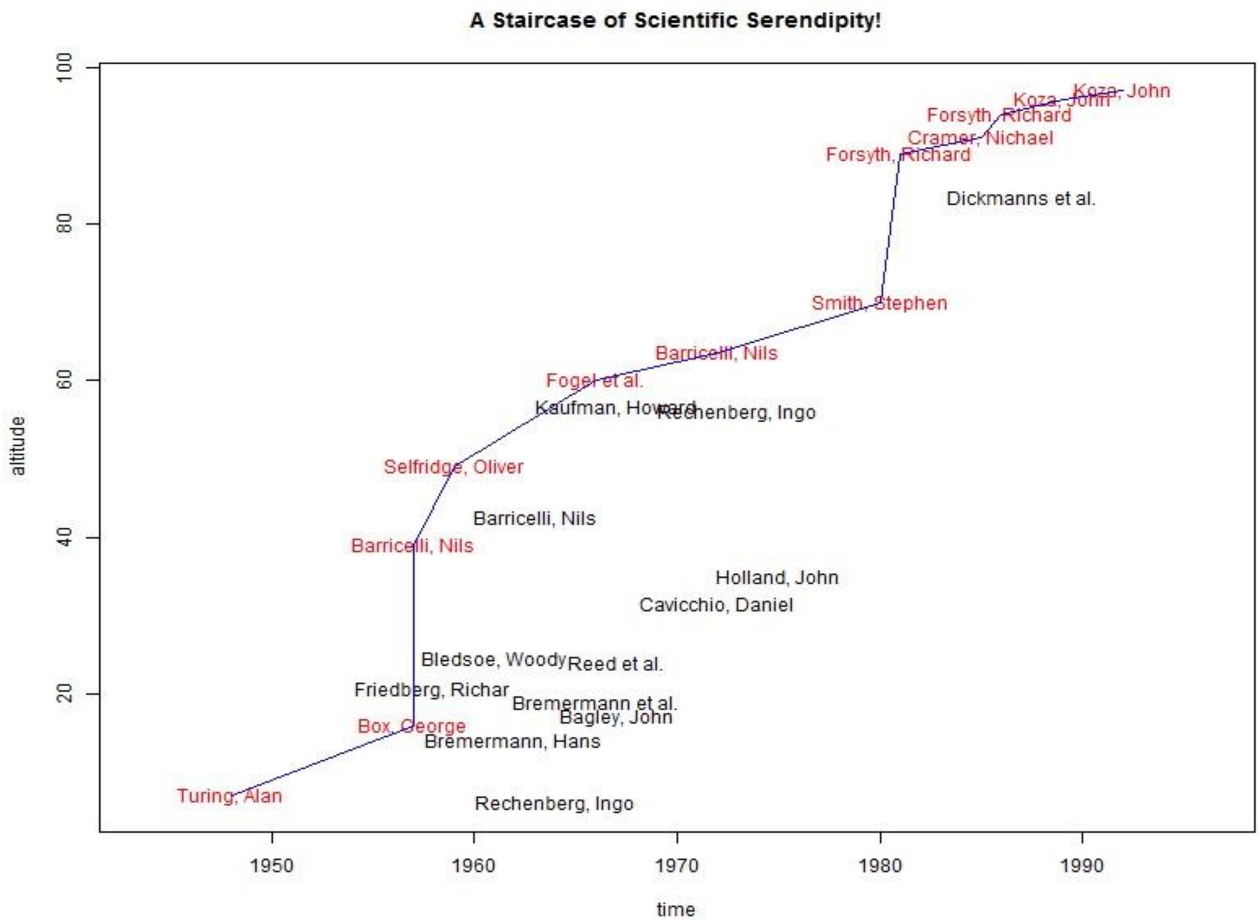
> "The scheme sketched is really a natural selection on the processing demons. If they serve a useful function they survive, and perhaps are even the source for other subdemons who are themselves judged on their merits." -- Selfridge, O. (1959). p. 14.
> "...the generation procedure operates in parallel fashion, producing sets or populations of programs" -- Holland, J.H. (1962). p. 298.
> "symbioorganisms will consist of numbers, and numbers in the machine can be interpreted as instructions according to any arbitrary code which can be established by writing an interpretive program." -- Barricelli, N. (1963). p. 2.
> "Thus, nonregressive evolution proceeds to find better and better programs for attacking the problem in hand." -- Fogel, L., Owens & Walsh (1966). p. 12.
> "...this highlights the fact that the rules are really programs in a special-purpose language, which might lead to the conclusion that the system should ultimately generate LISP functions." -- Forsyth, R. (1981). p. 165.

Thus the conceptual ingredients of what became GP were "in the air" for many years before it was put into practice. This article, however, concentrates not so much on the concept as on its implementation in working software, on the principle that the Wright brothers rather than Leonardo da Vinci are generally recognized as the first to launch powered heavier than air flying. (Actually, they are not universally recognized: the history of powered flight turns out on investigation to be more a case of many small hops than one grand take-off; but that's another story.)

The Ascent of GP

I am not a historian of science, nor indeed a historian of any kind. However, when I received an invitation late in 2015 to give a keynote address at the EvoStar Conference in Porto in March 2016, I felt it incumbent upon me to do some historical investigation, if only to clarify my position in the story of GP. It was clear that I had been invited as a voice from the early pioneering days of GP, on the basis of being the author of BEAGLE, a rule-finder inspired by Darwinian principles (Forsyth, 1981). The justification for inviting me to stand up in front of 200 researchers knowledgeable about evolutionary computing was the notion that BEAGLE was arguably the first working GP system. Informally, I had occasionally made that claim myself, but would it stand up to serious scrutiny?

Oddly enough, there doesn't seem to be a generally agreed "scientific" way of settling priority questions in science. So I devised my own framework. Readers can judge its strengths and weaknesses from what follows. But this is not a detective mystery, so let's start by spoiling the punchline and revealing the result. Below is a graph depicting the ascent of GP from 1948 (when it was just a glint in Alan Turing's fertile imagination) to 1992 (when Koza's 819-page tome presented the world with what amounted to a mature technology).



This plot shows 25 names positioned on 2 axes. The names are authors of papers or reports describing evolutionary computing systems which possessed attributes we now consider distinctive of GP systems. (Actually, 24 of the papers describe working systems, one (Turing, 1948), is included out of respect and as an initial benchmark, since although it prefigures the entire field it doesn't describe a working program.)

The horizontal axis, time, is relatively straightforward: it is the year when the paper or report was published (except Turing's, dated when it was written). The vertical axis is named "altitude", alluding among other things to that famous flight over Greenland. It requires further explanation (below).

The hues, red or black, depend on whether the position (of the middle of the name) lies on the *efficient* or *Pareto frontier*, indicated by the blue connecting line. A point lies on this frontier if no other point is both earlier and higher in altitude. This division into positions on and off the efficient frontier inevitably does carry evaluative implications, so its basis needs to be made explicit, as will be attempted shortly. One caveat that should be made immediately is that none of these published researchers were engaged in a "contest" to score a high altitude rating. They were describing findings and explaining methods to fellow workers in their fields. In other words, even if we equate altitude on this graph with proximity to full-fledged modern GP, that is not what most of them were trying to achieve. To avoid misinterpretation, it should be stressed that a point lower on this graph could well have more value as a contribution to the development of evolutionary computation -- in some respects -- than a higher point, even if we accept that the graph is valid within its terms of reference.

A Spreadsheet of Serendipity

The graph above required me to track down and read more than 25 articles of potential relevance to the development of GP. In this task I took books by Goldberg (1989) and Fogel (1998) as well as William

Langdon's Genetic Programming Bibliography (http://www.cs.bham.ac.uk/~wbl/biblio/) as my starting points. To assess whether, or more accurately to what degree, an article described a GP system required compiling a list of defining or distinctive attributes of GP systems. To compile this list I began with the definition given by Bäck et al. (1997), quoted earlier, as well the distinctive features listed by Kinnear (1994):

- Tree-structured heritable material
- Variable-length heritable material
- Executable heritable material
- Syntax-aware crossover

I added the stipulations that the article should describe a working computer system and that it should have an obvious Darwinian or evolutionary basis. However, it soon became apparent that I was not going to find a clear-cut short list of necessary and sufficient conditions to determine conclusively whether an article described a GP system or not. This prompted me to settle on a weighted-sum model. This involved 12 features, weighted according to their perceived importance, including the four from Kinnear (1994) above, which between them accounted for 70% of the total weighting. The full list, with weightings, can be seen in the Appendix.

These weighted features then became columns in a simple spreadsheet where the rows represented the 25 articles selected, ordered by publication year. The spreadsheet can be accessed at
http://www.richardsandesforsyth.net/software/
by anyone who wants to look at the details or experiment with differing assumptions, such as altered weights or additional/alternative attributes. (Save GPsheet.xlsx from the above location.)

All that remained to be done was to fill in the cells of this grid, i.e. to decide whether or not (1 or 0) the system concerned exhibited the attribute in question. If you obtain the spreadsheet, you will see that 30 of the 300 cells contain the number 0.5, meaning that in 10 percent of these supposedly binary decisions I was unable to make a firm judgement. This isn't wholly due to lack of decisiveness or understanding on my part; rather it underlines the influence of John Koza. Since the influential books by Koza (1992; 1994), a kind of template for describing systems of this nature in print has become customary. But earlier researchers weren't working with shared assumptions that in describing a GP system one would normally record explicitly the mode of crossover, the rate of mutation, the number of generations, the allowable operators and so on. Even parameters like population size were not always made explicit.

Thus, to the extent to which my choice of attributes, weightings and the rest is reasonable (which remains to be further discussed), this simple spreadsheet gives us a perspective on the development of an important subfield of computer science. This perspective does not force us to pick a single Eureka moment as the definitive start date. It gives us a more nuanced view; indeed a view from several angles, so to speak, since it is possible to adjust many of the parameters involved to observe their effects on the overall picture.

When it comes to the matter of assigning credit for discoveries, the efficient frontier does provide relevant information. I think it does make sense to regard the researchers on the efficient frontier as contributors to the main line of development of GP. It shows that GP arose from a process involving 12 particularly significant contributions by 11 researchers (nine authors and two co-authors). I suspect this is a worthwhile corrective to the mythic view of the solitary genius receiving a bolt of mystical inspiration. As long as we remember that the points off the efficient frontier are not failures but may themselves be contributions to a different line of development, we will not be tempted to make a simplistic division into sheep versus goats.

As regards the issue of small steps versus large leaps, some might feel tempted to note that the biggest jump on the graph shown above is associated with Barricelli (1957) and the second biggest with Forsyth (1981); but doing so is surely placing more weight on the assumptions behind the model than they can sensibly bear.

Limitations
It is fair to point out that this exercise has several limitations. In the first place, it relies heavily on my individual judgement. Human judgement is a fine and necessary thing, but it is well attested (see, for instance, Kahneman, 2011) that it is vulnerable to bias and prejudice. Multi-person judgement, if it can be elicited in conditions designed to minimize the chance of "groupthink", is usually more reliable than that of a single individual. Thus it would help if more people were involved. This is perfectly possible in principle. In high-stakes cases the same sort of study could be arranged to deliver more reliable results, with the author-by-attribute grid combining many experts' judgements. It is essentially a question of time and effort. In the present instance, I had the motivation to put some time in, and a participant's point of view, which has advantages as well as drawbacks.

The attributes chosen in this instance are open to challenge, as are their weightings. Here again, involving more people would help. It is easy to envisage a panel of experts debating such choices until a consensus emerges. Again this is a matter of resources, rather than a flaw inherent in the method. In the present case, the worksheet is public and open to amendment. It provides a framework for debate.

There is also the need to guard against forming a misleading impression from the efficient frontier of some sort of unbroken single mainstream. It not a pedigree chart or even a relay race where a baton is passed from hand to hand. It does not imply that, for instance, Barricelli (1972) had read Fogel et al. (1966), or that Fogel et al. knew of the work of Selfridge (1959). Even a perusal of citations would not establish that conclusively. It merely links papers that, in a sense, made an unanticipated advance towards modern GP, or, strictly speaking, towards GP as defined by a particular set of characteristics.

In addition, there is always the possibility that (for example) a ground-breaking paper in Russian or Japanese from the 1960s which has languished unread by the English-speaking world could be rediscovered and change the whole shape of the progress chart shown above. This merely means that any conclusions reached from this study remain provisional. The more people involved in an exercise such as this, the less likely it would be that a major contribution was overlooked, thus the more credibility that could be attached to its findings.

Finally, the article by Dickmanns et al. (1987) is in German and my grasp of German is exceedingly tenuous. I would be grateful if a German-speaker could check and revise that row of the spreadsheet. More generally, if anyone wants to help improve this spreadsheet, currently hosted at
http://www.richardsandesforsyth.net/docs.html
please feel free to contact me. Ideally it could be hosted on a public forum and become a crowd-sourced repository of consensus opinion on this topic. I would be interested in hearing from anyone who wants to work towards such an objective.

Conclusion
This article has presented a pilot study, limited in scope. Its specific conclusions are open to various challenges and therefore highly tentative. Nevertheless, it introduces what I believe to be a novel but readily intelligible technique for approaching the vexed question of academic priority. With more resources, and appropriate amendments, this technique could be applied to shed light on the important, and often contentious, subject of innovation in a variety of scientific and technical fields.

Acknowledgement
I would like to thank Dr James McDermott for making helpful suggestions for improving an earlier draft of this article.

Declaration of Interest: The foregoing describes a piece of small-scale curiosity-driven research, not free from individual bias. As will have become apparent, I have a personal interest in rescuing BEAGLE (Forsyth, 1981) from oblivion. Hence my motivation for writing the present article. Nevertheless I have endeavoured to take a relatively dispassionate view of the topic.

References

Bäck, T., Hammel, U. & Schwefel, H-P. (1997). Evolutionary computation: comments on the history and current state. *IEEE Transactions on Evolutionary Computation*, 1(1), 3-17.

Bagley, J.D. (1967). *The Behaviour of Adaptive Systems which Employ Genetic and Correlation Algorithms*. Doctoral Dissertation, University of Michigan.

Barricelli, N.A. (1957). Simbiogenetic evolution process realized by artificial methods. *Methodos*, 9(35-36), 143-182.

Barricelli, N.A. (1962). Numerical testing of evolution theories. Part I. *Acta Biotheoretica*, 16(1/2), 69-98.

Barricelli, N.A. (1963). Numerical testing of evolution theories. Part II. *Acta Biotheoretica*, 16(3/4), 99-126.

Barricelli, N.A. (1972). Numerical testing of evolution theories. *Journal of Statistical Computation and Simulation*, 1, 97-127.

Bledsoe, W. W. (1961). The Use of Biological Concepts in the Analytical Study of Systems. *Technical Report PRI 2, Panoramic Research, Inc., Palo Alto, California*. [Also 1961, Presented at ORSA-TIMS National Meeting, San Francisco, California, November 10.]

Box, G. (1957). Evolutionary operation: a method for increasing industrial productivity. *Applied Statistics*, 6(2), 3-23.

Bremermann, H.J. (1962).Optimization through evolution and recombination, Part 1. In: M.C. Yovits et al. (1962) eds. *Self-Organizing Systems*. Washington, DC: Spartan Books.

Bremermann, H.J., Rogson, M. & Salaff, S. (1966). Global properties of evolution processes. In: H.H. Pattee et al. (1966) eds. *Natural Automata and Useful Simulations*, 3-42. London: Spartan-McMillan.

Cavicchio, D.J. (1972). Reproductive adaptive plans. *Proceedings of the ACM 1972 Annual Conference*, 1-11.

Copeland, B.J. (2012). *Turing: Pioneer of the Information Age*. Oxford: Oxford University Press.

Cramer, N.L. (1985). A representation for the adaptive generation of simple sequential programs. In: J.J. Grefenstette. (1985) ed. *Proc. International Conference on Genetic Algorithms and their Applications*, 24-26 July, Carnegie-Mellon U., Pittsburgh.

Dickmanns, D., Schmidhuber, J. & Winklhofer, A. (1987). *Der genetische Algorithmus: Eine Implementierung in Prolog*. Fortgeschrittenenpraktikum, Institut f. Informatik, Lehrstuhl Prof. Radig, Tech. Univ. Munich, 1987. http://people.idsia.ch/~juergen/geneticprogramming.html

Fogel, D.B. (1998) ed. *Evolutionary Computation, the Fossil Record*. Piscataway, NJ: IEEE Press & Wiley-Blackwell.

Fogel, L.J., Owens, A.J. & Walsh, M.J. (1966). *Artificial Intelligence through Simulated Evolution*. New York: Wiley.

Forsyth, R.S. (1981). BEAGLE -- a Darwinian approach to pattern recognition. *Kybernetes*, 10, 159-166.

Forsyth, R.S. & Rada, R. (1986). *Machine Learning: Applications in Expert Systems and Information Retrieval*. Chichester: Ellis Horwood.

Friedberg, R.M. (1958). A learning machine. Part I. *IBM Journal of R & D*, 2(1), 2-13.

Goldberg, D.E. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. Reading, Massachusetts: Addison-Wesley Co.

Holland, J.H. (1962). Outline of a logical theory of adaptive systems. *Journal of the ACM*, 9(3), 297-314.

Holland, J.H. (1975). Adaptation in Natural and Artificial Systems. Ann Arbor: U. Michigan Press.

Joranson, P. N. *(*1958*).* Improvement of hardwoods through *genetics*. *Pulp and Paper Magazine of Canada*, 59(10), 193-201.

Kahneman, D. (2011). *Thinking, Fast and Slow*. London: Penguin Books.

Kaufman, H. (1967). An experimental investigation of process identification by competitive evolution. *IEEE Trans. Systems Science & Cybernetics*, SSC-3(1), June 1967, 11-16.

Keats, Jonathon (2006). John Koza has built an invention machine. *Popular Science*, 19 April 2006.

http://www.popsci.com/scitech/article/2006-04/john-koza-has-built-invention-machine
Kinnear, K.E. (1994) ed. *Advances in Genetic Programming*. Cambridge, Massachusetts: MIT Press.
Koza, J. (1992). *Genetic Programming: on the Programming of Computers by Means of Natural Selection*. Cambridge Massachusetts: MIT Press.
Koza, J. (1994). *Genetic Programming II: Automatic Discovery of Reusable Programs*. Cambridge Massachusetts: MIT Press.
Meltzer, B. & Michie, D. (1969) eds. *Machine Intelligence 5*: Edinburgh: Edinburg University Press.
Merton, R.K. (1957). Priorities in scientific discovery. *American Sociological Review*, 22(6), 635-659.
Merton, R.K. (1961). Singletons and multiples in scientific discovery: a chapter in the sociology of science. *Proc. American Philosophical Society*, 105, 470486.
Miller, J.F., & Thomson, P. (2000). Cartesian Genetic Programming. In R. Poli, W. Banzhaf, W.B. Langdon, J.F. Miller, P. Nordin, and T.C. Fogarty (eds.) Genetic Programming, *Proceedings of the 4th European Conference (EuroGP 2000),* 121–132, LNCS, Berlin: Springer-Verlag.
Rechenberg, I. (1965). *Cybernetic Solution Path of an Experimental Problem*. Royal Aircraft Establishment. Library Translation 1122.
Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Stuttgart: Frommann-Holzboog Verlag.
Reed, J., Toombs, R. & Barricelli, N.A. (1967). Simulation of Biological Evolution and Machine Learning I. *Journal of Theoretical Biology*, 17, 319-342.
Scerri, E.R. (1998). The Evolution of the Periodic System. *Scientific American*, September 1998, 279, 78-83.
http://www.scientificamerican.com/article/the-evolution-of-the-periodic-system/
Scerri, E.R. (2011). *The Periodic Table: a very short Introduction*. Oxford: Oxford University Press.
Selfridge, O.G. (1959). Pandemonium: a paradigm for learning. *Proceedings of Symposium held at The National Physical Laboratory*, November 1958, HMSO.
Smith, S. (1980). *A Learning System Based on Genetic Adaptive Algorithms*. PhD Thesis, University of Pittsburgh.
Turing, A.M. (1948). Intelligent Machinery. *Unpublished NPL Report*.
http://www.alanturing.net/intelligent_machinery/
Wikipedia (2016). Genetic Programming. https://en.wikipedia.org/wiki/Genetic_programming

Appendix

The table below describes the attributes used to characterize each contribution to the development of GP. It also gives the attributes' weightings as in the spreadsheet used to give the 2-dimensional plot shown above. The score labelled "altitude" on the vertical axis of that plot was computed as a weighted sum

$$A_i = D_i \times \sum(W_j \times C_{ij})$$

where A_i is the altitude score of item i, C_{ij} is 0 or 1 (occasionally 0.5) indicating whether characteristic j belongs to item i and W_j is the weighting assigned to characteristic j. D_i acts as a gatekeeper: it is the 0/1 score on the first attribute, "Clear Darwinian Basis", which is treated exceptionally. Ideally this should be 1 for every row in the worksheet, since if the paper concerned didn't employ an evolutionary approach, it would be discussing some other kind of optimization. In theory, therefore, it should be redundant. However, there was a borderline case (Friedberg, 1958) which received 0.5 on this attribute.

It will be seen that the first five attributes are dominant: the first acts as a switch, the next four contribute 70% of the total weighting.

Attribute	Weight	Description
Clear Darwinian basis	*	1 if the system uses an analogue of evolution by natural selection (0.5 in doubtful case)
Variable-length heritable material	15	1 if the structures being evolved can vary in length, zero otherwise
Tree-structured heritable material	20	1 if the structures being evolved have a tree-like form, zero otherwise
Syntax-aware crossover	10	1 if the crossover operator must know how to slice the structures being evolved at syntactically appropriate boundaries, zero if it operates blindly
Population members executable as programs	25	1 if the structures being evolved are executable expressions, zero otherwise
Population size exceeds two	10	1 if the size of the population of structures being evolved is 3 or more, zero otherwise (some early systems used tiny populations)
Uses crossover operation	5	1 if the system employs crossing of 2 or more parental structures to create novel structures, zero if mutation of only a single parent is used ('sexual' versus 'asexual' reproduction)
Uses mutation operator	2	1 if mutation (some kind of random change) is used in generating new structures, zero otherwise
Export of executable software	5	1 if the structures generated can be exported as software to be executed externally from the generating system, zero otherwise
Genotypes incorporate looping	2	1 if the structures being evolved can express a fundamental programming construct, namely repetition of a section, zero otherwise
Explicit submodule generation	2	1 if the structures being evolved can incorporate another fundamental programming construct, namely a generated subroutine, zero otherwise
System applied by others than originator	2	1 if the generating system was used by others, zero if it was only used by its originator
Selection at topmost level	2	1 if the system used the so-called "Pittsburgh" approach, in which the whole structure is subject to evolutionary optimization; zero if it used the so-called "Michigan" approach, in which only some portions of an overall structure are subject to evolutionary optimization

Note: As mentioned earlier, in cases of doubt, some papers were given a rating of 0.5 on some of these supposedly binary attributes.