

Algorithms, Games, and Evolution*

Erick Chastain¹, Adi Livnat², Christos H. Papadimitriou³, and Umesh V. Vazirani⁴

- 1 Computer Science Department, Rutgers University
erick@cs.rutgers.edu
- 2 Department of Biological Sciences, Virginia Tech
Blacksburg, VA 24061, USA
adi@vt.edu
- 3 Computer Science Division, University of California, Berkeley
Berkeley, CA, 94720, USA
christos@cs.berkeley.edu
- 4 Computer Science Division, University of California, Berkeley
Berkeley, CA, 94720, USA
vazirani@cs.berkeley.edu

Abstract

Even the most seasoned students of evolution, starting with Darwin himself [1], have occasionally expressed amazement at the fact that the mechanism of natural selection has produced the whole of Life as we see it around us. From a computational perspective, it is natural to marvel at evolution's solution to the problems of robotics, vision and theorem proving! What, then, is the complexity of evolution, viewed as an algorithm? One answer to this question is 10^{12} , roughly the number of sequential steps or generations from the earliest single celled creatures to today's Homo Sapiens. To put this into perspective, the processor of a modern cell phone can perform 10^{12} steps in less than an hour. Another answer is 10^{30} , the degree of parallelism, roughly the maximum number of organisms living on the Earth at any time. Perhaps the answer should be the product of the two numbers, roughly 10^{42} , to reflect the total work done by evolution, viewed as a parallel algorithm.

Here we argue, interpreting our recently published paper [2], that none of the above answers is really correct. Viewing evolution as an algorithm poses an additional challenge: recombination. Even if evolution succeeds in producing a particularly good solution (a highly fit individual), its offspring would only inherit half its genes, and therefore appear unlikely to be a good solution. This is the core of the problem of explaining the role of sex in evolution, known as the "queen of problems in evolutionary biology" [3].

The starting point of [2] is the diffusion-equation-based approach of theoretical population geneticists [4], who analyze the changing allele frequencies (over the generations) in the gene pool, consisting of the aggregate of the genetic variants (or "alleles") over all genes (or "loci") and over all individuals in a species. Taking this viewpoint to its logical conclusion, rather than acting on individuals or species or genes, evolution acts on this gene pool, or genetic soup, by making it more "potent", in the sense that it increases the expected fitness of genotype drawn randomly from this soup. Moreover, for much genetic variation [5], this soup may be assumed to be in the regime of weak selection, a regime where the probability of occurrence of a certain genotype involving various alleles at different loci is simply the product of the probabilities of each of its alleles. In this regime, we show in [2] that evolution in the regime of weak selection can be formulated as a game, where the recombining loci are the players, the alleles in those loci are possible moves or actions of each player, and the expected payoff of each player-locus is

* This work was partially supported by National Science Foundation Grant CCF-1064785, National Science Foundation Grant CCF-0964033, Templeton Foundation Grant 39966, and National Science Foundation Grant CCR-0905626.



precisely the organism's expected fitness across the genotypes that are present in the population. Moreover, the dynamics specified by the diffusion equations of theoretical population geneticists is closely approximated by the dynamics of multiplicative weight updates (MWUA) [6].

The algorithmic connection to MWUA brings with it new insights for evolutionary biology, specifically, into the question of how genetic diversity is maintained in the presence of natural selection. For this it is useful to consider a dual view of MWUA [7], which expresses “what each gene is optimizing” as it plays the game. Remarkably this turns out to be a particular convex combination of the entropy of its distribution over alleles and cumulative expected fitness. This sheds new light on the maintenance of diversity in evolution.

All of this suggests that the complexity of evolution should indeed be viewed as 10^{12} , but for a subtle reason. It is the number of steps of multiplicative weight updates carried out on allele frequencies in the genetic soup. A closer examination of this reveals further that the accurate tracking of allele frequencies over the generations requires the simulation of a quadratic dynamical system (two parents for each offspring). Moreover the simulation of even simple quadratic dynamical systems is known to be PSPACE-hard [8]. This suggests that the tracking of allele frequencies might require large population sizes for each species, putting into perspective the number 10^{30} . Finally, it is worth noting that in this view there is a primacy to recombination or sex, which serve to provide robustness to the mechanism of evolution, as well as the framework within which MWUA operates.

1998 ACM Subject Classification F.1.1 Models of Computation

Keywords and phrases evolution, recombination, coordination games, multiplicative weight updates

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2014.45

Category Invited Talk

References

- 1 Darwin, F. ed. 1887. The life and letters of Charles Darwin, including an autobiographical chapter. Vol. 2. London: John Murray.
- 2 Chastain, E., Livnat, A., Papadimitriou, C., Vazirani, U. “Algorithms, Games, and Evolution,” PNAS, vol. 111, no. 29, 10620–10623, 2014.
- 3 Bell, G. 1982. The Masterpiece of Nature: the Evolution and Genetics of Sexuality. University of California Press, Berkeley.
- 4 Kimura, M. 1985. The Neutral Theory of Molecular Evolution. Cambridge University Press.
- 5 Nei, M. 2005. “Selectionism and neutralism in molecular evolution,” Mol. Biol. Evol., 22:2318–2342.
- 6 Arora, S., Hazan E., Kale S. “The Multiplicative Weights Update Method: A Meta-Algorithm and Applications,” *Theory of Computing*, 8(1): 121–164, 2012.
- 7 Orecchia L., Mahoney M. “Implementing Regularization Implicitly Via Approximate Eigenvector Computation,” Proc. 28th Int’l Conf. Machine Learning, pp. 121–128, 2011.
- 8 Arora, S., Rabani, Y., Vazirani, U., “Simulating quadratic dynamical systems is PSPACE-complete,” Proceedings of the 26th Annual ACM Symposium on Theory of Computing, pp. 459–467, 1994.