One Operator, One Landscape

Terry Jones
Santa Fe Institute
1399 Hyde Park Road
Santa Fe, NM 87501, USA
terry@santafe.edu

January 1995

Abstract

The use of the term "landscape" is increasing rapidly in the field of evolutionary computation, yet in many cases it remains poorly, if at all, defined. This situation has perhaps developed because everyone grasps the imagery immediately, and the questions that would be asked of a less evocative term do not get asked. This paper presents an important consequence of a new model of landscapes. The model is general enough to encompass most of what computer scientists would call search, though it is not restricted to either the field or the viewpoint. The consequence is a "one-operator, one-landscape" view of search algorithms that is particularly relevant for algorithms that search via the use of multiple operators, and hence to genetic algorithms and other members of the evolutionary computing family. Crossover and selection landscapes are presented as siblings of the traditional mutation landscape. The model encourages a perspective on search algorithms that makes a clear division between landscape structures and navigation upon them. This division is the basis for the design of new search algorithms that combine elements of existing algorithms, an example of which is a crossover hillclimber. The model also establishes a strong connection with the heuristic state space search algorithms of Artificial Intelligence.

Submitted to the *Twelfth International Conference on Machine Learning.*

---

1 Introduction

The biological metaphor of a *fitness landscape* was introduced by Wright in the early 1930s [19]. Recently, Manderick et al. [12], drawing on work in physics by Weinberger [16], brought landscapes to the attention of the genetic algorithm (GA) community. The metaphor has also been widely used in other fields, including anthropology, chemistry, economics and immunology. Despite this attention and the clear and important results of Manderick et al., no formal model of fitness landscape exists that is appropriate for consideration of evolutionary algorithms (EAs). This claim will be supported in the following section.

This paper presents an important consequence of a new model of landscapes reported in [9]. This model generalizes that used by Weinberger and others, and allows a parsimonious view of GAs and other EAs that is also consistent with our understanding of other search algorithms from Artificial Intelligence (AI) and Operations Research (OR). In this model, a landscape is a graph whose edges are determined by the choice of an operator. Every operator, in combination with other choices (including representation and fitness function), creates a landscape graph. When a search algorithm makes use of the operator, it is traversing an edge of the landscape graph. From this perspective, one can adopt a low-level view of the working of a GA that involves not one, but *three* landscapes that are being simultaneously explored in a complex interrelated fashion. A high-level view in which a GA is seen as taking steps on a landscape graph whose vertices correspond to entire populations is in no way precluded or discouraged—this is the perspective adopted by work on GAs as Markov processes.

In the remainder of this paper, I argue why we need a formal model of landscapes, show how landscapes can be generated for operators that act on and/or produce multiple individuals, illustrate this with crossover and selection landscapes, describe the workings of a GA from this perspective, argue for a view of search as comprised of structures (landscapes) and navigation, show that this point of view has long been held in AI, and show how this point of view can be used to consistently discuss hybrid algorithms, to construct new algorithms, and as a basis for general landscape statistics.

2 Why Do We Need A New Model of Landscapes?

To answer this, it is important to understand why we are interested in employing the word "landscape" in the first place. The answer to this is presumably that we hope to use the imagery commonly associated with the term. By imagining or visualizing algorithms as operating on landscapes, we hope to increase our understanding of their behavior through consideration of peaks, local optima, ridges and other landscape-related terms. In addition, the imagery might help in the conceptual design of new algorithms, or raise other interesting questions about processes operating on these structures. A good definition of a landscape may also provide a foundation for mathematical analysis, such as the calculation of statistics predicting properties of the landscape. To support a claim that we need a new model, it is necessary to argue that the current notions of landscapes are not doing a sufficiently good job. A claim that a new model of landscapes should be adopted needs to demonstrate why the new model is better. I will argue that the current use of the term—both informal and formal—in discussing GAs and other EAs leaves much to be desired. I will address the problems that arise in both situations, arguing that folkloric notions of landscapes are often badly erroneous and that the existing formal models, while sufficient for the purposes they were constructed, are not appropriate for the study of EAs.

Before outlining some problems with current notions of landscapes, it is important to establish that to define a landscape implies the need for a definition of neighborhood. Perhaps the most frequently used landscape-related term is "peak." But what is a peak? A simple definition is a point whose neighbors are all less high than it is. It is not possible to sensibly define the word "peak" without defining what constitutes neighborhood. In fact, virtually all terms related to landscapes have this requirement.

The first problem with current notions of landscapes is that the need to be specific about neighborhood is not widely attended to. As a result, in many cases, it is not even clear what is meant by the word "landscape." The expressions "fitness function" and "fitness landscape" are often used interchangeably, both formally and informally. A fitness *function* is a function: a mapping from one set of objects to another. There is no notion of neighborhood, and one is not needed. A fitness *landscape* requires a notion of neighborhood: $f(x) = x^4 - 2x + 1$ does *not* define a fitness landscape, it is simply a function. Discussion of landscapes, both informal and formal, rarely mentions how neighborhood is defined.

A second problem is that even when neighborhood is defined, it is often done so incorrectly. Informally, people will staunchly defend the hypercube as *the* landscape when an algorithm processes bit strings, even if the algorithm never employs an operator that always flips exactly one bit chosen uniformly at random in an individual. Certainly this hypercube may have vertices that fit the above definition of a peak, but the relevance of these points to such an algorithm is far from clear. Even when mutation is employed in a GA, the mutation operator does *not* induce a hypercube graph. The mutation operator used in a GA does not always alter a single bit in an individual and hence the mutation landscape graph is not a hypercube. For binary strings of length n, the graph is the complete graph K_{2^n} augmented with a loop from each vertex to itself. With a per-bit mutation probability of p, the bidirectional edge between two vertices, v_1 and v_2 with Hamming distance d is labeled with the transition probability $p^d(1-p)^{n-d}$. This is the probability that a single application of the mutation operator transforms v_1 into v_2 (or vice-versa).

Third, a single landscape is often used as the framework for considering multiple operators. The best example is the use of a hypercube graph landscape as a basis for consideration of both mutation and crossover in a GA. Apart from the fact that the hypercube is not induced by either operator, this position leads to statements such as "crossover is making large jumps on the landscape." Crossover is viewed here in terms of the landscape structure that mutation is popularly thought to create. Mutation takes single steps, but crossover causes large jumps. Why is this? Even when the mutation probability in a GA is set to zero, people will insist on talking about local maxima from the point of view of mutation, despite the fact that these clearly have no relevance whatsoever to the algorithm. The new model claims that each operator is taking single steps on its own landscape graph. It is possible to consider the effect of crossover in terms of the mutation graph, but it is equally possible to consider mutation in terms of the crossover graph. The model insists that we explicitly acknowledge this, rather than, as is now common, comparing one operator in terms of the structure defined by another without acknowledging it.

Fourth, there is a tremendous bias towards thinking of GAs in terms of a mutational landscape. This is difficult to detect. A good illustration comes again from the consideration of the word "peak." As discussed above, for something to be a peak, it must have neighbors. If we define neighborhood in terms of the operators we employ, it should seem strange that we can use the word peak for one operator but not for another. Why is there no notion of a peak under crossover? Why is the term reserved solely for mutation? A GA creates new individuals through crossover and through mutation. In some instances, the number of individuals created via crossover will be greater than the number created through mutation. In these cases, shouldn't we be more concerned about peaks (local optima) with respect to crossover than with respect to mutation? In fact, if the domain of a fitness function has a point whose fitness is higher than that of all others, the mutation landscape will always have a single global optimum and no local optima, as any point can be reached from any other via one mutation. By this reasoning, we should not be concerned with local optima from mutation's point of view, since they do not exist! The current notions of landscape pay no heed to operator transition probabilities, and this is clearly important. The proposed model of landscapes incorporates these probabilities and removes the mutational bias, giving each operator its own landscape, complete with peaks.

Fifth, the current models of landscape cannot be simply extended to other search algorithms. In fact, it is not even possible to use them consistently within the field of EAs. Current models typically require fixed dimensionality and a distance metric. This makes it difficult to describe, for instance, the landscape that Genetic Programming (GP) is operating on. The proposed model makes it simple to view GP as occurring on landscapes, as well as making connections to many search algorithms in other fields, such as those in Artificial Intelligence [10, 11].

Formal definitions of landscapes can be found in the work of Weinberger [17, 18] and in a large number of papers by Schuster, Stadler, Fontana and others in theoretical chemistry (see [2] for example). Both of these models were formulated while considering specific systems; the dynamics of spin glasses and the folding and evolution of RNA molecules. It is not surprising then that they are not directly applicable to GAs and other algorithms. There are three primary difficulties that need to be addressed:

- These models are relevant to systems in which an operator acts on a single individual and produces another individual. That is, one RNA molecule is converted to another RNA molecule or one Hamiltonian circuit in a graph is converted into another.

- The systems under consideration all involve a single operator. For example, an operator which changes a spin in a spin glass or a point mutation operator which changes a nucleotide in an RNA molecule.
- All possible outcomes from an application of an operator have equal probability of occurrence.

These differences make the application of these landscape models to GAs problematic. In GAs and other evolutionary algorithms, none of the above are true. A GA has operators that act on and produce multiple individuals, it employs multiple operators, and the possible results of operator application are not equiprobable. These differences can all be incorporated into a more general landscape model. This model, which is outlined in the next section, also resolves the five problems described at the start of this section.

3 Outline Of A Model

This section presents a very brief description of a model of landscapes. A full description can be found in [9]. In this model, a landscape, $\mathcal{L}$, can be viewed as a directed graph $G_{\mathcal{L}} = (V, E)$. An operator, ϕ, can be thought of as a stochastic event that occurs in some context $v \in V$ and whose outcome is a random variable W with some probability distribution function. The probability of the event $W = w$ for a specific $w \in V$ and context v, will be denoted by $\phi(v, w)$. The ϕ-neighborhood of v, $N_{\phi}(v)$, is $\{w \in V : \phi(v, w) > 0\}$. If $\phi(v, w) > 0$, then $G_{\mathcal{L}}$ will contain a directed edge from v to w labeled $\phi(v, w)$. Vertices in $G_{\mathcal{L}}$ therefore correspond to the possible inputs and outputs of operators and there is no need to restrict these to correspond to individuals. A vertex will correspond to a multiset of individuals. This allows the construction of landscapes for operators that act on and/or produce more than a single individual.

Under this model, when an algorithm employs several operators, there are several landscapes. If we consider mutation, crossover and selection to be operators, a GA is making transitions on three landscapes; a mutation landscape, a crossover landscape and a selection landscape. Figure 1 illustrates this process. It is easy to see how this model removes the problems

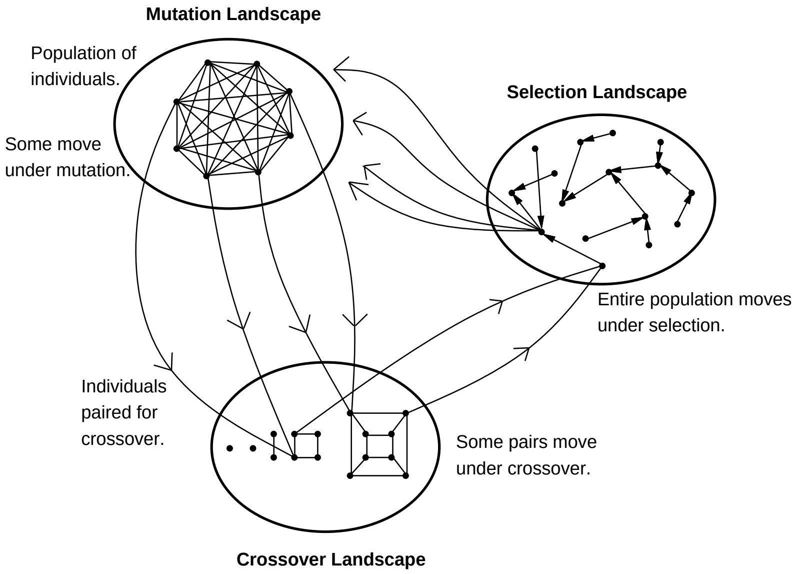

Figure 1: A simplified view of a GA operating on three landscapes. The landscape graphs are idealizations of far larger structures, and self-loops in the graph, created when an operator does not affect a vertex, have been omitted. The GA is seen as taking steps on the mutation landscape, then pairing individuals (probably according to fitness) thereby forming vertices on the crossover landscape upon which moves are made before the entire population is gathered into a vertex on the selection landscape where a step is taken. Finally, the population is decomposed into individuals which again correspond to vertices on the mutation landscape.

associated with the two formal models discussed above. Operators may act on and produce any number of individuals. We are not restricting attention to algorithms that employ a single operator—we simply adopt the view that if there are multiple operators at work, then there are multiple landscapes. Labeling edges with probabilities allows for consideration of operators whose outcomes are not all equiprobable.

In addition, if one adopts this model, the difficulties posed by the current notions of landscapes are removed. The model insists that we focus on operators as that which defines neighborhood. Until an operator has been specified, no landscape can exist. We will not be tempted to discuss crossover in the context of the mutation landscape, since crossover has its own landscape and the two have nothing to do with each other.[1] The bias towards thinking in terms of mutation can potentially be removed, as crossover and selection produce their own landscapes in the new model.

Finally, a landscape can be defined without any need for dimensionality or distance metrics. Though the model has no such requirements, it does not preclude the use of dimensionality and distance metrics if they can be defined. These are absent, or not trivially defined when we manipulate lisp S-expressions in genetic programming or permutations of integers when solving ordering problems. Should we therefore conclude that a GA operating on these representations cannot be described in landscape terms? This is not necessary. The new model allows us to seamlessly define landscapes for these representations in a natural way, allowing a common view of genetic programming and other algorithms that do not explicitly operate on bit strings. For this reason, the model encompasses far more than simply the landscapes that correspond to a collection of fixed-length binary strings. For example, it can be applied without change as a framework for thinking about heuristic state space search algorithms in AI.

4 Search as Structure and Strategy

The model advocates a "one operator, one landscape" view that allows the identification of a number of structures upon which a search algorithm operates. This collection of landscapes, determined by choices of representation, fitness function and operators, are only part of the process of search. What remains is the process of navigating within these structures in an attempt to locate the object of the search. I have termed this the "navigation strategy" of the search algorithm [9]. In a GA, the navigation strategy is responsible for determining population size, how the initial population is to be created, when the algorithm should be halted, how often to employ crossover and mutation, how to pair individuals for crossover, how to select part of the population to preserve if the generation gap is less than one and so on. Each of these decisions affects how the search on the various landscapes will proceed.

[1] This is not to say that we can ignore the result of employing both in an algorithm.

I believe that this division of search into structure and navigation is an important step towards the integration of GAs into the community of search algorithms developed in fields such as AI and OR, where this distinction has long been recognized. The reason is that in both cases, the division produces a picture of search algorithms as algorithms that search graphs. Many well-known search algorithms, such as A* [6], are explicitly designed to search graphs. A view of evolutionary algorithms as searchers within graphs (landscapes) has much in common with views of search algorithms in these other fields. For example, Pearl [14] (page 15) describes problem-solving, "the task of finding or constructing an object with given characteristics," as having three rudimentary requirements that AI has given the names "database," "operators or production rules," and "control strategy." The first two of these components form a "state space" and the control strategy is used to explore it. Our division into landscapes and navigation strategy is identical. Naturally there are differences in applications, for example, AI control strategies are sometimes designed to search graphs that are trees and EAs usually search several graphs simultaneously. The division into structure and strategy has at least two important consequences for EAs:

- Results from AI and OR may inform the study of EAs. These include optimality results that have been proven for search algorithms such as A*, and work done on the theory of heuristic functions [14]. The choice of heuristic functions in AI/OR and the choice of fitness functions in EAs have much in common. Both are used to label the vertices of graphs as a basis for guiding search. Within AI/OR, originating with the work of Doran and Michie [1], much attention has been paid to discovery, comparison, admissibility and automatic generation of heuristic (or evaluation) functions and their use to label the vertices of graphs to facilitate search. On a smaller and less formal scale, the effect of choice of fitness function and method of fitness scaling has been considered in EAs. If one regards EA fitness functions as heuristic functions, and considers them in this light (as an estimate of the distance to a goal), an extremely promising measure of search difficulty is obtained [10]. This measure correlates well with the performance of a GA. In particular, it provides an indicator of GA hardness that does not appear to suffer from the problems encountered with other indicators of difficulty. For example, this measure does not misclassify the problems designed by Grefenstette [5] to illustrate that deception is neither necessary nor sufficient for a problem to be hard for a GA. It also accounts for the surprising results found in Royal Road functions [3] and in the Tanese functions [4]. It correctly reports that Horn and Goldberg's massively rugged problem is easy to solve [7] and produces accurate measures of a number of other problems. It also correctly predicted that the question of whether binary or Gray coding was more useful for a problem was dependent on the number of bits in the encoding. The development of this measure was a direct consequence of considering the connection between EAs and other search methods that is made plain by the model of landscapes of this paper.

- The second consequence of the division into structure and navigation is that when two search algorithms are so divided, they can be recombined to form new algorithms. The new algorithm may exhibit better performance than either of the originals, and may also throw light on the original algorithms. For example, a GA can be divided into three landscapes and a navigation strategy. Hillclimbing algorithms can be divided into a single landscape and a navigation strategy. It is possible to take a landscape from the GA and use a hillclimbing navigational strategy to search it. This has been done in the *crossover hillclimbing* algorithm in [8]. The result outperforms both the GA and the hillclimber on several problems that have been examined. Further examination of this algorithm showed the importance of the dual roles played by crossover in a GA. This led to a new test for determining whether the use of crossover is producing gains for the GA over those that could be obtained with simple macromutation. New combinations of structure and strategy such as this may also involve search algorithms from outside EAs. For example, a search algorithm such as A^* could be used to make local improvements in the mutation or crossover graph during the run of a GA. Similar approaches have been taken in hybrid "memetic" algorithms, which have used hillclimbing via mutation to improve individuals [13]. The model of this paper provides a formal basis for this sort of algorithm, making explicit the fact that general graph searching algorithms can be used and that they may run on any of the GA's landscapes.

5 Usefulness Of The Metaphor

The term "landscape" has something powerfully seductive about it. The imagery it evokes is so appealing that further thought can be completely suspended. We use the imagery of the landscape metaphor in an attempt to increase our understanding of algorithms, to raise and possibly answer questions about them, and to suggest new approaches. Much of this imagery tends to rely rather heavily on the simple properties that we see in physical three dimensional landscapes. It is not clear just how many of the ideas scale up to landscapes with hundreds or thousands of dimensions. It is quite possible that the simplicity and beauty of the metaphor is actually damaging in some instances, for example by diverting attention from the actual process or by suggesting appealing, simple and incorrect explanations. All of this has been put very well by Provine [15] (pp. 307–317), which should be required reading for people interested in employing the metaphor.

The ambiguities surrounding the term and its use originated with Wright, and were not identified until 1985 [15]. These problems can also be found in the field of evolutionary computing. Given this, it is worth asking whether it is better to abandon the term or to use it and try to be more precise about what is actually meant. There is something to be said for abandoning it—after all, in this and other models, a landscape is simply a graph. On the other hand, it seems unlikely that the term will just go away. In addition, the metaphor, however distant it may sometimes be from reality, *has* given rise to new ideas and intuitions. This paper has opted to adopt the term, with the hope that it will lessen, rather than increase, the vagueness with which it is applied.

6 Conclusion

This paper has attempted to motivate the use of a new model of fitness landscapes. The model has much to recommend it. A consequence of the model is that each operator employed by a search algorithm generates its own landscape graph. The model allows a consistent definition of landscapes for operators that act on or produce multiple individuals, and landscapes for problems where individuals are not fixed length strings. For example, problems whose individuals are permutations of integers or lisp S-expressions are seamlessly accommodated. This in turn allows for the definition of terms such as "peak" that can be applied to any operator, rather than just mutation. The model incorporates edge transition probabilities as an important part of landscapes. A view of search as comprised of structures (landscapes) and navigation upon them is strongly encouraged by the model, and it was pointed out that this is the view of search that has long existed in Artificial Intelligence.

By defining a landscape as a labeled graph, insisting on "one operator, one landscape," and viewing search as structure and strategy, an important connection with heuristic state space search is established. This allows a view of GAs and other evolutionary algorithms as being close relatives of search algorithms in Artificial Intelligence and Operations Research. Far from being simply an interesting observation, the connection has produced good results. First, by considering GA fitness functions as heuristic functions, a measure of problem difficulty has been developed that correlates extremely well with GA performance. Second, the division into structure and strategy invites the construction of new algorithms and this has lead to a crossover hillclimbing algorithm whose performance is superior to the GA on a number of problems and whose simplicity has led to a simple method for determining whether crossover is helpful for a given problem. It is clear that in addition to providing a consistent and widely applicable definition of landscapes that removes the difficulties of other models, the model of this paper provides practical insights into some of the problems that are faced by those who consider the theory of evolutionary algorithms.

7 Acknowledgments

Thanks to Joseph Culberson, Walter Fontana, Stephanie Forrest, Ron Hightower, John Holland, Stuart Kauffman, Melanie Mitchell, Jesús Mosterín, Una-May O'Reilly, Richard Palmer, Gregory Rawlins, Mike Simmons, Derek Smith, and Peter Stadler for the many conversations over the last three years that have been important for the development of the ideas in this paper. Many thanks to the Santa Fe Institute for making communication of this kind a way of life. This research was supported in part by grants to the Santa Fe Institute, including core funding from the John D. and Catherine T. MacArthur Foundation; the National Science Foundation (PHY-9021427);

and the U.S. Department of Energy (DE-FG05-88ER25054).

References

[1] J. Doran and D. Michie. Experiments with the graph traverser program. *Proceedings of the Royal Society of London (A)*, 294:235–259, 1966.

[2] W. Fontana, P. F. Stadler, E. G. Bornberg-Bauer, T. Griesmacher, I. Hofacker, M. Tacker, P. Tarazona, E. D. Weinberger, and P. Schuster. RNA folding and combinatory landscapes. *Physical Review E*, 47(3):2083–2099, 1993.

[3] S. Forrest and M. Mitchell. Towards a stronger building-blocks hypothesis: Effects of relative building-block fitness on GA performance. In *FOGA-92, Foundations of Genetic Algorithms*, pages 109–126, Vail, Colorado, 26–29 July 1992.

[4] S. Forrest and M. Mitchell. What makes a problem hard for a genetic algorithm? Some anomalous results and their explanation. *Machine Learning*, 13:285–319, 1993.

[5] J. J. Grefenstette. Deception considered harmful. In L. D. Whitley, editor, *Foundations of Genetic Algorithms*, volume 2, pages 75–91, San Mateo, CA, 1992. Morgan Kaufmann.

[6] P. E. Hart, N. J. Nilsson, and B. Raphael. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, 1968.

[7] J. Horn and D. E. Goldberg. Genetic algorithm difficulty and the modality of fitness landscapes. In L. D. Whitley, editor, *Foundations of Genetic Algorithms*, volume 3, San Mateo, CA, 1994. Morgan Kaufmann. (To appear).

[8] T. C. Jones. Crossover, macromutation, and population-based search. In L. J. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, 1995. (submitted).

[9] T. C. Jones. *Evolutionary Algorithms, Fitness Landscapes and Search.* PhD thesis, University of New Mexico, Albuquerque, NM, March 1995. (expected).

[10] T. C. Jones and S. Forrest. Fitness distance correlation as a measure of problem difficulty for genetic algorithms. In L. J. Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, 1995. (submitted).

[11] T. C. Jones and S. Forrest. Genetic algorithms and heuristic search. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1995. (submitted).

[12] B. Manderick, M. De Weger, and P. Spiessens. The genetic algorithm and the structure of the fitness landscape. In R. K. Belew and L. B. Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 143–150, San Mateo, CA, 1991. Morgan Kaufmann.

[13] P. Moscato. On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms. Technical Report 826, Caltech Concurrent Computation Program, Pasadena CA, September 1989.

[14] J. Pearl. *Heuristics: Intelligent Search Strategies for Computer Problem Solving.* Addison-Wesley, Reading, MA, 1984.

[15] W. B. Provine. *Sewall Wright and Evolutionary Biology.* University of Chicago Press, Chicago IL., 1986.

[16] E. D. Weinberger. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biological Cybernetics*, 63:325–336, 1990.

[17] E. D. Weinberger. Fourier and taylor series on fitness landscapes. *Biological Cybernetics*, 65:321–330, 1990.

[18] E. D. Weinberger. Measuring correlations in energy landscapes and why it matters. In H. Atmanspacher and H. Scheingraber, editors, *Information Dynamics*, pages 185–193. Plenum Press, New York, 1991.

[19] S. Wright. The roles of mutation, inbreeding, crossbreeding and selection in evolution. In *Proceedings of the sixth international congress of genetics*, volume 1, pages 356–366, 1932.