Recombination Induced Hypergraphs: A New Approach to Mutation-Recombination Isomorphism

PAUL GITCHOFF
Department of Biology, Center of Computational Ecology, Yale University, New Haven, CT 06520, USA

GÜNTER P. WAGNER
Department of Biology, Center of Computational Ecology, Yale University, New Haven, CT 06520, USA,
e-mail: gpwag@peaplant.biology.yale.edu

Received October 18, 1995; accepted February 14, 1996

Natural selection acts on genetic variation that comes from two principal sources: mutation and recombination. Because of the inherent differences between mutation and recombination, it is often assumed that they are qualitatively different ways to explore the genotype space. In this paper a new way of constructing recombination spaces is introduced and the topological features of the resulting hypergraphs are analyzed. It is shown that types which are neighbors in the point mutation space are also neighbors in the recombination space, i.e., mutation and recombination spaces are homomorphic. This implies that the shapes of the fitness functions explored by mutation and recombination are similar. However, the potential of one- and two-point recombination operators to explore the fitness landscape may differ dramatically from uniform recombination operators or mutation operators because of the limited number of recombinant types they can produce. © 1996 John Wiley & Sons, Inc.

Keywords: Recombination, genetic algorithms, hypergraphs, configuration spaces

1. INTRODUCTION

The concept of fitness landscapes is amongst the most influential ideas of modern evolutionary biology [21]. The notion was introduced by Sewall Wright [26] in 1931 and one version if it is based on the assumption that each genotype can be associated with a fitness value, which collectively represent the fitness landscape. Extensions of this concept have been used to explain phenotypic adaptations [24], and were formalized by Lande [18] and Barton and Turelli [2]. The advantage of the landscape metaphor is that the topological features of this hypothetical landscape are related to the ability of natural selection to reach certain genotypic or phenotypic states. The same mathematical structure appears also outside biology, for instance in the Hamiltonians of spin glasses [19], or the cost or performance functions in search and optimization problems [11], and in all forms of evolutionary algorithms [13, 22]. The central importance of landscape models motivates efforts to develop a general theory of landscapes [25].

Underlying the idea of a landscape is the notion that the types, be it genotypes, phenotypes or other abstract objects (e.g., paths in the traveling salesman problem), are arranged in some kind of topological order, called the configuration space. Recently it was noted that the structure of configuration spaces is defined by the way new types come into existence [16]. For instance, two genotypes are neighbors in a
configuration spaces caused by point mutations on strings are easy to construct. Given one type and a one-point mutation operator (e.g., bit flipping), one can look at all the types that are reachable by one application of the mutation operator. The set of all mutant genotypes provides a natural notion of topological neighborhood. This approach leads to a graph which is isomorphic to a hypercube. The construction of recombination spaces is more difficult since recombinations involve two parental types. One approach is to construct recombination spaces where the elementary objects (vertices) are unordered pairs of types. For instance, Culberson [8] constructs the one-point crossover space by taking pairs of complementary strings as elementary objects. This makes the comparison of recombination spaces with the mutation space difficult. However, it is possible to construct a homomorphism of mutation and recombination spaces [8]. In a work similarly motivated, Jones allows all pairs of strings to be elements in the recombination space [15]. In this article, we introduce a new way to construct recombination spaces in which the elements or vertices are single types. The resulting graph is a hypergraph (for background on hypergraphs see [4]).

2. INTUITIVE BACKGROUND

Evolution by natural selection can be subdivided in two classes of processes: selection and transmission [1]. This distinction is not congruent with the distinction between replicators and interactors that emerged in the units of selection debate [9, 14], but is implicitly used in much of population genetics theory [12]. Selection is considered as an operator acting on the frequency distribution of genotypes \( p_{\text{selection}} \), but note that \( p\mu \) is the distribution of the parental genotypes after selection. The genotype distribution in the next generation is caused by another operator, called transmission \( p_{\text{transmission}} \). The transmission function describes the consequences of all the genetic processes producing the genotypes of the next generation, for instance mutation, recombination etc. The same abstraction is used to predict the selection response in quantitative genetics [4, 10, 18]. First one considers the change of the character mean value due to selection \( \mathbb{Z} \xrightarrow{\text{selection}} \mathbb{Z}_w \). The difference is called the selection differential \( S = \mathbb{Z}_w - \mathbb{Z} \). The mean value of the next generation is then predicted on the basis of the heritability \( \mathbb{Z}' = \mathbb{Z} + h^2 S \). The difference between the character mean in the previous generation before selection and the offspring generation is called the selection response \( R = \mathbb{Z}' - \mathbb{Z} = h^2 S \).

Since there are two types of operators, selection- and transmission-operators, it is not entirely clear which one shall be used to construct the configuration space [15, 16]. Here we decided to use solely the transmission operator to construct configuration spaces for two reasons. First, the processes underlying the transmission operator, namely mutation and recombination, are the ones which generate new types in the course of adaptive search. They are thus the processes determining which types are reachable from a given type set and thus define “neighborhood” in a way relevant for the efficiency of the search procedure. Second, the sets on which the selection and transmission operators are acting are of very different kind. The selection operators acts on distributions over types while the transmission operators can be defined in a way in which they act on individual types directly.

3. FORMALIZATION

The method we introduce to construct recombination spaces, is based on the topological notion of an “interval” rather than neighborhood. (Note that the two dimensional analog of an interval is a circle, etc.) The idea is, given two types (for instance two 0-1 strings) and apply a recombination procedure, like one-point crossover, the resulting set of possible recombinant sequences can be considered as topological “in-between” the two parental sequences. This is intuitively justified by the fact that recombinant types are “mixtures” of the parental types. The parental types thus mark the two endpoints of a (generalized) interval in recombination space. The interval is filled with all the recombinant types of the two parental ones. Below it is shown, that the sets of recombinants, called recombination sets, can be considered as the generalized edges of a hypergraph.

In this section we introduce the basic notation which will be used to construct recombination spaces. Most notably we need the notion of a recombination operator.

3.1. THE GENERAL RECOMBINATION OPERATOR

Let \( T \) be any set. We take \( T \) to represent the set of “types,” which may be strings of bits, vectors, DNA or RNA sequences etc. We define a recombination operator on \( T \) to be a function \( \mathcal{R}: T^2 \rightarrow \wp(T) \), that is the recombination operator maps every pair of elements of \( T \) to a subset of \( T \) (i.e., an element of the power set \( \wp(T) \)) such that \( \forall s,t \in T \), the following conditions hold:

\[
\begin{align*}
\mathcal{R}(s,t) &= \mathcal{R}(t,s) \\
\mathcal{R}(s,s) &= \{s\} \\
\{s,t\} &\subseteq \mathcal{R}(s,t) \\
\mathcal{R}(s,u) &\subseteq \mathcal{R}(s,t) \quad \text{for } \forall u \in \mathcal{R}(s,t).
\end{align*}
\]
We will call $\mathcal{R}(s,t)$ the recombination set of $s$ and $t$. This definition captures some intuitive notions of the properties of genetic recombination. The first condition is a simple symmetry requirement. The second condition says that no new types are created from a single type by recombination. The third condition is only for notational convenience and the fourth condition defines the topological implications of recombination. It essentially says that recombinants are more “similar” to the parental types than the two parental types. This is the idea that recombinants are mixtures of the two parental types. Furthermore this condition says, that the more “similar” two types are the fewer types can be created by recombination. As shown below, all these conditions are met by the conventional one point, two point and uniform recombination. It essentially says that recombinants are more similar to the parental types than the two parental types.

This definition of a recombination operator applies to arbitrary type sets, not just strings and vectors. It is thus not associated with a particular mechanism of recombination such as crossover. However, most of the results below will be derived for Boolean or other strings where the recombination process is implied by the syntactical structure of the objects. This is a more transparent approach and we will therefore mostly talk about strings and vectors.

### 3.2. Recombination of Boolean Vectors

Let $T = C(n)$ be the set of all Boolean vectors of size $n$, then $a = (a_1, ..., a_n) \in C(n)$ iff $a_i = 1$ or 0 $\forall i$. Of course, $|C(n)| = 2^n$. We define $\{x_i\}$, a set of one-point crossover functions, for $1 \leq i \leq n$.

$$x_i : C(n) \rightarrow C(n), (a_1, ..., a_n) x_i (b_1, ..., b_n) = (a_1, ..., a_i, b_{i+1}, ..., b_n).$$

We specify $x_a$ as representing the null crossover: $a \times_a b = a$. Note that this inclusion of null crossover is a slight deviation from usage in genetic algorithms theory, where only “effective” recombination events are included. For reasons of notational symmetry it is useful to include it. Further note that this function is idempotent, $a \times_i a = a$, and associative ($a \times_i b) \times_i c = a \times_i (b \times_i c)$.

We may use this function to define the one-point recombination operator:

$$\mathcal{R}_i(a, b) = \bigcup_{1 \leq i \leq n} \{a \times_i b, b \times_i a\}.$$  

Analogously, we may define a two-point crossover function $\{x_{ij}\}, 1 \leq i < j \leq n$.

$$\{a_1, ..., a_n\} \times_{ij} (b_1, ..., b_n) = (a_1, ..., a_i, a_{i+1}, ..., a_j, a_{j+1}, ..., a_n).$$

We identify each function $x_{im}$ with $x_i$, and we allow in addition the null crossover $x_{nn} = x_n$. Again the definition of the recombination operator follows:

$$\mathcal{R}_i(a, b) = \bigcup_{1 \leq i \leq n} \{a \times_i b, b \times_i a\}.$$  

Finally, we define the recombination operator for uniform recombination as:

$$\mathcal{R}_u(a, b) = \{c_1, ..., c_n\}; c_i = a_i \ or \ b_i \forall i.$$  

All these recombination operators satisfy the definition of recombination operators. Furthermore for all $a$ and $b$ in $C(n)$, we have

$$\mathcal{R}_i(a, b) \subseteq \mathcal{R}_j(a, b) \subseteq \mathcal{R}_u(a, b).$$

For $a$, $b$ in $C(n)$, let $H = d(a, b)$ denote the Hamming distance. Then we have:

$$\|\mathcal{R}_i(a, b)\| = \begin{cases} 2H \text{ when } a \neq b \\ 1 \text{ when } a = b \end{cases}$$  

$$\|\mathcal{R}_j(a, b)\| = \begin{cases} H(1-1)+2 \text{ when } a \neq b \\ 1 \text{ when } a = b \end{cases}$$  

$$\|\mathcal{R}_u(a, b)\| = 2^H.$$  

### 3.3. Uniqueness of One-Point Recombination Sets

Here we will prove a proposition, which is important for the algebraic theory of the fitness landscapes over the one-point recombination space (Stadler and Wagner, in prep.). It says that the recombination set is unique if the parental types are different at more than two sites.

Proposition: if $d(a, b) > 2$ and $d(s, t) > 2$ then

$$\mathcal{R}_i(a, b) = \mathcal{R}_i(s, t) \leftrightarrow \{a, b\} = \{s, t\}.$$  

Of course the implication $\rightarrow$ is trivial. For proving the other implication $\leftarrow$, we need the following Lemma. In the proof of this lemma we ignore all sites which are identical between the two vectors $a$ and $b$.

Lemma: if $s = a \times_i b$ and $t = b \times_i a$ then for all $k \neq i$

$$s \times_k t \notin \mathcal{R}_i(a, b)$$  

$$(t \times_k s) \notin \mathcal{R}_i(a, b)$$

For the proof of this lemma it is useful to introduce the following notation:

$$a \parallel b = (a_1, ..., a_i, b_{i+1}, ..., b_n)$$

such that $a \parallel b = a \times_i b$, but $a \parallel b \notin T$ and $a \times_i b$ is a function. Hence, $s = a \parallel b$ and $t = b \parallel a$. Now let us consider $k < i$,

$$s \times_k t = (a \parallel b) \times_k (b \parallel a) = a \parallel b \parallel a \in \mathcal{R}_i(a, b)$$
and for k=i also $s \times_i t \in \mathcal{R}_i(a,b)$. For $t \times_k s$ an analogous result holds. This proves the Lemma.

The proposition about the uniqueness of one-point recombination sets will be proven by considering the equivalent statement $\neg \left( \mathcal{R}_i(a,b) = \mathcal{R}_i(s,t) \wedge \{a,b\} \neq \{s,t\} \right)$. Let us assume

$$\mathcal{R}_i(a,b) = \mathcal{R}_i(s,t) \wedge \{a,b\} \neq \{s,t\},$$

then $\{s,t\} \subset \mathcal{R}_i(a,b)$. Note that $\{s,t\}$ is a proper subset of the recombination set of $a$ and $b$, since $d(a,b) = 4$ implies that $d(\mathcal{R}_i(a,b)) = 2d(a,b) > 4$. Now if $\{s,t\} \subset \mathcal{R}_i(a,b)$ and $d(s,t) = d(a,b) = H$, then there exists an $i < H - 1$ such that $s = (a \times_i b)$ and $t = (b \times_i a)$. This statement actually uses a result from the next section, summarized in Figure 1. Because of the Lemma proved above, the assumption can not be true and $\mathcal{R}_i(a,b) = \mathcal{R}_i(s,t) \rightarrow \{a,b\} = \{s,t\}$ follows immediately.

We conjecture that the recombination sets are unique for any $k$-point recombination operator if the “parental” strings have a distance larger than $k + 1$.

4. RECOMBINATION SPACES

Given a type set $T$ and a recombination operator $\mathcal{R}$, we seek to understand the topological features induced by $\mathcal{R}$ on the unordered set $T$. The information about the topology induced on $T$ is contained in the set of all recombination sets. We thus call the object $R(T) = (T, \mathcal{R})$ a recombination space if $3 = \{\mathcal{R}(s,t) : s,t \in T\}$. $R(T)$ can be seen as a hypergraph with the vertex set $T$ and the recombination sets are the generalized edges.

There are two limiting cases where the resulting hypergraph does not provide any notion of distance, proximity, or neighborhood. In the minimal limit, we have $\mathcal{R}(s,t) = \{s,t\}$ for all $s,t \in T$. The resulting hypergraph has an isomorphic copy of the 2-complete hypergraph. In the maximal limit, we have $\mathcal{R}(s,t) = T \forall s \neq t$. The resulting hypergraph has only the universal edge. Both cases are compatible with our notion of recombination operators, but induce no topological structure on $T$.

4.1. Boolean Recombination Spaces are Homomorphic to Point-Mutation Spaces

Let us now consider $R(C(n)) = (C(n), \mathcal{R})$, with $\mathcal{R} = \{\mathcal{R}(a,b) : a, b \in C(n)\}$, and $i \in \{1, 2, \Omega\}$. We may note in passing that neither of the Boolean recombination spaces are topological spaces, even after adding the empty set to $\mathcal{R}$. Note that a pair $(M, \Xi)$ with $\Xi \subseteq \sigma(M)$ is a topological space if three conditions are met:

1) $\emptyset \in \Xi, M \in \Xi$,
2) $A \in M, B \in M \Rightarrow A \cap B \in M$, and
3) $N \subseteq \Xi \Rightarrow \bigcup_{A \in \Xi} A \in \Xi.

Neither $\mathcal{R}_1$ nor $\mathcal{R}_2$ contain $T$ and are thus disqualified as topological spaces. $T \in \mathcal{R}_3$ because $\mathcal{R}(a,b) = T$ if $d(a,b) = n$, but the third condition is not met by any of the recombination spaces. For example take $A$ and $B$ to be nonoverlapping recombination sets, then $A \cup B$ can not be a recombination set, i.e., there are no $s,t \in C(n)$ for which $A \cup B$ this is the recombination set.

Now we want to construct the recombination space for Boolean vectors, which will lead to the following result:

Proposition: the hypercube is embedded in all three recombination spaces $R_1(C(n)), R_2(C(n))$ and $R_3(C(n))$. In other words, the point mutation space for bit strings is homomorphic with the corresponding recombination spaces.

Let $N_n(x)$ be the number of unordered pairs of elements in $C(n)$ whose Hamming distance is $x$. Then $N_n(x)$ takes on values as follows:

$N_0(x) = \binom{n}{x}$, $N_1(x) = \binom{n}{x} - \binom{n}{x-1}$, and $N_2(x) = \binom{n}{x} - \binom{n}{x-1} - \binom{n}{x-2}$.
\[ N_n(x) = \frac{1}{2^n} \binom{n}{x} \frac{n!}{x!(n-x)!}. \]

To begin our construction we identify each of the 2^n singleton sets \( \mathcal{R}(c,c) = \{c\} \) with a vertex of the hypergraph. Consider now the pairs of vertices having Hamming distance one. Then for \( i = 1,2,\Omega \), we have \( \mathcal{R}_i(a,b) = \{a,b\} \). Thus in each recombination hypergraph an edge of rank 2 exists between each pair of vertices separated by a Hamming distance of one. There are \( n2^{n-1} \) such pairs in \( C(n) \). These are the only rank-2 edges in each graph. This implies that in each case, the recombination space contains an isomorphic copy of the well-known n-dimensional hypercube of point mutation space.

Let us conditionally accept the hypercube as the support structure of each space \( R_i(C(n)) \). We must still investigate the remaining edges. Given two vertices \( a \) and \( b \) with Hamming distance 2, then in general \( \mathcal{R}_i(a,b) = \{a,b,c,d\} \). Note that \( d(a,c) = d(a,d) = d(b,c) = d(b,d) = 1 \). Therefore \( \mathcal{R}_i(a,b) \) is a two dimensional face (square) on the n-dimensional hypercube. In fact, the same face will be generated exactly twice, since \( \mathcal{R}_i(a,b) = \mathcal{R}_i(c,d) \). Thus there are \( n(n-1)2^{n-2} \) generalized edges of order 4, each corresponding to a two-dimensional face of the n-dimensional hypercube.

We may go no further in a simultaneous investigation of the three recombination spaces \( R_i(C(n)) \) and \( R_k(C(n)) \), since strings at Hamming distance greater than 2 generate different recombination sets under the three operators. It has been noted elsewhere that under free recombination every pair of elements generates a hypercube of dimension equal to the Hamming distance between them [8]. With this in mind, and recalling that \( \mathcal{R}_i(a,b) \subseteq \mathcal{R}_j(a,b) \subseteq \mathcal{R}_k(a,b) \) (see Section 3) we will investigate the minimal recombination space generated by one-point crossover.

We may note that a square is a circumference of a two-dimensional hypercube. We generalize this fact for all edges of the space \( R_i(C(n)) \) in the following Lemma:

Lemma: \( \mathcal{R}_i(a,b) \) is a circumference of the \( d(a,b) \)-dimensional hypercube \( \mathcal{R}_i(a,b) \), which is embedded in the n-dimensional hypercube.

With this result one may construct the \( N_n(n) \) recombination sets of order 2x, for x = 2,...,n. Figure 1 depicts a generalized recombination set \( \mathcal{R}_i(a,b) \), with \( d(a,b) = k \). Note that vertices connected in the recombination set and in fact also connected in the corresponding hypercube of the point mutation space. The hypercube thereby provides a natural support structure of the space \( R_i(C(n)) \). Incidentally the recombination spaces have the same metric as the point-mutation spaces.

At the end of this section we want to add a few notes on strings (or vectors) with more than two possible states on each position. It will be shown that most of what has been shown in the Boolean case is also true for the multi-allele case.

\[ T = \{ (a_1,...,a_n) : a_i \in S_i \forall i \}. \]

The definitions of the recombination operators remain the same as in the Boolean case, and the recombination spaces \( R_i(T); i \in \{1,2,\Omega \} \) are defined analogously.

Recombination operators are binary. A (local) consideration of \( \mathcal{R}_i(a,b) \) is essentially an examination of only up to two alleles at the n positions. With this in mind, it is natural to define a distance on \( R_i(T) \) equal to the Hamming distance. The expressions for the orders of recombination sets that are identical to the ones obtained in the Boolean case. Again the sense of neighborhood is the same as in the corresponding point mutation space.

4.2. Recombination Closure Sets

The notion of a recombination set has been introduced to apply to pairs of types. In this section we extend this notion to arbitrary subsets of the type set. The intuitive motivation for this development is that it might be interesting to know which part of the type set can be covered by recombining a given set of types to start with. Another question of interest is how different recombination operators differ in covering the search space.

Let \( T \) be any type set and \( \mathcal{R} : T^2 \to \mathcal{P}(T) \) be a recombination operator acting on a pair of elements of \( T \). Now we introduce the notion of a recombination set of an arbitrary subset of \( T \). Let \( S \subseteq T \), then the recombination set of \( S \) is

\[ \Phi(S) = \bigcup_{s \subseteq S} \mathcal{R}(s,t) \]

This function is an automorphism on the power set of \( T \) \( \Phi(\cdot) : \mathcal{P}(T) \to \mathcal{P}(T) \). In general, \( \Phi([s,t]) = \mathcal{R}(s,t) \forall s,t \in T \). Furthermore the following two relations hold: a) \( S \subseteq \Phi(S) \) and b) if \( R \subseteq S \) then \( \Phi(R) \subseteq \Phi(S) \). All this holds for the general recombination operators defined in section 3.

It is not true in general that \( \Phi(\mathcal{P}(S)) = \mathcal{P}(\Phi(S)) \). However, there is a natural extension which has this property; we form it via repeated application of the operator:

\[ \overline{S} = \Phi^k(S) \text{ such that } \Phi(\overline{S}) = \overline{S}. \]

This operator represents closure under the \( \Phi(\cdot) \) operator. It clearly satisfies the properties (a) and (b) of the \( \Phi(\cdot) \) operator by extension. In addition it is obvious that

\[ \overline{\overline{S}} = \overline{S} \text{ and } (\cdot) \]

is in fact a closure operator. We call it recombination closure operator.
Let $S$ be the set of types in an initial population (the so-called population support) then $\bar{S}$ represents the set of all possible descendants of $S$, formed by recombination only. A subset $S$ of $T$ which satisfies $\bar{S} = S$ may be called recombination-closed. Such a population cannot produce any new types without the benefit of some other operator (e.g., mutation). Clearly $T$ and $\{\}$ are both recombination closed. It may be worthwhile to consider recombination closure under the string recombination operators studied above.

Let $T$ be any multi-alle type space. For any $a, b \in T$, under $R_1$ or $R_2$ we have $\{a, b\} = R(a, b)$. Given this fact, it is easy to show that recombination-closure under $R_1, R_2$ and $R_\text{II}$ are equivalent.

Another result is that for any recombination closed set $S \subseteq T$ there exists at least one proper subset $K \subset S$ such that $S = \bar{K}$. The smallest such set has the cardinality $\min \{k | f \} = \max_i S_i$, where $T = S_1 \times \ldots \times S_n$. In the case of the Boolean type space $\Omega$, $K$ is simply any $\{a, b\}$ with $d(a, b) = n$, any two types which differ in the maximal number of sites.

5. DISCUSSION

We have outlined a general method to study the topological features induced by recombination processes. The aim was to provide a topological framework to aid the analysis of population genetic models as well as the performance of genetic algorithms. It is not within the scope of the present article to provide such an analysis, but we want to mention a few applications in which this topological framework can be useful.

The configuration space induced by recombination on Boolean vectors is a hypergraph. The vertex set of the hypergraph is the set of individual types. This and the work of Culberson [8] and Jones [15] indicate that recombination induced configuration spaces cannot be represented by simple graphs, like the hypercube. The choice is either to have a simple graph with more complex vertices, as done by Culberson and Jones, or to have simple vertices representing single types but more complex generalized edges, i.e., a hypergraph [4]. In this article, we chose hypergraphs to represent recombination spaces because then the vertex sets of mutation and recombination spaces are identical.

Based on this construction the homomorphisms between recombination and mutation spaces can be directly seen. This is a generalization of Culberson's [8] result of one-point crossovers. In addition, we obtain a homomorphism among the three recombination spaces, one-point, two-point and uniform crossover. Furthermore it has been shown here that the recombination space contains a copy of the hypercube or the Hamming graph. Finally all the types in a recombination set are also connected in the hypercube. This means that the Hamming-distance is a metric in the recombination space as it is in the point-recombination space. Point mutation and string recombination induce the same metric on the set of types. From this perspective the topologies of the fitness landscapes over these configuration spaces are comparable. This is of interest for models of natural selection as well as evolutionary algorithms using mutation and recombination, since it means that both processes effectively are climbing the same kind of landscape. However, this does not imply that each of these search operators (mutation and different recombination operators) may perform equally well.

Compare for instance the one-point cross over and the uniform crossover. If we start with two bit strings $a$ and $b$ with Hamming distance $H$, and let the recombination process explore the landscape. In both cases the same part of the type set will be covered, because the recombination closure sets are identical (see Section 4.2). The only difference is that the one-point crossover takes more time to cover the space of possibilities. However, since the recombination set for one-point crossover produces only $2H$ types in the first round of reproduction, but uniform crossover potentially produces all possible $2^n$ types in one step, there is a constraint possible for the one-point crossover. Imagine for instance that all the types in the set $R_\text{I}^{-1}(a, b) = R_\text{I}(a, b) - \{a, b\}$ have much lower fitness than the types $a$ and $b$, then a population starting with $a$ and $b$ will not be able to improve in fitness because all the immediate recombinants are selected against. In contrast, if there are any types fitter than $a$ or $b$, then uniform recombination will eventually find them. In the case of one-point crossover, the result will depend on the types initially present. For a recent study on the rate of statespace coverage by one-point crossover see [7].

A more careful comparison of different operators requires a measure for the difficulty of the landscape. A variety of measures have been proposed to measure the difficulty of an optimization problem (for an overview see [15]). A recently proposed approach is to decompose the fitness function into the eigensfunctions of the Laplacian operator of the configuration space [25]. In the case where the configuration space can be represented as a simple graph the theory of graph Laplacians can be used [5]. However the extension to recombination spaces requires additional developments. The problem is that hypergraph Laplacians [6, 23] are not directly applicable, because the hypergraph does not contain all the necessary information. Not all elements in a recombination set are functionally equivalent, as assumed in the generalized edges of a hypergraph. In particular the two "parental" types, which generate the recombination set, need to be known to define the possible "paths" connecting the types in a recombination space. To accommodate this information requires a slightly more complicated structure than hypergraphs (Stadler and Wagner, in prep.).

The main advantage of a topological approach proposed here is that the tools can be applied to more general cases than the recombination of strings. Examples are recombination of parse trees in genetic programming [17], or any other type set that requires special operators. It has been known for many
years that epistatic gene interactions leads to nonlinear genetic complementation maps, and the inheritance of phenotypic characters is not necessarily modeled by a linear string [20]. However, when ever the recombination process conforms with our relatively unconstrained definition of a recombination operator (see Section 3) the induced recombination space can be studied as a hypergraph.

ACKNOWLEDGMENT

The authors want to thank the following individuals for comments on an earlier version of this manuscript: Leo Buss, Harald Freund, Terry Jones, and Sean Rice. GPW is grateful for numerous discussions on the ideas motivating this study: Leo Buss, Jim Crutchfield, Terry Jones, Junhyong Kim, Manfred Laubichler, Sean Rice, and Peter Stadler. The financial support by NSF grant BIR-9400642 and the Yale Institute for Biospheric Studies is gratefully acknowledged. This is contribution # 33 of the Yale Center for Computational Ecology.

REFERENCES