



Analysing Equilibrium States for Population Diversity

Johannes Lengler¹ · Andre Opris² · Dirk Sudholt²

Received: 11 July 2023 / Accepted: 10 March 2024
© The Author(s) 2024

Abstract

Population diversity is crucial in evolutionary algorithms as it helps with global exploration and facilitates the use of crossover. Despite many runtime analyses showing advantages of population diversity, we have no clear picture of how diversity evolves over time. We study how the population diversity of $(\mu + 1)$ algorithms, measured by the sum of pairwise Hamming distances, evolves in a fitness-neutral environment. We give an exact formula for the drift of population diversity and show that it is driven towards an equilibrium state. Moreover, we bound the expected time for getting close to the equilibrium state. We find that these dynamics, including the location of the equilibrium, are unaffected by surprisingly many algorithmic choices. All unbiased mutation operators with the same expected number of bit flips have the same effect on the expected diversity. Many crossover operators have no effect at all, including all binary unbiased, respectful operators. We review crossover operators from the literature and identify crossovers that are neutral towards the evolution of diversity and crossovers that are not.

Keywords Evolutionary algorithms · Runtime analysis · Diversity · Population dynamics

1 Introduction and Motivation

Evolutionary algorithms are general-purpose algorithms for optimisation and design that maintain a *population* (multiset) of candidate solutions and create new solutions by

✉ Andre Opris
Andre.Opris@uni-passau.de

Johannes Lengler
johannes.lengler@inf.ethz.ch

Dirk Sudholt
Dirk.Sudholt@uni-passau.de

¹ Institute of Theoretical Computer Science, ETH Zürich, Zurich, Switzerland

² Faculty of Computer Science and Mathematics, University of Passau, Passau, Germany

applying operators such as crossover, mutation and selection. The use of a population helps with exploration, is important for escaping local basins of attraction, and is the basis for efficient use of crossover operators [1]. However, this only applies if the population contains dissimilar individuals, commonly referred to as *diversity*. A major challenge in applying evolutionary algorithms is the fact that the population may collapse to copies of the same search point. Maintaining population diversity is an important aspect of evolutionary algorithms [2–5]. There exist many mechanisms that explicitly encourage or force the population to become more diverse. Several studies confirmed the benefits of such explicit diversity-preserving mechanisms on various test functions [6–12]. For some operators, like lexica selection, it is even known that diversity decreases the runtime of this operator [13].

Many theoretical and practical results show that even low levels of population diversity can improve runtime. Even on the most simple benchmark ONEMAX, which counts the number of bits set to 1 in a bit string of length n , the standard $(2 + 1)$ GA (with mutation rate $1/n$) is by a constant factor faster than the fastest mutation-based evolutionary algorithm without crossover [14–16]. This is due to the beneficial effects of crossover, which can exploit even small amounts of diversity. For the more complex monotone function HOTTOPIC, the same effect reduces the exponential optimisation time of the $(\mu+1)$ EA to $O(n \log n)$ for the $(\mu+1)$ GA if μ is a large constant and the algorithms are started close to the optimum [17]. Finally, it was also shown to benefit memetic (hybrid) evolutionary algorithms on HURDLE functions [18].

Examples where crossover between more diverse individuals can help include REAL ROYAL ROAD functions [19] and JUMP $_k$. For JUMP $_k$, it is necessary to cross a fitness valley of size k . The $(\mu+1)$ GA can do this with crossover in time $O(4^k)$ if sufficiently diverse individuals exist, while mutation-based operators need $\Omega(n^k)$ trials [20]. However, the original approach by Jansen and Wegener, later improved by Kötzing, Sudholt and Theile, only showed that sufficiently diverse individuals appear for unrealistically small crossover probabilities [20, 21]. Dang et al. [22] showed that a more modest improvement of roughly a factor n is still possible when always performing crossover. This study showed that diversity emerges naturally on the set of all search points with $n - k$ ones, and that on this set crossover serves as a catalyst for boosting population diversity. However, the full benefits of crossover can still be obtained if the $(\mu+1)$ GA is equipped with diversity-preserving mechanisms [23].

So there is no shortage of results showing that diversity can be beneficial. Despite these results, our understanding of how population diversity evolves is still very limited. Even on ONEMAX, for a standard $(\mu+1)$ EA, we do not have a complete picture. While there are upper bounds whose leading constants decrease with μ up to $\mu = o(\sqrt{\log n})$ [16], lower bounds that are tight including leading constants are only known for $\mu = 2$ [24]. For JUMP $_k$, empirical results suggest that the improvement by crossover is much larger than the mentioned factor of n from the theoretical analysis [22]. In both scenarios, the main obstacle is understanding the population dynamics and the evolution of diversity.

When considering problems with large degrees of *neutrality*, that is, contiguous regions of the search space (with respect to the Hamming neighbourhood) of equal fitness, or plateaus in the fitness landscape, our understanding of population diversity is also not well developed. Many important problems feature neutrality, and functions

with plateaus have been analysed in the literature in the context of runtime analysis of evolutionary algorithms. This includes, for example, (1) the *hidden subset problem* [25–28], where the fitness only depends on a small fraction of all variables, and it is not known which variables are relevant and which ones only lead to neutral changes, (2) *majority functions* returning the majority bit value [29, 30], (3) the *moving Hamming ball* benchmark [31] from dynamic optimisation where a Hamming ball around a moving target must be tracked and the fitness areas within and outside of the Hamming ball are both flat, and (4) the PLATEAU_k function [32, 33], a variant of ONEMAX in which the best k fitness levels are turned into a neutral region, except for the optimum at $\bar{1}$. However, except for [32, 33] the above results either concern populations of size 1 or do not give detailed insights into the diversity of the population. The aforementioned work on JUMP [22] does give insights into the population diversity as part of the analysis, however these insights are limited to the specific set of search points with $n - k$ ones.

We aim to initiate the systematic theoretical analysis of population diversity in steady-state algorithms to gain insights into how diversity evolves, how quickly diversity evolves, and which factors play a role in the evolution of diversity. In contrast to previous work, we do not consider functions with specific fitness gradients but take an orthogonal approach. We study how the population diversity, defined as the sum of pairwise Hamming distances in the population, evolves in the absence of fitness-based guidance as found in a completely neutral environment, that is, a flat fitness function.

We consider general classes of $(\mu+1)$ EAs and $(\mu+1)$ GAs equipped with various mutation and crossover operators. As diversity measure S , we consider twice the sum of pairwise Hamming distances of population members. We show that, for all unbiased mutation operators (as will be defined later), the diversity in all $(\mu+1)$ EAs is pushed towards an equilibrium state S_0 that depends on the population size μ , the expected number χ of bits flipped during mutation, and the problem size n :

$$S_0 := \frac{(\mu - 1)\mu^2\chi n}{2(\mu - 1)\chi + n}.$$

At this equilibrium the expected Hamming distance between two random population members (with replacement) is roughly $(\mu - 1)\chi$ if $2(\mu - 1)\chi \ll n$, i.e. increasing linearly with the population size μ and the mutation strength χ , and roughly $n/2$ if $2(\mu - 1)\chi \gg n$, respectively. The term $n/2$ makes sense as this is the expected average Hamming distance in a uniform random population.

We show that, for reasonable parameters, the expected time to decrease the diversity below $(1 + \varepsilon)S_0$, with $\varepsilon > 0$ constant, when starting with any larger diversity is bounded by $O(\mu^2 \ln n)$. This bound grows very mildly with the problem size n . On the other hand, the expected time to increase diversity above $(1 - \varepsilon)S_0$, when starting with less diversity, can be $\Omega(n)$ even for $\mu = 2$ and can thus be larger by a factor n/μ for small values of μ . We prove a general bound of $O(n \ln n)$ for all mutation operators. However, the situation is much better for tail-bounded unbiased mutation operators, i.e. for an unbiased mutation operator mut for which the Hamming distance $H(\text{mut}(y), y)$ between an offspring $\text{mut}(y)$ and its parent y decays exponentially:

$\Pr(H(\text{mut}(y), y) \geq \ell) \leq q^{-\ell}$ for some base $q = 1 + \Omega(1)$ and all $\ell \geq 0$. For those, the expected time for increasing diversity is again $O(\mu^2 \ln n)$ if μ is moderately large ($\mu = \Omega(\sqrt{\ln n})$). For very small values of μ , we can still show a bound of $O(\ln^2(n))$ for tail-bounded mutation operators. Most mutation operators used in practice are tail-bounded, including standard bit mutation. However, there also exist heavy-tailed mutation operators which are not tail-bounded [34]. Table 1 gives a summary on all upper time bounds shown in this work.

We also show that, surprisingly, the dynamics are to a very large extent independent of the specifics of the algorithm¹:

- For fixed $\chi > 0$, every unbiased mutation operator which flips χ bits in expectation, leads to the same dynamics. For example, standard bit mutation with mutation rate $1/n$ has the same dynamics as the mutation operator in Randomised Local Search (RLS) that always flips one bit.
- Large classes of crossover operators, including uniform crossover and k -point crossover, have no effect on the dynamics.

For these reasons, we systematically classify which crossover operators have an effect on the dynamics of population diversity. In Sect. 5 we show that crossover operators are neutral with respect to diversity if and only if they satisfy a certain characteristic equation. Consequently, we call such operators *diversity-neutral*. In Sect. 6, we investigate this property further and show that it is implied if the crossover is respectful with a mask independent of the order of the parents, see Sect. 2 for formal definitions. Moreover, we will show that unbiased crossover operators are diversity-neutral if and only if they are respectful, i.e., if and only if the offspring are in the convex hull of the parents. Finally, in Sect. 6.2 we apply our classification, building on results from [35], to classify five crossover operators from the literature as diversity-neutral, and seven other operators as not diversity-neutral.

An extended abstract with parts of the results was presented at GECCO 2023 [36]. This improved and extended manuscript includes all omitted proofs, many further details and discussions, a refinement of the upper bound on the expected time to approach the equilibrium state from a population of low diversity (Theorem 12 (ii) and (14) in Theorem 14) and a new upper bound that further refines this result for tail-bounded mutation operators (Theorem 12 (iii) and (15) in Theorem 14).

1.1 Motivation for Studying Flat Landscapes

There are two major motivations for our study of a flat fitness landscape. One reason is that, informally, they could provide *upper bounds* on the population diversity that we obtain in many non-neutral environments. While we suspect that counterexamples exist, we also suspect that for many “natural” fitness functions, diversity in non-flat environments is generally *lower* than diversity in flat environments. After all, selection tends to favour individuals which are similar to each other, since it systematically promotes individuals which have a similar trait (namely, high fitness). In contrast, in a

¹ By “dynamics”, we mean the expected change per generation, the equilibrium value S_0 , and our upper bounds for the expected time to reduce or increase diversity towards S_0 .

Table 1 Overview of upper bounds on the expected time to approximate or skip over the equilibrium state S_0 (cf. Theorem 14) for $(\mu + 1)$ -evolutionary and genetic algorithms with $\mu \geq 2$ meeting the conditions of Corollary 9. $S(P_t)$ is the diversity of the population at time t . Given $\varepsilon \in (0, 1]$ we have $T_{\varepsilon, \downarrow} := \inf\{t \mid S(P_t) \leq (1 + \varepsilon)S_0\}$ (expected time to reduce excess diversity) and $T_{\varepsilon, \uparrow} := \inf\{t \mid S(P_t) \geq (1 - \varepsilon)S_0\}$ (expected time to generate diversity). Assumptions are stated in the second column. Definitions for *respectful* and *tail-bounded* are given in Sect. 2. Finally, $\ln^+(x) := \max\{1, \ln(x)\}$

Upper time bound	Assumptions
General bounds:	
$E(T_{\varepsilon, \downarrow}) \in O\left(\frac{\mu \cdot \min\{\mu, n/\chi\}}{\varepsilon} \cdot \ln\left(\frac{1+n/(\mu\chi)}{\varepsilon}\right)\right)$	
$E(T_{\varepsilon, \uparrow}) \in O\left(\frac{n}{\varepsilon\chi} \cdot \ln\left(\frac{1+n/(\mu^2\chi)}{\varepsilon}\right)\right)$	
$E(T_{\varepsilon, \uparrow}) \in O\left(\frac{\min\{\mu^2, n/\chi\} + \frac{1}{\chi} \log_q\left(\frac{\mu n}{\varepsilon\chi}\right)}{\varepsilon} \cdot \ln\left(\frac{1+\mu n^2/(\varepsilon\chi^2 \ln q)}{\varepsilon}\right)\right)$	respectful crossover, q -tail-bounded mutation
Bounds simplified for $\varepsilon, \chi = \Omega(1)$:	
$E(T_{\varepsilon, \downarrow}) \in O(\mu^2 \ln^+(n/(\mu\chi)))$	$\mu\chi = O(n)$
$E(T_{\varepsilon, \downarrow}) \in O(\mu n/\chi)$	$\mu\chi = \Omega(n)$
$E(T_{\varepsilon, \uparrow}) \in O(n \ln^+(n/(\mu^2\chi)))/\chi$	$\mu^2\chi = \Omega(n)$
$E(T_{\varepsilon, \uparrow}) \in O(n/\chi)$	respectful crossover, $(1 + \Omega(1))$ -tail-bounded mutation, $\mu^2\chi \leq \ln n$
$E(T_{\varepsilon, \uparrow}) \in O(\mu^2 \ln(n) + \ln^2(n))$	respectful crossover, $(1 + \Omega(1))$ -tail-bounded mutation, $\ln n \leq \mu^2\chi \leq n$
$E(T_{\varepsilon, \uparrow}) \in O(\mu^2 \ln n)$	

flat fitness landscape any new offspring is accepted, allowing the population to spread out without restrictions imposed by the topology of the search space. Thus, there is some hope that the diversity bounds of this paper may still hold as *upper bounds* in many non-neutral environments.

The second reason is that, in addition to ONEMAX and JUMP_k mentioned earlier, there are several processes of interest to the runtime analysis community that feature large degrees of neutrality or very low selective pressure, either continuously or temporarily.

- For the well-known LEADINGONES function, if the best-so-far fitness value is k then the bits at positions $k + 2, \dots, n$ receive no fitness signal and thus this subspace of the hypercube is a perfectly neutral environment. The dynamics of a $(\mu+1)$ EA or $(\mu+1)$ GA in *accepted steps* are similar to the dynamics studied in the following.
- *Clearing* [1, 8] is a diversity-preserving mechanism in which an individual of high fitness “clears” a region around itself, i.e. the fitness of “cleared” individuals is replaced with a plateau of low fitness. The evolution of the population happens on a flat fitness function, except for the fact that winner individuals are guaranteed to survive and continuously spawn offspring close to them.
- A similar process can be seen for $(\mu+1)$ EAs on HOTTOPIC functions. It has been shown that after an improving individual is found, the offspring of this individual may essentially evolve free from selective pressure for a while, as if they were in a fitness-neutral environment. The defects accumulated in this phase cause the $(\mu + 1)$ EA to take exponential time on HOTTOPIC if μ is a large constant [37].
- Selection pressure can also be absent if an evolutionary algorithm uses inappropriate parameter settings or operators that are not suitable. Examples of inefficient parameter settings are given in [38]. Selection pressure was also found to be nearly absent when using fitness-proportional replacement selection in probabilistic crowding [9] or when using stochastic pure ageing, where individuals are being removed from the population probabilistically [39]. So our results may be helpful to understand the effects of bad EA designs or parameter choices.

We emphasise that all these scenarios are unique in their own way: fitness plateaus have a topology that may be different from the hypercube; the clearing diversity mechanism continuously injects offspring of the current winners into the population; the phases without selective pressure in HOTTOPIC optimisation only last for a certain amount of time. These unique traits do affect the dynamics of population diversity. Therefore, our results only apply partially to those situations. Nevertheless, we believe that our study is a good starting point for better understanding such specific situations.

1.2 Related Work in Population Genetics

We remark that in biology, specifically in population genetics, evolution in the absence of selection has been studied as well, justified by the fact that many loci (bits) have little effect on the overall fitness of the organism [40, Chapter 3] and the hypothesis that evolution is largely driven by genetic drift [41]. According to [42], the $(\mu+1)$ EA is known in population genetics as the *Moran model* and the diversity measure is

known as *gene diversity* according to [43]. The gene diversity can be derived from *allele frequencies* (in our case: frequencies of bit values) as $\mu^2 \cdot \sum_{i=1}^n f_{i,0} \cdot f_{i,1}$, where $f_{i,b}$ for $i \in \{1, \dots, n\}$ and $b \in \{0, 1\}$ is the frequency of bit value b in the population at position i (cf. the discussion after Definition 5). For allele frequencies, equilibria on flat fitness landscapes are well-known [44].

However, despite these close links there are important differences between population genetics literature and our work. Firstly, apart from very few exceptions [45–47], studies in population genetics consider a fixed, constant number of loci (see, e.g. [44, 48, 49]). In contrast, our work deals with equilibria for gene diversity on strings of arbitrary length n (where n is often also used to parameterise the mutation strength). This is especially important because we focus on the expected time to approach (or skip over) the equilibrium state, and how this time depends on n . Apart from the exceptions cited above, we could not find directly related work in population genetics on the convergence speed or speed of adaptation that goes beyond $O(1)$ loci. Secondly, we directly study the effect of algorithmic components like mutation and crossover, and show how their properties affect the process. Finally, our work covers a much broader range of mutation and crossover operators, many of which are not natural in the context of population genetics.

2 Preliminaries

For $x, y \in \{0, 1\}^n$, the *Hamming distance* $H(x, y)$ is the number of positions in which x and y differ. For $k, \ell \in \mathbb{N}$ with $k \leq \ell$ we write $[k] := \{1, 2, \dots, k\}$ and $[k, \ell] := \{k, \dots, \ell\}$. By a *flat* (or *fitness-neutral*) fitness function we mean the function $f(x) = 0$ for all $x \in \{0, 1\}^n$. For $x \in \{0, 1\}^n$ we mean by $|x|_1$ the number of ones and by $|x|_0$ the number of zeros, respectively. By $\vec{i} \in \{0, 1, 2\}^n$ we mean $\vec{i} := (i, \dots, i)$ for $i \in \{0, 1, 2\}$. We write $\ln^+(z) := \max\{1, \ln z\}$.

2.1 Algorithms

We define the following schemes of a steady-state EA without crossover and a steady-state GA using crossover. The former starts with some initial population, selects a parent uniformly at random and creates an offspring y through some mutation operator. Then y replaces a worst search point z in the current population if its fitness is no worse than the fitness of z . The steady-state GA picks two parents uniformly at random with replacement and applies some crossover operator to the two parents, followed by some mutation operator applied to the offspring. The mutant replaces the worst member of the population if it is no worse.

We deliberately do not specify operators for initialisation, crossover and mutation at this point to obtain a scheme that is as general as possible. Note that Algorithm 2 chooses two parents with replacement. It is straightforward to adapt our results to selecting parents without replacement (that is, ensuring that two *different* parents are recombined), see Remark 19 in Sect. 5. Parent selection is assumed to be uniform. For our setting, this is no restriction: assuming the fitness function is flat and ties are broken

Algorithm 1: Scheme of a Steady-State $(\mu + 1)$ EA

```

1  $t \leftarrow 0$ 
   Initialise  $P_0$  as a multiset of  $\mu$  search points
   while termination criterion not met do
2   Select  $x \in P_t$  uniformly at random
      $y \leftarrow \text{mutation}(x)$ 
     Select  $z \in P_t$  uniformly at random from all search points with minimum fitness in  $P_t$ 
     if  $f(y) \geq f(z)$  then
3      $P_{t+1} \leftarrow (P_t \cup \{y\}) \setminus \{z\}$ 
4    $t \leftarrow t + 1$ 

```

Algorithm 2: Scheme of a Steady-State $(\mu + 1)$ GA

```

1  $t \leftarrow 0$ 
   Initialise  $P_0$  as a multiset of  $\mu$  search points
   while termination criterion not met do
2   Select  $x_1 \in P_t$  uniformly at random
     Select  $x_2 \in P_t$  uniformly at random
      $y \leftarrow \text{crossover}(x_1, x_2)$ 
      $y' \leftarrow \text{mutation}(y)$ 
     Select  $z \in P_t$  uniformly at random from all search points with minimum fitness in  $P_t$ 
     if  $f(y') \geq f(z)$  then
3      $P_{t+1} \leftarrow (P_t \cup \{y'\}) \setminus \{z\}$ 
4    $t \leftarrow t + 1$ 

```

uniformly at random, every selection method based on fitness values or rankings of search points boils down to uniform selection.

Algorithm 2 is a generalisation of Algorithm 1: if we choose a crossover operator that returns an arbitrary parent (called *boring crossover* in [35], the algorithm performs a mutation of a parent chosen uniformly at random as in Algorithm 1. It is also straightforward to implement a crossover probability p_c , that is, to apply some crossover operator c with probability p_c . In this case the crossover operator in Line 2 of Algorithm 2 performs a boring crossover with probability $1 - p_c$ and otherwise executes c .

In both schemes, in case of a fitness tie between the offspring and z , the offspring is preferred. This reflects a common strategy and it is useful for exploring plateaus. If the offspring is removed in case of equal fitness, or if z is selected from all search points with minimum fitness in $P_t \cup \{y'\}$ instead of P_t , steps removing the offspring are idle steps. In the latter case, if the fitness function is flat, an idle step occurs with probability $1/(\mu + 1)$. Idle steps do not affect the equilibrium states of population diversity, but they slow down the process by a factor $\mu/(\mu + 1)$, see Remark 20.

2.2 Mutation and Crossover Operators

One contribution of the paper is to characterise *diversity-neutral* crossover operators, which we will define in Sect. 5. In preparation for this, we define important properties of mutation and crossover operators.

An m -ary operator takes m search points $x_1, \dots, x_m \in \{0, 1\}^n$ as input and outputs $y \in \{0, 1\}^n$, where y may be random. For example, mutation operators are unary (1-ary) operators, and crossover operators are most often binary (2-ary), although crossover operators with higher arity exist as well. We will use the notion of *unbiased* operators by Lehre and Witt [50]. Intuitively, an operator is unbiased if it treats bit values and bit positions symmetrically. Formally, we require the following.

Definition 1 A m -ary operator $\text{op}(x_1, \dots, x_m)$ is *unbiased* if the following holds for all $x_1, \dots, x_m \in \{0, 1\}^n$. Let $D(y \mid x_1, \dots, x_m) := \Pr(\text{op}(x_1, \dots, x_m) = y)$.

(i) For every permutation of n bit positions σ we have

$$D(y \mid x_1, \dots, x_m) = D(\sigma(y) \mid \sigma(x_1), \dots, \sigma(x_m)).$$

(ii) For every $z \in \{0, 1\}^n$ we have (\oplus denoting exclusive OR)

$$D(y \mid x_1, \dots, x_m) = D(y \oplus z \mid x_1 \oplus z, \dots, x_m \oplus z).$$

Most mutation operators are unbiased, including standard bit mutation and the heavy-tailed mutation operators used in *fast EAs/GAs* [34]. Many of them (like standard bit mutation) are *tail-bounded*, which means that the probability of creating a certain offspring decreases exponentially in the Hamming distance to its parent.

Definition 2 For $q > 1$, an unbiased mutation operator mut is *q -tail-bounded* if $\Pr(H(\text{mut}(y), y) \geq \ell) \leq q^{-\ell}$ for every $\ell \geq 0$.

Many crossover operators are also unbiased, but not all of them are. A detailed discussion by Friedrich et al. can be found in [35]. For an unbiased, q -tail-bounded mutation operator which flips χ bits in expectation we have $\chi \leq \sum_{\ell \geq 1} q^{-\ell} \leq 1/(q - 1)$. Thus if $q \in 1 + \Omega(1)$ then $\chi = O(1)$.

A crossover operator is *respectful* [51] if components on which all parents agree are passed on to the offspring, i.e., the output is in the convex hull of the inputs (a characteristic of *geometric crossovers* [52]). For our purposes, the following description via masks is useful.

Definition 3 An m -ary operator op is *respectful* if it chooses a possibly random mask $a \in [m]^n$ (where the probabilities may depend on the parents) such that the i -th bit of y is taken from x_{a_i} .

Note that the mask in Definition 3 is not unique in positions in which parents have the same bit. We will consider respectful operators where the mask does not depend on the order of the parents. Here we restrict ourselves to binary operators.

Definition 4 For a binary respectful operator c , let $M(a, x_1, x_2)$ be the probability of $c(x_1, x_2)$ choosing the mask $a \in \{1, 2\}^n$. We call the mask *order-independent* if $M(a, x_1, x_2) = M(a, x_2, x_1)$ for all $x_1, x_2 \in \{0, 1\}^n$ and $a \in \{1, 2\}^n$, and we then say for short that c has an *order-independent mask* (OIM).

Since a respectful operator can be described by different masks, it can happen that the same respectful operator can either be described by an order-independent mask or by a mask that does depend on the order. For all our results, the *existence* of an order-independent mask is sufficient, so our results also apply to the case described above.

A respectful operator trivially has an OIM if the mask is created without considering the parents. Uniform crossover, biased uniform crossover (where each bit is chosen independently from parent x_1 with a given probability $c \in [0, 1]$) and k -point crossover are examples of respectful crossovers with OIM. Note that OIM does not imply symmetry between the parents. For instance, the operator which always returns the first parent, that is, $M(\bar{1}, x_1, x_2) = 1$, has an OIM since the mask $\bar{1}$ does not depend on the order of the parents; in fact, it does not depend on the parents at all. On the other hand, the bitwise AND operator is respectful, but does not have an OIM, as for bits where both parents differ, the mask has to reflect the unique parent having a bit value of 0. We give a formal proof in Lemma 22.

2.3 Diversity Measure

We consider the sum of Hamming distances as a natural and standard [43] diversity measure:

Definition 5 For a population $P_t = \{x_1, \dots, x_\mu\}$ and a search point $y \in \{0, 1\}^n$ we define

$$S(y) := S_{P_t}(y) := \sum_{i=1}^{\mu} H(x_i, y)$$

and

$$S(P_t) := \sum_{i=1}^{\mu} S_{P_t}(x_i) = \sum_{i=1}^{\mu} \sum_{j=1}^{\mu} H(x_i, x_j).$$

The double sum includes the Hamming distance of each pair x_i, x_j with $i \neq j$ twice. If instead we sum over all (i, j) with $i < j$, we would obtain $S(P_t)/2$. Other re-scalings are also interesting. The average value of $S(y)$ with $y \in P_t$ is $S(P_t)/\mu$. The expected Hamming distance of two uniform random points in P_t drawn with replacement is $S(P_t)/\mu^2$, and without replacement it is $S(P_t)/(\mu^2 - \mu)$. Since those values differ only by a fixed factor from $S(P_t)$, all our results transfer straightforwardly to these other measures.

The sum of Hamming distances is one of the oldest and most popular diversity metrics [43]. It can be calculated with $O(\mu n)$ operations [43], which is linear in the input size and hence optimal for all diversity measures that take into account all of a population’s genetic information. The idea is simply to count for each bit position i how many individuals have a 1 at position i . If this number is c_i , the contribution to $S(P_t)$ is $2c_i(\mu - c_i)$ as this is the number of pairs of population members that

have different values at bit i . Consequently, $S(P_t) = 2 \sum_{i=1}^n c_i(\mu - c_i)$ holds.² In the context of a $(\mu+1)$ EA, the vector c_1, \dots, c_n and thus $S(P_t)$ can be updated after one generation with $O(n)$ operations, which is again optimal. According to [53], the sum of Hamming distances has two desirable properties. Firstly, diversity increases when adding a new search point that is not yet contained in the population (called *monotonicity in species* [54]). Secondly, the diversity does not decrease when replacing the population with one where all pairs of solutions have a distance at least as large as the previous one (*monotonicity in distance* [54]). It does not fulfil the *twinning* property, stating that diversity should remain constant when adding a clone of a search point into the population [54], and it may be maximised by a population forming clusters of search points such that the clusters have a large Hamming distance [53]. In fact, $c_i(\mu - c_i)$ is maximised by $c_i := \lfloor \mu/2 \rfloor$ and thus $S(P_t) \leq \mu^2 n/2$. (However, we will often prefer the trivial bound $S(P_t) \leq \mu^2 n$ for the sake of simplicity.)

2.4 On Stationary Distributions and Mixing Times

A steady-state $(\mu+1)$ EA or $(\mu+1)$ GA can be described by a Markov chain over the state space of all possible populations. For most mutation and crossover operators, the Markov chain is irreducible if the algorithm runs on a flat fitness landscape. For example, standard bit mutation has a non-zero probability to create any offspring y from any parent x . Thus, there is a positive probability of creating any population P_2 from any initial population P_1 in a sequence of at most μ generations. The Markov chain is also usually aperiodic since there is a positive probability of adding a clone of the search point being removed, and hence there is a positive self-loop probability. In this case, by the fundamental theorem of Markov chains [55, Theorem 6.2], there exists a unique stationary distribution. The expected time to approach the stationary distribution is called *mixing time* and there is a well-established machinery for bounding mixing times (see, e.g. [56]). However, the Markov chain lives on the space of all possible populations, which has size $2^{n\mu}$, and even a logarithmic mixing time would be of order $\Omega(n\mu)$. Compared to this, our bound for approaching or crossing the equilibrium state from above is $O(\mu^2 \ln^+(n/\mu))$, which can be much smaller. We do not believe that such results can be directly deduced from mixing times.

Of course, the diversity $S(P_t)$ performs a random walk on a much smaller state space. But this is in general not a Markov chain, since there may be very different populations having the same overall diversity, and the possible values of $S(P_{t+1})$ depend on the details of the populations, not only on the value $S(P_t)$.

² Note that the summand $2c_i(\mu - c_i)c_i(\mu - c_i)$ can be written as $2\mu^2 \cdot ((c_i/\mu)(1 - c_i/\mu))$ where, in the notion of biology, c_i/μ is the allele frequency of the allele (bit value) 1 and $(1 - c_i/\mu)$ is the frequency of allele 0. Hence $S(P_t)$ can be easily computed from allele frequencies.

3 Drift of Population Diversity for Steady-State EAs Without Crossover

Now we will compute the expected change of $S(P_t)$, i.e. $E(S(P_{t+1}))$ for a given P_t . We break the process down into several steps, and work out a unifying formula for a very general situation, see Corollary 9. This includes the $(\mu+1)$ EA for any unbiased mutation operator (Theorem 10), but as we will see later in Sect. 5, it also includes the $(\mu+1)$ GA with a large variety of crossover operators.

We start with a lemma which describes the expected change for a fixed value of the offspring y' .

Lemma 6 Consider a population $P_t = \{x_1, \dots, x_\mu\}$ and a search point $y' \in \{0, 1\}^n$. Let $P_{t+1} := (P_t \cup \{y'\}) \setminus \{x_d\}$ for a uniformly random $d \in [\mu]$. Then

$$E(S(P_{t+1})) = \left(1 - \frac{2}{\mu}\right) S(P_t) + \frac{2(\mu - 1)}{\mu} S_{P_t}(y').$$

Proof For any fixed $d \in [\mu]$, let $P_{t+1}^{-d} := (P_t \cup \{y'\}) \setminus \{x_d\}$. Then

$$S(P_{t+1}^{-d}) = \sum_{z \in P_{t+1}^{-d}} \sum_{z' \in P_{t+1}^{-d}} H(z, z').$$

The double sum contains summands $H(y', x_j)$ for all $j \in [\mu] \setminus \{d\}$ and summands $H(x_i, y')$ for all $i \in [\mu] \setminus \{d\}$ as well as a summand $H(y', y') = 0$. By virtue of $H(x_i, x_j) = H(x_j, x_i)$, this equals

$$= \sum_{i=1, i \neq d}^{\mu} \sum_{j=1, j \neq d}^{\mu} H(x_i, x_j) + 2 \sum_{i=1, i \neq d}^{\mu} H(x_i, y').$$

Compared to $S(P_t)$, the double sum is missing summands $H(x_d, x_j)$ for all $j \in [\mu] \setminus \{d\}$ and summands $H(x_i, x_d)$ for all $i \in [\mu] \setminus \{d\}$ as well as a summand $H(x_d, x_d) = 0$. Thus, this is equal to

$$\begin{aligned} &= \sum_{i=1}^{\mu} \sum_{j=1}^{\mu} H(x_i, x_j) + 2 \sum_{i=1, i \neq d}^{\mu} H(x_i, y') - 2 \sum_{i=1}^{\mu} H(x_i, x_d) \\ &= S(P_t) + 2 \sum_{i=1, i \neq d}^{\mu} H(x_i, y') - 2 \sum_{i=1}^{\mu} H(x_i, x_d). \end{aligned} \tag{1}$$

Owing to the uniform choice of d , we get

$$E(S(P_{t+1})) = \frac{1}{\mu} \sum_{d=1}^{\mu} S(P_{t+1}^{-d}) = S(P_t) + \frac{2}{\mu} \sum_{d=1}^{\mu} \sum_{i=1, i \neq d}^{\mu} H(x_i, y') - \frac{2}{\mu} \sum_{d=1}^{\mu} \sum_{i=1}^{\mu} H(x_i, x_d).$$

The first double sum contains terms $H(x_i, y')$ for all $i \in [\mu]$ exactly $\mu - 1$ times. The second double sum equals $S(P_t)$. Thus,

$$= \left(1 - \frac{2}{\mu}\right) S(P_t) + \frac{2(\mu-1)}{\mu} \sum_{i=1}^{\mu} H(x_i, y').$$

□

The next lemma tells us how, for given x, z , mutating x changes the distance from a fixed search point z in expectation. Interestingly, if the mutation operator is unbiased then the result depends only on the expected number of bit flips, but not on the exact nature of the mutation operator.

Lemma 7 *Let $x, z \in \{0, 1\}^n$, and let y be the random search point obtained from x by an unbiased mutation operator that flips χ bits in expectation. Then*

$$E(H(z, y)) = \chi + \left(1 - \frac{2\chi}{n}\right) H(z, x).$$

Proof Let p_i be the probability of flipping the i -th bit of x . By unbiasedness, we have $p_i = p_j$ for all $i, j \in [n]$. We also have $\sum_{i=1}^n p_i = \chi$. Since all p_i are equal, this implies $np_i = \chi$, or $p_i = \chi/n$.

There are $H(z, x)$ positions on which x and z differ. In expectation, $\chi/n \cdot H(z, x)$ of them are flipped, and each flip decreases the distance from z by one. There are $n - H(z, x)$ positions on which x and z agree. Each such flip increases the distance from z by one, and their expectation is $\chi/n \cdot (n - H(z, x))$. Hence, $E(H(z, y))$ equals

$$H(z, x) - \frac{\chi H(z, x)}{n} + \frac{\chi(n - H(z, x))}{n} = \chi + (1 - 2\chi/n) H(z, x).$$

□

Lemmas 6 and 7 together allow us to derive how the diversity evolves if we create the offspring as a mutation of a given string y .

Theorem 8 *Consider a population $P_t = \{x_1, \dots, x_\mu\}$ and let $y \in \{0, 1\}^n$. Let y' be the random search point obtained from y by an unbiased mutation operator which flips χ bits in expectation. Let $P_{t+1} = (P_t \cup \{y'\}) \setminus \{x_d\}$ for a uniform random $d \in [\mu]$. Then*

$$E(S(P_{t+1})) = \left(1 - \frac{2}{\mu}\right) S(P_t) + 2(\mu - 1)\chi + \frac{2(\mu - 1)}{\mu} \left(1 - \frac{2\chi}{n}\right) S(y).$$

Proof Note that $S(y) = \sum_{i=1}^{\mu} H(x_i, y)$. Then by Lemma 6, the law of total probability and linearity of expectation

$$E(S(P_{t+1})) = \left(1 - \frac{2}{\mu}\right) S(P_t) + \frac{2(\mu - 1)}{\mu} E(S(y')). \tag{2}$$

On the other hand, by Lemma 7 and again linearity of expectation, for all $i \in [n]$,

$$E(H(x_i, y')) = \chi + \left(1 - \frac{2\chi}{n}\right) H(x_i, y). \tag{3}$$

Summing (3) over all i yields

$$E(S(y')) = \sum_{i=1}^{\mu} E(H(x_i, y')) = \mu\chi + \left(1 - \frac{2\chi}{n}\right) \sum_{i=1}^{\mu} H(x_i, y) = \mu\chi + \left(1 - \frac{2\chi}{n}\right) S(y). \tag{4}$$

Plugging (4) into (2) yields the theorem. □

Remarkably, Theorem 8 depends only on $S(y)$, not on y itself. This means that all y with the same value of $S(y)$ yield the same dynamics. Moreover, by the law of total probability the same still holds with $E(S(y))$ if y is random. The following corollary describes the special case that $E(S(y)) = S(P_t)/\mu$. As we will see later, this special case covers many interesting situations. In particular, it covers the $(\mu+1)$ EA, where the parent is chosen at random, and it covers the $(\mu+1)$ GA with any unbiased, respectful crossover operator.

Corollary 9 Consider a population $P_t = \{x_1, \dots, x_{\mu}\}$. Consider any process that

1. creates y by any random procedure such that $E(S(y)) = S(P_t)/\mu$;
2. creates y' from y by an unbiased mutation operator which flips χ bits in expectation;
3. sets $P_{t+1} = (P_t \cup \{y'\}) \setminus \{x_d\}$ for a uniformly random $d \in [\mu]$.

Then

$$E(S(P_{t+1})) = \left(1 - \frac{2}{\mu^2} - \frac{4(\mu - 1)\chi}{\mu^2 n}\right) S(P_t) + 2(\mu - 1)\chi.$$

Proof We apply Theorem 8 with a random y . By the law of total probability,

$$\begin{aligned} E(S(P_{t+1})) &= \left(1 - \frac{2}{\mu}\right) S(P_t) + 2(\mu - 1)\chi + \frac{2(\mu - 1)}{\mu} \left(1 - \frac{2\chi}{n}\right) E(S(y)) \\ &= \left(1 - \frac{2}{\mu}\right) S(P_t) + 2(\mu - 1)\chi + \frac{2(\mu - 1)}{\mu} \left(1 - \frac{2\chi}{n}\right) \frac{S(P_t)}{\mu} \\ &= \left(1 - \frac{2}{\mu} + \frac{2}{\mu} - \frac{2}{\mu^2} - \frac{4(\mu - 1)\chi}{\mu^2 n}\right) S(P_t) + 2(\mu - 1)\chi, \end{aligned}$$

and cancelling the $2/\mu$ -terms yields the corollary. □

As an immediate consequence, the $(\mu+1)$ EA with any unbiased mutation operator meets the conditions of Corollary 9. In the previous lemmas and theorems, P_t could be any fixed population. Since we now study the $(\mu+1)$ EA, the following theorem will denote by P_t the (random) t -th population of the $(\mu+1)$ EA.

Theorem 10 Consider any $(\mu+1)$ EA from Algorithm 1 with any unbiased mutation operator flipping χ bits in expectation and a population size of μ on a flat fitness function. Then for all populations P_t

$$E(S(P_{t+1}) \mid P_t) = \left(1 - \frac{2}{\mu^2} - \frac{4(\mu - 1)\chi}{\mu^2 n}\right) S(P_t) + 2(\mu - 1)\chi.$$

Proof This is an immediate consequence of Corollary 9, where $y \in P_t$ is chosen randomly, since such a random parent y satisfies

$$E(S(y) \mid P_t) = \frac{1}{\mu} \sum_{i=1}^{\mu} \sum_{j=1}^{\mu} H(x_i, x_j) = \frac{S(P_t)}{\mu}.$$

□

Finally we describe the probability that a strong increase of $S(P_t)$ occurs if we use arbitrary respectful m -ary operators, followed by a mutation operator. A limited increase by at most a factor of two can occur due to crossover. In case of tail-bounded mutation operators, the probability of any further increase becomes exponentially small. We will use this insight to derive upper bounds on the expected time until the equilibrium state is reached from below.

Lemma 11 Consider a population $P_t = \{x_1, \dots, x_\mu\}$. Let *mut* be any mutation operator. Consider any process that

1. creates y by any respectful m -ary operator from x_1, \dots, x_μ ;
2. creates $y' := \text{mut}(y)$;
3. selects any $d \in [\mu]$ and sets $P_{t+1} = (P_t \cup \{y'\}) \setminus \{x_d\}$.

Then for every $k \geq 0$

$$\Pr(S(P_{t+1}) \geq \min\{2S(P_t), S(P_t) + 2\mu n\} + 2k\mu \mid P_t) \leq \Pr(H(y, y') \geq k \mid P_t).$$

Proof For brevity, we fix P_t and omit conditioning on P_t from the notation. We first show that

$$S(P_t \cup \{y\}) \leq \min\{2S(P_t), S(P_t) + 2\mu n\}. \tag{5}$$

Note that the left hand side is a population of size $\mu + 1$, while the right hand side is a population of size μ . To see $S(P_t \cup \{y\}) \leq S(P_t) + 2\mu n$, we simply observe that $H(x_i, y) \leq n$ for all i , and $S(P_t \cup \{y\})$ contains 2μ summands of that form. To prove the other part, $S(P_t \cup \{y\}) \leq 2S(P_t)$, consider a position $\ell \in [n]$. Since y was obtained by a respectful operator, there exists $i \in [\mu]$ such that $y_\ell = (x_i)_\ell$. Hence,

$$2 \sum_{j=1}^{\mu} |y_\ell - (x_j)_\ell| = \sum_{j=1}^{\mu} (|(x_i)_\ell - (x_j)_\ell| + |(x_j)_\ell - (x_i)_\ell|) \leq \sum_{j=1}^{\mu} \sum_{j'=1}^{\mu} |(x_j)_\ell - (x_{j'})_\ell|, \tag{6}$$

where the last inequality follows because every summand with indices (i, j) or (j, i) also appears in the double sum, except that the summand for $j = i$ appears only once, but that summand is zero. Summing over all ℓ in (6) yields $2 \sum_{j=1}^{\mu} H(y, x_j)$ on the left hand side and $\sum_{j=1}^{\mu} \sum_{j'=1}^{\mu} H(x_{j'}, x_j)$ on the right hand side. Thus,

$$2 \sum_{j=1}^{\mu} H(y, x_j) \leq \sum_{j'=1}^{\mu} \sum_{j=1}^{\mu} H(x_{j'}, x_j),$$

and the claim in (5) follows from

$$S(P_t \cup \{y\}) = S(P_t) + 2 \sum_{j=1}^{\mu} H(y, x_j) \leq S(P_t) + \sum_{j'=1}^{\mu} \sum_{j=1}^{\mu} H(x_{j'}, x_j) = 2S(P_t).$$

Next we observe that changing one bit in y can increase $H(x_i, y)$ by at most one. The diversity $S(P_t \cup \{y\})$ contains each of the summands $H(x_i, y)$ and $H(y, x_i)$ exactly once for each i , and does not involve y otherwise (since $H(y, y) = 0$). Hence,

$$S(P_t \cup \{y'\}) \leq S(P_t \cup \{y\}) + 2\mu H(y, y') \leq \min\{2S(P_t), S(P_t) + 2\mu n\} + 2\mu H(y, y').$$

Finally, $S(P_{t+1}) \leq S(P_t \cup \{y\})$ since $P_{t+1} \subset P_t \cup \{y\}$. Together, we have

$$S(P_{t+1}) \leq \min\{2S(P_t), S(P_t) + 2\mu n\} + 2\mu H(y, y').$$

Hence, the event “ $S(P_{t+1}) \geq \min\{2S(P_t), S(P_t) + 2\mu n\} + 2\mu k$ ” implies that $H(y, y') \geq k$, and thus, conditioned on any fixed P_t ,

$$\Pr(S(P_{t+1}) \geq \min\{2S(P_t), S(P_t) + 2\mu n\} + 2\mu k) \leq \Pr(H(y, y') \geq k),$$

as required. □

4 Equilibria and Time Bounds

The preceding results give immediate insights about an equilibrium state for the population diversity. Define

$$\alpha := 2(\mu - 1)\chi \quad \text{and} \quad \delta := \frac{2}{\mu^2} + \frac{4(\mu - 1)\chi}{\mu^2 n}, \tag{7}$$

then Corollary 9 and Theorem 10 state that

$$E(S(P_{t+1}) \mid P_t) = (1 - \delta)S(P_t) + \alpha. \tag{8}$$

This condition was described in [57] as *negative multiplicative drift with an additive disturbance* (in [57] only lower hitting time bounds were given, while we will prove upper bounds). An equilibrium state with zero drift is attained for

$$S_0 := \frac{\alpha}{\delta} = \frac{(\mu - 1)\mu^2\chi n}{2(\mu - 1)\chi + n}$$

since then $E(S(P_{t+1}) \mid S(P_t) = S_0) = (1 - \delta) \cdot \frac{\alpha}{\delta} + \alpha = \frac{\alpha}{\delta} = S_0$.

If $(\mu - 1)\chi \ll n$ then the equilibrium is close to $(\mu - 1)\mu^2\chi$ and the average Hamming distance is $(\mu - 1)\chi$, growing linearly in the population size and linearly in the mutation strength χ . If $(\mu - 1)\chi \gg n$ then the equilibrium is close to $\mu^2 n/2$, that is, the average Hamming distance between population members is roughly $n/2$. This equals the expected Hamming distance between population members in a uniform random population. Note that the average Hamming distance at the equilibrium is at most

$$\frac{S_0}{\mu^2} = \frac{(\mu - 1)\chi n}{2(\mu - 1)\chi + n} \leq \frac{(\mu - 1)\chi n}{\max\{2(\mu - 1)\chi, n\}} = \min\{(\mu - 1)\chi, n/2\},$$

hence bounded by the value $n/2$ for a uniform random population. It is bounded from below by

$$\frac{S_0}{\mu^2} = \frac{(\mu - 1)\chi n}{2(\mu - 1)\chi + n} \geq \frac{(\mu - 1)\chi n}{2 \max\{2(\mu - 1)\chi, n\}} = \min\{(\mu - 1)\chi/2, n/4\}.$$

Hence, in order to obtain an average Hamming distance of $\Theta(n)$, we must have $\mu\chi = \Omega(n)$. In particular, for $\chi = \Theta(n)$, the population size must be at least linear in n .

We stress again that for given μ and n , the equilibrium value α/δ only depends on the expected number χ of flipped bits. For example, both RLS mutation, which flips exactly one bit, and standard bit mutation with mutation rate $1/n$ have the same value $\chi = 1$ and hence the same equilibrium state. Recently, another type of mutation operator has become quite popular, where the probability p_k of flipping k bits has a *heavy tail* [34]. Usually, it scales as $p_k \sim k^{-\tau}$ for some $\tau > 1$. In many applications, all values of τ lead to similar results. However, here they lead to qualitatively different behaviour due to different values of χ . More precisely, $\tau > 2$ leads to $\chi = \Theta(1)$ [58], which gives the same dynamics as standard bit mutation with slightly different mutation rate $\Theta(1/n)$. In particular, $\alpha/\delta = \Theta(\mu^3)$ for $\mu \leq n$. On the other hand, $\tau \in (1, 2)$ leads to $\chi = \Theta(\sum_{k=1}^n k \cdot p_k) = \Theta(\sum_{k=1}^n k^{1-\tau}) = \Theta(n^{2-\tau})$. Thus for $\mu \leq n^{\tau-1}$ the equilibrium state is $\alpha/\delta = \Theta(\mu^3\chi) = \Theta(\mu^3 n^{2-\tau})$. For constant μ , this means that the equilibrium state jumps from $\Theta(1)$ to $n^{\Omega(1)}$ as τ crosses the threshold $\tau = 2$. For $\tau = 2$, we get an intermediate regime of $\chi = \Theta(\log n)$ [58].

For another perspective on the equilibrium state we consider the distance $D(P_t) := S(P_t) - \alpha/\delta$. With (8) this changes as

$$E(D(P_{t+1}) \mid P_t) = E(S(P_{t+1}) \mid P_t) - \alpha/\delta = (1 - \delta)S(P_t) + \alpha - \alpha/\delta = (1 - \delta)D(P_t).$$

Hence, the distance from the equilibrium state shows a multiplicative drift. However, note crucially that $D(P_t)$ may take *positive and negative* values and the multiplicative drift theorem [59] is not applicable. The process is quite different from the usual situation of multiplicative drift, in which the target state is reached quickly. In fact, the equilibrium state $S(P_t) = \alpha/\delta$ may never be reached, since it might not be achievable due to rounding issues or if the diversity changes in large steps. However, we will show that the diversity will quickly reach an *approximation* of the equilibrium state, or that the equilibrium state will be overshoot.

The following theorem gives three upper time bounds. When starting with a diversity of $S(P_t) > (1 + \varepsilon)\alpha/\delta$, we bound the expected time to reach a diversity at most $(1 + \varepsilon)\alpha/\delta$. This reflects a scenario where a population has an above-average diversity and we ask how long it takes for diversity to reduce. Similarly, starting with a population of little or no diversity, $S(P_t) < (1 - \varepsilon)\alpha/\delta$, we give two different bounds for the expected time for diversity to increase to at least $(1 - \varepsilon)\alpha/\delta$. This reflects the expected time to generate diversity. The first bound holds in general, and the second bound holds for combinations of an arbitrary respectful crossover operator with a tail-bounded mutation operator. As it might be of independent interest, we formulate this theorem for general finite stochastic processes $(X_t)_{t \geq 0}$ in \mathbb{N}_0 whose drift is bounded from above or below by $(1 - \delta)X_t + \alpha$, respectively, where $\alpha, \delta > 0$ are arbitrary values.

Theorem 12 *Fix $0 < \varepsilon \leq 1$ and suppose that $\alpha, \delta > 0$. Let $(X_t)_{t \geq 0}$ with $X_t \in \{0, \dots, X_{\max}\}$ be a stochastic process. Let $T_{\varepsilon, \downarrow} := \inf \{t \mid X_t \leq (1 + \varepsilon)\frac{\alpha}{\delta}\}$ and $T_{\varepsilon, \uparrow} := \inf \{t \mid X_t \geq (1 - \varepsilon)\frac{\alpha}{\delta}\}$.*

(i) *If $E(X_{t+1} \mid X_t = x) \leq (1 - \delta)x + \alpha$ for all $x > \frac{\alpha}{\delta}(1 + \varepsilon)$ then*

$$E(T_{\varepsilon, \downarrow}) \leq \frac{4}{\varepsilon\delta} \ln \left(\frac{2\delta X_{\max}}{\varepsilon\alpha} \right).$$

(ii) *Assume there is $\Delta_{\max} > 0$ such that for all $x < (1 - \varepsilon)\frac{\alpha}{\delta}$ and all $t \geq 0$ we have $X_{t+1} \leq \alpha/\delta + \Delta_{\max}$ whenever $X_t = x$ and we have $E(X_{t+1} \mid X_t = x) \geq (1 - \delta)x + \alpha$. Then*

$$E(T_{\varepsilon, \uparrow}) \leq \frac{4\Delta_{\max}}{\varepsilon\alpha} \ln \left(\frac{2\alpha + 2\delta\Delta_{\max}}{\varepsilon\alpha} \right). \tag{9}$$

(iii) *If $E(X_{t+1} \mid X_t = x) \geq (1 - \delta)x + \alpha$ for all $x < (1 - \varepsilon)\frac{\alpha}{\delta}$ and there is $M \geq 0$ such that for all t and all values $X_1, \dots, X_t < (1 - \varepsilon)\frac{\alpha}{\delta}$:*

$$\Pr(X_{t+1} > \alpha/\delta + M \mid X_1, \dots, X_t) \leq \frac{\varepsilon\alpha}{2X_{\max}}, \tag{10}$$

then

$$E(T_{\varepsilon, \uparrow}) \leq \frac{8M}{\varepsilon\alpha} \ln \left(\frac{2\alpha + 2\delta M}{\varepsilon\alpha} \right). \tag{11}$$

Proof We use a direct argument, similar to the proof of the tail bound for multiplicative drift [60].

(i): We are interested in the hitting time $T_{\varepsilon, \downarrow}$. For this hitting time, it is irrelevant how the process transitions further from states $X_t \leq \alpha/\delta \cdot (1 + \varepsilon)$, and we may change these transitions arbitrarily without affecting $T_{\varepsilon, \downarrow}$. Hence, even though the condition $E(X_{t+1} \mid X_t = x) \leq (1 - \delta)x + \alpha$ is only required for $x > \alpha/\delta \cdot (1 + \varepsilon)$, we may safely assume that the condition holds for all $x \in \{0, 1, \dots, X_{\max}\}$, e.g., by making the process transition to $X_{t+1} = 0$ from all $X_t \leq \alpha/\delta \cdot (1 + \varepsilon)$. With this assumption, we show by induction on t that

$$E(X_t \mid X_0) \leq \sum_{i=0}^{t-1} (1 - \delta)^i \alpha + (1 - \delta)^t X_0. \tag{12}$$

For the base case $t = 0$ we have $E(X_0 \mid X_0) = X_0$ and $\sum_{i=0}^{t-1} (1 - \delta)^i \alpha + X_0 = X_0$ as the sum is empty. Now assume the claim holds for $E(X_t \mid X_0)$. Using the law of total expectation $E(E(X \mid Y, Z) \mid Z) = E(X \mid Z)$

$$\begin{aligned} E(X_{t+1} \mid X_0) &= E(E(X_{t+1} \mid X_t) \mid X_0) \\ &\leq E((1 - \delta)X_t + \alpha \mid X_0) = (1 - \delta)E(X_t \mid X_0) + \alpha. \end{aligned}$$

Applying the induction hypothesis, we get

$$\begin{aligned} E(X_{t+1} \mid X_0) &\leq (1 - \delta) \left(\sum_{i=0}^{t-1} (1 - \delta)^i \alpha + (1 - \delta)^t X_0 \right) + \alpha \\ &= \sum_{i=0}^{t-1} (1 - \delta)^{i+1} \alpha + (1 - \delta)^{t+1} X_0 + \alpha \\ &= \sum_{i=0}^t (1 - \delta)^i \alpha + (1 - \delta)^{t+1} X_0. \end{aligned}$$

From (12), we get, bounding the sum by an infinite series $\sum_{i=0}^{\infty} (1 - \delta)^i = \frac{1}{\delta}$ and using $1 - \delta \leq e^{-\delta}$ as well as $X_0 \leq X_{\max}$,

$$E(X_t \mid X_0) \leq \sum_{i=0}^{t-1} (1 - \delta)^i \alpha + (1 - \delta)^t X_0 \leq \frac{\alpha}{\delta} + e^{-\delta t} \cdot X_{\max}.$$

Choosing $t := \ln(X_{\max} \cdot \delta/\alpha \cdot 2/\varepsilon)/\delta$, we obtain

$$E(X_t \mid X_0) \leq \frac{\alpha}{\delta} + \frac{1}{X_{\max}} \cdot \frac{\alpha}{\delta} \cdot \frac{\varepsilon}{2} \cdot X_{\max} = \frac{\alpha}{\delta} \cdot \left(1 + \frac{\varepsilon}{2}\right).$$

By Markov’s inequality we get, for all values of X_0 ,

$$\Pr\left(X_t \geq \frac{\alpha}{\delta} \cdot (1 + \varepsilon)\right) \leq \frac{\frac{\alpha}{\delta} \cdot \left(1 + \frac{\varepsilon}{2}\right)}{\frac{\alpha}{\delta} \cdot (1 + \varepsilon)} = \frac{1 + \varepsilon/2}{1 + \varepsilon}$$

and thus $\Pr(X_t < \frac{\alpha}{\delta} \cdot (1 + \varepsilon)) \geq 1 - \frac{1+\varepsilon/2}{1+\varepsilon} = \frac{\varepsilon/2}{1+\varepsilon} \geq \frac{\varepsilon}{4}$, where the last inequality follows from $\varepsilon \leq 1$.

In case $X_t > \frac{\alpha}{\delta} \cdot (1 + \varepsilon)$ we repeat the above arguments with a further phase of t steps. Here we exploit that the above bound was made independent of X_0 . The expected number of phases required is at most $4/\varepsilon$. This gives an upper bound of $4t/\varepsilon$.

(ii): Since we are only interested in $T_{\varepsilon,\uparrow}$, we may assume that the process becomes stationary afterwards, i.e. $X_{T_{\varepsilon,\uparrow}} = X_{T_{\varepsilon,\uparrow}+1} = X_{T_{\varepsilon,\uparrow}+2} = \dots$. Moreover, we may assume $X_0 < (1 - \varepsilon)\alpha/\delta$, since otherwise there is nothing to show. Define $Y_{\max} := \alpha/\delta + \Delta_{\max}$ and $Y_t := Y_{\max} - X_t$. Then $0 \leq Y_t \leq Y_{\max}$ for all $t \geq 0$ by assumption on X_t .

Let $\varepsilon' := \frac{\varepsilon\alpha}{\delta Y_{\max} - \alpha}$, $\delta' := \delta$, $\alpha' := \delta Y_{\max} - \alpha$. Then the event “ $X_t \geq (1 - \varepsilon)\frac{\alpha}{\delta}$ ” is equivalent to the event “ $Y_t \leq (1 + \varepsilon')\frac{\alpha'}{\delta'}$ ”, because

$$\begin{aligned} (1 + \varepsilon')\frac{\alpha'}{\delta'} &= (1 + \varepsilon')(Y_{\max} - \frac{\alpha}{\delta}) = Y_{\max} - \frac{\alpha}{\delta} + \varepsilon' \frac{\delta Y_{\max} - \alpha}{\delta} \\ &= Y_{\max} - \frac{\alpha}{\delta} + \varepsilon \frac{\alpha}{\delta} = Y_{\max} - (1 - \varepsilon)\frac{\alpha}{\delta}, \end{aligned}$$

and because $Y_t = Y_{\max} - X_t$. We can describe $T_{\varepsilon,\uparrow}$ as the first point in time when $Y_t \leq (1 + \varepsilon')\frac{\alpha'}{\delta'}$, since this is equivalent to $X_t \geq (1 - \varepsilon)\frac{\alpha}{\delta}$.

Moreover, the same calculation shows that for all $y > (1 + \varepsilon')\frac{\alpha'}{\delta'}$ the event “ $Y_t = y$ ” implies $X_t < (1 - \varepsilon)\frac{\alpha}{\delta}$, so that the drift bound for X_t is applicable. Hence, for any such y , the drift of Y_t is

$$\begin{aligned} E(Y_{t+1} \mid Y_t = y) &= E(Y_{\max} - X_{t+1} \mid X_t = Y_{\max} - y) \\ &= Y_{\max} - E(X_{t+1} \mid X_t = Y_{\max} - y) \\ &\leq Y_{\max} - (1 - \delta)(Y_{\max} - y) - \alpha = (1 - \delta')y + \alpha'. \end{aligned}$$

Therefore, the prerequisites of part (i) are satisfied by Y_t with parameters ε' , δ' and α' . Hence, part (i) applied to Y_t gives

$$E(T_{\varepsilon,\uparrow}) \leq \frac{4}{\varepsilon'\delta'} \cdot \ln\left(\frac{2\delta'Y_{\max}}{\varepsilon'\alpha'}\right) \leq \frac{4\Delta_{\max}}{\varepsilon\alpha} \ln\left(\frac{2\alpha + 2\delta\Delta_{\max}}{\varepsilon\alpha}\right).$$

(iii): Consider the stochastic process $Y_t := \min\{X_t, \alpha/\delta + M\}$. We will show that this process satisfies the conditions in (ii) with $\Delta_{\max} := M$, $\alpha' := \alpha/2$, $\delta' := \delta/2$ and $\varepsilon' := \varepsilon$. Note that this will imply the claim since the condition “ $Y_t < (1 - \varepsilon')\alpha'/\delta'$ ” is equivalent to “ $X_t < (1 - \varepsilon)\alpha/\delta$ ” and since the right hand sides of (9) (with M , α' , δ' , ε' instead of Δ_{\max} , α , δ , ε) and of (11) coincide. The condition $Y_{t+1} \leq \alpha/\delta + \Delta_{\max}$ is

trivially satisfied by definition of Y_t . Moreover, by definition of Y_t we may write

$$\begin{aligned} Y_{t+1} &= X_{t+1} - \left(X_{t+1} - \frac{\alpha}{\delta} - M\right) \cdot \mathbb{1}\{X_{t+1} > \frac{\alpha}{\delta} + M\} \\ &\geq X_{t+1} - X_{\max} \cdot \mathbb{1}\{X_{t+1} > \frac{\alpha}{\delta} + M\}. \end{aligned}$$

Hence, for any $x < (1 - \varepsilon)\frac{\alpha}{\delta}$, we can bound

$$\begin{aligned} \mathbb{E}(Y_{t+1} \mid Y_t = x) &= \mathbb{E}(Y_{t+1} \mid X_t = x) \\ &\geq \mathbb{E}(X_{t+1} \mid X_t = x) - X_{\max} \cdot \mathbb{E}(\mathbb{1}\{X_{t+1} > \frac{\alpha}{\delta} + M\} \mid X_t = x) \\ &= \mathbb{E}(X_{t+1} \mid X_t = x) - X_{\max} \cdot \Pr(X_{t+1} > \frac{\alpha}{\delta} + M \mid X_t = x) \\ &\geq (1 - \delta)x + \alpha - X_{\max} \cdot \frac{\varepsilon\alpha}{2X_{\max}}, \end{aligned}$$

where the last step holds by the two prerequisites of (iii). Using $x < (1 - \varepsilon)\frac{\alpha}{\delta} < \frac{\alpha}{\delta}$, we can continue

$$\begin{aligned} \mathbb{E}(Y_{t+1} \mid Y_t = x) &\geq (1 - \delta)x + \alpha - \frac{\varepsilon\alpha}{2} = (1 - \frac{\delta}{2})x + \frac{\alpha}{2} - \frac{\delta}{2}x + \frac{\alpha}{2} - \frac{\varepsilon\alpha}{2} \\ &\geq (1 - \frac{\delta}{2})x + \frac{\alpha}{2} - \frac{\delta}{2} \cdot (1 - \varepsilon)\frac{\alpha}{\delta} + \frac{\alpha}{2} - \frac{\varepsilon\alpha}{2} = (1 - \delta')x + \alpha'. \end{aligned}$$

This shows that the second condition of (ii) is satisfied, and concludes the proof. \square

We remark that there are also overshoot-aware multiplicative drift theorems [61] which could also be directly applied in the situation of Theorem 12, but those lead to poor results since the upper bounds include the expected overshoot, which may be very large.

To apply Theorem 12 to our situation, we first prove a bound on Δ_{\max} .

Lemma 13 Consider a population $P_t = \{x_1, \dots, x_\mu\}$ and any $y \in \{0, 1\}^n$. Let $P_{t+1} = (P_t \cup \{y\}) \setminus \{x_d\}$ for some $d \in [\mu]$. Then $|S(P_{t+1}) - S(P_t)| \leq 2(\mu - 1)n$.

Proof By Equation (1) we have

$$S(P_{t+1}) - S(P_t) = 2 \sum_{i=1, i \neq d}^{\mu} H(x_i, y') - 2 \sum_{i=1, i \neq d}^{\mu} H(x_i, x_d),$$

and the bound follows since both the positive and the negative term are at most $2(\mu - 1)n$. \square

Theorem 14 Let $\mu \geq 2$ and consider a steady-state evolutionary algorithms meeting the conditions of Corollary 9 with $\alpha := 2(\mu - 1)\chi$ and $\delta := \frac{2}{\mu^2} + \frac{4(\mu-1)\chi}{\mu^2 n}$ as in (7). Fix $0 < \varepsilon \leq 1$ and let $T_{\varepsilon, \downarrow} := \inf \{t \mid S(P_t) \leq (1 + \varepsilon)\frac{\alpha}{\delta}\}$ and $T_{\varepsilon, \uparrow} :=$

$\inf \{t \mid S(P_t) \geq (1 - \varepsilon)\frac{\alpha}{8}\}$. Then

$$E(T_{\varepsilon, \downarrow}) = O\left(\frac{\mu \cdot \min\{\mu, n/\chi\}}{\varepsilon} \cdot \ln\left(\frac{1 + n/(\mu\chi)}{\varepsilon}\right)\right), \tag{13}$$

$$E(T_{\varepsilon, \uparrow}) = O\left(\frac{n}{\varepsilon\chi} \cdot \ln\left(\frac{1 + n/(\mu^2\chi)}{\varepsilon}\right)\right). \tag{14}$$

Moreover, if step 1 of the GA in Corollary 9 is obtained by a respectful operator, and step 2 is obtained by a q -tail-bounded mutation operator for some $q \in (1, 2]$ then

$$E(T_{\varepsilon, \uparrow}) = O\left(\frac{\min\{\mu^2, n/\chi\} + \frac{1}{\chi} \log_q\left(\frac{\mu n}{\varepsilon\chi}\right)}{\varepsilon} \cdot \ln\left(\frac{1 + \mu n^2/(\varepsilon\chi^2 \ln q)}{\varepsilon}\right)\right). \tag{15}$$

Proof In order to apply Theorem 12 to our case of $(X_t)_{t \geq 0} = (S(P_t))_{t \geq 0}$, we may set $\Delta_{\max} := 2(\mu - 1)n$ by Lemma 13. Moreover, we have $S(P_t)_{\max} \leq \mu^2 n$, since two individuals have Hamming distance at most n and so the diversity is at most $2\binom{\mu}{2}n$. For (13), we have $1/\delta = \frac{\mu^2 n}{2n + 4(\mu - 1)\chi}$, which implies $1/\delta \in \Theta(\mu \cdot \min\{\mu, n/\chi\})$. Now (13) follows immediately by plugging this into the bounds from Theorem 12. For (14), note that $\Delta_{\max}/\alpha = n/\chi$. Thus, Theorem 12 implies

$$E(T_{\varepsilon, \uparrow}) \leq \frac{4n}{\varepsilon\chi} \ln\left(\frac{2}{\varepsilon} + \frac{2n}{\varepsilon\chi} \cdot \left(\frac{2}{\mu^2} + \frac{4(\mu - 1)\chi}{\mu^2 n}\right)\right) = O\left(\frac{n}{\varepsilon\chi} \cdot \ln\left(\frac{1 + n/(\mu^2\chi)}{\varepsilon}\right)\right),$$

where we could omit the last term in the logarithm since $(\mu - 1)/\mu^2 = O(1)$. For (15), we will show that we may apply part (iii) of Theorem 12 with $M := \min\{\alpha/\delta, 2\mu n\} + 2k\mu$ where k is defined as $k := \lceil \log_q(2S(P_t)_{\max}/(\varepsilon\alpha)) \rceil$. Let mut be the mutation operator in step 2 of Corollary 9. By Lemma 11, for all $x < (1 - \varepsilon)\frac{\alpha}{8}$, since $\alpha/\delta + M = \min\{2\alpha/\delta, \alpha/\delta + 2\mu n\} + 2k\mu > \min\{2x, x + 2\mu n\} + 2k\mu$,

$$\begin{aligned} & \Pr(S(P_{t+1}) \geq \alpha/\delta + M \mid S(P_t) = x) \\ & \leq \Pr(S(P_{t+1}) \geq \min\{2x, x + 2\mu n\} + 2k\mu \mid S(P_t) = x) \\ & \leq \Pr(H(\text{mut}(y), y) \geq k) \leq q^{-k} \leq \frac{\varepsilon\alpha}{2S(P_t)_{\max}}, \end{aligned}$$

where y is created by the respectful operator in step 1 of Corollary 9. Thus Equation (10) in part (iii) of Theorem 12 is satisfied, and

$$E(T_{\varepsilon, \uparrow}) \leq \frac{8M}{\varepsilon\alpha} \ln\left(\frac{2\alpha + 2\delta M}{\varepsilon\alpha}\right). \tag{16}$$

It remains to estimate this term. Since $\mu \geq 2$ and $S(P_t)_{\max} \leq \mu^2 n$, we bound $\alpha = 2(\mu - 1)\chi \geq \mu\chi$, and thus $k = \lceil \log_q(2S(P_t)_{\max}/(\varepsilon\alpha)) \rceil \leq \lceil \log_q(2\mu n/(\varepsilon\chi)) \rceil$.

Moreover, $\min\{1/\delta, 2\mu n/\alpha\} \leq \min\{\mu^2, \mu n/\chi, 2n/\chi\} = \min\{\mu^2, 2n/\chi\}$. Hence,

$$\begin{aligned} \frac{M}{\alpha} &= \min\{1/\delta, 2\mu n/\alpha\} + \frac{2k\mu}{\alpha} \leq \min\{\mu^2, 2n/\chi\} + \frac{2\mu \lceil \log_q(2\mu n/(\varepsilon\chi)) \rceil}{\alpha} \\ &= O\left(\min\{\mu^2, n/\chi\} + \frac{1}{\chi} \cdot \log_q\left(\frac{\mu n}{\varepsilon\chi}\right)\right). \end{aligned} \tag{17}$$

We want to plug this into (16), where the factor M/α appears twice. Since the second appearance is in the logarithm, we aim for a simpler (but much cruder) bound to use there. We claim that

$$\frac{M}{\alpha} = O\left(\frac{\mu n^2}{\varepsilon\chi^2 \ln q}\right). \tag{18}$$

To see (18), first note that the first summand in (17) is covered by (18) since $\min\{\mu^2, n/\chi\} \leq n/\chi \leq \mu n^2/(\varepsilon\chi^2 \ln q)$, where the second step holds because $n/\chi \geq 1$, $\mu \geq 1$, $\varepsilon < 1$ and $\ln q \leq \ln 2 < 1$. The second summand is covered because $\frac{1}{\chi} \log_q(\mu n/(\varepsilon\chi)) = O(\mu n/(\varepsilon\chi^2 \ln q))$ and $n \geq 1$.

Now we plug (17) and (18) into (16) (into the first and second appearance of M/α , respectively) and use $\delta = O(1)$. We obtain

$$E(T_{\varepsilon, \uparrow}) = O\left(\frac{\min\{\mu^2, n/\chi\} + \frac{1}{\chi} \log_q\left(\frac{\mu n}{\varepsilon\chi}\right)}{\varepsilon} \cdot \ln\left(\frac{1 + \mu n^2/(\varepsilon\chi^2 \ln q)}{\varepsilon}\right)\right),$$

as required. □

We can simplify these bounds as follows. If $\varepsilon = \Omega(1)$, it can be dropped from all upper bounds. Then the bound from (13) becomes $E(T_{\varepsilon, \downarrow}) \in O(\mu \cdot \min\{\mu, n/\chi\} \cdot \ln(1 + n/(\mu\chi)))$. For $\mu\chi = O(n)$ this is $O(\mu^2 \ln^+(n/(\mu\chi)))$ and for $\mu\chi = \Omega(n)$ it is $O(\mu n/\chi)$. The bound from (14) is $E(T_{\varepsilon, \uparrow}) \in O(n \ln^+(n/(\mu^2\chi))/\chi)$, and for $\mu^2\chi = \Omega(n)$ this becomes $O(n/\chi)$ as then the logarithmic term is $\Theta(1)$.

If additionally $q = 1 + \Omega(1)$ then $\ln q = \Omega(1)$ and the tail bound implies $\chi = O(1)$. If we also assume $\chi = \Omega(1)$ then the bound (15) simplifies to

$$E(T_{\varepsilon, \uparrow}) \in O\left(\left(\min\{\mu^2, n\} + \log(\mu n)\right) \cdot \ln\left(1 + \mu n^2\right)\right).$$

Then, if $\mu^2 = O(\log n)$ the upper bound simplifies to $E(T_{\varepsilon, \uparrow}) \in O(\mu^2 \ln(n) + \ln^2 n)$. If $\mu^2 \in \Omega(\log n) \cap O(n)$, the term $\ln^2(n)$ can be dropped (as $\min\{\mu^2, n\} = \Omega(\log n)$) and we obtain the bound $O(\mu^2 \ln n)$. All upper bounds are listed in Table 1.

It should be stressed that these upper bounds are not necessarily tight. We shall discuss the functional dependencies on parameters n, μ , and χ , but should keep in mind that we are discussing upper bounds for which we do not have matching lower bounds.

In terms of the mutation strength χ , note that the bounds $E(T_{\varepsilon, \downarrow}) \in O(\mu n/\chi)$ for $\mu\chi = \Omega(n)$ and $E(T_{\varepsilon, \uparrow}) \in O(n \ln^+(n/(\mu^2\chi))/\chi)$ are proportional to $1/\chi$, the

inverse of the expected number of flipping bits. However, the upper bound $E(T_{\varepsilon,\downarrow}) \in O(\mu^2 \ln^+(n/(\mu\chi)))$ for $\mu\chi = O(n)$ shows a mild dependency on χ . The refined upper bounds for $(1 + \Omega(1))$ -tail-bounded mutation operators are much smaller than the general upper bounds and show no asymptotic dependence on χ since $\chi = \Theta(1)$.

In terms of the dependence on n , the upper bound $E(T_{\varepsilon,\downarrow}) \in O(\mu^2 \ln^+(n/(\mu\chi)))$ for $\mu\chi = O(n)$ depends very mildly on n , so the speed of reducing diversity is almost unaffected by the problem dimension. For the expected time to generate diversity, the general bound $E(T_{\varepsilon,\uparrow}) \in O(n \ln^+(n/(\mu^2\chi))/\chi)$, which does not rely on a tail bound on the number of bit flips by the mutation operator, is at least linear in n . Indeed, consider the scenario with $\mu = 2, \chi = 1$, where the mutation operator flips all n bits with probability $1/n$, and does nothing otherwise. When starting with two identical individuals then $E(T_{\varepsilon,\uparrow}) = \Omega(n)$. This example shows that without restrictions on the mutation operator, the linear dependency in n can occur. On the other hand, with the exponential tail bound on the mutation operator we obtain $E(T_{\varepsilon,\uparrow}) \in O(\mu^2 \ln(n) + \ln^2(n)/\chi)$, which depends only mildly on n .

Exponential tail bounds are quite common. In particular, *standard bit mutation*, which flips all bits independently, satisfies exponential tail bounds for $\chi = \Theta(1)$. We remark that in the proof of (15), we only used the exponential tail bound for one specific value of k , and it would also be possible to obtain improved bounds for $E(T_{\varepsilon,\uparrow})$ from Theorem 12 for other mutation operators which do not satisfy exponential tail bounds, like the so-called *fast mutation operators* [34].

Note that Theorem 12 only estimates the expected time to pass the borders $(1 + \varepsilon)\frac{\alpha}{\delta}$ and $(1 - \varepsilon)\frac{\alpha}{\delta}$, respectively. It does not guarantee that the diversity hits the interval $[1 - \varepsilon, 1 + \varepsilon]\frac{\alpha}{\delta}$. We define a stopping time for hitting this interval as follows.

Definition 15 Given a positive constant $\varepsilon > 0$ and an initial population P_t we define the first time T_ε when the diversity $S(P_t)$ is in the equilibrium as

$$T_\varepsilon := \inf \{ t \mid S(P_t) \in [(1 - \varepsilon)\frac{\alpha}{\delta}, (1 + \varepsilon)\frac{\alpha}{\delta}] \}.$$

In general, without restriction such as in Corollary 16, T_ε does not need to be finite and it is possible that the process never comes close to the equilibrium. The simplest (artificial) example is the following. Suppose $\mu = 2$ (so $\mu \in o(\sqrt{n})$), $\varepsilon = \frac{1}{3}$, and $\chi = n$ (i.e. every bit is flipped with probability 1), we omit crossover and the population initialises with two clones. Then we have $S(P_t) \in \{0, 2n\}$ for every t and

$$[1 - \varepsilon, 1 + \varepsilon]\frac{\alpha}{\delta} = \frac{4}{3}n[\frac{2}{3}, \frac{4}{3}] = [\frac{8}{9}n, \frac{16}{9}n].$$

Therefore $T_\varepsilon = \infty$, but $T_{\varepsilon,\uparrow} \leq 1$ and $T_{\varepsilon,\downarrow} \leq 1$.

The following corollary gives a sufficient condition for not skipping over the interval of states close to the equilibrium. The key is that the diversity can change at most by $2(\mu - 1)n$ in the setting of Theorem 14.

Corollary 16 If $\varepsilon\mu^2\chi \geq n + 2(\mu - 1)\chi$ (for example if $\mu \geq 3\sqrt{n/\varepsilon}, 0 < \varepsilon \leq 1$ and $\chi = 1$) then $T_\varepsilon = T_{\varepsilon,\downarrow}$ if $S(P_0) > (1 + \varepsilon)\frac{\alpha}{\delta}$ and $T_\varepsilon = T_{\varepsilon,\uparrow}$ if $S(P_0) < (1 - \varepsilon)\frac{\alpha}{\delta}$, respectively (see Theorem 12 for the definition of $T_{\varepsilon,\uparrow}$ and $T_{\varepsilon,\downarrow}$).

Proof Suppose that $S(P_0) > (1 + \varepsilon)\frac{\alpha}{\delta}$. Let $t := T_{\varepsilon, \downarrow} - 1$. Then we obtain $S(P_{t+1}) \leq (1 + \varepsilon)\frac{\alpha}{\delta}$. Moreover,

$$\begin{aligned} S(P_t) - (1 - \varepsilon)\frac{\alpha}{\delta} &> (1 + \varepsilon)\frac{\alpha}{\delta} - (1 - \varepsilon)\frac{\alpha}{\delta} = \frac{4\varepsilon\mu^2\chi \cdot (\mu - 1)n}{2n + 4(\mu - 1)\chi} \\ &\geq \frac{(4n + 8(\mu - 1)\chi) \cdot (\mu - 1)n}{2n + 4(\mu - 1)\chi} = 2(\mu - 1)n. \end{aligned}$$

Since $S(P_t) - S(P_{t+1}) \leq 2(\mu - 1)n$ by Lemma 13, we obtain $S(P_{t+1}) \in [1 - \varepsilon, 1 + \varepsilon]\frac{\alpha}{\delta}$.

For the other direction, suppose $S(P_0) < (1 - \varepsilon)\frac{\alpha}{\delta}$. Let $t := T_{\varepsilon, \uparrow} - 1$. Then we obtain $S(P_{t+1}) \geq (1 - \varepsilon)\frac{\alpha}{\delta}$ and

$$(1 + \varepsilon)\frac{\alpha}{\delta} - S(P_t) > (1 + \varepsilon)\frac{\alpha}{\delta} - (1 - \varepsilon)\frac{\alpha}{\delta} \geq 2(\mu - 1)n.$$

Hence, $S(P_{t+1}) \leq S(P_t) + 2(\mu - 1)n \leq (1 + \varepsilon)\frac{\alpha}{\delta}$ and thus $S(P_{t+1}) \in [1 - \varepsilon, 1 + \varepsilon]\frac{\alpha}{\delta}$, as required. □

5 Steady-State GA with Crossover

Now we turn to steady-state GAs that perform crossover before applying mutation to the resulting offspring (see Algorithm 2). Quite surprisingly, for nearly all common crossover operators, including crossover does not change the diversity equilibrium.

A sufficient condition is the following, which we term *diversity-neutral*, as the diversity equilibrium does not change when applying such a crossover operator.

Definition 17 We call a crossover operator c *diversity-neutral* if it has the following property. For all $x_1, x_2, z \in \{0, 1\}^n$,

$$E(H(c(x_1, x_2), z) + H(c(x_2, x_1), z)) = H(x_1, z) + H(x_2, z). \tag{19}$$

We shall see in Sect. 6 that common crossover operators like uniform crossover and k -point crossover are diversity-neutral.

We will show that the $(\mu+1)$ GA with any diversity-neutral crossover operator meets the conditions of Corollary 9. Hence, we obtain the following theorem.

Theorem 18 Consider the $(\mu+1)$ GA with any diversity-neutral crossover operator c , any unbiased mutation operator flipping χ bits in expectation and a population size of μ on a flat fitness function. Then for all populations P_t ,

$$E(S(P_{t+1}) \mid P_t) = (1 - \delta)S(P_t) + \alpha = \left(1 - \frac{2}{\mu^2} - \frac{4(\mu - 1)\chi}{\mu^2 n}\right)S(P_t) + 2\chi(\mu - 1), \tag{20}$$

where δ, α are as in (7).

Proof Let $y = c(x_i, x_j)$ and $y' = c(x_j, x_i)$ be the random result of crossover operations, where x_i and x_j are the randomly selected parents. We only need to show that $E(S(y)) = S(P_t)/\mu$, then the theorem follows from Corollary 9. By definition of diversity neutral crossover, we have for all $k \in [\mu]$,

$$E(H(y, x_k) + H(y', x_k) \mid x_i, x_j) = H(x_i, x_k) + H(x_j, x_k).$$

Summing over all k yields $S(y) + S(y')$ inside the expectation on the left hand side, and $S(x_i) + S(x_j)$ on the right hand side. Therefore, $E(S(y) + S(y') \mid P_t) = S(x_i) + S(x_j)$. Now we use that x_i and x_j are chosen uniformly at random. Hence,

$$E(S(y) + S(y') \mid P_t) = \frac{1}{\mu^2} \sum_{i=1}^{\mu} \sum_{j=1}^{\mu} (S(x_i) + S(x_j)) = \frac{2S(P_t)}{\mu}. \tag{21}$$

By the symmetric choice of the parents x_i and x_j , $E(S(y) \mid P_t) = E(S(y') \mid P_t)$, and thus $E(S(y) \mid P_t) = \frac{1}{2}(E(S(y) + S(y') \mid P_t)) = S(P_t)/\mu$. \square

Remark 19 Theorem 18 still holds if we choose the parents without replacement.

Proof We show that (21) still holds in this case. The rest of the proof carries over. In order to choose the parents without replacement, we can first take x_j uniformly at random and x_i can then be every individual except x_j . So we obtain

$$\begin{aligned} E(S(y) + S(y') \mid P_t) &= \frac{1}{(\mu - 1)\mu} \sum_{i=1}^{\mu} \sum_{j=1, j \neq i}^{\mu} (S(x_i) + S(x_j)) \\ &= \frac{S(P_t)}{\mu} + \frac{1}{(\mu - 1)\mu} \sum_{i=1}^{\mu} \sum_{j=1, j \neq i}^{\mu} S(x_j) \\ &= \frac{S(P_t)}{\mu} + \frac{(\mu - 1)S(P_t)}{(\mu - 1)\mu} = \frac{2S(P_t)}{\mu}. \end{aligned}$$

The third equality holds, because we sum up $S(x_j)$ exactly $(\mu - 1)$ times for every $j \in [\mu]$. So indeed (21) still holds. \square

We will see later that every diversity-neutral crossover is automatically respectful. Thus, all three parts of Theorem 12 hold also for a $(\mu + 1)$ GA with any diversity-neutral crossover operator c .

We assumed in Algorithms 1 and 2 that they break ties in favour of the offspring. In flat landscapes this means that the offspring is never discarded. We now transfer our results to variants in which the algorithm may also discard the offspring.

Remark 20 If the $(\mu + 1)$ GA does not favour the offspring over parents but instead breaks ties uniformly at random, then the conclusion of Theorem 18 still holds with (20) replaced by

$$E(S(P_{t+1}) \mid P_t) = (1 - \tilde{\delta})S(P_t) + \tilde{\alpha} \quad \text{with} \quad \tilde{\delta} := \frac{\mu}{\mu + 1}\delta, \quad \tilde{\alpha} := \frac{\mu}{\mu + 1}\alpha,$$

where δ, α are as in (7). In particular, the process has the same equilibrium state $\tilde{\alpha}/\tilde{\delta} = \alpha/\delta$ and all three parts of Theorem 12 still holds with $\tilde{\delta}$ and $\tilde{\alpha}$ instead of δ and α . Note that the bounds in Theorem 14 are increased precisely by a factor $(\mu + 1)/\mu$ since the additional factors in the logarithms cancel out. Since $(\mu + 1)/\mu = \Theta(1)$, Theorem 14 still holds unchanged.

Proof We fix some value for P_t and suppress conditioning on P_t in the following. Let A_t denote the event that the individual which we remove is not the offspring. Note that our results obtained so far always assumed A_t . By the law of the total probability,

$$\begin{aligned} E(S(P_{t+1})) &= P(A_t) \cdot E(S(P_{t+1}) \mid A_t) + P(\bar{A}_t) \cdot E(S(P_{t+1}) \mid \bar{A}_t) \\ &= \frac{\mu}{\mu + 1} E(S(P_{t+1})) + \frac{1}{\mu + 1} S(P_t). \end{aligned}$$

Therefore, by (20),

$$\begin{aligned} E(S(P_{t+1})) &= \frac{\mu}{\mu + 1} (1 - \delta) S(P_t) + \frac{\mu}{\mu + 1} \alpha + \frac{1}{\mu + 1} S(P_t) \\ &= \left(1 - \frac{\mu}{\mu + 1} \delta\right) S(P_t) + \frac{\mu}{\mu + 1} \alpha. \end{aligned}$$

□

6 Classifying Diversity-Neutral Crossover Operators

In this section we classify several known crossover operators into diversity/neutral ones and those that are not diversity/neutral.

6.1 Structural Results

We start with structural results that connect diversity/neutral with the properties unbiased, respectful and having order-independent mask (OIM), see Sect. 2. We will show that for unbiased crossover operators, diversity/neutral is equivalent to respectful. However, outside the class of unbiased operators, this is not true. While every diversity/neutral operator is still respectful, we show that the converse is false in general, but holds for the very large class of respectful operators with OIM.

Lemma 21 *Every diversity/neutral crossover operator is respectful.*

Proof Let x_1, x_2 be parents that both have a one-bit in position i . Let c be a diversity/neutral crossover operator and \mathcal{E} be the event that the offspring $c(x_1, x_2)$ has a zero-bit in position i . We will assume $\Pr(\mathcal{E}) > 0$ and derive a contradiction. The case that both parents have a zero-bit in position i is handled similarly. Suppose that the event \mathcal{E} appears. Let z_0 and z_1 be two search points which are identical in all positions

except for position i , where z_0 has a zero-bit and z_1 has a one-bit at position i . Then

$$H(x_1, z_0) = H(x_1, z_1) + 1 \quad \text{and} \quad H(x_2, z_0) = H(x_2, z_1) + 1. \quad (22)$$

Moreover, since z_0 and z_1 differ in exactly one position, $H(y, z_0) - H(y, z_1) \in \{-1, 1\}$ for all $y \in \{0, 1\}^n$. In particular, $H(y, z_0) - H(y, z_1) \leq 1$, and it is a strict inequality if and only if y has a zero-bit in position i . For $y = c(x_1, x_2)$, this implies

$$E(H(c(x_1, x_2), z_0) - H(c(x_1, x_2), z_1)) = \Pr(\mathcal{E}) \cdot (-1) + \Pr(\bar{\mathcal{E}}) \cdot 1 < 1, \quad (23)$$

where the inequality is strict because we have assumed $\Pr(\mathcal{E}) > 0$. For $y = c(x_2, x_1)$ we obtain

$$E(H(c(x_2, x_1), z_0) - H(c(x_2, x_1), z_1)) \leq 1, \quad (24)$$

where this time we cannot claim a strict inequality since we have not made any assumption on $c(x_2, x_1)$. Adding up (23) and (24), we obtain

$$E(H(c(x_1, x_2), z_0) + H(c(x_2, x_1), z_0) - H(c(x_1, x_2), z_1) - H(c(x_2, x_1), z_1)) < 2.$$

But since c is diversity-neutral, the left hand side equals

$$H(x_1, z_0) + H(x_2, z_0) - H(x_1, z_1) - H(x_2, z_1) \stackrel{(22)}{=} 2,$$

a contradiction. Hence, the assumption $\Pr(\mathcal{E}) > 0$ must have been false, and therefore the offspring of x_1 and x_2 must have a one-bit in position i with probability 1. This concludes the proof. \square

Next, we show that the converse is not true.

Lemma 22 *Not every respectful crossover is diversity/neutral.*

Proof For $x_1, x_2 \in \{0, 1\}^n$ we define $c(x_1, x_2)$ as the bit-wise AND of x_1 and x_2 . The operator is respectful since $1 \text{ AND } 1$ is 1 and $0 \text{ AND } 0$ is 0 .

Now for any two search points $x_1, x_2 \in \{0, 1\}^n$ with $x_2 = \bar{x}_1$ and $z = \vec{0}$, we have $H(x_1, z) + H(x_2, z) = n$ as every bit is set to 1 in exactly one parent. However, $c(x_1, x_2) = c(x_2, x_1) = \vec{0}$ and so

$$E(H(c(x_1, x_2), z) + H(c(x_2, x_1), z)) = 0 \neq H(x_1, z) + H(x_2, z).$$

So this crossover is not diversity/neutral. \square

The counterexample from Lemma 22 has a strong bias towards setting bits to 0. It is thus not unbiased. Recall from Definition 4 that a respectful crossover operator has an order-independent mask (OIM) if the probability distribution for choosing masks does not depend on the order of parents. Now we show that adding OIM gives a sufficient condition to be diversity/neutral. Note that this implies that the AND operator used in the proof of Lemma 22 does not have OIM.

Lemma 23 *All respectful crossovers with OIM are diversity/neutral.*

Proof We show for all $x_1, x_2, z \in \{0, 1\}^n$ and for each bit i that

$$E(|c(x_1, x_2)_i - z_i| + |c(x_2, x_1)_i - z_i|) = |(x_1)_i - z_i| + |(x_2)_i - z_i|.$$

Taking the sum over all $i \in [n]$ turns all absolute differences of bits $|a_i - b_i|$ in the above expression into Hamming distances $H(a, b)$, yielding (19). If $(x_1)_i = (x_2)_i$ then the equation is immediate since the left hand side simplifies to $E(|(x_1)_i - z_i| + |(x_2)_i - z_i|)$ (since c is respectful) and the expression is deterministic.

If $(x_1)_i = 1 - (x_2)_i$ then c with OIM implies

$$\Pr(c(x_1, x_2)_i = (x_1)_i) = \Pr(c(x_2, x_1)_i = (x_2)_i) =: p.$$

With probability $q := 1 - p$, $c(x_1, x_2)_i = 1 - (x_1)_i = (x_2)_i$ and $c(x_2, x_1)_i = 1 - (x_2)_i = (x_1)_i$, respectively. Together,

$$\begin{aligned} E(|c(x_1, x_2)_i - z_i| + |c(x_2, x_1)_i - z_i|) &= |(x_1)_i - z_i|p + |(x_2)_i - z_i|q + |(x_2)_i - z_i|p + |(x_1)_i - z_i|q \\ &= |(x_1)_i - z_i| + |(x_2)_i - z_i|. \end{aligned}$$

□

Recall that diversity/neutral operators are respectful by Lemma 21. Hence, the following lemma shows that the converse of Lemma 23 is true for *unbiased* crossover operators. In other words, within the class of unbiased binary operators, the properties diversity/neutral and respectful are equivalent. Outside of this class, Lemma 22 shows that the terms are not equivalent.

Lemma 24 *Every respectful unbiased crossover has an OIM.*

Proof Let x_1, x_2 be parents for a respectful, unbiased crossover operator c with a corresponding probability distribution $D(y | x_1, x_2)$ where the condition is meant to be understood that x_1 is the first parent and x_2 is the second parent. Let I_{diff} be the set of components of x_1, x_2 which differ, i.e. $I_{\text{diff}} := \{i \in \{1, \dots, n\} \mid (x_1)_i \neq (x_2)_i\}$. Let I_{eq} be the set of components of x_1, x_2 which are equal, i.e. $I_{\text{eq}} := \{1, \dots, n\} \setminus I_{\text{diff}}$.

We show that c can be described as a respectful crossover with a mask created according to a probability distribution $M(a, x_1, x_2)$ which is order-independent. For bits $i \in I_{\text{eq}}$ the mask is irrelevant since c is respectful, and we (arbitrarily) define $a_i := 1$. For $y \in \{0, 1\}^n$ with $D(y | x_1, x_2) > 0$ choose a mask $a = (a_1, \dots, a_n) \in \{1, 2\}^n$ with probability $D(y | x_1, x_2)$ in the following way. For bits $i \in I_{\text{diff}}$ we choose a_i as the unique value from $\{1, 2\}$ such that $(x_{a_i})_i = y_i$. This is possible since $i \in I_{\text{diff}}$ implies $\{(x_1)_i, (x_2)_i\} = \{0, 1\}$. Applying the mask to x_1 and x_2 creates y . Since the corresponding mask is chosen with probability $D(y | x_1, x_2)$, each y is created with probability $D(y | x_1, x_2)$. Hence c is respectful.

It is left to show that the choice of the mask does not depend on the order of the parents for crossover. Define $w \in \{0, 1\}^n$ as $w_i = 0$ if $i \in I_{\text{eq}}$ and $w_i = 1$ otherwise. Then we obtain $x_1 \oplus w = x_2$ and $x_2 \oplus w = x_1$. Since c is unbiased we have

$$D(y \mid x_1, x_2) = D(y \oplus w \mid x_1 \oplus w, x_2 \oplus w) = D(y \oplus w \mid x_2, x_1).$$

So it is left to show the following. Let $a \in \{1, 2\}^n$. If we obtain $y \in \{0, 1\}^n$ with the mask a applied to (x_1, x_2) then we obtain $y \oplus w$ with the same mask a applied to (x_2, x_1) . Let $i \in \{1, \dots, n\}$.

If $i \in I_{\text{eq}}$ then applying the mask a to (x_1, x_2) gives $y_i = (x_1)_i$. Note that $(y \oplus w)_i = y_i = (x_1)_i = (x_2)_i$ which is also the i -th component of the offspring if we apply the mask a to (x_2, x_1) .

If $i \in I_{\text{diff}}$ then applying the mask a to (x_1, x_2) gives $y_i = (x_{a_i})_i$. If we apply a to (x_2, x_1) we obtain $1 - (x_{a_i})_i$ for the i -th bit of the offspring, which equals $(y \oplus w)_i$ (since $(x_1)_i$ and $(x_2)_i$ differ). □

By Lemma 23, every crossover with OIM is diversity/neutral. The next lemma shows that the converse is not true. Hence, the class of diversity/neutral crossovers is strictly larger than the class of crossovers with OIM.

Remark 25 Not every diversity/neutral crossover has an OIM.

Proof Let c be the crossover operator on (x_1, x_2) which returns a uniform random bit-string if $(x_1, x_2) = (\vec{1}, \vec{0})$, and in all other cases it returns either x_1 or x_2 with probability $1/2$ each. In particular, the latter case also applies for $(x_1, x_2) = (\vec{0}, \vec{1})$. Then in either case, for all $x_1, x_2, z \in \{0, 1\}^n$,

$$E(H(c(x_1, x_2), z)) = \frac{1}{2}(H(x_1, z) + H(x_2, z)).$$

For the case $(x_1, x_2) = (\vec{1}, \vec{0})$, this follows since both the left hand side and the right hand side are $n/2$ for all $z \in \{0, 1\}^n$, while for all other cases it is obvious. Hence, (19) is satisfied and c is diversity/neutral. On the other hand, it does not have an OIM, since for $(x_1, x_2) = (\vec{1}, \vec{0})$ the operator has a positive chance to use the mask that takes the first half of the result from x_1 and the second half from x_2 , but for $(x_1, x_2) = (\vec{0}, \vec{1})$ it never uses this mask. □

6.2 Classifying Known Crossover Operators

We now give examples of diversity/neutral crossover operators, based on [35]. By Lemma 23 it suffices to show that a crossover is respectful with OIM. For uniform crossover and k -point crossover, this is trivially true as they are based on masks that are chosen independently from the parents. The same holds for the boring crossover (recall that it simply returns one of the parents uniformly at random) as the mask is chosen uniformly from $\{\vec{1}, \vec{2}\}$.

Shrinking crossover [62] computes a mask by starting with a window $[\ell, r] = [1, n]$ and then shrinking this window by increasing ℓ and/or decreasing r until the sub-string

$x_1[\ell, r]$ has the same number of ones as $x_2[\ell, r]$. Then it swaps these two sub-strings. The creation of the mask treats both parents symmetrically.

Balanced uniform crossover [35] is respectful as it copies bit values on which both parents agree. If the parents differ in k positions, it chooses values for these bits uniformly at random from all sub-strings that have exactly $\lfloor k/2 \rfloor$ ones at these positions. The order of parents is irrelevant, hence the crossover has OIM.

Hence, we have shown the following theorem.

Theorem 26 *The following crossovers are diversity/neutral:*

1. *Uniform crossover with arbitrary crossover bias*
2. *k -point crossover for all k*
3. *Boring crossover*
4. *Shrinking crossover*
5. *Balanced uniform crossover*

We mention some crossover operators that are not diversity neutral. For details we refer to [35] and the original papers.

Alternating crossover [63] on x_1 and x_2 proceeds as follows. If x_1 has ones at positions i_1, \dots, i_k and x_2 has ones at positions $j_1, \dots, j_{k'}$, then for $k^* := \min\{k, k'\}$ alternating crossover produces a sorted sequence s_1, \dots, s_{2k^*} of these positions. It outputs a search point that has ones at positions $s_1, s_3, s_5, \dots, s_{2k^*-1}$.

Counter-based crossover [64] is a variant of uniform crossover ensuring that the offspring has the same number of ones as x_1 . It creates an offspring bit by bit, choosing values from x_1 and x_2 uniformly at random, but stopping once the offspring contains $|x_1|_1$ ones or $|x_1|_0$ zeros. In this case a suffix of all-zeros or all-ones, resp., is appended to obtain a bit string of length n with $|x_1|_1$ ones.

Zero length crossover [64] uses a different representation: a search point x with $|x|_1 = k$ and $x = 0^{a_1} 1 0^{a_2} 1 \dots 0^{a_k} 1 0^{a_{k+1}}$ is encoded as a vector of runs of zeros: $[a_1, a_2, \dots, a_{k+1}]$. The crossover operator combines encodings from both parents by choosing run lengths in between the run lengths found in both parents.

Map-of-ones-crossover [64] uses an array that contains all indices of 1-bits to represent a bit string. The crossover operator then chooses indices from a randomly chosen parent. In a sense, map-of-ones crossover is a uniform crossover on the map-of-ones representation.

Balanced two-point crossover [63] resembles a two-point crossover on the same representation. It randomly generates two cutting points $u \leq v$ and then it takes the first $u - 1$ entries of the map-of-ones of x_1 , the entries at positions $u \dots v$ from the map-of-ones of x_2 and the remaining entries from position $v + 1$ from x_1 again. Any duplicate entries are removed and replaced by entries from the positions $u \dots v$ in the map-of-ones of x_1 .

Theorem 27 *The following crossovers are not diversity/neutral:*

1. *Alternating crossover*
2. *Counter-based crossover*
3. *Zero length crossover*
4. *Map-of-ones crossover*

5. *Balanced two-point crossover*
6. *Bit-wise AND and bit-wise OR*

Proof An alternating crossover of 110 and 101 creates a sorted sequence of indices $[1, 1, 2, 3]$ and the offspring 110, irrespective of the order of the parents. For $z := 110$, the left-hand side of (19) is $E(H(c(110, 101), 110) + H(c(101, 110), 110)) = E(2 H(110, 110)) = 0$ and the right-hand side is $H(110, 110) + H(101, 110) = 2 \neq 0$.

Crossovers (2)–(5) were shown not to be respectful in [35], thus by the contraposition of Lemma 21 they are not diversity/neutral. Bit-wise AND was shown not to be diversity/neutral in the proof of Lemma 22. Bit-wise OR can be treated analogously. \square

7 Conclusions and Future Work

We have shown that it is possible to understand the dynamics of population diversity in flat fitness environments in a very general sense, and that it is surprisingly unaffected by most specifics of the algorithm. We particularised our results on tail-bounded mutation operators which are very common in practice. For example, standard bitwise mutation and local search belong to this class.

Of course, our study is only the first step. Possible extensions would include other classes of algorithms like generational GAs or the effect of diversity-enhancing mechanisms [1] on the dynamics, in particular on the equilibrium state. Note that it is not clear a priori that such a state exists, since the dynamics might be too complex to reduce them to a single number. Future work could also try to establish connections with population genetics, where the $(\mu+1)$ EA is known as Moran model [42] (cf. the discussion at the end of Sect. 1.1).

The most pressing question is how the dynamics change with selective pressure. We conjectured that for “reasonable” situations, the diversity for flat fitness functions is an upper bound on the diversity for non-flat functions. Can this be made precise? For which non-flat fitness functions can we still characterise how the population diversity evolves over time? These questions have important theoretical and practical implications, yet they are wide open.

Acknowledgements This work originated at Dagstuhl seminar 22081 “Theory of Randomized Optimization Heuristics” and benefited from Dagstuhl Seminar 22182 “Estimation-of-Distribution Algorithms: Theory and Applications”. We thank Jon Rowe, Duc-Cuong Dang and Tiago Paixão for useful discussions and pointers to the literature, and we thank an anonymous GECCO reviewer for pointing out an algebra mistake in the first version of this manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL.

Declarations

Conflict of interest The authors have no conflict of interest to declare that are relevant to the content of this article.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Sudholt, D.: The benefits of population diversity in evolutionary algorithms: a survey of rigorous runtime analyses. In: *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization*, pp. 359–404. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-29414-4_8
2. Squillero, G., Tonda, A.: Divergence of character and premature convergence: a survey of methodologies for promoting diversity in evolutionary optimization. *Inf. Sci.* **329**, 782–799 (2016). <https://doi.org/10.1016/j.ins.2015.09.056>
3. Glibovets, M., Gulayeva, N.: A review of niching genetic algorithms for multimodal function optimization. *Cybern. Syst. Anal.* **49**(6), 815–820 (2013). <https://doi.org/10.1007/s10559-013-9570-8>
4. Črepinšek, M., Liu, S.-H., Mernik, M.: Exploration and exploitation in evolutionary algorithms: a survey. *ACM Comput. Surv.* **45**(3), 1–33 (2013). <https://doi.org/10.1145/2480741.2480752>
5. Shir, O.M.: Niching in Evolutionary Algorithms. In: *Handbook of Natural Computing*, pp. 1035–1069. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-540-92910-9_32
6. Friedrich, T., Oliveto, P.S., Sudholt, D., Witt, C.: Analysis of diversity-preserving mechanisms for global exploration. *Evol. Comput.* **17**(4), 455–476 (2009). <https://doi.org/10.1162/evco.2009.17.4.17401>
7. Oliveto, P.S., Sudholt, D., Zarges, C.: On the benefits and risks of using fitness sharing for multimodal optimisation. *Theor. Comput. Sci.* **773**(C), 53–70 (2019). <https://doi.org/10.1016/j.tcs.2018.07.007>
8. Osuna, E.C., Sudholt, D.: On the runtime analysis of the clearing diversity-preserving mechanism. *Evol. Comput.* **27**(3), 403–433 (2019). https://doi.org/10.1162/evco_a_00225
9. Covantes Osuna, E., Sudholt, D.: Runtime analysis of crowding mechanisms for multimodal optimisation. *IEEE Trans. Evol. Comput.* **24**(3), 581–592 (2020). <https://doi.org/10.1109/TEVC.2019.2914606>
10. Osuna, E.C., Sudholt, D.: Runtime analysis of restricted tournament selection for bimodal optimisation. *Evol. Comput.* **30**(1), 1–26 (2022). https://doi.org/10.1162/evco_a_00292
11. Oliveto, P.S., Zarges, C.: Analysis of diversity mechanisms for optimisation in dynamic environments with low frequencies of change. *Theor. Comput. Sci.* **561**, 37–56 (2015). <https://doi.org/10.1016/j.tcs.2014.10.028>
12. Lissovoi, A., Witt, C.: A runtime analysis of parallel evolutionary algorithms in dynamic optimization. *Algorithmica* **78**(2), 641–659 (2017). <https://doi.org/10.1007/s00453-016-0262-4>
13. Helmuth, T., Lengler, J., Cava, W.L.: Population diversity leads to short running times of lexicae selection. In: *Parallel Problem Solving from Nature (PPSN 2022)*, vol. 13399, pp. 485–498. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-14721-0_34
14. Sudholt, D.: How crossover speeds up building-block assembly in genetic algorithms. *Evol. Comput.* **25**(2), 237–274 (2017). https://doi.org/10.1162/EVCO_a_00171
15. Corus, D., Oliveto, P.S.: Standard steady state genetic algorithms can hillclimb faster than mutation-only evolutionary algorithms. *IEEE Trans. Evol. Comput.* **22**(5), 720–732 (2018). <https://doi.org/10.1109/TEVC.2017.2745715>
16. Corus, D., Oliveto, P.S.: On the benefits of populations for the exploitation speed of standard steady-state genetic algorithms. *Algorithmica* **82**(12), 3676–3706 (2020). <https://doi.org/10.1007/s00453-020-00743-1>
17. Lengler, J.: A general dichotomy of evolutionary algorithms on monotone functions. *IEEE Trans. Evol. Comput.* **24**(6), 995–1009 (2019). <https://doi.org/10.1109/TEVC.2019.2917014>
18. Nguyen, P.T.H., Sudholt, D.: Memetic algorithms outperform evolutionary algorithms in multimodal optimisation. *Artif. Intell.* **287**(1), 103345 (2020). <https://doi.org/10.1016/j.artint.2020.103345>
19. Jansen, T., Wegener, I.: Real royal road functions—where crossover probably is essential. *Discrete Appl. Math.* **149**(1–3), 111–125 (2005). <https://doi.org/10.1016/j.dam.2004.02.019>

20. Jansen, T., Wegener, I.: On the analysis of evolutionary algorithms—a proof that crossover really can help. *Algorithmica* **34**(1), 47–66 (2002). <https://doi.org/10.1007/s00453-002-0940-2>
21. Kötzing, T., Sudholt, D., Theile, M.: How crossover helps in pseudo-Boolean optimization. In: Proceedings of the 13th Annual Genetic and Evolutionary Computation Conference (GECCO 2011), pp. 989–996. ACM, New York, NY, USA (2011). <https://doi.org/10.1145/2001576.2001711>
22. Dang, D.-C., Friedrich, T., Kötzing, T., Krejca, M.S., Lehre, P.K., Oliveto, P.S., Sudholt, D., Sutton, A.M.: Escaping local optima using crossover with emergent diversity. *IEEE Trans. Evol. Comput.* **22**(3), 484–497 (2018). <https://doi.org/10.1109/TEVC.2017.2724201>
23. Dang, D.-C., Friedrich, T., Krejca, M.S., Kötzing, T., Lehre, P.K., Oliveto, P.S., Sudholt, D., Sutton, A.M.: Escaping local optima with diversity-mechanisms and crossover. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2016), pp. 645–652. ACM, Denver, USA (2016). <https://doi.org/10.1145/2908812.2908956>
24. Oliveto, P.S., Sudholt, D., Witt, C.: Tight bounds on the expected runtime of a standard steady state genetic algorithm. *Algorithmica* **84**(6), 1603–1658 (2022). <https://doi.org/10.1007/s00453-021-00893-w>
25. Cathabard, S., Lehre, P.K., Yao, X.: Non-uniform mutation rates for problems with unknown solution lengths. In: Proceedings of the 11th Workshop on Foundations of Genetic Algorithms (FOGA 2011), pp. 173–180. ACM, New York, NY, USA (2011). <https://doi.org/10.1145/1967654.1967670>
26. Doerr, B., Doerr, C., Kötzing, T.: Unknown solution length problems with no asymptotically optimal runtime. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2017), pp. 1367–1374. ACM, Berlin, Germany (2017). <https://doi.org/10.1145/3071178.3071233>
27. Doerr, B., Doerr, C., Kötzing, T.: Solving problems with unknown solution length at almost no extra cost. *Algorithmica* **81**(2), 703–748 (2019). <https://doi.org/10.1007/s00453-018-0477-7>
28. Einarsson, H., Gaury, M.M., Lengler, J., Meier, F., Mujika, A., Steger, A., Weissenberger, F.: The linear hidden subset problem for the $(1 + 1)$ -EA with scheduled and adaptive mutation rates. *Theor. Comput. Sci.* **785**(2), 150–170 (2019). <https://doi.org/10.1016/j.tcs.2019.05.021>
29. Bian, C., Qian, C., Tang, K., Yu, Y.: Running time analysis of the $(1+1)$ -EA for robust linear optimization. *Theor. Comput. Sci.* **843**, 57–72 (2020). <https://doi.org/10.1016/j.tcs.2020.07.001>
30. Doerr, C., Krejca, M.S.: Run time analysis for random local search on generalized majority functions. *IEEE Trans. Evol. Comput.* **27**(5), 1385–1397 (2022). <https://doi.org/10.1109/TEVC.2022.3216349>
31. Dang, D., Jansen, T., Lehre, P.K.: Populations can be essential in tracking dynamic optima. *Algorithmica* **78**(2), 660–680 (2017). <https://doi.org/10.1007/s00453-016-0187-y>
32. Antipov, D., Doerr, B.: Precise runtime analysis for plateau functions. *ACM Trans. Evol. Learn. Optim.* **1**(4), 1–28 (2021). <https://doi.org/10.1145/3469800>
33. Eremeev, A.V.: On non-elitist evolutionary algorithms optimizing fitness functions with a plateau. In: *Mathematical Optimization Theory and Operations Research—19th International Conference, MOTOR 2020*, pp. 329–342. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49988-4_23
34. Doerr, B., Le, H.P., Phuoc, H., Makhmara, R., Nguyen, T.D.: Fast genetic algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2017), pp. 777–784. ACM, New York, NY, USA (2017). <https://doi.org/10.1145/3071178.3071301>
35. Friedrich, T., Kötzing, T., Radhakrishnan, A., Schiller, L., Schirneck, M., Tennigkeit, G., Wietheger, S.: Crossover for cardinality constrained optimization. *ACM Trans. Evol. Learn. Optim.* **3**(2), 1–32 (2023). <https://doi.org/10.1145/3603629>
36. Lengler, J., Opris, A., Sudholt, D.: Analysing equilibrium states for population diversity. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2023). ACM, New York, NY, USA (2023). <https://dl.acm.org/doi/abs/10.1145/3583131.3590465>
37. Lengler, J., Zou, X.: Exponential slowdown for larger populations: the $(\mu + 1)$ -EA on monotone functions. *Theor. Comput. Sci.* **875**(1), 28–51 (2021). <https://doi.org/10.1016/j.tcs.2021.03.025>
38. Lehre, P.K.: Negative drift in populations. In: 11th International Conference on Parallel Problem Solving from Nature (PPSN 2010), vol. 6238, pp. 244–253. Springer, Berlin, Heidelberg (2011). https://doi.org/10.1007/978-3-642-15844-5_25
39. Oliveto, P.S., Sudholt, D.: On the runtime analysis of stochastic ageing mechanisms. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2014), pp. 113–120. ACM, New York, NY, USA (2014). <https://doi.org/10.1145/2576768.2598328>
40. Kingman, J.F.C.: *Mathematics of Genetic Diversity*. CBMS-NSF Regional Conference Series in Applied Mathematics. Society for Industrial and Applied Mathematics, Philadelphia (1980). <https://doi.org/10.1137/1.9781611970357>

41. Kimura, M.: The neutral theory of molecular evolution. *Sci. Am.* **241**(5), 98–129 (1979)
42. Paixão, T., Badkobeh, G., Barton, N., Corus, D., Dang, D.-C., Friedrich, T., Lehre, P.K., Sudholt, D., Sutton, A.M., Trubenová, B.: A unified model of evolutionary processes. *J. Theor. Biol.* **383**, 28–43 (2015). <https://doi.org/10.1016/j.jtbi.2015.07.011>
43. Wineberg, M., Oppacher, F.: The underlying similarity of diversity measures used in evolutionary computation. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2003), pp. 1493–1504. Springer, Berlin, Heidelberg (2003). https://doi.org/10.1007/3-540-45110-2_21
44. Kimura, M.: Genetic variability maintained in a finite population due to mutational production of neutral and nearly neutral isoalleles. *Genetics Res.* **11**(3), 247–270 (1968). <https://doi.org/10.1017/S0016672300011459>
45. Chatterjee, K., Pavlogiannis, A., Adlam, B., Nowak, M.A.: The time scale of evolutionary innovation. *PLoS Comput. Biol.* **10**(9), 1003818 (2014). <https://doi.org/10.1371/journal.pcbi.1003818>
46. Pérez Heredia, J., Trubenová, B., Sudholt, D., Paixão, T.: Selection limits to adaptive walks on correlated landscapes. *Genetics* **205**(2), 803–825 (2017). <https://doi.org/10.1534/genetics.116.189340>
47. Trubenová, B., Krejca, M.S., Lehre, P.K., Kötzing, T.: Surfing on the seascape: adaptation in a changing environment. *Evolution* **73**(7), 1356–1374 (2019). <https://doi.org/10.1111/evo.13784>
48. Griffiths, R.C.: Neutral two-locus multiple allele models with recombination. *Theor. Popul. Biol.* **19**(2), 169–186 (1981). [https://doi.org/10.1016/0040-5809\(81\)90016-2](https://doi.org/10.1016/0040-5809(81)90016-2)
49. Aulbach, B., Haderl, K.P.: Convergence to equilibrium in the classical model of population genetics. *J. Math. Anal. Appl.* **102**(1), 220–232 (1984). [https://doi.org/10.1016/0022-247X\(84\)90215-4](https://doi.org/10.1016/0022-247X(84)90215-4)
50. Lehre, P.K., Witt, C.: Black-box search by unbiased variation. *Algorithmica* **64**(4), 623–642 (2012)
51. Radcliffe, N.J.: The algebra of genetic algorithms. *Ann. Math. Artif. Intell.* **10**(4), 339–384 (1994). <https://doi.org/10.1007/BF01531276>
52. Moraglio, A.: Towards a geometric unification of evolutionary algorithms. PhD thesis, University of Essex (2007)
53. Ulrich, T.: Exploring structural diversity in evolutionary algorithms. Doctoral thesis, ETH Zurich, Zürich (2012). <https://doi.org/10.3929/ethz-a-007562769>
54. Solow, A.R., Polasky, S.: Measuring biological diversity. *Environ. Ecol. Stat.* **1**(2), 95–103 (1994). <https://doi.org/10.1007/BF02426650>
55. Motwani, R., Raghavan, P.: Randomized Algorithms. Cambridge University Press, Cambridge (1995). <https://doi.org/10.1017/CBO9780511814075>
56. Levin, D.A., Peres, Y., Wilmer, E.L.: Markov Chains and Mixing Times. American Mathematical Society, Providence, RI (2008). <https://doi.org/10.1090/mbk/058>
57. Doerr, B.: Lower bounds for non-elitist evolutionary algorithms via negative multiplicative drift. *Evol. Comput.* **29**(2), 305–329 (2021). https://doi.org/10.1162/evco_a_00283
58. Newman, M.E.J.: Power laws, Pareto distributions and Zipf’s law. *Contemp. Phys.* **46**(5), 323–351 (2005). <https://doi.org/10.1080/00107510500052444>
59. Doerr, B., Johannsen, D., Winzen, C.: Multiplicative drift analysis. *Algorithmica* **64**(4), 673–697 (2012). <https://doi.org/10.1007/s00453-012-9622-x>
60. Doerr, B., Goldberg, L.A.: Adaptive drift analysis. *Algorithmica* **65**(1), 224–250 (2013). <https://doi.org/10.1007/s00453-011-9585-3>
61. Buzdalov, M., Doerr, B., Doerr, C., Vinokurov, D.: Fixed-target runtime analysis. *Algorithmica* **84**(6), 1762–1793 (2022). <https://doi.org/10.1007/s00453-021-00881-0>
62. Chen, J.-S., Hou, J.-L.: A combination genetic algorithm with applications on portfolio optimization. In: Advances in Applied Artificial Intelligence, pp. 197–206. Springer, Berlin, Heidelberg (2006). https://doi.org/10.1007/11779568_23
63. Meinel, T., Berthold, M.R.: Crossover operators for multiobjective k -subset selection. In: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation (GECCO 2009), pp. 1809–1810. ACM, New York, NY, USA (2009). <https://doi.org/10.1145/1569901.1570173>
64. Manzoni, L., Mariot, L., Tuba, E.: Balanced crossover operators in genetic algorithms. *Swarm Evol. Comput.* **54**, 100646 (2020). <https://doi.org/10.1016/j.swevo.2020.100646>