On the Benefits and Risks of Using Fitness Sharing for Multimodal Optimisation[☆]

Pietro S. Oliveto[a], Dirk Sudholt[a], Christine Zarges[b,*]

[a] *Department of Computer Science, The University of Sheffield, Sheffield S1 4DP, UK*
[b] *Department of Computer Science, Aberystwyth University, Aberystwyth SY23 3DB, UK*

Abstract

Fitness sharing is a well-known diversity mechanism inspired by the idea that individuals in the population that are close to each other have to share their fitnesses in a similar way to how species in nature occupying the same ecological environment have to share resources. Thus, by derating the fitness of close individuals one hopes to encourage the population to spread out more.

Previous runtime analyses of fitness sharing studied a variant where selection was based on populations instead of individuals. We study the conventional fitness sharing mechanism based on individuals and use runtime analysis to highlight its benefits and dangers on the well-known bimodal test problem TwoMax, where diversity is crucial for finding both optima. In contrast to population-based sharing, a (2+1) evolutionary algorithm (EA) with conventional fitness sharing does not guarantee to find both optima in polynomial time even when problem specific knowledge is used to estimate the distance between individuals; however, a (μ+1) EA with $\mu \geq 3$ always succeeds in expected polynomial time. We further show theoretically and empirically that large offspring populations in (μ+λ) EAs can be detrimental as creating too many offspring in one particular area of the search space can make all individuals in this area go extinct. We conclude the paper with an empirical study indicating that similar conclusions may be drawn when

[☆] An extended abstract of this work with parts of the results and without most of the proofs has been presented at *Parallel Problem Solving from Nature (PPSN)* in 2014 [1].

[*] Corresponding author.

Email addresses: p.oliveto@sheffield.ac.uk (Pietro S. Oliveto), d.sudholt@sheffield.ac.uk (Dirk Sudholt), c.zarges@aber.ac.uk (Christine Zarges)

using the genotypic distance that has to be relied upon when no problem specific knowledge is available.

Keywords: Evolutionary computation, diversity mechanisms, fitness sharing, multimodal optimisation, runtime analysis

1. Introduction

Many real-world optimisation problems are multimodal by nature, i. e., they have a number of different local optima and may have more than one global optimum. Nature-inspired techniques have proven to be very popular and powerful to tackle these types of problems [2] and different optimisation goals have been discussed in the literature [3]. Taking a global perspective, one is for example interested in locating a single (local or global) optimum. However, in practice it is often more important to identify a multitude of different optima, either in a simultaneous or sequential fashion. Our analyses therefore concentrates on the multi-local aspect of multimodal optimisation, i. e., where the goal is that the set of local optima is contained in the population by the end of the run.

In evolutionary computation, diversity mechanisms are commonly used to tackle multimodal optimisation problems [4, 5, 6], particularly in the context of a multi-local perspective. The main idea is to try and introduce *niches* in the population to prevent the algorithm from converging to a single solution such that different niches explore different peaks of the fitness landscape. Thus, in this context, niches are often understood as narrow, connected areas of the search space. In contrast to the numerous and widespread applications, the amount of theoretical research rigorously proving the effectiveness of diversity mechanisms is limited. Nevertheless, some previous theoretical work on diversity mechanisms for multimodal optimisation using nature-inspired techniques exists. Most notably, runtime analyses are available where the performance of diversity mechanisms is evaluated in terms of their optimisation time (i. e., the number of fitness function evaluations required to find the global optimum or a set of optima).

Friedrich et al. [7] showed that diversity mechanisms may be necessary by analysing population-based evolutionary algorithms (EAs) for a bimodal function called TwoMax. TwoMax is a function of unitation, that is, the fitness only depends on the number of 1-bits $|x|$ in the considered search point x. Hence, while the bit string x is the *genotype*, the number of ones

$|x|$ may be considered the corresponding *phenotype*, from which the fitness $\max\{|x|, n-|x|\}$ is then derived (where n is the length of the bit string). The function is easy from the perspective of localising the two local optima in a sequential fashion, for example, by using local search coupled with a restart strategy. On the other hand, the function is very challenging from a multi-local perspective, since the two local optima are as far away as possible from each other. Friedrich et al. [7] proved that a population-based EA with realistic population size (i. e., at most sublinear in the problem size) would fail at locating efficiently both optima of TwoMax if no diversity mechanism is used. They then showed that some diversity mechanisms make the algorithm efficient (i. e., *fitness sharing*, which we consider here in more detail, and *deterministic crowding*, where offspring compete for survival only with their parents) while others do not (i. e., *avoiding genotype duplicates* and *avoiding duplicates of equal fitness*). Recently, it has been proven that also the *clearing mechanism*, where resources are only assigned to the best individual of a subpopulation, makes population-based EAs efficient for the TwoMax function [8].

Diversity mechanisms have also been shown to enhance the capabilities of the recombination operator by favouring the emergence of dissimilar individuals. Fischer and Wegener [9] were the first to rigorously study this effect by analysing the performance of a genetic algorithm (GA) using fitness sharing on colouring problems inspired by the Ising model. They showed that the diversity mechanism helps the exploration of large plateaus and proved a speedup of order n over the simple (1+1) EA on one-dimensional Ising models with n nodes. Sudholt [10] further proved an exponential speedup for a GA for the Ising model on trees, using the fact that fitness sharing is powerful enough to allow the algorithm to tunnel through shallow fitness valleys. Recently, Dang et al. [11] showed that several diversity mechanisms allow an exponential speedup in the time required to escape from the local optima via recombination for the standard Jump_k benchmark function. Diversity mechanisms can also enhance the capabilities of the recombination operator for hillclimbing the OneMax function. A (2+1) GA with genotype diversity optimises the function in half the expected time (i.e., $(e/2+o(1))n\log n$) required by EAs only using standard bit mutation with fixed mutation rate [12], while GAs without diversity have been experimentally shown to be slower and the best upper bound on the expected runtime known is $(3/4+o(1))en\log n$ [13]. It is also worth noting that an analysis for the Balance benchmark function has shown that diversity mechanisms allow the efficient optimisation of deceptive functions in the context of dynamic optimisation [14].

Fitness sharing [15, 16] is amongst the best known diversity mechanisms. It is featured in many surveys [4, 5, 6] and used in many practical applications (see, e. g. [17, 18, 19, 20, 21], to name just a few). In this scheme niche formation is induced by using a *sharing function* that derates the fitness of an individual by an amount related to its '*distance*' to the rest of the population. It is inspired by the idea that individuals in the same niche of the search space, i. e., individuals that are close to each other, have to share resources (their fitness)—similarly to how species in nature occupying the same ecological environment have to share resources. As a result individuals are encouraged to increase their distance from other individuals, thus spread out in the search space.

The effectiveness of fitness sharing may vary considerably according to which measure is used to define the distance between individuals. In particular, different fitness sharing functions are obtained according to how the distance between individuals is defined [4]. Fitness sharing can use distances defined on a genotypic or phenotypic level [4, 5, 6, 16, 22]. *Genotypic sharing* [4] uses genotypic distances like the Hamming distance to measure how close individuals of the population are to each other. *Phenotypic sharing* refers to distances in the decoded parameter space [22], which in turn depends on the encoding used. For example, if the genotype encodes a vector of real values, the phenotypic distance is commonly defined as the Euclidean distance between two such vectors [22]. In our case of functions of unitation, the phenotypes correspond to the number of ones in a search point, hence the number of ones may be used as distance measure to optimise the class of functions of unitation [23]. Note that when using phenotypic sharing in this way, we are in fact using problem-specific knowledge: we exploit the fact that we are dealing with a function of unitation. Such knowledge may not be available in a general black-box setting, and in this case only genotypic sharing may be used.

For the theoretical work presented in this paper we will use the phenotypic distance for three main reasons. The first is that this allows to highlight problems that may be encountered by using fitness sharing, even when problem knowledge is incorporated in the diversity mechanism. The second is that using phenotypic sharing allows for easier comparisons with previous results available in the literature, since they used this distance measure [7, 8]. The third reason is that the analysis is considerably simplified compared to genotypic sharing but similar conclusions on algorithmic behaviour may be observed. This is indicated by the empirical analysis for genotypic sharing presented in Section 7.

Previous theoretical work on fitness sharing has concentrated on a somewhat unusual implementation of the sharing mechanism. Rather than selecting *individuals* based on their shared fitness $f(x, P)$, selection was done on a level of populations, creating a population that maximises the overall shared fitness of the population [7, 9, 10, 11, 14]. While maximising the shared fitness of the population is indeed what is sought in fitness sharing, this approach has the drawback that the fitness of all the possible combinations of individuals needs to be examined. For large populations this is prohibitive as the number of populations that need to be examined is $\binom{\mu+\lambda}{\lambda}$ (see Section 2.1 for a detailed discussion).

In this paper we analyse the performance of the conventional fitness sharing approach based on individuals to match the approach taught in surveys and tutorials [4, 5, 6] and the way that fitness sharing is used in practice. As pointed out by Goldberg and Richardson [15], shared fitness values can be used with any selection mechanism. However, to allow for comparison with previous work on the effectiveness of fitness sharing for multimodal optimisation we use the same analytical framework, i. e., a standard (μ+λ) EA using the shared fitness values within the selection for replacement, and the same example function, i. e., the simple bimodal function TwoMax consisting of two different symmetric branches [7, 24, 25].

In the context of multimodal optimisation one crucial parameter of the algorithm is the population size, since this determines the number of local optima that can be found simultaneously. Lower bounds for the population size have been investigated in different settings [16, 26] and this work further adds to the understanding of the influence of this parameter.

A (μ+1) EA using the unconventional approach (i. e., maximising the phenotypic shared fitness of the population) can efficiently optimise TwoMax for any population size $\mu \geq 2$ [7]. The reason is that, in any population, the individuals with the smallest and the largest number of ones are always accepted for the next generation. Our analysis shows that using the conventional (phenotypic) sharing approach leads to considerably different behaviours of evolutionary algorithms.

We first concentrate on the effects of the parent population in Section 4. A population of size $\mu = 2$ is not sufficient to guarantee that the (μ+1) EA finds both optima in polynomial time. If the two individuals are initialised on the same branch, then there is a high probability that they will both find the same local optimum. Furthermore, there is a chance that the algorithm fails also when the two individuals are initialised on opposite branches. This leads to a worse failure probability than that of a simple crowding algorithm or that of a (1+1) EA that is restarted twice. On the other hand Section 5 shows that for $\mu \geq 3$, once the population is close enough to one optimum, individuals descending the branch heading towards the other optimum are accepted. This threshold, that allows successful runs with probability 1, lies further away from the local optimum as the population size increases.

Concerning the effects of the offspring population, in Section 6 we show that large values of λ can be detrimental. We rigorously prove that increasing the offspring population of a (μ+1) EA to a (μ+λ) EA, with $\mu = 2$ and $\lambda \geq 2$ a constant, results in an overcrowding that can make a (sub-)population go extinct. For the special case of $\lambda = 2$ we also prove an increased failure probability. We complement this result with an empirical analysis that suggests that the (μ+1) EA is successful if $\lambda < \lfloor \mu/2 \rfloor$ and that it almost always fails for $\lambda \geq \mu$. We conclude the paper with an empirical analysis indicating that similar algorithmic behaviour to that proven theoretically also occurs if no problem specific knowledge is available and genotypic sharing is used.

2. Analytical Framework

In our analyses, we consider a simple bimodal function consisting of two different symmetric branches (i. e., OneMax and ZeroMax) and we have defined both 0^n and 1^n to be global optima (see Figure 1). Formally:

$$\textsc{TwoMax}(x) := \max\left\{\sum_{i=1}^{n} x_i,\ n - \sum_{i=1}^{n} x_i\right\} \tag{1}$$

Moreover, we consider a standard (μ+λ) EA as shown in Algorithm 1 using standard bit mutation with mutation probability $1/n$, uniform random selection of parents and truncation selection for selection for replacement. However, instead of the raw fitness, it uses the shared fitness value in the truncation selection.

We consider fitness sharing as introduced by Goldberg and Richardson [15]. Throughout this work, $|x|$ denotes the number of 1-bits in x. The *shared fitness* of an individual $x \in P$ is

$$f(x, P) := \frac{f(x)}{\sum\limits_{y \in P} sh(x, y)}$$

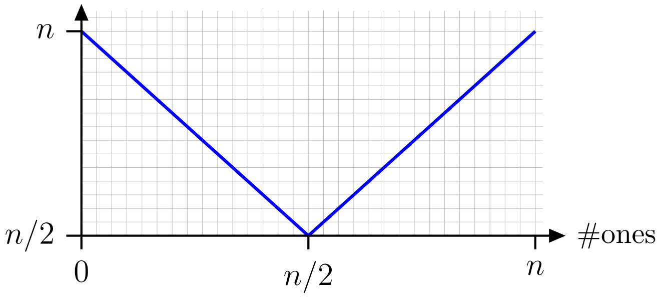

Figure 1: Sketch of the function TwoMax.

Algorithm 1 (μ+λ) EA with fitness sharing

```
Let $t = 0$ and initialise $P_0$ as a population of $\mu$ individuals chosen uniformly at random from $\{0,1\}^n$.
repeat
    for each $1 \leq i \leq \lambda$ do
        Select a parent $x \in P$ uniformly at random from the population.
        Let $x^i := x$. Flip each bit in $x^i$ independently with probability $1/n$.
    end for
    Create a new population $P_{t+1}$ by selecting the $\mu$ best individuals according to their shared fitness in $P_t \cup \bigcup_{i=1}^{\lambda} \{x^i\}$, breaking ties towards favouring offspring over parents, breaking remaining ties uniformly at random.
    Let $t := t + 1$.
until stopping criterion met
```

and the *sharing function* is

$$sh(x, y) := \max\left\{0, 1 - \left(\frac{d(x, y)}{\sigma}\right)^{\alpha}\right\}.$$

Here, $d(x, y)$ is the distance between the two individuals x and y and σ is the *sharing distance* beyond which individuals do not share fitness. More precisely, if $d(x, y) < \sigma$ then $sh(x, y) > 0$ and the shared fitness of x and y is lower than their true fitness. We say that then x and y *share fitness*. If $d(x, y) \geq \sigma$ then $sh(x, y) = 0$ and x and y do not share fitness. We consider fitness sharing with phenotypic sharing as in [7], where the distance between individuals is based on the number of ones: $d(x, y) := \big||x| - |y|\big|$. Note that d is a distance metric in phenotype space, that is, $d(x, y) = 0$ implies that x

and y have identical phenotypes, even though their genotypes might be very different. We use $\sigma = n/2$ (as in [7]) as this is the smallest distance that allows us to discriminate between the two branches. The parameter α is a constant, typically set to 1, that regulates the shape of the sharing function. We use the standard value $\alpha = 1$ and obtain

$$f(x, P) := \frac{f(x)}{\sum\limits_{y \in P} \max\left\{0, 1 - \frac{\big||x| - |y|\big|}{n/2}\right\}}.$$

For $s := \mu + \lambda$, let $P := \{x_1, x_2, \dots, x_s\}$ denote the extended population of current search points and the new offspring, labelled such that

$$|x_1| \leq |x_2| \leq \dots \leq |x_s|.$$

Let

$$D_j := \sum_{i=1}^{s} \min\left\{\big||x_j| - |x_i|\big|, \frac{n}{2}\right\}$$

denote the sum of phenotypic distances of x_j to all other members of the extended population. Individual distances are capped at the sharing distance $n/2$ so that the shared fitness can be written as

$$f(x_i, P) = \frac{f(x_i)}{s - \frac{D_i}{n/2}} = \frac{f(x_i)}{s - 2D_i/n}.$$

Since we are particularly interested in the multi-local perspective and aim at analysing the global exploration capabilities of the population-based EA, we call a run *successful* if it manages to find both optima of TwoMax (i. e., a population is reached that contains both 0^n and 1^n) efficiently. The expected number of generations for this to happen is called *expected running time.*

In the remainder we say that an event happens *with overwhelming probability (w. o. p.)* if it occurs with probability at least $1 - 2^{-\Omega(n^{\varepsilon})}$ for some constant $\varepsilon > 0$.

2.1. On the Time Complexity of Implementing Fitness Sharing

Before analysing the optimisation time, we discuss the overhead from implementing fitness sharing in terms of the classical notion of computation time. To this end, we assume that fitness values $f(x)$ are already known and accessible in time $O(1)$.

Computing sharing function values $sh(x, y)$. In what follows we denote by $T(n)$ the time to compute a sharing function $sh(x, y)$. A naive implementation would give $T(n) = \Theta(n)$ for both phenotypic and genotypic distances. If the phenotype $|x|$ is stored when computing $f(x)$, the phenotypic sharing function can be computed in additional time $O(1)$. Another approach that works for phenotypic and genotypic sharing is to update $sh(x, y)$ according to the respective value of x's parent and checking any bits flipped during mutation. Since in expectation only a constant number of bits have to be reconsidered, this leads to a constant expected time (and $O(n)$ preprocessing time) for each value $sh(x, y)$.

With both population-based and the conventional individual-based fitness sharing we need to compute or maintain $sh(x, y)$ for all individuals x, y from the union of parents and offspring. These can be stored in a $(\mu+\lambda) \times (\mu+\lambda)$ matrix that takes time $\Theta((\mu + \lambda)^2 T(n))$ to compute initially, but can be updated in time $\Theta(\lambda(\mu+\lambda)T(n))$ in each generation as only distances between the λ offspring and the other $\mu + \lambda - 1$ search points need to be computed.

Lemma 1. *Let $T(n)$ be the time to compute $sh(x, y)$ for any two search points x, y. Then for a population P of μ parents and λ offspring in a (μ+λ) EA a $(\mu + \lambda) \times (\mu + \lambda)$ matrix of all values $sh(x, y)$ can be created in time $\Theta((\mu + \lambda)^2 T(n))$ and updated in each generation in time $O(\lambda(\mu + \lambda)T(n))$.*

Computing shared fitness values $f(x, P)$. In order to compute a shared fitness $f(x, P)$ from $f(x)$, we need to compute $\sum_{y \in P} sh(x, y)$. This sum can be computed from scratch in time $\Theta(|P|)$, assuming that the sharing values are available from a table with $O(1)$ access time. It can further be computed more efficiently by using incremental steps. If we have stored $\sum_{y \in P'} sh(x, y)$ for a population P' with $O(1)$ access time, we can compute

$$\sum_{y \in P} sh(x, y) = \sum_{y \in P'} sh(x, y) - \sum_{y \in P' \setminus P} sh(x, y) + \sum_{y \in P \setminus P'} sh(x, y) \qquad (2)$$

in time $O(1 + |P' \setminus P| + |P \setminus P'|)$. This is $O(1)$ if P and P' only differ in one element, and $O(\lambda)$ if they differ in at most λ elements.

Time complexity of individual-based fitness sharing. The conventional individual-based fitness sharing computes $f(x, P) = f(x)/\sum_{y \in P} sh(x, y)$ for the same population P of μ parents and λ offspring. Using (2) and the arguments from the previous paragraph, given a matrix of all $sh(x, y)$ values, all $\mu + \lambda$

values $f(x, P)$ values can be computed incrementally in time $O(\lambda(\mu + \lambda))$. The preprocessing time at the start of the run is $O((\mu + \lambda)^2)$. Along with Lemma 1, we obtain the following time bounds.

Theorem 2. *Let $T(n)$ be the time to compute $sh(x, y)$ for any two search points x, y. Then the overhead from individual-based fitness sharing in one generation of the (μ+λ) EA is $O(\lambda(\mu + \lambda)T(n))$, with an additional preprocessing time at the start of the run of $\Theta((\mu + \lambda)^2 T(n))$.*

Time complexity of population-based fitness sharing. Given a population P of μ parents and λ offspring, population-based fitness sharing looks for a subpopulation $P' \subseteq P$ of size $|P'| = \mu$ that maximises the shared fitness of the population, $f(P') = \sum_{x \in P'} f(x, P')$. Note that there are $\binom{\mu+\lambda}{\mu}$ possibilities to choose P' and we are not aware of an efficient algorithm that is faster than computing all $\binom{\mu+\lambda}{\mu}$ shared population fitnesses.

We describe the most efficient way we could find, based on computing $f(P')$ values incrementally. We iterate over all possible population of size μ that can be formed from $\mu + \lambda$ parents and offspring. Chase's TWIDDLE algorithm [27] outputs a sequence $P_1, P_2, \dots$ of all such size-μ populations in time $O\left(\binom{\mu+\lambda}{\mu}\right)$, and this sequence has the property that two subsequent populations only differ in one element.

Now consider two populations P_i, P_{i+1}, both of size μ, such that P_{i+1} differs from P_i in just element: $P_{i+1} = (P_i \setminus \{z\}) \cup \{w\}$. Further assume $\sum_{y \in P_i} sh(x, y)$ are stored for all $x \in P_i$ with $O(1)$ access time. Then for all $x \in P_i \cup P_{i+1}$, by (2)

$$\sum_{y \in P_{i+1}} sh(x, y) = \sum_{y \in P_i} sh(x, y) - sh(x, z) + sh(x, w)$$

can be computed in time $O(1)$.

So if $f(P_i)$ and all $\sum_{y \in P_i} sh(x, y)$ are known, $f(P_{i+1})$ can be computed as

$$\begin{aligned} f(P_{i+1}) &= \sum_{x \in P_i \cup P_{i+1}} f(x, P_{i+1}) + f(w, P_{i+1}) \\ &= \sum_{x \in P_i \cup P_{i+1}} \frac{f(x)}{\sum_{y \in P_{i+1}} sh(x, y)} + f(w, P_{i+1}) \end{aligned}$$

where the elements in the last sum can be computed in total time $O(\mu)$ and $f(w, P_{i+1})$ can be computed in time $O(\mu)$ as well.

So we can compute shared population fitness values for all size-μ populations and find a best one in time $O\left(\lambda(\mu+\lambda)T(n) + \binom{\mu+\lambda}{\mu}\mu\right)$ per generation and initial preprocessing time $O((\mu+\lambda)^2T(n))$.

Theorem 3. *Let $T(n)$ be the time to compute $sh(x, y)$ for any two search points x, y. Population-based fitness sharing in a (μ+λ) EA can be implemented in such a way that the overhead from fitness sharing is time $O((\mu+\lambda)^2T(n))$ for preprocessing and time $O\Big(\lambda(\mu+\lambda)T(n) + \binom{\mu+\lambda}{\mu}\mu\Big)$ per generation.*

3. General Results

Phenotypic fitness sharing, along with the shape of the TwoMax function, implies that an individual with a better fitness than that of any other individual in the population will always survive, as it has a better fitness than the individual with the closest number of ones, and it has a larger phenotypic distance to other individuals. This means that in a (μ+1) EA the current best fitness never decreases; this also holds if multiple individuals have the same current best fitness, as only one individual is removed by selection.

Lemma 4. *Let $P = \{x_1, \dots, x_s\}$ with $|x_1| \leq \cdots \leq |x_s|$. If $f(x_1) > f(x_2)$ then $f(x_1, P) > f(x_2, P)$. Likewise, if $f(x_{s-1}) < f(x_s)$ then $f(x_{s-1}, P) < f(x_s, P)$.*

As a result, the (μ+1) EA never decreases its current best fitness and finds at least one optimum in expected time $O(\mu n \log n)$.

Proof. We prove the first statement. The second statement will follow by symmetry, swapping the meaning of zeros and ones. By definition of phenotypic fitness sharing,

$$f(x_1, P) = \frac{f(x_1)}{\mu + 1 - \frac{D_1}{n/2}} \quad \text{and} \quad f(x_2, P) = \frac{f(x_2)}{\mu + 1 - \frac{D_2}{n/2}}$$

Since $f(x_1) > f(x_2)$, it is sufficient to show that $D_1 > D_2$ to prove the statement. This follows by definition of D_j since, according to how the individuals are labelled, for all $3 \leq i \leq \mu + 1$ if x_1 shares fitness with x_i then

x_2 also shares fitness with x_i. It further holds that $\big||x_1| - |x_i|\big| > \big||x_2| - |x_i|\big|$ as $|x_1| < |x_2|$ while $\big||x_1| - |x_1|\big| = \big||x_2| - |x_2|\big| = 0$ and $\big||x_2| - |x_1|\big| = \big||x_1| - |x_2|\big|$.

The time bound follows from standard fitness level arguments: For an individual x with $f(x) = i$, $n/2 \leq i \leq n-1$, we have either $|x| = i$ (if $|x| \geq n/2$) or $|x| = n - i$ (if $|x| \leq n/2$); see (1). We consider the case $|x| = i$: To improve the fitness it suffices to flip one of the remaining i 1-bits and leave all other bits unchanged. The probability for this event is $\binom{i}{1} \cdot 1/n \cdot (1-1/n)^{n-1} \geq i/(en)$. Since the probability of selecting x as parent is $1/\mu$, the probability for a fitness improvement during a generation is at least $i/(e\mu n)$. Since the waiting times are geometrically distributed, we get an upper bound of

$$e\mu n \sum_{i=n/2}^{n-1} \frac{1}{i} = O(\mu n \log n)$$

for the expected number of fitness evaluations to increase the fitness from $n/2$ to n. The case $|x| = n - i$ is proven by considering the remaining i 0-bits in the very same way. □

The symmetry between $f(x_1, P)$ vs. $f(x_2, P)$ and $f(x_{s-1}, P)$ vs. $f(x_s, P)$ follows from swapping the meaning of zeros and ones. This also applies to further statements, where for simplicity we omit symmetric statements.

The following Main Lemma gives sufficient and necessary conditions on when the shared fitness of one individual is better than another.

Lemma 5 (Main Lemma). *Let $P = \{x_1, \dots, x_s\}$ with $|x_1| \leq \dots \leq |x_s|$ and fix $1 \leq i \leq s-1$. If $f(x_i) - f(x_{i+1}) = |x_{i+1}| - |x_i| > 0$ and $|x_s| - |x_1| \leq n/2$,*

$$\begin{aligned} f(x_{i+1}, P) \geq f(x_i, P) &\Leftrightarrow f(x_i) \cdot (2i - s) + D_i \geq s \cdot n/2 \\ &\Leftrightarrow f(x_{i+1}) \cdot (2i - s) + D_{i+1} \geq s \cdot n/2. \end{aligned}$$

The same holds if all inequalities "$\geq$" are replaced by strict inequalities "$>$". Moreover, for $i = s - 1$

$$f(x_s, P) > f(x_{s-1}, P) \Leftrightarrow |x_s| > \sum_{i=1}^{s-1} |x_i| - \frac{n}{2} \cdot (s-4).$$

Proof. Note that $|x_s| - |x_1| \leq n/2$ implies that all pairs of individuals do share fitness. We have

$$f(x_i, P) = \frac{f(x_i)}{s - \frac{D_i}{n/2}}.$$

Comparing D_i and D_{i+1}, for the latter the distance to $x_1, \dots, x_{i-1}$ is higher by $|x_{i+1}| - |x_i|$, and the distance to $x_{i+2}, \dots, x_s$ is lower by $|x_{i+1}| - |x_i|$:

$$\begin{aligned} D_{i+1} &= D_i + (i-1) \cdot (|x_{i+1}| - |x_i|) + (s-i-1) \cdot (|x_i| - |x_{i+1}|) \\ &= D_i + (2i - s) \cdot (|x_{i+1}| - |x_i|). \end{aligned}$$

Using the shorthand $h := |x_{i+1}| - |x_i|$,

$$f(x_{i+1}, P) = \frac{f(x_{i+1})}{s - \frac{D_{i+1}}{n/2}} = \frac{f(x_i) - h}{s - \frac{D_i + (2i-s)h}{n/2}}.$$

Now $f(x_{i+1}, P) \geq f(x_i, P)$ is equivalent to

$$\begin{aligned} & \frac{f(x_i) - h}{s - \frac{D_i + (2i-s)h}{n/2}} \geq \frac{f(x_i)}{s - \frac{D_i}{n/2}} \\ \Leftrightarrow \ & \frac{f(x_i) - h}{sn/2 - D_i - (2i-s)h} \geq \frac{f(x_i)}{sn/2 - D_i} \\ \Leftrightarrow \ & (f(x_i) - h) \cdot (sn/2 - D_i) \geq f(x_i) \cdot (sn/2 - D_i - (2i-s)h) \\ \Leftrightarrow \ & f(x_i) \cdot (2i-s)h + h \cdot D_i \geq h \cdot sn/2 \\ \Leftrightarrow \ & f(x_i) \cdot (2i-s) + D_i \geq sn/2. \end{aligned}$$

In the last step we used $h > 0$. The same calculations hold if "$\geq$" is replaced by "$>$" throughout. The second equivalence from the statement follows from

$$\begin{aligned} f(x_i) \cdot (2i-s) + D_i &= (f(x_{i+1}) + h) \cdot (2i-s) + D_{i+1} - h(2i-s) \\ &= f(x_{i+1}) \cdot (2i-s) + D_{i+1}. \end{aligned}$$

The second statement follows by simply applying the first statement:

$$\begin{aligned}
&f(x_s, P) > f(x_{s-1}, P)\\
\Leftrightarrow\ &f(x_s)(2(s-1)-s) + D_s > s \cdot (n/2)\\
\Leftrightarrow\ &f(x_s)(s-2) + (s-1)|x_s| - \sum_{i=1}^{s-1} |x_i| > s \cdot \frac{n}{2}\\
\Leftrightarrow\ &(n-|x_s|)(s-2) + (s-1)|x_s| - \sum_{i=1}^{s-1} |x_i| > s \cdot \frac{n}{2}\\
\Leftrightarrow\ &n(s-2) + |x_s| - \sum_{i=1}^{s-1} |x_i| > s \cdot \frac{n}{2}\\
\Leftrightarrow\ &|x_s| > \sum_{i=1}^{s-1} |x_i| - \frac{n}{2} \cdot (s-4).
\end{aligned}$$

□

Lemma 5 implies the following structural insight: If the population is located on one branch and the shared fitness values of two neighbouring (in the number of 1-bits) search points compare favourably for the higher search point, then the shared fitness strictly increases for all search points further up the branch. More precisely, Lemma 5 gives a condition for the individual of lowest raw fitness (i. e., x_s) to be accepted by selection. Concerning the (μ+1) EA, the condition clearly shows that for $\mu = 2$ at least $n/2$ bits have to flip (i. e., $|x_3| - |x_2| \geq n/2$). On the other hand, for $\mu \geq 3$ offspring with lower fitness values are accepted once the population is close enough to the optimum 0^n. This threshold is further away from the optimum as the population size increases. If mutation was only allowed to flip one bit and $\mu = 3$, then it is necessary that both x_1 and x_2 reach the local optimum before decreasing moves are accepted (i. e., $|x_1| + |x_2| = 0$). For $\mu = 4$ the sum of 1-bits in the first 4 individuals can be up to $|x_1| + |x_2| + |x_3| + |x_4| \leq n/2$ for any decreasing move to be accepted by the (μ+1) EA.

In general, the conditions from Lemma 5 are true for x_{s-1} and x_s if

$|x_{s-1}| < n/2$ and two individuals are in the optimum 0^n as then

$$\begin{aligned} f(x_{s-1})(s-2) + D_{s-1} &\geq (n - |x_{s-1}|)(s-2) + (s-2)|x_{s-1}| - \sum_{i=1}^{s-2} |x_i| \\ &\geq n(s-2) - (s-4)|x_{s-1}| \\ &> n(s-2) - (s-4)n/2 = sn/2. \end{aligned}$$

Lemma 6. *If $P = \{x_1, \dots x_s\}$, $|x_1| \leq \dots \leq |x_s|$, with $|x_{s-1}| < n/2$ and $|x_1| = |x_2| = 0$ then $f(x_{s-1}, P)(s-2) + D_{s-1} > sn/2$.*

4. Population Size $\mu = 2$ Is Not Enough

We first investigate the case of the (2+1) EA, showing that a population size of $\mu = 2$ is not sufficient to guarantee finding both optima. The following lemma gives sufficient and necessary conditions for a single individual on a branch to survive.

Lemma 7. *Let $\mu = 2$ and $P = \{x_1, x_2, x_3\}$ with $|x_1| < n/2 < |x_2| \leq |x_3|$ and $|x_3| - |x_1| \leq n/2$. Let $d_1 := n/2 - |x_1|$ and $d_2 := |x_2| - n/2$, then*

$$\begin{aligned} &f(x_1, P) > f(x_2, P) \Leftrightarrow \\ &d_2 < \left(\frac{3}{2} + \frac{7d_1}{n + 6|x_1|}\right) \cdot d_1 + \frac{(|x_3| + |x_2|)(f(x_2) - f(x_1))}{n/2 + 3|x_1|}. \end{aligned}$$

For $|x_3| = |x_2|$ the statement implies that x_1 survives if the distance from $n/2$ to x_2 is less than around $3/2$ times the distance from $n/2$ to x_1. The condition for survival sharpens when $|x_3| > |x_2|$; however, as x_2 and x_3 are likely to result from a mutation of one another, $|x_3| - |x_2|$ is bounded from above by the number of bits flipped in that mutation.

Proof of Lemma 7. We use the shorthand x_i for $|x_i|$. The claim follows from Lemma 4 if $f(x_1) > f(x_2)$, hence we assume in the following that $f(x_1) \leq$

$f(x_2)$. Then

$$\begin{aligned} f(x_2,P) &< f(x_1,P) \\ \Leftrightarrow \frac{x_2}{3-2D_2/n} &< \frac{n-x_1}{3-2D_1/n} \\ \Leftrightarrow x_2(3n/2-D_1) &< (n-x_1)(3n/2-D_2) \\ \Leftrightarrow x_2(3n/2-(x_2-x_1+x_3-x_1)) &< (n-x_1)(3n/2-(x_2-x_1+x_3-x_2)) \\ \Leftrightarrow x_2(3n/2+2x_1-x_2-x_3) &< (n-x_1)(3n/2+x_1-x_3) \\ \Leftrightarrow x_2(3n/2+2x_1-x_2) &< (n-x_1)(3n/2+x_1)-x_3(n-x_1-x_2) \\ \Leftrightarrow x_2(n/2+3x_1) &< (n-x_1)(3n/2+x_1)-(x_3+x_2)(n-x_1-x_2) \\ \Leftrightarrow x_2(n/2+3x_1) &< (n-x_1)(3n/2+x_1)+(x_3+x_2)(f(x_2)-f(x_1)) \end{aligned}$$

and this is equivalent to

$$x_2 < \frac{(n-x_1)(3n/2+x_1)}{n/2+3x_1} + \frac{(x_3+x_2)(f(x_2)-f(x_1))}{n/2+3x_1}.$$

The right-hand side terms can be simplified as follows.

$$\begin{aligned} \frac{(n-x_1)(3n/2+x_1)}{n/2+3x_1} &= \frac{3n^2-x_1n-2x_1^2}{n+6x_1} \\ &= \frac{n}{2}+\frac{5n^2/2-4x_1n-2x_1^2}{n+6x_1} \\ &= \frac{n}{2}+\frac{6n(n/2-x_1)-2(n/2-x_1)^2}{n+6x_1} \\ &= \frac{n}{2}+\left(\frac{n}{2}-x_1\right)\cdot\frac{5n+2x_1}{n+6x_1} \\ &= \frac{n}{2}+\left(\frac{n}{2}-x_1\right)\cdot\left(\frac{3}{2}+\frac{7(n/2-x_1)}{n+6x_1}\right). \end{aligned}$$

Together, $f(x_1,P) > f(x_2,P)$ is equivalent to

$$x_2-\frac{n}{2} < \left(\frac{3}{2}+\frac{7(n/2-x_1)}{n+6x_1}\right)\cdot\left(\frac{n}{2}-x_1\right)+\frac{(x_3+x_2)(f(x_2)-f(x_1))}{n/2+3x_1}. \quad \square$$

The following theorem states that with a probability greater than 1/2, the (2+1) EA will end up with both individuals in the same optimum, leading to an exponential running time from there. We remark that two independent runs of a (1+1) EA as well as a (2+1) EA with deterministic crowding are more efficient as they both find both optima with probability exactly 1/2, leading to expected runtimes of $O(n \log n)$ [7].

Theorem 8. *The (2+1) EA with fitness sharing with probability* $1/2 + \Omega(1)$ *will reach a population with both members in the same optimum, and then the expected time for finding both optima from there is* $\Omega(n^{n/2})$*.*

Proof. Using that $2^{-n}\binom{n}{i} \le 2^{-n}\binom{n}{n/2} = \Theta(1/\sqrt{n})$ for any $0 \le i \le n$, it is easy to show that with probability $1 - O(n^{1/3}/\sqrt{n}) = 1 - o(1)$ for both initial search points x_1, x_2 we have $|x_1|, |x_2| \notin [n/2 - n^{1/3}, n/2 + n^{1/3}]$. By symmetry, with probability $1/2 - o(1)$, x_1 and x_2 are on the same branch. Since at least $n^{1/3}$ bits would have to be flipped in one mutation, the probability of a mutation jumping from one branch to the other is then at most $1/(n^{1/3}!) = 2^{-\Omega(n^{1/3} \log n)}$, and the probability of this happening in expected polynomial time is still of the same order. This implies that w. o. p. no individuals on the opposite branch will be created in polynomial time as long as no offspring of decreasing fitness are ever accepted on the current branch. In the following we prove by contradiction that such offspring are always rejected.

Assuming both search points and the offspring are all on the same branch, w. l. o. g. the left branch, and labelling them by x_1, x_2, x_3 with $|x_1| \le |x_2| \le |x_3|$, by Lemma 5

$$f(x_3, P) \ge f(x_2, P) \Leftrightarrow f(x_2) + D_2 \ge 3 \cdot \frac{n}{2} \tag{3}$$

where $D_2 = (|x_2| - |x_1|) + (|x_3| - |x_2|) = |x_3| - |x_1|$. Then $f(x_2) + D_2 = n - |x_2| + |x_3| - |x_1| \le n + |x_3| - |x_2|$. This implies that (3) only holds if $|x_3| - |x_2| \ge n/2$, which is a contradiction since there are no points on the left branch differing in more than $n/2$ one-bits. Hence, the claim that no offspring on the left branch of worse fitness than x_2 are ever accepted, is proved. By Lemma 4, 0^n will be reached in expected time $O(n \log n)$. In a further expected $2 \cdot (1 - 1/n)^n = O(1)$ generations, the extended population will contain a clone of 0^n, and from then on any offspring x_3 with $0 < |x_3| \le n/2$ will be rejected. Then the expected time to create an individual on the other branch is $\Omega(n^{n/2})$ since at least $n/2$ bits need to flip.

The claimed probability $1/2 + \Omega(1)$ follows from considering the following additional event, which is disjoint from the above. The algorithm also fails if, using the notation from Lemma 7, $3\sqrt{n}/4 \le d_2 \le \sqrt{n}$ (probability at least

0.02) and $\sqrt{n}/3 \geq d_1 \geq 0$ (probability at least 0.21). If then in the first generation a clone of x_2 is generated (probability at least $1/2 \cdot (1-1/n)^n > 1/8$), we have

$$\left(\frac{3}{2} + \frac{7d_1}{n+6|x_1|}\right) \cdot d_1 + \frac{(x_3+x_2)(f(x_2)-f(x_1))}{n/2+3x_1} \leq \frac{\sqrt{n}}{3} \cdot \frac{3}{2} + O(1) < \frac{3\sqrt{n}}{4} \leq d_2$$

if n is large enough. Now Lemma 7 implies $f(x_1, P) < f(x_2, P) = f(x_3, P)$, hence, x_1 will be removed. Then we are in the same situation as when initialising two individuals on the same branch. □

However, there is still a constant probability that the (2+1) EA finds both optima in polynomial expected time. This holds if the EA is initialised with its two search points on different branches, and if these two search points maintain similar fitness values throughout the run.

Theorem 9. *The (2+1) EA with fitness sharing with probability* $\Omega(1)$ *will find both optima in time* $O(n \log n)$*.*

Since the proof of Theorem 9 is quite long, we first provide a sketch of the proof. Let x_1, x_2 be the two initial search points and $d_1 := n/2 - |x_1|$ and $d_2 := |x_2| - n/2$. With probability $\Omega(1)$, x_1 and x_2 are on opposite branches and have similar fitness: $(3/4)\sqrt{n} \leq d_1, d_2 \leq \sqrt{n}$.

Now, assume w. l. o. g. that when a new offspring is created and the population contains x_1, x_2, x_3 in order of their numbers of ones, that x_2 and x_3 are on the same branch. If $f(x_1) > f(x_2)$, Lemma 4 implies that $f(x_1, P) > f(x_2, P)$ and $f(x_2, P) < f(x_3, P)$ if $|x_3| > |x_2|$. Then x_1 is guaranteed to survive.

Now assume $f(x_1) \leq f(x_2)$. It is easy to derive from Lemma 7 and further arguments using $|x_3| - |x_1| > n/2$ that $f(x_1, P) > f(x_2, P)$ follows if $d_1 \geq (2/3) \cdot d_2$. Intuitively, this means that if x_1 and x_2 have a similar fitness—d_1 and d_2 being within a factor of 2/3—then x_1 is guaranteed to survive.

We then define a potential function that indicates a distance to a population where the lower-fitness individual is at risk of dying. For a current population $P = \{x_1, x_2\}$ define

$$g(P) := \min\{d_1, d_2\} - (2/3) \cdot \max\{d_1, d_2\}.$$

This ensures that $g(P) \geq 0 \Rightarrow f(x_1, P) > f(x_2, P)$. The potential of the initial population P_0 is comfortably large: $g(P_0) \geq \sqrt{n}/12$. If $d_1 \leq d_2 - k$ for some $k \in \mathbb{N}$, the potential increases by k if d_1 increases by k. However, the potential only decreases by $(2/3)k$ if d_2 increases by k. Moreover, increasing d_1 is easier than increasing d_2 as the former contains more “incorrect” bits (cf. Lemma 13 in [28]). This shows that, whenever the potential changes, it increases in expectation by 1/3.

A straightforward application of the simplified drift theorem [29, 30] shows that with overwhelming probability the potential never decreases below $\sqrt{n}/24$ in $2^{\Omega(\sqrt{n})}$ steps. So, with overwhelming probability x_1 survives until both optima are reached.

These arguments are made rigorous in the following proof.

Proof of Theorem 9. Let x_1, x_2 be the two initial search points and define $d_1 := n/2 - |x_1|$ and $d_2 := |x_2| - n/2$. Assume for simplicity that $\sqrt{n}$ is a multiple of 4. We claim that with probability $\Omega(1)$ we have

$$\frac{3}{4}\sqrt{n} \leq d_1, d_2 \leq \sqrt{n}, \tag{4}$$

i. e., x_1 and x_2 are on opposite branches and have similar fitness. The probability of these inequalities holding for x_1 is

$$2^{-n} \sum_{d=3/4\cdot\sqrt{n}}^{\sqrt{n}} \binom{n}{n/2-d} \geq \frac{\sqrt{n}}{4 \cdot 2^n} \cdot \binom{n}{n/2-\sqrt{n}} = \Omega(1)$$

where the last step follows from bounding the binomial coefficient from below by $\Omega(2^n/\sqrt{n})$ [31, Lemma 8]. By symmetry, the same holds for d_2 and hence the probability of (4) is $\Omega(1) \cdot \Omega(1) = \Omega(1)$.

Now, assume w. l. o. g. that when a new offspring is created and the population contains x_1, x_2, x_3 in order of their numbers of ones, that x_2 and x_3 are on the same branch. The case where x_1 and x_2 are on the same branch is symmetric.

If $f(x_1) > f(x_2)$, Lemma 4 implies that $f(x_1, P) > f(x_2, P)$ and $f(x_2, P) < f(x_3, P)$ if $|x_3| > |x_2|$. Then x_1 is guaranteed to survive.

In the following we assume $f(x_1) \leq f(x_2)$. The probability of flipping at least $\sqrt{n}/6$ bits in one mutation is at most $1/(\sqrt{n}/(6))! = 2^{-\Omega(\sqrt{n}\log n)}$ and the probability that this happens in expected polynomial time is still of the same order. So in the following we work under the assumption that such a mutation does not happen.

For $|x_3| - |x_1| \leq n/2$ we know from Lemma 7 that $f(x_1, P) > f(x_2, P)$ follows if

$$d_2 < \left(\frac{3}{2} + \frac{7d_1}{n + 6|x_1|}\right) \cdot d_1 + \frac{(|x_3| + |x_2|)(f(x_2) - f(x_1))}{n/2 + 3|x_1|}$$

which is implied by

$$d_2 \leq \frac{3}{2} \cdot d_1$$

or, equivalently,

$$d_1 \geq \frac{2}{3} \cdot d_2. \tag{5}$$

The same holds for $|x_3| - |x_1| > n/2$ as then $d_1 + d_2 + \sqrt{n}/6 > n/2$, along with $d_1 \geq \frac{2}{3} \cdot d_2$, implies

$$d_1 = \frac{2}{5} \cdot d_1 + \frac{3}{5} \cdot d_1 \geq \frac{2}{5} \cdot d_1 + \frac{2}{5} \cdot d_2 > \frac{2}{5} \cdot \left(\frac{n}{2} - \frac{\sqrt{n}}{6}\right),$$

hence $f(x_1) = n/2 + d_1 = 7/10 \cdot n - O(\sqrt{n})$. The shared fitness of x_1 is thus, using $D_1 = n/2 + |x_2| - |x_1| \geq n - \sqrt{n}/6$,

$$f(x_1, P) = \frac{f(x_1)}{3 - \frac{n - \sqrt{n}/6}{n/2}} = \frac{f(x_1)}{1 + \frac{1}{3\sqrt{n}}} = \frac{7}{10} \cdot n - O(\sqrt{n}).$$

The shared fitness of x_2 is smaller, for n large enough, even in the best case where x_2 does not share with x_1 and $|x_3| = n$:

$$f(x_2, P) \leq \frac{f(x_2)}{3 - \frac{n/2 + n - |x_2|}{n/2}} = \frac{f(x_2)}{2 - \frac{n - |x_2|}{n/2}} = \frac{n}{2} < f(x_1, P).$$

This establishes (5) as sufficient condition for the survival of x_1, regardless of whether $|x_3| - |x_1| \leq n/2$.

For a current population $P = \{x_1, x_2\}$ define a potential

$$g(P) := \min\{d_1, d_2\} - \frac{2}{3} \cdot \max\{d_1, d_2\}. \tag{6}$$

Intuitively, the potential indicates a distance to a population where the lower-fitness individual is at risk of dying. For $d_1 \leq d_2$ we have

$$g(P) \geq 0 \;\Leftrightarrow\; d_1 \geq \frac{2}{3} \cdot d_2 \;\Rightarrow\; f(x_1, P) > f(x_2, P).$$

using Lemma 7. Now we show that the potential with high probability never decreases to 0, which implies that x_1 survives until both optima are reached eventually.

For the initial population P_0 we have $g(P_0) \geq (3/4)\sqrt{n} - (2/3)\sqrt{n} \geq \sqrt{n}/12$.

Assume again w. l. o. g. that $d_1 \leq d_2$. We claim that while $g(P) \leq \sqrt{n}/12$, there is a positive drift towards higher potential values. Note that $g(P) \leq \sqrt{n}/12$ implies

$$d_2 - d_1 \geq \frac{d_2}{3} - \frac{\sqrt{n}}{12} \geq \frac{3\sqrt{n}}{12} - \frac{\sqrt{n}}{12} = \frac{\sqrt{n}}{6}.$$

As we do not allow jumps of this length, if $d_1 < d_2$ then the same will hold for the distances in the next generation. In other words, the roles of d_1 and d_2 in the min and max terms of (6) do not change.

If P_t is the current population at generation t, and $P_{t+1} = \{x'_1, x'_2\}$ then

$$\begin{aligned} &\mathrm{E}\left(g(P_{t+1}) - g(P_t) \mid P_t\right) \\ =& \sum_{d=1}^{\infty} d \cdot \mathrm{Prob}\left(|x'_1| = |x_1| - d\right) - \sum_{d=1}^{\infty} \frac{2d}{3} \cdot \mathrm{Prob}\left(|x'_2| = |x_2| + d\right). \end{aligned}$$

Finding an improvement by d is easier for x_1 than for x_2 as the former contains more 'incorrect' bits. Formally, Lemma 13 in [28] along with the symmetry of TwoMax implies that

$$\mathrm{Prob}\left(|x'_1| = |x_1| - d\right) \geq \mathrm{Prob}\left(|x'_2| = |x_2| + d\right).$$

So we get

$$\begin{aligned} &\mathrm{E}\left(g(P_{t+1}) - g(P_t) \mid P_t\right) \\ \geq& \sum_{d=1}^{\infty} d \cdot \mathrm{Prob}\left(|x'_1| = |x_1| - d\right) - \sum_{d=1}^{\infty} \frac{2d}{3} \cdot \mathrm{Prob}\left(|x'_1| = |x_1| - d\right) \\ =& \frac{1}{3} \cdot \mathrm{Prob}\left(|x'_1| < |x_1|\right) \\ \geq& \frac{(1/3) \cdot |x_1|}{2en} \end{aligned}$$

as the probability of selecting x_1 as parent and increasing its number of zeros is at least $|x_1|/(2en)$. Now, if $|x_1| \leq n/6$ then $d_1 \geq (2/3) \cdot n/2$ and (5) is always true. Hence, we can assume $|x_1| \geq n/6$ and get

$$\mathrm{E}\left(g(P_{t+1}) - g(P_t) \mid P_t\right) \geq \frac{1}{36e}.$$

Using the simplified drift theorem [29, 30], we see that for $a := \sqrt{n}/24$ and $b := \sqrt{n}/12$ the first condition is satisfied for $\varepsilon := 1/6$. The second condition on jump lengths follows by standard arguments: for all $d \in \mathbb{N}$

$$\mathrm{Prob}\left(|x_2'| = |x_2| + d\right) \leq \mathrm{Prob}\left(|x_1'| = |x_1| - d\right) \leq \frac{1}{d!} \leq \frac{2}{2^d}.$$

This shows that with probability $2^{-\Omega(\sqrt{n})}$ the potential never decreases below $\sqrt{n}/24$ in $2^{\Omega(\sqrt{n})}$ steps.

If individuals on both branches survive, by standard arguments (cf. Theorem 4 in [7]) both optima will be reached in expected time $cn \log n$ for some constant $c > 0$. By Markov's inequality, the probability of not having done so after $c' \cdot cn \log n$ generations is at most $1/c'$. Choosing c' large enough and taking into account all failure probabilities, the (2+1) EA finds both optima in time $O(n \log n)$ with probability $\Omega(1)$. □

5. Population Size $\mu \geq 3$ Always Finds Both Optima

A population of size $\mu = 2$ may fail, but we show that a (μ+1) EA with fitness sharing and $\mu \geq 3$ always finds both optima in expected time $O(\mu n \log n)$.

The following lemma is an extension of the Main Lemma (Lemma 5) to the case where an individual $x_{\mu+1}$ is on the other branch compared to the rest of the population. In particular, a stronger condition is given such that $x_{\mu+1}$ will survive selection when $f(x_\mu) > f(x_{\mu+1})$. The proof is similar to the one for the Main Lemma.

Lemma 10. *Let $|x_\mu| < n/2$, $|x_{\mu+1}| > n/2$ and $f(x_\mu) > f(x_{\mu+1})$. Also let $d_\mu := n/2 - |x_\mu|$ and $d_{\mu+1} := |x_{\mu+1}| - n/2$. Then*

$$f(x_\mu) \cdot (\mu - 1) \cdot \frac{d_\mu}{d_\mu - d_{\mu+1}} + D_\mu \geq (\mu + 1) \cdot \frac{n}{2} \Rightarrow f(x_{\mu+1}, P) \geq f(x_\mu, P).$$

Proof. By considering that, for all $1 \leq i \leq \mu - 1$, the summands of $D_{\mu+1}$ are bounded as

$$\begin{aligned}\min\big(\big||x_{\mu+1}| - |x_i|\big|, n/2\big) \geq\ & \min\big(\big|n/2 - |x_i|\big|, n/2\big)\\ &= n/2 - |x_i|\\ &= d_\mu + |x_\mu| - |x_i|\\ &= d_\mu + \min\big(\big||x_\mu| - |x_i|\big|, n/2\big)\end{aligned}$$

we bound $D_{\mu+1}$ as follows:

$$D_{\mu+1} \geq D_\mu + (\mu - 1) \cdot d_\mu$$

Hence, given that $f(x_{\mu+1}) = f(x_\mu) - d_\mu + d_{\mu+1}$, we get

$$f(x_{\mu+1}, P) = \frac{f(x_{\mu+1})}{\mu + 1 - \frac{D_{\mu+1}}{n/2}} \geq \frac{f(x_\mu) - d_\mu + d_{\mu+1}}{(\mu+1) - \frac{D_\mu + (\mu-1)(|x_{\mu+1}| - |x_\mu|) - (\mu-1)d_{\mu+1}}{n/2}}$$

Now $f(x_{\mu+1}, P) \geq f(x_\mu, P)$ is implied by

$$\frac{f(x_\mu) - d_\mu + d_{\mu+1}}{(\mu+1) - \frac{D_\mu + (\mu-1)(|x_{\mu+1}| - |x_\mu|) - (\mu-1)d_{\mu+1}}{n/2}} \geq \frac{f(x_\mu)}{\mu + 1 - \frac{D_\mu}{n/2}}.$$

This is equivalent to

$$\begin{aligned}&\big(f(x_\mu) - d_\mu + d_{\mu+1}\big)\big((n/2)(\mu+1) - D_\mu\big)\\ &\geq f(x_\mu)\big((n/2)(\mu+1) - D_\mu - (\mu-1)(|x_{\mu+1}| - |x_\mu|) + (\mu-1)d_{\mu+1}\big)\\ &\Leftrightarrow f(x_\mu)(\mu-1)((|x_{\mu+1}| - |x_\mu|) - d_{\mu+1}) \geq (d_\mu - d_{\mu+1})\big((\mu+1)(n/2) - D_\mu\big)\\ &\Leftrightarrow f(x_\mu)(\mu-1)d_\mu + (d_\mu - d_{\mu+1})D_\mu \geq (d_\mu - d_{\mu+1})(\mu+1)(n/2).\end{aligned}$$

In the last equivalence we used that $|x_{\mu+1}| - |x_\mu| = d_{\mu+1} + d_\mu$.

Since $d_\mu > d_{\mu+1}$, the inequality is equivalent to

$$f(x_\mu)(\mu-1)\frac{d_\mu}{d_\mu - d_{\mu+1}} + D_\mu \geq \frac{(\mu+1) \cdot n}{2}. \qquad \square$$

The following lemma states that if there is a bounded number r of individuals in one optimum then they will have better shared fitness than the next sub-optimal individual. This implies that r such individuals survive in the (μ+1) EA; the same holds if there are more than r such individuals in the extended population as only one individual is being removed.

Lemma 11. *Let $P = \{x_1, \ldots, x_s\}$ with $|x_1| \le \cdots \le |x_s|$. Assume $|x_1| = \cdots = |x_r| = 0 < |x_{r+1}|$ and $|x_s| < n$. If $r \le 2$ or if both $|x_{r+1}| \ge n/2$ and $r \le s/2$, then for all $1 \le i \le r$ we have $f(x_1, P) = \cdots = f(x_r, P) > f(x_{r+1}, P)$. In particular, if the current population of the (μ+1) EA contains at least two individuals 0^n, two such individuals always survive.*

Proof. As $f(x_1, P) = \cdots = f(x_r, P)$, we only need to show the claim for $i = 1$.

If $|x_{r+1}| < n/2$, we assume pessimistically that x_{r+1} shares fitness with the same individuals as $x_1, \ldots, x_r$, namely $x_1, \ldots, x_\ell$ for some $\ell \ge r + 1$. Then we have

$$D_1 = |x_{r+1}| + \sum_{j=r+2}^{\ell} (|x_j| - |x_1|) + (s - \ell) \cdot \frac{n}{2} \tag{7}$$

hence

$$\begin{aligned} D_{r+1} &= r|x_{r+1}| + \sum_{j=r+2}^{\ell} (|x_j| - |x_{r+1}|) + (s - \ell) \cdot \frac{n}{2} \\ &= D_1 + |x_{r+1}|(2r - \ell). \end{aligned}$$

Now the claim follows from

$$\begin{aligned} & n(sn/2 - D_{r+1}) > (n - |x_{r+1}|)(sn/2 - D_1) \\ \Leftarrow\ & n(sn/2 - D_1 - |x_{r+1}|(2r - \ell)) > (n - |x_{r+1}|)(sn/2 - D_1) \\ \Leftrightarrow\ & n|x_{r+1}|(2r - \ell) < |x_{r+1}|(sn/2 - D_1) \\ \Leftrightarrow\ & n(2r - \ell) + D_1 < sn/2. \end{aligned}$$

From (7) we see that $D_1 - n\ell$ is largest for $\ell = r + 1$, in which case

$$\begin{aligned} n(2r - \ell) + D_1 = n(r - 1) + |x_{r+1}| + (s - r - 1) \cdot n/2 \\ < nr/2 - n + sn/2 \le sn/2 \end{aligned}$$

if $r \le 2$. If $|x_{r+1}| \ge n/2$ we have $f(x_1, P) = n/r$ and

$$D_{r+1} \le rn/2 + (s - r - 1) \cdot (n - |x_{r+1}|),$$

leading to

$$f(x_{r+1}, P) \le \frac{x_{r+1}}{s - r - (s - r - 1) \cdot (n - |x_{r+1}|) \cdot 2/n}.$$

The above term is strictly increasing with x_{r+1}, hence along with $x_{r+1} < n$ we have

$$f(x_{r+1}, P) < \frac{n}{s-r} \leq \frac{n}{r} = f(x_1, P).$$

□

With these lemmas we are ready to prove the main result of this section.

Theorem 12. *Let $\mu \geq 3$. The (μ+1) EA with fitness sharing will find both optima of* TwoMax *with probability 1 in expected time $O(\mu n \log n)$.*

Proof. By Lemma 4, in expected time $O(\mu n \log n)$ one of the two optima is found. W. l. o. g. we assume the 0^n optimum is found. In expected time $O(\mu)$ a clone of 0^n is created (i. e., $|x_2| = 0$) and by Lemma 11 x_1 and x_2 (or clones thereof) will survive for the rest of the run.

We show that then the individual with the largest number of ones, $x_{\mu+1}$ (or a clone thereof), will always survive. If $|x_\mu| = |x_{\mu+1}|$ then $x_{\mu+1}$ or a clone survive. If $n/2 \leq |x_\mu| < |x_{\mu+1}|$ then $f(x_{\mu+1}) > f(x_\mu)$ and the claim follows from Lemma 4. If $|x_\mu| < n/2$ then Lemma 6 implies $f(x_{s-1})(s-2) + D_{s-1} > sn/2$ (where $s = \mu{+}1$). If $|x_{\mu+1}| \leq n/2$, by the Main Lemma this condition is equivalent to $f(x_{\mu+1}, P) > f(x_\mu, P)$. Otherwise, the same conclusion follows from Lemma 10 as $d_\mu/(d_\mu - d_{\mu+1}) > 1$. So, in all cases $x_{\mu+1}$ survives.

The expected time for $x_{\mu+1}$ reaching 1^n is again $O(\mu n \log n)$ and can be proven in the same way we proved the time bound in Lemma 4. □

Our analysis has revealed two very different behaviours. It is possible that the whole population climbs up one branch. But once a sufficiently large overall fitness value has been obtained – at the latest when two individuals have found an optimum – then the population expands towards lower fitness values as then the individuals with the smallest and the largest numbers of 1-bits always survive.

6. Too Large Offspring Population Sizes

Fitness sharing works for the (μ+1) EA, but for larger offspring populations it can have undesirable effects: if a cluster of individuals creates too many offspring, sharing decreases the shared fitness of all individuals in the cluster, and the cluster may go extinct. We consider this problem of overpopulation for $\mu = 2$ and $\lambda \geq \mu$ with $\lambda = O(1)$. In this setting we cannot guarantee convergence to populations with both optima any more, i. e., depending on λ we can lose one or even both optima.

Assume that all individuals are in the same optimum. With probability $\Omega(1)$, we create $\lambda - 1$ copies and one point with distance 1 to the optimum. Then, $f(x_1, P) = \ldots = f(x_{\lambda+1}, P) = n/((\lambda + 2) - 2/n)$ and $f(x_{\lambda+2}, P) = (n-1)/((\lambda+2) - (\lambda+1) \cdot 2/n)$. We see that $f(x_i, P) < f(x_{\lambda+2}, P)$ for all $i \in \{1, \ldots, \lambda+1\}$ and $\lambda \geq 2$. Thus, selection picks $x_{\lambda+2}$ and one of the optimal points.

Following the same argumentation, we lose both optima if $\lambda \geq 6$: If mutation creates $\lambda - 2$ copies and two points with distance 1 to the optimum (also with probability $\Omega(1)$), we have

$$\begin{aligned} f(x_1, P) = \ldots = f(x_\lambda, P) &= \frac{n}{\lambda + 2 - 2 \cdot 2/n} \\ &< \frac{n-1}{\lambda + 2 - \lambda \cdot 2/n} = f(x_{\lambda+1}, P) = f(x_{\lambda+2}, P) \end{aligned}$$

for $\lambda \geq 6$.

In exactly the same way we show that both optima are lost with probability $\Omega(1)$ if $\lambda \geq 6$ even if they are on different branches, i. e., we create $\lfloor \lambda/2 \rfloor$ offspring on the left branch and $\lceil \lambda/2 \rceil$ on the right branch where exactly one offspring on each branch has distance 1 to the optimum and the remaining offspring are copies.

Offspring populations can also decrease diversity in the following way.

Lemma 13. *With probability $1 - o(1)$, the $(2+\lambda)$ EA with fitness sharing, $\lambda \geq 2$ and $\lambda = O(1)$ will, at some point of time before an optimum is reached, obtain a population with both members on the same branch.*

The following proof mainly uses that in a single iteration with probability $\Omega(1)$ only copies of x_1 and x_2 are created. We then show that if $f(x_1) \neq f(x_2)$ and if we have a surplus of offspring on the branch with smaller fitness (also probability $\Omega(1)$), this branch goes extinct. If $f(x_1) = f(x_2)$ in iteration t we have $f(x_1) \neq f(x_2)$ in iteration $t+1$ with probability $\Omega(1)$ and if $f(x_1) \neq f(x_2)$ in iteration t we still have $f(x_1) \neq f(x_2)$ in iteration $t+1$ with probability $\Omega(1)$. Thus, with probability $1 - 2^{-\Omega(n)}$ there are $\Omega(n)$ iterations with $f(x_1) \neq f(x_2)$ before an optimum is reached and consequently, with probability $1 - 2^{-\Omega(n)}$, one branch will take over the whole population before an optimum is reached.

Proof. Let x_1, x_2 be the individuals of the current population. As in Theorem 8 with probability $1 - o(1)$ we have $|x_1|, |x_2| \notin [n/2 - n^{1/3}, n/2 + n^{1/3}]$ after initialisation and thus, the probability to create an offspring on the other branch is $2^{-\Omega(n^{1/3} \log n)} = o(1)$.

Furthermore, with probability $1 - 2^{-\Omega(n)}$, $f(x_1), f(x_2) \leq (1/2 + \varepsilon)n$, $0 < \varepsilon < 1/2$ constant, holds for the first $\Omega(n)$ many iterations: After initialisation we have $f(x_1), f(x_2) \leq (1/2 + \varepsilon')n$, $0 < \varepsilon' < \varepsilon$ a sufficiently small constant, with probability $1 - 2^{-\Omega(n)}$. In order to gain a progress of $(\varepsilon - \varepsilon') \cdot n$ at least $(\varepsilon - \varepsilon') \cdot n$ many bits have to flip. Due to Chernoff bounds, the probability to achieve this in $o(n)$ many iterations is $2^{-\Omega(n)}$.

Assuming that we have two individuals on different branches after initialisation (otherwise there is nothing to prove), we now show that with probability $1 - 2^{-\Omega(n)}$ we will lose the individual on one of the two branches before an optimum is reached.

We use that with probability $\Omega(1)$ only copies of x_1 and x_2 are created in an iteration. Thus, all individuals on the same branch have the same fitness value. Let x_L, x_R denote an individual on the left and right branch, and δ_L, δ_R the number of offspring on the left and right branch, respectively. Let $d_{i,j} = \min\{n/2, \big||x_i| - |x_j|\big|\}$. We observe that $d_{L,R}$ is the same for all pairs of x_L and x_R and $d_{L,L} = d_{R,R} = 0$. Moreover, $D_L = \delta_R \cdot d_{L,R}$ and $D_R = \delta_L \cdot d_{L,R}$.

We observe that $\delta_L = \delta_R = \lambda/2$ holds with probability $\Omega(1)$ if λ is even. If λ is odd, we have $\delta_L = \lfloor \lambda/2 \rfloor$ and $\delta_R = \lceil \lambda/2 \rceil$ with probability $\Omega(1)$.

We first consider the case $f(x_L) \neq f(x_R)$, w. l. o. g. $f(x_L) > f(x_R)$. If λ is even, the above observation implies $D_L = D_R$ and thus, $f(x_L, P) > f(x_R, P)$. For odd λ we conclude $D_L > D_R$ and thus, $f(x_L, P) > f(x_R, P)$. Hence, only individuals on the left branch survive with probability $\Omega(1)$ since $\mu \leq \delta_L + 1$.

Now consider $f(x_1) = f(x_2) = k$. For odd λ we use exactly the same argument as above: the branch with $\lceil \lambda/2 \rceil$ offspring has lower shared fitness and thus, only individuals on the other (i. e., the left) branch survive. For even λ we need to be more careful since from the above argumentation we can only conclude $f(x_L, P) = f(x_R, P)$ and we pessimistically assume that we select individuals on two different branches in this case. However, we see that a successful mutation occurs with probability at least $\binom{n-k}{1} \cdot (1/n) \cdot (1 - 1/n)^{n-1} \geq (n-k)/(en)$. This is $\Omega(1)$ as long as $k = \Omega(n)$. Thus, with probability $\Omega(1)$ we create $\lambda - 1$ copies and one improved offspring. Since the offspring has larger shared fitness, we have $f(x_L) \neq f(x_R)$ in the next iteration.

In summary: If $f(x_1) = f(x_2)$ in iteration t we have $f(x_1) \neq f(x_2)$ in iteration $t+1$ with probability $\Omega(1)$. If $f(x_1) \neq f(x_2)$ in iteration t we still have $f(x_1) \neq f(x_2)$ in iteration $t+1$ with probability $\Omega(1)$ (since it suffices to only create copies of x_1 and x_2). We conclude that with probability $1-2^{-\Omega(n)}$ there are $\Omega(n)$ iterations with $f(x_1) \neq f(x_2)$ before an optimum is reached. Since in this situation with probability $\Omega(1)$ one branch will take over the whole population, this happens with probability $1-2^{-\Omega(n)}$ before an optimum is reached. □

In order to show that the $(2+\lambda)$ EA also reaches a population with both members in the same optimum we additionally need to show that the population will not be stuck somewhere on the branch and that individuals cannot traverse back to the other branch. We consider this for the special case of $\lambda = 2$.

Theorem 14. *With probability* $1-o(1)$, *the* $(2+2)$ *EA with fitness sharing will, at some point of time, reach a population with both members in the same optimum. The expected time for finding both optima from there is* $\Omega\big(n^{n/2}\big)$.

Proof. Assume that both individuals are on the same branch. This happens with probability $1-o(1)$ before an optimum is reached (see Lemma 13).

The extended population of the $(2+2)$ EA has 4 individuals. We apply Lemma 5 and see that $f(x_3, P) \geq f(x_2, P) \Leftrightarrow D_2 \geq 2n$ where $D_2 = d_{2,1} + d_{2,3} + d_{2,4}$ since $d_{2,2} = 0$. Since all individuals are on the same branch $d_{i,j} \leq n/2$. This implies that $D_2 \leq 3n/2$ and thus, $f(x_3, P) < f(x_2, P)$.

We first show that a current best individual is never lost. If there is a single best individual in the population, this will never be lost since $f(x_1, P) > f(x_2, P) > f(x_3, P)$ (Lemma 4 and 5 as discussed above). If there are 3 or 4 best individuals, we are guaranteed to select at least one of them for the next generation since $\mu = 2$. In case there are 2 best individuals, we again use the above argumentation to prove that $f(x_3, P) < f(x_2, P)$. Thus, we are guaranteed to select at least one of the two best individuals for the next generation.

Since due to the above argumentation we never lose a single best individual, a single improved offspring of a best individual will always be accepted. Thus, we will reach a population with both members in the same optimum.

The claim about the expected time to find both optima follows as in Theorem 8. □

$\mu\backslash\lambda$	1	2	3	4	5	6	7	8	9	10	11	12
2	23	0	0	0	0	0	0	0	0	0	0	0
3	100	27.7	0	0	0	0	0	0	0	0	0	0
4	100	60.2	32	0	0	0	0	0	0	0	0	0
5	100	79.3	64.4	2.5	0	0	0	0	0	0	0	0
6	100	100	82.4	68.7	26.1	0	0	0	0	0	0	0
7	100	100	93.6	86.1	76.8	15.6	0	0	0	0	0	0
8	100	100	100	92.6	87.4	81.6	6.4	0	0	0	0	0
9	100	100	100	99.6	95.7	89.4	82.8	3.9	0	0	0	0
10	100	100	100	100	97.2	95.7	91.8	84.3	3.2	0	0	0
11	100	100	100	100	100	98	94.5	92.9	80.5	2	0.1	0
12	100	100	100	100	100	99	97.8	97.2	94.5	73.8	2.9	0

Table 1: Success rates as percentages of the (μ+λ) EA with phenotypic fitness sharing on TwoMax in 1000 runs, stopped after 100000 generations, and once both optima were found.

7. Experiments for Phenotypic and Genotypic Fitness Sharing

We first present a set of experiments, shown in Table 1, where we ran (μ+λ) EAs for $n = 100$ bits and varying values of $2 \leq \mu \leq 12$ and $1 \leq \lambda \leq 12$. We recorded the success rate as the number of runs where both optima were found within 100000 generations. The table shows a clear distinction between efficient and inefficient behaviour: for $\lambda < \lfloor \mu/2 \rfloor$ runs were always successful, whereas runs for $\lambda \geq \mu$ always failed (except for one run with $\lambda = \mu = 11$).

We further ran experiments to test the performance of genotypic fitness sharing, that is, repeating the above experiments but using Hamming distance as distance measure in the (μ+λ) EAs. Table 2 shows the resulting success rates with sharing radius $\sigma = n/2$ to match the setting from Table 1. Apart from the (2+1) EA, (3+1) EA, and (4+1) EA, all algorithms were unable to find both peaks. The reason could be that the sharing radius needs to be chosen differently. With $\sigma = n/2$, since two uniform random individuals will have Hamming distance $n/2$ in expectation, this means that any two initial individuals will either not share fitness, or share so little that the effect of fitness sharing is negligible.

Table 3 shows success rates when repeating the experiment with a sharing radius of $\sigma = n$, where all individuals always share fitness. One can see that the success rates show a similar pattern compared to Table 1 for phenotypic

$\mu \backslash \lambda$	1	2	3	4	5	6	7	8	9	10	11	12
2	56	0	0	0	0	0	0	0	0	0	0	0
3	76	0	0	0	0	0	0	0	0	0	0	0
4	5	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0

Table 2: Success rates as percentages of the (μ+λ) EA with genotypic fitness sharing and $\sigma = n/2$ on TwoMax in 100 runs, stopped after 100000 generations, and once both optima were found.

sharing, albeit numbers are generally smaller. For the (μ+1) EA success rates seem to converge to 1 with increasing μ, but a few runs still fail. We suspect that this is due to few runs that are initialised with all individuals on one branch.

To test this, we also ran experiments for a modified, favourable initialisation where we drew μ individuals independently and uniformly at random, and then checked whether the population contains at least one individual with $n/2 + \sqrt{n}$ ones and at least one individual with at least $n/2 + \sqrt{n}$ zeros. If this was not the case, the population was discarded and μ new individuals were drawn independently and uniformly at random. The term $n/2 + \sqrt{n}$ was chosen such that two individuals are firmly placed on their respective branches, from which mutations to the other branch are unlikely. We note without giving a formal proof that the probability of having at least $n/2 + \sqrt{n}$ ones is bounded from below by a positive constant $c > 0$ and hence the probability of initialising a population as described is at least $1 - 2(1-c)^{\lambda} = 1 - 2^{-\Omega(\lambda)}$. This means that for λ not too small, only a small fraction of initial populations is discarded. Table 4 shows that when unlucky initialisations are excluded, success rates of 100% are achieved for small λ.

$\mu\backslash\lambda$	1	2	3	4	5	6	7	8	9	10	11	12
2	48	0	0	0	0	0	0	0	0	0	0	0
3	71	38	0	0	0	0	0	0	0	0	0	0
4	81	50	28	0	0	0	0	0	0	0	0	0
5	90	65	48	21	0	0	0	0	0	0	0	0
6	95	67	64	47	18	0	0	0	0	0	0	0
7	94	78	63	53	34	21	0	0	0	0	0	0
8	97	83	69	63	60	40	20	0	0	0	0	0
9	95	86	80	79	63	44	41	20	1	0	0	0
10	100	92	77	69	52	54	49	40	19	5	0	0
11	98	93	88	83	73	68	60	49	31	12	3	0
12	99	98	97	87	85	64	62	64	45	38	18	5

Table 3: Success rates as percentages of the (μ+λ) EA with genotypic fitness sharing and $\sigma = n$ on TwoMax in 100 runs, stopped after 100000 generations, and once both optima were found.

$\mu\backslash\lambda$	1	2	3	4	5	6	7	8	9	10	11	12
2	100	0	0	0	0	0	0	0	0	0	0	0
3	100	89	0	0	0	0	0	0	0	0	0	0
4	100	99	67	0	0	0	0	0	0	0	0	0
5	100	96	85	55	0	0	0	0	0	0	0	0
6	100	100	94	78	43	0	0	0	0	0	0	0
7	100	100	98	89	73	35	0	0	0	0	0	0
8	100	100	99	92	83	63	31	0	0	0	0	0
9	100	100	100	99	89	8	54	24	2	0	0	0
10	100	100	100	99	94	79	68	57	35	0	0	0
11	100	100	100	99	96	86	8	67	46	27	3	0
12	100	100	100	100	98	93	88	75	66	48	41	2

Table 4: Success rates as percentages of the (μ+λ) EA with genotypic fitness sharing and $\sigma = n$ on TwoMax in 100 runs, stopped after 100000 generations, and once both optima were found. Here all runs were initialised using rejection sampling that the initial population contains at least one individual with at least $n/2 + \sqrt{n}$ ones and at least one individual with at least $n/2 + \sqrt{n}$ zeros.

8. Conclusions

This work sheds light on advantages and disadvantages of fitness sharing in multimodal optimisation, particularly in the context of a multi-local perspective where we are interested in locating different global or local optima. To allow for easy comparison with previous work, we used a common analytical framework (i. e., (μ+λ) EA) and example problem (i. e., TwoMax).

Our main contribution is the rigorous theoretical analysis of the conventional fitness sharing mechanism which selects individuals based on their shared fitness (rather than performing selection on a level of populations as done in previous theoretical work) when phenotypic sharing is used. We concentrated on the influence of the population sizes μ and λ as crucial parameters. Regarding the parent population, our analyses show that a population size μ of at least 3 is required to guarantee finding both optima of TwoMax in polynomial time. We also prove that large offspring population sizes λ can cause overpopulation which results in the extinction of whole clusters of search points. The latter results are accompanied by experiments suggesting that the (μ+1) EA is successful if $\lambda < \lfloor \mu/2 \rfloor$ and that it almost always fails for $\lambda \geq \mu$. These findings highlight the risks of using fitness sharing with inappropriate parameters and highlight the need for a better understanding of algorithm parameters. We concluded the paper with an empirical analysis of the genotypic sharing that has to be used when no problem specific knowledge is available. The experiments indicate that similar conclusions on algorithmic performance may be made when the Hamming distance is used. We leave rigorous theoretical proofs of this as an open problem for future work.

In the future it would also be interesting to extend the analyses of fitness sharing and other diversity mechanisms to problems beyond TwoMax. Promising candidates for such work are the set of theory-affine multimodal benchmark functions introduced in [32] or dynamic problems.

9. Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 618091 (SAGE) and by the EPSRC under Grant Agreement No. EP/M004252/1.

References

[1] P. S. Oliveto, D. Sudholt, C. Zarges, On the runtime analysis of fitness sharing mechanisms, in: T. Bartz-Beielstein, J. Branke, B. Filipic, J. Smith (Eds.), Proceedings of Parallel Problem Solving from Nature (PPSN XIII), Vol. 8672 of Lecture Notes in Computer Science, Springer, 2014, pp. 932–941.

[2] M. Preuss, Multimodal Optimization by Means of Evolutionary Algorithms, Springer, 2015.

[3] B. Doerr, N. Hansen, C. Igel, L. Thiele, Theory of Evolutionary Algorithms (Dagstuhl Seminar 15211), Dagstuhl Reports 5 (5) (2016) 57–91.

[4] S. W. Mahfoud, Niching methods, in: T. Bäck, D. B. Fogel, Z. Michalewicz (Eds.), Handbook of evolutionary computation, IOP Publishing and Oxford University Press, 1997, pp. C6.1:1–4.

[5] O. M. Shir, Niching in evolutionary algorithms, in: G. Rozenberg, T. Bäck, J. N. Kok (Eds.), Handbook of Natural Computing, Springer, 2012, pp. 1035–1070.

[6] G. Squillero, A. Tonda, Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization, Information Sciences 329 (2016) 782–799, special issue on Discovery Science.

[7] T. Friedrich, P. S. Oliveto, D. Sudholt, C. Witt, Analysis of diversity-preserving mechanisms for global exploration, Evolutionary Computation 17 (4) (2009) 455–476.

[8] E. Covantes Osuna, D. Sudholt, Analysis of the clearing diversity-preserving mechanism, in: C. Igel, D. Sudholt, C. Witt (Eds.), Proceedings of the 14th ACM/SIGEVO Conference on Foundations of Genetic Algorithms (FOGA), ACM Press, 2017, pp. 55–63.

[9] S. Fischer, I. Wegener, The one-dimensional Ising model: Mutation versus recombination, Theoretical Computer Science 344 (2–3) (2005) 208–225.

[10] D. Sudholt, Crossover is provably essential for the Ising model on trees, in: H. Beyer, U. O'Reilly (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), ACM Press, 2005, pp. 1161–1167.

[11] D.-C. Dang, T. Friedrich, T. Kötzing, M. S. Krejca, P. K. Lehre, P. S. Oliveto, D. Sudholt, A. M. Sutton, Escaping local optima with diversity mechanisms and crossover, in: T. Friedrich, F. Neumann, A. M. Sutton (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), ACM Press, 2016, pp. 645–652.

[12] D. Sudholt, How crossover speeds up building-block assembly in genetic algorithms, Evolutionary Computation 25 (2) (2017) 237–274.

[13] D. Corus, P. S. Oliveto, Standard steady state genetic algorithms can hillclimb faster than mutation-only evolutionary algorithms, IEEE Transactions on Evolutionary Computationdoi:10.1109/TEVC.2017.2745715.

[14] P. S. Oliveto, C. Zarges, Analysis of diversity mechanisms for optimisation in dynamic environments with low frequencies of change, Theoretical Computer Science 561, Part A (2015) 37–56.

[15] D. E. Goldberg, J. Richardson, Genetic algorithms with sharing for multimodalfunction optimization, in: J. J. Grefenstette (Ed.), Proceedings of the International Conference on Genetic Algorithms (ICGA), Lawrence Erlbaum Associates, 1987, pp. 41–49.

[16] A. D. Cioppa, C. D. Stefano, A. Marcelli, On the role of population size and niche radius in fitness sharing, IEEE Transactions on Evolutionary Computation 8 (6) (2004) 580–592.

[17] D. E. Goldberg, Genetic Algorithms for Search, Optimization, and Machine Learning, Addison-Wesley, 1989.

[18] Y. Kuwahara, Multiobjective optimization design of yagi-uda antenna, IEEE Transactions on Antennas and Propagation 53 (6) (2005) 1984–1992.

[19] M. Salazar-Lechuga, J. E. Rowe, Particle swarm optimization and fitness sharing to solve multi-objective optimization problems, in: Proceedings

of the 2005 IEEE Congress on Evolutionary Computation (CEC), Vol. 2, IEEE Press, 2005, pp. 1204–1211.

[20] Y. Zheng, Z. Y. Dong, Y. Xu, K. Meng, J. H. Zhao, J. Qiu, Electric vehicle battery charging/swap stations in distribution systems: Comparison study and optimal planning, IEEE Transactions on Power Systems 29 (1) (2014) 221–229.

[21] R. C. Peralta, A. Forghani, H. Fayad, Multiobjective genetic algorithm conjunctive use optimization for production, cost, and energy with dynamic return flow, Journal of Hydrology 511 (2014) 776–785.

[22] K. Deb, D. E. Goldberg, An investigation of niche and species formation in genetic function optimization, in: Proceedings of the 3rd International Conference on Genetic Algorithms, Morgan Kaufmann Publishers Inc., 1989, pp. 42–50.

[23] P. K. Lehre, P. S. Oliveto, Theoretical analysis of stochastic search algorithms, in: M. G. Resende, R. Marti, P. M. Pardalos (Eds.), Handbook of Heuristics (In press), Springer International Publishing, 2018, preliminary version available on Arxiv: https://arxiv.org/abs/1709.00890. doi:10.1007/978-3-319-07153-4_35-1.

[24] M. Pelikan, D. E. Goldberg, Genetic algorithms, clustering, and the breaking of symmetry, in: M. Schoenauer, K. Deb, G. Rudolph, X. Yao, E. Lutton, J. J. M. Guervós, H.-P. Schwefel (Eds.), Proceedings of Parallel Problem Solving from Nature (PPSN VI), Vol. 1917 of Lecture Notes in Computer Science, Springer, 2000, pp. 385–394.

[25] C. V. Hoyweghen, D. E. Goldberg, B. Naudts, From Twomax to the Ising model: Easy and hard symmetrical problems, in: W. B. Langdon, E. Cantú-Paz, K. E. Mathias, R. Roy, D. Davis, R. Poli, K. Balakrishnan, V. G. Honavar, G. Rudolph, J. Wegener, L. Bull, M. A. Potter, A. C. Schultz, J. F. Miller, E. K. Burke, N. Jonoska (Eds.), Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Morgan Kaufmann, 2002, pp. 626–633.

[26] S. W. Mahfoud, Population size and genetic drift in fitness sharing, in: L. D. Whitley, M. D. Vose (Eds.), Proceedings of the Third Workshop on

Foundations of Genetic Algorithms (FOGA), Morgan Kaufmann, 1994, pp. 185–223.

[27] P. J. Chase, Transposition graphs, SIAM Journal on Computing 2 (2) (1973) 128–133.

[28] B. Doerr, D. Johannsen, C. Winzen, Multiplicative drift analysis, Algorithmica 64 (2012) 673–697.

[29] P. S. Oliveto, C. Witt, Simplified drift analysis for proving lower bounds in evolutionary computation, Algorithmica 59 (3) (2011) 369–386.

[30] P. S. Oliveto, C. Witt, Erratum: Simplified drift analysis for proving lower bounds in evolutionary computation, ArXiv e-prints 1211.7184.

[31] B. Doerr, C. Winzen, Ranking-based black-box complexity, Algorithmica 68 (3) (2014) 571–609.

[32] T. Jansen, C. Zarges, Example landscapes to support analysis of multimodal optimisation, in: J. Handl, E. Hart, P. R. Lewis, M. López-Ibáñez, G. Ochoa, B. Paechter (Eds.), Proceedings of Parallel Problem Solving from Nature (PPSN XIV), Vol. 9921 of Lecture Notes in Computer Science, Springer, 2016, pp. 792–802.