



# Runtime analysis of convex evolutionary search algorithm with standard crossover

Tina Malalanirainy<sup>a,\*</sup>, Alberto Moraglio<sup>a</sup>

University of Exeter, UK



## ARTICLE INFO

### Keywords:

Convex evolutionary search  
Quasi-concave landscapes  
Runtime upper bound  
Standard crossover  
Unifying theory

## ABSTRACT

Evolutionary Algorithms (EAs) with no mutation can be generalized across representations as Convex Evolutionary Search algorithms (CSs). However, the crossover operator used by CSs does not faithfully generalize the standard two-parents crossover: it samples a convex hull instead of a segment. Segmentwise Evolutionary Search algorithms (SEs) are defined as a more faithful generalization, equipped with a crossover operator that samples the metric segment of two parents. In metric spaces where the union of all possible segments of a given set is always a convex set, a SE is a particular CS. Consequently, the representation-free analysis of the CS on quasi-concave landscapes can be extended to the SE in these particular metric spaces. When instantiated to binary strings of the Hamming space (resp.  $d$ -ary strings of the Manhattan space), a polynomial expected runtime upper bound is obtained for quasi-concave landscapes with at most polynomially many level sets for well-chosen population sizes. In particular, the SE solves Leading Ones in at most  $288n \ln[4n(2n+1)]$  expected fitness evaluations when the population size is equal to  $144 \ln[4n(2n+1)]$ .

## 1. Introduction

EAs are known to efficiently solve a large number of problems. Do these problems share any characteristics? More precisely, can we define a class containing all these problems that are efficiently solved by EAs? A representation-free analysis of EAs can be used to answer these questions.

Indeed, such class can be drafted from a geometrical description of the search performed by EAs. For example, the CS [1] has been defined as a representation-free EA generalizing EA with no mutation across representations. The CS samples the convex hull of the selected population at each generation (see Fig. 1 for an illustration). Then, the class of quasi-concave landscapes has been defined as to have convex canonical level sets only [1]. Finally, quasi-concave landscapes with at most polynomially many level sets have been shown to be efficiently solved by the CS in the Hamming (resp. Manhattan) space [2].

However, the CS is not a faithful generalization of EAs with no mutation and with a standard two-parents crossover [3]. observed that crossover operators (including all mask-based crossovers for binary strings [4]) sample an offspring from a segment formed by the two parents. Whereas, the CS makes use of the convex hull recombination that samples an offspring from the convex hull formed by the selected population [1]. Hence, we introduce a more faithful generalization of EAs with no mutation and with a standard two-parents crossover called SES

(Segmentwise Evolutionary Search Algorithm). The SES makes use of the geometric crossover [4] which samples an offspring from the segment formed by two parents. We shall therefore extend the runtime analysis of the CS on quasi-concave landscapes to the SES.

Necessary preliminaries are given in Section 6.1.1 along with the definition of quasi-concave landscapes. The Segmentwise Evolutionary Search algorithm (SES) is defined in Section 4. We show in Section 5 that the SES is a particular CS in some specific metric spaces. The runtime analysis of the SES on a quasi-concave landscape is done in Section 6. Theoretical results are compared to empirical results for the SES in Section 7. Finally, Section 8 presents the conclusion.

## 2. Literature review

The unification of different algorithms into a single algorithm often results in a generalized algorithm with interesting properties. In [5], hydraulic actuators are unified across different types of valves to result in an optimal tuned cascade controller. In [6], ILC (Iterative Learning Control) algorithms are unified across different initial conditions to result in a robust ILC algorithm. In this paper, EAs with no mutation and with a standard two-parents crossover are unified across representations into an algorithm called SES. The SES is shown to search polynomial quasi-concave landscapes in at most polynomial time.

Abbreviations: EA, evolutionary algorithm; CS, convex evolutionary search algorithm; SES, segmentwise evolutionary search algorithm.

\* Corresponding author.

E-mail address: [malala@aims.ac.za](mailto:malala@aims.ac.za) (T. Malalanirainy).

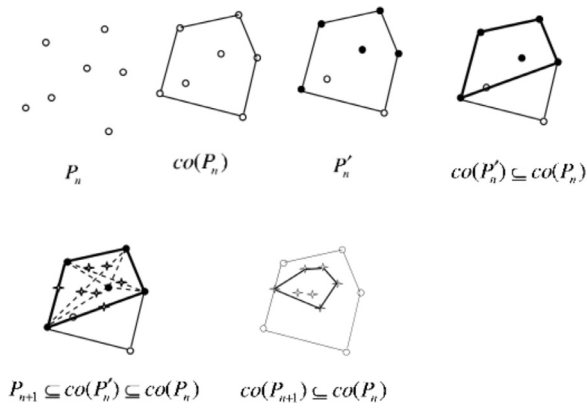


Fig. 1. Convex Search in the Euclidean Space. The figure is taken from [1].

In the context of EAs analysis, a theory that can be instantiated to different representations will be referred to as a unifying theory. This work is a unifying theory using results from schema theory, modelization of evolving population, fitness landscape theory, and runtime analysis.

### 2.1. Schema theory

Holland's schema theorem [7] is the main result on the increase in the number of strictly improving solutions in one generation. It says that we can find some template (called schema) corresponding to a subset of fit solutions in each generation, that increases in size in the next generation. Holland's schema theorem can only be applied to problems where schemata can be defined and only holds for infinitely large populations. The schema theorem has been criticized in [8] for only taking into account the case where a given schema is lost because of the disruptive effect of the genetic operators. Most importantly, the weakness of the schema theorem is due to the limitation of its scope to the one step variation in the number of individuals with a given schema [9].

Radcliffe [10] extended Holland's schema theorem to general non-string representations using equivalence relations. In [11], Holland's schema theorem has been extended to Genetic Programming (GP).

According to Goldberg [12], there exist schemata whose elements always generate offspring that are fitter than their parents. In particular, Goldberg's "building blocks" are short, low order, and highly fit schemata. By using an exact evolution equation, Stephens and Waelbroeck [13] determined the "building blocks" of a GA and showed that they need not be as Goldberg's. Initial steps towards the theoretical analysis of building blocks for GP have been taken in [14].

In [1], a schema is associated to a convex set to define a canonical level set. This results in a class of landscapes called quasi-concave.

### 2.2. Modelization of the evolving population

The evolution of a population has been modelized as a dynamical system for GA [15,16] and GP [17]. A unifying framework that links the dynamics of the population of a GA to the dynamics of the population of a GP has been given in [18].

In [19], a directed graph has been used to model the effect of each operator of the EA of interest on the search space. This model can be applied to any representation. Another graph modelization of the dynamics of EAs is given in [20] as a complex system.

In [21], statistical mechanics have been used to see all possible populations as points whose union makes up the *phase space*. Then, the evolution of a population can be seen as a *trajectory* in this *phase space*. This model is not limited to GA.

In [4], segments have been used to model the effect of crossover. As segments can be defined in any metric space, this model is unifying crossovers across representations.

### 2.3. Fitness landscape theory

Fitness landscape theory can be used to determine how the choice of landscape affects the evolution of the population [22].

One aspect of fitness landscape theory consists of studying the fitness landscape induced by the genetic operator(s) of the EA to be analyzed. The fitness landscape induced by recombination and the fitness landscape induced by mutation have been shown to be homomorphic for GA (resp. GP) in [23]. A similar result was obtained in [24] through a different approach.

In [1], quasi-concave landscapes are defined from the search performed by the convex hull recombination. This links quasi-concave landscapes to EAs with no mutation using a convex hull recombination (also called CSSs).

### 2.4. Runtime analysis

In runtime analysis, we are interested in estimating the number of fitness evaluations needed by an EA to find the first optimal solution in a fitness landscape. There are several runtime analysis methods:

- Markov chains can be used to model EAs [16]. These models can then be used for the runtime analysis of EAs [25].
- Tail inequalities are used to produce runtime upper bounds that hold with overwhelming probability from expected runtimes [26]. Markov's inequality and Chernoff bounds [27] are often used for this method.
- Typical run investigation is the study of the global behaviour of the algorithm. As the global behaviour of the algorithm is predictable with high probability, the corresponding runtime result holds with overwhelming probability [26].
- The artificial fitness levels method is a general approach that uses a partition of the search space into fitness levels. It has been initially used to analyze the (1 + 1) EA on various pseudo-Boolean problems in [26].
- The potential function method [26] is an extension of the artificial fitness levels method. It is used when computing the probability of leaving a fitness level is too costly. We first work with an easier function (which is the potential function), then we take into account the difficult fitness function. The potential function is used to measure the progress of the algorithm, while the fitness function is used to decide whether an offspring is accepted or not. The drift analysis method [28] is a particular case of the potential function. It makes use of a function that measures how far from the global optimum a population is.

The runtime analysis methods above are often applied separately for different representations. Moreover, research is focused on improving runtime analysis methods to obtain the most accurate possible result. In this paper, the artificial fitness levels method is applied uniquely to different representations via a representation-free approach to study EAs with no mutation and with a standard two-parents crossover.

In the literature, the effect of crossover on the runtime has been studied on EAs using both mutation and crossover, mostly for binary representations. The contribution of crossover has been studied on artificial pseudo-boolean problems: OneMax [29,30], jump function [31,32], and royal road functions [33,34]. Then, graph colouring problems inspired by the Ising model have been considered where binary strings representations were used [35,36]. Later on, individuals were represented as a sequence of edges for the all-pairs shortest path problem [37]. To our knowledge, the first unifying runtime analysis method has been introduced in [2] for the study of CSSs on quasi-concave landscapes.

## 3. Background

The representation-free analysis discussed in this paper is based on the findings of Moraglio and Sudholt [2].

1. A representation-free algorithm is defined from existing EAs, by determining the geometric object(s) described by their genetic operator(s).

**Example 1.** In [2], the Convex Evolutionary Search algorithm (CS) is defined as a representation-free algorithm generalizing EAs with no mutation across representations. The genetic operator of the CS describes a convex set.

2. A representation-free landscape corresponding well to the representation-free algorithm is determined. The representation-free landscape is chosen in such a way that its level sets are made up of the geometrical object(s) corresponding to the representation-free algorithm.

**Example 2.** Quasi-concave landscapes are representation-free landscapes whose level sets are defined as convex sets [2].

3. An upper-bound on the expected runtime of the representation-free algorithm on the representation-free landscape is computed through the fitness levels method [26].

**Example 3.** The representation-free runtime result on the analysis of the CS on a quasi-concave landscape, that was first published in [2], is recalled in Theorem 1.

These are explained in more details in the following subsections. Let us start with necessary preliminaries.

### 3.1. Segments and convex sets

Let  $S$  be a search space endowed with a metric  $D$ . We recall that a metric function  $D$  is a mapping from  $S \times S \rightarrow \mathbb{R}_+$  that satisfies for any  $x, y$  and  $z$  in  $S$ :

1.  $D(x, y) = D(y, x)$ ,
2.  $D(x, z) \leq D(x, y) + D(y, z)$ ,
3.  $D(x, y) = 0$  if and only if  $x = y$ .

We start by recalling the notions of segments and convex sets in a discrete metric space  $(S, D)$ . The discrete metric space  $(S, D)$  can be seen as a graph. The elements of  $(S, D)$  are the nodes of the graph and the distance between any two nodes is the length of the shortest paths between them. This length is the number of edges in the path.

**Definition 1 (Segment).** Let  $(S, D)$  be a metric space, and let  $x$  and  $y$  be elements of  $S$ . The segment between  $x$  and  $y$  is the union of the shortest paths between  $x$  and  $y$ . That is,  $[x, y]_D = \{z \in S \mid D(x, z) + D(z, y) = D(x, y)\}$ . The points  $x$  and  $y$  are extremes of the segment  $[x, y]_D$ .

**Example 4.** In the two-dimensional Hamming space  $(\{0, 1\}^2, \text{HD})$ , the segment  $[00, 11]$  is the union of the shortest paths between 00 and 11. The shortest paths between 00 and 11 are:  $\{00, 01, 11\}$  and  $\{00, 10, 11\}$ . Hence,  $[00, 11] = \{00, 01, 10, 11\}$ . Consequently, the same segment can have more than a pair of extremes, unlike the case of the Euclidean space. For instance, we have that  $[00, 11] = [01, 10]$ .

We shall now recall the notion of convexity in a discrete metric space.

**Definition 2 (Geodesic convexity [38]).** Let  $(S, D)$  be a metric space. A subset  $C$  of  $S$  is geodesically convex if all shortest paths between any two points of  $C$  are included in  $C$ . That is,  $[x, y]_D \subseteq C$  for all  $x, y$  in  $C$ .

**Example 5.** Let  $n \geq 2$ , the set  $\{0, 1\}^n$  is geodesically convex for the Hamming (resp. Manhattan) distance. All singletons and segments of length one are geodesically convex for the Hamming (resp. Manhattan) distance.

We will use the term *convex set* for *geodesically convex set* in the rest of the paper. Let  $A$  be a subset of the metric space  $(S, D)$ . We finally recall the notion of convex hull of a subset  $A$ , which is central to the analysis of the CS.

**Definition 3 (Convex hull [38]).** Let  $(S, D)$  be a metric space. The convex hull of a subset  $A$  of  $S$  is the smallest convex set containing  $A$ . In particular, it is the intersection of all convex sets containing  $A$ . The convex hull of  $A$  is denoted  $co(A)$ .

**Example 6.** Let HD denote the Hamming distance. In the metric space  $(\{0, 1\}^2, \text{HD})$ , the convex hull of the set  $\{01\}$  is  $co(\{01\}) = \{01\}$ . The convex hull of the set  $\{00, 10\}$  is itself and is equal to the segment  $[00, 10]$ . The convex hull of the set  $\{01, 10\}$  is  $co(\{01, 10\}) = \{0, 1\}^2$ .

### 3.2. Generalized schemata for strings in $M_{d, \text{HD}}$ and $M_{d, \text{MD}}$

A schema can be seen as a subset of the search space whose elements match some template. In particular, schemata can be used to define convex sets [2].

Traditional schemata of binary strings of length  $n$  are templates with  $n$  positions, where each position is either 0, 1, or the ‘do not care’ symbol  $*$ . We start by recalling the notion of schemata [39] for strings on a finite alphabet.

**Definition 4.** A schema in the set  $\{0, 1, \dots, d-1\}^n$  is a template with  $n$  positions where a position is either:

- Free to take any value in the set  $\{0, 1, \dots, d-1\}$ ,
- Restricted to take values in a non-empty strict subset of  $\{0, 1, \dots, d-1\}$ .

A free position is denoted  $*$ , whereas a restricted position is denoted  $*_A$  where  $A$  is the set of admissible values.

**Example 7.** All the elements of the set  $\{0, 1, 2\}^5$  match the schema  $*****$ . The smallest schema matching the elements 00123 and 21103 is  $*_{\{0,2\}}*_{\{0,1\}}1*_{\{0,2\}}3=*_{02}*_{01}1*_{02}3$ .

Schemata corresponding to convex sets of the metric space  $M_{d, \text{HD}} = (\{0, 1, \dots, d-1\}^n, \text{HD})$  and the metric space  $M_{d, \text{MD}} = (\{0, 1, \dots, d-1\}^n, \text{MD})$  are determined. We shall:

- Prove that any schema corresponds to a convex set for the Hamming distance,
- Determine the schemata corresponding to a convex set for the Manhattan distance.

Indeed, schemata corresponding to convex sets of the metric space  $M_{d, \text{HD}} = (\{0, 1, \dots, d-1\}^n, \text{HD})$  (resp.  $M_{d, \text{MD}} = (\{0, 1, \dots, d-1\}^n, \text{MD})$ ) have been used without proof in [2]. We provide the following results for completeness.

#### 3.2.1. Hamming distance

We first show that all schemata are convex sets in the metric space  $M_{d, \text{HD}} = (\{0, 1, \dots, d-1\}^n, \text{HD})$ .

**Lemma 1.** Any schema in the metric space  $M_{d, \text{HD}}$  is a convex set.

**Proof.** Let  $S$  be a schema in the metric space  $M_{d, \text{HD}}$ , whose admissible values at position  $i$  are the elements of a subset  $A_S(i)$  of  $\{0, 1, \dots, d-1\}$  for  $0 \leq i \leq n-1$ . Let  $x$  and  $y$  be two elements of  $S$ . We show that the segment  $[x, y]$  is contained in  $S$ .

Let  $z \in [x, y]$ , the value of  $z(i)$  is either  $x(i)$  or  $y(i)$ . As both  $x$  and  $y$  belong to  $S$ , then  $x(i)$  and  $y(i)$  belong to the set  $A_S(i)$  of admissible values at position  $i$  of  $S$ . Hence,  $z(i)$  also belongs to  $A_S(i)$ . Thus,  $[x, y]$  is contained in  $S$ . Therefore, the schema  $S$  is a convex set.  $\square$

#### 3.2.2. Manhattan distance

We now determine the schemata that are convex sets in the metric space  $M_{d, \text{MD}} = (\{0, 1, \dots, d-1\}^n, \text{MD})$ .

**Lemma 2.** Let  $0 \leq k \leq l \leq d-1$  and let  $[k, l]$  denote the set  $\{k, k+1, \dots, l-1, l\}$ . The only convex schemata of the metric space  $(\{0, 1, \dots, d-1\}^n, \text{MD})$  are those that only use symbols  $*_{[k, l]}$  and/or  $*$  and/or fixed values.

**Proof.** Let  $S$  be a schema in the metric space  $M_{d,MD}$ , whose admissible values at position  $i$  are the elements of a subset  $A_S(i)$  of  $\{0, 1, \dots, d-1\}$  for  $0 \leq i \leq n-1$ .

Let  $x$  and  $y$  be two elements of  $S$ . We determine the conditions under which the segment  $[x, y]$  is contained in  $S$ .

Let  $z \in [x, y]$ , the value of  $z(i)$  belongs to  $[\min\{x(i), y(i)\}, \max\{x(i), y(i)\}]$ . As both  $x$  and  $y$  belong to  $S$ , then  $x(i)$  and  $y(i)$  belong to the set  $A_S(i)$  of admissible values at position  $i$  of  $S$ . Hence,  $z(i)$  belongs to  $A_S(i)$  if:

$$[\min\{x(i), y(i)\}, \max\{x(i), y(i)\}] \subseteq A_S(i). \quad (1)$$

In order to ensure that  $[x, y]$  is contained in  $S$  for any  $x, y \in S$ , we must ensure that the inclusion above holds for any  $x(i), y(i) \in A_S(i)$ . Necessarily,  $A_S(i)$  must be a set of consecutive values such that:

$$A_S(i) = \left[ \min_{x(i), y(i) \in A_S(i)} \{x(i), y(i)\}, \max_{x(i), y(i) \in A_S(i)} \{x(i), y(i)\} \right]. \quad (2)$$

□

### 3.3. Convex evolutionary search algorithm (CS)

The CS uses a multi-parental crossover called convex hull recombination.

**Definition 5** (Convex hull recombination [2]). The (uniform) convex hull recombination returns an offspring sampled uniformly at random from the convex hull formed by its parents.

**Example 8.** Let us consider the elements  $x = 112$ ,  $y = 101$ , and  $z = 022$  of the metric space  $(\{0, 1, 2\}^3, HD)$ . The convex hull of the set  $P' = \{x, y, z\}$  is equal to the schema  $co(P') = *01**12$ . The uniform convex hull recombination consists of sampling an element of  $*01**12$  uniformly at random.

An illustration of the search performed by the CS (called *convex search*) in the Euclidean space is given in Fig. 1. Starting with a population  $P_n$ , a set  $P'_n$  of parents is first selected from the convex hull of  $P_n$  (i.e.,  $co(P_n)$ ). Then, offspring are generated through the convex hull recombination of the set of parents  $P'_n$  (i.e., offspring are sampled uniformly at random from  $co(P'_n)$ ). This yields a new population  $P_{n+1}$ . A pseudo-code corresponding to the CS [2] is shown in Algorithm 1.

---

#### Algorithm 1 Convex evolutionary search algorithm.

---

```

1: Input: population size  $\mu$ 
2: Output: individual in the last population
3: Initialise population uniformly at random
4: while population has not converged to the same individual do
5:   Rank individuals on fitness
6:   if there are at least two fitness values in the current population then
7:     remove all individuals with the worst fitness
8:   end if
9:   Create new population:
10:  for counter in  $\{1, 2, \dots, \mu\}$  do
11:    Apply the CONVEX HULL RECOMBINATION to the remaining
        individuals in the current population to create an individual
12:  end for
13: end while
14: Return any individual in the last population

```

---

We can see that the convex hull formed by the selected individuals forms the set of reachable solutions for the CS.

### 3.4. Quasi-concave landscapes

We expect the CS to perform well on concave landscapes. Hence, we consider a generalisation across representations of quasi-concave

functions on continuous domain to combinatorial spaces, called quasi-concave landscapes [2].

**Definition 6** (Canonical fitness level set [2]). Let  $S$  denote the search space, and let  $f$  be a fitness function on  $S$ . The codomain of the fitness function  $f$  is finite with values  $a_0 < a_1 < \dots < a_q$ . The canonical level set  $A_{\geq j}$  is defined for  $0 \leq j \leq q$  as  $\{x \in S \mid f(x) \geq a_j\}$ .

This definition is different from Wegener's [26], as Wegener's level set corresponds to  $A_j = A_{\geq j} \setminus A_{\geq j+1} = \{x \in S \mid f(x) = a_j\}$ .

**Example 9.** Let LO be the pseudo-Boolean function returning the number of leading ones in a binary string of length  $n$ . A canonical level set  $A_{\geq j}$  of LO is given by:

$$A_{\geq j} = \{x \in \{0, 1\}^n \mid LO(x) \geq j\}, \quad (3)$$

$$= \underbrace{111 \dots 1}_{j \text{ times}} \underbrace{*** \dots *}_{n-j \text{ times}}, \quad (4)$$

where  $0 \leq j \leq n$ .

**Definition 7** (Quasi-concave Landscape [2]). A problem belongs to the class of quasi-concave problems iff all its canonical level sets are convex sets.

Equivalently, a problem  $f : S \rightarrow \mathbb{R}$  belongs to the class of quasi-concave problems iff for all subsets  $C$  of  $S$ , we have:

$$f(z) \geq \min_{x \in C} f(x), \quad (5)$$

for any  $z \in co(C)$  [2].

**Example 10.** LO belongs to the class of quasi-concave problems with respect to the Hamming distance [2]. Indeed, its canonical level sets are convex sets with respect to the Hamming distance (see Example 9 and Lemma 1).

The notion of (geodesic) convexity requires a metric  $D$  on the search space  $S$ . Therefore, the resulting triplet  $(S, f, D)$  forms a fitness landscape [22]. A quasi-concave fitness landscape has two parameters  $q$  and  $r$  that have been introduced in [2]. They are defined as follows:

- $q$  is the index of the smallest canonical level set. As the largest canonical level set is denoted  $A_{\geq 0}$ ,  $q+1$  is therefore the number of distinct canonical level sets.
- $r$  is the smallest ratio between the sizes of two consecutive canonical level sets:

$$r = \min_{0 \leq j \leq q} \frac{|A_{\geq j+1}|}{|A_{\geq j}|}. \quad (6)$$

**Example 11.** The parameters of the quasi-concave landscape  $(\{0, 1\}^n, LO, HD)$  are:

- $q = n$ ,
- $r = 0.5$ .

Indeed, the smallest canonical level set is  $A_{\geq n} = \underbrace{*** \dots *}_{n \text{ times}}$  and any two consecutive canonical level sets only differ by one 'don't care' symbol  $*$ . The quasi-concave landscape  $(\{0, 1\}^n, LO, HD)$  is said to be polynomial (in  $n$ ) because both of its parameters are at most polynomial (in  $n$ ).

We now define fitness functions on the space  $\{0, 1, \dots, d-1\}^n$  that yield quasi-concave landscapes for the Hamming and the Manhattan distances.

**Definition 8.** Let  $a$  be a fixed string of  $\{0, 1, \dots, d-1\}^n$ , the fitness function  $SX_a : \{0, 1, \dots, d-1\}^n \rightarrow \mathbb{R}^+$  is defined as follows: the fitness  $SX_a(b)$  of a string  $b$  is given by the length of the longest suffix of  $a$  that is also a suffix of  $b$ .



**Example 12.** In  $\{0, 1, 2\}^4$ , let  $a = 2021$  and let  $b = 1021$ . The longest suffix of  $a$  that is also a suffix of  $b$  is 021. Hence, the fitness  $SX_a(b)$  of  $b$  is 3.

**Proposition 1.** Let  $a$  be a fixed string of  $\{0, 1, \dots, d-1\}^n$ . The fitness landscape  $(\{0, 1, \dots, d-1\}^n, SX_a, MD)$  (resp.  $(\{0, 1, \dots, d-1\}^n, SX_a, HD)$ ) is quasi-concave with parameters  $q = n$  and  $r = \frac{1}{d}$ .

**Proof.** It is enough to define the canonical level sets of the problem and to show that they are convex sets in the metric space  $(\{0, 1, \dots, d-1\}^n, MD)$ . Let  $a = a_1 a_2 \dots a_n$ , the possible lengths of a suffix of  $a$  are:  $n, n-1, \dots, 2, 1$ , and 0.

Let  $A_{\geq j}$  be the canonical level set containing all strings whose fitness value is at least  $j$ . This means that an element of  $A_{\geq j}$  is of the form:

$$a'_1 \dots a'_{n-j} a_{n-j+1} \dots a_{n-1} a_n, \quad (7)$$

where  $1 \leq j \leq n$ . We have the following results:

$$A_{\geq 0} = ** \dots *$$

$$A_{\geq 1} = ** \dots * a_n$$

$$A_{\geq 2} = ** \dots * a_{n-1} a_n$$

...

$$A_{\geq k} = ** \dots * a_{n-k+1} \dots a_n$$

...

$$A_{\geq n} = \{a_1 a_2 \dots a_n\}$$

By Lemma 2, we know that each canonical level set is a convex set in the metric space  $(\{0, 1, \dots, d-1\}^n, MD)$ . Moreover,  $A_{\geq j+1}$  is always contained in  $A_{\geq j}$  by construction. The number  $q+1$  of distinct level sets is therefore  $n+1$ . The smallest ratio  $r$  between the sizes of two consecutive canonical level sets is  $\frac{1}{d}$ . The same reasoning is used along Lemma 1 for the metric space  $(\{0, 1, \dots, d-1\}^n, HD)$ .  $\square$

**Remark 1.** Let  $a$  be a fixed string of  $\{0, 1, \dots, d-1\}^n$ . For any string  $b$  of  $\{0, 1, \dots, d-1\}^n$ , let  $PX_a(b)$  be the length of the longest prefix of  $a$  that is also a prefix of  $b$ . Using the same reasoning as above, we also find that the fitness landscape  $(\{0, 1, \dots, d-1\}^n, PX_a, MD)$  (resp.  $(\{0, 1, \dots, d-1\}^n, PX_a, HD)$ ) is quasi-concave. In particular, for  $d=2$  and  $a=11\dots 1$  the fitness function  $PX_a$  is Leading Ones.

### 3.5. Runtime analysis of the CS

In [2], an upper bound on the runtime of the CS on a quasi-concave landscape of parameters  $q$  and  $r$  is estimated through the fitness levels method [26]. Each of the  $q+1$  canonical level sets of the quasi-concave landscapes are assumed to be visited once. Moreover, the set of reachable solutions is assumed to coincide with the smallest canonical level set containing it. In particular, if  $P'$  is a population corresponding to the set of remaining individuals after selection then  $co(P')$  coincide with a level set  $A_{\geq j}$ . The expected number of improving offspring (i.e., elements of  $co(P')$  belonging to the level set  $A_{\geq j+1}$ ) for a population size  $\mu$  is bounded below by:

$$\mu \min_{0 \leq j \leq q-1} \frac{|A_{\geq j+1}|}{|A_{\geq j}|} \geq \frac{\mu r}{4}. \quad (8)$$

We recall that  $P_{(S,D)}^{Cov}(m)$  denotes the probability that the convex hull (with respect to the metric  $D$ ) of  $m$  elements sampled uniformly at random from  $S$  covers  $S$ :

$$P_{(S,D)}^{Cov}(m) = \Pr[co(P') = S \mid P' = \text{Unif}_m(S)]. \quad (9)$$

The probability  $P_{(S,D)}^{Cov}(m)$  can simply be written  $P_S^{Cov}(m)$  when it is clear from the context what metric  $D$  is considered.

We also recall the main runtime result of [2] on the analysis of the CS on a quasi-concave landscape of parameters  $q$  and  $r$  in a metric space  $(S, D)$ .

**Theorem 1.** [2] The CS with population size  $\mu$  finds a global optimum within  $q$  generations and  $\mu q$  fitness evaluations with probability at least

$$\left[ P_S^{Cov}\left(\frac{\mu r}{4}\right) \right]^{q+1} - q \exp\left(-\frac{9\mu r}{32}\right). \quad (10)$$

The runtime result of Theorem 1 is representation-free. Indeed, the probability  $P_S^{Cov}\left(\frac{\mu r}{4}\right)$  can only be computed for a specific representation. The runtime result of Theorem 1 has therefore been instantiated to strings on a finite alphabets for the Hamming and the Manhattan distances in [2], by specifically computing the probability above. We consider  $d$ -ary strings of length  $n$  on the alphabet  $\{0, 1, 2, \dots, d-1\}$ .

#### 3.5.1. Hamming distance

The probability  $P_{M_{d,HD}}^{Cov}(m)$  is the probability that the schema matching all the  $m$  elements of  $P'$  with respect to the Hamming distance is

$$\underbrace{*** \dots *}_{n \text{ times}}$$

**Lemma 3.** [2] We assume that  $d \geq 2$ , for any convex set  $C$  of the metric space  $M_{d,HD}$  we have  $P_C^{Cov}(m) \geq P_{M_{d,HD}}^{Cov}(m)$  where,

$$P_{M_{d,HD}}^{Cov}(m) \geq 1 - dn \left(1 - \frac{1}{d}\right)^m. \quad (11)$$

A lower bound on the population size for which the success probability is at least 0.5 has been estimated in [2] using Theorem 1. The formula shown below is adapted from the formula of Theorem 11 and the formula of Corollary 12 of [2], where  $q+2$  should read  $2q+1$ .

**Theorem 2.** [2] Let  $d \geq 2$ , if the population size  $\mu$  is at least:

$$\frac{4d}{r} \ln[2dn(2q+1)], \quad (12)$$

then the CS finds a global optimum on a quasi-concave landscape on the metric space  $M_{d,HD}$  with probability at least 0.5 within  $\mu q$  fitness evaluations.

#### 3.5.2. Manhattan distance

The probability  $P_{M_{d,MD}}^{Cov}(m)$  is the probability that the schema matching all the  $m$  elements of  $P'$  with respect to the Manhattan distance is

$$\underbrace{*** \dots *}_{n \text{ times}}$$

**Lemma 4.** [2] We assume that  $d \geq 2$ , for any convex set  $C$  of the metric space  $M_{d,MD}$  we have  $P_C^{Cov}(m) \geq P_{M_{d,MD}}^{Cov}(m)$  where,

$$P_{M_{d,MD}}^{Cov}(m) \geq 1 - 2n \left(1 - \frac{1}{d}\right)^m. \quad (13)$$

A lower bound on the population size for which the success probability is at least 0.5 has been estimated in [2] using Theorem 1. The formula shown below is adapted from the formula of Theorem 14 and the formula of Corollary 15 of [2], where  $q+2$  should read  $2q+1$ .

**Theorem 3.** [2] Let  $d \geq 2$ , if the population size  $\mu$  is at least:

$$\frac{4d}{r} \ln[4n(2q+1)], \quad (14)$$

then the CS finds a global optimum on a quasi-concave landscape on the metric space  $M_{d,MD}$  with probability at least 0.5 within  $\mu q$  fitness evaluations.

## 4. Segmentwise evolutionary search algorithm

Despite generalizing EAs with no mutation across representations, the CS does not faithfully describe EAs with no mutation that perform a standard two-parents crossover. Indeed, the crossover operator of the CS can use more than two parents. Hence, we define a generalization of EAs with no mutation across representations whose crossover operator faithfully describes a standard two-parents crossover.

The notion of formal evolutionary algorithm with geometric crossover was first introduced in [3]. The Segmentwise Evolutionary Search Algorithm (SESS) is a specific type of formal EAs shown in Algorithm 2. We start by defining the search operator used by the SES.

---

**Algorithm 2** Segmentwise Evolutionary Search Algorithm.
 

---

```

1: Input: population size  $\mu$ 
2: Output: individual in the last population
3: Initialise population uniformly at random
4: while population has not converged to the same individual do
5:   Rank individuals on fitness
6:   if there are at least two fitness values in the current population
       then
7:     remove all individuals with the worst fitness
8:   end if
9:   Create new population:
10:  for counter in  $\{1, 2, \dots, \mu\}$  do
11:    Randomly and uniformly pick two individuals from the re-
        maining individuals in the current population
12:    Recombine them through GEOMETRIC CROSSOVER to create a
        new individual
13:  end for
14: end while
15: Return any individual in the last population
  
```

---

**Definition 9** (geometric crossover [4]). The (uniform) geometric crossover returns an offspring sampled uniformly at random from the segment formed by its two parents.

**Example 13.** Let us consider the elements  $x = 010$  and  $y = 110$  of the metric space  $(\{0, 1\}^3, \text{HD})$ . The segment  $[x, y]$  is equal to the schema  $*10$ . The geometric crossover of the elements  $x$  and  $y$  consists of sampling an element of  $*10 = \{010, 110\}$  uniformly at random.

Lines 11 and 12 of Algorithm 2 tell us that a pair of individuals is sampled uniformly at random out of the set of all possible pairs of selected individuals. This means that the distribution of the pairs of selected individuals is uniform on the set of selected individuals.

Offspring are sampled uniformly at random from a segment. As the notion of segment can be defined for any representation, the SES is representation independent.

#### 4.1. Offspring distribution

We shall now determine an analytical formula describing the offspring distribution. This will be useful for the runtime analysis of the SES.

Let us denote  $P'$  the set of parents that are selected from a population  $P$ . The set of reachable solutions  $R(P')$  from the set of parents  $P'$  is the set of solutions that can be reached by repeated application of a search operator to the set of parents  $P'$ . In particular, the set  $R(P')$  of reachable solutions for the geometric crossover is the union of all the segments that can be formed out of the elements of  $P'$ . When  $P'$  is a subset of a metric space  $(S, D)$ , then:

$$R(P') = \bigcup_{x, y \in P'} [x, y]_D. \quad (15)$$

To ease the notation, the set  $\bigcup_{x, y \in P'} [x, y]_D$  is denoted  $\text{Seg}(P')$ .

**Example 14.** In the two-dimensional Hamming space  $(\{0, 1\}^2, \text{HD})$ , let us consider the subset  $A = \{00, 01, 11\}$ . The set  $\text{Seg}(A)$  is the union of the segments  $[00, 00], [01, 01], [11, 11], [00, 01], [00, 11]$  and  $[01, 11]$ . Hence,  $\text{Seg}(A) = \{0, 1\}^2$ .

The probability distribution of the reachable solutions need not be uniform on  $R(P')$ . Indeed, if  $x$  and  $y$  are elements of  $P'$  then the probability for sampling an offspring in the segment  $[x, y]$  is  $\frac{1}{|[x, y]|}$ . Let  $\alpha_{s, P'}$  be

the number of pairs of elements of  $P'$  yielding the segment  $s$ . The total number of pairs that can be formed out of the elements of  $P'$  is  $|P'|^2$  and each pair has probability

$$\frac{1}{|P'|^2} \quad (16)$$

to be sampled. This is not the case for segments. Indeed, two distinct pairs may form the same segment. Hence, the probability for sampling the segment  $s$  is not uniform and is given by:

$$\frac{\alpha_{s, P'}}{|P'|^2}. \quad (17)$$

**Example 15.** In  $(\{0, 1\}^2, \text{HD})$ , let  $P' = \{00, 01, 10, 11\}$ . The probability for sampling a pair of elements of  $P'$  is uniform and is equal to  $\frac{1}{4^2}$ .

- Let  $s$  be the segment  $[00, 11]$ . We have  $\alpha_{s, P'} = 4$ . Indeed,

$$s = [00, 11] = [11, 00] = [01, 10] = [10, 01]. \quad (18)$$

The probability for sampling the segment  $s$  is therefore  $\frac{4}{4^2} = \frac{1}{4}$ .

- Let  $s_{00}$  be the segment  $[00, 00]$ , we have  $\alpha_{s_{00}, P'} = 1$ . The probability for sampling  $s_{00}$  is therefore  $\frac{1}{4^2}$ . This is also the probability for sampling each of the segments  $s_{01}$ ,  $s_{10}$  and  $s_{11}$ .

- Let  $s_{\{00, 10\}}$  be the segment  $[00, 10]$ , we have  $\alpha_{s_{\{00, 10\}}, P'} = 2$  as:

$$s_{\{00, 10\}} = [10, 00]. \quad (19)$$

The probability for sampling the segment  $s_{\{00, 10\}}$  is  $\frac{2}{4^2}$ . This is also the probability for sampling each of the segments  $s_{\{00, 01\}}$ ,  $s_{\{10, 11\}}$ , and  $s_{\{01, 11\}}$ .

There are 9 distinct segments that can be formed out of the elements of  $P'$ . Those segments are:

$s$ ,  $s_{00}$ ,  $s_{01}$ ,  $s_{10}$ ,  $s_{11}$ ,  $s_{\{00, 10\}}$ ,  $s_{\{00, 01\}}$ ,  $s_{\{10, 11\}}$ , and  $s_{\{01, 11\}}$ . We obtain one by adding up the probabilities for sampling each one of them. In particular, we have:

$$\text{Seg}(P') = s \cup s_{00} \cup s_{01} \cup s_{10} \cup s_{11} \cup s_{\{00, 10\}} \cup s_{\{00, 01\}} \cup s_{\{10, 11\}} \cup s_{\{01, 11\}}. \quad (20)$$

More generally, the set  $\text{Seg}(P')$  can be rewritten as the union of the distinct segments that can be formed out of the elements of  $P'$ . That is, there exists  $p \leq |P'|^2$  such that:

$$\text{Seg}(P') = \bigcup_{i=1}^p s_i. \quad (21)$$

As  $s_1, s_2, \dots, s_p$  are the only segments that can be formed out of the elements of  $P'$ , we have:

$$\sum_{i=1}^p \frac{\alpha_{s_i, P'}}{|P'|^2} = 1. \quad (22)$$

**Theorem 4.** Let  $z$  be a reachable solution and let  $\mathbb{1}_s$  be the indicator function on the segment  $s$ . The probability for sampling  $z$  is given by:

$$\Pr(z \in \text{Seg}(P')) = \sum_{i=1}^p \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{\mathbb{1}_{s_i}(z)}{|s_i|}. \quad (23)$$

We also have:

$$\begin{aligned} \sum_{z \in \text{Seg}(P')} \Pr(z \in \text{Seg}(P')) &= \sum_{z \in \text{Seg}(P')} \sum_{i=1}^p \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{\mathbb{1}_{s_i}(z)}{|s_i|}, \\ &= 1. \end{aligned} \quad (24)$$

**Proof.** By construction. Indeed,  $\text{Seg}(P')$  is the set of reachable solutions from  $P'$  and  $\text{Seg}(P')$  is the union of all segments of  $P'$  (i.e., segments whose extremes are elements of  $P'$ ). Algorithm 2 tells us that:

- the probability for sampling any pair of elements of  $P'$  is uniform (i.e., is the same for any pair of elements of  $P'$ ),

- the probability for sampling an offspring in a segment of  $P'$  is uniform (i.e., is the same for any element of that segment).

The result follows as the sum of the probabilities of all outcomes must equal one.  $\square$

This theorem gives us an explicit formula for the non-uniform offspring distribution on the set  $Seg(P')$ .

## 5. SES versus CS

In order to extend the runtime analysis of the CS to the SES, we compare the set of offspring sampled by the search operator of the CS to the set of offspring sampled by the search operator of the SES for the same population of selected individuals. Let  $A$  denote the set corresponding to the population of selected individuals. The convex hull  $co(A)$  of  $A$  corresponds to the set of offspring sampled by the search operator of the CS [2]. Whereas, the union  $Seg(A)$  of all the segments that can be formed out of the elements of  $A$  corresponds to the set of offspring sampled by the search operator of the SES (see Section 4.1).

### 5.1. Relationship between $Seg(A)$ and $co(A)$

Let  $X$  be a subset in a metric space  $(S, D)$ . We recall that  $Seg(X)$  is the union of all segments that can be formed out of the elements of  $X$ . Proposition 4.1.2. of [38] gives the following result for any subset  $A$  of a metric space  $(S, D)$ :

$$co(A) = A \cup Seg(A) \cup Seg(Seg(A)) \cup \dots \cup Seg(\dots(Seg(A))\dots) \cup \dots$$

This implies the following proposition:

**Proposition 2.** *Let  $(S, D)$  be a metric space and let  $A$  be a subset of  $S$ . The set  $Seg(A)$  is always included in the set  $co(A)$ .*

Besides the proof of Proposition 4.1.2. given in [38], an alternative proof for Proposition 2 is given below for the sake of clarity.

**Proof.** By definition,  $co(A)$  is the smallest convex set containing  $A$ . Hence  $A \subseteq co(A)$ . It follows that  $Seg(A) \subseteq Seg(co(A))$ . Since the set  $co(A)$  is convex, all segments whose extremes are points of  $co(A)$  are included in  $co(A)$ . Therefore,  $Seg(co(A)) \subseteq co(A)$ . Thus,  $Seg(A) \subseteq co(A)$ .  $\square$

**Lemma 5.** *Let  $(S, D)$  be a metric space and let  $A$  be a subset of  $S$ , we have  $co(Seg(A)) = co(A)$ .*

**Proof.** On the one hand, we have  $Seg(A) \subseteq co(A)$  by Proposition 2. This implies that  $co(Seg(A)) \subseteq co(co(A)) = co(A)$ . On the other hand,  $A \subseteq Seg(A) \subseteq co(Seg(A))$ . As a result,  $co(Seg(A))$  is a smaller convex set that contains  $A$ . Necessarily,  $co(Seg(A)) = co(A)$ .  $\square$

**Theorem 5.** *Let  $(S, D)$  be a metric space and let  $A$  be a subset of  $S$ . The following statements are equivalent:*

1.  $Seg(A)$  is a convex set,
2.  $Seg(A) = co(A)$ .

**Proof.** On the one hand, if  $Seg(A) = co(A)$  then the set  $Seg(A)$  is convex as  $co(A)$  is. On the other hand, if  $Seg(A)$  is a convex set then  $Seg(A) = co(Seg(A))$ . The result follows from Lemma 5.  $\square$

In metric spaces where the sets  $Seg(A)$  and  $co(A)$  coincide for all  $A$ , the SES performs a certain form of convex search. We shall now compare the sets  $Seg(A)$  and  $co(A)$  for all subsets  $A$  in the metric space  $M_{d,HD}$  (resp.  $M_{d,MD}$ ). We already know that  $Seg(A)$  is either:

- strictly contained in  $co(A)$ ,
- or equal to  $co(A)$ ,

for any subset  $A$  (see Proposition 2). We are now interested in finding out whether the sets  $Seg(A)$  and  $co(A)$  are equal.

### 5.1.1. Hamming distance

Recall that the Hamming distance between  $x$  and  $y$  is the number of differing positions between them:

$$HD(x, y) = \sum_{k=1}^n [1 - \delta_{x(k), y(k)}], \quad (26)$$

where  $\delta_{i,j}$  is the Kronecker delta. That is,  $\delta_{i,j} = \begin{cases} 0 & \text{if } i \neq j, \\ 1 & \text{otherwise.} \end{cases}$

We show that  $Seg(A) = co(A)$  for any subset  $A$  of the metric space  $M_{2,HD} = (\{0, 1\}^n, HD)$ . We also show that  $Seg(A)$  need not be equal to  $co(A)$  for any subset  $A$  of the metric space  $M_{d,HD} = (\{0, 1, \dots, d-1\}^n, HD)$  where  $d > 2$ .

The following proposition can be easily derived from [38].

**Proposition 3.** *Any segment of the metric space  $M_{d,HD} = (\{0, 1, \dots, d-1\}^n, HD)$  is a convex set.*

**Proof.** Any segment of the  $n$ -dimensional Hamming space  $(\{0, 1, \dots, d-1\}^n, HD)$  is the Cartesian product of  $n$  segments of the one-dimensional space  $(\{0, 1, \dots, d-1\}, HD)$  [38]. A segment of  $(\{0, 1, \dots, d-1\}, HD)$  is either a single element or the union of two distinct elements. Hence, a segment of  $(\{0, 1, \dots, d-1\}, HD)$  is always a convex set. Since a Cartesian product of convex sets remains convex [38], any segment of the  $n$ -dimensional Hamming space  $(\{0, 1, \dots, d-1\}^n, HD)$  is also a convex set.  $\square$

Let  $A$  be a set in the metric space  $M_{d,HD} = (\{0, 1, \dots, d-1\}^n, HD)$  and let  $s_1, s_2, \dots, s_p$  be the  $p$  distinct segments that can be formed out of the elements of  $A$ :

$$Seg(A) = \bigcup_{1 \leq j \leq p} s_j. \quad (27)$$

A segment  $s_j = [x_j, y_j]$  corresponds to the schema  $*_{A_{s_j(1)}} *_{A_{s_j(2)}} \dots *_{A_{s_j(n)}}$ , where  $A_{s_j(i)} = \{x_j(i), y_j(i)\}$  is the set of admissible values at position  $i$ . We also recall that in the schema corresponding to  $co(A)$ , the admissible values at position  $i$  are the elements of  $\bigcup_{1 \leq j \leq p} A_{s_j(i)}$ . We have the following result:

**Lemma 6.** *The union  $Seg(A)$  of all the segments that can be formed out of the elements of  $A$  is a convex set if there exists  $1 \leq j \leq p$  such that:*

$$\bigcup_{1 \leq j \leq p} A_{s_j(i)} = A_{s_j(i)}, \quad (28)$$

at each position  $i$ .

**Proof.** If Eq. (28) is satisfied then there exists a segment  $s_j$  such that  $co(A) = s_j$ . Consequently,  $co(A)$  is contained in  $Seg(A) = \bigcup_{1 \leq j \leq p} s_j$ . Therefore, the sets  $co(A)$  and  $Seg(A)$  are necessarily equal.  $\square$

**Corollary 1.** *In the metric space  $M_{d,HD}$ , the set  $Seg(A)$ :*

- is always convex for any subset  $A$  when  $d = 2$ ,
- need not be convex for any subset  $A$  when  $d > 2$

**Proof.** In  $M_{2,HD}$ , the set  $\bigcup_{1 \leq j \leq p} A_{s_j(i)}$  contains either one or two elements. In both cases, we have:

$$\bigcup_{1 \leq j \leq p} A_{s_j(i)} = \{x_j(i)\} \cup \{y_j(i)\}, \quad (29)$$

$$= A_{s_j(i)}. \quad (30)$$

Consequently, the set  $Seg(A)$  is always convex in  $M_{2,HD}$ .

When  $d > 2$ , the set  $\bigcup_{1 \leq j \leq p} A_{s_j(i)}$  may contain more than two elements. In this case, it can not correspond to a set  $A_{s_j(i)}$ . As a result, the set  $Seg(A)$  need not be convex in  $M_{d,HD}$  when  $d > 2$ .  $\square$

**Example 16.** In the metric space  $(\{0, 1, 2\}^4, HD)$ , let  $A = \{0012, 2110, 2011\}$ . We have that:

$$Seg(A) = [0012, 2110] \cup [0012, 2011] \cup [2110, 2011], \quad (31)$$

$$=*_{02} *_{01} 1 *_{02} \cup *_{02} 01 *_{12} \cup 2 *_{01} 1 *_{01}, \quad (32)$$

and  $co(A) = *_{02} *_{01} 1 *$ . We can see that  $0111 \in co(A)$  but  $0111 \notin Seg(A)$ . Hence,  $Seg(A) \subsetneq co(A)$ .

### 5.1.2. Manhattan distance

Recall that the Manhattan distance between  $x$  and  $y$  is

$$MD(x, y) = \sum_{k=1}^n |x(k) - y(k)|. \quad (33)$$

We show that  $Seg(A) = co(A)$  for any subset  $A$  in the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$ .

The following proposition can be easily derived from [38].

**Proposition 4.** Any segment in the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$  is a convex set.

**Proof.** Any segment of the  $n$ -dimensional Manhattan space  $(\{0, 1, \dots, d-1\}^n, MD)$  is the Cartesian product of  $n$  segments of the one-dimensional space  $(\{0, 1, \dots, d-1\}, MD)$  [38]. A segment of  $(\{0, 1, \dots, d-1\}, MD)$  is either a single element, two consecutive elements, three consecutive elements, ..., or  $d$  consecutive elements. Hence, a segment of  $(\{0, 1, \dots, d-1\}, MD)$  is always a convex set. Since a Cartesian product of convex sets remains convex [38], any segment of the  $n$ -dimensional Manhattan space  $(\{0, 1, \dots, d-1\}^n, MD)$  is also a convex set.  $\square$

Let  $A$  be a set in the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$  and let  $s_1, s_2, \dots, s_p$  be the  $p$  distinct segments that can be formed out of the elements of  $A$ :

$$Seg(A) = \bigcup_{1 \leq j \leq p} s_j. \quad (34)$$

A segment  $s_j = [x_j, y_j]$  corresponds to the schema  $*_{A_{s_j}(1)} *_{A_{s_j}(2)} \dots *_{A_{s_j}(n)}$  where:

$$A_{s_j}(i) = [\min\{x_j(i), y_j(i)\}, \max\{x_j(i), y_j(i)\}], \quad (35)$$

is the set of admissible values at position  $i$ . We also recall that in the schema corresponding to  $co(A)$ , the admissible values at position  $i$  are the elements of  $[\min_{x,y \in A} \{x(i), y(i)\}, \max_{x,y \in A} \{x(i), y(i)\}]$ . We have the following result:

**Lemma 7.** In the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$ , the set  $Seg(A)$  is convex for any subset  $A$ .

**Proof.** In the schema corresponding to  $co(A)$ , the admissible values at position  $i$  are the elements of

$$[\min_{x,y \in A} \{x(i), y(i)\}, \max_{x,y \in A} \{x(i), y(i)\}]. \quad (36)$$

This means that there exists a segment  $s_j$  such that  $co(A) = s_j$ . As a result,  $Seg(A) = co(A)$  and is therefore a convex set.  $\square$

**Theorem 6.** Let  $A$  be a set, the union  $Seg(A)$  of all the segments that can be formed out of the elements of  $A$  is equal to the convex hull  $co(A)$  of  $A$  in the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$ .

**Proof.** Since the set  $Seg(A)$  is convex, it is equal to the set  $co(A)$  by Theorem 5.  $\square$

**Example 17.** In the metric space  $(\{0, 1, 2\}^4, MD)$ , let  $A = \{0012, 2110, 2011\}$ . We have that:

$$Seg(A) = [0012, 2110] \cup [0012, 2011] \cup [2110, 2011], \quad (37)$$

$$=*_{01} 1 * \cup *_{01} 1 *_{12} \cup 2 *_{01} 1 *_{01}, \quad (38)$$

and  $co(A) = *_{01} 1 *$ . We can see that  $co(A) = [0012, 2110]$  and  $Seg(A) = co(A)$ .

We restrict our study to metric spaces where the sets  $Seg(A)$  and  $co(A)$  coincide for all  $A$ . In these metric spaces, the runtime of the SES on quasi-concave landscapes can be analyzed by using an approach similar to that used for the CS in [2].

### 5.2. Expected number of improving offspring

The expected number of improving offspring is needed to compute the probability for covering a canonical level set for the SES. Let  $P'$  be a population corresponding to the set of remaining individuals after selection. The offspring are not uniformly distributed on  $Seg(P')$  as seen in Eq. (23). As our study is restricted to metric spaces where the sets  $Seg(A)$  and  $co(A)$  coincide for any subset  $A$ , this means that offspring are not uniformly distributed on  $co(P')$  as for the CS. Hence, when  $co(P')$  coincides with a level set  $A_{\geq j}$  then:

- the probability for sampling an offspring belonging to  $A_{\geq j+1}$  is no longer equal to  $\frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ ,
- the expected number of offspring belonging to  $A_{\geq j+1}$  for a population size of  $\mu$  is no longer equal to  $\mu \frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ .

Instead, pairs are sampled uniformly at random from the set of all possible pairs that can be made out of the elements of  $P'$ . The probability for sampling a pair belonging to  $A_{\geq j+1}$  when  $co(P')$  coincides with a level set  $A_{\geq j}$  is:

$$\left( \frac{|A_{\geq j+1}|}{|A_{\geq j}|} \right)^2. \quad (39)$$

We also need to estimate the probability that the segment formed by this pair is strictly contained in  $A_{\geq j+1}$  when  $co(P')$  coincides with a level set  $A_{\geq j}$ . The latter is given by the ratio of segments of  $P'$  that are strictly contained in  $co(P')$ .

#### 5.2.1. Ratio of segments of $A$ strictly contained in $co(A)$

Let  $A$  be a finite set in a discrete metric space. A segment whose extremes are elements of  $A$  is referred to as a *segment of  $A$* . We aim to compute a lower bound on the probability for sampling a pair of elements of  $A$  forming a segment that is strictly included in  $co(A)$  when pairs are uniformly distributed. To this end, we first estimate the ratio of segments of  $A$  covering its convex hull  $co(A)$ .

**Lemma 8.** The ratio of segments of  $A$  equating  $co(A)$  is bounded above by  $1/3$  whenever  $A$  contains at least two distinct elements.

**Proof.** Let  $[x_1, y_1]$  and  $[x_2, y_2]$  be two segments of  $A$  equating  $co(A)$ .

- We show that if two segments equating  $co(A)$  share an endpoint then they must share the other endpoint. If  $x_1 = x_2$  and  $y_1 \neq y_2$ , then the segment  $[y_1, y_2]$  is included in  $co(A)$ . This is because both  $y_i$  belong to  $co(A)$  and  $co(A)$  is a convex set. We have,

$$[x_1, y_1] = [x_1, y_2] \text{ and } [y_1, y_2] \subseteq [x_1, y_1]. \quad (40)$$

Therefore,  $y_2 \in [x_1, y_1]$ . As  $[x_1, y_1] = [x_1, y_2]$ , then  $y_2$  must be equal to  $y_1$ . This contradicts the initial assumption. Therefore, whenever  $x_1 = x_2$  then  $y_1 = y_2$  when  $[x_1, y_1]$  and  $[x_2, y_2]$  are both equal to  $co(A)$ .

- Let  $x_1 \neq x_2, y_2$  and  $y_1 \neq x_2, y_2$ . The segments  $[x_1, y_1]$  and  $[x_2, y_2]$  do not share any endpoint (though they may be equal in some specific metric spaces). We show that if  $[x_1, y_1]$  and  $[x_2, y_2]$  are both equal to  $co(A)$ , then the segments  $[x_1, x_2]$ ,  $[y_1, y_2]$ ,  $[x_1, y_2]$ , and  $[x_2, y_1]$  are not equal to  $co(A)$ . Without loss of generality, let us show that  $[x_1, x_2]$  is not equal to  $co(A)$ . Let us assume that  $[x_1, x_2] = co(A)$ . We have,

$$[x_1, x_2] = [x_1, y_1], \quad (41)$$

because  $[x_1, y_1] = co(A)$  by assumption. This implies that  $x_1 = y_2$ . Since,  $[x_1, y_1] = [x_2, y_2]$  we have:

$$[y_2, y_1] = [x_2, y_2], \quad (42)$$



by replacing  $x_1$  by  $y_2$  in the left hand side. Consequently,  $x_2 = y_1$ . As a result we have:

$$\begin{cases} x_1 = y_2, \\ x_2 = y_1, \end{cases} \quad (43)$$

which contradicts the initial assumption. Therefore,  $[x_1, x_2]$  is not equal to  $co(A)$ .

We conclude that whenever two segments of  $A$  that do not share endpoints are both equal to  $co(A)$ , there exist at least four segments of  $A$  that are not equal to  $co(A)$ . Thus, the ratio of segments of  $A$  equating  $co(A)$  is at most  $\frac{2}{2+4} = \frac{1}{3}$ .

If  $A = \{x_1, x_2, x_3\}$  where the three elements are distinct, and the segment  $[x_1, x_2]$  is equal to  $co(A)$ , then the segments  $[x_2, x_3]$  and  $[x_3, x_1]$  can not be equal to  $co(A)$ . Indeed, if they were they would be equal to  $[x_1, x_2]$  and  $A$  would only contain two distinct elements instead of three. Hence, the ratio of segments of  $A$  equating  $co(A)$  is at most  $\frac{1}{3}$ .

If  $A = \{x_1, x_2\}$  where the two elements are distinct, and the segment  $[x_1, x_2]$  is equal to  $co(A)$ , then the segments  $[x_1, x_1] = \{x_1\}$  and  $[x_2, x_2] = \{x_2\}$  can not be equal to  $co(A)$ . Hence, the ratio of segments of  $A$  equating  $co(A)$  is  $\frac{1}{3}$ .  $\square$

In the SES, pairs of parents are sampled uniformly at random from the selected population  $P'$ . This means that pairs of elements of  $P'$  are uniformly distributed on the set of reachable solutions  $Seg(P')$ . On metric spaces where  $Seg(P')$  coincides with  $co(P')$ , pairs of elements of  $P'$  are therefore uniformly distributed on  $co(P')$ . We can therefore use the uniform distribution of the pairs on the convex hull of the union of all possible pairs, for the analysis of the SES on such metric spaces.

**Theorem 7.** *We assume that the pairs of elements of  $A$  are uniformly distributed on  $co(A)$ . If  $A$  contains at least two distinct elements, then the probability for sampling a pair of elements of  $A$  forming a segment equating  $co(A)$  is bounded above by  $1/3$ .*

**Proof.** The probability for sampling a pair of elements of  $A$  forming a segment that is equal to  $co(A)$  is the ratio of segments of  $A$  equating  $co(A)$ . By Lemma 8, this ratio is bounded above by  $1/3$ . The result follows.  $\square$

**Corollary 2.** *We assume that the pairs of elements of  $A$  are uniformly distributed on  $co(A)$ . If  $A$  contains at least two distinct elements, then the probability for sampling a segment of  $A$  that is strictly included in  $co(A)$  is bounded below by  $2/3$ .*

**Proof.** This is the complementary of the event of sampling a segment of  $A$  equating  $co(A)$ . As the probability of its complementary is at most  $1/3$ , its probability is at least  $1 - 1/3 = 2/3$ .  $\square$

### 5.2.2. Probability for sampling improving solutions

We start by estimating a lower bound on the probability of sampling a strictly improving offspring in a quasi-concave landscape of parameters  $q$  and  $r$ . We recall that:

$$r = \min_{0 \leq j \leq q-1} \left( \frac{|A_{\geq j+1}|}{|A_{\geq j}|} \right). \quad (44)$$

**Theorem 8.** *The probability for sampling a strictly improving offspring from any selected population with at least two distinct individuals is bounded below by  $2r^2/3$ .*

**Proof.** Let  $P'$  denote the set of selected individuals. We assume that  $P'$  is contained in the canonical level set  $A_{\geq j}$ . The set  $co(P')$  is equal to the level set  $A_{\geq j}$  containing it. In this case, the probability for sampling an offspring belonging to  $A_{\geq j+1}$  (which is a strict subset of  $A_{\geq j}$ ) is given by:

$$\sum_{z \in A_{\geq j+1}} \sum_{i=1}^p \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{\mathbb{1}_{s_i}(z)}{|s_i|} \quad (45)$$

$$= \sum_{i=1}^p \sum_{z \in A_{\geq j+1}} \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{\mathbb{1}_{s_i}(z)}{|s_i|}, \quad (46)$$

$$= \sum_{i=1}^p \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}, \quad (47)$$

$$= \sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|} \quad (48)$$

$$+ \sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}, \quad (49)$$

$$\geq \sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'}}{|P'|^2}. \quad (50)$$

The bound in Inequality (50) is the probability to sample a segment of  $P'$  that is strictly included in  $A_{\geq j} = co(P')$ . By Corollary 2, the probability for sampling a segment of  $P'$  that is strictly included in  $co(P')$  is bounded below by  $2/3$  given that  $Seg(P') = co(P')$  and  $P'$  contains at least two distinct elements. Pairs are sampled uniformly at random from the set of all possible pairs that can be made out of the elements of  $P' \subseteq A_{\geq j}$ . Thus, the probability for sampling a pair that is included in  $A_{\geq j+1}$  is given by:

$$\left( \frac{|A_{\geq j+1}|}{|A_{\geq j}|} \right)^2 \geq \left[ \min_{0 \leq j \leq q-1} \left( \frac{|A_{\geq j+1}|}{|A_{\geq j}|} \right) \right]^2, \quad (51)$$

$$= r^2. \quad (52)$$

Consequently, the probability to sample a segment of  $P'$  that is strictly included in  $A_{\geq j} = co(P')$  is bounded below by  $\frac{2r^2}{3}$  given that  $Seg(P') = co(P')$  and  $P'$  contains at least two distinct elements.  $\square$

**Corollary 3.** *The expected number of strictly improving offspring for a population size of  $\mu$  is at least:*

$$\frac{2r^2\mu}{3}, \quad (53)$$

*if at least two distinct individuals are selected at each generation.*

**Proof.** In the worst case, all strictly improving offspring of the selected population have the same least probability of Theorem 8 to be sampled. The total number of offspring that is created is given by the population size  $\mu$ . Consequently, the expected number of strictly improving offspring among the  $\mu$  offspring is at least  $\frac{2r^2\mu}{3}$ .  $\square$

## 6. Runtime analysis of the SES

We compute an upper bound on the expected runtime of the SES on a quasi-concave landscape in a metric space where  $Seg(A) = co(A)$  for any subset  $A$ . As the SES performs a certain form of convex search in these metric spaces, the analysis used in [2] for the CS can be used as a guideline.

The SES finds a global optimum if the convex hull formed by the selected individuals always covers a higher level set than the one containing them. As level sets form a decreasing chain of sets with respect to the ‘contains’ order (see Definition 6), the condition above is satisfied whenever the convex hull formed by the selected individuals is always equal to the level set containing them. In combinatorial spaces, the latter happens with probability at least 0.5 for a well chosen population size. Indeed, the distribution of the offspring is not uniform on the level set. However, the distribution of pairs of parents is uniform on the level set and each offspring is created from a pair of parents. In this case, the

SES is expected to find a global optimum within  $2q$  generations where  $q + 1$  is the total number of distinct level sets.

Let  $m$  be a positive integer and let  $C$  be a non empty convex set whose elements are distributed as in Eq. (23). The set of  $m$  points drawn from  $C$  is denoted  $\text{NonUnif}_m(C)$ .

**Definition 10.** Let  $C$  be a convex set in a metric space  $(S, D)$  whose elements are distributed as in Eq. (23). The probability that the union of all the segments that can be made out of  $m$  points drawn from  $C$  equals  $C$  is:

$$P_{C,Seg}^{\text{Cov}}(m) = \Pr[Seg(P) = C \mid P = \text{NonUnif}_m(C)]. \quad (54)$$

In metric spaces where  $Seg(A) = co(A)$  for any subset  $A$ , the probability  $P_{C,Seg}^{\text{Cov}}(m)$  is equal to the probability that the convex hull of  $m$  points drawn from  $C$  equals  $C$ . That is:

$$P_{C,Seg}^{\text{Cov}}(m) = \Pr[Seg(P) = C \mid P = \text{NonUnif}_m(C)], \quad (55)$$

$$= \Pr[co(P) = C \mid P = \text{NonUnif}_m(C)], \quad (56)$$

$$\geq \min_{C \in C_S} \Pr[co(P) = C \mid P = \text{NonUnif}_m(C)], \quad (57)$$

where  $C_S$  denotes the set of convex sets on the entire search space  $S$ . Let us denote  $P_{S,Seg}^{\text{Cov}}(m)$  the probability  $\min_{C \in C_S} \Pr[co(P) = C \mid P = \text{NonUnif}_m(C)]$ . As in [2], the probability  $P_{S,Seg}^{\text{Cov}}(m)$  is monotonically increasing in  $m$  because additional samples can only increase the convex hull.

We assume a quasi-concave fitness function on the metric space  $(S, D)$  with fitness levels  $A_{\geq 0}, A_{\geq 1}, \dots, A_{\geq q}$ . Let  $P'_t$  denote the parents of generation  $t$ . The following lemma gives a lower bound on the probability that  $co(P'_{t+1})$  is equal to some  $A_{\geq j}$  given that  $co(P'_t)$  is equal to  $A_{\geq i}$  and  $i < j$ .

**Lemma 9.** The probability that the next generation of parents covers a higher level set than the level set covered by the current generation of parents is at least:

$$P_{S,Seg}^{\text{Cov}}\left(\frac{2r^2\mu}{3}\right) - \exp\left(-\frac{r^2\mu}{18}\right). \quad (58)$$

**Proof.** The probability  $P_{S,Seg}^{\text{Cov}}(m)$  is monotonically increasing in  $m$ . For a population size of  $\mu$ ,  $m$  is at least  $\frac{2r^2\mu}{3}$  by Corollary 3. Hence,  $P_{S,Seg}^{\text{Cov}}\left(\frac{2r^2\mu}{3}\right)$  is a lower bound on  $P_{S,Seg}^{\text{Cov}}(m)$ . Using Chernoff bound [27], the probability that the number of strictly improving offspring is smaller than  $\frac{2r^2\mu}{3}$ , is at most:

$$\Pr\left(\left|P'\right| \leq \frac{2r^2\mu}{3}\right) \leq \exp\left[-\frac{r^2\mu}{2} \cdot \left(\frac{1}{3}\right)^2\right]. \quad (59)$$

We define the worst-case typical behaviour to have exactly  $\frac{2r^2\mu}{3}$  strictly improving offspring in each level set as in Corollary 3.  $\square$

**Theorem 9.** The SES with population size  $\mu$  finds a global optimum within  $q$  generations and  $\mu q$  fitness evaluations with probability at least

$$\left[P_{S,Seg}^{\text{Cov}}\left(\frac{2r^2\mu}{3}\right)\right]^{q+1} - q \exp\left(-\frac{r^2\mu}{18}\right). \quad (60)$$

**Proof.** The reasoning is the same as in [2]. We assume that the probabilities for covering different level sets are independent. Each level set is visited taking into account  $A_{\geq 0}$ . Then, the probability that less than  $\frac{2r^2\mu}{3}$  strictly improving offspring are generated is removed at each step.  $\square$

The next step is to explicitly compute

$$P_{C,Seg}^{\text{Cov}}(m) = \Pr[co(P) = C \mid P = \text{NonUnif}_m(C)] \quad (61)$$

for specific representations.

When the selection of the  $m$  elements of  $C$  is uniform, then they equally contribute to the creation of their convex hull  $co(P)$ . This is for example the case for the CS [2].

When the  $m$  elements of  $C$  are not selected uniformly at random, they need not equally contribute to the creation of their convex hull  $co(P)$ . Each element must contribute at least once in the making of their convex hull  $co(P)$ . It remains to determine the maximum number of contributions. To this end, we introduce the notion of weight to measure the number of contributions of each of the  $m$  elements of  $C$ .

**Definition 11.** Let  $e_1, e_2, \dots, e_m$  be  $m$  samples from a non-empty convex set  $C$ . For each  $e \in C$ , we denote  $p(e)$  the probability to select  $e$ . The weight of the element  $e_i$  of  $C$  is defined as:

$$w_i = p(e_i) \cdot \text{lcd}_{e \in C} p(e), \quad (62)$$

where  $\text{lcd}$  stands for least common denominator.

**Example 18.** Let  $C = *1*$ ,  $C$  is a non-empty convex set of the metric space  $M_{2,HD} = (\{0,1\}^3, HD)$ . The elements of  $C$  are 010, 011, 110, and 111. Since the probability distribution on  $C$  is not uniform, the probabilities  $p(010)$ ,  $p(011)$ ,  $p(110)$ , and  $p(111)$  need not be the same. For the sake of illustration let:

$$\begin{cases} p(010) = \frac{1}{5}, \\ p(011) = \frac{1}{10}, \\ p(110) = \frac{1}{3}, \\ p(111) = \frac{1}{6}. \end{cases} \quad (63)$$

The weights of each of the element of  $C$  are therefore:

$$w(010) = \frac{1}{5} \cdot \text{lcd}\left(\frac{1}{5}, \frac{1}{10}, \frac{1}{3}, \frac{1}{6}\right) = \frac{1}{5} \cdot 30 = 6,$$

$$w(011) = \frac{1}{10} \cdot \text{lcd}\left(\frac{1}{5}, \frac{1}{10}, \frac{1}{3}, \frac{1}{6}\right) = \frac{1}{10} \cdot 30 = 3,$$

$$w(110) = \frac{1}{3} \cdot \text{lcd}\left(\frac{1}{5}, \frac{1}{10}, \frac{1}{3}, \frac{1}{6}\right) = \frac{1}{3} \cdot 30 = 10,$$

$$w(111) = \frac{1}{6} \cdot \text{lcd}\left(\frac{1}{5}, \frac{1}{10}, \frac{1}{3}, \frac{1}{6}\right) = \frac{1}{6} \cdot 30 = 5.$$

Under the non-uniform distribution of Eq. (63), the element 010 contributes up to 6 times in the making of the convex hull  $C$  of 010, 011, 110, and 111.

**Example 19.** For the CS, samples are selected uniformly at random from a non-empty level set  $A_{\geq j}$  [2]. Each of them has the same probability  $p(e) = \frac{1}{|A_{\geq j+1}|}$  to be selected. Indeed, an element  $e \in A_{\geq j}$  is selected if it belongs to  $A_{\geq j+1}$ . Hence,  $p(e)$  is the probability to sample an element  $e$  of  $A_{\geq j}$  given that this element belongs to  $A_{\geq j+1}$ . The result follows as offspring are uniformly distributed on  $A_{\geq j}$  with probability  $\frac{1}{|A_{\geq j}|}$ . We have:

$$w = p(e) \cdot \text{lcd}_{e \in C} p(e), \quad (64)$$

$$= 1. \quad (65)$$

Those  $m$  samples correspond to the selected individuals that will make up the set of parents of the next generation. This means that they are strictly improving offspring with respect to the current set of offspring. Hence, the selection probability (of the  $m$  selected individuals) is the probability to sample a strictly improving offspring in the convex hull of the current selected population.

Let  $P'_t$  be the current selected population. The elements of  $Seg(P'_t) = co(P'_t)$  are distributed as in Eq. (23). Moreover, the set  $co(P'_t)$  is equal to a level set  $A_{\geq j}$  in our case study. That is,  $P = P'_{t+1}$  and  $C = A_{\geq j+1}$  in Eq. (61). The selection probability is therefore the probability for sampling an element of  $A_{\geq j}$  that belongs to  $A_{\geq j+1}$ .

**Proposition 5.** The selection probability of elements of  $A_{\geq j}$  is at least:

$$\frac{1}{|A_{\geq j+1}| + 1}. \quad (66)$$

**Proof.** Let  $t$  be the current generation. The distribution of the offspring on  $co(P'_t) = A_{\geq j}$  is as in Eq. (23). The parents  $P'_{t+1}$  of the next generation are the offspring belonging to  $A_{\geq j+1}$ . The probability to select an offspring  $z_0$  from  $co(P'_t) = A_{\geq j}$  that belongs to  $A_{\geq j+1}$  is given by:

$$\frac{\sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0)|}{|s_i|}}{\sum_{z \in A_{\geq j+1}} \sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z)|}{|s_i|}} = \frac{\sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|}}{\sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}}. \quad (67)$$

We determine a lower bound on (67).

$$\begin{aligned} & \frac{\sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|}}{\sum_{i=1}^p \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}}, \\ & \frac{\sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|} + \sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|}}{\sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|} + \sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}}. \end{aligned}$$

By Corollary 2, the probability for sampling a segment  $s_i$  that is strictly included in  $A_{\geq j}$  is bounded below by  $2/3$  when  $P'$  contains at least two distinct individuals. Hence, a segment  $s_i$  is either contained in  $A_{\geq j+1}$  or equal to  $A_{\geq j+1}$  in the typical case. Therefore,

$$\sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|} \leq \sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}. \quad (68)$$

Consequently, a lower bound on (67) is given by:

$$\begin{aligned} & \frac{\sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|}}{\sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|} + \sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}}, \\ & \geq \frac{\sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|}}{\left[ \sum_{s_i \subseteq A_{\geq j+1}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|s_i(z_0) \cap s_i|}{|s_i|} \right] \max_{\substack{s_i \subseteq A_{\geq j+1} \\ s_i \cap \{z_0\} \neq \emptyset}} \frac{|A_{\geq j+1} \cap s_i|}{|s_i(z_0) \cap s_i|} + \sum_{\substack{s_i \not\subseteq A_{\geq j+1} \\ s_i \cap A_{\geq j+1} \neq \emptyset}} \frac{\alpha_{s_i, P'_t}}{|P'|^2} \cdot \frac{|A_{\geq j+1} \cap s_i|}{|s_i|}}, \\ & \geq \frac{1}{\max_{\substack{s_i \subseteq A_{\geq j+1} \\ s_i \cap \{z_0\} \neq \emptyset}} \frac{|A_{\geq j+1} \cap s_i|}{|s_i(z_0) \cap s_i|} + 1}, \\ & \geq \frac{1}{|A_{\geq j+1}| + 1}. \end{aligned}$$

**Proposition 6.** The selection probability of elements of  $A_{\geq j}$  is at most:

$$\frac{2}{|A_{\geq j+1}| + 1}. \quad (69)$$

**Proof.** The probabilities for selecting an offspring belonging to  $A_{\geq j+1}$  add up to one for each element of  $A_{\geq j+1}$ . An upper bound on the probability for selecting an offspring belonging to  $A_{\geq j+1}$  is obtained when all the remaining  $|A_{\geq j+1}| - 1$  offspring have the least probability of Proposition 5 to be selected. Hence, the largest probability for selecting an offspring belonging to  $A_{\geq j+1}$  (i.e., a parent for the next generation) is:

$$1 - \frac{|A_{\geq j+1}| - 1}{|A_{\geq j+1}| + 1} = \frac{2}{|A_{\geq j+1}| + 1}.$$

□

By Proposition 5 and Proposition 6 we have:

**Corollary 4.** The weight of a sample is at most two for the SES.

**Proof.** The lcd of all the selection probabilities is given by the denominator of the least possible probability given in Proposition 5. Any other

probability is bounded above by the largest possible probability given in Proposition 6. Hence, all weights are bounded above by:

$$\frac{2}{|A_{\geq j+1}| + 1} \cdot (|A_{\geq j+1}| + 1) = 2.$$

□

As a result, each of the  $m$  samples of Eq. (61) contributes between once and twice in the making of their convex hull in our analysis.

### 6.1. Instantiation of the analysis to strings on a finite alphabet

We specify the results of the analysis of the SES on quasi-concave landscapes to  $d$ -ary strings of length  $n$  on the alphabet  $\{0, 1, 2, \dots, d - 1\}$ . We will consider the same metrics used in [2] for the analysis of the CS :

- The Hamming distance HD,
- The Manhattan distance MD.

#### 6.1.1. Hamming distance

We know from Corollary 1 and Theorem 5 that the sets  $Seg(A)$  and  $co(A)$  coincide for any subset  $A$  of the metric space  $M_{2,HD}$ , but this need not be the case in the metric spaces  $M_{d,HD}$ , where  $d \geq 3$ . Thus, we restrict our analysis to the metric space  $M_{2,HD}$ .

We first estimate  $P_{M_{2,HD}}^{Cov}(m)$  which is a lower bound on the probability for covering a convex set  $C$  of  $M_{2,HD}$  with  $m$  samples from  $C$ .

**Lemma 10.** For any convex set  $C$  of the metric space  $M_{2,HD}$  we have  $P_{C,Seg}^{Cov}(m) \geq P_{M_{2,HD},Seg}^{Cov}(m)$ , where:

$$P_{M_{2,HD},Seg}^{Cov}(m) \geq 1 - 2n \left(1 - \frac{r^2}{6}\right)^m.$$

**Proof.** We will estimate:

$$P_{M_{d,HD},Seg}^{Cov}(m) = \Pr[co(P) = M_{d,HD} \mid P = \text{NonUnif}_m(M_{d,HD})], \quad (70)$$

for  $d = 2$ .

We saw in Lemma 1 that any schema corresponds to a convex set in the metric space  $M_{d,HD}$ . In particular, the schema corresponding to the entire search space is the only schema with the largest number of positions that are free to take more than one value. Moreover, each of these free positions take the maximum number of possible values. Therefore, the schema corresponding to any other convex set has at most  $n$  free positions. Each of these positions is free to take at most  $d$  values.

Let us now compute the probability  $\Pr[co(P) = M_{d,HD} \mid P = \text{NonUnif}_m(M_{d,HD})]$  for covering the entire search space from sampling  $m$  points from it. The schema corresponding to the entire search space is  $\underbrace{*** \dots *}_{n \text{ times}}$ .

The probability to sample an element from  $Seg(P) = co(P)$  is bounded below by  $\frac{1}{d^2}$  as all weights are bounded above by two (see [Corollary 4](#)). In the worst case scenario, all the improving offspring have the least probability  $\frac{1}{d^2}$  to be sampled. As the probability for sampling an improving offspring is at least  $\frac{2r^2}{3}$  (see [Theorem 8](#)), the probability for sampling an improving offspring with the least possible probability is at least  $\frac{2r^2}{3d^2}$ .

The don't care symbol is obtained at some position when each of the values 0, 1, ...,  $d-1$  appears at least once at this position. The probability that a value appears at this position in  $e_i$  is at least  $\frac{2r^2}{3d^2}$ . The probability that this value never appears at this position in  $e_i$  is therefore  $1 - \frac{2r^2}{3d^2}$ . The probability that this value never appears at this position in  $e_1, e_2, \dots$ , and  $e_m$  is therefore:

$$\prod_{i=1}^m \left(1 - \frac{2r^2}{3d^2}\right) = \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (71)$$

The probability that the value 0 never appears at this position OR the value 1 OR ... OR the value  $d-1$  is:

$$d \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (72)$$

Hence, the probability that each value appears at least once at that position is:

$$1 - d \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (73)$$

Thus, the probability for obtaining the don't care symbol at  $n$  positions is:

$$\left[1 - d \left(1 - \frac{2r^2}{3d^2}\right)^m\right]^n. \quad (74)$$

Hence, the probability for obtaining the schema  $\underbrace{*** \dots *}_{n \text{ times}}$  is at least:

$$\left[1 - d \left(1 - \frac{2r^2}{3d^2}\right)^m\right]^n \geq 1 - dn \left(1 - \frac{2r^2}{3d^2}\right)^m, \quad (75)$$

using Bernoulli's inequality.  $\square$

**Theorem 10.** *Let us consider a quasi-concave landscape on  $M_{2,HD}$ , whose canonical level sets are:  $A_{\geq 0}, A_{\geq 1}, \dots, A_q$ . Let also  $r = \min_{0 \leq j \leq q} \frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ . The SES with population size  $\mu$  finds a global optimum within at most  $q$  generations and  $\mu q$  fitness evaluations with probability at least:*

$$1 - 2n(2q+1) \exp \left[ - \min \left( \frac{r^2}{9}, \frac{1}{18} \right) \cdot r^2 \mu \right]. \quad (76)$$

**Proof.** We estimate a lower bound on  $\left[ P_{M_{d,HD},Seg}^{Cov} \left( \frac{2r^2 \mu}{3} \right) \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right)$  for  $d = 2$ .

$$\begin{aligned} & \left[ P_{M_{d,HD},Seg}^{Cov} \left( \frac{2r^2 \mu}{3} \right) \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq \left[ 1 - dn \left( 1 - \frac{2r^2}{3d^2} \right) \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq \left[ 1 - dn(q+1) \left( 1 - \frac{2r^2}{3d^2} \right) \right] - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq 1 - dn(q+1) \exp \left( - \frac{4r^4 \mu}{9d^2} \right) - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq 1 - dn(2q+1) \exp \left[ - \min \left( \frac{4r^2}{9d^2}, \frac{1}{18} \right) \cdot r^2 \mu \right]. \end{aligned}$$

The third line follows from Bernoulli's inequality. The fourth line is due to the fact that  $\ln(1+x)$  is bounded above by  $x$  whenever  $x < 0$ .  $\square$

**Corollary 5.** *Let us consider a quasi-concave landscape on  $M_{2,HD}$ , whose canonical level sets are:  $A_{\geq 0}, A_{\geq 1}, \dots, A_q$ . Let also  $r = \min_{0 \leq j \leq q} \frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ . The SES with population size:*

$$\mu \geq \frac{\ln [4n(2q+1)]}{r^2 \min \left( \frac{r^2}{9}, \frac{1}{18} \right)}. \quad (77)$$

*finds a global optimum within  $2q$  expected generations and  $2\mu q$  expected fitness evaluations.*

**Proof.** The result follows from solving in  $\mu$  the inequality:

$$1 - 2n(2q+1) \exp \left[ - \min \left( \frac{r^2}{9}, \frac{1}{18} \right) \cdot r^2 \mu \right] \geq \frac{1}{2}. \quad (78)$$

$\square$

Let one run of the SES be performed in  $q$  generations. If the population size satisfies the condition of [Corollary 5](#), then the expected number of runs before finding a global optimum (i.e., the expected hitting time) is at most  $\frac{1}{0.5} = 2$ . Hence, the expected number of generations and the expected number of fitness evaluations needed for finding a global optimum are respectively  $2q$  and  $2\mu q$ .

We apply the runtime result to the leading ones problem (LO). Indeed, the fitness landscape  $(\{0, 1\}^n, LO, HD)$  is quasi-concave with parameters  $q = n$  and  $r = 0.5$ .

**Theorem 11.** *In the metric space  $M_{2,HD} = (\{0, 1\}^n, HD)$ , Leading Ones is solved in  $2n$  expected generations by the SES when the population size is at least  $144 \ln[4n(2n+1)]$ .*

**Proof.** We apply the result of [Corollary 5](#) to Leading Ones by replacing  $q$  and  $r$  with their respective values for Leading Ones and by replacing  $d$  with 2.  $\square$

### 6.1.2. Manhattan distance

We know from [Theorem 6](#) that the sets  $Seg(A)$  and  $co(A)$  coincide for any subset  $A$  of the metric space  $M_{d,MD}$  for  $d \geq 2$ .

We first estimate the probability  $P_{M_{d,MD},Seg}^{Cov}(m)$  which is a lower bound on the probability for covering a convex set  $C$  of  $M_{d,MD}$  with  $m$  samples from  $C$ .

**Lemma 11.** *We assume that  $d \geq 2$ , for any convex set  $C$  of the metric space  $M_{d,MD}$  we have  $P_{C,Seg}^{Cov}(m) \geq P_{M_{d,MD},Seg}^{Cov}(m)$ , where:*

$$P_{M_{d,MD},Seg}^{Cov}(m) \geq 1 - 2n \left( 1 - \frac{2r^2}{3d^2} \right)^m.$$

**Proof.** We saw in [Lemma 2](#) that schemata using only the symbol  $*$  and/or  $*_{\{k,l\}}$  and/or fixed values correspond to a convex set in the metric space  $M_{d,MD}$ . In particular, the schema corresponding to the entire search space is the only schema with the largest number of positions that are free to take more than one value. Moreover, each of these free positions take the maximum number of possible values. Therefore, the schema corresponding to any other convex set has at most  $n$  symbols  $*$ .

Let us now compute the probability  $\Pr[co(P) = M_{d,MD} \mid P = \text{NonUnif}_m(M_{d,MD})]$  for covering the entire search space from sampling  $m$  points from it. The schema corresponding to the entire search space is  $\underbrace{*** \dots *}_{n \text{ times}}$ .

The probability to sample an element from  $Seg(P) = co(P)$  is bounded below by  $\frac{1}{d^2}$  as all weights are bounded above by two (see [Corollary 4](#)). In the worst case scenario, all the improving offspring have the least probability  $\frac{1}{d^2}$  to be sampled. As the probability for sampling an improving offspring is at least  $\frac{2r^2}{3}$  (see [Theorem 8](#)), the probability



for sampling an improving offspring with the least possible probability is at least  $\frac{2r^2}{3d^2}$ .

The don't care symbol is obtained at some position when each of the values 0 and  $d - 1$  appears at least once at this position. The probability that a value appears at this position in  $e_i$  is at least  $\frac{2r^2}{3d^2}$ . The probability that this value never appears at this position in  $e_i$  is therefore  $1 - \frac{2r^2}{3d^2}$ . The probability that this value never appears at this position in  $e_1, e_2, \dots$ , and  $e_m$  is therefore:

$$\prod_{i=1}^m \left(1 - \frac{2r^2}{3d^2}\right) = \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (79)$$

The probability that the value 0 never appears at this position OR the value  $d - 1$  never appears at this position is:

$$2 \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (80)$$

Hence, the probability that each value appears at least once at that position is:

$$1 - 2 \left(1 - \frac{2r^2}{3d^2}\right)^m. \quad (81)$$

Thus, the probability for obtaining the don't care symbol at  $n$  positions is:

$$\left[1 - 2 \left(1 - \frac{2r^2}{3d^2}\right)^m\right]^n \geq 1 - 2n \left(1 - \frac{2r^2}{3d^2}\right)^m, \quad (82)$$

using Bernoulli's inequality.  $\square$

**Theorem 12.** Let us consider a quasi-concave landscape on  $M_{d,MD}$ , whose canonical level sets are:  $A_{\geq 0}, A_{\geq 1}, \dots, A_q$ . Let also  $r = \min_{0 \leq j \leq q} \frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ . The SES with population size  $\mu$  finds a global optimum within at most  $q$  generations and  $\mu q$  fitness evaluations with probability at least:

$$1 - 2n(2q + 1) \exp \left[ - \min \left( \frac{4r^2}{9d^2} \frac{1}{18} \right) \cdot r^2 \mu \right]. \quad (83)$$

**Proof.** We estimate a lower bound on  $\left[ P_{M_{d,MD}}^{Cov} \left( \frac{2r^2 \mu}{3} \right) \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right)$ .

$$\begin{aligned} & \left[ P_{M_{d,MD}}^{Cov} \left( \frac{2r^2 \mu}{3} \right) \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq \left[ 1 - 2n \left( 1 - \frac{2r^2}{3d^2} \right)^{\frac{2r^2 \mu}{3}} \right]^{q+1} - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq \left[ 1 - 2n(q+1) \left( 1 - \frac{2r^2}{3d^2} \right)^{\frac{2r^2 \mu}{3}} \right] - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq 1 - 2n(q+1) \exp \left( - \frac{4r^4 \mu}{9d^2} \right) - q \exp \left( - \frac{r^2 \mu}{18} \right), \\ & \geq 1 - 2n(2q+1) \exp \left[ - \min \left( \frac{4r^2}{9d^2} \frac{1}{18} \right) \cdot r^2 \mu \right]. \end{aligned}$$

The third line follows from Bernoulli's inequality. The fourth line is due to the fact that  $\ln(1+x)$  is bounded above by  $x$  whenever  $x < 0$ .  $\square$

**Corollary 6.** Let us consider a quasi-concave landscape on  $M_{d,MD}$ , whose canonical level sets are:  $A_{\geq 0}, A_{\geq 1}, \dots, A_q$ . Let also  $r = \min_{0 \leq j \leq q} \frac{|A_{\geq j+1}|}{|A_{\geq j}|}$ . The SES with population size:

$$\mu \geq \frac{\ln[4n(2q+1)]}{\min \left( \frac{4r^2}{9d^2} \frac{1}{18} \right) \cdot r^2}. \quad (84)$$

finds a global optimum within  $2q$  expected generations and  $2\mu q$  expected fitness evaluations.

**Table 1**

General Runtime Results of the CS and the SES on a quasi-concave landscape of parameters  $q$  and  $r$  for a population size  $\mu$ .

Algorithm	Lower bound on expected number of improving offspring	Finds a global optimum within $q$ generations and $\mu q$ fitness evaluations with probability at least:
CS	$\frac{\mu r}{4}$	$\left[ P_S^{Cov} \left( \frac{\mu r}{4} \right) \right]^{q+1} - q \exp \left( - \frac{9\mu r}{32} \right)$ [2]
SES	$\frac{2\mu r^2}{3}$	$\left[ P_{S.Seg}^{Cov} \left( \frac{2\mu r^2}{3} \right) \right]^{q+1} - q \exp \left( - \frac{\mu r^2}{18} \right)$

**Table 2**

Theoretical smallest population size required for finding a global optimum with probability at least 0.5.

Algorithm	Metric Space	Population size threshold for finding a global optimum with probability at least 0.5
CS	$M_{d,HD}$	$\frac{4d}{\ln[2dn(2q+1)]}$ [2]
SES	$M_{2,HD}$	$\frac{r}{\ln[4n(2q+1)]}$ $r^2 \min \left( \frac{r^2}{9}, \frac{1}{18} \right)$
CS	$M_{d,MD}$	$\frac{4d}{\ln[4n(2q+1)]}$ [2]
SES	$M_{d,MD}$	$\frac{r}{\ln[4n(2q+1)]}$ $\min \left( \frac{4r^2}{9d^2} \frac{1}{18} \right) \cdot r^2$

**Proof.** The result follows from solving in  $\mu$  the inequality:

$$1 - 2n(2q+1) \exp \left[ - \min \left( \frac{4r^2}{9d^2} \frac{1}{18} \right) \cdot r^2 \mu \right] \geq \frac{1}{2}. \quad (85)$$

$\square$

Let one run of the SES be performed in  $q$  generations. If the population size satisfies the condition of Corollary 6, then the expected number of runs before finding a global optimum (i.e., the expected hitting time) is at most  $\frac{1}{0.5} = 2$ . Hence, the expected number of generations and the expected number of fitness evaluations needed for finding a global optimum are respectively  $2q$  and  $2\mu q$ .

We apply the runtime result to the fitness function of Definition 8 that yields a quasi-concave landscape of parameters  $q = n$  and  $r = \frac{1}{d}$  in the metric space  $M_{d,MD} = (\{0, 1, \dots, d-1\}^n, MD)$ .

**Theorem 13.** In the metric space  $(\{0, 1, \dots, d-1\}^n, MD)$ , if the population size is at least:

$$\mu \geq \frac{9d^6}{4} \ln[4n(2n+1)], \quad (86)$$

then the SES solves the longest common prefix (with a fixed string  $a$ ) problem in  $2n$  expected generations.

**Proof.** By Proposition 1 and Corollary 6.  $\square$

## 7. Experiment

We showed that in metric spaces where the sets  $Seg(A)$  and  $co(A)$  coincide for any subset  $A$ , SESs are particular CSs with a non-uniform offspring distribution. We analyzed the runtime of those specific SESs on quasi-concave landscapes, by extending the analysis of the CS in [2]. The runtime results are summarized in Table 1.

The theoretical smallest population sizes required for finding a global optimum are summarized in Table 2.

Note that the theoretical results obtained for the CS have already been verified experimentally in [2]. Hence, we shall only run experiments to verify the theoretical results obtained above for the SES. To this end:

- a quasi-concave landscape on the metric space  $M_{2,HD}$  (or  $M_{d,MD}$  for  $d \geq 3$ ) need be considered,
- the geometric crossover performed by the SES need be specified to the metric space considered.

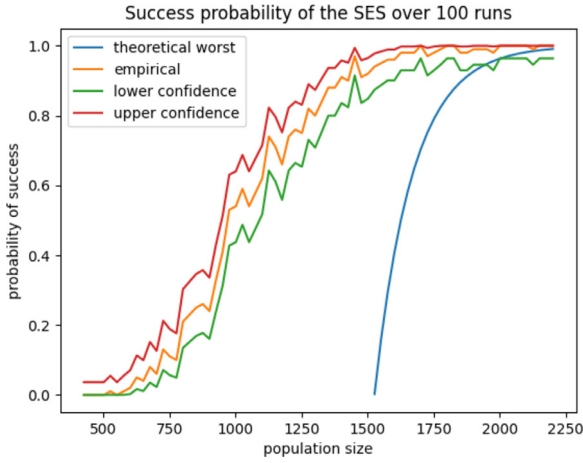


Fig. 2. Theoretical worst vs Empirical success probability of the SES for LO.

1. In  $M_{2,HD}$ , the geometric crossover of parent1 =  $a_1 a_2 \dots a_n$  and parent2 =  $a'_1 a'_2 \dots a'_n$  returns offspring =  $b_1 b_2 \dots b_n$ , where  $b_i$  is either  $a_i$  or  $a'_i$  for  $1 \leq i \leq n$ .
2. In  $M_{d,MD}$ , the geometric crossover of parent1 =  $a_1 a_2 \dots a_n$  and parent2 =  $a'_1 a'_2 \dots a'_n$  returns offspring =  $b_1 b_2 \dots b_n$ , where  $b_i$  is an integer between  $\min(a_i, a'_i)$  and  $\max(a_i, a'_i)$  for  $1 \leq i \leq n$ .

### 7.1. leading ones in $M_{2,HD}$

The fitness landscape  $(\{0, 1\}^n, LO, HD)$  is quasi-concave with parameters  $q = n$  and  $r = 0.5$  (see Example 11). Theorem 11 states that if the population size is at least  $144 \ln[4n(2n + 1)]$ , then the SES solves LO in  $2n$  expected generations. More precisely, the global optimum is found within  $q = n$  expected generations with probability at least 0.5 (Theorem 10). We shall verify these theoretical results for  $n = 100$ :

- we run the SES on LO one hundred times,
- we determine the empirical probability of success of the SES on LO (a success corresponds to finding the global optimum of LO),
- we compare this empirical probability of success to the theoretical worst probability of success of Theorem 10.

For  $n = 100$ , the theoretical population threshold for finding a global optimum with probability at least 0.5 is  $144 \ln[4n(2n + 1)] \simeq 1626.44$ . Hence, we shall consider population sizes ranging from 427 to 2227 with a step of 25. The plot also shows lower and upper 95% binomial confidence intervals using Clopper-Pearsons intervals. The result is shown in Fig. 2.

### 7.2. longest common prefix in $M_{d,MD}$

Let  $d \geq 2$  and let  $a$  and  $b$  be strings of the search space  $\{0, 1, \dots, d - 1\}^n$ . We recall that  $PX_a(b)$  returns the length of the longest prefix of  $b$  that is also a prefix of  $a$ . The fitness landscape  $(\{0, 1, \dots, d - 1\}^n, PX_a, MD)$  is quasi-concave with parameters  $q = n$  and  $r = 1/d$  (see Remark 1). Theorem 13 states that if the population size is at least  $\frac{9d^6}{4} \ln[4n(2n + 1)]$ , then the SES solves  $PX_a$  in  $2n$  expected generations. More precisely, the global optimum is found within  $q = n$  expected generations with probability at least 0.5 (Theorem 12). We shall verify these theoretical results for  $n = 10$ ,  $d = 3$  and  $a = 11 \dots 1$ :

- we run the SES on  $PX_a$  one hundred times,
- we determine the empirical probability of success of the SES on  $PX_a$  (a success corresponds to finding the string  $a$ ),
- we compare this empirical probability of success to the theoretical worst probability of success of Theorem 12.

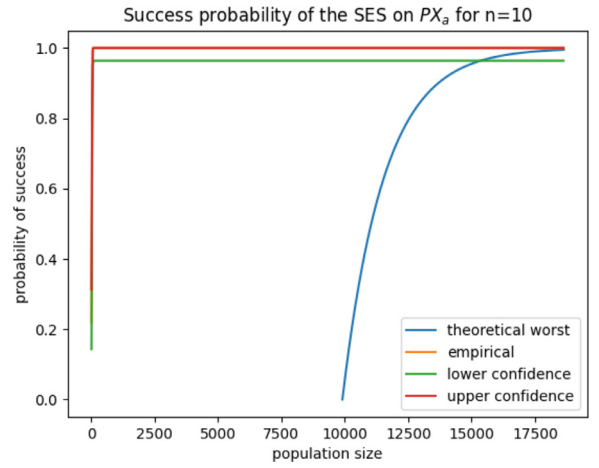


Fig. 3. Theoretical worst vs Empirical success probability of the SES for  $PX_a$ .

For  $n = 10$  and  $d = 3$ , the theoretical population threshold for finding a global optimum with probability at least 0.5 is  $\frac{9d^6}{4} \ln[4n(2n + 1)] \simeq 11044$ . Hence, we shall consider population sizes ranging from 20 to 18645 with a step of 25. The plot also shows lower and upper 95% binomial confidence intervals using Clopper-Pearsons intervals. The result is shown in Fig. 3.

### 7.3. Interpretation of results

The theoretical worst success probabilities are always smaller than or equal to the empirical success probabilities and approach the empirical success probabilities as the population size increases.

- The theoretical worst probability is a lower bound on the success probability. Hence, it is expected to sit below (or at best coincides with) the empirical success probability for each population size considered.
- Asymptotic approximations have been used to determine the formula for the theoretical worst success probability. Hence, the asymptotic values of the theoretical worst success probability approach the asymptotic empirical values of the success probability.

The theoretical worst success probability does not cover all population sizes. This is an immediate consequence of the worst case scenario analysis. The formula obtained for the theoretical worst success probability is only defined for population sizes larger than some threshold.

While comparing the lower bound obtained for binary strings of the metric space  $M_{2,HD}$  (see Fig. 2) to the lower bound obtained for 3-ary strings of the metric space  $M_{3,MD}$  (see Fig. 3), we find that the tightness of the lower bound depends on the representation considered. However, the theoretical lower bound on the success probability is always tight for large populations regardless of the representation considered (i.e., larger than 1626 for binary strings of  $M_{2,HD}$  and larger than 11,044 for 3-ary strings of  $M_{3,MD}$ ). Therefore, the corresponding runtime upper bound is also tight for large populations. In particular, the theoretical result is useful at estimating a minimal population size for which the algorithm is guaranteed to find a global optimum within at most two runs for any representation considered. In other words, we are concerned with finding a rule on the population size that guarantees a tight runtime upper bound.

Finally, the theoretical results on the analysis of the SES on a quasi-concave landscape presented in this paper were only instantiated to the metric spaces  $M_{2,HD}$  and  $M_{d,MD}$ . This is because the representation-free analysis presented in this paper is restricted to those metric spaces where the set  $Seg(A)$  and  $co(A)$  coincide for any subset  $A$ .

## 8. Conclusion

We developed a representation-free analysis of EAs with no mutation and with a standard two-parents crossover.

- We defined a generalization across representations of EAs with no mutation and with a standard two-parents crossover, that we named SES (Segmentwise Evolutionary Search algorithm).
- We considered the class of quasi-concave landscapes whose geometry matches that of the search performed by the SES.
- We analysed the runtime of the SES on quasi-concave landscapes. In this paper, the representation-free runtime results have been instantiated to:
  - binary strings of the metric space  $M_{2,HD}$ ,
  - $d$ -ary strings of the metric space  $M_{d,MD}$  for  $d \geq 2$ .

The SES solves quasi-concave landscapes with at most polynomially many level sets in at most polynomial expected time for well-chosen population sizes, in both metric spaces.

The approach used in this paper is a more universal runtime analysis of EAs with no mutation and with a standard two-parents crossover. Indeed, the runtime result can be instantiated to any representation whose corresponding metric space satisfy:

$$Seg(A) = co(A) \text{ for any subset } A. \quad (*)$$

However, the instantiation of the runtime result may differ for different representations. Here, the instantiation of the representation-free analysis to the metric spaces  $M_{2,HD}$  and  $M_{d,MD}$  for  $d \geq 2$  showed that quasi-concave landscapes are easily searched by SESs for a well-chosen population size. In future work, we will investigate the case of the metric space  $M_{d,HD}$  for  $d \geq 3$  and that of permutations metric spaces that do not satisfy (\*).

## Credit author statement

I certify that all material in this paper which is not my own work has been identified and that no material has previously been submitted to another journal.

## Declaration of Competing Interest

No conflict of interest.

## Acknowledgment

We would like to thank Dr Deise Santana Maia and Mr Adriaan Janse Van Vuuren for allowing us to run computational experiments on their machines.

## References

- [1] A. Moraglio, Abstract convex evolutionary search, in: Proceedings of the 11th Workshop Proceedings on Foundations of Genetic Algorithms, ACM, 2011, pp. 151–162.
- [2] A. Moraglio, D. Sudholt, Principled design and runtime analysis of abstract convex evolutionary search, *Evol. Comput.* 25 (2) (2017) 205–236.
- [3] A. Moraglio, Towards a Geometric Unification of Evolutionary Algorithms, University of Essex, 2008 Ph.D. thesis.
- [4] A. Moraglio, R. Poli, Topological interpretation of crossover, in: Genetic and Evolutionary Computation Conference, Springer, 2004, pp. 1377–1388.
- [5] N. Nedic, D. Prsic, L. Dubonjic, V. Stojanovic, V. Djordjevic, Optimal cascade hydraulic control for a parallel robot platform by pso, *Int. J. Adv. Manuf. Technol.* 72 (5) (2014) 1085–1098.

- [6] H. Tao, J. Li, Y. Chen, V. Stojanovic, H. Yang, Robust point-to-point iterative learning control with trial-varying initial conditions, *IET Control Theory Appl.* 14 (19) (2020) 3344–3350.
- [7] J. Holland, *Adaptation in Artificial and Natural Systems*, The University of Michigan Press, Ann Arbor, 1975.
- [8] L. Altenberg, The schema theorem and prices theorem, *Found. Gen. Alg.* 3 (1995) 23–49.
- [9] M.D. Vose, *The Simple Genetic Algorithm: Foundations and Theory*, volume 12, MIT press, 1999.
- [10] N.J. Radcliffe, Equivalence class analysis of genetic algorithms, *Compl. Syst.* 5 (2) (1991) 183–205.
- [11] W.B. Langdon, R. Poli, *Foundations of Genetic Programming*, Springer Science & Business Media, 2013.
- [12] D.E. Goldberg, *Genetic algorithms in search, optimization and machine learning* addison-wesley, 1989, Reading, MA (1989).
- [13] C. Stephens, H. Waelbroeck, Schemata evolution and building blocks, *Evol. Comput.* 7 (2) (1999) 109–124.
- [14] B. Burlacu, M. Affenzeller, S. Winkler, M. Kommenda, G. Kronberger, Methods for genealogy and building block analysis in genetic programming, in: *Computational Intelligence and Efficiency in Engineering Systems*, Springer, 2015, pp. 61–74.
- [15] M.D. Vose, G.E. Liepins, Punctuated equilibria in genetic search, *Complex Syst.* 5 (1) (1991) 31–44.
- [16] A.E. Nix, M.D. Vose, Modeling genetic algorithms with Markov chains, *Ann. Math. Artif. Intell.* 5 (1) (1992) 79–88.
- [17] J.R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, volume 1, MIT press, 1992.
- [18] C.R. Stephens, A. Zamora, Ec theory: a unified viewpoint, in: *Genetic and Evolutionary Computation Conference*, Springer, 2003, pp. 1394–1405.
- [19] T. Jones, et al., One operator, one landscape, Santa Fe Institute Technical Report (1995). 95–02
- [20] I. Zelinka, L. Tomaszek, P. Vasant, T.T. Dao, D.V. Hoang, A novel approach on evolutionary dynamics analysis—a progress report, *J. Comput. Sci.* 25 (2018) 437–445.
- [21] A. Prügel-Bennett, Modelling evolving populations, *J. Theor. Biol.* 185 (1) (1997) 81–95.
- [22] C.M. Reidys, P.F. Stadler, Combinatorial landscapes, *SIAM Rev.* 44 (1) (2002) 3–54.
- [23] P. Gitchoff, G.P. Wagner, Recombination induced hypergraphs: a new approach to mutation-recombination isomorphism, *Complexity* 2 (1) (1996) 37–43.
- [24] P.F. Stadler, G.P. Wagner, Algebraic theory of recombination spaces, *Evol. Comput.* 5 (3) (1997) 241–275.
- [25] Y. Zhou, J. He, A runtime analysis of evolutionary algorithms for constrained optimization problems, *IEEE Trans. Evol. Comput.* 11 (5) (2007) 608–619.
- [26] I. Wegener, Methods for the analysis of evolutionary algorithms on Pseudo-Boolean functions, in: *Evolutionary Optimization*, Springer, 2003, pp. 349–369.
- [27] R. Motwani, P. Raghavan, *Randomized Algorithms*, Cambridge University Press, 1995.
- [28] J. He, X. Yao, Drift analysis and average time complexity of evolutionary algorithms, *Artif. Intell.* 127 (1) (2001) 57–85.
- [29] D. Sudholt, Crossover speeds up building-block assembly, in: *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation*, ACM, 2012, pp. 689–702.
- [30] B. Doerr, C. Doerr, F. Ebel, From black-box complexity to designing new genetic algorithms, *Theor. Comput. Sci.* 567 (2015) 87–104.
- [31] T. Jansen, I. Wegener, The analysis of evolutionary algorithms—a proof that crossover really can help, *Algorithmica* 34 (1) (2002) 47–66.
- [32] T. Jansen, I. Wegener, On the analysis of evolutionary algorithms a proof that crossover really can help, in: *European Symposium on Algorithms*, Springer, 1999, pp. 184–193.
- [33] T. Jansen, I. Wegener, Real royal road functions where crossover provably is essential, *Discrete Appl. Math.* 149 (1–3) (2005) 111–125.
- [34] T. Storch, I. Wegener, Real royal road functions for constant population size, *Theor. Comput. Sci.* 320 (1) (2004) 123–134.
- [35] D. Sudholt, Crossover is provably essential for the ising model on trees, in: *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*, ACM, 2005, pp. 1161–1167.
- [36] S. Fischer, I. Wegener, The ising model on the ring: Mutation versus recombination, in: *Genetic and Evolutionary Computation Conference*, Springer, 2004, pp. 1113–1124.
- [37] B. Doerr, E. Happ, C. Klein, Crossover can provably be useful in evolutionary computation, *Theor. Comput. Sci.* 425 (2012) 17–33.
- [38] M.L. van De Vel, *Theory of Convex Structures*, volume 50, Elsevier, 1993.
- [39] J.H. Holland, et al., *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*, MIT Press, 1992.