This is a repository copy of *Runtime analysis of crowding mechanisms for multimodal optimisation*.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/147333/

Version: Accepted Version

Article:

Covantes Osuna, E. and Sudholt, D. orcid.org/0000-0001-6020-1646 (2019) Runtime analysis of crowding mechanisms for multimodal optimisation. IEEE Transactions on Evolutionary Computation. ISSN 1089-778X

https://doi.org/10.1109/TEVC.2019.2914606

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other users, including reprinting/ republishing this material for advertising or promotional purposes, creating new collective works for resale or redistribution to servers or lists, or reuse of any copyrighted components of this work in other works. Reproduced in accordance with the publisher's self-archiving policy.

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

Runtime Analysis of Crowding Mechanisms for Multimodal Optimisation

Edgar Covantes Osuna and Dirk Sudholt

***Abstract*—Many real-world optimisation problems lead to multimodal domains and require the identification of multiple optima. Crowding methods have been developed to maintain population diversity, to investigate many peaks in parallel and to reduce genetic drift. We present the first rigorous runtime analyses of probabilistic crowding and generalised crowding, embedded in a (μ+1) EA. In probabilistic crowding the offspring compete with their parent in a fitness-proportional selection. Generalised crowding decreases the fitness of the inferior solution by a scaling factor during selection. We consider the bimodal function TWOMAX and introduce a novel and natural notion for functions with bounded gradients. For a broad range of such functions we prove that probabilistic crowding needs exponential time with overwhelming probability to find solutions significantly closer to any global optimum than those found by random search. Even when the fitness function is scaled exponentially, probabilistic crowding still fails badly. Only if the exponential's base is linear in the problem size, probabilistic crowding becomes efficient on TWOMAX. A similar threshold behaviour holds for generalised crowding on TWOMAX with respect to the scaling factor. Our theoretical results are accompanied by experiments for TWOMAX showing that the threshold behaviours also apply to the best fitness found.**

***Index Terms*—Crowding methods, runtime analysis, probabilistic crowding, generalised crowding, theory.**

I. INTRODUCTION

PREMATURE convergence is one of the major difficulties in Evolutionary Algorithms (EAs), the population converging to a sub-optimal individual before the fitness landscape is explored properly. Real-world optimisation problems often lead to multimodal domains and so require the identification of multiple optima, either local or global [24], [26]. In multimodal optimisation problems, there exist many attractors for which finding a global optimum can become a challenge to any optimisation algorithm. A diverse population can deal with these multimodal problems as it can explore several hills in the fitness landscape simultaneously.

Crowding methods were introduced to preserve the population diversity, to investigate many peaks in parallel and to reduce the effect of genetic drift [4]. In general, parents and offspring compete in a replacement-oriented survival process. These methods are well-known techniques as covered in tutorials and surveys for diversity-preserving mechanisms [11], [25], [27], [30]. The first crowding mechanism was introduced by De Jong [4] and it was called *standard crowding*. In this mechanism the offspring replace the most similar parent from a random subpopulation of size CF (*crowding factor*). Later in the nineties, Mahfoud [14] changed De Jong's scheme in the following way. In a genetic algorithm (GA) all elements of the population are grouped into $\mu/2$ pairs (where μ is the population size and assuming μ to be even). Then, these groups are recombined and mutated. For each pair of offspring, two sets of parent-child tournaments are possible. Each offspring competes against the most similar parent according to a distance metric, either genotypic or phenotypic, and the offspring replace their closest parent according to a replacement rule.

Deterministic, *probabilistic* and *generalised crowding* are examples of Mahfoud's scheme with different replacement rules. In deterministic crowding the offspring replace their closest parent if it is at least as good [14]. In probabilistic crowding, the offspring compete against their most similar parent and the survivor is chosen with a probability proportional to their fitness [15]. Generalised crowding is a variant that generalises both deterministic and probabilistic crowding through the choice of a parameter called *scaling factor* $\phi \in [0, 1]$ that diminishes the impact of the inferior search point [10]. Similar to standard crowding, in *restricted tournament selection (RTS)*, the offspring replace their most similar individual from a random subpopulation of size w (*window size*) if it is at least as good [13]. Given such a variety of mechanisms to choose from, it is often not clear which mechanism is the best choice for a particular problem.

Most of the analyses and comparisons made between crowding techniques are assessed by means of empirical investigations using benchmark functions [2], [24], [26]. Theoretical runtime analyses have been performed that rigorously quantify the expected time needed to find one or several global optima [3], [9]. Both approaches are important to understand how these mechanisms impact the EA runtime and if they enhance the search for good individuals. These different expectations imply where EAs and which crowding mechanism should be used and, perhaps even more importantly, where they should not be used. Previous theoretical studies [1], [3], [9], [18] compared the expected running time of different diversity mechanisms when embedded in a simple baseline EA, the (μ+1) EA. All mechanisms were considered on the well-known bimodal function $\text{TWOMAX}(x) := \max\{n - \sum_{i=1}^{n} x_i, \sum_{i=1}^{n} x_i\}$. TWOMAX consists of two different symmetric slopes (or branches) ZEROMAX and ONEMAX with 0^n and 1^n as global optima, respectively, and the

---

E. Covantes Osuna and D. Sudholt are with the Department of Computer Science, University of Sheffield, Sheffield S1 4DP, U.K. e-mail: {ecovantes1,d.sudholt}@sheffield.ac.uk

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

goal is to evolve a population that contains both optima[1].

TWOMAX was chosen because it is simply structured, hence facilitating a theoretical analysis, and it is hard for EAs to find both optima as they have the maximum possible Hamming distance. The results allowed for a fair comparison across a wide range diversity mechanisms, revealing that some mechanisms like avoiding genotype and phenotype duplicates perform badly, while other mechanisms like fitness sharing, clearing, deterministic crowding and RTS perform surprisingly well (see Table I and Section II).

In this paper we provide rigorous theoretical runtime analyses accompanied by experimental studies for probabilistic crowding and generalised crowding. Our goal is to narrow the gap between theory and practice by rigorously assessing their performance and providing insights of when and why they perform well (or not) to enhance our understanding of their strengths and weaknesses, and by comparing the performance of both mechanisms to other diversity mechanisms analysed previously on TWOMAX.

For the (μ+1) EA with probabilistic crowding we show that the mechanism is unable to evolve solutions that are significantly closer to any global optimum than those found by random search, even when given exponential time. To this end, we introduce a novel notion of (α, β)-bounded gradients: a function has an (α, β)-bounded gradient if within a region within Hamming distance at most β to any global optimum, when making a local step towards an optimum, the fitness increases by at most a factor of α. For instance, with $\alpha = 2$ the fitness can at most double with every local step towards an optimum. This property naturally holds for many optimisation problems showing some degree of smoothness; we show this formally for ONEMAX, TWOMAX and the classical combinatorial optimisation problems MAXSAT and VERTEX COLOURING. For all these problems we show that the (μ+1) EA with probabilistic crowding with overwhelming probability needs exponential time to even get moderately close to any optimum.

The reason for this disastrous performance is that the selective pressure of the embedded fitness-proportional selection is too low when the gradient towards global optima is bounded. This even holds when scaling the fitness using exponential scaling, i. e., raising the fitness to a power of some base α, leading to functions $\alpha^{\text{ONEMAX}(x)}$ and $\alpha^{\text{TWOMAX}(x)}$ with an (α, n)-bounded gradient. We show a lower runtime bound of $2^{\Omega(n/\alpha)}$ for all functions with $(\alpha, n/\alpha)$-bounded gradient, revealing that every constant base α, and even values up to $\alpha = \mathrm{O}\big(n^{1-\varepsilon}\big)$ still lead to exponential times with overwhelming probability. Probabilistic crowding only becomes effective when choosing a base of $\alpha = \Omega(n)$ as only then the selection pressure becomes large enough to enable hill climbing. In this case probabilistic crowding with scaling is as successful on TWOMAX as deterministic crowding. Our results establish a threshold behaviour with respect to the gradient α.

We further provide the first runtime analysis of generalised crowding. For a scaling factor $\phi = 1$ we have probabilistic crowding and $\phi = 0$ yields deterministic crowding. We show that there is a threshold behaviour with respect to ϕ: we give a lower runtime bound of $2^{\Omega(\phi n/\alpha)}$ for all functions of $(\alpha, \phi n/\alpha)$-bounded gradient. This gives exponential times with overwhelming probability on ONEMAX and TWOMAX if $\phi = \Omega\big(n^{-1+\varepsilon}\big)$. Only if $\phi = \mathrm{O}(1/n)$, that is, if generalised crowding is extremely close to deterministic crowding, the (μ+1) EA with generalised crowding becomes effective on TWOMAX.

Our main theoretical results are accompanied by experiments for TWOMAX that further investigate the best fitness values found during a run. The results show conclusively how performance of probabilistic crowding degrades to that of random search for increasing problem sizes. By comprehensively covering the whole parameter range for probabilistic crowding with exponential scaling and generalised crowding our experiments show how the best fitness found steadily improves as the parameters α and ϕ approach the identified efficient regimes.

This article significantly extends a preliminary conference paper [3] that only studied probabilistic crowding without scaling on ONEMAX and TWOMAX. In this work, the analysis of probabilistic crowding has been rewritten entirely[2] and the notion of (α, β)-bounded gradients was introduced to provide stronger statements that apply to general functions with bounded gradients that may have many (even exponentially many) global optima[3]. This paper further provides the first rigorous runtime analyses for probabilistic crowding with exponential scaling and for generalised crowding, establishing threshold behaviours for both. We believe that the notion of (α, β)-bounded gradients introduced here is of independent interest and likely to find many further applications in the analysis of randomised search heuristics.

II. PREVIOUS WORK AND PRELIMINARIES

There has been a line of work comparing various diversity mechanisms on TWOMAX in the context of the simple (μ+1) EA. The (μ+1) EA starts with a population of size μ created uniformly at random (u. a. r.) and generates one offspring due to mutation; the resulting offspring competes with an individual selected u. a. r. from the subpopulation with worst fitness and the best individual replaces the worst (in case of ties, the offspring is preferred). Table I summarises all known results, including our contributions (shown in bold) and conditions involving population size μ and specific parameters of each diversity mechanism explained below. Results from [9] are adapted to our definition of TWOMAX; see [28] for details.

---

[1] In [9] an additional fitness value for 1^n was added to distinguish between a local optimum 0^n and a unique global optimum. There the goal was to find the global optimum, and all approaches had a baseline probability of $1/2$ of climbing up the right branch by chance. We use the same approach as [1], [18], and consider the original definition of TWOMAX and the goal of finding both global optima. The discussion and presentation of previous work from [9] is adapted to our setting. We refer to [28] for details.

[2] The previous analyses considered the fitness as progress measure; this was replaced by the Hamming distance to a particular global optimum to show that the (μ+1) EA with probabilistic crowding is unable to get within a certain Hamming distance from any optimum.

[3] This extension was motivated by a discussion with Kenneth De Jong and Pietro S. Oliveto during our talk at GECCO 2018. We thank both of them for this inspiring discussion.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

TABLE I

OVERVIEW OF RUNTIME ANALYSES FOR THE (μ+1) EA WITH DIVERSITY MECHANISMS ON TWOMAX, SHOWING THE PROBABILITY OF FINDING BOTH OPTIMA WITHIN (EXPECTED) TIME $\mathrm{O}(\mu n \log n)$. RESULTS DERIVED IN THIS PAPER ARE SHOWN IN BOLD.

Diversity Mechanism	**Success prob.**	**Conditions**
Plain (μ+1) EA [9]	$\mathrm{o}(1)$	$\mu = \mathrm{o}(n/\log n)$
No Duplicates [9]		
Genotype	$\mathrm{o}(1)$	$\mu = \mathrm{o}(\sqrt{n})$
Fitness	$\mathrm{o}(1)$	$\mu = \mathrm{poly}(n)$
Deterministic Crowding [9]	$1 - 2^{-\mu+1}$	all μ
Fitness Sharing ($\sigma = n/2$)		
Population-based [9]	1	$\mu \geq 2$
Individual-based [18]	1	$\mu \geq 3$
Clearing ($\sigma = n/2$) [1]	1	$\mu \geq \kappa n^2$
Prob. Crowding[1] (Th 8, Cor 10)	$2^{-\Omega(n)}$	all μ
Prob. Crowding with Scaling		
General bases $\boldsymbol{\alpha}$[1] (Th 9, Cor 10)	$2^{-\Omega(n/\alpha)}$	all $\alpha \geq 1$
Very large $\boldsymbol{\alpha}$ (Th 11)	$1 - 2^{-\mu+1}$	$\alpha \geq (1+\Omega(1))en$
Generalised Crowding		
General scaling factors $\boldsymbol{\phi}$[1] (Th 13)	$2^{-\Omega(\phi n)}$	all $\phi \leq 1$
Very small $\boldsymbol{\phi}$ (Th 12)	$1 - 2^{-\mu+1}$	$\phi \leq \frac{1-\Omega(1)}{e^2 n}$
Restricted Tournament Sel. [3]		
Small window size w	$\mathrm{o}(1)$	$\mu = \mathrm{o}(n^{1/w})$
Large window size w	$1 - 2^{-\mu'+3}$	$w \geq 2.5\mu \ln n$

[1] These results also hold for general function classes with bounded gradients.

As can be seen from previous works on TWOMAX, not all mechanisms succeed in finding both optima efficiently, that is, in expected time $\mathrm{O}(\mu n \log n)$ (the best known time bound for the (μ+1) EA with diversity-preserving mechanisms). Friedrich, Oliveto, Sudholt, and Witt [9] showed that the plain (μ+1) EA and the simple mechanisms like *avoiding genotype or fitness duplicates* are not able to prevent the extinction of one branch, ending with the population converging to one optimum, with high probability. *Deterministic crowding* with a sufficiently large population is able to reach both optima with probability $1 - 2^{-\mu+1}$ in expected time $\mathrm{O}(\mu n \log n)$ [9, Theorem 4]. This probability converges to 1 exponentially fast in μ; for instance, a small population size of $\mu = 10$ already gives a success probability of ≈ 0.998 and for $\mu = 30$ it grows to ≈ 0.9999999981. A *population-based fitness sharing* approach, constructing the best possible new population amongst parents and offspring, with $\mu \geq 2$ and a sharing radius of $\sigma = n/2$ is able to find both optima in expected optimisation time $\mathrm{O}(\mu n \log n)$ [9, Theorem 5]. The drawback of this approach is that all possible size μ subsets of this union of size $\mu + \lambda$ (where λ is the offspring population size) need to be examined. This is prohibitive for large μ and λ.

Oliveto, Sudholt, and Zarges [18] studied the original *fitness sharing* approach and showed that a population size $\mu = 2$ is not sufficient to find both optima in polynomial time; the success probability is only $1/2 - \Omega(1)$ [18, Theorem 1]. However, with $\mu \geq 3$ fitness sharing again finds both optima in expected time $\mathrm{O}(\mu n \log n)$ [18, Theorem 3]. Covantes Osuna and Sudholt [1] analysed the *clearing* mechanism and showed that it can optimise all functions of unitation—function defined over the number of 1-bits contained in a string—in expected time $\mathrm{O}(\mu n \log n)$ [1, Theorem 4.4] when the distance function and parameters like the clearing radius σ, the niche capacity κ (how many winners a niche can support) and μ are chosen appropriately. In the case of large niches, that is, with a clearing radius of $\sigma = n/2$, it is able to find both optima in expected time $\mathrm{O}(\mu n \log n)$ [1, Theorem 5.6]. Finally, Covantes Osuna and Sudholt [3] showed that *Restricted Tournament Selection* fails to find both optima even in exponential time when the window size w is too small [3, Theorem 3.4]. However, if w is large enough and $\mu' := \min(\mu, \log n)$, the mechanism can find both optima efficiently in expected time $\mathrm{O}(\mu n \log n)$ [3, Theorem 3.1].

The above works did not consider crossover as recombining individuals from different branches is likely to create poor offspring. We therefore consider a (μ+1) EA using mutation only.

A. Notation

Our notion of time is defined as the number of function evaluations before the (μ+1) EA achieves a stated goal such as finding a global optimum or finding both optima of TWOMAX. Since the (μ+1) EA is initialised with μ individuals, and subsequently generates one offspring in each generation, the number of function evaluations is equal to μ plus the number of generations needed to achieve the set goal. The additional term of μ is only relevant for unreasonably large population sizes and is being tacitly ignored when it is absorbed in a runtime bound (such as $\mathrm{O}(\mu n \log n)$) anyway.

We say that a function f is exponential if $f \geq 2^{\Omega(n^{\varepsilon})}$ for a positive constant $\varepsilon > 0$. A function f is exponentially small if and only if $1/f$ is exponential. An event A occurs with overwhelming probability if $1 - \Pr(A)$ is exponentially small.

B. Drift Theorems

Our analysis will make heavy use of a technique called *drift analysis*. In a nutshell, the progress of the algorithm is measured by a potential function such as the Hamming distance to an optimum where a potential of 0 indicates that an optimum has been found. The *drift* is then defined as the expected change of this potential in one generation.

The following *multiplicative drift theorem* gives an upper bound on the expected time until the potential reaches 0 and an optimum has been found. It requires that the drift is at least proportional to its current state. It also gives a tail bound showing that the probability of exceeding this time is very small.

Theorem 1 (Multiplicative drift theorem with tail bounds, [6]). *Let $\{X_t\}_{t \geq 0}$ be a sequence of random variables taking values in some set S. Let $g : S \to \{0\} \cup \mathbb{R}_{\geq 1}$ and assume that $g_{\max} := \max\{g(x) \mid x \in S\}$ exists. Let $T := \inf\{t \geq 0 : g(X_t) = 0\}$. If there exists $\delta > 0$ such that*

$$\mathrm{E}[g(X_{t+1}) \mid g(X_t)] \leq (1 - \delta) g(X_t)$$

then $\mathrm{E}[T] \leq (1 + \ln g_{\max})/\delta$ and for every $c > 0$ $\Pr(T > (\ln g_{\max} + c)/\delta) \leq e^{-c}$.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

For our negative results we make heave use of the so-called negative drift theorem (also known as simplified drift theorem). It states that, if the drift is negative within an interval of the state space that needs to be crossed, and the algorithm typically does not make large jumps, the time to cross the interval is exponential in the interval size. The theorem uses transition probabilities $p_{i,j}$ and the notation "$p_{k,k\pm d} \leq x$" as a shorthand for "$p_{k,k+d} \leq x$ and $p_{k,k-d} \leq x$".

Theorem 2 (Negative drift theorem [17], [19])**.** *Consider a Markov process $X_0, X_1, \dots$ on $\{0, \dots, m\}$ with transition probabilities $p_{i,j}$ and suppose there exist integers a, b with $0 < a < b \leq m$ and $\varepsilon > 0$ such that for all $a \leq k \leq b$ the drift towards 0 is*

$$\mathrm{E}[k - X_{t+1} \mid X_t = k] < -\varepsilon \tag{1}$$

Further assume there exist constants $r, \delta > 0$ (i. e., they are independent of m) such that for all $k \geq 1$ and all $d \geq 1$

$$p_{k,k\pm d} \leq \frac{r}{(1+\delta)^d}. \tag{2}$$

Let T be the first hitting time of a state at most a, starting from $X_0 \geq b$. Let $\ell = b - a$. Then there is a constant $c > 0$ such that $\Pr\left(T \leq 2^{c\ell/r}\right) = 2^{-\Omega(\ell/r)}$.

We also use a variant that is applicable to processes with large self-loop probabilities.

Theorem 3 (Negative drift with self-loops [23])**.** *The negative drift theorem with self-loops is identical to Theorem 2 with the following revised conditions, where $p_{k,k}$ is the self-loop probability at state k.*

$$\mathrm{E}[k - X_{t+1} \mid X_t = k] < -\varepsilon \cdot (1 - p_{k,k}) \tag{3}$$

$$p_{k,k\pm d} \leq \frac{r(1 - p_{k,k})}{(1+\delta)^d}. \tag{4}$$

III. PROBABILISTIC CROWDING

Recall that in probabilistic crowding, the offspring compete against the most similar parent according to a distance metric and the survivor wins proportionally according to their fitness. Without crossover, this means that the mutant y competes against its parent x using fitness-proportional selection. The idea is to use a low selection pressure to prevent the loss of niches of lower fitness [15]. Then the probability of the mutant y winning is given by $\frac{f(y)}{f(x)+f(y)}$, where f is the fitness function. The resulting (μ+1) EA is shown in Algorithm 1.

Algorithm 1 (μ+1) EA with probabilistic crowding

```
Initialise $P$ with $\mu$ individuals chosen u. a. r.
while stopping criterion not met do
    Choose $x \in P$ u. a. r.
    Create $y$ by flipping bits in $x$ independently w/prob. $1/n$.
    With probability $\frac{f(y)}{f(y)+f(x)}$ set $P = P \setminus \{x\} \cup \{y\}$.
```

There are several related theoretical analyses for fitness-proportional selection for the case of the ONEMAX function. The *Simple Genetic Algorithm (SGA)* has been analysed with fitness-proportional selection for parent selection in [16], [20], [21].

Most relevant to this work is the work by Happ, Johannsen, Klein, and Neumann [12], who analysed a variant of the (1+1) EA using fitness-proportional selection and showed that it needs exponential time to evolve a fitness of at least $(1+\varepsilon)n/2$ on ONEMAX with high probability. Their algorithm can be seen as a special case of the (μ+1) EA with probabilistic crowding for $\mu = 1$. Our result is similar to the result in [12], but it holds for arbitrary population sizes μ and it applies to general classes of functions with bounded gradients.

We first give a formal notion for bounded gradients.

Definition 4. *For two functions $\alpha := \alpha(n) \geq 1$ and $0 \leq \beta := \beta(n) \leq n$ we say that a function $f\colon \{0,1\}^n \to \mathbb{R}_0^+$ has (α, β)-bounded gradient if for every global optimum x^* of f and all search points x, y with $\mathrm{H}(y, x^*) < \mathrm{H}(x, x^*) \leq \beta$ we have*

$$f(y) \leq f(x) \cdot \alpha^{\mathrm{H}(x,x^*) - \mathrm{H}(y,x^*)}.$$

This definition, intuitively, states that each time the Hamming distance to an optimum x^* is decreased by 1, the fitness can only increase at most by a factor of α. This condition only has to hold for search points within a Hamming ball of radius β from the set of global optima; we do not care about search points further away from global optima.

Note that, if a function has (α, β)-bounded gradient then it also has (α', β')-bounded gradient for any larger gradient $\alpha' \geq \alpha$ and smaller radius $\beta' \leq \beta$. Thus when characterising a fitness function, the smallest possible value for α and the largest possible value of β will give the strongest conditions. Any result that assumes an (α', β')-bounded gradient also holds for an (α, β)-bounded gradient.

We argue that the condition of bounded gradients emerges naturally in many problems if there is some degree of smoothness. The condition can be natural and very easy to verify. For example, a function with maximum fitness $f_{\max}$ has $(2, \beta)$-bounded gradient if all solutions within Hamming distance at most β of any optimum have fitness at least $f_{\max}/2$. This is the case for TWOMAX.

In a more general sense, a sufficient condition is that the effect of each variable on fitness is limited. The following theorem formalises this.

Theorem 5. *Consider any pseudo-Boolean function f with $f_{\max} := \max_x f(x)$, for which flipping one bit only changes the fitness by a value in $[-d, d]$, for $d \in \mathbb{R}^+$. Then f is of $(1 + d/(f_{\max} - d\beta), \beta)$-bounded gradient for every $\beta < f_{\max}/d$. Choosing $\beta := f_{\max}/(2d)$ this yields a $(1 + 2d/f_{\max}, f_{\max}/(2d))$-bounded gradient.*

Proof: Consider an arbitrary optimum x^* and Hamming neighbours x, y with $\mathrm{H}(y, x^*) + 1 = \mathrm{H}(x, x^*) \leq \beta$. Then $f(x) \geq f(y) - d$ and $f(x) \geq f_{\max} - d \cdot \mathrm{H}(x, x^*) \geq f_{\max} - d\beta$ as every step of a shortest Hamming path from x^* to x decreases the fitness by at most d. Together,

$$\frac{f(y)}{f(x)} \leq \frac{f(x) + d}{f(x)} = 1 + \frac{d}{f(x)} \leq 1 + \frac{d}{f_{\max} - d\beta}.$$

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

The claim for general x, y with $\mathrm{H}(y, x^*) < \mathrm{H}(x, x^*) \leq \beta$ follows by a trivial induction. ■

For the functions ONEMAX and TWOMAX Theorem 5 applies with $d = 1$ and $f_{\max} = n$, yielding a very small value of α, $\alpha = 1 + 2/n$ within a large radius of $\beta = n/2$ around global optima. Note that for TWOMAX the gradient bound of $\alpha = 1 + 2/n$ even holds within the maximum Hamming distance of $\beta = n$ as for $x^* \in \{0^n, 1^n\}$ the fitness increases with increasing Hamming distance $\mathrm{H}(x, x^*)$ in the area of search points x with $\mathrm{H}(x, x^*) > n/2$ and thus $f(y)/f(x) < 1 \leq \alpha$ for $n/2 \leq \mathrm{H}(y, x^*) < \mathrm{H}(x, x^*)$. Hence TWOMAX is $(1 + 2/n, n)$-gradient bounded.

We also provide additional examples for (α, β)-bounded gradients in well-known combinatorial problems, showing that this notion can be applied in a much wider setting.

The famous MAXSAT problem provides a collection of clauses in conjunctive normal form (CNF) and asks for an assignment of variables that maximises the number of satisfied clauses. For instances with m clauses the optimal value is $f_{\max} \in [m/2, m]$ as one of the two assignments 0^n and 1^n must satisfy at least $m/2$ clauses. The MAXSAT problem is NP-hard, even in the case where all clauses only have two literals.

The well-known VERTEX COLOURING problem asks for an assignment of colours to vertices in an undirected graph such that no two adjacent vertices share the same colour. It is NP-hard in general, but we consider the special case of 2 colours for which efficient algorithms are known. For 2 colours we have a binary encoding where each bit indicates the colour of one corresponding vertex, and the fitness function is taken as the number of correctly coloured edges. A closely related problem is inspired from Ising models in physics: the encoding is the same, but an edge is correctly coloured if both vertices have *the same* colour. Both problems are equivalent for bipartite graphs[4]. For both settings we also have $f_{\max} \in [m/2, m]$ where m is the number of edges. This is because a random initial colouring colours $m/2$ edges correctly in expectation, hence solutions of fitness at least $m/2$ must exist.

Corollary 6. *Theorem 5 implies the following:*

1) ONEMAX *is of* $(1 + 2/n, n/2)$*-bounded gradient.*
2) TWOMAX *is of* $(1 + 2/n, n)$*-bounded gradient.*
3) MAXSAT *(maximising the number of satisfied CNF-clauses) with* m *clauses is of* $(1 + 4d/m, m/(4d))$*-bounded gradient if every variable only appears in at most* d *clauses.*
4) *Maximising the number of correctly coloured edges in* VERTEX COLOURING *with 2 colours or simple Ising models [7], [8], [29] in graphs with* m *edges is of* $(1 + 4d/m, m/(4d))$*-bounded gradient if the graph has maximum degree* d*. For* d*-regular graphs (all nodes have degree* d*) like cycles or toroids we have* $m = dn/2$ *yielding a* $(1 + 8/n, n/8)$*-bounded gradient.*

Examples of functions that are *not* gradient-bounded include functions with ridges such as LEADINGONES or RIDGE. For instance, the optimum of LEADINGONES, 1^n of fitness n, has a Hamming neighbour 01^{n-1} of fitness 0. Many pseudo-Boolean problems can be classed as having bounded gradients, though, and our analysis of probabilistic crowding will apply to large classes of such functions.

In the following we now fix an optimum $x^* \in \mathrm{OPT}$. We will show that the probability of reaching x^* efficiently is very small. For functions with multiple optima, we then apply a union bound to show that the probability of reaching any optimum is still very small.

The following lemma bounds the drift in the Hamming distance to a fixed optimum x^*. For simplicity we assume an (α, n)-bounded gradient, that is, the gradient is bounded by α everywhere ($\beta = n$). The following theorems will only require a much laxer condition on β, though. To ease readability, we use shorthands $\Pr(\Delta H = d)$ for $\Pr(\mathrm{H}(x, x^*) - \mathrm{H}(y, x^*) = d \mid x)$ and $\Pr(\Delta H = \pm d)$ for $\Pr(\Delta H \in \{-d, +d\})$ in the remainder.

Lemma 7. *Let* x *be the selected parent,* y *be the offspring, and* $z \in \{x, y\}$ *be the individual selected for survival. Then for any function* f *with* (α, n)*-bounded gradient and for all global optima* x^* *and all search points* x*,*

$$\mathrm{E}[\mathrm{H}(x, x^*) - \mathrm{H}(z, x^*) \mid x] \leq \frac{\mathrm{H}(x, x^*) - n/2}{n} + \frac{e(\alpha - 1)}{2}.$$

If the gradient is only (α, β)*-bounded for* $\beta < n$ *then the inequality holds for all* x, x^* *with* $\mathrm{H}(x, x^*) \leq \beta - \log n$ *when adding a term* $n^{-\omega(1)}$ *to the right-hand side.*

In the remainder of the paper we may abbreviate $\mathrm{H}(x, x^*)$ as $\mathrm{H}(x)$ for brevity if the second argument x^* is obvious.

Proof: We first analyse the expected distance of the mutant y before survival selection. In expectation $\mathrm{H}(x)/n$ bits that are different to x^* flip to agree with x^*. Likewise, $(n - \mathrm{H}(x))/n$ bits that agree with x^* flip to disagree with x^*. Hence $\mathrm{E}[\mathrm{H}(y)] = \mathrm{E}[\mathrm{H}(x)] + (n - \mathrm{H}(x))/n - \mathrm{H}(x)/n = \mathrm{E}[\mathrm{H}(x)] + (n - 2\mathrm{H}(x))/n$ and

$$\mathrm{E}[\mathrm{H}(x) - \mathrm{H}(y) \mid x] = \frac{2\mathrm{H}(x) - n}{n}. \quad (5)$$

We now use this inequality to analyse the distance difference $\mathrm{H}(z) - \mathrm{H}(x)$ after survival selection. Observe that this difference is 0 in case $z = x$. Hence only generations where y is selected for survival contribute to $\mathrm{E}[\mathrm{H}(x) - \mathrm{H}(z) \mid x]$. The latter can be written as follows.

$$\mathrm{E}[\mathrm{H}(x) - \mathrm{H}(z) \mid x] = \sum_{d=-\infty}^{\infty} \Pr(\Delta H = d) \cdot d \cdot \frac{f(y)}{f(x) + f(y)}$$

Using that

$$\frac{f(y)}{f(x) + f(y)} = \frac{1}{2} + \frac{1}{2} \cdot \frac{f(y) - f(x)}{f(y) + f(x)},$$

---

[4] There is a simple bijection between the two problems: flipping all vertices of one set of the bipartition turns all monochromatic edges into bichromatic edges and vice versa. The performance of any unbiased randomised search heuristic is identical for both problems if the graph is bipartite.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

we get

$$\begin{aligned}
&\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(z)\mid x]\\
&=\sum_{d=-\infty}^{\infty}\Pr(\Delta H=d)\cdot d\cdot\left(\frac{1}{2}+\frac{1}{2}\cdot\frac{f(y)-f(x)}{f(y)+f(x)}\right)\\
&=\frac{1}{2}\sum_{d=-\infty}^{\infty}\Pr(\Delta H=d)\cdot d\\
&\qquad+\frac{1}{2}\sum_{d=-\infty}^{\infty}\Pr(\Delta H=d)\cdot d\cdot\frac{f(y)-f(x)}{f(y)+f(x)}.
\end{aligned}$$

The first sum is $\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(y)]/2$ by definition of the expectation, and we already know from (5) that $\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(y)]/2=(\mathrm{H}(x)-n/2)/n$.

The second sum can be bounded as follows. Using that the gradient is bounded by $\alpha:=\alpha(n)$, we get for $d\geq 1$

$$\frac{f(y)-f(x)}{f(y)+f(x)}=\frac{f(y)/f(x)-1}{f(y)/f(x)+1}\leq\frac{\alpha^d-1}{\alpha^d+1}$$

and for $d\leq -1$ we get

$$\frac{f(y)-f(x)}{f(y)+f(x)}=\frac{1-f(x)/f(y)}{f(x)/f(y)+1}\geq\frac{1-\alpha^{|d|}}{1+\alpha^{|d|}}=-\frac{\alpha^{|d|}-1}{\alpha^{|d|}+1}$$

and thus

$$\begin{aligned}
&\frac{1}{2}\sum_{d=-\infty}^{\infty}\Pr(\Delta H=d)\cdot d\cdot\frac{f(y)-f(x)}{f(y)+f(x)}\\
&\leq\frac{1}{2}\sum_{d=-\infty}^{\infty}\Pr(\Delta H=d)\cdot|d|\cdot\frac{\alpha^{|d|}-1}{\alpha^{|d|}+1}.
\end{aligned}$$

Using that the summand for $d=0$ is 0 and the summand for $d>0$ is equal to the summand for $-d$, we get an upper bound of

$$\frac{1}{2}\sum_{d=1}^{\infty}\Pr(\Delta H=\pm d)\cdot d\cdot\frac{\alpha^d-1}{\alpha^d+1}.$$

To simplify this bound we exploit that $\Pr(\Delta H=\pm d)\leq 1/(d!)$ as it is necessary to flip at least d bits. There are $\binom{n}{d}$ ways of choosing d bits that are guaranteed to flip, and the probability of flipping the chosen bits is n^{-d}. The remaining $n-d$ bits can flip or stay the same. Hence the probability of flipping at least d bits is at most $\binom{n}{d}(1/n)^d\leq 1/(d!)$. Thus the second sum is bounded from above by

$$\frac{1}{2}\sum_{d=1}^{\infty}\frac{d}{d!}\cdot\frac{\alpha^d-1}{\alpha^d+1}=\frac{1}{2}\sum_{d=0}^{\infty}\frac{1}{d!}\cdot\frac{\alpha^{d+1}-1}{\alpha^{d+1}+1}.$$

Using $\frac{\alpha^{d+1}-1}{\alpha^{d+1}+1}=\frac{2\alpha^{d+1}}{\alpha^{d+1}+1}-1$, we get

$$\begin{aligned}
&\frac{1}{2}\left(\sum_{d=0}^{\infty}\frac{1}{d!}\cdot\frac{2\alpha^{d+1}}{\alpha^{d+1}+1}-\sum_{d=0}^{\infty}\frac{1}{d!}\right)\\
&=\frac{1}{2}\left(\alpha\sum_{d=0}^{\infty}\frac{1}{d!}\cdot\frac{2\alpha^{d}}{\alpha^{d+1}+1}-e\right).
\end{aligned}$$

Finally, we use that for all $\alpha\geq 1$, $\sum_{d=0}^{\infty}\frac{1}{d!}\cdot\frac{2\alpha^d}{\alpha^{d+1}+1}\leq e$, which is proven in Lemma 15 in the appendix. This yields the claimed bound $\frac{1}{2}(\alpha e-e)=e(\alpha-1)/2$.

For the last statement we bound the possible error introduced by a possible absence of a gradient bound beyond radius β. For all x with $\mathrm{H}(x)\leq\beta-\log n$, we have

$$\begin{aligned}
\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(z)\mid x]&=\sum_{d=-\infty}^{\beta-\mathrm{H}(x)}\Pr(\Delta H=d)\cdot d\cdot\frac{f(y)}{f(x)+f(y)}\\
&\quad+\sum_{d=\beta-\mathrm{H}(x)+1}^{\infty}\Pr(\Delta H=d)\cdot d\cdot\frac{f(y)}{f(x)+f(y)}
\end{aligned}$$

where the first sum is bounded by $e(\alpha-1)/2$ as before since for these values of d the gradient bound holds. The second sum is crudely bounded from above by

$$\sum_{d=\beta-\mathrm{H}(x)+1}^{\infty}\Pr(\Delta H=d)\cdot n\cdot 1\leq n\cdot\Pr(\Delta H\geq\log n).$$

Since at least $\log n$ bits have to flip in one mutation for $\Delta H\geq\log n$ to hold and the probability of flipping at least $\log n$ bits is at most $1/((\log n)!)=n^{-\omega(1)}$, we obtain an additive term of $n\cdot n^{-\omega(1)}=n^{-\omega(1)}$ as claimed. ■

Lemma 7 gives an important lesson. Assume that the survivor z was chosen uniformly between x and y. This would yield a highly inefficient blind random walk as the fitness is not taken into account. Then we would have

$$\begin{aligned}
&\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(z)\mid x]\\
&=\frac{1}{2}\cdot\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(y)\mid x]+\frac{1}{2}\cdot\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(x)\mid x]\\
&=\frac{\mathrm{H}(x)-n/2}{n}
\end{aligned}$$

using (5) and $\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(x)\mid x]=0$. Lemma 7 states that compared to this setting, a fitness-proportional selection of z only gives a bias of at most $e(\alpha-1)/2$. For OneMax and TwoMax, plugging in $\alpha:=1+2/n$, this bias is $\mathrm{O}(1/n)$ and hence vanishingly small. In other words, Lemma 7 quantifies the observation that in the considered context, fitness-proportional selection is very similar to uniform selection and each lineage behaves very similarly to a blind random walk. We now use Lemma 7 and the negative drift theorem (Theorem 2) to prove a strong negative result on the performance of the (μ+1) EA with probabilistic crowding.

More precisely, we will show that on functions with bounded gradient, including OneMax and TwoMax, the (μ+1) EA with probabilistic crowding does not perform noticeably better than random search. Note that, for any fixed optimum x^*, the expected Hamming distance of a search point chosen u. a. r. to x^* is $n/2$. The following theorem shows that the (μ+1) EA with probabilistic crowding does not evolve any solutions significantly closer to *any* optimum than $n/2$, even given exponential time.

Theorem 8. *Let $\xi,\varepsilon>0$ be constant. Let f be any function of $(\alpha,n/2)$-bounded gradient with $\alpha\leq 1+\varepsilon/(2e)-\Omega(1)$ and at most $2^{\xi n}$ global optima. With probability $1-2^{-\Omega(n)}$ the (μ+1)* EA *with probabilistic crowding will not have found a search point within Hamming distance $(1-\varepsilon)n/2$ of any optimum of f in 2^{cn} function evaluations, for every population size μ and small enough constants $\xi,c>0$ that may depend on ε.*

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

Proof: We first assume a (α, n)-bounded gradient and later on argue that the analysis still holds if $\beta := n/2$ is used instead. Fix a global optimum x^*. We show that no search point within Hamming distance $(1-\varepsilon)n/2$ of x^* is reached in 2^{cn} function evaluations with the claimed probability, and then use a union bound over all optima to prove the statement.

Note that the statement does not restrict the population size μ in any way. We may however safely assume that $\mu = 2^{o(n)}$ as if $\mu \geq 2^{c'n}$ for any constant $0 < c' < 1$, the statement follows immediately (for $c := c'$) as the first $2^{c'n}$ search points contain an optimal search point only with probability at most $2 \cdot 2^{-n} \cdot 2^{c'n} = 2^{-\Omega(n)}$ as $c' < 1$.

Further note that in the absence of crossover, probabilistic crowding evolves μ independent lineages as any offspring only competes directly with its parent. This allows us to consider a fixed lineage and later on apply a union bound over μ lineages.

We show that the probability of any fixed lineage reaching a Hamming distance of $(1-\varepsilon)n/2$ in 2^{cn} generations is $2^{-\Omega(n)}$. Taking the union bound over all lineages yields that the probability of reaching such a distance is bounded by $\mu \cdot 2^{-\Omega(n)} = 2^{o(n)} \cdot 2^{-\Omega(n)} = 2^{-\Omega(n)}$, which implies the claim.

Now focus on one lineage. By standard Chernoff bounds (see [5]), the probability of initialising the lineage with an initial search point of Hamming distance at most $(1-\varepsilon/2)n/2$ to x^* is $2^{-\Omega(n)}$. If this rare failure event does not happen, the lineage needs to decrease an initial distance from a value at least $(1-\varepsilon/2)n/2$ to a value at most $(1-\varepsilon)n/2$. We apply the negative drift theorem to the Hamming distance of the current individual in our lineage to x^* to show that this does not happen in 2^{cn} generations with probability $1 - 2^{-\Omega(n)}$. The interval chosen will be from $a := (1-\varepsilon)n/2$ to $b := (1-\varepsilon/2)n/2$; note that it has length $\varepsilon n/4$.

Let x be the selected parent, y be the offspring, and $z \in \{x, y\}$ be the individual selected for survival. We establish the two conditions of the negative drift theorem. The first condition follows from Lemma 7 as the drift is at most

$$\begin{aligned} \frac{\mathrm{H}(x) - n/2}{n} + \frac{e(\alpha-1)}{2} &\leq \frac{(1-\varepsilon/2)n/2 - n/2}{n} + \frac{e(\alpha-1)}{2} \\ &= -\frac{\varepsilon}{4} + \frac{e(\alpha-1)}{2} = -\Omega(1) \end{aligned}$$

as α is a constant strictly less than $1 + \varepsilon/(2e)$ by assumption.

The second condition follows easily from properties of standard bit mutation: the difference in Hamming distance $|\mathrm{H}(z) - \mathrm{H}(x)|$ is clearly bounded by the number of flipping bits. The probability of flipping d bits in a standard bit mutation is at most $1/(d!) \leq 2/2^d$ for all $d \geq 1$. This proves the second condition when choosing $r := 2$ and $\delta := 1$. Invoking the negative drift theorem yields that the probability of one lineage reaching a search point with a distance at most $(1-\varepsilon)n/2$, starting with a distance at least $(1-\varepsilon/2)n/2$, in $2^{c'\varepsilon n/8}$ steps, for some constant $c' > 0$, is at most $2^{-\Omega(\varepsilon n/4)} = 2^{-\Omega(n)}$. Choosing $c := c'\varepsilon/8$ yields the claimed time bound.

Applying the above arguments for every optimum and using a union bound over all at most $2^{\xi n}$ optima, the probability of finding any search point within Hamming distance $(1-\varepsilon)n/2$ of any optimum within the stated time is still bounded by $2^{\xi n} \cdot 2^{-\Omega(n)} = 2^{\Omega(n)}$ if ξ is sufficiently small.

Now, if β is lowered to $n/2$, that is, f only has an α-bounded gradient amongst search point with Hamming distance at most $n/2$ to any optimum, the above arguments essentially remain unaffected. The negative drift theorem only requires a drift estimate for all search points x within the considered interval, that is, for $\mathrm{H}(x) \leq (1-\varepsilon/2)n/2$. Since $(1-\varepsilon/2)n \leq n/2 - \log n$, the last statement from Lemma 7 applies, introducing an error term of $n^{-\omega(1)}$ that is absorbed in the drift bound $-\Omega(1)$. ■

Theorem 8 states that probabilistic crowding has a poor performance since it is not possible to evolve search points that are significantly better than those found by random search, even given exponential time. The fitness-proportional selection mechanism embedded in the selection does not give a high enough selection pressure to favour the fitter individuals. One standard way of improving the performance of fitness-proportional selection is to scale the fitness. An extreme way of using scaling is to use exponential scaling, that is, using $\alpha^{f(x)}$ instead of $f(x)$, for some base α.

A theoretical study in the context of evolutionary algorithms with fitness-proportional selection using this scheme for scaling the fitness was done in [16]. Neumann, Oliveto and Witt [16] showed for the first time that even with a large population, a mutation-based EA with fitness-proportional selection with high probability needs exponential time to find the global optimum of ONEMAX. However, by using exponential scaling with a base equal to the population size μ the EA turns into an efficient algorithm for ONEMAX if μ is large enough.

For f being ONEMAX or TWOMAX, scaling to α^f gives functions with an (α, n)-bounded gradient. An obvious question is how large α needs to be to yield a large enough selection pressure that leads to good performance in the (μ+1) EA with probabilistic crowding. For instance, $\exp(f(x))$ (that is, $\alpha := e$) scales the fitness exponentially, yielding enormously large fitness values of up to e^n. Is this drastic way of scaling the fitness sufficient? Note that Theorem 8 only applies to small values of α because of the condition $\alpha \leq 1 + \varepsilon/(2e) - \Omega(1)$. It is not clear whether α values mildly violating this condition will yield good performance.

The following theorem shows that for moderate α values, even constant and superconstant ones, performance is still very poor. The result applies to functions with any $(\alpha, n/\alpha)$-bounded gradient. This includes functions that only have a bounded gradient within a small radius around the set of optima as α can be chosen arbitrarily large to satisfy a gradient bound and to make the radius $\beta := n/\alpha$ arbitrarily small. In particular, the theorem applies to all examples from Corollary 6.

Theorem 9. *Let f be any function with an $(\alpha, n/\alpha)$-bounded gradient and f having at most $2^{\xi n/\alpha}$ global optima for a constant $\xi > 0$. With probability $1 - 2^{-\Omega(n/\alpha)}$ the (μ+1) EA with probabilistic crowding will not have found an optimum, or any search point within Hamming distance $n/(9\alpha)$ of any optimum, in $2^{cn/\alpha}$ function evaluations, for every population size μ and small enough constants $\xi, c > 0$.*

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

Note that the strength of the statement is parameterised by n/α in terms of the lower time bound, the probability bound, the number of optima and a lower bound on the Hamming distance to the set of optima. For $\alpha = \mathrm{O}(n^{1-\varepsilon})$, $\varepsilon > 0$ constant, we get an exponential lower bound that holds with overwhelming probability to get within Hamming distance $\mathrm{O}(n^{\varepsilon})$ of any optimum, even if there are exponentially many optima. The statement becomes trivial when $\alpha = \Omega(n)$ as then the claim may only give a lower bound of 1 generation.

Proof of Theorem 9: We assume that $n/\alpha = \omega(1)$ as the claim is trivial for $n/\alpha = \mathrm{O}(1)$. We fix a global optimum x^* and apply the negative drift theorem with self-loops to the distance interval $n/(9\alpha) \le \mathrm{H}(x, x^*) \le n/(8.5\alpha)$. Using the same arguments as in the proof of Theorem 8, the initial distance in each lineage is at least $n/(8.5\alpha)$ with probability $1 - 2^{-\Omega(n)}$, unless μ is so large that the claim holds trivially.

For search points x in the considered interval we estimate the positive and the negative summands in the drift separately. We pessimistically assume that the gradient is always exactly α as this clearly maximises the drift. As in the proof of Theorem 8 we first assume an (α, n)-bounded gradient and then explain how the arguments change for the radius $\beta = n/\alpha$ from the statement. For the positive summands, we pessimistically assume that all improvements are always accepted and then apply Lemma 3 in [22], which states that $\Pr(\Delta H = d) \le \frac{1.14}{d!} \cdot \left(\frac{\mathrm{H}(x)}{n}\right)^d$, to bound the probability of jumps closer to the optimum.

$$\begin{aligned}\sum_{d=1}^{\infty} \Pr(\Delta H = d) \cdot d &\le \sum_{d=1}^{\infty} \frac{1.14}{d!} \cdot \left(\frac{\mathrm{H}(x)}{n}\right)^d \cdot d \\ &= \frac{1.14\mathrm{H}(x)}{n} \sum_{d=0}^{\infty} \frac{1}{d!} \cdot \left(\frac{\mathrm{H}(x)}{n}\right)^d \\ &= \frac{1.14\mathrm{H}(x)}{n} \cdot e^{\mathrm{H}(x)/n} \le \frac{1.3}{8.5\alpha} \qquad (6)\end{aligned}$$

where the last inequality follows from $\mathrm{H}(x)/n \le 1/(8.5\alpha) \le 1/8.5$. Note for later use that the transition probabilities decrease exponentially in d.

The negative summands are bounded from above by pessimistically only considering the summand for $d = -1$ and assuming that the gradient is exactly α:

$$\begin{aligned}&\sum_{d=-\infty}^{-1} \Pr(\Delta H = d) \cdot d \cdot \frac{f(y)}{f(x) + f(y)} \\ &\le -\Pr(\Delta H = -1 \mid x) \cdot \frac{1/\alpha}{1 + 1/\alpha} \le -\frac{7.5}{17e\alpha} \qquad (7)\end{aligned}$$

where the last step follows from $\Pr(\Delta H = -1 \mid x) \ge (n - \mathrm{H}(x))/(en) \ge (n - n/8.5)/(en) = 7.5/(8.5e)$ and $1 + 1/\alpha \le 2$. Combining (6), (7) and $1.3/8.5 < 7.5/17e$, the drift is at most $\frac{1.3}{8.5\alpha} - \frac{7.5}{17e\alpha} = -\Omega(1/\alpha)$.

We claim that the term $\Omega(1/\alpha)$ is of the same order as the converse of the self-loop probability as this is required for the first condition of the negative drift theorem with self-loops. The probability of increasing the fitness is clearly bounded by the positive drift, which we already bounded from above by $\mathrm{O}(\mathrm{H}(x)/n) = \mathrm{O}(1/\alpha)$. The probability of decreasing the fitness is at most $\frac{1}{1+\alpha} = \mathrm{O}(1/\alpha)$ as the (μ+1) EA has to accept a fitness decrease of by a factor of α (or a larger power of α in the case of longer jumps). Using these arguments along with the familiar bound of $1/d!$ for the probability of changing the distance by an absolute value of d, it is also easy to see that the second condition for the negative drift theorem with self-loops is met. This implies the claimed lower bound.

Finally, the results hold for $\beta = n/\alpha$ instead of $\beta = n$ as argued in the proof of Theorem 8: in order to notice the absence of a gradient for a parent in the distance interval $n/(9\alpha) \le \mathrm{H}(x) \le n/(8\alpha)$, at least $(7/8)n/\alpha$ bits need to flip in one mutation. The probability for such an event is $2^{-\Omega(n/\alpha)}$.

Applying these arguments for all optima x^* and taking a union bound over at most $2^{\xi n/\alpha}$ optima still gives a probability bound of $1 - 2^{\xi n/\alpha} \cdot 2^{-\Omega(n/\alpha)}$ if ξ is chosen small enough. ■

We summarise our negative results for selected examples, including all problems from Corollary 6.

Corollary 10. *With overwhelming probability, the* (μ+1) EA *with probabilistic crowding requires exponential time*

1) *to get within Hamming distance* $(1 - \varepsilon)n/2$ *of any optimum on* OneMax *or* TwoMax *(by Theorem 8)*
2) *to find any optimum on scaled functions* $\alpha^{\text{OneMax}(x)}$ *and* $\alpha^{\text{TwoMax}(x)}$ *if* $\alpha = \mathrm{O}(n^{1-\varepsilon})$ *(by Theorem 9)*
3) *to get within Hamming distance* $m/(36d)$ *of any optimum for* MaxSat *on any instance with* m *clauses where each variable only appears in at most* d *clauses, provided* $m \le 4d(n-1)$ *(by Theorem 9 with* $\alpha := 4dn/m$*, the bound on* m *ensuring that* $1 + 4d/m \le \alpha$*)*
4) *to get within Hamming distance* $n/72$ *of any optimum for* Vertex Colouring *or the Ising model on any regular graph (by Theorem 9 with* $\alpha := 8$*).*

Corollary 10 shows that even exponential scaling with a base of $\alpha = \mathrm{O}(n^{1-\varepsilon})$ is not sufficient for optimising scaled versions of OneMax and TwoMax. The following theorem shows that a base of order n is sufficient to guarantee efficient runtimes on a scaled OneMax, and a high probability of finding both optima of TwoMax efficiently. This result also explains why Theorem 9 does not give a meaningful lower bound when $\alpha = \Omega(n)$.

Theorem 11. *The* (μ+1) EA *with probabilistic crowding and polynomial* μ *finds an optimum on* $\alpha^{\text{OneMax}(x)}$ *with* $\alpha \ge (1 + \Omega(1))en$ *in expected time* $\mathrm{O}(\mu n \log n)$*.*

On $\alpha^{\text{TwoMax}(x)}$ *with* $\alpha \ge (1 + \Omega(1))en$ *the* (μ+1) EA *with probabilistic crowding finds a population consisting of only global optima in expected time* $\mathrm{O}(\mu n \log n)$*. In that case the population contains both global optima with probability at least* $1 - 2^{-\mu+1}$*.*

Proof: We focus on one lineage only and show that the optimum of OneMax is reached in $\mathrm{O}(n \log n)$ mutation steps applied to said lineage. Let x be the current search point, then

$$\mathrm{E}[\mathrm{H}(x) - \mathrm{H}(z) \mid x] = \sum_{d=-\infty}^{\infty} \Pr(\Delta H = d) \cdot d \cdot \frac{\alpha^d}{\alpha^d + 1}.$$

Note that the summand for $d = 0$ is 0. For $d \ge 1$ we bound the summands from below by $\Pr(\Delta H = d) \cdot d \cdot \frac{\alpha}{\alpha+1}$ using

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

$\frac{\alpha^d}{\alpha^d+1} \geq \frac{\alpha}{\alpha+1}$. For $d \leq -1$ the summands are negative, hence we use $\frac{\alpha^d}{\alpha^d+1} \leq \frac{\alpha^{-1}}{\alpha^{-1}+1} = \frac{1}{1+\alpha}$ to bound these summands from below. Together we obtain a drift bound of

$$\begin{aligned}
&\frac{\alpha}{\alpha+1}\sum_{d=1}^{\infty}\Pr(\Delta H = d)\cdot d + \frac{1}{\alpha+1}\sum_{d=-\infty}^{-1}\Pr(\Delta H = d)\cdot d\\
&= \frac{\alpha-1}{\alpha+1}\sum_{d=1}^{\infty}\Pr(\Delta H = d)\cdot d + \frac{1}{\alpha+1}\sum_{d=-\infty}^{\infty}\Pr(\Delta H = d)\cdot d\\
&= \frac{\alpha-1}{\alpha+1}\sum_{d=1}^{\infty}\Pr(\Delta H = d)\cdot d + \frac{\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(y)\mid x]}{\alpha+1}.
\end{aligned}$$

Using that $\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(y)\mid x] \geq -1$ by (5) and lower-bounding the sum by the term of $d = 1$ (that is, flipping only one of $\mathrm{H}(x)$ incorrect bits), $\sum_{d=1}^{\infty}\Pr(\Delta H = d)\cdot d \geq \mathrm{H}(x)/(en)$, we get a drift of at least

$$\begin{aligned}
&\frac{\alpha-1}{\alpha+1}\cdot\frac{\mathrm{H}(x)}{en} - \frac{1}{\alpha+1}\\
&\geq \left(1-\mathrm{O}\left(\frac{1}{n}\right)\right)\cdot\frac{\mathrm{H}(x)}{en} - \frac{1}{(1+\Omega(1))en} = \Omega\left(\frac{\mathrm{H}(x)}{n}\right).
\end{aligned}$$

So the drift in one mutation step of the considered lineage is $\Omega(\mathrm{H}(x)/n)$. Since the lineage's current search point is mutated with probability $1/\mu$, the drift in one generation is $\Omega(\mathrm{H}(x)/(\mu n))$. Now the bound $\mathrm{O}(\mu n \log n)$ follows from a straightforward application of the multiplicative drift theorem with $\mathrm{H}(x)$ as distance function, $g_{\max} = n$ and $\delta = \Omega(1/(\mu n))$.

The same analysis also applies to TwoMax for the time one fixed lineage finds an optimum. However, the second statement of this theorem claims that *all* μ lineages will have found a global optimum in expected time $\mathrm{O}(\mu n \log n)$. Hence we need to show that even the slowest out of μ lineages will still finish in this time.

To this end, we exploit that the multiplicative drift theorem provides tail bounds. Since μ is polynomial, there is a constant $d > 0$ such that $\mu \leq n^d$. We then apply the multiplicative drift theorem's tail bound with parameter $c := d\ln(n) + 1$; this yields that the probability of a fixed lineage not finding an optimum in time $\mathrm{O}(\mu n \log n)$ is at most $e^{-c} \leq 1/(e\mu)$. By a union bound, the probability that there is a lineage that has not found an optimum in this time is at most $1/e$. If this happens, we repeat the above arguments with another, fresh period of $\mathrm{O}(\mu n \log n)$ generations. Since in every period all lineages find global optima with probability at least $1 - 1/e$, in expectation only $\mathrm{O}(1)$ such periods are needed before this happens. This implies a bound of $\mathrm{O}(\mu n \log n)$ generations.

The probability $1 - 2^{-\mu+1}$ follows from the fact that all lineages are independent. Every lineage has equal probabilities of ending up in 0^n or 1^n. The probability that, once all μ lineages have found global optima, they all end up in the same global optimum is $2 \cdot 2^{-\mu} = 2^{-\mu+1}$. Hence with probability $1 - 2^{-\mu+1}$ the population contains both global optima. ■

IV. GENERALISED CROWDING

Introduced by Galán and Mengshoel [10], generalised crowding uses the same pairing and replacement as probabilistic crowding, but it introduces a scaling factor ϕ that diminishes the fitness of the inferior search point. It allows wider ranges of replacement strategies by adjusting ϕ. In this crowding mechanism, the probability of accepting the offspring y over the parent x is given by:

$$\begin{cases} \frac{f(y)}{f(y)+\phi\cdot f(x)} & \text{if } f(y) > f(x),\\ 0.5 & \text{if } f(y) = f(x),\\ \frac{\phi\cdot f(y)}{\phi\cdot f(y)+f(x)} & \text{if } f(y) < f(x). \end{cases} \tag{8}$$

In the special case where $\phi = 1$ we obtain probabilistic crowding, and in case where $\phi = 0$ we essentially retrieve deterministic crowding as then the better offspring is selected with probability 1. A minor difference is that in case of ties, the offspring is kept with probability $1/2$ in generalised crowding whereas in deterministic crowding the offspring is always preferred in this case.

We use the techniques established for the analysis of probabilistic crowding to study the performance of generalised crowding in the context of TwoMax. Since we already know that the extreme parameter settings $\phi = 1$ and $\phi = 0$ are respectively inefficient and efficient, an obvious question is whether there is a threshold behaviour for the choice of ϕ and where this threshold is located. The following two theorems establish a threshold behaviour around $\phi = \Theta(1/n)$. This means that generalised crowding is only efficient if the parameters are chosen very close to those for deterministic crowding.

Theorem 12. *The* (μ+1) EA *with generalised crowding, polynomial* μ *and* $\phi \leq (1-\Omega(1))/(e^2 n)$ *finds an optimum on* OneMax *in expected time* $\mathrm{O}(\mu n \log n)$.

On TwoMax *it finds a population consisting of only global optima in expected time* $\mathrm{O}(\mu n \log n)$. *In that case the population contains both global optima with probability at least* $1 - 2^{-\mu+1}$.

Proof: As before, we only consider a single lineage and show a bound of $\mathrm{O}(n \log n)$ steps that evolve said lineage, keeping in mind an additional factor of μ for the expected waiting time for such an evolution step.

Using the same notation for x, y, z and $\mathrm{H}(\cdot)$ as before,

$$\begin{aligned}
\mathrm{E}[\mathrm{H}(x)-\mathrm{H}(z)\mid x] &= \sum_{d=1}^{\infty}\Pr(\Delta H = d)\cdot d\cdot\frac{f(y)}{\phi f(x)+f(y)}\\
&\quad+\sum_{d=-\infty}^{-1}\Pr(\Delta H = d)\cdot d\cdot\frac{\phi f(y)}{f(x)+\phi f(y)}.
\end{aligned}$$

Considering only the summand $d = 1$, the first sum is at least

$$\frac{\mathrm{H}(x)}{en}\cdot\frac{f(x)+1}{\phi f(x)+f(x)+1} \geq \frac{\mathrm{H}(x)}{en}\cdot\frac{1}{1+\phi}.$$

For the second sum, we again bound the probability of a jump of length $|d|$ by $1/(|d|!)$ (cf. proof of Lemma 7) and obtain a lower bound of

$$\begin{aligned}
&\sum_{d=-\infty}^{-1}\frac{1}{|d|!}\cdot d\cdot\frac{\phi f(y)}{f(x)+\phi f(y)}\\
&\geq -\sum_{d=1}^{\infty}\frac{1}{d!}\cdot d\cdot\frac{\phi}{1+\phi} = -\frac{\phi}{1+\phi}\sum_{d=0}^{\infty}\frac{1}{d!} = -\frac{e\phi}{1+\phi}.
\end{aligned}$$

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

Together, the drift $\mathrm{E}[\mathrm{H}(x) - \mathrm{H}(z) \mid x]$ is at least

$$\frac{\mathrm{H}(x)}{en} \cdot \frac{1}{1+\phi} - \frac{e\phi}{1+\phi} = \frac{1}{1+\phi}\left(\frac{\mathrm{H}(x)}{en} - e\phi\right) = \Omega\left(\frac{\mathrm{H}(x)}{n}\right)$$

as by assumption $\phi < (1-\Omega(1))/(e^2 n)$. Another application of the multiplicative drift theorem proves the claim.

The statement for TwoMax follows in the same way as in the proof of Theorem 11. ■

The following theorem gives a negative result very similar to Theorem 9 with ϕ/α replacing $1/\alpha$.

Theorem 13. *Let f be any function of $(\alpha, \phi n/\alpha)$-bounded gradient and f having at most $2^{\xi\phi n/\alpha}$ global optima for a constant $\xi > 0$. With probability $1-2^{-\Omega(\phi n/\alpha)}$ the* (μ+1) EA *with generalised crowding and parameter ϕ will not have found an optimum, or any search point within Hamming distance $\phi n/(9\alpha)$ of any optimum, in $2^{c\phi n/\alpha}$ function evaluations, for every population size μ and small enough constants $\xi, c > 0$.*

The proof is nearly identical to the proof of Theorem 9; the negative drift theorem is applied to the interval $\phi n/(9\alpha) \le \mathrm{H}(x) \le \phi n/(8.5\alpha)$ and calculations include an additional factor of ϕ (which retrieves Theorem 9 as a special case for $\phi = 1$). Details are omitted due to space restrictions.

Theorem 13 implies the following, recalling that TwoMax has an $(1+2/n, n)$-bounded gradient.

Corollary 14. *With probability $1-2^{-\Omega(\phi n)}$ the* (μ+1) EA *with generalised crowding and scaling factor ϕ will not have found any optimum, or any search point within Hamming distance $\phi n/9 - \mathrm{O}(1)$ of any optimum, in $2^{c\phi n}$ function evaluations, for every population size μ and a small enough constant $c > 0$.*

This is an exponential time with overwhelming probability if $\phi = \Omega\big(n^{-1+\varepsilon}\big)$ for a constant $\varepsilon > 0$.

V. Experimental Analysis

We provide an experimental analysis as well in order to see how closely the theory matches the empirical performance for reasonable problem sizes. Our analysis is focused on Algorithm 1, its scaled version and generalised crowding for the TwoMax function. We consider exponentially increasing population sizes $\mu \in \{2, 4, 8, \dots, 1024\}$ for a problem size $n = 100$ and for 100 runs.

Since we are interested in proving how good/bad these mechanisms are, we define the following outcomes and stopping criteria for each run. *Success*, both optima of TwoMax have been reached, i. e., the run is stopped if the population contains both 0^n and 1^n. *Failure*, once the run has reached $10\mu n \ln n$ generations and the population does not contain both optima. By [3, Lemma 3.3], this time period is long enough to allow any reasonable (μ+1) EA variant to find one or two global optima with high probability (unless the best fitness on a branch drops frequently). We report the mean of successes and failures for the 100 runs.

For probabilistic crowding (Algorithm 1), and as proved in Theorem 8, for all μ sizes, the method is not able to optimise TwoMax. In all runs the algorithm failed to reach even one optimum, let alone reaching both. Since the algorithm is not able to find any optimum of TwoMax, we ran additional experiments for $n \in \{32, 64, 128, \dots, 16384\}$ and population size $\mu = 32$ to observe how far the best lineages evolve from $n/2$ and/or how close the best individuals get to reach an optimum. In the following we will use box plots for representing some of the statistical data with an extended variant where additionally outliers are identified. In Fig. 1a, we show the best individuals obtained in each of the 100 runs and its variance. As soon as n increases, the best fitness in the population starts to concentrate around $n/2$ and reaching a fitness of $(1+\varepsilon)n/2$ becomes very difficult for all constants $\varepsilon > 0$ as n grows. Even the best outliers start to get closer and closer to the average of the population.

For probabilistic crowding with scaling we would like to observe how the hill-climbing capability of the method is improved with respect to the base α. In order to cover bases very close to 1 as well as larger ones, we vary $\alpha - 1$ exponentially: $\alpha - 1 \in \{2^{-\log n}, 2^{-((\log n)+1)}, \dots, 2^{(\log n)+2}\}$. We test the ($\mu$+1) EA with probabilistic crowding on α^{TwoMax} with $n = 1024$ (we know from Fig. 1a that this problem size is hard for the (μ+1) EA without scaling) and a population size $\mu = 32$ with the same stopping criteria.

In Fig. 1b, we show the best individuals obtained in each of 100 runs, its variance and the number of successes achieved. In this experimental analysis we explore a wide range of replacement rules that range from similar results from the classic probabilistic crowding with a performance close to $n/2$ to a more elitist replacement rule like deterministic crowding. From all the bases α analysed here, we can observe how scaling plays a crucial role in the optimisation process by allowing better individuals to survive and to reach both global optima of TwoMax. As shown in Theorem 11 and in Fig. 1b, an exponential scaling factor with a base of order n is needed to guarantee efficient runtimes on TwoMax.

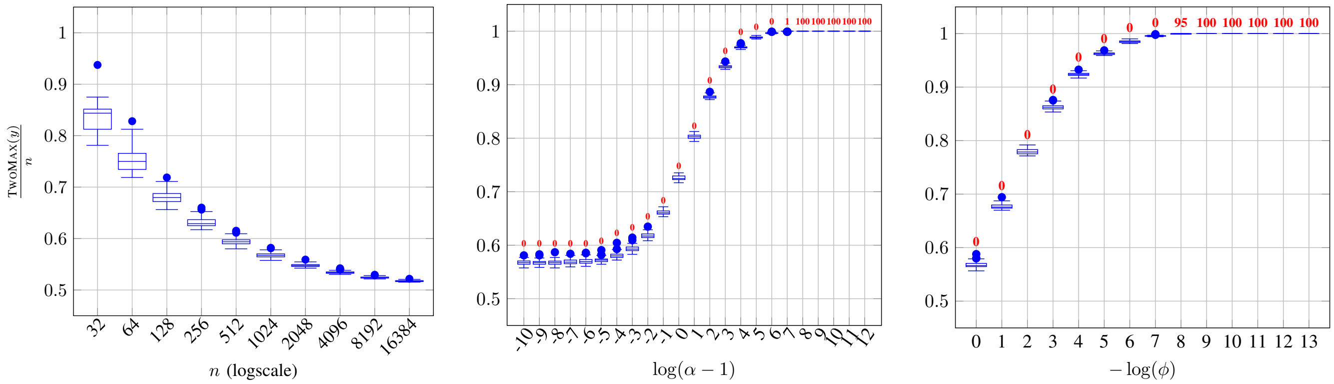
In the same way we analyse the (μ+1) EA with generalised crowding, varying the scaling factor exponentially as $\phi \in \{2^{-((\log n)+3)}, 2^{-((\log n)+2)}, \dots, 2^0\}$. In Fig. 1c, we show the best individuals obtained in each of the 100 runs, its variance and the number of successes achieved. And as stated from Theorem 12, generalised crowding is only efficient if the scaling factor ϕ approaches to 0, i. e., as the scaling factor is getting closer to 0, the algorithm starts emulating the behaviour of deterministic crowding, and the success rate reaches 100. When $\phi = 1$ we can observe a similar behaviour as probabilistic crowding in Fig. 1a.

VI. Conclusion

We have examined theoretically and empirically the behaviour of two different crowding mechanisms embedded into a simple (μ+1) EA on the large class of functions with bounded gradients, including TwoMax. We rigorously proved that probabilistic crowding fails miserably on TwoMax; it is not even able to evolve search points that are significantly better than those found by random search, even when given exponential time. The reason is that fitness-proportional selection for survival selection works very similar to uniform selection, and then the algorithm performs an almost blind search on μ independent lineages. Our negative results apply to a broad range of functions with bounded gradients,

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation



(a) Probabilistic Crowding (b) Probabilistic Crowding with Scaling (c) Generalised Crowding

Fig. 1. The normalised best fitness TwoMax$/n$ reached and number successes achieved among 100 runs at the time both optima were found or the $t = 10\mu n \ln n$ generations have been reached for (1a) $n \in \{32, 64, 128, \dots, 16384\}$ by the (μ+1) EA with probabilistic crowding, (1b) $n = 1024$ by the (μ+1) EA with probabilistic crowding with exponential scaling and base α with $\log(\alpha-1) \in \{-\log n, \dots, (\log n) + 2\}$ and, (1c) for $n = 1024$ by the (μ+1) EA with generalised crowding with exponentially decreasing scaling factor ϕ, choosing $-\log(\phi) \in \{0, \dots, (\log n) + 3\}$ and $\mu = 32$.

which also includes combinatorial problems like MaxSat and Vertex Colouring. For all considered problems, the (μ+1) EA with probabilistic crowding needs exponential time with overwhelming probability to even get reasonably close to any global optimum.

Even when scaling the fitness function exponentially to some base α, finding any optimum of TwoMax takes exponential time for any constant base α. We have shown a threshold behaviour with respect to α at $\alpha = \Theta(n)$ where probabilistic crowding with exponential scaling becomes efficient on TwoMax.

A similar threshold behaviour was proven for generalised crowding, where the fitness of the inferior search point in the comparison between parent and offspring is decreased by a factor of $\phi \in [0,1]$. Here the threshold between efficient and inefficient behaviour on TwoMax is located at $\phi = \Theta(1/n)$. In other words, the fitness of the inferior search point has to be vanishingly small for generalised crowding to be effective.

In addition to strengthening the theoretical foundation of EAs, we believe that our results are highly relevant for practice as they apply to many real-world settings. Our negative results for probabilistic crowding with and without scaling and generalised crowding apply to general classes of functions with bounded gradients, including highly multimodal problems as found in real-world problems. Most runtime analyses of EAs focus on the time to hit a global optimum exactly, which can be of limited relevance to practitioners. Our results go beyond global optimisation as they show that even the time to get within a certain Hamming distance to any optimum is exponential with overwhelming probability if the wrong mechanism is used or parameters are set incorrectly. Our theoretical and empirical results, including the threshold behaviours for parameters α and ϕ, provide solid guidance for practitioners on how to use crowding mechanisms most effectively.

APPENDIX

The following inequality was used in the proof of Lemma 7.

Lemma 15. *For all $\alpha \ge 1$,*

$$\sum_{d=0}^{\infty} \frac{1}{d!} \cdot \frac{2\alpha^d}{\alpha^{d+1}+1} \le e$$

Proof: The claim is obvious for $\alpha = 1$. We show the claim for $\alpha > 1$ by showing that the function is non-increasing in α.

Since $\alpha^{d+1} + 1 \ge 2$, the left-hand side is bounded by e^{α}. Despite being weaker than the claimed bound, it shows that the series is absolutely convergent for all $\alpha \ge 1$. The derivative of $\frac{\alpha^d}{\alpha^{d+1}+1}$ is calculated using the quotient rule $\left(\frac{f}{g}\right)' = \frac{f'g - g'f}{g^2}$, yielding

$$\frac{d\alpha^{d-1}(\alpha^{d+1}+1) - (d+1)\alpha^d \cdot \alpha^d}{(\alpha^{d+1}+1)^2} = \frac{d\alpha^{d-1} - \alpha^{2d}}{(\alpha^{d+1}+1)^2}.$$

Hence the derivative of the series is

$$\sum_{d=0}^{\infty} \frac{1}{d!} \cdot \frac{2d\alpha^{d-1} - 2\alpha^{2d}}{(\alpha^{d+1}+1)^2}.$$

The term for $d = 0$ is $-2/(\alpha+1)^2$. Hence the derivative is at most 0 if and only if

$$\sum_{d=1}^{\infty} \frac{1}{d!} \cdot \frac{2d\alpha^{d-1} - 2\alpha^{2d}}{(\alpha^{d+1}+1)^2} \le \frac{2}{(\alpha+1)^2}$$

or, equivalently,

$$\sum_{d=1}^{\infty} \frac{1}{d!} \cdot \frac{d\alpha^{d-1}(\alpha+1)^2}{(\alpha^{d+1}+1)^2} \le 1 + \sum_{d=1}^{\infty} \frac{1}{d!} \cdot \frac{\alpha^{2d}(\alpha+1)^2}{(\alpha^{d+1}+1)^2}. \tag{9}$$

We show that the left-hand side is at most e and the right-hand side is at least $1 + (e-1) = e$. The left-hand side equals

$$\sum_{d=1}^{\infty} \frac{1}{(d-1)!} \cdot \frac{\alpha^{d-1}(\alpha+1)^2}{(\alpha^{d+1}+1)^2} = \sum_{d=0}^{\infty} \frac{1}{d!} \cdot \frac{\alpha^{d+2} + 2\alpha^{d+1} + \alpha^d}{\alpha^{2d+4} + 2\alpha^{d+2} + 1}$$

The obvious inequality $\alpha^d - 1 \le \alpha^{d+2}(\alpha^d - 1)$ implies $\alpha^{d+2} + \alpha^d \le \alpha^{2d+2} + 1$ and $\alpha^{d+2} + 2\alpha^{d+1} + \alpha^d \le \alpha^{2d+4} + 2\alpha^{d+2} + 1$. Consequently, the left-hand side is at most $\sum_{d=0}^{\infty} \frac{1}{d!} = e$.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

The series on the right-hand side of (9) equals

$$\sum_{d=1}^{\infty} \frac{1}{d!} \cdot \frac{\alpha^{2d+2} + 2\alpha^{2d+1} + \alpha^{2d}}{\alpha^{2d+2} + 2\alpha^{d+1} + 1} \geq \sum_{d=1}^{\infty} \frac{1}{d!} = e - 1.$$

Together, this proves (9) and that the derivative is always at most 0. Hence the claim holds for all $\alpha \geq 1$. ■

ACKNOWLEDGMENT

The authors would like to thank the Consejo Nacional de Ciencia y Tecnología — CONACYT (the Mexican National Council for Science and Technology) for the financial support under the grant no. 409151 and registration no. 264342.

REFERENCES

[1] E. Covantes Osuna and D. Sudholt, “Analysis of the Clearing Diversity-Preserving Mechanism,” in *Proc. of FOGA ’17*, ACM, 2017, pp. 55–63.

[2] ——, “Empirical Analysis of Diversity-Preserving Mechanisms on Example Landscapes for Multimodal Optimisation,” in *Proc. of PPSN XV*, Springer, 2018, pp. 207–219.

[3] ——, “Runtime Analysis of Probabilistic Crowding and Restricted Tournament Selection for Bimodal Optimisation,” in *Proc. of GECCO ’18*, ACM, 2018, pp. 929–936.

[4] K. A. De Jong, “An Analysis of the Behavior of a Class of Genetic Adaptive Systems,” PhD thesis, 1975.

[5] B. Doerr, “Probabilistic Tools for the Analysis of Randomized Optimization Heuristics,” *ArXiv e-prints*, 2018.

[6] B. Doerr and L. A. Goldberg, “Drift Analysis with Tail Bounds,” in *Proc. of PPSN XI*, Springer, 2010, pp. 174–183.

[7] S. Fischer, “A polynomial upper bound for a mutation-based algorithm on the two-dimensional Ising model,” in *Proc. of GECCO ’04*, Springer, 2004, pp. 1100–1112.

[8] S. Fischer and I. Wegener, “The One-dimensional Ising Model: Mutation versus Recombination,” *Theoretical Computer Science*, vol. 344, no. 2–3, pp. 208–225, 2005.

[9] T. Friedrich, P. S. Oliveto, D. Sudholt, and C. Witt, “Analysis of Diversity-preserving Mechanisms for Global Exploration,” *Evolutionary Computation*, vol. 17, no. 4, pp. 455–476, 2009.

[10] S. F. Galán and O. J. Mengshoel, “Generalized Crowding for Genetic Algorithms,” in *Proc. of GECCO ’10*, ACM, 2010, pp. 775–782.

[11] N. N. Glibovets and N. M. Gulayeva, “A Review of Niching Genetic Algorithms for Multimodal Function Optimization,” *Cybernetics and Systems Analysis*, vol. 49, no. 6, pp. 815–820, 2013.

[12] E. Happ, D. Johannsen, C. Klein, and F. Neumann, “Rigorous Analyses of Fitness-proportional Selection for Optimizing Linear Functions,” in *Proc. of GECCO ’08*, ACM, 2008, pp. 953–960.

[13] G. R. Harik, “Finding Multimodal Solutions Using Restricted Tournament Selection,” in *Proc. of the 6th ICGA*, Morgan Kaufmann Publishers Inc., 1995, pp. 24–31.

[14] S. W. Mahfoud, “Niching methods for genetic algorithms,” PhD thesis, University of Illinois at Urbana-Champaign, 1995, p. 251.

[15] O. Mengsheol and D. Goldberg, “Probabilistic Crowding: Deterministic Crowding with Probabilistic Replacement,” in *Proc. of GECCO ’99*, 1999, pp. 409–416.

[16] F. Neumann, P. S. Oliveto, and C. Witt, “Theoretical Analysis of Fitness-proportional Selection: Landscapes and Efficiency,” in *Proc. of GECCO ’09*, ACM, 2009, pp. 835–842.

[17] P. S. Oliveto and C. Witt, “Erratum: Simplified Drift Analysis for Proving Lower Bounds in Evolutionary Computation,” *ArXiv e-prints*, 2012.

[18] P. S. Oliveto, D. Sudholt, and C. Zarges, “On the Runtime Analysis of Fitness Sharing Mechanisms,” in *Proc. of PPSN XIII*, Springer, 2014, pp. 932–941.

[19] P. S. Oliveto and C. Witt, “Simplified Drift Analysis for Proving Lower Bounds in Evolutionary Computation,” *Algorithmica*, vol. 59, no. 3, pp. 369–386, 2011.

[20] ——, “On the runtime analysis of the simple genetic algorithm,” *Theoretical Computer Science*, vol. 545, pp. 2–19, 2014.

[21] ——, “Improved time complexity analysis of the Simple Genetic Algorithm,” *Theoretical Computer Science*, vol. 605, pp. 21–41, 2015.

[22] T. Paixão, J. Pérez Heredia, D. Sudholt, and B. Trubenová, “Towards a Runtime Comparison of Natural and Artificial Evolution,” *Algorithmica*, vol. 78, no. 2, pp. 681–713, 2017.

[23] J. E. Rowe and D. Sudholt, “The choice of the offspring population size in the (1,λ) evolutionary algorithm,” *Theoretical Computer Science*, vol. 545, pp. 20–38, 2014.

[24] B. Sareni and L. Krahenbuhl, “Fitness sharing and niching methods revisited,” *IEEE Transactions on Evolutionary Computation*, vol. 2, no. 3, pp. 97–106, 1998.

[25] O. M. Shir, “Niching in Evolutionary Algorithms,” in *Handbook of Natural Computing*, Springer, 2012, pp. 1035–1069.

[26] G. Singh and K. Deb, “Comparison of Multi-modal Optimization Algorithms Based on Evolutionary Algorithms,” in *Proc. of GECCO ’06*, ACM, 2006, pp. 1305–1312.

[27] G. Squillero and A. Tonda, “Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization,” *Information Sciences*, vol. 329, pp. 782–799, 2016.

[28] D. Sudholt, “The Benefits of Population Diversity in Evolutionary Algorithms: A Survey of Rigorous Runtime Analyses,” *ArXiv e-prints*, 2018.

[29] D. Sudholt, “Crossover is Provably Essential for the Ising Model on Trees,” in *Proc. of GECCO ’05*, ACM Press, 2005, pp. 1161–1167.

[30] M. Črepinšek, S.-H. Liu, and M. Mernik, “Exploration and Exploitation in Evolutionary Algorithms: A Survey,” *ACM Comput. Surv.*, vol. 45, no. 3, 35:1–35:33, 2013.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TEVC.2019.2914606, IEEE Transactions on Evolutionary Computation

Edgar Covantes Osuna received the B.Eng. degree in Informatics from the Universidad Politécnica de Sinaloa, Mazatlán, Sinaloa, México, in 2009, the M.Sc in Information Science with honours from the Universidad Autónoma de Sinaloa, Culiacán, Sinaloa, México, in 2013 and the Ph.D. in Computer Science from the University of Sheffield in Sheffield, United Kingdom, in 2019. He has worked in the areas of technical support, web development and embedded systems. His main interest is in the runtime analysis of evolutionary algorithms and diversity-preserving mechanisms. He is interested in randomised algorithms, algorithmic analysis, multimodal, multiobjective and combinatorial optimisation, and decision support. His main expertise is the analysis of randomised search heuristics such as evolutionary algorithms and its use as a tool for multi-criteria decision aid and software development/programming.

Dirk Sudholt received his Master's and Ph.D. degrees in computer science, under the supervision of Prof. I. Wegener, from the Technische Universität Dortmund, Dortmund, Germany, in 2004 and 2008, respectively. He is a Senior Lecturer and Head of the Algorithms Group at the University of Sheffield, Sheffield, U.K. He has held Post-Doctoral positions at the International Computer Science Institute, Berkeley, CA, USA, and the University of Birmingham, Birmingham, U.K. The present paper marks his 100th refereed publication. His current research interests include runtime analysis of randomized search heuristics, such as evolutionary algorithms and swarm intelligence. Dr. Sudholt was a recipient of the EU's Future and Emerging Technologies Scheme (SAGE Project) and eight best paper awards at GECCO and PPSN. He is an Editorial Board Member of Evolutionary Computation and Natural Computing.

1089-778X (c) 2018 IEEE. Translations and content mining are permitted for academic research only. Personal use is also permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.