



This is a repository copy of *Emergence of Diversity and its Benefits for Crossover in Genetic Algorithms*.

White Rose Research Online URL for this paper:
<http://eprints.whiterose.ac.uk/101311/>

Version: Accepted Version

Proceedings Paper:

Dang, D., Friedrich, T., Kötzing, T. et al. (5 more authors) (2016) Emergence of Diversity and its Benefits for Crossover in Genetic Algorithms. In: Parallel Problem Solving from Nature – PPSN XIV. Parallel Problem Solving from Nature (PPSN 2016), 17th - 21st Sep, 2016, Edinburgh, UK. Lecture Notes in Computer Science, 9921 . Springer International Publishing .

https://doi.org/10.1007/978-3-319-45823-6_83

Reuse

Unless indicated otherwise, fulltext items are protected by copyright with all rights reserved. The copyright exception in section 29 of the Copyright, Designs and Patents Act 1988 allows the making of a single copy solely for the purpose of non-commercial research or private study within the limits of fair dealing. The publisher or other rights-holder may allow further reproduction and re-use of this version - refer to the White Rose Research Online record for this item. Where records identify the publisher as the copyright holder, users can verify any specific terms of use on the publisher's website.

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/>

Emergence of Diversity and its Benefits for Crossover in Genetic Algorithms

Duc-Cuong Dang¹, Tobias Friedrich², Timo Kötzing², Martin S. Krejca²,
Per Kristian Lehre¹, Pietro S. Oliveto³, Dirk Sudholt³, and Andrew M. Sutton²

¹ University of Nottingham, United Kingdom

² Hasso Plattner Institute, Potsdam, Germany

³ University of Sheffield, United Kingdom

Abstract. Population diversity is essential for avoiding premature convergence in Genetic Algorithms (GAs) and for the effective use of crossover. Yet the dynamics of how diversity emerges in populations are not well understood. We use rigorous runtime analysis to gain insight into population dynamics and GA performance for a standard $(\mu+1)$ GA and the Jump_k test function. By studying the stochastic process underlying the size of the largest collection of identical genotypes we show that the interplay of crossover followed by mutation may serve as a catalyst leading to a sudden burst of diversity. This leads to improvements of the expected optimisation time of order $\Omega(n/\log n)$ compared to mutation-only algorithms like the $(1+1)$ EA.

Keywords Genetic algorithms, crossover, diversity, runtime analysis, theory

1 Introduction

Genetic Algorithms (GAs) are powerful general-purpose optimisers that perform surprisingly well in many applications. Their wide-spread success is based on a number of factors: using populations to diversify search, using mutation to generate novel solutions, and using crossover to combine features of good solutions. Crossover can combine building blocks of good solutions, and help to focus search on bits where parents disagree. For both tasks the population needs to be diverse enough for crossover to be effective. A common problem in the application of GAs is the loss of diversity when the population converges to copies of the same search point, often called “premature convergence”.

Understanding population diversity and crossover has proved elusive. The first example function where crossover was proven to be beneficial is called Jump_k . In this problem, GAs have to overcome a fitness valley such that all local optima have Hamming distance k to the global optimum. Jansen and Wegener [7] showed that, while mutation-only algorithms such as the $(1+1)$ EA require expected time $\Theta(n^k)$, a simple $(\mu+1)$ GA with crossover only needs time $O(\mu n^2 k^3 + 4^k/p_c)$ where p_c is the crossover probability. This time is $O(4^k/p_c)$ for large k , and hence significantly faster than mutation-only GAs. However,

their analysis requires an unrealistically small crossover probability $p_c \leq 1/(ckn)$ for a large constant $c > 0$. Hence the analysis does not reflect the typical behaviour in GA populations with constant crossover probabilities $p_c = \Theta(1)$ as used in practice. Kötzing, Sudholt, and Theile [8] later refined these results towards a crossover probability $p_c \leq k/n$, which is still unrealistically small. Both approaches focus on creating diversity through a sequence of lucky mutations, relying on crossover to create the optimum once sufficient diversity has been created. Their arguments break down if crossover is applied frequently.

Here we provide a novel approach to show that diversity can also be created by frequent applications of crossover followed by mutation. For the maximum crossover probability $p_c = 1$ we prove that on Jump_k diversity emerges naturally in a population: the interplay of crossover, followed by mutation, can serve as a catalyst for creating a diverse range of search points out of few different individuals. This allows to prove a speedup of order $n/\log n$ for $k \geq 3$ compared to mutation-only algorithms such as the (1+1) EA. Both operators are vital: mutation alone requires $\Theta(n^k)$ expected iterations to hit the optimum from a local optimum. As shown in [8, Theorem 8] using only crossover with $p_c = \Omega(1)$ but no mutation, diversity reduces quickly, leading to inefficient runtimes for small population sizes ($\mu = O(\log n)$).

After defining the algorithm, the $(\mu+1)$ GA from [7], and the Jump_k function in Section 2, we elaborate on the population dynamics in Section 3, preparing the ground for the following runtime result. For the standard mutation rate, we show in Section 4 that the $(\mu+1)$ GA with $p_c = 1$, $\mu = O(n)$, $k = O(1)$ optimises Jump_k in expected time $O(\mu n \log(\mu) + n^k/\mu + n^{k-1} \log(\mu))$. Compared to the expected time $\Theta(n^k)$ for the (1+1) EA this corresponds to a speedup of order $n/\log n$ for $k \geq 3$ and $\sqrt{n/\log n}$ for $k = 2$, for the best possible choice of μ .

2 Preliminaries

The $\text{Jump}_k: \{0, 1\}^n \rightarrow \mathbb{N}$ class of pseudo-Boolean fitness functions was originally introduced by Jansen and Wegener [7]. The function value increases with the number of 1-bits in the bitstring until a *plateau* of local optima is reached consisting of all points with $n - k$ 1-bits. However, its only global optimum is the all-ones string 1^n . Between the plateau and the global optimum there is a gap of Hamming distance k which has to be “jumped over” for the function to be optimised. The function is formally defined as

$$\text{Jump}_k(x) = \begin{cases} k + |x|_1 & \text{if } |x|_1 = n \text{ or } |x|_1 \leq n - k, \\ n - |x|_1 & \text{otherwise,} \end{cases}$$

where $|x|_1 = \sum_{i=1}^n x_i$ is the number of 1-bits in x .

We will analyse the performance of a standard steady-state $(\mu+1)$ GA [7] with $p_c = 1$ using at each step uniform crossover (i.e., each bit of the offspring is chosen uniformly at random from one of the parents) and standard bit mutation (i.e., each bit is flipped with probability $p_m = \chi/n = \Theta(1/n)$). Algorithm 1 shows the pseudo code for the $(\mu+1)$ GA, tailored to $p_c = 1$.

Algorithm 1: $(\mu+1)$ GA with $p_c = 1$

```
1  $P \leftarrow \mu$  individuals, uniformly at random from  $\{0, 1\}^n$ ;  
2 while  $1^n \notin P$  do  
3   Choose  $x, y \in P$  uniformly at random;  
4    $z \leftarrow \text{mutate}(\text{crossover}(x, y))$ ;  
5    $P \leftarrow P \cup \{z\}$ ;  
6   Remove one element from  $P$  with lowest fitness, breaking ties u.a.r.;
```

3 Population Dynamics

The following lemma gives a bound on the expected time for the whole population to reach the plateau. It is proved using level-based arguments, similar to those in [3]. The proof is omitted due to space restrictions.

Lemma 1. *The expected time until the entire population of $(\mu+1)$ GA with $p_m = \Theta(1/n)$ reaches the plateau of Jump_k for $k = O(1)$ (or the optimum has been found) is $O(n\mu \log \mu + n \log n)$.*

In the remainder of the analysis we study the algorithm's behaviour once all individuals are on the plateau. Previous observations of simulations have revealed the following behaviour. Assume the algorithm has reached a population where all individuals are identical. We refer to identical individuals as a *species*, hence in this case there is only one species. Eventually, a mutation will create a different search point on the plateau, leading to the creation of a new species. Both species may shrink or grow in size, and there is a chance that the new species disappears and we go back to one species only.

However the existence of two species also serves as a catalyst for creating further species in the following sense. Say two parents 0001111111 and 0010111111 are recombined, then crossover has a good chance of creating an individual with $n - k + 1$ 1s, e.g. 0011111111. Then mutation has a constant probability of flipping any of the $n - k - 1$ unrelated 1-bits to 0, leading to a new species, e.g. 0011111011. This may lead to a sudden burst of diversity in the population.

Due to the ability to create new species, the size of the largest cluster performs an almost fair random walk. Once its size has decreased significantly from its maximum μ , there is a good chance for recombining two parents from different species. This helps in finding the global optimum as crossover can increase the number of 1s in the offspring, compared to its parents, such that fewer bits need to be flipped by mutation to reach the optimum. This is formalised in the following lemma.

Lemma 2. *The probability that the global optimum is constructed by a uniform crossover of two parents on the plateau with Hamming distance $2d$, followed by mutation ($p_m = \chi/n$), is*

$$\sum_{i=0}^{2d} \binom{2d}{i} \frac{1}{2^{2d}} \left(\frac{\chi}{n}\right)^{k+d-i} \left(1 - \frac{\chi}{n}\right)^{n-k-d+i} \geq \frac{1}{2^{2d}} \left(\frac{\chi}{n}\right)^{k-d} \left(1 - \frac{\chi}{n}\right)^{n-k+d} \quad (1)$$

Proof. For a pair of search points on the plateau with Hamming distance $2d$, both parents have d ones among the $2d$ bits that differ between parents, and $n - k - d$ 1s outside this area. Assume that crossover sets i out of these $2d$ bits to 1, which happens with probability $\binom{2d}{i} \cdot 2^{-2d}$. Then mutation needs to flip the remaining $k + d - i$ 0s to 1. The probability that such a pair creates the optimum is hence

$$\sum_{i=0}^{2d} \binom{2d}{i} \frac{1}{2^{2d}} \left(\frac{\chi}{n}\right)^{k+d-i} \left(1 - \frac{\chi}{n}\right)^{n-k-d+i}.$$

The second bound is obtained by ignoring summands $i < 2d$ for the sum. \square

Note that even a Hamming distance of 2, i.e. $d = 1$, leads to a probability of $\Omega(n^{-k+1})$, provided that such parents are selected for reproduction. The probability is by a factor of n larger than the probability $\Theta(n^{-k})$ of mutation without crossover reaching the optimum from the plateau. We will show that this effect leads to a speedup of nearly n for the $(\mu+1)$ GA, compared to the expected time of $\Theta(n^k)$ for the $(1+1)$ EA [5] and other EAs only using mutation.

The idea behind the analysis is to investigate the random walk underlying the size of the largest species. We bound the expected time for this size to decrease to $\mu/2$, and then argue that the $(\mu+1)$ GA is likely to spend a good amount of time with a population of good diversity, where the probability of creating the optimum in every generation is $\Omega(n^{-k+1})$ due to the chance of recombining parents of Hamming distance at least 2.

In the following we refer to $Y(t)$ as the size of the largest species in the population at time t . Define

$$\begin{aligned} p_+(y) &:= \Pr(Y(t+1) - Y(t) = 1 \mid Y(t) = y), \\ p_-(y) &:= \Pr(Y(t+1) - Y(t) = -1 \mid Y(t) = y), \end{aligned}$$

i.e., $p_+(y)$ is the probability that the size of the largest species increases from y to $y + 1$, and $p_-(y)$ is the probability that it decreases from y to $y - 1$.

The following lemma gives bounds on these transition probabilities, unless two parents of Hamming distance larger than 2 are selected for recombination (this case will be treated later in Lemma 4). We formulate the lemma for arbitrary mutation rates $\chi/n = \Theta(1/n)$ and restrict our attention to sizes $Y(t) \geq \mu/2$ as we are only interested in the expected time for the size to decrease to $\mu/2$.

Lemma 3. *For every population on the plateau of Jump_k for $k = O(1)$ the following holds. Either the $(\mu+1)$ GA with mutation rate $\chi/n = \Theta(1/n)$ performs a crossover of two parents whose Hamming distance is larger than 2, or the size $Y(t)$ of the largest species changes according to transition probabilities $p_-(\mu) = \Omega(1/n)$ and, for $\mu/2 \leq y < \mu$,*

$$\begin{aligned} p_+(y) &\leq \frac{y(\mu - y)(\mu + y)}{2\mu^2(\mu + 1)} \left(1 - \frac{\chi}{n}\right)^n + \mathcal{O}\left(\frac{(\mu - y)^2}{\mu^2 n}\right), \\ p_-(y) &\geq \frac{y(\mu - y)(\mu + \chi y)}{2\mu^2(\mu + 1)} \left(1 - \frac{\chi}{n}\right)^n - \mathcal{O}\left(\frac{\mu - y}{\mu n}\right). \end{aligned}$$

Proof. We call an individual belonging to the current largest species a y -individual and all the others non- y individuals. In each generation, there is either no change, or one individual is added to the population and one individual chosen uniformly at random is removed from the population. In order to increase the number of y -individuals, it is necessary that a y -individual is added to the population, and a non- y individual is removed from the population. Analogously, in order to decrease the number of y -individuals, it is necessary that a non- y individual is added to the population, and a y -individual is removed from the population.

Given that $Y(t) = y$, let $p(y)$ be the probability that a y -individual is created at time $t + 1$, and $q(y)$ the probability that a non- y individual is created.

We now estimate an upper bound on $p(y)$. We may assume that the Hamming distance between parents is at most 2 as otherwise there is nothing to prove. A y -individual can be created in the following three ways:

- Two y -individuals are selected. Crossing over two y -individuals produces another y -individual, which survives mutation if no bits are flipped, i.e., with probability $(1 - \chi/n)^n$.
- One y -individual and one non- y individual are selected. The crossover operator produces a y -individual with probability $1/4$ and mutation does not flip any bits with probability $(1 - \chi/n)^n$. If the crossover operator does not produce a y -individual, then to produce a y -individual at least one specific bit-position must be mutated, which occurs with probability $O(1/n)$. The overall probability is hence $(1/4)(1 - \chi/n)^n + O(1/n)$.
- Two non- y individuals are selected. These two individuals are either identical or have Hamming distance 2 (i.e., by assumption). In the first case they both have one of the k 0-bit positions of a y -individual set to 1. In the second case they either both have one of the k 0-bit positions of a y -individual set to 1 or they both have one of the $n - k$ 1-bit positions set to 0. In both cases, crossover cannot change the value of such bit. Thus, at least one specific bit-position must be flipped, which occurs with probability $O(1/n)$.

Taking into account the probabilities of the three selection events above, the probability of producing a y -individual is

$$\begin{aligned}
p(y) &= \left(\frac{y}{\mu}\right)^2 \left(1 - \frac{\chi}{n}\right)^n + 2 \left(\frac{y}{\mu}\right) \left(1 - \frac{y}{\mu}\right) \left[\left(\frac{1}{4}\right) \left(1 - \frac{\chi}{n}\right)^n + O\left(\frac{1}{n}\right) \right] \\
&\quad + \frac{(\mu - y)^2}{\mu^2} O\left(\frac{1}{n}\right) \\
&= \left(1 - \frac{\chi}{n}\right)^n \left(\frac{y}{\mu}\right) \left(\frac{y}{\mu} + \frac{\mu - y}{2\mu}\right) + O\left(\frac{y(\mu - y)}{\mu^2} \cdot \frac{1}{n}\right) + O\left(\frac{(\mu - y)^2}{\mu^2} \cdot \frac{1}{n}\right) \\
&= \frac{y(\mu + y)}{2\mu^2} \left(1 - \frac{\chi}{n}\right)^n + O\left(\frac{\mu(\mu - y)}{\mu^2} \cdot \frac{1}{n}\right)
\end{aligned}$$

We then estimate a lower bound on $q(y)$. In the case where $y = \mu$, a non- y individual can be added to the population if:

- two y -individuals are selected, and the mutation operator flips one of the k 0-bits and one of the $n - k$ 1-bits. This event occurs with probability $q(\mu) = k(n - k) \left(\frac{\chi}{n}\right)^2 \left(1 - \frac{\chi}{n}\right)^{n-2} = \Omega(1/n)$, where we used that $k = O(1)$. In the other case where $y < \mu$, then a non- y individual can be added to the population in the following two ways:
 - A y -individual and a non- y individual are selected. Crossover produces a copy of the non- y individual with probability $1/4$, which is unchanged by mutation with probability $(1 - \chi/n)^n$. Or with probability $1/4$, crossover produces an individual with $k - 1$ 0-bits. Mutation then creates a non- y -individual by flipping a single of the $n - k$ 1-bit positions. This event occurs with probability $(1/4)(n - k) \left(\frac{\chi}{n}\right) \left(1 - \frac{\chi}{n}\right)^{n-1} \geq (\chi/4) \left(1 - \frac{\chi}{n}\right)^n - O(1/n)$ using again that $k = O(1)$.
 - Two non- y -individuals are selected. In the worst case, the selected individuals are different, hence crossover produces an individual on the plateau with probability at least $1/2$, which mutation does not destroy with probability $(1 - \chi/n)^n$.

Assuming that $\mu/2 \leq y < \mu$ and n is sufficiently large, the probability of adding a non- y individual is

$$\begin{aligned} q(y) &\geq 2 \left(\frac{y}{\mu}\right) \left(1 - \frac{y}{\mu}\right) \left[\left(\frac{\chi+1}{4}\right) \left(1 - \frac{\chi}{n}\right)^n - O\left(\frac{1}{n}\right) \right] + \frac{1}{2} \left(1 - \frac{y}{\mu}\right)^2 \left(1 - \frac{\chi}{n}\right)^n \\ &= \frac{(\mu - y)(\mu + \chi y)}{2\mu^2} \left(1 - \frac{\chi}{n}\right)^n - O\left(\frac{\mu - y}{\mu} \cdot \frac{1}{n}\right). \end{aligned}$$

Multiplying $p(y)$ and $q(y)$ by the respective survival probabilities, we get

$$\begin{aligned} p_-(y) &\geq \left[\frac{(\mu - y)(\mu + \chi y)}{2\mu^2} \left(1 - \frac{\chi}{n}\right)^n - O\left(\frac{\mu - y}{\mu} \cdot \frac{1}{n}\right) \right] \left(\frac{y}{\mu + 1}\right) \\ &= \frac{(\mu - y)(\mu + \chi y)y}{2\mu^2(\mu + 1)} \left(1 - \frac{\chi}{n}\right)^n - O\left(\frac{(\mu - y)}{\mu} \cdot \frac{1}{n}\right). \end{aligned}$$

$$\begin{aligned} p_+(y) &= \left[\frac{y(\mu + y)}{2\mu^2} \left(1 - \frac{\chi}{n}\right)^n + O\left(\frac{y(\mu - y)}{\mu^2} \cdot \frac{1}{n}\right) \right] \left(\frac{\mu - y}{\mu + 1}\right) \\ &= \frac{(\mu^2 - y^2)y}{2\mu^2(\mu + 1)} \left(1 - \frac{\chi}{n}\right)^n + O\left(\frac{(\mu - y)^2}{\mu^2} \cdot \frac{1}{n}\right). \end{aligned}$$

Both equalities hold for values of y between $\mu/2$ and μ . □

Steps where crossover recombines two parents with larger Hamming distance were excluded from Lemma 3 as they require different arguments. The following lemma shows that conditional transition probabilities in this case are favourable in that the size of the largest species is more likely to decrease than to increase.

Lemma 4. *Assume that $y \geq \mu/2$ and the $(\mu+1)$ GA on Jump_k with $k = O(1)$ and mutation rate $\chi/n = \Theta(1/n)$ selects two individuals on the plateau with*

Hamming distance larger than 2, then for conditional transition probabilities $p_-^*(y)$ and $p_+^*(y)$ for decreasing or increasing the size of the largest species, $p_-^*(y) \geq 2p_+^*(y)$.

Proof. Assume that the population contains two individuals x and z with Hamming distance $2\ell \leq 2k$, where $\ell \geq 2$. Without loss of generality, let us assume that they differ in the first 2ℓ bit positions.

In the case that the majority individual y has ℓ 0-bits in the first 2ℓ positions, then a y -individual may be produced by creating the ℓ 0-bits and ℓ 1-bits in the exact positions by crossover and no mutation should occur. Alternatively, at least one exact bit has to be flipped by mutation. Then, the probability of producing a y -individual from x and z , and replacing a non y -individual with y is less than

$$p_+^*(y) \leq \left[\left(\frac{1}{2}\right)^{2\ell} \left(1 - \frac{\chi}{n}\right)^n + O\left(\frac{1}{n}\right) \right] \left(\frac{\mu - y}{\mu + 1}\right)$$

On the other hand, the probability of producing an individual on the plateau different from y , and replacing a y -individual is at least (for sufficiently large n)

$$p_-^*(y) \geq \left(\binom{2\ell}{\ell} - 1 \right) \left(\frac{1}{2}\right)^{2\ell} \left(1 - \frac{\chi}{n}\right)^n \left(\frac{y}{\mu + 1}\right) > 2p_+^*(y).$$

In the other case, assume that the majority individual y does not have ℓ 0-bits in the first 2ℓ bit-positions. Then the mutation operator must flip at least one specific bit among the last $n - 2\ell$ positions to produce y , which occurs with probability $O(1/n)$, while the probability to produce a non y -individual on the plateau is still $\Omega(1)$. \square

4 Standard Mutation Rate

In this section we state the main result. Herein we consider $p_m = 1/n$.

Theorem 1. *The expected optimisation time of the $(\mu+1)$ GA with $p_c = 1$, $p_m = 1/n$ and $\mu \leq \kappa n$, for some constant $\kappa > 0$, on Jump_k , $k = O(1)$, is $O(\mu n \log(\mu) + n^k/\mu + n^{k-1} \log(\mu))$.*

For $k \geq 3$ the best speedup compared to the expected time of $\Theta(n^k)$ for the $(1+1)$ EA [5] and other EAs only using mutation is of order $\Omega(n/\log n)$ for $\mu = \kappa n$. For $k = 2$ the best speedup is of order $\Omega(\sqrt{n/\log n})$ for $\mu = \Theta(\sqrt{n/\log n})$.

Note that for mutation rate $1/n$, the dominant terms in Lemma 3 are equal, hence the size of the largest species performs a fair random walk, up to a bias resulting from small-order terms. This confirms our intuition from observing simulations. The following lemma formalises this fact: in steps where the size $Y(t)$ of the largest species changes, it performs an almost fair random walk.

Lemma 5. *For the random walk induced by the size of the largest species, conditional on the current size y changing, for $\mu/2 < y < \mu$, the probability of increasing y is at most $1/2 + O(1/n)$ and the probability of decreasing it is at least $1/2 - O(1/n)$.*

We use these transition probabilities to bound the expected time for the random walk to hit $\mu/2$.

Lemma 6. *Consider the random walk of $Y(t)$, starting in state $X_0 \geq \mu/2$. Let T be the first hitting time of state $\mu/2$. If $\mu = O(n)$, then $E(T \mid X_0) = O(\mu n + \mu^2 \log \mu)$ regardless of X_0 .*

Proof. Let E_i abbreviate $E(T \mid X_0 = i)$, then $E_{\mu/2} = 0$ and $E_\mu = O(n) + E_{\mu-1}$ as $p_-(\mu) = \Omega(1/n)$ by Lemma 3.

For $\mu/2 < y < \mu$ the probability of leaving state y is always (regardless of Hamming distances between species) bounded from below by the probability of selecting two y -individuals as parents, not flipping any bits during mutation, and choosing a non- y individual for replacement (cf. Lemma 3, Lemma 4):

$$p_+(y) + p_-(y) \geq \frac{y^2}{\mu^2} \cdot \left(1 - \frac{1}{n}\right)^n \cdot \frac{\mu - y}{\mu + 1} \geq \frac{\mu - y}{24\mu}$$

as $y \geq \mu/2$, $\mu + 1 \leq 3\mu/2$ (since $\mu \geq 2$), and $(1 - 1/n)^n \geq 1/4$ for $n \geq 2$. Using conditional transition probabilities $1/2 \pm \delta$ for $\delta = O(1/n)$ according to Lemma 5, E_i is bounded as $E_i \leq \frac{24\mu}{\mu-i} + (\frac{1}{2} - \delta) E_{i-1} + (\frac{1}{2} + \delta) E_{i+1}$.

This is equivalent to $(\frac{1}{2} - \delta) \cdot (E_i - E_{i-1}) \leq \frac{24\mu}{\mu-i} + (\frac{1}{2} + \delta) \cdot (E_{i+1} - E_i)$. Introducing $D_i := E_i - E_{i-1}$, this is equivalent to

$$D_i \leq \frac{\frac{24\mu}{\mu-i} + (\frac{1}{2} + \delta) \cdot D_{i+1}}{\frac{1}{2} - \delta} \leq \frac{50\mu}{\mu-i} + \alpha \cdot D_{i+1}$$

for $\alpha := \frac{1+2\delta}{1-2\delta} = 1 + O(1/n)$, assuming n is large enough. From $E_\mu = O(n) + E_{\mu-1}$ we get $D_\mu = O(n)$, hence an induction yields $D_i \leq \sum_{j=i}^{\mu-1} \frac{50\mu}{\mu-j} \cdot \alpha^{j-i} + \alpha^{\mu-i} \cdot O(n)$.

Combining $\alpha = 1 + O(1/n)$ and $1 + x \leq e^x$ for all $x \in \mathbb{R}$, we have $\alpha^\mu \leq e^{O(\mu/n)} \leq e^{O(1)} = O(1)$. Bounding both α^{j-i} and $\alpha^{\mu-i}$ in this way, we get

$$D_i \leq O(n) + O(\mu) \cdot \sum_{j=i}^{\mu-1} \frac{1}{\mu-j} = O(n + \mu \log \mu)$$

as the sum is equal to $\sum_{j=1}^{\mu-i} 1/j = O(\log \mu)$.

Now, $D_{\mu/2+1} + D_{\mu/2+2} + \dots + D_i = (E_{\mu/2+1} - E_{\mu/2}) + (E_{\mu/2+2} - E_{\mu/2+1}) + \dots + (E_i - E_{i-1}) = E_i - E_{\mu/2} = E_i$. Hence we get $E_i = \sum_{k=(\mu/2)+1}^i D_k \leq O(\mu n + \mu^2 \log \mu)$. \square

Now we show that, when the largest species has decreased its size to $\mu/2$, there is a good chance that the optimum will be found within the following $\Theta(\mu^2)$ generations.

Lemma 7. *Consider the $(\mu+1)$ GA with $p_c = 1$ on Jump_k . If the largest species has size at most $\mu/2$ and $\mu \leq \kappa n$ for a sufficiently small constant $\kappa > 0$, the probability that during the next $c\mu^2$ generations, for some constant $c > 0$, the global optimum is found is $\Omega(1/(1 + n^{k-1}/\mu^2))$.*

Proof. We show that during the $c\mu^2$ generations the size of the largest species never rises above $(3/4)\mu$ with at least constant probability. Then we calculate the probability of jumping to the optimum during the phase, given this happens.

Let X_i , $1 \leq i \leq c\mu^2$ be random variables indicating the increase in number of individuals of the largest species at generation i . We pessimistically ignore self-loops, thus the size of the species either increases or decreases in each generation. Using the conditional probabilities from Lemma 5, we get that the expected increase in each step is $1 \cdot (1/2 + O(1/n)) - 1 \cdot (1/2 - O(1/n)) = O(1/n)$. Then the expected increase in size of the largest species at the end of the phase is

$$\mathbb{E}(X) = \sum_{i=1}^{c\mu^2} X_i = \sum_{i=1}^{c\mu^2} O(1/n) = (c'\mu^2)/n \leq c'\kappa\mu \leq (1/8)\mu,$$

where we use that $\mu \leq \kappa n$ and κ is chosen small enough.

By an application of Hoeffding bounds $\Pr(X \geq \mathbb{E}(X) + \lambda) \leq \exp(-2\lambda^2 / \sum_i c_i^2)$ with $\lambda = \mu/8$ and $c_i = 2$, we get that $\Pr(X \geq (2/8)\mu) \leq \exp(-c') = 1 - \Omega(1)$. We remark that the bounds also hold for any partial sum of the sequence X_i ([1], Chapter 1, Theorem 1.13), i. e. with probability $\Omega(1)$ the size *never* exceeds $(3/4)\mu$ in the considered phase of length $c\mu^2$ generations.

While the size does not exceed $(3/4)\mu$, in every step, there is a probability of at least $1/4 \cdot 3/4 = \Omega(1)$ of selecting parents from two different species, and by Lemma 2 the probability of creating the optimum is $\Omega(n^{-k+1})$.

Finally, the probability that at least one successful generation occurs in a phase of $c\mu^2$ is, using $(1 - (1 - p))^\lambda \geq (\lambda p / (1 + \lambda p))$ for $\lambda \in \mathbb{N}, p \in [0, 1]$ [2, Lemma 10], the probability that the optimum is found in one of these steps is

$$1 - \left(1 - \frac{1}{\Omega(n^{-k+1})}\right)^{c\mu^2} \geq \Omega\left(\frac{\mu^2 \cdot n^{-k+1}}{1 + \mu^2 \cdot n^{-k+1}}\right).$$

Finally, we assemble all lemmas to prove our main result.

Proof (of Theorem 1). The expected time for the whole population to reach the plateau is $O(\mu n \log(\mu) + n \log n)$ by Lemma 1. Once the population is on the plateau, we wait till the largest species has decreased its size to at most $\mu/2$. According to Lemma 6, the time for the largest species to reach size $\mu/2$ is $O(\mu n + \mu^2 \log \mu)$. By Lemma 7, the probability that in the next $c\mu^2$ steps the optimum is found is $\Omega(1/(1 + n^{k-1}/\mu^2))$. If not, we repeat the argument. The expected number of such trials is $O(1 + n^{k-1}/\mu^2)$ and the expected length of one trial is $O(\mu n + \mu^2 \log \mu) + c\mu^2 = O(\mu n + \mu^2 \log \mu)$. The expected time for reaching the optimum from the plateau is hence at most $O(\mu n + \mu^2 \log(\mu) + n^k/\mu + n^{k-1} \log(\mu))$.

Adding up all times and subsuming terms $O(\mu^2 \log(\mu)) = O(\mu n \log \mu)$ and $O(n \log n) = O(n^k/\mu + n^{k-1} \log \mu)$ completes the proof. \square

5 Conclusion

A rigorous analysis of the $(\mu+1)$ GA has been presented showing how the use of both crossover and mutation considerably speeds up the runtime for Jump_k

compared to algorithms using mutation only. Traditionally it has been believed that crossover may be useful only if sufficient diversity is readily available and that the emergence of diversity in the population is due to either mutation alone or should be enforced by the introduction of diversity mechanisms [4, 6, 9]. Indeed, previous work highlighting that crossover may be beneficial for Jump_k used unrealistically low crossover probabilities to allow mutation alone to create sufficient diversity. Conversely, our analysis shows that the interplay between crossover and mutation on the plateau of local optima of the Jump_k function quickly leads to a burst of diversity that is then exploited by both operators to reach the global optimum.

Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 618091 (SAGE) and from the EPSRC under grant no. EP/M004252/1 and is based upon work from COST Action CA15140 ‘Improving Applicability of Nature-Inspired Optimisation by Joining Theory and Practice (ImAppNIO)’.

Bibliography

- [1] A. Auger and B. Doerr. *Theory of Randomized Search Heuristics*. World Scientific, 2011.
- [2] G. Badkobeh, P. K. Lehre, and D. Sudholt. Black-box complexity of parallel search with distributed populations. In *Proc. of FOGA '15*, pp. 3–15, 2015.
- [3] D. Corus, D.-C. Dang, A. V. Eremeev, and P. K. Lehre. Level-based analysis of genetic algorithms and other search processes. In *PPSN XIII*, 2014.
- [4] D.-D. Dang, T. Friedrich, T. Kötzing, M. S. Krejca, P. K. Lehre, P. S. Oliveto, D. Sudholt, and A. M. Sutton. Escaping local optima with diversity mechanisms and crossover. In *GECCO '16*, pp. –, 2016. (To appear).
- [5] S. Droste, T. Jansen, and I. Wegener. On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science*, 276:51–81, 2002.
- [6] T. Friedrich, P. S. Oliveto, D. Sudholt, and C. Witt. Analysis of diversity-preserving mechanisms for global exploration. *Evolutionary Computation*, 17:455–476, 2009.
- [7] T. Jansen and I. Wegener. The analysis of evolutionary algorithms - a proof that crossover really can help. *Algorithmica*, 34:47–66, 2002.
- [8] T. Kötzing, D. Sudholt, and M. Theile. How crossover helps in pseudo-boolean optimization. In *Proc. of GECCO '11*, pp. 989–996, 2011.
- [9] P. S. Oliveto and C. Zarges. Analysis of diversity mechanisms for optimisation in dynamic environments with low frequencies of change. *Theoretical Computer Science*, 561:37–56, 2015.