POLITECNICO DI TORINO
Repository ISTITUZIONALE

Theory and Practice of Population Diversity in Evolutionary Computation

Original
Theory and Practice of Population Diversity in Evolutionary Computation / Sudholt, D., Squillero, G.. - (2024), pp. 1391-1409. (2024 Genetic and Evolutionary Computation Conference Companion, GECCO 2024 Companion Melbourne (AUS) July 14 - 18, 2024) [10.1145/3638530.3648430].

Availability:
This version is available at: 11583/3014656 since: 2026-08-19T15:00:27Z

Publisher:
Association for Computing Machinery

Published
DOI:10.1145/3638530.3648430

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

24 August 2026

Theory and Practice of Population Diversity in Evolutionary Computation

Dirk Sudholt

University of Passau
Passau, Germany

dirk.sudholt@uni-passau.de

Giovanni Squillero

Politecnico di Torino
Torino, Italy

giovanni.squillero@polito.it

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike International 4.0 License.
GECCO '24 Companion, July 14–18, 2024, Melbourne, VIC, Australia
© 2024 Copyright is held by the owner/author(s).
ACM ISBN 979-8-4007-0495-6/24/07

https://doi.org/10.1145/3638530.3648430

Instructors

Dirk Sudholt is a full professor at the University of Passau, Germany. He previously held a post as Senior Lecturer at the University of Sheffield, UK. He obtained his PhD in computer science in 2008 from the Technische Universität Dortmund, Germany, under the supervision of Prof. Ingo Wegener. His research focuses on the computational complexity of randomized search heuristics such as evolutionary algorithms and swarm intelligence. Most relevant to this tutorial is his work on runtime analysis of diversity mechanisms and the benefits of crossover in genetic algorithms. He has more than 130 refereed publications and won 9 best paper awards at GECCO and PPSN.

Giovanni Squillero works as a full professor of computer science at Politecnico di Torino, Italy. His research mixes artificial intelligence and soft computing, with particular emphasis on bio-inspired meta-heuristics and multi-agent systems; in more down-to-earth activities, he develops approximate optimization techniques able to achieve acceptable solutions with reasonable amount of resources. The industrial applications of his work range from electronic CAD to bioinformatics. He is credited as an author in 3 books, 39 journal articles, 11 book chapters, and 165 papers in conference proceedings; he is also listed among the editors in 14 volumes.

Relevant Publications

Practical diversity-promoting mechanism are surveyed in a paper (Squillero and Tonda, 2016):

> *"Divergence of Character and Premature Convergence: A Survey of Methodologies for Promoting Diversity in Evolutionary Optimization". Information Sciences, Special issue on Discovery Science, 329.*

Theoretical parts are described the following book chapter (Sudholt, 2020):

> *The Benefits of Population Diversity in Evolutionary Algorithms: A Survey of Rigorous Runtime Analyses. In Benjamin Doerr and Frank Neumann (Eds.): Theory of Evolutionary Computation - Recent Developments in Discrete Optimization, Natural Computing Series, Springer.*

Recent theoretical and empirical results are featured in Edgar Covantes Osuna's thesis (Covantes Osuna, 2019):

> *E. Covantes Osuna. Theoretical and Empirical Evaluation of Diversity-preserving Mechanisms in Evolutionary Algorithms: On the Rigorous Runtime Analysis of Diversity-preserving Mechanisms in Evolutionary Algorithms. PhD thesis, University of Sheffield, 2019.*

What this tutorial is about

- A view on population diversity from theory and practice
- Surveying approaches to maintain and promote diversity
- An overview of diversity-related runtime analyses of EAs
- Insights into which diversity mechanisms work best for particular problems – and why!
- Trying to understand how diversity mechanisms work, how to use them and how to improve on them.

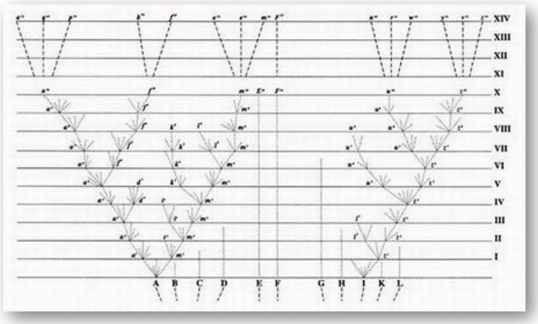
What this tutorial is not about

- Focus on static single-objective problems
- No in-depth coverage of island models/distributed EAs (see survey (Sudholt, 2015) and a tutorial at PPSN 2018

Agenda

1. Introduction
2. How Diversity Benefits Exploration – Case Study on TwoMax
3. How Diversity Benefits Crossover
4. Conclusions and Outlook

What is Diversity in EA?



The **only** illustration in Darwin's *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* (London: John Murray, 1859)

Diversity in Natural Evolution: Principle of Divergence

"The principle of divergence is simplicity itself: the more the coinhabitants of an area differ from each other in their ecological requirements, the less they will compete with each other; therefore natural selection will tend to favor any variation toward greater divergence" (Mayr, 1992)

"The principle, which I have designated by this term is of high importance, and explains, as I believe, several important facts" (Darwin, 1859)

- "[It] causes differences, at first barely appreciable, to steadily to increase, and the breeds to diverge in character, both from each other and from their common parent"
- "The varying descendants of each species try to occupy as many and as different places as possible in the economy of nature"

Diversity in Artificial Evolution: Premature Convergence

"Premature convergence is the well-known effect of losing population diversity too quickly, and getting trapped in a local optimum" (Eiben and Smith, 2015)

- That is: The tendency of an algorithm to converge towards a point where it was not supposed to converge to in the first place (probably an oxymoron)
- Premature convergence is the single most taxing problem of EC

Divergence of Character vs. Premature Convergence

- Holland was aware of this problem since his first works, see the discussion on the "lack of speciation" (Holland, 1992)
- Endemic problem: in EC a true environment is messing and there is no advantage to diversify

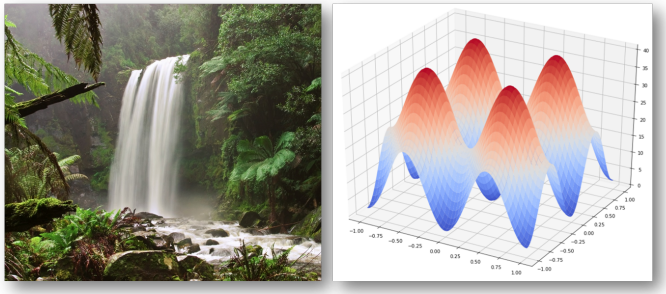
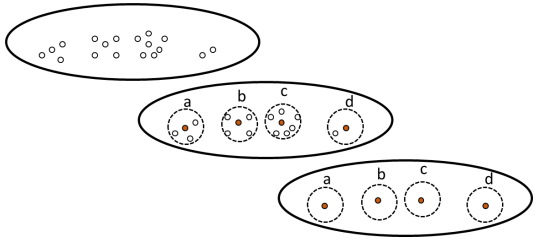


Figure: Environment vs. Fitness landscape

Niches

Niche: subspace in the environment with a **finite amount** of physical resources that can support different types of life



- Niches favor the divergence of character in natural evolution
- Are there niches in the fitness landscape? How to create niches since the environment is missing?

Levels in EC

Levels in Biology

- **Genotype**: the genetic constitution of an organism
- **Phenotype**: the composite of the organism's observable characteristics or traits
- **Fitness**: individual's ability to propagate its genes (well, almost)

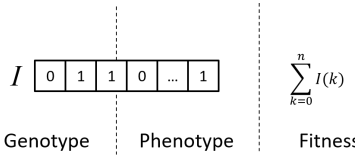
Levels in EC

Levels in EC — a modest proposal (Squillero and Tonda, 2016)

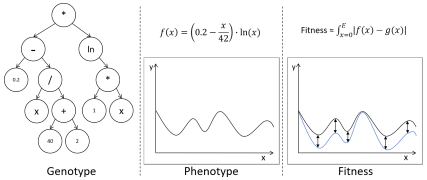
- **Fitness**: how well the candidate solution is able to solve the target problem
- **Genotype**: the internal representation of the individual, i.e., what is directly manipulated by genetic operators
- **Phenotype**: the candidate solution that is encoded in the genotype
  - the intermediate form in which the genotype needs to be transformed into for evaluating fitness
  - if genotype can be directly evaluated: genotype and phenotype coincide

Levels in EC

- Simple representation: Genotype and Phenotype coincide

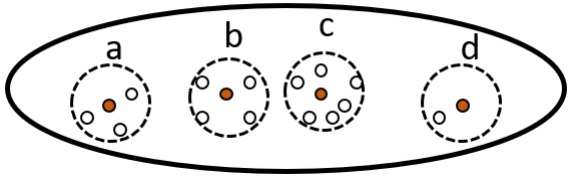


- More common situation: Genotype and Phenotype differ



Artificial Niches

- Niching in EC: grouping *similar* individual
  - similar spatial positions (i.e., islands)
  - similar genotypes (i.e., niching)
  - similar phenotypes
- Explicit vs. implicit neighborhood
- Several approaches are based on niching (far more than those who include "niching" in their name)

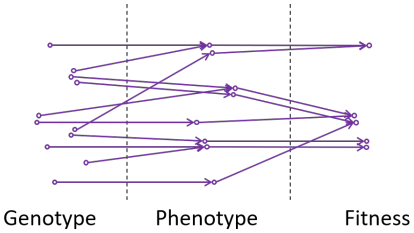


Diversity in EC

- Detecting whether two individuals are clones, i.e., identical, is often an easy task at any level
- Clone extermination
  - Same genotype
  - Same phenotype
  - Same fitness (*proxy*?)

Diversity in EC

- Individuals that look identical at a given level may be different at a previous level (aliasing)



- Different fitness values imply different phenotypes, different phenotypes imply different genotypes
  $F_x \neq F_y \implies P_x \neq P_y \implies G_x \neq G_y$, but the reverse implications are not true.

Diversity in EC

What is diversity?

- A distance metric, i.e., how far an individual is
  - ... from (a subset of) the whole population
  - ... from another single individual
- A property of the individual (amount of novelty), but at what level?
  - Phenotype
  - Genotype
  - Fitness
- A property of the whole population, but at what level?
  - Phenotype
  - Genotype
  - Fitness
- $F_x \approx F_y \implies P_x \approx P_y \implies G_x \approx G_y$?

Diversity in EC

How can diversity be measured?

- Fitness
  - Usually trivial
- Phenotype
  - Usually ad-hoc
- Genotype
  - Different genotypes in the population
  - GP subtree frequency
  - Edit distance (a.k.a., Levenshtein distance)
  - Entropy and free energy

Mechanisms for promoting diversity in EC

Caveat

The **end goal** an the optimization process is reaching better solutions in less time

- Promoting diversity has often been seen as the key factor to improve performances
- Promoting diversity is a mere **means goal** (yet a quite important one)
- No distinction is made here whether the means goal is to **preserve** existing diversity or to **increase** diversity

Mechanisms for promoting diversity in EC

A methodology for promoting diversity **alters** the **selection probability** of individuals during parent or survival selection.

Definition

$$\bar{p}_{x|\Psi} = p_{x|\Psi} \cdot \xi(x|\Psi)$$

- $\bar{p}_{x|\Psi}$ is the selection probability of individual x given that all individuals in set Ψ are also chosen
- $p_{x|\Psi}$ is the selection probability of individual x without diversity promotion when all individuals in set Ψ are also chosen
- $\xi(x|\Psi)$ is the corrective factor introduced by diversity promotion

- Mere definition: we **do not** imply that a mechanism operates explicitly on the selection operator, but its effects on selection probabilities are assessed to classify it

Mechanisms for promoting diversity in EC

Types of selection:

- Parent selection: α (usually non-deterministic)
- Survival selection: ω (usually deterministic)

Level of diversity promotion:

- Lineage
- Phenotype
- Genotype
- ~~Fitness~~: used as a proxy for either phenotype or genotype

Mechanisms for promoting diversity in EC

Lineage-based methodologies: The value of $\xi(\circ)$ does not depend on individual structure nor behavior, but it can be determined considering circumstances of its birth (e.g., time, position) — LBMs can be applied to any kind of problem, even in addition to other diversity preservation methods

- Allopatric selection [ω]
- Cellular EA [$\alpha\omega$]
- Deterministic crowding [$\alpha\omega$]
- Gender [α]
- Island model [$\alpha\omega$]
- Segregation [$\alpha\omega$]

Island model [$\alpha\omega$]

- The population is partitioned into sub-populations
- Only local interactions are allowed
- Periodically, individuals are moved between sub-populations (migrants)

Rationale

Different populations may explore different parts of the search space... but global interactions can be useful

Deterministic crowding [$\alpha\omega$]

- Offspring compete against parents for survival

Rationale

The offspring is *similar* to the parents and therefore compete for the same resources

- Flexible niching with implicit neighborhood
- Parents and offspring occupy the same niche
- No need for evaluating the similarity
- *Probabilistic* crowding
- *Generalised* crowding

Mechanisms for promoting diversity in EC

Genotype-based methodologies: Particularly effective when it is possible to define a sensible distance between genotypes — PBMs are often used to avoid overexploitation of peaks in the fitness landscape; promote the generation of new solutions very far from the most successful ones; preserve variability in the gene pool

- Clearing [$\alpha\omega$]
- Clone extermination [ω]
- Delta entropy/pseudo entropy [α]
- Diversifiers [$\alpha\omega$]
- Fitness sharing [$\alpha\omega$]
- Genetic Diversity Evaluation Method [$\alpha\omega$]
- Reference points partitioning [ω]
- Restricted tournament selection [ω]
- Sequential niching [$\alpha\omega$]
- Standard crowding [ω]
- Two-level diversity selection [α]

Clearing [$\alpha\omega$]

- Inside niches of a certain radius, the best k individuals retain their fitness while the rest are zeroed

Rationale

Set a hard limit to population density

- Flexible niching with explicit neighborhood

Fitness sharing [$\alpha\omega$]

- Scale down individual fitness in densely populated areas

Rationale

Set a hard limit to population density

- Flexible niching with explicit neighborhood

Mechanisms for promoting diversity in EC

Phenotype-based methodologies: Usually impractical, but sometimes fitness distance can be used as a proxy for phenotype distance

- Crowded-comparison operator [α]
- Extinction [ω]
- Gender [α]
- Hierarchical fair competition [$\alpha\omega$]
- Lexicase selection [α]
- Random immigrants [$\alpha\omega$]
- Restricted Tournament Selection [ω]
- Tarpeian method [$\alpha\omega$]
- Vector evaluated genetic algorithm [α]

Restricted Tournament Selection [ω]

- New individuals compete with the most similar individual in a random niche

Rationale

Set a hard limit to population density

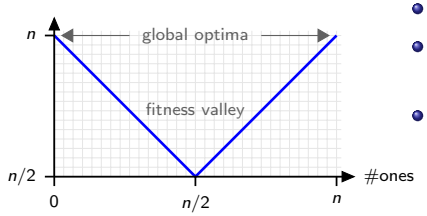
- Flexible niching with explicit neighborhood

Agenda

1. Introduction
2. How Diversity Benefits Exploration – Case Study on TwoMax
   - The TwoMax Function
   - No Diversity Mechanism
   - Avoiding Duplicates
   - Crowding Methods
   - Fitness Sharing
   - Clearing
   - Conclusions from TwoMax
3. How Diversity Benefits Crossover
4. Conclusions and Outlook

TwoMax function

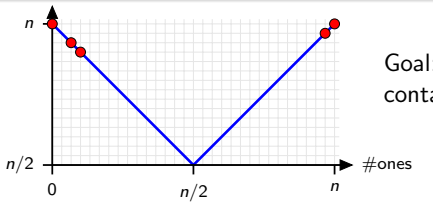
$$\textsc{TwoMax}(x) := \max\left\{\sum_{i=1}^{n} x_i, n - \sum_{i=1}^{n} x_i\right\}$$



- Popular test problem.
- Two symmetric branches.
- Optima have maximum Hamming distance.
- Optima have a large basin of attraction (Hamming ball of radius $n/2$)

Genotype: 0010111001101
Phenoype: number of ones
Fitness: number of ones or number of zeros

Goal for TwoMax



Goal: evolve a population containing both optima

- Goes beyond maximising fitness
  - Alternative view: imagine one optimum to be a local one. If you can find both, you have found the unique global optimum.
- Assessing inherent ability to maintain diversity for a long time
- Can we find different optima in a single run (no restarts)?
- Challenging benchmark for diversity mechanisms
  - May need to escape one optimum and traverse fitness valley
- Mechanisms that can do this are likely to perform well in other settings.

$(\mu+1)$ EA

Simple baseline EA to focus on effect of added diversity mechanisms.

Algorithm 1 $(\mu+1)$ EA

1: Initialise P with μ individuals chosen uniformly at random
2: **while** stopping criterion **not** met **do**
3: Choose $x \in P$ uniformly at random
4: Create y by flipping each bit in x independently w/ prob. $1/n$.
5: Choose $z \in P$ with worst fitness
6: **if** $f(y) \geq f(z)$ **then** y replaces z

Note:
No crossover as crossing individuals from different branches yields poor fitness.

Will discuss benefits of diversity for crossover in the next section.

Runtime Analysis

Goal

Bounds on the (expected) time for an algorithm to reach a target.

Target for TwoMax: evolve a population containing both optima.

Bounds are mathematically rigorous and scale with problem size n.

Methods

Combining techniques from analysis of algorithms, randomised algorithms, discrete mathematics and probability theory.

- Asymptotic notation $(O, \Omega, \Theta, o, \omega)$
- Markov chain analysis, tail bounds, drift analysis, ...

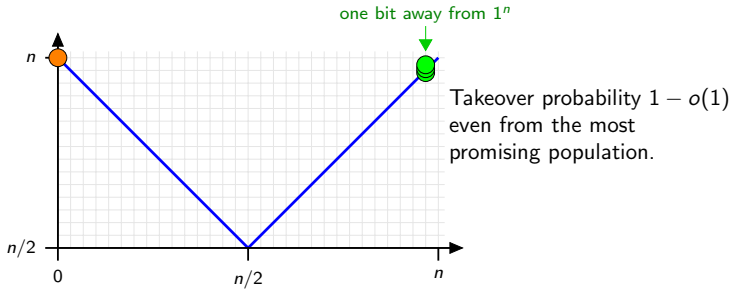
Weaknesses

- It's hard! Problems/algorithms must be reasonably simple.
- Results are usually specific to particular algorithms and particular problems or problem classes.

No Diversity Mechanism

Theorem (Adapted from Theorem 1 in Friedrich et al. (2009))

With high probability, the $(\mu+1)$ EA *with no diversity-preserving mechanism and* $\mu = o(n/\log n)$ *does not find both optima of* TwoMax *in time* n^{n-1}. *The expected time is* $\Omega(n^n)$.

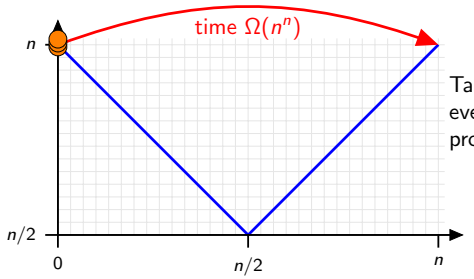


Takeover probability $1 - o(1)$ even from the most promising population.

No Diversity Mechanism

Theorem (Adapted from Theorem 1 in Friedrich et al. (2009))

With high probability, the $(\mu+1)$ EA *with no diversity-preserving mechanism and* $\mu = o(n/\log n)$ *does not find both optima of* TwoMax *in time* n^{n-1}. *The expected time is* $\Omega(n^n)$.



Takeover probability $1 - o(1)$ even from the most promising population.

Avoiding Genotype Duplicates (Clone Extermination)

Idea: only add offspring if their genotype is not contained in the population [genotype/ω].

Theorem (Adapted from Theorem 2 in Friedrich et al. (2009))

With high probability, the $(\mu+1)$ EA *with genotype diversity and* $\mu = o(n^{1/2})$ *does not find both optima of* TwoMax *in time* n^{n-2}. *The expected time is* $\Omega(n^{n-1})$.

Takeover is slightly more difficult, but the same analysis works in almost the same way.

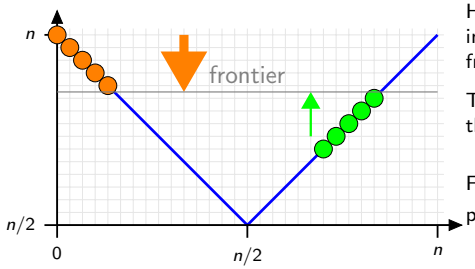
- It is not sufficient to create clones of 0^n; you have to create search points with a single 1-bit.
- From a most promising population, search points with fitness $n-1$ compete.
- Restriction on population size is slightly tougher: $\mu = o(n^{1/2})$ instead of $\mu = o(n/\log n)$.

Fitness Diversity

Idea: avoid storing multiple individuals with the same fitness. If an offspring x has the same fitness value as $y \in P$ then x replaces y [phenotype/ω].

Theorem (Adapted from Theorem 3 in Friedrich et al. (2009))

The expected time for the $(\mu+1)$ EA *with fitness diversity to find both optima of* TwoMax *is exponential.*



High-fitness and low-fitness individuals compete at the frontier.

The former tend to suppress the latter.

Fitness distance is a very bad proxy of phenotype distance!

Crowding for TwoMax

Idea: offspring directly compete with their parents.

Mutation-only EAs: offspring compete with their only parent. Then the population contains μ lineages that evolve independently.

Variants of Crowding

- Deterministic Crowding: offspring survives if no worse than parent.
- Probabilistic Crowding: offspring survives with a probability proportional to its fitness, $\frac{f(y)}{f(x)+f(y)}$.
- Generalized Crowding: mixing the above according to a parameter ϕ.
- Restricted Tournament Selection: offspring competes against a similar individual.

Deterministic Crowding

Idea: offspring directly compete with their parents and survive if no worse than parent [lineage-phenotype/ω].

The population contains μ lineages that evolve like $(1+1)$ EAs.

Theorem (Adapted from Theorem 4 in Friedrich et al. (2009))

The $(\mu+1)$ EA *with deterministic crowding on* TwoMax *reaches a population of global optima in expected time* $O(\mu n \log n)$. *Then the population contains both optima with probability* $1 - 2^{-\mu+1}$.

All μ lineages make independent choices whether to reach 0^n or 1^n. The probability all end up in the same optimum is $2 \cdot 2^{-\mu}$.

$(\mu+1)$ EA with Restricted Tournament Selection (RTS)

Idea: create a tournament of w randomly chosen individuals. The individual with the smallest distance to the offspring compete against each other and the better one survives [genotype/ω].

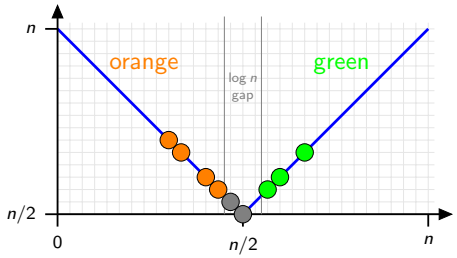
Distances used:

- Genotypic distance: Hamming distance
- Phenotypic distance: difference in the number of ones, $\mathrm{d}(z,y) := |\sum_{i=1}^{n} z_i - \sum_{i=1}^{n} y_i|$.

Large Window Sizes Are Effective

Theorem 3 in Covantes Osuna and Sudholt (2022) (informal)

If $\mu = o(\sqrt{n}/\log n)$ and $w \geq 2.5\mu \ln n$ then the $(\mu+1)$ EA with RTS is as effective as deterministic crowding.



Two types of failure:

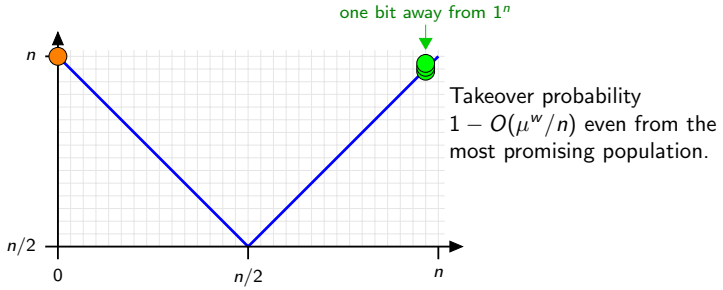
- bad initialisation (prob. $\sim 2^{-\mu}$)
- takeover

If $w \geq 2.5\mu \ln n$ then niches never compete (with high probability)

Small Window Sizes Can Fail

Theorem 3.4 in Covantes Osuna and Sudholt (2018a)

The probability that the $(\mu+1)$ EA with RTS finds both optima on TwoMax in time n^{n-1} is at most $O(\mu^w/n)$.
If $\mu^w \leq \varepsilon n$ the expected time for finding both optima is $\Omega(n^n)$.



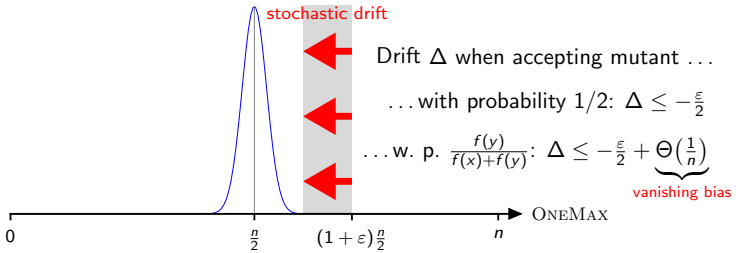
Takeover probability $1 - O(\mu^w/n)$ even from the most promising population.

Probabilistic Crowding

Idea: offspring y competes against parent x, accepted w/ prob. $\frac{f(y)}{f(x)+f(y)}$.

Theorem (Covantes Osuna and Sudholt (2018a, 2020))

With probability $1 - 2^{-\Omega(n)}$ *the* $(\mu+1)$ EA *with probabilistic crowding on* OneMax *or* TwoMax *does not find a search point with fitness at least* $(1+\varepsilon)n/2$ *in* 2^{cn} *function evaluations.*



Fitness-proportional selection is no better than a blind random walk!
Negative drift theorem (Oliveto and Witt, 2011, 2012): each lineage fails.

Generalised Crowding (Galán and Mengshoel, 2010)

Introduces a scaling factor ϕ that diminishes the fitness of the inferior search point.

Probability of accepting the offspring y over the parent x is:

$$\begin{cases} \frac{f(y)}{f(y)+\phi\cdot f(x)} & \text{if } f(y) > f(x), \\ 0.5 & \text{if } f(y) = f(x), \\ \frac{\phi\cdot f(y)}{\phi\cdot f(y)+f(x)} & \text{if } f(y) < f(x). \end{cases} \quad (1)$$

Special cases:
$\phi = 1$ gives probabilistic crowding
$\phi = 0$ gives deterministic crowding

Results for Generalized Crowding (Covantes Osuna and Sudholt, 2018a, 2020)

If $\phi < 1/(e^2 n)$, the $(\mu{+}1)$ EA with generalised crowding is as effective as deterministic crowding on TwoMax.

If $\phi = \Omega(n^{-1+\varepsilon})$, the expected time on TwoMax is exponential.

Fitness Sharing

Fitness Sharing

Idea: similar individuals have to share resources, so share their fitness.

Fitness is derated by a measure of similarity:

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

Possible distances:

- Genotypic distance [genotype/ω]: Hamming distance
- Phenotypic distance [phenotype/ω]: difference in the number of ones, $\mathrm{d}(z,y) := |\sum_{i=1}^{n} z_i - \sum_{i=1}^{n} y_i|$.

$(\mu{+}1)$ EA with Fitness Sharing

Theorem (Oliveto, Sudholt, and Zarges (2014, 2019))

The $(\mu{+}1)$ EA with phenotypic fitness sharing and $\mu \geq 3$ always finds both optima in expected time $O(\mu n \log n)$.

- **Phase 1:** larger true fitness means higher shared fitness $\rightarrow$ all individuals climb up
- **Phase 2:** at some point, at the latest once the population contains two copies of 0^n, the worst individual on the branch survives and tunnels through the fitness valley.

Note: $\mu \geq 3$ is needed, $\mu = 2$ is not enough:

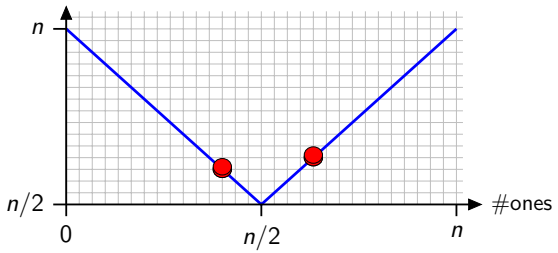
Theorem

Success probability of a (2+1) EA with fitness sharing is $0 < c < 1/2$ (i. e. worse than 2 independent runs)

Fitness Sharing with Offspring Populations

What about fitness sharing in a $(\mu{+}\lambda)$ EA?

Overpopulation: if a cluster creates many offspring, the shared fitness of all derates, and the whole cluster may go extinct.



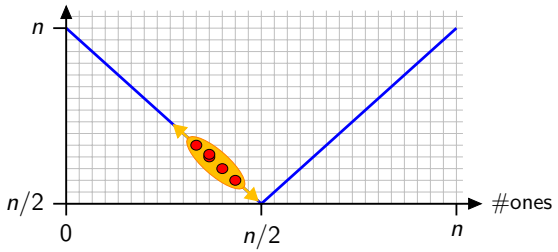
Results

With probability $1 - o(1)$ the $(2{+}\lambda)$ EA with $2 \leq \lambda = O(1)$ will reach a population on one branch only.
The (2+2) EA needs time $\Omega(n^{n/2})$ with probability $1 - o(1)$.

Population-based Fitness Sharing

Theorem (Friedrich et al. (2009))

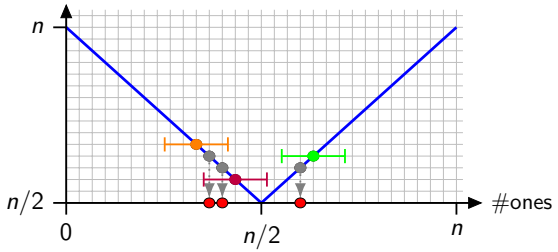
The $(\mu+1)$ EA *with phenotypic fitness sharing selecting the best population* $P^* = \arg\max\{\sum_{x\in P'} f(x, P') \mid |P'| = \mu\}$ *finds both optima on* TwoMax *for all* $\mu \geq 2$ *in expected time* $O(\mu n \log n)$.



Population is expanding both ways, tunneling through the fitness valley.

Clearing

Idea: best individuals are declared "winners" and individuals within a clearing radius σ have their fitness "cleared" (set to minimum) [genotype/ω] or [phenotype/ω].



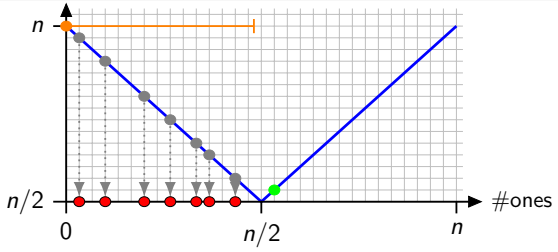
Advantages:

- Winners always survive (elitism)
- Cleared individuals perform random walk on a flat fitness landscape and can tunnel through fitness valleys

Clearing

Theorem (Covantes Osuna and Sudholt (2017, 2019), simplified)

The expected time for the $(\mu+1)$ EA *with clearing using genotypic of phenotypic distances,* $\mu \geq n^2/4$ *and* $\sigma = n/2$ *finding both optima on* TwoMax *is* $O(\mu n \log n)$.



If the population size is large enough, the cleared individuals perform a random walk and will tunnel through the fitness valley.

Conjecture: population size of $\mu = O(n)$ is sufficient as well.

Summary of Results on TwoMax

Diversity Mechanism	Success probability in time $O(\mu n \log n)$	Conditions
No Mechanism	$o(1)$	$\mu = o(n/\log n)$
No Genotype Duplicates	$o(1)$	$\mu = o(n^{1/2})$
Fitness Diversity	$o(1)$	
Deterministic Crowding	$1 - 2^{-\mu+1}$	
Probabilistic Crowding	$2^{-\Omega(n)}$	
Generalised Crowding	$1 - 2^{-\mu+1}$	$\phi \leq 1/(e^2 n)$
Restricted Tournament Selection	$\approx 1 - 2^{-\mu+1}$	$w \geq 2.5\mu \ln n$
Fitness Sharing	1	$\mu \geq 3$
Population-based Fitness Sharing	1	$\mu \geq 2$
Clearing	1	$\mu \geq n^2/4$

Conclusions from TwoMax Analysis

- The choice of diversity mechanism makes a huge difference (exponential time versus time $O(\mu n \log n)$).
- The choice of parameters and distance measures is crucial.
  - For fitness sharing, $\mu = 2$ vs. $\mu = 3$ makes a huge difference.
  - For restricted tournament selection, large window sizes w work while small ones don't.
  - Likewise for the population size in Clearing.
- Theory can guarantee which parameters are safe to use.
- However, the parameter choice is often conservative – experiments suggest that less extreme parameters also work well for the window size w in Restricted Tournament Selection and the population size μ in Clearing.

What About Other Functions?

Some analyses only apply to TwoMax (e.g. Fitness Sharing).

Other analyses generalise to wider function classes:

- Poor performance of Probabilistic Crowding generalises to a wide class of functions with bounded gradients (Covantes Osuna and Sudholt, 2020).
  - This can be fixed with very aggressive exponential scaling.
- Clearing analysis applies to generalisations of TwoMax, including multimodal problem classes (Jansen and Zarges, 2016).
- Empirical study on these multimodal problems was done in Covantes Osuna and Sudholt (2018b).
  - Fitness sharing (with genotypic distances) performed worse than expected.
  - Clearing was effective and the only mechanism able to tunnel through fitness valleys.

Diversity Mechanisms for Balance

Oliveto and Zarges (2015) did similar analyses on the Balance function that has a very different fitness landscape to TwoMax.

Diversity Mechanism	TwoMax	Balance
No Mechanism	☹	☹
No Genotype Duplicates	☹	☹
Fitness Diversity	☹	☺
Deterministic Crowding	☺	☹
Fitness Sharing	☺	☺☹

The worst mechanism for TwoMax is the best for Balance and the best for TwoMax performs poorly on Balance.

Agenda

1 Introduction

2 How Diversity Benefits Exploration – Case Study on TwoMax

3 How Diversity Benefits Crossover
- Real Royal Road Functions
- Hill Climbing
- Coloring Problems
- Escaping Local Optima on Jump

4 Conclusions and Outlook

Scheme of a Genetic Algorithm

Now: Examples where diversity benefits crossover in standard GAs.

Algorithm 2 Scheme of a $(\mu+\lambda)$ GA

```
Initialise P with μ individuals chosen u. a. r.
while stopping criterion not met do
    for i = 1, ..., λ do
        Choose p ∈ [0, 1] uniformly at random.
        if p ≤ p_c then
            Select two parents x_1, x_2.
            Let y := crossover(x_1, x_2).
        else
            Select a parent y.
        Flip each bit in y independently w/ prob. 1/n.
        Add y to P'.
    Let P contain the μ best individuals from P ∪ P'; break ties
    according to a specified tie-breaking rule.
```


Scheme of a Genetic Algorithm

- Recombination
  - mixes together two or more solutions to create the offspring
  - associated with the idea of exploration
- Mutation
  - performs a (usually small) change in an individual
  - associated with the idea of exploitation
- When all parents are **very similar**, the effectiveness of recombination is limited
- The ability to explore remote parts of the search space is impaired
- *Conventional wisdom* suggests that increasing **diversity** should be generally beneficial

Real Royal Road Functions (Jansen and Wegener, 2005)

First rigorous proof that crossover can make a difference between polynomial and exponential expected runtimes.

Denoting by $b(x)$ the length of the longest block consisting of ones only (e. g. $b(100\mathbf{111}011) = 3$), then

$$R_n(x) = \begin{cases} 2n^2 & \text{if } x = 1^n, \\ n|x|_1 + b(x) & \text{if } |x|_1 \leq 2n/3, \\ 0 & \text{otherwise.} \end{cases}$$

Theorem (Theorem 3 in Jansen and Wegener (2005), simplified)

The expected optimization time of the (n+1) GA with $0 < p_c < 1$ constant, breaking ties towards including individuals with the fewest duplicates in $P \cup P'$, on R_n is $O(n^4)$.

Diversity ensures *different* large blocks of 1s evolve and crossover recombines them: $1111111000 + 0001111111 = 1111111111$

Diversity and Crossover Speed Up Hill Climbing

Diversity and crossover even help on $\textsc{OneMax} := \sum_{i=1}^{n} x_i$.

Theorem (Sudholt (2012, 2017), strongly simplified)

The expected time of the (2+1) GA with $0 < p_c < 1$ constant, breaking ties towards including individuals with the fewest duplicates in $P \cup P'$, on $\textsc{OneMax}$ is $1.35n \ln n + O(n)$.

Twice as fast as the fastest mutation-only EA (up to small-order terms).

- Neutral mutations create different building blocks
- Diversity helps these to survive
- Uniform crossover easily recombines different building blocks: $1101001 + 1100101 = 1101101$.

How important is diversity in the tie-breaking?

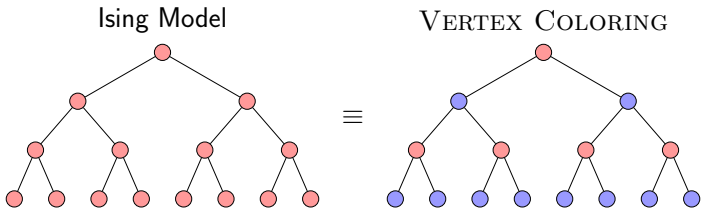
(2+1) GA with uniform random tie-breaking has expected time $2.22n \ln n$ (Corus and Oliveto, 2018, Oliveto et al., 2020).

Worse constant, though still better than mutation-only EAs (2.71).

Solving Coloring Problems with Diversity and Crossover

Ising Model: color a graph with 2 colors, maximising the number of edges whose end points have *the same* color.

- bit-flip symmetry
- all colorings that only use one color are optimal
- synchronisation problems: parts of graphs are colored differently
- on bipartite graphs it's equivalent to Vertex Coloring with 2 colors (seeking different colors):

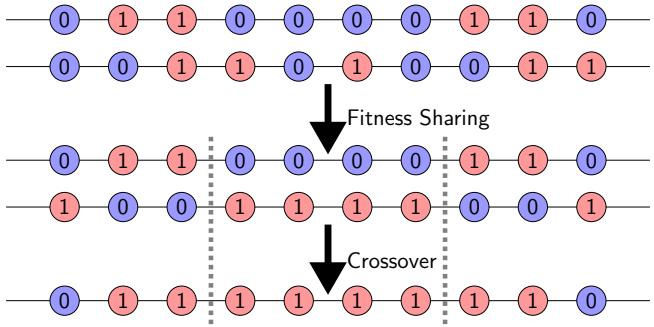


Coloring Rings

Theorem (Fischer and Wegener (2005), simplified)

A (2+2) GA with 2-point crossover and population-based fitness sharing colors rings in expected time $O(n^2)$.

Factor n faster than mutation-only EAs (semi-rigorous proof).

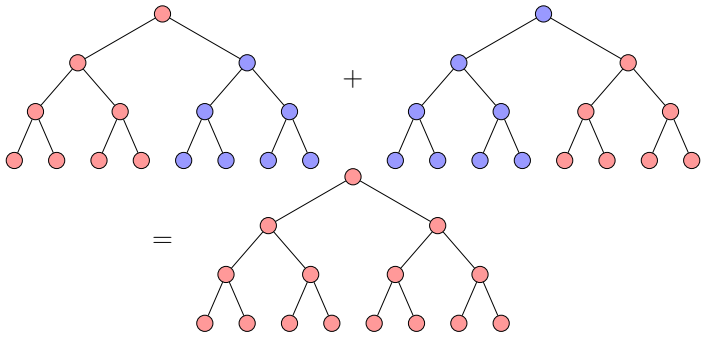


Coloring Trees

Theorem (Sudholt (2005))

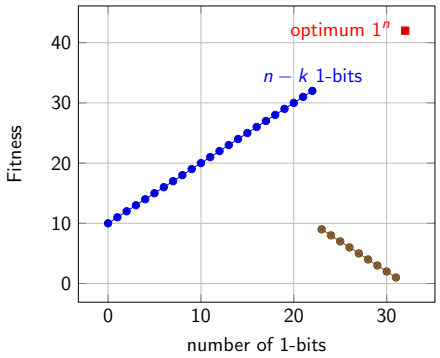
A (2+2) GA with 2-point crossover and population-based fitness sharing colors binary trees in expected time $O(n^3)$.

All mutation-only EAs need exponential expected time!



Jump_k: A Function With Tuneable Difficulty

Jump_k (Jansen and Wegener, 2002): "jump" of k bits required.



Genotypes of local optima: 111001110111 (k 0-bits).

Expected time of (1+1) EA on Jump_k is $O(n \log n + n^k)$.

Crossover on Jump_k

Success probability for mutation: $\sim n^{-k}$.

0 0 1 1 1 1 1 1
1 0 1 1 1 1 0 1

Crossover may set disagreeing bits to 1, agreeing 0s must all be flipped by mutation.

0 1 1 0 1 1 1 1
1 0 1 1 1 1 0 1

Best case (perfect diversity): prob. 4^{-k} of jump to optimum.

Superpolynomial gap for $k = \log n$

	4^k	n^k
$n = 10$	100	2099
$n = 100$	10,000	$2 \cdot 10^{13}$
$n = 1000$	1,000,000	$8 \cdot 10^{29}$

Escaping from Local Optima with Emerging Diversity

Superpolynomial gap for GAs avoiding genotype duplicates (Jansen and Wegener, 2002, Kötzing et al., 2011).

- Analyses used vanishing (unrealistic) crossover probabilities
- Crossover pessimistically assumed harmful for diversity

Does crossover really harm diversity without diversity mechanisms?

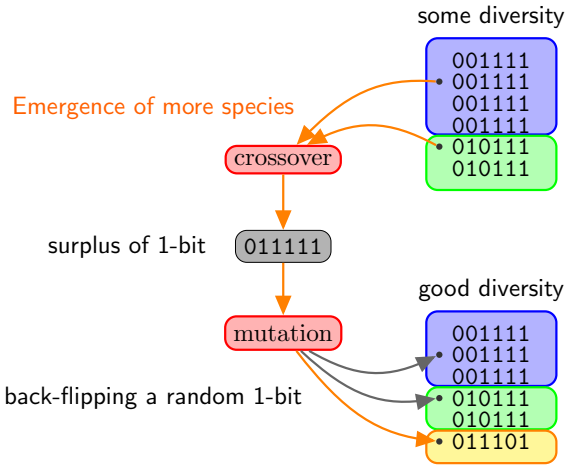
Theorem 6 in Dang et al. (2016a, 2018), simplified

The expected optimization time of the $(\mu+\lambda)$ GA with $p_c = 1$ and $\mu = n$ on Jump_k is $O(n^{k-1} \log n)$.

Observations from the analysis

- Diversity emerges naturally
- Crossover helps create diversity by mixing genes
- Interplay of mutation and crossover is a catalyst for diversity

How Crossover+Mutation Create Diversity on Jump



Escaping from Local Optima with Diversity Mechanisms

Can results be improved through explicit diversity mechanisms?

Dang et al. (2016b): use diversity mechanisms as a tie-breaking rule for selection.

Expected times on Jump (Dang et al., 2016b) (simplified for $k \geq \log n$)

Mechanism	**Best runtime**
No mechanism	$O(n^{k-1} \log n)$
Duplicate elimination	$O(n^{k-1})$
Duplicate minimization	$O(n^{k-1})$
Deterministic crowding	$O(ne^{5k}2^k)$
Convex hull maximisation	$O(n^2 \log n + 4^k)$
Hamming distance maximisation	$O(4^k)$
Fitness sharing	$O(4^k)$
Island model	$O(4^k)$

Conclusions on Crossover and Diversity

- In the absence of diversity, crossover is useless.
- Diversity mechanisms promote and maintain different "building blocks" in the population that can be combined effectively.
- Constant-factor speedups for hill climbing (OneMax).
- Larger speedups for Real Royal Roads, Jump and Coloring problems.

Agenda

1 Introduction
2 How Diversity Benefits Exploration – Case Study on TwoMax
3 How Diversity Benefits Crossover
4 Conclusions and Outlook

Conclusions

Maintaining and promoting diversity in evolutionary algorithms is a very important task.

Benefits of diversity

- Enhance the exploration capability of EAs
- Prevent premature convergence
- Improve robustness
- Enable crossover to work more effectively

Observations

- There is a multitude of diversity mechanisms and it is difficult to decide which mechanism performs best.
- Runtime analysis can quantify the performance of EAs enhanced with diversity mechanisms.
- Diversity can also emerge naturally, without any mechanisms.
- Which mechanism performs best depends on the problem.

Future Work

Future theoretical work

- Study other diversity mechanisms
- Performance on other multimodal problems (e. g. multimodal problem class by Jansen and Zarges (2016)).
- Diversity in other bio-inspired paradigms, e. g. PSO

Bridging theory and practice

- Theoretically analyse more practically relevant scenarios
- Extract practical guidelines from theoretical work
- Combine empirical and theoretical work
- Design diversity mechanisms that perform well across a wide range of problems

Acknowledgments

The presenters would like to thank:

- COST Action CA15140 'Improving Applicability of Nature-Inspired Optimisation by Joining Theory and Practice (ImAppNIO)' supported by COST (European Cooperation in Science & Technology).
- Edgar Covantes Osuna for proposing and completing an excellent PhD on the analysis of diversity mechanisms (funded by CONACYT).
- Dagstuhl seminar 17191 "Theory of Randomized Optimization Heuristics" for inspiring discussions on diversity.

References I

D. Corus and P. S. Oliveto. Standard steady state genetic algorithms can hillclimb faster than mutation-only evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 22(5):720–732, 2018.

E. Covantes Osuna. *Theoretical and Empirical Evaluation of Diversity-preserving Mechanisms in Evolutionary Algorithms: On the Rigorous Runtime Analysis of Diversity-preserving Mechanisms in Evolutionary Algorithms*. PhD thesis, University of Sheffield, 2019.

E. Covantes Osuna and D. Sudholt. Analysis of the Clearing Diversity-Preserving Mechanism. In *Proceedings of the 14th ACM/SIGEVO Conference on Foundations of Genetic Algorithms*, pages 55–63. ACM, 2017.

E. Covantes Osuna and D. Sudholt. Runtime Analysis of Probabilistic Crowding and Restricted Tournament Selection for Bimodal Optimisation. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 929–936. ACM, 2018a.

E. Covantes Osuna and D. Sudholt. Empirical Analysis of Diversity-Preserving Mechanisms on Example Landscapes for Multimodal Optimisation. In *Parallel Problem Solving from Nature – PPSN XV*, pages 207–219. Springer, 2018b.

E. Covantes Osuna and D. Sudholt. On the Runtime Analysis of the Clearing Diversity-Preserving Mechanism. *Evolutionary Computation*, 27(3):403–433, 2019.

E. Covantes Osuna and D. Sudholt. Runtime analysis of crowding mechanisms for multimodal optimisation. *IEEE Transactions on Evolutionary Computation*, 24(3):581–592, 2020.

E. Covantes Osuna and D. Sudholt. Runtime analysis of restricted tournament selection for bimodal optimisation. *Evolutionary Computation*, 30(1):1–26, 2022.

D.-C. Dang, T. Friedrich, T. Kötzing, M. S. Krejca, P. K. Lehre, P. S. Oliveto, D. Sudholt, and A. M. Sutton. Emergence of Diversity and its Benefits for Crossover in Genetic Algorithms. In *Proceedings of the 14th International Conference Parallel Problem Solving From Nature (PPSN)*. Springer, 2016a.

D.-C. Dang, T. Friedrich, M. S. Krejca, T. Kötzing, P. K. Lehre, P. S. Oliveto, D. Sudholt, and A. M. Sutton. Escaping Local Optima with Diversity-Mechanisms and Crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '16)*, pages 645–652. ACM Press, 2016b.

References II

D.-C. Dang, T. Friedrich, T. Kötzing, M. S. Krejca, P. K. Lehre, P. S. Oliveto, D. Sudholt, and A. M. Sutton. Escaping local optima using crossover with emergent diversity. *IEEE Transactions on Evolutionary Computation*, 22(3):484–497, 2018.

C. Darwin. *On the Origin of Species by Means of Natural Selection, or, The Preservation of Favoured Races in the Struggle for Life*. John Murray, 1859.

R. Dawkins. *The Extended Phenotype: The Long Reach of the Gene*. Oxford University Press, 2016.

A. Eiben and J. Smith. *Introduction to Evolutionary Computing*. Natural Computing Series. Springer Berlin Heidelberg, 2015.

S. Fischer and I. Wegener. The One-dimensional Ising Model: Mutation versus Recombination. *Theoretical Computer Science*, 344(2–3):208–225, 2005.

T. Friedrich, P. S. Oliveto, D. Sudholt, and C. Witt. Analysis of Diversity-preserving Mechanisms for Global Exploration. *Evolutionary Computation*, 17(4):455–476, 2009.

S. F. Galán and O. J. Mengshoel. Generalized Crowding for Genetic Algorithms. In *Proceedings of the 12th Annual Conference on Genetic and Evolutionary Computation (GECCO '10)*, pages 775–782. ACM, 2010.

J. H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, 1992.

T. Jansen and I. Wegener. On the analysis of evolutionary algorithms—a proof that crossover really can help. *Algorithmica*, 34(1):47–66, 2002.

T. Jansen and I. Wegener. Real royal road functions—where crossover provably is essential. *Discrete Applied Mathematics*, 149(1):111–125, 2005.

T. Jansen and C. Zarges. Example landscapes to support analysis of multimodal optimisation. In *Proceedings of the Parallel Problem Solving from Nature – PPSN XIV*, pages 792–802. Springer, 2016.

T. Kötzing, D. Sudholt, and M. Theile. How crossover helps in pseudo-Boolean optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '11)*, pages 989–996. ACM Press, 2011.

E. Mayr. Darwin's Principle of Divergence. *Journal of the History of Biology*, 25(3):343–359, 1992.

References III

P. S. Oliveto and C. Witt. Simplified Drift Analysis for Proving Lower Bounds in Evolutionary Computation. *Algorithmica*, 59(3):369–386, 2011.

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

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.

P. S. Oliveto, D. Sudholt, and C. Zarges. On the Runtime Analysis of Fitness Sharing Mechanisms. In *Parallel Problem Solving from Nature – PPSN XIII*, pages 932–941. Springer, 2014.

P. S. Oliveto, D. Sudholt, and C. Zarges. On the benefits and risks of using fitness sharing for multimodal optimisation. *Theoretical Computer Science*, 773:53–70, 2019.

P. S. Oliveto, D. Sudholt, and C. Witt. A tight lower bound on the expected runtime of standard steady state genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '20)*. ACM, 2020.

G. Squillero and A. Tonda. Divergence of character and premature convergence: A survey of methodologies for promoting diversity in evolutionary optimization. *Information Sciences*, 329:782–799, Feb. 2016.

D. Sudholt. Crossover is Provably Essential for the Ising Model on Trees. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '05)*, pages 1161–1167. ACM Press, 2005.

D. Sudholt. Crossover speeds up building-block assembly. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '12)*, pages 689–696, 2012.

D. Sudholt. Parallel evolutionary algorithms. In J. Kacprzyk and W. Pedrycz, editors, *Handbook of Computational Intelligence*, pages 929–959. Springer, 2015.

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

D. Sudholt. The Benefits of Population Diversity in Evolutionary Algorithms: A Survey of Rigorous Runtime Analyses. In B. Doerr and F. Neumann, editors, *Theory of Evolutionary Computation: Recent Developments in Discrete Optimization*. Springer, 2020.