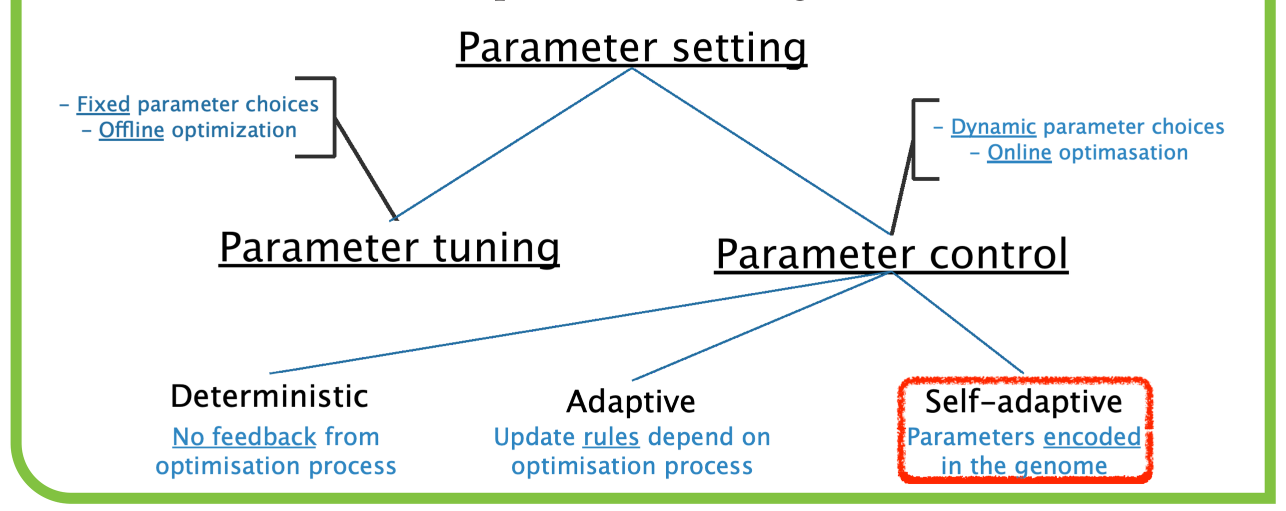
Self-adaptation via Multi-objectivisation: An Empirical Study

Xiaoyu Qin & Per Kristian Lehre @ University of Birmingham

PPSN 2022

Background

- Evolutionary Algorithms (EAs) are parameterised algorithms.
- Parameters setting can dramatically impact performance of EAs [1].
- Parameters setting is instance- and state-dependent [1].
- Classification scheme of parameters setting in EAs [2]:

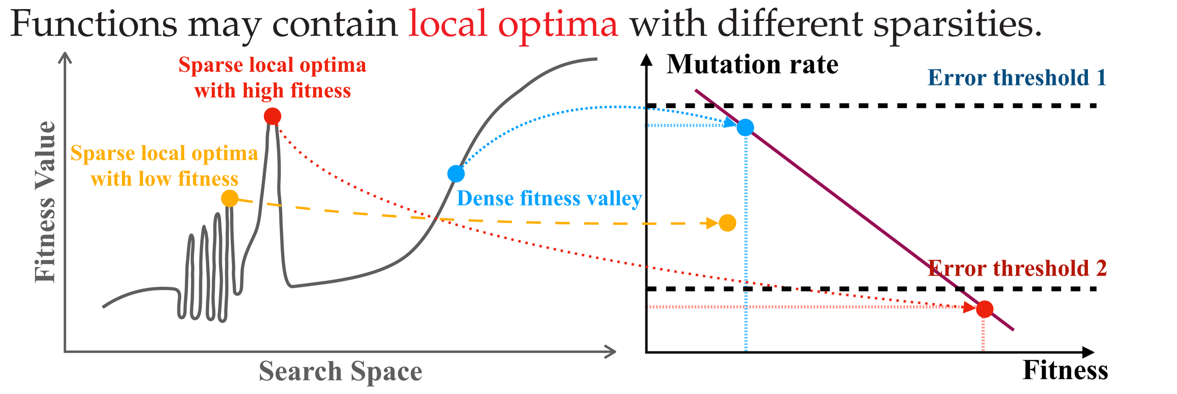


Motivation

- Elitist EAs can get stuck on local optima [7, 8].
- SparseLocalOpt ⇒ a kind of fitness landscapes with sparse deceptive regions (local optima) and dense fitness valleys [8]
- 3-tour. EA with sufficiently high mutation rate (close to the error threshold) can help [8] ⇒
  - ∘ Sparse local optimal individuals ⇒ higher chance to be selected but only a small percentage of such individuals survive mutation;
  - ∘ Dense fitness valley individuals ⇒ less chance of being selected but can have higher chance of surviving mutation.

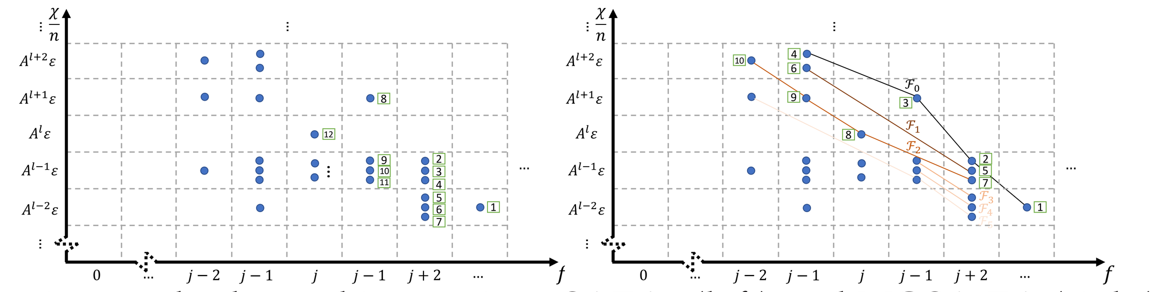
However,

- We need know the sparsity of local optima to set the mutation rate;
- Functions may contain local optima with different sparsities.



MOSA-EA

- Multi-objective sorting ⇒ Strict non-dominated Pareto fronts
- (μ, λ) selection ⇒ from sorted population
- Self-adapting mutation rate strategy ⇒ New mutation rate χ' is $A\chi$ with probability p_{inc}, or χ/A otherwise.
- Bit-wise mutation operator ⇒ New population.



Sorting methods used in previous SAEAs (left) and MOSA-EA (right)

Previous Work

The self-adaptive parameter control mechanism encodes the parameters in each individual and evolves the parameters together with its solution through variation operators.

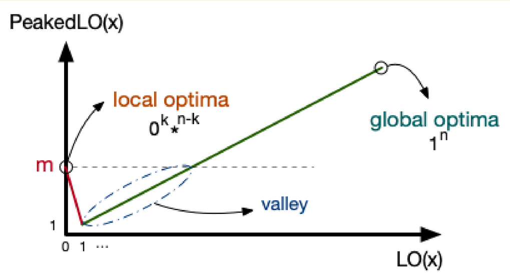
- Dang and Lehre [3] first presented that a self-adaptive population using two mutation rates can solve PeakedLO function which is a simple artificial two-peak function.
- Doerr et al. [4] rigorously analysed a self-adaptation mechanism on $(1, \lambda)$ EA on OneMax function.
- Case and Lehre [5] showed that the self-adaptation of mutation rate over a continuous interval can be effective on the unknown structure version of LeadingOnes function.

Algorithm 1 Framework for self-adaptive EAs

Require: Fitness function $f : \{0,1\}^n \to \mathbb{R}$. Population size $\lambda \in \mathbb{N}$. Sorting mechanism `Sort`. Selection mechanism P_{sel}. Self-adapting mutation rate strategy D_{mut}. Initial population $P_0 \in \mathcal{Y}^\lambda$.

```
for t in 0, 1, 2, ... until termination condition met do
    Sort(P_t, f)
    for i = 1, ..., λ do
        Sample I_t(i) ~ P_sel([λ]); Set (x, χ/n) := P_t(I_t(i)).
        Sample χ' ~ D_mut(χ).
        Create x' by mutating x with mutation rate χ'/n.
        Set P_{t+1}(i) := (x', χ'/n).
```

Theoretical Study [A]



$$\text{PeakedLO}_{m,k}(x) = \begin{cases} m & \text{if } x = \{0\}^k *^{n-k} \\ \text{LO}(x) & \text{otherwise.} \end{cases}$$

Tab. Runtime analyses of EAs on $\text{PeakedLO}_{m,k}$

Algorithm	$\text{PeakedLO}_{m,k}$	Runtime T
$(\mu+\lambda)$ EA	Any $n \geq k \in \Omega(n)$	$\Pr(T \leq e^{cn}) \leq e^{-\Omega(n)}$
(μ,λ) EA	Any $n \geq k \in \Omega(n)$	$\Pr(T \leq e^{cn}) \leq e^{-\Omega(n)}$
2-tour. EA	Some $n \geq k \in \Omega(n)$	$\Pr(T \leq e^{cn}) \leq e^{-\Omega(\lambda)}$
MOSA-EA	Any $n \geq k \in \Omega(n)$	$E[T] = O(n^2 \log(n))$

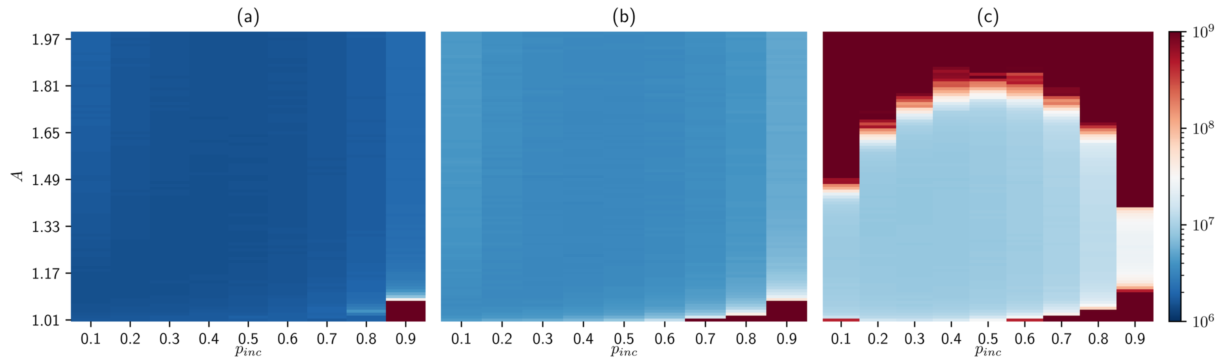
- The MOSA-EA can efficiently escape an artificial local optimum with unknown sparsity, while other fixed mutation rate EAs fail.

Proof Idea:

- The error thresholds for sparse and dense regions are different.
- The MOSA-EA maximises fitness and mutation rate on Pareto fronts.
- The mutation rates will be closed to its error threshold.
- Individuals with mutation rates larger than error thresholds will "vanish" in the next generation.
- Use the level-based theorem [9] to derive the runtime.

Hyper-parameters Do Not Need Careful Tuning [B]

- One of the aims of self-adaptation is to reduce the number of parameters that must be set by the user.
- MOSA-EA has three parameters ε, p_{inc} and A.
- Adding three new parameters to adapt one parameter seems contradictory to the aim of self-adaptation.
- However, as we see the figures to the right, these parameters need not to be tuned carefully.
- We use the same parameters setting of the MOSA-EA for all experiments in this study to show that the MOSA-EA does not require problem-specific tuning of the hyper-parameters.



Figs.1 Median runtimes of the MOSA-EA for different parameters A and p_{inc} on (a) OneMax, (b) LeadingOnes and (c) Funnel over 100 independent runs ($n = 100$).

MOSA-EA Adapts to Noise Level [B]

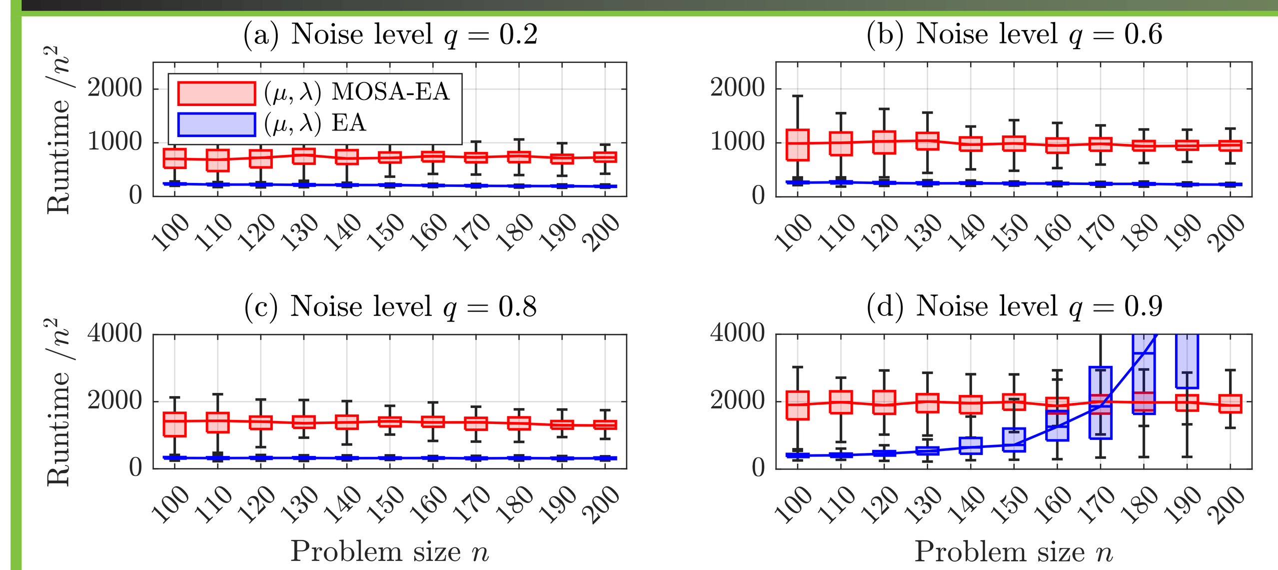


Fig.2 Runtimes of MOSA-EA and (μ, λ) EA with the fixed mutation rate $\chi/n = 1.386/n$ on LeadingOnes under one-bit noise with noise levels q.

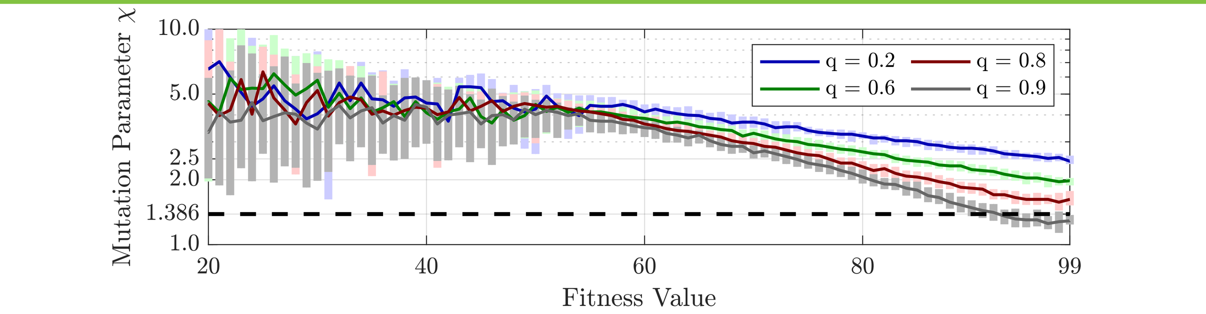


Fig.3 Real fitness and mutation parameter of the highest real fitness individual per generation of MOSA-EA on LeadingOnes under one-bit noise with noise levels q.

- Non-elitist EAs can cope with the higher levels of noise by reducing the mutation rate. However, we need to know the exact noise level to set a proper mutation rate. [6]
- For the noise model, MOSA-EA self-adapts the mutation rate to the noise level, while fixed mutation rate EA fails under some levels noise.

MOSA-EA Outperforms on Complex Problems [B]

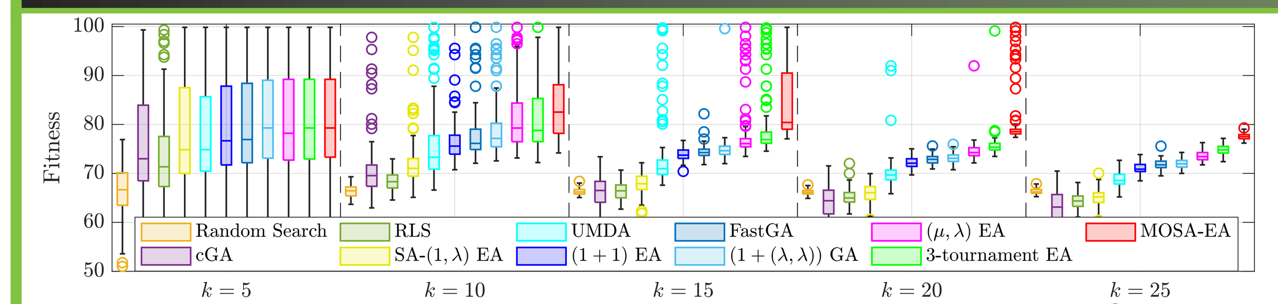


Fig.4 The highest fitness values found in the end of runs in 10^8 fitness evaluations on 100 random NK-Landscape instances ($n = 100$).

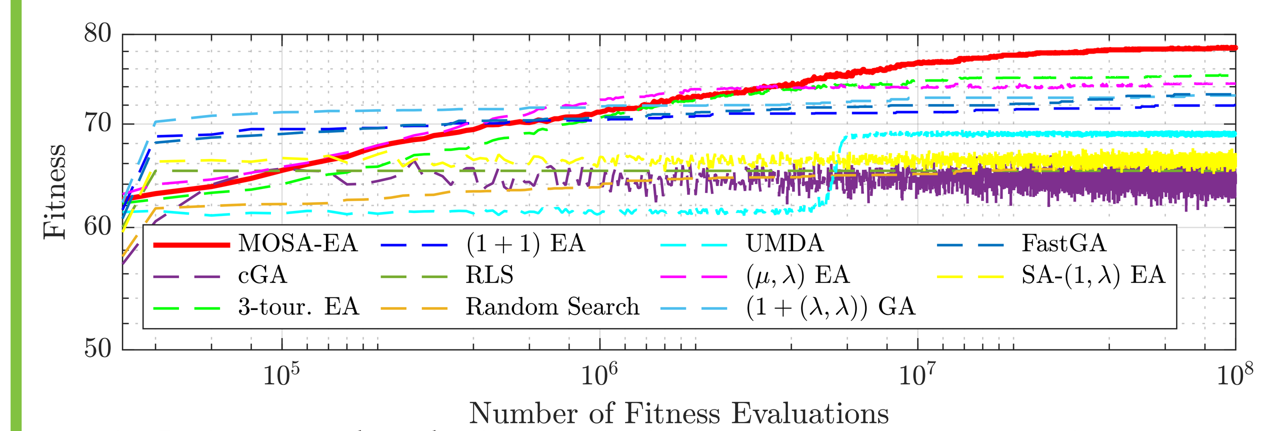


Fig.5 Fitness value during optimising a NK-Landscape instance

- The MOSA-EA increasingly outperforms the other algorithms for harder NK-Landscape and k-Sat instances.

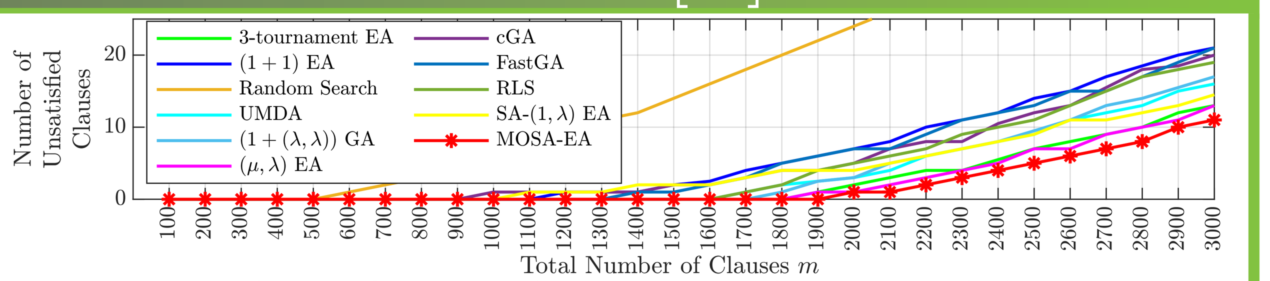


Fig.6 The medians of the smallest number of unsatisfied clauses found in 10^8 fitness evaluations on 100 random k-Sat instances with different total numbers of clauses m ($k = 5$, $n = 100$).

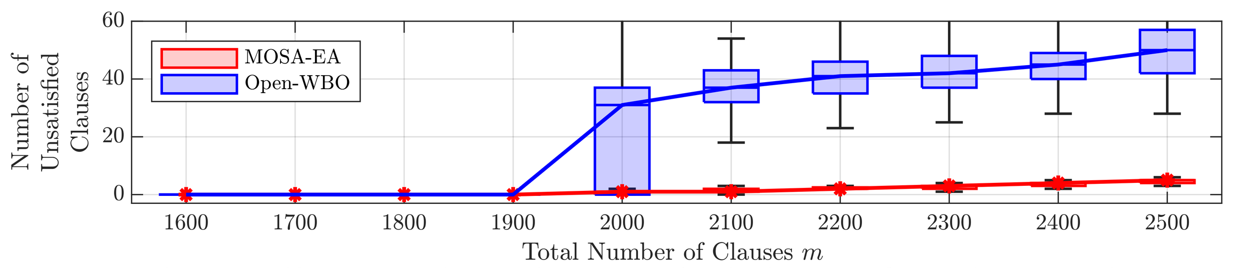


Fig.7 The smallest number of unsatisfied clauses found in one hour CPU-time on 100 random k-Sat instances with different total numbers of clauses m ($k = 5$, $n = 100$).

- In particular, the MOSA-EA outperforms the problem-specific Open-WBO algorithm, which was one of best solvers in MaxSat Evaluations in 2014, 2015, 2016, 2017.

Conclusion

- The MOSA-EA was proposed to optimise single-objective functions, treating parameter control via multi-objectivisation.
- The algorithm maximises fitness and mutation rates simultaneously.

Novelty: treats parameter control as another objective.

Significance: (1) can escape local optima with unknown sparsity. (2) self-adapt mutation rate to the noise level. (3) outperform other EAs on complex problems. (4) no need to set mutation rate manually.

Materials

Publications:

[A] Self-adaptation via Multi-objectivisation: A Theoretical Study. (with Per Kristian Lehre) In Proceedings of the Genetic and Evolutionary Computation Conference 2022 (GECCO'22).

[B] Self-adaptation via Multi-objectivisation: An Empirical Study. (with Per Kristian Lehre) To appear in the Parallel Problem Solving from Nature XVII (PPSN'22).

Code: https://github.com/ChengCheng-Qin/mosa-ea.git

References

[1] Benjamin Doerr and Carola Doerr. Theory of parameter control for discrete black-box optimization: Provable performance gains through dynamic parameter choices. *Theory of Evolutionary Computation*, pages 271–321, 2020. Publisher: Springer.

[2] A.E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, July 1999.

[3] Duc-Cuong Dang and Per Kristian Lehre. Self-adaptation of Mutation Rates in Non-elitist Populations. In *Parallel Problem Solving from Nature – PPSN XIV*, volume 9921, pages 803–813. Springer International Publishing, Cham, 2016.

[4] Benjamin Doerr, Carsten Witt, and Jing Yang. Runtime Analysis for Self-adaptive Mutation Rates. *Algorithmica*, 83(4):1012–1053, April 2021.

[5] Brendan Case and Per Kristian Lehre. Self-adaptation in non-Elitist Evolutionary Algorithms on Discrete Problems with Unknown Structure. *IEEE Transactions on Evolutionary Computation*, pages 1–1, 2020.

[6] Per Kristian Lehre and Xiaoyu Qin. More Precise Runtime Analyses of Non-elitist EAs in Uncertain Environments. In *Proceedings of the Genetic and Evolutionary Computation Conference*, page 9, Lille, France, 2021. ACM.

[7] Jens Jagerskupper and Tobias Storch. When the Plus Strategy Outperforms the Comma Strategyand When Not. In *2007 IEEE Symposium on Foundations of Computational Intelligence*, pages 25–32, Honolulu, HI, April 2007. IEEE.

[8] Duc-Cuong Dang, Anton Eremeev, and Per Kristian Lehre. Non-elitist Evolutionary Algorithms Excel in Fitness Landscapes with Sparse Deceptive Regions and Dense Valleys. In *Proceedings of the Genetic and Evolutionary Computation Conference*, Lille, France, 2021. ACM.

[9] Dogan Corus, Duc-Cuong Dang, Anton V. Eremeev, and Per Kristian Lehre. Level-Based Analysis of Genetic Algorithms and Other Search Processes. *IEEE Transactions on Evolutionary Computation*, 22(5):707–719, October 2018.