Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Improving Sampling in Evolution Strategies through Mixture-based Distributions built from Past Problem Instances

Stephen Friess, Peter Tino, Stefan Menzel, Bernhard Sendhoff, Xin Yao

2020

Preprint:

This is a post-peer-review, pre-copyedit version of an article published in Parallel Problem Solving From Nature (PPSN). The final authenticated version is available online at: https://doi.org/10.1007/978-3-030-58112-1_40



Improving Sampling in Evolution Strategies Through Mixture-Based Distributions Built from Past Problem Instances

Stephen Friess[1(✉)], Peter Tiňo[1], Stefan Menzel[2], Bernhard Sendhoff[2], and Xin Yao[1,3]

[1] CERCIA, School of Computer Science, University of Birmingham, Birmingham, UK
{shf814,p.tino,x.yao}@cs.bham.ac.uk
[2] Honda Research Institute Europe,
Carl-Legien-Str. 30, 63073 Offenbach a.M., Germany
{stefan.menzel,bernhard.sendhoff}@honda-ri.de
[3] Department of Computer Science and Engineering,
Southern University of Science and Technology, Shenzhen, China

Abstract. The notion of learning from different problem instances, although an old and known one, has in recent years regained popularity within the optimization community. Notable endeavors have been drawing inspiration from machine learning methods as a means for algorithm selection and solution transfer. However, surprisingly approaches which are centered around internal sampling models have not been revisited. Even though notable algorithms have been established in the last decades. In this work, we progress along this direction by investigating a method that allows us to learn an evolutionary search strategy reflecting rough characteristics of a fitness landscape. This latter model of a search strategy is represented through a flexible mixture-based distribution, which can subsequently be transferred and adapted for similar problems of interest. We validate this approach in two series of experiments in which we first demonstrate the efficacy of the recovered distributions and subsequently investigate the transfer with a systematic from the literature to generate benchmarking scenarios.

Keywords: Evolution strategies · Model-based optimisation · Continuous optimisation · Algorithm configuration · Transfer learning

1 Introduction

Within recent decades, the field of evolutionary computation has seen a surge of novel algorithms being proposed, frequently with the intent to operate on very specific problem domains. While this reflects on one hand the efficacy of population-based and evolutionary approaches for a wide range of applications, it also reflects deep rooted issues within the current state of the art. Particularly

© The Author(s) 2020
T. Bäck et al. (Eds.): PPSN 2020, LNCS 12269, pp. 583–596, 2020.
https://doi.org/10.1007/978-3-030-58112-1_40

in regards to: 1) A lack of a prescriptive theory on how to construct efficient algorithms for a given problem and 2) a lack of understanding on what constitutes and characterizes optimisation problems and the similarity thereof in a more generalized way. While the theorists cannot give definite answer to both questions at the moment, one may still legitimately ask whether or not it is possible to approach some of these problems from a pragmatic line of attack. For this reason, two popular trends have emerged within the optimisation community: 1) Research on meta-learning frameworks [13,18,23] and 2) research on transfer learning approaches [4,9,12,14,17]. Both try to boost the efficiency of optimisation algorithms by using prior knowledge from solving problem instances.

In our work, we progress at the intersection of both lines of research by building a model of a search strategy from individual runs which may be then subsequently transferred to similar problem instances. For this reason, we first give in Sect. 2 a brief overview discussing these two existing lines of research and give insight into the state-of-the-art. Section 3 explains the extensions we introduce to consolidate a search strategy. Further, we demonstrate its functionality on an illustrative benchmark function. In Sect. 4.1, we widen the range of considered benchmark problems to a selected variety of multimodal and valley-shaped problems. Subsequently, in Sect. 4.2 we consider the scenario of transferring search strategies across problem instances generated by translations, rotation and various non-linear transformations to the benchmark functions. We conclude our study with a summary in Sect. 5 and give an outlook on future work.

2 Knowledge from Problem Solving Exercises

In principle, within the optimisation community two approaches have been investigated within the recent decades. The first one being related to the construction of meta-learning frameworks for algorithm selection and configuration. The second one relating to instance-based transfer learning through candidate solutions. Both fields, while having gained strong traction with the recent years, can trace their origin back to much earlier roots. With seminal work on algorithm selection being done by Rice et al. [21] in the 1970s and research on transfer learning emerging from the discourse on lifelong machine learning systems in the 1990s [19]. However, their application towards the domain of optimisation has been only considered since recently within the 2000s [17,23].

Meta-learning frameworks attempt to harness high-level knowledge that can be subsequently used in the future to more efficiently solve related tasks. In the classical algorithm selection and algorithm configuration problem, this would equate to predicting the best performing algorithm or configuration for a given problem [13,18]. However, a key problem in optimisation lies in the first place in the extraction and computation of said task specific features. This poses especially an outstanding problem within the domain of continuous optimisation, where unlike in the combinatorial domain, problem features cannot be simply derived from the problem state or definition. Features thus have to be explicitly computed in a cheap and at best informative manner.

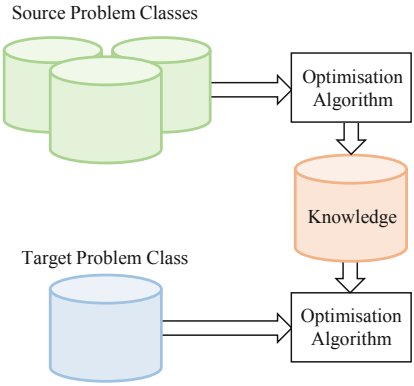

Fig. 1. Diagram of the archetypical pipeline for transfer learning. Roughly adapted from Pan et al. [19]. Similar setups are frequently encountered within literature on population-based optimisation (e.g. [4,5,14]). From previously solved source problem classes knowledge is extracted by the algorithm such that it can subsequently improve the performance on it on a new target problem class.

Transfer learning approaches on the other side may be seen as operating under more relaxed conditions. Essentially, what transfer learning assumes between two problem instances, is that beneficial knowledge which helped solving one problem instance, can be transferred either directly or by means of a transformation to a new problem instance. However, notably it introduces by this further uncertainties. The bulk of transfer learning literature in optimisation draws inspiration from instance-based transfer [19] by means of transferring high performing candidate solutions between tasks (e.g. [5,12,14,22]). Retrieval of the candidate solutions occurs either directly from the previous solved tasks [5,14] or through probabilistic sampling from a continuously built repository (e.g. [4,17]). As a way of determining the probabilistic weights, often times task similarity measures may be used [17]. However, in many scenarios instead simply solution similarity may be used as a proxy of task similarity [4,17]. In general, the lack of satisfying task similarity measures together with being prone to uncontrolled 'negative' knowledge transfer which degrades algorithm performance [5] are known problems of these approaches.

Interestingly, aside from these mentioned works, barely any of the recent literature tries to learn across problem instances explicitly by means of internal sampling models. Although quite notably, many popular algorithms rely upon operators drawing random variables from symmetrical distributions and thus have by default isotropy assumptions built in. However, this assumption becomes broken when given an optimisation problem which does not resemble a flat plane. Quite intuitively, the interplay between algorithm and optimisation problem should enforce characteristic search strategies and behaviors. Modern model-based algorithms [10,15] acknowledge this by adapting a distribution online during the optimisation run. However, they do not attempt to

memorize these in a more rough and abstract way, such that these can be transferred across problem instances. In many ways, this perspective might be also the only meaningful notion to realize transfer learning in continuous single-objective optimisation. In the following, we build up on our previous work [7,8] and try tackle the issue in a study using a variant of the popular (μ, λ)-Evolution Strategy for continuous optimisation. We explicitly incorporate strategy parameters through a windowing approach and harness systematics from the literature to build benchmarking scenarios.

3 Extending the Evolution Strategy

In the following, we consider continuous single-objective optimisation problems of the form $f : \chi \subseteq \mathbb{R}^d \rightarrow \mathbb{R}$, where χ denotes the search space and d its associated dimensionality. As a base we use a variant of the Evolution Strategy with (μ, λ) selection mechanism [1]. We keep out explicitly any recombination operators to have the framework reduced to its essentials. Meaning to sample mutations from a multivariate distribution and performing selection in an elitist manner. Note, that from an evolutionary perspective, mutation is the principle source of variation [16]. In many ways, this basic outline may resemble continuous variants of Evolutionary Programming. However, the elitist selection mechanism in Evolution Strategies has been implicated to contribute to performance improvements [2].

In the Evolution Strategy, population members $\mathbf{s}(\mathrm{j})$ are represented by tuples $\mathbf{s}(j) = [\mathbf{x}(j), \boldsymbol{\sigma}(j)]$, where $\mathbf{x}(j) = (x_1(j), \cdots, x_d(j))$ is the population member's representation in the solution space and $\boldsymbol{\sigma}(j) = (\sigma_1(j), \cdots, \sigma_d(j))$ are its strategy parameters. The latter can be considered to be a key feature of Evolution Strategy implementations. Strategy parameters essentially control the shape the normal distribution from which mutations

$$\Delta\mathbf{x}(j) \sim \mathcal{N}(\mathbf{0}, \mathrm{diag}[\boldsymbol{\sigma}(j)]) \tag{1}$$

for the individuals j are drawn which shift the individuals $\mathbf{x}'(j) = \mathbf{x}(j) + \Delta\mathbf{x}(j)$ in the solution space. Likewise, variation operators can be defined such that they also vary and recombine the strategy parameters of population members. However, we neglect this extension within our study.

3.1 Quality-Based Filtering of Mutations

In the following, we will further filter performed mutations according to their quality. Thus, we will distinguish between beneficial mutations as defined by

$$f(\mathbf{x}(j)^i_{before}) - f(\mathbf{x}(j)^i_{after}) \geq 0 \tag{2}$$

and detrimental mutations defined by

$$f(\mathbf{x}(j)^i_{before}) - f(\mathbf{x}(j)^i_{after}) < 0. \tag{3}$$

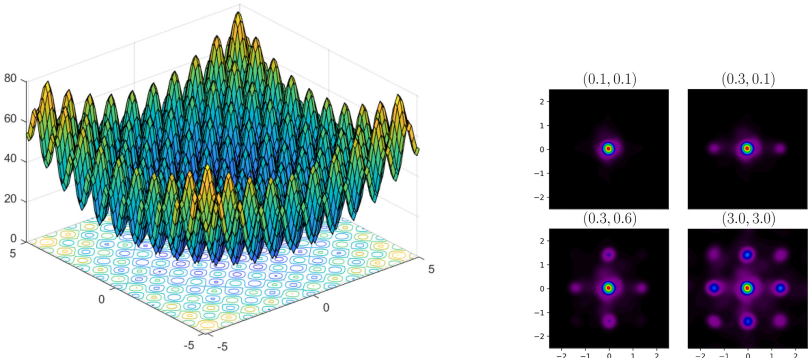

Fig. 2. Left panel: Rastrigin's benchmark function. Right panel: Search distributions for different pairs of strategy parameters $\boldsymbol{\sigma} = (\sigma_x, \sigma_y)$ derived from a 100 component mixture model of the distribution of beneficial mutations from 1000 runs under reweighing according to Eq. (4) & (5).

The idea is, that once we have stored statistics about mutations outside of the algorithm, we can use them to design improved search strategies. Specifically, by means of constructing empirical distributions which serve as basis for model-based mutation operators. These can be seen as reflecting globally averaged characteristics of the fitness landscape. In principle, one would intuitively be interested into enforcing beneficial mutations and suppressing detrimental mutations. However, distributions of detrimental mutations have been implicated to be strongly normal distributed [8]. It is also questionable from the perspective of algorithm design whether suppressing mutations comes at the expense of convergence properties, as every point in the search space should remain reachable by a small finite amount of probability. Thus, we focus in the following only on biasing the algorithm through distributions of beneficial mutations (Fig. 2).

3.2 Constructing Operators from Empirical Distributions

Choosing a Density Estimator. While by default, mutations are sampled in the Evolution Strategy from a multivariate normal distribution as given in Eq. (1), for empirical distributions one explicitly has to use a modeling technique. In principle, many techniques are available for this purpose. However, in the following we will use the Gaussian mixture model as it is a well-studied model which can act as universal density approximator. Mixture models reduce the input data to a small set of descriptive clusters which are parametrized by multivariate normal distributions, such that the full data distribution can then be expressed as $p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, with mixture coefficients π_k, which are normalized such that $\Sigma_{k=1}^{K} \pi_k = 1$, and determined together with means μ_k and covariances Σ_k by maximizing the log-likelihood through the expectation-maximization algorithm [3,20].

Incorporating Strategy Parameters. However, an outstanding problem still lies in the fact that the Evolution Strategy possesses strategy parameters $\boldsymbol{\sigma}$ which control the shape of the distribution from which mutations are sampled. Changing the shape of an empirical distribution as basis for improved sampling should not break the contained spatial information. Therefore, we simply window the empirical distribution with the multivariate normal distribution spanned by the strategy parameters as defined by Eq. (1). Effectively, this results in a reweighing of the mixture model where we replace the original mixture coefficients π_k with

$$r_k = \frac{\pi_k c_k}{\sum_{i=1}^{N} \pi_i c_i}, \tag{4}$$

where the coefficients c_k per mixture component quantify the average value of the normal distribution spanned by the strategy parameters over the k-th mixture component. This can be analytically calculated such that

$$\begin{aligned} c_k &:= \int_{\mathbb{R}^n} \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \Sigma_k)\, \mathcal{N}(\boldsymbol{x}|\mathbf{0}, \mathrm{diag}(\boldsymbol{\sigma}))\, \mathrm{d}^n\mathbf{x} \\ &= \int_{\mathbb{R}^n} \frac{\exp\left[-\frac{1}{2}\,(\mathbf{x}-\boldsymbol{\mu}_k)^T\, \Sigma_k^{-1}\,(\mathbf{x}-\boldsymbol{\mu}_k)\right]}{\sqrt{(2\pi)^d |\Sigma_k|}} \times \frac{\exp\left[-\frac{1}{2}\,\mathbf{x}^T\, \Sigma_{\boldsymbol{\sigma}}^{-1}\,\mathbf{x}\right]}{\sqrt{(2\pi)^d |\Sigma_{\boldsymbol{\sigma}}|}}\, \mathrm{d}^n\mathbf{x} \\ &= \frac{\exp\left(-\frac{1}{2}\,\boldsymbol{\mu}_k^T\, \Sigma_k^{-1} \boldsymbol{\mu}_k + \frac{1}{2}\,\boldsymbol{\mu}_k^T \left[\Sigma_k^{-1}\; \Sigma_c\;\; \Sigma_k^{-1}\right] \boldsymbol{\mu}_k\right)}{\sqrt{(2\pi)^d |\Sigma_k||\Sigma_c^{-1}||\Sigma_{\boldsymbol{\sigma}}|}} \times \int_{\mathbb{R}^n} \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_c, \Sigma_c)\, \mathrm{d}^n\mathbf{x} \\ &= \frac{\exp\left(-\frac{1}{2}\,\boldsymbol{\mu}_k^T \left[\Sigma_k^{-1}\,(\Sigma_k^{-1} + \Sigma_{\boldsymbol{\sigma}}^{-1})^{-1}\; \Sigma_{\boldsymbol{\sigma}}^{-1}\right] \boldsymbol{\mu}_k\right)}{\sqrt{(2\pi)^d |\Sigma_k||\Sigma_k^{-1} + \Sigma_{\boldsymbol{\sigma}}^{-1}||\Sigma_{\boldsymbol{\sigma}}|}}, \end{aligned} \tag{5}$$

where we further introduced $\Sigma_{\boldsymbol{\sigma}} := \mathrm{diag}(\boldsymbol{\sigma})$ and $\Sigma_c := (\Sigma_k^{-1} + \Sigma_{\boldsymbol{\sigma}}^{-1})^{-1}$.

Table 1. Benchmark functions used in this study, grouped from top to bottom according to landscape structure. 1st–3rd row: Unimodal and valley shaped problems. 4th–6th row: Multimodal problems with single global optimum and strong regularity. 7th–9th row: Difficult multimodal problems with single global optimum and high irregularity.

Name	Search Space	Function Definition
Sphere	$[-5.12, 5.12]^d$	$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2$
Bohachevsky	$[-100, 100]^d$	$f(\mathbf{x}) = \sum_{i=1}^{d-1} [x_i^2 + 2x_{i+1}^2 - 0.3\cos(3\pi x_i) - 0.4\cos(4\pi x_{i+1}) + 0.7]$
Rosenbrock	$[-5, 10]^d$	$f(\mathbf{x}) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$
Rastrigin	$[-5.12, 5.12]^d$	$f(\mathbf{x}) = 10\,d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$
Ackley	$[-32.768, 32.768]^d$	$f(\mathbf{x}) = -a\,\mathrm{e}^{0.2(\frac{1}{d}\sum_{i=0}^{d} x_i^2)^{0.5}} + \mathrm{e}^{-\frac{1}{d}\sum_{i=0}^{d}\cos(2\pi x_i)} + a + \mathrm{e}$
Griewank	$[-600, 600]^d$	$f(\mathbf{x}) = 1 + \frac{1}{4000}\sum_{i=1}^{d} x_i^2 + \frac{1}{4000}\prod_{i=1}^{d}\cos(x_i/\sqrt{i})$
Schwefel	$[-500, 500]^d$	$f(\mathbf{x}) = 418.9829d - \sum_{i=1}^{d-1} x_i \sin(\sqrt{\lvert x_i \rvert})$
Eggholder	$[-512, 512]^d$	$f(\mathbf{x}) = -(x_2+47)\sin(\sqrt{\lvert x_2 + x_1/2 + 47 \rvert}) - x_1\sin(\sqrt{\lvert x_1 - (x_2+47) \rvert})$
Rana	$[-512, 512]^d$	$f(\mathbf{x}) = x_1\sin(\sqrt{\lvert x_2+1-x_1 \rvert})\cos(\sqrt{\lvert x_1+x_2+1 \rvert}) + (x_2+1)\cos(\sqrt{\lvert x_2+1-x_1 \rvert})\sin(\sqrt{\lvert x_1+x_2+1 \rvert})$

4 Experimental Study

The following study is based upon the DEAP library for Evolutionary Computation [6] with the extensions as elaborated in Sect. 3. We first investigate in Sect. 4.1 whether distributions of beneficial mutations can be harnessed at all to realize performance improvements on a selected range of different continuous optimisation problems. Subsequently in Sect. 4.2 we investigate different transfer scenarios between problem instances. Particularly, we build these scenarios by harnessing existing systematics from the literature.

4.1 On the Efficacy of Distributions of Beneficial Mutations

In the following we conduct experiments over a range of 9 different optimisation problems listed in Table 1. We group these into unimodal and valley-shaped problems (1st–3rd row), multimodal problems with single global optimum and high regularity (4th–6th row) and multimodal problems with single global optimum and high irregularity (7th–9th row). All experiments are conducted with a population size of $\mu = 10$ and we generate at each generation $\lambda = 30$ offspring members by randomly selecting individuals and either cloning or mutating them with a 30% chance. In all experiments, the population is initialized randomly upon the entire search space, where we use additionally a penalization for the difficult multimodal problems by means of rejecting mutations crossing the search space boundaries. This is necessary, as otherwise in these problems lower optima could be reached in the outer areas. Strategy parameters are initialized such that $\sigma \in [0.1, 4.0]$ for the problems in row 1–6 in Table 1. For the difficult multimodal functions we re-adjust the upper boundaries, where we use for Schwefel's function $\sigma_{\max} = 400$, for Eggholder $\sigma_{\max} = 480$ and for Rana's function $\sigma_{\max} = 150$. We will elaborate further in the succeeding paragraph on the necessity of the re-adjustment. Experiments are conducted over 1000 generations and we accumulate data per experiment from 100 runs. Problem dimension is kept at $d = 2$ in all experiments, as this still allows the interpretation of the retrieved distributions and lifts problems of data sparsity arising with more degrees of freedom. The mixture model is constructed with a total number of $K = 50$ components.

Resulting minimum fitness curves per generation of the optimisation runs are plotted per problem group in Figs. 3 and 4. Where top rows are the runs using default mutation distributions, and the lower rows are runs which use distributions of beneficial mutations with and without considering strategy parameters. Further, median (dark blue), mean (grey) and individual runs (light blue) are plotted. Quite notably, across all considered problems the distribution of beneficial mutations significantly improves the search behavior. Particularly, it reduces late convergences by acting in a regularizing fashion. However, the inclusion of strategy parameters is only helpful when some regularity along the parameter axis can be harnessed. Otherwise, it's effect on the performance is detrimental. The approach can even be shown to work on the difficult multimodal functions of row 7–9 in Table 1. However, we openly admit that further precautions have to be taken for these experiments to work. In particular, for all three we had

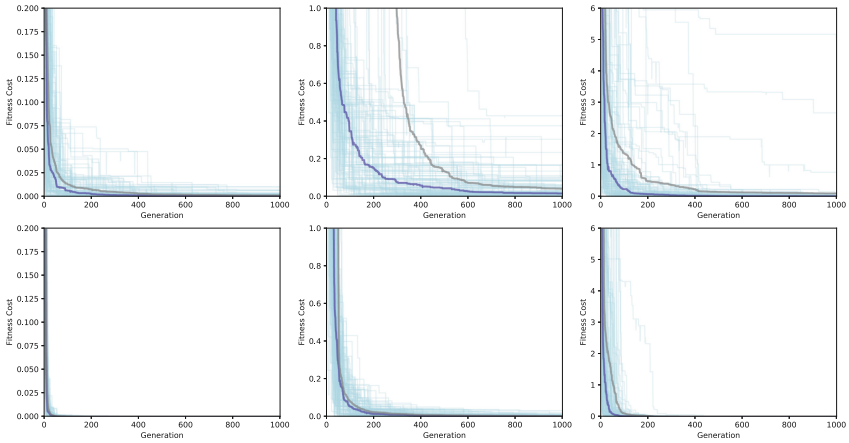

Fig. 3. Column 1–3: Fitness curves (light blue) for the unimodal Sphere, Bohachevsky's and Rosenbrock's function from 100 runs, as well as median (dark blue) and mean (dark grey) curves. Top row: With default sampling. Bottom row: With improved sampling using quality-based mutations. (Color figure online)

to re-adjust the upper bound of the strategy parameter to the previously mentioned values such that we achieved good convergence behavior in the runs with default sampling. Without taking these precautions, we were not able to achieve any improvements using the distribution of beneficial mutations. In fact, for the lower values of the strategy parameters we even found that the distributions of beneficial mutations were detrimental to the optimisation and encouraged premature convergence into local optima. We further list performance values of our experiments, as well as results from a statistical Wilcoxon rank sum test under normal approximation in Table 2. The results indicate that for a significance level of $\alpha = 0.05$, the null hypothesis can be rejected in all experiments.

4.2 Cross-Instance Transfer Scenarios

In the following section we will consider now cross-instance transfer learning scenarios. Meaning we try to transfer a mutation operator learned on a source optimisation problem to a target problem (c.f. Fig. 1) in the hope of realizing performance improvements. To generate variations of the source problem instances we apply in the following a systematic of transformations proposed by Hansen et al. [11].

Transformations of the Fitness Landscape. The following base transformations are designed to explicitly break the well-behavedness of our optimisation problems by acting upon the decision variables $\mathbf{x}$. *Ill-conditioning* introduces fast running components by a means of a linear rescaling

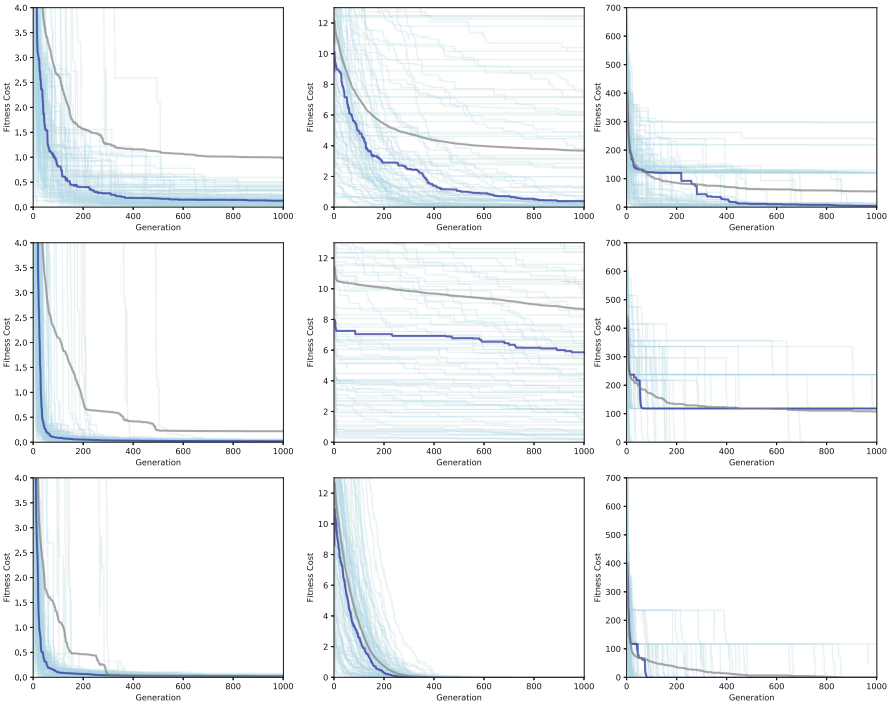

Fig. 4. Column 1–3: Fitness curves (light blue) for the multimodal Rastrigin's, Ackley's and Griewank's function from 100 runs, as well as median (dark blue) and mean (dark grey) curves. Top row: With default sampling. Middle row: With improved sampling considering strategy parameters. Middle row: With improved sampling considering strategy parameters. Bottom row: With improved sampling using quality-based mutations. (Color figure online)

$$T_{ill\text{-}c.} : \mathbb{R}^d \to \mathbb{R}^d, \;\; x_i \longmapsto x_i \; \alpha^{\frac{1}{2}\frac{i-1}{d-1}}, \tag{6}$$

where we choose $\alpha = 10$ in our experiments. The *asymmetrical* transformation breaks the symmetry of components x_i under sign transformations with

$$T_{asy} : \mathbb{R}^d \to \mathbb{R}^d, x_i \longmapsto \begin{cases} x_i^{1+\beta\frac{i-1}{d-1}\sqrt{x_i}} & \text{if } x_i > 0 \\ x_i & \text{otherwise} \end{cases}, \tag{7}$$

such that in the positive quadrant the components scale up exponentially. The *oscillatory* transformation introduces sinusoidal variability of the components by

$$T_{osc} : \mathbb{R}^d \to \mathbb{R}^d, x_i \longmapsto \text{sign}(x_i) \; \exp(\hat{x}_i + 0.049(\sin(c_1\hat{x}_i) + \sin(c_2\hat{x}_i))), \tag{8}$$

$$\hat{x} \longmapsto \begin{cases} \log(|x|) & \text{if } x \neq 0 \\ 0 & \text{otherwise} \end{cases}, \hat{c_1} \longmapsto \begin{cases} 10 & \text{if } x \neq 0 \\ 5.5 & \text{otherwise} \end{cases}, \hat{c_2} \longmapsto \begin{cases} 7.9 & \text{if } x \neq 0 \\ 3.1 & \text{otherwise} \end{cases}$$

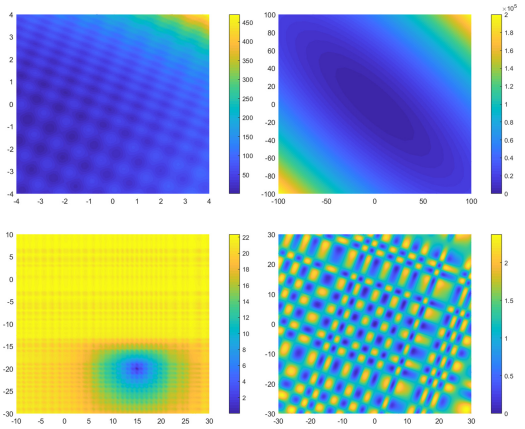

Fig. 5. From the top left corner clockwise: Altered variants of Rastrigin's (R_1), Sphere (S_1), Griewank's (G_1) and Ackley's (A_1) function.

Further, we also use counter-clockwise rotations $T_{rot}(\theta)$ by angle θ and translations T_{trans} of the global optimum.

Experimental Validation. We investigate the utility of the transformations in a set of 9 experiments with 4 transformed standard problems. Further, the transfer from source problem to target problem $P_0 \rightarrow P_1$, and likewise the transfer into the reverse direction $P_0 \rightarrow P_1$. We use in the following the sphere function S_1 with ill-conditioning, 45° rotation and extended search space to $[-100, 100]^2$, the Ackley's function A_1 with a translation of $\mathbf{t} = (-15, 20)$ and subsequently added oscillations and asymmetries, Rastrigin's function R_1 with 22.5° rotation,

Table 2. Medians $\tilde{f}_{\min}$, means $\overline{f}_{\min}$ and standard deviations $\sigma_{\min}$ of the minimum fitness after 1000 generations aggregated from 100 runs for default sampling using a normal distribution ($\mathcal{N}$) and improved sampling using a mixture model of quality-filtered mutations($\mathcal{M}$). Further, normalized ranks z and p-values for a two-tailed Wilcoxon rank sum test have been calculated. For a significance level of $\alpha = 0.05$ the null hypothesis can be considered to be rejected in all experiments.

Benchmark	$\tilde{f}_{\min}$ ($\mathcal{N}$)	$\overline{f}_{\min}$ ($\mathcal{N}$)	$\sigma_{\min}$ ($\mathcal{N}$)	$\tilde{f}_{\min}$ ($\mathcal{M}$)	$\overline{f}_{\min}$ ($\mathcal{M}$)	$\sigma_{\min}$ ($\mathcal{M}$)	$\lvert z \rvert$	p-value
Sphere	5.528e−4	1.303e−3	1.975e−3	2.360e−5	3.600e−5	3.630e−5	9.713e+0	2.670e−22
Bohachevsky	1.668e−2	4.888e−2	8.349e−2	2.754e−3	4.580e−3	5.222e−3	8.107e+0	5.180e−16
Rosenbrock	9.098e−3	1.013e−1	5.641e−1	6.096e−4	9.370e−4	1.081e−3	1.035e+1	4.390e−25
Rastrigin	1.924e−1	5.067e−1	7.974e−1	6.609e−3	9.327e−3	9.066e−3	1.133e+1	9.060e−30
Ackley	1.223e−1	9.722e−1	3.221e+0	2.957e−2	3.297e−2	1.575e−2	9.287e+0	1.580e−20
Griewank	3.686e−1	3.653e+0	5.780e+0	1.011e−4	1.747e−3	3.744e−3	1.172e+0	1.050e−31
Schwefel	3.523e+0	5.413e+1	7.509e+1	2.595e−3	1.189e+0	1.178e+1	1.201e+1	3.270e−33
Eggholder	4.959e+0	6.135e+1	7.786e+1	3.730e−5	1.131e+1	3.494e+1	9.935e+0	2.940e−23
Rana	1.652e+0	1.832e+1	2.939e+1	1.650e+0	4.359e+0	6.102e+0	2.622e+0	8.748e−3

Table 3. Medians $\tilde{f}_{\min}$, means $\overline{f}_{\min}$ and standard deviations $\sigma_{\min}$ of the minimum fitness after 1000 generations aggregated from 100 runs for default sampling (upper table) and transfer scenarios (bottom table). Further, normalized ranks z and p-values for a two-tailed Wilcoxon rank sum test are given. For a significance level of $\alpha = 0.05$ the null hypothesis can be considered to be rejected in all experiments.

Scenarios	$\tilde{f}_{\min}$	$\overline{f}_{\min}$	$\sigma_{\min}$	$\lvert z \rvert$	p-value
S_0^*	8.319e−4	1.443e−3	1.854e−3	–	–
R_0^*	2.173e−1	9.599e+0	4.118e+1	–	–
S_1	2.623e−3	1.859e−1	1.781e+0	–	–
A_1	1.639e−1	3.101e+0	6.812e+0	–	–
R_1	1.813e−1	6.901e−1	1.438e+0	–	–
G_1	3.298e+0	7.139e+0	8.870e+0	–	–
$S_0^* \to S_1$	5.786e−4	7.114e−4	7.064e−4	7.281e+0	3.306e−13
$S_1 \to S_0^*$	1.899e−4	2.299e−4	1.751e−4	6.744e+0	1.544e−11
$A_0 \to A_1$	2.839e−2	1.827e+0	5.716e+0	8.721e+0	2.771e−18
$A_1 \to A_0$	3.301e−2	3.774e−2	2.379e−2	9.928e+0	3.161e−23
$R_0 \to R_1$	5.689e−2	8.816e−2	8.894e−2	6.304e+0	2.902e−10
$R_1 \to R_0$	6.984e−2	1.051e−1	1.111e−1	5.603e+0	2.111e−8
$G_0 \to G_1$	3.034e−1	1.305e+0	4.749e+0	6.842e+0	7.837e−12
$G_1 \to G_0$	1.261e−1	1.773e+0	1.793e−1	3.531e+0	4.145e−4
$S_0^* \to G_0$	3.531e+0	5.110e+0	4.490e+0	3.592e+0	3.284e−4
$G_0 \to S_0^*$	6.850e−5	1.386e−4	1.488e−4	3.722e+0	1.974e−4

small shift $\mathbf{t} = (3, 2)$, extended search space to $[-100, 100]^2$ and added asymmetry, as well as Griewanks function G_1 with 20° rotation and added oscillations. Further, we denote the Sphere and Rastrigin's function with extended search spaces to $[-100, 100]^2$ as S_0^* and R_0^*. Heightmaps of most altered benchmark problems are plotted in Fig. 5. We find that in most considered transfer scenarios, performance improvements can be realized (Table 3). However, finding difficult and interesting scenarios without making them obvious is a bit of a hurdle. For example, in our experiments the scenario $S_0^* \to G_0$ features negative transfer, as the transferred distribution is simply adapted for a unimodal fitness landscape with small search space.

5 Conclusions

We have investigated in this paper an approach which allows us to learn an evolutionary search strategy reflecting rough and globally averaged characteristics of a fitness landscape. We represented this search strategy through flexible mixture-based distributions of beneficial mutations as basis for improved operators. Particularly, these distributions can be considered to be improved as they

enable us to lift the isotropy assumption usually built into mutation operators, thus ingrain the problem structure and redistribute probability weight radially to more appropriately balance exploration and exploitation on a given problem instance. The distribution can be further adapted through a Gaussian reweighing approach, thus emulating the role strategy parameters have for sampling with a default normal distribution. However, this only seems to be useful on a limited range of scenarios. We showed that unweighted distributions can indeed lead to performance improvements on a large variety problems, however prior good convergence properties of the default sampling approach seems to be an essential prerequisite. Further, we investigated systemically built transfer scenarios and could also realize performance improvements in these. However, we openly acknowledge the difficulty of finding meaningful and difficult transfer scenarios. Part of the problem stems from the fact, as it is unsure to which degree one can alter or change a problem such that it still may be attributed to be an instance of the former. However, introducing and investigating systematic transformations should be one the first key steps towards to resolving the issue. For the future, we plan to investigate the proposed framework in higher dimensions for improved transfer scenarios, as well as look into measures of problem similarity potentially by means of fitness landscape analysis.

Acknowledgements. This research has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement number 766186. It was also supported by the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (Grant No. 2017ZT07X386), Shenzhen Science and Technology Program (Grant No. KQTD2016112514355531), and the Program for University Key Laboratory of Guangdong Province (Grant No. 2017KSYS008).

References

1. Bäck, T.: Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming. Genetic Algorithms. Oxford University Press, Oxford (1996)
2. Bäck, T., Rudolph, G., Schwefel, H.P.: Evolutionary programming and evolution strategies: similarities and differences. In: Proceedings of the Second Annual Conference on Evolutionary Programming. Citeseer (1993)
3. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, Heidelberg (2006)
4. Da, B., Gupta, A., Ong, Y.: Curbing negative influences online for seamless transfer evolutionary optimization. IEEE Trans. Cybern. **99**, 1–14 (2018). https://doi.org/10.1109/TCYB.2018.2864345
5. Feng, L., Ong, Y.S., Jiang, S., Gupta, A.: Autoencoding evolutionary search with learning across heterogeneous problems. IEEE Trans. Evol. Comput. **21**(5), 760–772 (2017)
6. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., Gagné, C.: DEAP: evolutionary algorithms made easy. J. Mach. Learn. Res. **13**, 2171–2175 (2012)
7. Friess, S., Tiňo, P., Menzel, S., Sendhoff, B., Yao, X.: Representing experience through problem-tailored search operators. In: 2020 IEEE World Congress on Computational Intelligence (2020)

8. Friess, S., Tiňo, P., Menzel, S., Sendhoff, B., Yao, X.: Learning transferable variation operators in a continuous genetic algorithm. In: 2019 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2027–2033, December 2019. https://doi.org/10.1109/SSCI44817.2019.9002976
9. Gupta, A., Ong, Y.S., Feng, L.: Insights on transfer optimization: because experience is the best teacher. IEEE Trans. Emerg. Top. Comput. Intell. **2**(1), 51–64 (2018)
10. Hansen, N.: The CMA evolution strategy: a comparing review. In: Lozano, J.A., P. Larrañaga, P., Inza I., Bengoetxea E. (eds.) Towards a New Evolutionary Computation, pp. 75–102. Springer (2006). https://doi.org/10.1007/3-540-32494-1_4
11. Hansen, N., Finck, S., Ros, R., Auger, A.: Real-parameter black-box optimization benchmarking 2009: Noiseless functions definitions (2009)
12. Jiang, M., Huang, Z., Qiu, L., Huang, W., Yen, G.G.: Transfer learning-based dynamic multiobjective optimization algorithms. IEEE Trans. Evol. Comput. **22**(4), 501–514 (2017)
13. Kerschke, P., Hoos, H.H., Neumann, F., Trautmann, H.: Automated algorithm selection: survey and perspectives. Evol. Comput. **27**(1), 3–45 (2019)
14. Koçer, B., Arslan, A.: Genetic transfer learning. Exp. Syst. Appl **37**(10), 6997–7002 (2010)
15. Bengoetxea, E., Larrañaga, P., Bloch, I., Perchant, A.: Estimation of distribution algorithms: a new evolutionary computation approach for graph matching problems. In: Figueiredo, M., Zerubia, J., Jain, A.K. (eds.) EMMCVPR 2001. LNCS, vol. 2134, pp. 454–469. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44745-8_30
16. Losos, J.B.: The Princeton Guide to Evolution. Princeton University Press, Princeton (2017)
17. Louis, S.J., McDonnell, J.: Learning with case-injected genetic algorithms. IEEE Trans. Evol. Comput. **8**(4), 316–328 (2004)
18. Muñoz, M.A., Sun, Y., Kirley, M., Halgamuge, S.K.: Algorithm selection for black-box continuous optimization problems: a survey on methods and challenges. Inf. Sci. **317**, 224–245 (2015)
19. Pan, S.J., Yang, Q., et al.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
20. Pedregosa, F., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**(Oct), 2825–2830 (2011)
21. Rice, J.R., et al.: The algorithm selection problem. Adv. Comput. **15**(65–118), 5 (1976)
22. Ruan, G., Minku, L.L., Menzel, S., Sendhoff, B., Yao, X.: When and how to transfer knowledge in dynamic multi-objective optimization. In: 2019 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 2034–2041. IEEE (2019)
23. Smith-Miles, K.A.: Cross-disciplinary perspectives on meta-learning for algorithm selection. ACM Comput. Surv. (CSUR) **41**(1), 6 (2009)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

