Discovering Evolution Strategies via Meta-Black-Box Optimization

Robert Tjarko Lange  
Technical University Berlin

Tom Schaul, Yutian Chen  
DeepMind

Tom Zahavy, Valentin Dalibard  
DeepMind

Chris Lu  
University of Oxford

Satinder Singh  
DeepMind

Sebastian Flennerhag  
DeepMind

Figure 1: Discovering attention-based Learned Evolution Strategies (LES) via MetaBBO. At each meta-iteration one samples a set of inner loop tasks and a set of candidate LES parameters from a meta-evolutionary optimizer (EO). Afterwards, we run an inner loop search, compute normalized meta-fitness scores across the tasks and update the meta-EO. The resulting LES generalizes far beyond its meta-training setting and beats competitive ES baselines on challenging continuous control tasks.

ABSTRACT

Optimizing functions without access to gradients is the remit of black-box methods such as evolution strategies. While highly general, their learning dynamics are often times heuristic and inflexible — exactly the limitations that meta-learning can address. Hence, we propose to discover effective update rules for evolution strategies via meta-learning. Concretely, our approach employs a search strategy parametrized by a self-attention-based architecture, which guarantees the update rule is invariant to the ordering of the candidate solutions. We show that meta-evolving this system on a small set of representative low-dimensional analytic optimization problems is sufficient to discover new evolution strategies capable of generalizing to unseen optimization problems, population sizes and optimization horizons. Furthermore, the same learned evolution strategy can outperform established neuroevolution baselines on supervised and continuous control tasks. As additional contributions, we ablate the individual neural network components of our method; reverse engineer the learned strategy into an explicit heuristic form, which remains highly competitive; and show that it is possible to self-referentially train an evolution strategy from scratch, with the learned update rule used to drive the outer meta-learning loop.

CCS CONCEPTS

• Computing methodologies → Evolution strategies; Machine learning approaches;

KEYWORDS

evolution strategies, machine learning, meta-learning

ACM Reference Format:
Robert Tjarko Lange, Tom Schaul, Yutian Chen, Tom Zahavy, Valentin Dalibard, Chris Lu, Satinder Singh, and Sebastian Flennerhag. 2023. Discovering Evolution Strategies via Meta-Black-Box Optimization. In Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3583133.3595822
1 INTRODUCTION

Black-box optimization (BBO) methods are those general enough for the optimization of functions without access to gradient evaluations. Evolution Strategies (ES) are a class of BBO that iteratively refines the sufficient statistics of a (typically Gaussian) sampling distribution, based on the function evaluations (or fitness) of sampled candidates (population members). Their update rule is traditionally formalized by equations based on first principles, but the resulting specification is inflexible. On the other hand, the evolutionary algorithms community has proposed numerous variants of BBO, derived from very different metaphors, some of which have been shown to be equivalent. One way to attain flexibility without having to hand-craft heuristics is to learn the update rules of BBO algorithms from data, in a way that makes them more adaptive and scalable. This is the approach we take: We meta-learn a neural network parametrization of a BBO update rule, on a set of representative task families, while leveraging evaluation parallelism of different BBO instances on modern accelerators, building on recent developments in learned optimization [e.g. 3]. This procedure discovers novel black-box optimization methods via meta-black-box optimization, and is abbreviated by MetaBBO. Here, we investigate one particular instance of MetaBBO and leverage it to discover a learned evolution strategy (LES). The concrete LES architecture can be viewed as a minimal Set Transformer [2], which naturally enforces an update rule that is invariant to the ordering of candidate solutions within a batch of black-box evaluations. After meta-training, LES has learned to flexibly interpolate between copying the best-performing candidate solution and successive moving average updating. Additionally, we show that LES can be trained in a self-referential fashion, i.e. by bootstrapping its own update progress.

2 LES: INVARIANCE VIA SELF-ATTENTION

A fundamental property that any (learned) ES has to fulfill is invariance in the ordering of population members within a generation. Intuitively, the order of the population members is arbitrary and should therefore not affect the search distribution update. A natural inductive bias for an appropriate neural network-based parametrization is given by the dot-product self-attention mechanism. Our proposed learned evolution strategy processes a matrix \( F_t \) of population member-specific tokens, \( F_t \in \mathbb{R}^{N \times 3} \) consists of (1) the \( z \)-scored population fitness scores, (2) a centered rank transformation (lying within \([-0.5, 0.5]\)), and (3) a Boolean indicating whether the fitness score exceeds the previously best score. This fitness matrix is embedded into queries, keys and values using learned linear transforms with weights \( W_K, W_Q, W_V \). Recombination weights are then computed as attention scores over the members:

\[
W_t = \text{softmax} \left( \text{softmax} \left( \frac{F_t W_Q W^T W^K}{\sqrt{D}} \right) F_t W_V \right) \in \mathbb{R}^N
\]

The resulting recombination weights vary across the generations \( t \) and we additionally modulate the mean and standard deviation learning rates \( \alpha_m, \alpha_s, \alpha_t \in \mathbb{R}^D \) by a MLP with parameters \( \phi \). It processes a timestamp embedding and parameter-specific information provided by momentum-like statistics. The set of LES parameters are given by \( \theta = \{ W_K, W_Q, W_V, \phi \} \).

3 METABBO OF LEARNED ES

MetaBBO adopts a meta-evolution approach in order to optimize the parameters characterizing LES \( \theta \). We iterate the following steps:

Meta-sampling. At each meta-generation we sample a population of LES network parametrizations \( \theta_i \) for \( i = 1, \ldots, M \) meta-population members using a standard off-the-shelf ES algorithm.

Inner loop search. Next, we estimate the performance of different LES parametrizations on a set of tasks. We sample a set of inner loop optimization problems. For each task we initialize a mean \( \omega_0 \in \mathbb{R}^D \), standard deviation \( \sigma_0 \in \mathbb{R}^D \). Each LES instance is then executed on the batch of inner loop tasks (in parallel) with \( N \) population members and for a fixed set of inner-loop generations \( t = 1, \ldots, T \).

Meta-normalize. In order to ensure stable meta-optimization with optimization problems that can exhibit very different fitness scales, we normalize (\( z \)-score) the fitness scores within tasks and across meta-population members. We average the scores over tasks, and maximize over inner-loop population members & generations.

Meta-updating. Given the meta-performance estimate for the different LES parametrizations, we update the meta-ES search distribution and iterate. The choice of the meta-task distribution is important to ensure generalization of the meta-learned evolution strategy. We select a small set of classic BBO functions, to characterize a space of representative optimization problems. The problem space includes smooth functions with and without high curvature and multi-modal surfaces with global and local structure. We apply Gaussian additive noise to the fitness evaluations to mimic unreliable estimation and sample optima offsets.

4 SUMMARY OF RESULTS

Attention-based LES. We propose a novel self-attention-based ES parametrization, and demonstrate that it is possible to meta-learn BBO algorithms that outperform existing hand-crafted ES across optimization problems & compute resources (Figures 1 & 3 in [1]).

Strong generalization of LES. We investigate the importance of the meta-task distribution and meta-training protocol. In order to meta-evolve a well-performing ES, only a handful of core optimization classes are needed at meta-training time (Figures 4 & 8 in [1]).

Reverse-engineered DES. After ablating the neural network components, we recover an interpretable and analytically expressable ES. The discovered evolution strategy (DES) provides a simple to implement, competitive ES (Figures 5 & 17 in [1]).

Self-referential meta-evolution. We demonstrate how to generate a new LES starting from a blank-slate LES: A randomly initialized LES can bootstrap its own learning progress and self-referentially meta-learn its own weights (Figures 6 & 18 in [1]).

Summary. Our work highlights the potential of flexible attention-based parametrization & data-driven meta-learning of BBO.

REFERENCES

[1] Robert Tjarko Lange, Tom Schaul, Yutian Chen, Tom Zahavy, Valentin Dalibard, Chris Lu, Satinder Singh, and Sebastian Fleminghag. 2023. Discovering Evolution Strategies via Meta-Black-Box Optimization. ICLR (2023).

[2] Junho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. 2019. Set transformer: A framework for attention-based permutation-invariant neural networks. In International conference on machine learning. PMLR.

[3] Luke Metz, C. Daniel Freeman, James Harrison, Niru Maheswaranathan, and Jascha Sohl-Dickstein. 2022. Practical tradeoffs between memory, compute, and performance in learned optimizers. arXiv preprint arXiv:2203.11860 (2022).