An Invariant Information Geometric Method for High-Dimensional Online Optimization

Zhengfei Zhang ZF-ZHANG20@MAILS.TSINGHUA.EDU.CN
Yunyue Wei WEIYY20@MAILS.TSINGHUA.EDU.CN
Yanan Sui YSUI@TSINGHUA.EDU.CN
Tsinghua University, Beijing, China

Editors: A. Abate, M. Cannon, K. Margellos, A. Papachristodoulou

Abstract

Sample efficiency is crucial in optimization, particularly in high-dimensional black-box scenarios characterized by expensive evaluations and zeroth-order feedback. When computing resources are plentiful, Bayesian optimization is often favored over evolution strategies with this criterion. In this paper, we introduce a fully invariant evolution strategies algorithm, derived from its corresponding framework, that effectively rivals the leading Bayesian optimization method. Specifically, we first build the framework INVIGO that has proper computational costs, incorporates complete historical information, and is fully invariant. We then exemplify INVIGO on multi-dimensional Gaussian, which gives an invariant and scalable optimizer SYNCMA . The theoretical behavior and advantages of our algorithm over other Gaussian-based evolution strategies are further analyzed. Finally, We benchmark SYNCMA against leading algorithms in Bayesian optimization and evolution strategies on various high dimension tasks, including Mujoco locomotion tasks, rover planning task and synthetic functions. In all scenarios, SYNCMA demonstrates great competence, if not dominance, over other algorithms in sample efficiency, showing the underdeveloped potential of property oriented evolution strategies.

Keywords: Invariant optimizer, Information geometry, Evolution strategies, Bayesian optimization

1. Introduction

Many real-world continuous-space optimization problems do not have access to gradient information, and can only rely on zeroth-order evaluations. Moreover, these function evaluations are often costly and become less useful over time as the environment changes. Such tasks are then usually approached as online optimization problems with zeroth-order feedback and an ignorant initial. An ideal optimizer, to this end, should have high sample efficiency with reasonable computational complexity.

Bayesian optimization, with this criterion, is often the favored choice because it has empirically better sample efficiency in various machine learning scenarios (Shahriari et al., 2015; Frazier, 2018). Initially, this success is limited to low-dimensional problems due to its cubic computational complexity of the surrogate model (Rasmussen, 2003). Versatile scalable variants of Bayesian optimization have been developed recently (Eriksson et al., 2019; Binois and Wycoff, 2022), extending the dominance dimension of Bayesian optimization up to hundreds. In this paper, we use the name of high-dimension to denote dimensions ranging from dozens to hundreds, and the name of online optimization to include both Bayesian optimization and evolution strategies.

Evolution strategies usually has its computational complexity independent from sample size, which makes it a general method to apply. Over years of development, some theoretical frameworks and guidance are developed (Akimoto et al., 2012, 2014), through which covariance matrix adaptation evolution strategies (CMA-ES) (Hansen, 2016) and its variants (Abdolmaleki et al., 2017; Akimoto and Hansen, 2020) stand out. They are the current leading family of algorithms and achieve a balance between sample efficiency and computational cost. However, the lack of a solid theoretical foundation greatly hinders their development despite of many efforts invested (Arnold and Hansen, 2010; Brockhoff et al., 2012; Ba et al., 2016; Akimoto and Hansen, 2016; Shirakawa et al., 2018; Nishida and Akimoto, 2018). The potential of CMA family and even evolution strategies seem to be far from being explored. And we aim at exploring this potential from a property oriented perspective instead of developing a solid theory. In specific, we wonder, *can a fully invariant evolution strategies algorithm have great competence against Bayesian optimization in high-dimensional tasks?*

The invariant orientation is motivated by general optimization problems where gradient information is available. In this scenario, it is widely known that the performance of leading first-order optimizers such as AdaGrad (Duchi et al., 2011) and Adam (Kingma and Ba, 2014) are highly dependent on the curvature of the optimization objective. Since the curvature depends on the parameterization of the model, parameterization invariant optimizers are thus considered as promising ways when curvature is unknown. In achieving invariant optimizer, natural gradient (Amari, 1998) and further efforts (Transtrum and Sethna, 2012; Song et al., 2018) that concentrate on exploiting first or higher order structure in parameter space, are probably the most effective thread of research, yet less has been made. When gradient information is not available, sampling is used in information geometric optimization (IGO) (Ollivier et al., 2017) to estimate the natural gradient for specific parametric distributions. Similarly, geodesic modification is also explored (Bensadon, 2015) but the practical invariant capability is limited as in the general case.

Our contribution. We build the first invariant optimizer framework INVIGO for online optimization with ignorant initial and zeroth-order feedback, which adopts an approximation to the objective in IGO to allow everywhere differentiability, and a line search strategy to completely and scalably incorporate historical information. When further exemplified with multi-dimensional Gaussian that CMA optimizers built upon, the derived practical optimizer SYNCMA inherits all properties of INVIGO and has the same computation costs as CMA-ES. It is also the first time that historical information is stably incorporated for both mean and covariance parameters. In experiments that benchmark on high dimensional realistic tasks, where Bayesian optimizers usually dominant, and synthetic tasks, SYNCMA demonstrates great competence over other optimizers in sample efficiency.

2. Background

We study the optimization problem where a black-box function f needs to be optimized, and the optimizer is initially ignorant with only zeroth-order feedback available,

$$x^* = \operatorname{argmin}_{x\in\mathbb{R}^n} f(x). \tag{1}$$

To get rid of the potential complex nature of f, a global parametric sample distribution $\theta \mapsto p_\theta$ over the domain of x and a substitutional fitness function $g_{f,\theta}(x)$ are often considered. The relaxed

problem then becomes :

$$\theta^* = \text{argmin}_{\theta\in\Theta}\, \mathbb{E}_{p_\theta}[g_{f,\theta}(x)], \tag{2}$$

where fitness function g is manually selected to approximate f while maintaining good properties, such as integrable. Ideally, given g, $\theta^* \in \Theta$ should approximately equal to the delta distribution that has all its probability on x^*. Also, the fitness function generalizes the problem as specifically, i) if f has good properties, then it is naturally to set $g_{f,\theta}$ as f; ii) if g is indeed related with θ, e.g. p_θ-level function (Ollivier et al., 2017), then the problem has a time-varying environment that many online optimizers pursue.

2.1. Natural Gradient Flow with Zeroth-order Feedback

Equation (2) is often solved with natural gradient flow on the parameter space Θ when targeting on invariance (Amari, 1998; Kakade, 2001),

$$\frac{\mathrm{d}\theta}{\mathrm{d}t} = -\tilde{\nabla}_\theta \mathbb{E}_{p_\theta}[g_{f,\theta^t}(x)] = -\tilde{\nabla}_\theta L_{\theta^t}(\theta), \tag{3}$$

where θ^t denotes θ at time step t, $\tilde{\nabla}_\theta := I^{-1}(\theta)\nabla_\theta$ is the natural gradient with $I(\theta)$ the fisher information matrix, and $L_{\theta^t}(\theta) := \mathbb{E}_{p_\theta}[g_{f,\theta^t}(x)]$ denotes the loss function with Θ as its domain. The usage of natural gradient keeps this ODE invariant under smooth bijective transformation of parameter space. Its vanilla discrete version, i.e. natural gradient descent algorithm goes to,

$$\theta^{t+1} = \theta^t - h\tilde{\nabla}_\theta|_{\theta=\theta^t} L_{\theta^t}(\theta). \tag{4}$$

Here h denotes the learning rate. To practically compute the natural gradient in black-box setting, information geometric optimization (IGO) sets a sampling method given space Θ,

$$\tilde{\nabla}_\theta|_{\theta=\theta^t} L_{\theta^t}(\theta) = I^{-1}(\theta^t) \int g_{f,\theta^t}(x) \frac{\partial \ln p_\theta(x)}{\partial \theta}|_{\theta=\theta^t} p_{\theta^t}(\mathrm{d}x). \tag{5}$$

Different choices of distribution family Θ thus provide different optimization methods in the form of (4), we thus assume a finite computational cost $\mathcal{O}(H_\Theta)$ which widely holds (Akimoto et al., 2012; Hansen, 2016), and define the IGO complexity as a measurement accordingly.

Assumption 1 *For a given $x \in \mathbb{R}^n$ and $\theta \in \Theta$, $I^{-1}(\theta)\frac{\partial \ln p_\theta(x)}{\partial \theta}$ cost $\mathcal{O}(H_\Theta)$ time to compute.*

Definition 1 (IGO complexity) *When Assumption 1 holds, the IGO complexity $\mathcal{O}(H_\Theta N)$ denotes the computational complexity for single step updates when applying IGO to natural gradient method, i.e. to compute equation (5) with N samples.*

When discretizing with a given learning rate, errors with respect to the invariant property occur, which may accumulate to drastically change the trajectory. In gradient accessible setting with general loss function, the best invariant error achieved (Song et al., 2018) is 2-nd order invariant, representing the decrease of the error between the optimizer and some exactly invariant trajectories is $\mathcal{O}(h^2)$. Similar error order is achieved in the content of black-box setting for certain parametric distribution (Bensadon, 2015). There are some attempts to illustrate a fully invariant optimizer, such as IGO-ML in Ollivier et al. (2017), but they are not practically invented. In short, there is no practical algorithm that has a better order of invariance; when introducing historical information, e.g. momentum, current invariance order and even stability might be violated (Akimoto et al., 2014).

Definition 2 (Invariant property) *Let θ be the parameter of an optimizer using model p_θ and $\varphi(\theta)$ be an smooth bijective transformation of θ of the same optimizer using model $p'_{\varphi(\theta)} = p_\theta$. Let θ^t be the optimization trajectory when optimizing objective f, parameterized by θ and initialized at θ^0. And φ^t the optimization trajectory when optimizing objective f, parameterized by φ and initialized at $\varphi^0 = \varphi(\theta^0)$. We claim that the optimizer is invariant if $\forall t \in \mathbb{N}, \varphi^t = \varphi(\theta^t)$.*

3. An Invariant Optimizer Family with an Approximate Objective

In this section, we will overcome aforementioned challenges in invariance and stablity, presenting a fully invariant optimizer family with historical information incorporated.

3.1. Optimizing with the Approximate Objective

We start with replacing the loss function $L_{\theta^t}(\theta) \coloneqq \mathbb{E}_{p_\theta}[g_{f,\theta^t}(x)]$ in (3) that is only computational differentiable at point $\theta = \theta^t$. Given that g_{f,θ^t} is manually selected and $\forall b \in \mathbb{R}, \nabla_\theta E_{p_\theta}[g_{f,\theta^t}(x)] = \nabla_\theta E_{p_\theta}[g_{f,\theta^t}(x) + b]$, we assume g_{f,θ^t} to be non-negative without loss of generality. Then we denote reweighted distribution $q_\theta(x) \coloneqq \frac{p_\theta(x) g_{f,\theta^t}(x)}{L_{\theta^t}(\theta)}$ and decompose $\log L_{\theta^t}(\theta)$ as follow,

$$\log \frac{L_{\theta^t}(\theta)}{L_{\theta^t}(\theta^t)} = D_{KL}(q_{\theta^t} \| q_\theta) + D_{KL}(q_{\theta^t} \| p_{\theta^t}) - D_{KL}(q_{\theta^t} \| p_\theta). \tag{6}$$

where $D_{KL}(p\|q) \coloneqq \int p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x$ is the Kullback–Leibler (KL) divergence.

Inspired from decomposition (6), we claim $D_{KL}(q_{\theta^t} \| p_\theta)$ a good objective approximating $L_{\theta^t}(\theta)$. All the proofs and detailed derivations are set in Appendix[1] A.

Proposition 3 *The KL-divergence $D_{KL}(q_{\theta^t} \| p_\theta)$ is a substitution for $L_{\theta^t}(\theta)$ with the following properties.*

1. *The (natural) gradients for $\log L_{\theta^t}(\theta)$ and $-D_{KL}(q_{\theta^t} \| p_\theta)$ coincide at current point θ^t, further, for every $\theta \coloneqq (\theta^t + \delta\theta) \in \Theta$, $\nabla_\theta \log L_{\theta^t}(\theta) = -\nabla_\theta D_{KL}(q_{\theta^t} \| p_\theta) + O(\delta\theta)$.*
2. *Under Assumption 1 , computing natural gradient of $D_{KL}(q_{\theta^t} \| p_\theta)$ at any point $\theta \in \Theta$ costs the IGO complexity $O(H_\Theta N)$. While objective $L_{\theta^t}(\theta)$ in IGO is only differentiable at point θ^t.*

To utilize the everywhere differentiability of $D_{KL}(q_{\theta^t} \| p_\theta)$, we then frame discretized updates for θ^{t+1} as a step-size constrained optimization problem instead of vanilla descents in (4),

$$\theta_*^{t+1} = \operatorname{argmax}_\theta D_{KL}(q_{\theta^t} \| p_\theta) \ \ s.t.\ D_{KL}(p_{\theta^t} \| p_\theta) \le \epsilon^2/2. \tag{7}$$

The specific choice of the constraint comes from the definition of natural gradient (Amari, 1998; Martens, 2020) $-\tilde{\nabla}|_{\theta=\theta^t} L_{\theta^t}(\theta) \propto \lim_{\epsilon \to 0^+} \frac{1}{\epsilon} \operatorname{argmax}_{\delta\theta \ s.t.\ D_{KL}(p_{\theta^t} \| p_{\theta^t + \delta\theta}) \le \epsilon^2/2} D_{KL}(q_{\theta^t} \| p_{\theta^t + \delta\theta})$. More precisely, from this definition, the optimization problem (7) is approximately solving $\frac{\mathrm{d}\theta}{\mathrm{d}t} = -s(\theta)\tilde{\nabla} \log L_{\theta^t}(\theta)$ with $s(\theta) \coloneqq \frac{\epsilon}{\|\tilde{\nabla}|_{\theta=\theta^t} D_{KL}(q_{\theta^t} \| p_\theta)\|_{I^{-1}}}$ where $\|v\|_B \coloneqq \sqrt{v^T B v}$ for vector v and matrix B.

1. Please refer to https://github.com/Anoxxx/SynCMA-official for appendix and source codes.

3.2. Invariantly Incorporating Historical Information

When only local information is used in each iteration, historical information is helpful for the optimization, even if the environment is constantly changing over time (Yuan et al., 2016). We thus modify objective $D_{KL}(q_{\theta^t}\|p_\theta)$ to incorporate historical information. Here T denotes the horizon and the widely used exponential decay is applied with decay parameter $\lambda \in [0, 1)$. λ^0 is regarded as 1 in default.

$$\theta_*^{t+1} = \text{argmax}_\theta \sum_{\tau=0}^{T} \lambda^\tau D_{KL}(q_{\theta^{t-\tau}}\|p_\theta) \;\; s.t.\; D_{KL}(p_{\theta^t}\|p_\theta) \leq \epsilon^2/2. \tag{8}$$

Although problem (8) can be solved with strong duality and additional convex optimization, applying a simple natural Lagrange condition as a line search will yields more room for accessible invariant with proper computational cost. Here we denote $G^t(\theta) := \sum_{\tau=0}^{T} \lambda^\tau D_{KL}(q_{\theta^{t-\tau}}\|p_\theta)$,

$$\tilde{\nabla}_\theta|_{\theta=\theta^{t+1}}(-G^t(\theta) + \eta(\epsilon^2/2 - D_{KL}(p_{\theta^t}\|p_\theta))) = 0. \tag{9}$$

We name this algorithm family from iteratively solving (9) for different choice of parametric distribution family Θ as INVIGO. Further, when T is set to be large, we can always replace $G^t(\theta)$ with a self-evolved term $M^t(\theta)$ that retain the gradient information. In practice, it suffice to evolve only $\tilde{\nabla}_\theta M^t(\theta)$ as shown in the following section.

$$\tilde{\nabla}_\theta G^t(\theta) = -\tilde{\nabla}_\theta M^t(\theta) + \tilde{\nabla}_\theta D_{KL}(q_{\theta^t}\|p_\theta). \tag{10}$$

Assumption 2 *The chosen fitness function $g_{f,\theta^t}(x)$ and the Lagrange multiplier η are independent from the parameterization of θ.*

Theorem 4 (Invariant for INVIGO) *When assumption 1, 2 hold and the decay weight λ is independent from the parameterization of $\theta \in \Theta$, optimizers in INVIGO are invariant and the single step computational cost is $\mathcal{O}(\min(H_\Theta NT, H_\Theta N + K))$ where $\mathcal{O}(K)$ denotes the cost to compute $\tilde{\nabla}_\theta M^t(\theta)$.*

4. Exemplifying with Multi-dimensional Gaussian

We exemplify INVIGO with multi-dimensional Gaussian as our candidate distribution family Θ. To start with, we clarify the computational accessibility of multi-dimensional Gaussian for Assumption 1.

Proposition 5 (Theorem 4.1 in Akimoto et al. (2012)) *Suppose θ_m and θ_c are n- and $n(n+1)/2$-dimensional column representing mean m and covariance matrix C respectively. Then $\partial m/\partial\theta_m$ and $\partial vec(C)/\partial\theta_c$ are invertible at $\theta \in \Theta$ and,*

$$I_m^{-1}(\theta)\frac{\partial \ln p_\theta(x)}{\partial \theta_m} = (\frac{\partial m}{\partial \theta_m})^{-1}(x - m), \tag{11}$$

$$I_c^{-1}(\theta)\frac{\partial \ln p_\theta(x)}{\partial \theta_c} = (\frac{\partial vec(C)}{\partial \theta_c})^{-1} vec((x - m)(x - m)^T - C). \tag{12}$$

where the fisher information matrix $I(\theta)$ is consists of two blocks for mean and covariance respectively by parameterizations.

The requirement of fitness function $g_{f,\theta^t}(x)$ in Assumption 2 widely holds, such as the level function that reflect the probability to sample a value better than $f(x)$ according to p_{θ^t} in standard CMA-ES (Hansen, 2016). We choose this fitness function for comparing needs : In time step t, N samples $\{x_i^t\}$ are drawn from p_{θ^t} and we further denote $\hat{w}_i^t := \frac{g_{f,\theta^t}(x_i^t)}{\sum_i g_{f,\theta^t}(x_i^t)}$ as the normalized fitness for sample x_i^t. Finally, according to Proposition 5, parameter $\theta = (\theta_m, \theta_c) \mapsto \mathcal{N}(m, C)$ with $\theta_m \in \mathbb{R}^n$ and $\theta_c \in \mathbb{R}^{n(n+1)/2}$ representing mean and covariance respectively. We can thus split the Lagrange multiplier into constants pair $\eta = (\eta_m, \eta_c)$ in INVIGO without violating the invariant property. The Assumption 2 is satisfied thereby.

We use the parameterization $\theta = (m, C)$ for simplification sake through this section. Different parameterizations that meet the conditions in Proposition 5 will conduct different practical optimizers by following this section with minor modifications. The performance should be the same up to the transformation due to the invariant property.

4.1. An Invariant Optimizer with Historical Information : SYNCMA

We directly apply INVIGO and a maximum time horizon $T = t - 1$. By choosing such infinite horizon, the historical information is maximally used. To reduce the computational costs to the same as IGO, i.e. scalable to $\mathcal{O}(H_\Theta N)$ for single step update, we design $M^t(\theta)$ as follow,

$$\begin{aligned}\tilde{\nabla}_m M^t(\theta) &= \lambda_0(s_m^t + m^t - m), \\ \tilde{\nabla}_c M^t(\theta) &= \lambda_0((s_c^t + m^t - m)(s_c^t + m^t - m)^T - C) + Q_1^t + Q_2^t \circ m + Q_3^t m m^T.\end{aligned} \tag{13}$$

Here scalars $\lambda_0, Q_1^t \in \mathbb{R}$ and vectors $s_m^t, s_c^t, Q_2^t, Q_3^t \in \mathbb{R}^n$, with $\circ$ applying to two vectors $v_1, v_2 \in \mathbb{R}^n$ that denotes $v_1 \circ v_2 := v_1 v_2^T + v_2 v_1^T$. For brevity sake, we denote $d_i^t := x_i^t - m^t$, $d_w^t := \sum_i \hat{w}_i^t d_i^t$, $\hat{d}_w^t := d_w^t + m^t$ to represent statistics in a single generation, and $\hat{s}_m^{t-1} := s_m^{t-1} + m^{t-1}$, $\hat{s}_c^{t-1} := s_c^{t-1} + m^{t-1}$ to represent elements for history. Corresponding updates for hyperparameter $\lambda_0 \in \mathbb{R}$ and self-evolved terms $s_m^t, s_c^t, Q_2^t, Q_3^t \in \mathbb{R}^n$ that initially zero are shown below:

$$\lambda = \lambda_0 / \lambda_0 + 1, \tag{14}$$

$$s_m^t + m^t = \lambda \hat{s}_m^{t-1} + (1 - \lambda)\hat{d}_w^{t-1}, \tag{15}$$

$$s_c^t + m^t = \sqrt{\lambda}\hat{s}_c^{t-1} + \sqrt{1 - \lambda}\hat{d}_w^{t-1}, \tag{16}$$

$$Q_1^t = \lambda Q_1^{t-1} + \lambda \sum \hat{w}_i (d_i^{t-1} - d_w^{t-1})(d_i^{t-1} - d_w^{t-1})^T - \lambda_0 \sqrt{\lambda}\sqrt{1 - \lambda}\hat{d}_w^{t-1} \circ \hat{s}_c^{t-1}, \tag{17}$$

$$Q_2^t = \lambda Q_2^{t-1} - \lambda_0(\sqrt{\lambda} + \sqrt{1 - \lambda} - 2)(\sqrt{\lambda} * \hat{s}_c^{t-1} + \sqrt{1 - \lambda} * \hat{d}_w^{t-1}), \tag{18}$$

$$Q_3^t = \lambda Q_3^{t-1} - \lambda_0(\sqrt{\lambda} - 1)(\sqrt{1 - \lambda} - 1). \tag{19}$$

We now arrive at the single step update for next parameter $\theta^{t+1} = (m^{t+1}, C^{t+1})$. The resulting algorithm is named as SYNCMA to emphasize another prominent characterization, the synchronous update nature, as discussed in section 4.2, besides invariance. The final updates in single iteration with $z_m = \eta_m + \lambda_0 + 1$, $z_c = \eta_c + \lambda_0 + 1$, $\beta^t = \frac{1}{z_m}(d_w^t + \lambda_0 s_m^t)$ for brevity sake are shown below:

$$m^{t+1} = m^t + \beta^t, \tag{20}$$

$$\begin{aligned}C^{t+1} = &\frac{\eta_c}{z_c}(C^t + \beta^t(\beta^t)^T) + \frac{\lambda_0}{z_c}(s_c^t - \beta^t)(s_c^t - \beta^t)^T \\ &+ \frac{1}{z_c}(\sum_i \hat{w}_i (d_i^t - \beta^t)(d_i^t - \beta^t)^T + Q_1^t + Q_2^t \circ m^t + Q_3^t m^t (m^t)^T).\end{aligned} \tag{21}$$

4.2. Theoretical Comparison with Other CMA Optimizers

To the best of our knowledge, SYNCMA is the first fully invariant optimizer and the first CMA optimizer stably incorporating historical information in mean updates. We aim to characterize two additional properties of SYNCMA, clarify the absence of step-size adaption, and connect SYNCMA with CMA-ES in this subsection.

Synchronous Updates Given that INVIGO treats the current distribution θ as a single point in parameter space to update, the updates for mean and covariance matrix in SYNCMA naturally intertwine. This synchronous update nature allows SYNCMA to strictly follow Proposition 5, which is fundamental for all such CMA optimizers. In other CMA optimizers, updates in each iteration are sequentially performed, and thus only approximately follows Proposition 5, e.g. $m^{t+1} = U_m(m^t, (\sigma^t)^2\Sigma^t)$ and $\Sigma^{t+1} = U_c(m^{t+1}, (\sigma^t)^2\Sigma^t)$ where U_m and U_c are updates for mean and covariance, and σ_t is the additional step-size which SYNCMA lacks.

Effective Learning Rate As illustrated at the end of Section 3.1, INVIGO along with its derived algorithm SYNCMA have an effective rate that inversely proportion to gradient, which enhances their capabilities to escape local optima and saddle points over other optimizers. This property is further examined in experiments with rugged functions.

Absence of Step-size Adaption One of the direct consequences of directly exemplifying from INVIGO, i.e. synchronous updates, is the lack of the step-size σ_t adaption. Such step-size scaling the sampling region is one of the crucial components in lots modern optimizers (Hansen, 2016; Abdolmaleki et al., 2017). While we are fully aware of its importance, we exclude this part of research in the favor of strictly demonstrating INVIGO in a practical way, and sincerely regard an additional step-size in SYNCMA as a future topic.

Connection to CMA-ES. It is also worth building the connection between SYNCMA and the CMA family algorithms in Proposition 6, where two used approximations correspond to the fully incorporation of history and the synchronous nature of SYNCMA.

Proposition 6 *When i) the historical information is partially used for covariance, i.e.* $\tilde{\nabla}_m M^t(\theta) = 0$ *and* $\tilde{\nabla}_c M^t(\theta) = \lambda_0((s_c^t + m^t - m)(s_c^t + m^t - m)^T - C)$. *ii) all the higher order terms, when assuming* $\eta_c \approx z_c \gg 1, z_m \gg 1$, *are discarded.* SYNCMA *coincide with CMA-ES up to an external learning rate difference.*

5. Experiments

In this section, we intensively evaluate SYNCMA with other baselines in Mujoco locomotion tasks, rover planning task and synthetic functions. The criteria are chosen in the context of online optimization, focusing on full optimization procedures in the natural axis and sample efficiency when achieve a near global value. All optimization procedures are plotted with the shaded area bounded by quartiles and the solid line denoting the median performance over all trails.

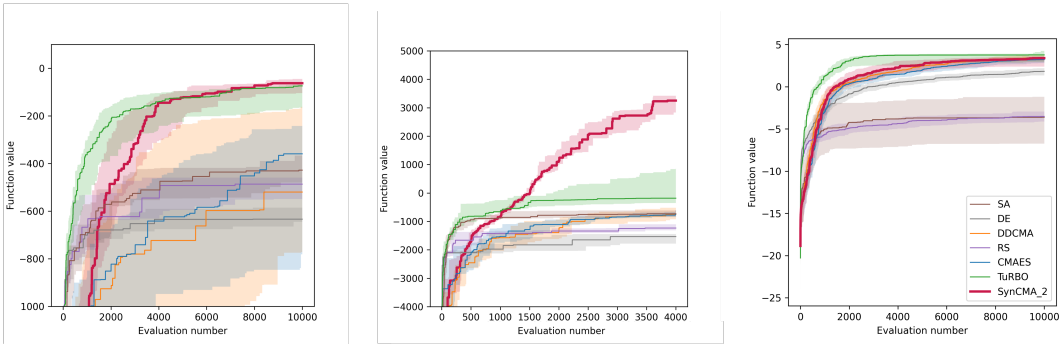
Baselines are chosen in a structured way. First, random search (RS) (Bergstra and Bengio, 2012) is chosen as the overall baseline. Then, two black-box optimizers, differential evolution (DE) (Storn and Price, 1997) and simulated annealing (SA) (Bouttier and Gavra, 2019) are chosen. Among the CMA optimizers, we choose CMA-ES and two of its leading variants DD-CMA (Akimoto and Hansen, 2020) and TR-CMA-ES (Abdolmaleki et al., 2017) for detailed comparison. Finally, the

Bayesian optimization method TuRBO (Eriksson et al., 2019) is used as the state-of-the-art baseline for BO. Parameters η_m, η_c of SYNCMA are set to constant that match the initial settings of the corresponding parameters in CMA-ES. λ_0 corresponds to a combination of several parameters in CMA-ES so we simply test with the constant value $\lambda_0 = 2$, which corresponds to the approximate counterparts in CMA-ES. We use this value throughout the main paper while there are better performances of SYNCMA with different λ_0 as shown in ablation studies in Appendix B.4. The initial distribution for all Gaussian based optimizers are identity matrix.

TR-CMA-ES is based on its original paper version implemented in Matlab due to precision problems in Python, and therefore we exclude TR-CMA-ES in the Mujoco locomotion and rover planning tasks as they are based on specific Python libraries. All other baselines are implemented with their fine-tuned version available online (Balandat et al., 2020; Duan et al., 2022). See Appendix B for details.

5.1. Mujoco Locomotion Task

We first evaluate SYNCMA and other baselines on the widely tested Mujoco locomotion tasks (Todorov et al., 2012), which are popular benchmarks for Bayesian optimization and reinforcement learning algorithms. To run sampling-based optimizers on Mujoco, we refer to (Mania et al., 2018) and optimize a linear policy: $\mathbf{a} = \mathbf{W}\mathbf{s}$, where $\mathbf{a}$ is the agent action and $\mathbf{s}$ is the environment state. The parameter matrix $\mathbf{W}$ are continuous and in the range of $[-1, 1]$. Among all 6 tasks, we dismiss the overly high dimensional task Humanoid(6392d) and test all other 5 tasks with batch size $N = 100$. Two results are shown here in figure 1(*a*), 1(*b*) with more results in Appendix B.1. While TuRBO dominates other baselines, SYNCMA outperforms TuRBO in 2 tasks and remains competitive with TuRBO for the other 3 tasks.



(*a*) Ant(888d) (*b*) HalfCheetah(102d) (*c*) Rover(60d)

Figure 1: Optimization procedure for two high dimensional Mujoco locomotion tasks over 10 trials and rover planning task over 100 trails. Index of SYNCMA indicate λ_0.

5.2. Rover Planning Task

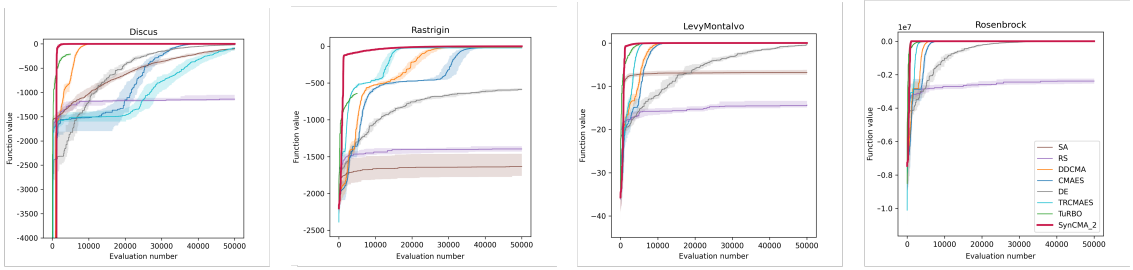
To further explore the empirical performance of SYNCMA in a realistic setting, we consider the rover trajectory optimization task, where a start position s and a goal position g are defined in the 2D plane, as well as a cost function $c(x)$ over the state space. The trajectories are described by a set of points to which a B-spline is fitted and the cost function is computed. The whole state space

is $x \in [0,1]^{60}$ and we make the batch size $N = 2n = 120$. A reward function to be optimized is defined to be non-smooth, discontinuous, and concave over the first two and last two dimensions of the state. The result in figure 1(c) shows that SYNCMA still exhibits competitive performance over other baselines.

5.3. Synthetic Function

We select 10 commonly used synthetic functions with dimension n arbitrarily set. This is the traditional test bed for black-box optimization and specific evolution strategies. These functions, including different characteristics such as multi-modal, ill-conditioned and ill-scaled, are scaled to a global minimum value 0 with shifted domain. The batch size is $N = 2n$ and the evaluation limit is the same for all optimizers except TuRBO, where the budget is fixed at 5,000 evaluations due to memory limitations in storing matrtix.

The full experiments are run with different dimensions of $n = \{32, 64, 128\}$ and the results are presented in two ways under the same evaluation budget: near global optimum performance and the whole optimization procedure. Some results are presented here and please refer to Appendix B.3 for the full experimental results.



(a) Discus (b) Rastrigin (c) LevyMontalvo (d) Rosenbrock

Figure 2: Optimization procedure in 4 typical synthetic functions with dimension $n = 64$ over 20 trails considering all optimizers. Index of SYNCMA indicate λ_0.

Table 1: Near global optimum performance on 64d synthetic functions(lower is better) over 20 trials with budget of 50000 evaluations, TuRBO is excluded due to memory limitation. Numbers in brackets indicates the median evaluation number needed for optimizers to achieve value better than 0.5.

Optimizer	Sphere	Discus	Schwefel	DiffPowers	LevyMontalvo	Rastrigin	Ackley
SA	0.6	100.7	0.1(2650)	37.5	6.8	1634.7	13.6
RS	793.9	1138.8	29902.6	2369.4	14.4	1395.9	11.8
DE	7.2	20.0	25.6	63.6	0.4(49700)	586.9	3.1
DDCMA	0.0(10047)	**0.0(12748)**	0.0(16820)	0.0(10164)	0.0(9087)	17.9	0.0(11757)
CMAES	0.0(13825)	0.0(43265)	0.0(16134)	0.0(18503)	0.0(9901)	21.4	0.0(15620)
TRCMAES	0.0(7185)	85.9	0.0(9905)	0.0(12040)	0.0(5515)	22.4	0.0(7869)
SYNCMA(**Ours**)	**0.0(3938)**	0.0(18820)	**0.0(1157)**	**0.0(1158)**	**0.0(2318)**	**0.2(42696)**	**0.0(7567)**

According to table 1 where TuRBO is excluded as it is unable to scale to this budget, SYNCMA demonstrate both superior optimization capability and efficiency over others. While other optimizers are

less efficient and fail to optimize high-conditional number multi-model function Rastrigin, ill-scaled function Discus and others. Further, full optimization procedures including TuRBO with maximum budget under storage limit are partially shown in figure 2, SYNCMA still outperforms others including TuRBO after first several hundreds evaluation from 32 to 128 dimension, demonstrating the capability of such optimizer derived from an invariant framework.

5.4. Ablation Study

The weight for historical information λ_0 is a parameter that substitutes a combination of several parameters in CMA optimizers, and is set constantly as $\lambda_0 = 2$. We thus study the sensitivity on this parameter for constant setting here. All of previous experiments are repeated for $\lambda_0 \in [0, 4]$, with results for $\lambda_0 = \{0, 1, 2, 4\}$ shown in Appendix B.4, from which we summarize several observations within range $[0, 4]$ here.

Sensitivity. When SYNCMA includes historical information, i.e. $\lambda_0 > 0$, SYNCMA consistently shows competitive performance.
Function Landscape. When their exists a fundamental subspace that covers the structure of the problem, as in Rastrigin, a higher λ_0 yields better performance and efficiency. Otherwise, as in LevyMontalvo, a higher λ_0 might be detrimental.
Dimensionality. Observed from tasks in Mujoco, synthetic functions, and rover planning, a higher dimension generally requires a higher λ_0.

6. Limitations

There is still much to explore in both framework INVIGO and optimizer SYNCMA . For the framework, we directly use Lagrange condition in each step to yield rooms for invariance and complexity, which need more endeavors to characterize its theoretical behavior or design novel per step subroutine. Moreover, our proposed framework and optimizer currently only work for certain parametric distribution families that IGO set. It is possible to generalize to a broader family of models such as neural networks, as we are based on a different objetive from the IGO objective. For the algorithm, the biggest point yet to explore is the step-size adaption which we excluded now for a focus on our invariant framework.

7. Conclusion

We present an invariant optimizer framework INVIGO that fully incorporates historical information, both of invariance and full incorporation are unprecedented. When exemplified with multi-dimensional Gaussian, our framework derives a invariant optimizer SYNCMA that retains the computational complexity as in information geometric optimization. With a straightforward invariant oriented motivation, SYNCMA shows competitive performance in both realistic and synthetic scenarios against leading Bayesian and evolution strategies optimizers.

We highlight its performance on high dimensional realistic problem as it shows the potential of a property oriented evolution strategies optimizer against Bayesian optimization optimizers. And we also defend the importance of fully invariant in optimization. In short, we believe the property oriented perspective is more approachable than inventing a rigorous theory that illustrates and improves current evolution strategies algorithms.

8. Acknowledgment

This project is funded by STI 2030—Major Projects 2022ZD0209400. Correspondence to Yanan Sui.

References

Abbas Abdolmaleki, Bob Price, Nuno Lau, Luis Paulo Reis, and Gerhard Neumann. Deriving and improving cma-es with information geometric trust regions. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 657–664, 2017.

Youhei Akimoto and Nikolaus Hansen. Online model selection for restricted covariance matrix adaptation. In *International Conference on Parallel Problem Solving from Nature*, pages 3–13. Springer, 2016.

Youhei Akimoto and Nikolaus Hansen. Diagonal acceleration for covariance matrix adaptation evolution strategies. *Evolutionary computation*, 28(3):405–435, 2020.

Youhei Akimoto, Yuichi Nagata, Isao Ono, and Shigenobu Kobayashi. Theoretical foundation for cma-es from information geometry perspective. *Algorithmica*, 64:698–716, 2012.

Youhei Akimoto, Anne Auger, and Nikolaus Hansen. Comparison-based natural gradient optimization in high dimension. In *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, pages 373–380, 2014.

Shun-Ichi Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

Dirk V Arnold and Nikolaus Hansen. Active covariance matrix adaptation for the (1+ 1)-cma-es. In *Proceedings of the 12th annual conference on Genetic and evolutionary computation*, pages 385–392, 2010.

Jimmy Ba, Roger Grosse, and James Martens. Distributed second-order optimization using kronecker-factored approximations. In *International Conference on Learning Representations*, 2016.

Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. In *Advances in Neural Information Processing Systems 33*, 2020.

Jérémy Bensadon. Black-box optimization using geodesics in statistical manifolds. *Entropy*, 17(1): 304–345, 2015.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

Mickael Binois and Nathan Wycoff. A survey on high-dimensional gaussian process modeling with application to bayesian optimization. *ACM Transactions on Evolutionary Learning and Optimization*, 2(2):1–26, 2022.

Clément Bouttier and Ioana Gavra. Convergence rate of a simulated annealing algorithm with noisy observations. *The Journal of Machine Learning Research*, 20(1):127–171, 2019.

Dimo Brockhoff, Anne Auger, and Nikolaus Hansen. On the effect of mirroring in the ipop active cma-es on the noiseless bbob testbed. In *Proceedings of the 14th annual conference companion on Genetic and evolutionary computation*, pages 277–284, 2012.

Qiqi Duan, Guochen Zhou, Chang Shao, Zhuowei Wang, Mingyang Feng, Yijun Yang, Qi Zhao, and Yuhui Shi. Pypop7: A pure-python library for population-based black-box optimization. *arXiv preprint arXiv:2212.05652*, 2022.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local bayesian optimization. *Advances in neural information processing systems*, 32, 2019.

Peter I Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018.

Nikolaus Hansen. The cma evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.

Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Horia Mania, Aurelia Guy, and Benjamin Recht. Simple random search of static linear policies is competitive for reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

James Martens. New insights and perspectives on the natural gradient method. *Journal of Machine Learning Research*, 21(146):1–76, 2020.

Kouhei Nishida and Youhei Akimoto. Psa-cma-es: Cma-es with population size adaptation. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 865–872, 2018.

Yann Ollivier, Ludovic Arnold, Anne Auger, and Nikolaus Hansen. Information-geometric optimization algorithms: A unifying picture via invariance principles. *The Journal of Machine Learning Research*, 18(1):564–628, 2017.

Carl Edward Rasmussen. Gaussian processes in machine learning. In *Summer school on machine learning*, pages 63–71. Springer, 2003.

Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1): 148–175, 2015.

Shinichi Shirakawa, Youhei Akimoto, Kazuki Ouchi, and Kouzou Ohara. Sample reuse via importance sampling in information geometric optimization. *arXiv preprint arXiv:1805.12388*, 2018.

Yang Song, Jiaming Song, and Stefano Ermon. Accelerating natural gradient with higher-order invariance. In *International Conference on Machine Learning*, pages 4713–4722. PMLR, 2018.

Rainer Storn and Kenneth Price. Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341, 1997.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

Mark K Transtrum and James P Sethna. Geodesic acceleration and the small-curvature approximation for nonlinear least squares. *arXiv preprint arXiv:1207.4999*, 2012.

Kun Yuan, Bicheng Ying, and Ali H Sayed. On the influence of momentum acceleration on online learning. *Journal of Machine Learning Research*, 17(192):1–66, 2016.