Natural Gradient Approach for Linearly Constrained Continuous Optimization

Youhei Akimoto[1] and Shinichi Shirakawa[2]

[1] Faculty of Engineering, Shinshu University, Nagano, Nagano, Japan
y_akimoto@shinshu-u.ac.jp
[2] College of Science and Engineering, Aoyama Gakuin University,
Sagamihara, Kanagawa, Japan
shirakawa@it.aoyama.ac.jp

Abstract. When a feasible set of an optimization problem is a proper subset of a multidimensional real space and the optimum of the problem is located on or near the boundary of the feasible set, most evolutionary algorithms require a constraint handling machinery to generate better candidate solutions in the feasible set. However, some standard constraint handling such as a resampling strategy affects the distribution of the candidate solutions; the distribution is truncated into the feasible set. Then, the statistical meaning of the update of the distribution parameters will change. To construct the parameter update rule for the covariance matrix adaptation evolution strategy from the same principle as unconstrained cases, namely the natural gradient principle, we derive the natural gradient of the log-likelihood of the Gaussian distribution truncated into a linearly constrained feasible set. We analyze the novel parameter update on a minimization of a spherical function with a linear constraint.

1 Introduction

The covariance matrix adaptation evolution strategy (CMA-ES) is a state-of-the-art randomized search heuristics in continuous domain [8–10, 12]. The CMA-ES maintains the Gaussian distribution, from which candidate solutions are drawn. It repeats the following: sample λ points from the Gaussian distribution, evaluate the fitness for each sample, update the parameters including the mean vector and the covariance matrix of the distribution in order to make the distribution likely to generate better solutions. Recently, it has been shown [2, 7] that the parameter update in the CMA-ES is partially interpreted as a natural gradient ascent on the parameter space of the Gaussian distribution, where the natural gradient is computed for the function defined below in (1). This idea is further generalized to the generic framework for arbitrary optimization, namely information-geometric optimization (IGO) [16].

Since the CMA-ES has been originally proposed for unconstrained continuous optimization, it often requires a treatment when solving a constrained problem. A number of constraint handling strategies have been proposed for evolution strategies and for more generic evolutionary algorithms [15]; e.g., adding an adaptive penalty to the fitness according to the constraint violation [11], repairing an infeasible point into the feasible region by a projection onto the boundary [4] or by a gradient based repair operator [13]. In this paper we consider the *resampling* strategy; an infeasible point is

T. Bartz-Beielstein et al. (Eds.): PPSN XIII 2014, LNCS 8672, pp. 252–261, 2014.

discarded and resampled until it drops into the feasible region. It can be applied even when the constraint functions are black-box.

It has been shown that under a linear constraint, the success probability, i.e. the probability of generating a better point, depends on the angle of the gradients of the constraint function and the objective function and the dependency of the success probability on the angle is different for the resampling [5] and a repair operator [4]. This obviously affects a success probability based parameter update such as step-size adaptation based on the 1/5 success rule [17]. Moreover, since the distribution of the generated point in the feasible region is truncated (in the case of resampling) or biased on the boundary (in the case of repairing), the update rules that are designed from a statistical viewpoint are affected. For example, when the original Gaussian distribution is parameterised by the mean vector m and the covariance matrix C, these parameters no more represent the mean vector and the covariance matrix of the truncated Gaussian distribution. Then the maximum likelihood estimators for m and C for the truncated distribution differ from the ones for the original distribution. Therefore, a treatment in parameter update is required, an example of which is proposed by [6] where the covariance matrix is actively reduced in the direction of the gradient of the constraint.

In this paper we study the effect of the constraint from a viewpoint of the natural gradient. When resampling method is employed, the distribution of the generated feasible points is a truncated probability distribution whose domain is limited to the feasible set. In this situation the natural gradient differs from the one computed for the non-truncated (original) probability distribution. Now a question arises as to if we can gain a better performance by computing the natural gradient on the manifold of the truncated Gaussian distributions limited to the feasible set.

To address the question we derive the natural gradient under a linearly constrained feasible domain and compare it with the original natural gradient theoretically and numerically. In Section 2 the IGO framework and the rank-μ update CMA-ES as an instantiation of the IGO are revisited. In Section 3 we derive the natural gradient under a linearly constrained feasible domain. In Section 4 we analyze the infinite-population model using the derived natural gradient on a linearly constrained spherical problem and perform simulations to compare the behavior of the derived algorithm with the original algorithm on a linearly constrained spherical problem. Finally in Section 5 we summarize this work and discuss required future works.

Notation. The inner product of $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^d$ is denoted by $\langle x, y \rangle$ and the norm of x by $\|x\| = \langle x, x \rangle^{1/2}$. For any matrix A, $[A]_{i,j}$ represents the (i, j)th element, $[A]_{i,:}$ (or $[A]_{:,j}$) the ith row (or the jth column, respectively.) For any symmetric matrix A of dimension d, let $\mathrm{vech}(A)$ denote the lower-left half vectorization of A such that $\mathrm{vech}(A)$ is the $d(d+1)/2$ dimensional column vector whose ith element is $[A]_{m_i,n_i}$ where $i = m_i + (d - n_i/2)(n_i - 1)$ for $1 \leqslant n_i \leqslant m_i \leqslant d$. We refer to [14] for the detail.

2 IGO Framework and the Rank-μ Update CMA

Formulation. We consider a constrained continuous minimization $\mathrm{argmin}_{x \in \mathrm{X}} f(x)$, where $\mathrm{X} \subset \mathbb{R}^d$ is the feasible set and f is the objective function defined over X. In the

following sections we assume that the feasible set is restricted by a linear function, namely $\mathrm{X} = \{x \in \mathbb{R}^d \mid \langle x, v\rangle \geqslant \alpha\}$ for some unit vector $v \in \mathbb{R}^d \setminus \{0\}$ and some $\alpha \in \mathbb{R}$.

Given a family $\mathcal{P}$ of probability distributions P_θ on X parameterized by a real vector $\theta \in \Theta$, the IGO framework formulates the joint problem on the parameter space Θ at each iteration t as follows

$$\theta^{t+1} = \mathrm{argmax}_{\theta\in\Theta} J_{\theta^t}(\theta), \quad \text{where} \quad J_{\theta^t}(\theta) := \int_{\mathrm{X}} W^f_{\theta^t}(x) P_\theta(\mathrm{d}x) \ . \tag{1}$$

Here θ^t is the value of the parameter at iteration t, $W^f_{\theta^t}$ defines the preference that is monotonic to f. The preference is defined based on the probability of sampling a better point; namely,

$$W^f_{\theta^t}(x) = w\left(P_{\theta^t}[y \in \mathrm{X} \mid f(y) \leqslant f(x)]\right) \ , \quad \text{where } w : [0,1] \to \mathbb{R}. \tag{2}$$

Another weight scheme is introduced in [1],

$$W^f_{\theta^t}(x) = -\left(\mu_{\mathrm{Leb}}[y \in \mathrm{X} : f(y) \leqslant f(x)]\right)^{2/d}, \tag{3}$$

where μ_{Leb} denotes the Lebesgue measure on $\mathbb{R}^d$. This is theoretically attractive; on an unconstrained monotonic convex quadratic composite function $g(x^{\mathrm{T}}Ax)$ with g strictly increasing, this weight value is $-cx^{\mathrm{T}}Ax$, where c is a constant independent of g, m, and C, and it enables us to derive the exact $J_{\theta^t}(\theta)$.

Natural Gradient. The *natural gradient* can be interpreted as the gradient of a function defined on the space of the probability distribution equipped with the Fisher metric. It can be also interpreted as the steepest ascent direction of the function with respect to the KL-divergence. Since the Fisher metric (and KL-divergence) is independent of the parameterization (coordinate system) of the probability distribution, the natural gradient is invariant to any re-parameterization of θ. Given a parameterization θ, the natural gradient is computed by the product of the inverse of the Fisher information matrix of θ and the vanilla gradient (gradient on the Euclidean space) of the log-likelihood of the probability distribution. We refer to [16] for further properties of the natural gradient.

Noting that $W^f_{\theta^t}$ in (2) or (3) is independent of θ, the natural gradient of J_{θ^t} is computed by

$$\tilde{\nabla} J_{\theta^t}(\theta) = \int_{\mathrm{X}} W^f_{\theta^t}(x) \tilde{\nabla} l(\theta; x) P_\theta(\mathrm{d}x) \ , \tag{4}$$

where $\tilde{\nabla}$ represents the map from a function to its natural gradient, and $l(\theta; x) = \ln p_\theta(x)$ denotes the log-likelihood at θ given x. Eq (4) is viewed as a weighted expectation of the natural gradient of the log-likelihood at θ.

Implementation of the Natural Gradient Ascent. The IGO algorithm performs the natural gradient ascent instead of exactly solving joint problem (1). Then iterate $\{\theta^t\}$ is defined by

$$\theta^{t+1} = \theta^t + \eta^t \tilde{\nabla} J_{\theta^t}(\theta)|_{\theta=\theta^t} \ , \tag{5}$$

where η^t denotes the step-size for the natural gradient ascent, aka the learning rate for the parameter update, which is sometimes replaced with a diagonal matrix whose diagonal entries are the learning rates for each element of the parameter vector. However, the integration in (4) cannot be performed analytically in advance unless f is known.

Therefore, we estimate (4) with samples $x_1, \ldots, x_\lambda$ drawn from P_{θ^t}. According to [16], we can approximate $W^f_{\theta^t}$ in (2) for each x_i as $W^f_{\theta^t}(x_i) \approx \hat{w}_{\mathrm{rk}(x_i)} = w((\mathrm{rk}(x_i) - 1/2)/\lambda)$, where $\mathrm{rk}(x_i)$ denotes the ranking of $f(x_i)$ among $f(x_1), \ldots, f(x_\lambda)$. With this, a Monte-Carlo estimate provides an approximation of (4) at $\theta = \theta^t$, namely

$$\tilde{\nabla} J_{\theta^t}(\theta)|_{\theta=\theta^t} \approx \frac{1}{\lambda} \sum_{i=1}^{\lambda} \hat{w}_{\mathrm{rk}(x_i)} \tilde{\nabla} l(\theta^t; x_i) \ . \tag{6}$$

The IGO implementation performs the natural gradient ascent (5) with replacing the natural gradient $\tilde{\nabla} J_{\theta^t}(\theta)|_{\theta=\theta^t}$ given in (4) with its approximation (6).

Rank-μ Update CMA. Considering the IGO implementation for unconstrained continuous optimization, i.e. $\mathrm{X} = \mathbb{R}^d$, with the Gaussian distributions on $\mathbb{R}^d$, it is known from [3] that the natural gradient of the log-likelihood of the Gaussian distribution is explicitly written in a special form. If the Gaussian distribution is parameterized by $\theta = [m^\mathrm{T}, \ \mathrm{vech}(C)^\mathrm{T}]$, where m and C are the mean vector and the covariance matrix, the parameter update in the IGO implementation reads

$$\begin{aligned} m^{t+1} &= m^t + \frac{\eta_m}{\lambda} \sum_{i=1}^{\lambda} \hat{w}_{\mathrm{rk}(x_i)}(x_i - m) \\ C^{t+1} &= C^t + \frac{\eta_C}{\lambda} \sum_{i=1}^{\lambda} \hat{w}_{\mathrm{rk}(x_i)}((x_i - m)(x_i - m)^\mathrm{T} - C) \ . \end{aligned} \tag{7}$$

This is called the rank-μ update [10] and is a component of the standard CMA [9].

In [3], $\mathrm{X} = \mathbb{R}^d$ is assumed to obtain the explicit form for the natural gradient. In other words, the natural gradient computed in the reference is the one on the manifold of (non-truncated) Gaussian distributions defined on $\mathbb{R}^d$. If X is a proper subset of $\mathbb{R}^d$ ($\mathrm{X} \subset \mathbb{R}^d$ and $\mathrm{X} \neq \mathbb{R}^d$) and the truncated Gaussian distribution is considered (sampling from a Gaussian distribution with resampling scheme as a constraint handling), the natural gradient on the manifold of the truncated Gaussian distributions on X is different from the one derived in [3] and the resulting natural gradient ascent differs from (7). This is the main concern of the paper.

3 Natural Gradient for Truncated Gaussian Distributions

Let $p_\theta(x)$ and $l(\theta; x)$ be the probability density function (p.d.f.) and the log-likelihood function (l.l.f.) induced by the Gaussian distribution P_θ with mean $m = m(\theta)$ and covariance matrix $C = C(\theta)$, i.e., $l(\theta; x) = -\frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln\det(C(\theta)) - \frac{1}{2}(x - m(\theta))^\mathrm{T} C^{-1}(\theta)(x - m(\theta))$ and $p_\theta = \exp(l(\theta; x))$. Then, $P_\theta(A) = \int_A p_\theta(x)\mathrm{d}x$ for any Lebesgue measurable $A \subset \mathbb{R}^d$. As in the rank-μ update CMA-ES, we consider $\theta = [m^\mathrm{T}, \ \mathrm{vech}(C)^\mathrm{T}]^\mathrm{T}$.

If the X is a proper subset of $\mathbb{R}^d$ and the resampling strategy is employed, the distribution of the samples in X is the Gaussian distribution truncated on X. Let $\bar{p}_\theta(x)$ and $\bar{l}(\theta; x)$ be the p.d.f. and l.l.f. of such a truncated Gaussian distribution $\bar{P}_\theta$ over X. Then, $\bar{p}_\theta(x) = p_\theta(x)/P_\theta(\mathrm{X})$ and $\bar{l}(\theta; x) = l(\theta; x) - \ln P_\theta(\mathrm{X})$ for $x \in \mathrm{X}$, where $P_\theta(\mathrm{X}) = \int_\mathrm{X} p_\theta(x)\mathrm{d}x = \mathbb{E}_{x\sim p_\theta}[\mathbb{I}\{x \in \mathrm{X}\}]$ is the probability of x being sampled in X from p_θ. If we implement an algorithm following the IGO framework, the natural gradient on the manifold of the truncated Gaussian distributions $\{\bar{P}_\theta \mid \theta \in \Theta\}$ is needed.

As the first attempt of the work, we derive the natural gradient of the l.l.f. of the Gaussian distribution truncated on $\mathrm{X} = \{x \in \mathbb{R}^d \mid \langle x, v\rangle \geqslant \alpha\}$. The following proposition and theorem provide the formula to compute the natural gradient explicitly.

Proposition 1. *Let φ and Φ be the p.d.f. and cumulative density function induced by the standard normal distribution $\mathcal{N}(0,1)$. Define φ_β be the p.d.f. for the normal distribution truncated onto $\{x \geqslant \beta\}$, that is, $\varphi_\beta(x) = \varphi(x)/(1-\Phi(\beta))$. Let N_β be a random variable obeying φ_β. Then, $\mu_1 := \mathbb{E}[N_\beta] = \varphi_\beta(\beta)$, $\mu_2 := \mathbb{E}[N_\beta^2] = \beta\varphi_\beta(\beta) + 1$, $\mu_3 := \mathbb{E}[N_\beta^3] = (\beta^2+2)\varphi_\beta(\beta)$ and $\mu_4 := \mathbb{E}[N_\beta^4] = (\beta^3+3\beta)\varphi_\beta(\beta) + 3$.*

Theorem 1. *Let the natural gradient $\tilde{\nabla}\bar{l}(\theta; x)$ of the l.l.f. of the truncated Gaussian distribution decomposed as $\tilde{\nabla}\bar{l}(\theta; x) = [\delta m(x)^{\mathrm{T}},\ \mathrm{vech}(\delta C(x))^{\mathrm{T}}]^{\mathrm{T}}$, where $\delta m(x) \in \mathbb{R}^d$ and $\delta C(x) \in \mathbb{R}^{d\times d}$ are the components corresponding to m and C respectively. Let $u = v/\|C^{1/2}v\|$ and define $y = x - m$. Then,*

$$\delta m(x) = \left[\frac{\mu_2 - \mu_1\langle u, y\rangle}{\tau_1}\right]y + \left[-\left(\frac{\tau_2}{\tau_1\tau_3 - \tau_2^2} - \frac{\mu_1}{\tau_1}\right)\langle u, y\rangle^2 + \left(\frac{\tau_3}{\tau_1\tau_3 - \tau_2^2} - \frac{\mu_2}{\tau_1}\right)\langle u, y\rangle - \frac{\tau_3\mu_1 - \tau_2\mu_2}{\tau_1\tau_3 - \tau_2^2}\right]Cu \quad \textit{and} \tag{8}$$

$$\delta C(x) = yy^{\mathrm{T}} - C + \left[\frac{(1-\tau_1)\langle u, y\rangle + \mu_1}{\tau_1}\right](yu^{\mathrm{T}}C + Cuy^{\mathrm{T}}) + \left[\left(2\frac{\tau_2\mu_1 - \tau_1\mu_2}{\tau_1\tau_3 - \tau_2^2} + 1\right) + \left(\frac{2\tau_1}{\tau_1\tau_3 - \tau_2^2} - \frac{2}{\tau_1} + 1\right)\langle u, y\rangle^2 - 2\left(\frac{\tau_2}{\tau_1\tau_3 - \tau_2^2} + \frac{\mu_1}{\tau_1}\right)\langle u, y\rangle\right]Cuu^{\mathrm{T}}C\ , \tag{9}$$

where μ_1, μ_2, μ_3 and μ_4 are as defined in Proposition 1 with $\beta = (\alpha - \langle v, m\rangle)/\|C^{1/2}v\|$, and $\tau_1 = \mu_2 - \mu_1^2$, $\tau_2 = \mu_3 - \mu_1\mu_2$, $\tau_3 = \mu_4 - \mu_2^2$.

We have omitted its proof due to the space limitation. Comparing to the natural gradient $\tilde{\nabla}l(\theta; x)$ of the l.l.f. for the non-truncated Gaussian distribution P_θ that can be expressed as $\delta m(x) = y$ and $\delta C(x) = yy^{\mathrm{T}} - C$, (8) and (9) have additional components characterized by $Cu = Cv/\|C^{1/2}v\|$. The coefficients are determined by β—a signed distance to the boundary normalized by the standard deviation $\|C^{1/2}v\|$ in the direction of v—and $\langle u, y\rangle$—a signed distance from the current mean m to the sample point x in the direction of $C^{1/2}v$. In the limit $\beta \to -\infty$, meaning that the constraint boundary is far away from the current mean and the situation is close to the unconstrained case, we have from Proposition 1 that $\mu_1 = \mu_3 = \tau_2 = 0$, $\mu_2 = \tau_1 = 1$, $\mu_4 = 3$, $\tau_3 = 2$, and (8) and (9) become identical to the natural gradient for the unconstrained case.

The natural gradient of the l.l.f. only depends on the manifold of the probability distributions. That is, it only depends on the feasible set X, but not on the objective function f. The parameter update (5) with (6) on the other hand depends on the selection scheme. More precisely, the adjustments $\overline{\delta m}$ and $\overline{\delta C}$ of the parameters is the weighted sum of $\delta m(x_i)$ and $\delta C(x_i)$ over $i = 1, \ldots, \lambda$, where the weight value is determined by the ranking of $f(x_i)$. In the next section we demonstrate on a linearly constrained spherical problem how much the derived natural gradient differs from the rank-μ update (7).

4 Study on a Linearly Constrained Spherical Problem

We consider the following linearly constrained spherical problem $\mathrm{argmin}_{x\in \mathrm{X}_\alpha} f(x) := g(\|x\|^2)$, where $\mathrm{X}_\alpha = \{x \in \mathbb{R}^d \mid \langle x, v\rangle \geqslant \alpha\}$ and g is strictly increasing. If $\alpha \geqslant 0$, the optimum is located on the boundary $x^* = \alpha v$, otherwise $x^* = (0, \ldots, 0)$ and the landscape

around the optimum is the same as the unconstrained sphere function. Therefore, we consider only $\alpha \geqslant 0$ in this work.

For ranking-based weight scheme as in (2), the natural gradient (4) generally needs to be approximated by (6). To understand and emphasize the benefit of use the natural gradient derived in the previous section, we employ the weight scheme (3). As mentioned after (3), we can compute the joint objective analytically, $J_{\theta^t}(\theta) = -c(\|m\|^2 + \mathrm{Tr}(C))$, on the unconstrained spherical problem and the natural gradients become $\overline{\delta m} = -2cCm$ and $\overline{\delta C} = -2cC^2$ with an appropriate constant factor c. This analytical natural gradient is the limit of the natural gradient estimate (6) w.r.t. $\lambda \to \infty$ [1] and it models the infinite-population behavior of the rank-μ update CMA (7).

If $\alpha = 0$, the volume of each sub level set $\mu_{\mathrm{Leb}}[y \in \mathrm{X} : f(y) \leqslant f(x)]$ is just halved compared to the unconstrained case and we still have similar results.

Lemma 1. *If $\alpha = 0$, the weight $W^f_{\theta^t}(x)$ defined in* (3) *is $-\tilde{c}x^{\mathrm{T}}x$, where $\tilde{c}$ is a constant independent of m and C. The natural gradient* (4) *with δm and δC derived in Theorem 1 reads $\overline{\delta m} = -2\tilde{c}Cm$ for m and $\overline{\delta C} = -2\tilde{c}C^2$ for C.*

Surprisingly, the natural gradient on the linearly constrained spherical problem is only different in length from the one on unconstrained spherical problem. This implies that the natural gradient update (5) with δm and δC in Theorem 1 reads the exact same parameter update as in the unconstrained case with an appropriate η^t. Therefore, all the results in [1] are carried over here as stated in the next theorem.

Theorem 2. *Let λ^t_1 denote the largest eigenvalue of $-C^{-1/2}\overline{\delta C}C^{-1/2}$. If C^0 is symmetric positive definite and $\eta^t\lambda^t_1 < 1$ for all $t \geqslant 0$, then C^t is symmetric positive definite. Moreover, if $\eta^t = \bar{\eta}/\lambda^t_1$ for $\bar{\eta} \in (0, 1/2]$, $\lim_{t\to\infty} \mathrm{Cond}(C^t) = 1$ and $\lim_{t\to\infty}\|C^{t+1}\|_F/\|C^t\|_F = \lim_{t\to\infty}\|m^{t+1}\|/\|m^t\| = 1 - \bar{\eta}$, where $\|\cdot\|_F$ denotes the Frobenius norm.*

The learning rate $\eta^t = \bar{\eta}/\lambda^t_1$ is taken from [1]. This theorem means, the C becomes proportional to the Hessian of $x^{\mathrm{T}}x$, namely the identity matrix, and m linearly converges towards the global optimum at the origin. For the detail, see [1].

To visualize the difference from the original parameter update (7) with $\lambda = \infty$ where the adjustment is $\mathbb{E}[W^f_{\theta^t}(x)(x - m)]$ and $\mathbb{E}[W^f_{\theta^t}(x)((x - m)(x - m)^{\mathrm{T}} - C)]$, we derive the explicit form of the expectation. Following proposition reads it when the weight scheme with baseline subtraction, $W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)]$, is introduced.

Proposition 2. *Let $y = x - m$, $u = v/\|C^{1/2}v\|$ and μ_1, μ_2, τ_1, τ_2 and τ_3 be as appeared in Theorem 1. If $\alpha = 0$, $\mathbb{E}[W^f_{\theta^t}(x)] = -\tilde{c}[\mathrm{Tr}(C) + \|m\|^2 + 2\mu_1 u^{\mathrm{T}}Cm + (\mu_2 - 1)u^{\mathrm{T}}C^2u]$, $\mathbb{E}[y] = \mu_1 Cu$, $\mathbb{E}[yy^{\mathrm{T}} - C] = (\mu_2 - 1)Cuu^{\mathrm{T}}C$, and*

$$\begin{aligned}
&\mathbb{E}[(W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)])y] \\
&\quad = -\tilde{c}[(\tau_2 - 2\mu_1)(u^{\mathrm{T}}C^2u) + 2(\tau_1 - 1)(u^{\mathrm{T}}Cm)]Cu - 2\tilde{c}\mu_1 C^2u - 2\tilde{c}Cm \quad \textit{and} \qquad (10)\\
&\mathbb{E}[(W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)])(yy^{\mathrm{T}} - C)] \\
&\quad = -2\tilde{c}C^2 - \tilde{c}[(\tau_3 - 4\mu_2 + 2)(u^{\mathrm{T}}C^2u) + 2(\tau_2 - 2\mu_1)(u^{\mathrm{T}}Cm)]Cuu^{\mathrm{T}}C \\
&\qquad - 2\tilde{c}(\mu_2 - 1)(C^2uu^{\mathrm{T}}C + Cuu^{\mathrm{T}}C^2) - 2\tilde{c}\mu_1(Cmu^{\mathrm{T}}C + Cum^{\mathrm{T}}C)\ . \qquad (11)
\end{aligned}$$

Note that the $\mathbb{E}[W^f_{\theta^t}(x)y] = \mathbb{E}[(W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)])y] + \mathbb{E}[W^f_{\theta^t}(x)]\,\mathbb{E}[y]$ and $\mathbb{E}[W^f_{\theta^t}(x)(yy^{\mathrm{T}} - C)] = \mathbb{E}[(W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)])(yy^{\mathrm{T}} - C)] + \mathbb{E}[W^f_{\theta^t}(x)]\,\mathbb{E}[yy^{\mathrm{T}} - C]$. We call them NG_n

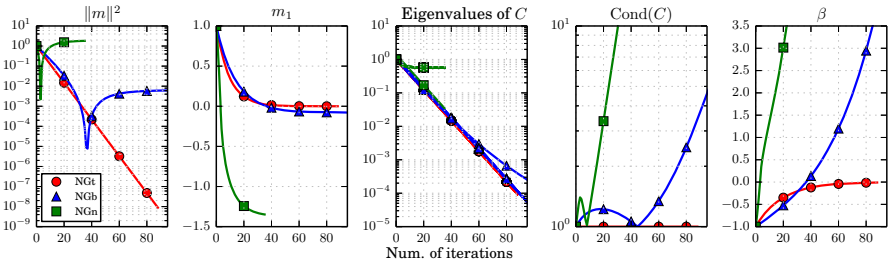

Fig. 1. Transitions of $\|m\|^2$, m_1, the eigenvalues of C, the condition number Cond(C), and β from left to right, respectively. For NG_n, the run stops after β reaches 7. In the center figure, the eigenvalue corresponds to the first coordinate is the larger one for NG_b, and the smaller one for NG_n.

(natural gradient computed on the non-truncated Gaussian manifold), in contrast to NG_t (natural gradient computed on the truncated Gaussian manifold) for $\overline{\delta m}$ and $\overline{\delta C}$ derived in Lemma 1. Moreover, we call (10) and (11) NG_b. The only difference between NG_n and NG_b is the offset of the weight, $-\mathbb{E}[W^f_{\theta^t}(x)]$, and the expectation of the weight in NG_b is forced to be zero. Note that this offset does not affect the natural gradient in NG_t derived Lemma 1 since the expectations of $\delta m(x)$ and $\delta C(x)$ taken over x are zero.

Fig. 1 illustrates the evolution of the parameters m and C following the natural gradient update (5) with the natural gradient $\tilde{\nabla} J_{\theta^t}(\theta) = [\overline{\delta m}^{\mathrm{T}}, \mathrm{vech}(\overline{\delta C})^{\mathrm{T}}]^{\mathrm{T}}$ where $\overline{\delta m}$ and $\overline{\delta C}$ are computed for NG_t, NG_n, and NG_b. For the constraint we set $v = e_1 = [1, 0, \ldots, 0]^{\mathrm{T}}$ and $\alpha = 0$. The step-size is $\eta^t = \bar{\eta}/\lambda^t_1$, where $\bar{\eta} = 0.1$ and λ^t_1 is the largest eigenvalue of $-C^{-1/2}\overline{\delta C}C^{-1/2}$ with corresponding $\overline{\delta C}$ for each variant. This step-size setting guarantees the positivity of the covariance matrix as stated in Theorem 2 for NG_t. The problem dimension is $d = 10$. To produce simple figures, the evolution starts from $m^0 = e_1$ and $C^0 = I$. Thanks to the symmetry, m stays on the first axis and C stays to be a diagonal matrix whose second to the last diagonal elements are equal.

As stated in Theorem 2, m and C converge linearly in NG_t while the condition number of C stays 1 forever. In contrast, m goes over the constraint boundary and tends to stop at some point in the infeasible area while the condition number of C grows up in NG_b and NG_n. The normalized and signed distance β from m to the constraint boundary then becomes large, which in the actual algorithms means that the probability of sampling a point in the feasible region decreases. Since m does not converge towards the optimum, the best-so-far point would not converge linearly towards the global optimum. The tendency of the plots does not depend on the choice of the learning rate, i.e. β does not converge to zero with any learning rate in NG_n and NG_b.

So far the natural gradient is analytically computed. This is considered the approximated behavior of the algorithm in the limit of $\lambda \to \infty$. In practice the population size $\lambda < \infty$ and $W^f_{\theta^t}(x)$ for each x_i and then $\overline{\delta m}$ and $\overline{\delta C}$ must be estimated using finite samples $x_1, \ldots, x_\lambda \sim \bar{p}_{\theta^t}(x)$.[1] We can approximate $W^f_{\theta^t}(x)$ as $\hat{W}^f_{\theta^t}(x) :=$

[1] The resampling can be performed efficiently as follows. Generate $z \sim \mathcal{N}(0, I_d)$. If $\langle z, C^{1/2}u\rangle < \beta$, generate $\tilde{z} \sim \mathcal{N}(0,1)$, resample it till $\tilde{z} \geqslant \beta$, then update $z = z + (\tilde{z} - \langle z, C^{1/2}u\rangle)C^{1/2}u$. Then, $m + C^{1/2}z$ obeys $\bar{p}_\theta$. So we only need to resample a standard normal random number $\tilde{z}$.

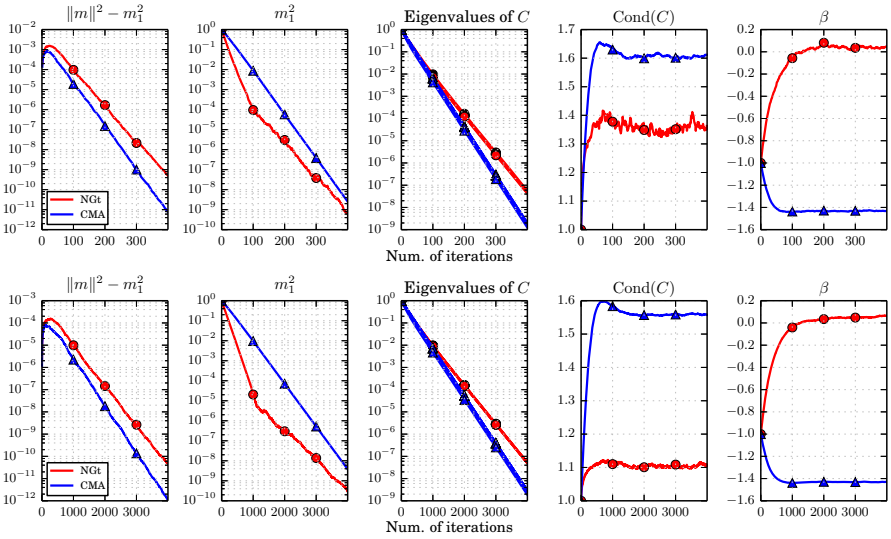

Fig. 2. Transitions of $\|m\|^2 - m_1^2$, m_1^2, the eigenvalues of C, the condition number Cond(C), and β for NG$_t$ and the rank-μ update CMA with $\eta_m = \eta_C = \bar{\eta} = 0.1$ (above) and $\eta_m = \eta_C = \bar{\eta} = 0.01$ (below). The graphs are the average over 30 trials.

$\left[\sum_{j\in\{k\in[\![1;\lambda]\!]|f(x_k)\leqslant f(x)\}}(1/\bar{p}_{\theta^t}(x_j))\right]^{2/d}$, where $\bar{p}_{\theta^t}(x) = p_{\theta^t}(x)/P_{\theta^t}(\mathrm{X})$ and $P_{\theta^t}(\mathrm{X}) = 1 - \Phi(\beta) = [1 - \mathrm{erf}(\beta/\sqrt{2})]/2$.[2] With this approximation we can estimate the weight with baseline subtraction[3], $W^f_{\theta^t}(x) - \mathbb{E}[W^f_{\theta^t}(x)]$, for each x_i as $w_i = \hat{W}^f_{\theta^t}(x_i) - \frac{1}{\lambda}\sum_{j=1}^{\lambda}\hat{W}^f_{\theta^t}(x_j)$. Then, $\overline{\delta m}$ and $\overline{\delta C}$ are approximated by the average of $w_i \cdot \delta m(x_i)$ and $w_i \cdot \delta C(x_i)$, respectively. We denote the estimated natural gradient by $\widehat{\delta m}$ and $\widehat{\delta C}$.

Fig. 2 compares the behaviors of NG$_t$ with that of the rank-μ update CMA (7) (denoted CMA) with the weight used in the standard CMA, $\hat{w}_{\mathrm{rk}(x_i)} = \max(0, \ln(\frac{\lambda+1}{2}) - \ln(\mathrm{rk}(x_i)))/\sum_{j=1}^{\lambda}\max(0, \ln(\frac{\lambda+1}{2}) - \ln(j))$. The problem dimension $d = 10$, the population size $\lambda = 1000$, and the learning rates $\eta_m = \eta_C = 0.1$ and 0.01 for the rank-μ update CMA. For NG$_t$, $\eta^t = \bar{\eta}/\max[\sigma(C^{-1/2}\widehat{\delta C}C^{-1/2}), \|C^{-1/2}\widehat{\delta m}\|]$ with $\bar{\eta} = 0.1$ and 0.01, where $\sigma(\cdot)$ represents the largest singular value. The graphs show the average of 30 independent runs for each method.

As you can see from the figure, in the rank-μ update CMA, β tends to stay at some point in negative, meaning that the mean vector is always away from the constraint boundary in the feasible domain and its distance is proportional to $\|C^{1/2}v\|$. The eigenvalue of C corresponding to $v = e_1$ becomes relatively smaller than the other eigenvalues.

[2] In practice, a scalar factor in $\hat{W}^f_{\theta^t}$ does not matter at all because the natural gradient is multiplied by η^t_m and η^t_C that are inversely proportional to the scalar factor as introduced below. Therefore, we can replace $1/\bar{p}_{\theta^t}(x)$ with $\exp(\|z\|^2/2)$, where $z = C^{-1/2}(x - m)$.

[3] As stated above, the baseline subtraction in NG$_t$ does not affect the expectation of the natural gradient, while it can reduce the estimation variance of the natural gradient.

On the other hand, NG_t results in smaller values of β and $\mathrm{Cond}(C)$ and they get even smaller if we decrease the learning rate or increase the population size.

5 Summary and Discussion

In this paper we derive the natural gradient of the l.l.f. of the truncated Gaussian distribution for linearly constrained optimization problems. Analysis on a linearly constrained spherical problem shows the infinite-population model using the derived natural gradient reads the same update as the exact natural gradient algorithm on a unconstrained spherical problem [1] and all the results proven in the reference hold. The simulation results exhibit different behavior of the derived algorithm and the rank-μ update CMA. The rank-μ update CMA tends to stay in the feasible set and the distance from the constraint boundary stays proportional to $\|C^{1/2}v\|$, where v is the normal vector of the constraint boundary, whereas the condition number and the normalized distance in the derived algorithm converges to smaller values than in the rank-μ udpate CMA.

We would like to remark that the simulation performed in Section 4 depends heavily on the weight scheme. As we see in Fig. 1 and Fig. 2, NG_n, NG_b, and the rank-μ update CMA result in different behavior, although their only difference is the weight value and the learning rate. Moreover, from a preliminary experiment we have observed that NG_t does not work as well as it is with the weight scheme (3) if we employ the CMA-type weight scheme or the fitness proportional weight $\hat{W}^f_{\theta^t}(x_i) = \|x\|^2$. The mean vector enters the infeasible region as we have observed in NG_n. Especially for the fitness proportional weight, we can derive a theoretical result for the infinite-population model that even if $\alpha > 0$, the natural gradient becomes the same as the one on the unconstrained sphere problem and the mean vector tends to converges to the origin that is in the infeasible domain. Further study on the weight scheme is highly required.

For other future works, we compare the derived algorithm with the existing treatment for the constrained problem such as [6]. To enhance the performance, we would need to incorporate a step-size control mechanism that is in general heavily affected by the constraint, and a projection of the mean vector to the feasible domain when it reaches the infeasible domain. Furthermore, we extend the formula for the natural gradient for a linearly constrained problem stated in Theorem 1 to problems with more general constraint.

Acknowledgments. This work is supported by JSPS KAKENHI Grant Number 25880012.

References

1. Akimoto, Y.: Analysis of a Natural Gradient Algorithm on Monotonic Convex-Quadratic-Composite Functions. In: Genetic and Evolutionary Computation Conference, pp. 1293–1300 (2012)
2. Akimoto, Y., Nagata, Y., Ono, I., Kobayashi, S.: Bidirectional relation between CMA evolution strategies and natural evolution strategies. In: Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G. (eds.) PPSN XI. LNCS, vol. 6238, pp. 154–163. Springer, Heidelberg (2010)

3. Akimoto, Y., Nagata, Y., Ono, I., Kobayashi, S.: Theoretical Foundation for CMA-ES from Information Geometry Perspective. Algorithmica 64, 698–716 (2012)
4. Arnold, D.V.: Analysis of a repair mechanism for the $(1, \lambda)$-ES applied to a simple constrained problem. In: Genetic and Evolutionary Computation Conference, pp. 853–860 (2011)
5. Arnold, D.V.: On the behaviour of the $(1, \lambda)$-ES for a simple constrained problem. In: Foundations of Genetic Algorithms, pp. 15–24 (2011)
6. Arnold, D.V., Hansen, N.: A (1 + 1)-CMA-ES for constrained optimisation. In: Genetic and Evolutionary Computation Conference Conference, pp. 297–304 (2012)
7. Glasmachers, T., Schaul, T., Sun, Y., Wierstra, D., Schmidhuber, J.: Exponential Natural Evolution Strategies. In: Genetic and Evolutionary Computation Conference, pp. 393–400 (2010)
8. Hansen, N.: Benchmarking a BI-population CMA-ES on the BBOB-2009 noisy testbed. In: Companion on Genetic and Evolutionary Computation Conference (2009)
9. Hansen, N., Auger, A.: Principled Design of Continuous Stochastic Search: From Theory to Practice. In: Borenstein, Y., Moraglio, A. (eds.) Theory and Principled Methods for the Design of Metaheuristics. Springer (2013)
10. Hansen, N., Muller, S.D., Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evolutionary Computation 11(1), 1–18 (2003)
11. Hansen, N., Niederberger, A.S.P., Guzzella, L., Koumoutsakos, P.: A Method for Handling Uncertainty in Evolutionary Optimization With an Application to Feedback Control of Combustion. IEEE Transactions on Evolutionary Computation 13(1), 180–197 (2009)
12. Hansen, N., Ostermeier, A.: Completely derandomized self-adaptation in evolution strategies. Evolutionary Computation 9(2), 159–195 (2001)
13. Harada, K., Sakuma, J., Ono, I., Kobayashi, S.: Constraint-handling method for multi-objective function optimization: Pareto descent repair operator. In: Obayashi, S., Deb, K., Poloni, C., Hiroyasu, T., Murata, T. (eds.) EMO 2007. LNCS, vol. 4403, pp. 156–170. Springer, Heidelberg (2007)
14. Harville, D.A.: Matrix Algebra from a Statistician's Perspective. Springer (2008)
15. Kramer, O.: A review of constraint-handling techniques for evolution strategies. Applied Computational Intelligence and Soft Computing 2010 (2010)
16. Ollivier, Y., Arnold, L., Auger, A., Hansen, N.: Information-geometric optimization algorithms: A unifying picture via invariance principles (2011), http://arxiv.org/abs/1106.3708
17. Rechenberg, I.: Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Frommann-Holzboog (1973)