Bidirectional Relation between CMA Evolution Strategies and Natural Evolution Strategies

Youhei Akimoto[1,2], Yuichi Nagata[1], Isao Ono[1], and Shigenobu Kobayashi[1]

[1] Tokyo Institute of Technology,
G5-21, 4259 Nagatsuta-cho, Midori-ku, Yokohama, 226-8502, Japan
{akimoto@fe., nagata@fe., isao@, kobayasi@}dis.titech.ac.jp
[2] Research Fellow of the Japan Society for the Promotion of Science

Abstract. This paper investigates the relation between the covariance matrix adaptation evolution strategy and the natural evolution strategy, the latter of which is recently proposed and is formulated as a natural gradient based method on the expected fitness under the mutation distribution. To enable to compare these algorithms, we derive the explicit form of the natural gradient of the expected fitness and transform it into the forms corresponding to the mean vector and the covariance matrix of the mutation distribution. We show that the natural evolution strategy can be viewed as a variant of covariance matrix adaptation evolution strategies using Cholesky update and also that the covariance matrix adaptation evolution strategy can be formulated as a variant of natural evolution strategies.

1 Introduction

Recently in the field of continuous function optimization, natural evolution strategies (NESs) have been proposed by Wierstra et al. [1] and developed by Sun et al. [2]. The NES utilizes the Gaussian mutation to generate new search points and adjusts the parameter of the mutation at each generation in order to improve the expected fitness under the mutation distribution. For the adjustment, the NES makes use of the natural gradient [3] of the expected fitness with respect to the parameter of the mutation distribution, which is referred to as a *natural evolution gradient*. Sun et al. [2] reported that the performance of the NES is competitive to that of the covariance matrix adaptation evolution strategy (CMA-ES, e.g. [4, 5]) on standard benchmarks.

Now an interesting question arises as to why they perform similarly despite their apparently different update rules for the parameters of the mutation distribution. Investigating the relation between the NES and the CMA-ES is beneficial to understand the algorithms. Comparing the NES with the CMA-ES, which is more intuitively understandable in terms of the update rules of the parameters of the mutation distribution, helps to understand how the NES adjusts the parameter of the mutation distribution. Describing the CMA-ES in the framework of the NES, which seems theoretically more tractable, allows deriving the convergence theory.

We investigate the relation between the NES and the CMA-ES. The rest of the paper is organized as follows. In section 2, we explain the concepts of the *evolution gradient* – the gradient of the expected fitness – and of the natural evolution gradient. In section 3, we elucidate the connection between the NES and the CMA-ES by deriving the explicit form of the natural evolution gradient, which is computed in [2] with an iterative computation of the inversion of the Fisher information matrix (FIM) of the mutation distribution. In section 4, we show that the CMA-ES employing global weighted recombination and rank-μ update without step-size adaptation can be formulated as a variant of natural evolution gradient based methods. Finally in section 5, we discuss the results and conclude this paper.

2 Formulation

The objective of minimization is to find the point $\mathbf{x}$ at which an objective function $f : \mathbb{R}^d \to \mathbb{R}$ has the minimal value. Both the NES and the CMA-ES search the optimal point via Gaussian mutation. Their algorithms repeat two steps at each generation: mutation step and update step. At the mutation step, their algorithms generate new points from a Gaussian distribution with mean $\mathbf{m}$ and covariance matrix $\mathbf{C}$. At the update step, their parameters $\theta = \langle \mathbf{m}, \mathbf{C} \rangle$ are updated to promote promising mutation by using the sample points. The update of θ in the NES, as we mention in the following subsections, is based on the gradient of the expected fitness under the mutation distribution, while that in the CMA-ES is related to the maximum likelihood estimation of the Gaussian distribution.

2.1 Evolution Gradient

The NES adjusts θ to optimize the expected fitness $J(\theta) = \mathbb{E}[f(\mathbf{x}) \mid \theta]$ of the next generation under the mutation distribution $\pi(\mathbf{x} \mid \theta)$ by using the natural evolution gradient – the natural gradient of $J(\theta)$. Preparatory to introducing the notion of the natural evolution gradient, we introduce the concept of the evolution gradient.

One of the most straightforward approaches to adjusting θ relies on the gradient $\nabla_\theta J(\theta)$ of $J(\theta)$. Let

$$\pi(\mathbf{x} \mid \theta) = \frac{1}{(2\pi)^{d/2} \det(\mathbf{C})^{1/2}} \exp\Big(-\frac{1}{2}(\mathbf{x} - \mathbf{m})^{\mathrm{T}} \mathbf{C}^{-1} (\mathbf{x} - \mathbf{m})\Big) \tag{1}$$

denote the probability density of the Gaussian distribution given parameter $\theta = \langle \mathbf{m}, \mathbf{C} \rangle$. The expected fitness under the mutation distribution $\pi(\mathbf{x} \mid \theta)$ is

$$J(\theta) = \int f(\mathbf{x}) \pi(\mathbf{x} \mid \theta) d\mathbf{x} \ . \tag{2}$$

Using the log-likelihood trick, we can express the gradient $\nabla_\theta J(\theta)$ with respect to θ as

$$\nabla_\theta J(\theta) = \nabla_\theta \int f(\mathbf{x})\pi(\mathbf{x} \mid \theta)d\mathbf{x} = \int \pi(\mathbf{x} \mid \theta) f(\mathbf{x}) \nabla_\theta \ln \pi(\mathbf{x} \mid \theta) d\mathbf{x} \ . \quad (3)$$

This is referred to as an evolution gradient. If the gradient $\nabla_\theta J(\theta^t)$ at the current location θ^t is given, one can update θ^{t+1} by shifting θ^t in the direction of the negative gradient, $-\nabla_\theta J(\theta^t)$, as $\theta^{t+1} = \theta^t - \eta \cdot \nabla_\theta J(\theta^t)$.

However, since the objective function is unknown, so is the evolution gradient. We alternatively utilize the Monte-Carlo approximation:

$$\nabla_\theta J(\theta) \approx \sum_{i=1}^{\lambda} \frac{f(\mathbf{x}_i)}{\lambda} \nabla_\theta \ln \pi(\mathbf{x}_i \mid \theta) \ , \quad (4)$$

where $\mathbf{x}_i$ are samples generated from $\pi(\mathbf{x} \mid \theta)$. To eliminate premature convergence attributed to disturbance of the estimation of evolution gradients and explicit the invariant property against order-preserving transformation, a ranking based fitness shaping is introduced in [1, 2]:

$$(f(\mathbf{x}_1)/\lambda, \ldots, f(\mathbf{x}_\lambda)/\lambda) \to (-\mathrm{w}_{R_1}, \ldots, -\mathrm{w}_{R_\lambda}), \quad \mathrm{w}_1 \geq \cdots \geq \mathrm{w}_\lambda \ . \quad (5)$$

Here the index R_i denotes the rank of $\mathbf{x}_i$ among $\mathbf{x}_1, \ldots, \mathbf{x}_\lambda$ with respect to f-values. That is, $f(\mathbf{x}_i)$ is the R_ith smallest among $f(\mathbf{x}_1), \ldots, f(\mathbf{x}_\lambda)$.

2.2 Natural Evolution Gradient

The natural gradient [3] has a background in information geometry, which studies the Riemannian geometric structure of the manifold of probability distributions. A result in information geometry states that the Fisher information matrix (FIM) defines a Riemannian metric tensor on the space of probability distributions [6] and that the direction of the steepest descent on a Riemannian manifold is given by the natural gradient, which is given by the conventional gradient premultiplied by the inverse matrix of the Riemannian metric tensor [3]. Thus, the natural gradient can be computed from the gradient and the FIM, and the natural gradient descent tends to converge faster than conventional one. Furthermore, the natural gradient descent is a variable metric method, which provides uniform convergence properties. In the field of machine learning, natural gradient learning has been used as an efficient method that can prevent from being stuck on plateaus [7].

The NES utilizes the natural evolution gradient – the natural gradient of the expected fitness – in lieu of the conventional one. If the FIM is invertible, the natural evolution gradient $\tilde{\nabla}_\theta J(\theta) = \mathbf{F}^{-1}(\theta) \nabla_\theta J(\theta)$ is given by the evolution gradient premultiplied by the inverse matrix of the FIM $\mathbf{F}(\theta)$. It is well-known that the FIM for a Gaussian distribution takes an explicit form. The (i, j) element of the FIM $\mathbf{F}(\theta)$ of the Gaussian distribution $\mathcal{N}(\mathbf{m}(\theta), \mathbf{C}(\theta))$ is

$$\mathbf{F}_{i,j} = \frac{\partial \mathbf{m}^{\mathrm{T}}}{\partial \theta_i} \mathbf{C}^{-1} \frac{\partial \mathbf{m}}{\partial \theta_j} + \frac{1}{2} \mathrm{tr}\Big(\mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta_i} \mathbf{C}^{-1} \frac{\partial \mathbf{C}}{\partial \theta_j} \Big). \quad (6)$$

We approximate the natural evolution gradient by replacing the exact gradient with its estimation, namely,

$$\tilde{\nabla}_\theta J(\theta) \approx \delta\theta = -\sum_{i=1}^{\lambda} \mathrm{w}_{R_i} \mathbf{F}^{-1}(\theta) \nabla_\theta \ln \pi(\mathbf{x}_i \mid \theta) \ . \tag{7}$$

Consequently, the NES framework repeats the estimation of the natural evolution gradient $\delta\theta^t$ by using the samples $\mathbf{x}_i^t \sim \pi(\cdot \mid \theta^t)$ generated at tth generation and the adjustment of the parameter by $\theta^{t+1} = \theta^t - \eta\delta\theta^t$. It can be considered that the NES transforms the minimization of $f(\mathbf{x})$ into the minimization of the expected fitness $J(\theta)$ under the mutation distribution $\pi(x \mid \theta)$ and minimizes $J(\theta)$ by using the estimation of the natural gradient of $J(\theta)$.

3 NES as a variant of CMA-ESs

Let $\mathbf{A}$ represent the Cholesky decomposition of the covariance matrix $\mathbf{C}$, or more rigorously, let $\mathbf{A}$ be the unique lower triangular matrix such that $\mathbf{C} = \mathbf{A}\mathbf{A}^{\mathrm{T}}$. Let θ be a $[d(d+3)/2]$-dimensional column vector consisting of the elements of $\mathbf{m}$ and the lower left elements of $\mathbf{A}$, more precisely,

$$\theta = \left[\mathbf{m}^{\mathrm{T}} \ \mathrm{vech}(\mathbf{A})^{\mathrm{T}}\right]^{\mathrm{T}}. \tag{8}$$

Here $\mathrm{vech}(\mathbf{A}) = \left[\, (\mathbf{A}_{1:d,1})^{\mathrm{T}} \ (\mathbf{A}_{2:d,2})^{\mathrm{T}} \ \ldots \ (\mathbf{A}_{d:d,d})^{\mathrm{T}} \,\right]^{\mathrm{T}}$ is a rearranging operator, where $\mathbf{A}_{k:d,k}$ is the sub-matrix in $\mathbf{A}$ at row k to d and column k (see e.g. [8]). In [1, 2], the mutation distribution is parameterized by (8). Sun et al. [2] proved in the case of the parameterization (8) that the exact FIM of $\pi(\mathbf{x} \mid \theta)$ becomes a block-diagonal matrix $\mathrm{diag}(\mathbf{F}_0, \ldots, \mathbf{F}_d)$ whose first block $\mathbf{F}_0$ is identical to $\mathbf{C}^{-1}$ and $k+1$th $(1 \leq k \leq d)$ block $\mathbf{F}_k$ is given by

$$\mathbf{F}_k = a_{k,k}^{-2} \mathbf{u}_k \mathbf{u}_k^{\mathrm{T}} + (\mathbf{C}^{-1})_{k:d,k:d} \ , \tag{9}$$

where $a_{i,i}^{-1}$ is the reciprocal of the ith diagonal element of $\mathbf{A}$, or identically, the ith diagonal element of $\mathbf{A}^{-1}$, and $\mathbf{u}_k$ is a $[d-k+1]$-dimensional column vector whose first element is one and all the other elements are zero. The required matrix inversion can be performed efficiently since $\mathbf{F}^{-1} = \mathrm{diag}(\mathbf{F}_0^{-1}, \ldots, \mathbf{F}_d^{-1})$. Moreover, the inverse of each block and the natural evolution gradient can be computed efficiently by an iterative method proposed in [2].

Although an efficient procedure is vital in implementing it on a computer, it makes it difficult to capture the mechanism of the update of θ. To analyze how the NES adjusts the parameter θ and to compare the NES with the CMA-ES, we derive the analytical inverse matrix of the FIM and extract the explicit form of the natural evolution gradient update rule.

3.1 Inverse of the Fisher Information Matrix

First, we derive the inverse matrix of each diagonal block of the FIM. Obviously, $\mathbf{F}_0^{-1} = \mathbf{C}$. Let $\mathbf{v}_k$ denote a d-dimensional column vector whose kth element is

one and all the other elements are zero, and $\mathbf{I}$ be the $[d-k+1]$-dimensional identity matrix. Then, the inverse matrix of $k+1$th diagonal block $\mathbf{F}_k$ of the FIM can be written as

$$\mathbf{E}_k = \begin{bmatrix}\mathbf{0}\ \mathbf{I}\end{bmatrix} \mathbf{A}\Big(\begin{bmatrix}\mathbf{0}\ \mathbf{0}\\ \mathbf{0}\ \mathbf{I}\end{bmatrix} - \frac{1}{2}\mathbf{v}_k\mathbf{v}_k^{\mathrm{T}}\Big)\mathbf{A}^{\mathrm{T}} \begin{bmatrix}\mathbf{0}\\ \mathbf{I}\end{bmatrix} \ . \tag{10}$$

To see this, it suffices to show the product of $\mathbf{E}_k$ and $\mathbf{F}_k$ becomes $\mathbf{I}_{d+1-k}$. Now, rewriting $\mathbf{F}_k$ in the form

$$\mathbf{F}_k = \begin{bmatrix}\mathbf{0}\ \mathbf{I}\end{bmatrix} \left(\mathbf{A}^{-\mathrm{T}}\mathbf{A}^{-1} + a_{k,k}^{-2}\mathbf{v}_k\mathbf{v}_k^{\mathrm{T}}\right) \begin{bmatrix}\mathbf{0}\\ \mathbf{I}\end{bmatrix} \tag{11}$$

and postmultiplying $\mathbf{F}_k$ by $\mathbf{E}_k$ we have $\mathbf{F}_k\mathbf{E}_k = \mathbf{I}$[3]. Therefore, $\mathbf{F}_k^{-1} = \mathbf{E}_k$.

3.2 Explicit Form of the Natural Evolution Gradient

The estimation $\delta\theta$ of the natural evolution gradient is given as a linear combination of the natural gradient of the log-likelihood for all samples. The partial derivative of the log-likelihood is

$$\frac{\partial}{\partial\theta_k}\ln\pi(\mathbf{x}\mid\theta) = \begin{cases}\mathbf{v}_k^{\mathrm{T}}\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m}) & \text{if } 1\le k\le d,\\ \mathbf{v}_{m_k}^{\mathrm{T}}\mathbf{R}\mathbf{v}_{n_k} & \text{otherwise,}\end{cases} \tag{12}$$

where m_k and n_k are the row and column indices of $\mathbf{A}$ corresponding to the kth element of θ, such that $1\le n_k\le m_k\le d$ and $m_k+\sum_{i=1}^{n_k-1} d+1-i = k-d$, and

$$\mathbf{R} = \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^{\mathrm{T}}\mathbf{A}^{-\mathrm{T}} - \mathrm{diag}(a_{1,1}^{-1},\ldots,a_{d,d}^{-1}) \ . \tag{13}$$

The natural gradient of the log-likelihood is obtained by premultiplying the gradient by the inverse of the FIM, which results in

$$\left[(\mathbf{x}-\mathbf{m})^{\mathrm{T}}\mathbf{C}^{-1}\mathbf{F}_0^{-\mathrm{T}}\ (\mathbf{R}_{1:d,1})^{\mathrm{T}}\mathbf{F}_1^{-\mathrm{T}}\ (\mathbf{R}_{2:d,2})^{\mathrm{T}}\mathbf{F}_2^{-\mathrm{T}}\ \ldots\ (\mathbf{R}_{d:d,d})^{\mathrm{T}}\mathbf{F}_d^{-\mathrm{T}}\right]^{\mathrm{T}} \ . \tag{14}$$

Since $\mathbf{F}_0^{-1} = \mathbf{C}$, $\mathbf{F}_0^{-1}\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m}) = (\mathbf{x}-\mathbf{m})$. For $1\le k\le d$, $\mathbf{R}_{k:d,k} = \begin{bmatrix}\mathbf{0}\ \mathbf{I}\end{bmatrix}\mathbf{R}\mathbf{v}_k$ and

$$\mathbf{F}_k^{-1}(\mathbf{R}_{k:d,k}) = \begin{bmatrix}\mathbf{0}\ \mathbf{I}\end{bmatrix}\mathbf{A}\Big(\mathrm{tril}(\mathbf{S}) - \frac{1}{2}\mathrm{diag}(s_1,\ldots,s_d) - \frac{1}{2}\mathbf{I}\Big)\mathbf{v}_k \ , \tag{15}$$

where $\mathbf{S} = \mathbf{A}^{-1}(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^{\mathrm{T}}\mathbf{A}^{-\mathrm{T}}$, s_k is the kth diagonal element of $\mathbf{S}$, and $\mathrm{tril}(\mathbf{S})$ denotes the lower triangular matrix whose $(i,\ j)$ element is identical to the $(i,\ j)$ element of $\mathbf{S}$ if $i\ge j$, zero otherwise[4].

---

[3] Since $\mathbf{A}$ is lower triangular, $\mathbf{A}^{-1}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] = \mathbf{A}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}^{-1}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] = \left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right]$ and $\mathbf{v}_k^{\mathrm{T}}\mathbf{A}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right] = a_{k,k}\mathbf{v}_k^{\mathrm{T}}$, and then the product $\mathbf{F}_k\mathbf{E}_k$ reduces to $\mathbf{F}_k\mathbf{E}_k = \mathbf{I} + a_{k,k}^{-1}\mathbf{u}_k\mathbf{v}_k^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] - \frac{1}{2}a_{k,k}^{-1}\mathbf{u}_k\mathbf{v}_k^{-\mathrm{T}}\mathbf{A}^{\mathrm{T}}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] - \frac{1}{2}a_{k,k}^{-1}\mathbf{u}_k\mathbf{v}_k^{-\mathrm{T}}\mathbf{A}^{\mathrm{T}}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] = \mathbf{I}$.

[4] According to $\mathbf{A}^{-1}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] = \mathbf{A}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}^{-1}\left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right] = \left[\begin{smallmatrix}\mathbf{0}\\ \mathbf{I}\end{smallmatrix}\right]$, $\mathbf{v}_k^{\mathrm{T}}\mathbf{A}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right] = a_{k,k}\mathbf{v}_k^{\mathrm{T}}$, $\mathrm{diag}(a_{1,1}^{-1},\ldots,a_{d,d}^{-1})\mathbf{v}_k = a_{k,k}^{-1}\mathbf{v}_k$ and $\mathbf{v}_k^{\mathrm{T}}\mathbf{A}^{\mathrm{T}}\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}^{-\mathrm{T}} = \mathbf{v}_k^{\mathrm{T}}$, the product of $\mathbf{F}_k^{-1}$ and $\mathbf{R}_{k:d,k}$ is reduces to $\mathbf{F}_k^{-1}(\mathbf{R}_{k:d,k}) = \left[\begin{smallmatrix}\mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}(\left[\begin{smallmatrix}\mathbf{0}&\mathbf{0}\\ \mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{S} - \frac{1}{2}\mathbf{v}_k\mathbf{S} - \mathbf{I} + \frac{1}{2}\mathbf{I})\mathbf{v}_k = \left[\begin{smallmatrix}\mathbf{0}&\mathbf{I}\end{smallmatrix}\right]\mathbf{A}(\mathrm{tril}(\mathbf{S}) - \frac{1}{2}\mathrm{diag}(s_1,\ldots,s_d) - \frac{1}{2}\mathbf{I})\mathbf{v}_k$.

Letting the estimated natural evolution gradient $\delta\theta$ be expressed in the block form $\delta\theta = -\left[\delta_0^{\mathrm{T}}\ \delta_1^{\mathrm{T}}\ \ldots\ \delta_d^{\mathrm{T}}\right]^{\mathrm{T}}$, then $\delta_0 = \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i - \mathbf{m})$ and

$$\delta_k = \left[\mathbf{0}\ \mathbf{I}\right] \mathbf{A}\Big(\mathrm{tril}(\mathbf{Y}) - \frac{1}{2}\mathrm{diag}(y_1, \ldots, y_d) - \frac{\sum_{i=1}^{\lambda} \mathrm{w}_i}{2}\mathbf{I}\Big)\mathbf{v}_k \tag{16}$$

for $1 \leq k \leq d$. Here $\mathbf{Y} = \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}\mathbf{A}^{-1}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^{\mathrm{T}}\mathbf{A}^{-\mathrm{T}}$ and y_i is the ith diagonal element of $\mathbf{Y}$.

3.3 Parameter Update Rules

We consider the update rules for $\mathbf{m}$ and $\mathbf{A}$ corresponding to $\theta^{t+1} = \theta^t - \eta \cdot \delta\theta^t$. Let $\mathbf{Y}^t = \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}\mathbf{A}^{-1}(\mathbf{x}_i^t - \mathbf{m})(\mathbf{x}_i^t - \mathbf{m})^{\mathrm{T}}\mathbf{A}^{-\mathrm{T}}$, $\delta\mathbf{m}^t = \delta_0$ and $\delta\mathbf{A}^t$ be a $[d \times d]$-dimensional lower triangular matrix whose $(i,\ j)$ element is identical to the $i+1-j$th element of δ_j for $i \leq j$, zero for $i > j$. Then analogous update rules for $\mathbf{m}^{t+1} = \mathbf{m}(\theta^{t+1})$ and $\mathbf{A}^{t+1} = \mathbf{A}(\theta^{t+1})$ can be written as $\mathbf{m}^{t+1} = \mathbf{m}^t + \eta \cdot \delta\mathbf{m}^t$ and $\mathbf{A}^{t+1} = \mathbf{A}^t + \eta \cdot \delta\mathbf{A}^t$, namely,

$$\mathbf{m}^{t+1} = \mathbf{m}^t + \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_j^t - \mathbf{m}^t) \tag{17}$$

$$\mathbf{A}^{t+1} = \mathbf{A}^t + \eta\mathbf{A}^t\Big(\mathrm{tril}(\mathbf{Y}^t) - \frac{1}{2}\mathrm{diag}(y_1^t, \ldots, y_d^t) - \frac{\sum_{i=1}^{\lambda} \mathrm{w}_{R_i}}{2}\mathbf{I}\Big)\ . \tag{18}$$

Suppose that w_i sum to one. The covariance matrix $\mathbf{C}^{t+1} = \mathbf{A}^{t+1}(\mathbf{A}^{t+1})^{\mathrm{T}}$ is

$$\begin{aligned}\mathbf{C}^{t+1} = \mathbf{A}^t\Big(\eta \cdot \mathrm{tril}(\mathbf{Y}^t) - \frac{\eta}{2}\mathrm{diag}(y_1^t, \ldots, y_d^t) + \frac{2-\eta}{2}\mathbf{I}\Big) \\ \cdot \Big(\eta \cdot \mathrm{tril}(\mathbf{Y}^t) - \frac{\eta}{2}\mathrm{diag}(y_1^t, \ldots, y_d^t) + \frac{2-\eta}{2}\mathbf{I}\Big)^{\mathrm{T}}(\mathbf{A}^t)^{\mathrm{T}}\ .\end{aligned} \tag{19}$$

Here, since $\mathbf{Y}^t$ is symmetric, $\mathrm{tril}(\mathbf{Y}^t) + \mathrm{tril}(\mathbf{Y}^t)^{\mathrm{T}} - \mathrm{diag}(y_1^t, \ldots, y_d^t) = \mathbf{Y}^t$, which reduces the last equality to

$$\begin{aligned}\mathbf{C}^{t+1} = (\frac{2-\eta}{2})^2\mathbf{C}^t + \eta\frac{2-\eta}{2}\mathbf{A}^t\mathbf{Y}^t(\mathbf{A}^t)^{\mathrm{T}} + \eta^2\mathbf{A}^t \\ \cdot \Big(\mathrm{tril}(\mathbf{Y}^t) - \frac{1}{2}\mathrm{diag}(y_1^t, \ldots, y_d^t)\Big)\Big(\mathrm{tril}(\mathbf{Y}^t) - \frac{1}{2}\mathrm{diag}(y_1^t, \ldots, y_d^t)\Big)^{\mathrm{T}}(\mathbf{A}^t)^{\mathrm{T}}.\end{aligned} \tag{20}$$

Notice $\mathbf{A}^t\mathbf{Y}(\mathbf{A}^t)^{\mathrm{T}} = \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i^t - \mathbf{m})(\mathbf{x}_i^t - \mathbf{m})^{\mathrm{T}}$. This equality (20) and the update rule (17) together are similar to the update rules for the mean vector and the covariance matrix used in the CMA-ES combining global weighted recombination and rank-μ update except for the third summand of (20) and the learning rate. Since the NES directly updates the Cholesky decomposition $\mathbf{A}$ of the covariance matrix $\mathbf{C}$ and equality (20) is related to rank-μ update, update rule (18) can be considered as a variant of the Cholesky update in [10].

From this explicit form of the NES, we can clearly see the difference between the NES and the CMA-ES. One different point is the third term of equality (20) due to the Cholesky update rule (18). Another point is in the learning late. The third point is the existence of step-size adaptation. We leave it to future work to study how these differences affect the performance of their optimization process.

In addition to enable us to compare the NES with the CMA-ES, the explicit forms (17) and (18) of the parameter update of the NES can reduce the time complexity of a single iteration of the NES from $\mathcal{O}(\lambda d^3)$ to $\mathcal{O}(\lambda d^2 + d^3)$[5]. This is because we can compute the estimated natural evolution gradient without computing the FIM and its inverse, as well as natural actor-critic reinforcement learning [11].

4 CMA-ES as a variant of NESs

The result in the previous section says that the NES can be viewed as a variant of CMA-ESs using Cholesky update when using the parameterization (8). The original NES parameterizes the mutation distribution as (8) to ensure the positivity and symmetry of the covariance matrix. An interesting question is whether there is a parameterization such that more standard CMA-ES which update the covariance matrix rather than the Cholesky factor of it is formulated as a variant of NESs. In this section, we answer the question in the affirmative.

Let

$$\theta = \left[\mathbf{m}^{\mathrm{T}}\ \mathrm{vec}(\mathbf{C})^{\mathrm{T}}\right]^{\mathrm{T}} \tag{21}$$

be a $d(d+1)$-dimensional column vector consisting of all the elements of the mean vector $\mathbf{m}$ and the covariance matrix $\mathbf{C}$, where $\mathrm{vec}(\cdot)$ denotes a rearranging operator from a matrix to a column vector such that $\mathrm{vec}(\left[\mathbf{a}_1\ \mathbf{a}_2\ \dots\ \mathbf{a}_d\right]) = \left[\mathbf{a}_1^{\mathrm{T}}\ \mathbf{a}_2^{\mathrm{T}}\ \dots\ \mathbf{a}_d^{\mathrm{T}}\right]^{\mathrm{T}}$ (see e.g. [8]). Let us consider the natural evolution gradient learning when using this parameterization.

Suppose that $\mathbf{C}$ is positive-definite and symmetric. The gradient of the log-likelihood of $\pi(\mathbf{x} \mid \theta)$ is

$$\nabla_\theta \ln \pi(\mathbf{x} \mid \theta) = \begin{bmatrix} \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m}) \\ \frac{1}{2}\mathrm{vec}(\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^{\mathrm{T}}\mathbf{C}^{-1} - \mathbf{C}^{-1}) \end{bmatrix}. \tag{22}$$

From (6), we have that the FIM of $\pi(\mathbf{x} \mid \theta)$ and its inverse matrix are, respectively,

$$\mathbf{F}(\theta) = \begin{bmatrix} \mathbf{C}^{-1} & \mathbf{0} \\ \mathbf{0} & \frac{1}{2}\mathbf{C}^{-1} \otimes \mathbf{C}^{-1} \end{bmatrix} \quad \text{and} \quad \mathbf{F}^{-1}(\theta) = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & 2\mathbf{C} \otimes \mathbf{C} \end{bmatrix}, \tag{23}$$

where $\otimes$ is the Kronecker product. Therefore, the natural gradient of the log-likelihood of $\pi(\mathbf{x} \mid \theta)$ is

$$\mathbf{F}^{-1}(\theta)\nabla_\theta \ln \pi(\mathbf{x} \mid \theta) = \begin{bmatrix} (\mathbf{x}-\mathbf{m}) \\ \mathrm{vec}((\mathbf{x}-\mathbf{m})(\mathbf{x}-\mathbf{m})^{\mathrm{T}} - \mathbf{C}) \end{bmatrix}. \tag{24}$$

[5] This is only a reduction if λ and d increase at the same time.

Since the natural evolution gradient $\delta\theta$ estimated from the samples $\mathbf{x}_i$ in the same way as (7) is a linear combination of the natural gradient of the log-likelihood, the natural evolution gradient can be estimated by

$$\delta\theta = \begin{bmatrix} -\sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i - \mathbf{m}) \\ -\mathrm{vec}(\sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^{\mathrm{T}} - \sum_{i=1}^{\lambda} \mathrm{w}_i \mathbf{C}) \end{bmatrix} . \tag{25}$$

Therefore, in the case of the parameterization (21), if $\mathbf{C}(\theta^t)$ is symmetric and nonsingular, the natural evolution gradient update produces the next parameter $\theta^{t+1} = \theta^t - \eta\delta\theta^t$ by

$$\theta^{t+1} = \begin{bmatrix} (1 - \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i})\mathbf{m}^t + \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i} \mathbf{x}_i^t \\ \mathrm{vec}((1 - \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i})\mathbf{C}^t + \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i^t - \mathbf{m}^t)(\mathbf{x}_i^t - \mathbf{m}^t)^{\mathrm{T}}) \end{bmatrix} . \tag{26}$$

Suppose $\sum_{i=1}^{\lambda} \mathrm{w}_i = 1$, $\mathrm{w}_i \geq 0$ for all i and $0 \leq \eta < 1$. Starting with an initial parameter θ^0 for which $\mathbf{C}(\theta^0)$ is positive definite and symmetric, the covariance matrix $\mathbf{C}(\theta^t)$ with this update rule is positive definite and symmetric for each t and the supposition that $\mathbf{C}$ is nonsingular and symmetric always holds. By letting $\mathbf{m}^{t+1} = \mathbf{m}(\theta^{t+1})$ and $\mathbf{C}^{t+1} = \mathbf{C}(\theta^{t+1})$, we have

$$\mathbf{m}^{t+1} = (1 - \eta)\mathbf{m}^t + \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i} \mathbf{x}_i^t \quad \text{and} \tag{27}$$

$$\mathbf{C}^{t+1} = (1 - \eta)\mathbf{C}^t + \eta \sum_{i=1}^{\lambda} \mathrm{w}_{R_i}(\mathbf{x}_i^t - \mathbf{m}^t)(\mathbf{x}_i^t - \mathbf{m}^t)^{\mathrm{T}} . \tag{28}$$

These update rules are the same as the CMA-ES combining global weighted recombination and rank-μ update except that $\mathbf{m}^t$ update and $\mathbf{C}^t$ update take a common learning rate η. Consequently, the CMA-ES can be considered as a variant of NESs using the parameterization (21) instead of (8), i.e., the CMA-ES implicitly utilizes the natural evolution gradient without the calculation of the FIM and its inverse matrix.

There are some remarks on the result:

1. In terms of natural gradient, the result justifies the form of the update rules in the CMA-ES, i.e., using $\mathbf{m}^t$ rather than $\mathbf{m}^{t+1}$ in (28) and using the same weights in the update of the covariance matrix as in that of the mean vector.

2. The differences $(\mathbf{C}^{t+1} - \mathbf{C}^t)/\eta$ in (20) and in (28) agree in the case $\eta \to 0$. This is because they represent the natural gradient and the natural gradient is independent of the choice of the parameterization. However, since a finite step ($\eta > 0$) in the natural gradient direction depends on the parameterization, the update rules do not agree for $\eta > 0$.

3. Since two partial derivatives $\frac{\partial}{\partial\theta_k} \int f(\mathbf{x})\pi(\mathbf{x} \mid \theta)d\mathbf{x}$ and $\frac{\partial}{\partial\theta_k} \int (f(\mathbf{x}) - b_k)\pi(\mathbf{x} \mid \theta)d\mathbf{x}$ agree for any b_k, $\sum_{i=1}^{\lambda}(f(\mathbf{x}_i)/\lambda\mathbf{I} - \mathrm{diag}(b_1, \ldots, b_d))\nabla_\theta \ln \pi(\mathbf{x}_i \mid \theta)$ is an unbiased estimator of the gradient as well as (4). When $b_i = 0$ for $1 \leq i \leq d$, $b_i = 1/\lambda$ for $d + 1 \leq i \leq d + d^2$, and ranking-based fitness shaping (5) is used,

the resulting update rule for $\mathbf{C}$ changes from (28) and, if $\mathbf{C}$ is positive-definite, the new one is $\mathbf{C}^{t+1} = \mathbf{C}^t + \eta \sum_{i=1}^{\lambda} (\mathrm{w}_{R_i} - 1/\lambda)(\mathbf{x}_i^t - \mathbf{m}^t)(\mathbf{x}_i^t - \mathbf{m}^t)^{\mathrm{T}}$. This is similar to Active-CMA [12] without rank-one update.

4. Fitness shaping is fundamental. If it is not used and the function values nearly vanish ($|f(\mathbf{x}_i)| \ll 1$), the natural gradient does and the parameter is not updated. An affine type fitness shaping $f(\mathbf{x}) \to a \cdot f(\mathbf{x}) + b$ does not affect the direction but does the length of the natural gradient. For example, a fitness shaping $f(\mathbf{x}_i) / \sum_{j=1}^{\lambda} f(\mathbf{x}_i)$ does not affect the direction of the natural gradient, but it normalizes the length in terms of the sum of the weights and shares the property with (5) that their values sum to one. However, in general, ranking-based fitness shaping (5) influences both the length and the direction of the natural gradient. In addition, so does the different learning rates (step size) for the mean vector and the covariance matrix in the CMA-ES. They might be important future works for further theoretical foundation of the CMA-ES.

5 Conclusion

These results state that the CMA-ES with global weighted recombination and rank-μ update can be formulated as the NES using the parameterization (21) and that their update rules corresponding to the mean and the covariance matrix of the mutation distribution are similar. The difference between the NES and the CMA-ES is essentially only in the parametrization of the mutation distribution. This is in agreement with the similar performances of them reported in [2].

The results also say that the CMA-ES can be formulated as a natural gradient learning which adjusts the parameter θ to minimize the average fitness $J(\theta)$ under the mutation distribution $\pi(\cdot \mid \theta)$. The theoretical foundation is important and profitable to justify, to understand the behavior of the CMA-ES, especially to analyse its convergence behavior, because the research dealing with the convergence properties of stochastic gradient learning (e.g. [13, 14]) may help to investigate the convergence behavior of the CMA-ES. Besides, this makes it easier to compare the CMA-ES with gradient based methods. We may be able to draw inspiration from such comparison about when and how evolutionary algorithms perform better than gradient based methods on rugged functions.

Future work would focus on studying how the differences between the NES and the CMA-ES affect the performances of their algorithms. In particular, it is interesting how we can treat the concept of step-size adaptation in the framework. This might lead to further understanding of the CMA-ES and could possibly help to construct the convergence theory of the CMA-ES. Another future work could be to incorporate ideas used in gradient based online learning into the covariance matrix adaptation. For example, the concepts of *optimal baseline* and *importance sampling* are integrated in the natural evolution strategies as optimal fitness baseline and importance mixing [2]. Furthermore, it is possibly useful to introduce learning rate adaptation [3] and to combine other gradient methods such as natural conjugate gradient methods used in several learning problems [15–17].

Acknowledgement

The authors are grateful to Nikolaus Hansen for his helpful comments.

References

1. Wierstra, D., Schaul, T., Peters, J., Schmidhuber, J.: Natural evolution strategies. In: Proceedings of CEC 2008 (2008) 3381–3387
2. Sun, Y., Wierstra, D., Schaul, T., Schmidhuber, J.: Efficient natural evolution strategies. In: Proceedings of GECCO 2009 (2009) 539–545
3. Amari, S.: Natural gradient works efficiently in learning. Neural Computation **10**(2) (1998) 251–276
4. Hansen, N., Ostermeier, A.: Completely deranomized self-adaptation in evolution strategies. Evolutionary Computation **9**(2) (2001) 159–195
5. Hansen, N., Müller, S., Koumoutsakos, P.: Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). Evolutionary Computation **11**(1) (2003) 1–18
6. Amari, S., Nagaoka, H.: Methods of Information Geometry. American Mathematical Society (2007)
7. Amari, S., Douglas, S.: Why natural gradient? In: Proceedings of ICASSP (1998) 1213–1216
8. Harville, D.A.: Matrix Algebra From a Statistician's Perspective. Springer-Verlag (2008)
9. Murray, M., Rice, J.: Differential geometry and statistics. Chapman & Hall/CRC (1993)
10. Suttorp, T., Hansen, N., Igel, C.: Efficient covariance matrix update for variable metric evolution strategies. Machine Learning **75**(2) (2009) 167–197
11. Peters, J., Schaal, S.: Natural actor-critic. Neurocomputing **71** (2008) 1180–1190
12. Jastrebski, G.A., Arnold, D.V.: Improving evolution strategies through active covariance matrix adaptation. In: Proceedings of CEC 2006 (2006) 9719–9726
13. Kuan, C.M., Hornik, K.: Convergence of learning algorithms with constant learning rates. Neural Networks, IEEE Transactions on **2**(5) (1991) 484–489
14. Kushner, H.J., Yin, G.G.: Stochastic approximation and recursive algorithms and applications. Springer Verlag (2003)
15. Gonzàlez, A., Dorronsoro, J.R.: Natural conjugate gradient training of multilayer perceptrons. Neurocomputing **71**(13-15) (2008) 2499–2506
16. Honkela, A., Tornio, M., Raiko, T., Karhunen, J.: Natural conjugate gradient in variational inference. In: Proceedings of ICONIP 2007. Volume 4985 of LNCS., Springer (2008) 305–314
17. Nishimori, Y., Akaho, S., Plumbley, M.D.: Natural conjugate gradient on complex flag manifolds for complex independent subspace analysis. In: Proceedings of ICANN 2008. Volume 5163 of LNCS., Springer (2008) 165–174