Comparison-Based Natural Gradient Optimization in High Dimension

Youhei Akimoto
Faculty of Engineering,
Shinshu University
y_akimoto@shinshu-u.ac.jp

Anne Auger
INRIA, Research centre
Saclay – Île-de-France
anne.auger@lri.fr

Nikolaus Hansen
INRIA, Research centre
Saclay – Île-de-France
nikolaus.hansen@lri.fr

ABSTRACT

We propose a novel natural gradient based stochastic search algorithm, VD-CMA, for the optimization of high dimensional numerical functions. The algorithm is comparison-based and hence invariant to monotonic transformations of the objective function. It adapts a multivariate normal distribution with a restricted covariance matrix with twice the dimension as degrees of freedom, representing an arbitrarily oriented long axis and additional axis-parallel scaling. We derive the different components of the algorithm and show linear internal time and space complexity. We find empirically that the algorithm adapts its covariance matrix to the inverse Hessian on convex-quadratic functions with an Hessian with one short axis and different scaling on the diagonal. We then evaluate VD-CMA on test functions and compare it to different methods. On functions covered by the internal model of VD-CMA and on the Rosenbrock function, VD-CMA outperforms CMA-ES (having quadratic internal time and space complexity) not only in internal complexity but also in number of function calls with increasing dimension.

Categories and Subject Descriptors

G.1.6 [**Numerical Analysis**]: Optimization—*Global optimization, Gradient methods, Unconstrained optimization*; F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems

Keywords

Covariance Matrix Adaptation, Natural Gradient, Hessian Matrix, Information Geometric Optimization, Theory

1. INTRODUCTION

Natural gradients, popularized in the context of machine learning by Amari [3], have been applied with success in various contexts like training of multilayer perceptron [4], variational inference [13], reinforcement learning [15, 16] or black-box optimization [14, 21]. In the setting of black-box optimization, given a family of probability distributions P_θ, a stochastic search algorithm is defined by an iterative update on P_θ. (This update should lead to convergence of P_θ towards a Dirac-delta distribution concentrated on optima of f.) The natural gradient can be elegantly used to define this update. Indeed, the original minimization problem of finding $\arg\min_{x\in\mathbb{R}^d} f(x)$ can be transformed into a joint optimization problem defined on Θ equipped with the Fisher metric. The joint problem can simply be minimization of the expectation of f over P_θ [7,21]. However this latter criterion is not invariant under monotonic transformations of f and can easily lead to unstable behavior. In order to achieve an invariant algorithm, a better choice for the joint problem is $\arg\max_{\theta\in\Theta} J(\theta)$ with $J(\theta) := \int_{x\in\mathbb{R}^d} W \circ f(x)\,\mathrm{d}P_\theta$, where $W \circ f$ is a θ-dependent rank-preserving transformation of f. The joint problem J is strictly comparison-based (or solely based on the f-ranks) and therefore invariant to monotonic transformations of f (see below for the precise definition) [5]. The natural gradient of $J(\theta)$, estimated by Monte Carlo samples from P_θ, governs the update of θ.

In contrast to some settings in machine learning where it is required to *estimate* the Fisher information matrix (FIM) and perform a numerical inversion in order to compute the natural gradient (e.g. for multi-layer perceptrons, see [4] for a discussion), in our context the inverse FIM can be explicitly derived provided a statistical model with known FIM is used. Consequently, efficient comparison-based optimization algorithms using natural gradients can be derived [1,5,7]. Interestingly, the covariance matrix adaptation evolution strategy (CMA-ES) [9, 10, 20]—a state-of-the-art evolutionary algorithm for continuous optimization—derives from the natural gradient framework sketched above when using the family of Gaussian distributions [1, 5]. The CMA-ES algorithm was however introduced independently of the natural gradient approach.

The CMA-ES parametrizes a Gaussian model with full covariance matrix, i.e. θ encodes a mean vector and full covariance matrix. It achieves invariance to general linear transformation of the search space. However in consequence, its space complexity and its internal time complexity per f-call are quadratic (see details below). For optimizing functions in higher dimensions, quadratic scaling becomes quickly too time consuming and linear scaling is desirable. To achieve linear scaling, several "variants" of CMA-ES have been proposed compromising on the general invariance to linear transformation: sep-CMA [18] restricts C to a diagonal matrix, MVA [17] and R1-NES [19] parameterize C by

GECCO'14, July 12–16, 2014, Vancouver, BC, Canada.

ACM 978-1-4503-2662-9/14/07 ...$15.00.
http://dx.doi.org/10.1145/2576768.2598258.

$I + vv^{\mathrm{T}}$. We propose here a novel comparison-based algorithm with linear time and space complexity derived from the natural gradient. Instead of a full covariance matrix, we parametrize the covariance matrix as $D(I + vv^{\mathrm{T}})D$ where D is a diagonal matrix of size d and v a vector in $\mathbb{R}^d$. The parameter update follows the natural gradient and two further techniques from CMA-ES are borrowed: an evolution path that low pass filters the change of the distribution mean (additionally used for the natural gradient update) and cumulative step-size adaptation, based on a similar idea. Both mechanisms enhance the performance of CMA-ES considerably [8].

The reminder of the paper is organized as follows. Section 2 is devoted to the introduction to the IGO framework and the CMA-ES. In section 3, we derive the novel linear time and space algorithm from the IGO framework combined with the cumulation concept borrowed from the CMA-ES. The proposed method is called VD-CMA. In section 4, we compare VD-CMA with the CMA-ES on a standard benchmark testbed, both in terms of function evaluations and cpu time. It is also compared with other linear time variants of evolution strategies. We conclude this paper with summary and further discussion in Section 5.

2. INFORMATION GEOMETRIC OPTIMIZATION AND CMA-ES

Information Geometric Optimization (IGO) is a general framework for optimization in arbitrary search spaces. IGO is based on invariance and consequently leads to comparison and natural gradient based optimization algorithms [5]. We give now some background about IGO, explain how parts of the CMA-ES algorithm are instantiations of IGO on the family of Gaussian distributions and detail other important concepts of CMA-ES that we borrow to construct our novel algorithm. While the IGO framework applies to arbitrary search spaces, we describe it conveniently on $\mathbb{R}^d$.

2.1 Information Geometric Optimization

Given an objective function to be minimized, $f : \mathbb{R}^d \mapsto \mathbb{R}$, and a family of probability distributions, P_θ on $\mathbb{R}^d$ with $\theta \in \Theta$, equipped with the Fisher metric, a joint optimization problem is defined on Θ as the maximization of the expectation $J(\theta)$ of a nonlinear scaling W of the objective function f over P_θ. More specifically, we consider at a given iteration t the current parameter θ^t and define $W^f_{\theta^t}(x) := w\big(q^f_{\theta^t}(f(x))\big)$, where $w : [0, 1] \to \mathbb{R}$ is a non-increasing function and $q^f_{\theta^t}(\bar{f})$ is the probability of sampling a point from the current distribution given θ^t into the sub level set $\{x \in \mathbb{R}^d : f(x) \leq \bar{f}\}$, defined as $q^f_{\theta^t}(\bar{f}) := \int_{f(x)\leq\bar{f}} p_{\theta^t}(x)\,\mathrm{d}x$. Hence, the function J also depends on θ (denoted by J_{θ^t} in the sequel) and is defined as $J_{\theta^t}(\theta) := \int_{\mathbb{R}^d} W^f_{\theta^t}(x) p_\theta(x)\,\mathrm{d}x$. This q-quantile based transformation of f is invariant under monotonic transformations of f and leads to comparison-based algorithms that show the same performance on f as on any increasing transformation of f, e.g., $f(x) = a\|x\|^b + c$ is equivalent for all $a, b > 0$ and $c \in \mathbb{R}$. The IGO update consists in the gradient ascent step

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

where $\tilde{\nabla} J_{\theta^t}(\theta)$ is the natural gradient of J_{θ^t} given by the product of the inverse of the FIM[1] $\mathcal{I}^{-1}(\theta)$ and the vanilla gradient $\nabla J_{\theta^t}(\theta)$. The gradient $\nabla J_{\theta^t}(\theta)$ is computed with the log-likelihood trick

$$\nabla J_{\theta^t}(\theta) = \int_{\mathbb{R}^d} W^f_{\theta^t}(x)\big(\nabla \ln p_\theta(x)\big) p_\theta(x)\,\mathrm{d}x \ . \tag{2}$$

The update (1) requires to evaluate the integral (2). This integral is naturally estimated by a Monte-Carlo method with λ samples drawn from the current distribution P_{θ^t}. Given a set of λ independent samples $x_i \sim P_{\theta^t}$, for $i = 1, \ldots, \lambda$, the quantile function $q^f_{\theta^t}(f(x_i))$ is approximated by $(\mathrm{rk}(x_i)+1/2)/\lambda$, with $\mathrm{rk}(x_i)$ the ranking of $f(x_i)$ among the λ samples, namely, $\mathrm{rk}(x_i) := |\{j : f(x_j) \leq f(x_i)\}|$. Hence $W^f_{\theta^t}(x_i)$ is approximated by $w_{\mathrm{rk}(x_i)} := w((\mathrm{rk}(x_i) + 1/2)/\lambda)$. With $w_i := w((i + 1/2)/\lambda)/\lambda$, the IGO algorithm update reads

$$\begin{aligned}\theta^{t+\delta t} - \theta^t &= \delta t \mathcal{I}^{-1}(\theta^t) \textstyle\sum_{i=1}^{\lambda} w_{\mathrm{rk}(x_i)} \nabla \ln p_{\theta^t}(x_i) \\ &= \delta t \mathcal{I}^{-1}(\theta^t) \textstyle\sum_{i=1}^{\lambda} w_i \nabla \ln p_{\theta^t}(x_{i:\lambda}) \ ,\end{aligned} \tag{3}$$

where $x_{i:\lambda}$ denotes the i^{th} best point ranked according to f.

2.2 The CMA-ES Algorithm

The CMA-ES algorithm [9, 10, 20], considered as state-of-the-art method for stochastic numerical optimization, was recently found to derive from the natural gradient, and more precisely from the IGO framework described in the previous section [1, 5]. For CMA-ES, P_θ is the family of Gaussian distributions $\mathcal{N}(m, \sigma^2 C)$ parametrized with a mean vector $m \in \mathbb{R}^d$ and a covariance matrix $\sigma^2 C$, with $\sigma \in \mathbb{R}_>$ the so-called global step-size, and $C \in \mathbb{R}^{d\times d}$ a positive-definite symmetric matrix. Because Gaussian distributions are considered, the FIM and its inverse are well known and the IGO update (1) can be computed analytically [1]. The update of $\theta = (m, \sigma, C)$ in CMA-ES combines different ideas. The update of m and C uses the natural gradient as prescribed by the IGO algorithm (3), however also using the so-called cumulation concept that smoothens and accelerates this update without compromising its stability [8]. Then, different learning rates (corresponding to the step-size δt of the gradient ascent step) for the mean and the covariance matrix updates are used. Last, the global step-size σ is independently adapted in order to accelerate the search performance and prevent premature convergence.

We give in the sequel a compact but thorough definition of the CMA-ES algorithm. After the initialization of m, σ and $C = I$ and so-called evolution paths $p_\sigma = p_C = 0 \in \mathbb{R}^d$, the CMA-ES repeats the following steps until a termination criterion is satisfied.

Step 1. Matrix Decomposition. Compute the square root $\sqrt{C}$ of C, where $\sqrt{C}$ is symmetric and positive-definite, and satisfies $C = \sqrt{C}\sqrt{C}$.

Step 2. Sampling, Evaluation and Ranking. Sample λ candidate solutions $x_i \sim \mathcal{N}(m, \sigma^2 C)$, for $i \in [\![1, \lambda]\!]$, as follows. Generate d-variate standard normal random vectors $z_i \sim \mathcal{N}(0, I)$ and compute $x_i = m + \sigma\sqrt{C} z_i$. For the later use we keep $\{z_i\}$ as well as $\{x_i\}$. Then, evaluate their objective values $f(x_i)$ for all $i \in [\![1, \lambda]\!]$. Rank the solutions according to f. In the following steps the subscript $i : \lambda$ denotes the index of ith best solution among λ current samples.

[1]The Fisher information matrix, FIM, is defined as $\mathcal{I}(\theta) = \int_{\mathbb{R}^d} \nabla \ln p_\theta(x)(\nabla \ln p_\theta(x))^{\mathrm{T}} p_\theta(x)\,\mathrm{d}x$.

Step 3. Cumulation. Update the evolution paths p_σ and p_C as

$$p_\sigma \leftarrow (1-c_\sigma)p_\sigma + \sqrt{c_\sigma(2-c_\sigma)\mu_{\mathrm{eff}}}\sum_{i=1}^{\mu} w_i z_{i:\lambda}$$
$$p_C \leftarrow (1-c_c)p_C + \frac{h_\sigma\sqrt{c_c(2-c_c)\mu_{\mathrm{eff}}}}{\sigma}\sum_{i=1}^{\mu} w_i(x_{i:\lambda}-m).$$

Here c_σ and c_c are the inverses of the backward time horizons for p_σ and p_C, respectively, and $\mu_{\mathrm{eff}} = \left(\sum_{i=1}^{\mu} w_i^2\right)^{-1}$. We let $h_\sigma = 1$ if $\|p_\sigma\|^2/d < (2+4/(d+1))(1-(1-c_\sigma)^{2t})$, where t is the iteration number starting from one, and $h_\sigma = 0$ otherwise.

Step 4. Update Parameters. Compute the natural gradient and update the parameters as follows:

$$m \leftarrow m + c_m\sum_{i=1}^{\mu} w_i\tilde{\nabla}_m \ln p_\theta(x_{i:\lambda}),$$
$$\sigma \leftarrow \sigma\exp\left((c_\sigma/d_\sigma)\left(\|p_\sigma\|/\chi_d - 1\right)\right),$$
$$C \leftarrow C + \underbrace{(1-h_\sigma)c_1c_c(2-c_c)C}_{\text{make up for variance loss in case of } h_\sigma=1} + c_\mu\underbrace{\sum_{i=1}^{\mu} w_i\tilde{\nabla}_C \ln p_\theta(x_{i:\lambda})}_{\text{rank-}(\mu\wedge d)\text{ update}} + c_1\underbrace{\tilde{\nabla}_C \ln p_\theta(m+\sigma p_C)}_{\text{rank-one update}}.$$

Here c_m is the learning rate for the m update, c_μ and c_1 are the learning rates of the so-called rank-μ and rank-one updates for C. The damping parameter for the σ update is denoted by d_σ. The symbol χ_d denotes the expected value of $\|\mathcal{N}(0,I)\| = \sqrt{2}\Gamma((d+1)/2)/\Gamma(d/2) \approx \sqrt{d}(1-1/(4d)+1/(21d^2))$. The approximated value is used in the algorithm. The natural gradient $\tilde{\nabla} = (\tilde{\nabla}_m, \tilde{\nabla}_C)$ of the log-likelihood of p_θ w.r.t. (m, C) is computed while σ is considered to be fixed.

When σ is fixed, the FIM $\mathcal{I}(\theta)$, where the parameter vector is $\theta = [m^{\mathrm{T}}, \mathrm{vec}(C)^{\mathrm{T}}]^{\mathrm{T}}$, becomes a block diagonal matrix $\mathrm{diag}(\mathcal{I}_m, \mathcal{I}_C)$ [2]. The diagonal blocks of $\mathcal{I}$ are given by $\mathcal{I}_m = \sigma^{-2}C^{-1}$ and $\mathcal{I}_C = 2^{-1}(C^{-1}\otimes C^{-1})$, where $\otimes$ denotes the Kronecker product. The vanilla gradients of the log-likelihood w.r.t. m and C are respectively $\nabla_m \ln p_\theta(x) = \sigma^{-2}C^{-1}(x-m)$ and $\nabla_C \ln p_\theta(x) = 2^{-1}C^{-1}\big(\sigma^{-2}(x-m)(x-m)^{\mathrm{T}} - C\big)C^{-1}$. Multiplying the inverse of $\mathcal{I}_m$ and $\mathcal{I}_C$ with the gradient $\nabla_m \ln p_\theta(x)$ and $\mathrm{vec}\big(\nabla_C \ln p_\theta(x)\big)$, we have the natural gradient $\tilde{\nabla}_m \ln p_\theta(x) = (x-m)$ and $\tilde{\nabla}_C \ln p_\theta(x) = \sigma^{-2}(x-m)(x-m)^{\mathrm{T}} - C$. Therefore, the update equations for m and C read

$$C \leftarrow C + c_\mu\sum_{i=1}^{\mu} w_i\big(\sigma^{-2}(x_{i:\lambda}-m)(x_{i:\lambda}-m)^{\mathrm{T}} - C\big) + c_1\big(p_Cp_C^{\mathrm{T}} - C\big),$$
$$m \leftarrow m + c_m\sum_{i=1}^{\mu} w_i(x_{i:\lambda}-m)\ .$$

The resulting m- and C-updates are similar to those used in the cross-entropy method (CEM) for continuous optimization [6] if $\sigma = 1$ and $c_1 = 0$, except that in CEM the m is updated first, which invariably leads to smaller variances in C and aggravates the problem of premature convergence.

The constants appearing in the algorithm are summarized in the following [8].

$$\begin{aligned}
\lambda &= 4 + \lfloor 3\ln(d)\rfloor, \quad \mu = \lfloor\lambda/2\rfloor, \quad c_m = 1,\\
w_i &= \frac{\ln((\lambda+1)/2) - \ln(i)}{\sum_{i=1}^{\mu}(\ln((\lambda+1)/2) - \ln(i))}, \quad c_\sigma = \frac{\mu_{\mathrm{eff}}+2}{d+\mu_{\mathrm{eff}}+5},\\
d_\sigma &= 1 + c_\sigma + 2\max(0, \sqrt{\tfrac{\mu_{\mathrm{eff}}-1}{d+1}} - 1),\\
c_c &= \frac{4+\mu_{\mathrm{eff}}/d}{d+4+2\mu_{\mathrm{eff}}/d}, \quad c_1 = \frac{2}{(d+1.3)^2+\mu_{\mathrm{eff}}},\\
c_\mu &= \min\left(1-c_1,\ \frac{2(\mu_{\mathrm{eff}}-2+1/\mu_{\mathrm{eff}})}{(d+2)^2+\mu_{\mathrm{eff}}}\right)\ .
\end{aligned} \tag{4}$$

The internal time complexity is $\mathcal{O}(d^3)$ for Step 1, $\mathcal{O}(\lambda d^2)$ for Step 2, $\mathcal{O}(\mu d)$ for Step 3 and $\mathcal{O}(\mu d^2)$ for Step 4. In practice, the matrix decomposition (Step 1) is done every $\lceil(10d(c_1+c_\mu))^{-1}\rceil$ iterations reducing the internal time complexity to $\mathcal{O}(d^2)$ per f-call. The space complexity is $\mathcal{O}(d^2+\mu d)$.

3. VD-CMA: A LINEAR VARIANT OF CMA-ES FOR HIGH DIMENSION OPTIMIZATION

We derive in this section a novel comparison-based algorithm using the IGO framework and additional features of CMA-ES. The algorithm aims at optimizing functions in high dimensions and should thus scale linearly with dimension d for its internal time (per f-call) and memory requirements. For this purpose we restrict the covariance matrix of the Gaussian model $\{\mathcal{N}(m, \sigma^2C)\}$ on $\mathbb{R}^d$ such that C has only $2d$ components to be adapted. More specifically, C is written in the form

$$C = D(I + vv^{\mathrm{T}})D\ , \tag{5}$$

where D is a diagonal matrix of dimension d and v is a vector in $\mathbb{R}^d$. This model is able to represent a scaling for each variable by D and a principle component, which is generally not parallel to an axis, by Dv. We parameterize the model by $\theta = (m \in \mathbb{R}^d,\ \sigma \in \mathbb{R},\ \theta_C \in \mathbb{R}^{2d})$ where θ_C is composed of two parts: $\theta_D \in \mathbb{R}^d$ whose ith element is the ith diagonal element of D, and $v \in \mathbb{R}^d$.

3.1 Preliminaries

To derive the parameter update equation for the model based on the natural gradient, we first derive the gradient of the log-likelihood $\ln p_\theta(x)$ and the FIM of the model. When computing the gradient and the FIM, we suppose that σ is fixed and the parameter vector considered is $\theta = [m^{\mathrm{T}}, v^{\mathrm{T}}, \theta_D^{\mathrm{T}}]^{\mathrm{T}}$.

Notations. Let V be the diagonal matrix whose ith diagonal element is the $i$the component of v. The normalization of v by its Euclidean norm is denoted by $\overline{v} = v/\|v\|$. Analogously, $\overline{V} = V/\|v\|$. The determinant of $I + vv^{\mathrm{T}}$, namely $1+\|v\|^2$, is denoted γ_v. Let $\odot$ denote the element-wise product operator and $\overline{\overline{v}} = \overline{v}\odot\overline{v}$. The vector whose elements are all one is denoted by $\mathbf{1}$ for any dimension. The vector e_i is the unit vector whose ith component is one and the others are zero.

LEMMA 3.1. *Let $x \in \mathbb{R}^d$ and let $y = \sigma^{-1}D^{-1}(x-m)$. The gradients of the log-likelihood of our model w.r.t. m, v*

and θ_D are

$$\nabla_m \ln p_\theta(x) = \sigma^{-2} C^{-1}(x-m),$$
$$\nabla_{\theta_D} \ln p_\theta(x) = D^{-1}\left[y \odot y - \gamma_v^{-1}\langle y, v\rangle\, y \odot v - \mathbf{1}\right]$$
$$\nabla_v \ln p_\theta(x) = \gamma_v^{-1}\left[\langle y, v\rangle\, y - \gamma_v^{-1}(\langle y, v\rangle^2 + \gamma_v) v\right] \ .$$

PROOF IDEA. It is known from Eq. (20) in [2] that the partial derivative of the log-likelihood of the Gaussian model $\{\mathcal{N}(m, \Sigma)\}$ parameterized by θ given x w.r.t. θ_i is computed as

$$\frac{\partial \ln p_\theta(x)}{\partial \theta_i} = \frac{\partial m^{\mathrm{T}}}{\partial \theta_i}\Sigma^{-1}(x-m) + \frac{1}{2}\operatorname{Tr}\left(\Sigma^{-1}\frac{\partial \Sigma}{\partial \theta_i}\Sigma^{-1}\left[(x-m)(x-m)^{\mathrm{T}} - \Sigma\right]\right) \ ,$$

where Tr denotes the trace. Substituting the partial derivatives and simplifying the above equality, we have the gradients. The details are omitted due to the space limitation. □

LEMMA 3.2. *The Fisher information matrix of our model is a block diagonal matrix* $\operatorname{diag}(\mathcal{I}_m, \mathcal{I}_C)$*, where* $\mathcal{I}_m = \sigma^{-2} C^{-1}$ *and* $\mathcal{I}_C = \begin{bmatrix} \mathcal{I}_{v,v} & \mathcal{I}_{v,D} \\ \mathcal{I}_{D,v} & \mathcal{I}_{D,D} \end{bmatrix}$*, where* $\mathcal{I}_{v,D} = \mathcal{I}_{D,v}^{\mathrm{T}}$ *and*

$$\mathcal{I}_{v,v} = \gamma_v^{-1}\left[\|v\|^2 I + (1-\|v\|^2)\gamma_v^{-1} v v^{\mathrm{T}}\right],$$
$$\mathcal{I}_{D,v} = \gamma_v^{-1} D^{-1} V \left[(2+\|v\|^2) I - v v^{\mathrm{T}}\right],$$
$$\mathcal{I}_{D,D} = \gamma_v^{-1} D^{-1}\left[2\gamma_v I + \|v\|^2 V^2 - V v v^{\mathrm{T}} V\right] D^{-1} \ .$$

PROOF IDEA. It is a well-know fact that the ijth element $[\mathcal{I}]_{i,j}$ of the FIM for the Gaussian distribution $\mathcal{N}(m, \Sigma)$ with parameter θ is $[\mathcal{I}]_{i,j} = \frac{\partial m}{\partial \theta_i}\Sigma^{-1}\frac{\partial m^{\mathrm{T}}}{\partial \theta_j} + \frac{1}{2}\operatorname{Tr}\left(\Sigma^{-1}\frac{\partial \Sigma}{\partial \theta_i}\Sigma^{-1}\frac{\partial \Sigma}{\partial \theta_j}\right)$. Remember $\theta = [m^{\mathrm{T}}, v^{\mathrm{T}}, \theta_D^{\mathrm{T}}]^{\mathrm{T}}$ and $\Sigma = \sigma^2 C = \sigma^2 D(I + vv^{\mathrm{T}})D$ in our case. The partial derivative $\partial m/\partial \theta_i = e_i$ if $\theta_i = m_i$ and $\partial m/\partial \theta_i = 0$ otherwise. On the other hand, $\partial \Sigma/\partial m_i = 0$. Therefore, the first term in the above equality appears only if both θ_i and θ_j are the elements of m, and the second term appears only if neither θ_i or θ_j is the element of m. This proves the block diagonal property of $\mathcal{I}$ and $\mathcal{I}_m = \Sigma^{-1} = \sigma^{-2} C^{-1}$ is an immediate consequence. The rests are derived from the second term by substituting the partial derivatives w.r.t. v and θ_D and simplifying it. □

To compute the natural gradient, the FIM must be invertible. Since $\mathcal{I}$ is block diagonal, the inverse is $\operatorname{diag}(\mathcal{I}_m^{-1}, \mathcal{I}_C^{-1})$ if both $\mathcal{I}_m$ and $\mathcal{I}_C$ are nonsingular. By Lemma 3.2 we know that $\mathcal{I}_m = (\sigma^2 C)^{-1}$ and it is nonsingular as long as C is nonsingular. The partitioned matrix inversion formula described in Theorem 8.5.11 in [12] shows that $\mathcal{I}_C$ is nonsingular if and only if the Schur complement $S_{v,v} = \mathcal{I}_{D,D} - \mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}$ of $\mathcal{I}_{v,v}$ is nonsingular, where the invertibility of $\mathcal{I}_{v,v}$ is guaranteed by the Sherman-Morrison formula (e.g., Corollary 18.2.10 in [12]) as long as $v \neq 0$. The Schur complement $S_{v,v}$ is given in the following lemma, whose proof is straight forward from the partitioned matrix inversion formula, Sherman-Morrison formula and Lemma 3.2.

LEMMA 3.3. *The Schur complement of* $\mathcal{I}_{v,v}$ *in* $\mathcal{I}_C$ *appeared in Lemma 3.2 is* $S_{v,v} = 2D^{-1}[I - 2\overline{V}^2 + \overline{v}\overline{v}^{\mathrm{T}}]D^{-1}$.

Unfortunately, $S_{v,v}$ becomes singular for some $\overline{v}$. For example, $\overline{v} = e_i$ for any i is a typical case that causes the singularity of $S_{v,v}$. Moreover, $S_{v,v}$ becomes singular for any $\overline{v}$ (hence for any v) in the case of $d = 2$. Indeed, noting that $\|\overline{v}\|^2 = 1$, it is easy to see the determinant of $S_{v,v}$ is zero. Thus, the invertibility of $\mathcal{I}_C$ is not guaranteed. Theoretically one can use the pseudo inverse of the FIM to define the natural gradient everywhere. However, due to arbitrarily small eigenvalues of the FIM around singular points that lead to arbitrarily long natural gradients, the resulting parameter update becomes unstable when the parameter approaches a singular point. Therefore it does not essentially solve this difficulty.

3.2 Modified Fisher Information Matrix with Reduced Off-diagonal Blocks

A simple way to avoid singularity of the FIM is to restrict it to the principal diagonal blocks. Then, the natural gradient for each block of parameters is computed independently while the other parameters are fixed. In our case, the block diagonalized FIM is nonsingular since $\mathcal{I}_{v,v}$ and $\mathcal{I}_{D,D}$ are nonsingular according to the Scherman-Morrison formula, provided $v \neq 0$ and D is positive definite. However, this leads to poor behavior, for example, on the rotated Cigar function defined in Table 1, where the principle component of the covariance matrix should not be axis parallel and v has to be learned adequately. To get nonsingularity without significantly compromising the performance, we use $\operatorname{diag}(\mathcal{I}_m, \mathcal{I}_C^{(\alpha)})$ in place of $\operatorname{diag}(\mathcal{I}_m, \mathcal{I}_C)$, where

$$\mathcal{I}_C^{(\alpha)} = \begin{bmatrix} \mathcal{I}_{v,v} & \alpha\mathcal{I}_{v,D} \\ \alpha\mathcal{I}_{D,v} & \mathcal{I}_{D,D} \end{bmatrix}, \quad \alpha \in [0,1] \ . \tag{6}$$

If $\alpha = 1$, $\mathcal{I}_C^{(\alpha)}$ is the original FIM, if $\alpha = 0$, $\mathcal{I}_C^{(\alpha)}$ is block diagonal, and α must be tuned such that $\mathcal{I}_C^{(\alpha)}$ remains nonsingular for any v and D, with $\alpha = 1$ desired. Since $\mathcal{I}_{v,v}$ is nonsingular except for $v = 0$, according to the partitioned matrix inversion formula $\mathcal{I}_C^{(\alpha)}$ is invertible if and only if the Schur complement $S_{v,v} = \mathcal{I}_{D,D} - \alpha^2 \mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}$ of $\mathcal{I}_{v,v}$ is nonsingular.

LEMMA 3.4. *The Schur complement of* $\mathcal{I}_{v,v}$ *in* (6) *is* $S_{v,v} = D^{-1}(A + b\overline{v}\overline{v}^{\mathrm{T}})D^{-1}$*, where* $b = -(1-\alpha^2)\|v\|^4\gamma_v^{-1} + 2\alpha^2$ *and* $A = 2I - (b + 2\alpha)\overline{V}^2$.

The proof is straightforward from Lemma 3.2 and Scherman-Morrison formula. Although the necessary and sufficient condition for $S_{v,v}$ to be nonsingular is not provided, the next proposition shows a sufficient α which leads to numerically stable $S_{v,v}^{-1}$.

PROPOSITION 3.5. *Let* $\gamma = \gamma_v^{-1/2}$*. If we choose*

$$\alpha = \min\left(1, \ \frac{[\|v\|^4 + (2-\gamma)\gamma_v / \max_i(\overline{v}_i)]^{1/2}}{2 + \|v\|^2}\right) \ , \tag{7}$$

then $S_{v,v}$ *is nonsingular and its inverse is given by*

$$S_{v,v}^{-1} = D\left[A^{-1} - (1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v})^{-1} b A^{-1}\overline{v}\overline{v}^{\mathrm{T}}A^{-1}\right]D \ .$$

PROOF IDEA. According to Lemma 3.4 and Scherman-Morrison formula, $S_{v,v}$ is invertible if A is nonsingular and $1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v} \neq 0$. Then, Scherman-Morrison formula provides the above explicit form of $S_{v,v}^{-1}$. Thus, it suffices to show the nonsingularity of A and $1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v} \neq 0$ under condition (7). Indeed, we prove $A_{i,i} \geq \gamma$ and $1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v} \geq \gamma$. The necessary and sufficient condition for $A_{i,i} \geq \gamma$ is $\alpha^2 \leq [\|v\|^4 + (2-\gamma)(1+\|v\|^2)/\max(\overline{v}_j)]/(2+\|v\|^2)^2$. If this condition holds, we have $0 < \overline{v}^{\mathrm{T}}A^{-1}\overline{v} \leq \gamma^{-1}\sum_i \overline{v}_i^2 \leq \gamma^{-1}\sum_i \overline{v}_i = \gamma^{-1}$. Noting that $\gamma \leq 1$, we have that $1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v} \geq \gamma$ holds if $b \geq \gamma(\gamma - 1)$. The necessary and sufficient condition for $b \geq$

$\gamma(\gamma-1)$ is $\alpha^2 \geq [\|v\|^4-(1-\gamma)\gamma(1+\|v\|^2)]/(\|v\|^4+2\|v\|^2+2)$. The α in (7) satisfies both inequalities. $\square$

The following theorem provides $\mathcal{O}(d)$ computation of the modified natural gradient of the log-likelihood with α introduced in Proposition 3.5.

THEOREM 3.6. *The modified natural gradient is computed as follows. Let $y = D^{-1}(x-m)/\sigma$ as above. Let α, A and b be those which appear in Proposition 3.5 and Lemma 3.4. Compute s and t as follows.*

1. $s \leftarrow y \odot y - \|v\|^2 \langle y, \overline{v}\rangle \gamma_v^{-1} y \odot \overline{v} - \mathbf{1}$
2. $t \leftarrow \langle y, \overline{v}\rangle y - 2^{-1}(\langle y, \overline{v}\rangle^2 + \gamma_v)\overline{v}$
3. $s \leftarrow s - \alpha\gamma_v^{-1}\left((2+\|v\|^2)\overline{v} \odot t - \|v\|^2 \langle \overline{v}, t\rangle \overline{v}\right)$
4. $s \leftarrow A^{-1}s - \left(1 + b\left\langle \overline{v}, A^{-1}\overline{v}\right\rangle\right)^{-1} b \left\langle s, A^{-1}\overline{v}\right\rangle A^{-1}\overline{v}$
5. $t \leftarrow t - \alpha\left[(2+\|v\|^2)\overline{v} \odot s - \langle s, \overline{v}\rangle \overline{v}\right]$

Then, $\tilde{\nabla}_m \ln p_\theta(x) = x - m$, $\tilde{\nabla}_v \ln p_\theta(x) = \|v\|^{-1} t$ and $\tilde{\nabla}_{\theta_D} \ln p_\theta(x) = Ds$.

PROOF IDEA. The modified natural gradient is the product of the inverse of $\mathrm{diag}(\mathcal{I}_m, \mathcal{I}_C^{(\alpha)})$ and the vanilla gradient given in Lemma 3.1. The inverse of $\mathrm{diag}(\mathcal{I}_m, \mathcal{I}_C^{(\alpha)})$ is $\mathrm{diag}(\mathcal{I}_m^{-1}, (\mathcal{I}_C^{(\alpha)})^{-1})$. As is shown in Lemma 3.2, $\mathcal{I}_m = \sigma^{-2}C^{-1}$ and its inverse is $\mathcal{I}_m^{-1} = \sigma^2 C$. Premultiplying the vanilla gradient w.r.t. m by $\mathcal{I}_m^{-1}$, we have $\tilde{\nabla}_m \ln p_\theta(x) = x - m$. The inverse of $\mathcal{I}_C^{(\alpha)}$ with α in Proposition 3.5 can be computed by using the partitioned matrix inversion formula as

$$\begin{bmatrix} \mathcal{I}_{v,v} & \alpha\mathcal{I}_{v,D} \\ \alpha\mathcal{I}_{D,v} & \mathcal{I}_{D,D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathcal{I}_{v,v}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D} \\ I \end{bmatrix} S_{v,v}^{-1} \begin{bmatrix} -\alpha\mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1} & I \end{bmatrix}$$

The inverse $\mathcal{I}_{v,v}^{-1}$ can be computed by the Scherman-Morrison formula and $S_{v,v}^{-1}$ is given in Proposition 3.5. Substituting them into the above equality and premultiplying the vanilla gradient given in Lemma 3.1 by the right hand side of the above equality, we obtain the natural gradient w.r.t. v and θ_D. The steps 2 to 7 are just a decomposition of the computation and can be computed in $\mathcal{O}(d)$. $\square$

3.3 The VD-CMA Algorithm

The overall algorithm is as follows. Initialize m and σ depending on the problem search space. Initialize $D = I$ and $v \sim \mathcal{N}(0, I/d)$ and $p_C = p_\sigma = 0$.

Step 1. Sampling, Evaluation and Ranking. Sample λ candidate solutions x_i, for $i \in [\![1, \lambda]\!]$, as follows. Generate d-variate standard normal random vectors $z_i \sim \mathcal{N}(0, I)$, compute $y_i = z_i + (\sqrt{1+\|v\|^2} - 1)\langle z_i, \overline{v}\rangle \overline{v}$ and $x_i = m + \sigma D y_i$. Then, $y_i \sim \mathcal{N}(0, I + vv^{\mathrm{T}})$ and $x_i \sim \mathcal{N}(m, \sigma^2 D(I + vv^{\mathrm{T}})D)$. Evaluate their objective values $f(x_i)$ for all $i \in [\![1, \lambda]\!]$. Rank solutions according to f.

Step 2. Cumulation. The evolution paths p_σ and p_C are updated in the same way as the original CMA does (see Sec. 2.2). Note that $z_i = C^{-1/2}(x_i - m)/\sigma$ in the original CMA whereas $z_i = (I + vv^{\mathrm{T}})^{-1/2} D^{-1}(x_i - m)/\sigma \neq C^{-1/2}(x_i - m)/\sigma$ in our case. We use z_i rather than $C^{-1/2}(x_i - m)/\sigma$ in order to achieve linear time update. Then, compute h_σ in the same way as in the original CMA.

Step 3. Update Parameters. Update the parameters as follows:

$$\begin{aligned} m &\leftarrow m + c_m \textstyle\sum_{i=1}^{\mu} w_i \tilde{\nabla}_m \ln p_\theta(x_{i:\lambda}), \\ \sigma &\leftarrow \sigma \exp\left((c_\sigma/d_\sigma)\left(\|p_\sigma\|/\chi_d - 1\right)\right), \\ v &\leftarrow v + c_\mu \textstyle\sum_{i=1}^{\mu} w_i \tilde{\nabla}_v \ln p_\theta(x_{i:\lambda}) \\ &\quad + (1-h_\sigma)c_1 \tilde{\nabla}_v \ln p_\theta(m + \sigma p_C), \\ D &\leftarrow D + c_\mu \textstyle\sum_{i=1}^{\mu} w_i \tilde{\nabla}_D \ln p_\theta(x_{i:\lambda}) \\ &\quad + (1-h_\sigma)c_1 \tilde{\nabla}_D \ln p_\theta(m + \sigma p_C). \end{aligned}$$

Here the natural gradients are computed following Theorem 3.6. The (constant) parameters are taken from (4) except for c_σ, c_1 and c_μ. Since the degrees of freedom in the covariance matrix is $2d$ compared with $d(d+1)/2$ in the original CMA, we expect that the natural gradient estimate is more reliable and larger values for c_1 and c_μ can be taken. The learning rate c_σ for σ is modified as well to achieve better scale-up with d. Let c_1^{old} and c_μ^{old} be the settings given in (4). Then, we set

$$c_\sigma = \frac{\sqrt{\mu_{\mathrm{eff}}}}{2(\sqrt{d} + \sqrt{\mu_{\mathrm{eff}}})},\ c_1 = \frac{d-5}{6} c_1^{\mathrm{old}},\ c_\mu = \min\left(1 - c_1, \frac{d-5}{6} c_\mu^{\mathrm{old}}\right).$$

The internal space complexity decreases to $\mathcal{O}(\mu d)$, the internal time complexity for each objective function call to $\mathcal{O}(d)$ compared to $\mathcal{O}(\mu d + d^2)$ and $\mathcal{O}(d^2)$ in the standard CMA, respectively.

4. EXPERIMENTS

We evaluated the new VD-CMA on benchmark functions described in Table 1, where the number of variables varies from 10 to 10^4. The VD-CMA is comparison-based and has the same performance on any composition of the functions by a monotonic transformation. The initial mean vector obeys $\mathcal{N}(3 \cdot \mathbf{1}, 2^2 \cdot I)$, except for $f_{\mathrm{ros}}(x)$ and $f_{\mathrm{rosrot}}(x)$ where m obeys $\mathcal{N}(0, 2^2 \cdot I)$ to avoid the symmetry; the initial stepsize is $\sigma = 2$. Runs are terminated as successful when a function value better than 10^{-10} is reached, otherwise when the number of function evaluations reached $10^5 \cdot d$. Only on the Rosenbrock functions about 15% of the runs were not successful due to the local minima and these are disregarded in the presentation. The code is implemented in Octave (single thread) and run on Debian 6.0 machine with Intel(R) Core(TM) i7-3770 3.4GHz CPU and 16 GB RAM.

Figure 1 shows a typical result on the 50 dimensional f_{ellcig} function, for which the inverse Hessian is proportional to $D_{\mathrm{ell}}^{-1}(I + (10^6 - 1)uu^{\mathrm{T}})D_{\mathrm{ell}}^{-1}$. First, D becomes proportional to D_{ell}^{-1}, then v starts to adapt, ending up with $v_i \approx \sqrt{(10^6-1)/d}$, such that $C = D(I + vv^{\mathrm{T}})D$ finally becomes closely proportional to the inverse Hessian. The reason of the later adaptation of v is that the function value is less sensitive to the direction of u than the coordinate-wise scaling by D_{ell} and the selection bias is only visible after $D \approx D_{\mathrm{ell}}^{-1}$. After learning the inverse Hessian, the speed of convergence is as fast as on f_{sph}. On the 50 dimensional f_{ellcig}, α is almost always one, whereas we observed a smaller α on lower dimensional functions (e.g., varying in $[0.75, 1]$ on the 10 dimensional f_{ellcig} and in $[0.65, 1]$ on the 10 dimensional f_{cig}).

Figure 2.(a) shows the number of function evaluations and CPU time in seconds, averaged over 10 independent runs on eight functions. The number of function evaluations scale up linearly on f_{sph}, f_{tab}, f_{cig}, f_{cigrot}, and slightly more than linear on f_{ell}, f_{ellcig}. On the Rosenbrock functions f_{ros} and

Table 1: Test function definitions. R is an orthogonal matrix and u is a unit vector, both are randomly generated for each run; D_{ell} is a diagonal matrix whose ith diagonal element is $10^{3\frac{i-1}{d-1}}$. The global minimum point is located at 1, $R^{\mathrm{T}}\mathbf{1}$ and 0 for f_{ros}, f_{rosrot} and all other functions, respectively.

Sphere	$f_{\mathrm{sph}}(x) = \sum_{i=1}^{d} x_i^2$	Tablet	$f_{\mathrm{tab}}(x) = 10^6 x_1^2 + \sum_{i=2}^{d} x_i^2$
Ellipsoid	$f_{\mathrm{ell}}(x) = f_{\mathrm{sph}}(D_{\mathrm{ell}} x)$	Cigar	$f_{\mathrm{cig}}(x) = x_1^2 + 10^6 \sum_{i=2}^{d} x_i^2$
Ellipsoid-Cigar	$f_{\mathrm{ellcig}}(x) = f_{\mathrm{cigrot}}(D_{\mathrm{ell}} x)$	Rot-Tablet	$f_{\mathrm{tabrot}}(x) = f_{\mathrm{sph}}(x) + (10^6 - 1)\langle x, u\rangle^2$
Rot-Ellipsoid	$f_{\mathrm{ellrot}}(x) = f_{\mathrm{ell}}(Rx)$	Rot-Cigar	$f_{\mathrm{cigrot}}(x) = 10^6 f_{\mathrm{sph}}(x) + (1 - 10^6)\langle x, u\rangle^2$
Rot-Rosenbrock	$f_{\mathrm{rosrot}}(x) = f_{\mathrm{ros}}(Rx)$	Rosenbrock	$f_{\mathrm{ros}}(x) = \sum_{i=1}^{d-1} \left[10^2 (x_i^2 - x_{i+1})^2 + (x_i - 1)^2\right]$

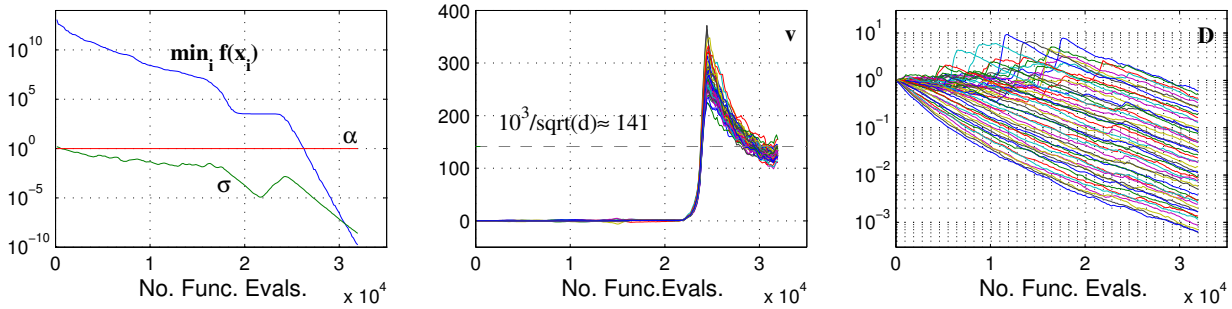

Figure 1: Single run on 50 dimensional f_{ellcig} with cigar axis $u = (1/\sqrt{d}, \ldots, 1/\sqrt{d})$. The minimum f-value at each iteration, α, σ (left), v values (center), and diagonal elements of D (right) are plotted.

f_{rosrot} the scale up is close to quadratic. On the other hand, it took 1.6e7, 1.0e7, 1.2e7 function evaluations on f_{tabrot} for $d = 10$, 20, 50, and 7.1e6, 1.3e7, 2.0e7 function evaluations on f_{ellrot} for $d = 10$, 20, 50, respectively, whereas the CMA requires 6.0e3, 1.6e4, 6.6e4 on f_{tabrot} and 6.2e3, 1.9e4, 1.0e5 function evaluations on f_{ellrot}. The inverse Hessian of these functions can not be well approximated by (5). Since the time complexity for each function evaluation is $\mathcal{O}(d)$, the total CPU time scales d times more than the number of function evaluations does.

Figure 2.(b) shows the speed up over the standard CMA. The reason of the speed up in terms of the number of function evaluations is the $(d-5)/6$ times larger learning rate for the C update. The effect is more pronounced on f_{ell}, f_{ellcig}, f_{tab} than on f_{cig} and f_{cigrot}, although they all have a Hessian matrix whose inverse can be represented by (5). This is because the standard CMA excels at learning single long components of C because of the cumulation in p_C. On f_{sph}, the performance does not differ much from standard CMA since C does not need to learn the shape and the σ update is dominative in determining the speed of convergence. When it comes to the total CPU time, our approach improves over the standard CMA for $d \geq 50$.

Figure 3 shows the scale up of VD-CMA, sep-CMA [18] and R1-NES [19], which are all linear time and space algorithms based on the same natural gradient principle.[2] Table 2 shows a comparison with the standard CMA, (1, 10)-AII [11] and (1, 10)-MVA [17]. The reason of better scale up of VD-CMA and sep-CMA than R1-NES even on f_{sph} is the cumulation employed to adapt σ. On f_{cig} and f_{cigrot} VD-CMA is more efficient than R1-NES, though both Gaussian models maintained by VD-CMA and R1-NES can adapt to the inverse Hessian. This is the effect of the cumulation for covariance adaptation. On f_{ell}, sep-CMA is faster than VD-CMA simply because the learning rate for the C update is higher. On the other hand, MVA and R1-NES, both of which restrict the covariance matrix to maintain only one long direction, do not solve f_{ell}, f_{ellcig} and f_{tab}. Since the model in our approach is richer than those maintained in sep-CMA and R1-NES, VD-CMA can solve more efficiently a larger class of functions including f_{ellcig}, f_{ros} and f_{rosrot} that are ill-conditioned and non-separable.

[2]The sep-CMA was introduced independently of the natural gradient, but later it was found in [2] to derive from the natural gradient framework.

Table 2: Average number of function evaluations to reach the target function value 10^{-9} on 20 dimensional functions among three runs except for MVA, where the average is computed over 70 runs. Data is taken from the references. Standard deviations are smaller than 3% of the average numbers except for MVA. On f_{ros}, no success was observed for MVA in 3.5×10^5 function evaluations.

	VD-CMA	CMA	(1, 10)-AII	(1, 10)-MVA
f_{ell}	9.4×10^3	2.0×10^4	1.2×10^4	no success
f_{ros}	2.0×10^4	2.1×10^4	2.1×10^4	5.7×10^4

5. DISCUSSION

Based on the IGO framework, we have derived a comparison-based stochastic search algorithm, VD-CMA, for continuous optimization in high dimension. To achieve internal computational time and space complexity linear in dimension, we have restricted the covariance matrix of the Gaussian distribution to $D(I + vv^{\mathrm{T}})D$. Since this model has singular

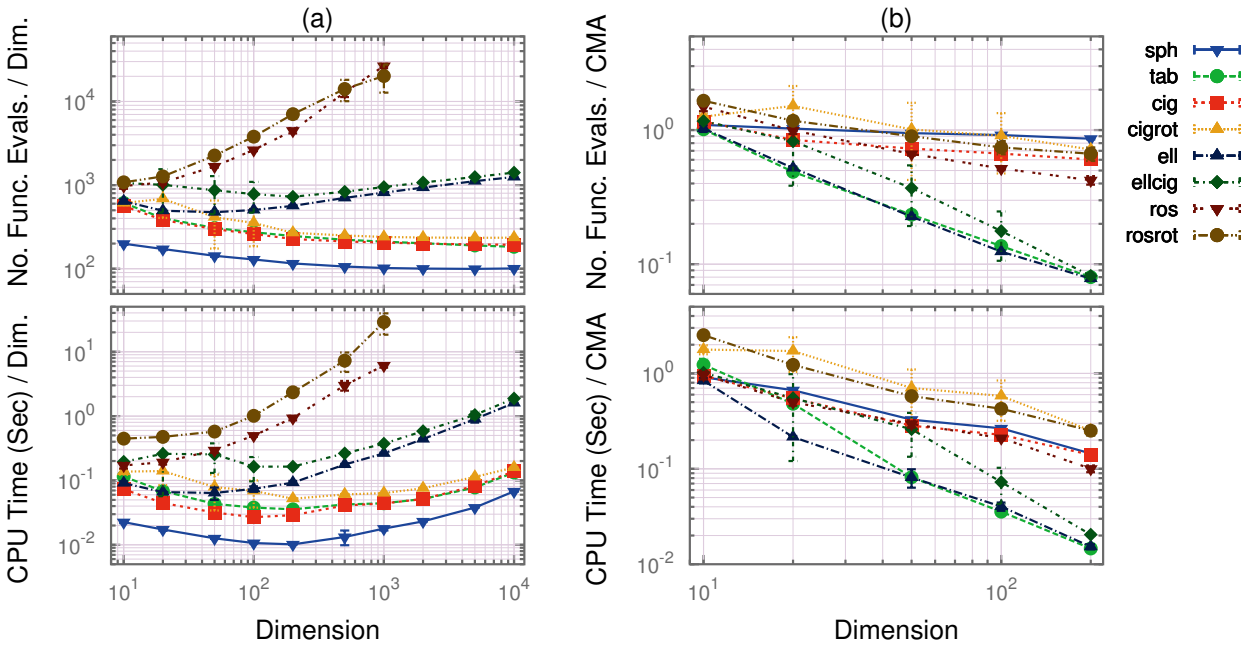

Figure 2: (a) Number of function evaluations (FEs) and CPU time [s] divided by d. (b) FEs and CPU time spent by VD-CMA divided by those spent by the CMA. Shown are average and standard deviation (error bar), from 10 independent runs.

points where the natural gradient is not defined and leads to unstable parameter update, we have defined a modified FIM to avoid the singularity and enable numerically stable computation of the natural gradient without significantly compromising the performance. Additionally to the natural gradient, we have incorporated the cumulation concept from the CMA-ES to make robust and accelerate the method.

We have shown the advantage of VD-CMA over the standard CMA in two aspects. One is the linear scaling of the internal time and memory usage w.r.t. the number of variables that is desired for optimizing functions in high dimension. The other is better scaling of the number of function evaluations thanks to the higher learning rates for the covariance update. The second aspect implies VD-CMA outperforms the standard CMA even on a low or moderate dimensional function where the inverse of the Hessian can be approximated by (5). On the other hand, if the model does not suit the inverse Hessian of a function, e.g., f_{ellrot} and f_{tabrot}, VD-CMA is inefficient compared to the standard-CMA. Compared to other linear time variants of the CMA-ES, it can solve a wider class of functions since we have a richer but linear number of elements of the covariance matrix.

We end with a remark on parallelization. As well as most evolutionary algorithms, one can benefit from parallelization. For the sake of simplicity we assume that λ processors are available. Then *sampling* and *evaluation* for each solution are performed in parallel. Moreover, step 2 and step 3 in Section 3.3 are parallelizable by computing $w_i z_{i:\lambda}$, $w_i(x_{i:\lambda} - m) = w_i \tilde{\nabla}_m \ln p_\theta(x_{i:\lambda})$, $w_i \tilde{\nabla}_v \ln p_\theta(x_{i:\lambda})$ and $w_i \tilde{\nabla}_D \ln p_\theta(x_{i:\lambda})$ for each $x_{i:\lambda}$ in parallel. Then, the number of floating point multiplications at each iteration on each processor reduces from $\mathcal{O}(\lambda d)$ to $\mathcal{O}(d)$. The other computation required is the sorting of $\mathcal{O}(\lambda)$ floating point numbers for ranking and the sum of $\mathcal{O}(\mu)$ floating point numbers for update, both of which are relatively cheap and can be also parallelized if needed. To further reduce the runtime, one can set the population size, λ, larger than the default value, which typically requires more function evaluations but smaller number of iterations. Large population also helps when the objective function is rugged.

Acknowledgements.
This work was supported by the ANR-2010-COSI-002 (SIMINOLE) and ANR-2012-MONU-0009 (NumBBO) grants of the French National Research Agency.

6. REFERENCES

[1] Y. Akimoto, Y. Nagata, I. Ono, and S. Kobayashi. Bidirectional Relation between CMA Evolution Strategies and Natural Evolution Strategies. In *Parallel Problem Solving from Nature – PPSN XI*, pages 154–163, 2010.

[2] Y. Akimoto, Y. Nagata, I. Ono, and S. Kobayashi. Theoretical Foundation for CMA-ES from Information Geometry Perspective. *Algorithmica*, 64:698–716, 2012.

[3] S. Amari. Natural Gradient Works Efficiently in Learning. *Neural Computation*, 10(2):251–276, 1998.

[4] S. Amari, H. Park, and K. Fukumizu. Adaptive method of realizing natural gradient learning for multilayer perceptrons. *Neural Computation*, 12:1399–1409, 2000.

[5] Y. Ollivier, L. Arnold, A. Auger, and N. Hansen. Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles. *arXiv:1106.3708*, 2011.

[6] P.-T. D. Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein. A Tutorial on the Cross-Entropy Method. *Annals of Operations Research*, (134):19–67, 2005.

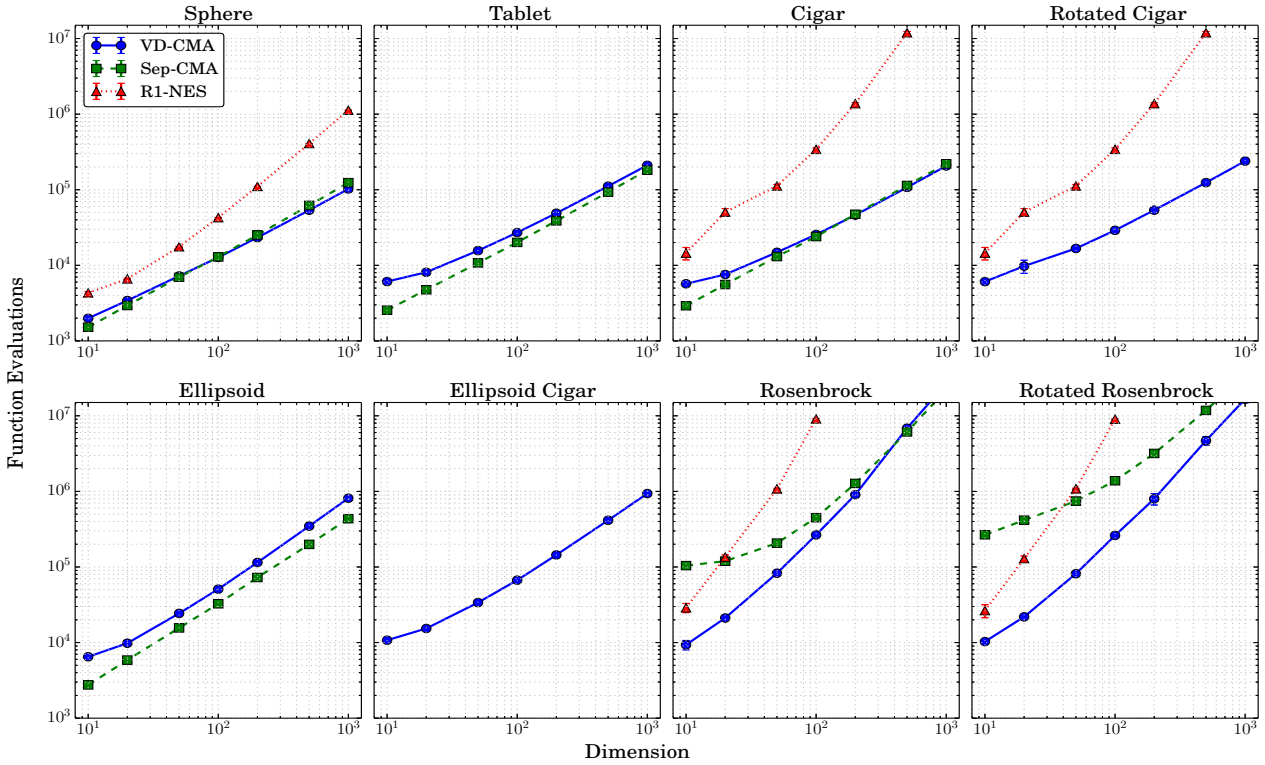

Figure 3: Number of function evaluations spent by VD-CMA, Sep-CMA and R1-NES. Shown are average and standard deviation (error bar), from 10 independe nt runs. Missing data implies it failed to reach the target within $2 \cdot 10^7$ function evaluations. Note that the error bars are hardly visible since the std. are small.

[7] T. Glasmachers, T. Schaul, Y. Sun, D. Wierstra, and J. Schmidhuber. Exponential Natural Evolution Strategies. In *Proceedings of Genetic and Evolutionary Computation Conference*, pages 393–400, 2010.

[8] N. Hansen and A. Auger. Principled Design of Continuous Stochastic Search: From Theory to Practice. In Y. Borenstein and A. Moraglio, editors, *Theory and Principled Methods for the Design of Metaheuristics*. Springer, 2013.

[9] N. Hansen, S. D. Muller, and P. Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation*, 11(1):1–18, 2003.

[10] N. Hansen and A. Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001.

[11] N. Hansen, A. Ostermeier, and A. Gawelczyk. On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating set Adaptation. In *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 57–64, 1995.

[12] D. A. Harville. *Matrix Algebra from a Statistician's Perspective*. Springer-Verlag, 2008.

[13] A. Honkela, M. Tornio, T. Raiko, and J. Karhunen. Natural conjugate gradient in variational inference. In *Neural Information Processing, 14th International Conference, ICONIP 2007*, pages 305–314, 2008.

[14] L. Malagò, M. Matteucci, and G. Pistone. Towards the geometry of estimation of distribution algorithms based on the exponential family. In *Foundations of Genetic Algorithms*, pages 230–242. 2011.

[15] A. Miyamae, Y. Nagata, I. Ono, and S. Kobayashi. Natural Policy Gradient Methods with Parameter-based Exploration for Control Tasks. In *Advances in Neural Information Processing Systems 23*, pages 1660–1668, 2010.

[16] J. Peters and S. Schaal. Natural actor-critic. *Neurocomputing*, 71(7-9):1180–1190, 2008.

[17] J. Poland and A. Zell. Main vector adaptation: A CMA variant with linear time and space complexity. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1050–1055, 2001.

[18] R. Ros and N. Hansen. A simple modification in CMA-ES achieving linear time and space complexity. *Parallel Problem Solving from Nature–PPSN X*, pages 296–305, 2008.

[19] Y. Sun, F. Gomez, T. Schaul, and J. Schmidhuber. A Linear Time Natural Evolution Strategy for Non-Separable Functions. *GECCO'13 Companion*, pages 61–62, 2013.

[20] T. Suttorp, N. Hansen, and C. Igel. Efficient covariance matrix update for variable metric evolution strategies. *Machine Learning*, 75(2):167–197, 2009.

[21] D. Wierstra, T. Schaul, J. Peters, and J. Schmidhuber. Natural evolution strategies. In *IEEE Congress on Evolutionary Computation*, pages 3381–3387, 2008.

APPENDIX

A. PROOFS

The following two well-known formulas are used repeatedly in the sequel.

PROPOSITION A.1 (SHERMAN-MORRISON). *Suppose that A is invertible and v is a column vector. Then, $A + vv^{\mathrm{T}}$ is invertible if and only if $1 + v^{\mathrm{T}}A^{-1}v \neq 0$ and the inverse is given by $A^{-1} - \frac{A^{-1}vv^{\mathrm{T}}A^{-1}}{1+v^{\mathrm{T}}A^{-1}v}$.*

PROPOSITION A.2 (PARTITIONED MATRIX INVERSION). *Suppose that T is nonsingular. Then, the partitioned matrix $[T, U; V, W]$ is nonsingular if and only if the Schur complement $S_T = W - VT^{-1}U$ of T is nonsingular and the inverse is given by*

$$\begin{bmatrix} T & U \\ V & W \end{bmatrix}^{-1} = \begin{bmatrix} T^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} -T^{-1}U \\ I \end{bmatrix} S_T^{-1} \begin{bmatrix} -T^{-1}V & I \end{bmatrix} \ .$$

Let $e_i \in \mathbb{R}^d$ be the unit vector whose ith element is one and the others are zero. Let $\delta_{i,j} = e_i^{\mathrm{T}} e_j$ which is one if $i = j$, 0 otherwise. In the following proofs we often use the following relations

$$\begin{aligned} \tfrac{\partial C}{\partial d_i} &= e_i e_i^{\mathrm{T}} D^{-1} C + C D^{-1} e_i e_i^{\mathrm{T}} \\ \tfrac{\partial C}{\partial v_i} &= D(e_i v^{\mathrm{T}} + v e_i^{\mathrm{T}}) D \ , \end{aligned}$$

where d_i is the ith diagonal element of D and v_i is the ith element of v.

PROOF OF LEMMA 3.1. The gradient w.r.t. m is proved in the main text. We are going to prove the gradients w.r.t. v and θ_D. Let $y = \sigma^{-1}D^{-1}(x - m)$. We start from the equation given in the main text:

$$\begin{aligned} \tfrac{\partial \ln p_\theta(x)}{\partial \theta_i} &= \tfrac{\partial m^{\mathrm{T}}}{\partial \theta_i} \Sigma^{-1}(x - m) \\ &\quad + \tfrac{1}{2} \operatorname{Tr}\big(\Sigma^{-1} \tfrac{\partial \Sigma}{\partial \theta_i} \Sigma^{-1} \left[(x - m)(x - m)^{\mathrm{T}} - \Sigma\right]\big) \ . \end{aligned}$$

Each element of $\nabla_v \ln p_\theta(x)$ is given as

$$\begin{aligned} &\tfrac{\partial \ln p_\theta(x)}{\partial v_i} \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} \tfrac{\partial C}{\partial v_i} C^{-1} [(x - m)(x - m)^{\mathrm{T}} - C]\right) \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} \tfrac{\partial C}{\partial v_i} C^{-1} [D y y^{\mathrm{T}} D - C]\right) \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} D (e_i v^{\mathrm{T}} + v e_i^{\mathrm{T}}) D C^{-1} [D y y^{\mathrm{T}} D - C]\right) \\ &= e_i^{\mathrm{T}} D C^{-1} [D y y^{\mathrm{T}} D - C] C^{-1} D v \\ &= e_i^{\mathrm{T}} D C^{-1} D y y^{\mathrm{T}} D C^{-1} D v - e_i^{\mathrm{T}} D C^{-1} D v \\ &= e_i^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} y y^{\mathrm{T}} (I + v v^{T})^{-1} v - e_i^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} v \\ &= e_i^{\mathrm{T}} \left(I - \tfrac{1}{1+\|v\|^2} v v^{\mathrm{T}}\right) y y^{\mathrm{T}} \left(I - \tfrac{1}{1+\|v\|^2} v v^{\mathrm{T}}\right) v \\ &\quad - e_i^{\mathrm{T}} \left(I - \tfrac{1}{1+\|v\|^2} v v^{\mathrm{T}}\right) v \\ &= e_i^{\mathrm{T}} \left(y - \tfrac{\langle y, v\rangle}{1+\|v\|^2} v\right) \left(\langle y, v\rangle - \tfrac{\langle y, v\rangle \|v\|^2}{1+\|v\|^2}\right) - e_i^{\mathrm{T}} \tfrac{1}{1+\|v\|^2} v \\ &= e_i^{\mathrm{T}} \tfrac{1}{1+\|v\|^2} \left[\langle y, v\rangle\, y - \tfrac{\langle y, v\rangle^2 + 1 + \|v\|^2}{1+\|v\|^2} v\right] \ . \end{aligned}$$

Rewriting it in a vector form, we have

$$\begin{aligned} \nabla_v \ln p_\theta(x) = (1 + \|v\|^2)^{-1} \big[&\langle y, v\rangle\, y \\ &- (1 + \|v\|^2)^{-1} (\langle y, v\rangle^2 + 1 + \|v\|^2) v\big] \ . \end{aligned}$$

Next, each element of $\nabla_{\theta_D} \ln p_\theta(x)$ is

$$\begin{aligned} &\tfrac{\partial \ln p_\theta(x)}{\partial [\theta_D]_i} \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} \tfrac{\partial C}{\partial d_i} C^{-1} [D y y^{\mathrm{T}} D - C]\right) \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} (e_i e_i^{\mathrm{T}} D^{-1} C + C D^{-1} e_i e_i^{\mathrm{T}}) C^{-1} [D y y^{\mathrm{T}} D - C]\right) \\ &= \operatorname{Tr}\left(C^{-1} e_i e_i^{\mathrm{T}} D^{-1} C C^{-1} [D y y^{\mathrm{T}} D - C]\right) \\ &= e_i^{\mathrm{T}} D^{-1} [D y y^{\mathrm{T}} D - C] C^{-1} e_i \\ &= e_i^{\mathrm{T}} [y y^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} D^{-1} - D^{-1}] e_i \\ &= e_i^{\mathrm{T}} [y y^{\mathrm{T}} (I - \tfrac{1}{1+\|v\|^2} v v^{\mathrm{T}}) D^{-1} - D^{-1}] e_i \\ &= e_i^{\mathrm{T}} [y y^{\mathrm{T}} D^{-1} - \tfrac{\langle y, v\rangle}{1+\|v\|^2} y v^{\mathrm{T}} D^{-1} - D^{-1}] e_i \\ &= y_i y_i d_i^{-1} - \tfrac{\langle y, v\rangle}{1+\|v\|^2} y_i v_i d_i^{-1} - d_i^{-1} \\ &= e_i^{\mathrm{T}} D^{-1} \left(y \odot y - \tfrac{\langle y, v\rangle}{1+\|v\|^2} y \odot v - \mathbf{1}\right) \ . \end{aligned}$$

Therefore, we have

$$\nabla_{\theta_D} \ln p_\theta(x) = D^{-1} \left[y \odot y - (1 + \|v\|^2)^{-1} \langle y, v\rangle\, y \odot v - \mathbf{1}\right] \ .$$

This ends the proof. □

PROOF OF LEMMA 3.2. We start from the well-known formula stated in the proof idea of Lemma 3.2 in the main text:

$$[\mathcal{I}]_{i,j} = \tfrac{\partial m}{\partial \theta_i} \Sigma^{-1} \tfrac{\partial m^{\mathrm{T}}}{\partial \theta_j} + \tfrac{1}{2} \operatorname{Tr}\big(\Sigma^{-1} \tfrac{\partial \Sigma}{\partial \theta_i} \Sigma^{-1} \tfrac{\partial \Sigma}{\partial \theta_j}\big) \ .$$

Since $\mathcal{I}_m$ is derived in the main text, we are going to derive an explicit form of $\mathcal{I}_C$.

Each element of $\mathcal{I}_{v,v}$ is

$$\begin{aligned} &[\mathcal{I}_{v,v}]_{i,j} \\ &= \tfrac{1}{2} \operatorname{Tr}\left(C^{-1} \tfrac{\partial C}{\partial v_i} C^{-1} \tfrac{\partial C}{\partial v_j}\right) \\ &= \tfrac{\sigma^4}{2} \operatorname{Tr}\left(C^{-1} (e_i v^{\mathrm{T}} + v e_i^{\mathrm{T}}) C^{-1} (e_j v^{\mathrm{T}} + v e_j^{\mathrm{T}})\right) \\ &= \operatorname{Tr}\left((I + v v^{\mathrm{T}})^{-1} e_i v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} e_j v^{\mathrm{T}}\right) \\ &\quad + \operatorname{Tr}\left((I + v v^{\mathrm{T}})^{-1} e_i v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} v e_j^{\mathrm{T}}\right) \\ &= e_i^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} v v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} e_j \\ &\quad + (v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} v) e_i^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} e_j \\ &= e_i^{\mathrm{T}} [(v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1} v)(I + v v^{\mathrm{T}})^{-1} \\ &\quad + (I + v v^{\mathrm{T}})^{-1} v v^{\mathrm{T}} (I + v v^{\mathrm{T}})^{-1}] e_j \\ &= e_i^{\mathrm{T}} \left[\tfrac{\|v\|^2}{1+\|v\|^2} (I + v v^{\mathrm{T}})^{-1} + \tfrac{1}{(1+\|v\|^2)^2} v v^{\mathrm{T}}\right] e_j \\ &= e_i^{\mathrm{T}} \left[\tfrac{\|v\|^2}{1+\|v\|^2} \left(I - \tfrac{1}{1+\|v\|^2} v v^{\mathrm{T}}\right) + \tfrac{1}{(1+\|v\|^2)^2} v v^{\mathrm{T}}\right] e_j \\ &= e_i^{\mathrm{T}} \left[\tfrac{\|v\|^2}{1+\|v\|^2} I - \tfrac{\|v\|^2}{(1+\|v\|^2)^2} v v^{\mathrm{T}} + \tfrac{1}{(1+\|v\|^2)^2} v v^{\mathrm{T}}\right] e_j \\ &= e_i^{\mathrm{T}} \left[\frac{\|v\|^2}{1 + \|v\|^2} I - \frac{\|v\|^2 - 1}{(1 + \|v\|^2)^2} v v^{\mathrm{T}}\right] e_j \\ &= e_i^{\mathrm{T}} \tfrac{1}{1+\|v\|^2} \left[\|v\|^2 I + \tfrac{1 - \|v\|^2}{1+\|v\|^2} v v^{\mathrm{T}}\right] e_j \ . \end{aligned}$$

Each element of $\mathcal{I}_{D,D}$ is

$$
\begin{aligned}
&[\mathcal{I}_{D,D}]_{i,j} \\
&= \tfrac{1}{2}\operatorname{Tr}\left(C^{-1}\tfrac{\partial C}{\partial d_i}C^{-1}\tfrac{\partial C}{\partial d_j}\right) \\
&= \tfrac{1}{2}\operatorname{Tr}\big(C^{-1}(e_ie_i^{\mathrm{T}}D^{-1}C + CD^{-1}e_ie_i^{\mathrm{T}}) \\
&\qquad \cdot C^{-1}(e_je_j^{\mathrm{T}}D^{-1}C + CD^{-1}e_je_j^{\mathrm{T}})\big) \\
&= \operatorname{Tr}\left(C^{-1}e_ie_i^{\mathrm{T}}D^{-1}CC^{-1}(e_je_j^{\mathrm{T}}D^{-1}C + CD^{-1}e_je_j^{\mathrm{T}})\right) \\
&= e_i^{\mathrm{T}}D^{-1}CC^{-1}(e_je_j^{\mathrm{T}}D^{-1}C + CD^{-1}e_je_j^{\mathrm{T}})C^{-1}e_i \\
&= e_i^{\mathrm{T}}D^{-1}e_je_j^{\mathrm{T}}D^{-1}CC^{-1}e_i + e_i^{\mathrm{T}}D^{-1}CD^{-1}e_je_j^{\mathrm{T}}C^{-1}e_i \\
&= e_i^{\mathrm{T}}D^{-2}e_j + e_i^{\mathrm{T}}(I - \tfrac{1}{1+\|v\|^2}vv^{\mathrm{T}})e_je_j^{\mathrm{T}}D^{-1}(I + vv^{\mathrm{T}})D^{-1}e_i \\
&= e_i^{\mathrm{T}}D^{-2}e_j + \left(\delta_{i,j} - \tfrac{v_iv_j}{1+\|v\|^2}\right)d_i^{-1}d_j^{-1}(\delta_{i,j} + v_iv_j) \\
&= e_i^{\mathrm{T}}D^{-2}e_j + \left(\delta_{i,j}^2 + \tfrac{\|v\|^2\delta_{i,j}v_iv_j}{1+\|v\|^2} - \tfrac{v_i^2v_j^2}{1+\|v\|^2}\right)d_i^{-1}d_j^{-1} \\
&= \left(2\delta_{i,j}^2 + \tfrac{\|v\|^2\delta_{i,j}v_iv_j}{1+\|v\|^2} - \tfrac{v_i^2v_j^2}{1+\|v\|^2}\right)d_i^{-1}d_j^{-1} \\
&= e_i^{\mathrm{T}}\tfrac{1}{1+\|v\|^2}D^{-1}\left[2(1+\|v\|^2)I + \|v\|^2V^2 - Vvv^{\mathrm{T}}V\right]D^{-1}e_j \ .
\end{aligned}
$$

Each element of $\mathcal{I}_{D,v}$ is

$$
\begin{aligned}
&[\mathcal{I}_{D,v}]_{i,j} \\
&= \tfrac{1}{2}\operatorname{Tr}\big(C^{-1}(e_ie_i^{\mathrm{T}}D^{-1}C + CD^{-1}e_ie_i^{\mathrm{T}}) \\
&\qquad \cdot C^{-1}D(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})D\big) \\
&= \operatorname{Tr}\left(C^{-1}e_ie_i^{\mathrm{T}}D^{-1}CC^{-1}D(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})D\right) \\
&= e_i^{\mathrm{T}}D^{-1}CC^{-1}D(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})DC^{-1}e_i \\
&= e_i^{\mathrm{T}}(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})DC^{-1}e_i \\
&= e_i^{\mathrm{T}}(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})(I - \tfrac{1}{1+\|v\|^2}vv^{\mathrm{T}})D^{-1}e_i \\
&= d_i^{-1}e_i^{\mathrm{T}}(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})e_i - \tfrac{v_i}{1+\|v\|^2}d_i^{-1}e_i^{\mathrm{T}}(e_jv^{\mathrm{T}} + ve_j^{\mathrm{T}})v \\
&= 2d_i^{-1}v_i\delta_{i,j} - \tfrac{d_i^{-1}v_i}{1+\|v\|^2}(\|v\|^2\delta_{i,j} + v_iv_j) \\
&= d_i^{-1}v_i\left(2\delta_{i,j} - \tfrac{1}{1+\|v\|^2}(\|v\|^2\delta_{i,j} + v_iv_j)\right) \\
&= d_i^{-1}v_i\left(2\delta_{i,j} - \tfrac{\|v\|^2}{1+\|v\|^2}\delta_{i,j} - \tfrac{1}{1+\|v\|^2}v_iv_j\right) \\
&= d_i^{-1}v_i\left(\tfrac{2+\|v\|^2}{1+\|v\|^2}\delta_{i,j} - \tfrac{1}{1+\|v\|^2}v_iv_j\right) \\
&= e_i^{\mathrm{T}}\tfrac{1}{1+\|v\|^2}D^{-1}V\left((2+\|v\|^2)I - vv^{\mathrm{T}}\right)e_j
\end{aligned}
$$

Altogether, with $\gamma_v = 1 + \|v\|^2$, we have

$$
\begin{aligned}
\mathcal{I}_{v,v} &= \gamma_v^{-1}\left[\|v\|^2 I + (1-\|v\|^2)\gamma_v^{-1}vv^{\mathrm{T}}\right], \\
\mathcal{I}_{D,v} &= \gamma_v^{-1}D^{-1}V\left[(2+\|v\|^2)I - vv^{\mathrm{T}}\right], \\
\mathcal{I}_{D,D} &= \gamma_v^{-1}D^{-1}\left[2\gamma_v I + \|v\|^2V^2 - Vvv^{\mathrm{T}}V\right]D^{-1}
\end{aligned}
$$

and $\mathcal{I}_{v,D} = \mathcal{I}_{D,v}^{\mathrm{T}}$ is clear by definition. This completes the proof. □

Proof of Lemma 3.3 and 3.4. Lemma 3.3 is the case that $\alpha = 1$ in Lemma 3.4. Hence, it suffices to prove Lemma 3.4.

The Schur complement of $\mathcal{I}_{v,v}$ is $S_{v,v} = \mathcal{I}_{D,D} - \alpha^2\mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}$ as it is stated in the main text. By applying the Sherman-Morisson formula, we have the inverse of $\mathcal{I}_{v,v}$

$$
\begin{aligned}
\mathcal{I}_{v,v}^{-1} &= \tfrac{1+\|v\|^2}{\|v\|^2}\left[I - \frac{\frac{1-\|v\|^2}{1+\|v\|^2}\overline{v}\overline{v}^{\mathrm{T}}}{1+\frac{1-\|v\|^2}{1+\|v\|^2}}\right] \\
&= \tfrac{1+\|v\|^2}{\|v\|^2}\left[I - \tfrac{(1-\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}}\right]
\end{aligned}
$$

provided $v \neq 0$.

First, observe

$$
\begin{aligned}
&\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D} \\
&= \tfrac{1+\|v\|^2}{\|v\|^2}\left[I - \tfrac{(1-\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}}\right]\left(\tfrac{2+\|v\|^2}{1+\|v\|^2}I - \tfrac{\|v\|^2}{1+\|v\|^2}\overline{v}\overline{v}^{\mathrm{T}}\right)VD^{-1} \\
&= \tfrac{1}{\|v\|^2}\left[I - \tfrac{(1-\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}}\right]\left((2+\|v\|^2)I - \|v\|^2\overline{v}\overline{v}^{\mathrm{T}}\right)VD^{-1} \\
&= \tfrac{1}{\|v\|^2}\Big[(2+\|v\|^2)I - \|v\|^2\overline{v}\overline{v}^{\mathrm{T}} - \tfrac{(1-\|v\|^2)(2+\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}} \\
&\qquad + \tfrac{(1-\|v\|^2)\|v\|^2}{2}\overline{v}\overline{v}^{\mathrm{T}}\Big]VD^{-1} \\
&= \tfrac{1}{\|v\|^2}\Big[(2+\|v\|^2)I \\
&\qquad - \tfrac{2\|v\|^2+2-\|v\|^2-\|v\|^4-\|v\|^2+\|v\|^4}{2}\overline{v}\overline{v}^{\mathrm{T}}\Big]VD^{-1} \\
&= \tfrac{1}{\|v\|^2}\left[(2+\|v\|^2)I - \overline{v}\overline{v}^{\mathrm{T}}\right]VD^{-1}
\end{aligned}
$$

Then,

$$
\begin{aligned}
&\mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D} \\
&= \tfrac{1}{\|v\|^2}D^{-1}V\left(\tfrac{2+\|v\|^2}{1+\|v\|^2}I - \tfrac{1}{1+\|v\|^2}vv^{\mathrm{T}}\right) \\
&\quad \cdot\left[(2+\|v\|^2)I - \overline{v}\overline{v}^{\mathrm{T}}\right]VD^{-1} \\
&= \tfrac{1}{\|v\|^2(1+\|v\|^2)}D^{-1}V\left((2+\|v\|^2)I - \|v\|^2\overline{v}\overline{v}^{\mathrm{T}}\right) \\
&\quad \cdot\left[(2+\|v\|^2)I - \overline{v}\overline{v}^{\mathrm{T}}\right]VD^{-1} \\
&= \tfrac{1}{\|v\|^2(1+\|v\|^2)}D^{-1}V\big((2+\|v\|^2)^2I - \|v\|^2(2+\|v\|^2)\overline{v}\overline{v}^{\mathrm{T}} \\
&\quad - (2+\|v\|^2)\overline{v}\overline{v}^{\mathrm{T}} + \|v\|^2\overline{v}\overline{v}^{\mathrm{T}}\big)VD^{-1} \\
&= \tfrac{1}{\|v\|^2(1+\|v\|^2)}D^{-1}V\big((2+\|v\|^2)^2I \\
&\quad - (2+\|v\|^2)^2I - (2+2\|v\|^2+\|v\|^4)\overline{v}\overline{v}^{\mathrm{T}}\big)VD^{-1} \\
&= \tfrac{1}{(1+\|v\|^2)}D^{-1}\overline{V}\big((2+\|v\|^2)^2I \\
&\quad - (2+2\|v\|^2+\|v\|^4)\overline{v}\overline{v}^{\mathrm{T}}\big)\overline{V}D^{-1}
\end{aligned}
$$

Finally we have

$$
\begin{aligned}
&S_{v,v} \\
&= D^{-1}\Big[2I + \tfrac{\|v\|^4}{1+\|v\|^2}\overline{V}^2 - \tfrac{\|v\|^4}{1+\|v\|^2}\overline{V}\overline{v}\overline{v}^{\mathrm{T}}\overline{V} \\
&\quad - \alpha^2\tfrac{(2+\|v\|^2)^2}{1+\|v\|^2}\overline{V}^2 + \alpha^2\tfrac{2+2\|v\|^2+\|v\|^4}{1+\|v\|^2}\overline{V}\overline{v}\overline{v}^{\mathrm{T}}\overline{V}\Big]D^{-1} \\
&= D^{-1}\Big[2I + \tfrac{\|v\|^4-\alpha^2(2+\|v\|^2)^2}{1+\|v\|^2}\overline{V}^2 \\
&\quad - \tfrac{\|v\|^4-\alpha^2(2+2\|v\|^2+\|v\|^4)}{1+\|v\|^2}\overline{V}\overline{v}\overline{v}^{\mathrm{T}}\overline{V}\Big]D^{-1} \\
&= 2D^{-1}\left[A + b\overline{\overline{v}}\overline{\overline{v}}^{\mathrm{T}}\right]D^{-1} \ .
\end{aligned}
$$

This ends the proof. □

Proof of Proposition 3.5. In the main text, a sufficient condition is derived:

$$
\begin{aligned}
\alpha^2 &\leq [\|v\|^4 + (2-\gamma)(1+\|v\|^2)/\max(\overline{\overline{v}}_j)]/(2+\|v\|^2)^2 \\
\alpha^2 &\geq [\|v\|^4 - (1-\gamma)\gamma(1+\|v\|^2)]/(\|v\|^4 + 2\|v\|^2 + 2) \ .
\end{aligned}
$$

It is obvious that the first inequality is satisfied by α in (7). Therefore, we are going to show that the second one is satisfied by this α.

First, notice that the RHS of the second inequality is smaller than one. Therefore, the inequality holds if $\alpha = 1$. If $\alpha < 1$, the inequality reduces to

$$
\begin{aligned}
&\alpha^2 \geq [\|v\|^4 - (1-\gamma)\gamma(1+\|v\|^2)]/(\|v\|^4 + 2\|v\|^2 + 2) \\
\Leftrightarrow &[\|v\|^4 + (2-\gamma)(1+\|v\|^2)/\max(\overline{v}_j)]/(2+\|v\|^2)^2 \\
&\geq [\|v\|^4 - (1-\gamma)\gamma(1+\|v\|^2)]/(\|v\|^4 + 2\|v\|^2 + 2) \\
\Leftrightarrow &(\|v\|^4 + 2\|v\|^2 + 2)\|v\|^4 \\
&+ (\|v\|^4 + 2\|v\|^2 + 2)(2-\gamma)(1+\|v\|^2)/\max(\overline{v}_j) \\
&\geq (\|v\|^4 + 4\|v\|^2 + 4)\|v\|^4 \\
&\quad - (\|v\|^4 + 4\|v\|^2 + 4)(1-\gamma)\gamma(1+\|v\|^2) \\
\Leftrightarrow &-2(1+\|v\|^2)\|v\|^4 \\
&+ (\|v\|^4 + 2\|v\|^2 + 2)(2-\gamma)(1+\|v\|^2)/\max(\overline{v}_j) \\
&\geq -(\|v\|^4 + 4\|v\|^2 + 4)(1-\gamma)\gamma(1+\|v\|^2) \\
\Leftrightarrow &-2\|v\|^4 + (\|v\|^4 + 2\|v\|^2 + 2)(2-\gamma)/\max(\overline{v}_j) \\
&+ (\|v\|^4 + 4\|v\|^2 + 4)(1-\gamma)\gamma \geq 0 \\
\Leftarrow &-2\|v\|^4 + (\|v\|^4 + 2\|v\|^2 + 2)(2-\gamma) \\
&+ (\|v\|^4 + 4\|v\|^2 + 4)(1-\gamma)\gamma \geq 0 \\
\Leftrightarrow &-(\|v\|^4 + 4\|v\|^2 + 4)\gamma^2 + 2(1+\|v\|^2)\gamma + 4(1+\|v\|^2) \geq 0 \\
\Leftrightarrow &(\|v\|^2+2)^2\gamma^2 - 2(1+\|v\|^2)\gamma - 4(1+\|v\|^2) \leq 0
\end{aligned}
$$

By solving the right-most side inequality, we obtain the sufficient condition for γ

$$
\begin{aligned}
&\frac{(1+\|v\|^2) - \sqrt{(1+\|v\|^2)^2 + 4(2+\|v\|^2)^2(1+\|v\|^2)}}{(2+\|v\|^2)^2} \leq \gamma \\
\leq &\frac{(1+\|v\|^2) + \sqrt{(1+\|v\|^2)^2 + 4(2+\|v\|^2)^2(1+\|v\|^2)}}{(2+\|v\|^2)^2} \ .
\end{aligned}
$$

The LHS is negative, so $\gamma = (1+\|v\|^2)^{-1/2}$ satisfies the left inequality. The RHS is

$$
\begin{aligned}
&\frac{(1+\|v\|^2) + \sqrt{(1+\|v\|^2)^2 + 4(2+\|v\|^2)^2(1+\|v\|^2)}}{(2+\|v\|^2)^2} \\
>&\frac{\sqrt{4(\|v\|^2+2)^2(1+\|v\|^2)}}{(2+\|v\|^2)^2} = \frac{2\sqrt{1+\|v\|^2}}{2+\|v\|^2} \\
\geq &\frac{1}{\sqrt{1+\|v\|^2}} = \gamma \ .
\end{aligned}
$$

Hence, the right inequality is satisfied. Therefore, the second inequality is satisfied, which completes the proof. □

Proof of Theorem 3.6. In the main text we have shown that $\tilde{\nabla}_m \ln p_\theta(x) = x - m$. We are going to show that $\tilde{\nabla}_v \ln p_\theta(x) = \|v\|^{-1} t$ and $\tilde{\nabla}_{\theta_D} \ln p_\theta(x) = Ds$ with s and t computed in step 4 and step 5.

The modified natural gradient w.r.t. θ_C is computed as

$$
\begin{aligned}
&\begin{bmatrix} \mathcal{I}_{v,v} & \alpha\mathcal{I}_{v,D} \\ \alpha\mathcal{I}_{D,v} & \mathcal{I}_{D,D} \end{bmatrix}^{-1} \begin{bmatrix} \nabla_v \ln p_\theta(x) \\ \nabla_{\theta_D} \ln p_\theta(x) \end{bmatrix} \\
&= \begin{bmatrix} \mathcal{I}_{v,v}^{-1}\nabla_v \ln p_\theta(x) \\ 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D} \\ I \end{bmatrix} \\
&\quad \cdot S_{v,v}^{-1} \begin{bmatrix} -\mathcal{I}_{D,v}\alpha\mathcal{I}_{v,v}^{-1} & I \end{bmatrix} \begin{bmatrix} \nabla_v \ln p_\theta(x) \\ \nabla_{\theta_D} \ln p_\theta(x) \end{bmatrix} \\
&= \begin{bmatrix} \mathcal{I}_{v,v}^{-1}\nabla_v \ln p_\theta(x) \\ 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D} \\ I \end{bmatrix} S_{v,v}^{-1} \\
&\quad \cdot \left(\nabla_{\theta_D} \ln p_\theta(x) - \alpha\mathcal{I}_{D,v}\mathcal{I}_{v,v}^{-1}\nabla_v \ln p_\theta(x)\right) \\
&= \begin{bmatrix} \mathcal{I}_{v,v}^{-1}\nabla_v \ln p_\theta(x) \\ 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}D \\ I \end{bmatrix} D^{-1}S_{v,v}^{-1} \\
&\quad \cdot D^{-1}\Big(\underbrace{D\nabla_{\theta_D} \ln p_\theta(x)}_{=:s_1} - \alpha D\mathcal{I}_{D,v}\underbrace{\mathcal{I}_{v,v}^{-1}\nabla_v \ln p_\theta(x)}_{=:t_2/\|v\|}\Big) \\
&= \begin{bmatrix} t_2/\|v\| \\ 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}D \\ D \end{bmatrix} D^{-1}S_{v,v}^{-1}D^{-1} \\
&\quad \cdot \underbrace{(s_1 - (\alpha/\|v\|)D\mathcal{I}_{D,v}t_2)}_{=:s_3} \\
&= \begin{bmatrix} t_2/\|v\| \\ 0 \end{bmatrix} + \begin{bmatrix} -\alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}D \\ D \end{bmatrix} \underbrace{D^{-1}S_{v,v}^{-1}D^{-1}s_3}_{=:s_4} \\
&= \begin{bmatrix} \underbrace{t_2/\|v\| - \alpha\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}Ds_4}_{=:t_5/\|v\|} \\ Ds_4 \end{bmatrix} \ .
\end{aligned}
$$

According to Lemma 3.1, it is easy to see that s_1 is computed by s in step 1. In the proof of Lemma 3.4, we have derived the explicit forms of $\mathcal{I}_{v,v}^{-1}$. From this and Lemma 3.1, we have

$$
\begin{aligned}
t_2 &= \|v\| \left(\tfrac{1+\|v\|^2}{\|v\|^2}\left[I - \tfrac{(1-\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}}\right]\right) \\
&\quad \cdot \left(\gamma_v^{-1}\left[\langle y, v\rangle\, y - \gamma_v^{-1}(\langle y, v\rangle^2 + \gamma_v)v\right]\right) \\
&= \tfrac{1}{\|v\|}\left[I - \tfrac{(1-\|v\|^2)}{2}\overline{v}\overline{v}^{\mathrm{T}}\right]\left[\langle y, v\rangle\, y - \gamma_v^{-1}(\langle y, v\rangle^2 + \gamma_v)v\right] \\
&= \tfrac{1}{\|v\|}\Big[\langle y, v\rangle\, y - \gamma_v^{-1}(\langle y, v\rangle^2 + \gamma_v)v \\
&\qquad - \tfrac{(1-\|v\|^2)}{2\|v\|}\langle y, v\rangle^2\,\overline{v} + \tfrac{(1-\|v\|^2)}{2\gamma_v}(\langle y, v\rangle^2 + \gamma_v)v\Big] \\
&= \tfrac{1}{\|v\|}\Big[\langle y, v\rangle\, y - \tfrac{(1-\|v\|^2)}{2\|v\|}\langle y, v\rangle^2\,\overline{v} - \tfrac{1}{2}(\langle y, v\rangle^2 + \gamma_v)v\Big] \\
&= \langle y, \overline{v}\rangle\, y - 2^{-1}(\langle y, \overline{v}\rangle^2 + \gamma_v)\overline{v} \ .
\end{aligned}
$$

Hence $t_2 = t$ in step 2. According to Lemma 3.2,

$$
\begin{aligned}
s_3 &= s_1 - (\alpha/\|v\|)\gamma_v^{-1}V\left[(2+\|v\|^2)I - vv^{\mathrm{T}}\right]t_2 \\
&= s_1 - \alpha\gamma_v^{-1}\overline{V}\left[(2+\|v\|^2)I - \|v\|^2\overline{v}\overline{v}^{\mathrm{T}}\right]t_2 \\
&= s_1 - \alpha\gamma_v^{-1}\left[(2+\|v\|^2)\overline{v} \odot t_2 - \|v\|^2\langle\overline{v}, t_2\rangle\,\overline{v}\right] \ ,
\end{aligned}
$$

which is equivalent to s in step 3. According to Proposition 3.5,

$$
\begin{aligned}
s_4 &= \left[A^{-1} - (1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v})^{-1}bA^{-1}\overline{v}\overline{v}^{\mathrm{T}}A^{-1}\right]s_3 \\
&= A^{-1}s_3 - (1 + b\overline{v}^{\mathrm{T}}A^{-1}\overline{v})^{-1}b\left\langle A^{-1}\overline{v}, s_3\right\rangle A^{-1}\overline{v}
\end{aligned}
$$

and $s_4 = s$ in step 4. In the proof of Lemma 3.4, we have derived the explicit forms of $\mathcal{I}_{v,v}^{-1}\mathcal{I}_{v,D}$. With this, we have

$$\begin{aligned} t_5 &= t_2 - \alpha\|v\| \left(\tfrac{1}{\|v\|^2} \left[(2 + \|v\|^2) I - \overline{v}\overline{v}^{\mathrm{T}} \right] V \right) s_4 \\ &= t_2 - \alpha \left[(2 + \|v\|^2) I - \overline{v}\overline{v}^{\mathrm{T}} \right] \overline{V} s_4 \\ &= t_2 - \alpha \left[(2 + \|v\|^2) \overline{v} \odot s_4 - \left\langle \overline{\overline{v}}, s_4 \right\rangle \overline{v} \right] \quad . \end{aligned}$$

This is equivalent to t in step 5. Therefore, $\tilde{\nabla}_v \ln p_\theta(x) = \|v\|^{-1} t_5 = \|v\|^{-1} t$ and $\tilde{\nabla}_{\theta_D} \ln p_\theta(x) = D s_4 = D s$. $\square$