Extended Hamiltonian Learning on Riemannian Manifolds: Numerical Aspects

Simone Fiori

***Abstract*—The present paper delivers the second part of a study initiated with the contribution S. Fiori, *Extended Hamiltonian Learning on Riemannian Manifolds: Theoretical Aspects*, IEEE Transactions on Neural Networks, Vol. 22, No. 5, pp. 687 – 700, May 2011, which aimed at introducing a general framework to develop a theory of learning on differentiable manifolds by extended Hamiltonian stationary-action principle. The present paper discusses the numerical implementation of the extended Hamiltonian learning paradigm by making use of notions from geometric numerical integration to numerically solve differential equations on manifolds. The general-purpose integration schemes as well as the discussion of several cases-of-interest show that the implementation of the dynamical learning equations exhibits a rich structure. The behavior of the discussed learning paradigm is illustrated via several numerical examples and discussions of cases-of-study. The numerical examples confirm the theoretical developments presented in this paper as well as of the first part of the present study.**

***Index Terms*—Extended Hamiltonian (second-order) learning; Riemannian manifold; Learning by constrained criterion optimization; Geometric numerical integration.**

I. Introduction

A CLASSICAL learning theory paradigm is based on criterion optimization over an Euclidean space by a gradient-based (first-order) learning algorithm. A more advanced instance of parameter learning is by constrained optimization. In such context, the constraints on parameters' values reflect the natural constraints presented by the learning problem. In this event, differential geometry is an appropriate mathematical instrument to formulate and to implement a learning theory. The contribution [21] aimed at introducing a general framework to develop a theory of learning on differentiable manifolds by extended Hamiltonian stationary-action principle, which leads to a second-order learning paradigm. The first part of the paper [21] presents a derivation of general equations of learning on Riemannian manifolds and discusses the relationship between the theory of dynamical learning by second-order differential equations on manifolds and the classical learning theory based on gradient-steepest-descent optimization. Over a parameter manifold M, the extended Hamiltonian learning dynamics is described by the system of differential equations:

$$\left\{ \begin{array}{rcl} \dot{x} & = & \frac{1}{\mathcal{M}} v, \\ \dot{v} & = & -\mathcal{M}\Gamma_x(v,v) - \nabla_x V - \mu v, \end{array} \right. \quad (1)$$

where the variable $x \in M$ represents a set of learning system's learnable parameters, variable v is proportional to the instantaneous learning velocity, constant $\mathcal{M} \geq 0$ denotes a mass term, function Γ_x denotes the Christoffel form associated to the metric of the real Riemannian manifold M, function $V : M \to \mathbb{R}$ describes a potential energy field, vector field $\nabla_x V$ denotes the Riemannian gradient of the potential energy function and constant $\mu > 0$ denotes a viscosity term. The potential energy function V arises from the casting of a given learning problem into an optimization problem. The learning dynamics of the system (1), described by the state-pair (x, v), evolves toward a stationarity state $(x_\star, 0)$. The point $x_\star \in M$ coincides with a local minimum of the potential energy function V over the manifold M. An interesting finding of paper [21] is the resemblance of dynamical learning with classical learning with 'momentum term'. A convergence analysis was carried out in terms of Lyapunov function and a fine-convergence analysis in the proximity of a stationary point of the learning-goal function, represented by a potential energy function for the dynamical system, was conducted as well. The fine-convergence analysis shows that second-order learning may improve the convergence ability of first-order learning if certain conditions are met. The second part of the paper [21] discusses several cases of learning by extended Hamiltonian systems on manifolds of interest in the scientific literature. The purpose of the calculations was to show how to derive the Christoffel form, the Riemannian gradient and the geodesic expressions by working in intrinsic coordinates only by means of the variational principle stated in the first part of the paper.

The formulation of the extended Hamiltonian learning equations on differentiable manifolds requires instruments from differential geometry. Likewise, the numerical implementation of the extended Hamiltonian systems on a computation platform requires instruments from *geometric numerical integration* [26]. The reason is that the numerical integration techniques which insist on flat spaces are unable (in general) to cope with curved parameter spaces. For a general parameter manifold, the classical Euler-type stepping method should be replaced with a numerical method that is capable of coping with the curved nature of the parameter space. The first fundamental equation of the system (1) may be solved numerically by a step-forward algorithm based on the notion of *manifold retraction*, that will be explained in section II. The second fundamental equation of the system (1) may be solved numerically by a step-forward algorithm based on the differential-geometrical notion of *parallel translation* (or by a numerical technique known as *vector translation*), whose definition will be recalled in section II. The general-purpose integration schemes presented in section II as well as their particularization to several cases-of-interest discussed in section III will show that the implementation of the dynamical learning equations exhibits a rich structure in terms of mathematical analysis and implementation issues. In particular, section III discusses cases of learning by extended Hamiltonian systems on manifolds of interest in the scientific literature, namely, the Stiefel manifold, the special orthogonal group, the unit hypersphere, the Grassmann manifold, the group of symmetric positive definite matrices, the flag manifold and the real symplectic group of matrices. Section IV illustrates the features of the discussed learning algorithms via comparative numerical experiments and section V concludes the paper and suggests further research topics of interest along the line of the present research.

II. Numerical integration of extended Hamiltonian systems

In general, the extended Hamiltonian learning system (1) is not solvable in closed form, therefore, it is necessary to resort to a nu-

The author is with Dipartimento di Ingegneria dell'Informazione, Facoltà di Ingegneria, Università Politecnica delle Marche, Via Brecce Bianche, Ancona I-60131, Italy. (eMail: s.fiori@univpm.it)

merical approximation of the exact solution. The present section aims at explaining the fundamental mathematical tools to solve differential equations on manifolds and how to apply these tools to the solution of the extended Hamiltonian learning system in subsection II-D. In particular, the numerical implementation makes use of the notion of parallel translation and of the concept of manifold retraction, which are recalled in the subsections II-A, II-B and II-C. Subsection II-E discusses a relationship between the extended Hamiltonian learning method and the gradient method. For the theory of differentiable manifolds and Lie groups, readers may refer to [49].

A. Definitions and notation

Let M be a real differentiable manifold of dimension r. At a point $x \in M$, the tangent space to the manifold M is denoted as T_xM. The symbol TM denotes the tangent bundle defined as $TM \stackrel{\text{def}}{=} \{(x, v) | x \in M, v \in T_xM\}$.

Let the algebraic group (G, m, i, e) be a Lie group, namely, let G be endowed with a differentiable manifold structure, which is further supposed to be Riemannian. Here, operator $m : G \times G \to G$ denotes group multiplication, operator $i : G \to G$ denotes group inverse and $e \in G$ denotes group identity, namely $m(x, i(x)) = e$ for every $x \in G$. The algebraic and the differential-geometric structures need to be compatible, namely, the map $(x, y) \mapsto m(x, i(y))$ needs to be smooth for every $x, y \in G$. To the Lie group G, a Lie algebra $\mathfrak{g} \stackrel{\text{def}}{=} T_eG$ is associated.

A Riemannian manifold M is endowed with a inner product $\langle \cdot, \cdot \rangle_x : T_xM \times T_xM \to \mathbb{R}$. Symbol $\mathbb{E}^r$ denotes a real Euclidean space of dimension r and symbol $\langle \cdot, \cdot \rangle^E$ denotes an Euclidean inner product in $\mathbb{E}^r$. Recall that any tangent space T_xM of a real differentiable manifold $M \ni x$ of dimension r is isomorphic to an Euclidean space $\mathbb{E}^r$. A local metric $\langle \cdot, \cdot \rangle_x$ also defines a local norm $\|v\|_x \stackrel{\text{def}}{=} \sqrt{\langle v, v \rangle_x}$, for $v \in T_xM$. Let $\psi : M \to \mathbb{R}$ denote a differentiable function. Symbol $\nabla_x \psi \in T_xM$ denotes the Riemannian gradient of function ψ with respect to a metric $\langle \cdot, \cdot \rangle_x$. The Riemannian gradient is defined as the unique tangent vector in T_xM that satisfies the metric compatibility condition

$$\langle \nabla_x \psi, v \rangle_x = \langle \partial_x \psi, v \rangle^{\mathrm{E}} \text{ for any } v \in T_xM, \tag{2}$$

where symbol ∂_x denotes Euclidean gradient evaluated at x.

In the following, an over-dot will denote the derivative $\frac{d}{dt}$, while a double over-dot will denote the derivative $\frac{d^2}{dt^2}$. The standard notation for covariant and contravariant tensors indices as well as Einstein's convention on summation indices are made use of throughout the present section.

For $u, v \in T_xM$, it holds $\langle u, v \rangle_x = g_{ij} u^i v^j$, where g_{ij} denote the components of the metric tensor associated to the inner product $\langle \cdot, \cdot \rangle_x$. The functions g_{ij} depend on coordinates x^k and the symmetry property $g_{ij} = g_{ji}$ holds. The functions g^{ij} denote the components of the dual metric tensor, namely, $g_{ik} g^{kj} = 0$ for $i \neq j$ and $g_{ik} g^{ki} = 1$. The Christoffel symbols of the first kind associated to a metric tensor of components g_{ij} are defined as:

$$\Gamma_{kji} \stackrel{\text{def}}{=} \frac{1}{2} \left(\frac{\partial g_{ji}}{\partial x^k} + \frac{\partial g_{ik}}{\partial x^j} - \frac{\partial g_{kj}}{\partial x^i} \right),$$

in the Levi-Civita formulation. The Christoffel symbols of the second kind are defined as $\Gamma^h_{ik} \stackrel{\text{def}}{=} g^{hj} \Gamma_{ikj}$. The Christoffel symbols of the second kind are symmetric in the covariant indices, namely, $\Gamma^h_{ij} = \Gamma^h_{ji}$. The Christoffel form $\Gamma_x : T_xM \times T_xM \to \mathbb{E}^r$ is defined in extrinsic coordinates by $[\Gamma_x(v, w)]^k \stackrel{\text{def}}{=} \Gamma^k_{ij} v^i w^j$.

On a Riemannian manifold M with Christoffel form $\Gamma_\cdot(\cdot, \cdot)$, a curve $c : [0, 1] \to M$ that satisfies the second-order differential equation

$$\ddot{c} + \Gamma_c(\dot{c}, \dot{c}) = 0 \tag{3}$$

is termed *geodesic*. A geodesic curve that satisfies the initial conditions $c(0) = x \in M$ and $\dot{c}(0) = v \in T_xM$ is denoted as $c_{x,v}(t)$. For some manifolds of interest in machine learning and for some specific metric structures, geodesic curves may be expressed in closed forms. However, in some cases either the closed-form expression requires extensive numerical computations to be evaluated or its closed-form is unknown. For these reasons, some constructions have been envisaged to approximate geodesic curves. See, for example, the method based on repeated manifold-projections explained in [32] tailored to the case of shape-manifolds or the more general method explained in [40] that solves the problem of finding a geodesic arc joining given endpoints in a connected complete Riemannian manifold.

Given two points $x_1, x_2 \in M$ and provided they may be joined by a geodesic arc $c_{x,v}(t)$, with $t \in [0, 1]$, $c_{x,v}(0) = x_1$ and $c_{x,v}(1) = x_2$, the Riemannian distance between the given points in M may be defined as:

$$d(x_1, x_2) \stackrel{\text{def}}{=} \int_0^1 \langle \dot{c}_{x,v}, \dot{c}_{x,v} \rangle^{\frac{1}{2}}_{c_{x,v}} dt. \tag{4}$$

In fact, the notion of geodesics on a manifold M turns the manifold into a metric space.

The kinetic energy functional associated to the system (1) is the symmetric bilinear form $\frac{1}{2} \mathcal{M} \langle \dot{x}, \dot{x} \rangle_x$, namely $K_x(\dot{x}, \dot{x}) \stackrel{\text{def}}{=} \frac{1}{2} \mathcal{M} g_{ij} \dot{x}^i \dot{x}^j$. On a Riemannian manifold, the metric tensor is positive-definite, hence $K_x(\dot{x}, \dot{x}) \geq 0$ on every trajectory $x(t) \in M$. (Such property does not hold true on other manifolds of interest in neural learning, such as pseudo-Riemannian manifolds [19].) The potential energy function V depends on the coordinates x^k only. The notation used throughout the paper to denote matrices, namely, by lower-case letters instead upper-case letters, is used to keep consistency with the notation introduced in paper [21].

B. Parallel translation of a tangent vector along a geodesic arc

Given a curve on a Riemannian manifold M, a point $x \in M$ on the curve and a tangent vector $v \in T_xM$ to the curve, if such a vector is moved to another point on the curve by a parallel translation in the sense of ordinary geometry, in general it will not be tangent to the curve anymore. This observation suggests that it is necessary to define a notion of "parallel translation along a curve" that is compatible with the geometry of the manifold that the curve belongs to[1].

In the present work, the main interest lies on parallel translation of a tangent vector along a geodesic curve on a Riemannian manifold only, thus the present subsection focuses on seeking for a solution to such a problem. The basic requirement of parallel translation of a vector along a curve is that it keeps the translated vector tangent to the curve at any point. Clearly, this condition by itself is too weak to give rise to a unique solution to the problem of parallel translation. In Riemannian geometry, it is additionally required that parallel translation preserves the inner product between the translated tangent vector and the curve's tangent. A parallel-translation rule may be constructed as follows. Let $x(t) \in M$ denote a given curve on a Riemannian manifold M endowed with a metric tensor of components g_{ij}, with $x(0)$ being simply denoted as x, and let the curve $v(t) \in T_{x(t)}M$, $t \in [0, 1]$ denotes parallel translation of a tangent vector $u \in T_xM$ identified with $v(0)$. The angle-preservation

[1]Indeed, the notion of parallel translation is fundamental in differential geometry, because it enables comparing vectors that belong to different tangent spaces. It is intimately related with the fundamental concept of *covariant derivative* of a vector field with respect to a vector.

property of parallel-translation implies that the sought-for curve $v(t)$ must meet the formal condition:

$$\langle v(t), \dot{x}(t) \rangle_{x(t)} = \text{constant over the curve } x(t). \tag{5}$$

Differentiating the above equation with respect to the parameter t yields:

$$\frac{d}{dt}(g_{ij} v^i \dot{x}^j) = 0, \tag{6}$$

namely:

$$\frac{\partial g_{ij}}{\partial x^k} \dot{x}^k \dot{x}^j v^i + g_{ij} \dot{x}^j \dot{v}^i + g_{ih} \ddot{x}^h v^i = 0. \tag{7}$$

Now, assume that the components x^h describe a geodesic arc on the Riemannian manifold M. Therefore, it holds $\ddot{x}^h = -\Gamma^h_{kj} \dot{x}^k \dot{x}^j$. Using such identity in the equation (7) yields:

$$\left(\left(\frac{\partial g_{ij}}{\partial x^k} - g_{ih} \Gamma^h_{kj} \right) \dot{x}^k v^i + g_{ij} \dot{v}^i \right) \dot{x}^j = 0. \tag{8}$$

Straightforward calculations with Christoffel symbols show that the terms within the inner parentheses sum up to Γ_{ikj}, therefore, the above equation may be rewritten as:

$$g_{hj} (\Gamma^h_{ik} v^i \dot{x}^k + \dot{v}^h) \dot{x}^j = 0, \tag{9}$$

which is equivalent to:

$$\langle \Gamma_x(v, \dot{x}) + \dot{v}, \dot{x} \rangle_x = 0. \tag{10}$$

A sufficient condition for the curve $v(t)$ to represent an angle-preserving parallel translation of the given vector $u \in T_x M$ along the given curve $x(t) \in M$ is:

$$\dot{v} + \Gamma_x(v, \dot{x}) = 0, \;\; v(0) = u. \tag{11}$$

The above Levi-Civita parallel translation equation may be interpreted as an evolution law of the tangent vector $v(0)$ that keeps the vector tangent to the manifold along a geodesic curve and that preserves the angle with the geodesic curve.

The parallel translation of a vector $u \in T_x M$ along a geodesic arc $c_{x,v}(t)$ is denoted by $\tau_{c_{x,v}(t)}(u)$ and is illustrated in the Figure 1. A noticeable property of any geodesic curve $c_{x,v}(t)$, that will be

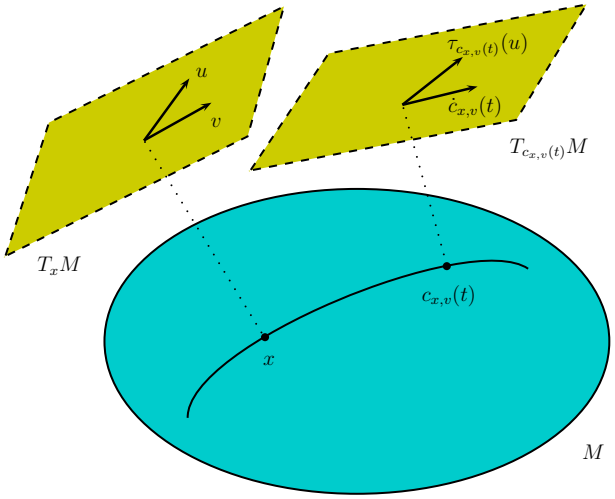

Fig. 1. Notion of parallel translation on a Riemannian manifold M. The tangent vector $u \in T_x M$ is parallel-translated along the geodesic arc $c_{x,v}(t)$.

invoked later in the paper, is that it parallel-translates itself. Such a property may be verified by observing that setting $v = \dot{x}$ in the equation (11) yields equation (3).

In order to set up the parallel translation equation (11), in the case that the symmetric Christoffel form $\Gamma_x(v, v)$ with $x \in M$ and $v \in T_x M$, is known, it is necessary to calculate from it the bilinear Christoffel form $\Gamma_x(u, w)$, with $x \in M$ and $u, w \in T_x M$. Such a result may be achieved via the following identity:

$$4\Gamma_x(u, w) = \Gamma_x(u + w, u + w) - \Gamma_x(u - w, u - w), \tag{12}$$

where clearly it makes sense to compute the quantities $u \pm w$ as the summands belong to the same tangent space.

The parallel translation equation (11) is, in general, difficult to solve, hence the operator $\tau_{c_{x,v}(t)}$ may not be available in closed form for manifolds of interest. Approximations of the exact parallel translation are available, as the 'Schild's ladder' construction [35] and the 'vector translation' method [44]. In practice, vector translation is based on the following idea:

1) Embed the manifold M of dimension r into the Euclidean ambient space $\mathbb{E}^r$.
2) Parallel-translate the vector $u \in T_x M$ in the sense of ordinary geometry in the ambient space to a point $y \in M$.
3) Project the vector u into the tangent space $T_y M$ by means of a suitable projector $\pi_y : \mathbb{E}^r \to T_y M$, for $y \in M$.

The vector-translation operator associated to the above procedure is, thus, $\pi_y(u)$, which moves the vector u from $T_x M$ to $T_y M$. The projection operator $\pi_y : \mathbb{E}^r \to T_y M$ is defined by the two conditions $\pi_y(z) \in T_y M$ for $y \in M$ and $z \in \mathbb{E}^r$, and $\langle v, z \rangle^E = \langle v, \pi_y(z) \rangle_y$ for all $v \in T_y M$. Depending on the manifold's geometric structure, vector translation may be much less expensive to compute than parallel translation.

C. Manifold retractions and exponential maps

A retraction map $R : TM \to M$ is defined such that any restriction $R_x : T_x M \to M$, for $x \in M$, satisfies conditions [44]:

1) Any restriction R_x is defined in some open ball $B(0, \varrho_x)$ of radius $\varrho_x > 0$ about $0 \in T_x M$ and is continuously differentiable;
2) It holds $R_x(v) = x$ if $v = 0$;
3) Let $v(t) \in B(0, r_x) \subset T_x M$ denote any smooth curve, with $t \in \mathbb{I} \ni 0$ and $v(0) = 0$. The curve $y(t) = R_x(v(t))$, for $t \in \mathbb{I}$, lies in a neighborhood of $x = y(0)$. It holds $\dot{y} = dR_x|_v(\dot{v})$ and $dR_x|_v : T_v T_x M \to T_{R_x(v)} M$ denotes a tangent map. For $t = 0$, it holds $dR_x|_0 : T_0 T_x M \to T_{R_x(0)} M$. Identify $T_0 T_x M \cong T_x M$ and $T_{R_x(0)} M = T_x M$. In order for R_x to be a retraction, the map $dR_x|_0$ must equate the identity map in $T_x M$.

In practice, a retraction $R_x(v)$ sends a tangent vector $v \in T_x M$ to a manifold $M \ni x$ into a neighbor of point x, as exemplified in the Figure 2.

On a Riemannian manifold, given a geodesic curve $c_{x,v}(t)$, $t \in [0, 1]$, associated to the metric structure, it is defined a map termed *exponential map* as $c_{x,v}(1)$. Any exponential map of a Riemannian manifold is a retraction (the converse is not necessarily true, though). Another class of retractions is the one based on the projection of Euclidean retractions on the manifold of interest [52].

D. Numerical integration methods

In the massive case (namely, whenever $\mathcal{M} \neq 0$), the extended Hamiltonian learning system (1) may be re-normalized by setting:

$$\tilde{\mu} \stackrel{\text{def}}{=} \frac{\mu}{\mathcal{M}}, \;\; \tilde{V} \stackrel{\text{def}}{=} \frac{V}{\mathcal{M}}, \;\; \tilde{v} \stackrel{\text{def}}{=} \frac{v}{\mathcal{M}}. \tag{13}$$

The equations of learning stay the same except that the mass term $\mathcal{M}$ disappears. It may be assumed, thus, that $\mathcal{M} = 1$ without loss of generality, so that the equations of learning become:

$$\begin{cases} \dot{x} = v, \\ \dot{v} + \Gamma_x(v, v) = -\nabla_x V - \mu v. \end{cases} \tag{14}$$

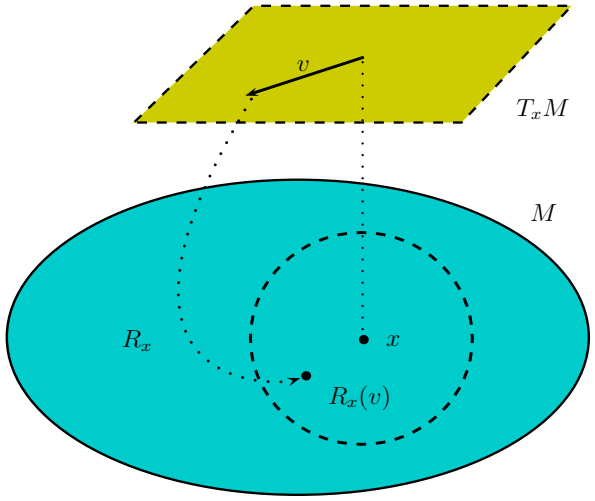

Fig. 2. Notion of retraction on a manifold M. The dashed circle represents the neighborhood of the point $x \in M$ that the retracted tangent vector $v \in T_xM$, namely $R_x(v)$, belongs to.

The system (14) represents an instance of a differential system over a tangent bundle (see, e.g., [7]).

The basic idea to solve numerically the general extended Hamiltonian system of equations over the manifold M is to replace the continuous-time state-variables $x(t) \in M$, $v(t) \in T_{x(t)}M$ with discrete-time state-variables $x_k \in M$, $v_k \in T_{x_k}M$ by ensuring that the state-variables belong to the respective spaces. This operation requires a numerical integration scheme of the fundamental equations.

The first fundamental equation in (14) may be solved by the help of the notion of retraction, namely, by the discrete-time learning rule:

$$x_{k+1} = R_{x_k}(\eta v_k), \ k = 0, 1, 2, 3, \ldots, \tag{15}$$

where $\eta > 0$ denotes an integration step-size (usually $\eta \ll 1$). The retraction-based stepping rule (15) may be regarded as an extension of the Euler method from Euclidean spaces to curved manifolds. In fact, for a Euclidean space $\mathbb{E}^r$, the rule (15) would read as a Euler method, because for $x \in \mathbb{E}^r$ it holds $T_x\mathbb{E}^r = \mathbb{E}^r$, hence $R_x(v) = x + v$. The numerical solution of differential equations on Euclidean spaces may benefit from other methods than Euler's, such as, for instance, the Runge-Kutta family of stepping methods. Likewise, the numerical solution of differential equations on manifolds may benefit from the extension of such methods to curved spaces [9], [38]. In the present context, however, the retraction-based stepping rule (15) is suitable, because the purpose of the numerical implementation is to find an optimal configuration of the learning system and not to approximate the solution of the learning differential equation (14) precisely at any time, unlike in some applications (e.g., in astronomy [26]) where the purpose of a stepping method is to approximate the *whole trajectory* $x(t) \in M$ in a satisfactory way. The second fundamental equation in (14) may be solved in different ways, according to the structure of the tangent bundle of the base-manifold M. In particular, three cases were individuated, which give rise to the three different types of solution described below.

Type I solution: In those cases in which the tangent bundle is trivial, namely, the velocity variable v evolves over a single tangent space (or may be replaced with a variable which evolves over a single linear space, as it is the case for Lie groups), the second fundamental equation in (14) may be solved numerically by Euler stepping on the common tangent space. In fact, the common tangent space to a manifold M of dimension r is a vector space isomorphic to $\mathbb{E}^r$ which affords Euler stepping. The complete stepping method reads:

$$\begin{cases} x_{k+1} = R_{x_k}(\eta v_k), \\ v_{k+1} = (1 - \eta\mu)v_k - \eta\Gamma_{x_k}(v_k, v_k) - \eta\nabla_{x_k}V, \end{cases} \tag{16}$$

for $k = 0, 1, 2, 3, \ldots$. The retraction $R : TM \to M$ may be chosen arbitrarily.

Type II solution: In those cases where the tangent bundle is non-trivial and a closed-form expression for geodesic curves and parallel translation along a geodesic curve are known, the second fundamental equation of dynamics in (14) may be implemented by a parallel-translation-based rule. Namely, the velocity $v_{k+1} \in T_{x_{k+1}}M$ is computed by the parallel translation of the Euler-stepped velocity along the geodesic curve $c_{x_k,v_k}(t)$ extended for the short time $t = \eta$. In order for the procedure to be consistent, it is necessary that the endpoint of the geodesic arc $c_{x_k,v_k}(\eta)$ coincides exactly with the stepped-forward point x_{k+1} computed by the retraction (15). A straightforward way to ensure that such consistency condition holds is to take $R_{x_k}(\eta v_k) = c_{x_k,v_k}(\eta)$ in the stepping rule (15). It is further necessary to pay attention to another aspect related to parallel translation. In some circumstances the formula for the parallel translation $\tau_{c_{x,v}(t)}(u)$ of an arbitrary tangent vector $u \in T_xM$ different from the initial tangent vector $v \in T_xM$ to the geodesic is available. In this case, the complete stepping method reads:

$$\begin{cases} x_{k+1} = c_{x_k,v_k}(\eta), \\ v_{k+1} = \tau_{c_{x_k,v_k}(\eta)}((1 - \eta\mu)v_k - \eta\nabla_{x_k}V), \end{cases} \tag{17}$$

for $k = 0, 1, 2, 3, \ldots$. In some circumstances, however, the parallel translation $\tau_{c_{x,v}(t)}(u)$ of a tangent vector $u \in T_xM$ different from the initial tangent vector $v \in T_xM$ to the geodesic is unavailable and the only available operation is the self-parallel-translation $\tau_{c_{x,v}(t)}(v)$ of the initial tangent vector $v \in T_xM$ to the geodesic. As recalled in section II, any geodesic curve parallel-translates itself, therefore, whenever the closed-form of a geodesic arc on a given manifold with a given metric is known, the self-parallel translation formula is known too, as it holds:

$$\tau_{c_{x,v}(t)}(v) = \dot{c}_{x,v}(t). \tag{18}$$

In this case, the complete stepping method (17) may be approximated as follows:

$$\begin{cases} a_k \stackrel{\text{def}}{=} (1 - \eta\mu)v_k - \eta\nabla_{x_k}V, \\ x_{k+1} = c_{x_k,a_k}(\eta), \\ v_{k+1} = \dot{c}_{x_k,a_k}(\eta), \end{cases} \tag{19}$$

for $k = 0, 1, 2, 3, \ldots$. Such a scheme ensures that only self-parallel-translation is invoked and that $v_{k+1} \in T_{x_{k+1}}M$ indeed.

Type III solution: In the case that the tangent bundle is non-trivial and no closed-form expression for the parallel translation is available, it is possible to resort to the notion of vector translation that replaces the parallel translation operation invoked in the Type II solution. The complete stepping method reads:

$$\begin{cases} x_{k+1} = R_{x_k}(\eta v_k), \\ v_{k+1} = \pi_{x_{k+1}}((1 - \eta\mu)v_k - \eta\nabla_{x_k}V), \end{cases} \tag{20}$$

where $k = 0, 1, 2, 3, \ldots$ and operator $\pi_y : \mathbb{E}^r \to T_yM$, for $y \in M$, denotes projection. Namely, the velocity $v_{k+1} \in T_{x_{k+1}}M$ is computed by projecting the Euler-stepped velocity. The retraction $R : TM \to M$ may be chosen arbitrarily.

E. Relationship with the retracted-gradient method

Recall from [21] that the conditions which allows the extended Hamiltonian system to improve over the gradient method may be met by properly selecting the damping coefficient of the dynamical system. In fact, let H denote the Hessian of the potential energy function V at a point of stationarity and let $\lambda_{\max}$ denote the

largest eigenvalue of the symmetric positive-definite matrix H. If the condition $\mu^2 > 2\lambda_{\max}$ holds, the dynamical system converges faster than the gradient-steepest descent one in the proximity of such stationarity point.

The gradient-steepest-descent rule for a learning problem formulated on the parameter manifold M and described in terms of a cost function $V : M \to \mathbb{R}$ reads:

$$\dot{x} = -\nabla_x V. \tag{21}$$

The above differential equation on the manifold M may be solved numerically by the help of a retraction scheme, namely, by the algorithm:

$$x_{k+1} = R_{x_k}(-\eta \nabla_{x_k} V), \tag{22}$$

with $\eta > 0$ playing the role of learning stepsize and $k = 0, 1, 2, \ldots$. The retraction-based learning rule (22) generalizes the exponential-map-based Euler method on manifolds studied in [24] (see also [3] for an analysis of the error structure of the exponential-map-based Euler method on manifolds).

In principle, it is possible to simplify the expressions for the implementation of the second extended Hamiltonian equation over nontrivial tangent bundles, by choosing the learning stepsize parameters and the damping coefficient such that $\eta\mu = 1$. In this case, equations (17), (19) and (20) reduce, respectively, to parallel translation or vector translation of the scaled Riemannian gradient of the potential energy function $-\frac{1}{\mu}\nabla_x V$. In the further hypothesis that the steps along the geodesic curve are short enough, consider the further simplification that the parallel/vector translation of $\nabla_{x_k} V$ to the point x_{k+1} approximately equals $\nabla_{x_{k+1}} V$, namely that $v_k \approx -\frac{1}{\mu}\nabla_{x_k} V$. In such over-simplified scenario, equation (15) would read:

$$x_{k+1} \approx R_{x_k}\left(-\frac{1}{\mu^2}\nabla_{x_k} V\right), \tag{23}$$

which shows that, under the above hypotheses, the iteration (15) collapses into a retracted-gradient iteration.

Hence, denoting again with $\lambda_{\max}$ the largest eigenvalue of the Hessian matrix associated with the potential energy function V at a point of stationarity, an extended Hamiltonian learning algorithm with stepsize η and damping coefficient μ such that

$$\sqrt{2\lambda_{\max}} < \mu < \eta^{-1} \tag{24}$$

is expected to improve over a retracted Riemannian gradient algorithm with learning stepsize η^2 and hence it is worth invoking.

III. Numerical implementation on special manifolds

The present section explains in details how the numerical integration methods of section II-D can be applied to manifolds of interest. In particular, on the line of the previous contribution [21], the present section deals with the Stiefel manifold, the special orthogonal group, the unit hypersphere, the Grassmann manifold, the group of symmetric positive definite matrices, the flag manifold and the real symplectic group of matrices. An explanation about the circumstances where the specific manifolds studied in the following are important was reported in the manuscript [21].

A. Implementation of the extended Hamiltonian system over the compact Stiefel manifold

The compact Stiefel manifold is defined as $\mathrm{St}(n,p) \stackrel{\text{def}}{=} \{x \in \mathbb{R}^{n\times p} | x^T x = e_p\}$, where $p \leq n$, superscript ^{T} denotes matrix transpose and symbol e_p denotes a $p \times p$ identity matrix. In the case of the compact Stiefel manifold $\mathrm{St}(n,p)$, two metrics are considered, namely, the Euclidean and the canonical metric. In both cases, the expressions of the geodesic arcs are available. Hence, the Hamiltonian dynamical system (14) may be implemented either by a Type II and a Type III solution. An expression of the projection operator $\pi_y : \mathbb{R}^{n\times p} \to T_y\mathrm{St}(n,p)$, for $y \in \mathrm{St}(n,p)$, is:

$$\pi_y(u) = \begin{cases} (e - \frac{1}{2}yy^T)u - \frac{1}{2}yu^T y & \text{for the Euclidean metric,} \\ u - yu^T y & \text{for the canonical metric.} \end{cases} \tag{25}$$

The simplest expressions arise in the case of canonical metric, therefore, the canonical metric is the metric of choice in the present implementation. Moreover, in principle the optimal learning point does not depend on the chosen metrics [15]. In the case of canonical metric, it holds [21]:

$$\Gamma_x(v,v) = -vv^T x - xv^T(e_n - xx^T)v, \tag{26}$$

$$c_{x,v}(t) = [x\ q] \exp\left(t \begin{bmatrix} x^T v & -r^T \\ r & 0_p \end{bmatrix}\right) \begin{bmatrix} e_p \\ 0_p \end{bmatrix}, \tag{27}$$

$$\nabla_x V = \partial_x V - x\partial_x^T V x, \tag{28}$$

where q and r denote the factors of the compact QR decomposition of the matrix $(e_n - xx^T)v$ and 0_p denotes a zero $p \times p$ matrix. The self-parallel-translation formula, as obtained from equation (18), is given by:

$$\tau_{c_{x,v}(t)}(v) = [v\ -xr^T] \exp\left(t \begin{bmatrix} x^T v & -r^T \\ r & 0_p \end{bmatrix}\right) \begin{bmatrix} e_p \\ 0_p \end{bmatrix}. \tag{29}$$

By the expressions of the Riemannian gradient of the potential energy function and of the Christoffel form, it is obtained the extended Hamiltonian system:

$$\begin{cases} \dot{x} &= v, \\ \dot{v} &= -vv^T x - xv^T(e_n - xx^T)v - \\ & \quad (\partial_x V - x(\partial_x V)^T x) - \mu v. \end{cases} \tag{30}$$

According to the expression of the geodesic on the Stiefel manifold endowed with the canonical metric and of the self-parallel-translation formula (29) corresponding to the canonical metric, the extended Hamiltonian system (30) may be implemented by the Type-II solution:

$$\begin{cases} a_k \stackrel{\text{def}}{=} (1-\eta\mu)v_k - \eta\left(\partial_{x_k} V - x_k(\partial_{x_k} V)^T x_k\right), \\ (q_k, r_k) \stackrel{\text{def}}{=} \mathrm{qr}((e_n - x_k x_k^T)a_k, 0), \\ P_k \stackrel{\text{def}}{=} \exp\left(\eta \begin{bmatrix} x_k^T a_k & -r_k^T \\ r_k & 0_p \end{bmatrix}\right) \begin{bmatrix} e_p \\ 0_p \end{bmatrix}, \\ x_{k+1} = [x_k\ q_k] P_k, \\ v_{k+1} = [a_k\ -x_k r_k^T] P_k, \end{cases} \tag{31}$$

where $\eta > 0$ is a stepsize for the extended Hamiltonian learning system, $\mathrm{qr}(\cdot, 0)$ denotes the compact QR factorization operator and $k = 0, 1, 2, \ldots$.

Moreover, according to the expression of the geodesic on the Stiefel manifold endowed with the canonical metric and of the projector (25) corresponding to the canonical metric, the extended Hamiltonian system (30) may be implemented by the Type-III solution:

$$\begin{cases} (q_k, r_k) \stackrel{\text{def}}{=} \mathrm{qr}((e_n - x_k x_k^T)v_k, 0), \\ x_{k+1} = [x_k\ q_k] \exp\left(\eta \begin{bmatrix} x_k^T v_k & -r_k^T \\ r_k & 0_p \end{bmatrix}\right) \begin{bmatrix} e_p \\ 0_p \end{bmatrix}, \\ a_k \stackrel{\text{def}}{=} (1-\eta\mu)v_k - \eta\left(\partial_{x_k} V - x_k(\partial_{x_k} V)^T x_k\right), \\ v_{k+1} = a_k - x_{k+1} a_k^T x_{k+1}, \end{cases} \tag{32}$$

with $\eta > 0$ being a stepsize for the extended Hamiltonian learning system and $k = 0, 1, 2, \ldots$.

In both cases, the learning state is represented by the pair $(x_k, v_k) \in T\mathrm{St}(n,p)$ for $k \in \mathbb{N}$.

B. Implementation of the extended Hamiltonian system over the unit hypersphere

The unit hypersphere is defined as $\mathrm{S}^{n-1}\stackrel{\mathrm{def}}{=}\left\{x\in\mathbb{R}^n|x^Tx=1\right\}$. Although the dynamics over the unit hypersphere S^{n-1} might be viewed as a special case of the dynamics over a Stiefel manifold (namely, $\mathrm{St}(n,1)$), it is worth discussing the numerical implementation of the dynamical-learning system (14) over a unit hypersphere as a separate issue because the expression of the geodesic curve is particularly simple and because a closed-form solution of the parallel-translation equation (11) is available. Hence, in the case of the manifold S^{n-1}, both a Type-II and a Type-III numerical implementations are available.

Recall that, for the unit-hypersphere endowed with the canonical metric, it holds [21]:

$$\Gamma_x(v,v)=-x\|v\|^2, \tag{33}$$

$$c_{x,v}(t)=x\cos(\|v\|t)+v\|v\|^{-1}\sin(\|v\|t), \tag{34}$$

$$\nabla_xV=(e_n-xx^T)\partial_xV, \tag{35}$$

where symbol $\|\cdot\|$ denotes vector 2-norm. By gathering the expressions of the Christoffel operator and of the Riemannian gradient of the potential energy, the following extended Hamiltonian system is obtained:

$$\left\{\begin{array}{rcl}\dot{x}&=&v,\\ \dot{v}&=&\|v\|^2x-(e_n-xx^T)\partial_xV-\mu v.\end{array}\right. \tag{36}$$

By making use of the identity (12), it is found that, for two arbitrary vectors $v,w\in T_x\mathrm{S}^{n-1}$, it holds:

$$\Gamma_x(v,w)=-x(v^Tw), \tag{37}$$

hence the parallel-translation equation (11) for a vector $u\in T_x\mathrm{S}^{n-1}$ along a geodesic arc $c_{x,v}(t)$ particularizes to:

$$\dot{w}(t)-c_{x,v}(t)\dot{c}_{x,v}^T(t)w(t)=0,\ \ w(0)=u, \tag{38}$$

whose solution $w(t)$ represents parallel translation of the tangent vector $u\in T_xM$. The solution of such equation may be adapted from [30] (for a non-unitary initial tangent vector v) and the corresponding parallel-translation operator reads:

$$\tau_{c_{x,v}(t)}(u)=(e_n-\|v\|^{-2}vv^T)u+u^Tv\|v\|^{-2}\dot{c}_{x,v}(t). \tag{39}$$

The component of the vector u orthogonal to the initial velocity of the geodesic $c_{x,v}$ is mapped by an affine translation while the component of the vector u parallel to the initial velocity of the geodesic is 'copied' along the geodesic. By the expression (34), the geodesic's tangent vector field computes as:

$$\dot{c}_{x,v}(t)=-x\|v\|\sin(\|v\|t)+v\cos(\|v\|t). \tag{40}$$

Hence, the parallel translation on the hypersphere S^{n-1} of the tangent vector $u\in T_x\mathrm{S}^{n-1}$ along the geodesic arc $c_{x,v}$ of an extent t computes by:

$$\tau_{c_{x,v}(t)}(u)=[e_n+(\|v\|^{-2}(\cos(\|v\|t)-1)v-\sin(\|v\|t)x)v^T]u. \tag{41}$$

In view of numerical implementation, an expression of a projector $\pi_y:\mathbb{R}^n\to T_y\mathrm{S}^{n-1}$, for $y\in\mathrm{S}^{n-1}$ is of use. According to the canonical metric, it may be chosen as:

$$\pi_y(u)=(e_n-yy^T)u. \tag{42}$$

According to the expression of the geodesic over the unit hypersphere associated to the canonical metric and of the parallel-translation operator (41), the extended Hamiltonian system (36) may be implemented by a Type-II solution as:

$$\left\{\begin{array}{rcl}x_{k+1}&=&x_k\cos(\eta\|v_k\|)+v_k\|v_k\|^{-1}\sin(\eta\|v_k\|),\\ v_{k+1}&=&[e_n+\|v_k\|^{-2}(\cos(\|v_k\|\eta)-1)v_kv_k^T-\\ &&\sin(\|v_k\|\eta)x_kv_k^T][(1-\eta\mu)v_k-\\ &&\eta(e_n-x_kx_k^T)\partial_{x_k}V],\end{array}\right. \tag{43}$$

with $\eta>0$ being a stepsize for the extended Hamiltonian learning system and $k=0,1,2,\ldots$.

In addition, according to the expression of the geodesic over the unit hypersphere associated to the canonical metric and of the projector (42), the extended Hamiltonian system (36) may be implemented by a Type-III solution as:

$$\left\{\begin{array}{rcl}x_{k+1}&=&x_k\cos(\eta\|v_k\|)+v_k\|v_k\|^{-1}\sin(\eta\|v_k\|),\\ v_{k+1}&=&\left(e_n-x_{k+1}x_{k+1}^T\right)[\eta(x_kx_k^T-e_n)\partial_{x_k}V+\\ &&(1-\eta\mu)v_k],\end{array}\right. \tag{44}$$

with $\eta>0$ being a learning stepsize and $k=0,1,2,\ldots$.

In both cases, the learning state is represented by the variable-pair $(x_k,v_k)\in T\mathrm{S}^{n-1}$ for any $k\in\mathbb{N}$.

A differentiable manifold closely related to the unit hypersphere is the oblique manifold [51], defined as:

$$\mathrm{OB}(n)\stackrel{\mathrm{def}}{=}\{x\in\mathbb{R}^{n\times n}|\mathrm{diag}(x^Tx)=e_n\}, \tag{45}$$

where the operator $\mathrm{diag}(\cdot)$ returns the zero matrix except for the main diagonal that is copied from the main diagonal of its argument. The geometry of the oblique manifold may be easily studied on the basis of the geometry of the unit hypersphere.

C. Implementation of the extended Hamiltonian system over the special orthogonal group

The manifold of special orthogonal matrices is defined as $\mathrm{SO}(n)\stackrel{\mathrm{def}}{=}\left\{x\in\mathbb{R}^{n\times n}|x^Tx=e_n,\ \det(x)=1\right\}$. The manifold $\mathrm{SO}(n)$ of special orthogonal matrices may be regarded as a Lie group, with Lie algebra:

$$\mathfrak{so}(n)=\{\omega\in\mathbb{R}^{n\times n}|\omega^T+\omega=0\}, \tag{46}$$

hence it affords an implementation of Type I. The geometrical expressions of use in the present section associated with the special orthogonal group endowed with the canonical metric, read [21]:

$$\Gamma_x(x\omega,x\omega)=-x\omega^2, \tag{47}$$

$$c_{x,x\omega}(t)=x\exp(t\omega), \tag{48}$$

$$\nabla_xV=\frac{1}{2}\left(\partial_xV-x\partial_x^TVx\right). \tag{49}$$

By replacing the expression of the Christoffel form and of the Riemannian gradient into the general extended Hamiltonian learning equations (14), straightforward calculations lead to the special orthogonal group dynamics:

$$\left\{\begin{array}{rcl}\dot{x}&=&x\omega,\\ \dot{\omega}&=&-\frac{1}{2}\left(x^T\partial_xV-\partial_x^TVx\right)-\mu\omega.\end{array}\right. \tag{50}$$

The kinetic energy has expression $K=-\frac{1}{2}\mathrm{tr}(\omega^2)$.

It is worth noting that the second equation of the system (50) describes a dynamics over the Lie algebra $\mathfrak{so}(n)$, which is a linear space (namely, the tangent space to the manifold at the origin), while the second equation of the general system (14) describes a dynamics over a collection of tangent spaces, different from each other. This fact noticeably simplifies the numerical integration of the equations (50) with respect to the general case. In particular, the second equation may be integrated numerically by a Euler stepping method, while the first one may be integrated via a suitable retraction

of the special orthogonal group. Namely, the system (50) may be implemented by a Type-I solution as:

$$\left\{\begin{array}{lcl} x_{k+1} & = & R_{x_k}(\eta x_k \omega_k), \\ \omega_{k+1} & = & \omega_k + \eta\left(-\frac{1}{2}\left(x_k^T \partial_{x_k} V - (\partial_{x_k} V)^T x_k\right) - \mu\omega_k\right), \end{array}\right. \tag{51}$$

where $\eta > 0$ plays the role of learning stepsize for the extended Hamiltonian learning system. The learning state is represented by the pair (x_k, ω_k), with $x_k \in \mathrm{SO}(n)$ and $\omega_k \in \mathfrak{so}(n)$ for $k \in \mathbb{N}$.

For a list of suitable retractions of the orthogonal group (besides the exponential map associated to the geodesic), readers might see the publication [17]. For example, an alternative approach to the exponential-map based retraction would be the Cayley retraction. However, such solution does not look appealing because of some numerical problems related to the singularities of the Cayley map which were recently underlined in [29].

A differentiable manifold closely related to the special orthogonal group is the special Euclidean manifold, denoted as $\mathrm{SE}(n)$, that finds application, primarily, in computer graphics and in robotics (see, e.g., [54]). The special Euclidean manifold is a set of $(n+1) \times (n+1)$ matrices defined as:

$$\mathrm{SE}(n) \stackrel{\mathrm{def}}{=} \left\{ \left[\begin{array}{cc} r & \delta \\ 0 & 1 \end{array}\right] \middle| r \in \mathrm{SO}(n),\ \delta \in \mathbb{R}^n \right\}. \tag{52}$$

The geometrical features of the special Euclidean manifold may be deduced from those of the special orthogonal group for what concerns the geometrical integration of differential equations on the tangent bundle $T\mathrm{SE}(n)$.

D. Implementation of the extended Hamiltonian system over the Grassmann manifold

A Grassmann manifold $\mathrm{Gr}(n,p)$ is a set of subspaces of $\mathbb{R}^{n\times n}$ spanned by p independent vectors. A representation of any of such subspace may be assumed as the equivalence class $[x] = \{x\rho | x \in \mathrm{St}(n,p),\ \rho \in \mathbb{R}^{p\times p},\ \rho^T\rho = e_p\}$. In practice, an element $[x]$ of the Grassmann manifold $\mathrm{Gr}(n,p)$ is represented by a matrix in $\mathrm{St}(n,p)$ whose columns span the space $[x]$. In the case of learning on the Grassmann manifold $\mathrm{Gr}(n,p)$, closed-form expressions for the geodesic curves as well as of the parallel translation operator may be taken advantage of, hence an implementation of Type II may be summoned. The geometrical expressions of use in the present section associated with the Grassmann manifold endowed with the canonical metric, read [21]:

$$c_{[x],v}(t) = [x\beta\ \alpha] \left[\begin{array}{c} \cos(\sigma t) \\ \sin(\sigma t) \end{array}\right] \beta^T, \tag{53}$$

$$\nabla_x V = (e_n - xx^T)\partial_x V, \tag{54}$$

where $\alpha\sigma\beta^T$ denotes the compact singular value decomposition of the matrix v. The parallel translation of a vector $u \in T_{[x]}\mathrm{Gr}(n,p)$ along the geodesic $c_{[x],v}(t)$ with $v \in T_{[x]}\mathrm{Gr}(n,p)$ is given by [10]:

$$\tau_{c_{[x],v}(t)}(u) = \left([x\beta\ \alpha] \left[\begin{array}{c} -\sin(\sigma t) \\ \cos(\sigma t) \end{array}\right] \alpha^T + e_n - \alpha\alpha^T\right) u. \tag{55}$$

According to the expression of the geodesic over the Grassmann manifold associated to the canonical metric and of the parallel-translation operator (55), the extended Hamiltonian system may be implemented by a Type-II solution as:

$$\left\{\begin{array}{l} (\alpha_k, \sigma_k, \beta_k) \stackrel{\mathrm{def}}{=} \mathrm{svd}(v_k, 0), \\ x_{k+1} = [x_k\beta_k\ \alpha_k] \left[\begin{array}{c} \cos(\eta\sigma_k) \\ \sin(\eta\sigma_k) \end{array}\right] \beta_k^T, \\ a_k \stackrel{\mathrm{def}}{=} (1-\eta\mu)v_k - \eta(e_n - x_k x_k^T)\partial_{x_k} V, \\ v_{k+1} = \left([x_k\beta_k\ \alpha_k] \left[\begin{array}{c} -\sin(\eta\sigma_k) \\ \cos(\eta\sigma_k) \end{array}\right] \alpha_k^T + e_n - \alpha_k\alpha_k^T\right) a_k, \end{array}\right. \tag{56}$$

where $\eta > 0$ is a stepsize for the extended Hamiltonian learning system, $\mathrm{svd}(\cdot, 0)$ denotes the compact singular value decomposition operator and $k = 0, 1, 2, \ldots$. The learning state is represented by the pair $([x_k], v_k) \in T\mathrm{Gr}(n,p)$ for $k \in \mathbb{N}$. An alternative retraction for the Grassmann manifold to the geodesic-based one is discussed in [34] and is based on the QR decomposition instead of the SVD decomposition.

E. Implementation of the extended Hamiltonian system over a flag manifold

A flag in a finite dimensional vector space $\Psi \subset \mathbb{R}^n$ is a sequence of subspaces Φ_i, where each subspace Φ_i is a proper subspace of the next, namely $\{0\} \subset \Phi_1 \subset \Phi_2 \subset \ldots \subset \Phi_r = \Psi$. Setting $d_i \stackrel{\mathrm{def}}{=} \dim \Phi_i$, then it holds that $0 = d_0 < d_1 < d_2 < \cdots d_r = p$, where $p \leq n$ is the dimension of the space Ψ. Any flag may be split into a direct sum of vector spaces $\Psi_i \subset \mathbb{R}^n$ of dimensions $n_i \stackrel{\mathrm{def}}{=} \dim \Psi_i > 0$ such that $n_i = d_i - d_{i-1}$, for $i = 1, \ldots, r$, namely $\Psi \stackrel{\mathrm{def}}{=} \Psi_1 \oplus \Psi_2 \oplus \Psi_3 \oplus \ldots \oplus \Psi_r$. The dimension of the space Ψ satisfies $p = \sum_{i=1}^r n_i \leq n$. The set of all such vector spaces is termed flag manifold[2] and is denoted by $\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$. Points on the flag manifold $[x] \in \mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ are represented by matrices $x \in \mathrm{St}(n,p)$ that obey the following rule: all the matrices $x\mathrm{diag}(\rho_1, \rho_2, \ldots, \rho_r)$ with $\rho_i \in \mathbb{R}^{n_i \times n_i}$, $\rho_i^T\rho_i = e_{n_i}$, $i = 1,\ 2,\ \ldots,\ r$, represent the same point on the flag manifold as the matrix x. It is convenient to partition the matrix representing a given point $[x] \in \mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ as $x = [x_{(1)}\ x_{(2)}\ \ldots\ x_{(r)}]$, $x_{(i)} \in \mathbb{R}^{n\times n_i}$, $i = 1,\ 2,\ \ldots,\ r$. It is worth remarking the following properties:

- If $n_1 = n_2 = \ldots = n_r = 1$, then the flag manifold $\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ reduces to the Stiefel manifold $\mathrm{St}(n,r)$.
- If $r = 1$, then the flag manifold $\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ reduces to the Grassmann manifold $\mathrm{Gr}(n, n_1)$.

The geometrical expressions of interest in the present section associated with the flag manifold endowed with the geometrical structure devised in [39] read:

$$c_{[x],v}(t) = \exp\left(t(\tilde{v}x^T - x\tilde{v}^T)\right) x,\ \tilde{v} \stackrel{\mathrm{def}}{=} \left(e_n - \frac{1}{2}xx^T\right) v, \tag{57}$$

$$(\nabla_x V)_{(i)} = \left(e_n - x_{(i)}x_{(i)}^T\right)\partial_{x_{(i)}} V - \sum_{j\neq i} x_{(j)}\partial_{x_{(j)}}^T V x_{(i)}. \tag{58}$$

The above expressions are associated to the canonical metric of the Stiefel manifold which is assumed as the metric of choice for the flag manifold [39]. In order to implement the extended Hamiltonian equations for the generalized flag manifold by the scheme retraction/vector-translation, it is useful to write an expression for the projector $\pi_{[y]} : \mathbb{R}^{n\times p} \to T_{[y]}\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$, for $[y] \in \mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$. Such a projector may be expressed, block-wise, as:

$$\pi_{[y]}(u)_{(i)} = \left(e_n - y_{(i)}y_{(i)}^T\right) u_{(i)} - \sum_{j\neq i} y_{(j)}u_{(i)}^T y_{(i)}. \tag{59}$$

The dynamical system associated to the flag manifold $\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ may be numerically implemented via the Type-III solution:

$$\left\{\begin{array}{l} \tilde{v}_k \stackrel{\mathrm{def}}{=} \left(e_n - \frac{1}{2}x_k x_k^T\right) v_k, \\ x_{k+1} = \exp\left(\eta(\tilde{v}_k x_k^T - x_k\tilde{v}_k^T)\right) x_k, \\ a_k \stackrel{\mathrm{def}}{=} (1-\eta\mu)v_k - \eta(e_n - x_k x_k^T)\partial_{x_k} V, \\ v_{k+1} = \pi_{[x_{k+1}]}(a_k), \end{array}\right. \tag{60}$$

[2]The definition given here amends the incorrect definition given in the paper [21].

where $\eta > 0$ is a stepsize for the extended Hamiltonian learning system and $k = 0, 1, 2, \ldots$. The learning state is represented by the pair $([x_k], v_k) \in T\mathrm{Fl}(n, n_1, n_2, \ldots, n_r)$ for $k \in \mathbb{N}$.

F. Implementation of the extended Hamiltonian system over the manifold of symmetric positive-definite matrices

The manifold of symmetric positive definite matrices is defined as $\mathrm{S}^+(n) \stackrel{\mathrm{def}}{=} \{x \in \mathbb{R}^{n\times n} | x^T - x = 0,\ x > 0\}$. The manifold $\mathrm{S}^+(n)$ of symmetric positive-definite matrices, endowed with its canonical metric, has associated the following geometric quantities:

$$\Gamma_x(v, v) = -vx^{-1}v, \tag{61}$$

$$c_{x,v}(t) = x^{\frac{1}{2}} \exp(tx^{-\frac{1}{2}} v x^{-\frac{1}{2}}) x^{\frac{1}{2}}, \tag{62}$$

$$\nabla_x V = \frac{1}{2} x \left(\partial_x V + \partial_x^T V \right) x. \tag{63}$$

The geodesic distance between two give symmetric positive-definite matrices may be calculated according to the definition (4). The distance associated to the canonical metric of the manifold of symmetric positive-definite matrices is given by $d(x, y) = \| \log(x^{-1}y) \|_{\mathrm{F}}$, for $x, y \in \mathrm{S}^+(n)$ (see, e.g., [36]). In the above expression, symbol $\| \cdot \|_{\mathrm{F}}$ denotes Frobenius norm.

Recall that $T_x\mathrm{S}^+(n) = \mathrm{S}(n)$, namely, the set of $n \times n$ symmetric matrices. Hence, all tangent spaces coincide with each other. Such observation leads to an implementation of Type I. By gathering the expressions of the Christoffel operator and of the Riemannian gradient of the potential energy function, the following extended Hamiltonian learning system equations are obtained:

$$\left\{ \begin{array}{rcl} \dot{x} &=& v, \\ \dot{v} &=& vx^{-1}v - \frac{1}{2} x \left(\partial_x^T V - \partial_x V \right) x - \mu v. \end{array} \right. \tag{64}$$

The kinetic energy function has expression $K = \frac{1}{2}\mathrm{tr}((vx^{-1})^2)$.

The second equation of the system (64) describes a dynamics over the vector space of symmetric matrices and may be integrated numerically by a Euler stepping method, while the first one may be integrated via a retraction of the manifold of symmetric positive definite matrices. Namely, the system (64) may be implemented by a Type-I solution:

$$\left\{ \begin{array}{rcl} x_{k+1} &=& R_{x_k}(\eta v_k), \\ v_{k+1} &=& \eta \left[v_k x_k^{-1} v_k - \frac{1}{2} x_k (\partial_{x_k}^T V - \partial_{x_k} V) x_k \right] + \\ & & (1 - \eta\mu) v_k, \end{array} \right. \tag{65}$$

where $\eta > 0$ plays the role of learning step-size for the extended Hamiltonian learning system and $k = 0, 1, 2, \ldots$. The learning state is represented by $x_k \in \mathrm{S}^+(n)$ and $v_k \in \mathrm{S}(n)$ for $k \in \mathbb{N}$.

A suitable retraction map of the manifold of symmetric positive definite matrices is the exponential map associated to the geodesic, namely $R_x(v) = x^{\frac{1}{2}} \exp\left(x^{-\frac{1}{2}} v x^{-\frac{1}{2}} \right) x^{\frac{1}{2}}$.

A related problem arises about the manifold $\mathrm{S}^+(n, p)$ of symmetric fixed-rank positive-semidefinite matrices, that may be defined as:

$$\mathrm{S}^+(n, p) \stackrel{\mathrm{def}}{=} \{ U R^2 U^T | U \in \mathrm{St}(n, p),\ R^2 \in \mathrm{S}^+(p),\ p < n \}. \tag{66}$$

The interest in symmetric fixed-rank positive-semidefinite matrices stems from the observation that, with the growing use of low-rank approximations of matrices as a way to retain tractability in large-scale applications, there is a need to extend the calculus of positive definite matrices to their low-rank counterparts [4].

G. Implementation of the extended Hamiltonian system over the group of real symplectic matrices

The manifold of real symplectic matrices is defined as follows:

$$\mathrm{Sp}(2n, \mathbb{R}) = \{ x \in \mathbb{R}^{2n\times 2n} | x^T q x = q \},\ q = \left[\begin{array}{cc} 0_n & e_n \\ -e_n & 0_n \end{array} \right]. \tag{67}$$

The manifold $\mathrm{Sp}(2n, \mathbb{R})$ is regarded as a Lie group with Lie algebra:

$$\mathfrak{sp}(2n, \mathbb{R}) = \{ h \in \mathbb{R}^{2n\times 2n} | h^T q + q h = 0_{2n} \}. \tag{68}$$

For simplicity, the dimension n is not indicated in the notation of the fundamental skew-symmetric matrix q. The real symplectic group is supposed to be endowed with the Bloch-Crouch-Marsden-Sayal metric as explained in [21]. Upon such a choice, the real symplectic group has associated the following geometric quantities [21]:

$$\Gamma_x(v, v) = -vx^{-1}v + xv^T q x q x^{-1} v - v v^T q x q, \tag{69}$$

$$\nabla_x V = \frac{1}{2} x q \left(\partial_x^T V x q - q x^T \partial_x V \right). \tag{70}$$

The complete extended Hamiltonian system describing a learning dynamics over the real symplectic group may now be obtained by inserting the expressions of the Christoffel matrix function Γ_x and the expression of the Riemannian gradient $\nabla_x V$ of the potential energy function into the general extended Hamiltonian system (14). Calculations lead to the following expressions:

$$\left\{ \begin{array}{rcl} \dot{x} &=& xh, \\ \dot{h} &=& h^T h - h h^T - \frac{1}{2} q \left((\partial_x V)^T x q - q x^T \partial_x V \right) - \mu h. \end{array} \right. \tag{71}$$

The kinetic energy function has expression $K = \frac{1}{2}\mathrm{tr}(h^T h)$.

Likewise the case of the orthogonal group $\mathrm{SO}(n)$, the second equation describes a dynamics over the Lie algebra $\mathfrak{sp}(2n, \mathbb{R})$. Hence, the second equation may be integrated numerically by a Euler stepping method, while the first one may be integrated via a suitable retraction of the symplectic group. Namely, the system (71) may be implemented by a Type-I solution as:

$$\left\{ \begin{array}{rcl} x_{k+1} &=& R_{x_k}(\eta x_k h_k), \\ h_{k+1} &=& \eta \left(h_k^T h_k - h_k h_k^T \right) + (1 - \eta\mu) h_k - \\ & & \frac{1}{2} \eta q \left(\partial_{x_k}^T V x_k q - q x_k^T \partial_{x_k} V \right), \end{array} \right. \tag{72}$$

where the parameter $\eta > 0$ again plays the role of learning stepsize for the extended Hamiltonian learning system and $k = 0, 1, 2, \ldots$. The learning state is represented by the couple $x_k \in \mathrm{Sp}(2n, \mathbb{R})$ and $h_k \in \mathfrak{sp}(2n, \mathbb{R})$ for $k \in \mathbb{N}$. Possible retraction of the real symplectic group are $R_x(v) = x \exp(x^{-1}v)$, whose properties were studied in the contributions [20], [27], and the Cayley map, as explained in [1].

IV. Numerical experiments

The present section aims at illustrating the numerical behavior of the extended Hamiltonian learning paradigm. In the present manuscript, no particular attention is paid to the question of the computational complexity of the learning algorithms explained in section III nor to the optimization of matrix-type expressions to attain minimal redundancy. The implementation of the learning algorithms of section III requires the repeated computation of QR and SVD factorizations as well as of matrix exponentiation, which makes their complexity of order $O(p^3)$ whenever matrices of size $p \times p$ are involved. The numerical issues behind QR and SVD factorizations are well known [25]. Advancements on the exponentiation of structured matrices may be read of in the papers [6], [46], [48] and references therein. The present section discusses the following cases: 1) Experiments of learning over the manifold $M = \mathbb{R}^2$ that aim at illustrating numerically the behavior of the extended Hamiltonian system when the damping condition (24) is/is-not verified, in comparison to the behavior of the gradient learning system. 2) An experiments of learning over the manifold $M = \mathbb{R}$ that aims at illustrating the benefit of the initial momentum in escaping from the plateaus of the potential energy function landscape. 3) A discussion of learning over the manifold $M = \mathrm{SO}(2)$ that aims at illustrating the need of appropriate geometrical integration of the extended Hamiltonian

learning equations. 4) Experiments of learning over the manifold $M = \mathrm{S}^{n-1}$ on applications such as minor component analysis and blind channel deconvolution. 5) Experiments of learning over the manifold $M = \mathrm{S}^{+}(n)$ on positive-definite symmetric matrix averaging. In the following subsections, the parameter t is interpreted as a time parameter ranging in the interval $[0,\ 1]$ and is measured in seconds. All the experiments were coded in MATLAB.

A. Experiments of learning on the space $\mathbb{R}^2$

In order to illustrate numerically the theoretical developments about the comparison between a dynamical system and a gradient-based system recalled in subsection II-E, the present subsection deals with a learning problem in the manifold $M = \mathbb{R}^2$ which affords an easy graphical rendering of the behavior of such methods. The potential energy function is $V(x) = \frac{1}{2}x^T Ax$, with $A \in \mathrm{S}^{+}(2)$ and the sought-for optimal connection pattern $x_\star$ in the present unconstrained case coincides with the origin of the plane $\mathbb{R}^2$. It is of particular interest here to illustrate numerically the behavior of the dynamical system when the condition (24) is not verified and when it is verified. In the present case, the eigenvalue $\lambda_{\max}$ coincides with the maximum eigenvalue of the matrix A. The Figure 3 corresponds to the case that $\mu = 0.2\sqrt{2\lambda_{\max}}$. Such a choice of the damping parameter corresponds to an under-damped dynamical system that exhibits a slowly-converging oscillating behavior. The

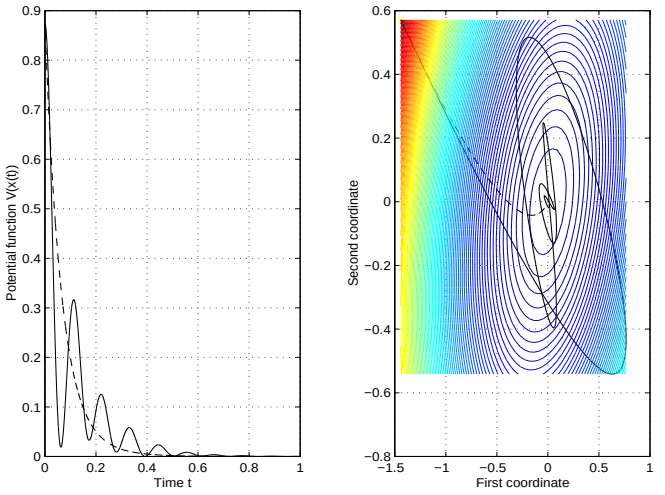

Fig. 3. Experiment of learning on the plane $\mathbb{R}^2$: Under-damped dynamical system. Left-hand panel: Course of the criterion function $V(x(t))$. Right-hand panel: Learning seen over the parameter-manifold $\mathbb{R}^2$. (Solid line: Extended Hamiltonian learning algorithm. Dashed line: Gradient algorithm.)

Figure 4, conversely, corresponds to the case that $\mu = 1.2\sqrt{2\lambda_{\max}}$. Such a choice of parameters meets condition (24) and corresponds to a damped dynamical system that exhibits a non-oscillating behavior and that converges faster than the gradient-based learning system (both with the same learning stepsize $\eta = 0.001$).

B. Experiments of learning on the space $\mathbb{R}$

In order to illustrate the benefit of the initial momentum in escaping from the plateaus of the potential energy function landscape, an experiment of learning over the manifold $M = \mathbb{R}$ is discussed. The potential energy function used in the present numerical test is illustrated in the Figure 5 in the top-right panel. The results reported in the Figure 5 corresponds to the case that $x_1 = \frac{98}{100}x_0$. The initial point x_0 corresponds to a flat zone of the potential energy function and the shown numerical results confirm numerically that the gradient

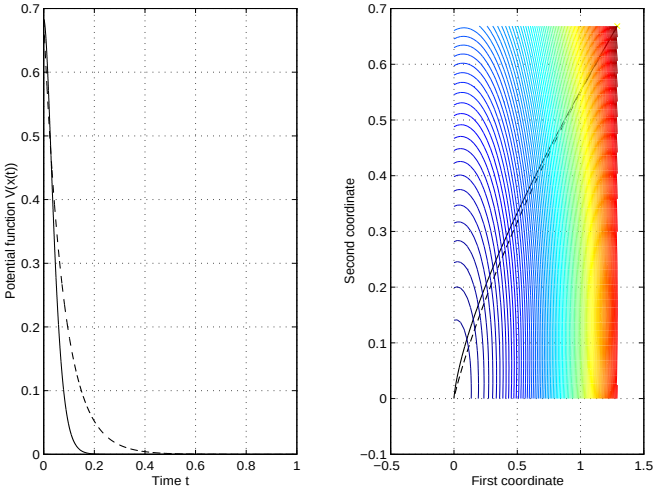

Fig. 4. Experiment of learning on the plane $\mathbb{R}^2$: Damped dynamical system that meets condition (24). Left-hand panel: Course of the criterion function $V(x(t))$. Right-hand panel: Learning seen over the parameter-manifold $\mathbb{R}^2$. (Solid line: Extended Hamiltonian learning algorithm. Dashed line: Gradient algorithm.)

method is unable to escape the plateau while the dynamical system is able to move out of the plateau thanks to the initial kinetic energy corresponding to the impressed initial momentum.

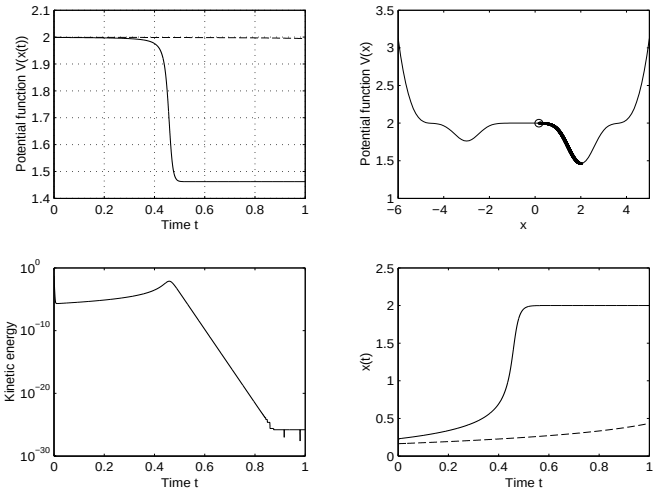

Fig. 5. Experiment of learning on the line $\mathbb{R}$: Effect of the initial momentum in escaping from the plateaus of the potential energy function landscape. Top-left panel: Course of the criterion function $V(x(t))$. Top-right panel: Shape of the potential function $V(x)$ and learning course represented by a dotted curve (while the starting point x_0 is denoted by an open circle). Bottom-left panel: Kinetic energy. Bottom-right panel: Course of $x(t)$. (Solid line: Extended Hamiltonian learning algorithm. Dashed line: Gradient algorithm.)

C. About learning on the space $\mathrm{SO}(2)$

To render the problem that arises about the numerical implementation of the dynamical learning equations on manifolds, it is worth revisiting the explanatory example discussed in the concluding section of paper [21] about the low-dimensional manifold $\mathrm{SO}(2)$. By embedding the space $\mathrm{SO}(2)$ into the space $\mathbb{R}^{2\times 2}$, any element of $\mathrm{SO}(2)$ may be regarded as a 2-by-2 real-valued matrix $x = \left[\begin{smallmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{smallmatrix}\right]$ whose entries must satisfy the constrains: $x_{11}^2 + x_{21}^2 = 1$, $x_{22}^2 + x_{12}^2 = 1$, $x_{11}x_{12} + x_{21}x_{22} = 0$ and $x_{11}x_{22} - x_{12}x_{21} = 1$.

The extended Hamiltonian system learning equation for the variable $x \in \mathrm{SO}(2)$ in (50) is a special case of a general differential equation $\dot{x} = F(x)$ on the manifold $\mathrm{SO}(2)$, with $F : x \in \mathrm{SO}(2) \mapsto F(x) \in T_x\mathrm{SO}(2)$, namely, it may be broken down into a set of four differential equations of the type $\dot{x}_{ij}(t) = F_{ij}(x_{11}(t), x_{12}(t), x_{21}(t), x_{22}(t))$. In the present case, it holds that $F(x) = x\omega$ with $\omega = \left[\begin{smallmatrix} 0 & \Omega \\ -\Omega & 0 \end{smallmatrix}\right]$, where $\Omega = \Omega(x(t)) \in \mathbb{R}$. The Euler stepping technique of numerical calculus to solve the above system of differential equations would read $x_{k+1} = x_k + \eta F(x_k)$, with $\eta > 0$ denoting a learning stepsize. As explained in the conclusions of paper [21], such numerical stepping method does not take into account the constraints on the entries of matrix x, namely, it generates a trajectory $k \mapsto x_k$ in the ambient space $\mathbb{R}^{2\times 2}$ rather than in the feasible space $\mathrm{SO}(2)$. Namely, starting from a point $x_k \in \mathrm{SO}(2)$, it would produce a new point $x_{k+1} \notin \mathrm{SO}(2)$. The reason of such behavior is that the Euler numerical integration techniques insist on the flat space $\mathbb{R}^n$ and do not cope with curved manifolds. An appropriate numerical stepping method is the one reported in equations (51) based on the notion of manifold retraction.

It is instructive to investigate in detail on the effect of the Euler stepping method in the solution of the differential equation $\dot{x} = x\omega$. In such a context, the Euler stepping equation reads:

$$x_{k+1} = x_k(e_2 + \eta\omega_k), \text{ with } \omega_k = \begin{bmatrix} 0 & \Omega_k \\ -\Omega_k & 0 \end{bmatrix}. \tag{73}$$

As the starting point x_0 satisfies $x_0^T x_0 = e_2$, it holds:

$$x_1^T x_1 = (e_2 + \eta\omega_0)^T x_0^T x_0 (e_2 + \eta\omega_0) = e_2 - \eta^2\omega_0^2. \tag{74}$$

Note that $\omega_k^2 = -\Omega_k^2 e_2$, hence $x_1^T x_1 = (1 + \eta^2\Omega_0^2)e_2$. The first step x_1 looses the normality of the columns of an additive amount $\eta^2\Omega_0^2$ and changes its determinant from 1 to $\sqrt{1 + \eta^2\Omega_0^2}$. However, it keeps the orthogonality of the columns of the matrix x (such phenomenon is peculiar of the case $\mathrm{SO}(2)$ only and does not copy to the general case $\mathrm{SO}(n)$ with $n > 2$). For the next step, it holds that:

$$\begin{aligned} x_2^T x_2 &= (e_2 + \eta\omega_1)^T x_1^T x_1 (e_2 + \eta\omega_1) \\ &= (1 + \eta^2\Omega_0^2)(e_2 - \eta^2\omega_1^2) \\ &= (1 + \eta^2\Omega_0^2)(1 + \eta^2\Omega_1^2)e_2. \end{aligned} \tag{75}$$

By reasoning by induction, it is readily verified that the matrix x_k keeps monotonically loosing the normality of its two columns of an identical amount, while it retains the mutual orthogonality of the columns. On the other hand, the geometric integration paradigm applied to the present case, with geodesic-based retraction, reads:

$$x_{k+1} = x_k \exp(\eta\omega_k), \text{ with } \omega_k = \begin{bmatrix} 0 & \Omega_k \\ -\Omega_k & 0 \end{bmatrix}. \tag{76}$$

Note that $\exp(\eta\omega_k) = \left[\begin{smallmatrix} \cos(\eta\Omega_k) & \sin(\eta\Omega_k) \\ -\sin(\eta\Omega_k) & \cos(\eta\Omega_k) \end{smallmatrix}\right]$. It is straightforward to verify that the rule (76) implies that $x_{k+1}^T x_{k+1} = x_k^T x_k = e_2$.

D. Experiments of learning on the unit hypersphere

The first experiment of learning on the unit hypersphere concerns minor component analysis (MCA). Minor component analysis is a statistical data analysis technique that aims at determining the eigenvector of a given symmetric positive-definite matrix corresponding to its smallest eigenvalue. Minor component analysis has several applications in machine learning and signal processing (see, e.g., [53]). The potential energy function associated to minor component analysis is:

$$V(x) = \frac{1}{2} x^T A x, \tag{77}$$

with $A \in \mathrm{S}^+(n)$ being a covariance matrix whose minor eigenpair is sought for. In the present experiment $n = 10$, $\eta = 0.1$ and $\mu = 1.01\sqrt{2\lambda_{\max}}$, with $\lambda_{\max}$ being the maximal eigenvalue of the matrix A and a Type-III implementation was selected. Any stationary point $x_\star \in \mathrm{S}^{n-1}$ of the extended Hamiltonian learning system satisfies $\nabla_{x_\star} V = 0$, namely $(e_n - x_\star x_\star^T)Ax_\star = 0$. The latter condition may be written equivalently as $Ax_\star = 2V(x_\star)x_\star$, which confirms that the extended Hamiltonian system (14) over the manifold S^{n-1} with the potential energy function (77) evolves toward the eigenvector $x_\star \in \mathrm{S}^{n-1}$ of the covariance matrix A corresponding to the minimal eigenvalue $2V(x_\star)$. The Figure 6 shows learning curves of the extended Hamiltonian algorithm compared to the retracted Riemannian gradient algorithm, averaged over 100 independent trials. In particular, the top panel shows the value of the criterion function $V(x(t))$ as well as the value of the minimal criterion function $V(x_\star)$. The middle panel shows the kinetic energy $K_{x(t)}(\dot{x}(t), \dot{x}(t))$ while the bottom panel shows the value of the consistency index $C(t) \stackrel{\text{def}}{=} x^T(t)v(t)$. The consistency index measures the numerical precision of parallel translation and should take values as close as possible to zero. The results show that the extended Hamiltonian learning algorithm is effective and converges much faster than the corresponding retracted gradient-based learning algorithm endowed with the same learning stepsize value η.

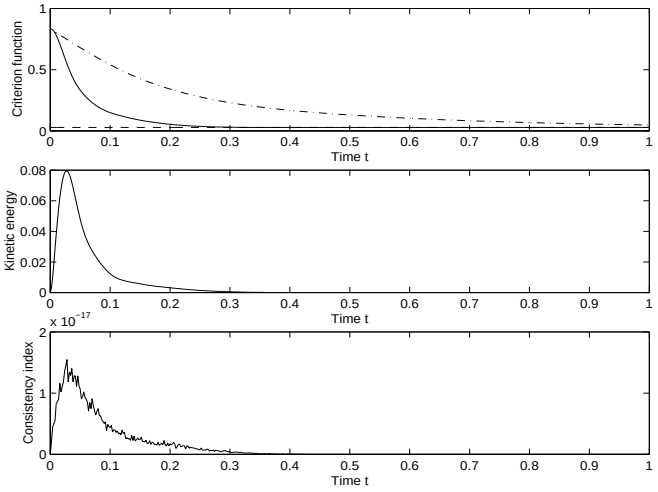

Fig. 6. Experiment of learning on the unit hypersphere: Minor component analysis. Top panel: Criterion function $V(x(t))$. Middle panel: Kinetic energy $K_{x(t)}(\dot{x}(t), \dot{x}(t))$. Bottom panel: Consistency index $C(t)$. (Solid line: Extended Hamiltonian learning algorithm. Dot-dashed line: Retracted Riemannian gradient algorithm. Dashed-line: Target learning criterion value (minimal potential energy function).) Curves averaged over 100 independent trials on the same learning problem.

The second experiment of learning on the unit hypersphere concerns blind channel deconvolution. Blind deconvolution is a signal processing technique that aims at recovering a signal distorted by a transmission system. Deconvolution arises naturally when dealing with finite multi-path interference on a signal, such as in marine seismic exploration [41] and in speech dereverberation [8]. Recent notable applications are to bar code reconstruction [11], image integrity verification in forensics [50], high-resolution mapping of transcription factor binding sites in DNA chains [33] and to the thermodynamic characterization of thin films [13]. In particular, the 'Bussgang' family of blind deconvolution algorithm was introduced in [2]. For a detailed explanation of the problem, the blind deconvolution theory and of the experiments see, e.g., [14], [16]. The blind deconvolution ability of an algorithm is measured in terms of the Inter-Symbol Interference (ISI) figure that should take values as close as possible to zero. In the present experiment $n = 14$, $\eta = 0.5$ and $\mu = 1$ as the result of validation and a Type-III implementation was selected. The

Figure 7 shows a detail of the behavior of the extended Hamiltonian learning algorithm on a noisy channel with signal-to-noise ratio ranging in $\{1, 5, 10, 20, \infty\}$ dB. The Figure 8 shows the ISI figure as well as the runtime figures of the Extended Bussgang algorithm, the Bussgang algorithm, the Bussgang algorithms with natural gradient, the Extended Bussgang algorithm with natural gradient and the Extended Hamiltonian learning algorithm. These figures confirm that the extended-Hamiltonian-learning algorithm applied to a real-world blind channel deconvolution problem converges steadily and that its learning performances are comparable to those of existing algorithm at a lower computational cost: In fact, the extended-Hamiltonian-learning algorithm applied to a blind channel deconvolution problem exhibits the lowest ISI index and is the fastest to run.

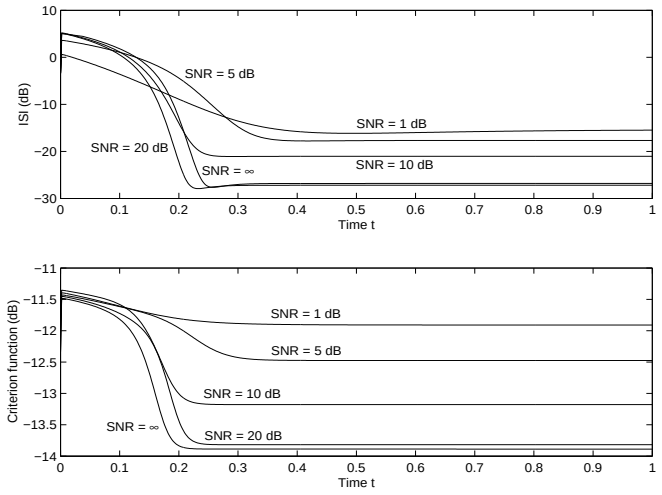

Fig. 7. Experiment of learning on the unit hypersphere: Blind channel deconvolution. Top panel: ISI figure during learning corresponding to signal-to-noise ratios (SNR) of $\{1, 5, 10, 20, \infty\}$ dB. Bottom panel: Criterion function $V(x(t))$.

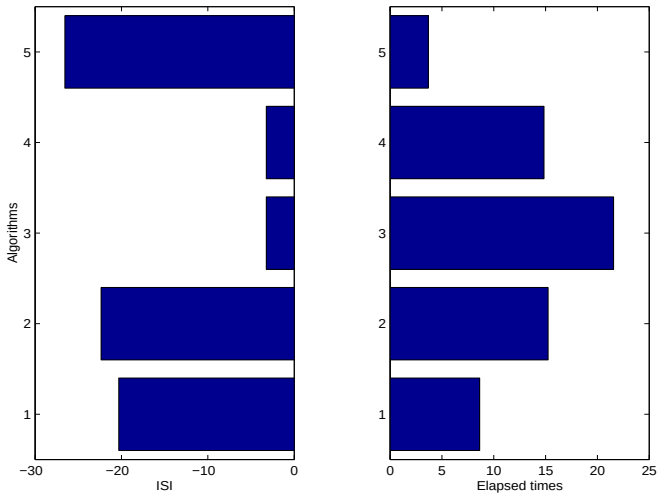

Fig. 8. Experiment of learning on the unit hypersphere: Blind channel deconvolution. Left-hand panel: ISI figures. Right-hand panel: Runtime figures (in seconds). (1 – Extended Bussgang algorithm, 2 – Bussgang algorithm, 3 – Bussgang algorithms with natural gradient, 4 – Extended Bussgang algorithm with natural gradient, 5 – Extended Hamiltonian learning algorithm.)

E. Experiments of learning on the manifold of symmetric positive-definite matrices

Symmetric positive-definite matrices play an important role in machine learning and applications (see, e.g., [18], [36]). Notable applications are to medical imagery analysis [23], data clustering and template matching [43] as well as automatic and intelligent control [5]. The ensemble statistical features of data and the algorithms to estimate them are of prime importance in intelligent data processing. In particular, computing the mean value of a set of data is a widely used technique to smooth out irregularities in the data and to filter out the noise and the measurement errors [22]. Learning the average matrix out of a set of N matrices $s_k \in \mathrm{S}^+(n)$ may be achieved by seeking the matrix $x \in \mathrm{S}^+(n)$ that minimizes the spread function

$$V(x) = \frac{1}{N} \sum_{k=1}^{N} d^2(x, s_k), \tag{78}$$

where $d : \mathrm{S}^+(n) \times \mathrm{S}^+(n) \to \mathbb{R}$ denotes a distance function (see subsection III-F). The function (78) measures the variance of the samples s_k around the point x and the empirical mean value of the distribution is defined as the point that minimizes such variance. In the present experiment $N = 50$, $n = 5$, $\eta = 0.1$ and $\mu = 2$ and a Type-I implementation was selected. The Figure 9 shows learning curves of the extended Hamiltonian algorithm compared to the retracted Riemannian gradient algorithm.

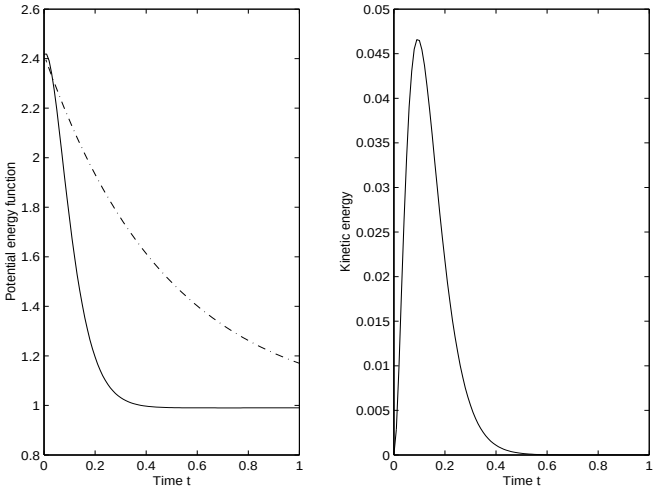

Fig. 9. Experiment of learning on the manifold of symmetric positive-definite matrices: Averaging. Left-hand panel: Criterion function $V(x(t))$. Right-hand panel: Kinetic energy $K_{x(t)}(\dot{x}(t), \dot{x}(t))$. (Solid line: Extended Hamiltonian learning algorithm. Dot-dashed line: Retracted Riemannian gradient algorithm.)

V. Conclusion

The present paper aims at discussing the numerical implementation of the extended Hamiltonian learning paradigm introduced in the contribution [21]. The main idea is to make use of two different notions to integrate the differential equations of the system (14). In particular, the differential equation in the state-variable $x \in M$ is integrated numerically by the help of retraction maps, while the differential equation in the velocity-variable $v \in T_x M$ is integrated numerically by the help of parallel-translation-based or vector-translation-based step-forward methods. The general-purpose discussion of section II as well as the discussion of the cases-of-interest of section III show that the implementation of the dynamical learning equations requires

a large effort in terms of mathematical analysis and exhibits a rich structure in terms of implementation possibilities.

The contribution [21] and the present contribution aim together at illustrating the author's current knowledge about the formulation of a learning theory based on the extended Hamiltonian stationary-action principle and to discuss its implementation on a computation platform. Such contributions open new perspectives in the theory of extended Hamiltonian learning that are briefly discussed below.

Keeping fixed the discretization parameter η, hence with the same numerical precision, the dynamical learning algorithm may converge faster than the gradient-steepest-descent algorithm. The theory also provides a hint about how to choose the learning parameters on the basis of a fine-convergence analysis. It is to be noted that the same learning stepsize η is used within the expression of the numerical integration rule of the first dynamical equation via retraction and within the expression of the numerical integration rule of the second dynamical equation via parallel/vector translation. In principle, two discretization stepsizes might be used instead. Moreover, it is known that the appropriate manipulation of the learning stepsize parameter during the training process can improve the learning ability of a learning machine [37]. A special emphasis is to be put on those stepsize-adaptation methods that are parameter-independent, that is, that do not imply the need for the user to tune parameters whose values exert influence on the performance of the algorithm.

Considerable research into the training of neural networks by gradient-based learning rules has been undertaken in the past years and it has been known that the introduction of a momentum term into the training equation can accelerate the training process considerably [42], [45]. However, it is still not clear how to choose the initial momentum which corresponds, in the context of extended Hamiltonian learning, to the initial speed v_0.

Much of the discussion in the physics and engineering literature concerning damped systems focuses on systems subject to viscous damping $-\mu v$ even though viscous damping occurs rarely in real physical systems. Other types of dissipative forces, such as the Rayleigh drag, exist in real systems. Frictional or drag forces that describe the motion of an object through a fluid or gas are rather complicated. One of the simplest empirical mathematical model is taken to be of the form $-\mu\|v\|^{\epsilon-1}v$, where ϵ denotes a damping exponent [47]. Such a model might replace the linear damping term in the system (14) to explore richer damping phenomena and need careful theoretical investigation in order to understand how to embed a non-linear damping term in the equations of dynamics without violating any underlying geometrical constraint.

The dynamical system (14) described by the state-pair $(x, v) \in TM$ evolves toward a stationarity state $(x_\star, 0)$, where the point $x_\star \in M$ coincides with a local minima of the potential energy function V over the manifold M. Such observation may be exploited in neural learning where the potential energy function is set to a learning goal: The optimal parameters of a machine are those that minimize the goal function. Speculations were raised by the possibility of choosing the potential energy function V from physics and to understand what will be the behavior of the extended hamiltonian learning system under any such potential field. As an instance, the contribution [12] discusses a particle-interaction-type potential function, termed *information potential*, in the context of machine learning. Also, the contribution [28] shows that electrostatic-type potential energy functions gives new perspectives to support vector machines. moreover, recently, attention has been paid to the study of non-smooth Hamiltonian systems [31]: Such analysis would be worth extending to the case of Hamiltonian systems whose state-variables evolve on manifolds.

To end with, it is worth mentioning an open question of seemingly pure speculative interest yet. The differential system on the tangent bundle TM (1) describes the state of a system that evolve until it attains a local minimum of the potential energy function V. If $M = \mathbb{R}$ and a Euclidean metric is selected, then the system (1) collapses to a free damped oscillator such as, for instance, the mass-spring-damper system:

$$\mathcal{M}\ddot{x} + \kappa x + \mu\dot{x} = 0, \tag{79}$$

where $\kappa > 0$ denotes the spring's constant of elasticity. What makes the oscillator free is that the right-hand side of the above equation equals zero. By replacing the right-hand side by a function of time, for example $F\cos(\omega t)$, with $F, \omega \in \mathbb{R}^+$, and the linear term κx with a non-linear term, one obtains a nonlinear oscillator [47]. Nonlinear oscillator theory has several applications and speculation arose about the possibility of extending the theory of non-linear oscillators to manifolds by adding a time dependency to the system (1).

VI. Acknowledgments

Part of the present work was written when I was a visiting scientist at the Bioinformatics Institute (BII-A*STAR, Biopolis, Singapore). I wish to gratefully thank Dr. Hwee-Kwan Lee for the opportunity to visit BII and the Institute members for the warm hospitality and for the constructive comments about the present research topic. I wish to gratefully thank the anonymous reviewers and the associate editor for the careful and stimulating comments to the present manuscript.

References

[1] V.I. Arnol'd and A.B. Givental', *Symplectic geometry*, in "Dynamical Systems IV: Symplectic Geometry & Its Applications" (V.I. Arnol'd and S.P. Novikov, Ed.s), Enciclopaedya of Mathematical Sciences, Vol. 4, pp. 1 – 138, Springer-Verlag (Second Edition), 2001

[2] S. Bellini, *Blind equalization*, Alta Frequenza, Vol. 57, pp. 445 – 450, 1988

[3] A. Bielecki, *Estimation of the Euler method error on a Riemannian manifold*, Communications in Numerical Methods in Engineering, Vol. 18. pp. 757 – 763, 2002

[4] S. Bonnabel and R. Sepulchre, *Riemannian metric and geometric mean for positive semidefinite matrices of fixed rank*, SIAM Journal of Matrix Analysis and Applications, Vol. 31, No. 3, pp. 1055 – 1070, 2009

[5] Y. Chen and J.E. McInroy, *Estimation of symmetric positivedefinite matrices from imperfect measurements*, IEEE Transactions on Automatic Control, Vol. 47, No. 10, pp. 1721 – 1725, October 2002

[6] E. Celledoni and S. Fiori, *Neural learning by geometric integration of reduced 'rigid-body' equations*, Journal of Computational and Applied Mathematics (JCAM), Vol. 172, No. 2, pp. 247 – 269, December 2004

[7] J. Cortés, A. Van Der Schaft and P.E. Crouch, "Characterization of gradient control systems," SIAM Journal on Control and Optimization, Vol. 44, No. 4, pp. 1192 – 1214, 2005

[8] M.J. Daly and J.P. Reilly, *Blind deconvolution using bayesian methods with application to the dereverberation of speech*, Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004), Vol. 2, pp. 1009 – 1012, May 2004

[9] N. del Buono and L. Lopez, *Runge-Kutta type methods based on geodesics for systems of ODEs on the Stiefel manifold*, BIT Numerical Mathematics, Vol. 41, No. 5, pp. 912 – 923, 2001

[10] A. Edelman, T.A. Arias and S.T. Smith, *The geometry of algorithms with orthogonality constraints*, SIAM Journal on Matrix Analysis Applications, Vol. 20, No. 2, pp. 303 – 353, 1998

[11] S. Esedoglu, *Blind deconvolution of bar code signals*, Inverse Problems, Vol. 20, pp. 121 – 135, 2004

[12] D. Erdogmus and J.C. Principe, *Generalized information potential criterion for adaptive system training*, IEEE Transactions on Neural Networks, Vol. 13, No. 5, pp. 1035 – 1044, September 2002

[13] E.F.S. Filho, D.B. Haddad and L.A.L. de Almeida, *Thermal Hysteresis Characterization Through Blind Deconvolution*, Proceedings of the 17^{th} International Conference on Systems, Signals and Image Processing (IWSSIP 2010, Rio de Janeiro, Brazil, 17-19 June 2010), pp. 352 – 355, 2010

[14] S. Fiori, *A fast fixed-point neural blind deconvolution algorithm*, IEEE Transactions on Neural Networks, Vol. 15, No. 2, pp. 455 – 459, March 2004

[15] S. Fiori, *Formulation and integration of learning differential equations on the Stiefel manifold*, IEEE Transactions on Neural Networks, Vol. 16, No. 6, pp. 1697 – 1701, November 2005
[16] S. Fiori, *Geodesic-based and projection-based neural blind deconvolution algorithms*, Signal Processing, Vol. 88, No. 3, pp. 521 – 538, March 2008
[17] S. Fiori, *Lie-group-type neural system learning by manifold retractions*, Neural Networks (Elsevier), Vol. 21, No. 10, pp. 1524 – 1529, December 2008
[18] S. Fiori, *Learning the Fréchet mean over the manifold of symmetric positive-definite matrices* (Invited keynote), Cognitive Computation (Springer), Vol. 1, No. 4, pp. 279 – 291, December 2009
[19] S. Fiori, *Learning by natural gradient on noncompact matrix-type pseudo-Riemannian manifolds*, IEEE Transactions on Neural Networks, Vol. 21, No. 5, pp. 841 – 852, May 2010
[20] S. Fiori, *Averaging over the Lie group of optical systems transference matrices*, Frontiers of Electrical and Electronic Engineering in China (Springer), Special issue of the "Sino foreign-interchange Workshop on Intelligence Science and Intelligent Data Engineering" Part A, Vol. 6, No. 1, pp. 137 – 145, March 2011
[21] S. Fiori, *Extended Hamiltonian learning on Riemannian manifolds: Theoretical aspects*, IEEE Transactions on Neural Networks, Vol. 22, No. 5, pp. 687 – 700, May 2011
[22] S. Fiori and T. Tanaka, *An algorithm to compute averages on matrix Lie groups*, IEEE Transactions on Signal Processing, Vol. 57, No. 12, pp. 4734 – 4743, December 2009
[23] P.T. Fletcher and S. Joshi, *Riemannian geometry for the statistical analysis of diffusion tensor data*, Signal Processing, Vol. 87, No. 2, pp. 250 – 262, February 2007
[24] D. Gabay, *Minimizing a differentiable function over a differentiable manifold*, Journal of Optimization Theory and Applications, Vol. 37, No. 2, pp. 177 – 219, 1982
[25] G. Golub and C. van Loan, *Matrix Computations*, The Johns Hopkins University Press, 3rd Edition, 1996
[26] E. Hairer, C. Lubich and G. Wanner, *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*, Springer Series in Computational Mathematics, 2nd Edition, 2006
[27] W.F. Harris and J.R. Cardoso, *The exponential-mean-log-transference as a possible representation of the optical character of an average eye*, Ophthalmic and Physiological Optics, 2006, Vol.26, No. 4, pp. 380 – 383, 2006
[28] S. Hochreiter, M.C. Mozer and K. Obermayer, *Coulomb classifiers: Generalizing support vector machines via an analogy to electrostatic systems*, Advances in Neural Information Processing Systems (NIPS, MIT Press) 15, pp. 545 – 552, 2003
[29] G. Hori and T. Tanaka, *Pivoting in Cayley tranform-based optimization on orthogonal groups*, Proceedings of the Second APSIPA Annual Summit and Conference, pp. 181 – 184, Biopolis, Singapore, 14-17 December 2010
[30] S. Huckemann, T. Hotz and A. Munk, *Intrinsic MANOVA for Riemannian manifolds with an application to Kendall's space of planar shapes*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 4, pp. 593 – 603, April 2010
[31] D.M. Kaufman and D.K. Pai, *Geometric Numerical integration of inequality constrained, nonsmooth Hamiltonian systems*, 2010, arXiv preprint available at `http://arxiv.org/abs/1007.2233`
[32] E. Klassen, A. Srivastava, W. Mio and S.H. Joshi, *Analysis of planar shapes using geodesic paths on shape spaces*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 26, No. 3, pp. 372 – 383, March 2004
[33] D.S. Lun, A. Sherrid, B. Weiner, D.R. Sherman and J.E. Galagan, *A blind deconvolution approach to high-resolution mapping of transcription factor binding sites from ChIP-seq data*, Genome Biology, Vol. 10, No. 12, pp. 142 – 144, December 2009
[34] G. Meyer, S. Bonnabel and R. Sepulchre, *Regression on fixed-rank positive semidefinite matrices: A Riemannian approach*, Journal of Machine Learning Research, Vol. 12, pp. 593 – 625, February 2011
[35] C.W. Misner, K.S. Thorne and J.A. Wheeler, *Gravitation*, W.H. Freeman Publisher, 1973
[36] M. Moakher and P.G. Batchelor, *Symmetric Positive-Definite Matrices: From Geometry to Applications and Visualization*. Chapter in 'Visualization and Processing of Tensor Fields' (Joachim Weickert and Hans Hagen, Ed.s), Springer Series in Mathematics and Visualization, pp. 285 – 298, 2006
[37] M. Moreira and E. Fiesler, *Neural networks with adaptive learning rate and momentum terms*, Technical report 95-04 of the Institut Dalle Molle d'Intelligence Artificielle Perceptive, 1995
[38] H. Munthe-Kaas, *High order Runge-Kutta methods on manifolds*, Applied Numerical Mathematics, Vol. 29, No. 1, pp. 115 – 127, January 1999
[39] Y. Nishimori, S. Akaho and M.D. Plumbley, *Riemannian optimization method on the flag manifold for independent subspace analysis*, Proc. of the 6th International Conference on Independent Component Analysis and Blind Signal Separation (ICA'06), Lecture notes in computer science, Vol. 3889, pp. 295 – 302, Springer, Berlin, 2006
[40] L. Noakes, *A global algorithm for geodesics*, Journal of the Australian Mathematical Society (Series A), Vol. 64, pp. 37 – 50, 1998
[41] B. Nsiri, J.-M. Boucher and T. Chonavel, *Multichannel blind deconvolution application to marine seismic*, Proceedings of the OCEANS 2003 (22-26 Septembre 2003, San Diego, CA, USA), Vol. 5, 2761 – 2766, September 2003
[42] V.V. Phansalkar and P.S. Sastry, *Analysis of the back-propagation algorithm with momentum*, IEEE Transactions on Neural Networks, Vol 5, No. 3, pp. 505 – 506, May 1994
[43] N. Prabhu, H.-C. Chang and M. Deguzman, *Optimization on Lie manifolds and pattern recognition*, Pattern Recognition, Vol. 38, No. 12, pp. 2286 – 2300, December 2005
[44] C.-H. Qi, K.A. Gallivan and P.-A. Absil, *Riemannian BFGS algorithm with applications*, Recent Advances in Optimization and its Applications in Engineering (M. Diehl et al., Ed.s), Part 3, pp. 183 – 192, Springer-Verlag Berlin Heidelberg, 2010
[45] G. Qiu, M.R. Varley and T.J. Terrell, *Accelerated training of backpropagation networks by using adaptive momentum step*, Electronics Letters, Vol. 28, No. 4, pp. 377 – 378, February 1992
[46] V. Ramakrishna and F. Costa, *On the exponentials of some structured matrices*, Journal of Physics A: Mathematical and General, Vol. 37, pp. 11613 – 11627, 2004
[47] M.A.F. Sanjuán, *The effect of nonlinear damping on the universal escape oscillator*, International Journal of Bifurcation and Chaos, Vol. 9, No. 4, pp. 735 – 744, 1999
[48] F. Silva-Leite and P. Crouch, *Closed forms for the exponential mapping on matrix Lie groups, based on Putzer's method*, Journal of Mathematical Physics, Vol. 40, No. 7, pp. 3561 – 3568, July 1999
[49] M. Spivak, *A Comprehensive Introduction to Differential Geometry*, 2nd Edition, Berkeley, CA: Publish or Perish Press, 1979
[50] A. Swaminathan, M. Wu and K.J.R. Liu, *Digital image forensics via intrinsic fingerprints*, IEEE Transactions on Information Forensics and Security, Vol. 3, No. 1, pp. 101 – 117, March 2008
[51] N.T. Trendafilov and R.A. Lippert, *The multimode Procrustes problem*, Linear Algebra and Its Applications, Vol. 349, pp. 245 – 264, 2002
[52] B. Vandereycken and S. Vandewalle, *A Riemannian optimization approach for computing low-rank solutions of Lyapunov equations*, SIAM Journal on Matrix Analysis Applications, Vol. 31, No. 5, pp. 2553 – 2579, 2010
[53] L. Xu, E. Oja, and C.Y. Suen, *Modified Hebbian learning for curve and surface fitting*, Neural Networks, Vol. 5, pp. 441 – 457, 1992
[54] J. Wallner and N. Dyn, *Convergence and C^1 analysis of subdivision schemes on manifolds by proximity*, Computer Aided Geometric Design, Vol. 22, pp. 593 – 622, 2005

Simone Fiori received the Italian Laurea (Dr. Eng.) *cum laude* in Electronic Engineering in July 1996 from the University of Ancona (Italy) and the Ph.D. degree in Electrical Engineering (circuit theory) in March 2000 from the University of Bologna (Italy). He is currently serving as Adjunct Professor at the Faculty of Engineering of the Università Politecnica delle Marche (Ancona, Italy). His research interests include unsupervised learning theory for artificial neural networks, linear and non-linear adaptive discrete-time filter theory, vision and image processing by neural networks, continuous-time and discrete-time circuits for stochastic information processing, geometrical methods for machine learning and signal processing. He is author of more than 145 refereed journal and conference papers. Dr. Fiori was the recipient of the 2001 "E.R. Caianiello Award" for the best Ph.D. dissertation in the artificial neural network field and the 2010 "Rector Award" as a proficient researcher of the Faculty of Engineering of the Università Politecnica delle Marche. He is currently serving as Associate Editor for Neurocomputing (Elsevier), Computational Intelligence and Neuroscience (Hindawi) and Cognitive Computation (Springer).