

# Quantization and clustering on Riemannian manifolds with an application to air traffic analysis

Alice Le Brigant<sup>a</sup>, Stéphane Puechmorel<sup>a</sup>

<sup>a</sup>ENAC, Université de Toulouse, Toulouse France

---

## Abstract

The goal of quantization is to find the best approximation of a probability distribution by a discrete measure with finite support. When dealing with empirical distributions, this boils down to finding the best summary of the data by a smaller number of points, and automatically yields a  $K$ -means-type clustering. In this paper, we introduce Competitive Learning Riemannian Quantization (CLRQ), an online quantization algorithm that applies when the data does not belong to a vector space, but rather a Riemannian manifold. It can be seen as a density approximation procedure as well as a clustering method. Compared to many clustering algorithms, it requires few distance computations, which is particularly computationally advantageous in the manifold setting. We prove its convergence and show simulated examples on the sphere and the hyperbolic plane. We also provide an application to real data by using CLRQ to create summaries of images of covariance matrices estimated from air traffic images. These summaries are representative of the air traffic complexity and yield clusterings of the airspaces into zones that are homogeneous with respect to that criterion. They can then be compared using discrete optimal transport and be further used as inputs of a machine learning algorithm or as indexes in a traffic database.

*Keywords:* air traffic analysis, clustering, optimal quantization, Riemannian geometry.

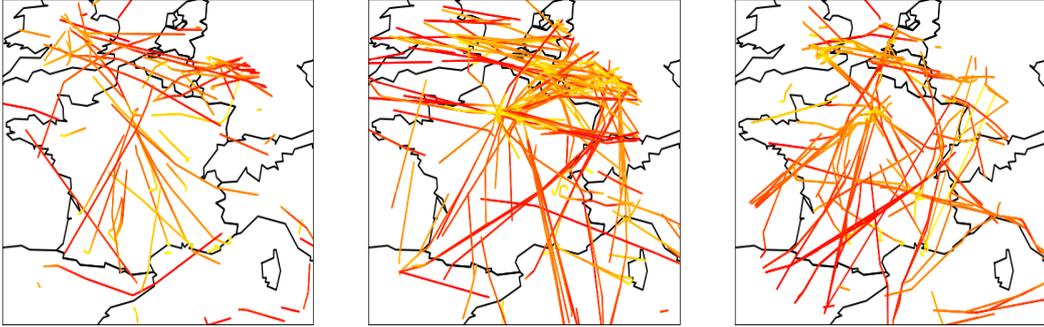
*2010 MSC:* 62H30, 62H35, 53-04.

---

## 1. Introduction

Most of the statistical tools developed so far are dedicated to data belonging to vector spaces, since it is the most convenient setting for algorithms as well as for theoretical derivations. However, when dealing with real world applications, such a framework may not fit with the structure of the data. It is obviously the case for geostatistics over world-sized datasets, but it is also true in many other fields: shapes in computer vision, diffusion tensor images in neuroimaging, signals in radar processing do not belong to an Euclidean space, but rather to a differentiable manifold. Riemannian geometry provides a convenient framework to deal with such objects. It allows a straightforward generalization of basic statistical notions such as means and medians [2, 14, 18], covariances [29], Gaussian distributions [31], and of methods based on linear operations, such as principal component analysis [12, 32]. The use of these statistical tools has met a growing interest in various fields, including shape analysis [19], computational anatomy [12], medical imaging [13], probability theory [4], and radar signal processing [1, 21].

In air traffic management (ATM), a major concern is the ability to infer an estimation of the complexity as perceived by a controller from the knowledge of aircraft trajectories in a given airspace, as depicted in Fig. 1. Many interdependent factors are involved in the cognitive process of a human controller, making the problem extremely difficult to solve, if even possible. However, there is a consensus among the experts on the importance of traffic disorder. As detailed in [43], a good way to estimate traffic disorder is to assume the spatial distribution of the aircraft velocities to be Gaussian and use the covariance function as an indicator for local complexity. This model requires to estimate a covariance matrix at each sample point of the airspace, of which we adopt a planar representation through a stereographic projection, and therefore yields a mapping from the plane to the space of symmetric, positive definite (SPD) matrices. Such a mapping will be referred to as an SPD image in the sequel. Working directly with the SPD images is an extremely computationally expensive task, that is unrealistic in practice. Moreover, the information provided is highly redundant, making it cumbersome to use in statistical analysis. To cope with this problem, we seek to produce summaries of the images. To do so, we model an SPD image as a realization of a random field with values



**Fig. 1.** Traffic in the French airspace during one-hour periods of time. Increasing variations of speed are shown from yellow to red.

in the space of SPD matrices. The values collected at each sample point in the image define an empirical probability distribution, supported on the space of SPD matrices. We propose to produce a summary of this empirical distribution using optimal quantization.

Optimal quantization is concerned with finding the best approximation, in the sense of the Wasserstein distance, of a probability distribution  $\mu$  by a discrete measure  $\hat{\mu}_n$  with finite support  $|\text{supp}\hat{\mu}_n| \leq n$  (see [15] or the survey paper [27], and references therein). When dealing with empirical distributions, this boils down to finding the best summary of the data by a smaller number of points. In the same setting, quantization naturally yields a clustering of the data points, which coincides with the solution given by the  $K$ -means algorithm. In our application, probability measures are supported on the space of SPD matrices, and therefore points actually refer to SPD matrices. The main challenge lies in the fact that SPD matrices do not form a vector space, but rather a Riemannian manifold [28], whereas most work on optimal quantization is suited for vector data [6, 25], or functional data [3, 26]. Nonetheless, the case of probability distributions on Riemannian manifolds has recently received attention [16, 20]. In particular, the asymptotic behavior of the quantization error, i.e., the evolution of the error made by approximating  $\mu$  by  $\hat{\mu}_n$  as  $n \rightarrow \infty$ , was studied for the manifold case in [16]. However, to the best of our knowledge, no numerical schemes have yet been introduced in this setting.

In this work, we introduce Competitive Learning Riemannian Quantization (CLRQ), a Riemannian counterpart of Competitive Learning Vector Quantization (see for example [27]). It is a gradient descent algorithm that computes the best approximation  $\hat{\mu}_n$  of a probability measure  $\mu$  over a Riemannian manifold using observations sampled from  $\mu$ . For empirical distributions, this allows to summarize a manifold-valued dataset of size  $N$  by a smaller number  $n \ll N$  of points, which additionally correspond to the centers of a  $K$ -means-type clustering. Recall that the classical Riemannian  $K$ -means algorithm computes at each step a partition of the data space (of the size of the desired number of clusters) and computes the geometric (Karcher) mean of each cell of the partition [34]. These two steps are usually very computationally expensive in the Riemannian setting, where the geodesic distance doesn't always have a closed form and the mean can only be obtained by an iterative procedure. Other clustering procedures relying on the estimation of distance-based kernels or graphs, such as mean-shift [33], or persistence-based clustering [10], require at least the computation of the matrix of pairwise-distances between the data points, if not the distances from the current (new) point to all the data points at each step. It is also the case of kernel-based approaches which first transport the data into a Reproducing Kernel Hilbert Space (RKHS) before applying a clustering procedure, less expensive in the vector setting. In [37], the authors propose to accelerate the procedure through random projections, which require to compute the distance matrix of a small subset of points instead of the whole dataset. In contrast, CLRQ is an online algorithm (i.e., processing one data point at a time) and requires at each step as many distance computations as the number of desired clusters, which is usually very small compared to the number of data. This leads us to argue that CLRQ is well-suited for large datasets.

After proving the convergence of CLRQ under the relevant assumptions, we use it to perform air traffic complexity analysis as presented above. Applying the algorithm to the empirical distribution of an SPD image yields two desirable

results: (1) a summary of the image through  $\hat{\mu}_n$ , and (2) a clustering of the image, and thereby of the corresponding airspace, into different zones homogeneous with respect to complexity. The latter means that to each point of the image is assigned a level of complexity according to the class it belongs to, and the former, that different traffic images can easily be compared through the comparison of their summaries, using the Wasserstein distance. This is an interesting prospect, since it allows for the indexation of air traffic situations in view of creating automatized decision-making tools to help air traffic controllers. Naturally, CLRQ can conceivably be applied to many other application-driven problems, requiring either the construction of summaries or the clustering of geometric data. A natural example of such data with a positive curvature is given by GPS positions on the Earth (spherical data); in negative curvature, one can think of Gaussian distributions, which can be parameterized by the hyperbolic half-plane in the univariate case, and by higher dimensional Riemannian symmetric spaces in the multivariate case [8, 22].

The paper is organized as follows. In Section 2, we introduce the Riemannian setting and give an example of a statistical object on a manifold (the Riemannian center of mass), before introducing the context of optimal quantization. In Section 3, we present the CLRQ algorithm and show its convergence. After showing some simple illustrations on the circle, the sphere and the hyperbolic half-plane in Section 4, we present our main application in air traffic management in Section 5.

## 2. Mathematical setup

### 2.1. Notions of Riemannian geometry

Let us begin by introducing some notations and reminding some basic notions of Riemannian geometry. We consider a differentiable manifold  $M$  of dimension  $d$  equipped with a Riemannian metric, i.e., a smoothly varying inner product  $\langle \cdot, \cdot \rangle_x$  defined on each tangent space  $T_x M$  at  $x \in M$ . Recall that  $T_x M$  is a linear approximation of  $M$  at point  $x$ , and contains all tangent vectors to  $M$  at  $x$ . The subscript  $x$  in the metric will often be omitted and the norm associated to the Riemannian metric  $\langle \cdot, \cdot \rangle$  will be denoted by  $\| \cdot \|$ . Vector fields on  $M$  are mappings  $X : M \rightarrow TM$  that associate to each point  $x$  a tangent vector  $X_x \in T_x M$ . It is possible to take the derivative of a vector field with respect to another using an affine connection, that is, a functional  $(X, Y) \mapsto \nabla_X Y$  that acts on pairs of vector fields according to the following rules: for any vector fields  $X, Y, Z$  and smooth real-valued function  $f$ ,

$$\begin{aligned}\nabla_{fX+Y}Z &= f\nabla_X Z + \nabla_Y Z, \\ \nabla_X(fY) &= X(f)Y + f\nabla_X Y, \quad \nabla_X(Y+Z) = \nabla_X Y + \nabla_X Z.\end{aligned}$$

If a vector field is defined along a curve  $\gamma(t)$ , i.e.,  $V(t) := X_{\gamma(t)}$ , then its covariant derivative along  $\gamma$  is denoted by  $\frac{DV}{dt}(t) := \nabla_{\gamma'(t)} V$ . There is only one affine connection that is symmetric, meaning  $XY - YX = \nabla_X Y - \nabla_Y X$ , and compatible with the Riemannian metric, that is

$$\frac{d}{dt}\langle U, V \rangle = \langle \frac{DU}{dt}, V \rangle + \langle U, \frac{DV}{dt} \rangle,$$

for any vector fields  $U, V$  along a curve  $\gamma$ . It is called the Levi-Civita connection and we will always take covariant derivatives with respect to that connection. The geodesics of  $M$  are the curves  $\gamma$  satisfying the relation  $\frac{D\gamma'}{dt} = 0$ , which implies that their speed has constant norm  $\|\gamma'(t)\| = \text{const}$ . They are also the local minimizers of the arc length functional  $l$ :

$$l: \gamma \mapsto \int_0^1 \|\gamma'(t)\| dt$$

where in the previous expression curves are assumed, without loss of generality, to be defined over the interval  $[0, 1]$ . The exponential map at point  $x$  is the mapping, denoted by  $\exp_x$ , that maps a tangent vector  $v$  of an open ball  $B_0 \subset T_x M$  centered in  $0 \in T_x M$  to the endpoint  $\gamma(1)$  of the geodesic  $\gamma : [0, 1] \rightarrow M$  verifying  $\gamma(0) = x$ ,  $\gamma'(0) = v$ ,

$$\exp_x(v) = \gamma(1).$$

Intuitively, the exponential map moves the point  $x$  along the geodesic starting from  $x$  at speed  $v$  and stops after covering the length  $\|v\|$ . The image  $\exp_x(B_0)$  is called a normal neighborhood of  $x$ . Conversely, the inverse of the exponential map associates to any  $y$  in this normal neighborhood the vector  $\exp_x^{-1}(y)$  that sends  $x$  on  $y$  through the

exponential map, and will be denoted by the more intuitive notation  $\vec{x}\bar{y}$ . We assume that  $M$  is complete, i.e., that the exponential map at  $x$  is defined on the whole tangent space  $T_x M$ . By the Hopf-Rinow theorem, we know that  $M$  is also geodesically complete, that is, any two points  $x, y \in M$  can be joined by a geodesic of shortest length. This minimal length defines the geodesic distance between  $x$  and  $y$ , denoted in the sequel by  $d(x, y)$ . For further details, we refer the reader to standard textbooks such as [9] or [17].

## 2.2. The Riemannian center of mass

Before introducing optimal quantization, let us briefly give an example of generalization of a fundamental statistical notion to the manifold setting: the Riemannian center of mass of a probability distribution, also called the Fréchet mean [14]. Consider a probability measure  $\mu$  on  $M$  with density with respect to the Riemannian volume form and a compact support  $K = \text{supp}\mu$ . As a compact set,  $K$  is contained in a geodesic ball  $B(a, R) = \{x \in M, d(a, x) < R\}$ , and we assume that this ball is convex, i.e., that for all points  $x, y \in \bar{B}(a, R)$ , there exists a unique minimizing geodesic from  $x$  to  $y$  (sufficiently small geodesic balls are convex [11, Theorem 5.14]). Let  $X$  be a random variable following the law  $\mu$ . The Fréchet mean  $\bar{x}$  of  $X$  generalizes the minimum-square-error property of the Euclidean mean

$$\bar{x} = \mathbb{E}_\mu(X) = \arg \min_{a \in M} \int_M d(x, a)^2 \mu(dx). \quad (1)$$

It can be characterized as the point where the gradient of the above functional vanishes, which gives

$$\int_M \vec{\bar{x}} \bar{x} \mu(dx) = 0. \quad (2)$$

This is due to the fact that the gradient of the functional  $f : a \mapsto d(x, a)^2$  is given by  $\text{grad}_a f = -2\vec{a}\bar{x}$  (see Lemma 1 in the Appendix), and to the assumptions we made on  $M$  and  $\text{supp}\mu$ . The same assumptions guarantee that  $\bar{x}$  exists and is unique [29]. When  $\mu$  is an empirical probability measure equally distributed on  $N$  data points  $x_1, \dots, x_N$ , the mean can be computed using a gradient descent algorithm, where the update is given by

$$\bar{x} \leftarrow \exp_{\bar{x}} \left( \frac{1}{N} \sum_{i=1}^N \vec{\bar{x}} \bar{x}_i \right).$$

This algorithm [29] is sometimes denoted by the name Karcher flow.

## 2.3. Optimal Riemannian quantization

Optimal quantization addresses the problem of approximating a random variable  $X$  with distribution  $\mu$  by a simplified version  $q(X)$  where  $q : M \rightarrow \Gamma \subset M$  is a measurable function with an image  $\Gamma$  of cardinal at most  $n$ . The function  $q$  is called an  $n$ -quantizer and is chosen to minimize the  $L^p$  criteria, called the *distortion function*,

$$\mathbb{E}_\mu \{d(X, q(X))^p\}. \quad (3)$$

Since  $q$  takes only a finite number of values, the distribution of  $q(X)$  will be a finite sum of point measures. It is well known that, since any  $n$ -quantizer  $q$  of image  $\Gamma \subset M$  satisfies for all  $x \in M$ ,  $d(x, q(x)) \geq \inf_{a \in \Gamma} d(x, a)$ , with equality if and only if  $q(x) = \arg \min_{a \in \Gamma} d(x, a)$ , the optimal quantizer is the projection to the nearest neighbor of  $\Gamma$ . Moreover, if  $|\Gamma| < n$  and  $|\text{supp}\mu| \geq n$ , one easily checks that  $q$  can always be improved, in the sense of criteria (3), by adding an element to its image. This means that an optimal  $n$ -quantizer has an image of exactly  $|\Gamma| = n$  points. Therefore, the optimal  $n$ -quantizer for criteria (3) is of the form  $q_\Gamma : M \rightarrow \Gamma = \{a_1, \dots, a_n\}$ , where the  $a_i$ 's are pairwise distinct, and

$$q_\Gamma(\cdot) = \sum_{i=1}^n a_i \mathbf{1}_{C_i(\Gamma)}(\cdot), \quad C_i(\Gamma) = \{x \in M, d(x, a_i) \leq d(x, a_j) \forall j \neq i\}.$$

The ties in  $C_i(\Gamma)$  are arbitrarily broken. The set  $C_i(\Gamma)$  is the  $i^{\text{th}}$  Voronoi cell associated to  $\Gamma$  and the union of all these cells form the Voronoi diagram. The quantization problem is therefore equivalent to the approximation of  $\text{supp}\mu$  by an  $n$ -tuple  $(a_1, \dots, a_n) \in M^n$  minimizing the cost function  $F_{n,p} : M^n \rightarrow \mathbb{R}_+$ ,

$$F_{n,p}(a_1, \dots, a_n) = \mathbb{E}_\mu \left\{ \min_{1 \leq i \leq n} d(X, a_i)^p \right\} = \int_M \min_{1 \leq i \leq n} d(x, a_i)^p \mu(dx). \quad (4)$$

This cost function is obtained by evaluating the distortion function (3) at  $q = q_\Gamma$ . Notice that if we try to approximate  $\mu$  by a single point  $a \in M$  (i.e.,  $n = 1$ ) with respect to an  $L^2$  criterion ( $p = 2$ ), we retrieve the definition (1) of the Riemannian center of mass.

Finally, there is a third way, in addition to (3) and (4), of expressing the quantization problem. It is also equivalent to the approximation of the measure  $\mu$  by the closest discrete measure  $\hat{\mu}_n$  supported by  $n$  points, with respect to the Wasserstein distance of order  $p$

$$W_p(\mu, \hat{\mu}_n) = \inf_P \int d(u, v)^p dP(u, v). \quad (5)$$

Here the infimum is taken over all measures  $P$  on  $M \times M$  with marginals  $\mu$  and  $\hat{\mu}_n$ . One can construct an optimal discrete approximation  $\hat{\mu}_n$  (i.e., a minimizer of (5)) from an optimal  $n$ -tuple  $(a_1, \dots, a_n)$  (i.e., a minimizer of (4)), and vice-versa, using

$$\hat{\mu}_n = \sum_{i=1}^n \mu \{C_i(\Gamma)\} \delta_{a_i}, \quad (6)$$

and then we have

$$F_{n,p}(a_1, \dots, a_n) = W_p(\mu, \hat{\mu}_n).$$

This is well known for the vector case [15] and applies verbatim to measures on manifolds. In the sequel, we will focus on the second formulation (4) of the quantization problem.

The first question that arises is the existence of a minimizer of (4). Since we have assumed that  $\mu$  has compact support, this existence is easily obtained.

**Proposition 1.** *Let  $M$  be a complete Riemannian manifold and  $\mu$  a probability distribution on  $M$  with density and a compact support. Then the distortion function  $F_{n,p}$  is continuous and admits a minimizer.*

**Proof.** Just as in the vector case [27, Proposition 1.1], for any  $x \in M$ , the function  $M^n \rightarrow \mathbb{R}_+$ ,  $\alpha = (a_1, \dots, a_n) \mapsto \min_{1 \leq i \leq n} d(x, a_i)$  is 1-Lipschitz for the distance  $d'(\alpha, \beta) := \max_{1 \leq i \leq n} d(a_i, b_i)$ , where  $\beta = (b_1, \dots, b_n)$ . Therefore it is continuous, and so is its  $p^{\text{th}}$  power. Since  $K = \text{supp } \mu$  is compact, for all  $\alpha \in M^n$  and all  $\beta$  in a neighborhood  $B(a_1, r_1) \times \dots \times B(a_n, r_n)$  of  $\alpha$ , we have

$$\forall x \in K, \quad \min_{1 \leq i \leq n} d(x, b_i)^p \leq \min_{1 \leq i \leq n} \left\{ \sup_{y \in K} d(y, a_i) + r_i \right\}^p < \infty.$$

So by dominated convergence,  $F_{n,p}$  is continuous. Recall that as a compact set,  $K$  is contained in a geodesic ball  $B(a, R)$ . If  $\alpha = (a_1, \dots, a_n) \in M^n$  is such that  $d(a, a_i) > 2R$  for at least one  $a_i$ , then for all  $x \in K$ ,  $d(x, a_i) \geq d(a, a_i) - R > R$ , and so the same  $n$ -tuple where  $a$  replaces  $a_i$  is a better candidate to minimize  $F_{n,p}$ . We can therefore limit our search to  $\bar{B}(a, 2R)$ , which is a closed and bounded subset of the complete manifold  $M$ , and thus compact. The continuous function  $F_{n,p}$  reaches a minimum on this compact, which is an absolute minimum.  $\square$  The elements of a minimizer  $\alpha = (a_1, \dots, a_n)$  are called *optimal  $n$ -centers* of  $\mu$ . The minimizer  $\alpha$  is in general not unique, first of all because any permutation of  $\alpha$  is still a minimizer, and secondly because any symmetry of  $\mu$ , if it exists, will transform  $\alpha$  into another minimizer of  $F_{n,p}$ . For example, any rotation of the optimal  $n$ -centers of the uniform distribution on the sphere preserves optimality.

The second question that comes naturally is: how does the error one makes by approximating  $\mu$  by  $\hat{\mu}_n$  (as given by (6)) evolve when the number  $n$  of points grows? The  $n^{\text{th}}$  *quantization error* is defined by

$$V_{n,p}(\mu) = \inf_{(a_1, \dots, a_n) \in M^n} F_{n,p}(a_1, \dots, a_n) = \inf_{(a_1, \dots, a_n) \in M^n} \int_M \min_{1 \leq i \leq n} d(x, a_i)^p \mu(dx).$$

In the vector case, Zador's theorem [15, Theorem 6.2] tells us that it decreases to zero as  $n^{-p/d}$ , and that the limit of  $n^{p/d} V_{n,p}(\mu)$  can be expressed in terms of the  $p^{\text{th}}$  *quantization coefficient*, i.e., the limit (which is also an infimum) when  $\mu$  is the uniform distribution on the unit square of  $\mathbb{R}^d$

$$Q_p([0, 1]^d) = \lim_{n \geq 1} n^{p/d} V_{n,p} \left\{ \mathcal{U}([0, 1]^d) \right\}.$$

More precisely, if  $\mu$  has an absolutely continuous part (with respect to the Lebesgue measure) with density  $h$ , then the limit of the  $n^{\text{th}}$  quantization error is equal to  $\mathcal{Q}_p([0, 1]^d) \|h\|_{d/(d+p)}$ . Moreover, when  $\mu$  has no singular part, the asymptotic empirical distribution of the optimal  $n$ -centers is proportional to  $h^{d/(d+p)}$ .

In the case of a Riemannian manifold  $M$ , the moment condition of the flat case generalizes to a condition involving the curvature of  $M$ . The following term measures the maximal variation of the exponential map at  $x \in M$  when restricted to a  $(d - 1)$ -dimensional sphere  $S_\rho \subset T_x M$  of radius  $\rho$

$$A_x(\rho) = \sup_{v \in S_\rho, w \in T_x S_\rho, \|w\| = \rho} \|d_v \exp_x(w)\|.$$

The following generalization of Zador's theorem to Riemannian quantization was proposed by Iacobelli [16].

**Theorem 1** ([16, Theorem 1.4 and Corollary 1.5]). *Let  $M$  be a complete Riemannian manifold without boundary, and let  $\mu = h \, \text{dvol} + \mu_s$  be a probability measure on  $M$ , where  $\text{dvol}$  denotes the Riemannian volume form and  $\mu_s$  the singular part of  $\mu$ . Assume there exist  $x_0 \in M$  and  $\delta > 0$  such that*

$$\int_M d(x, x_0)^{p+\delta} d\mu(x) + \int_M A_{x_0} \{d(x, x_0)^p\} d\mu(x) < \infty.$$

Then

$$\lim_{n \rightarrow \infty} n^{p/d} V_{n,p}(\mu) = \mathcal{Q}_p([0, 1]^d) \|h\|_{d/(d+p)},$$

where  $\|\cdot\|_r$  denotes the  $L^r$ -norm. In addition, if  $\mu_s = 0$  and  $(a_1, \dots, a_n)$  are optimal  $n$ -centers, then

$$\frac{1}{n} \sum_{i=1}^n \delta_{a_i} \xrightarrow{D} \lambda h^{d/(d+p)} dx \quad \text{as } n \rightarrow \infty,$$

where  $\xrightarrow{D}$  denotes convergence in distribution and  $\lambda$  is the appropriate normalizing constant.

In this work, we are interested in finding numerical schemes to compute the optimal  $n$ -centers  $\alpha = (a_1, \dots, a_n)$  in practice from potentially large sets of data. To do so, we will search for the critical points of the distortion function.

### 3. Competitive Learning Riemannian Quantization

#### 3.1. Differentiability of the distortion function

We assume that the only knowledge that we have of the probability measure  $\mu$  that we want to approximate is through an online sequence of i.i.d. observations  $X_1, X_2, \dots$  sampled from  $\mu$ . A classical algorithm used for quadratic ( $p = 2$ ) vector quantization is the *Competitive Learning Vector Quantization* algorithm, a stochastic gradient descent method based on the differentiability of the distortion function  $F_{n,2}$ . We propose here a natural extension of this method to our setting, i.e., a compactly-supported probability measure on a complete Riemannian manifold. It relies on the differentiability of the distortion function.

**Proposition 2.** *Let  $\alpha = (a_1, \dots, a_n) \in M^n$  be an  $n$ -tuple of pairwise distinct components and  $p > 1$ . Then  $F_{n,p}$  is differentiable and its gradient in  $\alpha$  is*

$$\text{grad}_\alpha F_{n,p} = \left( -p \int_{\mathring{C}_i(\alpha)} \frac{\overrightarrow{a_i x}}{\|a_i x\|^{p-1}} \frac{\overrightarrow{a_i x}}{\|a_i x\|} \mu(dx) \right)_{1 \leq i \leq n} \in T_\alpha M^n,$$

where  $\mathring{C}_i(\alpha)$  is the interior of the  $i^{\text{th}}$  Voronoi cell of  $\alpha$  and  $\overrightarrow{xy} := \exp_x^{-1}(y)$  denotes the vector that sends  $x$  on  $y$  through the exponential map. In particular, the gradient of the quadratic distortion function is given by

$$\text{grad}_\alpha F_{n,2} = \left( -2 \int_{\mathring{C}_i} \overrightarrow{a_i x} \mu(dx) \right)_{1 \leq i \leq n} = -2 \left( \mathbb{E}_\mu \mathbf{1}_{\{X \in \mathring{C}_i\}} \overrightarrow{a_i X} \right)_{1 \leq i \leq n}. \quad (7)$$

**Remark 1.** The first observation we can make is that optimal  $n$ -centers are Riemannian centers of mass of their Voronoi cells, as characterized by (2). Hence, the term  $n$ -centers is as appropriate in the Riemannian setting as it is in the vector case. More generally, for any value of  $p$ , each  $a_i$ ,  $i \in \{1, \dots, n\}$ , is the  $p$ -mean of its Voronoi cell, i.e., the minimizer of

$$a \mapsto \int_{\hat{C}_i(a)} d(x, a)^p \mu(dx).$$

Therefore, the optimal  $n$ -centers are always contained in the compact support of  $\mu$ .

**Remark 2.** The second observation is that the opposite direction of the gradient is, on average, given by the vectors inside the expectation. Competitive learning quantization consists in following this direction at each step  $k$ , that is, updating only the center  $a_i$  corresponding to the Voronoi cell of the new observation  $X_k$ , in the direction of that new observation. In the Riemannian setting, instead of moving along straight lines, we simply follow geodesics using the exponential map.

**Proof.** Let  $\alpha = (a_1, \dots, a_n) \in M^n$  be an  $n$ -tuple of pairwise distinct components,  $w = (w_1, \dots, w_n) \in T_\alpha M^n$  a tangent vector to  $\alpha$ , and let  $(-\epsilon, \epsilon) \ni t \mapsto \alpha(t) = (a_1(t), \dots, a_n(t))$  be a variation of  $\alpha$  in the direction given by  $w$ , i.e., such that  $a_i(0) = a_i$  et  $a_i'(0) = w_i$  for all  $i = 1, \dots, n$ . The functional  $t \mapsto \min_{1 \leq i \leq n} d(x, a_i(t))^p$  is differentiable for all  $x \notin \cup_{1 \leq i \leq n} \partial C_i(\alpha(t))$ , that is  $\mu$ -almost everywhere since  $\mu(\partial C_i(\alpha(t))) = 0$  for all  $i$  (this is shown in the Appendix, Lemma 2). Its derivative in  $t = 0$  is given by

$$\begin{aligned} \left. \frac{d}{dt} \right|_{t=0} \min_{1 \leq i \leq n} d(x, a_i(t))^p &= \sum_{i=1}^n \mathbf{1}_{\{x \in \hat{C}_i\}} \left. \frac{d}{dt} \right|_{t=0} d(x, a_i(t))^p = - \sum_{i=1}^n \mathbf{1}_{\{x \in \hat{C}_i\}} \frac{p}{2} [d(x, a_i(0))^2]^{p/2-1} 2 \left\langle \overrightarrow{a_i(0)x}, a_i'(0) \right\rangle \\ &= - \sum_{i=1}^n \mathbf{1}_{\{x \in \hat{C}_i\}} p \|\overrightarrow{a_i x}\|^{p-2} \left\langle \overrightarrow{a_i x}, w_i \right\rangle. \end{aligned}$$

To go from the first to the second equality, we have used the well known property that for any given  $x$ , the gradient of the function  $f : a \mapsto d(x, a)^2$  is given by  $\text{grad}_a f = -2 \exp_a^{-1} x = -2\overrightarrow{a x}$  (see Lemma 1 in the Appendix). We obtain by Cauchy-Schwarz, since  $x \in B(a, R)$  and  $a_i \in B(a, 2R)$  (recall that any  $n$ -tuple containing a coordinate outside of  $B(a, 2R)$  is a worse candidate than the same  $n$ -tuple where  $a$  replaces  $a_i$ ),

$$\left| \left. \frac{d}{dt} \right|_{t=0} \min_{1 \leq i \leq n} d(x, a_i(t))^p \right| \leq p \sum_{i=1}^n \|\overrightarrow{a_i x}\|^{p-1} \|w_i\| \leq p(3R)^{p-1} \sum_{i=1}^n \|w_i\|.$$

Therefore, by dominated convergence,  $t \mapsto \psi_{n,p}(\alpha(t))$  is differentiable and its differential in  $\alpha$  at  $w$  is given by

$$T_\alpha F_{n,p}(w) = \sum_{i=1}^n \left\langle -2 \int \mathbf{1}_{\{x \in \hat{C}_i\}} \|\overrightarrow{a_i x}\|^{p-1} \frac{\overrightarrow{a_i x}}{\|\overrightarrow{a_i x}\|} \mu(dx), w_i \right\rangle = \langle \langle \text{grad}_\alpha F_{n,p}, w \rangle \rangle,$$

where  $\langle \langle \cdot, \cdot \rangle \rangle$  denotes the  $L^2$  metric on  $M^n$ , which gives the desired result for the gradient.  $\square$

### 3.2. The algorithm

Assume that we have access to independent and identically distributed (i.i.d.) observations  $X_1, X_2, \dots$  sampled from  $\mu$ . We choose a sequence of positive steps  $(\gamma_k)_{k \geq 1} \subset (0, 1)$  satisfying the usual conditions

$$\sum_{k \geq 1} \gamma_k = +\infty, \quad \sum_{k \geq 1} \gamma_k^2 < +\infty. \quad (8)$$

We propose a procedure called Competitive Learning Riemannian Quantization, described in Algorithm 1. The steps are chosen to be in  $[0, 1]$  so that at each iteration, the center that is updated stays in the same Voronoi cell, guaranteeing that the centers stay pairwise distinct (if initially pairwise distinct).

Now we show the convergence of Algorithm 1, using a theorem from Bonnabel [5].

---

**Algorithm 1** Competitive Learning Riemannian Quantization
 

---

**Input:**  $X_1, X_2, \dots$  sampled from  $\mu$ ,  $\{\gamma_k\}_{k \geq 1}$  satisfying (8)

**Output:**  $\hat{\mu}_n$  summary of  $\mu$

$k \leftarrow 0$ , select  $\{a_1(0), \dots, a_n(0)\}$  at random

**repeat**

$i \leftarrow \operatorname{argmin}_j d(X_{k+1}, a_j(k))$

$a_i(k+1) \leftarrow \exp_{a_i(k)} \left\{ \overrightarrow{\gamma_{k+1} a_i(k) X_{k+1}} \right\}$

$a_j(k+1) \leftarrow a_j(k) \quad \forall j \neq i.$

$k \leftarrow k+1$

**until** convergence

$\{C_1, \dots, C_n\} \leftarrow$  Voronoi diagram of  $\{a_1(k), \dots, a_n(k)\}$

$\hat{\mu}_n \leftarrow \sum_{i=1}^n \mu(C_i) \delta_{a_i(k)}$

---

**Proposition 3.** Assume that the injectivity radius of  $M$  is uniformly bounded from below by some  $I > 0$ , and let  $(\alpha(k))_{k \geq 0}$  be computed using Algorithm 1 and samples from a compactly supported distribution  $\mu$ . Then  $F_{n,2}(\alpha(k))$  converges a.s. and  $\operatorname{grad}_{\alpha(k)} F_{n,2} \rightarrow 0$  as  $k \rightarrow \infty$  a.s.

**Proof.** The proof relies on [5, Theorem 1], and therefore we adopt the same notations. The cost function  $F_{n,2}$  and its gradient can be respectively expressed as expectations of the functions

$$Q(x, \alpha) = \min_{1 \leq i \leq n} d(x, a_i)^2,$$

$$H(x, \alpha) = \operatorname{grad}_{\alpha} Q(x, \alpha) = \left( -2 \mathbf{1}_{C_i}(x) \overrightarrow{a_i x} \right)_{1 \leq i \leq n}.$$

This is once again due to Lemma 1, given in the Appendix. As stated in Remark 1, the  $n$ -centers  $\alpha(k) = (a_1(k), \dots, a_n(k))$  are each the barycenter of their Voronoi cell and therefore always stay in the same compact ball  $B(a, R)$  as the data  $X_1, X_2, \dots$ . Since  $\|\mathbf{1}_{C_i(x)} \overrightarrow{y x}\| \leq 2R$  for all  $y, x \in K \subset B(a, R)$  and  $i = 1, \dots, n$ , the gradient  $H$  is bounded on  $K \times K$ . That is, all the assumptions of [5, Theorem 1] are verified and Algorithm 1 converges.  $\square$

### 3.3. Link to clustering

Notice that if the measure  $\mu$  one wants to approximate is the empirical measure of a dataset  $x_1, \dots, x_N$ ,  $N \gg n$ ,

$$\mu = \frac{1}{N} \sum_{k=1}^N \delta_{x_k},$$

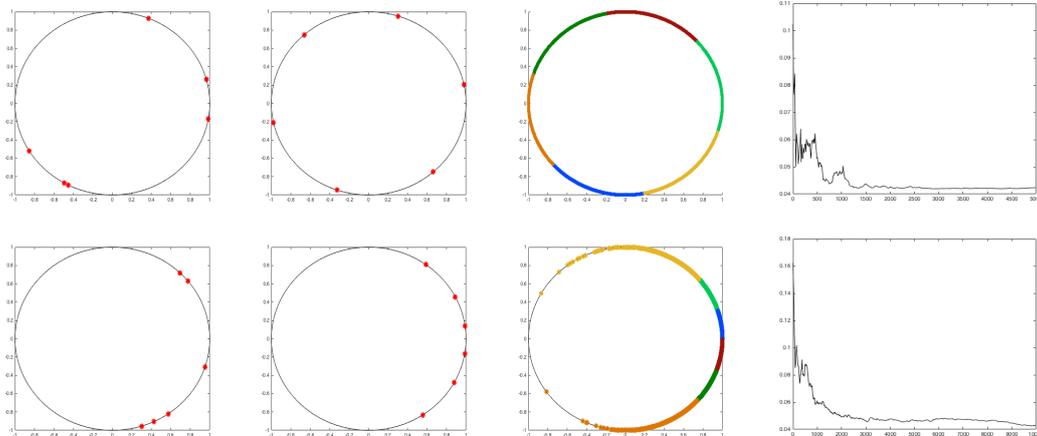
then the weight of the atom  $a_i$  in the optimal approximation is simply given by the proportion of data points that fall inside its Voronoi cell, that is,

$$\hat{\mu}_n = \sum_{i=1}^n \frac{|C_i(\Gamma)|}{N} \delta_{a_i}.$$

The atoms  $a_1, \dots, a_n$  minimize the distortion function, i.e., in this case, the sum of the squared geodesic distances within each Voronoi cell

$$F_{n,2}(a_1, \dots, a_n) = \sum_{i=1}^n \sum_{x_k \in C_i} d^2(x_k, a_i). \quad (9)$$

Notice that this is exactly the cost function of  $K$ -means clustering, an algorithm that iteratively replaces the current center of each cluster by the cluster center of mass until achieving convergence. Algorithm 1 (where the observations  $X_k$  are simply chosen randomly among  $x_1, \dots, x_N$ ) is an online version of the  $K$ -means algorithm in the Riemannian setting. Just as for classical  $K$ -means, the clusters are given by the Voronoi cells.



**Fig. 2.** Quantization of the uniform (top) and von Mises (bottom) distributions. From left to right: initial positions of the  $n$ -centers, final positions of the  $n$ -centers, associated Voronoi diagram, and evolution of the  $L^1$ -Wasserstein distance between the initial distribution and its quantized version.

#### 4. Examples

Now let us show some toy examples on manifolds of constant sectional curvature: the circle, the 2-sphere and the hyperbolic plane. Fig. 2 shows results of optimal quantization of size  $n = 6$  of the uniform (top row) and the von Mises (bottom row) distributions on the circle. The considered von Mises distribution is centered in 0 with concentration parameter  $K = 5$ . From left to right, Fig. 2 shows the initialization of Competitive Learning Riemannian Quantization (CLRQ), the optimal centers obtained after convergence, and the clustering (or Voronoi diagram) of the data taken as input of the algorithm. The centers are initialized uniformly on  $(-\pi, \pi]$  and  $[-\pi/2, \pi/2]$  respectively,  $S^1$  being identified with  $(-\pi, \pi]$ . In order to reduce dependency on the initialization, each step  $k$  is repeated a certain number  $m$  of times. In other words, the same step size is used several times. In the uniform case,  $m = 10$  is sufficient to obtain a visually satisfying result. For the von Mises distribution, we choose  $m = 50$ .

On the right-hand side of Fig. 2, we plot the evolution of the Wasserstein distance between the initial distribution and its quantized version. Recall that the quantization cost function (5) involves the  $L^2$ -Wasserstein distance. As shown in [30], the computation of the  $L^p$ -Wasserstein distance between two measures on the circle can be reduced to the same operation in the unit interval  $[0, 1]$  by "cutting" the circle at a certain point  $s \in S^1$ , i.e., by identifying it with a unit length fundamental domain for  $\mathbb{R}/\mathbb{Z}$ . However, when the two measures are not evenly distributed, the optimal cut is easier to find in the  $L^1$  case, therefore we choose to compute the weaker  $L^1$ -Wasserstein (or Kantorovich-Rubinstein) distance using the algorithm introduced in [7]. We plot the distance between the measure  $\mu$  and its approximation at each step  $k$  of Algorithm 1,

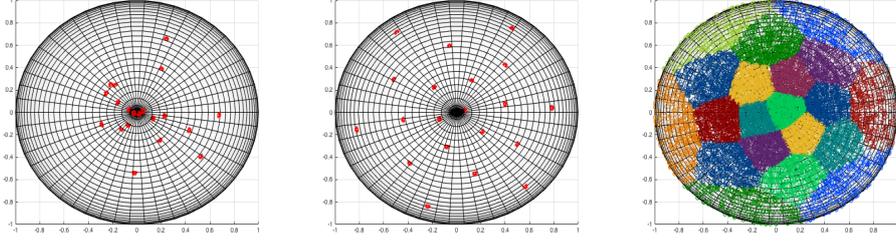
$$\hat{\mu}_n(k) = \sum_{i=1}^n \mu\{C_i(k)\} \delta_{a_i(k)},$$

where  $(a_1(k), \dots, a_n(k))$  are the  $n$ -centers at step  $k$ ,  $C_1(k), \dots, C_n(k)$  are the corresponding Voronoi cells, and  $\delta_x$  is the Dirac distribution at  $x \in M$ . Assuming that  $N \gg n$ , we can approximate  $\mu$  by the empirical measure of the observations  $x_1, \dots, x_N$ ,

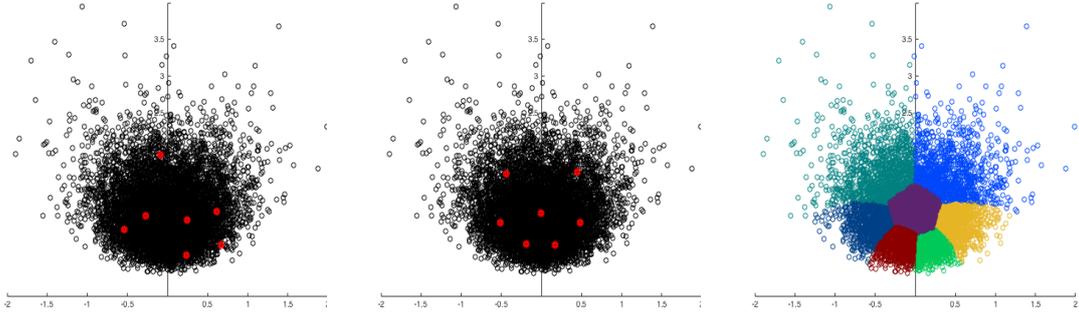
$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N \delta_{x_i}.$$

In order to compare two discrete measures with the same number of points, we then identify  $\hat{\mu}$  and  $\hat{\mu}_n(k)$  with the measures obtained on the reunion of their supports by completing with zero masses. For both the uniform and the von Mises examples, the Wasserstein distance decreases as expected.

Next, we show examples on the sphere. Fig. 3 displays, in the same order as Fig. 2, the initialization and results of the CLRQ algorithm applied to the von Mises distribution with concentration parameter  $K = 5$ , where each step contains  $m = 20$  iterations at the same step size. Finally, to show an example in the negative curvature setting, we



**Fig. 3.** Quantization of the von Mises distribution on the 2-sphere. From left to right: initialization of the centers, optimal centers, and the associated Voronoi diagram.



**Fig. 4.** Quantization of the Gaussian distribution on the hyperbolic half-plane. From left to right: initialization of the centers, optimal centers, and the associated Voronoi diagram.

use the work of Said et al. [31] regarding the definition and generation of Gaussian distributions on the space of SPD matrices to generate Gaussian samples on the hyperbolic half-plane. Recall that the hyperbolic half-plane is one of the models of 2-dimensional hyperbolic geometry, and can be identified with the complex upper-half plane  $\mathbb{H}^2 = \{z = x + iy, (x, y) \in \mathbb{R} \times \mathbb{R}_+^*\}$  where the length of an infinitesimal element  $dx + idy$  at point  $x + iy$  is measured by

$$ds^2 = \frac{dx^2 + dy^2}{y^2}.$$

The special linear group  $SL_2$  acts on  $\mathbb{H}^2$  from the left through the Moebius transformation:  $SO_2 \times \mathbb{H}^2 \rightarrow \mathbb{H}^2$ , defined by

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \cdot z = \frac{az + b}{cz + d}.$$

This action is transitive since for all  $(x, y) \in \mathbb{R} \times \mathbb{R}_+^*$ ,

$$\begin{pmatrix} y^{1/2} & xy^{-1/2} \\ 0 & y^{-1/2} \end{pmatrix} \cdot i = x + iy.$$

Noticing that  $SO_2$  is the stabilizer of  $i$ , we can identify  $\mathbb{H}^2$  with  $SL_2/SO_2$ , which is also homeomorphic to the space of  $2 \times 2$  SPD matrices of determinant 1 [17]. The space of SPD matrices of determinant 1 is therefore homeomorphic to  $\mathbb{H}^2$ , and the homeomorphism is given by  $\Phi : P \mapsto L \cdot i$ , where  $L$  is the upper-triangular matrix of the Cholesky decomposition of the SPD matrix  $P = L^T L$ . To generate a Gaussian sample in  $\mathbb{H}^2$ , we generate a Gaussian sample of SPD matrices using [31, Proposition 6], renormalize and transport them to the hyperbolic half-plane using  $\Phi$ . The first row of Fig. 4 shows the initialization and results of the CLRQ algorithm applied to observations sampled from the Gaussian distribution centered in  $i$  and with standard deviation  $\sigma = 0.5$ . Each step contains  $m = 20$  iterations.

## 5. Application to air traffic complexity analysis

### 5.1. Context

This work was motivated by complexity analysis in air traffic management (ATM). ATM deals with the means of organizing the aircraft trajectories in a given airspace so as to ensure both safety and efficiency. One of the most important part of ATM is the air traffic control (ATC) that acts on flying or taxiing aircraft in such a way that separation norms are satisfied at all time. Nowadays, most of the ATC is surveillance based, relying primarily on information coming from the radars to give instructions to planes. Even in country-sized airspaces the amount of traffic to be controlled is far beyond the limits of a single operator and the area under the responsibility of an air traffic controller (ATCO) has to be kept tractable. As a consequence, the whole airspace must be divided into elementary cells, known as control sectors, that are allotted to a team of ATCOs. A major concern arising in the design of such sectors is to ensure that the ATCOs workload is equally balanced over the airspace. Highly complex areas, involving many flows crossings and altitude changes, like those encountered close to the largest airports, must be kept small, while low complexity ones, with either a small aircraft density or a simple airspace structure may be made large. Finding a complexity indicator that can be computed using only airspace and traffic information and that closely mimics the perceived workload is a difficult, still unsolved problem [39, 40].

In France, the current operational indicator is the number of aircraft present in a control section at a given time. It is used by the European network manager to compute departure slots so that the value given is unlikely to be exceeded. This crude indicator is known to be too coarse as in some situations, ATCOs are willing to accept more aircraft than the threshold, while in others, they experience control overload with fewer aircraft. This is mainly due to differences in the traffic structure, that may be highly organized, thus easy to control, or on the contrary close to random and requiring a lot of attention. It is worth noticing that an incomplete knowledge of the complexity perceived by the ATCOs induces unnecessary delays, as some take-off slots would have been improved if the real acceptable number of aircraft in control sectors had been known.

In research, one of the most widely used indicators is the dynamic density [38]. It combines influential factors, like the number of maneuvering aircraft, the number of level changes and so on, to output a single positive real value representing the complexity level. Although quite pertinent from an operational point of view, the dynamic density is a tool requiring a lot of tuning, involving experiments with a panel of ATCOs. These experiments would be costly to implement in practice and cannot be adapted to different airspaces without having to re-tune from scratch. For the same reason, it is quite difficult to use dynamic density for assessing the performance of new concepts, since in such a case there is no reference situation or existing operational context that may be used to perform the tuning phase.

On the other hand, purely geometrical indicators have been introduced [41, 42], that are free of the dynamic density limitations. While perfectly suited to infer an intrinsic notion of complexity, they do not model all the aspects of the workload, as perceived by a human operator. The approach taken in the present work may be viewed as a mixture between the two previous ones: it relies on an intrinsic description of traffic, but does not produce a complexity value. Instead, a summary of the situation is issued, that serves as an input to a classification or machine learning algorithm. Even if this last phase looks very similar to a kind of dynamic density evaluation, it is far less complex:

- The traffic summary itself requires no tuning, while the influential factors taken into account in the dynamic density have weights that must be adjusted during the learning phase.
- Complexity classes are computed instead of complexity levels: a clustering algorithm will first segment the traffic dataset into homogeneous subsets, then a workload value will be associated by experts to each of them. This process is lightweight, since only the representative in each class has to be expertized.
- Adaptation to different airspaces is an easy task for the same reason: experts will evaluate only the representative situation in each class.

The first step is to model the spatial distribution of the aircraft velocities as a Gaussian law. Then, the covariance function is used as an indicator of traffic complexity. This assumption makes sense from an operational point of view as it represents the local organization, that is the most influential factor on ATCCO workload.

### 5.2. Estimating the covariance matrices

Although we will in practice consider time-windows, we start by considering a given airspace at a fixed time  $t$  containing  $N$  aircraft either flying or taxiing. We respectively denote by  $z_i$  and  $v_i$  the position and speed of the aircraft  $i$ ,  $1 \leq i \leq N$ , at time  $t$ . Since the altitude of an aircraft plays a special role and does not appear on controllers displays, we choose to disregard it and adopt a planar representation through a stereographic projection. An underlying Gaussian field model is assumed for the relation between the velocity and the position, whose variance function will be interpreted as a pointwise measure of the traffic complexity.

A non parametric Nadaraya-Watson type approach [24, 35] was taken to estimate the mean and variance functions at point  $z$ :

$$\begin{aligned}\hat{m}(z) &= \frac{\sum_{i=1}^N v_i K_h(z - z_i)}{\sum_{j=1}^N K_h(z - z_j)}, \\ \hat{\Sigma}(z) &= \frac{\sum_{i=1}^N \{v_i - \hat{m}(z)\} \{v_i - \hat{m}(z)\}^\top K_h(z - z_i)}{\sum_{j=1}^N K_h(z - z_j)}.\end{aligned}\tag{10}$$

The weights are given by a kernel function  $K$ , i.e., a positive, symmetric function integrating to one, scaled by a factor  $h > 0$ :  $K_h(x) = h^{-1}K(x/h)$ . Since most kernels have compact support, the estimations are based in practice on a number of observations much smaller than the size  $N$  of the sample. The estimator  $\hat{\Sigma}$  has been studied in [36] where it is shown to be asymptotically normal. Evaluating it at positions  $z_1, \dots, z_N$  yields a series of symmetric, positive definite matrices  $\hat{\Sigma}(z_1), \dots, \hat{\Sigma}(z_N)$  with empirical distribution

$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N \delta_{\hat{\Sigma}(z_i)},$$

where  $\delta_\Sigma$  denotes the Dirac mass at  $\Sigma$ . In order to obtain a summary of the traffic complexity, we propose to quantize  $\hat{\mu}$  using the CLRQ algorithm on the space of SPD matrices.

### 5.3. The geometry of SPD matrices

For the sake of completeness, let us briefly recall the most commonly used Riemannian structure [28] on the space  $\mathcal{P}_d$  of  $d \times d$  symmetric, positive definite matrices. Note that in this application, we are simply interested in the case  $d = 2$ . The Euclidean dot product on the space  $\mathcal{M}_d$  of square matrices of size  $d$  is given by the Frobenius inner product  $\Sigma_1 \cdot \Sigma_2 = \text{tr}(\Sigma_1^\top \Sigma_2)$ , where  $\text{tr}$  denotes the trace function. As an open subset of the vector space  $\mathcal{M}_d$ ,  $\mathcal{P}_d$  is a manifold where the tangent vectors are symmetric matrices. It can be equipped with a Riemannian metric invariant with respect to the action of the general linear group  $GL_d \times \mathcal{P}_d \rightarrow \mathcal{P}_d$ ,  $(A, \Sigma) \mapsto A^\top \Sigma A$ . At the identity, this metric is given by the usual Euclidean scalar product  $\langle W_1, W_2 \rangle_{\text{Id}} = W_1 \cdot W_2 = \text{tr}(W_1^\top W_2)$ , and at  $\Sigma$ , we ask that the value of the scalar product does not change when the tangent vectors are transported back to the identity via the action of  $A = \Sigma^{-1/2}$ , i.e.,

$$\langle W_1, W_2 \rangle_\Sigma = \langle \Sigma^{-1/2} W_1 \Sigma^{-1/2}, \Sigma^{-1/2} W_2 \Sigma^{-1/2} \rangle_{\text{Id}} = \text{tr}(\Sigma^{-1/2} W_1 \Sigma^{-1} W_2 \Sigma^{-1/2}).\tag{11}$$

As shown in [28], the associated geodesic distance is given by

$$d(\Sigma_1, \Sigma_2) = \sqrt{\sum_{i=1}^d \log^2(\lambda_i(\Sigma_1^{-1/2} \Sigma_2 \Sigma_1^{-1/2}))},\tag{12}$$

where we use the notation  $\lambda_i(\Sigma)$ ,  $i = 1, \dots, d$ , to denote the eigenvalues of  $\Sigma$ . Recall that in order to update the centers of the discrete approximation in the CLRQ algorithm, we need the exponential map, i.e., a mapping that associates to each point  $\Sigma$  and tangent vector  $W$  at  $\Sigma$  the end point of the geodesic starting from  $\Sigma$  at speed  $W$ . In the case of metric (11), it is given by

$$\exp_\Sigma(W) = \Sigma^{1/2} \exp(\Sigma^{-1/2} W \Sigma^{-1/2}) \Sigma^{1/2},$$

where the  $\exp$  on the right-hand side denotes the matrix exponential. Finally, we also need the inverse mapping, i.e., the logarithm map

$$\log_{\Sigma_1}(\Sigma_2) = \overrightarrow{\Sigma_1 \Sigma_2} = \Sigma_1^{1/2} \log(\Sigma_1^{-1/2} \Sigma_2 \Sigma_1^{-1/2}) \Sigma_1^{1/2},$$

where the log on the right-hand side denotes the matrix logarithm. Note that the matrix logarithm is well defined for any symmetric matrix  $\Sigma$  due to the factoring out, in the logarithm series, of the rotation matrices of the spectral decomposition  $\Sigma = UDU^T$ , yielding  $\log(UDU^T) = U \log(D)U^T$ . Endowed with this Riemannian structure, the space of SPD matrices is a geodesically complete Riemannian manifold with infinite injectivity radius at every point (i.e., no cut locus), and so all the exposed theoretical results are valid in this case.

#### 5.4. Real data analysis

##### 5.4.1. Segmenting and constructing summaries

We now have all the tools to construct summaries of the traffic complexity in a given airspace during a certain time period. As input, we consider an image such as the ones displayed in the first row of Fig. 5, showing the traffic over Paris, Toulouse and Lyon during a one-hour period of time. The color is related to the norm of the velocity, increasing from yellow to red. To simplify, we center and reduce the velocities  $v_i$ . The samples  $(z_i, v_i)$  are seen as observations arriving in a random order, and the covariance matrix at  $z_i$  is estimated according to (10) using a truncated Gaussian kernel  $K(x) = 1/\sqrt{2\pi} e^{-x^2/2} \mathbf{1}_{|x|<r}$ . The truncation of size  $r$  avoids useless computations. The procedure is described in detail in Algorithm 2.

---

#### Algorithm 2 CLRQ for air traffic complexity analysis

---

**Input:** positions  $z_1, \dots, z_N$  and speeds  $v_1, \dots, v_N$ , steps  $\{\gamma_k\}_{k \geq 1}$  satisfying (8)

**Output:** summary  $\hat{\mu}_n$  and clustering  $\{c_1, \dots, c_n\}$

select  $z'_1, \dots, z'_n$  randomly among  $\{z_1, \dots, z_N\}$  and compute  $\hat{\Sigma}(z'_1), \dots, \hat{\Sigma}(z'_n)$  using (10)

$k \leftarrow 0, A_i(0) \leftarrow \hat{\Sigma}(z'_i)$  for  $i \in \{1, \dots, n\}$

**repeat**

choose  $z'$  randomly among  $\{z_1, \dots, z_N\}$  and compute  $\Sigma_{k+1} = \hat{\Sigma}(z')$  using (10)

$i \leftarrow \operatorname{argmin}_j d(\Sigma_{k+1}, A_j(k))$

$A_i(k+1) \leftarrow \exp_{A_i(k)} \left\{ \gamma_{k+1} \overrightarrow{A_i(k)\Sigma_{k+1}} \right\}$

$A_j(k+1) \leftarrow A_j(k) \quad \forall j \neq i$

$k \leftarrow k+1$

**until** convergence

$\{C_1, \dots, C_n\} \leftarrow \text{Voronoi diagram of } \{A_1(k), \dots, A_n(k)\}$

$c_i \leftarrow \{z_k \mid \hat{\Sigma}(z_k) \in C_i\}$  for  $i \in \{1, \dots, n\}$

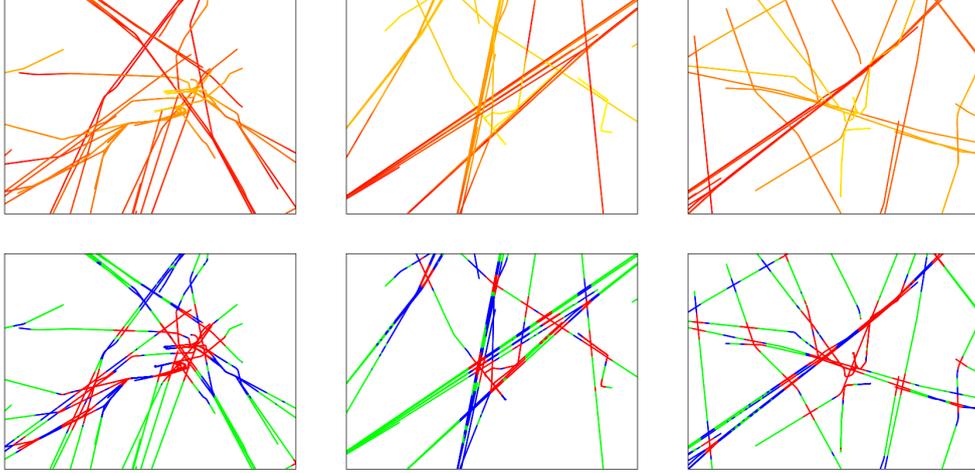
$\hat{\mu}_n \leftarrow \sum_{i=1}^n \frac{|C_i|}{N} \delta_{A_i(k)}$

---

In practice, we usually look for  $n = 3$  centers, i.e., the best approximation by three points. Indeed, we have found that generically, the centers can be ordered for the Loewner order when  $n = 3$  but not always for  $n > 3$ . (Recall that the Loewner order is a partial order on the space of SPD matrices, according to which  $A \geq B$  if  $A - B$  is positive semi-definite.) This can be explained by the fact that the Riemannian metric (11) sorts by rank, and therefore the covariance matrices are segmented into those with both eigenvalues close to zero (since the velocities are centered with respect to the neighborhood average, parallel velocities yield an estimated covariance matrix close to zero), those with only one eigenvalue close to zero, and the others. In the second row of Fig. 5, these clusters are respectively shown in green, blue and red. As could be expected, the first cluster corresponds to zones with either an isolated trajectory or parallel trajectories, the second to simple crossings or variations of speed in a given trajectory, and the third to zones with high density and crossings involving many trajectories. Naturally, the choice of the size  $r$  of the kernel's support has a great influence on the clustering, and it should be adjusted according to the minimum distance authorized between two aircraft in a zone considered as non conflictual.

##### 5.4.2. Importance of the Riemannian structure

In Fig. 6, we consider the whole French airspace over different one-hour periods of time. The clusterings obtained using CLRQ is shown in the middle column. To illustrate the importance of the Riemannian setting with respect to the Euclidean one, we show results of Competitive Learning Vector Quantization (CLVQ) on the same datasets, i.e., the same algorithm where the centers are updated using straight lines (linear interpolations between the matrix



**Fig. 5.** Traffic in the airspaces over Paris (top left), Toulouse (top middle) and Lyon (top right) during a one-hour period of time. Increasing variations of speed are shown from yellow to red. Clustering of these airspaces is achieved using CLRQ (bottom row). The zones in red correspond to covariance matrices of high Loewner order, in blue of intermediate Loewner order and in green low order. They can be interpreted as zones of decreasing complexity.

coefficients) and the distances are computed using the Frobenius norm. These results are shown in the right column of Fig. 6, and are less convincing. Many crossings and convergence points of different trajectories, such as the area around Paris, are put in the same class as isolated trajectories by the algorithm. This is contrary to what we want, since the former present a high level of complexity for controllers in contrast to the latter. Moreover, unlike with CLRQ, the matrices obtained as centers of the CLVQ clustering cannot be ordered in general, and thus the clusters cannot be interpreted as representing increasing levels of complexity. In Fig. 7, we show that the initialization has little influence on the segmentation of the airspace, which is satisfactory.

#### 5.4.3. Comparing summaries

We now propose a way to compare different summaries. A natural way to do so is through discrete optimal transport, which allows one to compute the distance between two discrete measures

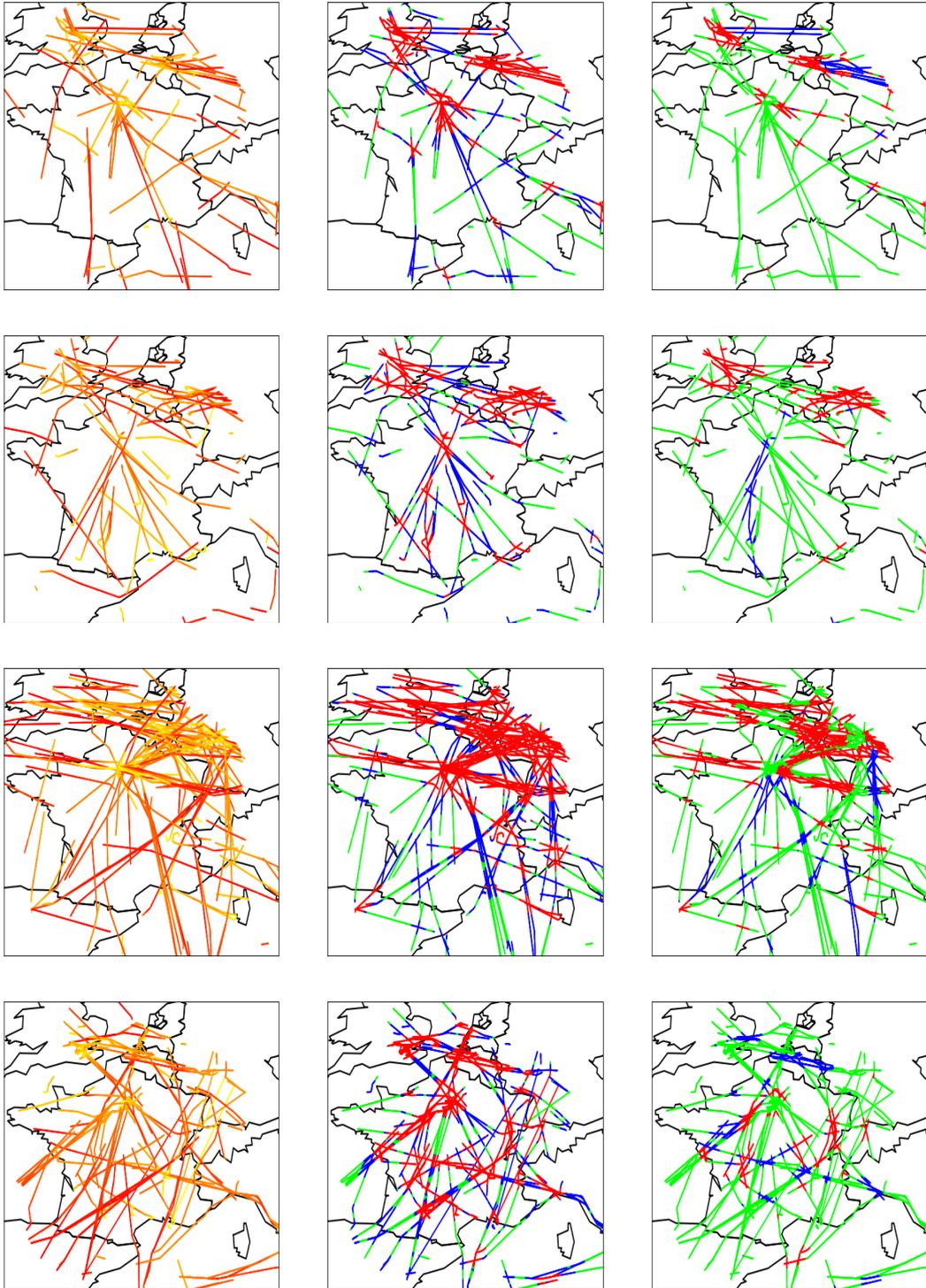
$$\mu = \sum_{i=1}^m \mu_i \delta_{A_i} \quad \text{and} \quad \nu = \sum_{j=1}^n \nu_j \delta_{B_j}.$$

In our case, the  $A_i$ 's and  $B_j$ 's are SPD matrices. Optimal transport seeks to transport the mass from  $\mu$  to  $\nu$  in a way that minimizes a certain cost. Formally, a transport plan between  $\mu$  and  $\nu$  is a matrix  $\pi = (\pi_{ij})_{i,j}$  with non-negative entries that verifies the two following properties for all  $i \in \{1, \dots, m\}$  and  $j \in \{1, \dots, n\}$ ,

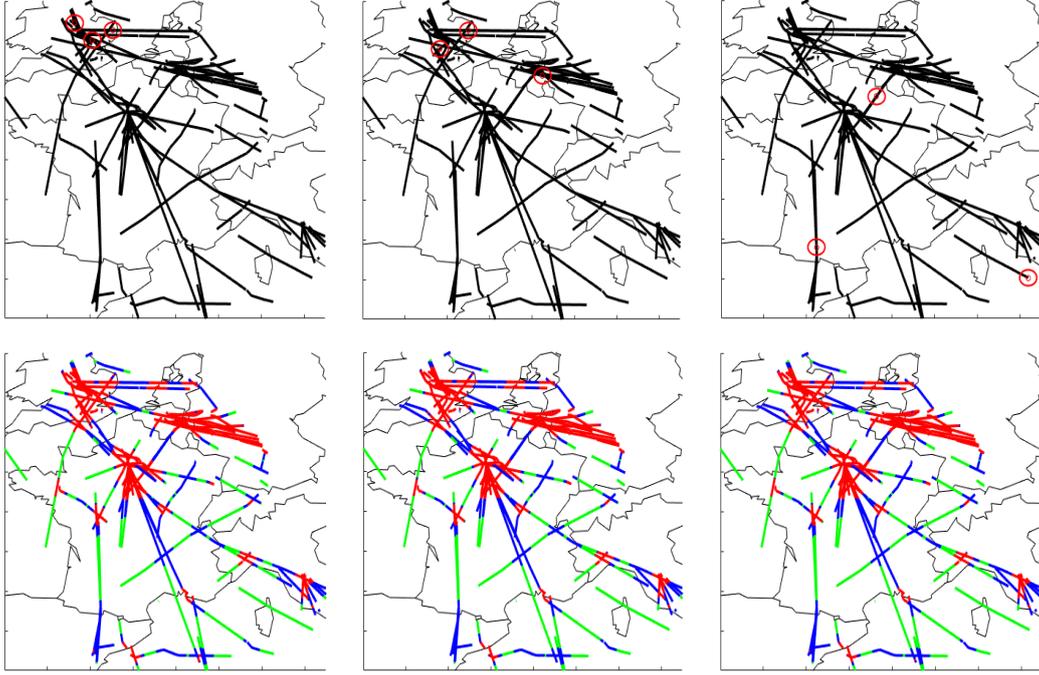
$$\sum_{j=1}^n \pi_{ij} = \mu_i \quad \text{and} \quad \sum_{i=1}^m \pi_{ij} = \nu_j.$$

The set of transport plans between  $\mu$  and  $\nu$  is denoted by  $\Gamma(\mu, \nu)$ . Intuitively, the value  $\pi_{ij}$  represents the part of the mass  $\mu_i$  transported from  $A_i$  to  $B_j$  to reconstruct  $\nu_j$ . Here, we measure the cost of transporting an infinitesimal unit of mass from  $A_i$  to  $B_j$  by the corresponding geodesic distance  $d(A_i, B_j)$ . The optimal transport plan is chosen to minimize the global cost, i.e., to be a solution of the following minimization problem

$$W_2(\mu, \nu) = \min \left\{ \sum_{i=1}^m \sum_{j=1}^n \pi_{ij} d(A_i, B_j)^2; \pi \in \Gamma(\mu, \nu) \right\}. \quad (13)$$



**Fig. 6.** Traffic in the French airspace during one-hour periods of time (left column). Increasing variations of speed are shown from yellow to red. We compare the clustering of these airspaces obtained using Riemannian quantization (middle column), versus vector quantization (right column). In the former case the affine-invariant distance (12) is used, and in the latter the Frobenius norm.



**Fig. 7.** Clustering of the French airspace with 3 different initializations. The neighborhoods chosen to compute the initial centers are shown by red circles (top row), and the corresponding clustering are shown below (bottom row).

The value of this minimal cost is the discrete  $L^2$ -Wasserstein distance between  $\mu$  and  $\nu$ . In general, the linear programming problem (13) is difficult and many different algorithms have been developed to solve it in various special cases [23]. However, in our setting, the discrete measures involved are supported by a very small number of points (3 in the examples shown) and problem (13) presents no difficulties. The matrix of distances between the summaries corresponding to the clusterings shown in the middle column of Fig. 6 is

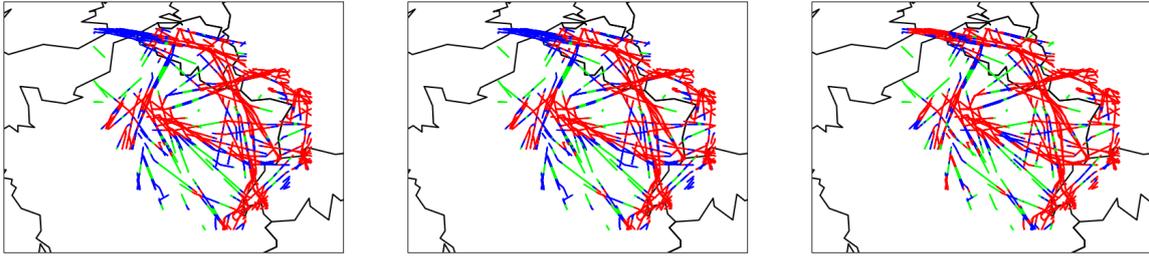
$$\begin{bmatrix} 0.00 & 1.92 & 6.74 & 4.55 \\ 1.92 & 0.00 & 8.31 & 6.07 \\ 6.74 & 8.31 & 0.00 & 1.22 \\ 4.55 & 6.07 & 1.22 & 0.00 \end{bmatrix}.$$

As expected, the first two situations are deemed similar but very different from the third one which has a much more complex traffic. The last situation is intermediary. In comparison, the different summaries of Fig. 7 obtained for the same traffic situation but different initializations are at small distances from one another, as can be seen from the following distance matrix

$$\begin{bmatrix} 0.000 & 0.033 & 0.031 \\ 0.033 & 0.000 & 0.016 \\ 0.031 & 0.016 & 0.000 \end{bmatrix}.$$

#### 5.4.4. Evaluation of the quantization results

The French territory is divided in five en-route control centers (North, East, West, South-East and South-West), which are responsible for controlling aircraft in the corresponding area of the French airspace. Each of them is partitioned into airspace blocks, known as elementary control sectors, and designed in such a way that the heaviest expected traffic is still manageable. For that purpose, the geometry of the elementary sectors is carefully chosen, so that the number of aircraft is small enough, but at the same time crossing points are far from the boundaries. Since elementary control sectors are adapted to worst-case situations, they are no longer optimal for light to medium loaded traffic. Grouping of sectors is then triggered, to obtain larger airspace blocks thus reducing the number of ATCOs



**Fig. 8.** Clustering of the French Eastern control center from 6:36 to 6:51 AM, obtained by 3 different clustering methods (from left to right): CLRQ, Riemannian  $K$ -means, and Kernelized Gaussian Random Projection.

needed. On the other hand, when the traffic starts to densify, degrouping occurs to cope with the increased load. This mechanism has a major effect on controllers workload, and is easily observed when conducting experiments with control simulators. It is also a good a posteriori indicator of complexity.

We evaluate the performance of the CLRQ algorithm on a dataset corresponding to the air traffic over the Eastern control center during a time period of a few hours. First, we compare the clustering given by quantization to other clustering methods. Fig. 8 shows, from left to right, the clusterings obtained for a fifteen-minute time slot by CLRQ, Riemannian  $K$ -means, and Kernelized Gaussian Random Projection (KGRP) [37]. Recall, the  $K$ -means algorithm iteratively replaces the current center of each cluster by the cluster center of mass (as defined in Section 2.2) until achieving convergence. CLRQ is an online version of the Riemannian  $K$ -means algorithm. KGRP projects the data points in a RKHS (Reproducing Kernel Hilbert Space) generated by a randomly selected subset of the data points, before performing  $K$ -means in that vector space. This enables to use the less computationally expensive vector version of  $K$ -means on non-vector data. As suggested in [37], we use a Gaussian kernel, we set the dimension of the RKHS to 300, and we use a subset of 100 points to generate it. Without optimizing any of the corresponding codes, CLRQ improves computation time by a factor of at least 10 with respect to  $K$ -means and KGRP. Both of these could be made faster with a better optimized computation of the Voronoi diagrams computed at each step, but CLRQ could also be optimized through parallel computations. Comparison of optimized versions of these codes is postponed to future work. As expected, the clustering results of CLRQ and  $K$ -means are very similar, and both algorithms attain approximately the same value of intra-cluster variance (9) (6.489 and 6.481 respectively), meaning that the online version (CLRQ) compares well to the offline version. In contrast, the KGRP method yields a much larger value (approximately 12.3), since  $K$ -means is performed in the projection space instead of the space of SPD matrices.

Finally, we confront our results to the perception of an ATCO. For this dataset, we have access to an ATCO's perception of the complexity of the traffic, evaluated every five minutes, in real time, in the form of an index between 0 and 5. Since the ATCO's perception of the traffic complexity can be influenced by the recent past as well as by the anticipation of upcoming situations, it is not easy to infer the true real-time complexity of the traffic from it. In other words, a non-zero time interval between reality and human reaction seems inevitable. Keeping this in mind, we compare this perceived complexity index, shown in Fig. 9 by the blue curve in the fifth row, to mean complexity indices computed using the summaries obtained for the three different clustering algorithms (CLRQ,  $K$ -means and KGRP) and for  $m = 15$  time slots of fifteen minutes each. Note that comparison to CLVQ (online vector  $K$ -means) is not pursued because the centers cannot be ordered in that case. At each time  $t_k$ ,  $k = 1, \dots, m$ , and for each summary  $\hat{\mu}_n(t_k) = w_1(t_k)\delta_{A_1(t_k)} + w_2(t_k)\delta_{A_2(t_k)} + w_3(t_k)\delta_{A_3(t_k)}$ , the mean complexity is computed through a weighted sum of three different complexity levels  $\lambda_1, \lambda_2, \lambda_3$ , where the weights are those of the clusters

$$c_\lambda(t_k) = \lambda_1 w_1(t_k) + \lambda_2 w_2(t_k) + \lambda_3 w_3(t_k).$$

Here, the weights are numbered according to the order of the corresponding cluster center ( $w_1, w_2, w_3$  correspond to  $A_1 < A_2 < A_3$  with respect to the Loewner order) and are given by the proportions  $w_i(t) = |C_i(t)|/N$  of matrices in the dataset belonging to the clusters. For each algorithm,  $\lambda = (\lambda_1, \lambda_2, \lambda_3)$  is chosen to minimize the mean square error to

the human perception  $I(t)$  of the complexity

$$\lambda = \arg \min_{\lambda} \sum_{k=1}^m |c_{\lambda}(t_k) - I(t_k)|^2.$$

The details of the summary weights are shown in the three first rows ( $w_1$  in green,  $w_2$  in blue,  $w_3$  in red). As expected, CLRQ and  $K$ -means give very similar results. The minimal errors (i.e., mean square errors to the ATCO's perception) for CLRQ and  $K$ -means are very close (58.0 and 58.7 respectively), and they are lower than that of KRGP (73.4), which probably loses precision in the projection phase. Moreover, the two drops of complexity perceived by the ATCO around 7:21 and 8:51 are clearly visible from the CLRQ and  $K$ -means mean complexities, and more so than from that of KRGP. The first of these two time periods corresponds to a decision to regroup, probably motivated by the simplification of the traffic.

Also, it is important to notice that the complexity perceived by the ATCO is not directly indexed on the number of aircraft in the considered airspace (shown on the last row), and neither are the clustering summaries. For example, using the Wasserstein distance we can notice that the summary of the 10<sup>th</sup> time slot (from 8:36 to 8:51) is slightly more similar to (at distance 1.49 of) the CLRQ summary of the last time slot (from 9:51 to 10:06), despite an increase of 42% in the number of aircraft, than to the summary of the following 11<sup>th</sup> time slot (8:51 to 9:06, at distance 1.57), which represents an increase of only 8%. This is already a significant improvement on the operational complexity index based on the number of aircraft.

## 6. Conclusion and future work

We have proposed a gradient descent type algorithm to find the best finite discrete approximation of a probability measure on a Riemannian manifold. This algorithm is adapted to large sets of data as it is online, and yields a clustering on top of a finite summary of the non-Euclidean data of interest. It is convergent when the manifold is complete and when its injectivity radius is uniformly bounded from below by a positive constant. We have used it to compute summaries of air traffic images in the form of a small number of covariance matrices representing different levels of local complexity, with associated weights corresponding to the occurrence of these levels of complexity in the images. These summaries can then be compared through discrete optimal transport, yielding a simple way to compare and index air traffic situations.

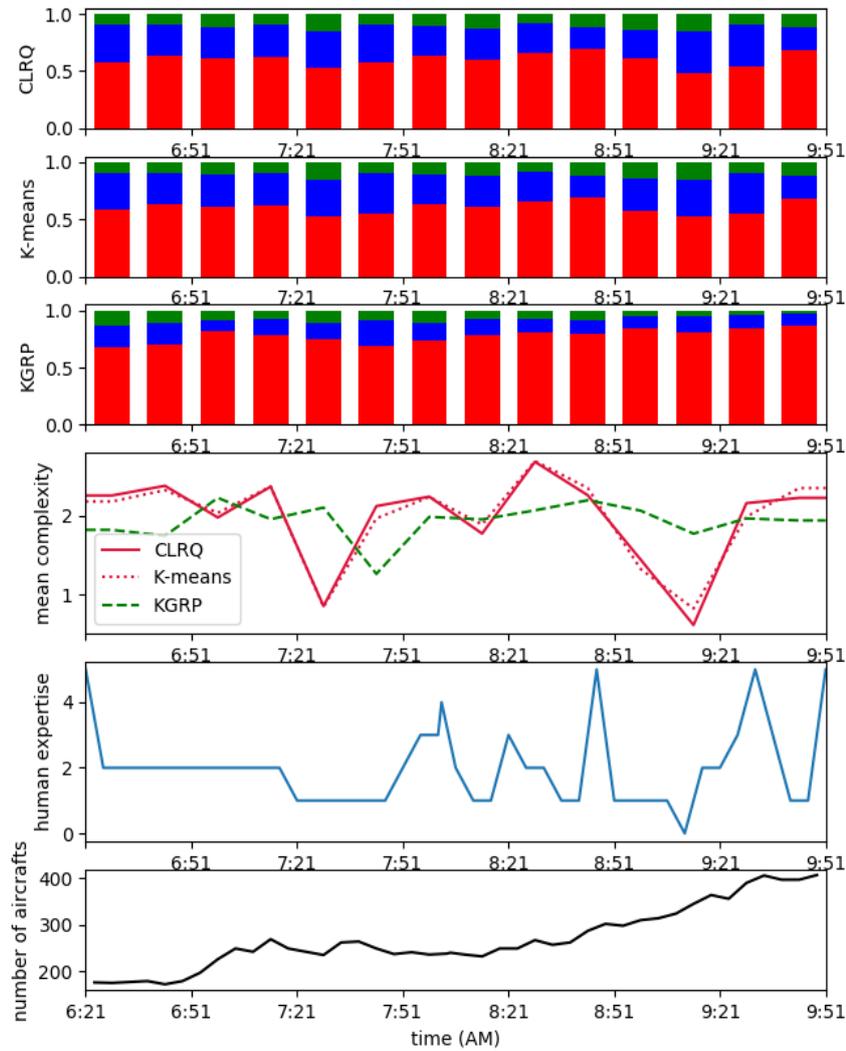
A still open question is to find an optimal number of centers for the quantization procedure. A possible approach is to use the associated clustering and to apply standard quality indicators like the silhouette [44]. In such a case, one starts with a minimal guess on the number of centers and increases it until the clustering is good enough.

In future work, we will also consider best finite constrained quantization, i.e., restriction to finite approximations with equal weights. Within this frame, one wants to find an optimal set of centers  $(a_1, \dots, a_n)$  such that the distribution:

$$\frac{1}{n} \sum_{i=1}^n \delta_{a_i}$$

best approximates the true sample density. In the air traffic complexity application, this may be used for finding an optimal segmentation of the airspace in areas of equal complexity, thus making the process of control sector design fully automatic. It may also be used in a dynamic way, since the quantization process presented above is an online procedure and will self-adapt to changing situations. Changes in the shape of control sectors must however be limited to allow ATCOs to seamlessly switch to new airspace partitions. This last point is currently under investigation in the context of manual or semi-automatic dynamic airspace design, so that our procedure may benefit from the results of this study.

Finally, only the covariance information was used in the application, but the mean function that yields the expected direction, and the local density bear some important information. Working on the product manifold  $\mathbb{R}^+ \times \mathbb{R}^d \times \mathcal{P}_d$  (with  $d = 2$  or  $d = 3$  for the air traffic framework) allows taking them into account. The gradient algorithm is readily adapted to this case, since geodesics on products are easily seen to be curves whose projections on factors are geodesics. A fully fledged complexity indicator may then be issued.



**Fig. 9.** Comparison of CLRQ to Riemannian  $K$ -means and Kernelized Gaussian Random projection (KGRP), and to the expertise of an ATCO. The first three rows show the weights  $w_1$  (green),  $w_2$  (blue), and  $w_3$  (red) of the summaries performed on the French Eastern control center for 15 time slots of 15 minutes each and for each different method. The fourth row shows the mean complexity indexes computed from these summaries to best fit the perceived complexity index of an ATCO shown on the fifth row. The last row shows the evolution of the number of aircrafts.

## Acknowledgements

We would like to thank Salem Said and Lionel Bombrun for sharing with us their code to generate Gaussian samples in the hyperbolic half-plane.

## Appendix

**Lemma 1.** Let  $x \in M$ . The gradient of the function  $f : a \mapsto d^2(x, a)$  is given by

$$\text{grad}_a f = -2 \log_a x = -2\vec{ax}.$$

**Proof.** Let  $a \in M$ ,  $u \in T_a M$ , and construct a curve  $(-\epsilon, \epsilon) \ni t \mapsto a(t)$  passing through  $a(0) = a$  at speed  $a'(0) = u$ , so that the differential of  $f$  at  $a$  in  $u$  is written, in terms of the norm associated to the Riemannian metric,

$$T_a f(u) = \left. \frac{d}{dt} \right|_{t=0} f\{a(t)\} = \left. \frac{d}{dt} \right|_{t=0} d^2(x, a(t)) = \left. \frac{d}{dt} \right|_{t=0} \|v(t)\|^2,$$

where  $v(t) := \overrightarrow{xa(t)} = \log_x a(t)$ . Consider the sheaf of geodesics

$$\gamma(t, s) := \exp_x\{sv(t)\}, \quad (t, s) \in (-\epsilon, \epsilon) \times [0, 1],$$

starting at  $\gamma(t, 0) = x$  and ending at  $\gamma(t, 1) = a(t)$ . The vector field  $J(s) := \partial\gamma/\partial t(0, s)$  then defines a Jacobi field along  $\gamma(s) := \gamma(0, s)$ , measuring the way the geodesics spread out from  $x$ , and its initial velocity is given by ([9], p.113)

$$\left. \frac{DJ}{ds} \right|(s) = \left. \frac{D}{ds} \left[ \left. \frac{\partial}{\partial t} \right|_{t=0} \exp_x\{sv(t)\} \right] \right| = \left. \frac{D}{ds} [d(\exp_x)_{sv(0)}\{sv'(0)\}] \right| = \left. \frac{D}{ds} [s d(\exp_x)_{sv(0)}\{v'(0)\}] \right|.$$

Identifying the tangent spaces  $T_x M \cong T_{v(0)}(T_x M)$ , we get

$$\left. \frac{DJ}{ds} \right|(0) = d(\exp_x)_0\{v'(0)\} + 0 \times \left. \frac{D}{ds} \right|_{s=0} d(\exp_x)_{sv(0)}\{v'(0)\} = v'(0),$$

and so,

$$T_a f(u) = 2 \langle v'(0), v(0) \rangle = 2 \left\langle \left. \frac{DJ}{ds} \right|(0), \gamma'(0) \right\rangle.$$

Since  $J$  verifies the classical Jacobi equation in terms of the curvature tensor  $\mathcal{R}$  of  $M$ ,  $D^2 J/ds^2 = -\mathcal{R}(J, \gamma')\gamma'$ , we get  $\langle D^2 J/ds^2, \gamma' \rangle = 0$ , leading to the scalar product  $\langle DJ/ds, \gamma' \rangle$  being constant (since  $D\gamma'/ds = 0$ ). This gives, since  $J(0) = 0$ ,

$$\langle J(s), \gamma'(s) \rangle = \left\langle \left. \frac{DJ}{ds} \right|(0), \gamma'(0) \right\rangle s + \langle J(0), \gamma'(0) \rangle = \left\langle \left. \frac{DJ}{ds} \right|(0), \gamma'(0) \right\rangle s,$$

yielding  $\langle DJ/ds(0), \gamma'(0) \rangle = \langle J(1), \gamma'(1) \rangle$ . To conclude, notice that  $J(1) = \partial\gamma/\partial t(0, 1) = u$  and, setting  $\tilde{\gamma}(s) = \gamma(1-s)$ , that  $\gamma'(1) = -\tilde{\gamma}'(0) = -\vec{ax}$  and so finally

$$T_a f(u) = -2\langle u, \vec{ax} \rangle,$$

which completes the proof.  $\square$

**Lemma 2.** Let  $\mu$  be a probability measure with density with respect to the Riemannian volume form on  $M$ ,  $\alpha = (a_1, \dots, a_n) \in M^n$  and  $C_i(\alpha)$  denote the Voronoi cell associated to  $a_i$  for all  $i \in \{1, \dots, n\}$ . No mass is assigned by  $\mu$  to the boundaries of the Voronoi diagram

$$\mu\{\partial C_i(\alpha)\} = 0.$$

**Proof.** For any  $n$ -tuple  $\alpha = (a_1, \dots, a_n)$ , the  $i^{\text{th}}$  Voronoi cell can be written

$$C_i(\alpha) = \cap_{j \neq i} H(a_i, a_j), \quad \text{where } H(a, b) = \{x \in K, d(x, a) \leq d(x, b)\},$$

which gives

$$\partial C_i(\alpha) = C_i(\alpha) \cap \overset{\circ}{C}_i(\alpha)^c = C_i(\alpha) \cap \{\cap_{j \neq i} \overset{\circ}{H}(a_i, a_j)\}^c = \cup_{j \neq i} \partial H(a_i, a_j) \cap C_i(\alpha).$$

Now for any  $i \neq j$ , the subset  $\partial H(a_i, a_j) = \{x \in K, d(x, a_i) = d(x, a_j)\}$  defined as the kernel of the submersion  $x \mapsto d(a_i, x) - d(a_j, x)$  is a submanifold of  $M$  of codimension 1, yielding  $\mu\{\partial H(a_i, a_j)\} = 0$  and therefore  $\mu\{\partial C_i(\alpha)\} = 0$ .  $\square$

## References

- [1] M. Arnaudon, F. Barbaresco, L. Yang, Riemannian medians and means with applications to radar signal processing, *IEEE Journal of Selected Topics in Signal Processing* 7 (2013), 595-604.
- [2] M. Arnaudon, F. Barbaresco, L. Yang, Medians and means in Riemannian geometry: existence, uniqueness and computation, in: *Matrix Information Geometry*, Springer, 2013, pp. 169-197.
- [3] G. Biau, L. Devroye, G. Lugosi, On the performance of clustering in Hilbert spaces, *IEEE Transactions on Information Theory* 54 (2008), 781-790.
- [4] J. Bigot, R. Gouet, T. Klein, A. López, Geodesic PCA in the Wasserstein space by Convex PCA, in: *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques* 53, 2017, pp. 1-26.
- [5] S. Bonnabel, Stochastic gradient descent on Riemannian manifolds, *IEEE Transactions on Automatic Control* 58 (2013), 2217-2229.
- [6] G. Bouchitté, C. Jimenez, R. Mahadevan, Asymptotic analysis of a class of optimal location problems, *Journal de mathématiques pures et appliquées* 95 (2011), 382-419.
- [7] C. A. Cabrelli, U. M. Molter, The Kantorovich metric for probability measures on the circle, *Journal of Computational and Applied Mathematics* 57 (1995), 345-361.
- [8] M. Calvo, J. M. Oller, An explicit solution of information geodesic equations for the multivariate normal model, *Statistics & Risk Modeling* 9 (1991), 119-138.
- [9] M. P. do Carmo, *Riemannian geometry*, Birkhäuser, 1992.
- [10] F. Chazal, L. J. Guibas, S. Y. Oudot, P. Skraba, Persistence-based clustering in riemannian manifolds, *Journal of the ACM* 60 (2013), 41.
- [11] J. Cheeger, D. G. Ebin, Comparison theorems in Riemannian geometry, *American Mathematical Soc.*, 2008.
- [12] P. T. Fletcher, C. Lu, S. M. Pizer, S. Joshi, Principal geodesic analysis for the study of nonlinear statistics of shape, *IEEE transactions on medical imaging* 23 (2004), 995-1005.
- [13] P. T. Fletcher, S. Joshi, Riemannian geometry for the statistical analysis of diffusion tensor data, *Signal Processing* 87 (2007), 250-262.
- [14] M. Fréchet, Les éléments aléatoires de nature quelconque dans un espace distancié, *Ann. Inst. H. Poincaré* 10 (1948), 215-310.
- [15] S. Graf, H. Luschgy, *Foundations of quantization for probability distributions*, Springer, 2007.
- [16] M. Iacobelli, Asymptotic quantization for probability measures on Riemannian manifolds, *ESAIM: Control, Optimisation and Calculus of Variations* 22 (2016), 770-785.
- [17] J. Jost, *Riemannian geometry and geometric analysis*, Springer, 2008.
- [18] H. Karcher, Riemannian center of mass and mollifier smoothing, *Communications on pure and applied mathematics* 30 (1977), 509-541.
- [19] D. G. Kendall, Shape manifolds, procrustean metrics, and complex projective spaces, *Bulletin of the London Mathematical Society* 16 (1984), 81-121.
- [20] B. Kloeckner, Approximation by finitely supported measures, *ESAIM: Control, Optimisation and Calculus of Variations* 18 (2012), 343-359.
- [21] A. Le Brigant, Computing distances and geodesics between manifold-valued curves in the SRV framework, *Journal of Geometric Mechanics* 9 (2017), 131-156.
- [22] M. Lovrić, M. Min-Oo, E. A. Ruh, Multivariate normal distributions parametrized as a Riemannian symmetric space, *Journal of Multivariate Analysis* 74 (2000), 36-48.
- [23] Q. Mérigot, E. Oudet, Discrete optimal transport: complexity, geometry and applications, *Discrete & Computational Geometry* 55 (2016), 263-283.
- [24] E. A. Nadaraya, On estimating regression, *Theory of Probability & Its Applications* 9 (1964), 141-142.
- [25] G. Pagès, H. Pham, J. Printems, Optimal quantization methods and applications to numerical problems in finance, in: *Handbook of computational and numerical methods in finance*, Springer, 2004, pp. 253-297.
- [26] G. Pages, Quadratic optimal functional quantization of stochastic processes and numerical applications, in: *Monte Carlo and Quasi-Monte Carlo Methods*, Springer, 2008, pp. 101-142.
- [27] G. Pagès, Introduction to vector quantization and its applications for numerics, *ESAIM: proceedings and surveys* 48 (2015), 29-79.
- [28] X. Pennec, P. Fillard, N. Ayache, A Riemannian framework for tensor computing, *International Journal of computer vision* 66 (2006), 41-66.
- [29] X. Pennec, Intrinsic statistics on Riemannian manifolds: Basic tools for geometric measurements, *Journal of Mathematical Imaging and Vision* 25 (2006), 127.
- [30] J. Rabin, J. Delon, Y. Gousseau, Transportation distances on the circle, *Journal of Mathematical Imaging and Vision* 41 (2011), 147.
- [31] S. Said, L. Bombrun, Y. Berthoumieu, J. H. Manton, Riemannian Gaussian distributions on the space of symmetric positive definite matrices, *IEEE Transactions on Information Theory* 63 (2017), 2153-2170.
- [32] S. Sommer, F. Lauze, S. Hauberg, M. Nielsen, Manifold valued statistics, exact principal geodesic analysis and the effect of linear approximations, in: *European conference on computer vision*, Springer, 2010, pp. 43-56.
- [33] R. Subbarao, P. Meer, Nonlinear mean shift over Riemannian manifolds, *International journal of computer vision* 84 (2009), 1.
- [34] P. Turaga, A. Veeraraghavan, A. Srivastava, R. Chellappa, Statistical computations on Grassmann and Stiefel manifolds for image and video-based recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), 2273-2286.
- [35] G. S. Watson, Smooth regression analysis, *Sankhyā: The Indian Journal of Statistics, Series A* (1964), 359-372.
- [36] J. Yin, Z. Geng, R. Li, H. Wang, Nonparametric covariance model, *Statistica Sinica* 20 (2010), 469.
- [37] K. Zhao, A. Alavi, A. Wiliem, B. C. Lovell, Efficient clustering on riemannian manifolds: A kernelised random projection approach, *Pattern Recognition* 51 (2016), 333-345.
- [38] I. V. Laudeman, S. G. Shelden, R. Branstrom, C. L. Brasil, *Dynamic Density: An Air Traffic Management Metric* (1998).
- [39] M. Prandini, L. Piroddi, S. Puechmorel, S. L. Brazdilova, Toward Air Traffic Complexity Assessment in New Generation Air Traffic Management Systems, *IEEE Transactions on Intelligent Transportation Systems* 12 (2011), 809-818.
- [40] A. Cook, H. Blom, F. Lillo, R. Mantegna, S. Miccichè, D. Rivas, R. Vázquez, M. Zanin, Applying complexity science to air traffic management, *Journal of Air Transport Management* 42 (2015), 149-158.

- [41] K. Lee, E. Feron, A. Prichett, Air Traffic Complexity : An Input-Output Approach, in: Proceedings of the US Europe ATM Seminar, 2007, pp. 2-9.
- [42] D. Delahaye, S. Puechmorel, Air traffic complexity based on dynamical systems, in: Proceedings of the 49th CDC conference, 2010.
- [43] G. Mykoniatis, F. Nicol, S. Puechmorel, A New Representation of Air Traffic Data Adapted to Complexity Assessment, in: Proceedings of ALLDATA2018, 2018, pp. 28-33.
- [44] P. J. Rousseeuw, Silhouettes: A graphical aid to the interpretation and validation of cluster analysis, Journal of Computational and Applied Mathematics 20 (1987), 53-65.