Information Geometry for Safety Data Processing

Florence Nicol, Stéphane Puechmorel

Information Geometry for Safety Data Analysis

Florence Nicol

Ecole Nationale de l'Aviation Civile
7, Avenue Edouard Belin,
F-31055 Toulouse FRANCE
Email: nicol@recherche.enac.fr

Stephane Puechmorel

Ecole Nationale de l'Aviation Civile
7, Avenue Edouard Belin,
F-31055 Toulouse FRANCE
Email: stephane.puechmorel@enac.fr

***Index Terms*—curve clustering, probability distribution estimation, functional statistics, safety data analysis.**

***Abstract*—Air traffic system generates huge amounts of data, most of them being only partly exploited. One important issue is the functional nature of the samples that consist mainly in curves. Analyzing this kind of highly structured data requires dedicated tools. Most of the time, the functions considered are expanded on a truncated Hilbert basis then usual multivariate statistics tools are applied on them. When additional constraints are put on the curves, like in applications related to air traffic where operational considerations are to be taken into account, this approach is no longer applicable. The present article surveys some recent results obtained by the authors using a new framework where curves are represented as regularized currents. The question of high dimensional data with intrinsic redundancy is also discussed and a possible extension of the already obtained results is sketched.**

I. INTRODUCTION

The air traffic system outputs huge volumes of data coming from surveillance systems, like Radar or ADS-B, and from monitoring and maintenance sources. Although very different in sampling rates and samples dimensions, they share in common the fact that almost all the time, data can be represented as functions.

As an example, clustering aircraft trajectories is a central problem in Air Traffic Management (ATM). It arises naturally when designing arrival and departure procedures and when some special constraints are to be considered: noise exposure, pollutants emission. For this application, one must first identify the major flows and then assess the statistical structure of the discrepancy of the real flight paths around them.

In the en-route phase, clustering is applied in order to extract mean flight paths, which are further used to optimally design the airspace elements (sectors and airways). A special instance of this problem is the automatic generation of safe and efficient trajectories, but in such a way that the resulting flight paths are still manageable by human operators. Several algorithms can automatically produce conflict free trajectories but generally fail to issue planning schemes that may be used by air traffic controller in the event of a system failure. Clustering may be a solution to this problem using a two phase procedure: in a first step, a set of flight plans or intended reference business trajectories is submitted to the optimizer which outputs a conflict free planning. In a second step, a clustering algorithms extracts medial lines from this trajectories set, which form a dynamic route system. If it is simple enough, a controller may use it in a manual control context.

Finally, a very important applicative field of trajectory analysis is related to safety data. This subject covers in fact many different applications, depending on the data source. When only radar tracks are used, only a small subset of the aircraft states can be inferred. However, it is still possible to get extract valuable information: runway adherence condition is a worked example of that. On-board data logging facilities give access to an almost complete picture of the aircraft state: the main issue when trying to analyze it is the volume and the internal redundancy of the samples. Tools dealing with such a situation are yet to be designed. A work in progress tries to pave the way to future systems, where almost independent features will be identified.

In all the above situations, a common denominator is the need for algorithms that take into account the functional nature of the samples. After a brief description of previous related works in general statistics and on the targeted air traffic application, a new framework based on densities associated with curve systems will be introduced. It relies on currents that can be seen as an extension of the delta distribution. An application to major flows identification will be presented. In a second part, a lie group modeling will allow to take into account flows orientations and velocities. Finally, this work will be further extended to flag manifolds, that may be viewed as an abstract model for independent subspaces analysis. While still in early stage of research, it represents the first attempt to treat the issue of functional data with values in a structured, high dimensional state space.

II. Previous related work

The field of functional data is quite recent and active. The results collected in [1], [2] give a picture of the multivariate algorithms that have been transposed to the functional framework.

The most obvious way of dealing with such data is simply to sample at evenly spaced times and to collect the values obtained in a finite dimensional vector. This procedure was used in a study conducted by the Mitre corporation on behalf of the Federal Aviation Authority (FAA) [3]. Due to the very high dimension of the vectors considered, clever numerical workarounds must be found. In the Mitre study, random projections reduce the dimensionality, then a classical PCA is performed. The huge computational cost of the required singular values decomposition is thus alleviated, allowing use on real recorded traffic over several months.

The most important limitation of the plain sampling approach is that the shape of the trajectories is no taken into account when applying the clustering procedure. Furthermore, there is no simple mean to put a constraint on the mean trajectory produced in each cluster: curvature may be quite arbitrary even if samples individually comply with flight dynamics.

Another approach is taken in [4], where an underlying graph structure is assumed. It is a variation of the original work described in [5]. It is well adapted to road traffic as vehicles are bound to follow predetermined road segments. For air traffic applications, it may be of interest for investigating present situations, using the airways and beacons as a structure graph, but will mis-classify aircraft following direct routes which is a quite common situation, and is unable to work on an unknown airspace organization. This last limitation prevents its use for airspace redesign applications or in the context of trajectory based operations. Varying the similarity measure will put the emphasis on different aspects of the trajectories [6],without removing the aforementioned drawbacks

An interesting vector field based algorithm is presented in [7]. An salient feature is the ability to distinguish between close trajectories with opposite orientations. It falls within the frame of landmarks based algorithms, first introduced for shape analysis by Kendall [8].

Due to the functional nature of trajectories, that are basically mappings defined on a time interval, some authors resort to techniques based on times series, as surveyed in [9], [10]. The algorithms pertaining to this category are based on sequences, possibly in conjunction with dynamic time warping [11].

Finally, as mentioned at the beginning of the section, one can assume that data come from an unknown underlying stochastic process whose sample paths belong to a given Hilbert space with countable basis. Using truncated expansions of this basis, functional data revert for implementation to usual multivariate algorithms. For the same reason, model-based clustering may be used in the context of functional data even if no notion of probability density exists in the original infinite dimensional Hilbert space [12]. A nice example of a model-based approach working on functional data is funHDDC [13].

III. Curve systems

When dealing with aircraft trajectories, some specific characteristics must be taken into account. First of all, flight paths consist mainly in straight segments connected by arcs of circles, with transitions that may be assumed smooth up to at least the second derivative. This last property comes from the fact that pilot's actions result in changes on aerodynamic forces and torques and a straightforward application of the equations of motion. When dealing with sampled trajectories, this induces a huge level of redundancy within the data, the relevant information being concentrated around the transitions. It is well known from numerical analysis that it renders covariance and related matrices ill-conditioned, inducing instabilities in the algorithms. From a practical standpoint, random pre-conditioners will greatly improve the situation, but render the interpretation of the final results difficult. At the same time, it is highly desirable that trajectories produced by algorithms be of smallest possible curvature, so that they may represent potential flight paths.

A second issue comes from the fact that trajectories of interest are defined as mapping from a time interval $[a, b]$ to $\mathbb{R}^3$ which is not the usual setting for functional data statistics: most of the work is dedicated to real valued mappings and not to vector ones. A simple approach will be to assume independence between coordinates, so that the problem falls within the standard case. However, even with this simplifying hypothesis, vertical dimension must be treated in a special way as both the separation norms and the aircraft maneuverability are different from those in the horizontal plane. This issues becomes even more important when dealing with on-board recorded flight data as the dimension of the state space may be in the order of several hundreds, with a strong correlation between coordinates.

Within the frame of functional data statistics, a clever choice of the hilbert state space may yield a solution to both problems. As a matter of fact, one can consider Sobolev spaces of the form:

$$\mathcal{H}_L = \{f \mid \int \|Lf(x)\|^2 dx < +\infty\} \tag{1}$$

where L is a differential operator with 0 kernel. Taking for example $L = Id + AD_{xx}$, with $A > 0$ a tuning parameter, allows the acceleration of the curve to be considered. Suitable hilbert basis in such a case can be obtained using green functions of the L^*L operator, with L^* the adjoint of L, yielding the so-called spline basis [14]. Such Sobolev spaces are reproducing kernel hilbert spaces (RKHS), with kernel the green function, which allows computation saves when evaluating inner products or distances. This relationship is exposed in detail in the above reference. The spline basis has been proved asymptotically optimal under special choices of L, fully justifying its use when working on truncated expansion.

When no obvious choice of L can be made or when the green function cannot be computed, the method is less straightforward to apply. Furthermore, the nice optimality property is generally lost when working with general basis: the truncation threshold may become so high that the method will no compete with plain sampling. Finally, geometric features of curves, like curvature, cannot be easily formulated as a Sobolev norm, and may not be preserved in the algorithms outputs.

A different approach is considered in this work, building upon geometry of shapes [15] and information geometry [16]. Curves will be modeled as smooth currents [17], then regularized using a kernel functions. Given a complete set of trajectories, it will give rise to a density, that may enter an optimization process. As one of the most important requirement about flight paths is that they must be as straight as possible, an entropy criterion was selected. In the next section, the contributions coming from [16], [18], [19] are summarized.

IV. THE ENTROPY OF A SYSTEM OF CURVES

Let a set $\gamma_1, \dots \gamma_N$ of smooth curves with value in a bounded open subset Ω of finite dimensional vector space E be given. If $U_k, k = 1 \dots P$ is a partition of Ω, then one can construct a density estimator for the curve system as:

$$d_k = \lambda^{-1} \sum_{i=1}^{N} \int_0^1 1_{U_k}(\gamma_i(t))\, dt \tag{2}$$

where the normalizing constant λ is obtained as:

$$\lambda = \sum_{k=1}^{P} \sum_{i=1}^{N} \int_0^1 1_{U_k}(\gamma_i(t))\, dt \tag{3}$$

$$= \sum_{i=1}^{N} \int_0^1 \sum_{k=1}^{P} 1_{U_k}(\gamma_i(t))\, dt \tag{4}$$

and since $U_k, k = 1 \dots P$ is a partition:

$$\lambda = \sum_{i=1}^{N} \int_0^1 dt = N \tag{5}$$

Density can be viewed as an empirical probability distribution with the U_k considered as bins in an histogram. Proceeding the same way as in non-parametric density estimation [20], a kernel smoother is used in place of the characteristic functions 1_{U_k}:

$$d \colon x \mapsto \frac{\sum_{i=1}^{N} \int_0^1 K(\|x - \gamma_i(t)\|)\, dt}{\sum_{i=1}^{N} \int_\Omega \int_0^1 K(\|x - \gamma_i(t)\|)\, dt dx} \tag{6}$$

it comes:

$$\int_\Omega K(\|x - \gamma_i(t)\|)\, dx = \int_{\mathbb{R}^2} K(\|x\|)\, dx$$

provided that Ω contains the set:

$$\{x \in \mathbb{R}^2, \inf_{i=1\dots N, t\in[0,1]} \|x - \gamma_i(t)\| \le A\}$$

where the interval $[-A, A]$ contains the support of K. When the kernel has unit integral, the expression of the density simplifies to:

$$d \colon x \mapsto N^{-1} \sum_{i=1}^{N} \int_0^1 K(\|x - \gamma_i(t)\|)\, dt \tag{7}$$

A common choice for K is the Epanechnikov function:

$$K \colon x \mapsto \left(1 - x^2\right) 1_{[-1,1]}(x)$$

that is presented here in its un-normalized version. The density map of one day of traffic over France is considered and pictured on figure 2: Unfortunately, this simple procedure suffers a severe flaw: it tends to overemphasis areas when the aircraft are slow, as they will represent a longer time occupancy for the same size. A change is made so as to overcome this issue, yielding:

$$\tilde{d} \colon x \mapsto \frac{\sum_{i=1}^{N} \int_0^1 K(\|x - \gamma_i(t)\|)\, \|\gamma_i'(t)\| dt}{\sum_{i=1}^{N} l_i} \tag{8}$$

with l_i the length of curve i. A unit integral kernel is assumed in the above expression. The resulting density map is more satisfying, as shown in figure 3: It turns out that the expression can be interpreted as a sum of regularized

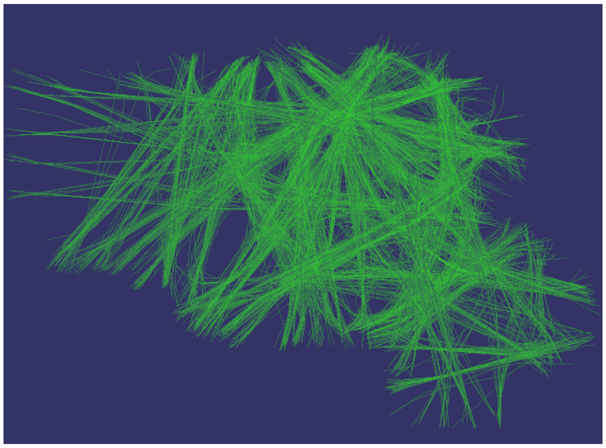

Fig. 1. Traffic over France the 12th February 2013

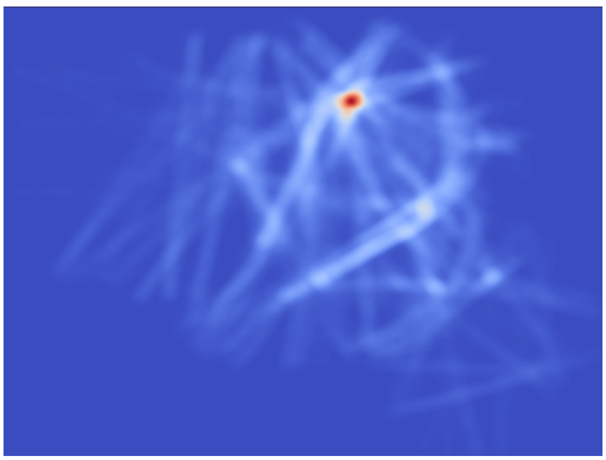

Fig. 2. Associated density

smooth currents [17], with the curves γ_i considered as 0-currents and K the smoothing kernel. Other choices of K may be more natural in this context: Gaussian kernels will be a good starting point, but are not compact (this is not in issue in real-world applications as the truncation error is generally negligible). Varying the bandwidth of K will change the amount of smoothing applied to the original curves: very low ones tend to reproduce accurately the trajectories while larger ones will spread the density over Ω. All the theoretical properties of smooth currents apply verbatim to the density of a curve system. It will be used in a future work to compute limiting objects.

The density defined that way is obviously positive and of unit integral: its entropy is thus well defined and can be computed as:

$$E(\gamma_1, \ldots, \gamma_N) = -\int_\Omega \tilde{d}(x) \log\left(\tilde{d}(x)\right) dx \qquad (9)$$

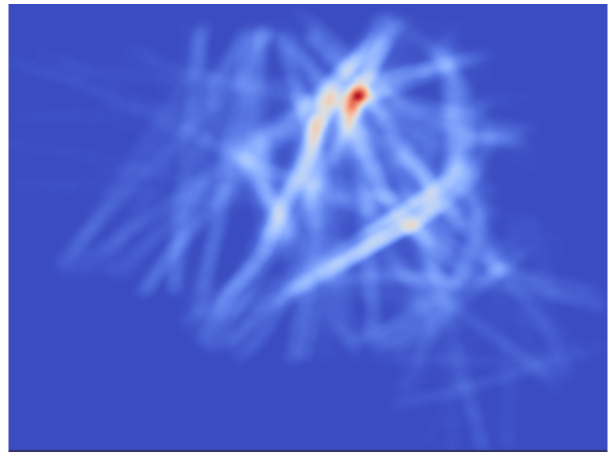

Fig. 3. Geometric density for the 12th February 2013

Since the entropy is minimal for concentrated distributions, it is quite intuitive to figure out that seeking for a curve system $(\gamma_1, \ldots, \gamma_N)$ giving a minimum value for $E(\gamma_1, \ldots, \gamma_N)$ will induce the following properties:

- The images of the curves tend to get close on to another.
- The individual lengths will be minimized: it is a direct consequence of the fact that the density has a term in γ' within the integral that will favor short trajectories.

Using a standard gradient descent algorithm on the entropy produces an optimally concentrated curve system, suitable for use in an unsupervised clustering algorithm. The displacement field for trajectory j is oriented at each point along the normal vector to the trajectory, with norm given by:

$$\int_\Omega \left.\frac{\gamma_j(t) - x}{\|\gamma_j(t) - x\|}\right|_{\mathcal{N}} K'\left(\|\gamma_j(t) - x\|\right) \log \tilde{d}(x) dx \|\gamma_j'(t)\| \qquad (10)$$

$$-\left(\int_\Omega K\left(\|\gamma_j(t) - x\|\right) \log \tilde{d}(x)) dx\right) \left.\frac{\gamma_j''(t)}{\|\gamma_j'(t)\|}\right|_{\mathcal{N}} \qquad (11)$$

$$+\left(\int_\Omega \tilde{d}(x) \log(\tilde{d}(x)) dx\right) \left.\frac{\gamma_j''(t)}{\|\gamma_j'(t)\|}\right|_{\mathcal{N}}, \qquad (12)$$

where the notation $v_{|\mathcal{N}}$ stands for the projection of the vector v onto the normal vector to the trajectory. An overall normalizing constant:

$$\frac{1}{\sum_{i=1}^N l_i},$$

has to put in front of the expression to get the true gradient of the entropy.

Applying the gradient based algorithm to the traffic situation studied above, yields after convergence the picture presented in figure 4.

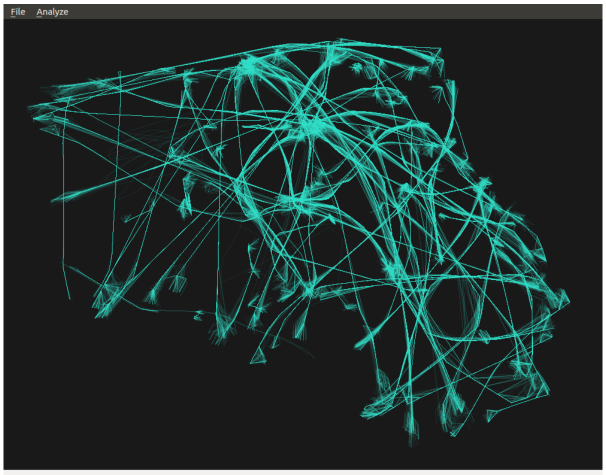

Fig. 4. Bundling one day of traffic

The clustering procedure was also used in the context of automated trajectory planners. In the example presented in figure 5, 30 flights were simulated on the depicted route system (the figure is a snapshot of the situation, so that not all aircraft are present). A multi-agent system was applied in order to solve all conflicts, but with no constraints put on trajectories: the resulting planning scheme makes each aircraft move in a free flight fashion, and may not be controlled by a human ATC controller. Applying the minimum entropy clustering scheme allows the emergence of a route system. Since conflicts are introduced by this procedure, an iterative application of the multi-agent solver and clustering must be used. After convergence, the conflict-free planning of figure 6 was obtained.

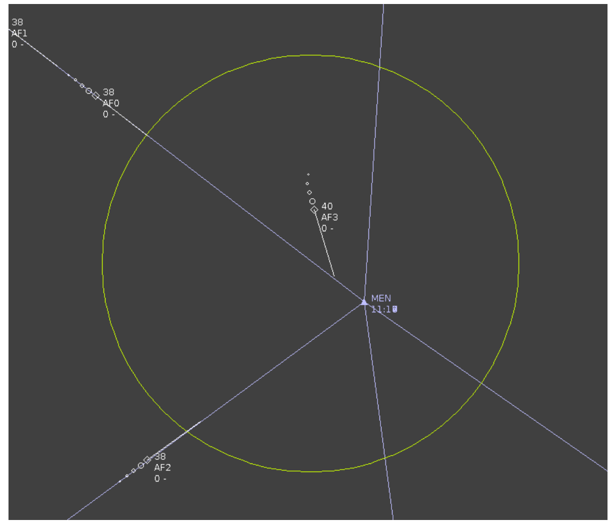

Fig. 5. Initial flight plan.

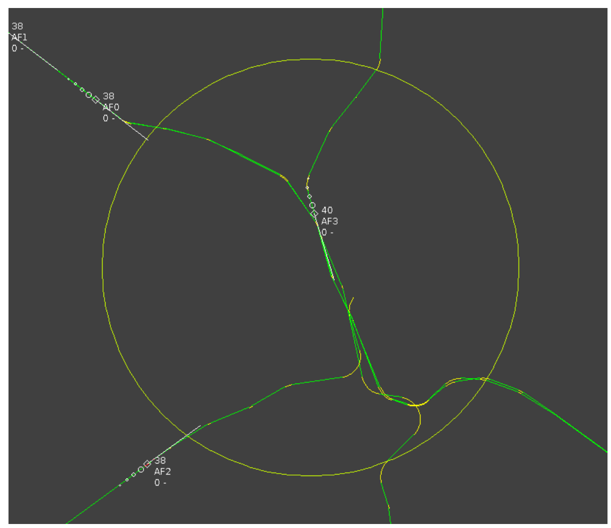

Fig. 6. Entropy minimal curve system from the initial flight plan.

V. Extension to trajectories in a Lie group

While satisfactory in terms of traffic flows, the previous approach suffers from a severe flaw when one considers flight paths that are very similar in shape but are oriented in opposite directions. Since the density is insensitive to direction reversal, flight paths will tend to aggregate while the correct behavior will be to ensure a sufficient separation in order to prevent hazardous encounters. Taking aircraft headings into account in the clustering process is then mandatory when such situations have to be considered.

Solving this issue is not as straightforward as one may think at a first glance. It is always possible to add a penalty term related to the inner product of speed vectors for each pair of curves, but tuning its relative importance is quite difficult and may impair the overall convergence of the algorithm. A classical workaround that several authors have used in usual multivariate PCA is to enrich the state space of the curves. One can consider for example a state vector of the form:

$$\begin{pmatrix} \gamma(t) \\ \gamma'(t) \end{pmatrix}. \tag{13}$$

Instead of using the speed vector to enrich the state, successive positions may be used:

$$\begin{pmatrix} \gamma(t) \\ \gamma(t+\tau) \end{pmatrix}. \tag{14}$$

where the time shift τ has to be chosen to capture instantaneous motion of the aircraft. While satisfactory, two points are not addressed by this formulation:

- Just comparing speed vectors (or couples) using standard euclidean distance is not scale invariant: if velocities are multiplied by a fixed positive scalar, then the inner product will scale with the square of this amount. This is not relevant in most applications as speed vectors comparison is generally assumed not to change under translations/rotations/scalings.
- Salient features of the speed vector in an operational context are magnitude and heading. There are not treated as independent parameters in the aforementioned representation.

Keeping the operational requirements in mind, it is more natural to describe the aircraft state as a centroid position and attitude. Since only observed trajectories are considered here, attitude will be described by a unit vector pointing in the direction of the motion and a positive real number giving the velocity. From a mathematical point of view, such a situation may be described by a group action on a reference vector, chosen here to be:

$$\left(\frac{0_d}{e_1}\right),$$

with e_1 the first basis vector, and 0_d the origin in $\mathbb{R}^d$. It is equivalent to model the state in 13 as a linear transformation:

$$0_d \otimes e_1 \mapsto T(t) \otimes A(t)(0_d \otimes e_1) = \gamma(t) \otimes \gamma'(t),$$

where $T(t)$ is the translation mapping 0_d to $\gamma(t)$ and $A(t)$ is the composite of a scaling and a rotation mapping e_1 to $\gamma'(t)$. Considering the vector $(\gamma(t), 1)$ instead of $\gamma(t)$ allows a matrix representation of the translation $T(t)$:

$$\left(\frac{\gamma(t)}{1}\right) = \left(\begin{array}{c|c} Id & \gamma(t) \\ \hline 0 & 1 \end{array}\right)\left(\frac{0_d}{1}\right).$$

From now, all points will be implicitly considered as having an extra last coordinate with value 1, so that translations are expressed using matrices. The origin 0_d will thus stand for the vector $(0, \ldots, 0, 1)$ in $\mathbb{R}^{d+1}$. Gathering things yields:

$$\left(\frac{\gamma(t)}{\gamma'(t)}\right) = \left(\begin{array}{c|c} T(t) & 0 \\ \hline 0 & A(t) \end{array}\right)\left(\frac{0_d}{e_1}\right). \quad (15)$$

The previous expression makes it possible to represent a trajectory as a mapping from a time interval to the matrix Lie group $\mathcal{G} = \mathbb{R}^d \times \Sigma \times S\mathbb{O}(d)$, where Σ is the group of multiples of the identity, $S\mathbb{O}(d)$ the group of rotations and $\mathbb{R}^d$ the group of translations. Please note that all the products are direct. The $A(t)$ term in the expression (15) can be written as an element of $\Sigma \otimes S\mathbb{O}(d)$. Starting with the defining property $A(t)e_1 = \gamma'(t)$, one can write $A(t) = \|\gamma'(t)\| U(t)$ with $U(t)$ a rotation mapping $e_1 \in \mathbb{S}^{d-1}$ to the unit vector $\gamma'(t)/\|\gamma'(t)\| \in \mathbb{S}^{d-1}$. For arbitrary dimension d, $U(t)$ is not uniquely defined, as it can be written as a rotation in the plane $\mathcal{P} = \text{span}(e_1, \gamma'(t))$ and a rotation in its orthogonal complement $\mathcal{P}^\perp$. A common choice is to let $U(t)$ be the identity in $\mathcal{P}^\perp$ which corresponds in fact to a move along a geodesic (great circle) in $\mathbb{S}^{d-1}$. This will be assumed implicitly in the sequel, so that the representation $A(t) = \Lambda(t)U(t)$ with $\Lambda(t) = \|\gamma'(t)\|\text{Id}$ becomes unique.

The Lie algebra $\mathfrak{g}$ of $\mathcal{G}$ is easily seen to be $\mathbb{R}^d \times \mathbb{R} \times \mathbf{Asym}(d)$ with $\mathbf{Asym}(d)$ is the space of skew-symmetric $d \times d$ matrices. An element from $\mathfrak{g}$ is a triple (u, λ, A) with an associated matrix form:

$$M(u, \lambda, A) = \left(\begin{array}{c|c} \begin{array}{c|c} 0 & u \\ \hline 0 & 0 \end{array} & 0 \\ \hline 0 & \lambda Id + A \end{array}\right). \quad (16)$$

The exponential mapping from $\mathfrak{g}$ to $\mathcal{G}$ can be obtained in a straightforward manner using the usual matrix exponential:

$$\exp((u, \lambda, A)) = \exp(M(u, \lambda, A)).$$

The matrix representation of $\mathfrak{g}$ may be used to derive a metric:

$$\langle (u, \lambda, A), (v, \mu, B) \rangle_{\mathfrak{g}} = \mathbf{Tr}\left(M(u, \lambda, A)^t M(v, \mu, B)\right).$$

Using routine matrix computations and the fact that A, B being skew-symetric have vanishing trace, it can be expressed as:

$$\langle (u, \lambda, A), (v, \mu, B) \rangle_{\mathfrak{g}} = n\lambda\mu + \langle u, v \rangle + \mathbf{Tr}\left(A^t B\right). \quad (17)$$

A left invariant metric on the tangent space $T_g\mathcal{G}$ at $g \in \mathcal{G}$ is derived from (17) as:

$$\langle\!\langle X, Y, \rangle\!\rangle_g = \langle g^{-1}X, g^{-1}Y \rangle_{\mathfrak{g}},$$

with $X, Y \in T_g\mathcal{G}$. Please note that $\mathcal{G}$ is a matrix group acting linearly so that the mapping g^{-1} is well defined from $T_g\mathcal{G}$ to $\mathfrak{g}$. Using the fact that the metric (17) splits, one can check that geodesics in the group are given by straight segments in $\mathfrak{g}$: if g_1, g_2 are two elements from $\mathcal{G}$, then the geodesic connecting them is:

$$t \in [0, 1] \mapsto g_1 \exp\left(t \log\left(g_1^{-1} g_2\right)\right).$$

where $\log$ is a determination of the matrix logarithm. Finally, the geodesic length is used to compute the distance $d(g_1, g_2)$ between two elements g_1, g_2 in $\mathcal{G}$. Assuming that the translation parts of g_1, g_2 are respectively u_1, u_2, the rotations U_1, U_2 and the scalings $\exp(\lambda_1), \exp(\lambda_2)$ then:

$$d(g_1, g_2)^2 = (\lambda_1 - \lambda_2)^2 + \quad (18)$$

$$\mathbf{Tr}\left(\log\left(U_1^t U_2\right) \log\left(U_1^t U_2\right)^t\right) + \|u_1 - u_2\|^2. \quad (19)$$

An important point to note is that the scaling part of an element $g \in \mathcal{G}$ will contribute by its logarithm: this ensures the scale invariance of the metric as any common scaling factor will cancel out in the distance computation. The part related to unit matrices can be interpreted as finding an optimal sequence of rotations that will make the eigenvectors coincident and compute the total angular variation needed. It has to be noted that when dealing with planar curve, the value obtained that way is exactly the angle between the speed vectors. Finally, the metric differs from the affine invariant one in that orientation is explicitly taken into account.

Based on the above derivation, a flight path γ with state vector $(\gamma(t), \gamma'(t))$ will be modeled in the sequel as a curve with values in the Lie group $\mathcal{G}$:

$$\Gamma : t \in [0, 1] \mapsto \Gamma(t) \in \mathcal{G},$$

with:

$$\Gamma(t).(0_d, e_1) = (\gamma(t), \gamma'(t)).$$

In order to make the Lie group representation amenable to statistical thinking, it is needed to define probability densities on the translation, scaling and rotation components that are invariant under the action of the corresponding factor of $\mathcal{G}$.

VI. Density estimation on $\mathcal{G}$

The kernel density estimator presented in the first part of the paper cannot be used directly as it relies explicitly on a vector space structure. It has to be adapted for the Lie group case, so as to ensure invariance by the group action.

Since the Lie group $\mathcal{G}$ is a direct product, the three components can be treated separately.

The translation part is the additive group $\mathbb{R}^d$ and its action is given by a vector translation. As a consequence, its group structure is identical to the one of $\mathbb{R}^d$ considered as a real vector space: the density estimation on it will not differ from the one presented in IV. The same estimator will be used, with either the epanechnihov or the Gaussian kernel.

Concerning the $S\mathbb{O}(d)$ component, it well known that an arbitrary rotation U can be represented as a sequence of points on spheres of decreasing dimension: the procedure was already detailed in the section V. Estimating a density on $S\mathbb{O}(d)$ can thus be done using kernels on spheres: it becomes a classical problem in directional statistics.

A commonly used choice is the von Mises-Fisher (vMF) distribution on $\mathbb{S}^{d-1}$ which is denoted $\mathcal{M}(m, \kappa)$ and given by the following density expression [21]:

$$K_{VMF}(x; m, \kappa) = c_d(\kappa)\, e^{\kappa m^T x},\ \kappa > 0,\ x \in \mathbb{S}^{d-1}, \quad (20)$$

where

$$c_d(\kappa) = \frac{\kappa^{d/2-1}}{(2\pi)^{d/2} I_{d/2-1}(\kappa)} \quad (21)$$

is a normalization constant with $I_r(\kappa)$ denoting the modified Bessel function of the first kind at order r. The vMF kernel function is an unimodal p.d.f. parametrized by the unit mean-direction vector μ and the concentration parameter κ . κ is a smoothing parameter that plays the role of the inverse of the bandwidth parameter as defined in the linear kernel density estimation. Large values of κ imply greater a concentration around the mean direction and lead to undersmoothed estimators whereas small values provide oversmoothed circular densities [22]. Indeed, if $\kappa = 0$, the vMF kernel function reduces to the uniform circular distribution on the hypersphere. Note that the vMF kernel function is by design invariant by the action of $S\mathbb{O}^d$. The VNf density enjoys properties analogous to those of multivariate Gaussian distribution: a central limit theorem exists in the context of spheres, with limiting distribution the VMf.

The vMF distribution may be expressed by means of the spherical polar coordinates of $x \in \mathbb{S}^{d-1}$ [23].

Given the random vectors X_i, $i = 1, \ldots, n$, in $\mathbb{S}^{d-1}$, the estimator of the spherical distribution is given by:

$$\begin{aligned}\widehat{f}(x) &= \frac{1}{n} \sum_{i=1}^{n} K_{VMF}(x; X_i) \\ &= \frac{c_d(\kappa)}{n} \sum_{i=1}^{n} e^{\kappa X_i^T x},\ \kappa > 0,\ x \in \mathbb{S}^{d-1}.\end{aligned}$$

The density estimator on $S\not{V}^d$ is constructed component-wise, using the previous expression on each of the spheres involved in its decomposition.

As for the scaling component of $\mathcal{G}$, the usual kernel functions such as the Gaussian and the Epanechnikov kernel functions are not suitable due to a bias introduced in close to the domain boundary at 0. An asymmetrical kernel function on $\mathbb{R}^+$ such as the log-normal kernel function is a more convenient choice. Moreover, this p.d.f. is invariant by the action of the multiplicative group of positive real numbers. Let $R_1, \ldots, R_n$ be univariate random variables from a p.d.f. which has bounded support on $[0; +\infty[$. The radial density estimator may be defined by means of a sum of log-normal kernel functions as follows:

$$\widehat{g}(r) = \frac{1}{n} \sum_{i=1}^{n} K_{LN}(r; \ln R_i, h), r \geq 0, h > 0,$$

where

$$K_{LN}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma x} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}$$

is the log-normal kernel function and h is the bandwidth parameter. The resulting estimate is the sum of bumps defined by log-normal kernels with medians R_i and variances $(e^{h^2}-1)e^{h^2}R_i^2$. Note that the log-normal (asymmetric) kernel density estimation is similar to the kernel density estimation based on a log-transformation of the data with the Gaussian kernel function: in practice, the estimator IV is used on the logarithms of the scale components.

VII. Unsupervised entropy clustering

The derivation of the algorithm used for clustering curves with value in the Lie group is similar to the vector space case (10). The kernel has to be replaced with the respective VMf and Log-normal ones on the rotation and scaling components. The term $\|\gamma'(t)\|$ in the original expression of the density, that is required to ensure invariance under reparametrization of the curve, has to be changed according to the metric in $\mathcal{G}$ and is replaced by $\langle\!\langle \gamma'(t), \gamma'(t)\rangle\!\rangle_{\gamma(t)}^{1/2}$. The density at $x \in \mathcal{G}$ is thus:

$$d_{\mathcal{G}}(x)) = \frac{\sum_{i=1}^{N} \int_0^1 K\left(x, \gamma_i(t)\right) \langle\!\langle \gamma_i'(t), \gamma_i'(t)\rangle\!\rangle_{\gamma_i(t)}^{1/2} dt}{\sum_{i=1}^{N} l_i} \quad (22)$$

where l_i is the length of the curve in $\mathcal{G}$, that is:

$$l_i = \int_0^1 \langle\!\langle \gamma_i'(t), \gamma_i'(t)\rangle\!\rangle_{\gamma_i(t)}^{1/2} dt \quad (23)$$

The expression of the kernel evaluation $K\left(x, \gamma_i(t)\right)$ is split into three terms. In order to ease the writing, a point x in $\mathcal{G}$ will be split into x^r, x^s, x^o components where the exponent r, s, t stands respectively for translation, scaling and rotation. Given the fact that K is a product of component-wise independent kernels it comes:

$$K\left(x, \gamma_i(t)\right) = K_t\left(x^t, \gamma_i^t(t)\right) K_s\left(x^s, \gamma_i^s(t)\right) K_o\left(x^o, \gamma_i^o(t)\right)$$

where:

$$K_t(x^t, \gamma_i^t(t)) = \text{ep}\left(\|x^t - \gamma_i^t(t)\|\right) \quad (24)$$

$$K_s(x^s, \gamma_i^s(t)) = \frac{1}{x^s \sigma \sqrt{2\pi}} \exp\left(-\frac{\left(\log x^s - \log \gamma_i^s(t)\right)^2}{2\sigma^2}\right) \quad (25)$$

$$K_o(x^o, \gamma_i^o(t)) = C(\kappa) \exp\left(\kappa \mathbf{Tr}\left(x^{ot} \gamma_i^o(t)\right)\right) \quad (26)$$

with $C(\kappa)$ the normalizing constant making the kernel of unit integral. Please note that the expression given here is valid for arbitrary rotations, but for aircraft trajectory the heading is the only observable parameter unless on-board data is downlinked. As a consequence it is enough to consider it only a standard von-mises distribution on $\mathbb{S}^{d-1}$:

$$K_o(x^o, \gamma_i^o(t)) = C(\kappa) \exp\left(\kappa x^{ot} \gamma_i^o(t)\right)$$

with normalizing constant as given in (21). The entropy of the system of curves is obtained from the density in $\mathcal{G}$:

$$E(d_{\mathcal{G}}) = -\int_{\mathcal{G}} d_{\mathcal{G}}(x) \log d_{\mathcal{G}}(x) d\mu_{\mathcal{G}}(x) \quad (27)$$

with $d\mu_{\mathcal{G}}$ the left Haar measure. Using again the fact that $\mathcal{G}$ is a direct product group, $d\mu$ is easily seen to be a product measure, with dx^t, the usual Lebesgue measure on the translation part, dx^s/x^s on the scaling part and the lebesgue measure dx^o on $\mathbb{S}^{d-1}$ for the rotation part. It turns out that the $1/x^s$ term in the expression of dx^s/x^s is already taken into account in the kernel definition, due to the fact that it is expressed in logarithmic coordinates. The same is true for the Von-Mises kernel, so that in the sequel only the (product) lebesgue measure will appear in the integrals.

Finding the system of curves with minimum entropy requires a displacement field computation as detailed in [16]. For each curve γ_i, such a field is a mapping $\eta_i\colon [0,1] \to T\mathcal{G}$ where at each $t \in [0,1]$, $\eta_i(t) \in T\mathcal{G}_{\gamma_i(t)}$.Compare to the original situation where only spatial density was considered, the computation must now be conducted in the tangent space to $\mathcal{G}$. Even for small problems, the effort needed becomes prohibitive. The structure of the kernel involved in the density can help in cutting the overall computations needed. Since it is a product, and the translation part is compactly supported, being an epanechnikov kernel, one can restrict the evaluation to points belonging to its support. Density computation will thus be made only in tubes around the trajectories.

Second, for the target application that is to cluster the flight paths into a route network and is of pure spatial nature, there is no point in updating the rotation and scaling part when performing the moves: only the translation part must change, the other two being computed from the trajectory. The initial optimization problem in $\mathcal{G}$ may thus be greatly simplified.

Let ϵ be an admissible variation of curve γ_i, that is a smooth mapping from $[0,1]$ to $T\mathcal{G}$ with $\epsilon(t) \in T_{\gamma_i(t)}\mathcal{G}$ and $\epsilon(0) = \epsilon(1) = 0$. We assume furthermore that ϵ has only a translation component. The derivative of the entropy $E(d_{\mathcal{G}})$ the t curve γ_i is obtained from the first order term when γ_i is replaced by $\gamma_i + \epsilon$. First of all, it has to be noted that $d_{\mathcal{G}}$ is a density and thus has unit integral regardless of the curve system. When computing the derivative of $E(d_{\mathcal{G}})$, the term

$$-\int_{\mathcal{G}} d_{\mathcal{G}}(x) \frac{\partial_{\gamma_i} d_{\mathcal{G}}(x)}{d_{\mathcal{G}}(x)} d\mu_{\mathcal{G}}(x) = -\int_{\mathcal{G}} \partial_{\gamma_i} d_{\mathcal{G}}(x) d\mu_{\mathcal{G}}(x)$$

will thus vanish. It remains:

$$-\int_{\mathcal{G}} \partial_{\gamma_i} d_{\mathcal{G}}(x) \log d_{\mathcal{G}}(x) d\mu_{\mathcal{G}}(x)$$

The density $d_{\mathcal{G}}$ is a sum on the curves, and only the i-th term has to be considered. Starting with the expression from (22), one term in the derivative will come from the denominator. It computes the same way as in [16] to yield:

$$\left.\frac{\gamma_i^{t\prime\prime}(t)}{\langle\!\langle \gamma_i'(t), \gamma_i'(t)\rangle\!\rangle_{\mathcal{G}}}\right|_{\mathcal{N}} E(d_{\mathcal{G}}) \tag{28}$$

Please note that the second derivative of γ_i is considered only on its translation component, but the first derivative makes use of the complete expression. As before, the notation $|_{\mathcal{N}}$ stands for the projection onto the normal component to the curve.

The second term comes from the variation of the numerator. Using the fact that the kernel is a product $K^t K^s K^o$ and that all individual terms have a unit integral on their respective components, the expression becomes very similar to the case of spatial density only and is:

$$-\left(\int_{\mathcal{G}} K\left(x, \gamma_i(t)\right) \log d_{\mathcal{G}}(x) d\mu_{\mathcal{G}(x)}\right) \left.\frac{\gamma_i^{t\prime\prime}(t)}{\langle\!\langle \gamma_i'(t), \gamma_i'(t)\rangle\!\rangle_{\mathcal{G}}^{1/2}}\right|_{\mathcal{N}} \tag{29}$$

$$+\int_{\mathbb{R}^d} e(t) K^{t\prime}\left(x^t, \gamma_i^t(t)\right) \log d_{\mathcal{G}}(x) \langle\!\langle \gamma_i'(t), \gamma_i'(t)\rangle\!\rangle_{\mathcal{G}}^{1/2} dx^t \tag{30}$$

with:

$$e(t) = \left.\frac{\gamma_i^t(t) - x^t}{\|\gamma_i^t(t) - x^t\|}\right|_{\mathcal{N}}$$

VIII. Results

Although the original Lie group formulation can be simplified in the framework of aircraft trajectories, the computation cost is still too high to be used in practical applications. A preprocessing phase was thus applied to the dataset in order to limit the number of terms used in the expression of the entropy or its gradient. The information about a trajectory can be summarized by the horizontal component of its centroid and its mean heading, yielding a three dimensional point. Using it within an octree allows the density computation to be limited not only to spatial areas close enough to existing trajectories but also to curves with similar orientation. It becomes possible to save a lot of useless kernel evaluations in estimating the density on the rotation part.

The arrivals and departures at Toulouse Blagnac airport were analyzed. The algorithm performs well as indicated on Figure 7. Four clusters are identified, with mean lines represented through a spline smoothing between landmarks. It is quite remarkable that all density based algorithms were unable to separate the two clusters located at the right side of the picture, while the present one clearly show a standard approach procedure and a short departure one.

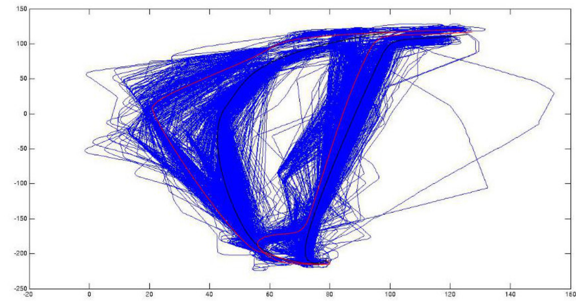

Fig. 7. Bundling trajectories at Toulouse airport

For the case of Figure 7, that represents 1784 trajectories, the computation time is 5 minutes on a XEON 3Ghz machine and with a pure java implementation.

IX. Toward analysis of high dimension data

When dealing with data coming from quick access recorder (QARs), the size of the samples arises as a new issue. It is no longer possible to use the techniques presented in the previous sections as the estimation of the density will become prohibitive in terms of computational complexity. Furthermore, the various components appearing in the samples may be of very different nature: it is thus required to perform first a segmentation into homogeneous groups. A promising approach is the one described in [24] where the data is represented as point in a flag of invariant subspaces. It is in some sense a generalization of the classical PCA decomposition, but with principal subspaces used instead of vectors: the obvious benefit is that less computation is needed and that all vectors pertaining to the same subspace may be considered equivalent. This gives a clue for a adaptation of the entropy based algorithms to very high dimensional data:

- Compute the flag decompositions;
- For a sample point, get its distance to each of the subspaces involved in the flag. Please note that it can be viewed as a kind of generalized coordinates, the usual expansion on a orthonormal basis being recovered by letting the flag by composed only of 1-dimensional subspaces;
- Use the previous expansion as the input data for an entropy based algorithm.

This work is still in early research stage, however it seems to be a very promising way of addressing the issue of QARs data clustering. Two important points are yet to be solved:

- Find an optimal way to construct the flag of invariant subspaces;

- Adapt the entropy based algorithm so as to be able to deal with samples with dimension in the order of 10, typical of what is provided by the flag expansion.

X. Conclusion and future work

The entropy associated with a system of curves has proved itself efficient in unsupervised clustering application where the information coming from geometry is of primary importance. Extending this setting to the case of Lie group valued curves adds the ability to take into account features of the data that are not of spatial nature, like heading or scale. On several test problems, the clusters produced extract relevant aeronautical information, yielding one of the first mathematically sound algorithm able to deal with such data. The next major step will be the introduction of invariant subspaces decompositions, that will naturally produce a summary of very high dimensional samples like produced by the QARs. It is anticipated to pave the way for a new generation of algorithms able to operate in this context.

References

[1] J. Ramsay and B. Silverman, *Functional Data Analysis*, ser. Springer Series in Statistics. Springer New York, 2006.

[2] F. Ferraty and P. Vieu, *Nonparametric Functional Data Analysis: Theory and Practice*, ser. Springer Series in Statistics. Springer, 2006.

[3] M. Enriquez, "Identifying temporally persistent flows in the terminal airspace via spectral clustering," in *ATM Seminar 10*, FAA-Eurocontrol, Ed., 06 2013.

[4] M. El Mahrsi and F. Rossi, "Graph-based approaches to clustering network-constrained trajectory data," in *New Frontiers in Mining Complex Patterns*, ser. Lecture Notes in Computer Science, A. Appice, M. Ceci, C. Loglisci, G. Manco, E. Masciari, and Z. Ras, Eds. Springer Berlin Heidelberg, 2013, vol. 7765, pp. 124–137.

[5] M. Ester, H. peter Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise." AAAI Press, 1996, pp. 226–231.

[6] J. Kim and H. S. Mahmassani, "Spatial and temporal characterization of travel patterns in a traffic network using vehicle trajectories," *Transportation Research Procedia*, vol. 9, pp. 164 – 184, 2015, papers selected for Poster Sessions at The 21st International Symposium on Transportation and Traffic Theory Kobe, Japan, 5-7 August, 2015.

[7] F. N., K. J. T., S. C. E., and S. C.T., "Vector field k -means: Clustering trajectories by fitting multiple vector fields," in *Eurographics Conference on Visualization (EuroVis)*, Preim, P. Rheingans, and H. Theisel, Eds., 2013.

[8] D. G. Kendall, "Shape manifolds, Procrustean metrics, and complex projective spaces," *Bulletin of the London Mathematical Society*, vol. 16, no. 2, pp. 81–121, 1984. [Online]. Available: http://blms.oxfordjournals.org/content/16/2/81.abstract

[9] T. W. Liao, "Clustering of time series data - a survey," *Pattern Recognition*, vol. 38, pp. 1857–1874, 2005.

[10] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: A survey," *International Journal of Computer Applications*, vol. 52, no. 15, pp. 1–9, August 2012, full text available.

[11] W. Meesrikamolkul, V. Niennattrakul, and C. Ratanamahatana, "Shape-based clustering for time series data," in *Advances in Knowledge Discovery and Data Mining*, ser. Lecture Notes in Computer Science, P.-N. Tan, S. Chawla, C. Ho, and J. Bailey, Eds. Springer Berlin Heidelberg, 2012, vol. 7301, pp. 530–541.

[12] A. Delaigle and P. Hall, "Defining probability density for a distribution of random functions," *The Annals of Statistics*, vol. 38, no. 2, pp. 1171–1193, 2010.

[13] C. Bouveyron and J. Jacques, "Model-based clustering of time series in group-specific functional subspaces," *Advances in Data Analysis and Classification*, vol. 5, no. 4, pp. 281–300, 2011.

[14] G. Wahba, *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, 1990. [Online]. Available: http://epubs.siam.org/doi/abs/10.1137/1.9781611970128

[15] P. W. Michor and D. Mumford, "Riemannian geometries on spaces of plane curves," *J. Eur. Math. Soc. (JEMS)*, vol. 8, pp. 1–48, 2006.

[16] S. Puechmorel and F. Nicol, "Entropy minimizing curves with application to automated flight path design," *Lecture notes in computer science, Geometric Science of Information 2015" in MDPI Entropy*, 2015.

[17] P. Griffiths and J. Harris, *Principles of Algebraic Geometry*, ser. Wiley Classics Library. Wiley, 2011. [Online]. Available: https://books.google.fr/books?id=Sny48qKdW40C

[18] S. Puechmorel, "Geometry of curves with application to aircraft trajectory analysis." *Annales de la facult des sciences de Toulouse*, vol. 24, no. 3, pp. 483–504, 07 2015.

[19] IARIA, Ed., *Unsupervised Aircraft Trajectories Clustering: a Minimum Entropy Approach*.

[20] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, 09 1962. [Online]. Available: http://dx.doi.org/10.1214/aoms/1177704472

[21] K. Mardia and P. Jupp, *Directional Statistics*, ser. Wiley Series in Probability and Statistics. Wiley, 2009.

[22] E. García-Portugués, R. M. Crujeiras, and W. González-Manteiga, "Kernel density estimation for directional–linear data," *Journal of Multivariate Analysis*, vol. 121, pp. 152–175, 2013.

[23] K. V. Mardia, "Statistics of directional data," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 37, no. 3, pp. 349–393, 1975.

[24] T. Marrinan, J. R. Beveridge, B. Draper, M. Kirby, and C. Peterson, *Numerical Mathematics and Advanced Applications - ENUMATH 2013: Proceedings of ENUMATH 2013, the 10th European Conference on Numerical Mathematics and Advanced Applications, Lausanne, August 2013*. Cham: Springer International Publishing, 2015, ch. Flag Manifolds for the Characterization of Geometric Structure in Large Data Sets, pp. 457–465. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-10705-9_45