Curve clustering based on second order information: application to bad runway condition detection

Florence Nicol, Stéphane Puechmorel, Baptiste Gregorutti, Cindie Andrieu

Curve clustering based on second order information: application to bad runway condition detection

Florence Nicol, *ENAC, Université de Toulouse*, Stéphane Puechmorel, *ENAC, Université de Toulouse*
Baptiste Gregorutti, *Safety Line,* and Cindie Andrieu, *Safety Line.*

***Abstract*—In air transportation, a huge amount of data is continuously recorded such as radar tracks that may be used for improving flight as well as airport safety. However, all known statistical algorithms such as clustering procedures, even those based on functional data, are unable to distinguish between a safety critical flight and another one departing from standard behavior, but otherwise safe. It is the case in airport safety when radar measurements are used for detecting incidents on airport surface. In this paper, we propose a change of paradigm by switching from a functional data framework to a geometrical one by representing curves as points in a shape manifold. In this way, any intrinsic structure of the data that is amenable to geometry can be directly encoded in the representation space. Based on an extension of a classical distance between shapes, a new one is defined, that explicitly takes into account the second derivative and can be related to slippery. Its properties are investigated in a first part, then some results on datasets of synthetic and real trajectories are presented.**

***Index Terms*—curve clustering, outlier detection, similarity measure, shape manifold, functional data, air traffic management, airport safety.**

I. Introduction

In the context of aviation safety, a crucial issue consists in assessing runway adherence condition and detecting incidents on airport surface by observing only the radar tracks of landing aircraft, e.g. trajectory deviation and abnormal deceleration may be good indicators of runway bad state. Indeed, landing aircraft must brake in a quite short time, putting a stringent condition on the adherence coefficient of the runways. Bad weather conditions like rain, snow and icing can dramatically lower it, increasing the landing distance and making maneuvers more difficult to perform. Rubber deposited by the wheels during braking may also impair the friction coefficient of the runway making it more slippery.

To estimate the runway adherence condition, a direct measurement of it on the pavement is usually performed: it implies sending a vehicle with a dedicated tool, which will interfere with the ongoing traffic. With the expected increase in air traffic, this procedure is particularly restrictive: based on recent studies, European air traffic was expected to grow, yielding an increase of 1.2 million flights in 2023, 14% more than in 2016 [1]. In the context of the COVID-19 pandemic, the massive drop in air traffic and the traffic evolution significantly have reduced this problem. However, it still arises at hub airports and this procedure has a significant operational cost and environmental impact. Major framework programs such as SESAR (Single European Sky Air traffic management Research) in Europe aim to improve air traffic management (ATM) by investigating innovative solutions on management and traffic flow analysis. In particular, for assessing runway adherence condition, we need for reliable, continuous and consistent tools that not requiring runway closure. The objective is to develop spatio-temporal similarity measures for trajectories that will be used in clustering or outliers detection methods.

S. Puechmorel and F. Nicol are with Université de Toulouse and Ecole Nationale de l'Aviation Civile (ENAC), 31055 Toulouse, France.
B. Gregorutti and C. Andrieu are with Safety Line, 75015 Paris, France.

A. Motivations for Functional Data Analysis

The mathematical problem arising from the application presented above falls within the general framework of functional data statistics. Indeed, aircraft trajectories are functional objects mapping time to position, even if we will most of the time observe discretized samples of trajectories, such as radar measurements.

In the functional data framework, the commonly used algorithms developed for multivariate analysis are inoperative and may induce some numerical instabilities. Several works were dedicated to the extension of multivariate algorithms to sample paths of Hilbert processes. As a starting point, data is first expanded on a truncated Hilbert basis [2] then the vectors of expansion coefficients enters a standard finite dimensional analysis. A clever choice of the representation space and basis allows to take into account the a priori knowledge about the studied process. In [3], an EM functional clustering algorithm is presented with adaptive basis in each group, yielding an efficient numerical method to deal with this issue. Another class of methods relies on a non-parametric approach [4], [5]. A recent work [6] pertaining to this approach presents a hierarchical clustering principle, with application to electric power consumption.

In ATM, some works have already used functional data statistics, such as functional principal component analysis (FPCA), functional regression and random forest. FPCA is performed on aircraft trajectories in [7], [8], yielding some insights on the nowadays traffic and then allow to forecast the expected one. Such a functional decomposition is used in [9] for bundled simplification of 2D and 3D curve-sets. New curves are next generated after deforming the cluster centroids to manipulate the underlying data in a simple way while preserving their statistical properties. The problem of short to mid-term aircraft trajectory prediction is considered in [10]. This approach is based on a local linear functional regression

using wavelet decomposition. In [11], random forests for functional data are used for minimizing the risk of accidents and identifying explanatory factors in the context of aviation safety. In the closed context of road traffic, the derivative information was used in a functional analysis of speed profiles of vehicles [12], providing a functional modeling of space-speed profiles of vehicles that leads to a complex nonparametric regression problem.

A major asset of working with functional data framework is the ease of adding a priori information by carefully selecting the Hilbert space and basis. Unfortunately, the dimension of the samples produced that way may be high, and varies with the geometric features of the sample paths. In particular, the presence of high curvature values will increase the number of expansion coefficients needed to keep a good approximation of the original function.

Moreover, the choice of the L^p metric as a measure of similarity between trajectories may be problematic in many situations and does not reflect the high internal structure of aircraft trajectories. As noted in [13], this metric suffers from several drawbacks, especially in registration of functional data when functions are shifted, owing to time lags or general differences in dynamics and alternative metrics should be implemented in a functional data analysis.

For overcoming these difficulties, several methods are based on warping distance such as DTW [14], LCSS [15], EDR [16] and ERP [17]. Alternative distances focus on the shape of trajectories, such as the Hausdorff distance [18], the Fréchet distance [19] and the SSPD distance [20]. A comprehensive review of these distances used in distance-based trajectory clustering is given in [20].

B. The Riemannian framework

An alternative solution consists in switching from a functional data framework to a geometrical one, by representing trajectories in a Riemannian framework. Indeed, curves can be made amenable to functional data statistics by representing them as points on the so-called shape manifold, that is formally defined as a quotient of the manifold of smooth immersions from $[0,1]$ to $\mathbb{R}^n$. It is not a Hilbert space, but it can be provided with Riemannian metrics that allow geodesic distance computation, although some care must be taken to avoid degeneracy of the metric [21].

In the shape space literature, curves are considered as geometric objects and are represented in the so-called shape space, that is the set of immersions quotiented by the group of smooth diffeomorphisms of the interval $[0,1]$. An overview of various notions of shape spaces is given in [22]. Three family of metrics on shape space can be distinguished: the almost local metrics, the Sobolev-type metrics and the right-invariant metrics on the diffeomorphism group of the ambient space. Note that this kind of metrics verify the reparametrization invariance property.

More general almost local metrics and Sobolev-type inner metrics on the space of plane curves were considered in [23] and some algorithms [24] use a shape manifold representation in order to derive a metric between sample paths. The special case of order one Sobolev type metric is presented in [25] and a similar approach takes into account a greater amount of information on the space separating two curves by computing the distance in the manifold itself rather than in one tangent plane [26].

Having it at hand, statistical analysis of functional data such as trajectories clustering may be performed in a standard way. It worth mention that most of the time a mean of computing the centroid of a set of curves is mandatory. In [13], elastic metric for curves has been integrated into statistical analysis of functional data. This so-called Elastic Functional Data Analysis performs alignment, principal component analysis, and modeling of multidimensional and unidimensional functions using the square-root velocity framework developed in [25], [27]. Nice libraries such as the R package `fdasrvf` have been developed [28].

Considering methods falling in this category, it appears that the requirements of the landing trajectory analysis are not fulfilled: enforcing a full parametrization invariance as in the case of shape-based approaches prevents the use of longitudinal acceleration that enters the non-slip condition. On the other side, Sobolev-like metrics are designed to be tailored to specific needs. Motivated by a real use case where one wants to assess runway adherence condition by observing only the radar tracks of landing aircraft, we propose in this paper to use a Riemannian framework for curves with velocity information.

C. Related works in Trajectory Clustering

The mathematical problem arising from the estimation of the runway adherence condition falls within the general framework of curve clustering. Several well established algorithms may be used for performing clustering on a set of trajectories.

First, in sample-based approaches, time samples on trajectories are used in a classical multivariate analysis. Due to the huge number of points, a dimension reduction procedure must be applied first. Random projections followed by a spectral clustering was successfully applied in a study conducted by the Mitre corporation on behalf of the Federal Aviation Authority (FAA) [29]. The most important limitation of this approach is that the shape of the trajectories is not taken into account when applying the clustering procedure unless a resampling procedure based on arclength is applied.

Secondly, density-based algorithms rely on a density map on the state space and a definition of similarity measure. They aim to attract trajectories towards areas of high density and iterate the process until convergence. This kind of algorithms are well suited for already structured graph structure, most implementation derived from the original work presented in [30]. The distance-based density-based algorithms developed in [31], [32] are well adapted to road traffic as vehicles are bound to follow predetermined segments but are unable to work on an unknown airspace organization.

Due to the functional nature of trajectories, that are basically mappings defined on a time interval, it seems more appropriate to resort to techniques based on times series as surveyed in [33], [34], or functional data statistics, with standard references [2], [4]. Again, in both approaches, a

distance between pairs of trajectories or, in a weaker form, a measure of similarity must be available. The algorithms of the first category are based on sequences, possibly in conjunction with dynamic time warping [35], while in functional data analysis, samples are assumed to come from an unknown underlying function belonging to a given Hilbert space. Both approaches yield similar end algorithms, since functional data revert for implementation to usual finite dimensional vectors of expansion coefficients on a suitable truncated basis. 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 as mentioned in [5]. A nice example of a model-based approach working on functional data is the R package `funHDDC` [3], [36].

The purpose of this paper is to introduce a new kind of Riemannian metric that is especially adapted to the clustering of curves where the velocity is a discriminating feature. In such a case, the original shape space approach cannot be used since the parametrization invariance will wipe out the velocity information. A partial parametrization invariance is introduced, yielding a bundle shape space model on which a relevant metric can be defined. The design of the metric was based on the equations of motion and reflects the internal structure of the data. The numerical implementation of the path computation of the new metric is given. For simplicity, we will use a discrete approximation of the total energy of the sampled path rather than a practical computation algorithm based on a shooting method. The minimal approximate energy will be obtained by a standard optimization algorithm and will be next used in the clustering phase.

The paper is organized as follows. In Section II, we formally describe the theoretical framework and define an adapted metric taking into account the velocity information. In Section III, the numerical implementation of the new metric and the K-medoids clustering method [37], [38] is described in a general setting. This algorithm allows to implement another metric than the usual Euclidean distance. Next, Section IV illustrates the performance of clustering algorithms with the new metric for low adherence detection on a set of simulated and real aircraft trajectories. These results are also compared with competing distances. Finally, some comments and future works are drawn in Section V.

II. Theoretical framework

This section is dedicated to the definition of a relevant metric for curves taking into account velocity information. The framework presented here will overcome the limitations of the shape approach through the use of a metric in which the slip characteristic of the landing path is explicitly considered.

A. Slip detection

Landing aircraft may experience slip during deceleration phase when the runway is in degraded conditions. It may result from icing, snow, bad runway surface state but also from pilot's actions, namely a too high braking action or a sharp turn. In this last case, it is not related to runway condition and must not trigger a maintenance action from the airport services.

Slip can be detected on-board by comparing wheel rotation rate with aircraft velocity and computing the so-called wheel slip factor:

$$\lambda = \frac{\omega_w - \omega_a}{\max(\omega_w, \omega_a)} \tag{1}$$

where ω_w is the wheel angular velocity and $\omega_a = V_a/R_w$ is the expected angular velocity that can be computed as the ratio of the aircraft velocity to the wheel radius. Please note that on the real vehicle, several wheels are used, and the λ coefficient has to be understood as a mean value. Furthermore, due to tire elasticity, λ is not zero even if there is no actual slip: this is due to the fact that when a traction or a braking force is applied, the rubber will stretch, resulting in the tire outer part actually traveling more or less than expected from rigid body dynamics. This information is not yet downlinked in real time to ground centers and thus cannot be used in the intended application. From the ground standpoint, λ cannot be observed without on-board information, but some aspects of the landing or taxiing aircraft behavior may still be inferred. It is assumed in the sequel that Coulomb's law for friction [39] is applicable, so that the contact force F_c depends only on aircraft weight and tire/runway conditions:

$$F_c \leq \mu g M \tag{2}$$

with M the aircraft mass, g the gravity of Earth and μ the adhesion coefficient. Without slip, μ is equal to the static friction coefficient μ_s and F_c can be increased until it reaches the upper bound in (2). At that point, slip occurs and μ drops to the value of the dynamic friction coefficient μ_d. The contact force F_c remains constant until it falls below $\mu_d g M$. In real world experiments, this simple behavior is no longer valid and one has to expression μ as a function of λ [40]. Within this frame, the expression of the contact force is $F_c = \mu(\lambda) g M$, which is valid for both non-slip and slip case. Furthermore, in the case of aircraft, aerodynamics forces are exerted, with a net result of a braking force F_a that adds to the actual brakes action, but does not contribute to the friction analysis. Putting things together, the equation of motion along the aircraft trajectory γ can be expressed as:

$$\ddot{\gamma}(t) = \frac{F_a(t)}{M} + \mu(\lambda(t)) g \vec{u} \tag{3}$$

where $\vec{u}$ is a unit vector in the direction of the contact force F_c. Without making additional assumptions, it is not possible to use (3) for slip detection. However, if actions taken are assumed to be optimal, then F_a and $\vec{u}$ will be collinear so as to maximize the net braking effect. The expression of the aircraft dynamics becomes:

$$\ddot{\gamma}(t) = \left(K(t) + \mu(\lambda(t)) g\right) \vec{u} \tag{4}$$

where the coefficient $K(t)$ accounts for the aerodynamic braking force intensity. As aircraft must loose speed fast, μ will be close to the maximum at least during the landing and the beginning of taxi. The same applies for K, as it will not impair adherence. It can then be deduced that aircraft will try

to keep the ratio between longitudinal and normal acceleration as high as possible.

An observable measurement of slip condition can be deduced from the previous remark by considering the angle θ between the acceleration and speed vectors:

$$\sin(\theta(s)) = \frac{\kappa(s)\|D_s\gamma(s)\|^2}{\|D_{ss}\gamma(s)\|} = \frac{\det\left(D_s\gamma(s), D_{ss}\gamma(s)\right)}{\|D_s\gamma(s)\|\|D_{ss}\gamma(s)\|} \quad (5)$$

where γ is the aircraft trajectory and κ its curvature. In the above expression and through all the document, the symbol D_s stands for the partial derivative with respect to variable s. Higher order derivatives are written similarly as $D_{s_1\dots s_1, s_2\dots s_2,\dots}$ by repeating the variable p times to indicate a partial derivative of order p.

In good runway conditions, the longitudinal acceleration will be high and nearly constant, at least in the first part of the landing trajectory. As a consequence, one can expect θ to be relatively small and be proportional to $\det\left(D_s\gamma, D_{ss}\gamma\right)$. Reciprocally, under slip conditions, a trade off has to be made between path following and deceleration: the angle θ will thus increases towards the limiting value $\pm\pi/2$.

Please note that θ is not defined in the portions of the trajectory where the acceleration or the velocity vanishes. While the second situation is highly uncommon, as it will indicate a stop during landing or taxiing, the first one can be encountered when the aircraft is not braking nor turning. In such parts of the trajectory, it is not possible to infer an adherence condition, and curve comparison can be done only on a geometrical basis. In the addressed application, aircraft in the observed part of the landing trajectory are decelerating, so that the above issue is not a concern.

B. An adapted metric in the space of trajectories

Recall that a smooth curve $\gamma\colon [0,1] \to \mathbb{R}^2$ will be an immersion when the derivative $D_s\gamma$ is everywhere non vanishing in $]0,1[$. The set of such curves will be denoted by $\mathbf{Imm}([0,1],\mathbb{R}^2)$. It has the structure of Banach manifold, its tangent space at $\gamma \in \mathbf{Imm}([0,1],\mathbb{R}^2)$ being the vector space $C^\infty([0,1],\mathbb{R}^2)$. A tangent vector is thus a couple (γ, h) with γ the base curve and h an element of $C^\infty([0,1],\mathbb{R}^2)$ that is interpreted as an infinitesimal displacement field along γ. This point can be formalized using the following notion of admissible variation.

Definition 1. Let γ be a smooth loop. An admissible variation of γ is a smooth mapping $\Phi\colon]-\epsilon,\epsilon[\times[0,1] \to \mathbb{R}^2, \epsilon > 0$, such that $\Phi(0,\bullet) = \gamma(\bullet)$.

An admissible variation defines a tangent vector $(\gamma, D_t\Phi|_{t=0})$. The extension to more general immersions is quite straightforward [21]. In the same reference, the variation formula is used to derive a Riemannian metric on the quotient space $\mathbf{Imm}(\mathbb{S}^1,\mathbb{R}^2)/\mathbf{Diff}([0,1],\mathbb{R}^2)$. These notions can be easily generalized to the case of open curves.

In the present work, a similar approach will be taken. However, due to the fact that the slip condition must come into play, it is not meaningful to keep invariance under change of parametrization. Instead, a weaker invariance by affine change of parametrization will be obtained at the end. Furthermore, curves with vanishing second derivative must be excluded since the slip angle θ in (5) is not defined at points where $D_{ss}\gamma(s) = 0$. The last condition boils down to the requirement that the curve $s \in [0,1] \mapsto (\gamma, D_s\gamma)$ be an immersion. The space of such objects will be denoted by $\mathbf{Imm}^*([0,1],\mathbb{R}^2)$. An mentioned above, this is not an issue for the application, due to the restriction of the braking part of the trajectory. In a more general setting, it may be necessary to segment curves into non vanishing second derivative parts and compare only them.

The slip angle in (5) has a nice variational interpretation as indicated in the next lemma.

Lemma 1. *Let $\gamma\colon [0,1] \to \mathbb{R}^2$ be a smooth loop and Φ an admissible variation of it. Let ϕ be a smooth path such that $\phi(0) = \gamma(1)$, $\phi(1) = \gamma(0)$. Then:*

$$D_tA(0) = \int_0^1 \det\left(D_t\Phi(0,s), D_s\Phi(0,s)\right) ds \quad (6)$$

where $A(t)$ is the net area enclosed by the loop $\Phi(t,\bullet), \phi$ for $t \in]-\epsilon,\epsilon[$.

Proof. For any $t \in]-\epsilon,\epsilon[$, the loop $\Phi(t,\bullet), \phi$ is piecewise smooth so that the net area $A(t)$ can be computed using Stoke's formula:

$$A(t) = \frac{1}{2}\int_0^1 \det(\Phi(t,s), D_s\Phi(t,s))ds \quad (7)$$

$$+ \frac{1}{2}\int_0^1 \det(\phi(s), D_s\phi(s))ds \quad (8)$$

Taking the derivative with respect to t yields:

$$D_tA(t) = \frac{1}{2}\left(\int_0^1 \det(D_t\Phi(t,s), D_s\Phi(t,s))ds \quad (9)\right.$$

$$\left. + \int_0^1 \det(\Phi(t,s), D_{st}\Phi(t,s))ds\right) \quad (10)$$

Recalling that the end points of $\Phi(t,\bullet)$ are fixed, one can use integration by parts to get:

$$D_tA(t) = \frac{1}{2}\left(\int_0^1 \det(D_t\Phi(t,s), D_s\Phi(t,s))ds \quad (11)\right.$$

$$\left. - \int_0^1 \det(D_s\Phi(t,s), D_t\Phi(t,s))ds\right) \quad (12)$$

and finally, using the alternating property of the determinant:

$$D_tA(t) = \int_0^1 \det(D_t\Phi(t,s), D_s\Phi(t,s))ds \quad (13)$$

and the conclusion follows by letting $t = 0$. □

Using Lemma 1, the integral :

$$\int_0^1 |\sin(\theta(s))|\,\|D_s\gamma(s)\|ds = \quad (14)$$

$$\int_0^1 \frac{|\det\left(D_s\gamma(s), D_{ss}\gamma(s)\right)|}{\|D_{ss}\gamma(s)\|}ds \quad (15)$$

may be interpreted as the total infinitesimal area swept by the curve γ when moved in the direction $D_{ss}\gamma$.

The integral in (14) is invariant under affine change of parametrization. Given a curve γ defined on an arbitrary interval $[a, b]$, it is thus possible to go back to fixed interval $[0, 1]$. As a consequence, any curves will be assumed to be defined on $[0, 1]$. Please note however that general invariance under change of parametrization cannot be obtained, and will not be meaningful as the information sought after is clearly dependent on the velocity.

The slip angle θ can be obtained in a convenient way using the fact that the curves take their values in $\mathbb{R}^2$ and can thus also be considered as complex valued mappings. Within this frame, it comes for the expression of the slip angle θ if curve γ:

$$e^{i\theta} = \frac{D_s\gamma}{|D_s\gamma|}\frac{\overline{D_{ss}\gamma}}{|D_{ss}\gamma|}. \tag{16}$$

Using the complex representation, the following proposition gives access to the variation of the slip angle.

Proposition 1. *Let $\Phi\colon]-\epsilon, \epsilon[\times[0,1]$ to $\mathbb{C}$ be an admissible variation of γ. Let $\theta(t, s)$ be the slip angle at s of the curve $s \mapsto \Phi(t, s)$. Then:*

$$D_t e^{i\theta} = \frac{1}{|D_s\Phi|} D_{st}\Phi|_N \frac{D_{ss}\overline{\Phi}}{|D_{ss}\Phi|} N + \frac{1}{|D_{ss}\Phi|}\overline{D_{sst}\Phi|_{\widetilde{N}}}\frac{D_s\Phi}{|D_s\Phi|}\widetilde{N}$$

where the notations N and $|_N$ (resp. $\widetilde{N}$ and $|_{\widetilde{N}}$) denotes the unit normal vector and the component normal to the curve $s \mapsto \Phi(t, s)$ (resp. $s \mapsto D_s\Phi(t, s)$).

Proof. The proposition is a direct computation of the derivative. The only point to note is that:

$$D_t\left(\frac{D_s\Phi}{|D_s\Phi|}\right) = \frac{D_{st}\Phi}{|D_s\Phi|} - \frac{D_s\Phi}{|D_s\Phi|}\frac{\Re\left(D_{st}\Phi.D_s\overline{\Phi}\right)}{|D_s\Phi|^2} \tag{17}$$

$$= \frac{1}{|D_s\Phi|} D_{st}\Phi|_N\, N \tag{18}$$

and the equivalent derivation for $D_{ss}\Phi$:

$$D_t\left(\frac{D_{ss}\Phi}{|D_{ss}\Phi|}\right) = \frac{1}{|D_{ss}\Phi|}\overline{D_{sts}\Phi|_{\widetilde{N}}}\,\widetilde{N}.$$

□

Proposition 2. *Under the same assumptions as in Proposition 1:*

$$(D_t\theta)^2 = \left(\frac{D_{st}\Phi|_N}{|D_s\Phi|} - \frac{D_{sst}\Phi|_{\widetilde{N}}}{|D_{ss}\Phi|}\right)^2$$

Proof. Using the fact that $|e^{i\theta}| = 1$ and Proposition 1, it comes:

$$(D_t\theta)^2 = \frac{1}{|D_s\Phi|^2}|D_{st}\Phi|_N|^2 + \frac{1}{|D_{ss}\Phi|^2}|D_{sst}\Phi|_{\widetilde{N}}|^2 + 2\frac{1}{|D_s\Phi||D_{ss}\Phi|}\Re\left(D_{st}\Phi|_N.\overline{\overline{T}}.D_{sst}\Phi|_{\widetilde{N}}\overline{T}\right)$$

with:

$$T = \frac{D_s\Phi}{|D_s\Phi|}, \qquad \widetilde{T} = \frac{D_{ss}\Phi}{|D_{ss}\Phi|}.$$

Since the curve is planar, $N = iT$ (resp. $\widetilde{N} = i\widetilde{T}$). This combined with the expression of the normal component $D_{st}\Phi|_N = \langle D_{st}\Phi, N\rangle$ (resp. $D_{sst}\Phi|_{\widetilde{N}} = \langle D_{sst}\Phi|_{\widetilde{N}}, \widetilde{N}\rangle$) yields:

$$\Re\left(D_{st}\Phi|_N.\overline{\overline{T}}.D_{ss}D_t\Phi|_{\widetilde{N}}\overline{T}\right) = -\langle D_{st}\Phi, N\rangle\langle D_{sst}\Phi|_{\widetilde{N}}, \widetilde{N}\rangle$$

and the result follows. □

An admissible variation Φ of a curve γ defines a tangent vector to γ in the space $\mathbf{Imm}^*([0,1],\mathbb{R}^2)$ by $D_t\Phi|_{t=0}$. Such a tangent vector is thus a couple (γ, u) where $u \in C^\infty([0,1],\mathbb{R}^2)$. The above expression for $(D_t\theta)^2$ allows to define a semi-norm in the tangent space to the curve γ by integration along the curve.

Definition 2. Let (γ, u) be a tangent vector to γ. The semi-norm $M_{(\gamma,u)}$ is defined by:

$$M^2_{(\gamma,u)} = \int_0^1 \left(\frac{D_s u|_N}{|D_s\gamma|} - \frac{D_{ss}u|_{\widetilde{N}}}{|D_{ss}\gamma|}\right)^2 ds$$

with N (resp. $\widetilde{N}$) the normal vector to γ (resp. $D_s\gamma$).

Given two curves γ_1, γ_2, a path between them in $\mathbf{Imm}^*([0,1],\mathbb{R}^2)$ is smooth mapping $\Phi\colon [0,1]\times[0,1] \to \mathbb{R}^2$ such that:

- $\Phi(0,\bullet) = \gamma_1(\bullet), \Phi(1,\bullet) = \gamma_2(\bullet)$
- For all t in $]0,1[$, $\Phi(t,\bullet) \in \mathbf{Imm}^*([0,1],\mathbb{R}^2)$.

Definition 3. Let Φ be a path between γ_1 and γ_2. Its θ-deformation energy is defined as:

$$E(\Phi) = \int_0^1 \left(|D_t\Phi(t,0)|^2 + |D_t\Phi(t,1)|^2\right) dt + \int_0^1\int_0^1 \left(\frac{D_{st}\Phi|_N}{|D_s\Phi|} - \frac{D_{sst}\Phi|_{\widetilde{N}}}{|D_{ss}\Phi|}\right)^2 ds\, dt.$$

The first term is needed as the second part involves only derivatives of the curve and will not take endpoints into account. The above definition of the energy is valid for curves in $\mathbf{Imm}^*([0,1],\mathbb{R}^2)$, but is not invariant by affine change of parametrization. Furthermore, it is based on a semi-metric, resulting in possible degeneracies. The path energy is thus modified according to Definition 4.

Definition 4. Let Φ be a path between γ_1 and γ_2. Its total deformation energy is defined as:

$$\widetilde{E}(\Phi) = \int_0^1 \left(|D_t\Phi(t,0)|^2 + |D_t\Phi(t,1)|^2\right) dt + \int_0^1\int_0^1 |D_t\Phi|^2|D_s\Phi|\, ds\, dt + \lambda\int_0^1\int_0^1 \left(\frac{D_{st}\Phi|_N}{|D_s\Phi|} - \frac{D_{ss}D_t\Phi|_{\widetilde{N}}}{|D_{ss}\Phi|}\right)^2 |D_s\Phi|\, ds\, dt$$

where $\lambda > 0$ tunes the relative importance of shape and slip angle.

The first double integral is a Sobolev Riemannian metric that accounts for shape deformation. The free parameter λ must be tuned in applications. The energy allows the computation of geodesic path between curves.

Definition 5. The geodesic path Φ_0 between γ_1 and γ_2 is, when it exists, defined by:

$$\Phi_0 = argmin_{\Phi \in I} \widetilde{E}(\Phi)$$

where I is the set of paths from γ_1 to γ_2.

The energy of a geodesic path defines a similarity measure between curves. It will be used in the sequel within a clustering algorithm.

III. Numerical implementation

This section describes the numerical implementation of the path computation of the new metric described in Section II-B. Then, the clustering procedures will be presented.

A. Geodesic path computation

Two approaches can be used for computing a geodesic path Φ_0 minimizing the total energy. The first one relies on a system of differential equations involving $D_t\Phi$ that is solved starting at $t = 0$ from the first curve. An iterative shooting procedure reduces gradually the distance at $t = 1$ to the second curve, yielding an approximate geodesic path. This algorithm is appealing as it does not require an approximation to the entire path Φ_0, but just of the curve $s \mapsto \Phi(t, s)$ for a fixed t. Its two main drawbacks are first the need to establish the equation of geodesics, that may be quite complicated, and second the numerical instabilities in the shooting phase. It will nevertheless be investigated in a future work. The second approach is to approximate the full path Φ by its samples $\Phi_{ij} = \Phi(x_{ij}), i = 0 \dots n, j = 0 \dots m$ on a grid defined by the points $x_{ij} = (t_j, s_i)$. In the current implementation, the grid is evenly spaced. A numerical differentiation procedure is applied to obtain the derivatives D_s, D_{ss}, D_t and takes the form of matrices, still denoted with D_s, D_{ss}, D_t, that are applied to the matrix $M = (\Phi_{ij})$ of path sample points. Since the derivative with respect to s must be applied columnwise, the derivative $D_s\Phi$ is approximated by $D_s.M$. Conversely, derivation is performed rowwise for $D_t\Phi$, so that the approximation is $M.D_t$. Finally, the first and the last column of Φ are held constant, since they represent sampled version of the endpoint curves γ_1, γ_2.

Using the discrete approximation, the total energy of the sampled path described by the matrix M is a sum of three terms, hereafter denoted by E_1, E_2, E_3. Their respective expressions are given in equations

$$E_1(M) = \sum_{j=0}^{m} |(M.D_t)_{0j}|^2 + |(M.D_t)_{nj}|^2 \tag{19}$$

$$E_2(M) = \sum_{j=0}^{m} \sum_{i=0}^{n} |(M.D_t)_{ij}|^2 |(D_s.M)_{ij}| \tag{20}$$

$$E_3(M) = \lambda \sum_{j=0}^{m} \sum_{i=0}^{n} \left(\frac{(D_s.M.D_t|_N)_{ij}}{|(D_s.M)_{ij}|} \right. \tag{21}$$

$$\left. - \frac{(D_{ss}.M.D_t|_{\widetilde{N}})_{ij}}{|(D_{ss}.M)_{ij}|} \right)^2 |(D_s.M)_{ij}|. \tag{22}$$

The optimal approximate path is obtained by a standard numerical optimization algorithm, chosen here to be LM-BFGS [41] that is well suited to high dimensional problem. Initializing Φ with a linear homotopy ensure convergence in a few iterations on most problems. The minimal energy obtained is used in the following clustering phase.

B. Clustering methods

K-medoids algorithm [37], [38] is an efficient clustering technique that aims to partition unlabelled data into clusters. Unlike the well known K-means algorithm [42] which is sensitive to outliers, K-medoids algorithm is more robust and uses an actual point in the cluster to represent it. As this method produces realisctic cluster centers, it may be preferred for aircraft trajectory clustering. The most benefit of K-medoids algorithm compared to the K-means is that there is no need to explicitly compute the cluster centers since we just need to store the indices of the trajectories minimizing the total distance to other trajectories in that cluster. Note that K-medoids requires the use of a metric distance that satisfies the triangular inequality. It may be a serious drawback for distances that are not metrics. As recommanded in [20], a hierarchical cluster analysis (HCA) should be prefered.

As a matter of fact, it is possible to perform the clustering with a pairwise distance matrix as input data instead of the raw data. Thereby, we can derive an algorithm with the metric defined in Section II in order to classify a set of n spatiotemporal trajectories $\{x_1, \dots, x_n\}$. Similarly to K-means and other clustering techniques such as HCA, the number of clusters K is assumed to be known. In practice, the knowledge of the data can help the user to estimate the value of K. In the case of aviation safety, one could choose $K = 3$ corresponding to three levels of alerts: low, medium, high. A fourth cluster may be added in the procedure for detecting outliers trajectories. In general, the number of clusters is not known and has to be computed from the data. In [43], the authors introduced the Silhouette criterion as a graphical method for interpreting and validating the results of a clustering algorithm. Broadly speaking, the Silhouette score measures the tightness and the separation of the clusters. More formally, for an observed trajectory x_i, it is defined as:

$$s(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))},$$

where $a(i)$ is the distance between x_i and all other trajectories in the same cluster and $b(i)$ is the distance between x_i and all other trajectories in the next nearest cluster. The overall Silhouette score, which ranges between -1 and 1, is then the mean value of the score computed on each sample, i.e.

$$s = \frac{1}{n} \sum_{i=1}^{n} s(i).$$

Hence, the number of clusters K can be chosen by maximizing the Silhouette criterion. It can also be used as a validation metric of a clustering method: the higher value of s is, the better is the clustering. Note that the mean values of $a(i)$ and $b(i)$ can be interpreted respectively as the intra-cluster variance

and the inter-cluster variance. Knowing that, a higher value of s indicates that the clusters are dense and well separated while a lower value of s refers to an incorrect clustering. A score around zero indicates overlapping clusters.

Note also that the K-medoids can be sensitive to the initialization of the algorithm and may find a local minimum. This is actually a classical drawback of the clustering methods. A common solution is to repeat the clustering with different initializations and to retain the best one with respect to the Silhouette score.

IV. Numerical experiments

In the following, the new metric defined in Section II is called the *slippery metric*. The slippery metric described is tailored to take into account information on acceleration and should be able to detect abnormal deceleration levels that may indicate a potential degraded runway condition. This metric has an affine invariance property by change of parametrization, speed and acceleration of trajectories remaining unchanged except for a multiplicative constant. In this section, the performance of this metric is compared with other existing competitors: one geometric metric, the Square Root Velocity (SRV) metric [25]–[27], and seven other distances described in [20].

Geometric metrics are relevant for clustering trajectories having different morphologies, and for which second-order information is not required; this is the case when some aircraft trajectories deviate from the nominal one. As noted in [44], the SRV metric is invariant under change of parametrization: if the two curves are reparametrized in the same way, the distance does not change. However, if the curves are reparametrized in different ways, the distance changes. Even if the SRV metric is not tailored to take into account information on acceleration, it might be able to detect parametrization differences between curves in addition to changes in the shape. Usually, an optimal matching procedure is used to produce a distance, that can detect differences in the shapes of the curves irrespective of their parametrization. This should not take into account the velocity information and is not be suitable for slip detection. In this section, we will consider both kinds of SRV metric: the first one will be called the *SRV metric before alignment* and the second one will be called the *SRV metric after alignment*. We have used the algorithm developed in [45], implemented in the Geomstats python package [44], [46].

In addition to the SRV metrics, [20] provides a comprehensive review of different distances used in the literature, especially in the context of vehicle trajectory clustering. But most of the used competitor distances are not metrics. For this reason, in the following first subsection, we will perform a hierarchical cluster analysis (HCA) in a validating step on simulated datasets. In the second subsection, the performance of the two SRV metrics and the slippery metric will be compared by using a K-medoids clustering procedure to a real dataset of landing aircraft tracks.

A. The simulation procedure

As already mentioned, it is virtually impossible to ensure that an observed trajectory that deviates from the nominal one is the result of a bad adherence condition. To perform such a task, a physical measurement is mandatory and the only way to correlate runway quality to landing tracks is to have recorded trajectories right after or before a measurement.

A more tractable approach is to simulate landing and taxiing in a realistic fashion, so that the detection performance of the methods can be estimated. Very good models of wheeled ground vehicles are available in the literature [40] and the case of braking aircraft has been considered in [47]. For the purpose of the study, a very detailed model is not needed, but a good conformance to observations is required. The approach taken was to use the model of Figure 1. The aircraft is controlled

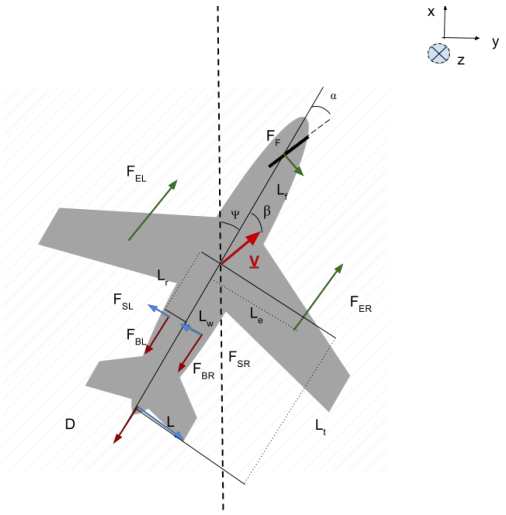


Fig. 1: Aircraft physical model.

using the engine thrust and the front wheel steering angle. The landing gear tires train is replaced by a single equivalent tire. The simulation is performed by solving the equations of motion using a Runge-Kutta 45 scheme with adaptive step. The control loop consists of two Proportional-integrated-derivative (PID) regulators: the inner acts on the steering angle of the front wheel and corrects lateral deviations from the intended path while the outer controls deceleration. When the mobile is skating, the first control loop takes precedence and consumes the apart of the available friction force to minimize the lateral error. The remaining friction is transfered to the outer loop to apply some deceleration. It is clear that this process will increase the slip angle in a trajectory turn.

A simulator was coded in Java and applied to synthetic tracks with tunable adherence coefficient. Since slippery can be detected easily within the simulation loop, all output trajectories are flagged as ”skating” or ”normal”. Two datasets were generated from this simulator to compare the slippery metric with all others competitors. In order to more precisely compare the behavior of the SRV metrics and the slippery metrics, other more tractable datasets were simulated following a circle shape and different parametrizations of curves.

B. The aircraft trajectory simulator datasets and comparison with all competitors

1) Simulated trajectories: Landing and taxiing trajectories are simulated following a circle representing the nominal trajectory. The initial landing point is randomly disturbed as well as the initial velocity. In this way, it may induce a skating at the beginning of the landing trajectory which is then corrected by the regulators as described above. Note that, following the action of the PID regulators, simulated skating trajectories end before the normal ones. Moreover, all the paths take the same initial direction following the nominal trajectory as illustrated in Figure 2. The dataset of simulated trajectories is composed of 16 skating trajectories and 284 normal trajectories. As the analysis of landing trajectories requires to take into account information on acceleration, simulated trajectories have not been normalized.

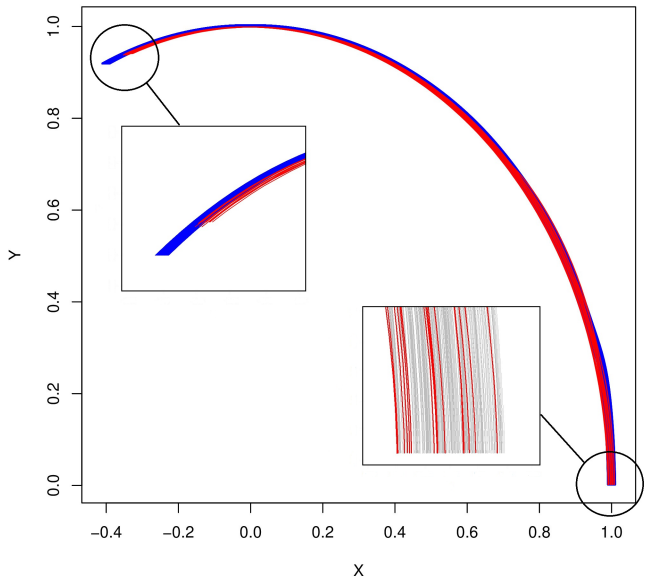


Fig. 2: Simulated trajectories: 284 normal trajectories (blue) and 16 skating trajectories (red).

The slippery metric is compared with both SRV metrics and the distances described in [20]: some warping-based distances such as DTW [14], LCSS [15], EDR [16] and ERP [17], and some shape-based distances, such as the SSPD distance [20], the Hausdorff distance [18] and the Fréchet distance [19]. The discrete Fréchet distance was preferred, as requiring a lower computational cost. Because the discrete Fréchet distance, just like SSPD, DTW, LCSS and EDR distances are not metric, a HCA clustering method is then performed in order to detect two clusters: the normal trajectories cluster and the skating trajectories one. In this framework, *complete*, *median*, *ward* and *centroid* linkage criteria have been compared. Finally, following [20], the *ward* criterium has been chosen, usually giving the best results and producing spherical clusters. Note that the LCSS and EDR warping-based distances require to set a spatial threshold that makes the distance robust to noise. A high threshold value can produce all similar distances, while by using a too low threshold, only very close curves are considered similar. We set this threshold to 0.01 after comparing different results.

TABLE I: number of skating and normal trajectories.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*normal*	*skating*	*normal*
Slippery	0	284	16	0
SRV without alignment	0	284	16	0
SRV with alignment	0	284	16	0
ERP	0	284	16	0
DTW	0	100	16	184
EDR	10	284	6	0
LCSS	10	284	6	0
Discrete Fréchet	0	284	16	0
Hausdorff	0	284	16	0
SSPD	5	188	11	96

Table I displays the number of trajectories in each cluster returned by the HCA procedure. The output trajectories are then flagged as "skating" or "normal". The slippery metric and both SRV metrics, as well the ERP warping-based metric, the Hausdorff and discrete Fréchet shape-based distances, perform well and correctly detect the 16 skating trajectories cluster and the 284 normal trajectories one. The DTW warping-based distance produces one cluster containing only normal trajectories and one cluster composed of all the skating trajectories, but also a large number of normal trajectories. The EDR and LCSS warping-based distances and the SSPD distance detect one cluster with some skating trajectories, the others being clustered with the normal trajectories.
Table II displays some evaluation measures for DTW, EDR, LCSS and SSPD distances: the false negative rate (FNR), the false positive rate (FPR) and some indices measuring how similar the returned clusters are to the skating and normal labels. FNR represents the miss rate, that is the rate of skating trajectories in the cluster of returned normal trajectories corresponding to an undetected skidding, distances with high value miss rate are not suitable from a safety point of view. FPR corresponds to the false alarm rate, that is the rate of normal trajectories falsely detected as skating. A high FDR value corresponds to a situation generating significant costs for ATM as noted in Section I. The Matthews correlation coefficient takes on a value between 0 and 1, with 0 meaning that the two returned clusters are completely different from the skating and normal clusters. Higher values indicate greater similarity and an indicator of 1 means that the output trajectories are correctly clustered as skating and normal.

TABLE II: clustering evaluation.

Metrics	*FNR*	*FPR*	*Matthews*
DTW	0	0.92	0.168
EDR/LCSS	0.034	0	0.602
SSPD	0.259	0.90	0.164

Even if the DTW miss rate is 0, the DTW false alarm rate is very high (92%), as the SSPD distance (90%). EDR and LCSS both produce the lowest miss rate and false alarm rate values, but the Matthews correlation coefficient indicates that more than half of skidding trajectories are not detected by using EDR and LCSS distances.

As noted previously, skating trajectories can be distinguished from normal ones due to their end points. This shape argument could explain why the SRV metrics and the Hausdorff, discrete Fréchet and ERP distances work well for clustering skid and non-skid situations. For this reason, we have truncated the simulated trajectories, so that all trajectories end almost at the same location. We assess clustering for ERP, discrete Fréchet, Hausdorff distances, both SRV metrics and the slippery one in Table III and Table IV .

TABLE III: number of skating and normal trajectories.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*normal*	*skating*	*normal*
Slippery	0	284	16	0
SRV without alignment	0	284	16	0
SRV with alignment	0	284	16	0
ERP	1	145	15	139
Discrete Fréchet	1	145	15	139
Hausdorff	1	145	15	139

TABLE IV: clustering evaluation for the ERP, Hausdorff and Discret Fréchet distances.

FNR	*FPR*	*Matthews*
0.007	0.96	0.21

Results in Table III show that the ERP metric, as well as the Hausdorff and Discrete Fréchet distances, no longer correctly detect the two skating and normal clusters. While the low miss rate value shows that few skating trajectories are falsely detected as normal, the false alarm rate value is very high. Moreover, the Matthews correlation coefficient indicates that the predicted clusters are not similar to the skating and normal clusters.

Finally, only the slippery metric and both SRV metrics seem to be able to correctly cluster trajectories in a skid and non-skid situation. This result is surprising, particularly that of the SRV with alignment, which is probably explained by a weak numerical bias, detected by the HCA procedure. The result from the SRV without alignment is coherent with the fact that, before the optimal matching procedure, the differences in parametrization may be detected, when differences in shapes are especially not enough significant to hide this effect.

2) Slightly deformed simulated trajectories: In order to assess the robustness of the results obtained by using the slippery metric and the SRV metrics, we have slightly deformed trajectories by using a random coefficient (around 1%) so that simulated landing trajectories are now randomly flattened following an ellipsoid and are more realistic.

When HCA is performed on the entire trajectories, both SRV metrics and the slippery metric perform well. We then focused on the beginning of the trajectories before the PID regulators act skidding. Now, both SRV metrics can no longer correctly detect the two clusters while the slippery metric can do so. Results from Table V show that almost all skating trajectories are clustered with normal ones. These two metrics produce a very high false alarm rate (around 90%) and the low Matthews correlation coefficient indicates that the clustering algorithm does not correctly distribute the trajectories into clusters. In Figure 3, we can visualize the clusters produced

TABLE V: number of skating and normal trajectories.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*normal*	*skating*	*normal*
Slippery	0	284	16	0
SRV without alignment	2	186	14	198
SRV with alignment	3	213	13	71

TABLE VI: clustering evaluation.

Metrics	*FNR*	*FPR*	*Matthews*
SRV without alignment	0.011	0.93	0.141
SRV with alignment	0.014	0.84	0.315

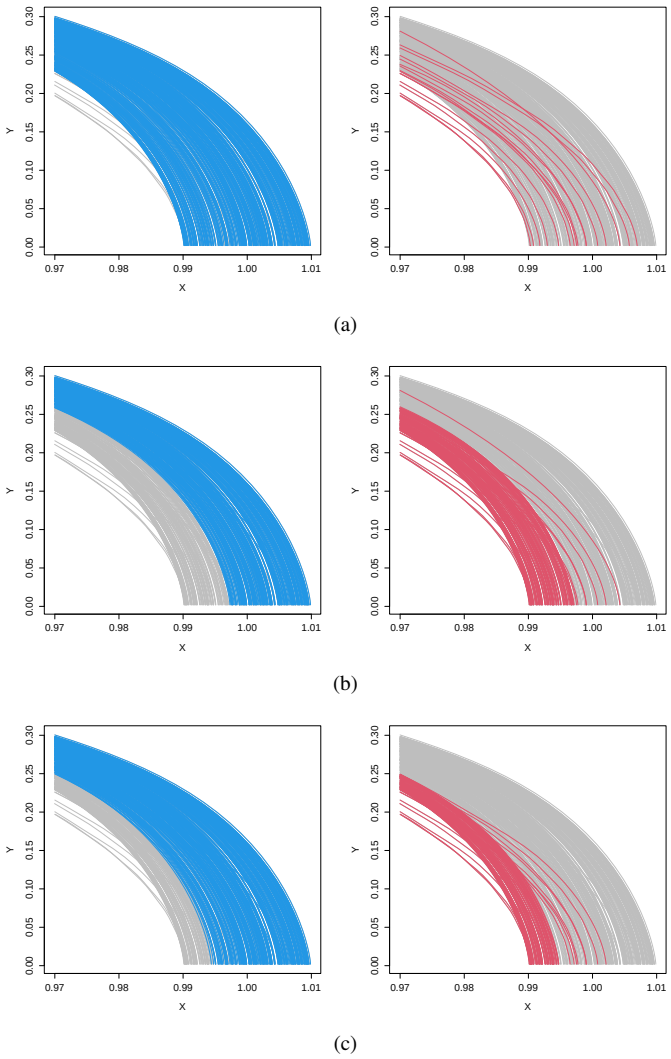


Fig. 3: HCA with (a) the slippery metric, (b) the SRV metric without alignment and (c) the SRV metric with alignment. Two predicted clusters: cluster 1 (blue) and cluster 2 (red).

from HCA with the slippery metric and both SRV metrics. When using the SRV metrics, almost all the skating trajectories are clustered with the normal trajectories whose the starting points have low values. It seems that normal trajectories are distributed in the two predicted clusters in equal proportions relatively to the starting point, the skating trajectories being clustered in one of these clusters.

Finally, it seems that, among all the competitors, only the

SRV metrics might be able to correctly detect the skidding in certain situations. These results are quite surprising in the case of deformed simulated trajectories. We suspect that the differences in the shapes along the entire trajectories are not sufficiently high to hide the parametrization differences or a numerical bias. However, the datasets generated by the simulator may eventually suffer from biases due to the action of the PID regulators. This is the case of the end points of trajectories, but other effects may be present.

C. Application to more tractable simulated datasets

In order to highlight the effect of shape differences, we have compared results from the SRV metrics and the slippery metric with more tractable datasets. In the following datasets, we have considered two profiles of velocity along the curves: a constant speed profile corresponding to skating and a braking one at the beginning of the curves. The datasets of simulated curves are composed of 30 curves, mixing the two profiles of velocity in equal proportions.

1) Trajectories of the same shape: We first consider trajectories following the shape of a circle with initial point being randomly disturbed. Each curve i is a set of positions

$$traj(i) = (a_i \cos\theta + c_i, b_i \sin\theta + c_i), \quad \theta \in [0, \pi/2],$$

where c_i is evenly distributed in an interval of amplitude α and $a_i = b_i = 1$.

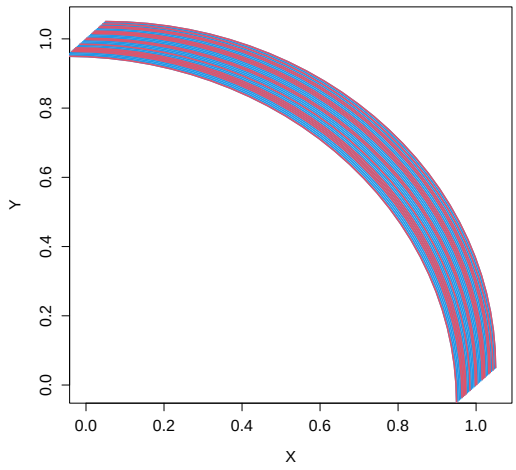


Fig. 4: Simulated circle trajectories: 15 constant speed trajectories (blue) and 15 braking trajectories (red).

The SRV metric is not sensitive to differences in tranlation and should vanish. Surprisingly, the HCA procedure can detect the two clusters of speed profile, even in the case of the SRV without alignment. When examining the distance matrices produced by the SRV metric with alignment, the distance values of similar speed profile curves is around of 10e-14 and 10e-5 for different speed profile curves. These differences seem negligible at first glance, however they are anyway detected by a HCA procedure. In this case, the clustering result is clearly due to a numerical bias. On the other hand, the SRV metric without alignment produces similar distance values for similar speed profile curves, but non negligible values for different speed profile curves around 0.15. This result is less surprising because, as noted in [44], the SRV metric may be sensitive to differences in reparametrization.

2) Trajectories with deviations from the initial shape: We consider the circle-shaped trajectories as in the previous example. However, some of them exhibit a Gaussian-like deviation from the nominal circle-shaped curve. This corresponds to the shape of landing trajectories with lateral deviations from the centreline. For each curve i, this deviation d_i can be inflated by a multiplicative coefficient r as follows

$$d_i(t) = (r * \exp(-0.5 * (t - 0.5)^2/\sigma^2), \quad t \in [0, 1],$$

where t represents a time location and σ a fixed standard deviation. Two situations were considered with different values of the coefficient r : low-medium deviations ($r = 1\%$, 5% and 10%) and medium-high deviations ($r = 1\%$, 10% and 30%). Again, for each scenario, the two speed profiles are equally distributed into the three subgroups curves.

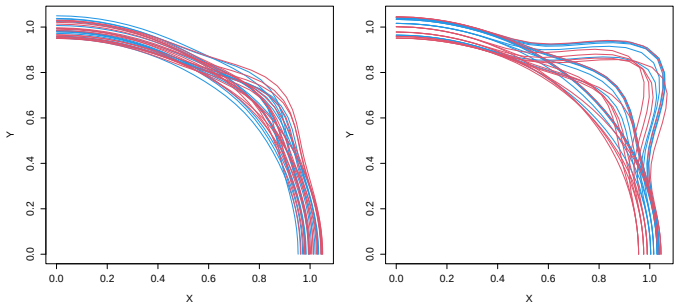


Fig. 5: Simulated deviated trajectories: 15 constant speed trajectories (blue) and 15 braking trajectories (red). Low deviations (left) and high deviations (right).

Table VII shows that, with low-medium deviations, the SRVF metric with alignment can no longer correctly detect the two clusters. Both skating and braking trajectories are similarly clustered into the predicted clusters in equal proportions. Moreover, the miss rate (50%), the false alarm rate (50%) and the zero Matthews correlation coefficient mean that the clustering algorithm randomly distributes the trajectories into clusters. Note that the SRVF metric without alignment can detect skating and braking when deviations are relatively low.

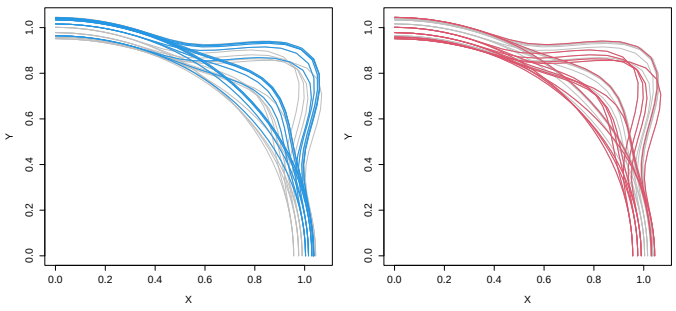
TABLE VII: number of braking and skating curves for low-medium deviations.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*braking*	*skating*	*braking*
Slippery	0	15	15	0
SRV without alignment	0	15	15	0
SRV with alignment	10	10	5	5

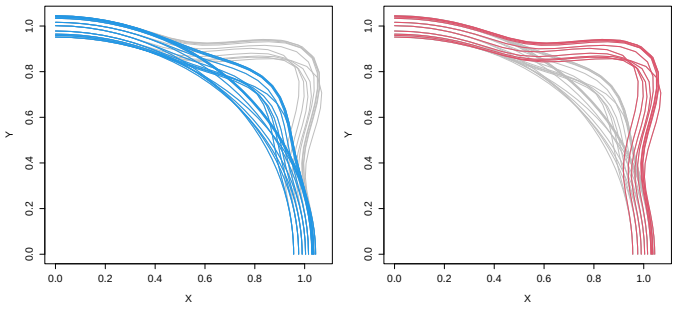
Table VIII displays the results when the differences in the shape are higher. The SRV metric without alignment now favors shape detection to the detriment of speed profile detection. In Figure 6, we can visualize that both SRV metrics cluster the curves relatively to the shape differences while the slippery metric performs well.

TABLE VIII: number of braking and skating curves for medium-high deviations.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*braking*	*skating*	*braking*
Slippery	0	15	15	0
SRV without alignment	10	10	5	5
SRV with alignment	10	10	5	5



(a)



(b)

Fig. 6: HCA with (a) the slippery metric, (b) the SRV metric without and with alignment. Two predicted clusters: cluster 1 (blue) and cluster 2 (red).

3) Trajectories with slightly deformed deviations: Finally, the previous datasets are slightly deformed by adding Gaussian distributed noise $\mathcal{N}(0, \varepsilon = 0.05)$ along the curves. A smoothing splines regression technique is next applied to recover smooth trajectories. The resulting dataset is then composed of slightly deformed curves relatively to the previous example.

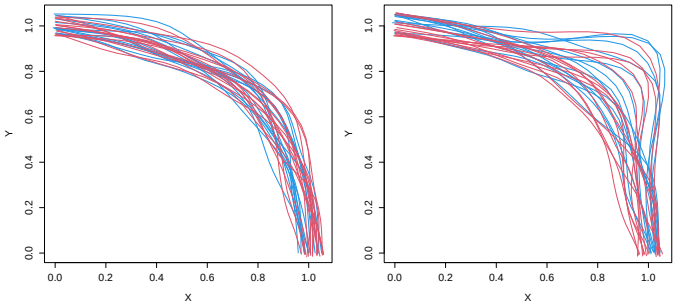


Fig. 7: Simulated slightly deformed deviated trajectories: 15 constant speed trajectories (blue) and 15 braking trajectories (red). Low deviations (left) and high deviations (right).

Both SRV metrics are now unable to correctly cluster the curves as skating and braking. Even if the evaluation criteria are better for the SRV without alignment than for the SRV with alignment, the false alarm rate remains important (20%).

TABLE IX: Low-medium deviations: number of braking and skating curves for slightly deformed curves.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*braking*	*skating*	*braking*
Slippery	0	15	15	0
SRV without alignment	0	11	15	4
SRV with alignment	7	8	8	7

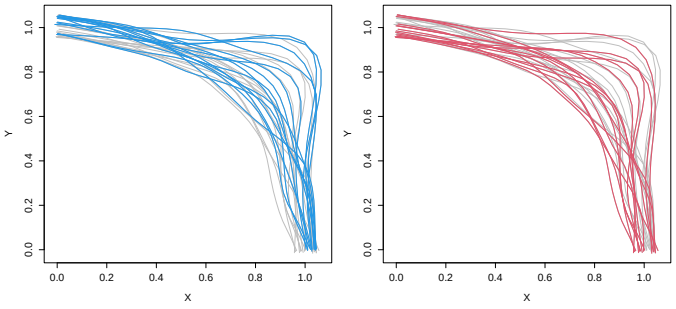
TABLE X: clustering evaluation.

Metrics	*FNR*	*FPR*	*Matthews*
SRV without alignment	0	0.2	0.76
SRV with alignment	0.47	0.47	0.07

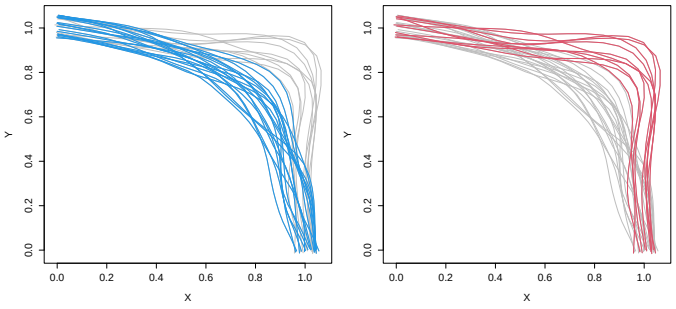
Finally, with medium-high deviations, we obtain the same results as those found in the previous examples: the parametrization detection is hidden by the shape differences by both SRV metrics while it is clearly identified by the slippery metric.

TABLE XI: Medium-high deviations: number of braking and skating curves for slightly deformed curves.

Metric	*Cluster 1*		*Cluster 2*	
	skating	*braking*	*skating*	*braking*
Slippery	0	15	15	0
SRV without alignment	10	10	5	5
SRV with alignment	10	10	5	5



(a)



(b)

Fig. 8: HCA with (a) the slippery metric, (b) the SRV metric without and with alignment. Two predicted clusters: cluster 1 (blue) and cluster 2 (red).

These results highlight those obtained with the aircraft trajectory simulator datasets. When the curves have the same

shape, both SRV metrics perform well, due to a numerical bias for the SRV metric with alignment. When the curves have different morphologies, the SRV with alignment behaves as expected and detects only differences in shape. The numerical bias seems to be hidden in this case. This observation does not hold for the SRV without alignment, which depends on the parametrization of the curves. Even if this metric is not tailored to take into account information on acceleration, it is able to perform well in certain situations when shape differences are not high enough to hide differences in parametrization. As an alternative, the slippery metric that is tailored to detect the information on acceleration performs well in each studied case.

D. Application to Radar tracks

In this section, we apply the K-medoids clustering method to a database of radar tracks in order to detect bad runway condition and skidding. Note that an HCA procedure was also performed and analogous results were obtained.

1) Description of the data: The Radar tracks used here are a set of landings coming from A-SMGCS[1] which is a system that provides air traffic controllers with services to maintain airport safety and capacity. These data are generated by sensor fusion of Primary Surveillance Radar (PSR), Secondary Surveillance Radar (SSR) using onboard transponders and multilateration Automatic Dependent Surveillance Broadcast (ADS-B).

Each landing trajectory T is defined as a set of spatiotemporal data, i.e. includes GPS position x_k, y_k and the aircraft speed v_k:

$$\begin{aligned} T &= (p_1, \ldots, p_n) \\ &= (\gamma(t_1), \ldots, \gamma(t_n)), \end{aligned}$$

where $p_k = (x_k, y_k, v_k)$ and $\gamma : [0,1] \longrightarrow \mathbb{R}^3$. Note that deceleration values are derived from aircraft speed as a preprocessing step. The dataset consists of 357 landing trajectories.

We have chosen to focus on the runway exit curve only since slippery is mainly visible in curves. Then, following experts, as skidding is a loss of grip with the runway, a degraded runway condition is defined as follows:

- The aircraft deviates from the centreline
- The deceleration value is greater than the runway grip coefficient (which is low when the runway is slippery)

Following this definition, skidding is not only reduced to a lateral deviation of the trajectory. The aircraft deceleration value is indeed used as the indicator of the runway state. A low deceleration level indicates that the runway does not respond to the brakes.

2) Clustering-based bad runway condition detection: The K-medoids clustering method detailed in Section III-B is performed on the Radar tracks data. Two kind of metrics are used: both SRV metrics [25], [26], i.e. without and with alignment, which are not supposed to take into account the deceleration of the aircraft and the slippery metric defined in Section II.

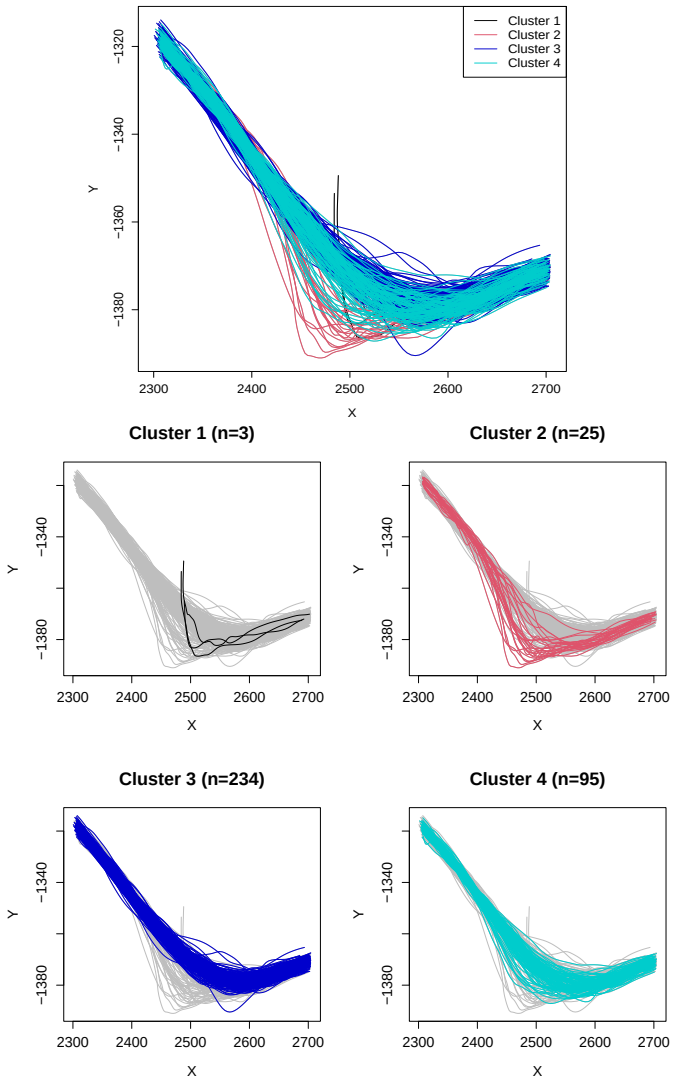


Fig. 9: K-Medoids clustering (SRV metric without alignment).

Figures 9 and 10 show that using both SRV metrics, the trajectories are well clustered by the algorithm relatively to their morphologies. Indeed, cluster 2 contains mostly trajectories with a large lateral deviation, while cluster 3 corresponds to aircraft trajectories following the runway centreline (reference trajectory). Cluster 4 contains mostly standard trajectories even if there is several trajectories with slight lateral deviations. Cluster 1 corresponds to outliers trajectories. Note that, as expected, the SRV metric with alignment is more efficient for clustering trajectories morphologies than the SRV metric without alignment. We have previously seen that the latter metric may detect differences in parametrization that could influence the clustering. On the other hand, these differences in parametrization may be hidden by the shape differences and produce false alarm situations. Moreover, as explained before, if lateral deviations in landing trajectories correspond to abnormal behavior, it does not necessarily represent skidding. In practice, the detection of bad runway conditions has to take into account the deceleration profile (see Section II). It will be

[1]http://www.eurocontrol.int/articles/advanced-surface-movement-guidance-and-control-systems-smgcs

interesting to compare these results with the one obtained by using the slippery metric that is built for taking into account the second-order information.

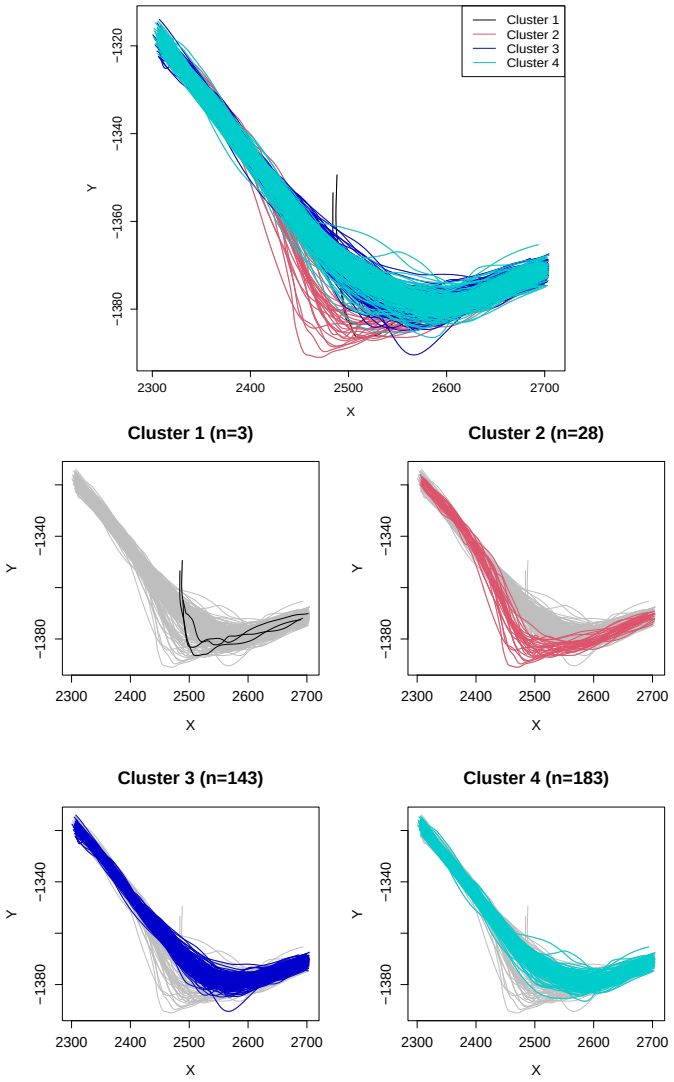


Fig. 10: K-Medoids clustering (SRV metric with alignment).

When adding the information of the deceleration in the metric, the results are slightly different. Indeed, from a geometric point of view, results are quite similar to the one described with the SRVF metric. However, if cluster 2 contains the most deviant trajectories, we observed one deviant trajectory in cluster 4 (Figure 11).

In-depth examination shows that it cannot be considered as a slipped trajectory. Indeed, Figure 12 compares this trajectory and its deceleration profile with the medoid of cluster 2 (deviant trajectories). Again the blue trajectory deviates from the red one but the deceleration profiles are significantly different: the deceleration level of the blue profile is higher than the red medoid deceleration profile. This means that the runway is not slippery, since there is a sufficient grip to obtain a good braking action. From an operational point of view, it seems the aircraft speed was high when arriving on exiting the runway which led on high braking.

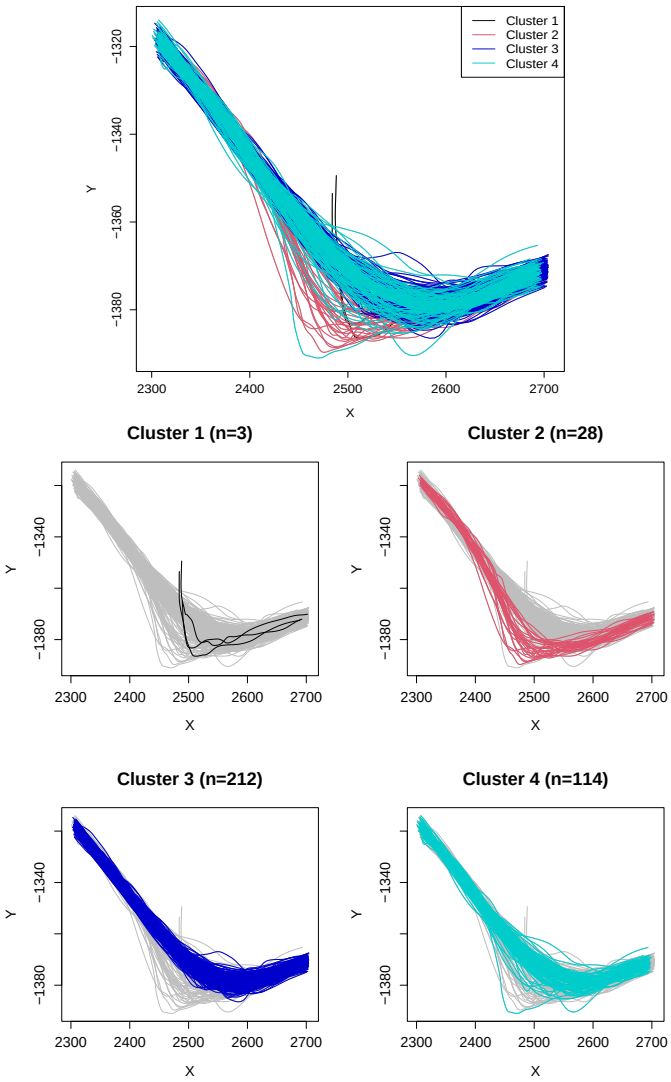


Fig. 11: K-Medoids clustering with the slippery metric.

Briefly speaking, unlike the SRV metric which is only based on the geometric shape, the benefits of the metric with deceleration is twofold: firstly, it detects lateral deviations, and secondly, it removes false positives by taking into account the deceleration part. Finally, an expert study shows that there was 3.7% of slippery alerts in the dataset, among which 92.3% are in cluster 2 (red trajectories). This results confirms the good performance of the proposed metric.

V. Conclusion and Future Work

The ability to detect bad runway conditions without resorting to intrusive on-site measurements will be of major interest both for economic considerations and for passenger experience. In the future, aircraft will be able to downlink their flight and taxi parameters, but until then, only radar tracks may be used. Clustering landing trajectories into skating and normal ones falls within the frame of functional data. However, some extra information must be added to really distinguish between trajectories experiencing slipping and those resulting from late breaking action of the pilot.

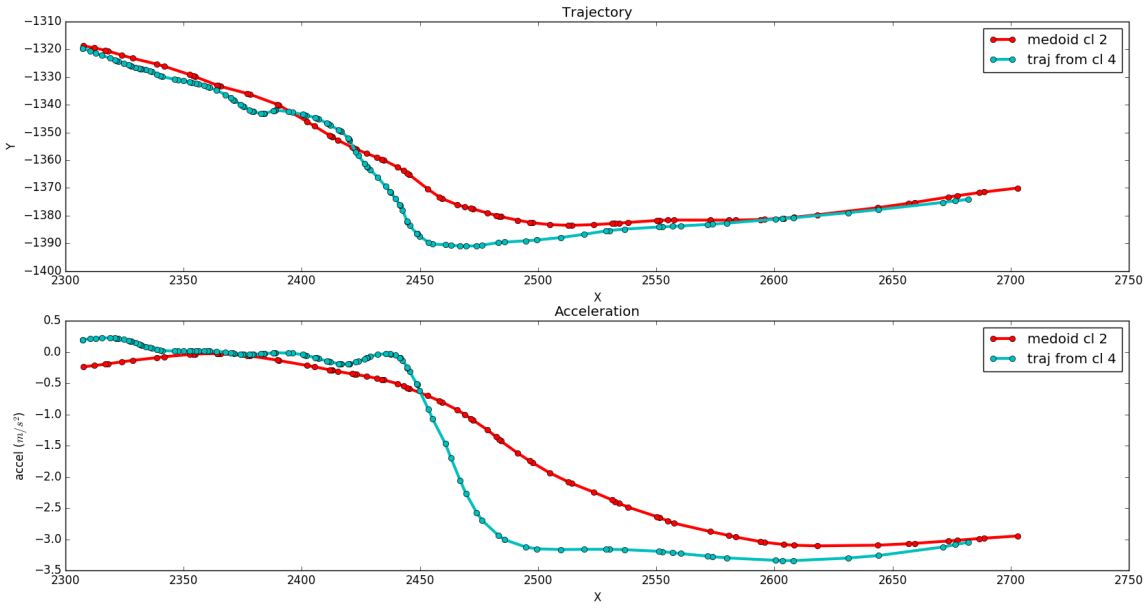


Fig. 12: The most deviant trajectory of cluster 4 (blue), the medoid of cluster 2 (red) and their related deceleration profiles.

In this work, a new distance between curves especially tailored for bad runway adherence conditions has been introduced. Using the slip angle as a measure of skating, a deformation energy is built, which associates the square derivative of the slip angle to an admissible variation of a trajectory. Using the fact that all curves are planar, its expression can be reduced to a simple semi-metric. Finally, its integration along a smooth homotopy between two curves gives an energy. Minimizing it among all possible such homotopies allows the computation of a geodesic path, with associated minimal distance. Using this new metric within a clustering algorithm yields a new methods for clustering landing trajectories that outperforms usual shape-based metrics on both simulated and real data. The cost of distance evaluation is still compatible with offline use of the procedure, in the order of ten minutes for 15,000 pairs of trajectories. It a future implementation, parallel evaluation of the distance matrix will be used, with an expected speed-up close to the number of available compute nodes.

The metric introduced is of Finsler type. The investigation of some properties of it, like its spray, was beyond the scope of the present study which focused on practical applications. It will be for further research.

REFERENCES

[1] Annual Analysis of the EU Air Transport Market. European Commision, 2016. [Online]. Available: https://ec.europa.eu/transport

[2] J. Ramsay and B. Silverman, Functional Data Analysis, ser. Springer Series in Statistics. Springer, 2005. [Online]. Available: https://books.google.co.uk/books?id=mU3dop5wY_4C

[3] 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. [Online]. Available: http://dx.doi.org/10.1007/s11634-011-0095-6

[4] F. Ferraty and P. Vieu, Nonparametric Functional Data Analysis: Theory and Practice, ser. Springer Series in Statistics. Springer New York, 2006. [Online]. Available: https://books.google.fr/books?id=lMy6WPFZYFcC

[5] 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.

[6] M. Boullé, R. Guigourès, and F. Rossi, Advances in Knowledge Discovery and Management: Volume 4. Cham: Springer International Publishing, 2014, ch. Nonparametric Hierarchical Clustering of Functional Data, pp. 15–35. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-02999-3_2

[7] F. Nicol, "Functional principal component analysis of aircraft trajectories," in Proceedings of the 2nd International Conference on Interdisciplinary Science for Innovative Air Traffic Management, Toulouse, France, Jul. 2013. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-00867957

[8] ——, "Statistical Analysis of Aircraft Trajectories: a Functional Data Analysis Approach," in Proceedings of the 3rd International Conference on Big Data, Small Data, Linked Data and Open Data, ser. ALLDATA 2017 Proceedings, Venice, Italy, Apr. 2017, pp. pp.51–56/ISBN: 978–1–61 208–457–2. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-01799104

[9] C. Hurter, S. Puechmorel, F. Nicol, and A. Telea, "Functional Decomposition for Bundled Simplification of Trail Sets," IEEE Transactions on Visualization and Computer Graphics, vol. PP, no. 99, pp. 1 – 1, Aug. 2017. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-01587221

[10] K. Tastambekov, S. Puechmorel, D. Delahaye, and C. Rabut, "Aircraft trajectory forecasting using local functional regression in Sobolev space," Transportation research. Part C, Emerging technologies, vol. 39, pp. pp 1–22, Feb. 2014. [Online]. Available: https://hal-enac.archives-ouvertes.fr/hal-00924360

[11] B. Gregorutti, B. Michel, and P. Saint-Pierre, "Grouped variable importance with random forests and application to multiple functional data analysis," Computational Statistics and Data Analysis, vol. 90, pp. 15 – 35, 2015.

[12] C. Andrieu, "Functional modeling of speed profiles adapted to the infrastructure and methodology of construction of an aggregated speed profile," Theses, Université Paul Sabatier - Toulouse III, Sep. 2013.

[13] A. Srivastava and E. Klassen, Functional and Shape Data Analysis, 01 2016.

[14] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," vol. 10, pp. 359–370, 1994.

[15] M. Vlachos, D. Gunopoulos, and G. Kollios, "Discovering similar multidimensional trajectories," in Proceedings of the 18th International Conference on Data Engineering, ser. ICDE '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 673–. [Online]. Available: http://dl.acm.org/citation.cfm?id=876875.878994

[16] L. Chen, M. Özsu, and V. Oria, "Robust and fast similarity search for moving object trajectories," in Proceedings of the 2005 ACM SIGMOD international conference on Management of data, ser. ACM, 2005, pp. 491–502.

[17] L. Chen and R. Ng, "On the marriage of lp-norms and edit distance," in Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30, ser. VLDB '04. VLDB Endowment, 2004, pp. 792–803. [Online]. Available: http://dl.acm.org/citation.cfm?id=1316689.1316758

[18] F. Hausdorff, Grundz uge der mengenlehre.

[19] M. Fréchet, "Sur quelques points du calcul fonctionnel," vol. 22, pp. 1–72, 1906.

[20] P. C. Besse, B. Guillouet, J.-M. Loubes, and F. Royer, "Review and perspective for distance-based clustering of vehicle trajectories," IEEE Transactions on Intelligent Transportation Systems, vol. 17, no. 11, pp. 3306–3317, Nov 2016.

[21] P. W. Michor and D. Mumford, "Vanishing geodesic distance on spaces of submanifolds and diffeomorphisms." Documenta Mathematica, vol. 10, pp. 217–245, 2005. [Online]. Available: http://eudml.org/doc/125727

[22] M. Bauer, M. Bruveris, and P. W. Michor, "Overview of the Geometries of Shape Spaces and Diffeomorphism Groups," ArXiv e-prints, May 2013.

[23] P. W. Michor and D. Mumford, "An overview of the riemannian metrics on spaces of curves using the hamiltonian approach," Applied and Computational Harmonic Analysis, vol. 23, no. 1, pp. 74 – 113, 2007, special Issue on Mathematical Imaging. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1063520307000243

[24] D. Mumford, Colloquium De Giorgi 2009. Pisa: Scuola Normale Superiore, 2012, ch. The geometry and curvature of shape spaces, pp. 43–53. [Online]. Available: http://dx.doi.org/10.1007/978-88-7642-387-1_4

[25] A. Srivastava, E. Klassen, S. H. Joshi, and I. H. Jermyn, "Shape analysis of elastic curves in euclidean spaces," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 1415–1428, 2011.

[26] A. Le Brigant, "Computing distances and geodesics between manifold-valued curves in the SRV framework," ArXiv e-prints, Jan. 2016.

[27] J. D. Tucker, W. Wu, and A. Srivastava, "Generative models for functional data using phase and amplitude separation," Computational Statistics and Data Analysis, vol. 61, pp. 50 – 66, 2013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167947312004227

[28] "CRAN package fdasrvf," URL: https://cran.r-project.org/web/packages/fdasrvf/index.html.

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

[30] M. Ester, H. P. 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.

[31] 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.

[32] 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.

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

[34] 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.

[35] 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.

[36] "CRAN package funHDDC," URL: https://cran.r-project.org/web/packages/funHDDC/index.html.

[37] L. Kaufman and P. Rousseeuw, Clustering by Means of Medoids, ser. Delft University of Technology : reports of the Faculty of Technical Mathematics and Informatics. Faculty of Mathematics and Informatics, 1987. [Online]. Available: https://books.google.fr/books?id=HK-4GwAACAAJ

[38] ——, Finding Groups in Data: an introduction to cluster analysis. Wiley, 1990.

[39] V. L. Popov, Contact Mechanics and Friction: Physical Principles and Applications. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, ch. Coulomb's Law of Friction, pp. 133–154. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-10803-7_10

[40] R. Rajamani, Vehicle Dynamics and Control, ser. Mechanical Engineering Series. Springer US, 2011. [Online]. Available: https://books.google.fr/books?id=cZJFDox4KuUC

[41] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," Mathematical Programming, vol. 45, no. 1, pp. 503–528, Aug 1989.

[42] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics. University of California Press, 1967, pp. 281–297. [Online]. Available: https://projecteuclid.org/euclid.bsmsp/1200512992

[43] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," Journal of Computational and Applied Mathematics, vol. 20, no. Supplement C, pp. 53 – 65, 1987.

[44] A. L. Brigant, "Geomstats: Shape analysis of curves with the Square Root Velocity metric," URL: https://geomstats.github.io/notebooks/10_practical_methods__shape_analysis.html#.

[45] A. Le Brigant, "A discrete framework to find the optimal matching between manifold-valued curves," Journal of Mathematical Imaging and Vision, vol. 61, 01 2019.

[46] "Geomstats: open-source Python package Geomstat for computations, statistics, and machine learning on nonlinear manifolds," URL: https://geomstats.github.io/index.html.

[47] L. Jones, "Modélisation des forces de contact entre le pneu d'un avion et la piste," Ph.D. dissertation, 2012, thèse de doctorat dirigée par Bes, Christian et Boiffier, Jean-Luc Mathématiques appliquées et systèmes industriels Toulouse, ISAE 2012. [Online]. Available: http://www.theses.fr/2012ESAE0019