



HAL
open science

4D trajectories : a functional data perspective

Stéphane Puechmorel, Daniel Delahaye

► **To cite this version:**

Stéphane Puechmorel, Daniel Delahaye. 4D trajectories : a functional data perspective. DASC 2007, 26th Digital Avionics Systems Conference, Oct 2007, Dallas, United States. pp 1.C.6-1 - 1.C.6-12, 10.1109/DASC.2007.4391832 . hal-00938181

HAL Id: hal-00938181

<https://enac.hal.science/hal-00938181v1>

Submitted on 5 May 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

4D TRAJECTORIES: A FUNCTIONAL DATA PERSPECTIVE

Puechmorel, S. ENAC Dept. MI, Toulouse FRANCE

Delahaye, D. DTI/R&D, Toulouse FRANCE

Abstract

3D or 4D trajectories are fundamental objects within the frame of ATM and can be defined as mappings from a bounded interval to the space R^3 . However, this functional aspect is almost never used in applications. A great improvement over existing procedures for statistical analysis of trajectories can be obtained by explicitly considering trajectory data as functional data. Since raw functional data is by essence infinite dimensional, no computation can be made unless a finite representation has been found: all the complexity of functional data analysis is hidden in this stage. The present paper will first analyze the functional aspect of trajectories in order to extract operationally relevant quantities then will give a way of representing optimally aircraft trajectories with a finite number of parameters, allowing further processing. Applications to trajectory prediction and classification will be briefly discussed at the end.

Introduction

Aircraft trajectory is one of the most fundamental objects within the frame of ATM. However, partly due to the fact that aircraft positions are most of the time represented as radar plots, the time dependence is generally overlooked so that many trajectory statistics conducted in ATM are spatial only. Even in the most favorable setting, with time explicitly taken into account, trajectory data is expressed as an ordered list of plots labeled with a time stamp, forgetting the underlying aircraft dynamics. On the opposite, functional data analysis will work with the whole trajectory as a mapping from a time interval to the state space (most of the time R^3 or R^6 when speed is considered as being part of the aircraft state). There is quite a long history of functional data in statistics [1,2], but to our knowledge no application to the ATM field has been done yet. Furthermore, this way of thinking induces a change of paradigm encompassing more than bare statistics: functional representation makes

an assumption on the underlying phenomenon giving rise to the observed data so that a great wealth of information can be taken into account. As an example, aircraft trajectories are mainly made of straight lines and arcs of circles. Using a linearized model of flight dynamics, such trajectories can be encoded using only piecewise constant mappings (namely the commands applied to the system) so that the whole data can be expressed using only a small number of samples. On the opposite, the collection of radar plots describing the same trajectory can have tenths more samples, nearly all of them redundant. As a consequence, data analysis based on the first representation will use only the salient features of the trajectory while with the second a lot of useless information will be injected. In the present paper, we will analyze first the possible representation for functional data, then some application will be presented.

Air Traffic Management Applications

Regular aircraft trajectory representation (ordered list of plots labeled with time) does not contain typical ATM operational information due to the limited information given by a list of plots. The new representation described in this paper takes into account typical ATM operational features such as curvature and torsion. These relevant features are naturally encoded in this new representation and can be used for turns detection or any other maneuvers. Based on these new trajectories representation many ATM operational applications may be investigated.

Aircraft Trajectories Data Compression

As it has been previously mentioned, ATM systems manage aircraft trajectories and control them in order guaranty safety and airspace capacity. Currently those trajectories are represented by the mean of plot lists which are manipulated by ATM software. Every day, all aircraft trajectories are registered into large database for which huge capacity is needed. Based on this new trajectory

representation for which redundancy has been removed, the trajectories database may be strongly improved from the capacity point of view. As a matter of fact, only spline decomposition coefficients have to be registered in the database then reducing the needed capacity. This compressed trajectory format may also be used for improving the trajectories transmission between ATM entities.

Aircraft Trajectories Distance Computation

Although trajectories are well understood and studied, relatively little investigation on the precise comparison of trajectories is presented in the literature. A key issue in performance evaluation of ATM decision support tools (DST) is the distance metric that determines the similarity of trajectories. This new trajectory representation will enable the definition and fast computation of a rigorous mathematical distance between aircraft trajectories. Based on this new distance computation several ATM applications may be implemented.

Aircraft Model Inference

All aircraft models are based on ODEs(Ordinary Differential Equation), including tabular ones. Control input includes condition and model parameters. The model refinement (and computational complexity) ranges from tabular to many degrees of freedom. The aircraft model inference consists in answering the following question: *Given a parametrized model and a goal trajectory, can we infer the best parameter values?*

A model can be viewed as a mapping from the control space into the trajectory space. The way to answer the previous question is then given by the closest model to the goal trajectory. In order to find the closest model in this trajectory space, a reliable trajectory distance is needed. The model inference problem has to solve the accuracy-smoothness dilemma: *Over-fitted models are generally poor predictors*. The previous construction gives the shortest path (and thus the distance) between the goal trajectory and the trajectory set which can be synthesized by the model. All deformations from the goal trajectory to the trajectory set are taken into account but many of those transformations are not feasible from a physical point of view. The solution consists in defining a smooth path between trajectories by an infinitesimal displacement field and by adding a field smoothness requirement to the

energy functional. On the other hand, if we decide to control the maximum distance between the goal trajectory and a given parametrized model, it is possible to adjust the most economical model to the current phase of flight. As a matter of fact, computation effort is proportional to the degrees of freedom used and to the evolution equation used. Then, the relevant question is the following: *Can we switch from different level of accuracy depending on flight phases and computational power available?* The solution consists in switching from one model of given accuracy to another by following a shortest path in the trajectory space.

Trajectory Prediction

Air traffic management research and development has provided a substantial collection of decision support tools that provide automated conflict detection and resolution [3,4,5], trial planning [6], controller advisories for metering and sequencing [7,8], traffic load forecasting [9,10], weather impact assessment [11,12,13]. The ability to properly forecast future aircraft trajectories is central in many of those decision support tools. As a result, trajectory prediction (TP) and the treatment of trajectory prediction uncertainty continue as active areas of research and development (e.g. [14-18]). Accuracy of TP is generally defined as point spatial accuracy (goal attainment) or as trajectory following accuracy. The last one can be rigorously defined by the mean of trajectory space. The first one is a limit case of the second by adding a weight function in the energy functional. Since we may prescribe smoothness accuracy of a simplified model relative to a finer one, may be computed. Finally such trajectory distance allows comparison between TP models. When we refer to trajectory prediction errors for a specific DST, we are typically comparing the predicted trajectory for a specific DST to the actual trajectory to be experienced by an aircraft. Discrepancies between these two types of trajectories typically affect the performance of the DST.

Radar Tracker Evaluation

The goal of a radar tracker is to eliminate the residual noise coming from the radars. It is a key element of the ATM system and its accuracy is one of the factors which determines the separation norm. In order to validate such trackers, an exact reference trajectory is generated and perturbed by a white gaussian noise. This perturbed trajectory is

then used as input of the tested tracker. The tracker generates an estimated trajectory which is compared to the reference trajectory. In order to do such comparison, a reliable trajectory distance is needed.

Alternative Route Synthesis

Airspace congestion is related to aircraft located in the same area during the same period of time. Then, when congestion has to be minimized, algorithms have to separate aircraft in time (slot allocation), in space (route allocation) or both (bi-allocation). When route allocation is investigated, associated algorithms need alternative routes set in order to spread the traffic on them. A route is said to be alternative to another if it is different enough based on a trajectory distance.

Major Flows Definition

When radar tracks are observed over a long period of time in a dense area, it is very easy to identify major flows connecting major airports. The expression "major flows" is often used but never rigorously defined. Based on an exact trajectory distance and a learning classifier, it is possible to answer the following questions: *Given a set of observed trajectories, can we split it into "similar" trajectory classes?* If yes, classes with highest number of elements will rigorously define the major flows. Given those classes and a new trajectory, can we tell if it belongs to a major flow and which one? The principle of the major flows definition is to use shape space to represent trajectory shapes as points and to use a shape distance (the shape of a trajectory is the path followed by an aircraft, that is the projection in the 3D space of its 4D trajectory. The speed on the path has no impact).

Trajectories as Mappings

In the following all aircraft trajectories will be described as mappings from a time interval $[a, b]$ to a state space E with E either R^3 or R^6 depending on the fact that speed is assumed to be part of aircraft state or not. Extension to trajectories on a sphere (typically long haul flights) will be sketched only.

Notations and Terminology

The reference for this section is [19]. Let $\gamma[a, b] \rightarrow E$ be a trajectory. The origin of the

trajectory is $\gamma(a)$ and the destination is $\gamma(b)$.

Those two points are called the endpoints of the trajectory. All trajectories are assumed to be at least continuously differentiable (class C^1) so that the length of a trajectory $\gamma[a, b] \rightarrow E$ is well defined as:

$$l(\gamma) = \int_a^b \|\gamma'(t)\| dt$$

If $\|\gamma'(t)\| = 0$ for some $t \in (a, b)$ the point t is said to be singular. A parametrized curve of class C^p (or more concisely a C^p curve) will be a C^p mapping from an **open** time interval (a, b) to the state space E with no singular points. Any C^1 curve can be parametrized by arclength. Let $\gamma(a, b) \rightarrow E$ be such a curve. Defining the mapping $s(a, b) \rightarrow (0, l(\gamma))$ by:

$$s(t) = \int_a^t \|\gamma'(t)\| dt$$

we see that by the non singularity assumption on γ , $s'(t) = \|\gamma'(t)\| > 0$ for any $t \in (a, b)$, so that s is an invertible mapping. Now, $\gamma \circ s^{-1}$ is a mapping from the open interval $(0, l(\gamma))$ to E satisfying:

$$\|(\gamma \circ s^{-1})'\| = \|(\gamma' \circ s^{-1}) \circ (s^{-1})'\| = 1$$

In the following, we will simply write $\gamma(s), s \in (0, l(\gamma))$ for a curve parametrized by arclength, dropping the variable t .

Remark 1. One must be careful with the respective definitions of trajectories and curves: a curve is defined on an open interval and thus has no endpoints. Nevertheless, any trajectory $\gamma : [a, b] \rightarrow E$ has an associated curve, namely $\gamma : (a, b) \rightarrow E$. It is generally more convenient to deal with curves to avoid special treatment of the endpoints.

Remark 2. The non singularity assumption on the underlying curve is very natural when dealing with aircraft trajectories in R^3 since it is not

possible for an aircraft to stop except at the endpoints of the trajectory.

Remark 3. While the case $E = R^3$ is very natural and intuitive, care must be taken when $E = R^6$ since all the preceding definitions apply in a completely different setting: for example, the non singularity assumption does not implies nowhere zero speed, but only that speed and acceleration cannot both vanish at the same time.

The arclength parametrization allows to define very important geometrical quantities when $E = R^3$.

Definition 1. Let:

$$\gamma(0, l) \rightarrow R^3$$

be a C^1 curve parametrized by arclength. The unit tangent vector to γ at $s \in (0, l)$ is:

$$\tau(s) = \gamma'(s)$$

It is clear from the definition of parametrization by arclength that $\tau(s)$ is a unit vector.

Definition 2. Let:

$$\gamma(0, l) \rightarrow R^3$$

be a C^2 curve parametrized by arclength. The curvature of γ at $s \in (0, l)$ is:

$$K(s) = \|\gamma''(s)\|$$

The curvature can be explicitly computed even if the curve γ is not parametrized by arclength. The general formula is:

$$K(t) = \frac{\|\gamma'(t) \wedge \gamma''(t)\|}{\|\gamma'(t)\|^3}$$

with \wedge the vector cross product. Curvature is of primary importance for ATM related studies since as mentioned before aircraft trajectories are mainly made of straight lines and arcs of circle and so have piecewise constant curvature. If at point t the curvature is not zero, the curve is said to be biregular at t . For a curve γ parametrized by

arclength, the unit normal vector $\nu(s)$ is defined at all biregular points by:

$$\nu(s) = \frac{\gamma''(s)}{K(s)}$$

Remark 4. A straight line has everywhere zero curvature. However, it is clearly possible to define a unit normal vector.

At a biregular point, $\tau(s)$ and $\nu(s)$ are well defined. Taking their cross product gives a new vector $\beta(s) = \tau(s) \wedge \nu(s)$. If the curve γ is assumed to be C^3 , it can be shown that $\dot{\beta}(s)$ and $\nu(s)$ are collinear:

$$\dot{\beta}(s) = T(s)\nu(s)$$

The real number $T(s)$ is called the torsion of the curve at s and represents an obstruction for the curve to be planar. As for the curvature, it is possible to compute the torsion even if the curve is not parametrized by arclength:

$$T(t) = -\frac{\det(\gamma'(t), \gamma''(t), \gamma'''(t))}{\|\gamma'(t) \wedge \gamma''(t)\|^2}$$

Torsion is not so useful as curvature for en-route data analysis since only a few number of trajectories have non zero torsion. However, it is very relevant in terminal areas.

Remark 5. The $E = R^6$ case is again very different, since the geometric meaning of curvature and torsion is not obvious in this setting. Furthermore, the extra degrees of freedom will impose using higher order derivatives in order to build up an equivalent description. A complete treatment goes beyond the scope of the present paper and has little interest for our purpose (in practical applications, the speed information, when available, is used to improve estimates of curvature and torsion and not to study a trajectory in R^6).

The fact that the last two quantities are nearly piecewise constant functions for aircraft trajectories makes the previous representation very efficient for ATM applications. Most of the time, the relevant assumption will be that the torsion is piecewise

constant, with value 0 in all flight phases except those occurring in terminal areas. Curves with constant torsion and curvature are helices, so that our characterization of aircraft trajectories by piecewise constant curvature and torsion curves will induce a representation by pieces of helices.

All the previous derivations rely on the fact that the first three derivatives of the trajectory are available. In applications however this is nearly never the case since even the speed has to be computed. Moreover, real data are corrupted by noise, making evaluation of derivatives quite cumbersome. To obtain good estimates of curvature and torsion, we will first compute an expansion of the trajectory on an infinite dimensional basis that will be truncated to obtain a finite representation of the trajectory. Since we know that the relevant information is hidden behind curvature and torsion, that is quantities related to first, second and third derivative, the decomposition basis has to be chosen so that those values are easy to compute.

Spline Smoothing

A nice introduction (in French) to spline smoothing applied to aircraft trajectories can be found in a technical report from K. Mehadhebi [20]. In their simplest form, splines are piecewise polynomial approximations to real function, with the extra regularity assumption that the reconstructed mapping has to be of prescribed class C^r . In the case of functions taking their values in R^p , splines will be constructed coordinate-wise. Let $I = [a, b]$ be a real interval and let:

$$a = t_0 < t_1 < \dots < t_N = b$$

be a subdivision of I with $N+1$ points (those points are called the knots of the spline). A polynomial spline on the subdivision is a mapping $S : [a, b] \rightarrow R$ such that for all $i = 0 \dots N-1$,

$S : [t_i, t_{i+1}) \rightarrow R$ is a polynomial P_i . If

$y_i, i = 0 \dots N$ are prescribed values to be taken by the spline at points $t_i, i = 0 \dots N$, then, applying the definition of a spline yields the following condition:

$$\forall i = 0 \dots N-1, P_i(t_i) = y_i$$

Note that the last point y_N is not taken into account since the polynomial P_{N-1} is only defined on a semi-open interval $[t_{N-1}, t_N)$. Assuming now that all polynomials P_i are of the same degree d , the number of coefficients to be estimated is $(d+1)N$. The interpolation condition gives N relations so that dN coefficients remain to be estimated. Since we want some smoothness of the spline, we can impose conditions on the inner points:

$$\forall i = 1 \dots N-1, P_{i-1}^{(k)}(t_i) = P_i^{(k)}(t_i)$$

with k ranging from 0 (continuity) to the requested regularity degree. A common choice is to have C^{d-1} splines, in which case smoothness conditions will give $d(N-1)$ relations. Gathering interpolation and smoothness conditions yield a total of $(d+1)N - d$ relations. To have a fully determined system, d relations are still needed. Classically, those will be obtained by fixing the values of the first $d-1$ derivatives of P_{N-1} at t_N (the 0-th order derivative is the polynomial itself and so this conditions can be used to explicitly take the value y_N into account). Although it is possible to compute an interpolating spline based on the previous derivations, it is more enlightening to take a different approach for trajectories representation. First, one can note that a 0-th order spline is indeed a piecewise constant function and thus is entirely defined by the interpolation data (no smoothness is required). It is then clear that the interpolating spline can be written as a weighted combination of elementary piecewise constant functions:

$$S(t) = \sum_{i=0}^{N-1} y_i 1_{[t_i, t_{i+1})}(t)$$

In the general case, we will seek for a decomposition on basis functions that are minimally compactly supported splines. For regularity d and a spline function made from N polynomials, smoothness requirement will induce $d(N+1)$ relations. The compact support assumption implies that the derivatives of order $0 \dots d-1$ vanish at endpoints, thus adding $2d$ relations. Finally, the

spline is required to take a non zero value at some point so that another relation is added for a net total of $d(N+1)+1$. Since the number of free coefficients is $(d+1)N$, a minimal spline will be obtained by equating this number with the number of relations, giving $N = d+1$. The basis functions can be recursively computed by the Cox-De boor algorithm:

$$\psi_{j,0}(t) = 1_{[t_j, t_{j+1})}(t)$$

$$\psi_{j,k}(t) = \frac{t-t_j}{t_{j+k}-t_j} \psi_{j,k-1}(t) + \frac{t_{j+k+1}-t}{t_{j+k+1}-t_{j+1}} \psi_{j+1,k-1}(t)$$

The general expression of a spline of smoothness d will thus be:

$$S(t) = \sum_{j=0}^{N-1-d} \alpha_j \psi_{j,d}(t)$$

with the weights α_j determined by interpolation condition (note that it may be necessary to add extra node in order to meet the condition). While it is possible with this model to efficiently model an aircraft trajectory and to compute any derivatives using the recursion formula, a major drawback is that the spline function has to exactly interpolate the prescribed values, which is not relevant when data is corrupted by noise. A way of dealing with such data is to seek for a minimal square error. Namely, assuming that the smoothing spline has the form:

$$S(t) = \sum_{j=1}^P \alpha_j \psi_{j,d}(t)$$

then the optimal set of coefficients $(\alpha_j)_{j=1 \dots P}$ will minimize the square error:

$$\sum_{i=1}^N (y_i - S(t_i))^2$$

Note that the number of elementary splines has no longer to be directly related to the number of prescribed values. An elementary computation yields the following equation for the coefficients:

$$\sum_{j=1}^P \alpha_j \left(\sum_{i=1}^N \psi_{k,d}(t_i) \psi_{j,d}(t_i) \right) = \sum_{i=1}^N \psi_{k,d}(t_i) y_i$$

or in a more compact matrix form:

$$G\alpha = R$$

with G the matrix:

$$G_{jk} = \sum_{i=1}^N \psi_{j,d}(t_i) \psi_{k,d}(t_i)$$

R the vector:

$$R_j = \sum_{i=1}^N \psi_{j,d}(t_i) y_i$$

and $\alpha = (\alpha_1, \dots, \alpha_P)$. G is a gram matrix and will be definite positive if the vectors:

$$\Psi_j = (\psi_{j,d}(t_1), \dots, \psi_{j,d}(t_N)), j = 1 \dots P$$

are linearly independent. In this case, $\alpha = G^{-1}R$. Computation can be made efficient by pre-computing the gram matrix and storing its Cholesky decomposition. The overall algorithm for smoothing spline construction is the following:

Selection of sample times for spline building using Cox-De boor recursion. Since elementary splines are piecewise polynomial functions, it is possible to pre-compute the coefficients of those polynomials, thus avoiding recursion for subsequent functions evaluations.

Computation of the gram matrix G for the sample times on the trajectories. Sample times used in spline definition will be obtained by downsampling those times.

Computation of R and linear system $G\alpha = R$ solving.

Following [21], smoothing splines can be defined in the setting of reproducing kernel Hilbert spaces (RKHS). Assuming that the measures obey the following model:

$$y_i = \phi_i \gamma + \varepsilon_i$$

with γ an unknown mapping to be approximated, ϕ_i a **continuous** linear functional (most of the time, ϕ_i is the evaluation functional at time t_i , that is $\phi_i \gamma = \gamma(t_i)$). Since ϕ_i is continuous, by the Riesz representation theorem it exists a vector v_i such that for any f :

$$\phi_i f = \langle f, v_i \rangle$$

Assuming that all vectors lie in a RKHS with reproducing kernel K , the following relation holds:

$$\begin{aligned}\phi_i K(t, \cdot) &= \langle K(t, \cdot), v_i(\cdot) \rangle \\ &= v_i(t)\end{aligned}$$

So that finally, for any f :

$$\phi_i f = \langle f(t), \phi_i K(t, \cdot) \rangle$$

This important property allows us to replace the evaluation of a bounded linear functional by an inner product: it will be the key ingredient in establishing the general smoothing spline formula. For reason that will become clear in the following, the initial RKHS will be written as direct sum of two orthogonal subspaces:

$$H = H_0 \oplus H_1$$

with H_0 a finite dimensional subspace, spanned by basis vectors (ψ_1, \dots, ψ_p) and H_1 a RKHS with reproducing kernel K_1 . A smoothing spline will be a vector $S \in H$ realizing the minimum of the functional E :

$$E(\gamma) = \sum_{i=1}^n (y_i - \phi_i \gamma)^2 + \mu \|P_{H_1} \gamma\|_{H_1}^2$$

with μ a positive real number, P_{H_1} the orthogonal projection on H_1 and the ϕ_i be bounded linear functionals as before. μ is used to tune the trade off between interpolation (first sum in the expression of E) and smoothness (related to the square norm appearing in E). By writing $\gamma = \gamma_0 + \gamma_1$ with $\gamma_0 \in H_0, \gamma_1 \in H_1$ and using the representation property for RKHS:

$$\phi_i \gamma = \phi_i \gamma_0 + \langle \gamma_1, \phi_i K_1 \rangle_{H_1}$$

Furthermore, since by assumption H_0 admits a finite basis:

$$\phi_i \gamma_0 = \sum_{j=1}^p \alpha_j \phi_i \psi_j \quad \text{if } \gamma_0 = \sum_{j=1}^p \alpha_j \psi_j$$

It thus makes sense to define:

$$\theta_i(t) = \phi_i K_1(\cdot, t), \eta_{ij} = \phi_i \psi_j$$

E can be rewritten as:

$$E(\gamma) = \sum_{i=1}^n \left(y_i - \sum_{j=1}^p \alpha_j \eta_{ij} - \langle \gamma_1, \theta_i \rangle \right)^2 + \mu \|\gamma_1\|_{H_1}^2$$

Putting:

$$e_i = y_i - \sum_{j=1}^p \alpha_j \eta_{ij} - \langle \gamma_1, \theta_i \rangle, i = 1 \dots n$$

and taking partial derivatives with respect to the $\alpha_k, k = 1 \dots p$ shows that for the optimal γ :

$$\sum_{i=1}^n \eta_{ik} e_i = 0, k = 1 \dots p$$

Using again differential calculus in the Hilbert space H_1 we have at the optimum:

$$\forall \eta \in H_1, \langle \eta, \mu \gamma_1 - \sum_{i=1}^n \theta_i e_i \rangle = 0$$

so that:

$$\gamma_1 = \mu^{-1} \sum_{i=1}^n \theta_i e_i$$

Gathering the two parts, the optimal γ has an expansion:

$$\gamma = \sum_{j=1}^p \alpha_j \psi_j + \sum_{j=1}^n \beta_j \theta_j$$

This result is known as the representer theorem [21]. Plugging the expansion of the optimal γ in the expression of E yields:

$$E(\gamma) = \|y - \Sigma \alpha - G \beta\|^2 + \mu \|G \beta\|^2$$

With $y = (y_1, \dots, y_n)$, $\alpha = (\alpha_1, \dots, \alpha_p)$, $\beta = (\beta_1, \dots, \beta_n)$ and the matrices:

$$\Sigma = (\eta_{i,j}), G = (\langle \theta_i, \theta_j \rangle)$$

If the Gram matrix G is full rank, then a standard least square computation shows that the coefficient vectors α, β satisfy the equations:

$$\begin{cases} \Sigma' \beta = 0 \\ y = \Sigma \alpha + (G + \mu Id) \beta \end{cases}$$

Specific routines have been developed for solving such linear systems. In our setting some modification to the original work has to be done. First of all, since our data is three dimensional, we need to replace the ϕ_i in the previous expression by

bounded linear mappings to R^3 , still denoted by ϕ_i . The energy functional to be minimized becomes:

$$E(\gamma) = \sum_{i=1}^n \|\gamma_i - \phi_i \gamma\|^2 + \mu \|P_{H_1} \gamma\|_{H_1}^2$$

The reproducing property is now:

$$\forall v \in R^3, \langle \gamma(t), v \rangle_{R^3} = \langle \gamma, K(t, v, \cdot) \rangle_H$$

with the kernel K a mapping with values in vector space of endomorphisms of R^3 . Since K is linear in v , it is convenient to use the notation $K(t, \cdot)v$ instead of $K(t, v, \cdot)$. All the previous derivations remain valid in this setting, and the representer theorem applies verbatim, except for the fact that the expansion of the optimal γ now involves vector coefficients. Putting $\theta_i(t)(v) = \phi_i K(t, v, \cdot)$ gives a mapping with values in endomorphisms of R^3 . The optimal γ has expansion:

$$\gamma(t) = \sum_{j=1}^p \alpha_j \psi_j(t) + \sum_{j=1}^n \theta_j(t)(\beta_j)$$

with $\beta_j \in R^3$.

To obtain a RKHS for our problem, we will assume that all trajectories are defined on the same time interval $[0,1]$. The Sobolev space $W^m([0,1])$ is the space of all mappings from $[0,1] \rightarrow R^3$ whose derivatives up to order m exist (in the weak sense) and are square summable. Let L be a differential operator of order m . Then:

$$W^m([0,1]) = Ker L \oplus W_1^m$$

with W_1^m a RKHS (in the previous sense). Furthermore, the Green function of L gives immediately access to the reproducing kernel of W_1^m . Recalling that our purpose is to obtain piecewise constant curvature and torsion, the choice of L will be made in order to satisfy as well as possible this requirement. Ideally, L should be the derivative of the curvature (or the torsion), but unfortunately, since the expression of the curvature is nonlinear, an optimal L cannot be found. The standard way of dealing with the problem is to

require only that the square norm of the third derivative be as small as possible, thus giving:

$$L = \begin{pmatrix} D^3 & 0 & 0 \\ 0 & D^3 & 0 \\ 0 & 0 & D^3 \end{pmatrix}$$

with D the operator of differentiation with respect to time. Simple computation shows that this particular choice gives piecewise polynomial splines of degree 5 on each of the three components: all the results previously given will apply. Note that many standard spline smoothing algorithms use instead cubic splines that correspond to an approximate minimum total curvature. Because of the approximation made on the criterion, curves reconstructed by quintic splines will not be of piecewise constant curvature. In order to have a better approximation, it is convenient to add a penalty term so that the norm of the velocity be as constant as possible: in this case, curves tend to be parametrized by arclength, thus giving a better approximation of curvature by the second derivative. The corresponding composite criterion is:

$$L + \mu \begin{pmatrix} D & 0 & 0 \\ 0 & D & 0 \\ 0 & 0 & D \end{pmatrix}$$

with $\mu > 0$ a positive real number controlling the importance paid to constant velocity. Functions optimal with respect to this new criterion have been intensively studied in the literature (at least for the cubic case) and are named splines under tension (μ is the tension parameter). Increasing μ results in a curve with sharper turns and more straight line segments, while setting $\mu = 0$ gives the initial quintic (or cubic) splines. As it has been pointed out by K. Mehadehbi, it is interesting to make μ depend on time, so that it is possible to more accurately model turns by relaxing tension while low curvature areas will be associated to high tension, yield almost straight lines which is operationally sound. The choice of the tension parameter remains anyway a highly heuristic process. In order to more accurately take into account the real curvature and more generally the

shape of the trajectory to be reconstructed, a different approach combining a shape optimization and a velocity interpolation will be taken. Recall that shape parameters like curvature or torsion are easily defined only when the curve is parametrized by arclength. Assuming that such a parametrization has been obtained, the interpolation criterion has to be modified: with time parametrization, measured position y_i at time t_i has to be as close as possible to the smoothed value $\gamma(t_i)$ while with arclength parametrization, there is no direct relationship between measurement y_i and the corresponding $\gamma(s(t_i)) = \gamma(s_i)$. The only way of dealing with the problem is to require that $\gamma(s_i)$ be a point on the curve closest to y_i . Unfortunately, such a point is generally not unique (take for example as γ a circle and as y_i the center of this circle), and to correct this flaw, we have to add an extra assumption forcing uniqueness. A classical choice is to select s_i as the supremum of the set of those s for which $\gamma(s)$ is closest to y_i . The assumption on s_i insuring uniqueness will be implicit now, but has to be kept in mind when implementing algorithms.

The smoothing criterion to be minimized by the smoothing curve γ will thus be of the form:

$$\sum_{i=1}^n \min_s \|y_i - \gamma(s)\|^2 + \mu \|P\gamma\|^2$$

The optimal γ minimizing this criterion is known as a regularized principal curve (see [22]).

If $\|P\gamma\|^2$ has to be rotation invariant (meaning that curves that can be obtained by rotations of the same curve will be equally penalized), it can be proved using Schur lemma that $P^t P$ has to be scalar, precluding the use of vector valued kernels.

Remark 6. Since most of the relevant P for aircraft trajectory studies are diagonal, this restriction has little impact: interpolation can be done with scalar kernels component-wise. Furthermore, computational speed is greatly increased when using such kind of kernels: a speed-up of 9 is theoretically possible since the complexity of linear algebra algorithms used to find

the coefficients of spline expansion is $o(N^3)$ with N the dimension of the matrices involved. With general vector kernels, this amounts to a complexity of order $3^3 = 27$ while the diagonal case reduces to 3 independent resolutions of one dimensional problems, thus yielding an overall complexity of order 3.

In [22], it is proposed to find a solution to the principal curve problem by assuming that all curves γ have an expansion as a weighted sum of a reproducing kernel:

$$\gamma^j(s) = C^j + \sum_{i=1}^m \alpha_i^j K(s_i, s), j = 1 \dots 3$$

with C^j real constants and K a reproducing kernel satisfying $K(x, y) = \langle PK(x, \cdot), PK(y, \cdot) \rangle$ (K can be easily obtained as the green function of the operator $P^t P$). In this setting, the kernel of $P^t P$ has to be the space of constant functions, but extension to different finite dimensional kernels is immediate by replacing the constants C^j with mappings from the kernel of $P^t P$. It is very important to note that contrary to the standard splines case, the nodes positions s_i are now free parameters of the problem that must be chosen prior to performing curve fitting. Putting:

$$\alpha_i = (\alpha_i^1, \alpha_i^2, \alpha_i^3)$$

and assuming that the data has been de-trended (so that the term corresponding to the mapping from the kernel of $P^t P$ is 0), the regularized principal curve associated with the measured positions y_i is obtained as the solution of the following optimization program:

$$\min_{(\alpha_i), (\omega_j)} \sum_{j=1}^m \left\| y_j - \sum_{i=1}^n \alpha_i K(s_i, \omega_j) \right\|^2 + \mu \sum_{i,j=1}^n \langle \alpha_i, \alpha_j \rangle K(s_i, s_j)$$

Because of the minimization on the parameters (ω_j) , it is no longer possible to use a standard solver and an iterative algorithm has to be considered. [22] propose to first minimize over the

(ω_j) , then solve in a second step for the (α_i) , iterating those two stages until no further improvement is obtained. Theoretical analysis of convergence rate has been done in the same paper, showing that a nearly optimal learning rate can be obtained. As in the smoothing splines case, we will enforce arclength parametrization by adding an extra term in $\|\dot{\gamma}\|^2$: the resulting kernel will be associated with splines in tension.

Once the optimal shape of the trajectory has been obtained by the previous algorithm, it remains to infer a parametrization matching the really observed 4D trajectory. Such a parametrization is a monotonic real mapping: it will be obtained again using a spline interpolation, but this time using a well known cubic spline interpolation. It must be noted that monotonic splines exist, thus insuring that the parametrization has by construction the required properties, however, on practical cases, it appears that the simple cubic splines algorithm produces admissible results with a much lower computational cost. For that reason, only this last option has been considered in our final implementation of spline smoothing.

Using Functional Data

Once a tractable (i.e. finite dimensional) representation of the original functional data has been obtained, it is relatively easy to extend classical vector algorithms to the functional framework. For the purpose of aircraft trajectory analysis, two applications have been investigated:

- Trajectory smoothing and prediction.
- Trajectory classification.

We will now briefly explain how the use of functional data representation can improve over the existing procedures. Trajectory smoothing application is obvious, since functional representation of sampled data is obtained by solving a smoothing problem. For prediction, short term forecasting can be again directly obtained by evaluating the functional representation at the desired time. However, smoothing splines, either standard or shape based are generally poor predictors for medium to long term forecasting:

some knowledge about the future behavior of the trajectory has to be incorporated to the model. Some information from flight plan can be added is available by considering those intended position as measures, but with a low confidence level (this is obtained simply by weighting the correspond error term in the smoothing problem). Since 4D positions are generally badly defined in flight plans, it is far better to use the regularized principal curve algorithm first, then to use the 1D cubic spline smoothing to infer the time/arclength mapping, with a low confidence on values obtained from the flight plan). For long term forecasting, only statistical procedures are relevant. Since it amounts to finding the most probable trajectory based on observation, it is in fact based on the next item that is trajectories classification.

Classification of trajectories is a statistical procedure aimed at splitting a dataset into subsets with high internal coherence and at finding a best representing member for each of those subsets. Many algorithms exist for performing this task and fall in one of those two classes:

- Supervised classification, where the number of classes is known and an already classified learning dataset is available.
- Unsupervised classification where the classes are inferred by the algorithm.

Supervised classification can be readily applied in the functional context since almost all algorithms require only a measure of similarity between data. In our setting, such a measure can be obtained by computing Sobolev distances between functional data (for operational purpose, an interesting distance is obtained by summing the L^2 norm of the differences of trajectories, derivative of trajectories and second derivatives of trajectories). For the unsupervised case, most algorithms include a heuristic step: we have not yet investigated such algorithms for the purpose of aircraft trajectory analysis.

Conclusion and Future Work

Because trajectories are functional objects it is highly desirable to stay within this context as long

as possible. Since existing statistical procedure within the frame of ATM are nearly all the time based on sampled data, a way of optimally inferring a functional model has been presented. Since relevant operational quantities are related to the shape of trajectories, it is interesting to first model the data as a regularized principal curve, then to fit an ad-hoc parametrization to match the 4D aspect. This algorithm improves over the smoothing splines or splines in tension algorithms already in use. Furthermore, it is much easier to enforce piecewise constant curvature or torsion using shape models. The price that has to be paid for this extra flexibility is a more computationally intensive procedure, involving an iterative optimization stage. Taking into account the computation power of nowadays computers, it turns out that computational time penalty compared with standard smoothing splines is acceptable. A extension of the shape algorithm using an optimal tension parameter is currently under development. At the same time, a dataset of arrival trajectories at Paris CDG will be analyzed using the functional data approach.

References

- [1] Ramsay, J.O., Silverman, B.W., 1997, Functional Data Analysis, Springer-Verlag .
- [2] Ramsay, J.O., Silverman, B.W., 2002, Applied Functional Data Analysis, Springer-Verlag .
- [3] Erzberger H. , R.A. Paielli , D.R. Isaacson , M.M. Eshowl, 1997, Conflict Detection in the Presence of Prediction Error, Saclay France, Proceeding of FAA/Eurocontrol ATM Seminar.
- [4] Brudnicki D.J. , A.L. McFarland, 1997, User Request Evaluation tool (URET) Conflict Probe Performance and Benefits Assessment, Saclay France, Proceeding of FAA/Eurocontrol ATM Seminar.
- [5] Vink A., 1997, EATCHIP Medium Term Conflict Detection: Part 1 EATCHIP Context, Saclay France, Proceeding of FAA/Eurocontrol ATM Seminar.
- [6] McNally B.D. , R.E. Bach , W. Chan, 1998, Field Test Evaluation of the CTAS Conflict Prediction and Trial Planning Capability, Boston MA, Proceeding of the AIAA-1998-4480 AIAA GNC Conference.
- [7] Swensen H.N. et al, 1997, Design and Operational Evaluation of the Traffic Management Advisor at the Forth Worth Air Route Traffic Control Center, Saclay France, Proceeding of FAA/Eurocontrol ATM Seminar.
- [8] Coppenbarger R. , R. Lanier , D. Sweet , S. Dorsky, 2004, Design and Development of the EnRoute Descent Advisor (EDA) for Conflict-Free Arrival Metering, Providence RI, Proceeding of the AIAA-2004-4875 AIAA GNC Conference.
- [9] Meckiff C. , R. Chone , J.P. Nicolaon, 1998, The Tactical Load Smoother for Multi-Sector Planning, Orlando FL, Proceeding of FAA/Eurocontrol ATM Seminar.
- [10] Masalonis, A et al, 2004, Using Probabilistic Demand Prediction for Traffic Flow Management Decision Support, Providence RI, Proceeding of the AIAA-2004-5231 AIAA GNC Conference.
- [11] Kirk D.B. et al, 2004, Problem Analysis Resolution and Ranking (PARR) Development and Assessment, Santa Fe NM, Proceeding of FAA/Eurocontrol ATM Seminar.
- [12] Sud V. et al, 2001, Air Traffic Flow Management Collaborative Routing Coordination Tools, Montreal PQ, Proceeding of the AIAA-2001-4112 AIAA GNC Conference.
- [13] Evans J. et al, 2003, Reducing Severe Weather Delays in Congested Airspace with Weather Support for Tactical Air Traffic Management, Budapest Hungary, Proceeding of FAA/Eurocontrol ATM Seminar.
- [14] Slattery R. , Y. Zhao, 1997, Trajectory Synthesis for Air Traffic Automation, AIAA Journal Guidance Control and Dynamics, Vol 20-2, pp 232-238.
- [14] Swierstra S. and Green S., 2003, Common Trajectory Prediction Capability for Decision Support Tools, Budapest Hungary, Proceeding of FAA/Eurocontrol ATM Seminar.
- [16] S. Mondoloni and S.M. Pagli one and S. Green, 2002, Trajectory Modeling Accuracy for Air Traffic Management Decision Support Tools, Toronto ON, Proceeding of the ICAS Conference.
- [17] H.F. Ryan and M. Paglione and S. Green, 2004, Review of Trajectory Accuracy Methodology and Comparison of Error Measure Metrics,

Providence RI, Proceeding of the AIAA-2004-4787
AIAA GNC Conference.

[18] S.Mondoloni and I. Bayraktuta, 2005, Impact
of Factors, Conditions and Metrics on Trajectory
Prediction Accuracy, Baltimore MD, Proceeding of
FAA/Eurocontrol ATM Seminar.

[19] Berger, M. Gostiaux, B., 1988, Differential
geometry: Manifolds, Curves and Surfaces,
Springer-Verlag.

[20] Mehadhebi, K. Roux, S., 2002, Algorithme de
Spline à Rigidité Variable: Application au Lissage

de Trajectoires Radar, Technical Report N° NT02-
250, CENA-MSS.

[21] Wahba, G., 1990, Spline Models for
Observational Data, SIAM.

[22] Smola, A.J, Williamson, R.C, Mika, S.,
Schölkopf, B., 1999, Regularized Principal
Manifolds, Springer-Verlag, LNAI 1572, pp. 214-
229 .

*26th Digital Avionics Systems Conference
October 21, 2007*