

# Hybrid clustering/HMM constrained-based learning for Aircraft Trajectory Prediction

Harris Georgiou <sup>#1</sup>, Nikos Pelekis <sup>#2</sup>, David Scarlatti <sup>\*3</sup>, Stylianos Sideridis <sup>#4</sup>, Yannis Theodoridis <sup>#5</sup>

<sup>#</sup> *Data Science Lab, University of Piraeus  
Piraeus, Greece*

<sup>1</sup> hgeorgiou@unipi.gr

<sup>2</sup> npelekis@unipi.gr

<sup>4</sup> ssider@unipi.gr

<sup>5</sup> ytheod@unipi.gr

<sup>\*</sup> *Boeing Research & Technology Europe  
Madrid, Spain*

<sup>3</sup> David.Scarlatti@boeing.com

**Abstract**—Aircraft trajectory prediction (TP) is a challenging and inherently data-driven time-series modeling problem. Adding annotation parameters further increases the complexity of the search space, especially when ‘blind’ optimization algorithms are employed. In this paper, flight plans, localized weather and aircraft properties are introduced as trajectory annotations (or semantics), which enable modeling in a space higher than the typical 4-D spatio-temporal domain. A two-phase hybrid approach is employed for the core TP task: (a) clustering using properly designed semantic-aware similarity functions as distance metrics; and (b) a hidden Markov model (HMM) for each cluster, using non-uniform graph-based spatial grid and exploiting flight plans as constraints for a parametric probabilistic model for the emissions. The proposed method is applied in real radar tracks and weather data for a one-month dataset of flights in Spanish airspace. Using parametric Gaussians as the base for the emissions model and confidence interval estimations for the associated errors, the proposed method exhibits exceptionally low HMM complexity and per-waypoint prediction accuracy of a few hundred meters compared with submitted flight plans.

## I. INTRODUCTION

The increasing use of portable devices, such as navigation systems, and the wide range of location-aware applications has led to a huge amount of mobility data being produced daily. As a result, a plethora of research issues has emerged in order to manage or analyze such data. One of the most challenging data analytics tasks is to transform data to actionable knowledge by means of exploiting historical mobility patterns in order to gauge what the moving entities may do in the future [1] [2] [3].

The problem of predictive analytics over mobility data in the aviation domain involves applications where aircrafts are traced in real-time in order to compute e.g. short- or long-term predictions. Short-term prediction, which is time-critical and asks for immediate response, facilitates the efficient planning, management, and control procedures, while assessing traffic conditions in the air transportation field. The latter is extremely important as safety, credibility and cost are critical and a decision should be taken by considering adversarial to

the environment conditions in order to act immediately. On the other hand, long-term prediction enhances current plans to achieve cost efficiency or, when contextual information is provided (e.g., weather conditions), to ensure public safety. In the future of the Air Traffic Management (ATM), trajectories will be used as the core component of many procedures [4].

Recently, there has been plenty of work on location and trajectory prediction in the mobility, and especially in the aviation domain. Swierstra and Green [5] provide a system engineering approach for investigating important design issues and tradeoffs, such as the balance between TP accuracy and computational speed. Regarding en route climb TP, one of the major aspects of ATM decision support tools, Coppensbarger [6] discusses the exploitation of real-time aircraft data, such as aircraft state, aircraft performance, pilot intent and atmospheric data for improving ground-based TP. The problem of climb TP is also discussed in Thipphavong et al. [7], as it constitutes a very important challenge in ATM. In that work, an algorithm that dynamically adjusts modeled aircraft weights is developed, exploiting the observed track data to improve the accuracy of TP for climbing flights. Real-time evaluation with actual air traffic data shows a significant improvement on the prediction of the trajectory altitude, as well as the time to reach the top-of-climb.

Instead of following the typical approach operating on the raw data collected from various sensors, in this work we make use of semantically enhanced data, which are inferred by some enrichment and/or annotation method. This way, raw trajectory data is transformed into multidimensional sequences (semantic trajectory data) that form a more realistic representation model of the complex every-day life; mobility of aircrafts belongs to this broad class.

More specifically, in this paper we define the semantic-aware variation of the TP problem: in the so-called *Future Semantic Trajectory Prediction* (FSTP) problem, starting from a base point, we aim to predict various intermediate semantically interesting positions, until a target point or region is reached. For instance, given the flight plan submitted by an airline for a specific flight, we make a prediction of the

4-D spatio-temporal flight trajectory from departure to destination airport, including its important intermediate semantics: ‘top-of-climb’, ‘top-of-descent’, etc.

Our methodology exploits on clustering and HMM tasks. In particular, the merits and contributions of our work are summarized as follows:

- we define the FSTP problem and propose a novel clustering and model-based methodology to effectively tackle it;
- we devise a novel HMM-based representation for semantic trajectories, for indexing purposes in order to effectively and efficiently perform  $k$ -NN search in the semantic trajectory database;
- we provide an extensive empirical study over real aircraft-related data (radar tracks, flight plans, weather, etc.).

The rest of the paper is organized as follows: Section II presents the related work and Section III provides the FSTP problem formulation; Section IV presents our framework for FSTP purposes; Section V presents our experimental study; Section VI provides additional analysis and discussion on the results; finally, section VII concludes the paper and points out some interesting research directions for future work.

## II. RELATED WORK

During the last few years, there is a mainstream trend of using stochastic models for retrieval, with HMM approach being the most popular [8] as it has proved its efficiency in modeling a wide range of sequences of observations. In general terms, a system is assumed to have the Markovian property if its future situations depend only on its current situation and not on the old ones. Exhibiting high accuracy in modeling sequential data, Markovian assumption has given rise to a wide range of extensions of Markov chains. The HMM approach models the evolution of a system by a set of states and transitions between them, each one accompanied by a probability that is typically extracted by analyzing historic data. HMMs have been successfully applied in various domains, such as speech recognition, music retrieval, human activity recognition, consumer pattern recognition and in many other domains. Consequently, it is a clear opportunity to apply them in the domain of mobility data analysis.

In the context of trajectory prediction, the flight route and all the associated information (weather, semantic data, etc), are encoded into discrete values that constitute the HMM states; then, the trajectory itself is treated as an evolution of transitions between these states, using the raw trajectory data of a large set of flights for training, plus spatio-temporal constraints (locality) to reduce the dimensionality of the problem. Ayhan and Samet [9] introduce a novel stochastic approach to aircraft trajectory prediction problem, which exploits aircraft trajectories modelled in space and time by using a set of spatio-temporal data cubes. They represent airspace in 4-D joint data cubes consisting of aircraft’s motion parameters (i.e., latitude, longitude, altitude, and time) enriched by weather conditions. They use Viterbi algorithm [10] to compute the most likely sequence of states derived by

a HMM, which has been trained over historical surveillance and weather conditions data. The algorithm computes the maximal probability of the optimal state sequence, which is best aligned with the observation sequence of the aircraft trajectory. In their experimental study, they demonstrate that their methodology efficiently predicts aircraft trajectories by comparing the prediction results with the ground truth aligned trajectories, with the error being reasonably low for one-hour flights (it resides within the boundaries of the highest spatial resolution, 8-13 km). In another work by Ayhan and Samet [11], the authors investigate the applicability of the HMM for TP on only one phase of a flight, specifically the climb after takeoff. A stochastic approach such as the HMM can address the TP problem by taking environmental uncertainties into account and training a model using historical trajectory data along with weather observations. In that work, a time series clustering algorithm is employed to generate an optimal sequence of weather observations, employing  $k$ -Nearest Neighbors ( $k$ -NN) search using Dynamic Time Warping (DTW) Euclidean distance. The results show robust performance and high TP accuracy, proving that HMM can be applied equally well for single-phase prediction, as well as complete-flight prediction.

## III. PROBLEM FORMULATION & OVERVIEW

In this section, we provide preliminary definitions and formulate the problem in hand. A list of symbols used in the current as well as the sections that follow appears in Table I.

The (raw) trajectory  $T$  of an aircraft is defined as a 4-D polyline consisting of a sequence of  $|T|$  pairs  $(p_i, t_i)$ ,  $i = 0, \dots, |T|-1$ , where  $p_i$  is a 3-D point  $(x_i, y_i, z_i)$  in the 3-D space and  $t_i$  is a timestamp, assuming linear interpolation between two consecutive pairs  $(p_i, t_i)$  and  $(p_{i+1}, t_{i+1})$ .  $T$  can be partitioned into a sequence of (raw) sub-trajectories; formally, a (raw) sub-trajectory  $T'$  of a (raw) trajectory  $T$  valid in the interval  $[t_i, t_j]$ ,  $t_0 \leq t_i < t_j \leq t_{|T|-1}$ , is defined as the portion of  $T$  between timestamps  $t_i$  and  $t_j$ . Having the above definitions in hand, we define their semantic-aware variants.

*Definition 1 (Enriched Point):* An enriched point  $r_i$  corresponds to a (raw) pair  $(p_i, t_i)$ , and is defined as a triple  $\langle p_i, t_i, v_i \rangle$ , where  $v_i$  is a multi-dimensional vector consisting of categorical and/or numerical variables that annotate the raw point with associated context data.

Examples of  $v_i$  attribute values could be any user-defined tag or annotation valid regarding the specific application (e.g. consider annotations made by an event recognition module that detects the ‘top-of-climb’ or ‘top-of-descent’, etc) or any numerical variable that can be attached to  $p_i$ , such as weather information (e.g. temperature, wind speed, humidity, etc).

*Definition 2 (Semantic Trajectory):* A semantically enriched trajectory  $R$  corresponds to a (raw) trajectory  $T$  of a moving object, which is defined as the sequence of the enriched points of  $T$ .

Given the previous definitions and assuming a historic semantic trajectory database (STD), we define the FSTP problem as follows.

**Problem 1 (Future Semantic Trajectory Prediction - FSTP):** Given (i) a STD consisting of semantic trajectories  $R$ , (ii) a distance function  $dist()$  that quantifies the dissimilarity between two semantic trajectories, (iii) a *–incomplete–* semantic trajectory  $\langle (p_0, t_0, v_0), (p_1, t_1, v_1), \dots, (p_{i-1}, t_{i-1}, v_{i-1}) \rangle$  of a moving object  $o$ , recorded at past  $i$  time instances, (iv) a (enriched) flight plan  $F$ , and (v) a target region  $G$ , predict the semantic trajectory  $R_F: \langle (p^*_i, t_i, v^*_i), (p^*_{i+1}, t_{i+1}, v^*_{i+1}), \dots, (p^*_t, v^*_t) \rangle$ , where  $p^*$  is located in  $G$ , i.e.,  $o$ 's anticipated sequence of enriched points until it reaches  $G$ , where  $R_F \in STD$  and satisfies the following property:

$$R_F = \{R_i | \text{argmin}_i (dist(F, R_i)), \forall i = 1, \dots, N\} \quad (1)$$

TABLE I  
TERMS AND DEFINITIONS

Term	Definition
$(p_i, t_i)$	a 3-D point $p_i = (x_i, y_i, z_i)$ along with its timestamp $t_i$
$T(T)$	a raw trajectory (sub-trajectory, resp.) consisting of a set of pairs $(p_i, t_i)$
$r_i$	an enriched point corresponding to a (raw) timestamped point $(p_i, t_i)$
$R_F$	a semantic trajectory consisting of a set of enriched points $r_i$
STD	a semantic trajectory database consisting of a set of semantic trajectories
$N$	number of available flight plans & actual routes (dataset size)
$FP_i$	flight plan of flight $\{i\}$ , as a sequence of waypoints
$RT_i$	actual route of flight $\{i\}$ , as a sequence of waypoints
$SW_i$	semantics of flight $\{i\}$ , as a sequence of waypoints
$g(\dots)$	semantic-aware similarity function for comparing two enriched trajectories
$d_{n,m}$	semantic-aware distance between trajectories $n$ and $m$ based on $g(\dots)$
$K$	number of trajectory clusters
$C_k$	set of trajectories assigned to cluster $k$
$R_k$	medoid, i.e., representative “mean” trajectory for cluster $k$
$L_k$	length of trajectories in cluster $k$ , i.e., number of waypoints in $R_k$
$\theta^k$	HMM for cluster $k$ designed based on $R_k$
$S_{kt}$	state $\{t\}$ of $\theta^k$ , i.e., for waypoint $\{t\}$ of $R_k$
$E_{kt}$	emission $\{t\}$ of $\theta^k$ , i.e., for waypoint $\{t\}$ of $R_k$
$O_k$	maximum-likelihood flight plan associated with medoid $R_k$
$\mu_{kt}$	mean values of the per-cluster, per-waypoint FP/RT deviations
$\sigma_{kt}$	standard deviations of the per-cluster, per-waypoint FP/RT deviations
$err_{kt}$	half-width of confidence interval for $\mu_{kt}$
$q_X$	query flight plan, i.e., input for the query stage
$H_b$	maximum-likelihood actual route for $q_X$ based on $\theta^b$ and $R_{kb}$ where $b$ is the best-matched cluster ( $X \leftarrow b$ ).

Note that a flight plan announced by an airline is a low-resolution raw trajectory that consists of the waypoints and times that the aircraft is constrained to pass through. Thus, in order for a semantic trajectory  $R$  to be comparable with a flight plan  $F$ , the latter should be enriched / annotated, so as to encapsulate the same information as the semantic trajectories (e.g. weather information should be attached to the corresponding waypoints). The annotation of the flight plan is performed with exactly the same method as the one followed in order for raw aircraft trajectories to be enriched.

In the following and in order to simplify the presentation, we assume that  $i = 1$ , in other words we only have knowledge of the starting position of the aircraft. Thus, our problem is informally described as the task to *predict the semantic trajectory of a flight between the departure and the destination airport, given a corresponding historic database of semantic trajectories and a flight plan*. Nevertheless, our approach is applicable to  $i > 1$  as well.

#### IV. THE PROPOSED FSTP FRAMEWORK

Our approach for addressing the FSTP problem is as follows (also schematically illustrated in Fig. 1): we first cluster the historic STD by using  $dist()$  and for each cluster we build a HMM; then, given a flight plan, we probabilistically recognize the cluster (through the respective HMM) that  $F$  matches best. The 2<sup>nd</sup> step is a classification problem, where the set of classes corresponds to the set of cluster identifiers  $C: \langle C_1, C_2, \dots, C_K \rangle$ . More specifically, given an unclassified flight plan  $F$  and the set  $\theta^k, 1 \leq k \leq K$ , of the HMMs representing each cluster (i.e., class), we search for the model that maximizes the likelihood of having generated  $F$ :

$$C_F = \{C_k | \text{argmax}_k [P(F | \theta^k)]\} \quad (2)$$

Interestingly, this problem can be straightforwardly solved by applying the Forward algorithm [8].

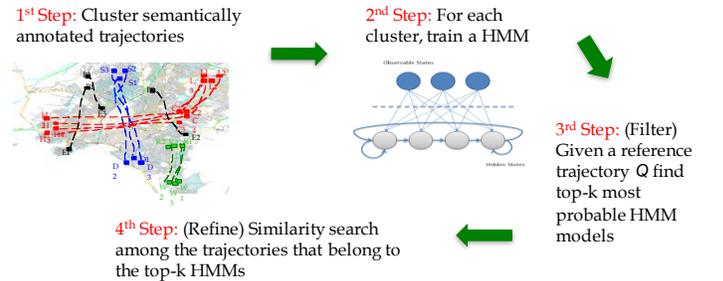


Fig. 1 The Future Semantic Trajectory Prediction (FSTP) framework

At the 3<sup>rd</sup> step, we focus on the cluster that is the most probable to find the solution and retrieve the top- $k$  most similar semantic trajectories by applying  $dist()$  in order to reach the final answer. Obviously, this approach is an approximate one (i.e. we cannot guarantee that the most similar trajectory belongs to the top- $k$  clusters), but our claim is that the way we cluster the STD and model each cluster with a HMM results in a very accurate solution. Of course, this approach provides a significant speedup w.r.t. exhaustively searching the entire STD.

In the following subsections, we provide details of each step of our methodology.

### A. Clustering Semantic Trajectories

Clustering semantic trajectories implies the partitioning of a STD into clusters (groups), so that each cluster contains similar semantic trajectories according to a specific similarity measure. Two semantic trajectories of aircrafts may be considered similar in many ways; they may have common departure and/or destination airports; they may fully or partly be close to each other throughout the flight, they may be fully or partly synchronous, or they may be disjoint in time but with similar behavior (e.g. same control operations as these are represented by their aircraft intent, etc.). For our task, we adopt the SemT-OPTICS approach proposed in [12]. The SemT-OPTICS algorithm is driven by the popular OPTICS clustering method [13]; it is tuned by the same parameters, namely *minPts* describing the number of elements required to form a cluster and *eps* describing the maximum distance (radius) to consider for a sufficiently dense cluster, and its outcome is also a reachability plot, upon which we automatically extract clusters and outliers using the  $\xi$ -clustering method, originally proposed in [13].

For our purposes, the dissimilarity between two enriched points is decomposed by two parts, one regarding their spatio-temporal dissimilarity and another regarding their dissimilarity on the semantic components. In particular, we adopt an appropriate modification of the function proposed in [12], which in its turn is a variant of Edit distance with Real Penalty (ERP) [14]:

**Definition 3 (distance between enriched points,  $D_r$ ):** Given two enriched points  $r_i$  and  $r_j$ , their distance  $D_r(r_i, r_j)$  is defined as:

$$D_r(r_i, r_j) = \lambda \cdot \text{dist}_e(r_i, r_j) + (1 - \lambda) \cdot \text{dist}_v(r_i, r_j) \quad (2)$$

$$\text{dist}_e(r_i, r_j) = \frac{\sqrt{w_1 \cdot \{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2\} + \frac{w_2}{w_1} \cdot (t_i - t_j)^2}}{\text{maxEuclideanDistance(STD)}} \quad (3)$$

$$\text{dist}_v(r_i, r_j) = 1 - \frac{v_i \cdot v_j}{\|v_i\|^2 + \|v_j\|^2 - v_i \cdot v_j} \quad (4)$$

where  $\text{dist}_e$  is the Euclidean distance in the 4-D space ( $x, y, z, t$ ), user-defined weights  $w_1$  and  $w_2 = 1 - w_1$  weight and bring into similar scale the spatial vs. the temporal dimension,  $\text{maxEuclideanDistance(STD)}$  is the coverage of the database in the 4-D space acting as a normalization factor in  $[0..1]$ ,  $\text{dist}_v$  is the Jaccard distance of the semantic components, and  $\lambda \in [0..1]$  is a user-defined parameter that tunes the relative importance between the two components.

In particular for the semantic distance, the part of the vector  $v_i$  consisting of the numerical variables is normalized to exclude scaling effects, whereas each categorical variable (described by a set of keywords) is transformed to vector of weights, the dimensionality of which corresponds to the vocabulary of all distinct keywords in the database. Thus, each keyword corresponds to a weight that is calculated by TF-IDF [15]. Overall,  $v_i$  is the concatenation of the two

(numerical and categorical) vectors. As the Jaccard distance maps the semantic similarity to the range  $[0..1]$ , it follows that  $D_r(r_i, r_j)$  always results into  $[0..1]$ .

Having defined distance  $D_r$  between two enriched points, distance  $D_R$  between two semantic trajectories is defined as follows:

**Definition 4 (distance between semantic trajectories,  $D_R$ ):** Given two semantic trajectories  $R_i$  and  $R_j$  of arbitrary length (i.e., arbitrary number of enriched points), their distance  $D_R(R_i, R_j)$  is defined as:

$$D_R(R_i, R_j) = \min \left\{ \begin{array}{l} D_R(T(R_i), T(R_j)) + D_r(r_{i,1}, r_{j,1}), \\ D_R(T(R_i), T(R_j)) + D_r(r_{i,1}, \text{gap}), \\ D_R(T(R_i), T(R_j)) + D_r(\text{gap}, r_{j,1}) \end{array} \right\} \quad (5)$$

where  $T(R_i)$  denotes the tail of  $R_i$ , namely the enriched points of  $R_i$  after removing the first enriched point of the  $i$ -th semantic trajectory ( $r_i, 1$ ), and  $\text{gap}$  is a virtual enriched point whose spatio-temporal value is the origin of the 4-D space of the entire dataset, while its semantic component corresponds to the zero vector.

The value of the gap element is given in a way similar with [14], where it is determined as the first value of the time scale for the time series (i.e., typically  $\text{gap} = 0$ ). Following a similar approach as in [12], it is trivial to prove that  $D_R$  is a metric.

Given the distance  $D_R$  and a corresponding clustering result  $C$  consisting of  $K$  clusters (noise could also be considered as a separate cluster), we define the average distance  $\overline{D_{R_l}^{C_k}}$  of a member  $R_l$  of a cluster  $C_k | k = [1, \dots, n]$  from all other  $m - 1$  members of the same cluster ( $|C_k| = m$ ) as:

$$\overline{D_{R_l}^{C_k}} = \frac{1}{m - 1} \sum_{i \in [m], i \neq l} D_R(R_l, R_i) \quad (6)$$

The member  $R_l$  which has the minimum average distance in cluster  $C_k$  is considered the medoid  $R_\mu^{C_k}$  of cluster  $C_k$ . Formally:

$$R_k = R_\mu^{C_k} = \arg \min_l \left( \overline{D_{R_l}^{C_k}} \right), l \in [m] \quad (7)$$

Thus, each clustering of semantic trajectories can be represented by the corresponding cluster medoids.

### B. Model-Based trajectory prediction via HMM

As already discussed earlier, HMM formulates the evolution of a system by a set of states and transitions between them, each one accompanied by a probability that is typically extracted by analyzing historic data. In the context of trajectory prediction, the flight route and all the associated information (weather, semantic data, etc.), are encoded into discrete values that constitute the HMM states. Then, the trajectory itself is treated as an evolution of transitions between these states, using the raw trajectory data of a large set of flights for training, plus spatio-temporal constraints (locality) to reduce the dimensionality of the problem. This approach is already being tested for trajectory prediction from

raw data and some very recent case studies show that its results on real data are very promising (e.g. see [9] [11]).

Mathematically speaking, a HMM is defined as a doubly stochastic process. It represents the joint probability of an unobservable (so is called *hidden*) finite state sequence  $H = \{S_t : t = 1, \dots, T\}$ , where  $S_t \in S = \{1, \dots, M\}$ , and an observable emissions sequence  $O = \{X_t : t = 1, \dots, T\}$ , where  $X_t = x \in \mathbf{R}$  is associated to the corresponding  $S_t$ . Formally, a HMM is defined by  $M$  distinct states,  $L$  distinct values that can be observed at each state transition, the  $M \times M$  matrix  $A = \{a_{ij}\}$  of transition probabilities, where  $a_{ij} = P(S_t = j / S_{t-1} = i)$  with  $1 \leq i, j \leq M$ , and  $S_t$  being the state at time  $t$  (stationary, first-order Markov chain), the  $M \times L$  matrix  $B = \{b_i(x)\}$  of the probabilities of emissions  $x$  at state  $i$ , where  $b_i(x) = P[X_t = x / S_t = i]$ , with  $X_t$  being the observed value produced by the HMM at time  $t$ , and the set  $\Pi = \{\pi_i\}$  of prior state probabilities, where  $\pi_i = P(S_1 = i)$ ,  $i = 1, \dots, M$ . Therefore, a HMM  $\theta$  is specified as a triple  $\theta = (A, B, \Pi)$ . According to this formulation, a HMM is the functional form:

$$P(S, X) = P(S_1) \prod_{t=2}^T P(S_t / S_{t-1}) \prod_{t=1}^T P(X_t / S_t) \quad (8)$$

As described earlier, the FSTP task involves the modeling of the complete trajectory and not just the forecasting of  $t \leq T$  future locations for a trajectory of length  $T$ . Hence, the entire (semantic) trajectory has to be transformed into a discrete-space domain, in order to be modelled into states and emissions sets. Moreover, if the trajectories are organized into different groups, e.g. by clustering, a separate HMM is associated with each one of these groups.

In order to apply a HMM-based modeling to already clustered semantic trajectories, the following steps are performed:

1. For each semantic trajectory cluster  $C_k$ ,  $k = \{1, \dots, K\}$ , a HMM  $\theta^k$  is trained, i.e., the model parameters  $(A, B, \pi)$  are estimated for maximum likelihood of the training set observation sequences for the corresponding cluster  $C_k$ . This is the HMM training phase and it is realized by applying the Expectation-Maximization (EM) algorithm.
2. For a new flight plan  $Q = FP$  that is used as a query, i.e., a prediction of its complete semantic path in the FSTP sense, the model likelihood of  $Q$  is computed for all  $K$  possible HMM models,  $P(Q|\theta^k)$ ,  $1 \leq k \leq K$ . Then  $Q$  is assigned to the cluster with the highest HMM likelihood according to Eq.(8). This is the HMM evaluation stage.

The *Expectation-Maximization (EM)* algorithm [16] used to find the local maximum of the HMM likelihood of the state-transition sequence, consists of two iterative steps: (a) the initialization or *Expectation* step (*E-step*) and the refinement or *Maximization* step (*M-step*). Following the procedure described in the previous section, the initial estimates for the model parameters  $\{A, B, \Pi\}$  are used to specify a HMM, which will be the starting point for the *E-step*. In the *M-step* the model parameters  $\{A, B, \Pi\}$  are adjusted in order to maximize the models' likelihood  $P(Q|\theta)$ . This process is repeated until a desired convergence limit is satisfied, i.e., the likelihood of the HMM cannot be further improved.

The *Expectation step (E-step)* involves estimating two terms. The first is the probability of being in state  $i$  at time  $t$  given the observed emissions sequence:

$$\gamma_t(i) = P(S_t = i / X_t = x; \theta) \quad (9)$$

The second is the probability that the process rests at state  $i$  at time  $t$  and moves to state  $j$  at  $t+1$  given the observed emissions sequence is:

$$\xi_t(i, j) = P(S_t = i, S_{t+1} = j / X_t = x; \theta) \quad (10)$$

These terms can be calculated via a dynamic programming method known as the *forward-backward algorithm*, which has a complexity of  $O(J^2T)$ .

In the *Maximization step (M-step)*, based on Eq.(9) and Eq.(10), the initial transition probabilities are estimated as:

$$\hat{\pi}'_i = \gamma_0(i) \wedge \hat{p}'_{ij} = \frac{\sum_{t=1}^T \xi_t(i, j)}{\sum_{t=1}^T \sum_{i \neq j} \xi_t(i, j)} \quad (11)$$

In the common case where the observations  $X_t = x$  are normally distributed given  $S_t = i$ , i.e.,  $P(X_t / S_t = i) \approx N(\mu_i, \sigma_i^2)$ , the parameters  $\mu_i$  and  $\sigma_i^2$  are estimated as:

$$\hat{\mu}'_i \cong \frac{\sum_{t=1}^T \gamma_t(i) X_t}{\sum_{t=1}^T \gamma_t(i)} \wedge \hat{\sigma}'_i \cong \frac{\sum_{t=1}^T \gamma_t(i) (X_t - \hat{\mu}'_i)^2}{\sum_{t=1}^T \gamma_t(i)} \quad (12)$$

### C. HMM design and parameters

In order to determine the state transition and the emissions probability matrices of the HMM, the set  $M$  of its hidden states and the associations between them have to be determined. If the semantic trajectories of  $N$  aircrafts belong to a certain cluster  $C_k$ , then the set of  $M$  hidden states can be defined in various ways. In the following, we present three of them:

**Exhaustive mapping:** If the complexity of the problem is low, i.e., the size of all  $C_k$  is small and/or the dimensionality is low, then the HMM states can be defined as the *union of the (distinct) enriched points* from all the trajectories within each group.

**Maximum-length representative:** In this case, the representative semantic trajectory denoted as  $R'_{C_k}$  for group  $C_k$  is defined by selecting the one that contains the *maximum* number of enriched points within this group.

**Cluster medoid:** Instead of selecting the member of  $C_k$  with the maximum length, the group itself is treated as a cluster and its *medoid* is the one that is used as the representative. Then, each of its semantic points is translated to a separate state, similarly to the maximum-length case. However, the medoid is statistically a much better representative than the maximum-length one, since it is created in terms of density-based optimization and, thus, exhibits minimum deviation from all the trajectories it represents. This approach has the additional advantage that the clusters  $C_k$  can also be discovered via a typical clustering process, employing a proper semantic-aware similarity metric that separates the original set of trajectories into compact groups, essentially making the whole process of designing the HMM unsupervised. Apparently, the medoid-based state

definition is the method adopted in this paper for the HMM design.

Concerning the observation space  $B$  of emissions for each hidden state of the model, this is much more problem-specific. Typically, the emissions are associated with some property or output from the system that is modelled by the HMM, in the sense that the system shifts between states internally and the emissions are the corresponding observations produced with every such transition. In other words, an emission is associated with some observable result from the state transitions, since the states themselves are not observed in a HMM. It is common to assume that the HMM emissions follow a Gaussian distribution in each state, if the number of observations allow such a statistical approximation (typically more than 30 unbiased samples).

The design of the state transitions space  $A$  and observation space  $B$  directly affects the way the corresponding matrices are created. At each time step  $t$ , an observation is emitted and the process can either stay at the same state or shift to another, updating the two matrices accordingly. This process is followed for every point in the trajectory and for every trajectory in group  $C_k$  that is modelled by the  $\theta^k$  HMM, essentially producing the corresponding cumulative probability density matrices  $\hat{A}$  and  $\hat{B}$  (estimated).

In most cases, the system is assumed to be stationary and, thus, the resulting HMM static. This means that all the probabilities in Eq.(8) through Eq.(11) become time-independent and can be approximated by the corresponding frequency distributions, provided that both the sizes of each trajectory group  $C_k$ , as well as the length of each trajectory itself, are large enough to produce statistically significant estimates. In practice, this means that each cluster should contain at least 30 or more members (trajectories).

For the final transition matrix  $A$ , the transition probabilities  $a_{ij}$  are computed as:  $a_{ij} = \frac{\hat{a}_{ij}}{\sum \hat{a}_{i*}}$ , where the sum  $\sum \hat{a}_{i*}$  is taken over all the next-states  $j \in S$ . As a result, the probability of staying at the same state is  $a_{ii}$  and for the last state in a transition sequence it is 1. Similarly for the final emissions matrix  $B$ , the emission probabilities  $b_i(x)$  are computed by dividing the number of observed emission  $x$  when arriving at state  $i$  by the total number of observed emissions for this state. If there are only  $l \in E = \{1, \dots, L = |X_t|\}$  distinct emission values in the observation set, then  $b_i(x) = b_{il} = \frac{\hat{b}_{ij}}{\sum \hat{b}_{i*}}$ , where the sum  $\sum \hat{b}_{i*}$  is taken over all the emissions  $l \in E$ . As described above, the corresponding probability density function can be approximated in continuous form by assuming a Gaussian as in Eq.(12), in which case  $b_i(x)$  is defined by the closed-form analytical formula for normal distribution.

Finally, for the initial state probabilities  $\pi = \{\pi_i\}, 1 \leq i \leq M$ , matrix  $\hat{A}$  is used again as for  $A$ , in order to produce relative frequency estimations. In other words:  $\pi_i = \frac{\sum \hat{a}_{i*}}{\sum \sum \hat{a}_{**}}$ , where the sum  $\sum \hat{a}_{i*}$  is taken over all the next-states  $j \in S$  and the sum  $\sum \sum \hat{a}_{**}$  is taken over the entire matrix  $\hat{A}$ .

Using these estimations for matrices  $A$  and  $B$ , as well as the initial state probabilities  $\pi$ , the HMM  $\theta = (A, B, \Pi)$  is fully

defined. Hence, for  $K$  clusters  $C_k$  of trajectories, each of the  $\theta^k$  HMM is defined similarly.

#### D. Medoid-based HMM: The hybrid approach

In the previous sections, HMM was described as the base model for the FSTP task. In this work, a novel HMM approach is proposed for addressing two problems, i.e., the inherent complexity issues of the HMM in general and the physical meaning of the model with regard to the FSTP task. The first part is addressed by employing a combination of flight plans and matching route points from the actual trajectory, instead of using the full-resolution trajectory itself. The second part is addressed by associating the emissions to a transformation of the combined flight plans and matching route points, in a way that actually correspond to the observed “output” of the model.

(a) *Waypoints as HMM states*: The general idea of this novel approach is described earlier (see: Introduction). A pair of departure and destination airports is selected for trajectory modeling, using a large set of flights between them (historic data). Flight plans (**FP**) and matching actual route points (**RT**), together with additional semantic information (**SW**) including weather parameters, aircraft type, wake category, weekday of flight, etc, are used as the training dataset. Each flight plan  $FP_i$  contains a limited number of reference waypoints, typically 40-50 times less than the full-resolution trajectory track (e.g. secondary radar). For each waypoint  $f_{ij}$  of a flight plan  $FP_i$ , a matching route point  $r_{ij}$  is the closest one from the full-resolution trajectory. Hence, there is a 1-on-1 association between the waypoints  $f_{ij}$  of a flight plan and its actual route points  $r_{ij}$ , producing a sequence of pairs  $\{f_{ij}, r_{ij}\}$  of length  $L = |FP_i| = |RT_i|$  for each flight in the dataset. Additionally, each of these flight plan or actual route points is “enriched” by a semantic vector  $s_{ij}$  that includes every item of information that is available for weather, aircraft, etc. Some of the properties such as the aircraft type and wake category do not change, but the exact values of (local) weather parameters may differ between the waypoints. Moreover, if the route points  $r_{ij}$  deviate significantly from the flight plan waypoints  $f_{ij}$  they refer to, the values of weather parameters may differ between these too. In practice, this means that each of the  $f_{ij}$  and  $r_{ij}$  is accompanied with its own semantic vector  $s_{ij}$  as per-point enrichment variables, namely  $sf_{ij}$  and  $sr_{ij}$ . Using this formulation, each flight in the training dataset corresponds to a triplet of  $\{FP_i, RT_i, SW_i\}$ , which becomes a series of  $\{f_{ij}, r_{ij}, \{sf_{ij}, sr_{ij}\}\}$  of length  $L$ .

The clustering stage of this approach essentially employs the semantic-aware similarity function we introduced earlier for comparing and grouping enriched routes together, i.e., against the  $\{RT_i, SW_i\}$  part. It should be noted that  $FP_i$  contains the waypoints which constitute a priori constraints of each flight and  $RT_i$  contains the actual route points that are the closest realizations of these constraints. In other words,  $\{RT_i, SW_i\}$  embodies information associated to each actual flight and at the same time *posterior* knowledge related to it. The spatio-temporal proximity between different trajectories is not only included but augmented with semantics during the

clustering stage, which produces a medoid as a semantic-aware representative for each cluster. This procedure essentially renders unnecessary any further discretization step for the design of the HMM. In practice, using the  $\{RT_i, SW_i\}$  in clustering and the resulting medoids as the baseline for producing corresponding HMMs is analogous to employing a discretization process with a *non-uniform grid*: each state is now associated to an enriched route point  $\{r_{ij}, sr_{ij}\}$  as a graph node and each state transition as a directed edge in this graph. It should be noted that the state-of-the-art approach used e.g. in [9] employs a uniform spatial grid, each point of the full-resolution trajectory is aligned to a grid cell and the HMM states are essentially the non-empty cells produced after processing the entire training dataset. Together with weather parameter discretization, this “blind” HMM approach normally produces an excessive number of states that correspond to a very large but sparse state transition matrix. In contrast, the novel approach proposed in this work produces multiple HMMs (one for each cluster) that exhibit very limited probabilistic path variability and very compact in size, since there are only  $L$  states and almost<sup>1</sup> sequential transitions between them.

(b) *FP/RT deviations as HMM emissions*: The physical meaning of the HMM emissions is the observed output of the system. Given the availability of both the flight plans  $FP_i$  and the corresponding actual routes  $RT_i$ , the proposed approach associates the emissions with the *deviation* between these two, since this is the actual output observed from the evolution of a flight. Specifically, the medoid of each cluster, i.e., the representative actual route, is used to specify the sequence of states for the corresponding HMM, as described above. Then, each individual flight plan  $FP_i$  is compared to the medoid of its assigned cluster to calculate the deviation for each of its  $L$  reference points, which become the emissions sequence. This deviation can be spatiotemporal-only or fully semantic-aware as in the similarity function used during the clustering stage. In this work, the first choice was employed for the emissions, i.e., the deviations include only the 3-D differences (signed) between the reference points, as this is the standard requirement for assessing the prediction accuracy of FSTP algorithms in most cases. For per-dimension deviations (lat/lon/alt), simple subtraction is used and the (signed) result is subsequently converted to meters. For 3-D deviation calculations, the Haversine method [17] is employed to calculate the spherical distance, enhanced with trapezoid approximation for surface-to-altitude distances, producing a full 3-D calculation for an arbitrary pair of points in the Earth’s atmosphere. For instance, for the area of the utilized training dataset, i.e., flights between Barcelona and Madrid in Spain<sup>2</sup>, the true geodesic resolution is 111.133 km/deg Lat (mean) and 83.921 km/deg Lon.

Since multiple flights are included in each cluster, the values of these emissions produce an empirical probability distribution function per state (i.e., reference waypoint), which under moderate statistical assumptions can be approximated by a parametric Gaussian. In other words, the final HMM for each cluster contains exactly  $L$  states<sup>3</sup> with sequential transition between them and a continuous-valued Gaussian distribution of emissions that correspond to individual point-wise deviations between flight plans and the cluster’s medoid. This is exactly why this “hybrid” approach that employs clustering before the HMM design produces a very efficient and truly scalable solution for the FSTP task.

In summary, the outline of the **training** phase of the proposed FSTP framework is described in Algorithm I.

---

**Algorithm I: Training of the Hybrid Clustering/HMM method**

---

**INPUT:**  $\mathbf{FP}=\{FP_i\}$ ,  $\mathbf{RT}=\{RT_i\}$ ,  $\mathbf{SW}=\{SW_i\}$ ,  $i=1,\dots,N$ , a set of flight plans, actual matching routes and associated semantics.

---

1. Let:  $z_{ij}=\{f_{ij}, r_{ij}, \{sf_{ij}, sr_{ij}\}\} \in \mathbf{Z}$  the corresponding data series of length  $L_z=|FP_i|=|RT_i|=|SW_i|$ , where  $f_{ij} \in FP_i$ ,  $r_{ij} \in RT_i$  and  $\{sf_{ij}, sr_{ij}\} \in SW_i$ .
2. Let:  $v_{ij}=\{f_{ij}, sf_{ij}\} \in \mathbf{Z}_f$  and  $u_{ij}=\{r_{ij}, sr_{ij}\} \in \mathbf{Z}_r$  sets of enriched flight plans and enriched actual routes, where  $f_{ij} \in FP_i$ ,  $r_{ij} \in RT_i$  and  $\{sf_{ij}, sr_{ij}\} \in SW_i$ .
3. [CLUSTERING] Let:  $D_r(r_i, r_j)$  the semantic-aware similarity function for comparing two enriched trajectories in  $\mathbf{Z}_f$  or  $\mathbf{Z}_r$ .
4. Cluster  $\mathbf{Z}_r$  into  $K$  distinct groups using  $D_r(r_i, r_j)$ . For each cluster  $C_k$ , produce the best-match (maximum similarity) representative or *medoid*  $R_k$ . Note:  $K$  is computed automatically by the clustering algorithm.
5. [HMM SETUP] For each cluster  $C_k$  with medoid  $R_k$ , design a HMM  $\theta^k$  with exactly  $L_k=|R_k|$  states and sequential transitions between them, i.e.:  $\{S_{k1} \rightarrow S_{k2} \rightarrow \dots \rightarrow S_{kL}\}$ .
6. Using  $D_r(r_i, r_j)$ , calculate the  $d_{n,k}=D_r(v_{n,*}, R_k)$  semantic-aware similarity between every enriched flight plan  $v_{n,*}$  ( $*=j \in \{1,\dots,L_k\}$ ) and the medoid  $R_k$ . In this study, FSTP accuracy is focused only on the 3-D spatial error (Lat/Lon/Alt).
7. For each state  $S_{kj}$  the corresponding emission  $E_{kj}$  is modelled by a Gaussian distribution of the  $d_{n,k}$  values, i.e., mean  $\mu_{kj}$  and standard deviation  $\sigma_{kj}$  of  $d_{n,k}$ , where:  $n \in C_k$ , i.e.:  $E_{kj} \sim N(d; \mu_{kj}, \sigma_{kj})$ .

---

**OUTPUT:**  $K$  clusters  $C_k$  with medoids  $R_k$ ,  $K$  HMMs  $\theta^k$  with states  $S_{kj}$  defined by the waypoints of  $R_k$  and emissions  $E_{kj}$  modelled by corresponding Gaussian distributions of means  $\mu_{kj}$  and standard deviation  $\sigma_{kj}$ .

---

After the last step in the training pipeline, each flight subset in cluster  $C_k$  is fully described by a representative semantic-aware medoid  $R_k$  and the corresponding Gaussian approximation for the emissions  $O_k$ :

---

<sup>1</sup> Employing non-sequential HMM state transitions in this approach requires the exploitation of multiple top- $l$  medoids or combinations of medoid segments per cluster. In this approach, one medoid per cluster is used, i.e.,  $l=1$ .

<sup>2</sup> LEBL-LEMD: lat = [40...43]°, lon = [-3...+3]°, alt = [0...40,000]ft.

---

<sup>3</sup> When a cluster contains flights with different number of reference points, the medoid length is used as the basis for the entire cluster, with flight expansions or cropping when needed. However, for statistical consistency, all errors and confidence intervals presented later on refer to the minimum-common-length  $L_k$ , i.e., all calculations are made using the entire cluster in all cases.

$$i \in C_k: \mu_{k^*} = E[(f_{ij} - r_{kj})], \sigma_{k^*} = \text{Var}[(f_{ij} - r_{kj})]^{1/2} \quad (13)$$

$$\Rightarrow (f_{ij} - r_{kj}) \approx E_{kj} \sim \mathcal{N}(d, \mu_{kj}, \sigma_{kj})$$

Then, the state transition within the  $R_k$  of each cluster essentially defines the maximum-likelihood emissions sequence as:  $O_k = \{\mu_{k1} \rightarrow \mu_{k2} \rightarrow \dots \rightarrow \mu_{kL}\}$ , i.e., through the means of each Gaussian pdf of the per-waypoint deviations. Hence, for each  $R_k$  a corresponding representative flight plan can be produced in the maximum-likelihood sense as:

$$O_k - R_k \approx E_k \Rightarrow O_k = R_k + E_k \approx R_k + \mu_k \pm \text{err}_k \quad (14)$$

where  $\text{err}_k^*$  is estimated as the half-width of the confidence interval for the mean  $\mu_k$ , i.e.:  $\text{err}_k^* = \frac{t_\alpha \cdot s_k}{\sqrt{n_k}}$ , where,  $n_k$  is the size of cluster  $k$ ,  $s_k$  is the sample standard deviation of the  $f_{ij} - r_{kj}$  differences and  $t_\alpha$  is the corresponding one-tailed t-Student value at significance level  $\alpha$ , all calculated separately for each reference waypoint ( $* = j \in \{1, \dots, L_k\}$ ).

Next, the exact route of the predicted trajectory, based on a query flight plan as input, is based on the maximum-likelihood estimation of emissions from the best-matching cluster. In summary, the outline of the **query** (or testing) phase of the proposed FSTP framework is described in Algorithm II.

---

**Algorithm II:** Testing of the Hybrid Clustering/HMM method

---

**INPUT:** Let:  $q_X = \{qf_{Xj}, qsf_{Xj}\} \in \mathbf{Z}_f$  a new enriched flight plan of  $L_z$  reference waypoints of a future flight, for which the actual route is to be predicted.

---

1. Let:  $d_{n,m} = g(u_n, u_m)$  a semantic-aware similarity function for comparing two enriched trajectories in  $\mathbf{Z}_f$ .
  2. Using  $g(\dots)$ , calculate the  $d_{X,k} = g(q_X, O_k)$  similarity between  $q_X$  and the  $K$  representative flight plans, i.e.:  $O_k = \{\mu_{k1} \rightarrow \mu_{k2} \rightarrow \dots \rightarrow \mu_{kL}\}$ .
  3. Let:  $b = \text{argmin}\{d_{X,k}\}$ ,  $k=1, \dots, K$ , i.e., the id of the cluster  $C_b$  for which the corresponding HMM  $\theta_b$  produces the best-matching  $O_b$  with respect to  $q_X$ .
- 

**OUTPUT:**  $q_X$  is assigned to cluster  $b$  and  $H_b$  is the maximum-likelihood estimation for its (future) actual route as:  $H_b = q_X - E_b \approx q_X - \mu_b \pm \text{err}_b$

---

## V. EXPERIMENTAL STUDY

### A. Dataset and experimental setup

The experimental setup for validating the proposed Hybrid Clustering/HMM approach is based on a selected set of flights between Madrid and Barcelona. More specifically, the flight plans (the latest submitted before departure), the radar tracks, weather data (actual) and additional aircraft properties are included in the enriched “linked” FP/RT flights dataset from April 2016. The specific pair of airports was selected as the one with the heaviest traffic on a monthly basis compared to any other airport pair in Spain. It involves different flight plans (reference waypoints) and multiple takeoff / landing approaches. In operational mode, each direction and each pair of airports will be associated with a separate Clustering/HMM

model, in order to capture the fine details and the specific statistics of each case.

Table II summarizes the dataset used in the experimental study. Fig. 2 and Fig. 3 illustrate examples of flights with moderate matching and severe between the submitted flight plan and the actual route flown. Fig. 4 presents the (colored) clusters produced for a 4+1 cluster setup, with one of them used to group outliers/noise (7 of 703), for the one-direction subset of flights used from the dataset. Fig. 5 illustrates the corresponding medoid for each cluster.

TABLE II  
SUMMARY OF THE DATASETS USED IN THE EXPERIMENTAL STUDY

Element	Description	Comments
Airport pairs (subsets)	(a) LEBL $\rightarrow$ LEMD: 693 flights (b) LEMD $\rightarrow$ LEBL: 703 flights	
Flight plans (FP)	Latest submitted FP for each flight.	Each FP consists of 11-18 reference waypoints
Actual route (RT)	Reference waypoints from the full radar track route, matched (closest) to the FP.	Waypoint matching was conducted only on the spatio-temporal basis (no semantics).
Weather (SW)	Latest NOAA weather parameters estimated via interpolation upon each reference point.	Wind speed, wind direction, temperature, humidity.
Other semantics (SW)	Additional parameters used in the enrichment process.	Aircraft type, wake (size), weekday.

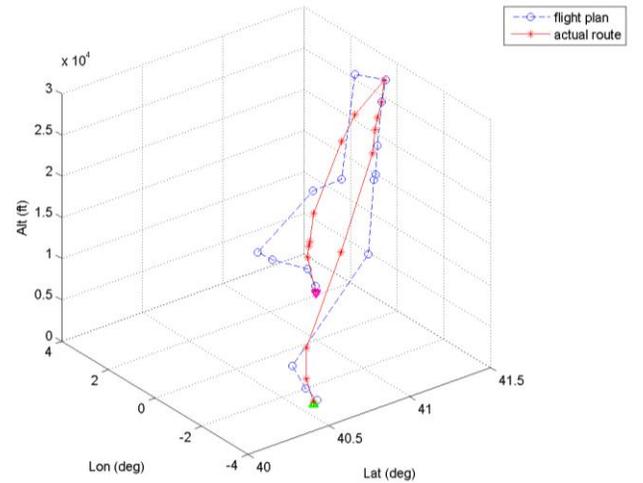


Fig. 2 Example of flight plan vs. actual route from Madrid to Barcelona, with intermediate mismatch between the two trajectories ( $\max\{d_{ij}\} > 20$  km).

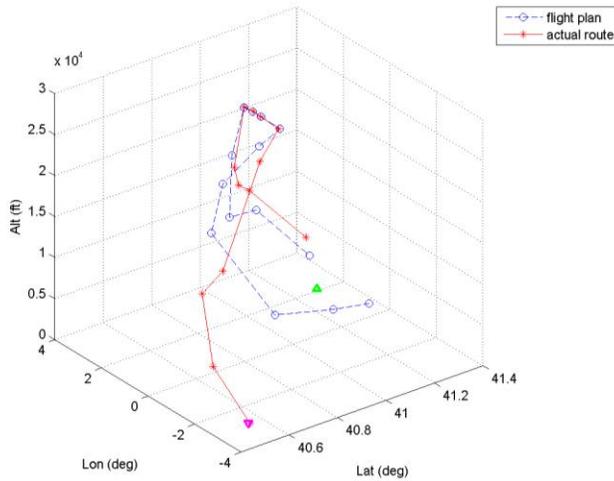


Fig. 3 Example of flight plan vs. actual route from Barcelona to Madrid, with distinct deviation between the two trajectories (different landing approach).

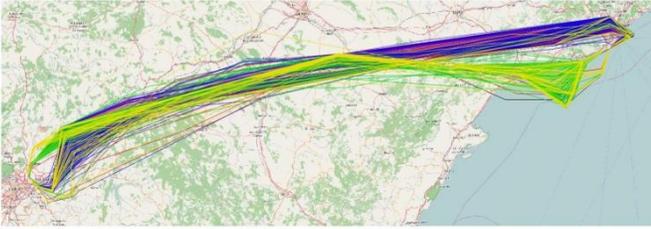


Fig. 4 Example of four main clusters (colored) and one cluster of noise/outliers (black) produced in the clustering phase upon the RT (actual routes) using the EDR semantic-aware similarity metric.

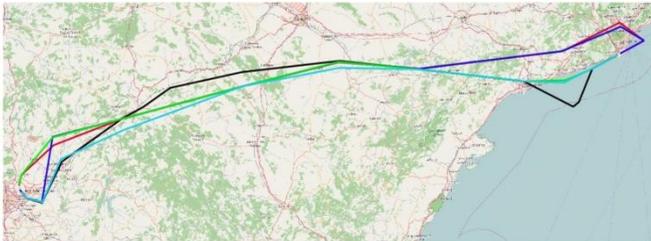


Fig. 5 The medoids of the four main clusters, representing 696 of 703 flights in the enriched FT/RT dataset.

### B. Semantic trajectory clustering phase

As described above, the first stage in the proposed approach is the clustering of the flights using a semantic-aware similarity metric. The parameters of the composite distance metric described in Eq.(2), Eq.(3) and Eq.(4) were established after extensive experimentation and evaluation of the quality (size vs. compactness) of the resulting clusters. More specifically, the spatio-temporal part was preferred over the semantic part ( $\lambda=3/4$ ), in order to produce easily interpretable visual plots for the clusters in 3-D, equally-weighted spatial dimensions ( $w_1=1/3$ ), as well as a requirement for time-invariant trajectory matching ( $w_2=0$ ). Using the flights of each direction separately between Madrid and Barcelona, the enriched  $\{RT_i, SW_i\}$  trajectories were clustered and the corresponding  $R_k$  medoid of each cluster was identified.

### C. HMM modeling phase

According to the description of the proposed approach, the results from the clustering stage are used as input for the next, i.e., the setup and training of the corresponding HMMs. No assumption is made with regard to their statistical distribution, but enforcing a sample size (cluster members) of at least 30 or more essentially enables the estimation of the confidence intervals via proper means, i.e. using sample variations and the t-Student distribution for hypothesis testing. Table III summarizes the results from the statistical significance analysis of the emissions model regarding the four main clusters (7 outliers excluded) of the experimental setup described above. For each cluster, the FP/RT deviations upon every waypoint over its members is used to produce a corresponding (Gaussian) pdf and subsequently the mean, sample stdev and confidence interval of the per-waypoint means are calculated, here for a significance level of  $\alpha=0.1$ . The resulting *half-width confidence interval* or HWCI, which is essentially the radius of the sphere around each reference waypoint, is the aggregated statistic that corresponds to the error term when calculating the maximum-likelihood of an estimated future route, according to Algorithm II (see step 5). It is the average per-waypoint “uncertainty” distance (in meters) through which each  $C_k$  member flight will pass with probability  $1-\alpha$ . TABLE III presents these statistics separately for each dimension (Lat/Lon/Alt) and as 3-D radius, in order to examine the accuracy and error sensitivity of the HMMs.

TABLE III

SUMMARY OF THE EMISSIONS MODEL PER CLUSTER. HWCI = HALF-WIDTH CONFIDENCE INTERVAL FOR PER-WAYPOINT FP/RT DEVIATIONS OVER THE FLIGHTS  $|C_k|$  IN EACH CLUSTER. THE MEANS VALUE, CONFIDENCE INTERVAL AND SAMPLE STDEV REFER TO THE ENTIRE FLIGHT PATH  $L_k$  DEFINED BY  $R_k$ .

$C_k$	$ C_k $	$L_k$	HWCI mean	HWCI mean: conf.int.range	HWCI mean: sample stdev
1	255	13	Lat: 194.5	Lat: 52.3	Lat: 138.9
			Lon: 48.3	Lon: 11.2	Lon: 29.9
			Alt: 29.6	Alt: 7.2	Alt: 19.2
			R = 208.5	R = 50.4	R = 133.9
2	228	14	Lat: 269.5	Lat: 72.0	Lat: 199.6
			Lon: 73.0	Lon: 33.4	Lon: 92.7
			Alt: 32.0	Alt: 6.3	Alt: 17.5
			R = 285.3	R = 77.5	R = 214.7
3	138	15	Lat: 440.1	Lat: 138.1	Lat: 397.8
			Lon: 112.8	Lon: 40.2	Lon: 115.8
			Alt: 48.7	Alt: 9.1	Alt: 26.2
			R = 460.9	R = 142.5	R = 410.4
4	75	11	Lat: 617.6	Lat: 128.1	Lat: 309.6
			Lon: 200.6	Lon: 73.0	Lon: 176.4
			Alt: 102.7	Alt: 16.1	Alt: 38.9
			R = 665.9	R = 141.0	R = 340.8

It should be noted that these estimations may differ significantly between the waypoints, due to the fact that the first and last ones are very “strict” constraints as part of standard takeoff and landing procedures, while intermediate ones can be traversed more “loosely” but shortcuts if necessary, e.g. to save time lost in flight delays. As an example, Fig. 6 illustrates the per-waypoint box plot for

cluster 1 as described above. The height of each bounding box is directly linked to the uncertainty associated with producing the maximum-likelihood FP/RT deviation from the HMM emissions in each waypoint. As expected, most of the waypoints just after takeoff and just before landing have the tightest confidence intervals.

Additionally, Fig. 7 illustrates the per-waypoint RMSE 3-D deviation between flight plans and actual routes for cluster 1, where it is evident that the largest values appear after the takeoff phase (wp.1-2) and just before entering the final approach to landing (wp.12-13). Finally, Fig. 8 illustrates the distributions of the confidence intervals (ranges) of Lat/Lon/Alt and inclusion radius R, providing an overview of the statistical uncertainty per dimension and in 3-D. This is an example of graphical representation of the distributions described in Table III (cluster 1), but here in standard box plot notation, i.e., with median, quartiles and extremes instead of mean and standard deviation.

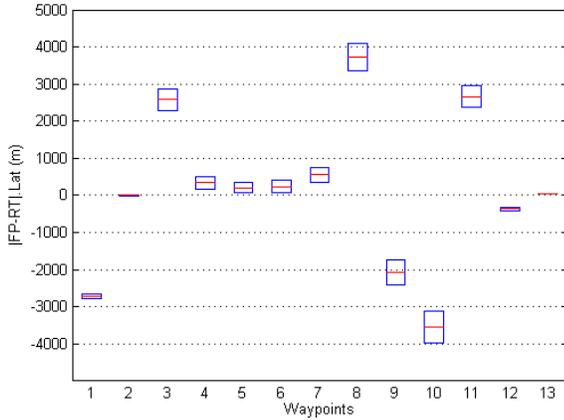


Fig. 6 Mean and confidence interval of the FP/RT Latitude deviations (in meters) within cluster 1 over the minimum common length of flight plans included.

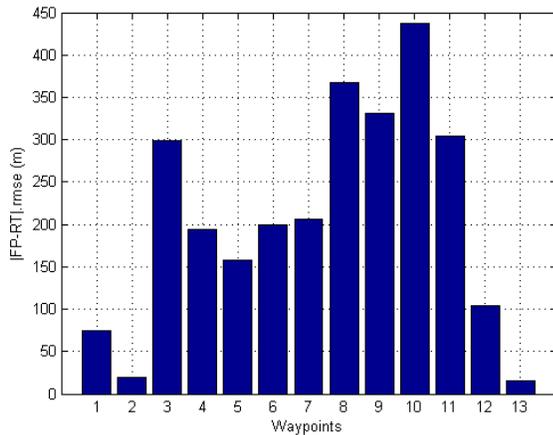


Fig. 7 Mean radius (in meters) of the sphere corresponding to the Lat/Lon/Alt confidence intervals of the FP/RT deviations (in meters) within cluster 1 over the minimum common length of flight plans included.

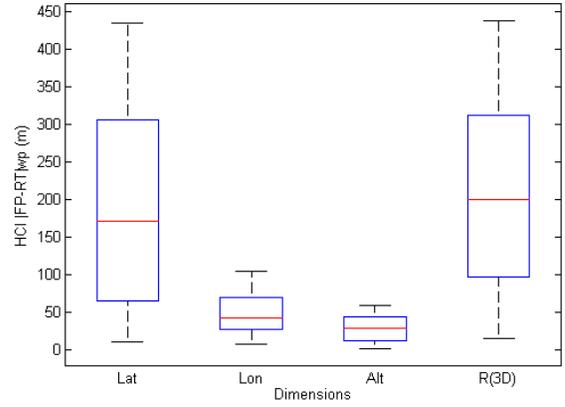


Fig. 8 Distributions of confidence intervals (ranges) of Lat/Lon/Alt and radius of inclusion sphere (in meters) within cluster 1 over the minimum common length of flight plans included.

## VI. DISCUSSION

As described earlier, the proposed method exploits the constraints imposed by the flight plans, i.e., the intended flight path, as well as other “enrichment” parameters such as localized weather and aircraft properties. The first stage incorporates the principle of “divide-and-conquer” via a semantic-aware clustering of actual flight routes, essentially grouping together flights that exhibit similarity not just in their spatio-temporal path but also in their semantic vectors.

As described earlier, each flight path is not modelled by the full-resolution spatio-temporal grid (radar track), but rather by the reference waypoints of the flight plans. This way, the length of the corresponding time series for each flight path is reduced by a factor of 1:50 on average, since in the current datasets described in Table II the full-resolution radar tracks of the flights vary in length between 680-730 points on average, with slightly varying sampling rate of about 5 seconds, while the corresponding FP/RT/weather linked data include 11-18 waypoints with arbitrary time intervals. Additionally, the grouping of flights via clustering in the “enriched” data space reduces the complexity of the corresponding flight paths that will be modelled next by the HMMs, further scaling down the complexity of the training phase. Finally, the modeling of the waypoints as HMM states in this non-uniform/sparse manner, as well as the modeling of deviations between flight plans and actual routes as the per-waypoint emissions, is a natural way to describe the probabilistic model of the FSTP in the context of HMMs.

The results presented in Table III as well as the plots in Fig. 6 through Fig. 8, demonstrate the robustness and the statistical significance of the Hybrid Clustering/HMM proposed here. According to the confidence intervals, flights in cluster 1 can be predicted (see Algorithm II) with accuracy of roughly:  $err_k = 208.5 \pm 50.4/2 = 183...234$  meters upon each waypoint of its submitted flight plan. In contrast, flights in the much smaller cluster 4 can be predicted with accuracy of roughly:  $err_{k^*} = 665.9 \pm 141/2 = 595...736$  meters upon each waypoint of its submitted flight plan. It should be noted that the significance level  $\alpha=0.1$  has some but not very large effect in these confidence intervals in terms of the order of magnitude of this

uncertainty. Since this is a one-tailed t-Student test (upper bound for errors,  $n > 30$ ),  $\alpha = 0.1$  ( $p = 90\%$ ) corresponds to  $t_\alpha = 1.282$ ,  $\alpha = 0.05$  ( $p = 95\%$ ) to  $t_\alpha = 1.645$  and  $\alpha = 0.01$  ( $p = 99\%$ ) to  $t_\alpha = 2.326$ . In other words, even at very high confidence levels ( $p = 99\%$ ) the corresponding interval is at most 81% wider than the values presented in TABLE III, which are already adequately tight. Even for cluster 4 with the smallest size (75), the confidence interval is expected to become (roughly) 1.2 km with probability  $p = 99\%$ , which is actually the worst-case and stricter error bound for this model setup; this is still an order of magnitude more accurate in absolute cross-track error than the current state-of-the-art with “blind” HMM for TP as in [9] [11], i.e., without using the flight plans as constraints.

According to the results in Table III as well as the per-dimension summary plot in Fig. 8, the confidence intervals (ranges) of Lat/Lon/Alt demonstrate significant differences. More specifically, Lat error bounds are consistently much larger than of the other dimensions, 4-3 times wider than Lon and 10 times wider than Alt. The same behavior was confirmed with configurations involving 9-10 clusters instead of four as presented here. This is a very important conclusion from these experiments, related to the fine-tuning of the similarity metric weights and the capability of “shaping” the error margins in a way that prediction uncertainty is equally distributed between the dimensions of the trajectory domain space. In other words, the errors in each dimension can be scaled appropriately so that the flattened ellipsoid that they produce becomes a sphere of radius  $R$ , i.e., approximately the same 3-D error distance in all dimensions.

In this hybrid approach, the FSTP task is addressed from the start. Both the clustering and the HMM stages are inherently designed to be dimensionality-invariant, in the sense that they can be applied in a unified way regardless of the presence or dimensionality of the semantics set **SW** that “enriches” the flight plan and actual route sets **FP** and **RT**, respectively. The complexity of having additional semantic dimensions over the spatio-temporal domain is addressed efficiently by employing a proper similarity function, which is typically some linear vector norm. This enables the incorporation of any number of semantic parameters to be used, e.g. weather conditions, aircraft properties, etc, with limited impact on the overall complexity. Moreover, the clustering is performed using a properly designed semantic-aware similarity function that takes into account the entire input space instead of just the spatio-temporal proximity of flight trajectories. In other words, the proposed approach addresses the fact of flight plans that may seem identical in the spatio-temporal domain but may produce quite different realizations of actual routes due to different weather conditions. This is extremely important for creating clusters that are compact, not only in the spatio-temporal domain (geodesic proximity) but in the full semantic-enriched domain. Hence, the design of HMM states based upon these enriched waypoints incorporates this integrated semantic-enriched information, not directly as with HMMs that employ separate discretization for each domain dimension, but indirectly by having different semantic-aware clusters. To put it another

way, in this hybrid approach the different medoids upon which HMMs are trained are produced taking into account the full semantic-enriched domain, hence they already incorporate any significant semantic information available without explicitly increasing the complexity of the HMM design.

Another important difference between the proposed approach and the typical HMM methods used for the FSTP task is the emissions model. For example, in [9] and [11] the emissions sequence is constructed via Dynamic Time Warping (DTW) clustering of weather “paths” that are related to a specific state transition sequence. This approach is an effective one for designing HMMs where the Viterbi algorithm estimates the most probable state transition sequence, i.e., spatio-temporal movement of the aircraft, from the “observed” emissions sequence. However, this modeling is not fully associated to the physical world: the weather conditions are not “emitted” by the point-to-point movement of the aircraft; it is an environmental observation that is inherently unrelated to what the state-transition model generates as a result. Furthermore, this standard approach requires discretization of the emissions into “buckets” and combination with the states, in order to produce a fully discrete HMM model. In contrast, the proposed approach employs a very limited number of states, mapped against the waypoints of flight plans and actual routes, and an analytical probabilistic model for approximating the per-state emissions. Practically, every state in each HMM is accompanied with a Gaussian pdf that approximates the empirical emissions (FP/RT deviations). This way, the HMM “explains” precisely what the system generates as output and at the same time it is trained analytically by statistical modeling, instead of using emissions sequences to estimate state transitions.

It is worth noting that the proposed hybrid approach produces a multi-level organization of the flights by employing a two-stage training with the dataset: During the clustering stage, any statistically significant semantic-aware dissimilarity produces different clusters, i.e., splits the entire dataset for flights into groups. Therefore, if these clusters are produced compact enough, the HMM in the second stage is inevitably deterministic, i.e., the state transitions inherently become sequential. In other words, the path-finding problem is for the most part addressed already during the clustering stage, while the HMM stage subsequently uses the corresponding FP/RT deviations sequence for producing a maximum-likelihood emissions probabilistic model. This approach is much more natural and expected by such a FSTP algorithm: given a complete set of parameters (aircraft characteristics, flight plan, weather conditions) as prior constraints, the proposed method produces as output the most probable sequence of deviations from a given flight plan, i.e., the maximum-likelihood actual route waypoints where the aircraft will fly through.

In should also be noted that the proposed method is inherently generic. It does not rely on spatio-temporal grid sizes or resolution, number of semantic parameters or discretization of them. It does rely on pre-flight constraints, more importantly the flight plan that is associated with each

actual route, an updated semantic vector for each reference point (primarily localized weather), as well as a specific pair of departure/destination airports. The basic assumption here is that (a) all flights submit a flight plan prior to take off and (b) that all pilots have more or less the same incentives to follow it. This is the core factor upon which the FP/RT deviations are used as the output of a maximum-likelihood probabilistic formulation (HMM emissions), with error bound that can be estimated accurately via confidence interval statistical analysis. This formulation can be extended to cases where a completely new flight plan is submitted for a route never flown before by anyone (“outlier” in clusters), assuming (b) and exploiting the flight plan similarly in combination with the “closest match” for the deviations model, e.g. for the closest pair of airports, geographical region, number of waypoints, total length of flight, etc.

The applicability of this proposed hybrid clustering/HMM method to other domains is limited only by the availability and mandatory provision of appropriate constraints as input, i.e., something analogous to the flight plans. This includes maritime domain and even land transportations, if/when specific spatio-temporal constraints are present and available beforehand, e.g. estimated time of arrival (ETA) at specific critical points on the intended route. In this respect, the aviation domain can be considered as the most difficult and demanding, as it includes high dimensionality and increased variability in the semantic parameters in the short-term (e.g. altitude-dependent, localized weather). In this work, the aviation domain was chosen precisely for these reasons and the feasibility study of the proposed method was concluded successfully.

## VII. CONCLUSION

In this paper, we presented a novel Hybrid Clustering/HMM method for addressing the TP problem in the aviation domain, building upon and substantially improving the “blind” probabilistic methods that are currently used in the context of TP. Furthermore, our work enhances the content of the spatio-temporal trajectories towards with annotations (semantic information) that includes flight plans, localized weather and aircraft properties that enables modeling in an arbitrary N-dimensional domain. Clustering is employed as a first processing phase, using properly designed semantic-aware similarity functions as distance metrics, in order to cluster, in order to produce compact groups of similar trajectory evolutions. Then, HMMs are trained for each cluster, using non-uniform graph-based spatial grid and exploiting flight plans as constraints for a parametric probabilistic model for the emissions.

The proposed hybrid method exhibits at least an order of magnitude better accuracy in terms of absolute cross-track error compared to the current state-of-the-art with “blind” HMM for TP, while at the same time exhibiting two to three orders of magnitude less processing and storage resources. Further work will be focused on improvements with respect to designing proper semantic-aware similarity metrics and segmented-trajectory models, for very large training datasets.

## ACKNOWLEDGMENT

This work was partially supported by projects datACRON and DART, which have received funding from the European Union’s Horizon 2020 programme under grant agreements No 687591 and No 699299, respectively.

## REFERENCES

- [1] G. Andrienko, N. Andrienko, P. Bak, D. Keim and S. Wrobel, *Visual analytics of movement*, Springer Science and Business Media, 2013.
- [2] N. Pelekis and Y. Theodoridis, *Mobility Data Management and Exploration*, Springer, 2014.
- [3] Y. Zheng, “Trajectory Data Mining: An Overview,” *Transactions on Intelligent Systems and Technology*, vol. 6, no. 3, pp. 1-41, 2015.
- [4] G. Enea and M. Poretta, “A comparison of 4D-trajectory operations envisioned for NextGen and SESAR, some preliminary findings,” in *ICAS 2012*, 2012.
- [5] S. Sip and S. M. Green, “Common Trajectory Prediction Capability for Decision Support Tools,” in *ATM 5th USA/Europa R&D seminar*, Budapest, 2003.
- [6] R. A. Coppenbarger, *Climb Trajectory Prediction Enhancement Using Airplane Flight-Planning Information*, American Institute of Aeronautics and Astronautics (AIAA-99-4147), 1999.
- [7] D. P. Thippavong, C. A. Schultz and et.al., “Adaptive Algorithm to Improve Trajectory Prediction Accuracy of Climbing Aircraft,” *Journal of Guidance, Control and Dynamics (JGCD)*, vol. 36, no. 1, 2013.
- [8] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257-286, 1989.
- [9] S. Ayhan and H. Samet, “Aircraft Trajectory Prediction Made Easy with Predictive Analytics,” in *Proceedings of ACM SIGKDD 2016*, 2016.
- [10] A. Viterbi, “Error bounds for convolutional codes and an asymptotically optimum decoding algorithm,” *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260-269, 1967.
- [11] S. Ayhan and H. Samet, “Time Series Clustering of Weather Observations in Predicting Climb Phase of Aircraft Trajectories,” in *IWCTS 2016*, Burlingame (CA), USA, 2016.
- [12] K. Patroumpas, E. Alevizos, A. Artikis, M. Voudas, N. Pelekis and Y. Theodoridis, “Online Event Recognition from Moving Vessel Trajectories,” *Geoinformatica*, vol. 21, no. 2, pp. 389-427, 2017.
- [13] M. Ankerst, M. Breunig and et.al., “OPTICS: Ordering points to identify the clustering structure,” in *Proceedings of SIGMOD 1999*, New York (NY), USA, 1999.
- [14] L. Chen and R. Ng, “On the marriage of edit distance and Lp norms,” in *Proceedings of VLDB 2004*, 2004.
- [15] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Journal Information Processing and Management*, vol. 24, no. 5, pp. 513-523, 1988.
- [16] O. Cappe, “Online EM Algorithm for Hidden Markov Models,” *Journal of Comp. & Graph. Stat.*, vol. 20, no. 3, pp. 728-749, 2011.
- [17] G. van Brummelen, *Heavenly Mathematics: The Forgotten Art of Spherical Trigonometry*, Princeton University Press, 2013.