DART: A Machine-Learning Approach to Trajectory Prediction and Demand-Capacity Balancing

SESAR
Belgrade, Serbia November 28-30 2017

Pablo Costas
DART Project

- **DART**: Data-driven Aircraft Trajectory prediction Research
  - SESAR 2020 Exploratory Research
  - *Topic ER-02-2015 - Data Science in ATM*
  - June 2016-June 2018 (currently ongoing)

- **Objective**: Address the suitability of applying big data techniques for **predicting multiple aircraft trajectories** based on data-driven models and accounting for **ATM network complexity** effects

- **Focus on**:
  - Single Trajectory Prediction (WP2)
  - Multiple (Collaborative) Trajectory Prediction (WP3)

**Extended Objective**: Iterative **multi-criteria optimization process**, considering different stakeholders interests

**Link to DatAcron H2020 project** (discrete events forecasting for moving entities)
DART Concept

Objectives
DART will deliver understanding on the suitability of applying data-driven and agent-based models for enhancing our abilities to increase predictability of aircraft trajectories.

Increasing predictability <-> Reducing uncertainty


**DART Scenarios**

This scenario aims at analyzing and evaluating machine learning algorithms for trajectory predictions from an individual trajectory perspective (i.e. without considering traffic) from the airspace users’ point of view.

Once detected the sectors demand-capacity **imbalances and the potential conflicts**, there will be selected **those flights to modify** in order to remove the imbalances and conflicts.

For those flights to modify: i) a new FP from AOs preferred list will be selected and ii) a new single trajectory will be predicted (WP2)

**Multiobjective optimization process:**

i. Minimizing the sector imbalances and potential conflicts.

ii. Minimizing the cost thought maximizing the adherence to the airlines preferred FPs.
Single Trajectory Prediction

- A **trajectory** can be defined as the time-evolution of the position of the aircraft’s center of mass (and other state variables).
- A **predicted trajectory** is a representation of the aircraft’s future trajectory, typically given by a **chronologically ordered sequence of aircraft states**, where each state includes variables such as position (of the center of mass), speeds and weight.
- When using models to predict aircraft motion, additional variables are required to predict a trajectory: **aircraft performance characteristics**, **atmospheric variables**, **aircraft intent** and **initial aircraft state**.

Data ingestion and feature extraction

- **Surveillance Data**. Radar tracks of the Spanish airspace controlled by EnAire, the Spanish Air Navigation Service Provider (ANSP).
- **Weather Data**. Forecasts (downloaded from NOAA in grib format), Significant Meteorological Information (SIGMET), Meteorological Aerodrome Report (METAR) and Terminal Aerodrome Forecast (TAF).
- **Flight Plans**. Standard dataset generated by Airspace Users (AU) and agreed with the ANSPs, that represents an intended flight or portion of a flight. The FPs considered within DART are those stored in the Spanish ATC operational system, and include all flight plan amendments associated to the originally filed FP (GIPV from SACTA).
- **Airspace structure**. The airspace is organized in accordance with the envisioned traffic flown and the availability of resources to manage that traffic. Includes both possible and applied sector configurations.
- **Re-constructed trajectory**. Extended trajectory information that includes additional aircraft state variables that are not included in the surveillance datasets (e.g., airspeeds, mass, and the like) with higher data sampling.
- **Aircraft Intent Description**. Semantic description of a trajectory that represents the set of instructions to be executed by the aircraft in order to realize its intended trajectory, equivalent to the commands issued by the pilot or the FMS to steer the aircraft.
Single Trajectory Prediction

Feature Extraction

Re-constructed trajectory. Extended trajectory information that includes additional aircraft state variables that are not included in the surveillance datasets (e.g., airspeeds, mass, weather conditions, ...) with higher data sampling.

Aircraft Intent Description. Semantic description of a trajectory that represents the set of instructions to be executed by the aircraft in order to realize its intended trajectory, equivalent to the commands issued by the pilot or the FMS to steer the aircraft.
Aircraft Intent Example

**Motion Profiles**

<table>
<thead>
<tr>
<th>1st DOF</th>
<th>2nd DOF</th>
<th>3rd DOF</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS (M)</td>
<td>HA (P)</td>
<td>TLP (GC)</td>
</tr>
<tr>
<td>TOD</td>
<td>TL (IDLE)</td>
<td></td>
</tr>
<tr>
<td>CAS=280kt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>h=4500ft</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HA (CAS)</td>
<td>HS (CAS)</td>
<td>HS (CAS)</td>
</tr>
<tr>
<td>HA (GEO)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Configuration Profiles**

- HL
- SB
- LG

**Aircraft Trajectory**

- FL320
- TOD
- M .88
- 280 KCAS
- 4500ft
- 180 KCAS

- N370945.72
- W0032438.01
- OT5
- A
- B
- 110
- R

**Motion Profiles**

- HA
- HL
- HHL
- HS
- HSB
- HLG

**Time**
Single Trajectory Prediction
Hidden Markov Models

Given a set of historical raw or reconstructed trajectories for specific aircraft types along with pertinent historical weather observations, we aim at learning a model that reveals the correlation between weather conditions and aircraft positions and predicts trajectories in the form of a time series.

States $S = \{S_1, S_2, ..., S_K\}$ are represented by reference points’ coordinates (latitude, longitude, altitude) that form aligned trajectories.

Transition probabilities $A = \{a_{ij}\}$, $1 \leq i, j \leq K$, i.e. $a_{ij}$ is the probability of an aircraft discretely transitioning from one state $S_i$ to another $S_j$ along its aligned trajectory, $T$.

Emission probabilities $B = \{b_i(o)\}$, $1 \leq i \leq K$ is the probability of discrete weather parameters having been observed at a specific state, $S_i$.

Initial probabilities $\pi = \{\pi_i\}$, $1 \leq i \leq K$ is the probability of an aligned trajectory beginning at a specific state, $S_i$. 
Single Trajectory Prediction
Hidden Markov Models

<table>
<thead>
<tr>
<th>Case</th>
<th>Route1</th>
<th>Training Set Size</th>
<th>Test Set Size</th>
<th>Route2</th>
<th>Training Set Size</th>
<th>Test Set Size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>#trjs  #pts</td>
<td>#trjs  #pts</td>
<td></td>
<td>#trjs  #pts</td>
<td>#trjs  #pts</td>
</tr>
<tr>
<td>1</td>
<td>LEAL-LEBL</td>
<td>1118  55116</td>
<td>200  9860</td>
<td>LEMD-LEIB</td>
<td>2572  125623</td>
<td>200  9769</td>
</tr>
<tr>
<td>2</td>
<td>LEAL-LEBL</td>
<td>1118  55116</td>
<td>19   937</td>
<td>LEMD-LEMH</td>
<td>1056  68141</td>
<td>19   1226</td>
</tr>
<tr>
<td>3</td>
<td>LEAL-LEBL</td>
<td>1118  55116</td>
<td>152  7493</td>
<td>LEPA-LEMD</td>
<td>5116  306128</td>
<td>152  9095</td>
</tr>
<tr>
<td>4</td>
<td>LEBL-LEMG</td>
<td>1704  127451</td>
<td>43   3216</td>
<td>LEMD-LEMH</td>
<td>1056  68141</td>
<td>43   2775</td>
</tr>
<tr>
<td>5</td>
<td>LEBL-LEMG</td>
<td>1704  127451</td>
<td>180  13463</td>
<td>LEPA-LEMD</td>
<td>5116  306128</td>
<td>180  10771</td>
</tr>
<tr>
<td>6</td>
<td>LEBL-LEAZL</td>
<td>2404  183343</td>
<td>41   3127</td>
<td>LEMD-LEAM</td>
<td>1434  70128</td>
<td>41   2005</td>
</tr>
<tr>
<td>7</td>
<td>LEBL-LEAZL</td>
<td>2404  183343</td>
<td>46   3508</td>
<td>LEMD-LEMH</td>
<td>1056  68141</td>
<td>46   2968</td>
</tr>
<tr>
<td>8</td>
<td>LEBL-LEAZL</td>
<td>2404  183343</td>
<td>164  12508</td>
<td>LEMG-LEMD</td>
<td>1403  75408</td>
<td>164  8815</td>
</tr>
<tr>
<td>9</td>
<td>LEBL-LEAZL</td>
<td>2404  183343</td>
<td>210  16016</td>
<td>LEPA-LEMD</td>
<td>5116  306128</td>
<td>210  12566</td>
</tr>
<tr>
<td>10</td>
<td>LEIB-LEBL</td>
<td>1360  53443</td>
<td>259  10178</td>
<td>LEPA-LEMD</td>
<td>5116  306128</td>
<td>259  15498</td>
</tr>
<tr>
<td>11</td>
<td>LEIB-LEBL</td>
<td>1360  53443</td>
<td>158  6209</td>
<td>LEPA-LEVC</td>
<td>1426  50467</td>
<td>158  5592</td>
</tr>
<tr>
<td>12</td>
<td>LEMG-LEBL</td>
<td>1563  114767</td>
<td>38   2790</td>
<td>LEMD-LEAM</td>
<td>1434  70128</td>
<td>38   1858</td>
</tr>
<tr>
<td>13</td>
<td>LEMG-LEBL</td>
<td>1563  114767</td>
<td>46   3378</td>
<td>LEMD-LEIB</td>
<td>2572  125623</td>
<td>46   2247</td>
</tr>
<tr>
<td>14</td>
<td>LEBZ-LEBL</td>
<td>2380  186299</td>
<td>40   3131</td>
<td>LEMD-LEIB</td>
<td>2572  125623</td>
<td>40   1954</td>
</tr>
</tbody>
</table>
Single Trajectory Prediction
Clustering + HMM

1st Step: Clustering semantic trajectories

2nd Step: For each cluster train a HMM

3rd Step: (Filter) Given a flight plan Q find top-k most probable HMM models

4th Step: (Refine) Similarity search among the semantic trajectories that belong to the top-k HMMs

“Annotated” Trajectories (FP, weather,...) Clustering with ad-hoc distance functions (not just spatio-temporal but weather, date, etc...)

Non-uniform graph-based spatial grid FP Waypoints are used as reference for HMM states

Waypoint-to-waypoint matching to medoids 3-D deviation (Haversine distance)

Semantic distance metric (points):
\[ D_{\text{dist}}(r_i, r_j) = \lambda \cdot \text{dist}_w(r_i, r_j) + (1 - \lambda) \cdot \text{dist}_s(r_i, r_j) \]
\[ \text{dist}_w(r_i, r_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \]
\[ \text{dist}_s(r_i, r_j) = 1 - \frac{\text{std}(r_i, r_j)}{\text{std}(r_i, r_j) + \text{std}(r_j, r_j)} \]

Semantic trajectory distance:
\[ D_{\text{st}}(R_i, R_j) = \lambda \cdot D_{\text{st}}^w(R_i, R_j) + (1 - \lambda) \cdot D_{\text{st}}^s(R_i, R_j) \]

SESAR INNOVATION DAYS 2017
Single Trajectory Prediction Clustering + HMM

Using the formulation above, this two-phase hybrid clustering/HMM approach was tested in a benchmark dataset of actual flight trajectories (around 1400 flights). One airport pair was considered from the Spain airspace (Barcelona/Madrid) and each direction was modeled separately, as it involves different flight plans and takeoff/landing approaches.

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.

Bearing clustering, represented in $t, \cos(\chi), \sin(\chi)$

Using the formulation above, this two-phase hybrid clustering/HMM approach was tested in a benchmark dataset of actual flight trajectories (around 1400 flights). One airport pair was considered from the Spain airspace (Barcelona/Madrid) and each direction was modeled separately, as it involves different flight plans and takeoff/landing approaches.

Figure illustrates the per-waypoint means and confidence intervals for Latitude in cluster 1 as described above. The height of each bounding box is directly linked to the uncertainty associated with producing the maximum-likelihood deviation from the HMM emissions in each reference waypoint, i.e., the difference between the flight plan and the aircraft actual route.

The height of each box, i.e., the size two central quartiles, is directly linked to the statistical uncertainty in predicting each dimension of the pair-wise deviations between flight plans and the cluster medoid.
Collaborative Trajectory Prediction

**Scope:** This scenario objective is to demonstrate how DART predictive analytics capability can help in trajectory forecasting when demand exceeds capacity (from a global perspective), at **planning phase** (pre-tactical).

**D>C**

- System capacity is not enough
- Some flights must be delayed → regulation
- Delays are expensive and problematic

Measures will be applied to the WP2 trajectories due to the imbalance between demand and capacity

**Goal:** Improve **global** predictability *(relying on accurate planning information)*
Collaborative Trajectory Prediction

Approach: Formulate a Markov Decision Process

- Solving MDP = planning
- Reinforcement Learning methods are considered appropriate
- Multi-agent RL approach inherently appropriate
- MDP requires a Reward Model
  - Reward functions take into account participation of trajectories to hotspots and delays imposed - later results consider AU's preferences in terms of strategic delay cost, as well.
  - *Other options resulted in a huge state-action space (i.e. keeping locations, or action = next target location)*

- State vector:
  - $d_i \in \mathbb{N}$ Imposed delays so far (1 per flight)
  - No need to keep locations in states
  - Time is factored out

- Actions per agent: 0 or 1 (while on ground) UNTIL a MAX delay is reached.
  - Available options limited by flight plan
- OR:
- Actions per agent: 0 ... MAX delay
  - Available options limited by flight plan

Reward Model

- Capacity: $c = \text{capacity of sector}$
- Demand: $\text{demand}(i, j) = \text{number of flights in time slot } j$
- DCB violations: $\text{hotspot}(c, i, j) = 1$ if $\text{demand}(i, j) > c$, 0 otherwise

\[
Rwd_A(s^t_A, s^{t+1}_A) = \lambda_1 \cdot C(s^t_A, s^{t+1}_A) + \lambda_2 \cdot D(s^t_A, s^{t+1}_A)
\]

C: A function that takes into account the number of hotspots and the “contribution” of flights in them (in terms of duration of being involved in a hotspot)

D: A function that depends only on the delays imposed to flights: Currently this is translated into strategic delay cost.

str: The strategy of agents - i.e. their chosen delay.

This function aims to reduce hotspots (via the minimization of flights contribution to delays) and delays (costs due to delays) imposed to flights
Collaborative Trajectory Prediction

Reinforcement Learning approach considered (in progress):

1. **Independent Learners** approach (*Ind-Colab-RL*)
   
   *Each agent (flight) is self-interested and learns by itself to resolve the DCB problem, by taking into account own state and measuring its own reward after each decision.*

2. **Sparse Collaborative Q-Learning – Agent-based decomposition – Edge based update** (*Ed-Colab-RL*)
   
   *This is a variant of the sparse cooperative edge-based Q-learning method. Multiple agents are jointly interacting with the environment and decide by taking into account joint state and measuring individual reward after each joint decision.*

3. **Sparse Collaborative Q-Learning – Agent-based decomposition – Agent based update** (*Ag-Colab-RL*)
   
   *This is a variant of the agent-based update sparse cooperative edge-based Q-learning method that allows agents to share their joint reward after joint decision.*
Collaborative Trajectory Prediction

Experiments’ set up

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>grid structure of sectors</td>
<td>$4 \times 4$</td>
</tr>
<tr>
<td>capacity of sectors, $C$</td>
<td>$[4, 10]$</td>
</tr>
<tr>
<td>number of planes, $N$</td>
<td>100</td>
</tr>
<tr>
<td>Duration and Step of Occupancy Counting Period</td>
<td>6</td>
</tr>
<tr>
<td>total time period duration $H$</td>
<td>180</td>
</tr>
<tr>
<td>maximum delay</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 1. Parameter values used during the simulated experiment.

Experimental Results: Results achieved for different sectors’ capacities
Collaborative Trajectory Prediction

Experimental Results: Learning curves showing convergence to solution.

Experimental Results: Results shown final demand to periods in sectors
Collaborative Trajectory Prediction

Experimental Results: Highlights

• The results confirm that the proposed multi-agent formulation provides a promising framework for tackling the DCB problem.
• All methods demonstrated very similar behavior by eradicating hotspots with Edge Based Update being slightly more effective compared to others in terms of the number of hotspots and mean delay achieved, but less efficient than Agent Based Update in terms of convergence speed.

Latest Experimental Results on Real-World Scenario

• A scenario has been drafted using real world data for the 23d of November 2016.
• Delay cost of the Flights has been added to the reward function.
• New deterministic rule (+DR) that facilitates the exploration, by pruning the state space has been applied. The deterministic rule allows flights NOT participating in hotspots to get delay equal to 0, this reduced the search space and thus increased methods computational efficiency.
Collaborative Trajectory Prediction

Latest Experimental Results on Real-World Scenario

<table>
<thead>
<tr>
<th>Method</th>
<th># flights delayed</th>
<th>Total delay</th>
<th>Mean delay (delayed flights)</th>
<th>Mean delay (all flights)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ind-Colab-RL</td>
<td>322</td>
<td>7871</td>
<td>24</td>
<td>2.14</td>
</tr>
<tr>
<td>Ind-Colab-RL + DR</td>
<td>473</td>
<td>16589</td>
<td>34</td>
<td>4.51</td>
</tr>
<tr>
<td>Ed-Colab-RL</td>
<td>280</td>
<td>4668</td>
<td>16</td>
<td>1.27</td>
</tr>
<tr>
<td>Ed-Colab-RL + DR</td>
<td>310</td>
<td>6866</td>
<td>22</td>
<td>1.87</td>
</tr>
<tr>
<td>Ag-Colab-RL</td>
<td>160</td>
<td>3743</td>
<td>23</td>
<td>1.02</td>
</tr>
<tr>
<td>Ag-Colab-RL + DR</td>
<td>271</td>
<td>8284</td>
<td>30</td>
<td>2.54</td>
</tr>
<tr>
<td>Actual</td>
<td>269</td>
<td>4130</td>
<td>16</td>
<td>1.12</td>
</tr>
</tbody>
</table>

Learning curves
Showing convergence to zero hotspots (left) and mean-delay (right)

Distribution of delay (mins) to flights achieved by RL methods compared to the actual ones.
Collaborative Trajectory Prediction

Latest Results from real world Scenario

• The methods manage to provide solutions to the DCB problem, imposing delays that result to zero hotspots.

• The new deterministic rule considerably increases the methods’ performance.

• Agent Based Update is more effective in terms of the number of flights delayed, also compared to the actual delays.
DART: A Machine-Learning Approach to Trajectory Prediction and Demand-Capacity Balancing

Thank you very much for your attention!

This project has received funding from the SESAR Joint Undertaking under the European Union’s Horizon 2020 research and innovation programme under grant agreement No [number]

The opinions expressed herein reflect the author’s view only. Under no circumstances shall the SESAR Joint Undertaking be responsible for any use that may be made of the information contained herein.