Clustering airport surface trajectories to identify anomalies

Andrew Churchill
Michael Bloem
Mosaic ATM
Motivation for airport surface anomaly detection

• The *airport surface* is a complex system, with a history of significant R&D to ensure safety and improve efficiency
  • Seems safer, in some sense, because *aircraft can stop*

• Many models and analyses of airport surface operations rely on representations of paths through the taxiway network, but these are expensive to generate

• One interesting application of such paths is *anomaly detection*  
  • Anomaly detection is a broad area of research across domains, often leveraging machine learning to create accurate models of typical/atypical system behavior  
    • Aircraft not on a known path are per se anomalies

• Increasing traffic density and an evolving fleet mix on/near the airport surface add significant complexity for controllers maintaining safe operations, so we are developing algorithmic foundations for an advisory tool
Modeling approach
Design for technical approach

• Clustering seems like a natural approach; some previous research in aviation (airborne trajectories) and in other domains on clustering/mining trajectories
• Desire results in both the spatial and temporal domains

• Initial attempts at clustering trajectories in space-time yielded critical observation that *addressing stops in surface trajectories is essential*
  ➢ Development of two-level hierarchical approach to separate space and time
• Measurement noise in position observations introduced significant complexity
  ➢ Development of filtering/resampling/preparation approach to “clean” trajectories
• Lack of obvious distance metric to use for time-based level of hierarchy
  ➢ Development of within-cluster, progress-along-centroid metric
• No standard methodology for setting hyper-parameters of clustering algorithm
  ➢ Development of constrained grid search and heuristic approach to calibrating model
Space-based clustering with DBSCAN

**Density-Based Spatial Clustering of Applications with Noise**

- “Views clusters as areas of high density separated by areas of low density”
- Builds clusters by iteratively finding “core” points in areas of high density
- **Points with too few or too distant neighbors are assigned to no cluster**
  - *Common approach to treat these as outliers or anomalies*
- **Benefits:**
  - Previously applied to airborne aircraft trajectory clustering
  - No assumption of cluster convexity
  - Automatically identifies trajectories as outliers (i.e., anomalies)
- **Challenges:**
  - Preparing data to match input assumptions
  - Calibrating hyper-parameters
trajectory processing (for each trajectory)

raw trajectory

- resampling evenly over time
- low-pass filter
- filtered trajectory

- resampling evenly over distance
- concatenate into vector & standardize

Level 1 clustering: space-based

Level 2 clustering: time-based (within each space-based cluster)
Filtering to remove measurement noise

• Other research prepared trajectory data for clustering by resampling trajectories at even intervals of time or distance traveled
• Measurement noise makes results of such resampling misleading, particularly when aircraft stop
Filtering to remove measurement noise

• Passing trajectory latitude and longitude time series through a low-pass filter removes measurement noise

• Noise still has some influence on trajectory shape at stopping points, but more aggressive filtering compromises other aspects of trajectories, like turns
Other data preparation steps

• Resampling evenly over distance
  • To prepare trajectories for shape-based clustering, we resampled the filtered trajectories evenly over distance into a standard number of points
  • Filtering ensured that stops had negligible impact on the shape of these resampled trajectories

• Concatenate and standardize
  • Clustering requires each trajectory to be represented as an equal-length vector
  • Trajectories evenly sampled over distance traveled were concatenated into a vector
  • Vectors standardized so that variations in latitude and longitude had equal contribution in distance metric and thus in clustering
trajectory processing (for each trajectory)

raw trajectory

resampling evenly over time

filtered trajectory

low-pass filter

concatenate into vector & standardize

[···]

Level 1 clustering: space-based

DBSCAN

space-based clusters & outliers

Level 2 clustering: time-based (within each space-based cluster)
Parameter selection for space-based clustering

• DBSCAN has two hyper-parameters: Within distance $\text{eps}$, there must be $\text{min\_samples}$ other points for a point to be labeled as “core”

• Somewhat competing desiderata:
  • Not too many outliers/anomalies: less likely that reporting false positives
  • High quality clusters: difficult to quantify; we used the Silhouette coefficient

• Silhouette coefficient definition (higher $\rightarrow$ “better defined”):
  • Silhouette coefficient $= \text{mean}(s_i) \in [0,1]$
  • $s_i = \frac{b_i - a_i}{\max(a_i,b_i)} \in [0,1]$
  • $a_i = \text{mean distance between point i and all others in the same cluster}$
  • $b_i = \text{mean distance between point i and all others in next nearest cluster}$
Parameter selection for space-based clustering

Constrained grid search

Each point represents results for a single pair of \texttt{eps} & \texttt{min_samples} parameter values.

Selected parameters: maximize the Silhouette coefficient while respecting upper bound on fraction of outliers.
trajectory processing
(for each trajectory)

- raw trajectory
  - resampling evenly over time
  - resampling evenly over distance
  - project onto space-based cluster centroid

- filtered trajectory
  - low-pass filter
  - concatenate into vector & standardize
  - extend by padding

Level 1 clustering: space-based

- DBSCAN
  - space-based clusters & outliers
  - compute centroids
  - space-based cluster centroids

Level 2 clustering: time-based

- DBSCAN
  - time-based clusters & outliers
  - compute centroids
  - time-based cluster centroids
Projecting onto space-based cluster centroid

• To develop time-based clustering metric, leverage results of shape-based clustering

• First project each point on the filtered trajectories to the nearest point on its cluster centroid trajectory

• This should almost completely remove variations and isolate the time dimension
Time-based clustering distance metric

• Metric is then the $\ell_1$-norm of the distance between the projected trajectories, measured along the cluster centroid trajectory and normalized by the length of the centroid trajectory

$$D(i, j) = \sum_{t=0}^{T} \left| d_i(t) - d_j(t) \right|$$

$d_i(t)$ = distance along cluster centroid, expressed as a fraction of total centroid distance

$D(i, j)$ is proportional to the total area between the lines
Study results
Study data

• Initial analysis focused on Charlotte Douglas International Airport (CLT), because of existing familiarity with operations there

• Results presented here use one week of ASDE-X surface surveillance data covering the active movement area

• Dataset well-balanced between arrivals and departures, but each was evaluated separately for simplicity

• Now extending results to longer samples and JFK airport
Characterization of shape-based clusters

- arrivals
- departures
- outliers

- roughly linear on semi-log plot
- \( \rightarrow \) roughly exponential decline in size with rank

- many more and much smaller arrival clusters
- \( \rightarrow \) arrivals use a greater variety of runways and taxiways
Largest shape-based arrival clusters
Variation within largest arrival cluster
Shape-based arrival anomaly
Small arrival cluster as anomaly
Largest shape-based departure clusters
Variation within largest departure cluster
Shape-based departure anomaly
Characterization of time-based clusters

Approximately 50% of shape-based clusters have 5% or fewer outliers when clustering in the time domain.

→ Within many shape clusters, there is considerable homogeneity with respect to time.
Time variation within largest shape-based cluster

Only one time-based cluster was identified.

Distance along shape cluster centroid trajectory [meters]

While clustered trajectories are mostly uninhibited, cluster centroid suggests typical stop for a few seconds at ~1250 m.

Outlier trajectories each had stops at different locations or of different durations.

Trajectory in cluster 0
Cluster 0 centroid
Outlier trajectory

Time since start of trajectory [seconds]
Stop in largest shape-based arrival cluster

Stop in time-based cluster centroid at ~1250 m is for crossing of runway 18/36C
Time variation within largest departure cluster

- The largest cluster had minimal delays reaching the runway.
- Other cluster represents some trajectories that had multiple similar stops.
- A number of trajectories were still left out of any time-based cluster because of the diversity in when and where flights stopped along their trajectory.

Graph details:
- Distance along shape cluster centroid trajectory [meters].
- Time since start of trajectory [seconds].
- Cluster 0 centroid.
- Trajectory in cluster 0.
- Cluster 1 centroid.
- Trajectory in cluster 1.
- Outlier trajectory.
Time-based departure anomaly

distance along shape cluster centroid trajectory [meters]
Continuing work

• Trajectory processing
  • Development of alternative (sophisticated) filtering approach (multi-modal UKF)
  • KFs should work better with real-time data than low-pass
  • “Point squashing” heuristic to bluntly remove observations at stops
  • Leveraging airport adaptation (graph representation) data?

• Space-based clustering
  • Exploration of SSPD metric
  • Looking at OPTICS or HDBSCAN instead of DBSCAN. These will have one fewer hyper-parameter to tune, and can work with clusters of varying densities.

• Time-based clustering
  • Explore alternative measures (e.g., velocity vs distance, hides duration of stop)
  • Leverage modes from KFs (e.g., stopped) to help identify operationally relevant events, e.g., failing stop at expected locations, stopping at unexpected locations