Abstract—Given the evolution of the National Airspace System (NAS), it has become more critical to effectively identify safety threats or emerging risks. In this paper, we propose an unsupervised anomaly detection algorithm to generate a model which can be applied in real time to detect terminal airspace anomalies. The algorithm is based on temporal logic learning, which provides formulas that are easy to be interpreted and hence facilitate human feedback for identifying only operationally significant anomalies. The proposed algorithm is demonstrated with airport surface surveillance data.

Keywords-component; anomaly detection; temporal logic; aviation safety

I. INTRODUCTION

The National Airspace System (NAS) is vast and complex, including many subsystems. A substantial amount of data is being recorded, whose size will grow as the National Airspace System (NAS) evolves with additional sensing and data collection capabilities and newly deployed systems. In this regard, it has become more critical to effectively identify (a priori unknown) safety threats or emerging risks. Aviation data are typically unlabeled, which leads to apply unsupervised learning approaches to support anomaly detection in the NAS.

Anomaly detection is the problem of identifying events or observations that do not conform to expected behaviors in a dataset. Especially for sequential (or time-series) data such as flight data, several techniques for anomaly detection have been proposed such as Gaussian Processes (GP) [1] and Hidden Markov Model (HMM) [2][3]. Among the anomalies detected by such methods, which are statistically significant, we can incorporate the human feedback to identify only operationally significant anomalies. In general, these methods infer a surface in a high-dimensional feature space which separates normal and anomalous data. However, it is in general hard to interpret the meanings of the surfaces, especially in on-line monitoring [4]. In this sense, we propose a Temporal Logic Learning based Anomaly Detection (TempAD) algorithm, which provides formulas that are easy to be interpreted in natural languages. The learned temporal logic formulas can express system properties such as bounds on time and physical parameters which have physical meanings. In the following sections, we present the data preprocessing procedure followed by the details of TempAD algorithm. Initial results for an illustrative case are then discussed with future work.

II. DATA PREPROCESSING

In this paper, we focus on the analysis on flight trajectories during the final approach, which is crucial part of safe air traffic operations and likely to have anomalous flights due to the complexity of operations, with the Airport Surface Detection Equipment – Model X (ASDE-X) data recorded at LaGuardia (LGA) Airport during the period of April 6 – 24, 2016 (19 days). The data contains the flight ID of a flight with its position (latitude, longitude, altitude) and speed (ground speed) information.

Among the trajectories in the ASDE-X data, only the flights arriving at the LaGuardia airport (LGA) are first filtered out. For example, there are 563 flights for a single day (April 6, 2016). Since a single airport contains multiple arrival patterns, instead of dealing with them as a whole, by examining groups with similar properties individually, data analysis and anomaly detection can be made more effectively and efficiently. In this sense, the flight data is first divided into clusters (groups with similar properties) with clustering techniques. Clustering techniques can be broadly categorized as hierarchical, partitioning, density-based, and grid-based methods [5][6]. Since our data is spatial in nature, and contains abnormal flights (or noises), we propose to use a density-based method, called DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [7]. DBSCAN has another advantage that there is no requirement to set the number of clusters a priori.

As the size of recorded data are different for each flight, the horizontal tracks are resampled into the same number of points (40 points in this paper) using linear interpolations. The resultant data, therefore, has the dimension of 563 (the number of flights) by 80 (40 for longitude and 40 for latitude) for the example day. Then, DBSCAN is applied to the resampled data to find clusters in the horizontal trajectories. By repeating the above procedure for all 19 days, it is found that there are 12
patterns in the arrival flights at LGA approaching from the northeast or the southwest, as shown in Fig. 1, with 2 to 4 patterns in each runway. Section III will describe a proposed algorithm that can detect anomalies within each pattern identified by DBSCAN.

Figure 1. Identified arrival trajectory patterns at LGA for 19 days (April 6 – 24, 2016)

III. ALGORITHM

The overall architecture of the proposed learning algorithm, called TempAD (Temporal Logic Based Anomaly Detection), for unsupervised anomaly detection consists of Training and Monitoring, as shown in Fig. 2:

- In the Training stage, a set of time-series data is fed into TempAD to generate a model that separates normal and anomalous time-series data; and
- In the Monitoring stage, the learned model is used to compute an anomaly score for an unseen time-series data.

A. Discrete Search

The temporal logic formula can express system properties including bounds on time and physical variables. For example, the following formula

$$\phi = \varphi_0 = \left( F_{[2,27]}(x,y) \in R_1 \land C_{[30,31]}(x,y) \in R_2 \right)$$  \hspace{1cm} (1)

where $R_1 = \{(x,y) \mid f_1(x,y) > 0\}$ and $R_2 = \{(x,y) \mid f_2(x,y) > 0\}$, consists of (i) its structure ($\varphi$) that determines “Finally” (F) or “Globally” (G), “and” (\&) or “or” (\lor) relations, and the form of the predicates (e.g., $f_1(x,y) > 0$) and (ii) its parameters (\theta) that specify the bounds on time and physical variables. This can be interpreted in natural language as “for the normal behavior, the system’s variables $x$ and $y$ should reach area $R_1$ (e.g., reach $x+3y > 27$) in any time between 2 and 27, and also should reside in area $R_2$ (e.g., maintain $y^2+3x+y < 15$) during the time steps between 30 and 31.” In the next section, we discuss the details of the discrete search for finding the structure of a formula and the continuous search for computing the parameters of the formula.

The form of the predicates can be any functions such as polynomials, exponentials, or logarithms. For example, in [4], only a simple form of predicates with a single variable such as $x < \pi$ ($x$ is a variable and $\pi$ is a constant) was used to describe the behavior of a system. However, this simple formula cannot accurately describe the aircraft behaviors since via the flight data analysis, it is found that aircraft trajectories in a cluster show a piecewise polynomial form with linear or quadratic polynomials, as shown in Fig. 4. Therefore, we propose a discrete search algorithm based on piecewise regression which can compute piecewise polynomial functions that can accurately describe the given flight data. To find such piecewise polynomial functions, it is required to determine (i) the number and locations of breakpoints that separate the centroid (mean) of aircraft trajectories into sections, and (ii) the order and coefficients of a polynomial in each section. For the breakpoints, it is observed through data analysis that a centroid for the arrival flights has no more than three breakpoints (that is, maximum of two). Also, for each section (divided by the
breakpoints) of the centroid, either linear or quadratic polynomials can accurately represent horizontal trajectories, and a linear polynomial can do for the vertical trajectories and speed. In this regard, the piecewise regression is performed by a grid search method that finds a piecewise polynomial of the minimum difference (or error) between the polynomial and the given centroid by considering (i) the number of break points up to three, (ii) the locations of breakpoints along the centroid, and (iii) the order of polynomial up to one (linear) or two (quadratic) depending on the dimension (horizontal, vertical, and speed). The discrete search results for the centroids of flights with piecewise regression are shown in Fig. 4. For example, in the horizontal dimension, the centroid is represented as two sections divided by the breakpoint at (–73.9241, 40.7140) and the polynomial representing each section is computed as: 

\[ -x + 1.0304y = 115.8777 \] for Section 1 (red) and 

\[ x + 74.9528y^2 - 6108.9y = 12454.6 \] for Section 2 (blue), where \( x \) is the longitude and \( y \) is the latitude (both in degrees).

\[ \mu_i = \begin{cases} \varepsilon & \text{if } r(s^{(i)}, \varphi_{\theta}) > \varepsilon \\ 2 & \text{otherwise} \end{cases} \]

where the robustness degree \( r(s^{(i)}, \varphi_{\theta}) \) is a signed distance of data point \( s_i \) from the learned formula \( \varphi_{\theta} \). The positive/negative and larger robustness degree of a time-series means that the time-series satisfies/violates formula \( \varphi_{\theta} \) and a larger perturbation is required for the time-series to violate/satisfy formula \( \varphi_{\theta} \), respectively.

This is to minimize the number of time-series data that formula \( \varphi_{\theta} \) classifies as anomalous. The continuous search part is a nonlinear optimization problem that is highly sensitive to an initial guess and noise, leading to numerical issues (e.g., numerical instability and slow convergence rate). To overcome these problems, we propose to use sequential convex programming [8] that approximates the original nonconvex function as a convex function in the trust region, and sequentially solves the approximate convex problem by updating the trust region.

IV. DEMONSTRATION WITH FLIGHT DATA

We have tested the performance of the developed TempAD algorithm with the arrival flights to the LaGuardia (LGA) airport for the period of 14 days (April 6 – 19, 2016) of the ASDE-X data, which contains 6,662 flights. In Table 1, the number of anomalous flights detected in each dimension is presented. During this period, the total number of 67 flights have been identified as anomalous, which is 1.01% (67/6,662) of the flights recorded for 14 days.

<table>
<thead>
<tr>
<th>Day</th>
<th>H</th>
<th>V</th>
<th>S</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>8</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>3</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>5</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>14</td>
<td>3</td>
<td>4</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>7</td>
<td>5</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>19</td>
<td>6</td>
<td>7</td>
<td>0</td>
<td>6</td>
</tr>
<tr>
<td>Total</td>
<td>47</td>
<td>54</td>
<td>2</td>
<td>36</td>
</tr>
</tbody>
</table>

H: Horizontal, V: Vertical, S: Speed, C: Combined

By analyzing the identified anomalies, it is found that the 67 anomalies can be categorized into four types: (i) 36 flights of go-around, (ii) 11 flights of horizontal deviation, (iii) 18 flights of altitude maneuver, and (iv) 2 flights of speed anomaly. To demonstrate the training and monitoring steps in Fig. 2, we present a go-around example. As shown in Fig. 5,
the models learned by TempAD (blue lines) describe the normal behaviors so that anomalous trajectories (red lines) can be detected. For example, the model in the horizontal dimension is computed as:

\[
G_{[t_0,t^*]}((x,y) \in R_1) \land G_{[t^*,t]}((x,y) \in R_2)
\]

(3)

where \(t_0\) is the time remaining to touchdown at the beginning, \(t^* = 178.08\) (around three minutes before touchdown), and \(R_1\) and \(R_2\) are the areas defined by linear and quadratic polynomials as:

\[
R_1 = \{(x,y) | -x + 0.8065y > 106.7516 \text{ and } -x + 1.4286y < 132.1086\}
\]

\[
R_2 = \{(x,y) | x + 57.2788y^2 - 4668.0y > 95178.9 \text{ and } x + 68.3416y^2 - 5571.1y < 113612.2\}
\]

With some visualization aids such as Fig. 5, the model in (3) can be interpreted in natural language as “an aircraft flying normally should reside in area \(R_1\) up to 3 minutes before touchdown and then \(R_2\) until touchdown.”

For the real-time monitoring purpose, the anomaly score is computed by using the concept of robustness degree that represents a signed distance of a time-series data to the learned model: if it is positive, then the data is normal, and if negative, then anomalous. As an example, Fig. 5 shows the models learned by TempAD (blue) and its use for monitoring for a go-around flight (red). Note that for the vertical and speed dimensions, the models are learned for both upper and lower bounds. It is shown that all the anomaly scores in the three dimensions are maintained positive until the vertical trajectory violates the lower bound (82 seconds before touchdown). After 68 seconds, the aircraft takes a go-around maneuver which is identified by the negative anomaly scores in all three dimensions.

V. CONCLUSIONS

In this work, an unsupervised anomaly detection algorithm is proposed using temporal logic learning. The learned model in the form of temporal logic formula are easy to be interpreted in natural language and can be used in real-time monitoring for safety. In the future, the proposed TempAD will be further extended (i) to incorporate the human feedback by facilitating the characteristics of the temporal logic, and (ii) to capture a causal relation, that is, to identify a precursor to the detected anomaly.

ACKNOWLEDGMENT

This work is supported by the NASA Big Data Analytics Project (NNA16BF04C). The authors are grateful to Dr. Nikunj C. Oza at NASA Ames for his valuable comments and support.

REFERENCES