Managing Flight Pattern Visualization Complexity

Paul Peng Yu Mou (ppymou@uwaterloo.ca)
Annie Cho (acho@uwaterloo.ca)
Jonathan Histon (jhiston@uwaterloo.ca)
University of Waterloo

Abstract

Air traffic control (ATC) is a complex task requiring controllers to maintain order and safety in a given airspace. Research into increasing the safety, capacity, and efficiency of the ATC system is enhanced by high quality tools for analyzing air traffic patterns. A common challenge in visualizing traffic patterns is the large number of flights that make visualizations complex and challenging to interpret. This paper builds on and extends existing flight pattern visualization techniques by incorporating a hierarchical clustering algorithm to greatly reduce visualization complexity while still preserving the core traffic patterns. The proposed technique analyzes traffic at different altitudes separately and displays flight directionality using color encodings and arrows. The approach and implementation of the visualization tool are illustrated with some sample visualizations generated with one thousand flight trajectories collected over one day.

Introduction

Many air traffic control (ATC) researchers depend on air traffic visualization tools for analyzing and assessing the feasibility of new operational concepts. For example, in order to increase staffing flexibility there is considerable interest in developing generic airspace, or sectors that have enough commonality that controllers could move between them with minimal or no retraining. A key step in identifying such sectors is examining their traffic patterns in order to identify candidate generic sectors (Cho et al., 2011). Differences in their traffic patterns can impact the need for training if controllers move from one sector to another. Not only do sector differences significantly limit the flexibility of controller deployment, it also incurs substantial costs in the form of retraining. As air traffic growth continues, the complexity of managing air sectors is only increasing and the retraining problem will likely be exacerbated.

The generic airspace operational concept highlights two important application areas associated with the development of visualization tools: conducting analysis and enhancing training received by controllers. To minimize retraining requirements, it is important to establish metrics to assess sector similarity so that controllers and analyst alike can be transferred to similar sectors. To increase retraining efficiency, it is important for a controller to quickly understand and compare traffic patterns in a given air sector. The development of improved methods of data visualization holds promise for addressing both of these problem areas.

A common problem with some existing visualization techniques is that once the number of flight trajectories becomes very large the visualization usually becomes overly cluttered and it becomes difficult to discern the underlying traffic patterns. This is partially addressed by existing tools such as MITRE's Dominant Flow Detector (DFD) system that utilizes clustering techniques to reduce visualization complexity (DeArmon J., 2000). The clutter issue faced in visualizing ground highway traffic is similar and is being extensively researched. Reducing visualization complexity using progressive clustering techniques (Rinzivillo et al ., 2008) and making traffic patterns more distinct by comparing route diversities (Liu et al., 2011) are two other research techniques that have been investigated.

In order to address the challenges of making flight visualization analysis more accessible, this paper proposes an explorative data visualization technique that seeks to build on MITRE's DFD work and apply relevant concepts used in ground traffic analysis to the area of flight visualization. The proposed method utilizes standard hierarchical clustering algorithm at different altitudes to reduce the visualization complexity and display directionality to allow controllers, analysts, and researchers to observe traffic patterns in airspaces more clearly. The technique developed in this paper utilizes standard software packages and could be easily incorporated into existing flight visualization solutions. The following sections describe previous work on visualization and visualization algorithms, the approach to developing the new visualization technique by using clustering algorithm, and present examples of its application.

Background

There is a wide range of traffic visualization tools that have been developed to help industry professionals manage the complexity of air traffic management (ATM). Examples of these tools include NASA's Future ATM Concepts Evaluation Tool (FACET) (Bilimoria et al., 2000), fully 3D immersive flight path visualization system (Bourgois et al., 2005), and GIS integrated tools such as ArcTools for designing and evaluating air traffic control sector.

FACET is used for modeling and assessing effectiveness of new ATM strategies and provides real-time simulation and visualization of complex air traffics. ArcTools and immersive 3D flight simulation help researchers identify potential traffic bottlenecks in an air sector to improve efficiency. While useful for managing and evaluating particular air sectors, these software are generally not helpful for speeding up controller retraining because they do not provide high level, intuitive visualization for comparing traffic dynamics across sectors. In order to address this issue, better visualization approaches are required.

In the area of visualization, Aaron Koblin's flight pattern visualization is perhaps one of the most well-known (Koblin, 2009). Koblin's work effectively conveys traffic density at different altitudes and displays the most traffic heavy air highways at different times of the day clearly. However, Koblin's work is shown in a 2D plane and does not display directional information, thereby forcing controllers or analysts to decipher a complex network of intersecting flights at different altitudes and pick out the most traveled airways manually. Detecting the direction of flights as well as differentiating between ascent and descent phases of flight are important for many ATM analysis and training purposes (Menon et al., 2004).

The technique developed in this paper is inspired by the use of clustering techniques both in ground highway traffic visualization and air sector classification. As applied to highway traffic, progressive cluster technique utilizes several distance functions to reduce cluster computation complexity; it first applies coarse distance functions that are usually fast to compute to quickly reduce the number of trajectories and progressively uses more complex distance functions on the remaining trajectories to create the final cluster. Clustering algorithms have also been extensively used in air traffic control research. For example, classifying air sectors into similar groups based on flight traffic and sector topologies using clustering algorithms have been used for sector similarity classification techniques in generic airspace research (Christien et al., 2002).

Building off previous researches, the technique presented in this paper groups flight trajectories into distinct altitude divisions; this allows traffic at each altitude division to be treated separately to reduce cluster computation time. Agglomerative hierarchical cluster algorithm is the clustering algorithm used in this paper. This technique is well known and readily implemented in existing commercial and open-source software. Briefly, the technique treats each distinct flight trajectory as a distinct cluster and iteratively combines clusters by comparing clustering similarity using a predefined distance function. The progression of the algorithm is frequently visualized via a dendrogram, a hierarchical tree-like structure. For an in-depth discussion of the clustering algorithm, please see Dr. Johnson's paper (Johnson S., 1967).

Approach

When developing a visualization tool dealing with data as complex as air traffic flight trajectories, two important factors must be considered in order to reduce the appropriate amount of complexity and provide the users with a more intuitive visualization. The first important factor that must be considered is how much information to present to the user at a time as using raw data directly in visualizations can result in overwhelmingly complex depictions of a traffic situation. A second important factor that must be considered is how (e.g., in what format) to present the information to the user.

The tool presented in this paper addresses these two factors using several different techniques. First, the tool provides the visualization by grouping similar traffic together using hierarchical clustering technique. Secondly, the visualization using this tool can separate traffic visualization at different altitudes. Thirdly, the tool provides directionality of major flight highways. Finally, the visualization is fully three dimensional and data can be interactively examined through zoom and rotation.

Before the trajectory data can be visualized, trajectories data are transformed to a representation using the first two hundred Fourier coefficients using Fast Fourier Transform. Each flight trajectory is then classified into an altitude division by splitting the altitude range of the dataset into increments specified by the user. The hierarchical clustering algorithm is applied to flight trajectories at each altitude division.

Fourier coefficient representation is used for two reasons. First, Fourier vector reduces memory requirements and computation time. Second, using the standard Fourier feature vector establishes a common metric to compare flight trajectories, an important procedure since flight trajectories data differ in duration and sampling rate.

The method proposed in this paper is implemented in MATLAB and utilizes the built-in hierarchical clustering algorithms and Fast Fourier Transform (FFT) functions. Pseudo-code of the tool developed is presented in the appendix of this paper.

The flight trajectories visualization is created using a robust clustering algorithm that does not require predefinition of the number of clusters. The hierarchical clustering approach depends on determining a degree of similarity between aircraft trajectories. For the technique developed in this paper, this similarity is determined using the Euclidean distance function between the 2D Fourier transform coefficients of the X and Y plane of the flight trajectories.

As described in Matlab (2012), the hierarchical clustering algorithm operates by repeatedly finding the pairs of trajectories with the smallest distance, or most similarity, between them. The pair is then treated as a single trajectory, and the process repeats, finding the next pair with the smallest distance (e.g. most similarity). This process repeats, combining the trajectories into clusters, until every trajectory is combined into one single cluster.

The result is a tree-like structure, with branches and leaves representing the increasingly consolidated trajectories. Each node in the tree represents the pairing of two trajectories, where one or both of those trajectories itself be a trajectory. The inconsistency coefficient for a node is a ratio of the distance between the two trajectories that are joined at the node, and the average distance between trajectories in each of the sub nodes. This provides a measure of whether the node represents the joining of two clusters of trajectories that are distinct (high inconsistency coefficient) or whether the distance between the two clusters being joined is very similar to the distance between trajectories within each of the clusters (low inconsistency coefficient).

The tree-like structure is searched for all nodes that have an inconsistency coefficient at or less than a user provided upper limit for inconsistency. In addition, all sub-nodes of that node must also have an inconsistency coefficient at or less than the user provided upper limit for inconsistency (Matlab, 2012). Thus, the user defined inconsistency ratio provides a measure of how aggressive the hierarchical clustering algorithm combines clusters; the higher the inconsistency coefficient, the fewer the resulting clusters. The set of such nodes represent the output of the algorithm and their representative trajectory can be used to develop simplified visualizations.

Instead of using altitude coordinate information in the distance calculation, which is already visually distinct when the data sets are viewed in the XZ plane or the YZ plane, the clustering algorithm is applied to each altitude divisions. This gives controllers and analysts the ability to quickly understand the differences between traffic patterns at different altitudes. For visualization, one sample from each cluster is shown in the resulting graph.

To convey directionality, the tool utilizes color coding and arrows. Ascending trajectory is colored in green while descending trajectory is colored red. For equal altitude flight, the trajectory is colored blue. Ascending and descending traffics are determined by taking the difference between consecutive altitude data; the actual direction of the flight is indicated using arrow on the trajectory only in the flight visualization with clustering analysis applied.

Application

In this section, the four functionalities of the tool are demonstrated using a dataset obtained from Enhanced Traffic Management System (ETMS) containing one thousand flights collected over a period of one day. For this analysis, altitudes are divided into 10 equal divisions and an inconsistency coefficient of one is used. A value of one creates relatively few clusters.

Figure 1 below shows the raw flight trajectory data generated by the tool. The figure shows the original flight trajectories; airports are readily identifiable at the locations showing multi-directional clustering of flight paths and where descent paths shown in red, and ascent path shown in green, intersect. The traffic volume in the flight network is also visible by assessing the density of the flight network. Figure 1 shows an overall picture of where the traffic merges and diverges as well as the general complexity of the airspace very well.

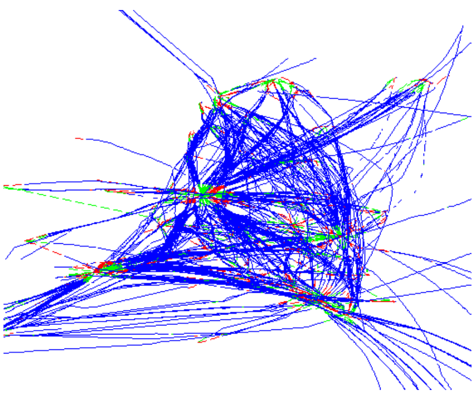

Figure 1. Original traffic data with directionality

The tool produces a second plot showing the results of the hierarchical clustering algorithm. This allows controllers or analysts to get a high level sense of the complexity of air sector and assess whether the clustered result is a good representation of the original traffic.

Applying the hierarchical clustering algorithm greatly reduces the complexity of the visualization while still preserving the underlying traffic patterns. Figure 2 shows the result of the technique: the visualization is simplified significantly while still providing the user with key interaction points, concentration areas, and the general directionality flow of the core traffics.

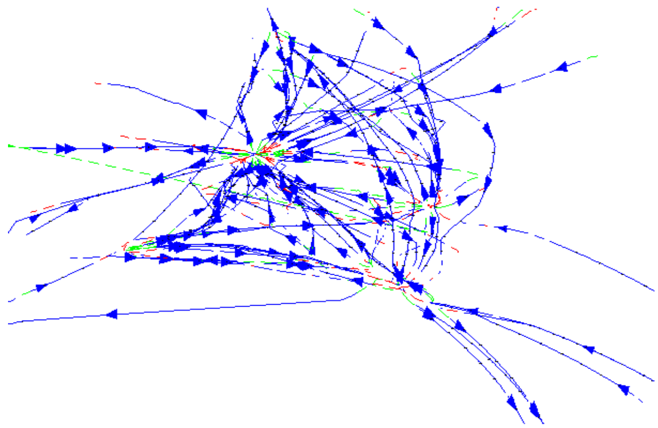

Figure 2: Clustered Flight Trajectory of All Altitude Divisions

The plot has a scrollbar that allows the user to interactively examine the core traffic patterns at different altitudes. This reduces the visualization complexity and allows core traffic patterns at different altitudes to be discerned easily. The traffic transition to different altitude is also made clear through this method. Figure 3 demonstrates this functionality. The lower altitude image on the left of Figure 3 displays more traffic variations as planes depart whereas higher altitude traffic shown on the right of the figure displays mostly straight line traffic patterns (predicted to consist of more en-route trajectories).

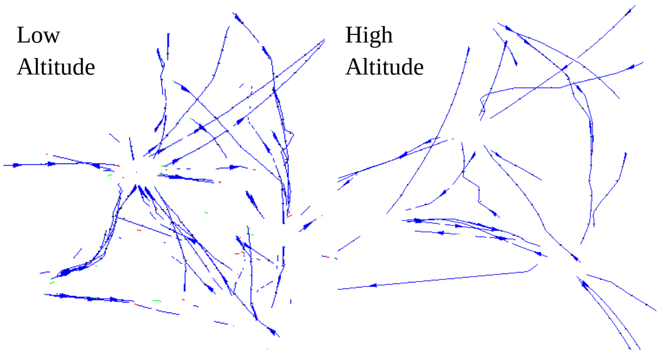

Figure 3. Clustered Flight Trajectory at Different Altitudes

The ability to visualize the core traffic at different altitudes and compare them should be useful for analysts trying to understand the complexity challenges of existing airspace as well as for controllers developing a better understanding of the traffic flows when learning a piece of new airspace.

Discussion

Using the tool proposed in this paper provides its users the advantage of assessing traffic density, directionality, and ascent and descent of flight trajectories at different altitudes at significantly reduced complexity.

However, the clustering result is sensitive to the inconsistency coefficient and requires careful tuning for each dataset. The altitude division is another parameter that must be manually adjusted. Moreover, this visualization tool does not provide precise coordinate information and trajectory speed in characterizing the airspaces. Instead, the developed tool is intended for high level analysis to help the users understand the large scale traffic structures in a given air sector. It is up to the user using the tool to understand the traffic patterns and visually classify the air sectors.

Conclusion

In summary, the tool being developed for this paper provides abundant opportunities for researchers to analyze air traffic patterns for various purposes, due to its ability to produce simplified visualizations of traffic density, directionality, as well as ascending and descending flight trajectories with customizable altitude ranges. This is an improvement over existing flight trajectory visualization as not only is the resulting visualization a succinct portrait of most traveled air highways, it also conveys directionality information and allows easy understanding of the traffics at different altitude. However, the tool developed in this paper is only intended as a high level tool for identifying and learning core traffic patterns and does not provide coordinate and aircraft information. This tool complements current research that use air traffic pattern analyses as it helps its users to quickly understand the similarity and differences between sectors and allows interactive control to produce different visualizations for various purposes. The tool developed utilizes standard methods and software packages and could be easily ported to any existing visualization solutions.

Future Work

To extend this work further and make the resulting visualization more accessible, it would be useful to allow clustering sensitivity to be varied interactively and vary trajectory line thickness based on underlying cluster size.

In terms of new features, a useful improvement to the proposed tool would be animating the trajectory to give air traffic controllers a better sense of time dynamics. Furthermore, incorporation of the weather data and flight times into the visualization would provide an important layer of information for an in-depth analysis of the traffic patterns. It would also be beneficial for traffic trajectory to display aircraft model to let controllers or analysts learn the altitude each aircraft type travels in.

References

Bilimoria, K., Sridhar, B., Chatterji, G., Sheth, K., Grabbe, S., (2000) FACET: Future ATM Concepts Evaluation Tool, 3rd USA/Europe Air Traffic Management R&D Seminar, Napoli, Italy

Boesel, J., Dearmon, J., Bourgois, M., Cooper, M., Duong, V., Hjalmarsson, J., Lange, M., Ynnerman, A., (2005). Interactive and Immersive 3D Visualization for ATC, 6th ATM Seminar, Baltimore, USA

Cho, A., Histon, J. (2011). Identification of Air Traffic Control Sectors with Common Structural Characteristics, 6th International Symposium on Aviation Psychology, Dayton OH.

Christien, R., Benkouar, A., Chaboud, T., Loubieres, P., (2002). Air traffic complexity indicators & ATC sectors classification, The 21st Digital Avionics Systems Conference, Irvine CA

DeArmon, J. (2000). Identifying Dominant Air Traffic Flows in Complex Airspace. MITRE Data Mining Issue (Vol. 4, No. 2, August 2000)

Guttman, J.A., Stein, E.S., Gromelski, S. (1995). The Influence of Generic Airspace on Air Traffic Controller Performance, Report # DOT/FAA/CT-TN95/38, William J. Hughes Technical Center: Atlantic City International Airport.

John, S. (1967). Hierarchical Clustering Schemes, Psychometrika (Vol. 32, No.3, September 1967): pp.241-254

Koblin, A. (2009). Flight Pattern. Retrieved on April 16th, 2012 at http://www.aaronkoblin.com/work/flightpatterns/index.html

Liu, He., Gao, Yuan, Lu., Liu, S., Qu, H., L.M, N. (2011). Visual Analysis of Route Diversity, Visual Analytics Science and Technology (VAST), 2011 IEEE Conference

MATLAB, "Hierarchical Clustering" , Matlab Statistics Toolbox Documentation, Matlab version R2012a. Obtained June 18, 2012 from http://www.mathworks.com/help/toolbox/stats/bq_679x-3.html

Menon, P.K., Sweriduk, G. D., Bilimoria, K. D., (2004). New Approach for Modeling, Analysis, and Control of Air Traffic Flow, Journal of Guidance, Control, and Dynamics (Vol. 27, No.5 October 2004): pp.737-744.

Rinzivillo S., Pedreschi D., Nanni M., Giannotti F., Andrienko N, Andrienko N G., (2008). Visually-driven Analysis of Movement Data by Progressive Clustering, Information Visualization (Vol. 7, No.5, June 2008):pp.225-239

Appendix

Pseudo-code explaining the flow of the tool developed in this paper.

```
function createFlightVisualization()
 % parse cleans the data and transforms flight
 % trajectory into fourier coefficients
 allFlights = parse(inputFile)

 % partitions each flight into an altitude group
 for each flight in allFlights,
  altitudeGroup = getAltitudeGroup(flight.altitude)
  altitudeDivisions[altitudeGroup].append(flight)
 end

 for each altitudeDivision in altitudeDivisions,
  resultCluster = ApplyHierarchicalClustering(altitudeDivision)
  plot(resultCluster)
 end

function ApplyHierarchicalClustering(flightsInAltitudeDivision)
 % High level description of the algorithm
 % 1. Create a new cluster for each flight in altitudeDvision
 % 2. Merge least dissimilar cluster as determined by a distance
   function
 % 3. If remaining clusters greater than one, go back to step 1,
   else stop
 %this is accomplished in MATLAB by using linkage() and
   cluster() functions

 %  second parameter determines the distance function used; in
   this case, weighted average distance is used
 z = linkage(flightInAltitudeDivision, 'weighted')

 % cutoff is the inconsistency coefficient; a value of 1.0 is used
   and limits the size of the returned cluster
 c = cluster(z,'cutoff',1.0)

 return c
```