

Algorithmic Traffic Abstraction and its Application to NextGen Generic Airspace

Girishkumar R. Sabhnani*, Arash Yousefi†

Metron Aviation Inc., Dulles VA, 20166

David P. Kierstead

Daniel H. Wagner, Associates, Inc.

Irina Kostitsyna‡, Joseph S. B. Mitchell§

State University of New York, Stony Brook, NY, 11794

Valentin Polishchuk¶

University of Helsinki, FI-00014

We develop traffic abstraction algorithms that, given a set of 4D Trajectories (4DTs), extract the traffic structure in terms of standard flows and critical points (conflict and merge points). We demonstrate the application of our techniques to enable the NextGen generic airspace concept. We also analyze historical demand data to evaluate the level of abstraction underlying the en route traffic within high-altitude sectors. Finally, we compare the structure of historical traffic to user preferred, wind-optimal, futuristic trajectories.

Nomenclature

G	Gradient.
H	Hessian.
U	Unit Vector.
4DTs	4D Trajectories.
ATC	Air Traffic Controller.
CIT	Controller Information Tool.
DSTs	Decision Support Tools.
HITL	Human in the Loop.
nm	Nautical Miles.
NRS	National Referencing System.
ZDC	Washington ARTCC.
ZFW	Fort Worth ARTCC.
ZOA	Oakland ARTCC.

I. Introduction

ANALYSIS of air traffic structures for a given set of flown or projected trajectories can serve multiple purposes. In particular, such analysis can be used for designing future airspace boundaries (e.g., sectors) and analyzing NextGen generic airspace concepts. As the current system gradually migrates to a satellite-based navigation system, and National Referencing System (NRS) replaces the existing ground-based Navigational Aids, aircraft will fly more direct and user-preferred trajectories, and traffic will be more scattered throughout the NAS. The existing sector boundaries are designed primarily based on the existing airways, and any airspace redesign requires analysis of futuristic traffic structures.

The concept of generic airspace has been proposed within the NextGen ConOps.^{1,2} By definition, such

*Senior Analyst, Advanced Research & Engineering, 45300 Catalina Ct.

†Principal Analyst, Advanced Research & Engineering, 45300 Catalina Ct., AIAA Senior Member

‡Ph.D. candidate, Dept. Computer Science

§Professor, Dept. Applied Mathematics and Statistics

¶Postdoctoral Researcher, Helsinki Institute for Information Technology

airspace can be managed by any controller from any facility with supporting procedures, structure, and automation levels. This will enable interchangeability of resources among different facilities in the NAS, and can potentially enhance the utilization of human controller resources. To enable such concepts, traffic characteristics, airspace structures, referencing systems, and control procedures must be generalized so that controllers can quickly understand and comprehend the traffic situation. Decision Support Tools (DSTs) and Enhanced Training Tools need to be developed to assist in controller training and expediting the time required for controllers to become familiar with traffic structure and local characteristics of sectors. Mogford et al.³ developed the Controller Information Tool (CIT) to study these issues and evaluate the benefit of DSTs in facilitating the goals of generic airspace. It was concluded that presenting the sector and traffic information is helpful for controllers before they start controlling on the D-side.

Traffic abstraction and conflicting points will help controllers to quickly comprehend the traffic structure. Tools similar to CIT require traffic abstraction and structure including the coordinates of standard flows and conflict points as inputs. In highly dynamic heterogeneous future airspace, automated techniques are needed to provide these inputs to generic airspace DSTs.

In this paper we present two methods of producing traffic abstractions and present examples of their application in generic airspace. The organization of paper is as follows: In Section II, we discuss the important elements of abstraction that must be considered in model development. Section III gives the details of various traffic abstraction methods. Section IV presents applications of the methods to analyze the NextGen generic airspace concept and to serve as input for the Air Traffic Controller (ATC) decision support and training tools. In Section V, we compare the structure of today’s demand with the futuristic user-preferred, wind-optimal demand. The conclusion and future work is discussed in Section VI.

II. Elements of Traffic Abstraction

Based on over-the-shoulder monitoring of controllers and analysis of historical traffic, Hansman et al.⁴ identified the following major factors in structure-based abstraction:

- **Standard Flows:** Standard flows are major flows of aircraft within a piece of airspace, such as a sector; the majority of traffic flies along these standard flows. In today’s ground-based navigation system, standard flows are mostly placed along the jet routes.
- **Critical Points:** Critical points are the intersections of standard flows. At these points there is a higher probability for loss of separation. Controllers perform more tasks when aircraft fly closer to critical points. Merge points and conflict points (with crossing traffic) are examples of critical points.
- **Groupings:** A Grouping is a sequence of aircraft flying the same or similar heading and speed. To control a Grouping, controllers only need to assign a certain heading and speed and require the aircraft to maintain their speed and heading until approaching a critical point, where additional instruction and conflict resolution actions may be needed. Therefore, controlling n aircraft within a Grouping requires less effort than controlling n individual aircraft that are scattered across a sector.

Figure 1 illustrates an example of these abstractions. A non-standard aircraft is one, which does not follow the standard flows. Our algorithms are aimed to produce this abstraction quickly and automatically in response to dynamic demand and resectorization.

Identifying standard flows can be thought of as a trajectory clustering problem. There is an excellent web survey⁵ covering different kinds of trajectory clustering problems. The most relevant kind of trajectory clustering of interest here is classified as *repetition* in Andrienko et al.⁶ Gudmundsson et al.’s book chapter⁷ also gives a good overview of the topic, along with some key applications beyond air traffic management.

III. Traffic Abstraction Methods

We describe two methods to identify standard flows. Dykes et al.⁸ suggest that the traffic density heat map approach used by both

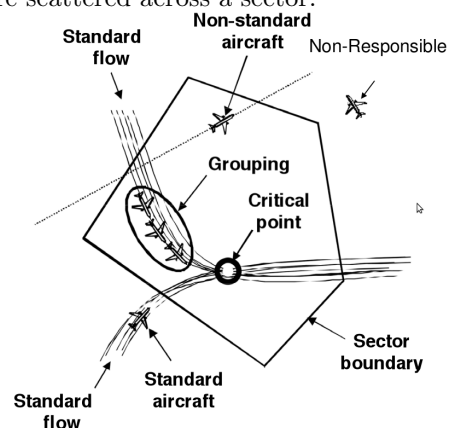


Figure 1. Four Types of Structure-Based Abstractions (source Hansman et al.⁴)

methods dates back to the work of Hagerstrand in 1970.

TRAFFIC DENSITY MAP We first describe the construction of a traffic density heat map in the 2D (lat,lon) plane. We discretize the space of the bounding rectangle of (the 2D projection of) the trajectory set T with a uniform grid G . The grid spacing in G is defined by a parameter $incr$ (in Nautical Miles (nm)). For each grid point g , the weight $w(g)$ is defined as the number of tracks that intersect a disc of radius ϵ (in nm and $incr < \epsilon$), centered at g . See Fig. 2(b) for an example of a traffic density map. The grid points are color-coded, with traffic density increasing from green to red. It should be easy to see that this construction extends to 3D (with altitude) and 4D (with altitude and time).

1. Greedy Trajectory Clustering

This method addresses the following problem: Given a set T of n aircraft trajectories (polygonal paths in 2D), does there exist a set of standard flows $D \subset T$, $|D| \leq k$, such that at least $c\%$ (measured as the sum of length) of the trajectories in T lie “close” (within ϵ) to at least one of the standard flows. A trajectory t is “close” to a standard flow d if a certain fraction f of its length $l(t)$ lies within the ϵ -fattening of d (see Fig. 2(a)). One may want to fix c and minimize k , or fix k and maximize c .

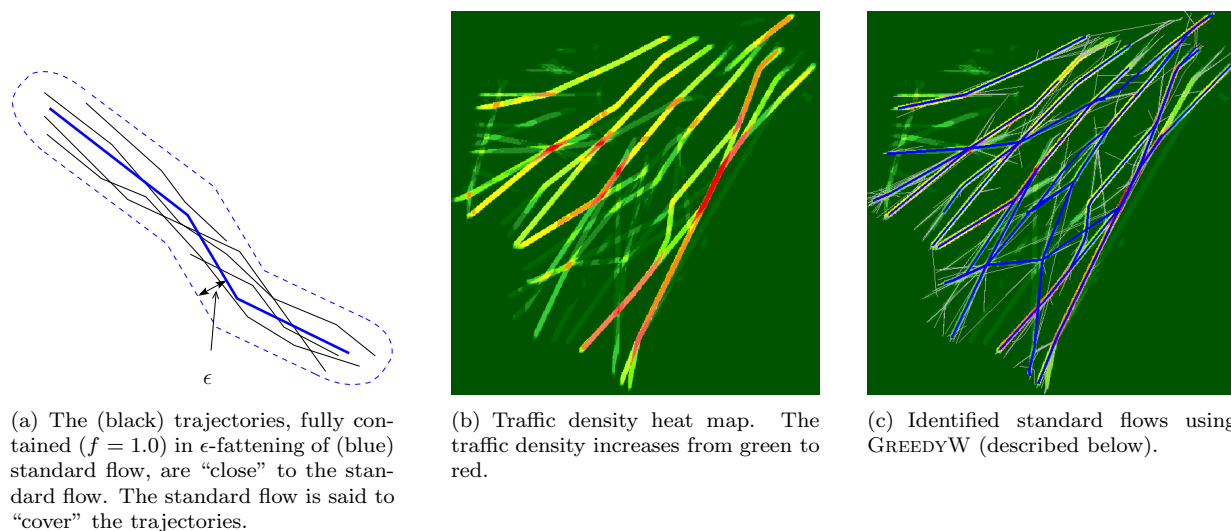


Figure 2. Greedy Clustering results for ZDC.

To determine these standard flows, we greedily select trajectories that “cover” many tracks from a given set of trajectories, and designate them as standard flows. Tracks that remain in the vicinity of a standard flow are claimed as covered (see Fig. 2(b)). The main assumption of this method is that there is at least one trajectory in the input set that exactly follows a standard flow, and can act as a representative for that flow.

Using the traffic density heat map, for each track t , we define the weight $w(t)$ as the sum of weights of all grid points that are within distance ϵ (in the ϵ -band) of t . Note that this ϵ is the same as the disc radius used in the construction of the traffic density map. The density of t is defined as $d(t) = w(t)/n(t)$, where $n(t)$ is the number of grid points in the ϵ -band of t . Intuitively, the density is weight per unit length of the track and is a good quantifier for the dominance of t . A track t' with a high $d(t')$ goes through (or lies completely in) the region of high traffic density. Thus, t' is a good candidate for standard flow.

Inspired by the greedy approximation algorithm for set cover, we devise a greedy algorithm for our problem; see Algorithm 1. Our basic heuristic, GREEDYT, selects the “best” track at each iteration as follows:

- GREEDYT “best” track is the one which has maximum coverage: $d \leftarrow \operatorname{argmax}_{\{t' \in T \text{ and } t' \notin C\}} \operatorname{cov}(t')$, where $\operatorname{cov}(t') = \sum_{\{t \in T \text{ and } t \notin C \text{ and } t \text{ “close” to } t'\}} l(t)$.

Finding the best track takes considerable computation time. To reduce it, we used two modifications of GREEDYT, which we called GREEDYD and GREEDYW:

- GREEDYD selects a track with the maximum density. Since a track with the maximum density may have a very low weight $w(t)$, it is not a good choice because it may cover very few tracks. We modify GREEDYD by introducing a constraint that the weight of the selected track is at least a fraction w_c of the maximum track weight. Thus, $d \leftarrow \operatorname{argmax}_{\{t \in T \text{ and } t \notin C \text{ and } w(t) > w_c \cdot w(t')\}} d(t)$, where $t' \leftarrow \operatorname{argmax}_{\{t \in T \text{ and } t \notin C\}} w(t)$.
- GREEDYW selects a track with maximum weight under the constraint that $d(t)$ is at least a fraction d_c of the maximum track density. Thus, $d \leftarrow \operatorname{argmax}_{\{t \in T \text{ and } t \notin C \text{ and } d(t) > d_c \cdot d(t')\}} w(t)$, where $t' \leftarrow \operatorname{argmax}_{\{t \in T \text{ and } t \notin C\}} d(t)$.

Algorithm 1 Greedy Trajectory Clustering

Input: Set T of tracks, Coverage threshold c %.

Output: Set D of standard flows.

$D \leftarrow \phi$

$C \leftarrow \phi$ /* Covered tracks /*

Coverage $\leftarrow 0$

while Coverage $< c$ % **do**

$d \leftarrow$ “best” track

$D \leftarrow D \cup d$

for all Tracks $t \in T$ **do**

if $t \notin C$ and t “close” to d **then**

$C \leftarrow t$

end if

end for

Coverage $\leftarrow \sum_{t \in C} l(t) / \sum_{t \in T} l(t)$

end while

Even though GREEDYT does not use the density weights from the heat map, we still use the discrete grid G to simplify the coverage computation (by intersecting the set of grid points in the vicinity of track and the standard flow). In addition, the traffic density map is updated in each iteration of Algorithm 1, for GREEDYD and GREEDYW, corresponding to the tracks in $T \setminus C$.

An instance of GREEDYW output for 24hr historical demand for the Washington ARTCC (ZDC) is shown in Fig. 2(c). The parameters used for this experiment are as follows: $incr = 1$ nm, $\epsilon = 2.5$ nm, $f = 70\%$, $c = 60\%$ and $d_c = 0.6$. Refer to Sabhnani⁹ for more information about how these parameters affect the results of the method.

2. Ridge Top Detection

This method primarily works in 2D. It starts by building the 2D traffic density heat map, and considers the traffic density heat map’s values as the third dimension to create the ridge domain as shown in Fig. 3(a). Let a twice-differentiable function $f : \mathbb{R}^2 \rightarrow \mathbb{R}$ represent the density heat map after passing through a Gaussian filter. The Gradient (\mathbf{G}) of f is:

$$\mathbf{G} = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{pmatrix} \quad (1)$$

\mathbf{G} points in the direction of maximum increase in f . The derivative of f in direction \mathbf{U} (a Unit Vector (\mathbf{U})) is $\mathbf{G}^T \mathbf{U}$.

The Hessian (\mathbf{H}), of f is:

$$\mathbf{H} = \nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix} \quad (2)$$

The second derivative of f in direction \mathbf{U} is $\mathbf{U}^T \mathbf{H} \mathbf{U}$. \mathbf{H} is symmetric and therefore it has two real eigenvalues e_1 and e_2 ($e_1 \leq e_2$) with corresponding eigenvectors \mathbf{V}_1 and \mathbf{V}_2 that are orthogonal.

At a ridge, $e_1 < 0$ and \mathbf{V}_1 is orthogonal to the ridge. Also $\mathbf{G}^T \mathbf{V}_1 = 0$ and \mathbf{V}_2 points along the ridge. Near a ridge, \mathbf{G} has a component in the direction of the ridge and \mathbf{V}_1 points roughly towards the ridge or away from it. We identify the location of ridges by checking for these conditions in f .

The output of this process is illustrated in Fig. 3(b). Once the location of the ridges is identified, we identify the standard flows based on the average heat value along the ridges. We filter out the ridges with low average heat values; see Fig. 3(c). Both Figs. 2(c) and 3(c) show the identified flows for ZDC using two different methods, and the similarity of the results is apparent.

The main advantage of the ridge method is that it does not rely on individual tracks; if multiple tracks together define a standard flow, but none of them individually follows the standard flow exactly, this method is likely to identify that flow.

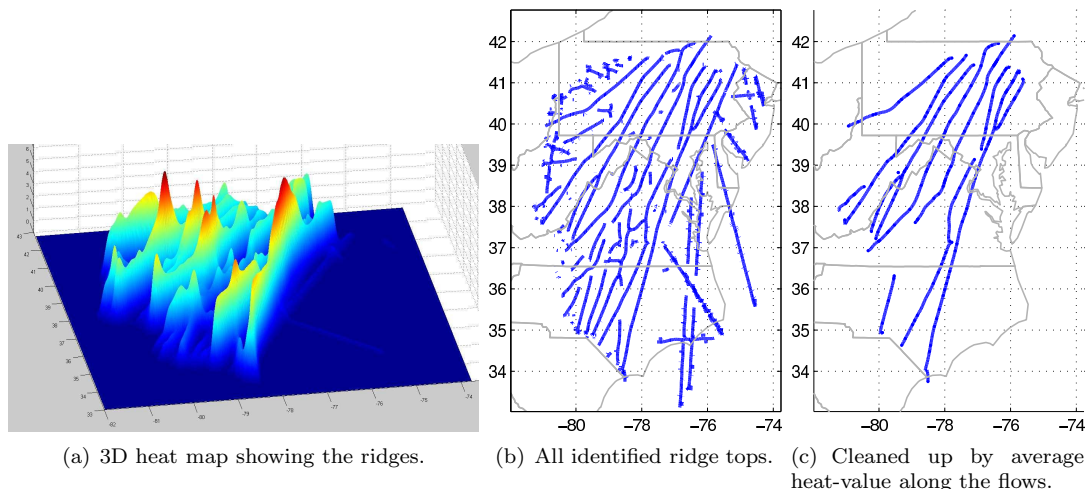


Figure 3. Ridge top detection results for ZDC.

3. Identifying Critical Points

A point where standard flows intersect can be one of the following: a) a merge point where two or more flows merge, b) a conflict point where two flows cross each other, and c) a diverge point where flows diverge. We consider all flow intersections as critical points in the traffic abstraction, even though a diverge point may be less critical for quantifying the ATC workload.

We follow a top-down method to identify the critical points: First identify the standard flows, and then identify the intersection points of two or more standard flows. This is in contrast to the bottom up method,¹⁰ in which intersections of individual tracks are clustered to identify critical points. For generic airspace purpose, we believe the top-down method is more relevant. The critical point, if present, should be translated back to intersection of some standard flows. This is important because a) it is unlikely that few aircraft approach a point in space-time from different directions without themselves belonging to some standard flow; b) even if that is the case, it is not really a part of an underlying traffic structure to which the ATC would want to revert in case of congestion or high demand volume. Note that the top-down method also considerably reduces the search space for finding the critical points, since the points of interest are only those at the intersections of identified standard flows.

For a critical point, it is important that the flows intersect in 4D, i.e. aircraft flying along the intersecting standard flows should interleave in space-time for the point to be considered critical. For experiments in this paper, since we use the 2D implementation of above methods; it is important to post-analyze the flow intersection points to check for interleave in altitude-time.

Consider an intersection point of two standard flows in 2D. Figure 4(a) shows the altitude-time profile of all aircraft that fly in the vicinity (2.5 nm) of this intersection point (in the 2D projection). The red points represent aircraft that fly along the first standard flow, the blue along the second flow, and the gray points represent aircraft that do not fly along either of the flows. We identify conflict points as pair of points (of different colors) that lie within 2,000 feet and 15 minutes of each other (shown as black points in Fig. 4(b)). Now, we can just count the number of these conflicts and weight the corresponding critical

points. An alternate way to weigh the critical point would be to cluster these conflict points and identify a critical range for altitude-time (as shown by rectangles in Fig. 4(b)). The size of these rectangles (sum or max) can be considered as weights of critical points. We are currently investigating standard k-means clustering methods to identify these critical intervals.

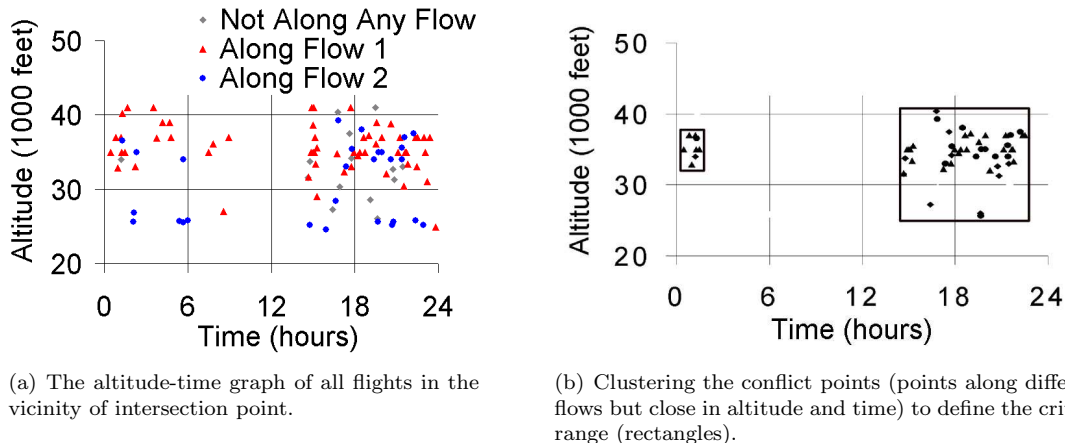


Figure 4. Analyzing altitude time profile of flights in the vicinity of a flow intersection point in 2D.

IV. Application to Generic Airspace

NextGen ConOps introduces changes in airspace usage and design by dynamically resectorizing the airspace in response to changes in traffic that may arise from convective weather activities or changes in demand profile. As a result, controllers may constantly need to adjust to the new airspace and may be assigned to a different position in the NAS. The concept of generic airspace¹¹ is proposed by NextGen to facilitate this shifting of positions.

Even though certain local and site-specific procedural and referencing knowledge are not transferable, it is possible to abstract some features that make a piece of airspace a good candidate for generic airspace. The presence of structure enables the controllers to develop abstract mental models of traffic, and this enhances situational awareness. A high-level of situational awareness results in lower airspace complexity and controller workload, which in turn reduces the required training and transition time for the controllers. Hence, a piece of airspace is a good candidate for generic airspace if the majority of its traffic is flying along a highly structured pattern. Here we use our flow abstraction methodologies to identify sectors with highly structured traffic patterns.

A. Analysis of Traffic Abstraction for Generic Airspace

EXPERIMENTAL SETUP

36 high-altitude sectors (with at least 10,000 feet overlapping with altitude range 24,000 – 99,900 feet) in the ZDC, Oakland ARTCC (ZOA), and Fort Worth ARTCC (ZFW) are analyzed. Historical flight data from May 1, 2008 (a day with no severe weather disruption) is used.

The day is divided into 12 two-hour time intervals, and the demand in each individual sector-time-interval pair is analyzed to be a candidate for generic airspace. For the greedy trajectory clustering method, standard flows are restricted to have at least 6 tracks in its vicinity. The choice of 6 tracks following the path of any standard flow translates to having 1 flight (on average) every 20 minutes. In addition, all flows with length less than 10 nautical miles were eliminated as outliers. The same experiment was repeated with 6 four-hour time intervals, and the threshold for number of tracks in vicinity of a standard flow was set to 12. We consider all intersection points of the standard flows (projected on to 2D) to be critical points.

RESULTS

For each sector, we discard the time intervals that have less than 15% of demand (total tz-hit counts)

compared to the peak demand time interval for the same sector. This is done because controllers, regardless of unfamiliarity of the new airspace, should be able to manage such sparse traffic demand.

Figure 5 contains three types of histograms that bucket the number of sector-time-intervals: in the left column with respect to the number of standard flows; in the middle column with respect to the number of critical points; and in the right column with respect to the ratio of demand along the standard flows. The ratio of demand is defined as the sum of track lengths along the flows over the total length of all tracks.

It is interesting to see that for most sector-time instances (for both two-hour and four-hour intervals) there are 1 – 4 standard flows, 0 – 5 critical points, and for almost half of the instances, 50% or more traffic followed the standard flows. This strongly suggests that there is enough structure in the demand that can be extracted and presented to the ATC. This is hardly surprising as the historical (and present day) demand is already structured along the jet routes. In section V we compare the underlying structure in current demand with projected futuristic demand.

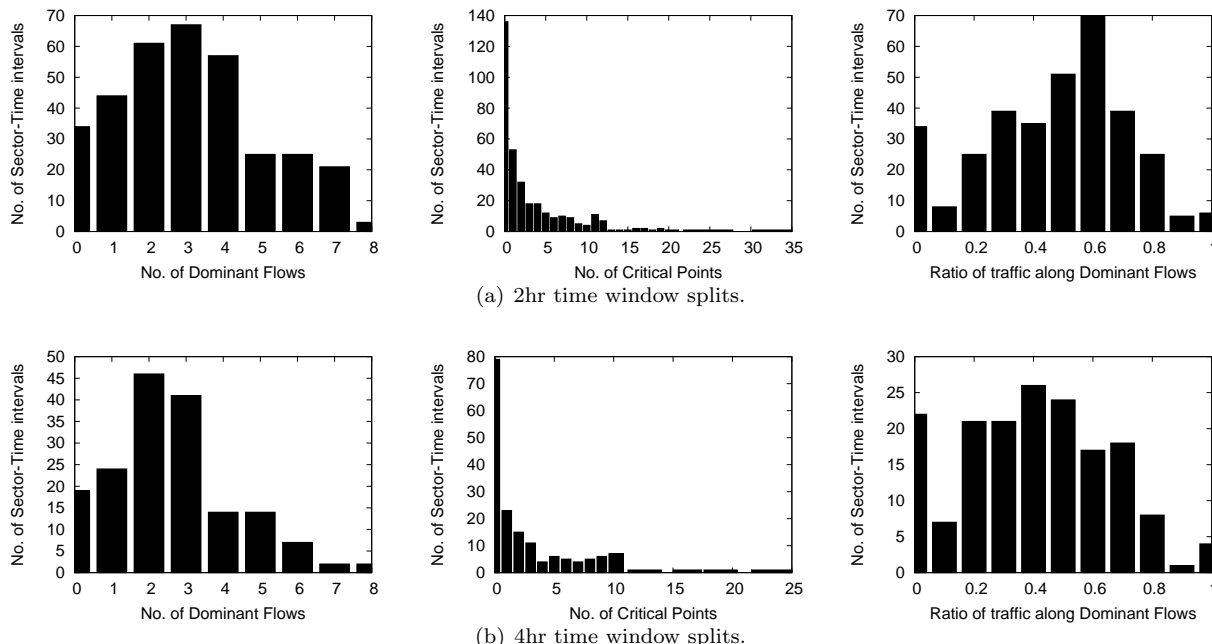


Figure 5. Histograms of traffic abstraction across all sectors.

Also, for the two-hour time-interval experiment, 31.15% of the instances (apart from the 95 sparse traffic instances) have 60% or more traffic along the identified standard flows, including 20.77% that have 6 or less critical points. These numbers indicate that significant chunks of airspace (in 4D) has very structured traffic, and can be considered as suitable for being regarded as generic airspace.

Figure 6 shows the variation in these abstraction elements for different two-hour time intervals for two specific sectors, ZDC19 and ZFW49, over the entire day. For both sectors, we also include a plot of the ratio of demand (total tz-hit counts) in each individual time interval to the peak demand time interval for that sector. This explains why certain time intervals had no standard flows and critical points. Both the number of standard flows and the number of critical points is less in ZDC19 compared to ZFW49. In addition, the ratio of traffic along the standard flows is much higher for ZDC19. These results indicate that traffic in ZDC19 is more structured compared to ZFW49. This information can be used in identifying candidates for generic airspace in NextGen.

B. Input for ATC Decision Support and Training Tools

Traffic abstractions can assist the ATC decision support and training tools by presenting a high-level structure in the projected demand. Auto-traffic abstraction methods are important in highly dynamic demand environments, as in case of weather events, since the demand structure changes significantly from the current operations. Such methods can quickly set up the visualization tools that assist ATC to develop the mental

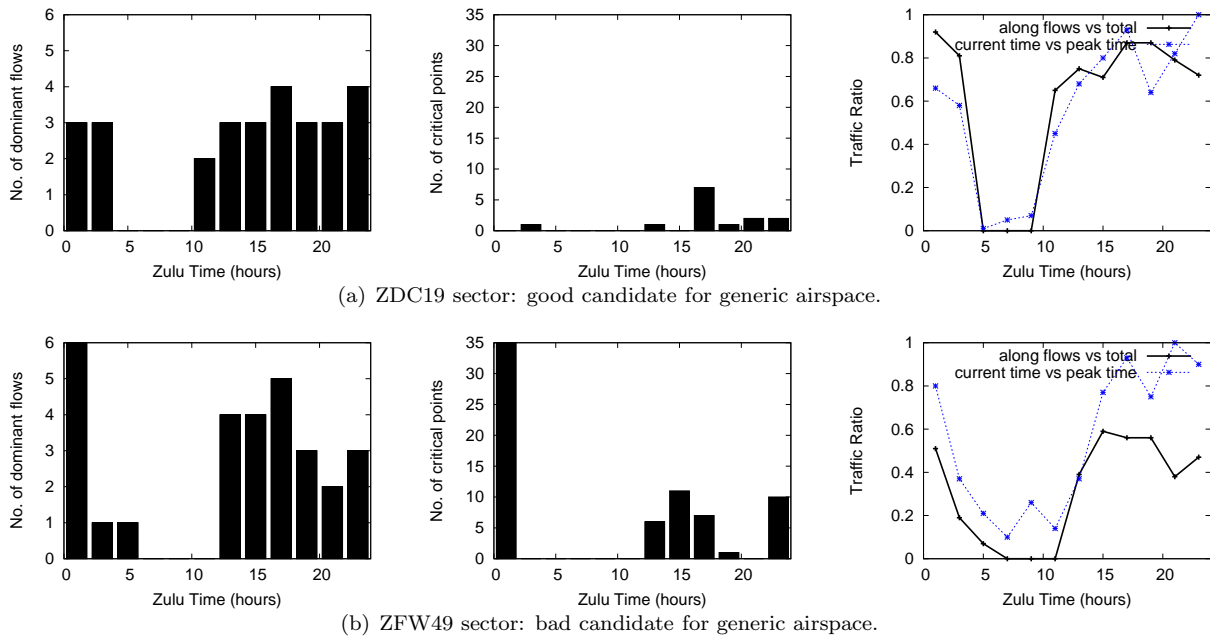


Figure 6. Traffic abstraction for two sectors for different 2hr time windows for the entire day.

model to adapt to these changes.

Mogford et al.³ developed the CIT to facilitate controller training, and to enable rapid realization of important attributes of sectors before controllers actually start controlling the traffic in an unfamiliar sector. CIT provides easily accessible sector information, and enables the controllers to displayed critical information while hiding less important items. The usefulness and usability of the CIT was evaluated by Human in the Loop (HITL) experiments.³

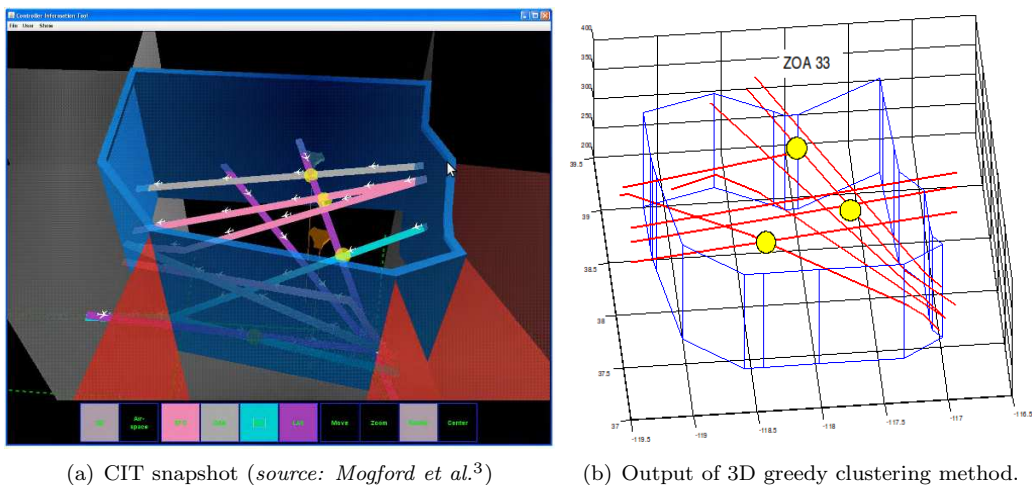


Figure 7. Standard flows in ZOA33.

Figure 7(a) shows the CIT's 3D view (as used in HITL) with major flows and critical points that were identified by interviewing the controllers for a specific sector. Figure 7(b) illustrates the same sector with algorithmically generated traffic abstraction. Standard flows and critical points are marked with ribbons and dots so that controllers can easily grasp the overall picture of traffic situation. For HITL simulations, the location of critical points and standard flows was solicited from the controllers during interviews and manually plotted on the CIT screen. We algorithmically extracted the standard flows and critical points

using historical trajectory data for a clear weather day. It is apparent from the figure that our automated algorithm was able to generate an image very similar to that generated through controller interview and manual plotting. (An exact correspondence was not possible because we did not know the precise time period for which the CIT image was generated.) Such an automation may facilitate rapid preparation of input data for future tools like CIT.

V. Historical vs Futuristic Demand Structure

Next, we compare the historical demand structure, in which the trajectories more or less fly along the underlying jet routes, with the predicted futuristic demand.¹² The predicted futuristic demand represents 2.0X (approximately twice the number of flights) wind-optimal, user-preferred trajectories. We restrict the comparison to only the 18 high-altitude ZFW sectors with the demand split into 12 two-hour intervals. Keeping the other parameters of methods the same (as in Section IVA), we just double the threshold (12 flights) for the standard flow identification methods in order to account for 2.0X traffic.

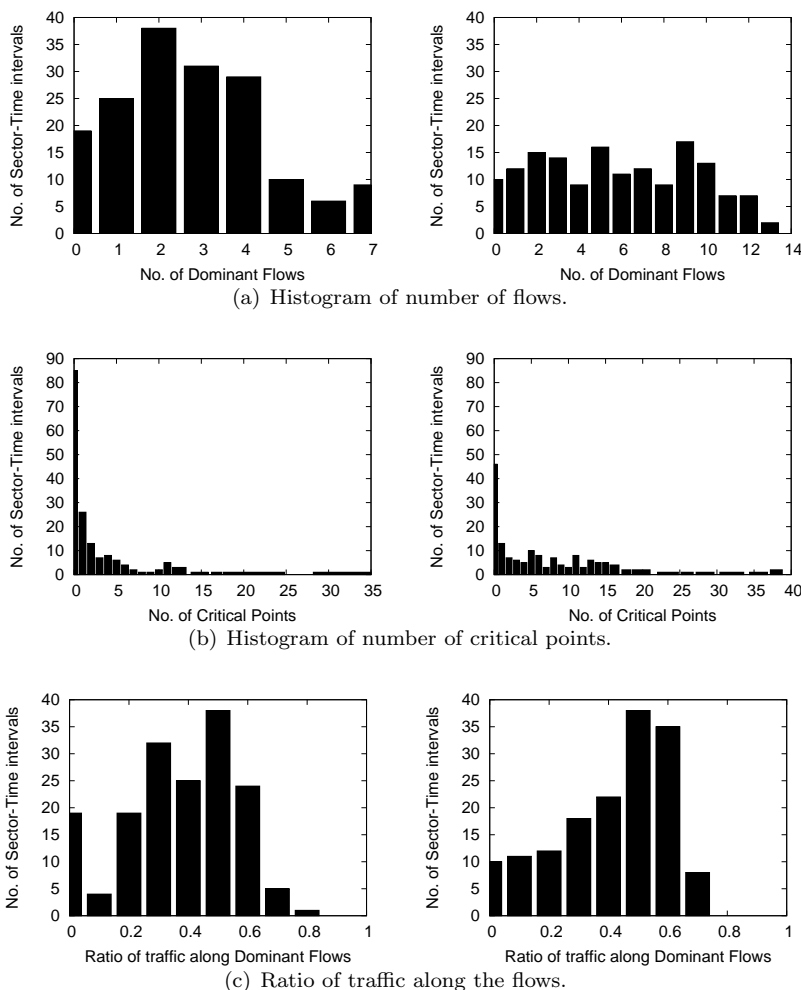


Figure 8. Comparison of structure in today's traffic (left) with futuristic predicted traffic (right).

Figure 8 shows different metrics for the comparison. Out of $18 \times 12 = 216$ sector-interval pairs, 167 such pairs for the historical track data and 154 pairs for the futuristic data were considered interesting as they had more than 15% of the peak traffic within each sector. Furthermore, the number of standard flows increased, whereas the number of critical points decreased significantly for the futuristic data, and the percent traffic along the flows remained similar. These observations are consistent with the intuition that by removing the constraint of ground-based navigation systems, aircraft will be more scattered throughout the NAS when

they fly their preferred, wind-optimal trajectories. Note that a significant proportion of demand complies with the dominant flows in both historical and futuristic demand.

VI. Conclusion and Future Work

To analyze NextGen concepts like generic airspace and dynamic sectorization, fast algorithmic traffic abstraction methods are needed. We developed two standard-flow and critical-point identification methods. We examined the application of these methods to identify sectors with highly structured traffic. The abstraction methods are applicable for identifying traffic structure in any piece of airspace, thus they will also be useful as the sector boundaries change dynamically. Furthermore, we applied the abstraction methods to both historical and futuristic track data, and compared the underlying structure of both traffic data.

In future, we would like to use these traffic abstraction results and modify sectorization methods to produce more generic sector designs that allow ease of ATC transition to the new airspace.

Acknowledgment

This research was sponsored by NextGen Airspace Program at NASA Ames Research Center under order number NNA08BA50D. We wish to thank Shannon Zelinski at NASA Ames Research Center for her support and contribution to this research.

References

- ¹“Concept of Operations for the Next Generation Air Transportation System, Ver, 2.0,” Tech. rep., Joint Planning and Development Office, Washington, DC, June 2007.
- ²Kopardekar, P., Bilimoria, K., and Sridhar, B., “Initial Concepts for Dynamic Airspace Configuration,” *AIAA 7th Aviation Technology, Integration and Operations Conference (ATIO)*, Belfast, Northern Ireland, September 2007.
- ³Mogford, R., “Generic Operations Research,” Presented at the Second Workshop on Dynamic Airspace Configuration (DAC), February 2009.
- ⁴Hansman, R. J., Li, L., Sandes, A., and Histon, J., “Complexity Factors Affecting Generic Airspace,” Presented at the Second Workshop on Dynamic Airspace Configuration (DAC), February 2009.
- ⁵<http://movementpatterns.pbworks.com/Patterns-of-Movement>,
- ⁶Andrienko, N. and Andrienko, G., “Designing visual analytics methods for massive collections of movement data,” *Cartographica v.42 (2)*, 2007, pp. 117–138.
- ⁷Gudmundsson, J., Laube, P., and Wolle, T., “Movement Patterns in Spatio-temporal Data,” *Encyclopedia of GIS*, 2008, pp. 726–732.
- ⁸Dykes, J. A. and Mountain, D. M., “Seeking structure in records of spatio-temporal behaviour: visualization issues, efforts and applications,” *Computational Statistics & Data Analysis*, Vol. 43, No. 4, 2003, pp. 581–603.
- ⁹Sabhnani, G. R., *Geometric Algorithms for Dynamic Airspace Sectorization*, Ph.D. thesis, Stony Brook University, NY, May 2009.
- ¹⁰Zelinski, S., “Defining Critical Points for Dynamic Airspace Configuration,” *26th International Congress of the Aeronautical Sciences*, Anchorage, AL, September 2008.
- ¹¹Guttman, J. A., Stein, E. S., and Gromeleski, S., “The influence of generic airspace on air traffic controller performance.” Atlantic City International Airport: Federal Aviation Administration William J. Hughes Technical Center (DOT/FAA/CT-TN95/38), 1995.
- ¹²Yousefi, A., Stefandis, K., Zadeh, A., and Lowther, M., “Airspace Class Scheduler: Analysis of Aircraft Equipage and Traffic Characteristics for Determining Optimal Allocation of Airspace Classes,” Metron Aviation Tech. Report, Order #NNA08BA50D, May 2009.