

# Automated Ionospheric Front Velocity Estimation Algorithm for Ground-Based Augmentation Systems

Eugene Bang and Jiyun Lee  
*Korea Advanced Institute of Science and Technology*

Jiwon Seo, Sam Pullen, and Sigrid Close  
*Stanford University*

## ABSTRACT

Ionospheric anomalies, which may occur during severe ionospheric storms, could pose integrity threats to Ground-based Augmentation System (GBAS) users [1], [2], [3]. The ionospheric threat for a Local Area Augmentation System (LAAS), a GBAS developed by the U.S. Federal Aviation Administration (FAA), was modeled as a spatially linear, semi-infinite “front” (like a weather front) with constant propagation speed. The model is parameterized by the slope (or gradient) of the front, its width, and its ground speed. Along with the magnitude of ionospheric gradients, the speed of the fronts in which these gradients are embedded is an important parameter for GBAS integrity analysis.

This paper proposes an automated velocity estimation algorithm for anomalous ionospheric fronts. To examine the performance of this automated algorithm, we obtained estimation results for the points of the current Conterminous U.S (CONUS) threat space and compared these estimates to those manually computed previously. This new algorithm proposed in this paper is shown to be robust to faulty measurement and modeling errors. In addition, this algorithm is used to populate the current threat space with newly-generated threat points obtained from the Long-Term Ionospheric Anomaly Monitoring tool [4]. A larger number of velocity estimates helps to better understand the motion of ionospheric fronts under geomagnetic storm conditions.

## INTRODUCTION

The ionosphere is a region of the atmosphere extending from about 50km to about 1000km above the earth’s surface. In this region, free electrons and ions generated by the sun’s radiation give rise to phase advance and

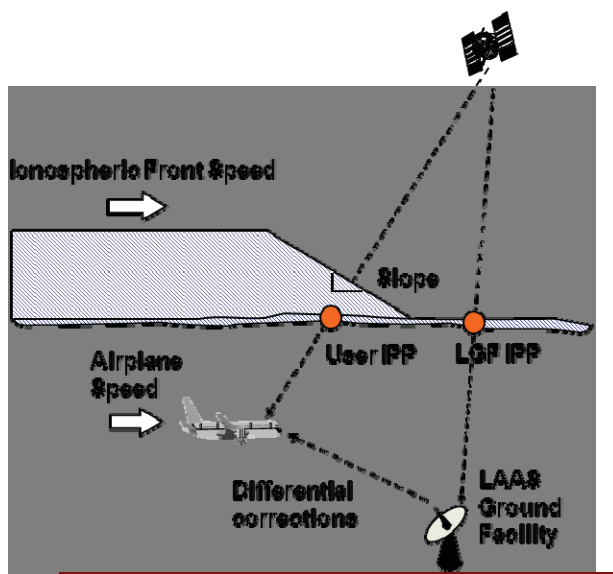
group delay of GPS signals [5]. The error introduced by the ionosphere to GPS pseudorange measurement can be up to tens of meters.

Local-area Differential GPS (DGPS) can remove almost all of ionosphere delay errors in normal condition because ionospheric spatial and temporal decorrelation errors are negligible between a reference station and users. Local Area Augmentation Systems (LAAS), a representative system of local area DGPS, supports aircraft precision approach and landing by providing differential corrections and integrity information to aviation users. However, unusual solar activity such as Coronal Mass Ejections (CME) or solar flares can cause anomalous ionospheric behaviors, as was observed during the November 2003 ionosphere storms. These ionospheric anomalies, which may exhibit large spatial or temporal decorrelation in a short baseline, could pose potential integrity threats to GBAS users.

These ionospheric anomalies caused spatial gradient in slant (i.e., along the actual path between satellite and receiver) ionospheric delays of as large as 413 mm/km over baseline of 40-100 km. The residual range error suffered by a LAAS user at the CAT I decision height (DH) could be as large as 8 meters, if this large gradient goes undetected by the LAAS ground facility (LGF) [3]. A configuration of the LGF, aircraft impacted by an ionospheric front, and a satellite is sketched in Figure 1. Thus, the development of the CONUS threat model for LAAS were required to simulate worst-case ionospheric errors for LAAS users, i.e., to determine their impact on LAAS users, and to develop additional mitigation strategies [6], [7].

However, since the CONUS threat model was based on only the last decade of observed anomalies, there is an

uncertainty as to whether or not threat-model bounds derived from past events are sufficient to bound future events. Considering the peak of the next solar cycle is expected in around 2013 and system integrity relies on an empirically-driven threat model, it is necessary to validate and update this threat model over the life cycle of the system by monitoring ionospheric behavior continuously [4]. In addition, ionospheric threat models must be derived for all regions where GBAS will become operational. For these reasons, a vast amount of ionospheric data should be analyzed continuously going forward. However, in an earlier work [1], a large portion of analysis on ionospheric storm data was done manually, including the estimation of ionospheric wave-front propagation velocities. Thus, the vast amount of data to be processed motivates the development of an automated ionospheric front velocity estimation algorithm.



**Figure 1: Illustration of an LAAS user impacted by an ionospheric wave front.**

## 2.0 DATA

High precision ionospheric measurements are required to estimate ionospheric front velocity more accurately. In this paper, two types of data were used for ionospheric front velocity estimation.

The current ionospheric threat model for LAAS was constructed using high-quality estimates of ionospheric delays produced by NASA's Jet Propulsion Laboratory (JPL). GPS data were collected from the network of Continuously Operating Reference Stations (CORS) and the Wide Area Augmentation Systems (WAAS) and post-processed by JPL in sophisticated post-processing algorithms [8]. The dates on which data were analyzed to define the current ionospheric threat model are shown in Table 1. These data were used to validate the performance

of the automated ionospheric front velocity estimation algorithm in Section 5.3.

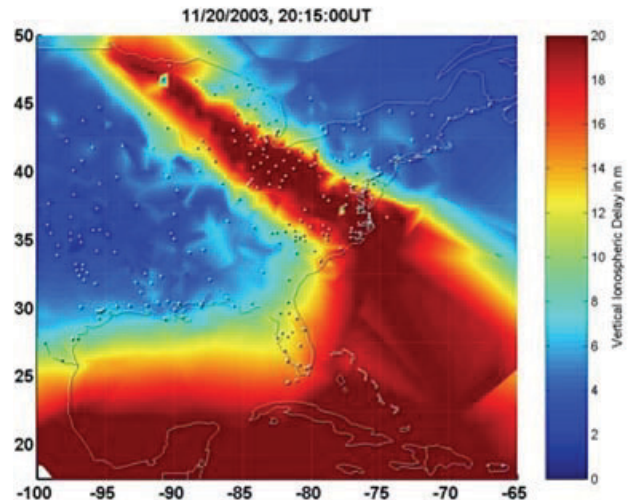
We also used the ionospheric delay data obtained from the Long term Ionospheric Anomaly Monitoring tool (LTIAM) [4] as input to generate additional threat points using this automated ionospheric front velocity estimation algorithm.

**Table 1: Ionospheric storm dates and geomagnetic conditions.**

Day (UT mm/dd/yy)	Kp	Dst	Geomagnetic Storm Class
04/06/00	8.3	-287	Severe
04/07/00	8.7	-288	Extreme
07/15/00	9.0	-289	Extreme
07/16/00	7.7	-301	Strong
09/07/02	7.3	-163	Strong
10/29/03	9.0	-345	Extreme
10/30/03	9.0	-401	Extreme
10/31/03	8.3	-320	Severe
11/20/03	8.7	-472	Extreme
07/17/04	6.0	-80	Moderate

## 3.0 METHODOLOGY

### 3.1 Parameters for the Ionospheric Wave Front Velocity



**Figure 2: Map of vertical ionospheric delays over the eastern U.S. on November 20, 2003 20:15 UT.**

An ionospheric wave front is assumed to be a straight and semi-infinite line and to move with constant speeds relative to the ground [1]. These assumptions are made to simplify the analysis of ionospheric wave-front velocities. Figure 2 is a snap shot of vertical ionospheric delay map on the well-known November 2003 ionospheric storm. As presented in Figure 2, the ionospheric wave front

observed at the edge of sharp transitions between red and blue regions could be modeled as an approximately linear line locally.

Figure 3 and Table 2 illustrate how we parameterize the velocity of the ionospheric wave front. In Figure 3, the ionospheric wave front is inclined and moves southwest, as indicated by the black arrow. In addition, the GPS satellite is moving southeast along the IPP track, as indicated by the orange dotted line. To model this ionospheric front motion, four parameters are defined: the orientation of the ionospheric wave front, the speed of the ionospheric wave front, and the direction and speed of Ionospheric Pierce Point (IPP). The orientation of the front,  $i$ , and the direction of IPP,  $\alpha$ , are the angles between the y-axis and the wave front (measured in a counterclockwise direction starting the y-axis), the x-axis and the IPP moving direction (measured in a counterclockwise direction starting the x-axis).

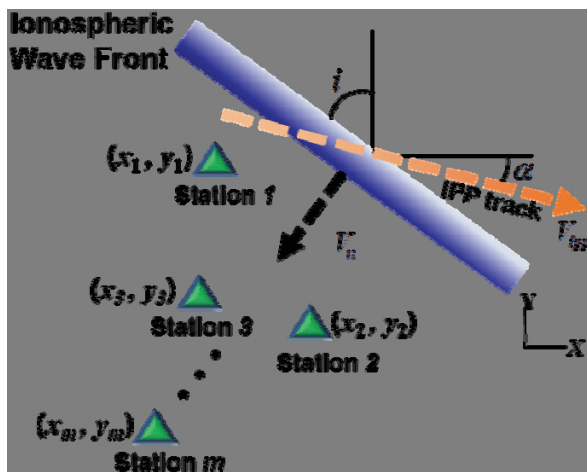


Figure 3: Illustration of the velocity parameters of the ionospheric wave front.

Table 2: Definition of velocity parameters of the ionospheric wave front.

Parameter	Definition
$\alpha$	Direction of ionospheric pierce point (IPP)
$V_{ipp}$	Speed of ionospheric pierce point (IPP)
$i$	Orientation of ionospheric wave front
$V_n$	Speed in normal direction of ionospheric wave front

Note that  $V_n$  contains the velocity component resulting from the movement of GPS satellite in addition to that of the actual ionospheric front motion with respect to the

ground. Thus, to determine the ionospheric wave-front velocity with respect to the ground, it is necessary to calculate the speed and direction of the IPP motion caused by the motion of GPS satellites.

### 3.2 Velocity computation of Ionospheric Pierce Point (IPP)

Under nominal conditions, the ionosphere may be thought of as a thin shell surrounding the earth with a mean ionosphere height,  $H_{iono}$ , of 350 km. The ionosphere Pierce Point (IPP) is defined as the point where the line-of-sight vector and the spherical shell at height  $H_{iono}$  intersect, as shown in Figure 4 [5].

The position of the IPP in the geodetic coordinate frame at a particular time can be calculated from the geometric relationship (parameterized by  $\chi$ ,  $\xi$ ,  $\Psi$ ) between the GPS satellite position and the user position. After two coordinate transformations, from geodetic LLA coordinates to ECEF coordinates and from ECEF coordinates to North-East-Down (NED) local frame coordinates, the position of IPP at a specific epoch in the NED local frame coordinate is determined. Because those geometric information are provided at regular time intervals, we can compute the velocity of IPP using the location of IPP and the data sampling time.

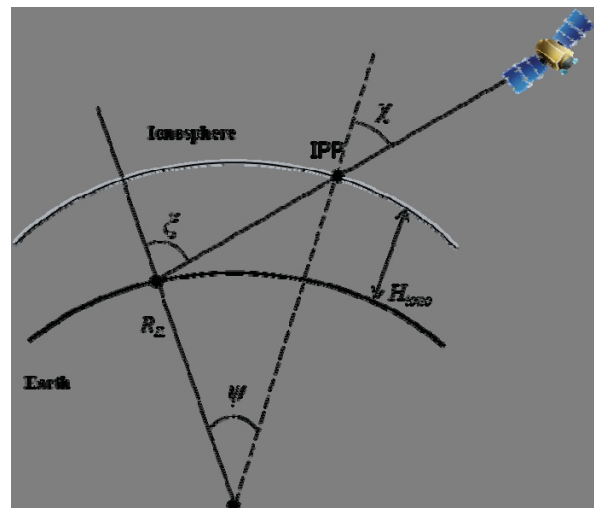


Figure 4: Ionosphere thin shell model and definition of the Ionosphere Pierce Point (IPP) and angles.

### 3.3 Ionospheric Front Velocity Computation Methods

As shown in Figure 3, after calculating the IPP speed and direction, there still are two parameters to be determined: the speed in normal direction of ionospheric wave front,  $V_n$ , and orientation of the ionospheric wave front,  $i$ . Note that, as mentioned above, IPP velocity

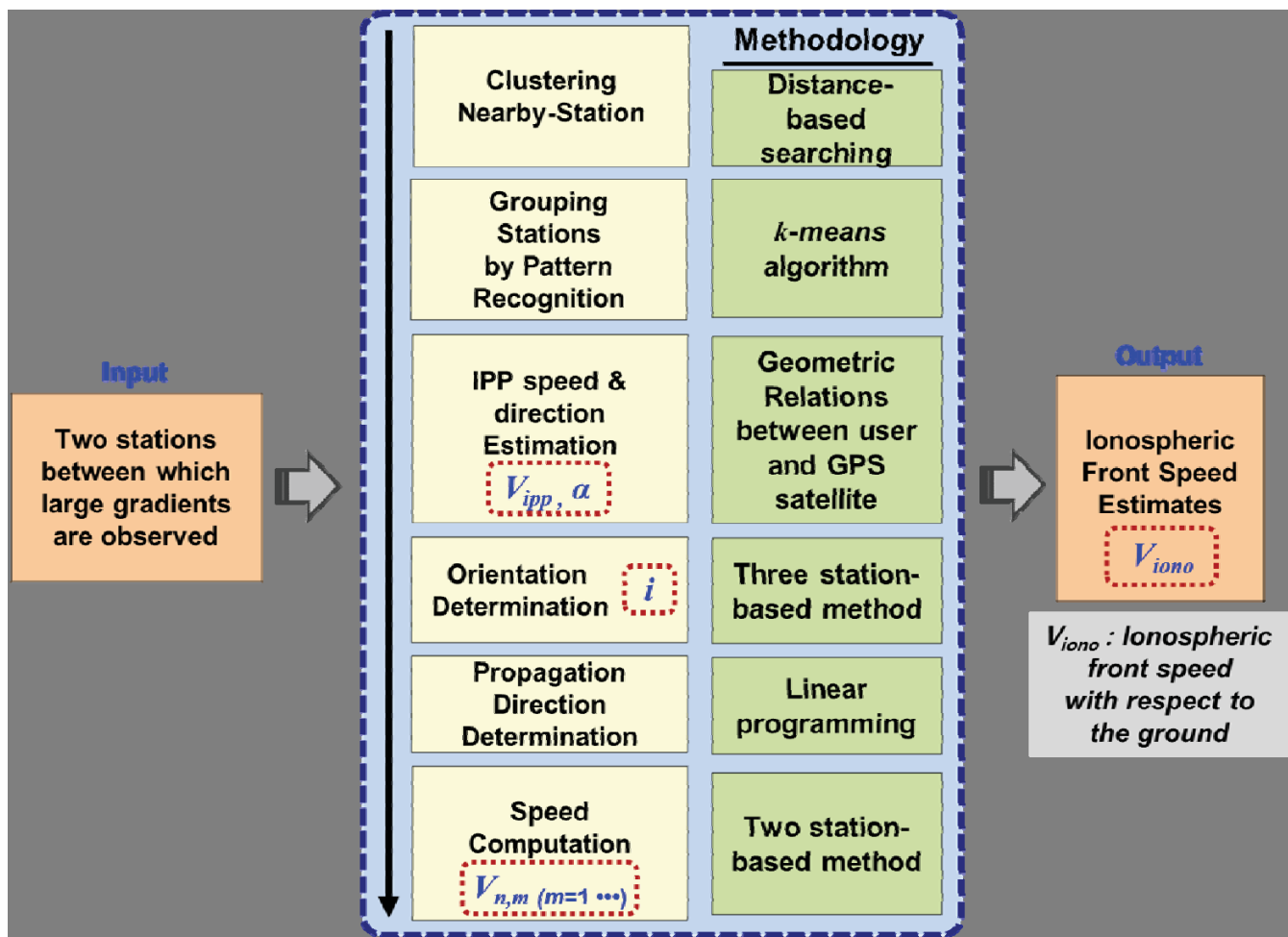


Figure 5: The automated ionospheric front velocity estimation algorithm.

component is included in  $V_n$  and thus this IPP velocity component will be removed from the normal speed of the front (See Subsection 4.7). To solve these two unknowns, two methods, a three-station based method and a two-station-based method, are employed.

In a previous work [9], an automated ionospheric velocity computation algorithm based only on the three-station-based method was developed to estimate the speed and direction of the ionospheric front. In this method, the travel time during which the ionospheric wave front swept through a pair of stations is measured to compute the speed and the third station is used to observe the ionospheric wave front direction [1]. However, this automated algorithm often returned faulty results corrupted by measurement errors and limited by the underlying assumptions of the simplified wave front threat model (the wave front is assumed to be a straight line and to move with a constant speed relative to the ground) [1]. While a manual two-station-based method is used to compensate for the weakness of this automated algorithm, this method was done manually.

Thus, in this paper, we combine the three-station-based method and two-station-based method to utilize advantages of each computation method for automation. The detailed procedures of those methods used to automatically compute the orientation and speed of the ionospheric wave front will be described in Subsections 4.5 and 4.7.

#### 4.0 AUTOMATED IONOSPHERIC FRONT VELOCITY ESTIMATION ALGORITHM

##### 4.1 Overview

In this paper, we develop an automated algorithm for ionospheric front velocity estimation. This automated algorithm is composed of six steps, indicated by the yellow boxes shown in Figure 5: Clustering Nearby-Stations, Grouping Stations by Pattern Recognition, IPP Speed and Direction Estimation, Orientation Determination, Propagation Direction Determination, and Speed Computation. The Green boxes show the methodology used in each corresponding step.

Two stations between which extremely large ionospheric spatial gradients are observed are taken as an input to the automated algorithm. Four parameters in blue font, the speed and direction of IPP ( $V_{ipp}$ ,  $\alpha$ ) and the orientation and speed of the front ( $i$ ,  $V_n$ ), are automatically computed from the six steps of the automated procedure. Once all four parameters are determined, the automated algorithm finally computes the ionospheric front speed estimates relative to the ground. Experimental results will be discussed in Section 5, and the details of each step of this automated algorithm are described in the following subsections.

#### 4.2 Clustering Nearby Stations

The automated ionospheric front velocity estimation algorithm first searches for nearby reference stations whose distances from the two stations (taken as the input to this algorithm) are less than a predefined threshold (e.g. 300 km). The underlying assumption in this step is that of the model of a linear, semi-infinite wave front with constant speed would be valid in a local area for nearby stations.

#### 4.3 Grouping Stations by Pattern Recognition

The second step selects stations that show a similar ionospheric delay pattern from those clustered by the first step of the automated algorithm.

The top plot in Figure 6 shows time history of ionospheric delay measurements observed from five CORS stations, and the lower plot illustrates the motion of the ionospheric front to the stations (only four stations are represented for simplicity). The numbers in red shown to the right of the names of the four stations indicate the order in which the stations were impacted by the wave front. As the ionospheric wave front approaches the stations, the ionospheric delays start increasing. When the wave front crosses the stations, the ionospheric delays of each station reach the local maximum value within the arcs. After the ionospheric front has passed by the stations, the delay measurements start decreasing. From this observation, the peak delay times (indicated in "P") are used to determine the travel time of the front from one station to others and the order in which the stations were affected by the wave front.

The patterns of the ionospheric delays observed by the stations affected by the same ionospheric front exhibit similar tendencies because they experience the same ionospheric spatial gradient embedded in the front. In previous works, stations whose ionospheric delay patterns are similar were selected manually, and thus this was a time consuming task. As the data to be processed increases, more time and effort, accordingly, are required. Thus, it is important to automate the process of

recognizing which ionospheric delays have a similar pattern in developing the automated estimation algorithm. In this work, we apply a pattern recognition technique, the  $k$ -means algorithm, to automatically choose stations that have similar ionospheric delay patterns.

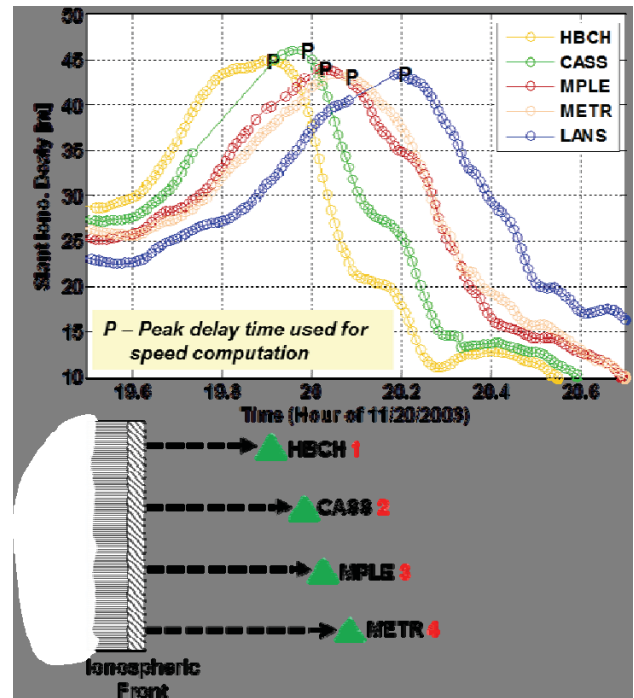


Figure 6: Concept of ionospheric front speed computation.

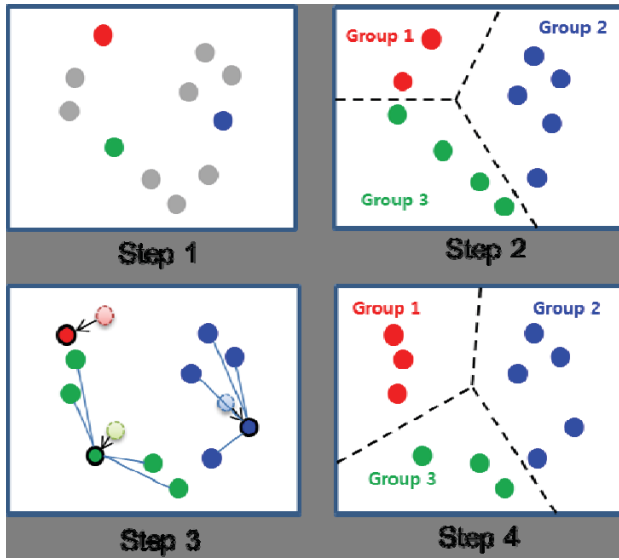
##### 4.3.1 $k$ -means Algorithm

The  $k$ -means clustering is one of clustering formulation the most widely used and studied. Given a set of  $n$ -data points in real  $L$ -dimensional space,  $\mathbf{R}^L$ , and integer  $k$ ,  $k$ -means clustering is a problem to determine a set of  $k$  points in  $\mathbf{R}^L$ , which are called centers, to minimize the mean squared distance (Euclidean norm) from each data point to its nearest center [10]. One of the most widely used methods for solving the  $k$ -means clustering problem is the  $k$ -means algorithm, a simple iterative scheme for finding a local minimum solution [10].

The  $k$ -means algorithm is based on the simple observation that the optimal placement of a center is at the centroid of the associated cluster [10], [11]. As illustrated in Figure 7, the  $k$ -means algorithm can be implemented as follows. The first step randomly selects initial  $k$  centers (in color) from the data set. The second step then creates  $K$ -groups by associating every individual point with the nearest initial randomized center point. Next, the centroid (circled in black) points of each of the  $k$  clusters become the new



center points. The second and third steps are iterated until a predefined level of convergence has been met.



**Figure 7: Illustration of principle of the *k*-means algorithm.**

#### 4.3.2 Application of *k*-means Algorithm

In pattern recognition, a feature vector indicates a vector whose components can represent the numerical features of any object, including length, width, color, number, shape, and so on. From the point of view of the application of the *k*-means algorithm, a set of  $n$  data points in real  $L$ -dimensional space  $\mathbf{R}^L$  used in the problem of *k*-means clustering can be considered as a sort of feature vector. To classify arbitrary ionospheric delay curves into the predefined total number of groups according to the type of ionospheric delay patterns, selection of the feature vector that can accurately reflect the characteristics of the ionospheric delay data is required. The better determined the feature vectors, the better performance of pattern recognition we can expect. Thus, the method to determine the components of the feature vector is important.

In this paper, we use 31 real coefficients obtained from a 30<sup>th</sup>-order polynomial curve fitted to ionospheric delay data as components of the feature vector. This vector becomes an input of the *k*-means algorithm. In addition, a  $k$  of 3, which is the other input of the *k*-means algorithm, is chosen to partition ionospheric delays into three clusters. This method searches for the optimal order of polynomial fit and  $k$  in offline analysis.

The three clusters may contain three different types of ionospheric delay patterns. One of the clusters among the three consists of the ionospheric delays whose pattern is similar to that of the ionospheric delays observed by one

of the two input stations (see Subsection 4.1). Another cluster among the three also contains the ionospheric delays whose pattern is similar to the ionospheric delay pattern observed by the other station. The remaining ionospheric delay patterns that are different from those of both two stations are separated into the third cluster. Because the distance between the two stations is short and the stations track the same satellite at a particular time interval where large ionospheric spatial gradients are observed, almost all of those ionospheric delay patterns might be similar.

In this procedure, thus, it is desirable to combine two types of ionospheric delay patterns that are similar to those of the two input stations. Finally, the stations that observed the ionospheric delays within the union of the two clusters, including the two stations, are used in the following procedure.

#### 4.4 Velocity Estimation of Ionospheric Piece Point

From Subsection 3.2, it was shown that the velocity of IPP can be directly estimated from the information of some geometric relations between the user and the GPS satellite and steps of coordinate transform.

#### 4.5 Orientation Determination

The goal of this step is to determine the orientation (represented as  $i$ ) of the ionospheric front using the three-station-based method. As shown in Figure 3, the ionospheric wave front sweeps through station 1, station 2, station 3, and then other stations sequentially. Let  $t_1$  be the instantaneous epoch at which the wave front crosses Station 1. Similarly,  $t_2$  and  $t_3$  are the times corresponding to Stations 2 and 3. The points  $(x_1, y_1)$ ,  $(x_2, y_2)$ , and  $(x_3, y_3)$  indicate the locations of the first three stations (green triangle) that are impacted by the front. Then, the relative positions of Stations 2 and 3 to Station 1 can be defined as follows:

$$dx_p = x_{p+1} - x_p; dy_p = y_{p+1} - y_p \quad (p=1,2) \quad (1)$$

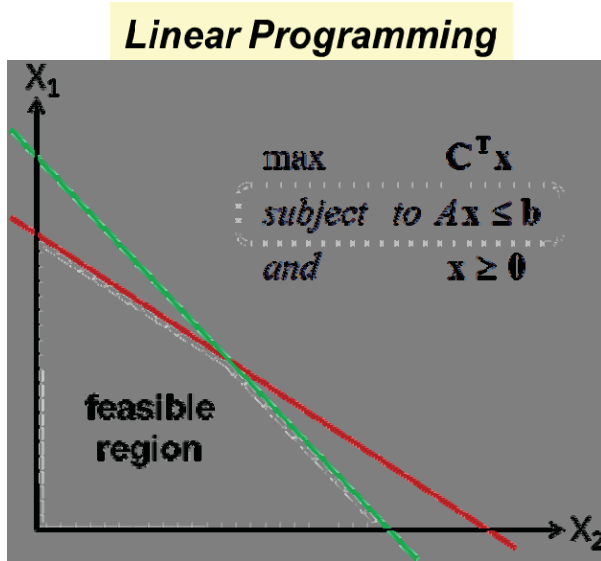
From all these given values following two equations are derived:

$$\begin{aligned} dy_1 + \tan(90-i) \cdot dx_1 &= V_n \cdot \sin(i) \cdot dt_1 \\ V_n \cdot \sin(i) \cdot dt_2 - \tan(90-i) \cdot dx_2 &= dy_2 \end{aligned} \quad (2)$$

where the orientation,  $i$ , is in degrees. Solving these two equations simultaneously, two unknowns,  $V_n$  and  $i$ , are obtained. Under the assumption that an ionospheric front moves in a constant direction, we fix the orientation parameter obtained in this step and solve for a speed using the two-station-based method (see Subsection 4.7).

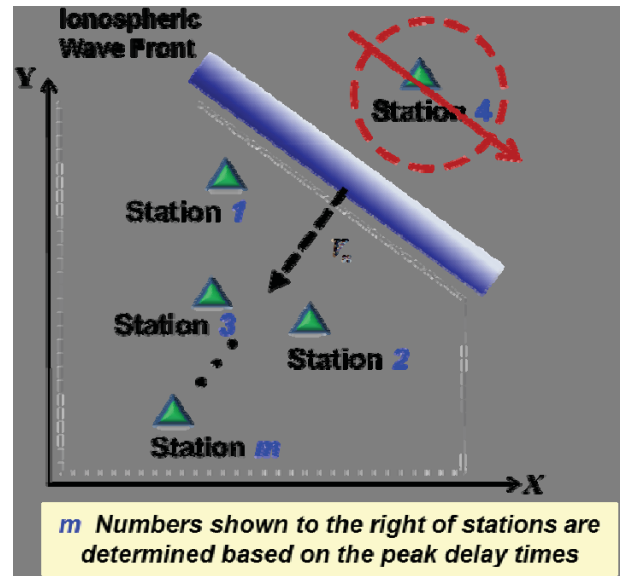
#### 4.6 Propagation Direction Determination

Once the orientation of the ionospheric front is obtained, this step determines the forward propagation direction of the front by using the order of peak delay times and the known locations of the stations. However, the peak delay times and locations of the stations do not always agree with the wave-front motion under the assumption of unvarying velocity. This uncertainty mainly comes from measurement errors. Thus, this step applies a linear programming method to remove any stations that provide faulty measurements.



**Figure 8: Feasible region (gray colored region) and non-feasible region are defined by a set of linear constraints (represented as gray dotted box) in the linear programming method.**

In the linear programming method, a feasible region (gray region) is defined by a set of linear constraints (represented in a gray dotted box) [12], as shown in Figure 8. Figure 9 illustrates how to apply the linear programming method in this work. The numbers shown to the right of the stations indicate the order in which stations are hit by the front. This order is determined based on the peak delay times. In this example, we assume that the wave-front actually propagates southwest. However, according to the order of the hit and the location of stations, the wave front appears to propagate southwest until the front impacts the third station and then moves backward (northeast) until the wave front hits the fourth station. This scenario is not feasible under the assumption that the ionospheric wave-front moves in an unvarying propagation direction in a local area during a short period. This implies that the fourth station observed faulty measurements, which should be removed to correctly estimate the propagation speed of the front.



**Figure 9: Illustration of application of the linear programming method to remove any stations that do not meet the assumption of linear front line and unvarying propagation direction.**

To do this, this step first determines an equation of the linear line, i.e., ionospheric wave front, based on given orientation calculated from the three-station-based method. The equation of the linear line is defined in a local frame coordinate that takes location of the first swept station as the origin of the coordinate and is as follows:

$$y - \tan(90 + i) \cdot x = 0 \quad (3)$$

If the substitution of the position of a station (relative position to the first station) makes the left side of Equation (3) greater than zero, a plus sign will be given to that station. In this way, the one region (gray colored) in Figure 9 where the order based on the peak delay time and the location of the stations agree well can be defined by substituting each position of the stations for Equation (3) and assigning a plus or minus sign to each station. Finally, the fourth station to which a minus sign is given, due to its location being opposite from the other stations, will be removed.

#### 4.7 Wave Front Speed Estimates

From the previous steps of the automated algorithm, three out of the four velocity parameters are determined. In this step, the final speed estimates of the wave front are obtained.

As described in Subsection 4.5, once the orientation parameter  $i$  is computed by the three-station-based

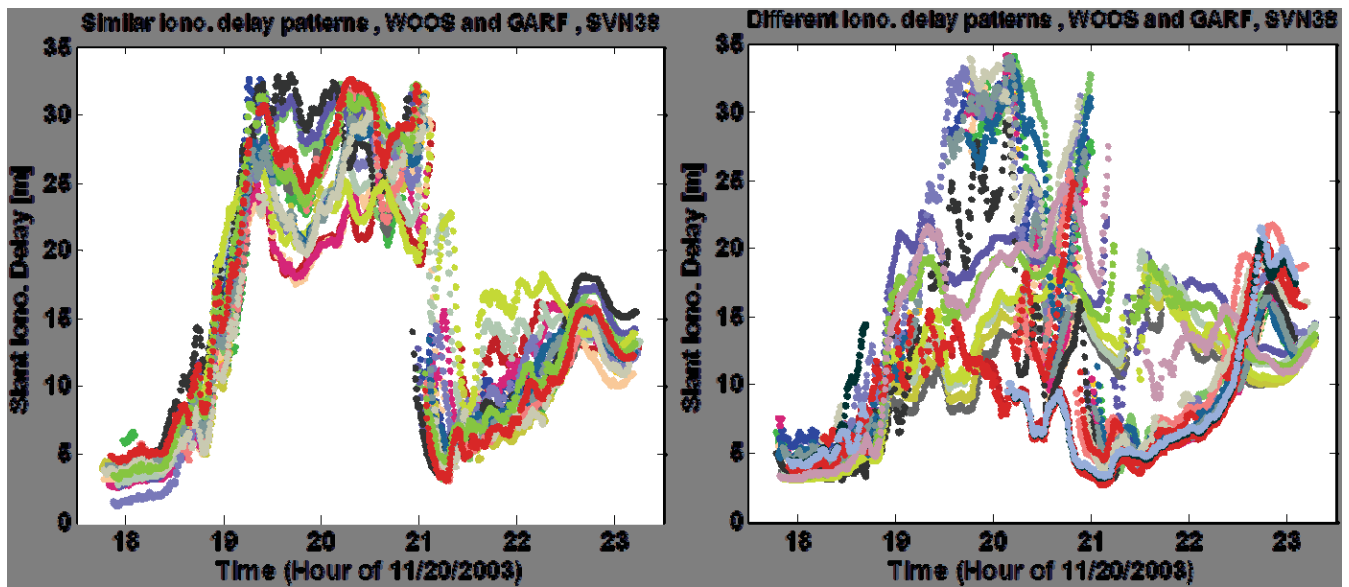


Figure 10: Results from the ‘Grouping Stations by Pattern Recognition’ of the automated algorithm. The left plot shows the ionospheric delay patterns, which are similar to those of GARF and WOOS stations, and the right plot shows ionospheric delays whose patterns are different from those of the two station, GARF and WOOS.

method, a range of the front speed in normal direction is estimated by constructing all feasible pairs of stations and computing speeds for the given front orientation using the two-station-based method. This procedure is as follows: Equation (1) indicates the relative position of the stations to the first impacted (by the front) station. Thus, in the two-station-based method,  $p$  can be from 1 to the value that is less than the total number of the stations ( $m$ ) by one. Changing Equation (2), following equation can be derived:

$$V_n = \frac{(dy_p + \tan(90 - i) \cdot dx_p) \cdot \cos(90 - i)}{dt_p} \quad (4)$$

$$(p = 1, 2, \dots, m - 1)$$

Because the front orientation is given, only  $V_n$  (the speed in normal direction of the front) is unknown. In Equation (4), upper part indicates the distance between two stations projected onto the line perpendicular to the wave front and lower indicates the travel time of the front that is determined based on the peak delay times. As mentioned above, this speed estimate contains the velocity component resulting from the movement of the GPS satellite in addition to that from wave front motion with respect to the ground. Thus, after taking the mean of the range of speeds, which is computed by solving Eq. (4) over all feasible pairs of stations, and removing the component of the IPP speeds from the mean value, as shown in Equation (5), the speed estimate (relative to the

ground) of the ionospheric wave front,  $V_{iono}$ , is finally obtained.

$$V_{iono} = V_n + V_{ipp} \times \cos(i - \alpha) \quad (5)$$

where the last term,  $V_{ipp} \times \cos(i - \alpha)$ , indicates the IPP speed projected into the normal direction of the front speed.

## 5.0 RESULTS

### 5.1 Results from Station-Grouping by Pattern Recognition

This section shows results from the step of ‘Grouping Stations by Pattern Recognition’ to confirm that the algorithm performs well in selecting stations that show a similar trend of ionospheric delays. Ionospheric delay estimates observed from stations in northern Ohio/Michigan on November 20, 2003 were processed by the k-means algorithm. The input stations, WOOS and GARF, which track PRN 8 were used in this example.

The left plot in Figure 10 shows ionospheric delay estimates observed by a group of CORS stations that show similar delay patterns to those of two stations, WOOS and GARF. The remaining stations that show different ionospheric delay patterns are well excluded by this algorithm, as shown in the right plot in Figure 10.



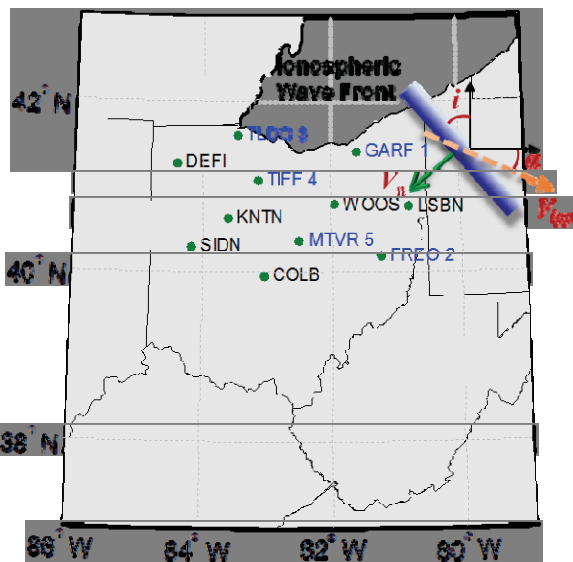
## 5.2 Results from case study

To examine the performance of the automated algorithm, the automated data processing was conducted for the November 20, 2003 ionospheric storm observed at 20:30 UT. This section shows example results from the test.

Figure 11 shows a map of the five CORS stations (in blue) in the state of Ohio used to form pairs of stations for speed computation. The blue bar represents the ionospheric wave front. The numbers shown to the right of the names of the station indicate the order in which the stations were impacted by the wave front. The four velocity parameters are shown in red. The final numerical result for the parameters defined in the front velocity model is presented in Table 3. The negative sign for  $\alpha$  and  $V_{iono}$  means that the GPS satellite moves southeast and the actual wave front moves southwest.

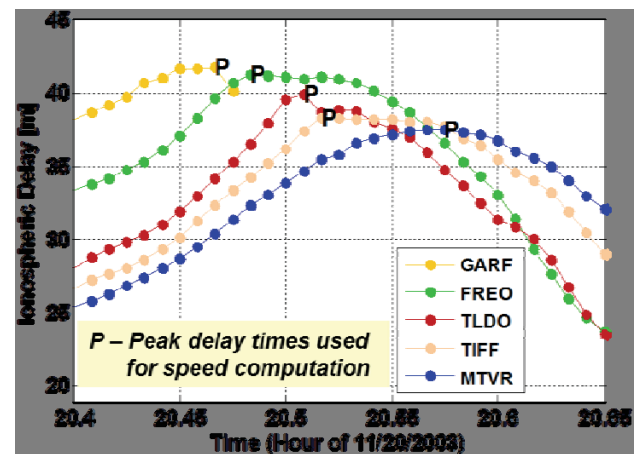
**Table 3: Result from automated procedures for Nov. 20, 2003 ionospheric storm observed at 20:30 UT.**

Parameter	Value
$\alpha$ (deg)	-2.49
$V_{ipp}$ (m/sec)	92.61
$i$ (deg)	43.53
$V_n$ (m/sec)	-648.78
$V_{iono}$ (m/sec)	-578.9



**Figure 11: A map of the five CORS stations (in blue), the order in which the stations were impacted by the front (shown to the right of stations marks) and four velocity parameters**

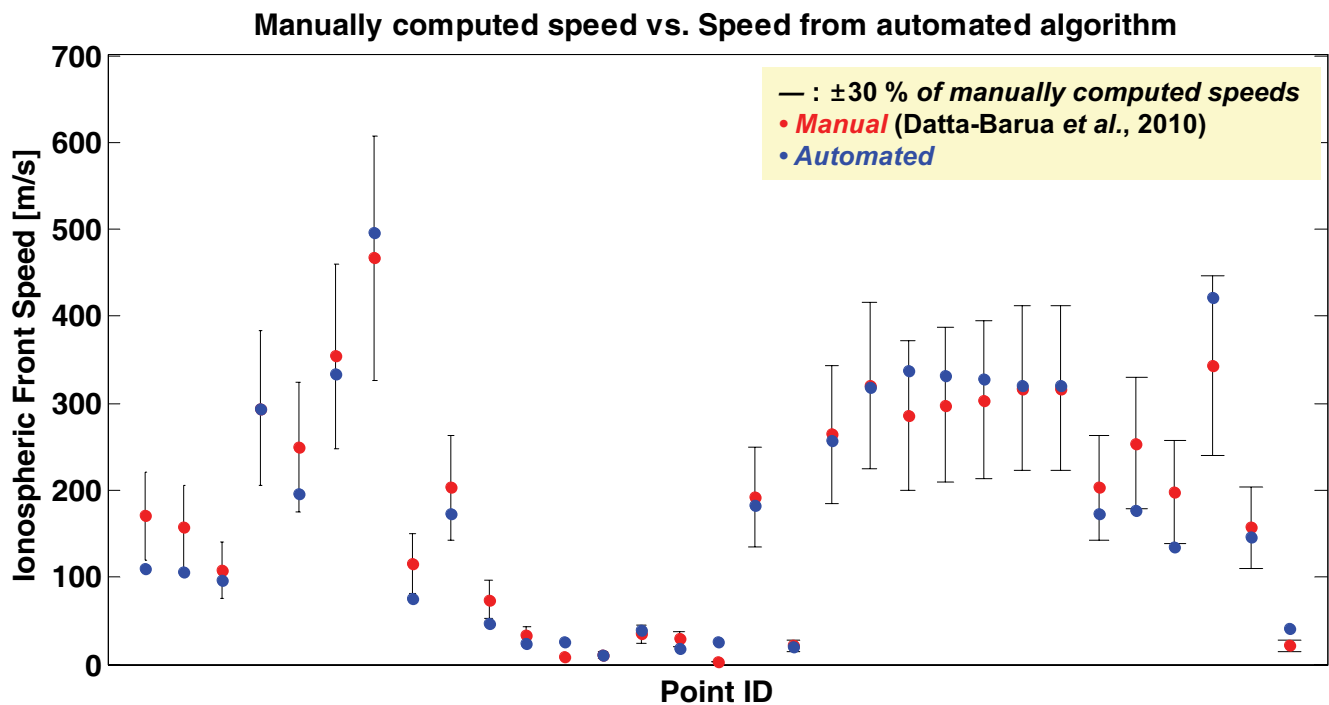
Figure 12 presents the time history of the ionospheric delay measurements observed from the five CORS stations. The points represented by the capital “P” indicate the peak delay times used in the automated algorithm to compute the speed and direction of the wave front. As shown in Figure 11, 12, the order of peak delay times and the location of stations agree well under the assumption of unvarying propagation direction. Accordingly, the orientation and forward propagation direction of the wave front are well estimated.



**Figure 12: Time history of the ionospheric delay measurements observed from five CORS stations and peak delay times (indicated by capital P).**

## 5.3 Performance Validation of the Automated Front Speed Estimation Algorithm

To validate the performance of the automated algorithm, we compared the estimation results for the points of the current CONUS threat space [1] to those computed manually from the previous study. In Figure 13, the ionospheric front speeds, estimated using the automated algorithm (blue dots), are compared to those manually estimated (red dots). Error bars are shown to represent the uncertainty in speed estimates due to measurement errors and inconsistencies between an actual ionospheric wave-front and the model of the front used in this study. From prior work, the speeds were manually estimated by carefully considering errors caused by faulty measurement and violation of assumptions made, i.e., the front is a straight line, semi-infinite and moves with constant speed. Almost all of speed estimates from the automated algorithm fall within error bars of  $\pm 30$  percent of the manually-estimated speeds. Furthermore, the mean of the deviation of automatically estimated speeds relative to manually-estimated speeds is 19.9 percent of the manually-estimated speeds, and this value is much less than the 30 percent. Thus, this new algorithm proposed in this paper is shown to be robust to faulty measurement and modeling errors.



**Figure 13: Comparison between the front speed estimates obtained from the automated algorithm and those manually estimated for performance validation of the automated ionospheric front speed estimation algorithm.**

#### 5.4 Newly-generated Threat Points

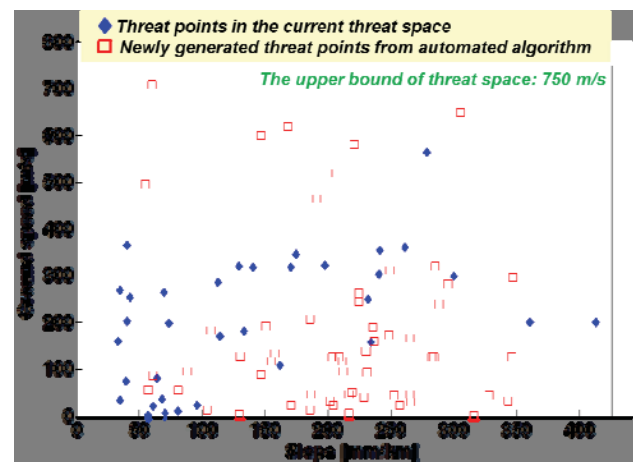
Figure 14 shows 58 additional threat points (represented as red squares) generated from the automated front velocity estimation algorithm and the threat points of the current CONUS threat space (represented as blue diamonds) are shown. Those additional threat points are identified by processing ionospheric delay data obtained from the LTIAM tool.

**Table 4: Ionospheric storm dates investigated for additional data process**

Day (UT dd/mm/yy)	10/29/03 10/30/03 10/31/03 11/20/03
Number of the newly generated points using the automated algorithm	58 points
Number of threat points in the current threat space	34 points

Table 4 lists the ionospheric storm dates investigated in this study. The estimated speeds are all within the bound of the current threat space. Previously, it took

considerable effort and time to identify one threat point. However, this automated ionospheric front velocity estimation algorithm enable us to populate the threat space with additional threat points with relatively little effort and time.



**Figure 14: Additional threat points (red squares) generated from the automated ionospheric front velocity estimation algorithm by processing ionospheric data on the four storm days (shown in Table 4).**

## 6.0 CONCLUSION

This study developed an automated algorithm for the velocity estimation of the anomalous ionospheric front. The current CONUS threat model has limitations due to the lack of sufficient data (uncertainty as to whether or not the existing model bounds could bound future events). Thus, it is not acceptable to rely upon the existing model indefinitely. In this circumstance, since the maximum of the next solar cycle is expected to occur in around 2013, ionospheric data should be monitored continuously going forward. Thus, the necessity of the development of this automated algorithm comes from the vast amount of data that should be processed.

This paper examines the performance of this automated algorithm by comparing automatically estimated speeds (parameterized by front speed, orientation, and propagation direction) to those manually estimated in previous work. This automated estimation algorithm proved to be robust to faulty measurements and uncertainties in modeling the threat model geometry.

A larger number of velocity estimates, to be produced by the automated data processing, would help to better understand the behavior of ionospheric wave fronts under geomagnetic storm condition. It will further benefit the GBAS design by understanding the full range of anomalous events.

## ACKNOWLEDGMENTS

The authors thank John Warburton of the FAA William J. Hughes Technical Center and his team for their support. We also would like to thank Jason Burns of the FAA and Per Enge, Todd Walter, and Juan Blanch of Stanford for their support of this work. Jiwon Seo and Sigrid Close acknowledge the support of the National Science Foundation (AGS 1025262-002) and the Office of Naval Research (N00014-10-1-0450). Eugene Bang was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0026599).

## REFERENCES

- [1] Datta-Barua, S., Lee, J., Pullen, S., Luo, M., Ene, A., Qui, D., Zhang, G., and Enge, P., "Ionospheric Threat Parameterization for Local Area GPS-Based Aircraft Landing Systems," *AIAA Journal of Aircraft*, Vol. 47, No. 4, Jul. 2010, pp. 1141-1151.
- [2] Pullen, S., Park, Y.S., and Enge, P., "Impact and mitigation of ionospheric anomalies on ground-based augmentation of GNSS," *Radio Science*, Vol. 44, RS0A21, Aug. 2009.
- [3] Lee, J., Datta-Barua, S., Zhang, G., Pullen, S., and Enge, P., "Observation of low-elevation ionospheric anomalies for ground-based augmentation of GNSS," *Radio Science*, Vol. 46, RS6005, Nov. 2011.
- [4] Lee, J., Jung, S., Pullen, S., "Enhancements of Long Term Ionospheric Anomaly Monitoring for the Ground-Based Augmentation System," *Proceedings of the 2011 International Technical Meeting of The Institute of Navigation*, San Diego, CA, Jan. 2011, pp. 930-941.
- [5] Misra, P., and Enge, P., *Global Positioning System: Signals, Measurements, and Performance*, 2nd ed., Ganga-Jamuna, Lincoln, MA, 2006.
- [6] Lee, J., Seo, J., Park Y., Pullen, S., and Enge, P., "Ionospheric Threat Mitigation by Geometry Screening in Ground-Based Augmentation Systems," *AIAA Journal of Aircraft*, Vol. 48, No. 4, July 2011, pp.1422-1433.
- [7] Seo, J., Lee, J., Pullen, S., Enge, P., and Close, S., "Targeted Parameter Inflation within Ground-Based Augmentation Systems to Minimize Anomalous Ionospheric Impact," *AIAA Journal of Aircraft*, accepted, 2011.
- [8] Lee, J., Pullen, S., Datta-Batua, S., and Enge, P., "Assessment of Ionosphere Spatial Decorrelation for Global Positioning System-Based Aircraft Landing Systems," *AIAA Journal of Aircraft*, Vol. 44, No. 5, September 2007, pp. 1662-1669.
- [9] Ene, A., Qui, D., Luo, M., Pullen, S., and Enge, P., "A Comprehensive Ionosphere Storm Data Analysis Method to Support LAAS Threat Model Development," *Proceedings of the 2005 National Technical Meeting of the Institute of Navigation*, San Diego, CA, Jan. 2005, pp. 110-130.
- [10] Kanungo, T., Mount, D., Netanyahu, N., Piatko, C., Silverman, R., Wu, A., "An Efficient k-Means Clustering Algorithm: Analysis and Implementation," *IEEE transactions on pattern analysis and machine intelligence*, Vol. 24, No. 7, Jul. 2002, pp. 881-892.
- [11] Faber V., "Clustering and the Continuous k-means algorithm," *Los Angeles Science*, vol. 22, pp. 138-144, 1994
- [12] Vaserstein, L., et al., *Introduction to linear programming*, Pearson Education, Upper Saddle River, N.J., 2003