

MATLAB Algorithm Availability Simulation Tool

Shau-Shiun Jan · Wyant Chan · Todd Walter

Published online: 18 February 2009
© Springer-Verlag 2009

Abstract This paper describes a set of MATLAB functions currently being developed for Space Based Augmentation System (SBAS) availability analysis. This toolset includes simulation algorithms that are constantly being developed and updated by various working groups. This set of functions is intended for use as a fast, accurate, and highly customizable experimental test bed for algorithm development. A user-friendly interface has also been developed for the tool. It is open source and can be downloaded from the Stanford GPS Research Laboratory web site (<http://waas.stanford.edu/~wwu/maast/maast.html>). Therefore, it provides a common ground for different working groups to compare their results. This paper demonstrates the utility of this toolset by analyzing the SBAS service volume models for the Conterminous United States. The paper describes the functionality provided within

the software, and shows an example set of contour plots which are the means for investigating how different algorithms and parameters impact availability.

Keywords Space Based Augmentation System (SBAS) · Global Positioning System (GPS) · Service Volume Model (SVM) · Wide Area Augmentation System (WAAS)

Introduction

The availability of the Wide Area Augmentation System (WAAS) (Enge et al. 1996) is determined by the computation of confidence estimates for the corrections to various error sources. Several groups are revising the algorithms for these confidence computations. Additionally the next generation algorithms are being designed. Service volume model (SVM) analysis (Poor et al. 1999) has been used by algorithm developers as a tool to assess relative performance benefits of algorithm or parameter changes. A general SVM tool would include the model algorithms, a facility for setting simulation configurations, and a means for assessing performance through simulation outputs. The model algorithms compute confidence estimates of, user differential range error (UDRE) (Alexander and Griffith 2000; RTCA SC-159 2006), grid ionosphere vertical error (GIVE) (Fries et al. 2001, RTCA SC-159 2006), code noise and multipath error (CNMP) (McGraw et al. 2000, Shallberg et al. 2001), and troposphere delay (TROP) (RTCA SC-159 2006). Simulation configurations include GPS satellite almanac, WAAS reference station (WRS) (Enge et al. 1996) information, user information, geostationary (GEO) satellites, ionosphere grid point (IGP) mask (Fries et al. 2001, RTCA SC-159 2006), and other parameters.

The GPS Tool Box is a column dedicated to highlighting algorithms and source code utilized by GPS engineers and scientists. If you have an interesting program or software package you would like to share with our readers, please pass it along; e-mail it to us at gpstoolbox@ngs.noaa.gov. To comment on any of the source code discussed here, or to download source code, visit our website at <http://www.ngs.noaa.gov/gps-toolbox>. This column is edited by Stephen Hilla, National Geodetic Survey, NOAA, Silver Spring, Maryland, and Mike Craymer, Geodetic Survey Division, Natural Resources Canada, Ottawa, Ontario, Canada.

S.-S. Jan (✉)
Department of Aeronautics and Astronautics,
National Cheng Kung University, Tainan 70101, Taiwan
e-mail: ssjan@mail.ncku.edu.tw

W. Chan · T. Walter
Department of Aeronautics and Astronautics,
Stanford University, California 94305, USA

The outputs of a SVM tool typically include an availability contour, and vertical/horizontal protection level (V/HPL) (RTCA SC-159 2006) contours.

This paper describes a set of MATLAB functions currently being developed for availability analysis. The toolset is called the MATLAB Algorithm Availability Simulation Tool or **MAAST**. The goal is to develop an experimental test bed for SVM analysis algorithm evaluation that is open-source, can be easily updated, has a friendly interface for the user to set simulation options and parameters, and can ultimately provide fast but reasonably accurate results.

The remainder of this paper is organized as follows. “Overall structures of MAAST” discusses the overall structure and flow of the MAAST software. The simulation process is detailed in “MAAST simulation process”. “MAAST simulation outputs” describes examples of outputs. “Conclusions” presents the summary and concluding remarks.

Overall structure of MAAST

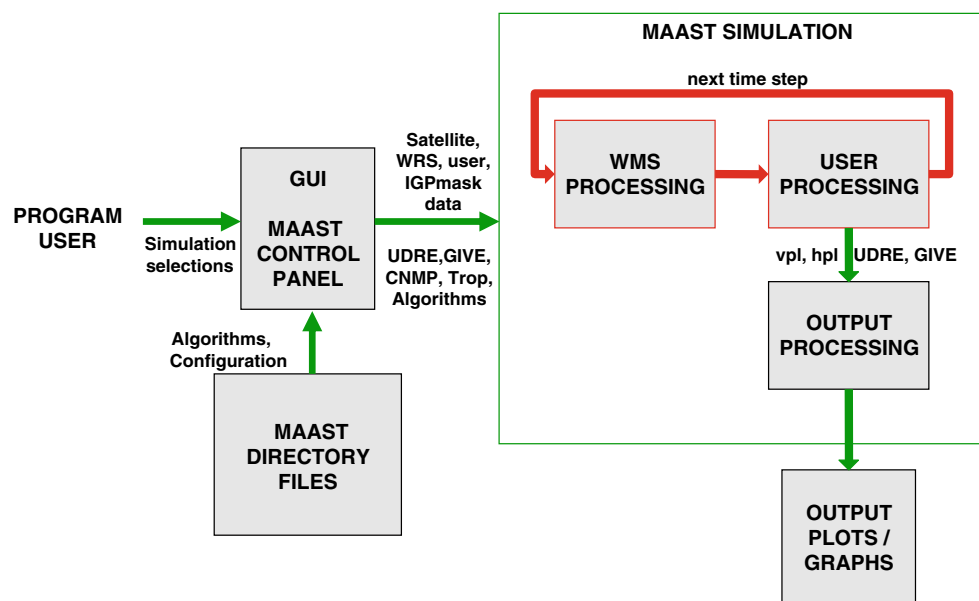
MAAST has four parts: MAAST directory files, graphical user interface (GUI), MAAST main program, and outputs. The MAAST directory files contain WAAS simulation algorithms and configurations. The GUI provides a control panel to allow the program user to make selections from the algorithm and simulation options. Then the selected algorithms and configurations will be fed into MAAST main program *svmrun.m*. The MAAST main program performs WAAS master station (WMS) processing, user processing, and output processing. The outputs provide several graphic options, for example, availability contour,

V/HPL contour, and UDRE/GIVE plots. This approach is summarized in Fig. 1.

The MAAST directory files include:

1. **WAAS reference station (WRS) list:** MAAST uses the 25 U.S. WAAS initial operational capability (IOC) reference stations. The associated file name in MAAST is *wrs25.dat*, which is in the format of (WRS number, WRS latitude in degrees, WRS longitude in degrees, WRS height in meters). It is easily configurable to accommodate different locations. Users can build their own station list in the same format as *wrs25.dat*. Most SBAS reference stations (for example, Japan’s MSAS, and Europe’s EGNOS) are already included in the GUI.
2. **User boundary files:** MAAST simulates users on a rectangular grid, but only the nodes contained inside the specified boundary (for example, CONUS, Alaska), will be used to calculate coverage and to fill in histogram data. The associated files are *usralaska.dat* and *usrconus.dat*, which specify polygon boundaries of Alaska and CONUS, respectively in the format of (latitude in degree, longitude in degree) If program users want to customize their own user boundary, then they need to build their user boundary in the same format as *usrconus.dat* or *usralaska.dat*. There are nine user boundary options in the GUI.
3. **GPS satellite almanacs:** MAAST accepts the standard YUMA almanac format. The default constellation is the standard constellation specified in Appendix B of the Minimum Operational Performance Standard (MOPS) (RTCA SC-159 2006). In addition, other constellations in the YUMA format may be used corresponding to a specified week. YUMA formatted

Fig. 1 Block diagram of MAAST



ephemeris files for GPS can be downloaded from the U.S. Coast Guard website (<http://www.navcen.uscg.gov/ftp/GPS/almanacs/yuma/>). Files must be stored in the same directory as MAAST and named *alm-yuma(week number).dat*, where week number can be specified in the GUI. In addition, the standard Galileo satellite constellation is included in the GUI as an option.

4. **Geostationary satellite list:** there are nine GEO satellites that are currently specified: AOR-E, AOR-W, ARTEMIS, IOR, POR, N107, N133, MTSAT-1, and MTSAT-2. The associated file in MAAST is *geo.dat*, which is in the format of [GEO PRN number, GEO latitude in degree, flag if the button should default to on, and name for the button]. If program users want to customize their own GEO list, then they need to build their own GEO list in the same format as *geo.dat* and replace *geo.dat* with their version. Note the PRNs need to be between 120 and 138 as per the WAAS MOPS (RTCA SC-159, 2006). The GUI has space for up to eleven options.

MAAST simulation process

An overview of the main simulation engine of MAAST is shown in the upper right section of Fig. 1. It is subdivided into three major components: WAAS master station (WMS) processing, WAAS user processing, and output processing. The corresponding MATLAB functions are *wmsprocess.m*, *usrprocess.m*, and *outputprocess.m*. The WMS processing and user processing blocks constitute the main computational loop and are stepped through in sequence at every time step. Time step resolution is chosen by the program user through the GUI. WMS processing simulates the computations of UDREs, GIVEs, and Message Type 28 (Walter et al. 2001) covariance matrices

performed by the WAAS master station using data gathered from reference stations. These computations are to be broadcast to WAAS users. User processing, on other hand, simulates the WAAS user’s computation of confidence bounds on clock/ephemeris and ionospheric corrections at the user site, from which VPL/HPL can be derived. The output processing block then takes all these data and creates visual outputs of VPL/HPL and availability contours, as well as UDRE and GIVE plots.

WAAS service availability at user locations are based on vertical and horizontal protection levels, which are determined from confidence estimates on corrections to the different error sources. Algorithms for these confidence estimates are being developed by several working groups. Aside from having predefined algorithm functions, MAAST offers common templates for including custom algorithms. This is achieved by defining standardized input and output arguments for each customizable algorithm function. This provides an efficient way for developers to test their own algorithm implementations against the whole system in a modular fashion. Selectable modules for this tool include algorithms for computing troposphere errors (TROP), code noise, and multipath errors (CNMP), and confidence bounds on GPS/GEO clock and ephemeris corrections (UDRE) and ionospheric corrections (GIVE).

The simulation does not include old but active data (OBAD) (RTCA SC-159 2006). Degradation of fast correction, range-rate correction, long-term correction, and en route data are not modeled, and all these degradation terms are defined in Appendix A of the WAAS MOPS (RTCA SC-159 2006).

To gain some perspective on how these algorithm modules fit in the simulation, refer to Figs. 2 and 3 for functional flowcharts of the simulated processing performed by the WAAS master station (WMS) and the WAAS user, respectively, to obtain confidence estimations.

Fig. 2 Functional flowchart of WMS processing

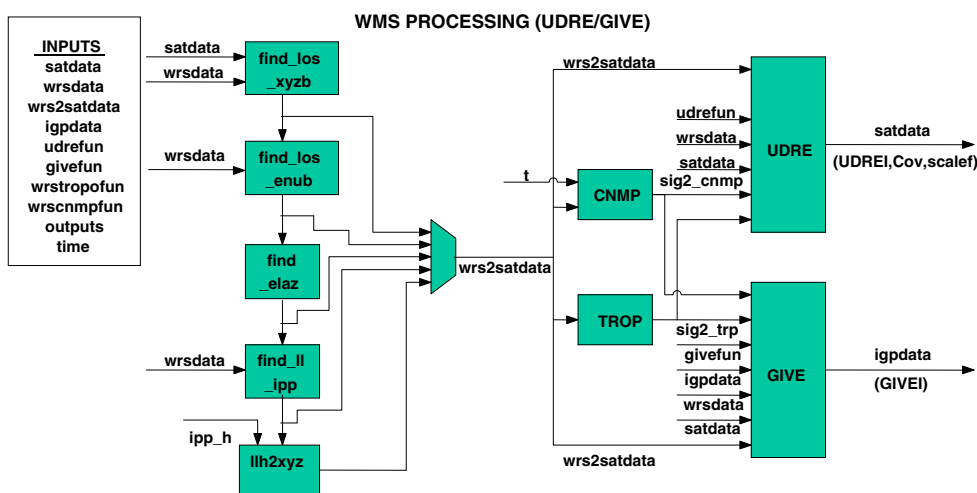
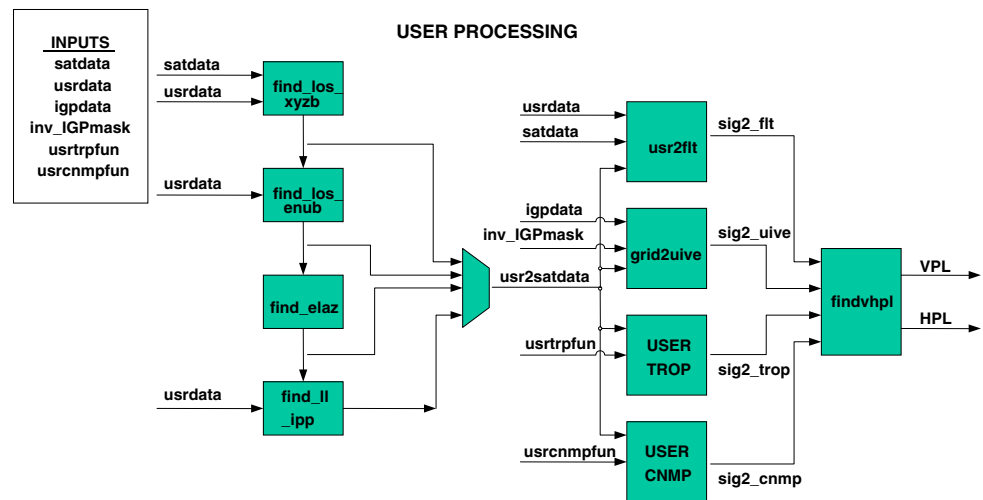


Fig. 3 Functional flowchart of user processing



WMS processing

In the simulation of master station processing, location data of reference stations and satellites for the current time step are passed through functional blocks (left half of Fig. 2) to compute relevant line-of-sight and ionospheric pierce point information for each reference station–satellite pair. Satellite and WRS information are input into the function *find_los_xyzb.m* to give line-of-sight vectors in ECEF coordinates. These are translated into east-north-up coordinates by the function *find_los_enub.m*. Elevation and azimuth are calculated by *find_elaz.m*. The function *find_ll_ipp.m* then computes ionospheric pierce point (IPP) locations. All these data are packaged into a matrix *wrs2satdata* which is passed into succeeding functions that need line-of-sight information. Each row of this matrix corresponds to a particular line-of-sight, while the columns correspond to information fields. The details of the column definitions corresponding to the fields of the matrix, as well as other relevant matrices used in the MAAST, can be found in *init_col_labels.m*.

After line-of-sight computation is done, the TROPO module takes elevation angles as input and generates troposphere error variances. The CNMP module takes as input the elevation angle and/or track time since last cycle slip of each pair and generates the noise and multipath error variance (Shallberg et al. 2001). Here it was assumed that the carrier phase is continuous while the satellite-to-reference station elevation angle exceeds the visibility limit, currently set as 5° by the WAAS MOPS and cycle slips never occur. Using this assumption, the times a satellite rises into view of a reference station are predetermined up to 1 s accuracy before entering the time step loop. This resulted in a marked improvement in the execution time of the track time calculations. The troposphere and CNMP error variances, together with line-of-sight information, are then fed into the UDRE module to generate indexed UDREs and

Message Type 28 (MT28) covariance matrices for each satellite. Likewise, the GIVE module uses this information, together with ionospheric pierce point data, to generate indexed GIVES for each ionospheric grid point.

User processing

User processing uses functional blocks similar to those used in WMS processing for computing line-of-sight data between the satellite-user pairs, as shown in Fig. 3. Using these line-of-sight data, the *udre2flt* module projects satellite UDREs with MT28 covariance matrices into fast and long-term correction variances σ_{fit}^2 for each user line-of-sight. Similarly the *grid2uive* module derives user ionospheric correction variances σ_{uive}^2 from ionospheric grid point GIVES. Implementation of these two modules is based on the WAAS MOPS. User processing uses its own selectable TROPO and CNMP algorithms (McGraw et al. 2000) independent of the selections made for WMS processing. User VPL and HPL for each time step are the final outputs of the user processing block.

MAAST simulation outputs

There are eight output plots currently available in MAAST: availability contour, VPL/HPL contours, UDRE/GIVE histograms, UDRE/GIVE contours, and coverage versus availability plot. There is a “percent” option in the outputs menu, and it has different definitions for the different outputs. In this paper, we choose 95% as an example.

The availability contour plots the availability as a function of user location. We compute the percentage of time that user vertical protection level (VPL) is less than the vertical alert limit (VAL), and the horizontal protection level (HPL) is below the horizontal alert limit (HAL) to determine the availability percentage contour for

continental U.S. (CONUS) or Alaska. The option of 95% here calculates the fraction of users within those regions that had a time availability of 95% or greater. This measure is referred to as coverage.

The VPL/HPL contours plot the VPL and HPL as a function of user location. The option of 95% here indicates that a user at each specific location had a VPL or an HPL equal to or below the value indicated by the color bar. A selection of 50%, for example, would display the median value.

The UDRE histogram plots the probability distributions of UDRE values and the confidences associated with the fast and long-term corrections ($3.29 \cdot \sigma_{fit}$). The GIVE histogram plots the probability distributions of GIVE values and user ionosphere vertical error (UIVE) values. The percent option box is not applicable to either of these plots.

The UDRE map generates a UDRE contour by gathering UDRE data at positions in the satellite orbits and interpolating values to the points in between. The GIVE map generates a GIVE contour by gathering GIVE values at the ionosphere grid points (IGPs). As in the VPL/HPL plots, the percent chosen indicates that the GIVE value at a location is the less than or equal to the displayed contour level 95% of the time.

After making algorithm, simulation, and outputs selections, users then click on the RUN button to begin simulation. The selected output plots are displayed after the simulation, and all relevant data are stored in a temporary binary file *outputs.mat*. Clicking the PLOT button will bypass the simulation process and instead plot the selected output options from data stored in the *outputs.mat*. This allows users to quickly plot other output options if algorithm and simulation configurations have not changed.

Figures 4, 5, 6, 7, 8 show several of the plots generated by a sample run. For this particular example, we chose WIPP ADDs for UDRE, GIVE, and WRS CNMP. We used AAD-B for user CNMP. The simulation was configured for a CONUS user grid, using the 25 IOC U.S. WRSs, satellite almanac from the WAAS MOPS, two GEOs (N107 and N133), 1° user grid and 300-second time steps over a 24 h simulation period.

Figure 4 shows availability contours for CONUS users. It indicates that the coverage for users with availability of at least 95% of time is 99.34% of the CONUS. Figures 5 and 6 show VPL and HPL contours. As described in “Conclusions”, these plots are contours that V/HPLs are less than corresponding values listed in the bottom color bars of the plots for 95% of time. For example, users in the cyan color area of Fig. 5 have a VPL less than or equal to 25 m 95% of time.

Figure 7 is the GIVE contour for CONUS and Alaska. The black circles shown in the plot correspond to the ionosphere grid points (IGPs). Figure 8 plots a UDRE

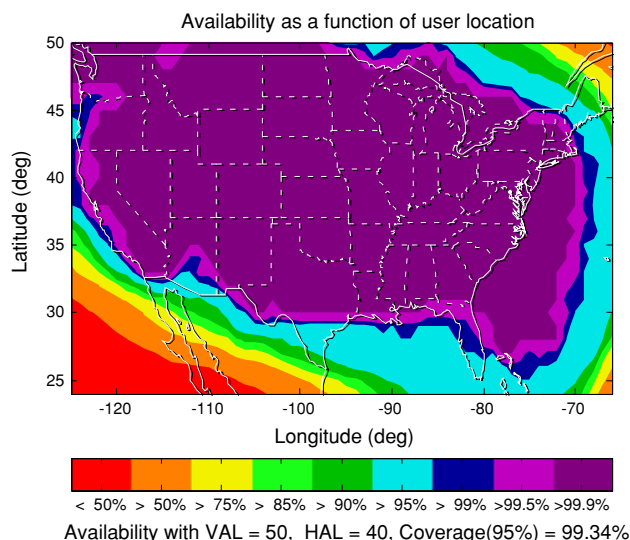


Fig. 4 Availability contour of CONUS

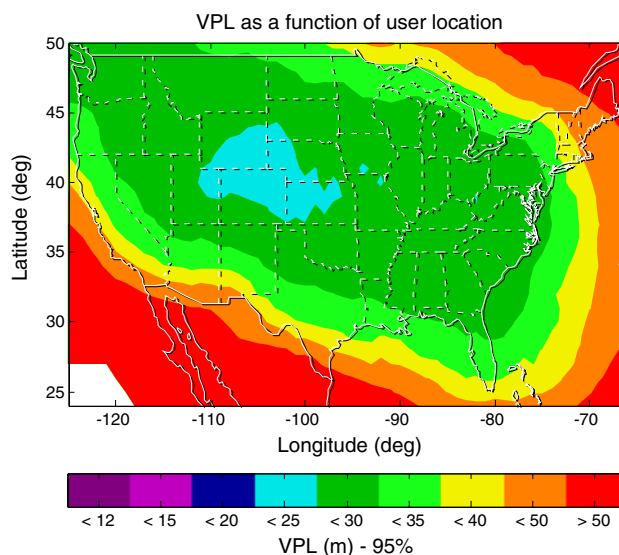


Fig. 5 VPL contour of CONUS

contour by gathering UDRE data at positions in the satellite orbits and interpolating values to the points in between.

Additional details regarding the UDRE and GIVE histogram plots (not shown here for the sake of brevity) and the MAAST graphical user interface are discussed in the additional documentation available from the MAAST web site (<http://waas.stanford.edu/~wwu/maast/maast.html>).

Conclusions

This paper presents a MATLAB Algorithm Availability Simulation Tool for SBAS availability analysis, which implements the real WMS algorithms. MAAST can be

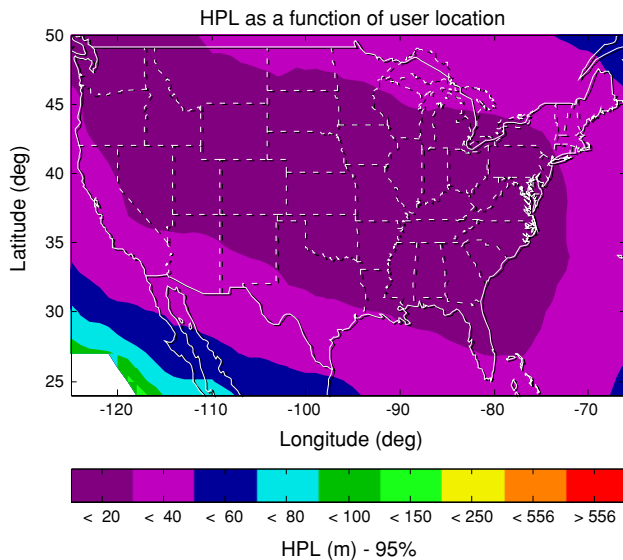


Fig. 6 HPL contour of CONUS

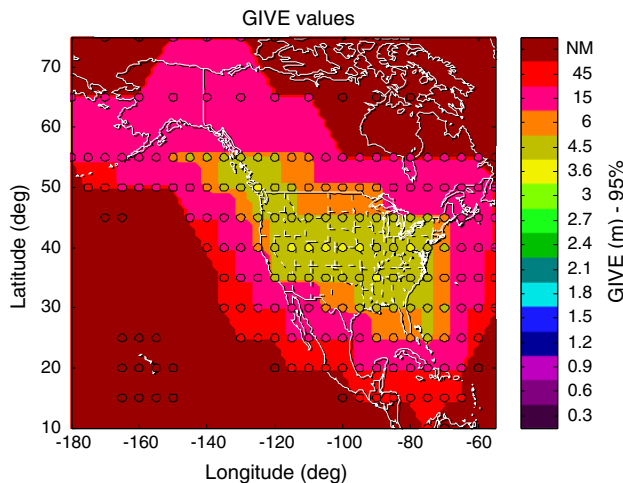


Fig. 7 GIVE contour of CONUS and Alaska

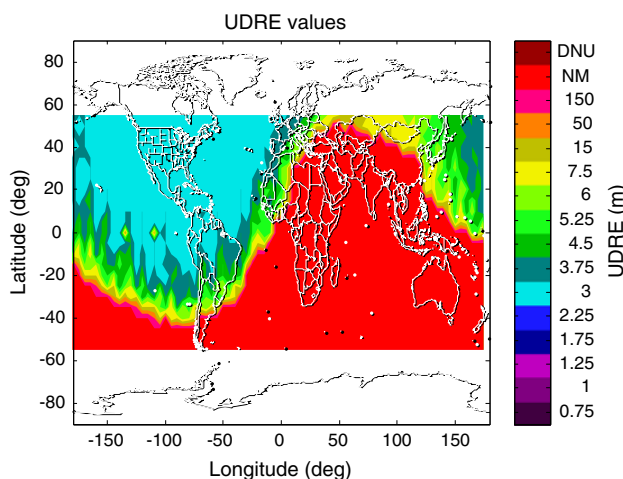


Fig. 8 UDRE contour

used to analyze the SBAS service volume models for CONUS and Alaska and to establish performance figures for certain baseline algorithms. A next step will be to investigate how to modify algorithms and parameters to improve availability and achieve lower VPLs using this tool. One key area of investigation will be how the incorporation of additional civil frequencies will improve availability.

MAAST was intended as an efficient and effective tool for algorithm development. It was not intended to guarantee that we will see exactly that level of availability at each location. In creating MAAST, a number of assumptions have been made. MAAST algorithms are for confidence bounding only; they do not model corrections. Furthermore, it is strictly deterministic, and does not model asset failures in a probabilistic manner. Despite these limitations, the results of this paper show that a simple yet powerful framework has been developed that allows us to rapidly model availability. MAAST can be a valuable tool for SBAS algorithm research.

Acknowledgments The authors would like to thank the Federal Aviation Administration (FAA) for supporting this research. The authors would also like to thank the Taiwan National Science Council for supporting this research under research grant NSC-97-2221-E-006-061.

References

Alexander S, Griffith C (2000) UDRE Monitor 8th revision Algorithm Description Document (UDRE ADD), WAAS Integrity Performance Panel (WIPP), November 20, 2000. (ADD is available to WIPP members only.)

Enge P, Walter T, Pullen S, Kee C, Chao Y-C, Tsai Y-J (1996) Wide Area Augmentation of the Global Positioning System. In: Proceedings of the IEEE, volume: 84 issue: 8, August, 1996

Fries R, Altshuler E, Griffith C, Walter T, Mannucci T, Sparks L, Pi X (2001) GIVE Monitor Algorithm Description Document (GIVE ADD), WIPP, January 22, 2001

MATLAB, The Math Works Inc. MATLAB is a software tool for doing mathematical computations (see <http://www.mathworks.com>)

McGraw GA, Murphy T, Brenner M, Pullen S, Van Dierendonck AJ (2000) Development of the LAAS Accuracy Models. In: Proceedings of ION GPS 2000, Salt Lake City, UT

Poor W, Chawla J, Greanias S, Hashemi D, Yen P (1999) A Wide Area Augmentation System (WAAS) availability model and its use in evaluating WAAS architecture design sensitivities. Global positioning system Vol. VI. The Institute of Navigation, Alexandria

RTCA SC-159 (2006) Minimum Operational Performance Standard for Global Positioning System/Wide Area Augmentation System Airborne Equipment, RTCA/DO-229D, December 13, 2006

Shallberg K, Shloss P, Altshuler E, Tahmazyan L (2001) WAAS Measurement Processing, Reducing the Effects of Multipath. In: Proceedings of ION GPS 2001, Salt Lake City, UT

Walter T, Hansen A, Enge, P (2001) Message Type 28. In: Proceedings of ION NTM 2001. Long Beach, CA