

GBAS airport availability simulation tool

Shuo-Ju Yeh¹ · Shau-Shiun Jan¹

Received: 3 April 2015 / Accepted: 12 May 2015 / Published online: 23 May 2015
© Springer-Verlag Berlin Heidelberg 2015

Abstract A global navigation satellite system augmentation system availability analysis tool has been developed to simulate a ground-based augmentation system (GBAS) prototype, an integrity monitor test bed, to evaluate its operational benefits at an airport of interest. The proposed availability simulation tool includes all GBAS ground facility algorithms as well as a graphical user interface that allows the user to modify simulation options and parameters. The output of the simulation tool is presented in a Stanford chart to help visualize the performance. The chart indicates the availability and integrity. The performance is evaluated primarily in the vertical position domain because of the weaker satellite geometry and more stringent required navigation performance as compared to those of the horizontal position domain. The simulation tool is implemented in Qt (<http://www.qt.io/>), an open-source cross-platform toolkit, allowing the tool to run on various devices. The computations are performed in the associated C++ code. The Newark Liberty International Airport (ICAO code: KEWR) is used as a simulation example to

demonstrate the utility of the developed tool for investigating how reduced error models impact GBAS availability at the airport.

Keywords Ground-based augmentation system (GBAS) · Integrity monitor test bed (IMT) · Availability

Introduction

The purpose of an integrity monitor test bed (IMT), which is the ground facility algorithm of a ground-based augmentation system (GBAS), is to determine the confidence bounds for the corrections of various error sources (Normark et al. 2001). To quantify the performance of an IMT, required navigation performance (RNP) of the Category I (CAT I) landing operation is used as the performance evaluation criterion (ICAO 2008). The four specific requirements of a RNP are accuracy, integrity, continuity, and availability. This work describes a GBAS airport availability simulation tool to assess relative performance while allowing models and parameters to be changed. The tool includes the model algorithms for determining the confidence bounds, an interface to set the simulation configuration, and a means for assessing performance through simulation outputs. The model algorithms are used to compute the statistics for confidence bounds estimation, including the B -values and standard deviations of the GBAS ground facility. Simulation configuration includes GPS satellite orbit parameters, IMT reference station information, user information, and the simulated error model. The output is a plot in a metric, called a Stanford chart, which shows the protection level and navigation sensor error (NSE) (<http://waas.stanford.edu/metrics.html>). This study also develops a graphical user interface (GUI) in the

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.

✉ Shau-Shiun Jan
ssjan@mail.ncku.edu.tw

¹ Department of Aeronautics and Astronautics, National Cheng Kung University, Tainan 70101, Taiwan

Qt toolkit (<http://www.qt.io/>) and C++ code for GBAS performance analysis. The developed software is called the IMT simulation tool. The GUI is used to evaluate the IMT algorithm performance that can quickly provide reasonable results.

The Qt and C++ source code for the GBAS airport availability simulation tool and samples of the configuration files are included in the *.zip file called GBAS_Airport_Availability_Simulation_Tool_SourceCode which can be downloaded from the Web site <http://tinyurl.com/kv94nad>; 64-bit and 32-bit executable files for LINUX and Windows are also available at this same Web site. The files are also available on the GPS Toolbox Web site at <http://www.ngs.noaa.gov/gps-toolbox/GBAS>.

We first describe the overall structure and flowchart of the developed software and then explain the process details of the simulations. The results of simulations are analyzed and discussed. Finally, conclusions and suggestions for future research are presented.

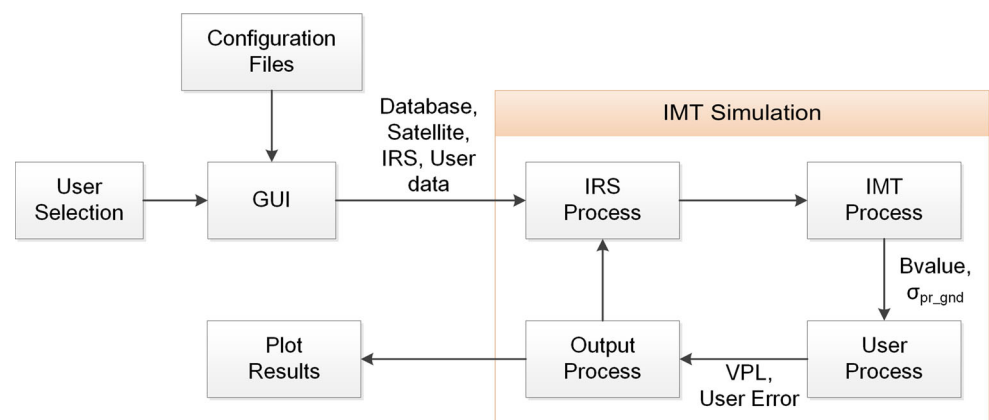
Architecture of IMT simulation tool

The IMT simulation tool is organized into four parts, namely configuration files, GUI, program, and outputs (Fig. 1). The configuration files of the IMT simulation tool are used for simulation configuration. The GUI provides a control panel for the user to set the configuration. The configuration files are used by the IMT simulation tool, which comprises the IMT reference station (IRS) process, IMT process, IMT user process, and output process, as shown in Fig. 1. Under the user-defined configurations, the output is given as a Stanford chart to show the performance of the IMT algorithm in terms of meeting RNP requirements.

The configuration files of the IMT simulation tool are as follows:

1. GPS satellite orbit parameters: IMT simulation tool requires both the GPS broadcast ephemeris and almanac for the IRS process. The GPS broadcast ephemeris is to compute the satellite positions used in the simulation tool, and the GPS almanac is applied to check and to monitor the consistency of the corresponding broadcast ephemeris. The input formats of the ephemeris and almanac for the IMT simulation tool are RINEX and YUMA, respectively. YUMA almanac files for GPS can be downloaded from the U.S. Coast Guard Web site. The ephemeris parameters from RINEX navigation messages can be downloaded from the Scripps Orbit and Permanent Array Center Web site. The files must be saved in the “data” folder in the main IMT simulation tool folder.
2. Integrity monitoring criterion: IMT uses the Gaussian over-bounding method to determine the threshold of integrity monitoring algorithms (Shively and Braff 2000). The measurement of a satellite is declared unsafe for use when its test statistic exceeds the threshold of any integrity monitoring in GBAS ground facility algorithms. The upper and lower thresholds are determined as the mean value plus and minus six standard deviations, respectively. The mean values and standard deviation values are computed based on the test statistics defined in the integrity monitoring algorithms of the GBAS ground facility. The upper and lower thresholds are calculated by substituting the current satellite elevation angle into a given quartic function. Therefore, to set the thresholds, two sets of coefficients are needed for the quartic function of the mean and the standard deviation. Six configuration files are used to record the coefficients. In the files, lines 1–5 are the coefficients of quartic function of the mean for each IRS. Similarly, lines 7–11 represent the coefficients for the standard deviation. The coefficients are in the format [4th coef. | 3rd coef. | 2nd coef. | 1st

Fig. 1 Flowchart of IMT simulation tool



coef. | constant]. Additionally, the sixth elements of lines 7–11 are the inflation factor for each IRS. The associated files are “SQM_Dvgc_Coe.txt,” “SQM_Power_Coe.txt,” “MQM_Acc_Coe.txt,” “MQM_Ramp_Coe.txt,” “MQM_Step_Coe.txt,” and “MQM_Inno_Coe.txt.” “MRCC_Threshold.txt” comprises 30 values in 30 lines (six lines for each IMT reference station (IRS) and three each for the upper and lower bounds). The three values of the bounds are for satellites with elevation angles of 0°–22.5°, 22.5°–65°, and 65°–90°, respectively. The configuration files must be in the same directory as IMT simulation tool.

IMT simulation tool process

The main simulation process is shown on the right of Fig. 1. The four parts are the IRS process, IMT process, user process, and output process. Users use the GUI to input the simulation time window, IRS positions, user position, satellite orbit files, integrity monitoring criterion parameters, and error models, which are all required to start the simulation. The IRS process block simulates the measurements, which are needed in the IMT algorithm, from the user-defined data and passes them to the IMT process block after measurement quality monitoring. The IMT process block computes the *B*-values, range corrections, and standard deviation of the ground facility from the data gathered from IRSs. The information is then broadcast to IMT users. The user process block simulates the IMT users that apply the IMT service, and calculates the NSE from the user-defined data. Using the information from the IMT process, the confidence bounds for the user positioning after applying range corrections, which is called the vertical protection level (VPL) (RTCA 2008), are computed. The output process block stores the VPL and NSE and creates a Stanford chart plot. Figures 2, 3, and 4 show the

individual flowcharts of simulating IRS, IMT, and IMT user, respectively. The figures describe how the modules are implemented in the simulation.

IMT reference station process (IRS process)

At the start of the IRS process simulation, measurements including carrier phase range, code phase range, and satellite positions are generated for the current time step. The carrier phase range and code phase range are determined using the geometric distance between the user position and the satellite position and the error model according to (RTCA 2008). Since the IMT simulation tool allows the users to evaluate the IMT performance in different conditions, the changes for the error model of the measurements could be introduced by user settings from the GUI. The correction of each satellite–IRS pair (channel) is determined. The measurement data and the location data for the IRS are passed to the functional block to determine the line-of-sight information for each channel. Inside the functional block, the information is fed into the find_los_enub function to obtain the line-of-sight vector in earth-centered, earth-fixed (ECEF) coordinates. The vector is then transformed to east-north-up frame. The elevation angle of a satellite is calculated using the find_elev function. These satellite data, which are associated with the IRSs, are packaged as irs2sat data and then passed to the quality monitoring (QM), which includes signal QM (SQM), data QM (DQM), and measurement QM (MQM). To build the monitoring criteria, DQM sets a fixed threshold according to previous studies (Normark et al. 2001) and SQM and MQM determine the thresholds from the information provided in the configuration files. Each QM determines its own test statistics to be compared to the thresholds in methods as follows: SQM takes the signal power from irs2sat data; DQM compares the difference

Fig. 2 Flowchart of IMT reference station process

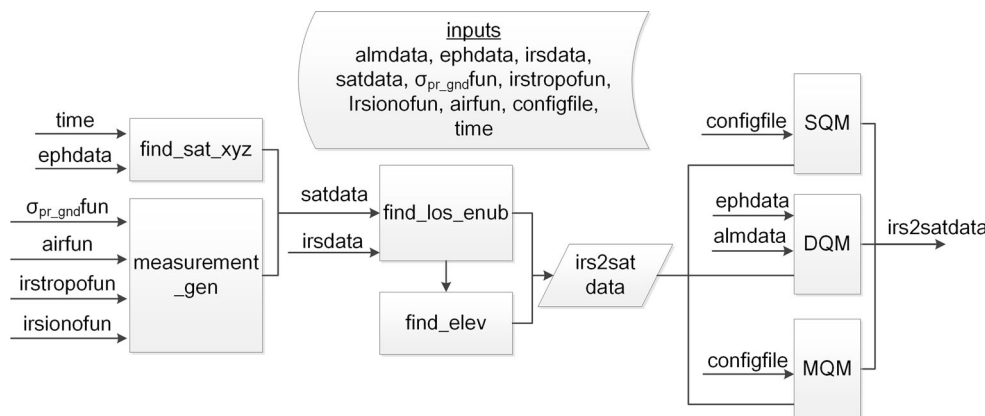


Fig. 3 Flowchart of IMT process

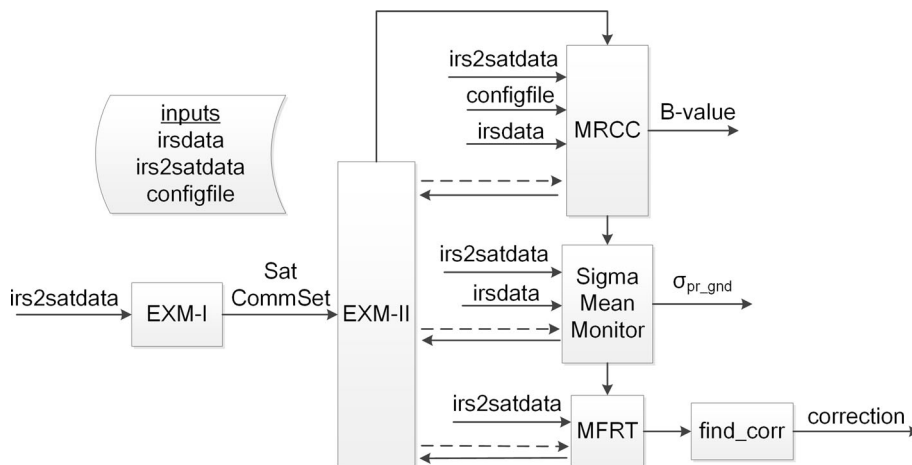
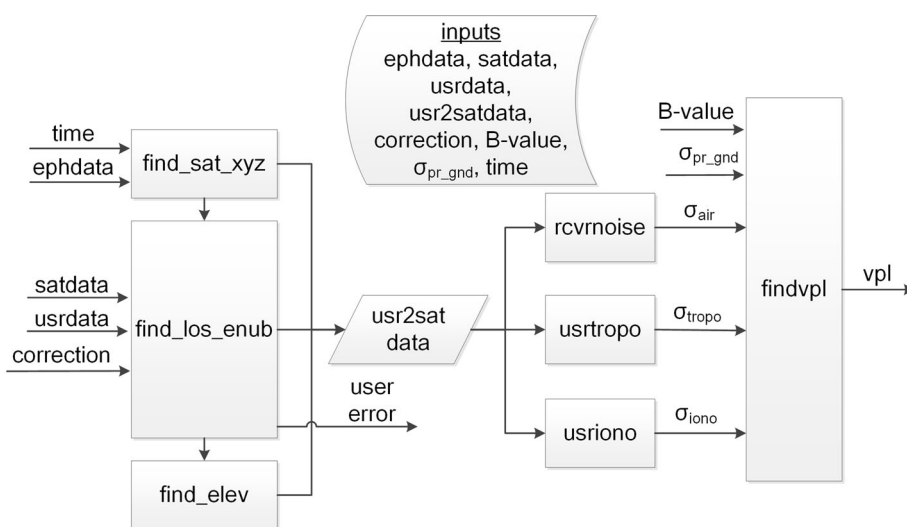


Fig. 4 Flowchart of IMT user process



between the satellite positions determined by almanac and ephemeris data; and MQM determines the tendency of code phase and carrier phase measurements. The channel is flagged if the channel fails any one of the QM tests.

IMT process

In the IMT process, the irs2sat data without a QM flag are input into stage one of the executive monitoring (EXM-I) block. EXM-I gathers the irs2sat data from each IRS and finds a satellite common set. The common set is the intersection of two satellite sets: One set includes satellites that are physically tracked at each IRS and the other set includes satellites without any QM flag. Secondly, the EXM-II block takes the satellite common set from EXM-I and attempts a series of monitoring including multiple-reference consistency check (MRCC), sigma-mean ($\sigma\mu$) monitoring, and message field range test (MFRT) blocks (Normark et al. 2001; Lee et al. 2006).

The MRCC block takes the counts of IRS, threshold information, and the irs2sat data as input to compute *B*-values as a test statistic. The *B*-value represents the consistency of the correction for each satellite across all IRSs. The $\sigma\mu$ monitoring ensures that the broadcast standard deviation of the ground facility error (σ_{pr_gnd}) over-bounds the true error distribution of the broadcast differential corrections. MFRT gathers the corrections from each channel and determines the average corrections for each satellite across IRSs. The purpose of MFRT is to confirm that the computed average pseudorange corrections and correction rates are within the confidence bounds. MRCC, $\sigma\mu$ monitoring, and MFRT flag the channels that failed the test and pass the data to EXM-II. A channel that has failed the test is excluded by EXM-II from the correction determination. When that happens, the *B*-values and corrections must be recomputed for all remaining channels and passed to MRCC to confirm the new set. Since excluding channels lowers the number of satellites in the common set, it is possible to exclude an IRS to find a new common set that

contains more than five satellites. If no channels fail the test, the MRCC, σ_{μ} monitoring, and MFRT output the B -value, σ_{pr_gnd} , and scaled corrections for each satellite in the common set, respectively.

IMT user process

The IMT user process uses blocks similar to those of the IRS process to compute the line-of-sight data between satellites and the user. Based on the location data of IRS, the find_los_enub function applies the corrections received from the IMT process and calculates the navigation sensor error (NSE) for the output process. The IMT user process uses three blocks to obtain the standard deviation of measurement σ_{pr} . The rcvrnoise block determines the airborne pseudorange error σ_{air} . The usrtropo block determines the model of tropospheric residual uncertainty σ_{tropo} . The usiono block determines the model of ionospheric residual uncertainty σ_{iono} . The findvpl block takes these three error models and the σ_{pr_gnd} from the IMT facility process as input to obtain σ_{pr} . Based on (RTCA 2008), the findvpl block obtains the VPL from the line-of-sight data, B -values from the IMT process, and σ_{pr} . The user VPL and NSE for each time step are the final outputs of the user process block.

IMT simulation tool outputs (output process)

The output process block gathers the VPL and NSE of all simulation time steps and displays them in a Stanford chart. The horizontal axis is the NSE of user position with respect to the true position from the GUI input. The vertical axis is the VPL for every time step. Each pixel in the Stanford chart is the occurrence number of a specific VPL–NSE pair. The color bar of each grid indicates the number of epochs in which the pair occurred. The pixel scale is quantized to 0.5 m. The vertical alarm limit for a Category I precision approach, indicated by a horizontal line, is set at 12 m (RTCA 2008). The points in the “System Unavailable” zone indicate an alarm condition that leads to availability loss and possible continuity failure. The Stanford chart is divided by a diagonal line into upper and lower triangles. For pixels on the diagonal line, NSE is equal to VPL. Therefore, pixels in the lower triangle indicate that the NSE exceeds the VPL calculated by the IMT algorithm. This integrity failure condition is called hazardously misleading information (HMI). The label in each zone indicates the number of epochs. The “CAT I Avail.” region displays a percentage. The more detailed definition of the Stanford chart can be found in the Stanford

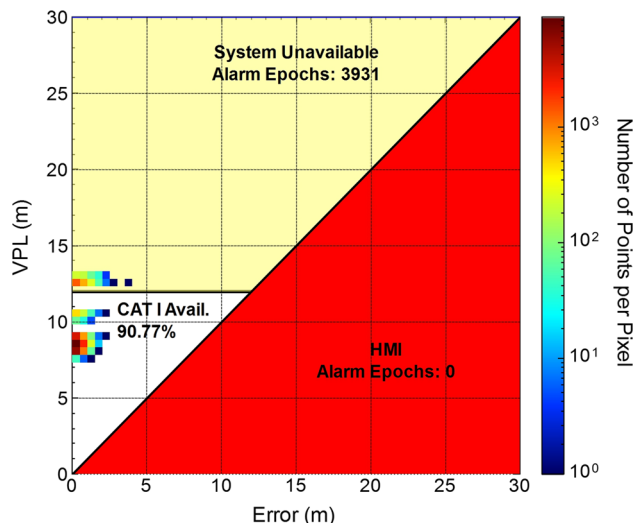


Fig. 5 Simulation result of KEWR with GAD-A receiver

GPS Laboratory WAAS Precision Approach Metrics Web site (<http://waas.stanford.edu/metrics.html>).

Figures 5 and 6 show two of the plots generated in a few test runs with GAD-A and GAD-C used in the σ_{pr_gnd} model, respectively. The GAD-A and GAD-C are the defined ground accuracy designators (GADs) that indicate the performance levels of GPS receiver technologies used in the GBAS reference stations (RTCA 2004). GAD-A represents a legacy receiver with a standard correlator design and a nominal antenna, and GAD-C specifies a receiver with a narrow correlator design and a multipath limiting antenna (RTCA 2004). Therefore, the GAD-C performance is expected to be better than that of GAD-A. The tropospheric residual was set at zero based on the long-term

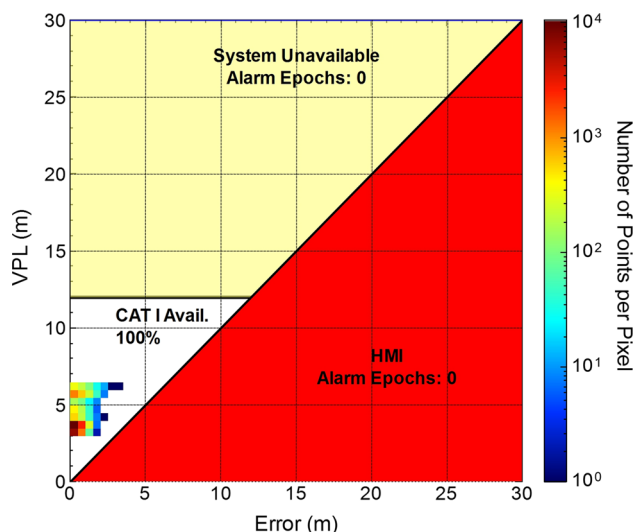


Fig. 6 Simulation result of KEWR with GAD-C receiver

availability test assumption (RTCA 2008). Ionospheric errors and receiver noise were applied. The simulation was configured for Newark Liberty International Airport (KEWR), with five IRSs and a 0.5-s time step over a 6-h simulation period. The simulation time started at 2013/4/22 0:00:00 and ended at 5:59:59. Figure 6 shows that the simulated IMT user with GAD-C receiver met all three safety indicators, namely accuracy, integrity, and continuity, with an availability of 100 % for the CAT I requirement during the simulation time period, and Fig. 5 indicates that the IMT user with GAD-A receiver has GBAS service availability of 90.77 %. As expected, the VPL with GAD-A is larger than that with GAD-C because the GAD-A has larger receiver noise than GAD-C.

Conclusions

We proposed a GBAS airport availability simulation tool that implements all GBAS ground facility algorithms. As shown in the simulation results, the simulation tool can be used to analyze the performance of a GBAS ground facility for a specific airport. The results are displayed in a Stanford chart. The IMT simulation tool has a GUI for users to modify parameters of the GBAS algorithms in order to evaluate the impact of the algorithm parameter changes on algorithm performance. A next step will be to modify the algorithms to include additional satellite constellations to improve availability.

The proposed GBAS airport availability simulation tool is an efficient and effective tool for GBAS planning at an airport of interest. The algorithms are for confidence bounding only and do not model asset failures in a probabilistic manner. Therefore, this simulation tool is not intended to guarantee that users will see exactly the same level of availability at the airport.

Acknowledgments This work was supported by the Taiwan Ministry of Science and Technology under grant NSC 102-2221-E-006-079-MY3. The simulation software tool is based on our previous work (Lin and Jan 2014) with significant changes in algorithms and codes.

References

ICAO (2008) Performance-based navigation (PBN) Manual. Document 9613, 3rd edition, International Civil Aviation Organization, Montreal, Quebec, Canada

- Lee J, Pullen S, Enge P (2006) Sigma-mean monitoring for the local area augmentation of GPS, Stanford University. *IEEE Trans Aerosp Electron Syst* 42(2):625–635
- Lin CW, Jan SS (2014) Algorithm and configuration availability tool for integrity monitoring test-bed. In: *Proceedings of ION ITM 2014*, Institute of Navigation, Jan 27–29, San Diego, California, pp 223–233
- Normark P-L, Xie G, Akos D, Pullen S, Luo M, Lee J, Enge P, Pervan B (2001) The next generation integrity monitor testbed (IMT) for ground system development and validation testing. In: *Proceedings of ION GPS 2001*, Institute of Navigation, Sept 11–14, Salt Lake City, Utah, pp 1200–1208
- RTCA (2004) Minimum aviation system performance standards for the local area augmentation system (LAAS). RTCA DO-245A, Washington, D.C., Dec 9
- RTCA (2008) Minimum operational performance standards for GPS local area augmentation system airborne equipment. RTCA DO-253C, Washington, D.C., Dec 16
- Shively C, Braff R (2000) An overbound concept for pseudorange error from the LAAS ground facility. In: *Proceedings of ION AM 2000*, Institute of Navigation, June 26–28, San Diego, California, pp 661–671



Shuo-Ju Yeh Shuo-Ju Yeh is a Ph.D. candidate in the Department of Aeronautics and Astronautics at National Cheng Kung University (NCKU), Taiwan. He received his B.S. and M.S. degrees in Aeronautics and Astronautics from NCKU in 2007 and 2009, respectively. His research focuses on the integrity monitoring machine design and implementation for GNSS to support the autonomous air traffic management.



Shau-Shiun Jan Shau-Shiun Jan is an Associate Professor of the Department of Aeronautics and Astronautics, NCKU, Taiwan. He directs the NCKU Communication and Navigation Systems Laboratory (CNSL). His research focuses on GNSS augmentation system design, analysis, and application. He received his Ph.D. degree in Aeronautics and Astronautics from Stanford University in 2003.