Improved User Position Monitor for WAAS

Todd Walter and Juan Blanch
Stanford University

ABSTRACT

The majority of the monitors in the Wide Area Augmentation System (WAAS) [1] focus on errors affecting individual error components. For example, the User Differential Range Error (UDRE) monitor [2] protects against large satellite clock and ephemeris errors. Its purpose is preventing errors larger than roughly four times σ_{UDRE} from being broadcast to the users. The UDRE monitor does not necessarily need to react to errors on the order of two to three times σ_{UDRE}, as such errors are expected to occur occasionally as part of a normal error distribution. Disallowing these smaller errors may limit the overall system availability. Although two sigma errors are not all that uncommon individually, it is very unlikely to have many independent two-sigma errors present at the same time. Such an occurrence could lead to unsafe position errors for the user. Consequently, WAAS has monitors to examine not just the individual component errors, but to also the aggregate effect of the component errors together. Chief among these monitors is the User Position Monitor (UPM) [3].

This paper describes an efficient new algorithm being developed for WAAS that significantly improves upon the prior UPM. This new algorithm is based on the sum of the squares of the normalized errors at each reference station. It has significantly improved the ability to detect error conditions that could be harmful to users. As a consequence, excess conservatism may be removed from other WAAS monitors, as this new UPM is better suited to identify potential threats. As these other monitors are also improved we expect this innovation to lead to higher availability while maintaining equal or better levels of integrity.

INTRODUCTION

The current UPM computes the real-time, WAAS-corrected, position estimates (using the reference station pseudorange measurements) and compares them against the known survey locations. The monitor uses tight thresholds, well below the broadcast protection levels, in order to determine if there may be a threat. Unfortunately, a position calculation only tests a specific combination of errors. A nearby user may observe a slightly different set of satellites or weight them differently due to its local conditions. Thus, while the UPM is generally good at detecting small ranging errors that can map to larger position errors, it is hard for it to quantitatively limit the worst-case user computed position error. As a result, the other monitors in WAAS retain a high level of conservatism to reduce the overall likelihood of two and three sigma errors.

A new UPM is being developed for WAAS that is based on the sum of the squares of the normalized errors at each reference station. As we will show, this chi-square based UPM tests not only the specific set of weights used by each reference station, but effectively evaluates all possible sets of weights including every possible subset. The new monitor only uses a single chi-square metric to perform this evaluation, so the computational cost is not higher than the existing UPM. In subsequent sections, we derive the analysis behind the chi-square UPM and demonstrate its efficacy using real WAAS data. We show that this new monitor rigorously bounds the observed position errors and how this bound may be applied not just at the reference stations, but also for all nearby users. This more effective monitor allows WAAS to reduce conservatism in the individual monitors, allowing the possibility of broadcasting smaller UDREs, leading to better availability.

CURRENT USER POSITIONING MONITOR

The current UPM uses reduced versions of the WAAS error bounds to calculate the reference station position errors and associated protection levels. The protection levels calculated by the UPM are much smaller than those

calculated by the users. The GIVE used by the UPM, $GIVE_{UPM}$, is much less conservative than the GIVE value broadcast to the users. The broadcast version has an inflation factor that depends on the self-consistency of the ionospheric measurements [4]. The $GIVE_{UPM}$ calculation requires greater inconsistency among the measurements in order for it to be inflated. In addition, the broadcast GIVE includes a protection term against the possibility of poorly observed ionospheric disturbances [5]. Ionospheric disturbances are infrequently present over North America and are usually well observed and identified before they affect users. Therefore, the $GIVE_{UPM}$ uses a greatly reduced term for this threat. Finally, $GIVE_{UPM}$ is an internal value and is not quantized. The large steps between the discrete GIVE values often leads to significant increases due to this quantization. In the end, $GIVE_{UPM}$ is typically one third of the value of the broadcast GIVE.

Instead of the broadcast UDRE value, the UPM uses the Kalman filter estimate from the corrections processor, $UDRE_{CP}$. As with the GIVE, this version of the UDRE is unquantized and neglects certain threats such as antenna biases and nominal signal deformation. As a result, it is typically only two-thirds the size of the broadcast UDRE, although it is much smaller when the broadcast UDRE becomes very large (15 m and larger). As a further step, the UPM neglects the $\delta UDRE$ term corresponding to the covariance parameters in Message Type 28 (MT28) [6]. This parameter increases the effective UDRE contribution to the protection level calculation. Its value increases for users farther from the reference station network. It has an average value of two, meaning that it typically doubles the magnitude of the UDRE in the users protection level calculation.

The variance used for satellite i in the UPM is

$$\sigma_{UPM,i}^2 = \sigma_{UDRE\_CP,i}^2 + \sigma_{UIRE\_UPM,i}^2 + \sigma_{CNMP,i}^2 + \sigma_{tropo,i}^2 \qquad (1)$$

where σ_{CNMP} is the estimated code noise and multipath bound for the reference station measurement to that satellite and σ_{tropo} is the standard MOPS tropospheric bounding term [7]. The vertical protection level (VPL) calculated by the UPM corresponds to

$$VPL_{UPM} = 3.5\sqrt{\left(\mathbf{G}^T \cdot \mathbf{W} \cdot \mathbf{G}\right)^{-1}} \qquad (2)$$

Where the smaller value of 3.5 (vs. 5.33 as specified in the MOPS) further reduces the evaluated error bound. The protection levels calculated by the UPM are typically less than a quarter of the value specified in the MOPS [7].

The UPM compares the position error against this reduced protection level and declares a fault if the error exceeds the protection level. If that were to occur, all associated UDREs and GIVEs would be set to Not Monitored and those values would be broadcast to the users. This action would effectively disable all WAAS positioning, as a large number of satellites would no longer be usable for either horizontal or vertical guidance. Fortunately, the UPM has never tripped in the history of WAAS operation. The UPM is seen as a monitor of last resort in the very unlikely event of an unforeseen fault mode.

ANALYSIS OF UPM NEAR TRIPS

This section describes some investigations into previously observed events where the vertical position error (VPE) as calculated by the UPM came close to or exceeded VPL_{UPM}. It was observed that there was a UPM trip on the shadow system for Iqaluit on November 1, 2011 during the period 13:24:11–26 UTC. The shadow system is a parallel hardware chain for evaluating prototype versions of software. In this case, the next intended software release for WAAS was being tested. There were frequent carrier smoothing resets and the residual errors were higher on multiple satellites from roughly hour 11:00 to 17:00. Figure 1 shows two related traces: the top shows the pseudorange residuals for all satellites normalized by the UPM sigma value (1), and the bottom shows the pseudorange residuals normalized by the full broadcast sigma values (except that $\delta UDRE$ is set to 1 as it was not recorded in the diagnostic data set, see Equation 5). As can be seen, the UPM sigmas are much smaller than the broadcast sigmas applied by the user.

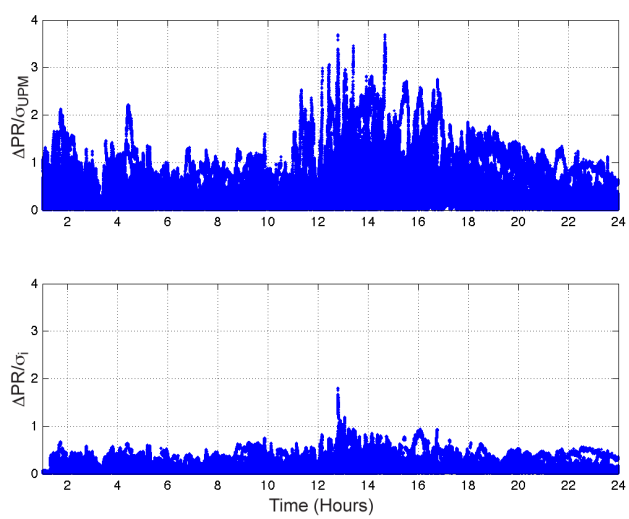

Figure 1. *Normalized pseudorange residuals by the UPM sigma (top) and broadcast sigma (bottom) for Iqaluit on November 1, 2011*

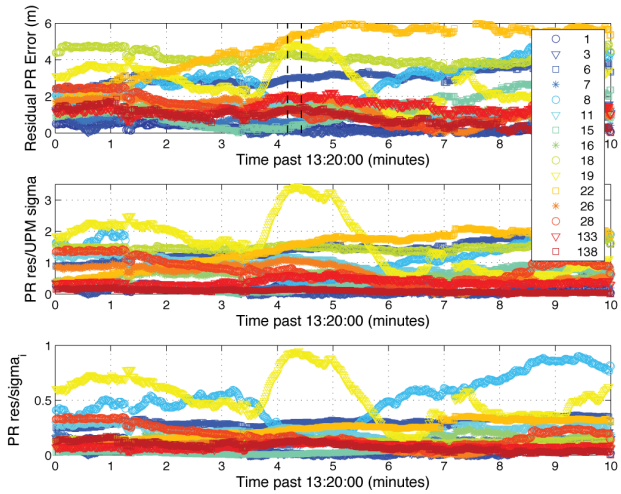

Figure 2. *Pseudorange residuals near the time of trip (indicated by vertical dashed lines in top plot), residuals normalized by UPM sigma (middle), and normalized by broadcast sigma (bottom) for Iqaluit on November 1, 2011.*

The smaller UPM sigmas result in much larger normalized residual errors. Many of the normalized pseudorange residuals corresponding to the UPM sigmas go above 2 and sometimes approach 4. The ratios corresponding to the broadcast UDRE and GIVE values are always below 2, and almost always below 1. The VPE divided by the VPL based on the broadcast values was always well below 0.18, so there never was any actual user threat. PRN 19 had the largest residual at the time, however, removing PRN 19 from the solution still led to UPM VPEs that were more than 50% of VPL_{UPM}.

The apparent UPM threat occurs because the UPM UIVE for PRN 19 is much smaller than the broadcast value and leads to a normalized residual error of 3.46. Further, the UPM VPL was calculated using 3.5 x σ_{vert} rather than 5.33 x σ_{vert} as specified in the MOPS. This deliberate reduction of the VPL is to make the monitor more sensitive. Unfortunately, in this case, the UPM is overly sensitive. The issue arises because the errors increase on some of the lines of sight, likely because of actual undersampling of an ionospheric disturbance, but the UPM GIVE does not get increased. Figure 2 shows the pseudorange residuals in the top plot. As can be seen, the error on PRN 19 increases by more than two meters leading up to the trip time (which occurred between the two vertical black dashed lines). The residual pseudorange error during the trip was 4.88 m while the UPM sigma was 1.41 m and the broadcast sigma was 5.14 m.

The peak pseudorange error ratio occurs at 13:24:23 UTC. The UPM VPE reaches 8.27 m while VPL_{UPM} is 8.04 m. Using the broadcast sigmas to weight the position solution reduces the VPE to 6.51 m and the actual corresponding VPL is 43.37 m. The large error on PRN 19 causes 5.62 m of vertical error by itself. The next largest contributor is PRN 22 and it only adds another 0.68 m to the UPM VPE. Thus, PRN 19 is by far the dominant cause of the error. It has a relatively high elevation angle of 64 degrees at the time.

The next event examined was a near-UPM trip at Chicago on April 23, 2012. Unlike the prior trip, this event is not dominated by a single large error. Instead, it appears that several smaller errors added coherently to create a larger position error. For the majority of the day the VPE was below 1.5 m, but around the times of the near trips it was between 3.5 and 4.5 m, which is certainly elevated, but not threatening. There were two periods of time when the UPM VPE exceeded 70% of VPL_{UPM}. The first occurred at 23:09:57 and the second between 23:53:56 and 23:54:42. Figure 3 shows the residual errors around the time of the two trips. The middle and bottom plots in this figure show the range error normalized by the UPM sigmas and the broadcast sigmas respectively. The first event involves PRN 30 whose error slowly increased while its UPM sigma also increased, but much more slowly. At the time of the peak, the ratio between the error and the UPM sigma had reached 2.2. PRN 30 contributed 2 m to the UPM VPE, which reached 4.5 m at this point. No other satellite contributed more than 0.67 m, but all contributed coherently creating a much larger sum. The other issue was that VPL_{UPM} at this point was 6.48 m. During this period of time, the broadcast

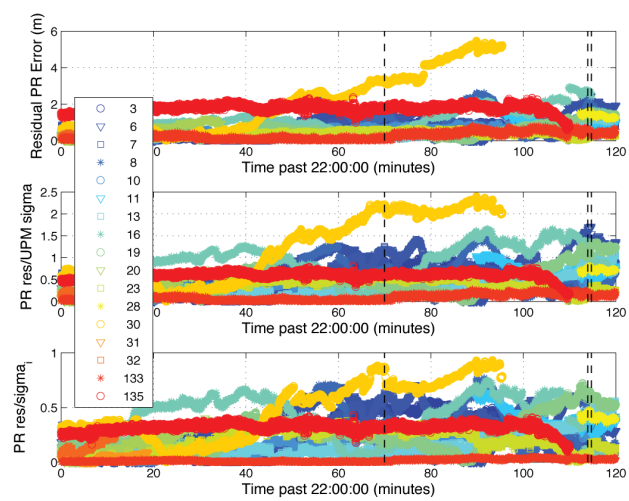

Figure 3. *Pseudorange residuals, UPM sigma values, and broadcast sigma values around the time of near UPM trips for Chicago on April 23, 2012.*

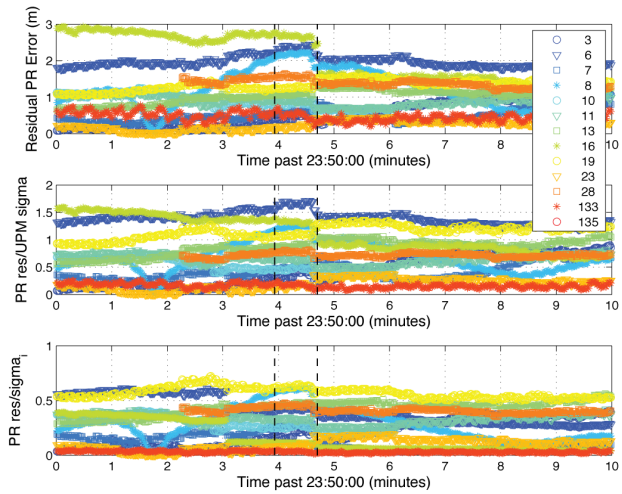

Figure 4. *Pseudorange residuals, UPM sigma values, and broadcast sigma values around the time of the second near UPM trip for Chicago on April 23, 2012*

UIRE was increasing from approximately 4 m to approximately 6 m, while the UPM UIRE remained constant near 2 m. This indicates that the underlying pseudorange error was also likely ionospheric in origin.

The second Chicago near trip also involved no single large satellite error. Figure 4 shows a zoomed in view around this region. PRN 6 has the largest ratio to the UPM sigma, but it is just above 1.5 sigma. At the same time, three other satellites (PRNs 8, 16, & 19) have errors above 1 sigma. Although this does not seem threatening, it is uncommon to have four satellites simultaneously above 1 sigma (because of the conservatism inherent in the sigma values). The largest contributor to VPE UPM was PRN 16, which caused 0.70 m of vertical error. The other satellites all contributed less, but all contributions were in the same direction. The individual errors are not particularly large, three had range errors greater than 2 m (PRNs 6, 8, & 16). However, it also is not common to simultaneously have three errors greater than 2 m. Again the UPM VPE was not particularly large (4.2 m), and the UPM VPL was very small (5.79 m). It is not known why three satellites had errors greater than 2 m while three others had errors larger than 1 m all at the same time nor why they would all add coherently. Perhaps the local ionosphere was more difficult than usual to correctly model.

The final event described is the near-UPM trip observed at Goose Bay on April 12, 2012 that occurred between 19:00:16 and 19:21:35 UTC. This near trip also was caused by multiple small errors rather than a single large error. Figure 5 shows the pseudorange residual errors (top), the errors normalized by UPM sigma (middle) and

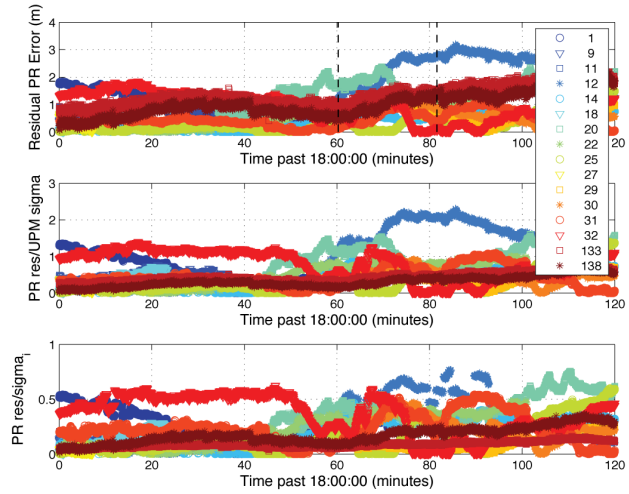

Figure 5. *Pseudorange residuals, UPM sigma values, and broadcast sigma values around the time of the near UPM trip at Goose Bay on April 12, 2012.*

the errors normalized by the broadcast sigma. Although PRNs 12 and 20 have somewhat elevated values, neither appears to be large enough to increase the UPM VPE to within 70% of the UPM VPL.

The peak ratio of VPE_{UPM} / VPL_{UPM} is ~74% and it occurs at 19:10:53 UTC. Table 1 shows the pseudorange residual, UPM sigma, the projection matrix element for the vertical direction ($s_{3,i}$) and the product of $s_{3,i}$ and the residual error, which provides the projection of the pseudorange error into the vertical position, at the time of the peak ratio. As can be seen in the last column, none of the satellite errors lead to a very large position error by themselves. However, the signs of the errors are essentially always opposite to the signs of the $s_{3,i}$ terms.

PRN	PR Residual (m)	UPM Sigma (m)	Vertical Projection Element ($s_{3,i}$)	Vertically Projected Error (m)
1	-1.4166	1.9228	0.2075	-0.2940
12	-2.2723	1.2058	0.3966	-0.9011
14	0.5509	0.9232	-0.6342	-0.3494
20	-2.1949	1.5631	0.2246	-0.4929
22	-1.0228	1.2592	0.3042	-0.3111
25	-0.0383	0.9578	-0.3345	0.0128
29	-1.3282	2.1549	0.2436	-0.3236
30	-1.1750	1.7451	0.2251	-0.2645
31	0.8619	0.9907	-0.6025	-0.5193
32	1.1518	1.1415	-0.0851	-0.0980
133	-1.4699	3.1018	0.0200	-0.0294
138	-0.9836	3.2830	0.0347	-0.0341

Table 1. *Breakdown of error contributions from the satellites in view at Goose Bay at 19:10:53 UTC on April 12, 2012.*

Thus, all of the errors (except for the smallest value) add coherently to create a larger vertical error of -3.6 m. Although the residual errors may be slightly elevated during this period, it appears that the loss of accuracy is caused by the unfortunate alignment of all of the errors. While this is expected to be an uncommon occurrence, it should not be surprising that, given enough data from many WREs, it would be observed on occasion.

In all cases the UPM correctly identifies periods of increased position error. Even though the user position errors are all small compared to the actual protection levels, the VPE can exceed the desired accuracy level of 4 m and in the worst case nearly reaches 8 m. However, user integrity is never threatened. There is no need to have the UPM trip in these events. Further, the UPM VPE is not always accurately reflecting the user's VPE because the difference in the sigmas lead the UPM to give more weight to erroneous satellites that are already indicated as less trustworthy to the users.

It would be better to have the UPM use the broadcast sigmas rather than creating special reduced sigmas that may neglect important and already well known protection terms (e.g. the undersampled threat term).

The LPV-200 procedure [8] [9] has a strong requirement to ensure that vertical position errors larger than ~10 m are rare. This could be interpreted as creating a desire to have the UPM trip before user VPEs would exceed ~10 m. For the specific case where the 8 m VPE was observed, the VPL was above the 35 m vertical alert limit (VAL) required to support LPV-200, so no special action is required in this case. However, we do want to ensure that some action is taken before a >10 m vertical error is likely to affect a user.

EFFECTS OF VISIBILITY/WEIGHTING ON UPM

To see the effect of weighting on the calculated position error, we look at a specific user geometry. This example was created using Stanford's Matlab Algorithm Availability Simulation Tool (MAAST) [10] and was previously described in [11]. In this example, the user has eight satellites in view as shown in Table 2. Figure 6 shows the elevations and azimuths of the satellites along with their PRN values. Table 2 also shows the PRN, elevation, azimuth, and one sigma confidence bound (σ_i). In addition, the fifth column shows the dependence of the vertical error to a pseudorange error on that satellite, s_{3i}. **S** is the projection matrix and is defined as $\mathbf{S} = (\mathbf{G}^T\mathbf{W}\mathbf{G})^{-1}\mathbf{G}^T\mathbf{W}$, where **G** is the geometry matrix and **W** is the weighting matrix, see Appendix J of [7]. This term multiplies the error on the pseudorange to determine the contribution to the vertical error. Thus a 1 m ranging error on PRN 2 would create a positive 59.5 cm vertical error for the user with this combination of satellites and weights. The final column in Table 1 shows the projection matrix values if PRN 8, a low elevation satellite, is not included in the position solution.

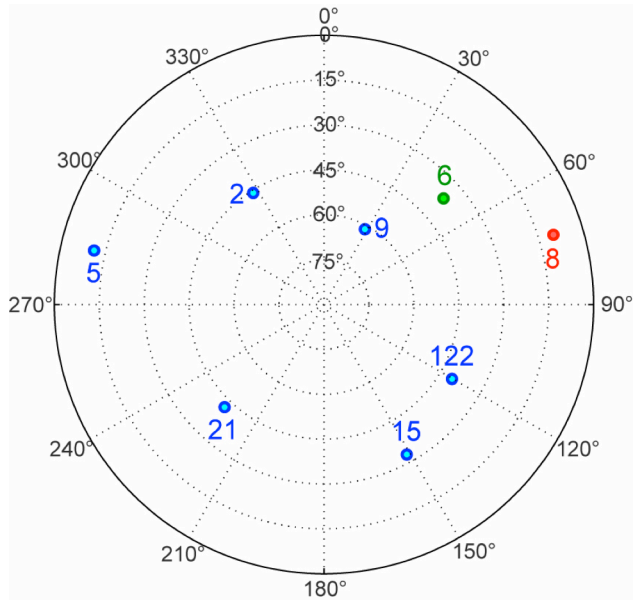

Figure 6. *Satellite elevation and azimuth values for a standard skyplot. PRN 8 is a low elevation satellite that if not included in the solution dramatically changes the influence of PRN 6.*

With the all-in-view solution, the user has a VPL of 33.3 m (HPL = 20.4 m). When PRN 8 is dropped, the VPL increases to 48.6 m (HPL = 20.5 m). Both values are below the 50 m Vertical Alert Limit (VAL) for LPV [8]. Either solution could be used for vertical guidance. Notice that the vertical error dependency changes

PRN	*EL*	*AZ*	σ_i	s_{3i}	s_{3i} without PRN 8
2	45.8°	-32.3°	2.34 m	0.595	0.451
5	11.2°	-76.8°	10.1 m	0.258	0.437
6	36.6°	48.4°	2.32 m	0.162	2.005
8	9.98°	73.0°	3.74 m	1.000	-
9	61.4°	28.5°	2.03 m	-1.928	-3.087
15	32.8°	151.0°	6.89 m	-0.015	0.174
21	42.3°	-136.0°	4.83 m	0.066	-0.003
122	40.6°	120.1°	6.19 m	-0.139	0.022

Table 2. *Vertical projection elements for all-in-view and PRN 8 out geometries.*

dramatically with the loss of PRN 8. In particular, PRN 6, which had little influence over the all-in-view VPE, now has a very strong impact on this subset VPE. Also notice that the other values change as well. PRNs 2, 21, and 122 lose influence while PRNs 5, 6, 9 and 15 become more important. More surprisingly, the influences of PRNs 15, 21, and 122 change sign; therefore, what led to a positive VPE for the all-in-view solution now leads to a negative VPE for this particular subset.

The changes in the s_{3i} values with subset or superset position solutions limit the ability to verify performance exclusively in the position domain. For example, if PRN 6 had a 25 m bias on its pseudorange, it would lead to a vertical error of greater than 50 m with PRN 8 missing, but just over 4 m for the all-in-view solution. This effect limits the ability of the current UPM to mitigate all possible threats. The fact that it is evaluated at all reference stations means that different values of s_{3i} are tested, but it does not guarantee that all threatening combinations will be caught. For this reason, we wanted to create a more effective UPM.

CHI-SQUARED UPM ALGORITHM

We wish to evaluate whether or not there are several larger than normal errors regardless of their sign or corresponding s_{3i} parameters. We further wish to quantitatively bound the positioning error any user may experience with the set of errors. We begin by looking at the positioning error and for simplicity we will begin with the vertical positioning error (VPE):

$$VPE = \sum_{i=1}^{n} s_{3,i} \times \varepsilon_i \tag{3}$$

where ε_i is the pseudorange error on satellite i.

This can be rewritten as:

$$VPE = \left| \sum_{i=1}^{n} s_{3,i} \times \frac{\varepsilon_i}{\sigma_i} \times \sigma_i \right| \tag{4}$$

where σ_i is the overbounding sigma according to the MOPS [7]:

$$\sigma_i^2 = \sigma_{flt,i}^2 + \sigma_{UIRE,i}^2 + \sigma_{air,i}^2 + \sigma_{tropo,i}^2 \tag{5}$$

When calculating σ_{flt} in this new UPM we will continue to neglect the effects of $\delta UDRE$ from MT28 (as is done in the current UPM). This adds conservatism to the calculation. Also it is consistent with the prior UPM algorithm and will be simpler to implement.

From the Cauchy-Schwarz inequality, we obtain:

$$VPE \le \sqrt{\sum_{i=1}^{n} s_{3,i}^2 \times \sigma_i^2} \times \sqrt{\sum_{i=1}^{n} \frac{\varepsilon_i^2}{\sigma_i^2}} \tag{6}$$

Then, because:

$$VPL = K_V \sqrt{\sum_{i=1}^{n} s_{3,i}^2 \times \sigma_i^2} \tag{7}$$

we can see that:

$$\frac{VPE}{VPL} \le \frac{1}{K_V} \sqrt{\sum_{i=1}^{n} \frac{\varepsilon_i^2}{\sigma_i^2}} \tag{8}$$

This yields a useful upper bound on the VPE relative to the VPL. The ratio of the error to the protection level is bounded by the square root of the sum of the squares of the normalized pseudorange errors (i.e. the square root of the chi-square metric) divided by the K factor. This upper bound is independent of geometry and weighting; the s_{3i} parameters can take on arbitrary values, they do not affect the right hand side of the equation. Thus, the upper bound holds for all subset geometries as well (since some of the s_{3i} could be set to zero). Using the chi-square metric is far more powerful than merely comparing the position error against the protection level which only checks one specific geometry and one specific set of weights.

Note that the pseudorange errors may contain a common clock term that will not affect positioning error. It is advisable to remove this common mode component before computing the chi-square value. In fact, when computing either the VPE or the HPE, additional error components that only affect the orthogonal direction may also be removed. That is, we can compute separate, specific chi-square values for the vertical and horizontal directions. This is advantageous because the HPL and VPL use different K factors and we can match them with corresponding chi-square evaluations. The vertical chi-square, χ_{vert}^2, is given by:

$$\chi_{vert}^2 = \boldsymbol{\varepsilon}^T \cdot \left(\mathbf{W} - \mathbf{W} \cdot \mathbf{G}_V \cdot \left(\mathbf{G}_V^T \cdot \mathbf{W} \cdot \mathbf{G}_V \right)^{-1} \cdot \mathbf{G}_V^T \cdot \mathbf{W} \right) \cdot \boldsymbol{\varepsilon} \tag{9}$$

where $\boldsymbol{\varepsilon}$ is the vector of pseudorange residuals, the weighting matrix is given by:

$$\mathbf{W}^{-1} = \begin{bmatrix} \sigma_1^2 & & & 0 \\ & \sigma_2^2 & & \\ & & \ddots & \\ 0 & & & \sigma_n^2 \end{bmatrix} \tag{10}$$

and $\boldsymbol{G_V}$ is given by:

$$\mathbf{G}_V = \begin{bmatrix} G_{1,1} & G_{1,2} & G_{1,4} \\ G_{2,1} & G_{2,2} & G_{2,4} \\ \vdots & \vdots & \vdots \\ G_{n,1} & G_{n,2} & G_{n,4} \end{bmatrix} \tag{11}$$

The second term in the parentheses in (9) removes the common clock term as well as error components that do not affect the vertical upper bound. Proof for this formula is provided in the appendix to this paper.

The horizontal chi-square χ^2_{horz}, is given by:

$$\chi^2_{horz} = \boldsymbol{\varepsilon}^T \cdot \left(\mathbf{W} - \mathbf{W} \cdot \mathbf{G}_H \cdot \left(\mathbf{G}_H^T \cdot \mathbf{W} \cdot \mathbf{G}_H \right)^{-1} \cdot \mathbf{G}_H^T \cdot \mathbf{W} \right) \cdot \boldsymbol{\varepsilon} \tag{12}$$

where $\boldsymbol{G_H}$ is given by:

$$\mathbf{G}_H = \begin{bmatrix} G_{1,3} & G_{1,4} \\ G_{2,3} & G_{2,4} \\ \vdots & \vdots \\ G_{n,3} & G_{n,4} \end{bmatrix} \tag{13}$$

The second term in the parentheses in (12) removes the common clock term as well as error components that do not affect the horizontal upper bound.

The chi-squared UPM monitor can now compare these chi-square values to the corresponding K factors, to ensure that the positioning errors will remain below the protection levels for all users. Thus, we want to ensure that:

$$\chi^2_{vert} \le K_V^2 \tag{14}$$

and

$$\chi^2_{horz} \le K_{H,PA}^2 \tag{15}$$

where $K_V = 5.33$ and $K_{H,PA} = 6.0$ as specified in [7]. In reality, we will use threshold values somewhat below these upper bounds to increase conservatism.

CHI-SQUARED UPM PERFORMANCE

The chi-square metrics from the previous section were computed for several prior days where elevated values were observed with the current UPM monitor. These days included November 1-2, 2011, April 12 and April 23, 2012 previously described. The metrics were computed for each WAAS reference station every second for the twelve days that had the highest legacy UPM values. Figure 7 shows the χ^2_{vert} and χ^2_{horz} values for Iqualit on November 1, 2011. These values can be compared to the results in Figure 1. Notice that there is almost no effect on the vertical measure, while the horizontal measure shows a distinct spike. However, this spike barely exceeds six, while the threshold for concern could be close to $K^2_{H,PA}$ or 36. Thus, the chi-square UPM correctly recognizes that while the errors are worse than normal, they do not threaten integrity, nor is the monitor particularly close to creating a false alert.

Figures 8 and 9 show results for all days at all stations. In all cases it was found that:

$$\frac{VPE}{VPL} \le \frac{\sqrt{\chi^2_{vert}}}{K_V} \quad \text{and} \quad \frac{HPE}{HPL} \le \frac{\sqrt{\chi^2_{horz}}}{K_{H,PA}} \tag{16}$$

It is perhaps surprising that the calculated ratio was occasionally right at the upper bound, indicating that the s_{3i} parameters were near worst-case, yielding the

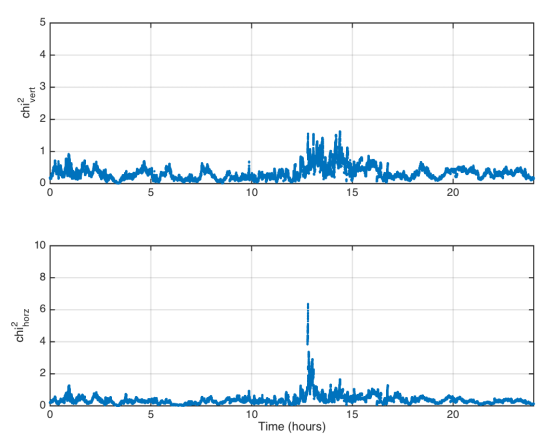

Figure 7. *Chi-squared UPM sigma metric values corresponding to vertical (top) and horizontal (bottom) for Iqaluit on November 1, 2011.*

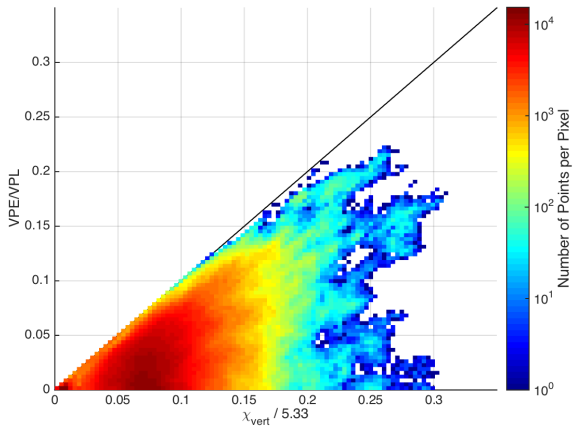

Figure 8. *Two-dimensional histogram of VPE/VPL versus* $\sqrt{\chi^2_{vert}}$ / 5.33 *for all WAAS reference stations on November 1-2, 2011.*

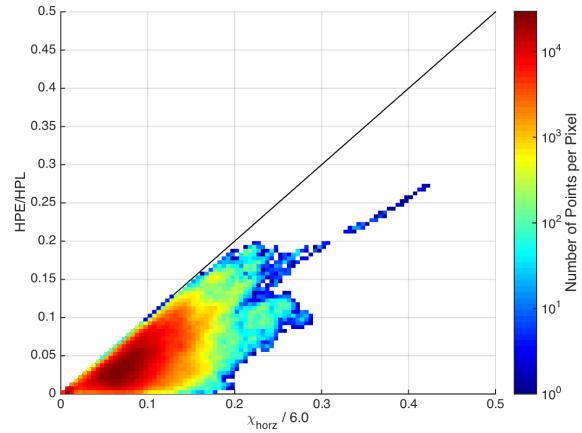

Figure 9. *Two-dimensional histogram of HPE/HPL versus* $\sqrt{\chi^2_{horz}}$ / 6.0 *for all WAAS reference stations on November 1-2, 2011.*

maximum possible error. However, it is reassuring that despite these worst-case positioning errors, the chi-square value did always provide an upper bound. We further see, that in all cases, the chi-square UPM is never particularly close to tripping. The square root of the chi-square values were below 35% of threshold for vertical and below 45% of threshold for horizontal. This is a substantial improvement over the current UPM which exceeded 70% of its threshold for many of these days and over 90% at Iqaluit on November 1, 2011. Notice that the Iqaluit event creates a noticeable spike in Figure 9, but is still well below causing an alert. The horizontal chi-square value indicates that even for the worst-case user would not have an HPE that exceeded 28% of the HPL

CONCLUSIONS

A new user position monitor is proposed that more effectively tests for simultaneously elevated error values across all corrected pseudoranges. This chi-square UPM is able to evaluate not just the threat for one specific user geometry and set of weights, but for all possible combinations of the errors. This new UPM is therefore more effective at identifying threats from multiple increased errors that individually may not appear threatening.

We have further shown that the risk of false alert for this new monitor is decreased compared to the existing monitor. This is because the current monitor uses reduced values of the UDREs and GIVEs to perform its evaluations. These reduced values neglect terms that already protect the user from potential threats. The current monitor may then incorrectly perceive a threat that has already been handled by the other monitors. The new UPM uses the broadcast UDRE and GIVE values together with with the observed errors. We have shown that the new chi-square metrics create upper bounds on the ratio of the potential position errors relative to the protection levels, thus fully protecting all users.

Because the new UPM is very effective at catching any potentially harmful error, it may be possible to eliminate some conservatism in the UDRE (and perhaps the GIVE) monitor. Any such reduction could lead directly to increased system availability while fully maintaining integrity.

ACKNOWLEDGMENTS

The authors would like to gratefully acknowledge the FAA Satellite Product Team for supporting this work under MOA contract number DTFAWA-15-A-80019. The opinions expressed in this paper are the authors' and this paper does not represent a government position on the future development of WAAS.

REFERENCES

[1] Lawrence, et al., "Wide Area Augmentation System (WAAS) - Program Status," ION GNSS 2007, Fort Worth, TX Sep. 2007 (ION, Virginia 2007) 892 – 899

[2] Wu, T. and Peck, Stephen, "An Analysis of Satellite Integrity Monitoring Improvement for WAAS," Proceedings of the 15th International Technical Meeting

of the Satellite Division of The Institute of Navigation (ION GPS 2002), Portland, OR, September 2002, pp. 756-765.

[3] Grewal, M.S., Habereder, H., and Schempp, T.R., "Overview of the WAAS Integrity Design," Proceedings of the 16th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS/GNSS 2003), Portland, OR, September 2003, pp. 2750-2759.

[4] Walter, T. et al., "Robust Detection of Ionospheric Irregularities," NAVIGATION, Journal of The Institute of Navigation, Vol. 48, No. 2, Summer 2001, pp. 89-100.

[5] Walter, T., Rajagopal, S., Datta-Barua, S., and Blanch, J., "Protecting Against Unsampled Ionospheric Threats," Proc. Beacon Satellite Symposium, Trieste, Italy Oct. 2004.

[6] Walter, T., Hansen, A., and Enge, P., "Message Type 28," Proceedings of the 2001 National Technical Meeting of The Institute of Navigation, Long Beach, CA, January 2001, pp. 522-532.

[7] Minimum Operational Performance Standards for Global Positioning System/Wide Area Augmentation System Airborne Equipment, RTCA DO229D Change 1, Feb. 2013.

[8] Cabler, H., and DeCleene, B., "LPV: New, Improved WAAS Instrument Approach," ION GPS 2002, Portland, OR 24-27 Sep. 2002 1013 – 1021.

[9] International Standards and Recommended Practices (SARPS), Annex 10 – Aeronautical Telecommunications, Vol. 1I – Radio Navigation Aids, 6th edn., ICAO, July 2006.

[10] Jan, S.S., Chan, W. Walter, T., and Enge, P., "Matlab Simulation Toolset for SBAS Availability Analysis," Proceedings of the 14th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 2001), Salt Lake City, UT, September 2001, pp. 2366-2375.

[11] Walter, T., Enge, P., and DeCleene, B., "Integrity Lessons from the WAAS Integrity Performance Panel (WIPP)," Proceedings of the 2003 National Technical Meeting of The Institute of Navigation, Anaheim, CA, January 2003, pp. 183-194.

APPENDIX: UPPER BOUND ON THE NORMALIZED POSITION ERROR BASED ON THE CHI-SQUARE OF THE MEASUREMENT ERRORS

ASSUMPTIONS

We consider the system of equations:

$$\mathbf{y} = \mathbf{G} \cdot \mathbf{x} + \boldsymbol{\varepsilon}$$

We consider the set of projection matrices **S** which produce an unbiased estimate of **x**, so that:

$$\mathbf{S} \cdot \mathbf{G} = \mathbf{I}$$

We consider a given row of **S**, for example the first one, we have:

$$\mathbf{s}_1 \cdot \mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

We define $\bar{\mathbf{G}}_j$ as the matrix obtained by removing the jth column of **G**. We have for our example of $j = 1$:

$$\mathbf{s}_1 \cdot \bar{\mathbf{G}}_1 = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}$$

This relation holds true for any j even if j is a set containing more than one row (and removing the corresponding columns from **G**).

RESULT

Let **W** be a positive definite weighting matrix and $\boldsymbol{\varepsilon}$ a vector of errors. For any $\mathbf{s}_j$ constrained by the equation above, it can be shown that we have:

$$\left|\mathbf{s}_j^T \cdot \boldsymbol{\varepsilon}\right| \le \sqrt{\mathbf{s}_j^T \cdot \mathbf{W}^{-1} \cdot \mathbf{s}_j} \times \sqrt{\boldsymbol{\varepsilon}^T \cdot \left(\mathbf{W} - \mathbf{W} \cdot \bar{\mathbf{G}}_j \cdot \left(\bar{\mathbf{G}}_j^T \cdot \mathbf{W} \cdot \bar{\mathbf{G}}_j\right)^{-1} \cdot \bar{\mathbf{G}}_j^T \cdot \mathbf{W}\right) \cdot \boldsymbol{\varepsilon}}$$

This result links the magnitude of the estimation error with an error bound computed assuming a zero mean Gaussian distribution.

PROOF

For any **x** we have:

$$\begin{aligned}\mathbf{s}_j^T \cdot \boldsymbol{\varepsilon} &= \mathbf{s}_j^T \cdot \left(\boldsymbol{\varepsilon} + \bar{\mathbf{G}}_j \cdot \mathbf{x}\right) \\ &= \mathbf{s}_j^T \mathbf{W}^{-\frac{1}{2}} \cdot \mathbf{W}^{\frac{1}{2}} \left(\boldsymbol{\varepsilon} + \bar{\mathbf{G}}_j \cdot \mathbf{x}\right)\end{aligned}$$

We apply the Cauchy-Schwarz inequality to the scalar product of these two vectors:

$$\left|\mathbf{s}_j^T \cdot \boldsymbol{\varepsilon}\right| \le \sqrt{\mathbf{s}_j^T \cdot \mathbf{W}^{-1} \cdot \mathbf{s}_j} \sqrt{\left(\boldsymbol{\varepsilon} + \bar{\mathbf{G}}_j \cdot \mathbf{x}\right)^T \mathbf{W} \cdot \left(\boldsymbol{\varepsilon} + \bar{\mathbf{G}}_j \cdot \mathbf{x}\right)}$$

This inequality is valid for any **x**. It is in particular true for the **x** that realizes the minimum of the right term. This minimum is achieved by the least squares estimate of **x**:

$$\mathbf{x} = -\left(\bar{\mathbf{G}}_j \cdot \mathbf{W} \cdot \bar{\mathbf{G}}_j\right)^{-1} \cdot \bar{\mathbf{G}}_j \cdot \mathbf{W} \cdot \boldsymbol{\varepsilon}$$

By replacing this expression in the inequality, we get:

$$\begin{aligned}\left|\mathbf{s}_j^T \cdot \boldsymbol{\varepsilon}\right| \le \sqrt{\mathbf{s}_j^T \cdot \mathbf{W}^{-1} \cdot \mathbf{s}_j} \times \\ \sqrt{\boldsymbol{\varepsilon}^T \cdot \left(\mathbf{W} - \mathbf{W} \cdot \bar{\mathbf{G}}_j \cdot \left(\bar{\mathbf{G}}_j^T \cdot \mathbf{W} \cdot \bar{\mathbf{G}}_j\right)^{-1} \cdot \bar{\mathbf{G}}_j^T \cdot \mathbf{W}\right) \cdot \boldsymbol{\varepsilon}}\end{aligned}$$

The equality is obtained when:

$$\mathbf{s}_j \propto \left(\mathbf{W} - \mathbf{W} \cdot \bar{\mathbf{G}}_j \cdot \left(\bar{\mathbf{G}}_j^T \cdot \mathbf{W} \cdot \bar{\mathbf{G}}_j\right)^{-1} \cdot \bar{\mathbf{G}}_j^T \cdot \mathbf{W}\right) \cdot \boldsymbol{\varepsilon}$$