OPTIMAL POSITIONING FOR ADVANCED RAIM

Juan Blanch, Todd Walter, Per Enge.

Stanford University

ABSTRACT

In the next decade, several important developments will have a major impact on civil aviation: the deployment of Galileo and Compass, the modernization of GPS, and the fact that all these core constellations will broadcast navigation signals in two distinct frequencies that fall in the L1/E1 and L5/E5, which fall in the Aeronautical Radio Navigation Satellite Service (ARNSS) space. As a consequence, even under conservative assumptions, the ranging sources will more than triple. In addition, the ionospheric delay will be estimated and removed by receivers using dual frequency. These developments can be exploited in all satellite navigation systems for aircraft. In particular, the increased redundancy and accuracy could dramatically improve the performance of Receiver Autonomous Integrity Monitoring (RAIM). In Advanced RAIM (ARAIM), they could help enable worldwide vertical guidance. For horizontal RAIM, it could help achieve worldwide coverage of lateral navigation down to fractions of a mile. It is therefore useful to evaluate which RAIM algorithms offer the best performance.

As shown in [1], [2] the performance of RAIM can be improved by optimally allocating the integrity budget and the continuity budget across the fault modes – in order to minimize the Protection Levels. The approach in [1] and [2] assumes that the position is centered at the most accurate all-in-view position. This approach guarantees the best accuracy under nominal conditions. However, it is possible to reduce the Protection Levels by choosing a position solution that minimizes it – therefore degrading accuracy. This approach has been exploited in NIORAIM within the framework of slope-based RAIM, where single faults are assumed [3] and accuracy constraints are not considered. It has also been exploited in the case of a simplified threat model where only constellation faults are assumed in [4].

The contribution of this paper consists on simultaneously optimizing the integrity allocation and the position solution, in taking into account additional constraints when generating the position solution - for example the accuracy, but not only -, and in doing it for any threat model (in particular multiple faults). This is done by casting the problem as a convex optimization problem. We will evaluate this algorithm by comparing its performance with algorithms where the position solution is not optimized, and showing how it could help achieve worldwide coverage of vertical guidance (LPV-200) under different sets of assumptions.

INTRODUCTION

Advanced RAIM (ARAIM) [5] performance is a function of the threat model, the constellation strength, and the user algorithm. The objective of this paper is to present a potential improvement on the ARAIM user algorithms within the solution separation ARAIM algorithms presented in [1] and [2], and more generally, on any form of RAIM (like horizontal RAIM). The objective of the user algorithm is to maximize availability while meeting the integrity and continuity requirements. We will first give an overview of ARAIM and a summary of the user algorithm, which is based on solution separation [1], [2]. Then, we will show how to decrease the Protection Levels by adjusting the all-in-view position, while staying within the same framework. Finally, we will evaluate the performance of the algorithm for vertical guidance.

ADVANCED RAIM OVERVIEW

In this section we give a short description of the ARAIM concept. For more details on this description, the reader should refer to [5].

Advanced RAIM definition

The provision of integrity for vertical guidance using mainly airborne monitors is referred to as Advanced RAIM to distinguish it from RAIM as it is used today for horizontal navigation [7], [8]. The increased level of integrity for vertical guidance requires a higher level of scrutiny in the generation of the Protection Levels. In turn, this results in expanded threat models, airborne algorithms that can handle the expanded threat models, and a ground monitoring system that can update the assumptions used by the airborne algorithms.

Requirements for vertical guidance (LPV-200)

At least six conditions must be met for LPV-200: the Vertical Protection Level (VPL) must be below 35 m, the Horizontal Protection Level (HPL) must be below 40 m, the Effective Monitor Threshold (EMT) must be below 15 m, the false alarm rate must be below 8 x 10^{-6} per approach, the 95% vertical accuracy must be below 4 m, and the 10^{-7} bound on the fault free vertical error must be below 10 m. A definition of these different figures of merit and guidance on how to compute them is given in [5]. With the above thresholds, whenever the VPL is met, the HPL is almost always met; in addition, the objective of minimizing the EMT is almost the same as the objective of minimizing the VPL; finally, the two last requirements have been determined to be formally very similar. For these reasons, in this paper we will focus on two of the requirements: the VPL and the accuracy. As will be seen, the false alarm rate requirement is taken into account in the VPL calculation.

Nominal error models

As discussed in [5] the different requirements target different probability levels and different levels of hazard severity. For this reason, there are at least two different pseudorange error models. One error model, which will be labeled the integrity nominal error model, is applied for the requirements that are in the Hazardous category. This error model is only used in the PL terms that guarantee the integrity of the error bound. The other error model, which will be labeled the accuracy nominal error model, is applied in the requirements that are in the Major category (which require less scrutiny). This error model is used to compute the terms in the PL that only affect continuity (the false alert rate), the EMT requirement, and the accuracy. Both errors are characterized by a Gaussian overbound with a maximum nominal bias. For a given geometry, the covariance of the measurements will be designated by C_{int} for the integrity error model and by C_{acc} for the accuracy error model. Both covariances are diagonal, and the formulas to compute each term is given in [1]. The maximum biases are designated by b_{int} and b_{acc} respectively. However, within this paper, we will assume b_{acc} to be zero.

Failure modes

In this paragraph we generalize the description of the failure modes beyond what was described in [2] and [5], so that known correlations between satellite faults can be exploited. In the fault free case, we have:

$$y = Gx + n \quad (1.1)$$

where y are the pseudorange measurements, G is the geometry matrix, x is the receiver location and clock offsets (one for each constellation), and n is the nominal noise (which is characterized by C_{int} and b_{int} for integrity purposes and by C_{acc} and b_{acc} for accuracy purposes as discussed above).

We define a fault mode as the random introduction of a new state $x_{fault,i}$ with an observation matrix $A_{fault,i}$. That is, with a given a priori probability $P_{ap,i}$ the measurement model is given by:

$$y = Gx + A_{fault,i}x_{fault,i} + n \quad (1.2)$$

This description of a fault includes all subset failures considered in [2] and [5]. For example, for a single satellite failure, $A_{fault,i}$ is a row vector with all zeros except for the satellite assumed to be affected, and $x_{fault,i}$ is the magnitude of the fault.

SOLUTION SEPARATION ARAIM ALGORITHM WITH OPTIMAL INTEGRITY ALLOCATION

Solution separation ARAIM algorithm

The solution separation ARAIM algorithm has been described in [1] for single satellite faults, and in [2] for a generalization to multiple failures. Here we describe the algorithm using the characterization of faults described above. The idea of solution separation ARAIM is to create a position error bound for each of the fault modes by computing a position solution unaffected by the fault and computing an error bound around this solution, and accounting for the difference between the all-in-view position solution and the fault tolerant position. In the description here, we will focus on the vertical coordinate (third coordinate in the East-North-Up coordinate system).

First we compute the all-in-view position:

$$\hat{x}_3^{(0)} = s^{(0)T} y$$

The coefficients $s^{(0)}$ can be computed using the least squares estimate corresponding to Equation (1.1) either using the covariance C_{acc} or C_{int}. Then, the position estimates corresponding to each of the fault modes i are computed as:

$$\hat{x}_3^{(i)} = s^{(i)T} y$$

The coefficients $s^{(i)}$ are computed using the least squares estimate corresponding to Equation (1.2) using the covariance C_{int}. In the case of a single (or multiple) satellite fault, this corresponds to the least square estimates computed excluding the satellites that might be faulty. For each fault mode we compute a threshold T_i as:

$$T_i = K_{fa,i}\,\sigma_{ss,i} + \left|\left(s^{(0)} - s^{(i)}\right)^T\right| b_{acc} \quad (1.3)$$

Where:

$$\sigma_{ss,i} = \sqrt{\left(s^{(0)} - s^{(i)}\right)^T C_{acc}\left(s^{(0)} - s^{(i)}\right)}$$

and $K_{fa,i}$ is related to the false alert rate and will be discussed below. The absolute value in (1.3) is taken element-wise. Then we check that:

$$\left|\hat{x}_3^{(i)} - \hat{x}_3^{(0)}\right| \le T_i$$

If all the tests pass, then the VPL is obtained by taking the maximum across the VPL_i, where VPL_i protects the user against each fault mode. VPL_i is defined by:

$$VPL_i = T_i + K_{md,i}\sigma_i + B_i$$

where:

$$\sigma_i = \sqrt{s^{(i)T} C_{\text{int}} s^{(i)}}$$

$$B_i = \left|s^{(i)T}\right| b_{\text{int}}$$

The factors $K_{md,i}$ must meet the following condition (P_{HMI} is the integrity budget):

$$\sum_{i=0}^{N_{faults}} 2P_{ap,i}\,Q\left(-K_{md,i}\right) = P_{HMI}$$

Similarly, the factors $K_{fa,i}$ discussed above must meet the continuity requirement (P_{fa} is the continuity budget):

$$\sum_{i=0}^{N_{faults}} 2Q(-K_{fa,i}) = P_{fa}$$

Note that we include the all-in-view case in the above equation. An simple choice for both $K_{md,i}$, and $K_{fa,i}$ is to allocate an equal allocation to each mode, both for the integrity and the continuity. The final VPL is given by:

$$VPL = \max_i VPL_i$$

Optimizations

It is possible to lower the VPL by optimizing the allocations of the continuity and the integrity, as it is explained in [2]. The optimization problem that is solved is the following:

$$\text{Minimize } \max_i K_{fa,i}\,\sigma_{ss,i} + \left|\left(s^{(0)} - s^{(i)}\right)^T\right| b_{acc} + K_{md,i}\sigma_i + B_i$$

$$\text{Subject to } \sum_{i=0}^{N_{faults}} 2P_{ap,i}\,Q\left(-K_{md,i}\right) = P_{HMI}$$

$$\sum_{i=0}^{N_{faults}} 2Q(-K_{fa,i}) = P_{fa}$$

Implementation of SS ARAIM with optimal integrity allocation

As shown in [2], most of the benefits of the optimization derive from the integrity allocation. As a consequence, a much simpler algorithm can be used which allocates the continuity as:

$$K_{fa,i} = -Q^{-1}\left(\frac{P_{fa}}{2\left(N_{faults}+1\right)}\right)$$

The VPL corresponding to the optimal allocation of the integrity is the solution to the equation:

$$\sum_{i=0}^{N_{faults}} 2P_{ap,i}\,Q\left(\frac{T_i + B_i - VPL}{\sigma_i}\right) = P_{HMI} \quad (1.4)$$

This equation can be solved iteratively using a simple interval half-interval search algorithm.

Choice of baseline ARAIM algorithm

The baseline SS ARAIM algorithm used here will be the one corresponding to the equal distribution of the continuity budget and the optimal distribution of the integrity budget, that is, the VPL is determined through Equation (1.4). There is another choice to make, and that is how to choose the all-in-view solution. For the baseline algorithm we take the all in view solution for which the accuracy is optimal, that is:

$$s^{(0)T} = e_3^T\left(G^T C_{acc}^{-1} G\right)^{-1} G^T C_{acc}^{-1} \quad (1.5)$$

$$e_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix}^T$$

Accuracy

The standard deviation of the position error using the accuracy error model is given by:

$$\sigma_{accuracy} = \sqrt{s^{(0)T} C_{acc} s^{(0)}}$$

In the case of the coefficients computed with Equation (1.5), the resulting standard deviation is:

$$\sigma_{best\ accuracy} = \sqrt{\left(G^T C_{acc}^{-1} G\right)^{-1}_{3,3}}$$

OPTIMIZING THE ALL IN VIEW POSITION

The previous sections have outlined the baseline algorithm. In this section we describe the contribution of this work. We have seen how it is possible to improve the VPL by carefully allocating the continuity budget and the integrity budget. Here we describe how to further reduce it by adjusting the all-in-view position coefficients. It was mentioned above that one could choose C_{int} or C_{acc} to compute the all-in-view coefficients. However, the computation of these coefficients is not limited to these two choices. The only constraints they need to meet are:

$$s^T G = e_3^T$$

Since there is a accuracy requirement, which we will note $\sigma^2_{required\ accuracy}$, we also need to have:

$$s^T C_{acc} s \leq \sigma^2_{required\ accuracy}$$

In the rest of the paper, we are going to assume (mostly to lighten the notations) that b_{acc} is zero. Also, we are not going to optimize the continuity allocation so that all $K_{fa,i}$ are equal. This is not a necessary step, but it lightens the notations and has very little impact on the final result.

Problem statement

Finding the position that minimizes the VPL while meeting the accuracy requirement is solving the following problem:

$$\begin{aligned}
&\underset{s, K_{md,i}}{\text{minimize}} \quad \max_i K_{fa} \sqrt{\left(s - s^{(i)}\right)^T C_{acc} \left(s - s^{(i)}\right)} + K_{md,i}\sigma_i + B_i \\
&\text{subject to} \quad s^T G = e_3^T \\
&\sum_i 2 p_{ap,i} Q\left(K_{md,i}\right) = P_{HMI} \\
&s^T C_{acc} s \leq \sigma^2_{required\ accuracy}
\end{aligned}$$

In the next paragraphs, we describe a method to solve this optimization problem.

Change of variables

First, a change of variables is performed to remove the first constraint. Be $s^{(0)}$ the all in view solution defined by Equation (1.5) , and be Q a matrix whose columns are basis of the null space of G^T. For any admissible s there exists z such that:

$$s = s^{(0)} + Qz$$

The problem is then equivalent to:

$$\begin{aligned}
&\underset{z, K_{md,i}}{\text{minimize}} \quad \max_i K_{fa} \sqrt{\left(Qz - s^{(i)} - s^{(0)}\right)^T C_{acc} \left(Qz - s^{(i)} - s^{(0)}\right)} + K_{md,i}\sigma_i \\
&\text{subject to} \quad \sum_i 2 p_{ap,i} Q\left(K_{md,i}\right) = P_{HMI} \\
&\left(s^{(0)} + Qz\right)^T C_{acc} \left(s^{(0)} + Qz\right) \leq \sigma^2_{required\ accuracy}
\end{aligned}$$

$s^{(i)} - s^{(0)}$ is in the null space of G^T, so there exists z_i such that:

$$s^{(i)} - s^{(0)} = Qz_i$$

Second, a change of variables is performed to normalize and simplify the term under the square root. For this, we determine R such that:

$$Q^T K_{fa}^2 C_{acc} Q = R^T R$$

This can be done by computing the Choleski factorization. We do the change of variables (note that x is no longer the position):

$$x = Rz \quad a_i = Rz_i$$

In addition, we have:

$$\begin{aligned}
s^{(0)T} C_{acc} Qz &= e_3^T \left(G^T C_{acc}^{-1} G\right)^{-1} G^T C_{acc}^{-1} C_{acc} Qz \\
&= e_3^T \left(G^T C_{acc}^{-1} G\right)^{-1} G^T Qz = 0
\end{aligned}$$

since Qz is in the null space of G^T. As a consequence:

$$\left(s^{(0)} + Qz\right)^T C_{acc} \left(s^{(0)} + Qz\right) = s^{(0)T} C_{acc} s^{(0)} + z^T Q^T C_{acc} Qz$$

The problem is then written:

$$\begin{aligned}&\underset{x,K_{md,i}}{\text{minimize}} \quad \max_i \sqrt{\left(x-a_i\right)^T\left(x-a_i\right)}+K_{md,i}\sigma_i+B_i\\&\text{subject to } \sum_i 2p_{ap,i}Q\left(K_{md,i}\right)=P_{HMI}\\&x^Tx\le K_{fa}^2\left(\sigma^2_{required\ accuracy}-\sigma^2_{best\ accuracy}\right)\end{aligned}\tag{1.6}$$

Geometric interpretation

To illustrate the problem, in this paragraph we assume that the integrity allocations are fixed. We also label:

$$K_{md,i}\sigma_i+B_i=R_i$$
$$K_{fa}\sqrt{\sigma^2_{required\ accuracy}-\sigma^2_{best\ accuracy}}=R_{accuracy}$$

The problem can be then written:

$$\begin{aligned}&\text{minimize} \quad \max\left|x-a_i\right|+R_i\\&\text{subject to } \left|x\right|\le R_{accuracy}\end{aligned}$$

This problem corresponds to finding the smaller circle containing the circles of center a_i and radius R_i whose center is within a circle centered at the origin and radius $R_{accuracy}$.

If we go back to the geometric interpretation, then for each possible center *VPL(x)* is the minimum radius containing all the circles.

Finding the optimal x

We go back to the original problem (Equation (1.6)). At the optimal point, like in [2], all the terms that need to be minimized are equal, so that the problem becomes:

$$\begin{aligned}&\underset{x,K_{md,i}}{\text{minimize}} \quad VPL\\&\text{subject to } VPL=\sqrt{\left(x-a_i\right)^T\left(x-a_i\right)}+K_{md,i}\sigma_i+B_i\\&\sum_i 2p_iQ\left(K_{md,i}\right)=PHMI\\&x^Tx\le K_{fa}^2\left(\sigma^2_{required\ accuracy}-\sigma^2_{best\ accuracy}\right)\end{aligned}$$

We now define the function *VPL(x)* as the solution of the equation:

$$\sum_i 2p_{ap,i}Q\left(\frac{VPL\left(x\right)-\alpha_i\sqrt{\left(x-a_i\right)^T\left(x-a_i\right)}-B_i}{\sigma_i}\right)=P_{HMI}$$

The problem can be written as:

$$\begin{aligned}&\underset{x}{\text{minimize}} \quad VPL\left(x\right)\\&\text{subject to } x^Tx\le K_{fa}^2\left(\sigma^2_{required\ accuracy}-\sigma^2_{best\ accuracy}\right)\end{aligned}\tag{1.7}$$

Iterative solution

To solve the problem in Equation (1.7) we use an iterative approach. To explain the approach, we first neglect the accuracy constraint, which we will re-introduce in the next paragraph. The iterations are based on a modified Newton method. The Newton method approximates the function VPL by a quadratic function [6]:

$$VPL\left(x+\Delta x\right)\simeq VPL\left(x\right)+\Delta x\nabla VPL\left(x\right)+\frac{1}{2}\Delta x^TH\left(x\right)\Delta x$$

In this equation $\nabla VPL\left(x\right)$ is the gradient of *VPL* and *H(x)* is the Hessian. At each step, the new estimate is the minimizer of the above quadratic function. The new estimate is given by:

$$x_{new}=x_{old}-g\left(x_{old}\right)$$

where *g* is defined by:

$$H\left(x_{old}\right)g\left(x_{old}\right)=\nabla VPL\left(x_{old}\right)$$

For numerical stability, this approach was modified because the Hessian is initially very close to zero. So instead, we compute the direction to follow as the solution of:

$$\left(I+H\left(x_{old}\right)\right)g\left(x_{old}\right)=\nabla VPL\left(x_{old}\right)$$

In addition, instead of updating the solution with this step, we perform a line search along this direction. That is, at each step we solve the minimization:

$$\min_t VPL\left(x_{old}-tg\left(x_{old}\right)\right)$$

This is the minimization of a scalar convex function, which is a straightforward numerical problem, so we will not give more details in this paper. (The convexity will be justified below).

Enforcing accuracy with a barrier function

The accuracy constraint can be implemented using a barrier function [6]. The barrier function used here was:

$$B\left(x\right)=-\log\left(\frac{1}{2}\left(K_{fa}^2\left(\sigma^2_{required\ accuracy}-\sigma^2_{best\ accuracy}\right)-x^Tx\right)\right)$$

We modify the minimization as:

$$\underset{x}{\text{minimize}} \quad VPL(x) + \mu B(x)$$

For this application, it is not necessary to update the parameter μ. Instead it was fixed at 0.01.

Gradients and Hessians

This approach is possible because, even though VPL(x) is only implicitly defined, it is possible to compute both the gradient and the Hessian. It can be shown that we have:

$$\nabla VPL(x) = \frac{\sum_i p_{ap,i} \frac{x - a_i}{\sigma_i |x - a_i|} p\left(\frac{VPL - |x - a_i| - B_i}{\sigma_i} \right)}{\sum_i \frac{p_{ap,i}}{\sigma_i} p\left(\frac{VPL - |x - a_i| - B_i}{\sigma_i} \right)}$$

$$\nabla\nabla VPL(x) = \sum_i \left(\nabla VPL(x) - \frac{x - a_i}{|x - a_i|} \right)\left(\nabla VPL(x) - \frac{x - a_i}{|x - a_i|} \right)^T \frac{VPL(x) - |x - a_i| - B_i}{\sigma_i^2} h_i(x) + \sum_i \left(I - \frac{(x - a_i)(x - a_i)^T}{|x - a_i|^2} \right) \frac{h_i(x)}{|x - a_i|}$$

Where we have defined:

$$h_i(x) = \frac{\frac{p_{ap,i}}{\sigma_i} p\left(\frac{L - |x - a_i| - B_i}{\sigma_i} \right)}{\sum_j \frac{p_{ap,j}}{\sigma_j} p\left(\frac{L - |x - a_j| - B_i}{\sigma_j} \right)}$$

In the above equation, the function p is the density of a zero mean unit Gaussian distribution.

For the barrier function, the formulas are given by:

$$\nabla B(x) = \frac{x}{\frac{1}{2}\left(K_{fa}^2 \left(\sigma_{required\ accuracy}^2 - \sigma_{best\ accuracy}^2 \right) - x^T x \right)}$$

$$\nabla\nabla B(x) = \frac{I}{\frac{1}{2}\left(K_{fa}^2 \left(\sigma_{required\ accuracy}^2 - \sigma_{best\ accuracy}^2 \right) - x^T x \right)} + \frac{x x^T}{\left(K_{fa}^2 \left(\sigma_{required\ accuracy}^2 - \sigma_{best\ accuracy}^2 \right) - x^T x \right)^2}$$

It is easy to verify by inspection that the Hessian of the function $VPL(x) + \mu B(x)$ is a positive semidefinite matrix. This shows that the function is convex.

Stopping criteria and implementation notes

The iterations are stopped either if they reach a maximum of 10 iterations or if the resulting improvement is less than 0.01 m. Tightening this last criteria and increasing the number of iterations did not yield a significant benefit. At every iteration, VPL(x), the gradient, and the Hessian must be computed. As mentioned above, the computation of VPL(x) requires itself an iterative algorithm, as mentioned above and in [2]. The gradient and the Hessian do not and are easily computed. We also note that each of the points a_i are singular points. In particular, the algorithm must be initialized away from x = 0 (although it is practical to choose a point close to it).

ALGORITHM EVALUATION

In this section we evaluate the algorithm under the conditions described in [5], using the VPL and accuracy as requirements for availability. A service volume model tool simulates the geometries experienced by users, computes the corresponding pseudorange error models, and calculates the VPL and accuracy as explained above. To illustrate the potential of the presented algorithm, Figure 1 shows a histogram of the ratio between the new VPL and the old VPL without the accuracy constraint and for a limited set of geometries.

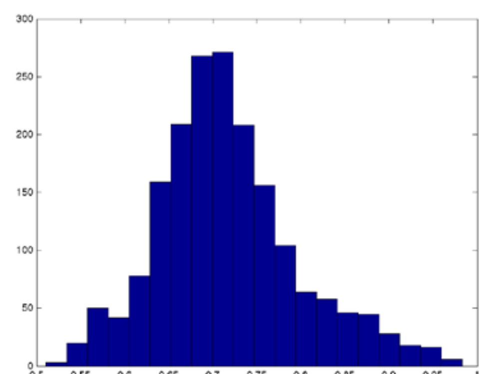

Figure 1. *Ratio between optimized VPL and baseline VPL.*

Although the improvement appears to be significant, it is the effect on availability that is relevant. We show two examples: a single constellation example and a dual constellation example. The exact error models are the ones given in [5]. However, here we are only interested in the availability improvement of the positioning optimization compared to the baseline algorithm chosen above. The figures show the 99.5% VPL as a function of location. The tables show the coverage of 99.5% availability of VPL < VAL = 35 m and 95% accuracy < 4 m.

Single fault ARAIM with one constellation

In the first example we consider a Galileo constellation of 30 satellites in 3 planes. The URA is 1 m, the URE is .67 m, and the bias (b_{int}) is 1 m. Geometries were simulated every 10 degrees and every 5 minutes for 24 hours. Figures 2 and 3 show the baseline VPL results and the new algorithm results respectively. The satellite prior probability of fault was assumed to be 10^{-5}. Table 1 shows the coverage figures.

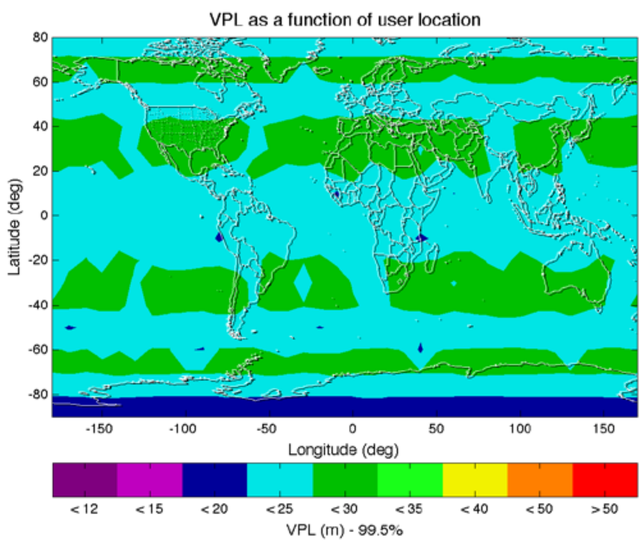

Figure 2. *Baseline results for a 30 satellite Galileo constellation*

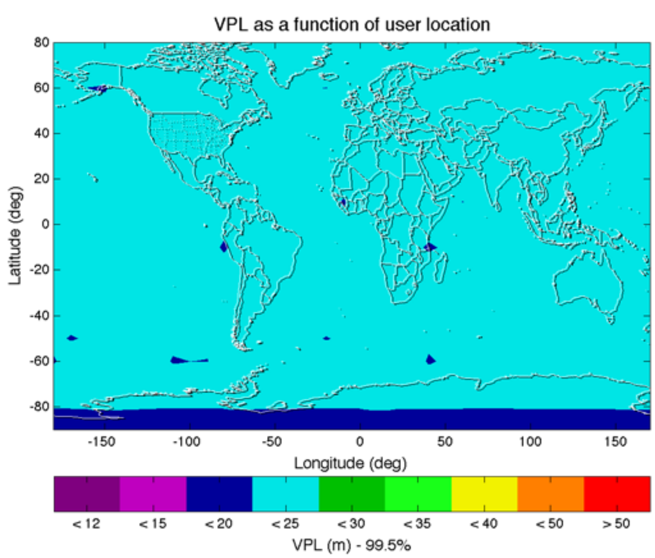

***Figure 3**. Position optimization results for a 30 satellite Galileo constellation*

	Baseline	***Optimal positioning***
Coverage of 99.5% VAL = 35 m	*100%*	*100%*
Coverage of 99.5% VAL = 25 m	*57%*	*100%*

Table 1. *Coverage results for a 30 satellite Galileo constellation*

Dual constellation ARAIM with constellation fault

In the second example we consider a Galileo constellation of 27 satellites in 3 planes and a GPS constellation of 24 satellites in 6 planes. For Galileo, the URA is 1 m, the URE is .67 m, and the bias is 1 m. For GPS, the URA is .75 m, the URE is .5 m, and the bias is .75 m. Geometries were simulated every 10 degrees and every 10 minutes for 10 sidereal days. Figures 4 and 5 show respectively the baseline VPL results and the new algorithm results. The satellite prior probability of fault was assumed to be 10^{-5}, and the prior probability of constellation fault was 10^{-4}. Table 2 shows the coverage figures.

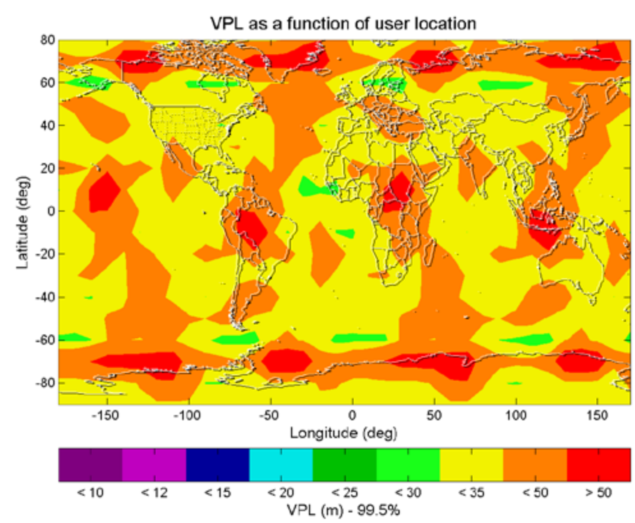

Figure 4. *Baseline results for GPS 24 and Galileo 27 with constellation fault*

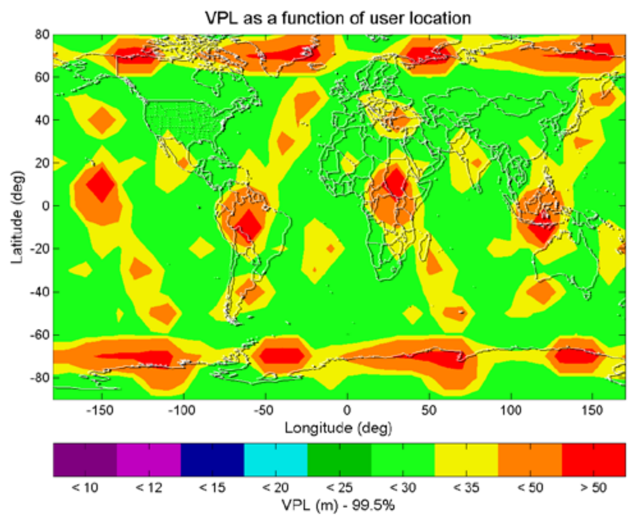

Figure 5. *Position optimization results for GPS 24 and Galileo 27 with constellation fault*

	Baseline	**Optimal positioning**
Coverage of 99.5% VAL = 35 m	60%	84.8%
Coverage of 99% VAL = 35 m	74%	92%

Table 2. *Coverage results for GPS 24 and Galileo 27 with constellation fault*

CONCLUSION

We have presented an improvement for the ARAIM user algorithm within the Solution Separation ARAIM framework. The algorithm optimizes the coefficients of the all-in-view position fix to decrease Protection Levels. The effect on Protection Levels is very significant. We observed up to a 20% reduction on all Protection Levels when there is accuracy margin, which translated into significant coverage improvements. The algorithm is more complex than the optimal integrity allocation only: it involves an iterative process, and about 10 times slower than the baseline algorithm. Therefore it would have to be simplified to be implemented in an airborne receiver.

ACKNOWLEDGEMENTS

This work was sponsored by the FAA GPS Satellite Product Team (AND-730).

REFERENCES

[1] GEAS Phase II Report. http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/library/documents/media/GEAS_PhaseI_report_FINAL_15Feb08.pdf

[2] Blanch, J., Walter, T., Enge, P. “RAIM with Optimal Integrity and Continuity Allocations Under Multiple failures” IEEE Transactions on Aerospace and Electronic Systems Vol. 46, No. 3, July 2010.

[3] Hwang, Patrick Y., Brown, R. Grover, "RAIM FDE Revisited: A New Breakthrough In Availability Performance With NIORAIM (Novel Integrity-Optimized RAIM)," *Proceedings of the 2005 National Technical Meeting of The Institute of Navigation*, San Diego, CA, January 2005, pp. 654-665.

[4] Lee, Young C., "Two New RAIM Methods Based on the Optimally Weighted Average Solution (OWAS) Concept", *NAVIGATION*, Vol. 54, No. 4, Winter 2007-2008, pp. 333-345.

[5] Blanch, Juan, Walter, Todd, Enge, Per, Wallner, Stefan, Fernandez, Francisco Amarillo, Dellago, Riccardo, Ioannides, Rigas, Pervan, Boris, Hernandez, Ignacio Fernandez, Belabbas, Boubeker, Spletter, Alexandru, Rippl, Markus, "A Proposal for Multi-constellation Advanced RAIM for Vertical Guidance," *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation* (ION GNSS 2011), Portland, OR, September 2011, pp. 2665-.

[6] S. Boyd, L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. p 243.

[7] RTCA DO-208, Minimum Operational Performance Standards for Airborne Supplemental Navigation Equipment Using Global Positioning System (GPS)

[8] RTCA DO-316, Minimum Operational Performance Standards for Global Positioning System/Aircraft Based Augmentation System Airborne Equipment