Exclusion for Advanced RAIM: Requirements and a Baseline Algorithm

Juan Blanch, Todd Walter, Per Enge
Stanford University

ABSTRACT

Advanced Receiver Autonomous Integrity Monitoring is a concept that extends RAIM to multi-constellation and dual frequency that could provide worldwide coverage of vertical guidance [1], [2]. A baseline algorithm for the user receiver has been described in [3]. This baseline algorithm included the computation of the Protection Levels, the Effective Monitor Threshold, the accuracy, and a preliminary description of an exclusion algorithm.

The objective of this paper is to develop a set of requirements for the exclusion algorithm (as there is for the computation of the Protection Levels) and to propose a baseline algorithm that fulfills these requirements, therefore attempting to complete the work started in [3].

INTRODUCTION

The purpose of the exclusion function is to provide an integrity error bound and a position solution when a fault is detected, therefore increasing both availability and continuity. Ideally, one would wish to simply exclude the satellite that seems to be causing the measurements to be inconsistent and compute a position solution and error bound as if that satellite had never been present (using for example the formulas described in [3]). However, adding the exclusion option increases the overall exposure to an integrity fault, so that it must be accounted in the integrity risk assessment. This can greatly complicate both the integrity analysis and the Protection Level calculation.

The main objective of the exclusion algorithm presented here is that in the presence of an unambiguous fault, the protection level is the same as if the excluded satellite had not been included in the first place. This constraint guarantees that the temporal behavior of the protection level is consistent, and that the Protection Level calculation after exclusion remains as simple as the all-in-view Protection Level calculation. We will show that this can be achieved through the use of exclusion confirmation tests. This type of test was described in [3], and has already been exploited in previous Fault Detection and Exclusion algorithms (as in [6]). A secondary objective of the approach proposed in this paper is to condition the integrity risk on the knowledge that exclusion has been performed, which means that a fault has probably occurred in a subset of satellites. This is achieved by using a specific allocation of the integrity risk across the different exclusion modes. In this paper we modify, simplify, and complete the approach introduced in [3].

We will first go through the requirements that must be fulfilled by the fault detection and exclusion algorithm as a whole. Then we will describe the exclusion confirmation tests, which are the main tool used here to fulfill the requirements when exclusion is performed. After briefly describing the different steps of the ARAIM airborne algorithm, we will derive an integrity equation. Based on this integrity equation, we will describe how to set the thresholds and Protection Levels to fulfill the integrity equation.

INTEGRITY REQUIREMENT

The integrity requirement is a probability requirement, and is therefore dependent on what is supposed to be known when computing the probability. In the integrity risk calculation done in RAIM and ARAIM, it is usually assumed that the probability of Hazardously Misleading Information is conditioned on the geometry of the satellites, the nominal error model, and the prior probabilities of fault. When considering exclusion, a question arises: do we condition the integrity risk by the knowledge that exclusion has been performed or do we assume that it is an event to which we can assign a probability? In the approach proposed here, we attempt to rigorously meet the integrity requirement considering exclusion as a random event. That is, the probability of HMI is not conditioned on whether a certain fault has been excluded. However, we will attempt to introduce a notion of conditional integrity through the choice of the integrity risk allocation across the different exclusion cases.

CONTINUITY REQUIREMENT

Continuity of service can be affected by flagged faults, satellites that go out of view (because of banking), or by faults detected by RAIM. The algorithm described here will assume that continuity is mostly affected by the former reasons and not the latter one. To further support this assumption, we consider the risk of a continuity failure due to a fault. Let us consider the probability of onset of any fault P_{onset}, and the continuity exposure time T_{exposure}. The probability that a pseudorange fault causes a detection (and attempted exclusion) is on the order of:

$$P\left(\text{detection during } T_{\text{exposure}}\right) = P_{onset} T_{\text{exposure}}$$

Using a value of 4 x 10^{-4}/hour for P_{onset}, and a T_{exposure} of 15s (defined in [5]), we get the estimate:

$$P\left(\text{detection during } T_{\text{exposure}}\right) = 2\times 10^{-4}\,\frac{15}{3600} \approx 10^{-6}$$

This is well below the continuity risk requirement of 8 x 10^{-6}/15s. Finally, as described in [1], the continuity requirement is a false alert requirement under fault free conditions. This approach is consistent with the approaches taken in the integrity monitoring algorithms for SBAS.

EXCLUSION CONFIRMATION TESTS

The exclusion confirmation tests are designed to limit the probability that the exclusion process leads to a false exclusion. The idea (already exploited in previous work, as in [6]) is to check whether excluding the fault source *j* is the only way to explain the large residuals. In other words, we make sure that the probability of having large residuals due to threat *i* is unlikely. This is done by going through all possible threats *i*. For each threat *i*, a statistic independent of whether fault *i* is present is formed. For a given threat *i*, the probability that these statistics exceed a certain threshold can be controlled through the choice of the threshold. In the case of single faults, the exclusion confirmation test is simply that if we have a threshold test that doesn't pass, it is only by removing one given satellite that we obtain a consistent set of measurements. The algorithm presented here generalizes this approach to multiple faults and links (what has been called in previous literature) the probability of false exclusion to the integrity requirement.

PROPOSED EXCLUSION ALGORITHM

In this section we describe the proposed fault detection and exclusion algorithm. The notations of [3] are used. We will only describe process for the Vertical Protection Level.

1) The solution separation statistics and thresholds are computed as indicated in [3].

2) If the solution separation statistics are within the thresholds, the all-in-view VPL, noted VPL_0 is computed.

3) If the threshold tests do not pass, the algorithm determines a subset such that: it is consistent (in the sense that the solution separation statistics are within the thresholds) and that there are no larger sets that are consistent. [3] shows that the determination can be done by computing the chi-square statistic of the subsets that exclude each candidate (is is not necessary to go through all the solution separation statistics within the subsets). Let us label *j* the fault mode that is excluded. The new position solution is labeled $\hat{x}^{(j)}$.

4) In this step, the confirmation test statistics are formed as follows. For each fault mode *i* such that *j* is not included in *i*, we compute the residual:

   $$\left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right|$$

   We compare it to a the residual $T_{excl,j,ij}$. If for all *i* we have:

   $$\left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij} \qquad (1)$$

   Then the exclusion of fault *j* is said to be confirmed. In this case the VPL, which we label VPL_j^C, is computed. In this case, the fault *j* is excluded for a period $T_{quarantine}$. This constant is determined by the time it takes the ground segment to either, remove the fault, or flag it as unhealthy.

5) If the confirmation test does not pass, we still exclude the fault but compute instead VPL_j^N, and the fault is not quarantined (at the next time step, all satellites are included and tested).

INTEGRITY EQUATION

This section evaluates the integrity risk resulting from the algorithm described above. As before, *i* is the index associated to each fault mode that is monitored, and *j* is the index associated to each fault source *j* that is excluded (for a method to determine faults to be monitored and the term $P_{not\ monitored}$, refer to [3], although some notations have been lightened to improve readability):

$$P(HMI) = P\left(\bigcup_{j}\{HMI,\ j \text{ excluded}\}\right)$$
$$P(HMI) = P\left(\bigcup_{i,j}\{HMI,\ j \text{ excluded, fault } i\}\right) + P_{not\_monitored} \tag{2}$$

In the next step we sort the indexes *(i,j)* in two groups. The first group contains the indexes such that fault *j* is not included in *i*. The second group contains the indexes such that fault *j* is included in *i* (abusing notation, we will note this $j \subset i$). For example, if *i* designates a fault in satellites 1 and 2, and *j* a fault in satellite 1, then *(i,j)* is in the second group.

$$\begin{aligned} &P\left(\bigcup_{i,j}\{HMI,\ j \text{ excluded, fault } i\}\right) = \\ &P\left\{\bigcup_{i,j/j\not\subset i}\left(HMI,\ j \text{ excluded, fault } i\right)\right\} \\ &+P\left\{\bigcup_{i,j/j\subset i}\left(HMI,\ j \text{ excluded, fault } i\right)\right\} \end{aligned} \tag{3}$$

The event that we have HMI, that *j* is excluded, and that there is a fault in *i* is the union of two events. This can be written:

$$\begin{aligned} &\{HMI,\ j \text{ excluded, fault } i\} = \\ &\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \\ &\cup\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^N, y \in \Omega_j^N, \text{ fault } i\right\} \end{aligned} \tag{4}$$

Ω_j^C is the region of the measurements *y* such that fault *j* is excluded and the confirmation tests pass. Ω_j^N is the region of the measurements *y* such that fault *j* is excluded but the confirmation tests don't pass. We therefore have:

$$\begin{aligned} &\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \subset \\ &\left(\begin{aligned}&\left\{\hat{x}_3^{(ij)} - x_3 > VPL_j^C - T_{j,ij,3} - b_3^{(ij)}, \text{ fault } i\right\} \\ &\cup\left\{\hat{x}_3^{(j)} - x_3 < -VPL_j^C + b_3^{(j)}, \text{ fault } i\right\}\end{aligned}\right) \\ &\text{and} \\ &\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^N, y \in \Omega_j^N, \text{ fault } i\right\} \subset \\ &\left(\begin{aligned}&\left\{\hat{x}_3^{(ij)} - x_3 > VPL_j^N - T_{j,ij,3} - b_3^{(ij)}, \text{ fault } i\right\} \\ &\cup\left\{\hat{x}_3^{(j)} - x_3 < -VPL_j^N + b_3^{(j)}, \text{ fault } i\right\}\end{aligned}\right) \end{aligned} \tag{5}$$

As in Appendix H of [3], these inclusions assume that the effect of fault *i* on $\hat{x}_3^{(j)}$ is a positive bias.

Faults for which an exclusion test exists

We first evaluate the first term (where the fault *i* is not included in *j*) of Equation (3). We have the upper bound:

$$\begin{aligned} &P\left\{\bigcup_{i,j/j\not\subset i}\left(HMI,\ j \text{ excluded, fault } i\right)\right\} \\ &\le \sum_{j}\sum_{i/j\not\subset i} P\left(\left(HMI,\ j \text{ excluded, fault } i\right)\right) \end{aligned} \tag{6}$$

Each exclusion case *j* is not considered separately. We have:

$$\begin{aligned} &P\left(HMI,\ j \text{ excluded, fault } i\right) = \\ &P\left(HMI,\ j \text{ excluded not confirmed, fault } i\right) \\ &+P\left(HMI,\ j \text{ excluded confirmed, fault } i\right) \end{aligned} \tag{7}$$

We look at the first term in Equation (7):

$$\begin{aligned} &P\left(HMI,\ j \text{ excluded not confirmed, fault } i\right) = \\ &P\left\{\left|\hat{x}_3^{(j)} - x\right| > VPL_j^N, y \in \Omega_j^N, \text{ fault } i\right\} \end{aligned}$$

$$\begin{aligned} &P\left\{\left|\hat{x}_3^{(j)} - x\right| > VPL_j^N, y \in \Omega_j^N, \text{ fault } i\right\} \le \\ &P\left\{\left|\hat{x}_3^{(j)} - x\right| > VPL_j^N, \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(j)}\right| \le T_{j,ij,3}, \text{ fault } i\right\} \end{aligned} \tag{8}$$

These are the same conditions as Appendix H in [3], so we can write:

$$\sum_i P\left(HMI, j \text{ excluded not confirmed, fault } i\right) \leq$$
$$2Q\left(\frac{VPL_j^N - b_3^{(j)}}{\sigma_j}\right) + \sum_{\substack{i/i \not\subset j \\ j \not\subset i}} Q\left(\frac{VPL_j^N - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}}\right) p_{fault,i} \quad (9)$$

We do not need to consider in the sum the threats such that i is included in j, because they are accounted in the first term.

For the second term of Equation (7), we have:

$$P\left(HMI, j \text{ excluded confirmed, fault } i\right) = P\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \quad (10)$$

Recalling the conditions under which the exclusion of fault j is said to be confirmed, we have:

$$\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \subset \left\{\begin{array}{l}\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(j)}\right| \leq T_{j,ij,3}, \\ \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij,3}, \text{ fault } i\end{array}\right\} \quad (11)$$

In particular we have:

$$P\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \leq P\left(\left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij,3}, \text{ fault } i\right) \quad (12)$$

The distribution of the residual $\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}$ is not affected by fault i, so we can write:

$$P\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \leq P\left(\left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij,3}\right) p_{fault,i} \quad (13)$$

Let us assume that $\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}$ has a zero mean normal distribution with standard deviation:

$$\sigma_{ss,3}^{(i,ij)2} = e_3^T \left(S^{(ij)} - S^{(i)}\right) C_{acc} \left(S^{(ij)} - S^{(i)}\right)^T e_3 \quad (14)$$

For the definition of the different terms in this Equation, please refer to [3] (please see the Appendix for additional comments on this assumption). We can write:

$$P\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \leq Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right) p_{fault,i} \quad (15)$$

Summing over the indices j, we obtain:

$$\sum_{i/j \not\subset i} P\left(HMI, j \text{ excluded confirmed, fault } j\right) \leq 2\sum_{i/j \not\subset i} Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right) p_{fault,i} \quad (16)$$

Faults for which a confirmation test does not exist

We now consider the second term in Equation (3). For the pairs in this group we have $j \subset i$, therefore:

$$\hat{x}^{(ij)} = \hat{x}^{(i)}$$

As a consequence:

$$\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, y \in \Omega_j^C, \text{ fault } i\right\} \subset \left(\begin{array}{l}\left\{\hat{x}_3^{(i)} - x_3 > VPL_j^C - T_{j,ij,3} - b_3^{(i)}, \text{ fault } i\right\} \\ \cup \left\{\hat{x}_3^{(j)} - x_3 < -VPL_j^C + b_3^{(j)}, \text{ fault } i\right\}\end{array}\right)$$

and

$$\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^N, y \in \Omega_j^N, \text{ fault } i\right\} \subset \left(\begin{array}{l}\left\{\hat{x}_3^{(i)} - x_3 > VPL_j^N - T_{j,ij,3} - b_3^{(i)}, \text{ fault } i\right\} \\ \cup \left\{\hat{x}_3^{(j)} - x_3 < -VPL_j^N + b_3^{(j)}, \text{ fault } i\right\}\end{array}\right) \quad (17)$$

We show in the Appendix that it is reasonable to assume that we have:

$$VPL_j^C - T_{j,ij,3} \geq VPL_0 - T_{0,i,3}$$

and

$$VPL_j^N - T_{j,ij,3} \geq VPL_0 - T_{0,i,3} \quad (18)$$

This is true for both the confirmed case and non-confirmed case. As a consequence we have:

$$\left\{\left|\hat{x}_3^{(i)} - x_3\right| > VPL_j^{N \text{ or } C} - T_{j,ij,3} - b_3^{(i)}, \text{ fault } i\right\} \subset \left(\begin{array}{l}\left\{\hat{x}_3^{(i)} - x_3 > VPL_0 - T_{0,i,3} - b_3^{(i)}, \text{ fault } i\right\} \\ \cup \left\{\hat{x}_3^{(j)} - x_3 < -VPL_j^{N \text{ or } C} + b_3^{(j)}, \text{ fault } i\right\}\end{array}\right) \quad (19)$$

From this point, we will neglect the probability of the event on the right hand side, as given the sign of the fault, it is much less likely than the left hand side (except for the case j=0).

This implies that for all indices j such that $j \subset i$ (excluding j =0):

$$\left\{ \bigcup_{j/j \subset i} \left(HMI, j \text{ excluded, fault } i \right) \right\} \subset \left\{ \hat{x}_3^{(i)} - x_3 > VPL_0 - T_{0,i,3} - b_3^{(i)}, \text{ fault } i \right\} \quad (20)$$

And:

$$P\left\{ \bigcup_{i,j/j \subset i} \left(HMI, j \text{ excluded, fault } i \right) \right\} \le$$
$$P\left\{ \hat{x}_3^{(0)} - x_3 < -VPL_0 + b_3^{(0)} \right\} + \sum_i P\left\{ \hat{x}_3^{(i)} - x_3 > VPL_0 - T_{0,i,3} - b_3^{(i)} \right\} p_{fault,i}$$
$$\le 2Q\left(\frac{VPL_0 - b_3^{(0)}}{\sigma_0} \right) + \sum_i Q\left(\frac{VPL_0 - T_{0,i,3} - b_3^{(i)}}{\sigma_i} \right) p_{fault,i} \quad (21)$$

This is very useful: it shows that the probability of HMI due to fault i and all exclusion cases of j included in i is bounded by the term already included in the no exclusion case (as long as the inequalities (18) hold).

Note: If we had not used the assumption that the effect of fault i on each subset solution has the same sign for each solution $\hat{x}_3^{(j)}$, we would have needed to add a factor of 2 to each term of the sum in Equation (21).

Final integrity equation

The final integrity equation is then:

$$2Q\left(\frac{VPL_0 - b_3^{(0)}}{\sigma_0} \right) + \sum_i Q\left(\frac{VPL_0 - T_{0,i,3} - b_3^{(i)}}{\sigma_i} \right) p_{fault,i} +$$
$$\sum_j \left(2Q\left(\frac{VPL_j^N - b_3^{(j)}}{\sigma_j} \right) + \sum_{\substack{i/i \not\subset j \\ j \not\subset i}} Q\left(\frac{VPL_j^N - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}} \right) p_{fault,i} \right) +$$
$$2\sum_j \sum_{i/j \not\subset i} Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}} \right) p_{fault,i} = PHMI_{VERT} - P_{not\ monitored} \quad (22)$$

In the case where there are no confirmation tests, the equation becomes:

$$2Q\left(\frac{VPL_0 - b_3^{(0)}}{\sigma_0} \right) + \sum_i Q\left(\frac{VPL_0 - T_{0,i,3} - b_3^{(i)}}{\sigma_i} \right) p_{fault,i} +$$
$$\sum_{j \ge 1} \left(2Q\left(\frac{VPL_j^N - b_3^{(j)}}{\sigma_j} \right) + \sum_{\substack{i/i \not\subset j \\ j \not\subset i}} Q\left(\frac{VPL_j^N - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}} \right) p_{fault,i} \right) +$$
$$= PHMI_{VERT} - P_{not\ monitored} \quad (23)$$

If in addition we made all the VPLs identical for each exclusion case, we would end up with an integrity equation very similar to the one described in [4].

Interpretation

Equation (22) has three contributions. The first one is:

$$2Q\left(\frac{VPL_0 - b_3^{(0)}}{\sigma_0} \right) + \sum_i Q\left(\frac{VPL_0 - T_{0,i,3} - b_3^{(i)}}{\sigma_i} \right) p_{fault,i} \quad (24)$$

It accounts for the all-in-view integrity risk as well as all the combinations of exclusion cases and threats for which there is no exclusion confirmation test.

The second one is:

$$\sum_{j \ge 1} \left(2Q\left(\frac{VPL_j^N - b_3^{(j)}}{\sigma_j} \right) + \sum_{\substack{i/i \not\subset j \\ j \not\subset i}} Q\left(\frac{VPL_j^N - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}} \right) p_{fault,i} \right) \quad (25)$$

It accounts for each exclusion case that has not been confirmed.

The third one is:

$$2\sum_j \sum_{i/j \not\subset i} Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}} \right) p_{fault,i} \quad (26)$$

It accounts for each confirmed exclusion case. At first sight it might seem puzzling that VPL_j^C does not appear in the equation. However, this equation is only valid if the conditions (18) hold. In addition, after a confirmed exclusion, and for consistency in the results, the VPL must be at least as large as it would be if the fault j had been flagged by a mechanism exterior to ARAIM.

INTEGRITY BUDGET ALLOCATION

In the previous section, we have seen which conditions must be fulfilled by VPL_0, VPL_j^N, VPL_j^C and the thresholds $T_{excl,i,ij,3}$. In this section we describe a practical way of determining them. To simplify the processing, we set as a constraint that the receiver should only need to compute the thresholds, standard deviations, and biases associated to exclusion j if j is actually excluded. This means that the allocation across the different modes must be made independent of the geometry.

Integrity allocation based on conditional integrity risk

We rewrite the integrity equation as follows:

$$2Q\left(\frac{VPL_0-b_3^{(0)}}{\sigma_0}\right)+\sum_i Q\left(\frac{VPL_0-T_{0,i,3}-b_3^{(i)}}{\sigma_i}\right)p_{fault,i}+\sum_j\left(\begin{array}{l}2Q\left(\frac{VPL_j^N-b_3^{(j)}}{\sigma_j}\right)+\sum_{\substack{i/i\not\subset j\\ j\not\subset i}} Q\left(\frac{VPL_j^N-T_{j,ij,3}-b_3^{(ij)}}{\sigma_{ij}}\right)p_{fault,i}\\ +\sum_{i/j\not\subset i}2Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right)p_{fault,i}\end{array}\right)=PHMI_{VERT}-P_{not\ monitored} \quad (27)$$

We can approximately say that the j^{th} term in this sum represents $P(HMI, j \text{ excluded})$, that is, the probability that HMI occurs and j is excluded:

$$P(HMI, j\ \text{excluded})=2Q\left(\frac{VPL_j^N-b_3^{(j)}}{\sigma_j}\right)+\sum_{\substack{i/i\not\subset j\\ j\not\subset i}} Q\left(\frac{VPL_j^N-T_{j,ij,3}-b_3^{(ij)}}{\sigma_{ij}}\right)p_{fault,i}+\sum_{i/j\not\subset i}2Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right)p_{fault,i} \quad (28)$$

The first step consists on allocating the available integrity budget to each exclusion case. We find a_j such that:

$$\sum_j a_j = 1 \quad (29)$$

We will then impose:

$$P(HMI, j\ \text{excluded})\le a_j\left(PHMI_{VERT}-P_{not\ monitored}\right) \quad (30)$$

As expressed above, any choice of positive coefficients fulfilling will work. At this point we do add a new constraint: we would like to have that the probability of HMI given that j is excluded is below the available integrity budget. That is:

$$P(HMI \mid j\ \text{excluded})\le PHMI-P_{not,monitored} \quad (31)$$

Writing Bayes formula we get:

$$\frac{P(HMI, j\ \text{excluded})}{P(j\ \text{excluded})}\le PHMI-P_{not,monitored} \quad (32)$$

The probability that j is excluded can be approximated by the probability of the fault j. The condition above is therefore written:

$$P(HMI, j\ \text{excluded})\le p_{fault,j}\left(PHMI-P_{not,monitored}\right) \quad (33)$$

This means that:

$$a_j = p_{fault,j} \quad (34)$$

In addition, we need to decide how to allocate among the confirmed and not confirmed case. For this we will introduce the parameter θ. We allocate $\theta p_{fault,j}$ to the non-confirmed case and $(1-\theta)p_{fault,j}$ to the confirmed case.

THRESHOLDS AND PROTECTION LEVEL FORMULAS

Continuity thresholds

For each of the exclusion functions, the continuity $T_{j,ij,3}$ thresholds are determined as in [3]. That is, they are set such that in the presence of fault j, the probability of exceeding the consistency threshold must be below the continuity requirement for the set that excludes j. Notice that a different interpretation of the continuity requirement could lead to different continuity allocations across the exclusion cases (as in [4]).

All-in-view VPL

The all-in-view VPL, noted VPL_0 is determined by the equation:

$$2Q\left(\frac{VPL_0-b_3^{(0)}}{\sigma_0}\right)+\sum_i Q\left(\frac{VPL_0-T_{0,i,3}-b_3^{(i)}}{\sigma_i}\right)p_{fault,i}=p_{no\ fault}\left(PHMI_{VERT}-P_{not\ monitored}\right) \quad (35)$$

In practice, $p_{nofault}$ is so close to one that we can assume that the VPL is determined as shown in [3].

Exclusion VPL with no confirmation

The equation determining VPL_j^N is:

$$2Q\left(\frac{VPL_j^N - b_3^{(j)}}{\sigma_j}\right) + \sum_{i/i\not\subset j} Q\left(\frac{VPL_j^N - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}}\right) p_{fault,i}$$
$$= \theta p_{fault,j}\left(PHMI_{VERT} - P_{not\ monitored}\right) \quad (36)$$

According to the integrity equation, it is only necessary to include the indexes i such that $j \not\subset i$. However, to ensure that the inequalities (18) hold, we include all threats that affect the position solution $\hat{x}_3^{(j)}$. This also ensures that the equation is formally similar to the all-in-view equation.

Exclusion thresholds

The exclusion thresholds must be such that we have:

$$\sum_{i/j\not\subset i} 2Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right) p_{fault,i} = \left(1-\theta\right) p_{fault,j}\left(PHMI_{VERT} - P_{not\ monitored}\right) \quad (37)$$

For the baseline algorithm, we can choose to allocate the integrity risk evenly across the different terms:

$$2Q\left(\frac{T_{excl,j,ij,3}}{\sigma_{ss,3}^{(i,ij)}}\right) p_{fault,i} = \frac{\left(1-\theta\right)}{N_j} p_{fault,j}\left(PHMI_{VERT} - P_{not\ monitored}\right) \quad (38)$$

In this equation, N_j is the number of terms in the sum of Equation (37). Finally:

$$T_{excl,j,ij,3} = Q^{-1}\left(\frac{\left(1-\theta\right)}{2N_j}\frac{p_{fault,j}}{p_{fault,i}}\left(PHMI_{VERT} - P_{not\ monitored}\right)\right)\sigma_{ss,3}^{(i,ij)} \quad (39)$$

Exclusion VPL with confirmation

The exclusion VPL with confirmation is computed exactly as the all-in-view VPL, but excluding fault *j*. It can be written:

$$2Q\left(\frac{VPL_j^C - b_3^{(j)}}{\sigma_j}\right) + \sum_{i/i\not\subset j} Q\left(\frac{VPL_j^C - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}}\right) p_{fault,i} =$$
$$\left(1 - \sum_{i/j\not\subset i} p_{fault,i}\right)\left(PHMI_{VERT} - P_{not\ monitored,j}\right) \quad (40)$$

The term $P_{not\ monitored}$ has changed (and a subscript *j* is added) because it might be different for the subset that excludes *j*. As for the all-in-view VPL, in practice the Equation can be considered to be:

$$2Q\left(\frac{VPL_j^C - b_3^{(j)}}{\sigma_j}\right) + \sum_{i/i\not\subset j} Q\left(\frac{VPL_j^C - T_{j,ij,3} - b_3^{(ij)}}{\sigma_{ij}}\right) p_{fault,i}$$
$$= PHMI_{VERT} - P_{not\ monitored,j} \quad (41)$$

As indicated above, the integrity equation is valid if we have the inequalities (18). We show in Appendix A that this will generally hold if the VPL is computed using Equation (41).

SUMMARY

There are three contributions in this paper: the description of a baseline exclusion algorithm for ARAIM that uses exclusion confirmation tests, the integrity risk equation that the Protection Levels and exclusion confirmation tests must fulfill, and a practical method to determine them. One of the goals of the algorithm presented here is to provide a set of conditions under which a fault (satellite or group of satellites, for example) can be excluded and treated as if it was flagged by the ground. This property allows the receiver to treat the satellites remaining after exclusion as an all-in-view situation, which greatly simplifies the processing. In addition, the allocation of the integrity risk among the different exclusion modes is determined to provide a notion of conditional integrity risk.

REFERENCES

[1] Working Group C, ARAIM Technical Subgroup, Interim Report, Issue 1.0, December 19, 2012. Available at http://ec.europa.eu/enterprise/newsroom/cf/_getdocument.cfm?doc_id=7793 and http://www.gps.gov/policy/cooperation/europe/2013/working-group-c/ARAIM-report-1.0.pdf

[2] Blanch, J., Walter, T., Enge, P., Wallner, S., Fernandez, F., Dellago, R., Ioannides, R., Pervan, B.,

Hernandez, I., Belabbas, B., Spletter, A., and Rippl, M., "Critical Elements for Multi-Constellation Advanced RAIM for Vertical Guidance," to appear in *Navigation, Journal of the Institute of Navigation*.

[3] Blanch, J., Walter, T., Enge, P., Lee, Y., Pervan, B., Rippl, M., Spletter, A., "Advanced RAIM user Algorithm Description: Integrity Support Message Processing, Fault Detection, Exclusion, and Protection Level Calculation," *Proceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012)*, Nashville, TN, September 2012.

[4] M. Joerger, S. Stevanovic, F-C. Chan, S. Langel, B. Pervan. "Integrity Risk and Continuity Risk for Fault Detection and Exclusion Using Solution Separation ARAIM," *Proceedings of the 26th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2013)*, Nashville, TN, September 2013.

[5] ICAO, Annex 10, Aeronautical Telecommunications, Volume 1 (Radio Navigation Aids), Amendment 84, published 20 July 2009, effective 19 November 2009. GNSS standards and recommended practices (SARPs) are contained in Section 3.7 and subsections, Appendix B, and Attachment D.

[6] B. S. Pervan, D. G. Lawrence, C. E. Cohen, B. R. Parkinson. "Parity space methods for autonomous fault detection and exclusion using GPS carrier phase," *Proceedings of IEEE PLANS 1996*.

ACKNOWLEDGEMENTS

The authors wish to thank all the members of the WG-C ARAIM subgroup. This research was sponsored by the F.A.A.

APPENDIX A

In this Appendix we show why it is reasonable to assume that the inequalities (18) hold. First we will assume that the VPL after exclusion is larger than the all-in-view:

$$VPL_j^{N\ or\ C} \geq VPL_0 \quad (42)$$

If it doesn't hold using the formulas, this inequality could be enforced by the receiver, if deemed necessary. We show that we have:

$$T_{j,ij,3} \leq T_{0,i,3} \quad (43)$$

We show it here in the case that C_{acc} and C_{int} coincide. In this case we would have:

$$\sigma_{ss,3}^{(0,i)2} = \sigma_3^{(i)2} - \sigma_3^{(0)2}$$
$$\sigma_{ss,3}^{(j,ij)2} = \sigma_3^{(ij)2} - \sigma_3^{(j)2} \quad (44)$$

Since j is included in i, we have:

$$\sigma_3^{(i)2} = \sigma_3^{(ij)2} \quad (45)$$

And:

$$\sigma_3^{(0)2} \leq \sigma_3^{(j)2} \quad (46)$$

Combining the last three equations yields:

$$\sigma_{ss,3}^{(j,ij)2} \leq \sigma_{ss,3}^{(0,i)2} \quad (47)$$

Because the continuity budget is divided among fewer subsets in the excluded case, we have (see formula in [3]):

$$K_{fa,3,j} \leq K_{fa,3,0} \quad (48)$$

Combining the two last equations, we get (43), which ends the proof.

APPENDIX B

Refining the integrity equation

More credit can be taken from Equation (11), as the proposed approach does not take into account the VPL. In [3], the following approximation was made:

$$P\left\{\begin{array}{l}\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(j)}\right| \leq T_{j,ij,3}, \\ \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij,3}, \text{ fault } i\end{array}\right\}$$
$$\simeq P\left\{\left|\hat{x}_3^{(j)} - x_3\right| > VPL_j^C, \left|\hat{x}_3^{(ij)} - \hat{x}_3^{(j)}\right| \leq T_{j,ij,3}, \text{ fault } i\right\}$$
$$P\left(\left|\hat{x}_3^{(ij)} - \hat{x}_3^{(i)}\right| > T_{excl,j,ij,3}\right) \quad (49)$$

Because this approximation is not strictly valid (although it is very good for strong geometries), in this paper we choose to only take credit for the exclusion confirmation test. At the expense of more complexity, it is possible to exploit this equation more than it is proposed here.

Use of continuity error model for the exclusion confirmation tests

As described above the exclusion tests use the continuity error model, as shown in Equation (14). This choice was made because there are already many layers of conservatism in the integrity bounding. For a more conservative approach, we can use instead:

$$\sigma_{ss,3}^{(i,ij)2} = e_3^T \left(S^{(ij)} - S^{(i)}\right) C_{\text{int}} \left(S^{(ij)} - S^{(i)}\right)^T e_3 = \sigma_3^{(ij)2} - \sigma_3^{(i)2} \qquad (50)$$

(as well as the corresponding bias).

Use of chi-square instead of solution separation in exclusion confirmation tests

For the description above, we have chosen to use the solution separation for the exclusion confirmation tests. It is possible that a better choice is given by the chi-square statistic that is independent of fault *i*, for example, in the case of fault *i*, it would be (using the notations of [3]):

$$y_{\hat{i}}^T W_{\hat{i}} y_{\hat{i}} - \left(G_{\hat{i}}^T W_{\hat{i}} y_{\hat{i}}\right)^T \left(G_{\hat{i}}^T W_{\hat{i}} G_{\hat{i}}\right)^{-1} G_{\hat{i}}^T W_{\hat{i}} y_{\hat{i}} \qquad (51)$$

Indeed, faults that would manifest themselves in the chi-square statistic might not be apparent in the solution separation statistic. Alternatively, we could add tests on the horizontal solution separation.