Baseline Advanced RAIM User Algorithm: Proposed Updates

Juan Blanch, Todd Walter, *Stanford University*

Carl Milner, *ENAC*

Mathieu Joerger, *Virginia Tech*

Boris Pervan, *IIT*

Denis Bouvet, *Thales Avionics*

ABSTRACT

As part of the development of the Advanced RAIM, a baseline user algorithm was developed to demonstrate the feasibility of the concept. A first version was published in 2012 and subsequent modifications were made available in different forums. The purpose of this paper is to describe a possible set of updates to the original baseline algorithm that integrates the latest safety analyses (the effect of temporal exposure and the effect of exclusion), that exploits the proposed default Integrity Support Data (the parameters describing the nominal error model and the faults), and that enables the use of techniques to drastically reduce the computational complexity.

INTRODUCTION

The Advanced Receiver Autonomous Integrity Monitoring (ARAIM) concept, an evolution of RAIM to multi-constellation and dual frequency signals, is currently being standardized within ICAO and RTCA/EUROCAE. A reference user algorithm was part of the report describing the initial ARAIM concept developed within the bilateral US-EU Working Group C [1,2,3]. This reference algorithm has been used to evaluate the expected ARAIM performance [4] and for early prototyping [5]. Although there is no plan to make the baseline algorithm a requirement in the standards, it remains a key input for their development, because it provides an acceptable method to implement ARAIM at the user receiver and demonstrates how the integrity support data must be interpreted.

As part of this standardization process, the initial baseline user algorithm needs to be updated and refined to address recent integrity and continuity analyses [6,7], and to reduce the computational load [8]. Also, proposed default values for the Integrity Support Data are now available for all the GNSS constellations that could be used in ARAIM. These default values are expected to be valid even if no Integrity Support Message is received and they may be seen as a lower bound on the expected performance. These lower bounds can be exploited to simplify the design of the algorithm and to improve the computational efficiency.

The purpose of this paper is to describe updates to the baseline algorithm that integrates these latest developments, and to clarify certain design choices. After reviewing the basic elements of an Advanced RAIM user algorithm, we will focus on the following points:

1) *Subset selection*: in [3], the subset selection is designed to minimize the number of subsets in the fault detection mode. For algorithms where the integrity allocation among exclusion modes is not optimized (to limit computational load), it can lead to less-than-ideal subset choices. Now that we have a set of default ISD, we can define a subset selection approach that is both simpler and better in terms of performance (computation and availability)

2) *Exclusion function*: after going over why we need to allocate the integrity budget to each exclusion option, we will justify the choice of exclusion options based on the continuity requirement [7]. More precisely, we will show that for the proposed ISD default values, it is sufficient to consider single fault modes (either satellite or constellation).

3) *Temporal exposure integrity analysis*: we will refine the Equations introduced in [6] and show how to exploit the maximum possible integrity risk contribution from a given fault mode. To do this, we will go over the impact of the faults that are not monitored on the integrity risk.

4) *Methods to reduce computational load*: we will integrate in the algorithm description the techniques described in [8,9]. These techniques can drastically reduce the computational load with very little performance impact, both in availability simulations and in real time implementations

OVERVIEW OF THE ALGORITHM

The structure of the baseline algorithm, as described in [1], [2], [3], and [6] at different stages of development, has not changed. Although we will attempt to have a complete description of the proposed updated algorithm, we will direct the user to the references for some of the details and justifications.

The main purpose of the algorithm is to provide a position estimate and protection levels such that both integrity and continuity requirements are met. For this algorithm, our intent is that these two requirements are demonstrated using mathematical proofs. At the same time, we also attempt to keep the algorithm relatively simple and computationally tractable. This does result in steps that may appear to be overly conservative. We try to point out where those tradeoffs occur.

To compute the position estimate and the protection level, the algorithm goes through the following steps:

1. Determine the nominal pseudorange error model
2. Determine the fault modes that need to be monitored as well as the probability of the faults that are not monitored (this is done based on the Integrity Support Data and the geometry)
3. Determine the candidates for exclusion (note that this is necessary even if the exclusion function is not triggered)
4. Test the consistency of the measurements using the fault detection statistics, or of one of the subsets in case the all-in-view set does not pass (this is the exclusion function)
5. Compute the protection level

We note that this order is not strict, and there may be advantages in performing some of those steps simultaneously. For example, performance may be optimized by doing 2, 4, and 5 simultaneously.

INTEGRITY REQUIREMENTS

The main change in the integrity requirement formulation with respect to previous versions has consisted in explicitly considering the effect of temporal exposure [6] (rather than relying on ad hoc justifications for the use of an instantaneous integrity risk). For LPV operations, the time interval associated to the integrity risk is 150 s. For RNP operations, the integrity risk is expressed as a probability per hour. Due to edge effects on the integrity risk computation over an interval, it turns out that the probability over a given interval is not proportional to the length of the interval, even for small probabilities. As a result, for RNP, we need to choose a length of interval T_{EXP} to assess the integrity risk. The equations that are presented here will assume one hour, but other choices may be acceptable.

An expression of the integrity risk (IR) that accounts for the Time-to-alert (TTA) and the exposure is given by:

$$IR = P\left(\exists\, t\, \in I_{EXP} \mid \left\{ \begin{array}{c} \min_{\tau \in [t-TTA,t]} \left(VPE(\tau)\right) > VAL \\ or \\ \min_{\tau \in [t-TTA,t]} \left(HPE(\tau)\right) > HAL \end{array} \right\} \& \text{ no alert at time } t \right) \quad (1)$$

Where:

  I_{EXP} is the exposure interval (of length T_{EXP})

  $VPE(\tau)$ and $HPE(\tau)$ are the vertical and horizontal position errors at time τ

  VAL and HAL are the vertical and horizontal alert limits (may be infinite)

To shorten the notations, we will note as HMI (for hazardously misleading information) the event specified in Equation (1).

Fault modes

We can go further in the evaluation of the IR by developing Equation (1) using the formula of total probability. Note that here the fault modes form a partition of the state of the measurements:

$$P(\text{fault } k,\ \text{fault } j) = 0 \text{ if } k \neq j$$

$$\sum_k P(\text{fault } k) = 1$$

The expression for IR becomes:

$$IR = \sum_k P(HMI, \text{fault } k)$$

Effect of temporal exposure on monitored faults

As described in [6], we can account for the effect of temporal exposure using two distinct bounds. For the modes that are monitored, we write that the probability of HMI over the interval can be replaced by the sum of the instantaneous HMI at discrete times t_0 to t_{Nes}:

$$P(HMI, \text{fault } k) = \sum_{i=0}^{N_{ES}-1} P(HMI \text{ at } t_i, \text{fault } k)$$

The determination of the number of N_{ES} is discussed in [6]. The definition of the HMI in Equation (1) implies that if there are no HMI events at a discrete temporal grid separated by TTA, then there is no HMI over the interval I_{EXP}. As a result, an upper bound on N_{ES} is given by T_{EXP}/TTA (this is 360 – 450) for RNP and 25 for LPV. Other methods can be used to exploit the temporal correlation of the nominal error to further reduce N_{es}. Let us now sum over all fault modes k that will be monitored:

$$\sum_{\text{monitored faults}} P(HMI, \text{fault } k) = \sum_{\text{monitored faults}} \sum_{i=0}^{N_{ES}-1} P(HMI \text{ at } t_i, \text{fault } k) = \sum_{i=0}^{N_{ES}-1} \sum_{\text{monitored faults}} P(HMI \text{ at } t_i, \text{fault } k)$$

After this step, we can see that the term inside the sum over the time steps is an instantaneous integrity risk. An important consequence of this relationship is that we now only need to classify the fault states as they affect time t_i only. In other words, we can write

$$\sum_{\text{monitored faults}} P(HMI \text{ at } t_i, \text{fault } k) = \sum_{\text{monitored faults}} P(HMI \text{ at } t_i, \text{fault } k \text{ present at } t_i)$$

Note that in the previous equation we have changed the partition of the fault states. We further develop this equation to make appear the prior probability of a fault being present at t_i:

$$\sum_{\text{monitored faults}} P(HMI \text{ at } t_i, \text{fault } k \text{ present at } t_i) =$$

$$\sum_{\text{monitored faults}} P(HMI \text{ at } t_i | \text{fault } k \text{ present at } t_i) P(\text{fault } k \text{ present at } t_i)$$

Now, if we assume a frozen geometry at time t_i, we can write that:

$$\sum_{\text{monitored faults}} P\left(HMI, \text{fault } k\right) = N_{ES} \sum_{\text{monitored faults}} P\left(HMI \text{ at } t_i | \text{fault } k \text{ present at } t_i\right) p_{fault,k}$$

where

$$p_{fault,k} = P\left(\text{fault } k \text{ present at } t_i\right)$$

Upper bound on integrity risk

Using the previous relationships, an upper bound on the integrity risk IR is then given by:

$$IR \leq N_{ES} \sum_{\text{monitored faults}} P\left(HMI \text{ at } t_i | \text{fault } k \text{ present at } t_i\right) p_{fault,k} + \sum_{\text{not monitored faults}} p_{fault,k}\left(T_{EXP}\right) \qquad (2)$$

where

$$p_{fault,k}\left(T_{EXP}\right) = P\left(\text{fault } k \text{ at any time over } I_{\exp}\right)$$

CONTINUITY REQUIREMENT

The effect of the temporal exposure extends to the continuity requirement in a similar way. For our purposes, here we will focus on the probability of alert: this is the probability that the user algorithm is unable to find a solution whose set of measurements appear to be consistent with the nominal error model (the consistency being defined by the solution separation statistic, as in defined in [1], [2], [3]). The equation for the probability of alert can be written as:

$$P\left(Alert\right) \leq N_{ES,CONT} \sum_{\text{faults } \in J} P\left(\text{Alert at } t_i | \text{fault } k \text{ present at } t_i\right) p_{fault,k} + \sum_{\text{faults } \notin J} p_{fault,k}\left(T_{EXP,CONT}\right)$$

$$(3)$$

where J is the set of candidates for exclusion (including the all-in-view case, which is the fault-free candidate) and $N_{ES,CONT}$ is the number of effective samples for continuity (which may be different than N_{ES}, the number of effective samples for integrity [6]). As with the integrity, the probability of alert is expressed per unit of time hour, so there is a choice to be made for the exposure time $T_{EXP,CONT}$. In [7] and [11], it is argued that assuming an overall continuity requirement of 10^{-5}/hour and with the expected values of the integrity support data, it is sufficient to consider single faults. For the algorithm we will consider that it is sufficient to have:

$$N_{ES,CONT} \sum_{\text{faults } \in J} P\left(\text{Alert at } t_i | \text{fault } k \text{ present at } t_i\right) p_{fault,k} \leq P_{FA} \qquad (4)$$

where $P_{FA} = 5 \cdot 10^{-7}$ (note that this probability includes the alerts due to a failed exclusion)

In the description of the exclusion function below, we will see how this inequality is linked to the definition of the fault detection thresholds (under each of the subsets).

INTEGRITY SUPPORT DATA

At this point it is necessary to introduce the parameters that will allow the receiver to bound the terms $p_{fault,k}$ and $p_{fault,k}(T_{EXP})$. Table 1 shows the parameters that constitute the Integrity Support Data (ISD) – note that this nomenclature has evolved to consider the fact that the integrity parameters may not be broadcast in an Integrity Support Message (ISM). Of those, the first three ones describe the nominal model. The next four ones describe the fault modes.

Table 1. *List of parameters derived from the ISD*

	Description	Source
$\sigma_{URA,i}$	standard deviation of the clock and ephemeris error of satellite *i* used for integrity	ISD + navigation data
$\sigma_{URE,i}$	standard deviation of the clock and ephemeris error of satellite *i* used for accuracy and continuity	ISD + navigation data
$b_{nom,i}$	maximum nominal bias for satellite *i* used for integrity	ISD
$R_{sat,i}$	fault rate of fault in satellite *i*	ISD
$R_{const,j}$	rate of a fault affecting more than one satellite in constellation *j*	ISD
$MFD_{sat,i}$	Mean time to notify for satellite *i*	ISD
$MFD_{const,i}$	Mean time to notify for constellation *j*	ISD

We note that the parameters included in Table 1 might be dependent on the frequency combination (single frequency or dual frequency), or on the mode of operation (horizontal guidance or vertical guidance) -this will be specified in the descriptions of the ISD for each constellation. We have:

$$P_{sat,i} = R_{sat,i} \cdot MFD_{sat,i}$$

$$P_{const,j} = R_{const,j} \cdot MFD_{const,j} \qquad (5)$$

where $P_{sat,i}$ is the prior probability of fault in satellite i at a given time and $P_{const,j}$ is the prior probability of a fault affecting more than one satellite in constellation j at a given time. We note that constellation providers might choose to either provide prior probability or a rate (more generally, out of the three fault parameters – rate, probability, and MFD-, two must be provided). The faults described by the ISD should be treated as independent events.

UPPER BOUND ON THE PROBABILITY OF A FAULT AFFECTING A USER OVER A GIVEN INTERVAL

In this section we go over the methods to evaluate the second term in Equation (2). The probability that a fault affects any part of an interval I_{exp} of length T_{exp} is given by:

$$P\left(fault\ k\ \text{in interval } I_{\exp} \right) = MFD_k \cdot R_k + T_{\exp} \cdot R_k = \left(1 + \frac{T_{\exp}}{MFD_k} \right) p_{fault,k} \qquad (6)$$

where

MFD_k is the mean fault duration of a fault

R_k is the corresponding fault rate

There are two contributors in this expression: the first term is the probability that the fault is present at the beginning of the exposure interval, and the second term is the probability that it appears during the exposure interval. If fault k corresponds to one of the primary events that is described in the ISD, then Equation (6) can be applied directly. If not, we either need to compute the parameters MFD_k and R_k (or $p_{fault,k}$), or compute an upper bound using the ISD parameters.

Mean fault duration for simultaneous faults

The mean fault duration MFD_k of a fault composed of simultaneous primary faults is given by [6], [10]:

$$MFD_k = \left(\sum_{i/B_{i,k}=1} \frac{1}{MFD_{event,i}} \right)^{-1} \qquad (7)$$

where $B_{i,k}$ is an function indicating whether the primary event i is included in fault k

(in this formula, the label 'event' replaces both 'sat' and 'const' and refers to the primary events defined in the ISD)

A consequence of this formula is that the probability of fault k in the interval I is given by:

$$p_{fault,k}(T) = \left(1 + T \sum_{i/B_{i,k}=1} \frac{1}{MFD_{event,i}}\right) p_{fault,k}$$

(8)

The state probability $p_{fault,k}$ is derived as in [1], [2], and [3].

Probability of events occurring in the same time interval

To simplify the computations, we define $P_{event,i}(T)$ the probability that one of the primary events has occurred during the interval of length T. According to Equation (6), the probability is noted:

$$P_{event,i}(T) = \left(1 + \frac{T}{MFD_{event,i}}\right) P_{event,i}$$

(9)

Similarly we define the probability $p_{fault,k}(T)$ that the set of faults defined by $B_{i,k}$ above has occurred at any point during an interval T. We have:

$$p_{fault,k}(T) \le \prod_{i=1}^{N_{events}} P_{event,i}^{B_{i,k}}(T)\left(1 - P_{event,i}(T)\right)^{1-B_{i,k}} = \overline{p}_{fault,k}(T)$$

(10)

These formulas can be used to conservatively determine the set of faults to be monitored.

DETERMINATION OF FAULT MODES

The ISD does not specify explicitly which fault modes need to be monitored or their corresponding prior probabilities. (This is because this list is potentially dependent on the user geometry.) This determination must be made by the receiver based on the contents of the ISD, which specifies the probabilities of events that can be treated as independent.

The determination of this list must consider the computational load and the effect on performance. These two objectives are not always competing.

This paragraph provides one possible method to establish a list of event combinations (the fault modes) to be monitored. The objective is to make sure that the sum of the probabilities of the modes that are not monitored do not exceed a pre-defined fraction of the total integrity budget (P_{THRES}). The list of fault modes that need to be monitored described here is only sufficient (there could be shorter lists that also meet the integrity requirements). The approach consists in moving fault modes from the list of not-monitored to the monitored list one by one until the remaining modes have a total probability below a pre-defined threshold. We want:

$$\sum_{k \text{ not monitored}} \overline{p}_{fault,k}(T_{EXP}) \le P_{THRES}$$

(11)

This approach is practical because we know that the sum of all the probabilities is one:

$$\sum_{k=0}^{2^{N_{events}}} \bar{p}_{fault,k}\left(T_{EXP}\right) = 1 \qquad (12)$$

(The sum above goes through all the possible combinations of the primary events, of which there are $2^{N_{events}}$.)

The condition expressed in Equation (11) can therefore be written:

$$\sum_{k \text{ monitored}} \bar{p}_{fault,k}\left(T_{EXP}\right) \geq 1 - P_{THRES} \qquad (13)$$

This way, it is only necessary to compute the probabilities (using Equation (10)) of the modes that will be monitored. We then need to decide the order in which the faults are considered. For fault detection only, the choice of modes is not critical, and the order included in [3] is adequate. For fault detection and exclusion, because we might be pre-allocating the integrity risk among exclusion choices, the choice of monitored faults can have a large impact on performance. For this reason, it is preferable to order the monitored modes from stronger to weaker (for all exclusion candidates). The order defined below was intended to work well with the ISD parameters defined in the draft ICAO SARPS Annex 10 (which we will refer as default ISD).

The order can be defined as follows:

1) the modes for which $\bar{p}_{fault,k}\left(T_{EXP}\right) > PHMI$, where *PHMI* is the predefined IR requirement. These modes need to be monitored. For the default ISD settings, this will include all the single satellites fault modes, and the constellation fault modes such that $\bar{p}_{fault,k}\left(T_{EXP}\right) > PHMI$ (Galileo, Beidou, and GLONASS)
2) the dual satellite faults (except GPS-GPS, which are very weak in the case of a Galileo constellation exclusion)
3) the satellite-constellation wide fault modes corresponding the constellation wide fault modes that are included in step 1 from stronger geometry to weaker geometry (GPS satellite – Galileo constellation, in the case of the default ISD parameters)
4) the remaining constellation wide fault modes with $\bar{p}_{fault,k}\left(T_{EXP}\right) > 0$

For the cases where faults with a higher degree need to be considered (where we define the degree as the number of primary events forming the composite fault mode), the faults can be removed by order of increasing degree. Within one degree it is recommended to start with the faults composed of independent satellite faults.

In this process, if a fault cannot be monitored (either in the all-in-view case or the exclusion options), it is not included in the list of fault modes and we move to the next one. Each fault mode *k* is characterized by the set of indices corresponding to the measurements that are not affected by the fault, which will be noted *idx$_k$*. The set *idx$_0$* corresponds to the full set of indices.

The integrity risk from the fault modes that are not monitored is bounded by $\bar{P}_{fault,not\ monitored}$, which is defined as:

$$\overline{P}_{fault,\ not\ monitored} = \sum_{k\ not\ monitored} \overline{p}_{fault,k}\left(T_{EXP}\right) \tag{14}$$

Note that it is possible to have a finer bound on $P_{fault,not\ monitored}$ by using Equation (8).

Fault consolidation

Fault consolidation (or grouping) can be used to reduce the number of monitored modes without a significant impact on performance.

After establishing the initial list above, the algorithm consolidates multiple satellite faults from the same constellation with the constellation wide fault. This is done as follows: for each constellation j, we note k_j the fault mode corresponding to the fault of constellation j only, and C_j the set of fault modes that are formed of satellite faults included in constellation j (and included in the list established above). If the following inequality holds:

$$\sum_{k \in C_j} p_{fault,k} \le F_C\, p_{fault,k_j} \tag{15}$$

where F_C is a tunable parameter, the fault modes in Cj are removed from the list and the probability of fault mode k_j is updated as follows:

$$p_{fault,k_j}^{\ (updated)} = p_{fault,k_j} + \sum_{k \in C_j} p_{fault,k} \tag{16}$$

Quasi-fixed subset list

As an alternative to the previous algorithm, the following orderworks well for the proposed default values:

1. faults for which $\overline{p}_{fault,k}\left(T_{EXP}\right) > PHMI$
2. dual satellite faults from two different constellations
3. dual constellation–satellite faults for the constellations for which $\overline{p}_{fault,k}\left(T_{EXP}\right) > PHMI$ (as few as possible, and choosing the strongest geometries under the weaker exclusion option)

The dual satellite faults within one constellation are grouped within the corresponding constellation fault.

When determining how many of the subsets in 3) need to be included, it is possible to use the less conservative bounds on the $\overline{p}_{fault,k}\left(T_{EXP}\right)$ described above.

Effect of exclusion function on the list of faults

The list of faults to be monitored is the list determined above. The new sets of indices used to compute the fault tolerant position solution will be given by:

$$idx_j \cap idx_k \tag{17}$$

where j refers to the exclusion candidate and k to the fault mode.

However, now this set of subsets will contain elements that are identical. We reduce this list by identifying a set of unique elements, which are re-indexed from $k = 0$ to $N_{fault\_modes,j}$ where $N_{fault\_modes,j}$ is the new number of fault modes (after identifying the identical sets). We label the new sets of indices $idx_k^{(j)}$.

For example, let us suppose that there are 6 satellites in view {1,2,3,4,5,6}, and that satellite 2 was excluded. If the original subsets k and k' were: {1,2,3,4,5} and {1,3,4,5} and satellite 2 is excluded, the resulting subsets from applying (17) will be identical. We can therefore group them.

The probabilities of the new list of fault modes will need to account for the grouping. Therefore, the probability of fault for each mode is given by:

$$p_{fault,k}^{(j)} = \sum_{k'|idx_k^{(j)}=idx_{k'}\cap idx_j} p_{fault,k'} \tag{18}$$

The index $k=0$ corresponds to the new all-in-view solution (that is, we have $idx_j = idx_0^{(j)}$).

EXCLUSION FUNCTION

We will not repeat the description of the computation of the fault tolerant estimates and the associated covariance, as these remain the same as in [1], [2], and [3]. We will however briefly go over the exclusion function and how it affects the evaluation of both the integrity and continuity.

The first step of the exclusion algorithm consists in finding a subset of measurements that is consistent (among the list of candidates, which we have labeled J above). A subset is determined to be consistent if it passes the solution separation tests. As shown in [1], it is possible to avoid testing all possible subsets by checking the chi-square statistic of each of the subsets. Because this statistic is an upper bound on the maximum solution separation statistic, the subset with the smallest chi-square statistic is very likely to be consistent, and thus a good candidate for exclusion. In this algorithm, any set among J that passes the consistency checks can be chosen.

Effect on integrity

The effect of the exclusion function is to choose one of the subset solutions $\hat{x}^{(j)}$ as the new position solution. Let us now go back to our upper bound on the integrity risk in Equation (2). It is important to note that this expression must consider the exclusion process. The HMI event includes all the possible exclusion options (see the paragraph "*Integrity allocation across exclusion options*" for the determination of the exclusion options). We can write as:

$$HMI = \bigcup_{j\in J}\left(HMI \ \& \ \hat{x} = \hat{x}^{(j)}\right)$$

Using this decomposition and the fact that the probability of the union of a set of events is bounded by the sum of the probability of each event, we have the upper bound:

$$IR \leq N_{es} \sum_{j \in J} \sum_{\text{monitored faults}} P\left(HMI \,\&\, \hat{x} = \hat{x}^{(j)} \mid \text{fault } k \text{ present at } t_i\right) p_{fault,k} \\ + \sum_{\text{not monitored faults}} p_{fault,k}\left(T_{EXP}\right)$$

(19)

This is the inequality that is at the root of the allocation of integrity across the exclusion candidates. We do point out that there are alternatives to this upper bound that can be used [11].

One of the conditions to choose candidate j is that the solution separations statistics for each of the monitored modes is below the detection threshold. This means that we can use the same bounding approach as in [1]. It does mean, however, that we must decide an allocation to each of the exclusion candidates, and that allocation must be independent of the measurements. The parameter that regulates this allocation will be labeled ρ_j.

Effect on continuity

The effect of the exclusion function on continuity can be seen in Equation (3). To go further, we will use the upper bound:

$$P\left(\text{Alert}\mid\text{fault } j\right) \leq P\left(\hat{x} = \hat{x}^{(j)}\right)$$

(20)

A condition for $\hat{x} = \hat{x}^{(j)}$ is that the solution separation tests between j (as the new all-in-view solution) and the subset solutions pass. This condition will set a constraint on the solution separation thresholds that will be exploited in the next section.

PROTECTION LEVEL EQUATION

The equations defining the protection levels with fault exclusion are formally identical to the fault detection protection levels as shown in [1]. The proposed changes are meant to account for the temporal exposure [6], and the fault exclusion function. For the fault exclusion function (already present in [3]), the changes are:

- the set of satellites that is considered (the subset determined to be consistent is now the all-in-view)

- the integrity allocation (which is now reduced to account for exclusion)

Horizontal Protection Level

As in [1], the horizontal protection level is computed by computing a protection level on the East and the North coordinate (q=1 and 2). Also, the script j corresponds to the exclusion candidate that has passed

the solution separation test. Based on the Equations above, we can define the $^{(j)}HPL_q$ (for q = 1 and 2) as the solution of the equation:

$$2\bar{Q}\left(\frac{^{(j)}HPL_q - {}^{(j)}b_q^{(0)}}{^{(j)}\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\ modes,j}} {}^{(j)}p_{fault,k}\bar{Q}\left(\frac{^{(j)}HPL_q - {}^{(j)}T_{k,q} - {}^{(j)}b_q^{(k)}}{^{(j)}\sigma_q^{(k)}}\right) =$$

$$\rho_j \frac{PHMI_{HOR}}{2N_{ES}}\left(1 - \frac{\bar{P}_{fault,not\ monitored}}{PHMI_{VERT} + PHMI_{HOR}}\right) \tag{21}$$

where:

$\bar{P}_{fault,not\ monitored}$ is a bound on $P_{fault,not\ monitored}$ (for example computed as described in Equation (14) to account to T_{EXP}),

$^{(j)}b_q^{(k)}$ are the nominal biases for the subset solution using the subset $idx_k^{(j)}$

$^{(j)}\sigma_q^{(k)}$ are the standard deviation of the subset position solution using the subset $idx_k^{(j)}$

$^{(j)}T_{k,q}$ is the solution separation thresholds for the exclusion candidate j. and the subset position solution using $idx_k^{(j)}$. We will go over their definition below.

$\bar{Q}$ is a modified normal tail cdf, which we will discuss below.

$PHMI_{VERT}, PHMI_{HOR}$ are the integrity allocations to the vertical and horizontal coordinates (see [1],[2],[3])

N_{ES} is the number of effective samples discussed in the first section of this paper and in [6]

ρ_j is a parameter adjusting the integrity allocation defined in the previous section. The set of parameters ρ_j is selected without the knowledge of the measurements (in particular, it must be independent of the exclusion option) and be such that:

$$\sum_{j=0}^{N_{fault\ modes}} \rho_j = 1$$

If the exclusion process results in the choice of j as the consistent solution, the HPL is given by:

$$^{(j)}HPL = \sqrt{^{(j)}HPL_1^2 + {}^{(j)}HPL_2^2} \tag{22}$$

Vertical Protection Level

Similarly, the Vertical Protection Level $^{(j)}VPL$ satisfies the following equation:

$$2\bar{Q}\left(\frac{^{(j)}VPL - {}^{(j)}b_3^{(0)}}{^{(j)}\sigma_3^{(0)}}\right) + \sum_{k=1}^{N_{fault\ modes,j}} {}^{(j)}p_{fault,k}\bar{Q}\left(\frac{^{(j)}VPL - {}^{(j)}T_{k,3} - {}^{(j)}b_3^{(k)}}{^{(j)}\sigma_3^{(k)}}\right) =$$

$$\rho_j \frac{PHMI_{VERT}}{N_{ES}}\left(1 - \frac{\bar{P}_{fault,not\ monitored}}{PHMI_{VERT} + PHMI_{HOR}}\right)$$

(23)

Integrity allocation across exclusion options (ρ_j)

The choice of the parameters ρ_j will be dependent on the continuity requirements and the receiver capabilities. One possible approach is to pre-select (that is, before knowing the measurements) the set of exclusion options that will be attempted, *J*.   This set will be a subset of all the monitored fault modes and includes the all-in-view (*j*=0). For example, above we argued that in Horizontal ARAIM, it is likely that this set would only need to include all single satellite faults and constellation-wide faults that must be monitored (P$_{const}$ equal or larger than 10^{-7}). For the indices *j* corresponding to these exclusion options, we can set:

$$\rho_j = \frac{1}{N_{exc} + 1}$$

(24)

where N_{exc} is the number of pre-selected exclusion options (excluding the all-in-view). Note that the PLs above will only be defined for the pre-selected exclusion options.

If the receiver has sufficient computational power, the HPL can be computed by solving the equation (as suggested in [12]):

$$\sum_{j \in J}\left[2\bar{Q}\left(\frac{HPL_q - {}^{(j)}b_q^{(0)}}{^{(j)}\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\ modes,j}} {}^{(j)}p_{fault,k}\bar{Q}\left(\frac{HPL_q - {}^{(j)}T_{k,q} - {}^{(j)}b_q^{(k)}}{^{(j)}\sigma_q^{(k)}}\right)\right] =$$

$$\frac{PHMI_{HOR}}{2N_{ES}}\left(1 - \frac{\bar{P}_{fault,not\ monitored}}{PHMI_{VERT} + PHMI_{HOR}}\right)$$

(25)

Such approach corresponds to a choice of the allocations ρ_j that makes all $^{(j)}HPL_q$ equal under all exclusion options. It will make the receiver more robust to faults, in the sense that it minimizes the worst-case PL in the case of a fault. It might however make it less robust to outages, in the sense that the PL will be worse when there is an outage and no fault, (because the PL will be close a FD PL corresponding to a geometry missing the satellites due a worst-case fault –in addition to the outage).

Threshold computation

The detection thresholds are defined by:

$$^{(j)}T_{k,q} = {}^{(j)}K_{fa,q}^{(k)} \; {}^{(j)}\sigma_{ss,q}^{(k)} \tag{26}$$

where ${}^{(j)}\sigma_{ss,q}^{(k)}$ is the standard deviation of the solution separation statistic between the candidate subset j and the subset k. To meet the probability of Alert constraint, the containments ${}^{(j)}K_{fa,q}^{(k)}$ can be constrained by Equation (4) and (20). We get:

$$N_{ES,CONT} \sum_{j \in J} p_{fault,j} \sum_{k=0}^{N_{fault\,modes,j}} \sum_{q=1}^{3} 2Q\left({}^{(j)}K_{fa,q}^{(k)}\right) \le P_{FA} \tag{27}$$

For the baseline algorithm, we propose to further simplify the constraint by exploiting the fact that $\sum_{j \in J} p_{fault,j} \le 1$. It is sufficient to have:

$$\sum_{k=0}^{N_{fault\,modes,j}} \sum_{q=1}^{3} 2Q\left({}^{(j)}K_{fa,q}^{(k)}\right) \le \frac{P_{FA}}{N_{ES,CONT}}$$

For this to be true, it is sufficient to have

$$^{(j)}K_{fa,1}^{(k)} = {}^{(j)}K_{fa,2}^{(k)} = Q^{-1}\left(\frac{P_{FA\_HOR}}{4N_{fault\,modes}N_{ES,CONT}}\right) \tag{28}$$

$$^{(j)}K_{fa,3}^{(k)} = Q^{-1}\left(\frac{P_{FA\_VERT}}{2N_{fault\,modes}N_{ES,CONT}}\right)$$

with the constraint $P_{FA} = P_{FA\_HOR} + P_{FA\_VERT}$.

Modified Q function for the PL equation

The *Q* function is defined as:

$$Q(u) = \frac{1}{\sqrt{2\pi}} \int_{u}^{+\infty} e^{-\frac{t^2}{2}} dt \tag{29}$$

In the PL equation, we use a modified version, which is labeled $\bar{Q}$. It is defined by

$$\bar{Q}(u) = Q(u) \text{ for } u > 0$$
$$\bar{Q}(u) = 1 \text{ for } u \le 0 \tag{30}$$

This modification ensures that a smaller standard deviation always results in a smaller contribution of the integrity risk.

Bounding the effect of imprecision in the Q function

Assume that there is an approximation function to $\bar{Q}$, denoted as $\bar{Q}'$ with the knowledge of its relative accuracy A. This section argues that the protection level equation using $\bar{Q}'$ with a modified requirement results in a valid PL.

$$2\bar{Q}\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\,modes}} p_{fault,k}\bar{Q}\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right) = \frac{PHMI_q}{N_{ES}}\left(1 - \frac{P_{fault,not\,monitored}}{PHMI}\right)$$

(31)

Here $IR(PL_q)$ is the IR requirement usually given by $\frac{PHMIq}{N_{ES}}\left(1 - \frac{P_{not\,monitored}}{PHMI}\right)$. Therefore, the protection level equation is:

$$IR(PL_q) > 2\bar{Q}\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\,modes}} p_{fault,k}\bar{Q}\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)$$

(32)

If we replace the true $\bar{Q}$ with approximation $\bar{Q}'$ where $\frac{\bar{Q}}{\bar{Q}'} < A$ and A is known i.e. 10% maximum relative error à $A = 1.1$ then:

$$2\bar{Q}\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\,modes}} p_{fault,k}\bar{Q}\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)$$

$$< A \times \left[2\bar{Q}'\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\,modes}} p_{fault,k}\bar{Q}'\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)\right].$$

(33)

The requirement $IR(PL_q)$ can be rewritten to account for inaccuracy of the $\bar{Q}'$, so we need to reduce the requirement from $IR(PL_q)$ to $\frac{IR(PL_q)}{A} = IR'(PL_q) < IR(PL_q)$ which will have very small impact on the PL_q obtained - it will increase from the value which solves (32).

Meeting the following rule (using the approximation $\bar{Q}'$):

$$IR'(PL_q) > 2\bar{Q}'\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{k=N} p_{fault,k}\bar{Q}'\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)$$

(34)

ensures that the global requirement (32) is met since:

$$IR(PL_q) = A \times IR'(PL_q) > A \times 2\bar{Q}'\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + A \times \sum_{k=1}^{k=N} p_{fault,k}\bar{Q}'\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)$$

$$> 2\bar{Q}\left(\frac{PL_q - b_q^{(0)}}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{k=N} p_{fault,k}\bar{Q}\left(\frac{PL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}}\right)$$

Therefore, it is sufficient to use the relative accuracy as a performance indicator for the assessment of any approximation function $\bar{Q}'$.

REDUCING THE COMPUTATIONAL LOAD

With the proposed default ISD parameters and the proposed subset selection method, the number of terms in the PL equation will be about 100. Most of those fault modes correspond to dual satellite faults. The subsets corresponding to these fault modes usually correspond to strong geometries (at least compared to the constellation wide fault modes). The contribution of those modes to the integrity risk is therefore expected to be small. It is possible to significantly reduce the computational load by exploiting the approach described in [9]. In this description, we will assume that nominal error parameters are the same for continuity and integrity (if they are not, the equations must be adapted).

Replacing the solution separation statistic by the sum of square residuals

In [1], we showed that the square root of the sum of squared normalized residuals is a bound on the normalized solution separation, that is:

$$\forall k \quad \frac{\left| \hat{x}_q^{(k)} - \hat{x}_q^{(0)} \right|}{\sigma_{ss,q}^{(k)}} \leq \sqrt{y^T \left(W - WG \left(G^T WG \right)^{-1} G^T W \right) y} \tag{36}$$

wherewe follow the notations of [1]..

This bound implies that we do not need to evaluate the subset position solutions to check the consistency of the measurements. It also means that we must use the chi-square statistics to define the threshold [9].

Computing a Protection Level without evaluating all subset solutions

Let us now consider a subset Ω of the list of fault modes. For example, it could designate all the *n*-2 and *n*-1 subsets. In [1] we showed that when using the all-in-view least squares position solution, we have the identity:

$$\sigma_{ss,q}^{(k)2} = \sigma_q^{(k)2} - \sigma_q^{(0)2} \tag{37}$$

Now let us assume that that we have an upper bound $\bar{\sigma}_q^{\Omega}$ of all the standard deviations $\sigma_q^{(k)}$ for all the subsets k in Ω:

$$\forall k \in \Omega \quad \sigma_q^{(k)} \leq \bar{\sigma}_q^{\Omega} \tag{38}$$

For any value of L and any k in Ω we have:

$$Q\left(\frac{L - T_{k,q}}{\sigma_q^{(k)}}\right) \leq Q\left(\frac{L - K_{fa,q}^{\chi^2}\sqrt{\bar{\sigma}_q^{\Omega 2} - \sigma_q^{(0)2}}}{\bar{\sigma}_q^{\Omega}}\right) \tag{39}$$

where $K_{fa,q}^{\chi^2}$ is a containment defined by the chi-square statistic.

The important feature of this upper bound is that it does not depend on the index k. We can further write:

$$\sum_{k \in \Omega} p_{fault,k} Q\left(\frac{L - T_{k,q}}{\sigma_q^{(k)}}\right) \leq \left(\sum_{k \in \Omega} p_{fault,k}\right) Q\left(\frac{L - K_{fa,q}^{\chi^2}\sqrt{\bar{\sigma}_q^{\Omega 2} - \sigma_q^{(0)2}}}{\bar{\sigma}_q^{\Omega}}\right) \tag{40}$$

This bound shows that we can replace many terms in the PL equation by one, as long as we have an upper bound on $\sigma_q^{(k)}$.

Upper Bound on the Error Covariances of Subsets with N-2 measurements

To obtain an upper bound on the standard deviations corresponding to the dual faults, we exploit the matrix inversion lemma as described in [9]. After some algebra, we get the key relationship:

$$\sigma_q^{(J)2} = \sigma_q^{(0)2} + e^T S_{(J)} \left(P_{(JJ)}\right)^{-1} S_{(J)}^T e \tag{41}$$

where

    J refers to the subset of measurements that is removed from the all-in-view (note that this J is not related to the list of exclusion candidates)

    e is the vector projecting the position solution on the coordinate of interest

    $P_{(JJ)}$ is obtained by selecting indices J in the rows and columns of the matrix

$$P = W - WG\left(G^T W G\right)^{-1} G^T W$$

    $e^T S_{(J)}$ is composed of the indices J of $e^T S$ where $S = \left(G^T W G\right)^{-1} G^T W$

The terms in Equation (41) are byproducts of the all-in-view position solution.

After normalizing the matrix P, as described in [9], we end up with the relationship:

$$\sigma_q^{(J)2} = \sigma_q^{(0)2} + s_{norm,(J)} \left(P_{norm,(JJ)} \right)^{-1} s_{norm,(J)}^{T}$$

The new matrix P_{norm} has all ones in the diagonal.

To go further, we use the following inequality [7]:

$$s_{norm,(J)} \left(P_{norm,(JJ)} \right)^{-1} s_{norm,(J)}^{T} \leq \left| s_{norm,(J)} \right|^{2} \max \left(\lambda \left(P_{norm,(JJ)}^{-1} \right) \right)$$

(41)

Where the term λ refers to the eigenvalue of the matrix. For a subset $J=(i,j)$ with two elements, the matrix $P_{norm,(JJ)}$ will have the form:

$$P_{norm,JJ} = \begin{bmatrix} 1 & p_{norm,ij} \\ p_{norm,ij} & 1 \end{bmatrix}$$

The eigenvalues of this matrix are given by $1 + p_{norm,ij}, 1 - p_{norm,ij}$. For the subsets with two elements, we get the upper bound:

$$\max \left(\lambda \left(P_{norm,(JJ)}^{-1} \right) \right) \leq \frac{1}{1 - \max_{i \neq j} \left(p_{norm,ij} \right)}$$

The term $\left| s_{norm,(J)} \right|^{2}$ can be bounded by simply taking the two largest components of s_{norm}.

Nominal biases

The effect of the nominal biases can be bounded by using the Cauchy-Schwarz inequality, which allows us to link the position domain nominal bias to the standard deviation upper bound [9].

Results

We show a set of coverage results obtained using the baseline algorithm with and without the proposed method to reduce the computational load. The details of the scenario are described below. The simulations were obtained using MAAST. The HPL that are computed correspond to the worst-case exclusion HPL (which is also very close to the worst-case outage case).

ISD and URA settings

Parameter	GPS	Galileo
R_{const}	1e-8/h	-
R_{sat}	1e-5/h	-
MFD	1 h	*1 h*
P_{const}	1e-8	1e-4
P_{sat}	1e-5	1e-4
b_{nom0}	0.0 (DF) m	0.0 (DF) m
URA	2.4 m	6 m

User:

- 24 h with 300 s steps
- 10 by 10 degree user grid

Receiver

- $P_{HMI} = 10^{-7}$
- H-ARAIM: $P_{FA} = 5 \times 10^{-7}$
- $P_{THRES} = 8 \times 10^{-8}$

Almanacs: GPS 24 – Galileo 24 (WG-C assumptions)

	RNP
Number of effective samples	360
Exposure window	1 hour

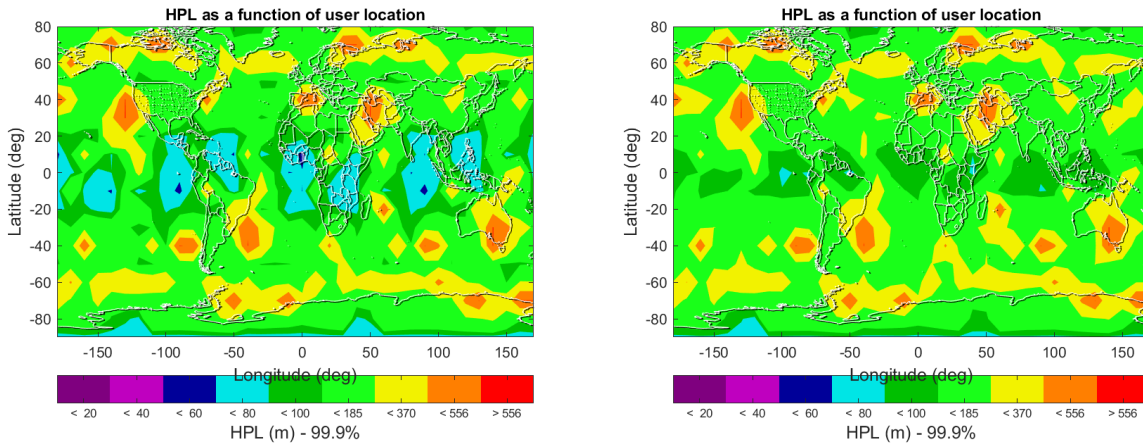*Figure 1. Simulation settings*

Figure 2. 99.9% HPL for proposed baseline algorithm with (left) and without (right) the computational reduction method
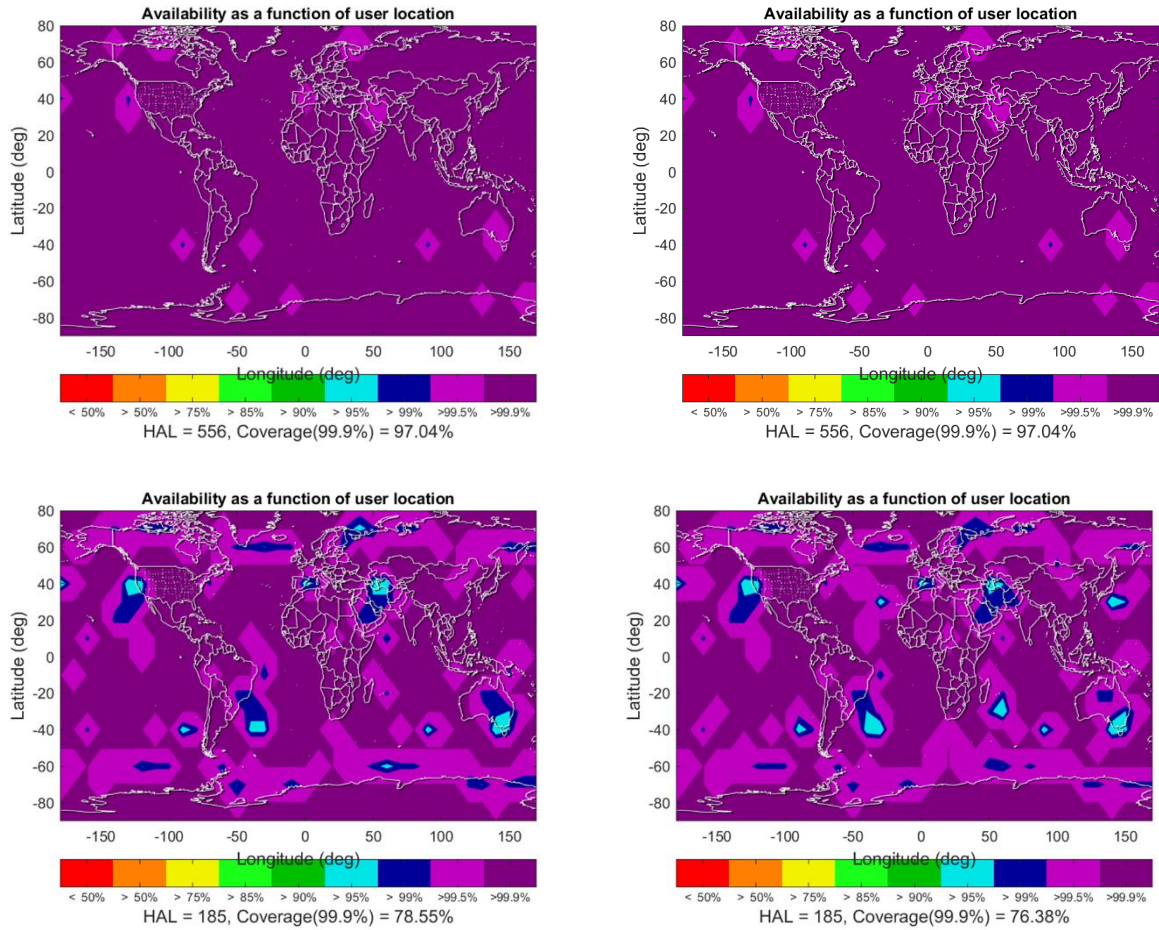
Figure 2. Coverage of 99.9 availability of HAL=556 m (top) and HAL = 185 m (bottom) for proposed baseline algorithm with (left) and without (right) the computational reduction method

These availability simulations show that the updates proposed in this paper would result in a user algorithm that provides excellent coverage of RNP, and that the proposed technique to reduce the computational load do not degrade performance significantly.

SUMMARY

We have described a set of updates to the baseline ARAIM algorithm described in [1],[2], [3]. These updates address the effect of temporal exposure on integrity and continuity, and clarify the effect of exclusion on the PL. In addition, the subset selection is modified to ensure that performance remains good under these changes. Finally, to reduce the computational complexity (which has been a concern during the development of ARAIM), we propose a technique that can drastically reduce the number of subsets while maintaining availability..

REFERENCES

[1] Blanch, J., Walter, T., Enge, P., Lee, Y., Pervan, B., Rippl, M., Spletter, A., Kropp, V., "Baseline Advanced RAIM User Algorithm and Possible Improvements," IEEE Transactions on Aerospace and Electronic Systems, Volume 51, No. 1, January 2015.

[2] Working Group C, ARAIM Technical Subgroup, Milestone 3 Report, February 26, 2016. Available at:

http://www.gps.gov/policy/cooperation/europe/2016/working-group-c/

http://ec.europa.eu/growth/tools-databases/newsroom/cf/itemdetail.cfm?item_id=8690

[3] WG-C Advanced RAIM Technical Subgroup Reference Airborne Algorithm Description Document, available at:

http://web.stanford.edu/group/scpnt/gpslab/website_files/maast/ARAIM_TSG_Reference_ADD_v3.1.pdf

[4] https://gps.stanford.edu/resources/tools/maast

[5] Phelts, R. Eric, Blanch, Juan, Gunning, Kazuma, Walter, Todd, Enge, Per, "Assessment of Aircraft Banking on ARAIM Performance," *Proceedings of the 31st International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2018)*, Miami, Florida, September 2018, pp. 2632-2641.

[6] Milner, Carl, Pervan, Boris, Blanch, Juan, Joerger, Mathieu, "Evaluating Integrity and Continuity Over Time in Advanced RAIM," *2020 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, Portland, Oregon, April 2020, pp. 502-514.

[7] Joerger, Mathieu, Zhai, Yawei, Martini, Ilaria, Blanch, Juan, Pervan, Boris, "ARAIM Continuity and Availability Assertions, Assumptions, and Evaluation Methods," *Proceedings of the 2020 International Technical Meeting of The Institute of Navigation*, San Diego, California, January 2020, pp. 404-420. https://doi.org/10.33012/2020.17152

[8] Blanch, Juan, Walter, Todd, Enge, Per, "Fixed Subset Selection to Reduce Advanced RAIM Complexity," *Proceedings of the 2018 International Technical Meeting of The Institute of Navigation*, Reston, Virginia, January 2018, pp. 88-98.

[9] J. Blanch and T. Walter, "Fast Protection Levels for Fault Detection With an Application to Advanced RAIM," in IEEE Transactions on Aerospace and Electronic Systems, vol. 57, no. 1, pp. 55-65, Feb. 2021, doi: 10.1109/TAES.2020.3011997.

[10] Blanch, J., Gunning, K., Walter, T., de Groot, L., and Norman, L. "Solution separation-based FD to mitigate the Effects of Local Threats on PPP Integrity" Proceedings of the IEEE/ION Position Location and Navigation Symposium (PLANS), Portland, Oregon, April 2020.

[11] Blanch J., Walter, T., and Enge, P.," Protection Levels after Fault Exclusion for Advanced RAIM," *NAVIGATION, Journal of The Institute of Navigation*, Vol. 64, No. 4, Winter 2017, pp. 505-513.

[12] Joerger, M., and B. Pervan. "Fault Detection and Exclusion Using Solution Separation and Chi-Squared RAIM." *IEEE Transactions on Aerospace and Electronic Systems*. 52.2. (2016).