WG-C Advanced RAIM Technical Subgroup Reference Airborne Algorithm Description Document

Version 3.1

June 20, 2019

List of main changes with respect to 3.0:

- Corrected σ_{noise} term to match AAD-B error model. As a consequence, the numerical values in the example have changed.
- Added Table 3 with proposed navigation parameters for RNP
- Refined and corrected PL approximation defined in Appendix B

The purpose of this document is to describe the airborne algorithm that is used in the ARAIM availability simulations within the WG-C Advanced RAIM Technical subgroup (ATSG). This document is an evolution of Annex A in [18]. It will be updated whenever there will be a change that has been agreed to by the group. The availability results should record the version number that has been used, the Integrity Support Message content, and the parameter settings. The starting point of the reference algorithm is the one described in [1].

1. INTRODUCTION

The GPS Evolutionary Architecture Study (GEAS) outlined an Advanced RAIM concept in the GEAS Phase II report [2], which has been further developed within the Working Group C ARAIM Technical subgroup (ARAIM SG) [3]. The integrity data used by the airborne receiver is contained in the Integrity Support Message (ISM) that is determined on the ground and broadcast to the airborne fleet [3], [4].

Since the GEAS Phase II Report [2], it has become apparent that multiple simultaneous faults cannot be ruled out, and therefore might need to be mitigated by the airborne receiver. The user algorithm described in [2] only covered the single fault case. Although it was indicated that the algorithm could be generalized to multiple failures, the exact implementation was not made explicit. Methods to compute the Protection Levels with threat models including multiple faults have been described in [5], [6], [7]. The present document describes each step of an ARAIM user algorithm based on these references and [1]. The primary focus of ARAIM is on vertical guidance. However, there is interest in applying ARAIM to improve horizontal navigation. This version describes how to set the algorithm input parameters for horizontal navigation.

Section 2 describes some of the performance requirements that need to be met by the ARAIM user algorithm, and motivates the need for additional availability criteria. Section 3 describes how the ISM should be interpreted by the user receiver and its relationship to the navigation requirements. Section 4 describes the main elements of the reference user algorithm step by step for ARAIM, and is an extension of the one described in the GEAS Phase II Report [2], including elements of [5], [6], and [7]. Section 5 proposes a method to compute the Protection Levels when exclusion is implemented. Section 6 specifies the simulation conditions that were used to evaluate ARAIM coverage in [18] and [21].

2. NAVIGATION REQUIREMENTS

Requirements for LPV-200 and LPV-250

The target operational level for ARAIM is LPV-200 [8], which is a relatively new operation and one that is incompletely specified in the ICAO Standards And Recommended Practices (SARPs) [9]. Currently, LPV-200 is only provided by SBAS. The SARPs contain both requirements and guidance material on the desired operational performance, including positioning performance, continuity, and availability. However, ARAIM will have different characteristics than current SBAS, and it is important to understand how these differences may affect operational behaviour and the feasibility of meeting LPV-200 requirements. In particular, there is a concern that the test statistics in ARAIM, while protecting against errors exceeding the VAL, could allow large errors to remain undetected (for vertical guidance, it is not sufficient to have position errors below the VAL). Therefore, it is necessary to understand the operational requirements of LPV-200 and ensure the final ARAIM algorithm addresses these concerns.

For continuity, the SARPs specify a continuity risk requirement of 8×10^{-6} per 15 s. For ARAIM, the airborne algorithm tests have a finite probability of false alert, which can cause a loss of continuity. For this reason, a fraction P_{FA} of the total continuity budget is allocated to the false alerts due to the airborne algorithm. For typical fault rate values, the probability of loss of continuity due to a fault is a fraction of the continuity requirement (see Appendix E).

The SARPs describe four positioning performance criteria:

- 4 m, 95% accuracy (vertical only);
- 10 m, 99.99999% fault-free accuracy (vertical only);
- 15 m, 99.999% Effective Monitoring Threshold (EMT) (vertical only); and
- 35 m vertical, 40 m horizontal, 99.99999% limit on the position error, (i.e., the VPL and HPL have to be below a VAL of 35 m and a HAL of 40 m respectively).

Two of the criteria: 95% accuracy and PL are described in Chapter 3 of Annex 10, Volume 1, of the ICAO SARPs [9]. The other two criteria: fault-free accuracy and EMT, are only described in the guidance material in Attachment D to Annex 10 which also provides more information on the previous two criteria. For the Wide Area Augmentation System (WAAS), it was determined by the Federal Aviation Administration (FAA) that if the VPL requirement is met, the other requirements in the vertical domain are also all met. This is because of the inherent accuracy of WAAS and that the VPL is driven by rare fault-modes. Any condition that supported a VPL below 35 m, also assured that the accuracy requirements and EMT would be met.

ARAIM will have different error characteristics than SBAS. Unlike any SBAS currently implemented, ARAIM makes use of the dual-frequency ionosphere-free pseudorange combination. Additionally, ARAIM will not use SBAS differential corrections (at least in the offline architecture). Therefore, it will likely have worse accuracy than current SBAS systems. Further, its method of error detection may allow fault modes to create larger position errors before they are identified and removed. Thus, conditions that support an ARAIM VPL below 35 m may not always lead to error characteristics that support LPV-200 operations.

Therefore, we introduce two additional real-time tests in the aircraft to ensure that every supported condition has error characteristics that meet the intent of the SARPs. Specifically an accuracy test and an EMT test are described in Section 4. A single accuracy test assures that both the 4 m 95% and the 10 m 99.99999% test are met (the tests are of identical form, but the 10 m test is more stringent). The EMT test prevents faults that are not large enough to ensure detection from creating vertical position errors greater than 15 m more often than 0.001% of the time.

The requirements for LPV-250 are less stringent than LPV-200. The positioning criterion is given by:

- 50 m vertical 40 m horizontal 99.99999% limit on the position error, (i.e., the VPL and HPL have to be below a VAL of 50 m and a HAL of 40 m respectively).

Horizontal ARAIM navigation requirements for RNP

RNP has multiple levels of performance [10]. RNP ‘x’ requires that the aircraft be positioned within ‘x’ nautical miles (NM) of the estimated position. For RNP 0.1 the true aircraft position must be within 0.1 NM of the estimated position. More specifically, the number after RNP specifies the 95% bound on the Total System Error (TSE), which is the combination of Flight Technical Error (FTE) and Navigation System Error (NSE). Further, RNP also specifies that 99.999% of the time, TSE shall be contained within twice the specified number. Thus, for RNP 0.1 95% of TSE values should be less than 0.1 NM and 99.999% of TSE values should be less than 0.2 NM. The availability of RNP is a function of the FTE, which is aircraft dependent. Therefore, to assess RAIM and ARAIM performance we need to make conservative assumptions on the FTE. NSE is typically allocated half of the budget (this is conservative as FTE is typically well below 100 m 95%). The corresponding requirement can be viewed as 95% of NSE should be less than 0.05 NM (~93 m) and 99.999% of NSE should be less than 0.1 NM (~185 m). Although the integrity requirement is specified at the $1 - 10^{-5}$ level, RAIM calculates this bound at the 10^{-7} level for comparison against the 99.999% NSE requirement. (To be consistent with RAIM, ARAIM will to the same.)

3. INTEGRITY SUPPORT MESSAGE AND RELATION TO NAVIGATION REQUIREMENTS

In this section, we describe how the ISM should be processed in order to meet the integrity and continuity requirements. Table 1 shows the parameters that are derived from the ISM (how they are computed will depend on the ISM format).

Table 1. List of parameters derived from the ISM

	Description	Source
$\sigma_{URA,i}$	standard deviation of the clock and ephemeris error of satellite *i* used for integrity	ISM + navigation data
$\sigma_{URE,i}$	standard deviation of the clock and ephemeris error of satellite *i* used for accuracy and continuity	ISM + navigation data
$b_{nom,i}$	maximum nominal bias for satellite *i* used for integrity	ISM
$P_{sat,i}$	prior probability of fault in satellite *i* per approach	ISM
$P_{const,j}$	prior probability of a fault affecting more than one satellite in constellation *j* per approach	ISM

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).

Table 2. Navigation requirement parameters for LPV-200 and LPV-250

Name	**Description**	**Value for LPV-200 (preliminary)**	**Value for LPV-250 (preliminary)**
PHMI	total integrity budget.	10^{-7} / approach	10^{-7}/approach
P_{FA}	continuity budget allocated to disruptions due to false alert. The total continuity budget is 8 x 10^{-6} /15 s [14] (because of the temporal correlation of the error, it is adequate to use this value per 150 s).	5 x 10^{-7} / approach	5 x 10^{-7}
P_{EMT}	probability used for the calculation of the Effective Monitor Threshold	10^{-5} / approach	NA

VAL	Vertical Alert Limit	35 m	50 m
HAL	Horizontal Alert Limit	40 m	40 m
EMTL	Effective Monitor Threshold Limit	15 m	NA

Table 2 shows the constants related to the navigation requirements for LPV-200.

Table 3. Navigation requirement parameters for RNP X

Name	**Description**	**Value for RNP X (preliminary)**
PHMI	total integrity budget	10^{-7}/hour
P_{Alert}	continuity budget allocated to disruptions due to false alert and failed exclusions (see Appendix E)	5 x 10^{-7}/hour
HAL	Horizontal Alert Limit	X *1852 m

3.1 Fault modes

The ISM (Table 1) provides:

- $P_{sat,i}$: probability of a fault on satellite *i*
- $P_{const,j}$: probability a fault affecting two or more satellites within a constellation *j*

In addition we define:

- N_{sat} : number of satellites in view
- N_{const}: number of constellations in view

Note: Throughout this document, "satellites in view" and "all-in-view" will refer to all the satellites that are selected by the receiver given its limitations, the ISM, and any additional constraints (like the approval of GNSS elements by States).

The N_{sat} + N_{const} single fault events characterized by the ISM should be treated as independent events. In particular, they are not exclusive. Therefore, in the integrity risk assessment, the probability of having simultaneous faults must be accounted.

This assessment should be done by considering the set of jointly exhaustive and mutually exclusive fault modes indexed by k, (where k=0 will refer to the fault free mode) formed of all the possible combinations of the events specified in the ISM, of which there are $2^{Nsat+Nconst}$ (the number of subsets in a set of size $N_{sat} + N_{const}$). To lighten the notations, we define:

$$\begin{aligned} P_{event,i} &= P_{sat,i} \\ P_{event,Nsat+j} &= P_{const,j} \\ N_{events} &= N_{sat} + N_{const} \end{aligned} \tag{1}$$

For example, k=1 could refer to a fault in satellite 1, and no fault in the other satellites or constellations. The probability of fault mode 1 would be given by $P_{sat,1}\prod_{i=2}^{N_{sat}}\left(1-P_{sat,i}\right)\prod_{j=1}^{N_{const}}\left(1-P_{const,j}\right)$. More generally, the probability of fault mode k is given by:

$$p_{fault,k} = \prod_{i=1}^{N_{events}} P_{event,i}^{B_{i,k}}\left(1-P_{event,i}\right)^{1-B_{i,k}} \tag{2}$$

where $B_{i,k}$ is equal to one if event i is in fault mode k and zero otherwise

As will be seen in the reference algorithm in section 4, it is not necessary to compute all fault modes.

For the integrity risk computation, it should be assumed that a fault causes the addition of an arbitrary bias to the affected satellite, or an arbitrary vector of biases in a group of satellites within a given constellation. We note this vector of fault biases β_i.

3.2 Nominal error model

When a satellite is not faulted, the contribution of the satellite to the pseudorange error is characterized by a normal gaussian $N(\mu, \sigma)$ such that:

$\sigma \leq \sigma_{URA,i}$, and $|\mu| \leq b_{nom,i}$ for integrity purposes
$\sigma \leq \sigma_{URE,i}$ and $\mu = 0$ for continuity (false alert or failed exclusion) purposes

There are two other contributors to the variance of the pseudorange error: the residual tropospheric delay, and the code noise and multipath. The residual tropospheric delay is characterized by a zero mean gaussian $N(0,\sigma_{tropo,i})$ with the variance specified in Appendix A.

The code noise and multipath should be characterized by a zero mean gaussian $N(0,\sigma_{user,i})$. At steady state, and under nominal conditions, this bound should be smaller than the one specified in Appendix A. The pseudorange error covariance is characterized by diagonal covariance matrices C_{int} (the nominal error model used for integrity) and C_{acc} (the nominal error model used for accuracy and continuity). For a dual frequency user, they are defined by:

$$\begin{aligned} C_{int}(i,i) &= \sigma_{URA,i}^2 + \sigma_{tropo,i}^2 + \sigma_{user,i}^2 \\ C_{acc}(i,i) &= \sigma_{URE,i}^2 + \sigma_{tropo,i}^2 + \sigma_{user,i}^2 \end{aligned} \tag{3}$$

For a single frequency user (and therefore only for H-ARAIM), the error model includes the residual ionospheric delay (see Appendix A).

3.3 Alert limit requirement

For a fixed set of fault biases β_i, the contribution of fault mode *k* to the integrity risk is given by:

$$IR_k\left(\beta_1,\cdots,\beta_{N_{events}}\right) = p_{fault,k} \cdot \mathrm{Prob}\left(\left\{\begin{array}{l}\text{vertical position error} > \text{VAL} \\ \text{or} \\ \text{horizontal position error} > \text{HAL}\end{array}\right\} \&\text{no alert} \mid \text{fault mode } k \text{ with biases } B_{i,k}\beta_i\right) \tag{4}$$

Since the fault biases are not known and the user must be protected against any possible bias size, the integrity requirement for all fault modes is:

$$\max_{\beta_1,\cdots,\beta_{N_{events}}} \sum_{k=0}^{2^{N_{events}}-1} IR_k\left(\beta_1,\cdots,\beta_{N_{events}}\right) \leq PHMI \tag{5}$$

3.4 Effective Monitor Threshold requirement

The effective monitor threshold (EMT) requirement can be stated as follows: for the fault modes such that $p_{fault,k} \geq P_{EMT}$, when no alert is present, and assuming that there is no nominal noise (i.e. the only errors are the fault biases) the maximum size of the vertical position error (the Effective Monitor Threshold) must be below EMTL.

3.5 False alert and accuracy requirements

The false alert requirement is given by:

$$\mathrm{Prob}\left(\text{ alert} \mid \text{no fault }\right) \leq P_{FA} \tag{6}$$

For the false alert and the accuracy requirements, the satellite contribution to the pseudorange error is characterized by $\sigma_{URE,i}$.

4. ARAIM USER ALGORITHM FOR FAULT DETECTION

The algorithm described here is an acceptable way of meeting the above requirements.

4. 1 Definitions

ΔPR: when computing the position solution, the vector of pseudorange measurements (after dual frequency correction or ionospheric delay correction, tropospheric correction, and smoothing are performed) minus the expected ranging values based on the location of the satellites and the position solution given at each iteration.

y: vector of pseudorange measurements minus the expected range for an all-in-view position solution

x: receiver position and clock states (offset with respect to a position close enough to the true position so that the linear approximation of the observation equation is valid)

G: geometry matrix in East North Up (ENU) coordinates with a clock component for each constellation

Q: tail probability of a zero mean unit normal distribution. The *Q* function is defined as:

$$Q(u) = \frac{1}{\sqrt{2\pi}} \int_{u}^{+\infty} e^{-\frac{t^2}{2}} dt \tag{7}$$

$\bar{Q}$: modified Q function defined by:

$$\begin{aligned} \bar{Q}(u) &= Q(u) \text{ for } u > 0 \\ \bar{Q}(u) &= 1 \text{ for } u \le 0 \end{aligned} \tag{8}$$

Q^{-1}*:*inverse of the *Q* function.

PL: Protection Level (Vertical or Horizontal). The *PL* is an output of the user receiver that is compared to the Alert Limit to determine the availability of an operation. The *PL* is formulated so that the integrity risk is below the requirement.

Note 1: For Alert Limit aware receivers it is not necessary to compute a *PL.*

Note 2: The modified *Q* function ensures that the computed integrity risk is conservative when the detection threshold is above the *PL* [20].

4.2 List of constants derived from the requirements

Table 4. Constants derived from the navigation requirements

Name	**Description**	**Value for LPV-200 or LPV-250 if applicable (preliminary)**	**Value for RNP (preliminary**
K_{ACC}	number of standard deviations used for the accuracy formula	1.96	N/A
K_{FF}	number of standard deviations used for the 10^{-7} fault free vertical position error	5.33	N/A

4.3 List of design parameters

The parameters in Table 5 can be changed within constraints. These parameters set:

- the allocation of the integrity budget between vertical and horizontal,
- the false alert rate allocation to the monitors in the vertical domain, horizontal domain,
- the false alert rate to chi-square test.
- the parameter used to limit the number of fault modes that are monitored by the airborne algorithm.

These different parameters should be adjusted as a function of the range of the expected ISM content, and the targeted operation. For example, for a horizontal operation, one could choose to allocate all the integrity budget to the horizontal dimension. Similarly P_{THRES} should be adjusted to remove most of the fault modes. If P_{THRES} is set too low, some fault modes that could be neglected are actually triple counted (because they are accounted in full in VPL, HPL1 and HPL2).

Table 5. Design parameters (tunable)

Name	**Description**	**Value for LPV-200 and LPV-250**	**Value for RNP**
$PHMI_{VERT}$	integrity budget for the vertical component	9.8×10^{-8}/ approach	0
P_{FA_VERT}	Probability of false alert allocated to the vertical mode	3.9×10^{-6}/ approach	0

P_{FA_HOR}	Probability of false alert allocated to the horizontal mode	9 x 10^{-8}/ approach	5 x 10^{-7} /h
P_{THRES}	threshold for the integrity risk coming from unmonitored faults	8 x 10^{-8}/ approach	4 x 10^{-8}
F_C	threshold used for fault consolidation (See Eq. (18))	0.01	0.01
$N_{ITER,MAX}$	maximum number of iterations to compute the PL	10	10
TOL_{PL}	tolerance for the computation of the Protection Level	5 x 10^{-2} m	5 x 10^{-2} m

The constraints on these parameters are:

$$PHMI_{HOR} = PHMI - PHMI_{VERT} > 0 \tag{9}$$

$$P_{THRES} < PHMI \tag{10}$$

The parameter F_C sets the maximum probability of the independent satellite fault modes within one constellation that are grouped within the corresponding constellation wide fault mode.

4.4 Pseudorange covariance matrices $\mathbf{C_{int}}$ and $\mathbf{C_{acc}}$

The first step of the proposed baseline ARAIM algorithm consists of computing the pseudorange error diagonal covariance matrices C_{int} (the nominal error model used for integrity) and C_{acc} (the nominal error model used for accuracy and continuity) as described in Equation (3)

Results of this step: C_{int} and C_{acc}

4.5 All-in-view position solution

To be included in the all-in-view position solution, a satellite must not have been flagged for a given period (this period has not been determined yet) and have a valid set of input parameters from the ISM. The all-in-view position solution $\hat{x}^{(0)}$ is computed using a process analogous to the one defined in Appendix E of [11], but adapted to multi-constellation. A weighted least-squares estimation is performed at each iteration. The update for $\Delta\hat{x}$ is given by:

$$\Delta\hat{x} = \left(G^T W G\right)^{-1} G^T W \Delta PR \tag{11}$$

The geometry matrix G is an N_{sat} by 3+N_{const} matrix, where N_{const} is the number of independent constellations. The first three columns of G are defined as in Appendix E of [11]. Each of the remaining columns corresponds to the clock reference of each constellation. Labeling the constellations from j=1 to N_{const}, we define:

$$\begin{aligned} &G_{i,3+j} = 1 \text{ if satellite } i \text{ belongs to constellation } j \\ &G_{i,3+j} = 0 \text{ otherwise} \end{aligned} \tag{12}$$

The weighting matrix W is defined as:

$$W = C_{int}^{-1} \tag{13}$$

ΔPR is the vector of pseudorange measurements minus the expected ranging values based on the location of the satellites and the position solution given by the previous iteration. When the position solution has converged, the last ΔPR is the vector y as defined above. Equation (11) assumes that all measurements are in a common reference coordinate system.

Results of this step: y, G, $\hat{x}^{(0)}$

4.6 Determination of the faults that need to be monitored and the associated probabilities of fault

As explained in section 3, The ISM does not specify explicitly which fault modes need to be monitored or their corresponding prior probabilities. This determination must be made by the receiver based on the contents of the ISM, which specifies the probabilities of events that can be treated as independent.
This paragraph provides a 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 on 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:

$$\sum_{k \text{ not monitored}} p_{fault,k} \leq P_{THRES} \tag{14}$$

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

$$\sum_{k=0}^{2^{Nevents}} p_{fault,k} = 1 \tag{15}$$

The condition expressed in Equation (14) can therefore be written:

$$\sum_{k \text{ monitored}} p_{fault,k} \geq 1 - P_{THRES} \tag{16}$$

This way, it is only necessary to compute the probabilities (using Equation (2)) of the modes that will be monitored. We then need to decide the order in which the faults are considered.

The order is defined as follows:

- From smallest degree to larger
- Within one degree, from larger to smaller $p_{fault,k}$

where the degree is the number of primary events forming the composite fault mode. If a fault cannot be monitored, 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:

$$\bar{P}_{fault,not\ monitored} = \sum_{k\ not\ monitored} p_{fault,k} \tag{17}$$

Fault consolidation

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:

$$\sum_{k\in C_j} p_{fault,k} \le F_C\, p_{fault,k_j} \tag{18}$$

where F_C is a fraction of 1, the fault modes in Cj are removed from the list and the probability of fault mode k_j is updated as follows:

$$p_{fault,k_j}{}^{(updated)} = p_{fault,k_j} + \sum_{k\in C_j} p_{fault,k} \tag{19}$$

Filtering the subsets

Among the subset faults determined in the previous section, there could be some that cannot be monitored (because the remaining satellites do not allow the receiver to compute a position). In this case, these events must be removed from the list of faults (and their integrity risk subtracted from the available budget). This is true, for example, of all subsets with three satellites or less belonging to one constellation, or four satellites or less belonging to two constellations. We note $P_{unobservable}$ their total probability and therefore an upper bound on their contribution to the integrity risk. An upper bound on the total integrity risk of the modes that are not monitored is given by:

$$P_{fault,not\ monitored} = \bar{P}_{fault,not\ monitored} + P_{unobservable} \tag{20}$$

Results of this step: $p_{fault,k}$,idx_k for k ranging from 1 to the maximum number of fault modes to be monitored ($N_{fault\ modes}$), $P_{fault,not\ monitored}$

4.7 Fault-tolerant positions and associated standard deviations and biases

The monitor chosen to protect against the list of fault modes determined in the previous section is solution separation. For each k from 1 to $N_{fault\ modes}$, the difference $\Delta\hat{x}^{(k)}$ between the fault-tolerant position $\hat{x}^{(k)}$ and the all-in-view position solution $\hat{x}^{(0)}$, the standard deviations, and test thresholds are determined. For $k \geq 0$, we definethe diagonal weighting matrix:

$$\begin{aligned} &W^{(k)}(i,i) = C_{int}^{-1}(i,i) \text{ if } i \text{ is in } idx_k \\ &W^{(k)}(i,i) = 0 \text{ otherwise} \end{aligned} \tag{21}$$

For all j such that:

$$\left(G^T W^{(k)}\right)_{3+j,*} = \begin{bmatrix} 0 & \cdots & 0 \end{bmatrix}^T \tag{22}$$

G must be redefined by removing its $3+j^{th}$ column. This happens if none of the satellites from constellation j is in idx_k.

The position solution tolerant to fault mode k is obtained by applying the corresponding weighted least squares to the residuals y:

$$\begin{aligned} &\Delta\hat{x}^{(k)} = \hat{x}^{(k)} - \hat{x}^{(0)} = \left(S^{(k)} - S^{(0)}\right) y \text{ where} \\ &S^{(k)} = \left(G^T W^{(k)} G\right)^{-1} G^T W^{(k)} \end{aligned} \tag{23}$$

The computation of $S^{(k)}$ should take advantage of the relationship between $S^{(0)}$ and $S^{(k)}$ through rank one updates (in the case of a multiple satellite fault mode, more than one rank update is necessary)[1] .

Let the index q = 1, 2, and 3 designate the East, North and Up components respectively. The variances of $\hat{x}_q^{(k)}$ for q from 1 to 3 are given by:

$$\sigma_q^{(k)2} = \left(G^T W^{(k)} G\right)^{-1}_{q,q} \tag{24}$$

The worst case impact of the nominal biases occurs when the nominal bias of each measurement has the same sign as the coefficient projecting the pseudorange onto the position. Since the absolute value of each nominal bias is bounded by $b_{nom,i}$ and the signs of the nominal biases are not known to the receiver (see List of Inputs), the worst case impact on the position solution $\hat{x}_q^{(k)}$ is given by:

$$b_q^{(k)} = \sum_{i=1}^{N_{sat}} \left| S_{q,i}^{(k)} \right| b_{nom,i} \tag{25}$$

We compute the variance of the difference, $\Delta\hat{x}_q^{(k)}$, between the all-in-view and the fault tolerant position solutions:

$$\sigma_{ss,q}^{(k)2} = e_q^T \left(S^{(k)} - S^{(0)} \right) C_{acc} \left(S^{(k)} - S^{(0)} \right)^T e_q \tag{26}$$

in which e_q denotes a vector whose q^{th} entry is one and all others are zero.

Results of this step: $\sigma_q^{(k)}, \sigma_{ss,q}^{(k)}, b_q^{(k)}$ for k from 0 to $N_{fault\ modes}$, and for q from 1, 2, and 3.

4.8 Solution separation threshold tests

Solution Separation Test

For each fault mode, there are three solution separation threshold tests, one for each coordinate. The thresholds are indexed by the fault index k and the coordinate index q and noted $T_{k,q}$. They are defined by:

$$T_{k,q} = K_{fa,q} \sigma_{ss,q}^{(k)} \tag{27}$$

where:

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

$$K_{fa,3} = Q^{-1}\left(\frac{P_{FA\_VERT}}{2N_{fault\ modes}} \right) \tag{29}$$

$Q^{-1}(p)$ is the (1-p)-quantile of a zero-mean unit-variance Gaussian distribution. Protection Levels can be computed only if for all k and q we have:

$$\left| \hat{x}_q^{(k)} - \hat{x}_q^{(0)} \right| \le T_{k,q} \tag{30}$$

If any of the tests fails, the service is not available without successful exclusion.

Note: If exclusion is attempted, the Protection Level must be modified to account for the additional integrity risk exposure (even if there is no detection). Section 5 describes a method to account for exclusion.

Note on χ^2 statistic

This test is not required, as it does not offer additional protection for faults listed in the threat model. The chi-square statistic for the all-in-view set is computed as follows:

$$\chi^2 = y^T\left(W_{acc} - W_{acc}G\left(G^T W_{acc} G\right)^{-1} G^T W_{acc}\right)y \tag{31}$$

In this equation, we have $W_{acc} = C_{acc}^{-1}$. As shown in [1], this chi-square statistic is an upper bound of all solution separation tests. Therefore, if a fault is detectable, it will manifest itself in this statistic. The threshold is defined by:

$$F\left(T_{\chi^2}, N_{sat} - 3 - N_{const}\right) = 1 - P_{FA\_CHI2} \tag{32}$$

The false alert allocation P_{FA_CHI2} should be set to have a negligible impact on the overall false alert budget, since it is only a sanity check. The operator $F(u, \deg)$ is the cdf of a chi-square distribution with *deg* degrees of freedom. If $\chi^2 > T_{\chi^2}$, but $\left|\hat{x}_q^{(k)} - \hat{x}_q^{(0)}\right| \le T_{k,q}$ for all q and k, the PL cannot be considered valid and exclusion cannot be attempted. In this case, the chi-square statistic is larger than expected, but none of the solution separation tests have failed, which suggests that the fault is outside the threat model. While the chi-square test is not linked to the threat model, it makes the algorithm more robust to violations of the threat model with no performance or computational penalty. A similar test is required for SBAS [11].

Results of this step: Thresholds $T_{k,q}$, decision on whether to continue with Protection Level calculation, attempt fault exclusion, or declare the HPL and VPL invalid.

4.9 Protection Levels

Vertical Protection Level (VPL)

The Protection Levels are determined by the integrity requirement. For the VPL, we need to make sure that the integrity risk (which is the sum of the contribution of each fault mode) is below the integrity risk allocated to the vertical error. The solution to the following equation provides a VPL that meets the required integrity allocation:

$$2\bar{Q}\left(\frac{VPL-b_3^{(0)}}{\sigma_3^{(0)}}\right)+\sum_{k=1}^{N_{fault\ modes}} p_{fault,k}\bar{Q}\left(\frac{VPL-T_{k,3}-b_3^{(k)}}{\sigma_3^{(k)}}\right)= PHMI_{VERT}\left(1-\frac{P_{fault,not\ monitored}}{PHMI_{VERT}+PHMI_{HOR}}\right) \quad (33)$$

In Equation (33), each term of the left hand side is an upper bound of the contribution of each fault to the integrity risk. The proof of safety associated to this Protection Level can be found in Appendix H of [1]. The output VPL must be within TOL_{PL} of the solution of this equation. There are several methods available to solve this equation. Appendix B of [1] proposes one of them, as well as an upper bound (which is actually close to the solution).

Horizontal Protection Level (HPL)

For the HPL computations, we first compute HPL_q for q=1 and 2. As for the VPL, HPL_q is the solution to the equation:

$$2\bar{Q}\left(\frac{HPL_q-b_q^{(0)}}{\sigma_q^{(0)}}\right)+\sum_{k=1}^{N_{fault\ modes}} p_{fault,k}\bar{Q}\left(\frac{HPL_q-T_{k,q}-b_q^{(k)}}{\sigma_q^{(k)}}\right)= \frac{PHMI_{HOR}}{2}\left(1-\frac{P_{fault,not\ monitored}}{PHMI_{VERT}+PHMI_{HOR}}\right) \quad (34)$$

The output HPL_q must be within TOL_{PL} of the solution of this equation. This equation can be solved using a half interval search as shown for the VPL in Appendix B. The HPL is given by:

$$HPL=\sqrt{HPL_1^2+HPL_2^2} \quad (35)$$

Accounting for possible double counting of integrity risk

Due to the pre-allocation of the integrity budget to each of the coordinates, there is the possibility that the computed contribution of integrity risk of a fault mode might exceed the probability of the fault mode. This can result in loss of performance. Let us consider mode k. The upper bound on the contribution to mode k is given by:

$$IR_k = p_{fault,k}\left(\bar{Q}\left(\frac{HPL_1-T_{k,1}-b_1^{(k)}}{\sigma_1^{(k)}}\right)+\bar{Q}\left(\frac{HPL_2-T_{k,2}-b_2^{(k)}}{\sigma_2^{(k)}}\right)+\bar{Q}\left(\frac{VPL-T_{k,3}-b_3^{(k)}}{\sigma_3^{(k)}}\right)\right) \quad (36)$$

If the term between parenthesis exceeds one, then IR_k exceeds $p_{fault,k}$. However, if we had chosen not to monitor mode k, IR_k would have been exactly $p_{fault,k}$, which would have resulted in a smaller Protection Level.

This possible loss of performance can be mitigated by: first, identifying the modes for which we are overestimating the integrity risk, second, by excluding them from the list of monitored faults, and, third, by recomputing the thresholds and Protection Levels with the new list. Specifically, we find the set of indices *k* such that:

$$\bar{Q}\left(\frac{HPL_1 - T_{k,1} - b_1^{(k)}}{\sigma_1^{(k)}}\right) + \bar{Q}\left(\frac{HPL_2 - T_{k,2} - b_2^{(k)}}{\sigma_2^{(k)}}\right) + \bar{Q}\left(\frac{VPL - T_{k,3} - b_3^{(k)}}{\sigma_3^{(k)}}\right) \geq 1 \tag{37}$$

Let us call this set I_{excl}. We exclude these modes from the list of monitored modes. Since they are now excluded from this list, we must account their integrity risk contribution in the term $P_{fault,not\ monitored}$ computed in Equation (20). We define $P_{fault,not\ monitored,new}$:

$$P_{fault,not\ monitored,new} = P_{fault,not\ monitored} + \sum_{k \in I_{excl}} p_{fault,k} \tag{38}$$

The new number of monitored fault modes is then:

$$N_{fault\_\text{modes,new}} = N_{fault\_\text{modes}} - \left|I_{excl}\right| \tag{39}$$

Note that the detection thresholds defined in Equations (27), (28), and (29) should be re-computed, as they depend on the number of monitored faults.

Results of this step: VPL and HPL

4.10 Accuracy, the fault free position error bound, and Effective Monitor Threshold (for LPV-200 only)

The standard deviation of the vertical position solution used for these two criteria is given by:

$$\sigma_{v,acc} = \sqrt{e_3^T S^{(0)} C_{acc} S^{(0)T} e_3} \tag{40}$$

The formulas for the two accuracy requirements are given by:

$$accuracy(95\%) = K_{ACC}\sigma_{v\_acc} \tag{41}$$

$$fault - free\left(10^{-7}\right) = K_{FF}\sigma_{v,acc} \tag{42}$$

Because 10 m / K_{FF} is smaller than 4 m / K_{ACC}, the fault-free test is the only one that needs to be evaluated by the aircraft. We therefore need to test:

$$\sigma_{v\_acc} \leq 1.87m \tag{43}$$

The Effective Monitor Threshold (EMT) can be defined as the maximum of the detection thresholds of faults that have a prior equal or above P_{EMT}. It is computed as follows:

$$EMT = \max_{k|p_{faul,k} \geq P_{EMT}} T_{k,3} \tag{44}$$

Results of this step: 95% accuracy, the 10^{-7} fault free position error bound, and EMT

4.11 Optimized positioning for weak geometries

An approach to minimize the Protection Levels by adjusting the position was described in [12]. As shown in this reference, there can be an improvement in the integrity error bound by choosing a solution position that is offset from the most accurate position solution under nominal conditions. For geometries where one of the subsets has a much larger standard deviation, this algorithm can be greatly simplified and is specified below. This approach should only be applied when a target protection level is not achieved (for example, for LPV-200 if the VPL exceeds 35 m or the EMT exceeds 15 m and $\sigma_{v\_acc} \leq 1.87m$). This part of the algorithm should be inserted after Equation (25).

Application to VPL

We describe the algorithm for the vertical protection level. At the end, we show how to use it to compute the horizontal protection level.

Step 1: Among the fault modes that are going to be monitored, and whose a priori probability is above PHMI, select the one with the largest $\sigma_3^{(k)}$. We define as s_{max} the corresponding coefficients (the third row of $S^{(k)}$). We also note s_{all} the third row of $S^{(0)}$. In addition we note $\sigma^2_{acc,req}$ the required accuracy for LPV 200 (=1.87^2).

Step 2: Compute:

$$\begin{aligned} a &= \left(s_{\max} - s_{all}\right)^T C_{acc} \left(s_{\max} - s_{all}\right) \\ b &= 2{s_{all}}^T C_{acc} \left(s_{\max} - s_{all}\right) \\ c &= {s_{all}}^T C_{acc} s_{all} - \sigma^2_{acc,req} \end{aligned} \tag{45}$$

Step 3: Compute:

$$t = \min\left(1, \frac{-b+\sqrt{b^2-4ac}}{2a}\right) \tag{46}$$

Step 4: Compute:

$$s = s_{all} + t\left(s_{\max} - s_{all}\right) \tag{47}$$

Once the all-in-view coefficients have been computed according to Equation (47), the algorithm to compute the test thresholds and the PLs described above is modified as follows. In Equation (23), the third row of $S^{(0)}$ is replaced with s, and in Equation (24), the standard deviation (k= 0 and q = 3) for the fault free case is given by:

$$\sigma_3^{(0)2} = s^T C_{\text{int}} s \tag{48}$$

The rest of the algorithm (Equations (25) through (44)) remains unchanged. A more detailed account of this method can be found in [13].

Note: If t = 1,we have $s_{all} = s_{max}$, which causes both the threshold and the statistic in the test (30) to be zero. This means that the test should always pass (the position solution is not affected by the fault corresponding to s_{max}). However, numerical errors can cause the test to fail. There are many ways to solve this issue. One of them is to force the test corresponding to s_{max} to pass whenever $\frac{-b+\sqrt{b^2-4ac}}{2a} \geq 1$.

Application to HPL

This algorithm modification can also be applied to each of the horizontal components. Although there is not an equivalent fault free accuracy requirement for RNP, a value of 20 m was chosen (so that the algorithm would not degrade excessively the horizontal accuracy).

5. ARAIM USER ALGORITHM FOR FAULT DETECTION AND EXCLUSION

This section describes a method to modify the Protection Levels when exclusion is implemented.

5. 1 Finding a consistent set

The first step of the exclusion algorithm consists in finding a subset of measurements that is consistent. A subset is determined to be consistent if it passes the solution separation tests described by Equation (30) . As shown in [1], it is possible to avoid testing all possible subsets by checking the chi-square statistic of each of the subsets. The chi-square statistic is defined by:

$$q_k = y^T \left(W^{(k)} - W^{(k)} G \left(G^T W^{(k)} G \right)^{-1} G^T W^{(k)} \right) y \tag{49}$$

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 order to perform the solution separation tests on the subset, we need to determine the list of faults to be monitored. In this algorithm, any set that passes the consistency checks can be chosen.

Results of this step: indices of set of candidate consistent measurements idx_j.

5.2 Determination of faults to be monitored

The list of faults to be monitored is the same list determined in section 4.6. The new sets of indices used to compute the fault tolerant position solution will be given by:

$$idx_j \cap idx_k \tag{50}$$

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 $idx_k^{(j)}$.

To illustrate this step, let us suppose that there are 6 satellites in view {1,2,3,4,5,6}, and that satellite 2 was excluded. If, for example,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 (50) 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:

$$p_{fault,k}^{(j)} = \sum_{k' | idx_k^{(j)} = idx_{k'} \cap idx_j} p_{fault,k'} \tag{51}$$

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

5.3 Solution separation threshold tests

The solution separation tests are formally identical to the all-in-view solution separation tests. The only difference is that now the all-in-view is the candidate subset determined above. We note ${}^{(j)}T_{k,q}$ the corresponding thresholds. They are defined by:

$${}^{(j)}T_{k,q} = {}^{(j)}K_{fa,q} \, {}^{(j)}\sigma_{ss,q}^{(k)} \tag{52}$$

where ${}^{(j)}\sigma_{ss,q}^{(k)}$ is the standard deviation of the solution separation statistic between the candidate subset *j* and the subset *k*. The containments ${}^{(j)}K_{fa,q}$ are defined using Equations (28) and (29) with $N_{fault\_modes,j}$.

Note: It is possible to modify the false alert allocations as long as the overall impact on the probability of loss of continuity remains the same (see Appendix E).

5.4 Protection levels

The equations defining the protection levels with fault exclusion are formally identical to the fault detection protection levels. The only 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

${}^{(j)}HPL_q$ (for q = 1 and 2) is the solution of the equation:

$$\begin{aligned}&2\bar{Q}\left(\frac{{}^{(j)}HPL_q - {}^{(j)}b_q^{(0)}}{{}^{(j)}\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{fault\ modes,j}} {}^{(j)}p_{fault,k}\bar{Q}\left(\frac{{}^{(j)}HPL_q - {}^{(j)}T_{k,q} - {}^{(j)}b_q^{(k)}}{{}^{(j)}\sigma_q^{(k)}}\right) = \\ &\rho_j \frac{PHMI_{HOR}}{2}\left(1 - \frac{P_{fault,not\ monitored}}{PHMI_{VERT} + PHMI_{HOR}}\right)\end{aligned} \tag{53}$$

where:

${}^{(j)}b_q^{(k)}, {}^{(j)}\sigma_q^{(k)}, {}^{(j)}T_{k,q}$ are computed using the new subsets $idx_k^{(j)}$

ρ_j is a parameter adjusting the integrity allocation. 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:

$$\sum_{j=0}^{N_{fault\ modes}} \rho_j = 1 \tag{54}$$

As in the fault detection case, the HPL is given by:

$${}^{(j)}HPL = \sqrt{{}^{(j)}HPL_1^2 + {}^{(j)}HPL_2^2} \tag{55}$$

Vertical Protection Level

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

$$2\bar{Q}\left(\frac{{}^{(j)}VPL-{}^{(j)}b_3^{(0)}}{{}^{(j)}\sigma_3^{(0)}}\right)+\sum_{k=1}^{N_{fault\ modes,j}}{}^{(j)}p_{fault,k}\bar{Q}\left(\frac{{}^{(j)}VPL-{}^{(j)}T_{k,3}-{}^{(j)}b_3^{(k)}}{{}^{(j)}\sigma_3^{(k)}}\right)= \rho_j PHMI_{VERT}\left(1-\frac{P_{fault,not\ monitored}}{PHMI_{VERT}+PHMI_{HOR}}\right) \tag{56}$$

Integrity allocation across exclusion options

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, which we note J_{exc}. This set will be a subset of all the monitored fault modes, and includes the all-in-view (j=0). For example, 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})(See Appendix E). For the indices j corresponding to these exclusion options, we set:

$$\rho_j = \frac{1}{N_{exc}+1} \tag{57}$$

where N_{exc} is the number of pre-selected exclusion options (excluding the all-in-view). In the case where only single satellite satellite exclusions are attempted, we have $N_{exc} = N_{sat}$. Note that the PLs above will only be defined for the pre-selected exclusion options.

Note 1: If the receiver has sufficient computational power, the HPL can be computed by solving the equation (as suggested in [19]):

$$\sum_{j\in J_{exc}}\left[2\bar{Q}\left(\frac{HPL_q-{}^{(j)}b_q^{(0)}}{{}^{(j)}\sigma_q^{(0)}}\right)+\sum_{k=1}^{N_{fault\ modes,j}}{}^{(j)}p_{fault,k}\bar{Q}\left(\frac{HPL_q-{}^{(j)}T_{k,q}-{}^{(j)}b_q^{(k)}}{{}^{(j)}\sigma_q^{(k)}}\right)\right]= \frac{PHMI_{HOR}}{2}\left(1-\frac{P_{fault,not\ monitored}}{PHMI_{VERT}+PHMI_{HOR}}\right) \tag{58}$$

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).

Note 2: The *PL* is treated here as an output that is to be compared with the Alert Limit. It is not a predictive value that indicates whether exclusion is available or not.

Horizontal Uncertainty Level

In some instances it may be advantageous to keep using the all-in-view solution position after a detection event, as it may provide lower error bounds. In this case, the user computes a position error bound (the Uncertainty Level) that meets the integrity requirement but that is directly dependent on the measurement residuals. An acceptable formula for the HUL is given by:

$$HUL = \sqrt{HUL_1^2 + HUL_2^2} \tag{59}$$

where:

$$HUL_q = \max_{\substack{k \in [\![0, N_{fault\ modes}]\!] \\ HPL_q - T_{k,q} - b_q^{(k)} > 0}} \left| \hat{x}_q^{(k)} - \hat{x}_q^{(0)} \right| + K_{md,q,k} \sigma_q^{(k)} \tag{60}$$

and:

$$K_{md,q,k} = \frac{HPL_q - T_{k,q} - b_q^{(k)}}{\sigma_q^{(k)}} \tag{61}$$

If necessary, a similar formula can be defined for the VUL.

6. BASELINE SIMULATION CONDITIONS

In this section we describe the simulation conditions that have been used for the evaluation of ARAIM coverage in [18] and [21].

Constellation configurations

Four constellation scenarios have been chosen which are meant to represent: a configuration which uses the reference almanac for each constellation (‘baseline’), a configuration in which one satellite has been removed in each constellation (‘depleted’),and a more optimistic configuration, consistent with the observed history of GPS and that assumes that Galileo will match the number of satellites expected for GPS, which is not unrealistic given Galileo replenishment strategy (‘optimistic’):

1. Baseline: GPS 24 (24-slot nominal GPS constellation), Galileo 24 (baseline)
2. Depleted: GPS 24-1 , Galileo 24-1
3. Expected: GPS 24 + 3, Galileo 24
4. Optimistic: GPS 24 + 3, Galileo 24+3

Table 6. Almanacs used for availability simulations

	GPS	Galileo
24-1	almmops-1.txt	almanac Galileo 24-1 Week 703.alm.txt
24	almmops.txt	almanac Galileo 24 Week 703.alm.txt
24+3	almgps24+3.txt	almanac Galileo 24 + 3 Spare Week 703.alm.txt

The almanacs can be downloaded here.

User mask angle

Table 7. User mask angle used in simulations

	GPS	Galileo
User mask angle in degrees	5 degrees	5 degrees

User grid and time steps

Users are simulated as follows:

- 5 by 5 degree user grid
- 10 sidereal days
- 600 s time steps

Evaluation criteria

- Coverage of 99.5% of LPV 200 and APV1/LPV 250 between -70 and 70 degrees latitude
- For coverage, user grid points are weighed by the cosine of the latitude to account for the relative area they represent

Availability criteria:

Table 8. Availability criteria used in simulations

	VAL	HAL	EMT	σ_{acc} threshold
LPV-200	35 m	40 m	15 m	1.87 m
APV 1 / LPV-250	50 m	40 m	-	-
RNP 0.1	-	185 m	-	-
RNP 0.3	-	556 m	-	-

Simulation settings

For the Milestone IIb Report, the ISM parameters have been set to:

- σ_{URA} = .5m, .75m, 1m, 1.5m, 2m, for LPV-200 and LPV-250 and 2.5m for Horizontal
- $\sigma_{URE} = 2/3\ \sigma_{URA}$
- b_{nom} = .75m
- $P_{sat} = 10^{-5}$
- $P_{const} = 10^{-4}, 10^{-8}$

APPENDIX A

Error Models for dual frequency

The error models that will be used for Advanced RAIM have not yet been fully determined. The final values will need to be consistent with the values developed for dual frequency SBAS. The error budgets that are included here reflect the best estimate.

Two error budgets for GPS and Galileo have been made use of to allow for a performance prediction in the frame of ARAIM. A preliminary Galileo user contribution to the error budget was described in tabular form [21].

Table A-1. Galileo Elevation Dependent SIS user error

(meters)	Galileo			
$\sigma_{n,user}^{Gal}$	5°	0.4529m	50°	0.2359 m
	10°	0.3553 m	55°	0.2339 m

(vs elevation)	15°	0.3063 m	60°	0.2302 m
	20°	0.2638 m	65°	0.2295 m
	25°	0.2593 m	70°	0.2278 m
	30°	0.2555 m	75°	0.2297 m
	35°	0.2504 m	80°	0.2310 m
	40°	0.2438 m	85°	0.2274 m
	45°	0.2396 m	90°	0.2277 m

However, at the moment, it is more likely that the error bound for Galileo will be the one used for GPS, which is specified below.

The $\sigma_{n,user}$ for GPS follows the formula provided in [14] for the Airborne Accuracy Designator – Model B (AAD-B) [15]:

$$\sigma_{n,user}^{GPS} = \sqrt{\frac{f_{L1}^4 + f_{L5}^4}{\left(f_{L1}^2 - f_{L5}^2\right)^2}} \sqrt{\left(\sigma_{MP}\right)^2 + \left(\sigma_{Noise}\right)^2}$$
$$\sigma_{MP}\left(\theta\right) = 0.13[\mathrm{m}] + 0.53[\mathrm{m}]\exp(-\theta / 10[\mathrm{deg}])$$
$$\sigma_{Noise}\left(\theta\right) = 0.11[\mathrm{m}] + 0.13[\mathrm{m}]\exp(-\theta / 6.9[\mathrm{deg}]) \tag{62}$$

where θ is the elevation angle in degrees. This represents an overbound of the error after carrier smoothing.

The tropospheric delay $\sigma_{n,tropo}$ can be modeled according to [16] as

$$\sigma_{n,tropo}\left(\theta\right) = 0.12[\mathrm{m}] \frac{1.001}{\sqrt{0.002001 + \left(\sin\left(\frac{\pi\theta}{180}\right)\right)^2}} \tag{63}$$

Nominal error model for single frequency (L1 or L5)

The standard deviation of the nominal error model for single frequency is given by:

$$\sigma_i^2 = \sigma_{URA,i}^2 + \sigma_{tropo,i}^2 + \sigma_{SFuser,i}^2 + \sigma_{iono,i}^2 \tag{64}$$

The third term, which bounds the code noise and multipath is defined here as a fraction of the code noise and multipath term used for dual frequency:

$$\sigma_{SFuser,i} = \sqrt{\frac{\left(f_{L1}^2 - f_{L5}^2\right)^2}{f_{L1}^4 + f_{L5}^4}}\sigma_{user,i} \quad (65)$$

(This correction undoes the correction made in [17] for dual frequency GPS and scales down the corresponding Galileo term.)

The ionospheric delay error models for GPS are well established. For L1, the standard deviation of the ionospheric delay error bound is assumed to be equal to $\sigma_{i,UIRE}$ as defined in Appendix J of [11]. That is:

$$\sigma_{iono,i}^2 = \sigma_{i,UIRE}^2 \quad (66)$$

In the case of Galileo, the ionospheric correction method for aviation receivers is still evolving, and the error bounds on the residual error have not been developed. It is however likely that the residual ionospheric error bounds for Galileo will be as good as the ones for GPS, in which case using the GPS model for Galileo is conservative for simulation purposes.

For L5, the error bound must account for the increased uncertainty due to the difference between the L1 and L5 frequencies f_{L1} and f_{L5}. We have in this case:

$$\sigma_{iono,i}^2 = \frac{{f_{L1}}^4}{{f_{L5}}^4}\sigma_{i,UIRE}^2 \quad (67)$$

APPENDIX B

Methods to Solve the VPL Equation

Iterative method

The VPL can be obtained by solving the following equation using a half interval search:

$$P_{exceed}\left(VPL\right) = PHMI_{VERT,ADJ} \quad (68)$$

where:

$$P_{exceed}\left(VPL\right) = 2\bar{Q}\left(\frac{VPL - b_3^{(0)}}{\sigma_3^{(0)}}\right) + \sum_{k=1}^{N_{fault\ \mathrm{modes}}} p_{fault,k}\bar{Q}\left(\frac{VPL - T_{k,3} - b_3^{(k)}}{\sigma_3^{(k)}}\right) \quad (69)$$

and:

$$PHMI_{VERT,ADJ} = PHMI_{VERT}\left(1-\frac{P_{sat,not\ monitored}+P_{const,not\ monitored}}{PHMI_{VERT}+PHMI_{HOR}}\right) \tag{70}$$

This search can be started with the lower and upper bounds which relate to full and even allocation of the integrity risk respectively and are given by:

$$VPL_{low,init} = \max\left\{\begin{array}{l} Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{2}\right)\sigma_3^{(0)}+b_3^{(0)}, \\ \max\limits_k Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{p_{fault,k}}\right)\sigma_3^{(k)}+T_{k,3}+b_3^{(k)} \end{array}\right\} \tag{71}$$

$$VPL_{up,init} = \max\left\{\begin{array}{l} Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{2\left(N_{faults}+1\right)}\right)\sigma_3^{(0)}+b_3^{(0)}, \\ \max\limits_k Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{p_{fault,k}\left(N_{faults}+1\right)}\right)\sigma_3^{(k)}+T_{k,3}+b_3^{(k)} \end{array}\right\} \tag{72}$$

The iterations stop when:

$$\left|VPL_{up}-VPL_{low}\right| \le TOL_{PL} \tag{73}$$

or when the number of iterations exceeds $N_{iter,max}$. The final VPL is given by VPL_{up} at the end of iteration. In the case of HPL_1 and HPL_2, the approach is identical, but the appropriate parameters must be changed.

Approximation Not Requiring an Iterative Algorithm

In cases where the argument of the Q function is larger than 1 for the value $VPL_{low,init}$, the function P_{exceed} is convex so a linear approximation provides a tight upper bound of the VPL.

$$\begin{aligned} VPL_{approx,upper} &= VPL_{low,init} + \\ &\left(PHMI_{VERT}-P_{exceed}\left(VPL_{low,init}\right)\right)\times \\ &\frac{VPL_{upper,init}-VPL_{low,init}}{P_{exceed}\left(VPL_{upper,init}\right)-P_{exceed}\left(VPL_{low,init}\right)} \end{aligned} \tag{74}$$

If VPL_{low_init} is not such that the argument of Q is more than 1 in all terms, we define a new VPL guess as:

$$VPL_{guess} = \max \left\{ \begin{array}{l} Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{2}\right)\sigma_3^{(0)} + b_3^{(0)}, \\ \max\limits_k \left(\max\left(Q^{-1}\left(\dfrac{PHMI_{VERT,ADJ}}{p_{fault,k}}\right), 1\right)\sigma_3^{(k)} + T_{k,3} + b_3^{(k)} \right) \end{array} \right\} \tag{75}$$

This guess is such that the condition above is fulfilled. If $P_{exceed}(VPL_{guess}) > PHMI_{VERT}$, then we take $VPL_{low,init} = VPL_{guess}$, otherwise the upper approximation is simply $\min(VPL_{guess}, VPL_{upper,init})$.

This approximation does not provide a bound as tight as the iterative method, but it might be sufficient.

Similarly, the function log P_{exceed} is concave, so a linear approximation provides a tight lower bound:

$$\begin{aligned} & VPL_{approx,low} = VPL_{low,init} + \\ & \left(\log PHMI_{VERT,ADJ} - \log P_{exceed}\left(VPL_{low,init}\right)\right) \times \\ & \frac{VPL_{upper,init} - VPL_{low,init}}{\log P_{exceed}\left(VPL_{upper,init}\right) - \log P_{exceed}\left(VPL_{low,init}\right)} \end{aligned} \tag{76}$$

This lower bound can be used to speed up the search.

APPENDIX C

Formulas for the determination of the list of monitored faults

Probability of subset fault

In the following equations, $P_{event,i}$ is the prior probability of the independent fault event i, which is included in the Integrity Support Message. The probability of the set of events $i_1, i_2, \ldots, i_r$, and no other fault is:

$$\begin{aligned} & \prod_{s=1,\cdots,r} P_{event,i_s} \prod_{s\neq 1,\cdots,r} \left(1 - P_{event,i_s}\right) = \\ & \prod_{k=1}^{N_{sat}+N_{const}} \left(1 - P_{event,k}\right) \prod_{s=1,\cdots,r} \frac{P_{event,i_s}}{1 - P_{event,i_s}} \\ & = P_{no\_fault} \prod_{s=1,\cdots,r} \frac{P_{event,i_s}}{1 - P_{event,i_s}} \end{aligned} \tag{77}$$

where:

$$P_{no\_fault} = \prod_{k=1}^{N_{sat}+N_{const}} \left(1 - P_{event,k}\right) \tag{78}$$

APPENDIX D

Numerical example for LPV-200

We consider the geometry defined by G:

```
G = [0.0225    0.9951   -0.0966  1  0;
      0.6750   -0.6900   -0.2612  1  0;
      0.0723   -0.6601   -0.7477  1  0;
     -0.9398    0.2553   -0.2269  1  0;
     -0.5907   -0.7539   -0.2877  1  0;
     -0.3236   -0.0354   -0.9455  0  1;
     -0.6748    0.4356   -0.5957  0  1;
      0.0938   -0.7004   -0.7075  0  1;
      0.5571    0.3088   -0.7709  0  1;
      0.6622    0.6958   -0.2780  0  1];
```

(79)

We assume that for all satellites:

$$\sigma_{URA,i}\text{=.75 m} \qquad \sigma_{URE,i}\text{=.50 m} \qquad P_{sat,i}=10^{-5}$$
$$b_{nom,i} = .5\text{ m} \tag{80}$$

For the two constellations we assume:

$$P_{const,j}=10^{-4} \tag{81}$$

Following the steps outlined in the paper and using the preliminary values introduced in the list of constants we have:

$$C_{\mathrm{int}} = diag\begin{pmatrix} [3.2899 & 1.2792 & 0.7901 & 1.4430 & 1.1847 \\ 0.7737 & 0.8233 & 0.7962 & 0.7871 & 1.2166] \end{pmatrix}$$

$$C_{acc} = diag\begin{pmatrix} [2.9774 & 0.9667 & 0.4776 & 1.1305 & 0.8722 \\ 0.4612 & 0.5108 & 0.4837 & 0.4746 & 0.9041] \end{pmatrix} \tag{82}$$

The subset fault modes is composed of all *n-1* subsets, as well as the two constellation fault modes. Let *k* and *k'* be the indexes corresponding to the two constellation fault modes. We have:

$$\sigma_3^{(k)} = 2.4600\text{ m} \qquad \sigma_3^{(k')} = 2.4359\text{ m}$$

$$\sigma_{ss,3}^{(k)} = 1.4290 \text{ m} \qquad \sigma_{ss,3}^{(k')} = 1.4217 \text{ m}$$
$$b_3^{(k)} = 2.8915 \text{ m} \qquad b_3^{(k')} = 2.0875 \text{ m}$$
(83)

(We do not write the standard deviations for all the other subsets). We have:

$$K_{fa,3} = Q^{-1}\left(\frac{P_{FA\_VERT}}{2N_{fault \text{ modes}}}\right) = Q^{-1}\left(\frac{3.9\times10^{-6}}{2\times12}\right) = 5.1083$$
(84)

The solution to Equation (33) is:

$$VPL = 18.3 \ m$$

The HPL is given by Equation (35) and is:

$$HPL = 13.45 \ m$$

The EMT is given by Equation (44) and is:

$$EMT = 7.2998 \ m$$

The standard deviation of the all-in-view given by Equation (40) is:

$$\sigma_{v,acc} = 1.3694 \text{ m}$$

APPENDIX E

One of the contributors to the loss of continuity in ARAIM is the probability that the algorithm ceases to provide a finite Protection Level. This can happen when the consistency check fails and it is not followed by a successful exclusion. This Appendix describes the relationship between the choice of detection and exclusion thresholds and the probability of alert P_{Alert}.

We assume that, at the most, a fault will make the ARAIM test trigger once. As a consequence, for a fault with probability of onset P_{onset} (be it a satellite or constellation fault) and an exposure time T_{exp} the probability that a consistency test including satellite i or constellation j will fail is:

$$P_{sat,cont,i} = P_{onset,sat,i} T_{\exp,i}$$
$$P_{const,cont,j} = P_{onset,const,j} T_{\exp,j}$$
(85)

Just like for the integrity evaluation, we need to take into account all possible combinations of faults. Using the methods that are used to determine the list of faults to be monitored, we compute the probabilities $P_{fault,cont,j}$, and form a list.

The probability that there is a failed exclusion given that fault *j* is present is bounded by the probability of a false alert on the subset that is not affected by fault *j*. We note $test_j$ the indicator of the event that the consistency check of subset *k* passes, that is:

$$test_j = 1_{\Omega_j}(y)$$

$$\Omega_j = \left\{ y \mid \forall k, q \left| {}^{(j)}\hat{x}_q^{(k)} - {}^{(j)}\hat{x}_q^{(0)} \right| \leq {}^{(j)}T_{k,q} \right\} \tag{86}$$

If no test is performed for fault *j*, and fault *j* is present, then in the worst case there will be an alarm. To simplify the notations, we set $test_j = 0$ for those cases.

The probability of loss of continuity due to an ARAIM alert P_{Alert} is bounded as follows:

$$P_{Alert} \leq \sum_j P_{fault,cont,\mathrm{j}} P\left(test_j = 0\right) \tag{87}$$

We have:

$$\begin{aligned} P\left(test_j = 0\right) &= P\left(\exists (k,q) \mid \left| {}^{(j)}\hat{x}_q^{(k)} - {}^{(j)}\hat{x}_q^{(0)} \right| > {}^{(j)}T_{k,q} \right) \\ &\leq \sum_{k=0}^{N_{fault\_modes,j}} \sum_{q=1}^{3} P\left(\left| {}^{(j)}\hat{x}_q^{(k)} - {}^{(j)}\hat{x}_q^{(0)} \right| > {}^{(j)}T_{k,q} \right) \\ &\leq \sum_{k=0}^{N_{fault\_modes,j}} \sum_{q=1}^{3} 2Q\left(\frac{{}^{(j)}T_{k,q}}{{}^{(j)}\sigma_{ss,q}^{(k)}} \right) \end{aligned} \tag{88}$$

Let us note *J* the set of subsets that will be tested. We can write:

$$P_{Alert} \leq \sum_{j \in J} P_{fault,cont,j} \sum_{k=0}^{N_{fault\_modes,j}} \sum_{q=1}^{3} 2Q\left(\frac{{}^{(j)}T_{k,q}}{{}^{(j)}\sigma_{ss,q}^{(k)}} \right) + \sum_{j \notin J} P_{fault,cont,j} \tag{89}$$

Vertical guidance

We have:

$$\sum_{j>0} P_{fault,cont,j} \leq \sum_{i=1}^{N_{sat}} P_{onset,sat,i} T_{\exp,i} + \sum_{j=1}^{N_{const}} P_{onset,const,j} T_{\exp,j} \tag{90}$$

If we assume, for continuity purposes, that:

$$\begin{aligned} P_{onset,sat,i} &\leq 10^{-5} / hour \\ P_{onset,\mathrm{const},i} &\leq 10^{-4} / hour \end{aligned} \tag{91}$$

Then we will have, with T_{exp} = 15 s:

$$\sum_{j>0} P_{fault,cont,j} \leq \left(10^{-5} N_{sat} + 10^{-4} N_{const}\right) \cdot \frac{15}{3600} = 4.2 \cdot 10^{-8} \left(N_{sat} + 10N_{const}\right) \tag{92}$$

For N_{sat} = 30 and N_{const} = 2, we have:

$$\sum_{j>0} P_{fault,cont,j} \leq 2.1 \cdot 10^{-6} / 15s \tag{93}$$

Going back to Equation (89), we can see that it is not necessary to attempt exclusion to meet the continuity requirement, since the total continuity budget is 8x10^{-6}. In particular, we can reserve 4x10^{-6} /15 s for the probability of false alert under fault free conditions.

Horizontal guidance

For horizontal guidance, the continuity requirement is tighter (ranging from 10^{-4} to 10^{-8} per hour). The above calculation applied to T_{exp} = 1 hour yields:

$$\sum_{j>0} P_{fault,cont,j} \leq 5 \cdot 10^{-4} \tag{94}$$

Since this exceeds the available continuity budget, it is necessary to attempt exclusion. The probability of alarm due to two or more simultaneous faults can be shown to be bound by:

$$\sum_{j \in S} P_{fault,cont,j} \leq \frac{1}{2}\left(\sum_{i=1}^{N_{sat}} P_{onset,sat,i} T_{\exp,i} + \sum_{j=1}^{N_{const}} P_{onset,const,j} T_{\exp,j}\right)^2 \leq 1.25 \cdot 10^{-7} / hour \tag{95}$$

where S is the set of fault modes formed of two or more simultaneous faults. If we assume a continuity budget of 10^{-6}, Equation (89) shows that it is sufficient to exclude single faults.

Equation (89) also shows that it is possible to adjust the thresholds within some constraints. Let us assume that for each test j, the thresholds have been chosen such that:

$$P\left(test_j = 0\right) \leq \sum_{k=0}^{N_{fault\ modes,j}} \sum_{q=1}^{3} 2Q\left(\frac{{}^{(j)}T_{k,q}}{{}^{(j)}\sigma_{ss,q}^{(k)}}\right) \leq P_{FA,j} \tag{96}$$

We will have:

$$P_{Alert} \leq \sum_{j \in J} P_{fault,cont,j} P_{FA,j} + \sum_{j \notin J} P_{fault,cont,j} \tag{97}$$

Therefore, any choice of $P_{FA,j}$ will work as long as the right hand side term in Equation(97) is below the probability of alert allocation.

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] Phase II of the GNSS Evolutionary Architecture Study, February 2010 http://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/library/documents/media/GEASPhaseII_Final.pdf

[3] 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 a Multi-Constellation Advanced RAIM", *NAVIGATION*, Vol. 60, No. 1, Spring 2013, pp. 53-69.

[4] Blanch, J., Walter, T., Enge, P. "Framework for analyzing ARAIM architectures," *Proceedings of the Institute of Navigation GNSS-12*, Nashville, September 2012.

[5] Blanch, J., Ene, A., Walter, T., Enge, P., "An Optimized Multiple Hypothesis RAIM Algorithm for Vertical Guidance," *Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2007)*, Fort Worth, TX, September 2007, pp. 2924-2933.

[6] 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.

[7] Ene, A., "Utilization of Modernized Global Navigation Satellite Systems for Aircraft-Based Navigation Integrity," Doctoral Research Dissertation, VDM VerlagDr. Müller, June 2009.

[8] DeCleene, B., "Performance Requirements for LPV Approaches to Minima as Low as 200 ft above Runway Threshold," *International Civil Aviation Organization Navigation Systems Panel (NSP), Technical Working Group (WG1) Meeting*, Brussels, Belgium, May 8 to 19, 2006

[9] 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.

[10] FAA, Advisory Circular 90-101A, "Approval Guidance for RNP Procedures with AR," February 23, 2011.

[11] WAAS Minimum Operational Performance Specification (MOPS), RTCA document DO-229D

[12] Blanch, J., Walter, T., and Enge, P.," Optimal Positioning for Advanced RAIM," *NAVIGATION*, Vol. 60, No. 4, Winter 2013, pp. 279-290.

[13] Blanch, J., Walter, T., Enge, P., Kropp, V.,"A Simple Position Estimator that Improves Advanced RAIM Performance," IEEE Transactions on Aerospace and Electronic Systems Vol. 51, No. 3, July 2015.

[14] RTCA/DO-208, "Minimum Operational Performance Standards for Airborne Supplemental Navigation Equipment Using Global Positioning System (GPS)," RTCA/DO-208, July 1991, prepared by RTCA Special Committee 159

[15] McGraw et. al, "Development of the LAAS Accuracy Models," *Proceedings of the ION GPS 2000*, Salt Lake City, UT

[16] "Airborne Supplemental Navigation Equipment Using the Global Positioning System (GPS)," Technical Standard Order (TSO) C-129, 10 December 1992, U.S. Federal Aviation Administration, Washington, D.C.

[17] 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
http://www.gps.gov/policy/cooperation/europe/2013/working-groupc/
ARAIM-report-1.0.pdf

[18] 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

[19] 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).

[20] Blanch, J. "Proposed updates to reference ARAIM airborne algorithm description document". Presented at WG-C Service Evolutions Subgroup, April 2017. Available upon request.

[21] Working Group C, ARAIM Technical Subgroup, Milestone 2 Report, Issue 1.0, February 11, 2015. Available at:
http://www.gps.gov/policy/cooperation/europe/2015/working-group-c/
http://ec.europa.eu/growth/tools-databases/newsroom/cf/itemdetail.cfm?item_id=8191