Advanced RAIM for Mega-Constellations

Juan Blanch, Sukrut Oak, Sam Pullen, Sherman Lo, Todd Walter

Stanford University

ABSTRACT

We describe a practical method to enable Advanced RAIM in mega-constellations, where the number of ranging sources and the fault rates could be very high (thus rendering standard approaches potentially impractical). The method is based on a combination of a branch-and-bound approach and on the Gershgorin circle theorem. We evaluate the performance of ARAIM in mega-constellations for fault rates that would be unfeasible with standard RAIM and ARAIM approaches. We find that for constellations where 100 to 200 satellites may be in view for a user, the proposed approach would enable the computation of finite and useful Protection Levels with fault rates as high as 10^{-2} per hour.

INTRODUCTION

Mega-constellations like OneWeb or Starlink could revolutionize satellite radionavigation by providing a very large number of high-power ranging signals. The number of ranging signals visible at any time by a user could increase from about forty now to more than a hundred.

In addition to the gains in accuracy, resilience, and availability, this increased redundancy can be exploited to compute error bounds on the estimated position. For error bounds with integrity, we will need a guarantee of performance for the ranging signals (either through service history or commitments from service providers), but with so many of them, we might be able to tolerate high fault rates.

Advanced Receiver Autonomous Integrity Monitoring (ARAIM) for multi-constellation provides a framework to compute integrity error bounds for any number of ranging signals with a known bound on the probability of fault (which may be different depending on the constellation and satellite). In theory, the ARAIM baseline algorithm could deal with arbitrarily high fault probabilities and any number of satellites, but such an approach might not be possible in practice. The ARAIM user algorithm monitors a set of faults and combinations of faults such that all other combinations of faults can be considered unlikely enough to be left un-monitored (Blanch (2015a)). For example, for a dual constellation GPS Galileo, the probabilities of fault lead the ARAIM user algorithm to monitor most dual faults (for the default constellation performance commitments).

In mega-constellations, two characteristics may cause the number of monitored faults to increase beyond what may be computationally practical: first, the large number of signals used in the solution could (beyond a 100), and second the potentially large probabilities of fault of the new satellites (since navigation is not the main mission of these satellites). This will result in a very large number of fault combinations. For example, with 50 ranging signals and a probability of fault of 10^{-3} per satellite, the algorithm might need to monitor all the simultaneous quadruple faults, of which there are more than two million. While this does not mean that the algorithm must compute as many subset solutions, it does mean that it needs to compute a bound on the worst subset standard deviation.

In our earlier work, we have described methods: 1) to identify many outliers (Blanch (2015b)) and 2), to compute PLs for fault detection when the probabilities of fault were large enough to have to consider multiple simultaneous faults that greatly reduced the computational load compared to standard approaches (Blanch (2021)). In this paper, we combine this method with a branch-and-bound approach.

More precisely, we provide practical formulas and methods to compute:

- The probabilities of the events that are not strictly mitigated by the redundancy check
- The prior probabilities of *p* simultaneous faults
- Upper bounds on the subset sigmas or, equivalently, the fault slopes for multiple simultaneous faults

All these quantities are key inputs to the computation of a protection level, which we also provide.

In the last section, using the service volume availability tool MAAST, we will show that it might be possible and practical to provide useful integrity error bounds (at the 10^{-7} level, for example) with satellite fault rates of 10^{-3} per hour or even higher in constellations with 200 satellites in view.

ADVANCED RAIM FRAMEWORK

In the ARAIM framework, the pseudorange error model is described by a discrete number of fault hypotheses H_k, where H_0 is the fault free hypothesis. Each fault hypothesis corresponds to the addition of an additional unknown state b_k in one or more measurements and has a known prior fault rate $R_{fault,k}$ and mean fault duration $T_{MFD,k}$. In the implementation of ARAIM concept for aviation, each of the fault hypotheses is derived from a set of primary faults. These primary faults correspond to single satellite faults and single constellation faults. These primary faults are assumed to be independent, thus enabling the derivation of the fault rates and mean fault durations for the composite faults.

STATE PROBABILITIES FOR COMPOSITE FAULTS

We consider *n* primary events. In our case, these will be satellite faults and constellation wide faults, as described in the ARAIM framework. An event is the occurrence of a fault in one or a subset of measurements. Mathematically, this is expressed by the addition of a new state in the observation equation. We will assume that these primary events are mutually independent. As a result, the probability of occurrence of simultaneous events is the product of the probability of those events. More precisely, let us consider *m* primary events i_1 through i_m. The probability of occurrence of these events (at least) is:

$$P\left(\text{at least } i_1,\cdots,i_m\right)=\prod_{k=1}^{m} p_{i_k}$$

The probability of occurrence of these events and only these is given by:

$$P\left(i_1,\cdots,i_m\right)=\prod_{k=1}^{m} p_{i_k}\prod_{j\neq i_k}\left(1-p_k\right) \tag{1}$$

Equation (1) is the formula that is used to compute the state probability of each composite fault $p_{fault,k}$ that will be used in the protection level (PL).

The probability of no faults is a particular case of the above formula. It is given by

$$P_0 = \prod_{k=1}^{n} \left(1 - p_k\right)$$

With this definition, we can re-write the probability as

$$P\left(i_1, \cdots, i_m\right) = \frac{\prod_{k=1}^{m} p_{i_k}\left(1 - p_{i_k}\right)}{P_0} = \frac{\prod_{k=1}^{m} q_{i_k}}{P_0} \quad (2)$$

Equation (2) is a form of Equation (1) that will be practical.

Let us now compute the sum of the probabilities of all events composed of *m* primary events. This probability will be later used to reduce the computational load compared to the baseline ARAIM approach. We need to sum over all combinations of m primary events:

$$P\left(\text{exactly } m \text{ primary events}\right) = \sum_{i_1 < \cdots < i_m} P\left(i_1, \cdots, i_m\right) = \frac{1}{P_0} \sum_{i_1 < \cdots < i_m} \prod_{k=1}^{m} q_{i_k}$$

There are $\begin{pmatrix} n \\ m \end{pmatrix}$ terms in this sum. To compute this sum, we use the Newton identities. The sum

$$e_m\left(q_1, \cdots, q_n\right) = \sum_{i_1 < \cdots < i_m} \prod_{k=1}^{m} q_{i_k}$$

is the k^{th} elementary symmetric polynomial. These polynomials can be expressed as a function of the power sums:

$$S_k\left(q_1, \cdots, q_n\right) = \sum_{i=1}^{n} {q_i}^k$$

The first four are:

$$e_1 = S_1$$

$$e_2 = \frac{1}{2}\left({S_1}^2 - S_2\right)$$

$$e_3 = \frac{1}{6}\left({S_1}^3 - 3S_2S_1 + 2S_3\right)$$

$$e_4 = \frac{1}{24}\left(S_1^4 - 6S_1^2S_2 + 3S_2^2 + 8S_1S_3 - 6S_4\right)$$

There is a general formula, but it is practical to compute them iteratively using the recursion formula

$$ke_k = \sum_{i=1}^{k} \left(-1\right)^{i-1} e_{k-i} S_i$$

Using these formulas, it is straightforward to compute the probability of exactly *k* events or an upper bound on the probability of at least *k* events. Note that the formula does not require all the probabilities to be equal.

A very practical upper bound is given by

$$\sum_{i_1<\cdots<i_m} \prod_{k=1}^{m} q_{i_k} \le \frac{S_1^m}{m!} = \frac{\left(\sum_{i=1}^{n} q_i\right)^m}{m!}$$

This formula is very accurate for relatively small values of *q*, and it is the one that we use in our code. It will also be useful to have formulas for the probability of having at least m events. Using the same approach, we have, with the same derivation:

$$P(\text{at least } m \text{ primary events}) = \sum_{i_1<\cdots<i_m} \prod_{k=1}^{m} p_{i_k} = e_m(p_1,\cdots,p_n) \le \frac{\left(\sum_{i=1}^{n} p_i\right)^m}{m!} \tag{3}$$

FAULT RATE PROBABILITIES FOR COMPOSITE FAULTS

One way to determine a sufficient list of monitored modes is by ensuring that all composite faults with more than m+1 primary events have a total probability of occurring (during the exposure time) that is only a fraction of the total integrity budget PHMI.

Let us now consider one fault mode with fault rate *R* and mean fault duration *M*. The probability of having a fault in an interval T_{EXP} is bounded by (and very well approximated by)

$$P(\text{fault in interval of length } T_{EXP}) \le R(T_{MFD} + T_{EXP}) = P\left(1 + \frac{T_{EXP}}{T_{MFD}}\right)$$

An upper bound on the fault rate of a composite fault composed of two primary faults 1 and 2 or more is given by

$$R = P_1 P_2 \left(\frac{1}{T_{MFD,1}} + \frac{1}{T_{MFD,2}}\right)$$

The corresponding mean fault duration is given by

$$T_{MFD} = \left(\frac{1}{T_{MFD,1}} + \frac{1}{T_{MFD,2}}\right)^{-1}$$

These formulas can be generalized to composite faults with *m* primary faults as follows Blanch 2020, Milner 2020:

$$R(i_1,\cdots,i_m) = \prod_{k=1}^{m} p_{i_k} \left(\sum_{k=1}^{m} \frac{1}{T_{MFD,i_k}}\right)$$

The mean fault duration is given by

$$T_{MFD,(i_1,\cdots,i_m)} = \left(\sum_{k=1}^{m} \frac{1}{T_{MFD,i_k}} \right)^{-1}$$

To compute the probability of having more than m faults over an interval T_{EXP}, we sum over all the sets with *m* events

$$P(m \text{ or more faults in interval of length } T_{EXP}) \le \sum_{i_1 < \cdots < i_m} \prod_{k=1}^{m} p_{i_k} \left(1 + T_{EXP} \left(\sum_{k=1}^{m} \frac{1}{T_{MFD,i_k}} \right) \right)$$

We cannot directly apply the Newton formulas above because of the term containing the mean fault durations. We can however compute an upper bound by considering the m lowest mean fault durations. The composite fault corresponding to these primary faults is the one with the shortest mean fault duration. We note it $T_{MFD,m,low}$. Using the approach above, we get:

$$P(m \text{ or more faults in interval of length } T_{EXP}) \le e_m(p_1,\cdots,p_n)\left(1+\frac{T_{EXP}}{T_{MFD,m,low}}\right)$$

As before with the state probabilities, we can bound it by

$$P(m \text{ or more faults in interval of length } T_{EXP}) \le \frac{\left(\sum_{i=1}^{n} p_i\right)^m}{m!}\left(1+\frac{T_{EXP}}{T_{MFD,m,low}}\right)$$

Now we note:

$$P_m(T_{EXP}) = e_m(p_1,\cdots,p_n)\left(\frac{1}{T_{EXP}}+\frac{1}{T_{MFD,m,low}}\right) \tag{4}$$

This is a bound on the probability of occurrence of any m simultaneous primary events.

Important case: identical mean fault duration

In the case all satellites have the same mean fault duration we can use the formula

$$P(m \text{ or more faults in interval of length } T_{EXP}) = \frac{\left(\sum_{i=1}^{n} p_i\right)^m}{m!}\left(1+m\frac{T_{EXP}}{T_{MFD}}\right) \tag{5}$$

DETERMINATION OF FAULTS MODES TO BE MONITORED

There are as many fault modes are there are combinations of primary faults. It is however not necessary to monitor them all, because many of them have probabilities that are very small. To limit the number of monitored modes, we determine the smallest integer *m* such that

$$P_{m+1}\left(T_{EXP}\right) \le \alpha P_{HMI}$$

where

α is a tunable parameter between zero and one

P_{HMI} is the available integrity budget

This can be done with an iteration starting at m=0 using Equation (5).

Once *m* is determined, the list of monitored modes is simply the list of faults composed of *m* or less primary modes (be they constellation wide faults or satellite faults).

MONITORING THE FAULT MODES AND PROTECTION LEVEL EQUATION

The Advanced RAIM algorithm described in Blanch 2015 uses solution separation to monitor the fault modes. That is, it tests that for all

$$\left|\hat{x}_q^{(k)} - \hat{x}_q^{(0)}\right| \le T_{k,q} \text{ with } T_{k,q} = K_{fa,q}^{(k)} \sigma_{ss,q}^{(k)}$$

where

$\hat{x}_q^{(k)}$ is a fault tolerant position solution for fault mode *k*

$\sigma_{ss,q}^{(k)}$ is the standard deviation of the solution separation $\hat{x}_q^{(k)} - \hat{x}_q^{(0)}$ under nominal conditions $T_{k,q}$ is a scalar set to meet a pre-determined false alert rate under fault free conditions.

The protection level in the coordinate *q*, PL_q, can be computed by solving the equation:

$$2N_{es}Q\left(\frac{PL_q}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{\text{fault modes}}} N_{es}\, p_{fault,k}\, Q\left(\frac{PL_q - T_{k,q}}{\sigma_q^{(k)}}\right) = P_{HMI,q} \tag{6}$$

where:

$p_{fault,k}$ is the prior probability of fault hypothesis H_k

$p_{fault,k}$ (T_{EXP}) is the prior probability of fault hypothesis H_k occurring at any time during the exposure time T_{EXP}

$PHMI_q$ is the integrity allocation to the q^{th} coordinate

N_{es} is the number of effective samples per hour (which translates the probabilities over a finite interval to per event probabilities. In the baseline ARAIM algorithm, it is determined by the required time to alert)

Consistency checks

The sum in Equation (6) can potentially contain up to millions of terms for satellite fault rates of 10^{-3} per hour and 100 satellites in view. In addition, each of these terms corresponds to the computation of a fault tolerant solution. This may be a problem, even for future processors. To reduce the computational load, we need a way to reduce the number of terms in this Equation and the number of fault detection tests. To achieve this, we will use the approaches described in [1] with a few modifications.

Reducing the number of tests

The number of tests can be easily reduced by exploiting the relationship between the normalized solution separation statistics and the sum of squared normalized residuals. More precisely, we have

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

where

W is the inverse of the covariance of the measurements (assumed to be diagonal)

G is the observation matrix

y is the vector of measurements (linearized about the solution)

This means that we can replace any number solution separation tests by the sum of squared residuals test (although we don't need to replace them all). If we set a threshold $K_{fa}^{\chi^2}$ and ensure that

$$\sqrt{y^T \left(W - WG\left(G^T WG\right)^{-1} G^T W\right) y} \leq K_{fa}^{\chi^2}$$

Then we are sure that we also have for any solution separation test

$$\frac{\left|\hat{x}_q^{(k)} - \hat{x}_q^{(0)}\right|}{\sigma_{ss,q}^{(k)}} \leq K_{fa}^{\chi^2}$$

Let us suppose for now that we check all the subsets using this method. The corresponding PL equation is now:

$$2N_{es} Q\left(\frac{PL_q}{\sigma_q^{(0)}}\right) + \sum_{k=1}^{N_{\text{fault modes}}} N_{es} p_{fault,k} Q\left(\frac{PL_q - K_{fa}^{\chi^2} \sigma_{ss,q}^{(k)}}{\sigma_q^{(k)}}\right) = P_{HMI,q}$$

While we have reduced the number of tests, we still have all the terms in the sum, so we still need to compute the fault tolerant standard deviations $\sigma_q^{(k)}$. Note that if we have $\sigma_q^{(k)}$, we can compute $\sigma_{ss,q}^{(k)}$ using the relationship:

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

The next step is to find bounds on $\sigma_{ss,q}^{(k)2}$ for groups of satellites. Let us suppose that we have *m* such groups, identified each by a set of indices Ω_j for *j* from 1 to m. For each of these groups we have a $\sigma_{ss,q}^{(\Omega_j)}$ such that for each k in Ω_j, we have

$$\sigma_{ss,q}^{(k)2} \leq \sigma_{ss,q}^{(\Omega_j)}$$

Because each of the terms in the PL sum are monotonously increasing functions of $\sigma_{ss,q}^{(k)}$, we have

$$2N_{es}Q\left(\frac{PL_q}{\sigma_q^{(0)}}\right)+\sum_{j=1}^{m}N_{es}p_{fault,k}Q\left(\frac{PL_q-K_{fa}^{\chi^2}\sigma_{ss,q}^{(\Omega_j)}}{\sigma_{ss,q}^{(\Omega_j)}}\right)\leq 2N_{es}Q\left(\frac{PL_q}{\sigma_q^{(0)}}\right)+\sum_{k=1}^{N_{\text{fault modes}}}N_{es}p_{fault,k}Q\left(\frac{PL_q-K_{fa}^{\chi^2}\sigma_{ss,q}^{(k)}}{\sigma_q^{(k)}}\right)$$

Therefore, the PL defined by

$$2N_{es}Q\left(\frac{PL_q}{\sigma_q^{(0)}}\right)+\sum_{j=1}^{m}N_{es}\left(\sum_{k\in\Omega_j}p_{fault,k}\right)Q\left(\frac{PL_q-K_{fa}^{\chi^2}\sigma_{ss,q}^{(\Omega_j)}}{\sigma_q^{(\Omega_j)}}\right)=P_{HMI,q} \qquad (7)$$

is valid. The important point here is that now there are only *m* terms in this sum. In the next two sections we describe how to obtain the terms in this sum.

Choice of fault groups

We will start describing the groups for the case where the constellation wide faults can be left un-monitored (that is, that their cumulative prior probability is below the available integrity budget). In that case, each group Ω_j is formed by all composite faults formed using *j* primary events. For example, if *j*=2, the group contains all the dual satellite faults. This is the approach in its simplest form.

Bound on the standard deviation of the fault tolerant solutions using Gershgorin circle theorem and branch-and-bound approach

The key to the approach described in this paper is the use of a computationally efficient method to compute the upper bound $\sigma_{ss,q}^{(\Omega_j)}$. We will partly use the upper bound developed in Blanch 2021, which is based on the Gershgorin circle theorem. This bound was shown to be several orders of magnitude faster than computing the list of standards deviations, even when applying rank-one updates. However, a direct application of these bounds can result in poor performance, because the bounds can be excessively conservative. The main contribution of this work is to combine this upper bound with a branch-and-bound approach.

First, let us define the function f that provides an upper bound over a class of subsets:

$$\sigma_{upper\_bound}=f\left(G,W,I,J,k\right)$$

where

I is the set of indices that are used

J = $\{j_1,\ldots,j_m\}$ is the set of indices out which we remove measurements (J must be a subset of I)

k is the number of simultaneous faults removed out of J (k must be smaller or equal to the number of elements in J)

For example, if J =[1:n], and k = 1, the function will return an upper bound of all the one-out subsets. If K = [2:n], and k=2, the function will return an upper bound of all the 2-out subsets where satellite one is never removed.

The function f is defined recursively as follows. The upper bound is initially computed using the minimum of the sigma obtained from:

- The Gershgorin bound (restricted to J)
- The subset that excludes all measurements in J (this can only be computed when I\J has sufficient satellites)

If this bound is too large (here a tunable threshold is used), for example compared to the all-in-view sigma, then the algorithm divides the subsets in as many branches as there are elements in *J*. The first branch bounds all the subsets that include j_1 (out of the all the subsets defined by J and k). This is computed by calling

$$\sigma_{upper\_bound} = f\left(G, W, I, J \setminus j_1, k\right)$$

The second branch bounds all the subsets with j_1 out and with j_2. This is computed by calling

$$\sigma_{upper\_bound} = f\left(G, W, I \setminus j_1, J \setminus \{j_1, j_2\}, k-1\right)$$

The p-th branch is computed by calling

$$\sigma_{upper\_bound} = f\left(G, W, I \setminus \left\{j_1, \cdots, j_{p-1}\right\}, J \setminus \left\{j_1, \cdots, j_p\right\}, k-p+1\right)$$

To illustrate the branching mechanism, if 0 means a measurement out and 1 a measurement in, the different branches correspond to the following rows

$$\begin{matrix} 1 & * & * & * & * \\ 0 & 1 & * & * & * \\ 0 & 0 & 1 & * & * \\ 0 & 0 & 0 & 1 & * \\ \cdots & \cdots & \cdots & \cdots & \cdots \end{matrix}$$

As a refinement, we can treat each sub-group bounded by in a branch as a term in the PL equation by combining this computation with the probability of subset fault.

Example for a set of k-out subsets

Figure 1 shows the histogram of 3-out subset sigmas for two examples (with 149 satellites and 100 satellites). These histograms were obtained by listing every possible 3-out geometry and computing the corresponding sigma. There are respectively 540274 and 152096 subsets. The algorithm proposed above returns an upper bound for all these subsets instead of computing all the subset sigmas (respectively 0.2045 and 1.075). For large number of satellites, the bounds provided by the algorithm are often extremely tight, as in these two cases.

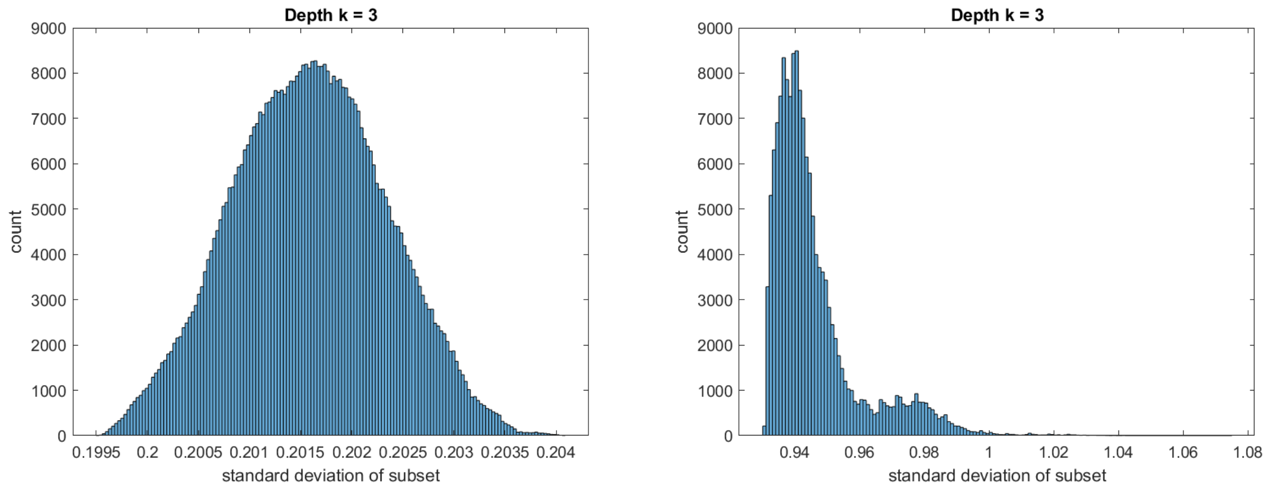

Figure 1. *Standard deviations ($\sigma^{(k)}$) of 3-out subsets for N_{sat} = 149 (left) and N_{sat} = 100 (right). There are respectively 540274 and 152096 subsets in these histograms. The proposed algorithm provides an upper bound of 0.2045 (left) and 1.075 (right)*

COMPUTATIONAL LOAD ESTIMATES

Figure 2 shows the number of subsets that would need to be monitored in the baseline algorithm as a function of the probability of satellite p_{sat} and the number of satellites in view. The criterion used to determine the depth of the subsets is the one described in the section "determination of fault modes to be monitored". For this plot, we used a target integrity risk of 10^{-7} per hour, and a mean fault duration of one hour is assumed for all primary faults.

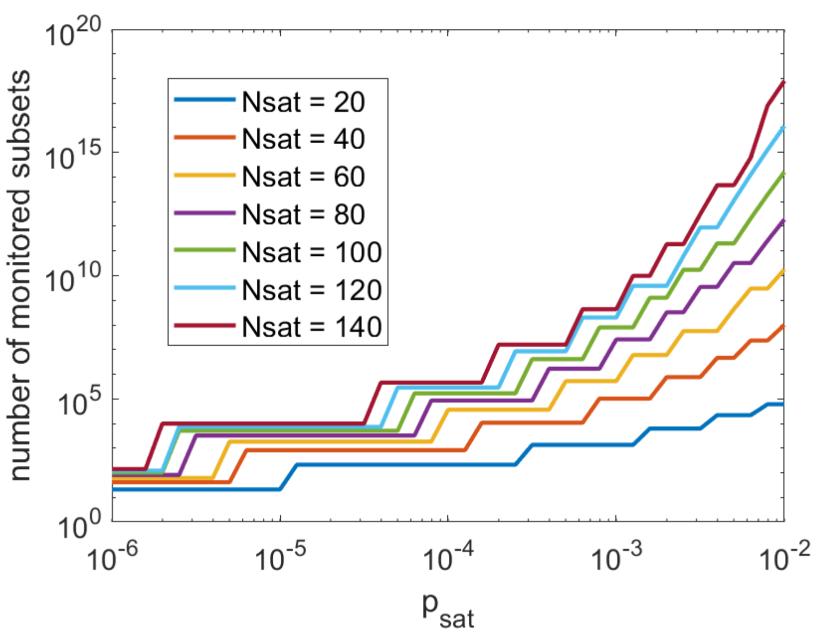

Figure 2. *Number of fault modes to be monitored as a function of p_{sat} and the number of satellites in view (Nsat). For number of satellites above 40 or p_{sat} above 10^{-3}, the baseline algorithm is impractical or even unfeasible.*

To illustrate the power of the proposed approach, Table 1 shows execution times (observed and estimated) for three geometries. The first row corresponds to values typical for a dual constellation ARAIM scenario, where about 100 subsets must be monitored. The computation of the PL took on the order of 10^{-3} s in a PC.

For the second row, a p_{sat} of 10^{-3} and Nsat = 100 leads to a number of monitored subsets on the order of 10^{7}. Because the time it takes to compute PL is approximately proportional to the number of subsets (the baseline algorithm computes the sigma for each subset successively), this would lead to an execution time of 100 s. In the third row, we increase p_{sat} to 10^{-2} and Nsat to 150, which leads to as many as 10^{17} monitored subsets. With our PC, this would take on the order 0f 10000 years to compute.

The last column of Table 1 shows the time it took with the proposed approach, which is only slightly more than in the dual constellation ARAIM scenario with low p_{sat}, which means that it is both feasible and practical.

Table 1. *Computational load estimates*

Number of satellites	Probability of satellite fault	Number of monitored subsets	Execution time for one geometry for baseline approach	Proposed approach
20	10^{-4}	100	10^{-3} s	10^{-3} s
100	10^{-3}	10^{7}	100 s	2 x 10^{-3} s
150	10^{-2}	10^{17}	10^{12} s ~ 30000 years	4 x 10^{-3} s

AVAILABILITY SIMULATIONS

We demonstrate the feasibility of this approach using an almanac corresponding to the Starlink constellation. This is a constellation that has more than 5000 satellites at altitudes between 330 km and 550 km with inclinations between 40 and 98 degrees. This constellation was chosen because the very large number of satellites would render the direct application of a baseline ARAIM algorithm unfeasible with high or even moderate fault rates.

For the ranging accuracy we made used the same models used for GPS with a URA of 1.5 m. We also assumed an elevation masking angle of 5 degrees.

The simulations were performed using the Stanford MAAST simulation tool with a 10 by 10 degree user grid and 300 time steps over 24 hours.

The HPL was computed as described in Equation (7). We will be using a target PHMI of 10^{-7} per hour, of which 90% are allocated to the modes that are not monitored (α parameter defined above). We will use a false alarm of 10^{-6} per hour.

We evaluated two scenarios with very high fault rates: 10^{-3} and 10^{-2} per hour. This is 100 to 1000 times worse than what GPS currently provides. Our purpose here is to demonstrate that we can compute a useful PL even with these high fault rates. Since we are only considering one constellation, we assumed that constellation wide faults could be neglected.

Table 2. *Integrity Support Data assumptions*

$P_{const,default}$	1×10^{-8}
$P_{sat,default}$	1×10^{-3}, $1\text{x}10^{-2}$
$R_{const, default}$	1×10^{-8}/h
$R_{sat, default}$	1×10^{-3}/h
$MFD_{const, default}$	1 hour
$MFD_{sat, default}$	1 hour
$\sigma_{URA,default, dual frequency}$ [m]	1.5 m
N_{ES}	450

In Figure 3, we show the HPL for one user over a period of 24 hours. A typical user would have between 100 and 200 satellites in view at any given time. With the criteria used to determine the list of monitored fault modes, a standard ARAIM algorithm would monitor all the fault modes composed of 6 or less primary faults. There are up 10^{10} such fault modes, which makes it unfeasible to apply it. With the approach described in this paper, it is not only possible but quite fast to compute the PL, since there are only 7 terms in the PL (Equation (7)). In our simulations, the computation of one PL took less than 0.002 seconds for 10^{-3} and less than 0.005 for 10^{-2}.

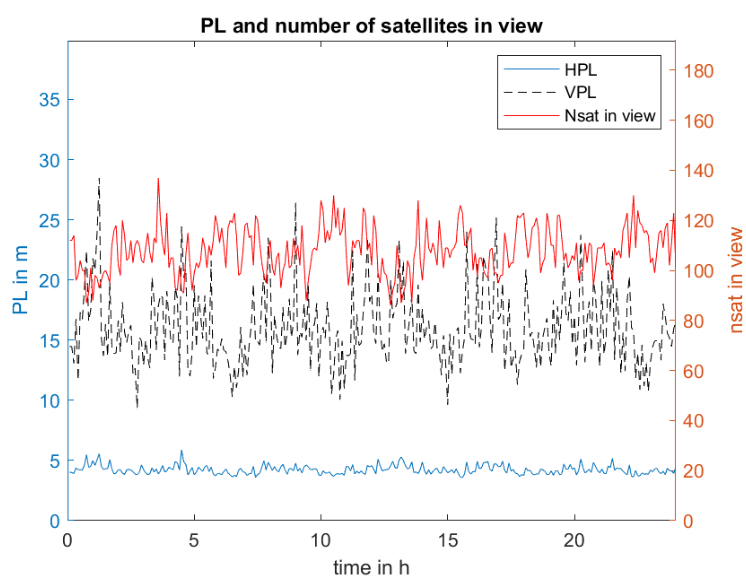

Figure 3. PL for a user located at lat. and lon. (0,0). For this user all simultaneous faults with 5 or less faults are monitored. There are about 7.5 x 10^7 such modes.

Figure 4 and 5 show a maps of the PL corresponding to the 99.5% percentile of the PLs over 24 hours for p_{sat} = 10^{-3} and 10^{-2}. For p_{sat} = 10^{-3}, the HPLs are below 10 m in most of the globe and the VPLs are mostly below 25 m (recall that we have assumed an arbitrary URA of 1.5 m). For $p_{sat} = 10^{-2}$, the PLs increase significantly, they are however still finite, and potentially useful. The main point here is that we can compute a PL in the first place in a practical way.

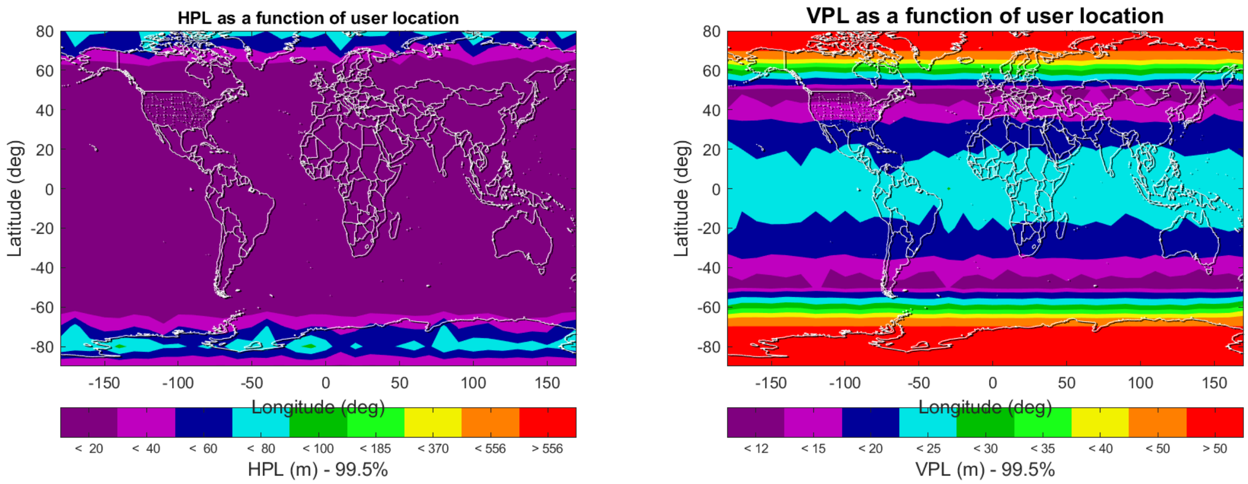

Figure 4. *99.5% Fault Detection PL maps for geometries corresponding to Starlink with fault rates of 10^{-3} per hour.*

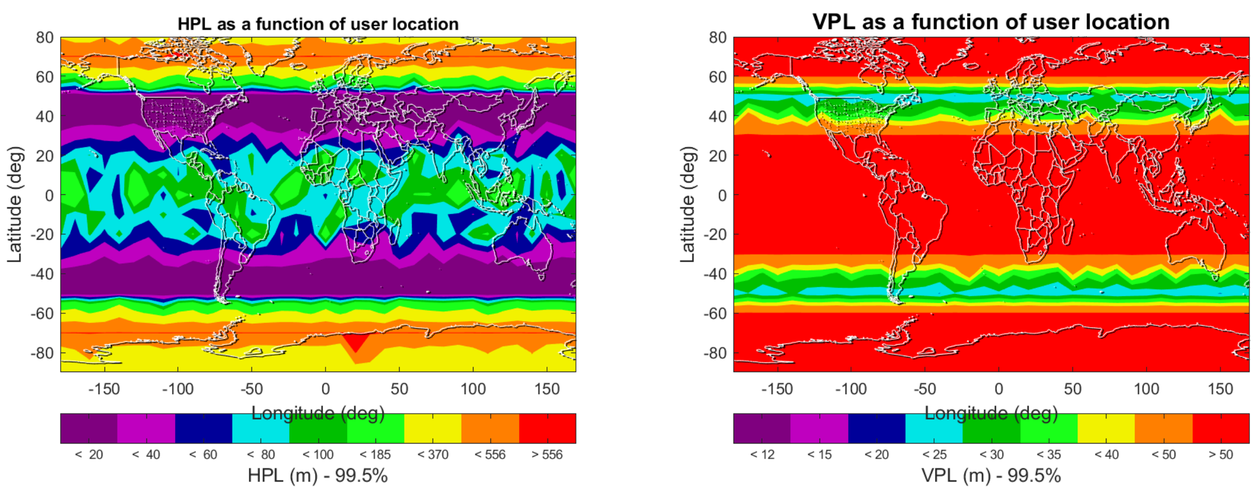

Figure 5. *99.5% Fault Detection PL maps for geometries corresponding to Starlink with fault rates of 10^{-2} per hour.*

A NOTE ON EXCLUSION FUNCTION

The PLs described above correspond to a fault detection PL. They are the PLs that would be obtained if no exclusion was attempted. If the consistent set used in final position fix was the result of an exclusion process, the formulation of the PL we would simply need to correct the integrity allocation in Equation (7) to account for the multiple risk exposure caused by the exclusion process (as it is done in Blanch 2021 and the Advanced RAIM reference algorithm). The search of the consistent set could be performed using for example a greedy search, as described in Blanch 2015b or Wendel 2022.

SUMMARY

We have described an implementation of ARAIM that enables the computation of integrity error bounds for mega-constellations with large fault rates. The key to the approach is to reduce the number of terms in the Protection Level equation by computing upper bounds on the standard deviations of the subset solutions rather than computing them explicitly. An example based on the Starlink constellation shows that, with this approach, it would be possible to compute useful PLs even when the probability of up to 6 simultaneous faults cannot be neglected.

REFERENCES

Advanced RAIM Technical Subgroup Reference Airborne Algorithm Description Document, v4.2 February 2023

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.

Blanch, J., Walter, T., Enge, P., " Fast Multiple Fault Exclusion with a Large Number of Measurements," *Proceedings of the 2015 International Technical Meeting of The Institute of Navigation*, Dana Point, California, January 2015.

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.

Blanch, J., and Walter, T., "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.

Blanch, Juan, Walter, Todd, Milner, Carl, Joerger, Mathieu, Pervan, Boris, Bouvet, Denis, "Baseline Advanced RAIM User Algorithm: Proposed Updates," Proceedings of the 2022 International Technical Meeting of The Institute of Navigation, Long Beach, California, January 2022, pp. 229-251. https://doi.org/10.33012/2022.18254

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.

Oak, S. et al “GNSS Augmentation by Low-Earth-Orbit (LEO) Satellites: Integrity Performance Under Non-Ideal Conditions,” in Proceedings of ION GNSS+2023, September 2023.

“Newton-Girard identities” available at: https://mathworld.wolfram.com/Newton-GirardFormulas.html

Wendel, Jan, "GNSS Pseudorange Fault Detection and Exclusion with Multiple Outliers," *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, Denver, Colorado, September 2022, pp. 1481-1495.
https://doi.org/10.33012/2022.18421