Secure Multi-constellation GNSS Receivers with Clustering-based Solution Separation Algorithm

Kewei Zhang
Networked Systems Security Group
KTH Royal Institute of Technology
kewei@kth.se

Panos Papadimitratos
Networked Systems Security Group
KTH Royal Institute of Technology
papadim@kth.se

Abstract*—Because of the limited satellite visibility, reduced signal reception reliability and constraining spatial geometry, e.g., in urban areas, the development of multi-constellation global navigation satellite systems (GNSS) has gained traction rapidly. GNSS-based applications are expected to handle observations from different navigation systems, e.g., GPS, GLONASS, BeiDou and Galileo, in order to improve positioning accuracy and reliability. Furthermore, multi-constellation receivers present an opportunity to better counter spoofing and replaying attacks, leveraging approaches take advantage of the redundant measurements. In particular, cluster-based solution separation algorithm (CSSA) proposes to detect and identify faulty/malicious signals in a single GPS constellation by checking the consistency of receiver positions calculated with different number of satellites. Intuitively, the algorithm targets directly the consequence of spoofing/replaying attacks: the victim receiver position error estimation. It works independently of how the attacks are launched, either through modifying pseudorange measurements or manipulating the navigation messages, without changing the receiver hardware. Multi-constellation GNSS receivers utilize all observations from different navigation systems, there are more than 30 available satellites at each epoch after Galileo and BeiDou systems become fully operational; in other words using abundant redundancy. Therefore, we introduce such a CSSA to a multi-constellation receiver. The work shows that a multi-constellation GNSS receiver equipped with our algorithm works effectively against a strong spoofing/replaying attacker that can manipulate a large number of signals, or even an entire constellation. The results show that CSSA with multi-constellation significantly improves the performance of detecting and identifying the malicious signals; particularly, when the adversary cannot control all the constellations, a multi-constellation receiver can identify the faults even the adversary induces very small errors to pseudorange measurements, comparing with a single constellation receiver. Moreover, when the attacker is powerful to manipulate most of signals of all the constellations, a multi-constellation receiver with CSSA can still detect and identify the faulty signals with high probability when the attacker tries to mislead the victim more than a couple of hundred meters from its true location.

TABLE OF CONTENTS

978-1-5386-6854-2/19/$31.00

1. INTRODUCTION

In the near future, when Galileo is operational and BeiDou completes its global coverage, there will be at least four global navigation satellite systems (GNSS), including GPS and GLONASS, providing global positioning, navigation and timing (PNT) services. Moreover, several countries are developing or have developed regional navigation satellite systems, such as Indian Regional Navigation Satellite System (IRNSS), Quasi-Zenith Satellite System (QZSS) by Japan, Doppler Orbitography and Radio-positioning Integrated by Satellite (DORIS) by France, and German precise and range rate equipment (PRARE) [1]. Currently, for any given point on the earth, a GNSS receiver can calculate its PVT with two full operational global navigation satellite systems: the GPS and the GLONASS. With more operational global navigation satellite systems in near future, notably Galileo and Beidou, there will be more than 30 available satellites for a receiver to use for navigation at most of the time [2], [3]. This enhances the GNSS receiver accuracy, integrity and availability, while the receivers can access the satellite signals of several constellations at the same time.

The security of a GNSS receiver is important, especially the ability to detect malicious signals and prevent a faulty position, velocity and time (PVT) solution [4], [5], induced by an adversary, not only for single constellation GNSS receivers, but also for multi-constellation receivers. In this paper, we care about replay and spoofing attacks that can change the receiver's PVT result, not jamming attacks that simply prevents the receivers from receiving GNSS signals. Attacks can be classified roughly as: 1) the attacker generates its own signals to mislead the victim, probably with estimated satellite positions; this type of attacker does not necessarily consider the detection schemes at the victim receiver; 2) the attacker transmits pre-recorded GNSS signals to the victim that overshadows authentic signals; 3) the attacker estimates real-time signal features, such as code phase and Doppler frequency, etc., and then transmits the newly constructed signals to the victim [6], [7]. The difference between a single constellation attacker and a multi-constellation attacker is that it requires more power, more knowledges about different constellations and more equipment for the later attacker to manipulate signals from several constellations.

The clustering-based solution separation algorithm (CSSA) was proposed to detect and identify faulty signals, and was evaluated for a GPS receiver [8], [9]. The algorithm is based on checking consistency of receiver positions that are calculated with different subsets of signals due to the redundancy. The algorithm is assumption-verification driven; particularly, under an assumption of combination of certain number of faults, the algorithm tries to cluster the position results that are calculated with different subsets; then the subsets that do not involve any presumed faulty signals should form a cluster, if such cluster could be found, the algorithm

concludes that the faulty signals are found; otherwise the algorithm concludes that the assumed faulty signals are not real faulty ones. Therefore, it means that the more available signals the receiver has, the more powerful the algorithm can be and the higher capability the algorithm can have to identify more faulty signals. Hence, CSSA can be more powerful when a GNSS receiver can use several constellation satellites.

Receiver autonomous integrity monitoring (RAIM) [10], [11] is an important method used in safety-critical GPS applications, which aims to detect satellite failures by checking consistency of pseudorange measurements because of redundant satellites. Articles of [8], [9] have evaluated the performance difference between CSSA with RAIM, and compared the two methods based on different number of faults and numeric error values. In this paper, we look into the facts how CSSA improves a receiver's performance against an adversary with different capacities from single constellation to multi-constellation.

The rest of the paper is organized as: Section 2 gives reviews of the multi-constellation GNSS receiver and the fundamentals of CSSA; Section 3 presents an adversary model for a multi-constellation GNSS receiver; afterwards, Section 4 illustrates how CSSA can be applied to the aforementioned receiver in order to detect and identify the forged signals; the evaluation results based on Matlab simulation are given in Section 5, followed by summary and conclusions at Section 6.

2. BACKGROUND

This section first provides the fundamentals of a GNSS receiver with multi-constellation capability, showing the difference between calculating a receiver's state with a single and a multi-constellation. Thereafter, we give the review of the algorithm, CSSA, including the outline, advantages and disadvantages.

Multi-constellation GNSS Receiver

A multi-constellation GNSS receiver position calculation can be modeled like a single constellation receiver as following:

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{v} \tag{1}$$

where $\boldsymbol{y}$ is the pseudorange measurements of several constellations, $\boldsymbol{H}$ denotes the observation matrix corresponding to the constellations, $\boldsymbol{x}$ is the receiver state, i.e., position and clock errors that can be written as $[x, y, z, \Delta t_1, ..., \Delta t_k]$, in which Δt_k is the clock error of the k^{th} constellation, and $\boldsymbol{v}$ is Gaussian noise. Therefore, in order to obtain the receiver's position, it requires at least $3 + n$ satellites, where n is the number of constellations and at least one satellite from each constellation.

During a receiver position calculation in a single constellation, e.g., the GPS, the observation matrix $\boldsymbol{H}$ is constructed at the first iteration while calculating the receiver's state using the least squares method:

$$H_0 = \begin{bmatrix} -\frac{X^1-X_0}{\rho_0^1} & -\frac{Y^1-Y_0}{\rho_0^1} & -\frac{Z^1-Z_0}{\rho_0^1} & 1 \\ -\frac{X^2-X_0}{\rho_0^2} & -\frac{Y^2-Y_0}{\rho_0^2} & -\frac{Z^2-Z_0}{\rho_0^2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ -\frac{X^k-X_0}{\rho_0^k} & -\frac{Y^k-Y_0}{\rho_0^k} & -\frac{Z^k-Z_0}{\rho_0^k} & 1 \end{bmatrix} \tag{2}$$

where the subscript 0 indicates the iteration number, k is the satellite number and the last column is for the clock error. For a multi-constellation GNSS receiver, we need to add a clock error element for each constellation, then the new observation matrix $\boldsymbol{H}$ is constructed for the first iteration as:

$$H_0 = \begin{bmatrix} -\frac{X_G^1-X_0}{\rho_{G,0}^1} & -\frac{Y_G^1-Y_0}{\rho_{G,0}^1} & -\frac{Z_G^1-Z_0}{\rho_{G,0}^1} & 1 & 0 & 0 \\ -\frac{X_E^2-X_0}{\rho_{E,0}^2} & -\frac{Y_E^2-Y_0}{\rho_{E,0}^2} & -\frac{Z_E^2-Z_0}{\rho_{E,0}^2} & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -\frac{X_R^k-X_0}{\rho_{R,0}^k} & -\frac{Y_R^k-Y_0}{\rho_{R,0}^k} & -\frac{Z_R^k-Z_0}{\rho_{R,0}^k} & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

where the subscript G, E and R indicates the constellation, and the fourth column is for the GPS clock error, the fifth column is for the Galileo clock error and the sixth column is for the GLONASS clock error.

Clustering-based Solution Separation Algorithm (CSSA)

The clustering-based solution separation algorithm is based on verifying the assumption of the faulty signals by checking the consistency of the receiver's estimated state. Outline of the algorithm can be described with Fig. 1 [9], where M is the assumed number of faulty signals. And the algorithm verifies different combinations of the presumed faulty signals to test whether the assumption is correct or not. The algorithm starts with $M = 0$ and increases M by one if the verification of all possible combinations of M faults fails, until the verification succeeds or M exceeds a threshold Thr. The verification process is illustrated in Fig. 2, where the faulty satellites obtained through the clustering process should match the assumed faults if the assumption is correct, otherwise it concludes the assumed faulty satellites are not the true ones. The parameters of the clustering process is obtained with one clean subset that has the worst geometry among all clean subsets with the assumed faulty satellites. When the verification succeeds, the confirmed M faulty signals are excluded from the final position calculation, otherwise increases M by one until it reaches the threshold. A single constellation GNSS receiver calculates its state with at least four satellites. Therefore, if we want to group some position results that are calculated with four clean satellites, we need at least five clean satellites to choose from. Thus, the maximum number of faults that can be identified by the algorithm is $Thr = N - 5$ [9], where N is the number of total available satellites.

3. ADVERSARY MODEL FOR A MULTI-CONSTELLATION GNSS RECEIVER

Equipped with multi-constellation GNSS feature, a receiver can perform acquisition and tracking processes on several GNSS constellations [12]. Instead of focusing on designing and implementing a multi-constellation GNSS receiver, this work emphasizes how to secure a receiver's state, i.e., PVT solution, with its observation measurements and navigation data. In this paper, we only consider that an adversary manipulates the pseudorange measurements, not the satellite positions, because different GNSSs are developing and implementing navigation message authentication (NMA) based signals that will prevent an attacker from forging the satellite positions through modifying the navigation messages [13], [14].

When an adversary mounts an attack process, one can choose to manipulate signals of one constellation with lower complexity and cost, or manipulate signals of several constella-

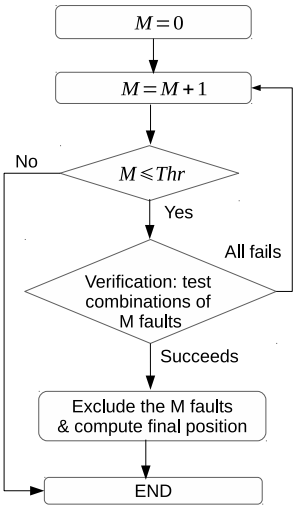

Figure 1. Outline of CSSA

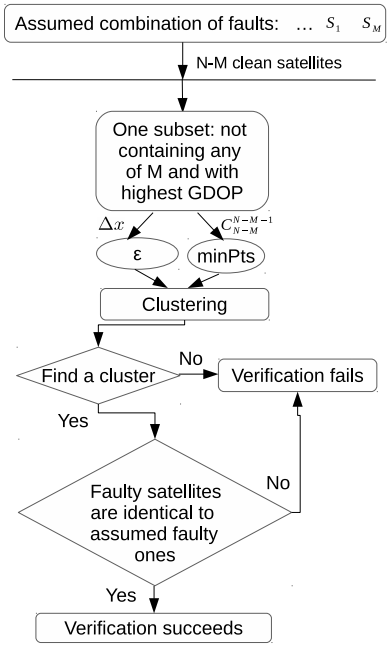

Figure 2. Verification process of CSSA

tions, which obviously requires more efforts and higher cost; but obviously can cause more harm to the victim from the adversary's point of view.

When applying CSSA to a single constellation GNSS receiver, it clusters the subsets by checking the euclidean distance between different subsets. As described in [8], [9], without being attacked, the receiver position error follows the distribution:

$$\boldsymbol{\Delta x} \sim \boldsymbol{N(0, \sigma_G)} \tag{4}$$

and the variance σ_G is

$$\sigma_G = \sqrt{Q_{11} + Q_{22} + Q_{33} + Q_{44}}\sigma_0 \tag{5}$$

where σ_0 is the standard deviation of the user equivalent range errors (UERE) [15] and $\boldsymbol{Q} = \boldsymbol{(H^T W^{-1} H)^{-1}}$, where $\boldsymbol{W}$ is the weight matrix. In Eq. 5, the first part of the right side is the position dilution of precision (GDOP), e.g., $GDOP = \sqrt{Q_{11} + Q_{22} + Q_{33} + Q_{44}}$.

By introducing a fault vector $\boldsymbol{f}$ to the receiver, with nonzero elements corresponding to different pseudorange measurements, the receiver state error follows a new distribution [8], [9]:

$$\boldsymbol{\grave{\Delta} x} \sim \boldsymbol{N((H^T W^{-1} H)^{-1} H^T W^{-1} f, \sigma_G)} \tag{6}$$

In order to analyze the euclidean distance between different type of subsets, we define X: clean subset position sets, and Y: faulty subset position sets, hence the distance can be classified to three types:

- d_{xx} is the distance between clean subsets: $d_{xx} = |x_i - x_j|$, $x_{i,j} \subset X$
- d_{yy} is the distance between faulty subsets: $d_{yy} = |y_i - y_j|$, $y_{i,j} \subset Y$
- d_{xy} is the distance between clean subsets and faulty subsets: $d_{xy} = |x_i - y_j|, \;\; x_i \subset X, \;\; y_j \subset Y$

At any given time, the position of the subset k with or without faults, follows one of these distribution, respectively:

$$\begin{aligned} &\boldsymbol{X} \sim \boldsymbol{N(P_0, \sigma_{G,k})} \\ &\boldsymbol{Y} \sim \boldsymbol{N(P_0 + (H_k^T W_k^{-1} H_k)^{-1} H_k^T W_k^{-1} f, \sigma_{G,k})} \end{aligned} \tag{7}$$

where P_0 is the actual receiver position and $\sigma_{G,k}$ is obtained based on subset k. Therefore, we can calculate the distribution of three categorized distances:

$$\begin{aligned} d_{xx} &= |x_i - x_j| \sim N(0, \sqrt{\sigma_{G,i}^2 + \sigma_{G,j}^2}) \\ d_{yy} &= |y_i - y_j| \sim N\Big((H_i^T W_i^{-1} H_i)^{-1} H_i^T W_i^{-1} f_i \\ &\quad - (H_j^T W_j^{-1} H_j)^{-1} H_j^T W_j^{-1} f_j, \sqrt{\sigma_{G,i}^2 + \sigma_{G,j}^2}\Big) \\ d_{xy} &= |x_i - y_j| \sim N\Big((H_j^T W_j^{-1} H_j)^{-1} H_j^T W_j^{-1} f, \\ &\quad \sqrt{\sigma_{G,i}^2 + \sigma_{G,j}^2}\Big) \end{aligned} \tag{8}$$

Eq. 8 is used to evaluate the performance of CSSA for a single constellation receiver because the algorithm clusters the subset positions based on their distances.

We introduce a new distance term, high dimensional distance (hd), for the clustering process when applying CSSA to detect and identify faulty satellites for multi-constellation satellite systems. The high dimensional distance describes the difference between two receiver states $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$:

$$\begin{aligned} hd = &\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \\ &+ c * |\Delta t_i^G - \Delta t_j^G| + c * |\Delta t_i^E - \Delta t_j^E| + \dots \end{aligned} \tag{9}$$

where (x_i, y_i, z_i) is the three dimensional position for the i^{th} state $\boldsymbol{x_i}$, Δt^G and Δt^E are the clock errors of the GPS and the Galileo, and more clock error elements can be added for more available constellations.

Accordingly we introduce a new term called 'state error' (SE) to describe the error between an estimated receiver state and the actual receiver state. The state error, SE, is written as:

$$SE = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} + |c\Delta t_1| + ... + |c\Delta t_k| \tag{10}$$

where $(\Delta x,\ \Delta y,\ \Delta z)$ are the three-dimensional position errors, and $|c\Delta t_k|$ is the error along the time dimension of the k^{th} constellation. The clock error of different constellation is independent from each other, therefore we can retrieve the distribution of the state error by considering its different components similarly as Eq. 4:

$$\boldsymbol{SE} \sim \boldsymbol{N}(\boldsymbol{0}, \boldsymbol{\sigma_{SE}}) \tag{11}$$

where

$$\sigma_{SE} = \sqrt{PDOP^2 + TDOP_1^2 + ... + TDOP_k^2}\sigma_0 \tag{12}$$

where $PDOP$ is the position dilution of precision, i.e., $PDOP = \sqrt{Q_{11} + Q_{22} + Q_{33}}$, and $TDOP_k$ is time dilution of precision (TDOP) of the k^{th} constellation.

Accordingly, the distribution of $\boldsymbol{SE}$ changes after introducing faults, $\boldsymbol{f}$, to the systems; we calculate the new state error with the same method as Eq. 6:

$$\grave{\boldsymbol{SE}} \sim \boldsymbol{N}((\boldsymbol{H^T W^{-1} H})^{-1}\boldsymbol{H^T W^{-1} f}, \boldsymbol{\sigma_{SE}}) \tag{13}$$

where $\boldsymbol{H}$ and $\boldsymbol{f}$ are matrix and vector corresponding to a multi-constellation receiver.

Similar to Eq. 8, the high dimensional distance, hd, between different receiver states can be written as:

$$\begin{aligned}
&hd_{x_i x_j} \sim N(0, \sqrt{\sigma_{SE,i}^2 + \sigma_{SE,j}^2}) \\
&hd_{y_i y_j} \sim N\Big((H_i^T W_i^{-1} H_i)^{-1} H_i^T W_i^{-1} f_i \\
&\quad - (H_j^T W_j^{-1} H_j)^{-1} H_j^T W_j^{-1} f_j, \sqrt{\sigma_{SE,i}^2 + \sigma_{SE,j}^2}\Big) \\
&hd_{x_i y_j} \sim N\Big((H_j^T W_j^{-1} H_j)^{-1} H_j^T W_j^{-1} f, \\
&\quad \sqrt{\sigma_{SE,i}^2 + \sigma_{SE,j}^2}\Big)
\end{aligned} \tag{14}$$

where $x_i, x_j \subset \{\text{clean states}\}$ and $y_i, y_j \subset \{\text{faulty states}\}$.

4. CSSA IN MULTI-CONSTELLATION GNSSS

For a multi-constellation GNSS receiver, an attacker has different attacking strategies or options due to one's technique and budget; in other words, the attacker can choose to manipulate signals of a single constellation or signals of several constellations. Therefore, the situation can be classified into three paths, as illustrated in Fig. 3, in which the receiver has already obtained the observation measurements and navigation data of different constellations. As Fig. 3 describes, there are three possibilities for the receiver: 1) the environment is clean, i.e., no faulty signal; 2) the attacker only manipulates fewer GNSS systems than those that the victim has access to; 3) the attacker manipulates all available GNSS systems that the receiver can access. The processes of applying CSSA in a multi-constellation receiver can be described as:

• Path determination: determines which one of the three aforementioned paths in Fig. 3 is the case for the victim receiver. The receiver performs this test by verifying whether the assumption $M = 0$ is correct or not for each single constellation. If the verification succeeds for all the GNSS systems, the receiver reaches the conclusion that all the GNSS constellations are clean. Otherwise, if the verification succeeds for some constellations and fails for others, the second path is determined, the receiver adopts corresponding steps for this path. The last possibility is that the verifications fails for each single constellation, then the path ③ is determined.

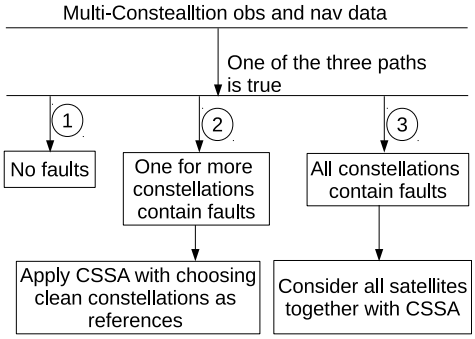

Figure 3. Outline of CSSA with multi-constellation GNSS

• If the path ① is deemed to be the case, the receiver concludes that the environment is clean and calculates its state with all available satellite signals.

• If the path ② is decided, i.e., both clean constellations and faulty constellations exist, the goal is to identify and exclude the faulty signals from the faulty constellations. For illustration, lets define that C_1 is the clean constellation and C_2 is the faulty one; then the receiver tests and identifies the faulty signals of C_2 by considering C_1 as reference.

Further steps can be taken as follows: the receiver state calculated by the reference constellation C_1 is considered as the real state. Thereafter, the receiver takes one satellite signal from constellation C_2 and uses it together with all signals of constellation C_1 to check whether all the signals are clean, as explained in Algorithm. 1. Because if the tested signal is clean, the high dimensional distance should follow a distribution as $hd_{x_{C_1}\hat{x}}$ of Eq. 14, where x_{C_1} is the receiver's state estimated with the clean constellation C_1, and $\hat{x}$ is the estimated receiver state using constellation C_1 together with the tested signal. With a false alarm probability requirement P_{FA}, it should satisfy: $1/2(1 - P\{|hd_{xx}| \leq hd_{x_{C_1}x_{tested}}\}) \leq P_{FA}$. However, if the tested signal is faulty, the estimated receiver's state follows the distribution as hd_{xy} of Eq. 14, then the above requirement satisfaction will not be fulfilled.

Once the receiver considers the tested signal as a clean signal, the receiver uses it to test the rest signals of C_2 together with other clean signals. Otherwise, the receiver discards the tested signal for the rest of processing steps.

• If the path ③ is determined, each constellation contains one or several faulty signals, the goal is to identify all the faulty signals from all the constellations.

First, the receiver applies CSSA to each constellation to check whether the number of faults for each constellation excesses the threshold $Threshold$ of the single constellation. If the number of faults of one or more constellations is below the threshold, CSSA can identify and exclude the faulty signals and then consider this constellation as a clean constellation. Then it uses the newly constructed clean constellation as the reference to identify the faulty signals of other constellations, like path ②.

Second, when the number of faults exceeds the threshold for all the constellations after applying CSSA to each single constellation, for instance, $M_1 > Threshold_1$ for C_1 and

ALGORITHM 1. Process illustration with two constellations after path ② is determined

Input: Clean constellation C_1 and faulty constellation C_2.
Output: The faulty signals of C_2
1: ***Initialization***: a receiver state $\boldsymbol{x}_{\boldsymbol{C_1}}$ calculated with all satellites of C_1; satellites $\{l_1 ... l_{N_2}\}$ in C_2.
2: *LOOP Process*
3: **while** take one satellite l_k from C_2 **do**
4: **for** each combination of $N_1 - 1$ satellites from C_1 **do**
5: calculate a receiver state based on $N_1 - 1$ satellites from C_1 together with l_k
6: **end for**{there are N_1 states in total}
7: calculate a new receiver state $\hat{\boldsymbol{x}}$ using all satellites of C_1 plus l_k
8: **if** $hd_{x\hat{x}}$ satisfies the first sub-equation of Eq. 14 under a requirement of flase alarm probability P_{FA} **then**
9: the tested signal l_k is clean
10: **else**
11: l_k is a faulty signal
12: **end if**
13: test another satellite of C_2
14: **end while**

ALGORITHM 2. Process illustration with two constellations after path ③ is determined

Input: Both faulty constellations C_1 and C_2
Output: The faulty signals of C_1 and C_2
1: ***Initialisation***: possible value sets of $\{(M_1, M_2)\}$ based on Eq. 15
2: **for** $(M_1, M_2) \in \{(M_1, M_2)\}$ **do**
3: clean satellites: $N_1 - M_1$ in C_1 and $N_2 - M_2$ in C_2;
4: calculate receiver states with $N_1 + N_2 - M_1 - M_2 - 1$ satellites that are taken from two constellations;
5: cluster the receiver states based on the high dimensional distance, hd;
6: **if** find clusters with size of $N_1 + N_2 - M_1 - M_2$ **then**
7: find the faulty satellites based on the found clusters;
8: **if** size(found faulty satellites) match the assumed (M_1, M_2) **then**
9: the assumed (M_1, M_2) is correct;
10: calculate the receiver final state by excluding the faults;
11: **end if**
12: **else**
13: try next value set (M_1, M_2);
14: **end if**
15: **end for**

$M_2 > Threhold_2$ for C_2, we then identify the faulty signals by considering all the constellations together. Therefore, the number of faulty signals M_1 and M_2 should satisfy the following restrictions:

$$\begin{cases} N_1 > M_1 > N_1 - 5 \\ N_2 > M_2 > N_2 - 5 \\ M_1 + M_2 \leq N_1 + N_2 - 6 \end{cases} \tag{15}$$

where N_1 and N_2 are total number of available satellites for C_1 and C_2, $N_1 - 5$ and $N_2 - 5$ are the thresholds of the corresponding single constellation, and $M_i < N_i$ due to requiring at least one satellite for the clock error. The last restriction is that the total number of faults should not be greater than the total number of satellites minus six (for two constellations), because it requires at least five satellites to calculate a receiver state for two constellation systems and it requires one more satellite for the clustering purpose. Thereafter, the receiver can iterate all the combinations of number sets (M_1, M_2), and verify the assumptions with a clustering process, as shown in Algorithm. 2. As the algorithm illustrates, with the assumption of number of faults in each constellation: (M_1, M_2), the receiver calculates its PVT state with $N_1 + N_2 - M_1 - M_2 - 1$ satellites, among of which there are $C_{N_1+N_2-M_1-M_2}^{N_1+N_2-M_1-M_2-1} = N_1 + N_2 - M_1 - M_2$ receiver states that should form a cluster and the distance used for the clustering is the high dimension distance, hd, in Eq. 9. It is possible that the receiver finds more than one cluster, then the receiver discards the clusters that produce different number of faults as the number set (M_1, M_2). If there is more than one cluster that produce the matched number of faults, as assumed, but with different satellites combination, the only method to check which one is correct is to verify whether the environment is clean by excluding the faulty satellites produced by each cluster.
The receiver continues testing each possible number set of (M_1, M_2) until it finds the deemed faulty satellites in each constellation, or it concludes that the number of faulty satellites exceeds the threshold $N_1 + N_2 - 6$, as the last restriction of Eq. 15.

5. EVALUATION RESULTS

We use Eq. 14 as the basis to evaluate the performance theoretically, with the help of RINEX data files, which are downloaded from the NASA's space geodesy data center [16]; the recored data has one hour length, with 30 seconds interval from one static station. In the first epoch, there are 10 GPS satellites that we call constellation 1, i.e., C_1, and 8 Galileo satellites that we call constellation 2, i.e., C_2. As discussed in Section 4, there are three possibilities for the receiver situation, as shown in Fig. 3. The path determination is to apply CSSA to a single GNSS constellation, which article [9] has analyzed in detail. The following evaluation is divided based on path ② and path ③.

Simulation Setup: The attacker manipulates k_1 satellite signals of C_1 and k_2 signals of C_2, and the attacker introduces errors to the pseudorange measurements with numeric values: $f_v \in \{50,\ 100,\ 200,\ 300,\ 500\}$m. For the evaluation of path ②, we let constellation C_1 be clean, then the receiver tries to detect and identify the faulty signals of constellation C_2 with the algorithm, and we evaluate the receiver's performance, given requirements of detection probability and false alarm probability. For path ③, we do the same performance evaluation on the receiver, based on an example of number of faulty signals (M_1, M_2).

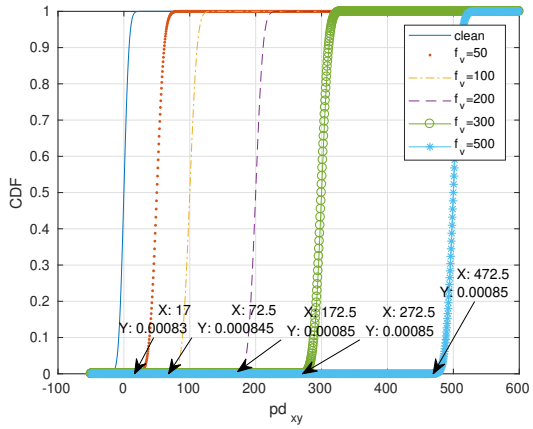

Figure 4. An example illustration: CDF of the states distance pd_{xy} for different pseudorange measurement errors for path ②

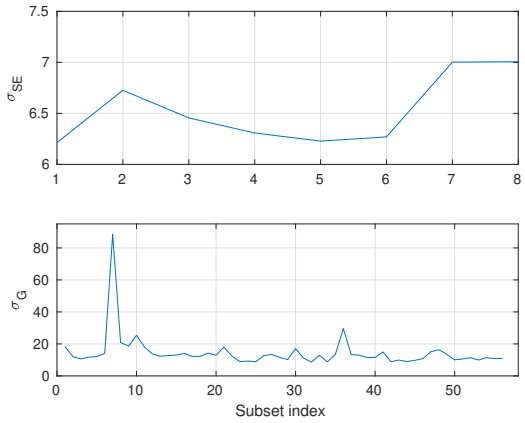

Figure 5. An example illustration: state variance for different cases: 1) top plot is the variance of constellation C_1 together with each one tested signal from C_2 when using CSSA with multi-constellation; 2) bottom plot is the variance of each subset at the iteration of $M = 2$ using CSSA with a single constellation.

Path ②

For the simulation of path ②, the attacker decides to attack the Galileo signals and ignore the GPS signals, because of some unknown reasons. And the attacker manipulates the number of satellites in numerical values: $\{2\ 4\}$. Because the threshold of the number of faults that can be detected for a single constellation is $Threshold_2 \leq N_2 - 5 = 3$, so the two faulty signals can be identified with CSSA for constellation C_2. In fact, in this case, the receiver can choose to use CSSA within a single constellation or with multi-constellation to identify the small number of faults. The important truth for this case is that the receiver state estimated by the clean constellation C_1 can be considered as the true receiver state, which can be used to evaluate other signals.

In the simulation, the attacker chooses four satellites to spoof by modifying the corresponding pseudorange measurements with the aforementioned numeric values. Given a required detection probability $P_{det}^{req} = 99.9\%$, meaning the probability of detecting the faulty signals is at least 99.9%, we obtain different values, v_0, that satisfy: $P\{pd_{xy} > v_0\} \geq 99.9\%$ based on CDF of pd_{xy}, for different pseudorange measurements error, f_v. As shown in Fig. 4, we know that $v_0 = 17$ for the case $f_v = 50$ m, similarly $v_0 = 72.5$ for the case $f_v = 100$ m, $v_0 = 172.5$ for the case $f_v = 200$ m, $v_0 = 272.5$ for the case $f_v = 300$ m and $v_0 = 472.5$ for the case $f_v = 500$ m. In the figure, the 'clean' line is calculated based on constellation C_1 that is known to the receiver after the path determination. The values, pd_{xy}^0, indicate that when the attacker modifies the pseudorange measurements with the corresponding f_v, the tested signal has 99.9% probability of being faulty if $hd_{x_{C_1}x_{tested}}$ is greater than the corresponding pd_{xy}^0.

However, as a receiver, the value f_v is unknown previously, therefore the receiver needs to assume f_v is small to obtain the hd_{xy}^0, in order to detect the small errors. In contrast, the hd_{xy}^0 cannot be too small, otherwise, the tested signal will be identified as a faulty when it is clean. When we look at the line with 'clean' legend of Fig. 4, given a false alarm requirement $P_{FA} = 0.01\%$, we have $P\{|pd_{xy}| \leq 23.8\} = 99.99\%$. That means that if we consider the tested signal with $hd_{x_{C_1}x_{tested}} > 23.8$ as a faulty signal, we have 0.01% false alarm probability. Therefore, in order to satisfy the P_{FA}, we have $P\{pd_{xy}^0 > 23.8\} = 99.29\%$ for $f_v = 50$ and $P\{pd_{xy}^0 > 23.8\} = 1.665e-14$ for $f_v = 100$. Then we can conclude that when the attacker modifies pseudorange measurements less than 50 m ($f_v = 70$ m for $P_{det}^{req} = 99.9\%$ in simulation), the algorithm cannot satisfy both $P_{det}^{req} = 99.9\%$ and $P_{FA} = 0.01\%$ for path 2.

As discussed before, when the number of faults is small, in order to identify the faulty satellites the receiver has an option to choose from using CSSA with a single constellation or with multi-constellation. However, the performance is different, because multi-constellation has more available satellites that provides better geometry condition than a single constellation, as shown in Fig. 5. σ_G and σ_{SE} are calculated based on Eq. 5 and Eq. 12. After abandoning the unacceptable subset that has $GDOP > 10$, we use the subset with the largest σ_G to calculate the value in order to satisfy the requirement of false alarm probability, we get $P\{|d| \leq 115\} = 99.99\%$ where d is the distance between the subset with the largest σ_G and the true position. It means that CSSA with single constellation, in this setup for two faults, can provide a false alarm probability $P_{FA} = 0.01\%$ only when the receiver is shifted farther than 115 m by the attacker. In other words, it gives higher false alarm probability when the attacker induces smaller position shift. Comparing with above CSSA in multi-constellation, CSSA with single constellation is less sensitive to the position error and provides less accuracy.

Path ③

When all of constellations contain faulty signals, the algorithm tries to determine whether it can identify the faulty signals for each constellation with CSSA. If the number of faults in some constellations satisfies $M \leq N - 5$, CSSA can identify the faulty signals in each of these constellation. Thereafter, these constellations become clean after excluding the faulty signals, thus the situation is switched to path ②.

However, when the attacker spoofs more signals, e.g., 6 satellites of C_1 and 4 of C_2, the two constellations need to collaborate in order to identify the faults because the number of faults excesses the threshold for both single constellation.

As Eq. 15 describes, in this setup we have:

$$\begin{cases} 10 = N_1 > M_1 > N_1 - 5 = 5 \\ 8 = N_2 > M_2 > N_2 - 5 = 3 \\ M_1 + M_2 \leq N_1 + N_2 - 6 = 12 \end{cases} \tag{16}$$

which gives us the following number sets that (M_1, M_2) can be:

$$(M_1, M_2) \subset \{(6,4),\ (6,5),\ (6,6),\ (7,4),\ (7,5),\ (8,4)\} \tag{17}$$

Therefore, CSSA needs to check each above possible set to see which one is correct, thus to identify and exclude the faulty signals. The receiver starts assuming that the number of faults for two constellations is $(6, 4)$, which means that there are four clean satellites in C_1 and four clean satellites in C_2, i.e., eight clean satellites in total. Therefore, if the assumption is correct, eight clean subsets should be clustered together while each subset contains seven clean satellites out of the eight ones. Since we are going to calculate the receiver's states with seven satellites, there will be C_{18}^7 subsets in total, but the following subsets need to be excluded: 1). the subsets only contain satellites from one constellation, since we calculate the receiver's state with two constellations; 2). the subsets have GDOP greater than six, which are considered as bad geometry in this paper. However, this threshold for GDOP can be configured for different applications and can be adjusted for different subset size at different number sets (M_1, M_2).

For all the subsets, we calculate the high dimension distance based on the two subsets with the highest state variance σ_{SE}, e.g., hd_{xx} in Eq. 14. And obtain one value based on requirement of the false alarm probability: $P\{|hd_{x_i x_j}| \geq hd_{xx}^0 = 69.79\} < P_{FA}$, where $P_{FA} = 0.1\%$, and 67.79 is the value used as the cluster radius for the clustering process. Thereafter, we can calculate the probabilities of that the distance between clean states and faulty states, e.g., $hd_{x_i y_j}$, is greater than hd_{xx}^0, for different modified pseudorange measurements, as shown in Fig. 6. In the figure, $X = 70$ is the value that satisfies the false alarm probability requirement. From the figure, we know that the algorithm can not identify the faultys signals when the attacker induces 100 m error to the pseudorange measurements; it can identify the faulty signals with probability of $1 - 0.1306 = 86.94\%$ when $f_v = 200$ (99.7% for $f_v = 230$ in simulation); the probability is almost 100% when the induced error is more than 300 m. The results of Fig. 6 show that the algorithm works efficiently when the induced pseudorange measurements error is more than 200 m, otherwise the detection probability is low while satisfying the false alarm probability $P_{FA} = 0.1\%$.

During the clustering step, the number of found clusters may be more than one, so the receiver first calculates the faulty satellites for each found cluster, and then takes further steps as: 1). the clusters gives different number of faulty satellites as the above assumed faulty signals are abandoned; 2). the left cluster with matched number of faulty satellites is considered as the correct one; however, if there are still more than one cluster left, the way to decide which one is correct is to check whether the environment is clean after excluding the concluded faulty satellites of the clusters from two constellations. The reason why there are more than one cluster found is because of $hd_{y_i y_j}$, as expressed in Eq. 14. When the attacker modifies the pseudorange measurements with a large value, the high dimension distance, $hd_{x_i y_j}$, between a clean state and a faulty state is much larger than the distance between clean states, $hd_{x_i x_j}$. That means that it

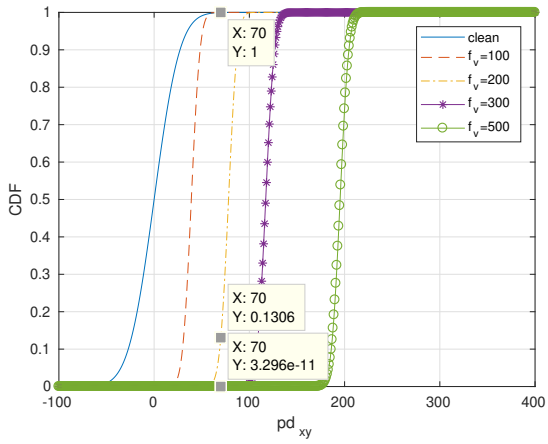

Figure 6. An example illustration: CDF of the states distance pd_{xy} for different modifications of pseudorange measurements for path ③

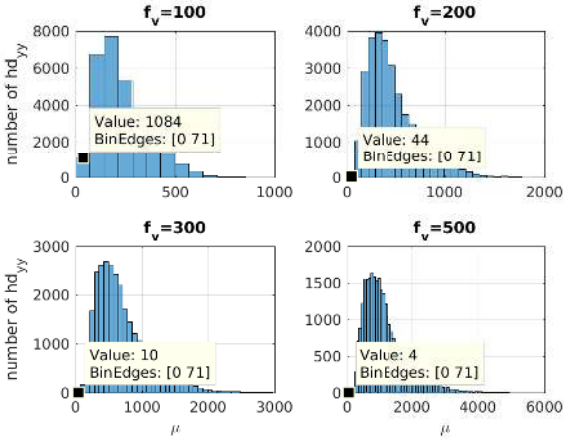

Figure 7. An example illustration: distribution of the mean value of hd_{yy} in Eq. 14 for different pseudorange measurement errors

has low probability of mixing the clean subsets with the faulty subsets during the clustering process.

However, the probability of inner-distance among the faulty subsets are comparable with inner-distance among the clean subsets is higher when the attacker introduces smaller errors to the pseudorange measurements, as shown in Fig. 7. The figure illustrates the distribution of the mean value, i.e., $(H_i^T W_i^{-1} H_i)^{-1} H_i^T W_i^{-1} f_i - (H_j^T W_j^{-1} H_j)^{-1} H_j^T W_j^{-1} f_j$, based on different pseudorange measurement errors, f_v.

In the figure, the bin size, i.e., 70, is the value obtained priorly based on the false alarm probability requirement and used as radius for the clustering process. And we calculate the mean value, μ, using the same manner as the state error, i.e., Eq. 10. The top-left plot shows there are many inner-distance, i.e., 1084, whose mean value is smaller than 70. We conclude that it is impossible to distinguish the high dimension distances among the clean subsets from those among the faulty subsets when the attacker introduces 100 m error to the pseudorange measurements of constellation C_1 and C_2 with $M_1 = 6$ and $M_2 = 4$. The top-right plot reveals that the inner-distances among the faulty subsets still have chance to mislead the

receiver from confusing them with the inner-distance among the clean subsets, but with low probability, when the induced measurement error is 200 m. The bottom two plots show that it will be easy for the receiver to tell the difference between the inner-dintance among the faulty subsets and that among the clean subsets, because there are very small number of high dimension distance that are close to each other for the faulty subsets. Therefore, when there are several found clusters, the best way to decide which one is correct is to check whether there still has faulty signals left, i.e., clean environment, after excluding the faulty signals produced by each cluster.

6. CONCLUSIONS

We evaluated the receiver's performance when the adversary manipulated fewer constellations than that the victim receiver has access to, given requirements of detection probability and false alarm probability. Moreover, we did the same performance evaluation when the adversary manipulated all available constellations to the receiver, and we further quantitatively analyzed the reasons why the inner-distance among faulty satellite subsets degrades the performance.

Particularly, when an attacker cannot manipulate all the available constellations, CSSA is able to detect and identify the faulty signals even when the attacker introduces small errors to the pseudorange measurements. For example, in our simulation setup, it can reach 99.9% detection probability and 0.01% false alarm probability for 70 m induced error. When the attacker has large power to modify most of signals of each constellation, CSSA can still identify the faulty signals of all constellations with a constraint: $Threshold \leq N_1 + ... + N_k - 6$, by clustering different satellite subsets, with several constellations collaboratively. The results show that CSSA provides solid detection probability with multi-constellation to identify a large number of faulty signals that CSSA is not able to do with a single constellation. In other words, with more available signals from several constellations, CSSA provides better performance comparing to a single constellation, in term of capacity, detection probability and false alarm probability, as well as accuracy.

7. ACKNOWLEDGMENT

This work has been partially supported by the Swedish Foundation for Strategic Research (SSF) SURPRISE project.

REFERENCES

[1] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007, pp. 397–405.

[2] L. Wang, P. D. Groves, and M. K. Ziebart, "Multi-constellation gnss performance evaluation for urban canyons using large virtual reality city models," *The Journal of Navigation*, vol. 65, no. 3, pp. 459–476, 2012.

[3] O. Montenbruck, P. Steigenberger, R. Khachikyan, G. Weber, R. Langley, L. Mervart, and U. Hugentobler, "Igs-mgex: preparing the ground for multi-constellation gnss science," *Inside Gnss*, vol. 9, no. 1, pp. 42–49, 2014.

[4] P. Papadimitratos and A. Jovanovic, "GNSS-based Positioning: Attacks and Countermeasures," *IEEE MILCOM*, San Diego, CA, 2008.

[5] M. L. Psiaki and T. E. Humphreys, "GNSS Spoofing and Detection," *Proceedings of the IEEE*, vol. 104, no. 6, pp. 1258–1270, 2016.

[6] K. Zhang and P. Papadimitratos, "GNSS receiver tracking performance analysis under distance-decreasing attacks," in *2015 International Conference on Location and GNSS (ICL-GNSS)*. IEEE, 2015, pp. 1–6.

[7] ——, "On the effect of the Distance-decreasing Attacks on Cryptographically Protected GNSS Signals," in *Proceedings of the 2019 International Technical Meeting of The Institute of Navigation*, January 2019, Accepted.

[8] K. Zhang, R. A. Tuhin, and P. Papadimitratos, "Detection and Exclusion RAIM Algorithm against Spoofing/Replaying Attacks," *International Symposium on GNSS 2015*, Kyoto, Japan, 2015.

[9] K. Zhang and P. Papadimitratos, "Clustering-based Solution Separation Algorithm against Spoofing/Replay Attacks," *GPS Solutions*, 2018, Under review.

[10] R. G. Brown, "A baseline GPS RAIM scheme and a note on the equivalence of three RAIM methods," *Navigation*, vol. 39, no. 3, pp. 301–316, 1992.

[11] M. Joerger, F.-C. Chan, S. Langel, and B. Pervan, "RAIM detector and estimator design to minimize the integrity risk," in *Proceedings of the 25th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2012). Nashville, TN*, 2012.

[12] S. Söderholm, M. Z. H. Bhuiyan, S. Thombre, L. Ruotsalainen, and H. Kuusniemi, "A multi-GNSS software-defined receiver: design, implementation, and performance benefits," *Annals of Telecommunications*, vol. 71, no. 7-8, pp. 399–410, 2016.

[13] K. Wesson, M. Rothlisberger, and T. Humphreys, "Practical cryptographic civil GPS signal authentication," *Navigation*, vol. 59, no. 3, pp. 177–193, 2012.

[14] I. Fernández-Hernández, V. Rijmen, G. Seco-Granados, J. Simon, I. Rodríguez, and J. D. Calle, "A navigation message authentication proposal for the galileo open service," *Navigation: Journal of the Institute of Navigation*, vol. 63, no. 1, pp. 85–102, 2016.

[15] A. Cina and M. Piras, "Stand-Alone Satellite-Based Global Positioning," *Orbit*, vol. 2, no. b2, p. a2, 2013.

[16] Crustal Dynamics Data Information System (CDDIS DAAC), "International GNSS Service, Hourly 30-second observation data." Accessed 2018-08-27. [Online]. Available: ftp://cddis.gsfc.nasa.gov/gnss/data/hourly/2018/180/11/

BIOGRAPHY

Kewei Zhang *is a Ph.D. candidate with the Networked Systems Security (NSS) group at KTH Royal Institute of Technology, Stockholm, Sweden. He earned his MSc degree in Wireless Systems at KTH, Sweden. His research is concerned with secure localization and positioning.*

Panos Papadimitratos *is a professor with the School of Electrical Engineering and Computer Science (EECS) at KTH Royal Institute of Technology, Stockholm, Sweden, where he leads the Networked Systems Security (NSS) group. He earned his Ph.D. degree from Cornell University, Ithaca, New York, in 2005. His research agenda includes a gamut of security and privacy problems, with emphasis on wireless networks. His webpage is* `www.people.kth.se/~papadim`.