2017

Using Range Information to Detect Spoofing in Platoons of Vehicles

Peter F. Swaszek
University of Rhode Island, swaszek@uri.edu

Richard J. Hartnett

See next page for additional authors

Citation/Publisher Attribution
Swaszek, Peter F., Hartnett, Richard J., Seals, Kelly C., "Using Range Information to Detect Spoofing in Platoons of Vehicles," *Proceedings of the 30th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2017)*, Portland, Oregon, September 2017, pp. 2838-2853.
Available at: https://www.ion.org/publications/abstract.cfm?articleID=15338

Using Range Information to Detect Spoofing in Platoons of Vehicles

Peter F. Swaszek, *University of Rhode Island*
Richard J. Hartnett, *U.S. Coast Guard Academy*
Kelly C. Seals, *U.S. Coast Guard Academy*

BIOGRAPHIES

Peter F. Swaszek is a Professor in the Department of Electrical, Computer, and Biomedical Engineering at the University of Rhode Island. His research interests are in statistical signal processing with a focus on digital communications and electronic navigation systems.

Richard J. Hartnett is a Professor of Electrical Engineering at the U.S. Coast Guard Academy, having retired from the USCG as a Captain in 2009. His research interests include efficient digital filtering methods, improved receiver signal processing techniques for electronic navigation systems, and autonomous vehicle design.

Kelly C. Seals is the Chair of the Electrical Engineering program at the U.S. Coast Guard Academy in New London, Connecticut. He is a Commander on active duty in the U.S. Coast Guard and received a PhD in Electrical and Computer Engineering from Worcester Polytechnic Institute.

ABSTRACT

GNSS are well known to be accurate providers of position information across the globe. Because of high signal availabilities, capable/robust receivers, and well-populated satellite constellations, operators typically believe that the location information provided by their GNSS receiver is correct. More sophisticated users are concerned with the integrity of the derived location information.

Attacks on GNSS availability and integrity are known as jamming and spoofing. Jamming involves the transmission of signals that interfere with GNSS reception so that the receiver is unable to provide a position or time solution; various methods to detect jamming, and possibly overcome it, have been considered in the literature. Spoofing is the transmission of counterfeit GNSS signals so as to mislead a GNSS receiver into reporting an inaccurate position or time. If undetected, spoofing might be much more dangerous than a jamming attack.

A variety of approaches have been proposed in the literature to recognize spoofing. Of interest here are methods which compare GNSS information to measurements available from other, non-GNSS sensors. Recent ION conferences have included several examinations of combining GNSS and non-GNSS data toward spoof detection.

This paper considers the use of range-only information to detect GNSS spoofing of a platoon of vehicles equipped with inter-vehicle communications: a statistical model of the problem is developed in which the spoofer is assumed to have limited geographical impact (i.e. only spoofs a subset, nominally one, of the vehicles in the platoon); under a Neyman-Pearson formulation the (generalized) likelihood ratio test to fuse the GNSS and range measurements is presented; examples are included to demonstrate the resulting performance.

INTRODUCTION

GNSS are well known to be accurate providers of position information across the globe. Because of high signal availabilities, capable/robust receivers, and well-populated satellite constellations, operators typically believe that the location information provided by their GNSS receiver is correct. More sophisticated users are concerned with the integrity of the derived location information; for example, RAIM algorithms were developed to address possible satellite failure modes.

Attacks on GNSS availability and integrity are known as jamming and spoofing. Both are based on the creation of radio signals in the GNSS band. Jamming involves the transmission of signals that interfere with GNSS reception so

that the receiver is unable to provide a position or time solution. Various methods to detect jamming, and possibly overcome it, have been considered in the literature. Spoofing is the transmission of counterfeit GNSS signals so as to mislead a GNSS receiver into reporting an inaccurate position or time. If undetected, spoofing might be much more dangerous than a jamming attack.

A variety of approaches have been proposed in the literature to recognize spoofing and can vary widely based upon the assumed capabilities and a priori knowledge of the spoofer. Many of these are based on the RF signal alone as, in some sense, they are the simplest to implement. Of interest here are methods which compare GNSS information to measurements available from other, non-GNSS sensors. Over 10 years ago Warner and Johnston [1] suggested such methods, calling them *sanity checks*; they did not further develop the idea. Recently there have been a few examinations of using different non-GNSS signals:

- In 2014 these authors considered the use of IMU data to detect spoofing of a Coast Guard ship [2]. Specifically, the pitch and roll measurements from the ship's gyrocompass were used to predict the relative spatial trajectory of a GPS antenna mounted high up on the ship. This movement was then correlated to the GPS measurements (with the linear motion of the ship being removed) to detect spoofing. The concept was that the spoofer would not correctly generate the "wiggle" due to the sea state and, hence, be identifiable.
- In 2014 and 2015 Khanafseh and Pervan employed RAIM residuals from a tightly coupled aircraft GPS/INS to detect spoofing [3, 4]. In this case, the tightly coupled INS and GPS system tracked the aircraft's motion due to winds. As above, if the spoofer does not generate this "wiggle" correctly, it could be detected.
- In 2015 Carson and Bevly discussed the use of range and bearing information with GPS positions to detect spoofing for a platoon of vehicles [5]. They assumed the availability of Relative Position Vectors (RPVs) between pairs of vehicles from the radar sensor. To detect spoofing of a single vehicle they compared these RPVs to the corresponding GPS difference vector, declaring spoofing if the difference was too great. Their focus was on a pair of vehicles only.
- In 2016 these authors presented methods to detect GNSS spoofing for a single vehicle with multiple ranges or pseudoranges from fixed beacons [6, 7]. These works included full descriptions of the statistical hypothesis tests (Neyman-Pearson criterion) with details on performance analysis and Monte Carlo examples.

This paper considers the use of range only (no bearing) information communicated amongst a platoon of vehicles to detect GNSS spoofing. Our initial interest in this problem formulation developed from a question on the spoofing resilience of a senior capstone project involving delta-wing UAVs, each equipped with both a GPS receiver and a wideband ranging sensor. The primary contribution of this paper is the explicit development of a GNSS spoof detection algorithm that fuses multiple GNSS positions with such range measurements. Examples with 3 and more vehicles are included to demonstrate its utility.

THE MEASUREMENTS

Figure 1 presents the problem scenario:

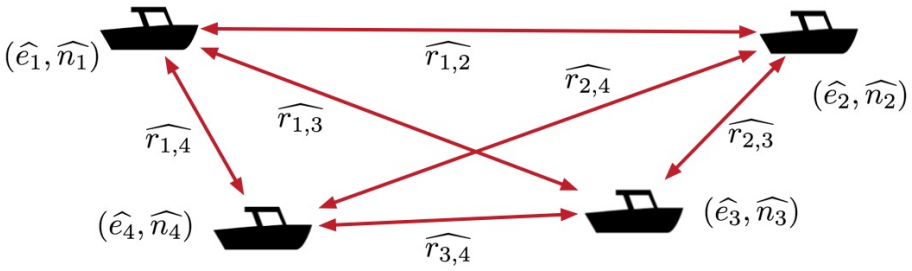

Figure 1: Example of a four vehicle platoon.

- We envision a platoon of m vehicles distributed over a plane (in the figure $m = 4$, individually numbered); we will naturally call the vehicles' coordinates east and north with notation (e_k, n_k), $k = 1, 2, \ldots m$. (A three-dimensional development is, of course, possible and is discussed later.)
- Each individual vehicle is equipped with a stand-alone GNSS receiver that yields a position measurement; the notation for measurements will employ hats so that $(\widehat{e_k}, \widehat{n_k})$ is the GNSS measurement at vehicle k. An obvious (and we think interesting) extension, but not considered here, is to limit the availability of GNSS to only some of the vehicles as occurs in some mesh network problems.
- Since inter-vehicle range is of interest as auxiliary information, let the variable $r_{j,k}$ represent the true range between vehicles j and k

$$r_{j,k} = \sqrt{(e_j - e_k)^2 + (n_j - n_k)^2} \tag{1}$$

  for $j, k \in \{1, 2, ...m\}$. Trivially, $r_{k,k} = 0$.
- Some (perhaps all) of the vehicles are equipped to estimate these ranges to the other vehicles. We will use the notation $\widehat{r_{j,k}}$, $j \neq k$, for those measurements. The set of measurements might not include all $\frac{m(m-1)}{2}$ possible ranges as some might be occluded by other vehicles or the environment, or the separation might be at too large for the equipment to measure the range. The development below allows for some of these measurements to be missing.
- Both sensors (GNSS and range) suffer from measurement error, perhaps to quite different degrees. Since the full platoon is expected to be geographically "close" we will assume independent (across vehicles) and identically distributed errors on the unspoofed GNSS measurements; the model for the spoofed measurements will be independent, but different (and unknown). Further, we will assume that the range measurements are independent, both amongst themselves and from the GNSS measurements, and identically distributed. We further assume that both sets of measurements are unbiased and will invoke Gaussian statistics. These assumptions are made to keep the formulation simple and are relaxed toward the end of the paper.
- All of the available measurements (both GNSS and range) are shared between the vehicles via some communications link; we ignore transmission latency and resolution issues.
- We assume that common range measurements, if available, are combined so that we have a unique set of measurements; in other words, we assume that $\widehat{r_{j,k}} = \widehat{r_{k,j}}$.

For notational simplicity we will occasionally employ boldface notation (e.g. $\mathbf{e}$ or $\widehat{\mathbf{r}}$) to represent a vector of variables or measurements.

THE HYPOTHESES

Our goal here is to test for GNSS spoofing which we define as the existence of radio signals that would result in an erroneous position solution at a GNSS receiver. For the problem formulation we initially assume that when a spoofer is present the interferring signal impacts *only one* vehicle, commenting on the extension to a subset of the vehicles in the conclusions. Thus, we have $m + 1$ situations, the null hypothesis, H_0, in which no spoofer is present, and m alternative hypotheses, $\mathrm{H}_1, \mathrm{H}_2, \ldots \mathrm{H}_m$ in which a spoofer is present and impacts that corresponding vehicle:

H_0: With no spoofer present each individual measurement is an accurate estimate of its respective variable:

$$\widehat{e_k} = e_k + w_{Ge,k} \qquad \widehat{n_k} = n_k + w_{Gn,k}$$

and

$$\widehat{r_{j,k}} = r_{j,k} + w_{R,jk}$$

in which the $w_{\cdot,\cdot}$ are measurement errors.

H_j: ($j = 1, 2, \ldots m$) The spoofer distorts the GNSS measurement only at vehicle j, the other $m - 1$ GNSS measurements and all of the range measurements are unaffected:

$$\widehat{e_k} = e_k + w_{Ge,k} \qquad \widehat{n_k} = n_k + w_{Gn,k}$$

for $k \neq j$,

$$\widehat{e_j} = e_S + w_{Se,j} \qquad \widehat{n_j} = n_S + w_{Sn,j}$$

and

$$\widehat{r_{j,k}} = r_{j,k} + w_{Rj,k}$$

in which e_S and n_S describe the (unknown) spoofed location for vehicle j.

To finish the characterization we need knowledge of the variances/covariances of the noise terms. Clearly the results will vary with the relative scales of these measurement errors. With the unbiased Gaussian assumption, we will use the notation σ_G, σ_S, and σ_R for the standard deviations of the GNSS under no spoofing, the GNSS under spoofing, and the ranging errors, respectively. More complex statistical models, including GNSS error correlations, are considered later in this paper. Finally, we recognize that the Gaussian model for errors in $\widehat{r}$ is strictly wrong; the measurement could never be negative. However, the inaccuracy of this assumption is negligible for the expected ranges and the assumption dramatically simplifies the development and analysis of the spoof detection algorithm.

HYPOTHESIS TESTING

The development of an $m+1$-ary hypothesis test starts by constructing the $m+1$ likelihood functions

$$L_j\left(data\right) = f\left(\widehat{\mathbf{e}}, \widehat{\mathbf{n}}, \widehat{\mathbf{r}} \,|\mathrm{H}_j\right)$$

Invoking the Gaussian assumption and mutual independence of the measurements yields

$$L_0\left(data\right) = \prod_{k=1}^{m} \frac{1}{2\pi\sigma_G^2} e^{-\frac{1}{2\sigma_G^2}\left[(\widehat{e_k}-e_k)^2+(\widehat{n_k}-n_k)^2\right]} \prod_{p=1}^{m-1}\prod_{q=p+1}^{m} \frac{1}{\sqrt{2\pi}\sigma_R} e^{-\frac{1}{2\sigma_R^2}(\widehat{r_{p,q}}-r_{p,q})^2}$$

and

$$\begin{aligned} L_j\left(data\right) = & \frac{1}{2\pi\sigma_S^2} e^{-\frac{1}{2\sigma_S^2}\left[(\widehat{e_j}-e_S)^2+(\widehat{n_j}-n_S)^2\right]} \\ & \times \prod_{k=1,k\neq j}^{m} \frac{1}{2\pi\sigma_G^2} e^{-\frac{1}{2\sigma_G^2}\left[(\widehat{e_k}-e_k)^2+(\widehat{n_k}-n_k)^2\right]} \prod_{p=1}^{m-1}\prod_{q=p+1}^{m} \frac{1}{\sqrt{2\pi}\sigma_R} e^{-\frac{1}{2\sigma_R^2}(\widehat{r_{p,q}}-r_{p,q})^2} \end{aligned}$$

(In both of these expressions the double product term assumes that all of the ranges are measured; it would be a simple step to reduce these as appropriate.) Removing extraneous multiplicative terms and taking logarithms, the log-likelihoods are equivalent to

$$l_0\left(data\right) = -\left[\sum_{k=1}^{m} \frac{\left(\widehat{e_k}-e_k\right)^2+\left(\widehat{n_k}-n_k\right)^2}{2\sigma_G^2} + \sum_{p=1}^{m-1}\sum_{q=p+1}^{m} \frac{\left(\widehat{r_{p,q}}-r_{p,q}\right)^2}{2\sigma_R^2} I_{p,q}\right] \tag{2}$$

and

$$l_j\left(data\right) = -\left[\frac{\left(\widehat{e_j}-e_S\right)^2+\left(\widehat{n_j}-n_S\right)^2}{2\sigma_S^2} + \sum_{k=1,k\neq j}^{m} \frac{\left(\widehat{e_k}-e_k\right)^2+\left(\widehat{n_k}-n_k\right)^2}{2\sigma_G^2} + \sum_{p=1}^{m-1}\sum_{q=p+1}^{m} \frac{\left(\widehat{r_{p,q}}-r_{p,q}\right)^2}{2\sigma_R^2} I_{p,q}\right] \tag{3}$$

in which we have added the indicator variables $I_{p,q}$ which equal unity if the p,q range is measured, zero if not. For convenience, we subtract $l_0(data)$ from each of these to yield the m test metrics (effectively log-likelihood ratios)

$$T_j = \frac{\left(\widehat{e_j}-e_j\right)^2+\left(\widehat{n_j}-n_j\right)^2}{2\sigma_G^2} - \frac{\left(\widehat{e_j}-e_S\right)^2+\left(\widehat{n_j}-n_S\right)^2}{2\sigma_S^2}$$

Unfortunately, for $j = 1, \ldots m$, each of these contains unknown quantities; specifically, e_S, n_S, e_j, n_j, and σ_S. A common approach, *generalized likelihoods* [8], replaces each unknown with its Maximum Likelihood Estimate (MLE) which we will represent using tildes (˜):

- Since e_S and n_S are defined under H_j we find their MLEs by maximizing the log-likelihood function in Eq. (3); this is equivalent to minimizing the expression within the brackets. Since e_S and n_S only appear in the first fraction the extremum is obviously at $\widetilde{e_S} = \widehat{e_j}$ and $\widetilde{n_S} = \widehat{n_j}$; after scaling by $2\sigma_G^2$ and taking a square root, the j^{th} test metric is

$$T_j = \sqrt{\left(\widehat{e_j} - \widetilde{e_j}\right)^2 + \left(\widehat{n_j} - \widetilde{n_j}\right)^2}$$

the distance between the GNSS location for vehicles j and the MLE of its location Of significance, we note that there is no need to estimate σ_S, that these test statistics are independent of the spoofer's noise level.

- The MLEs $\widetilde{e_j}$ and $\widetilde{n_j}$ are found by maximizing Eq. (2), recognizing that e_j and n_j implicitly appear in each $r_{j,k}$ term as in Eq. (1). The derivative of the log-likelihood function with respect to e_j is

$$\frac{\partial l_0}{\partial e_j} = \frac{1}{\sigma_G^2}\left(\widehat{e_j} - e_j\right) + \sum_{i=1, i\neq j}^{m} \frac{1}{\sigma_R^2}\left(\widehat{r_{i,j}} - r_{i,j}\right)\frac{e_j - e_i}{r_{i,j}} I_{i,j}$$

The n_j derivative is similar. Setting these derivatives to zero yields a set of necessary conditions for the MLEs. After algebraic manipulation each $\widetilde{e_j}$ must satisfy

$$\begin{aligned} \widetilde{e_j} &= \widehat{e_j} + \sum_{i=1, i\neq j}^{m} \frac{\sigma_G^2}{\sigma_R^2}\left(\widehat{r_{i,j}} - \widetilde{r_{i,j}}\right)\frac{\widetilde{e_j} - \widetilde{e_i}}{\widetilde{r_{i,j}}} I_{i,j} \\ &= \widehat{e_j} + \Delta_{e,j} \end{aligned} \tag{4}$$

with the definition

$$\widetilde{r_{i,j}} \equiv \sqrt{\left(\widetilde{e_i} - \widetilde{e_j}\right)^2 + \left(\widetilde{n_i} - \widetilde{n_j}\right)^2}$$

In the second line of Eq. (4) we recognize $\Delta_{e,j}$ as the east offset from the GNSS measurement to the MLE of location. In a similar fashion the n_j derivative yields the requirement on the MLE of the north component

$$\begin{aligned} \widetilde{n_j} &= \widehat{n_j} + \sum_{i=1, i\neq j}^{m} \frac{\sigma_G^2}{\sigma_R^2}\left(\widehat{r_{i,j}} - \widetilde{r_{i,j}}\right)\frac{\widetilde{n_j} - \widetilde{n_i}}{\widetilde{r_{i,j}}} I_{i,j} \\ &= \widehat{n_j} + \Delta_{n,j} \end{aligned} \tag{5}$$

and the test metric reduces to

$$T_j = \sqrt{\Delta_{e,j}^2 + \Delta_{n,j}^2} \tag{6}$$

While these expressions are quite sensible, that (1) the MLE of the true position is the GNSS position plus an offset due to the range measurements and the positions of the other vehicles in the platoon (and that each offset depends upon the relative accuracy of the GNSS and range measurements) and (2) that the test metrics are the nearness of the GNSS data to the best estimate of position, this result is not yet useful in that the expressions for the corrections are themselves functions of the MLEs of the vehicle positions. A method to iteratively solve these expressions is presented in the Appendix.

Normally in an m-ary hypothesis scenario (with a belief that all of the hypotheses are equally likely) we choose that hypothesis with the largest test metric. For a Neyman-Pearson formulation (controlled false alarm probability and maximum detection probability) the resulting test is

$$\max_{j=1,\ldots m} T_j \underset{\text{no spoofing}}{\overset{\text{spoofing}}{\gtrless}} \lambda \tag{7}$$

for some $\lambda > 0$. In words, if the largest of the test metrics (say T_k) is greater than the threshold, λ, we declare spoofing; otherwise, we declare no spoofing (H_0). The threshold λ is chosen to limit the probability of false alarm, declaring spoofing when no spoofing is present, to a low level (often represented by α, $0 < \alpha \ll 1$)

$$\mathrm{P_{fa}} = \mathrm{Prob}\left(\text{ any } T_j > \lambda | \mathrm{H}_0\right) \leq \alpha$$

We are also interested in the probability of detection, declaring spoofing when it indeed is occurring

$$\mathrm{P_d}(\mathrm{H}_k) = \mathrm{Prob}\,(\text{ any } T_j > \lambda | \mathrm{H}_k)$$

Since we have a test metric for each vehicle, T_j, we can extend a decision for spoofing to include an estimate of which vehicle is being spoofed; specifically, identifying the one with the largest test metric. Define the probability of declaring spoofing *and* correctly identifying the vehicle being spoofed as

$$\mathrm{P_d}^*(\mathrm{H}_k) = \mathrm{Prob}\left(\text{ any } T_j > \lambda \;\&\; \arg\left\{\max_j T_j\right\} = k | \mathrm{H}_k\right)$$

We tabulate both of these detection probabilities in the examples later in this paper.

THE CASE OF TWO VEHICLES – AN ASIDE

Consider the example of $m = 2$ vehicles (with one range measurement). In this case we can solve for the MLEs under H_0 exactly (see [6] for details of a similar development) as

$$\widetilde{e_1} = \widehat{e_1} + \Delta_e \qquad \widetilde{e_2} = \widehat{e_2} - \Delta_e$$

$$\widetilde{n_1} = \widehat{n_1} + \Delta_n \qquad \widetilde{n_2} = \widehat{n_2} - \Delta_n$$

with

$$\Delta_e = \frac{\sigma_G^2}{\sigma_R^2 + 2\sigma_G^2}\left(1 - \frac{\widehat{r}}{\breve{r}}\right)(\widehat{e_2} - \widehat{e_1}) \qquad \Delta_n = \frac{\sigma_G^2}{\sigma_R^2 + 2\sigma_G^2}\left(1 - \frac{\widehat{r}}{\breve{r}}\right)(\widehat{n_2} - \widehat{n_1})$$

in which $\breve{r}$ is defined as the distance between the GNSS measurements

$$\breve{r} = \sqrt{(\widehat{e_2} - \widehat{e_1})^2 + (\widehat{n_2} - \widehat{n_1})^2}$$

The result of these reductions on the two test metrics is interesting. Specifically,

$$T_1 = \sqrt{\Delta_e^2 + \Delta_n^2} = T_2$$

identical tests! In other words, if we decide spoofing with only two vehicles then we cannot say which vehicle is being spoofed and which has good GNSS data. Substituting in for the MLE offsets the test metric is

$$T = \sqrt{\Delta_e^2 + \Delta_n^2} = \frac{\sigma_G^2}{\sigma_R^2 + 2\sigma_G^2}\sqrt{(\breve{r} - \widehat{r})^2}$$

We can ignore the constant to yield

$$T = |\breve{r} - \widehat{r}| \underset{\mathrm{H}_0}{\overset{\mathrm{H}_{1,2}}{\gtrless}} \lambda$$

The optimum test for two vehicles is seen to be the comparison of the measured range to the GNSS derived range. If the two measurements are close to each other (within λ) we decide H_0; if the absolute difference is larger we decide that one of the two vehicles is being spoofed.

The simplicity of the $m = 2$ case allows for an approximate analysis of performance of this test (again, see similar work in [6] for details). Specifically, the false alarm and detection probabilities are

$$\mathrm{P_{fa}} \approx 2Q\left(\frac{\lambda}{\sqrt{2\sigma_G^2 + \sigma_R^2}}\right) \qquad \text{and} \qquad \mathrm{P_d} \approx Q\left(\frac{\lambda + \beta}{\sqrt{2\sigma_S^2 + \sigma_R^2}}\right) + Q\left(\frac{\lambda - \beta}{\sqrt{2\sigma_S^2 + \sigma_R^2}}\right)$$

in which $Q(x)$ is the standard Gaussian tail probability and β is the range difference caused by the spoofer (effectively, the projection of the spoofing offset onto the line connecting the true positions, see Figure 2). The ROC, or Receiver Operating Characteristic, curve is a plot of the probability of detection, $\mathrm{P_d}$, versus the probability of false alarm,

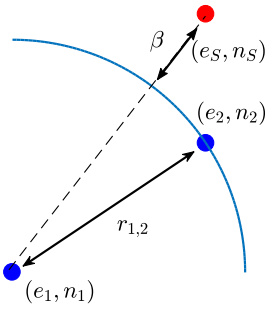

Figure 2: The definition of β.

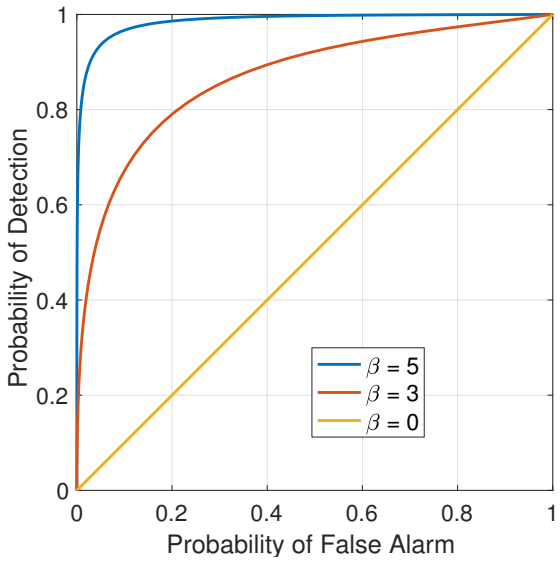

Figure 3: The performance of the $m = 2$ example for different values of β.

P_{fa}, for a hypothesis test. Since we normally desire high P_d and small P_{fa}, curves further to the upper left of the graph depict better performance. As an example, let $\sigma_G = \sigma_S = 1$ and $\sigma_R = 0.25$ meter, respectively. Figure 3 shows the ROC curves for $\beta = 0, 3$ and 5 meters (0 meaning that the spoofer creates a new location so that the distance between the two vehicles is unchanged). We note that the longer the along track offset, the better the detectability; orthogonal movement is unobservable!

MORE VEHICLES

While Eq. (6) defines the test metrics for the hypothesis test in Eq. (7), the MLEs of the east and north positions for each vehicle (as defined by Eqs. (4) and (5)) are sufficiently complicated that their direct solution (and, hence, their statistical characterization under the hypotheses) appears impossible except for the $m = 2$ case as developed above. To examine scenarios with three or more vehicles, we implemented a simple and efficient numerical procedure to solve for the vehicle location MLEs (details appear in the Appendix) and simulated the performance of the hypothesis test; these are presented below.

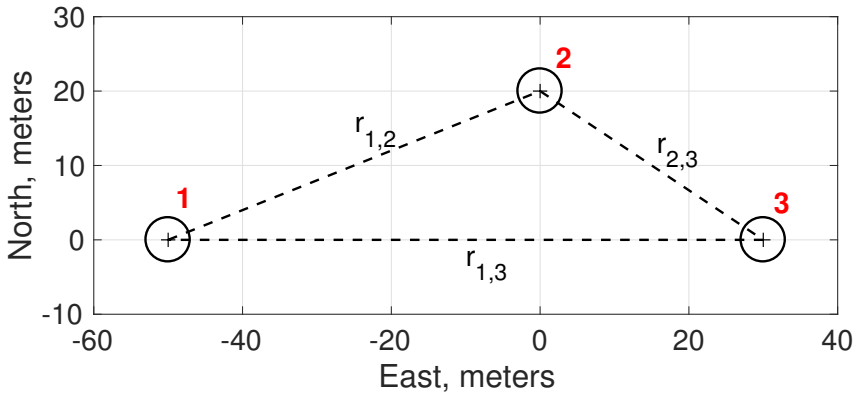

Figure 4: Example 1 – True locations of the three vehicles.

As a first example, let $m = 3$ and assume that all three range measurements are available. As shown in Figure 4, the vehicles are assumed to be at locations $(-50, 0)$, $(0, 20)$, and $(30, 0)$ meters; the black circles and plus signs show the three vehicles' true locations while the true ranges are the lengths of the dotted lines. Figure 5, subfigure (a), shows a typical set of measurements under H_0; the measurement standard deviations are assumed to be $\sigma_G = 1.0$ and $\sigma_R = 0.25$ meters:

- The larger black circles and plus signs (partially obscured) still show the three vehicles' true locations.
- The filled blue circles show the GNSS measurements of position.
- The blue lines show the range measurement drawn to scale between the GNSS measurements.

Since this example has no spoofing, the measurements look okay, close to the true configuration. Subfigure (b) shows the MLEs as red squares; the red lines show the range measurements relative to the MLEs and the values of the three test metrics (the distances from the GNSS measurements to the MLEs) are listed along the bottom, respectively. It appears that the MLE procedure has modified the locations somewhat (between about 0.5 and 1.5 meters). To get a better sense of the impact of the MLE computation, subfigure (c) zooms in on the area near vehicle 2. The blue lines (the range measurements) don't quite meet at the GNSS position for vehicle 2; $\widehat{r_{1,2}}$ was a little too short. When compared at the MLE location (in red, perturbed a bit toward the southwest), the range measurements appear to match better (recall that the range measurements are assumed to be four times more precise than the GNSS positions).

To understand what happens under spoofing we allow the spoofer to distort the GNSS position for vehicle 2 by 5 meters to the west (left in these figures). Figure 6, subfigure (a), shows a typical set of measurements under H_2; not only is that GNSS position far from truth, but the range measurements (the blue lines) don't meet well at the measured positions. The MLEs of position, shown in subfigure (b), perturb the three estimated positions to be a much better fit of the range measurements; this is particularly pronounced for vehicle 2 as seen in subfigure (c). The test metrics (distances moved) for all three vehicles are much larger than they were under H_0 (the MLE approach perturbs *all* of the positions trying to better fit the range data) with the term for vehicle 2 (3.5774 from Figure 6, subfigure (b)) being the largest. To observe how well the spoof detection algorithm works Figure 7 shows estimates of P_d and P_{fa} (10,000 point simulations each) for this spoofing situation (the left subfigure shows the entire ROC range; the right subfigure zooms in on small P_{fa}). Two ROCs are shown: the upper one (blue) is the performance of the detector correctly identifying spoofing, $P_d(H_2)$ versus P_{fa}, but not trying to identify which vehicle is being spoofed. The lower curve (red) is the probability of detecting both the occurrence of spoofing *and* that vehicle 2 is the one being spoofed, $P_d{}^*(H_2)$ versus P_{fa}. The two dots in each subfigure mark the performance at 1% false alarm rate; detection rates of 81% and 77%, respectively.

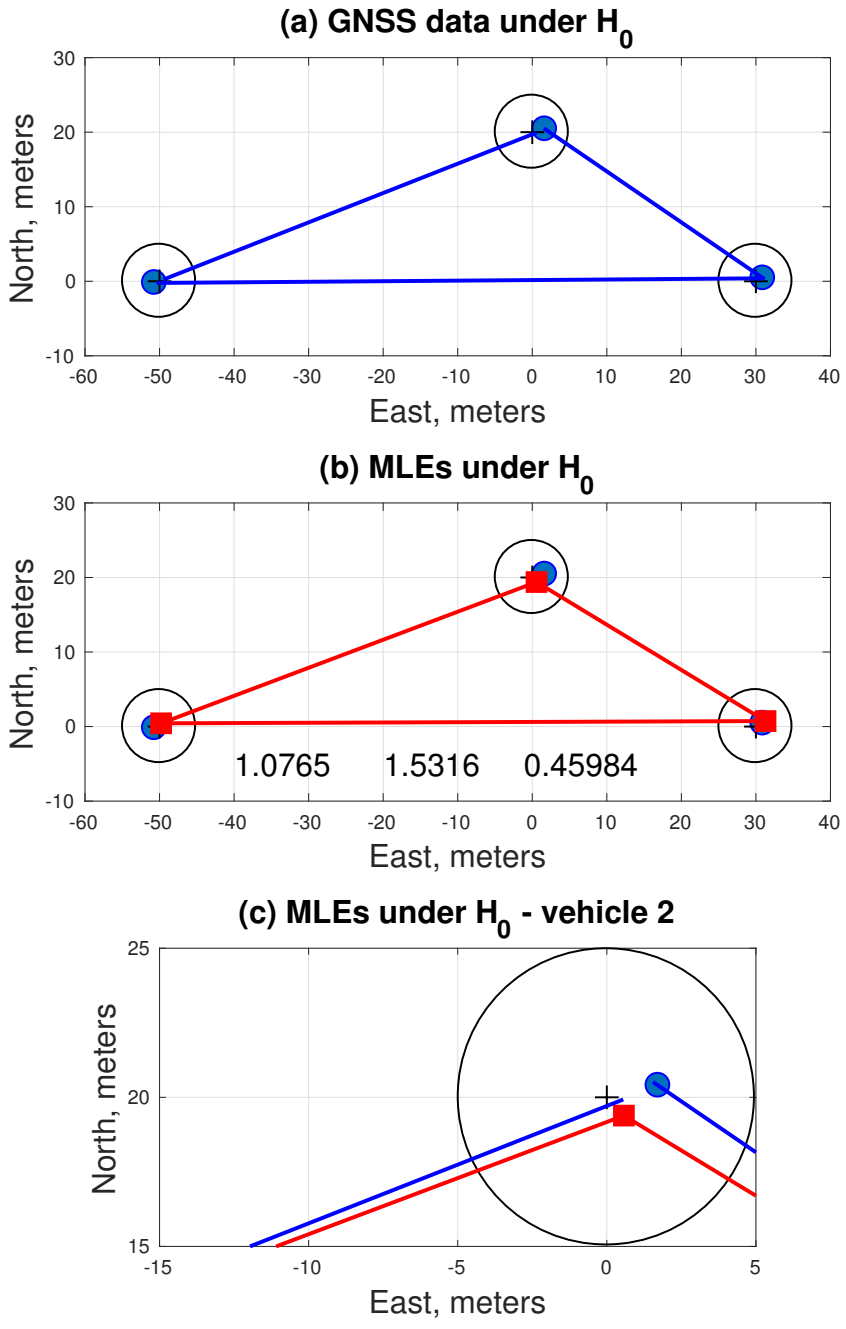

Figure 5: Example 1 – A typical situation under H_0: (a) the GNSS and range measurements; (b) the MLE estimates of position; (c) close-up of the results for vehicle 2.

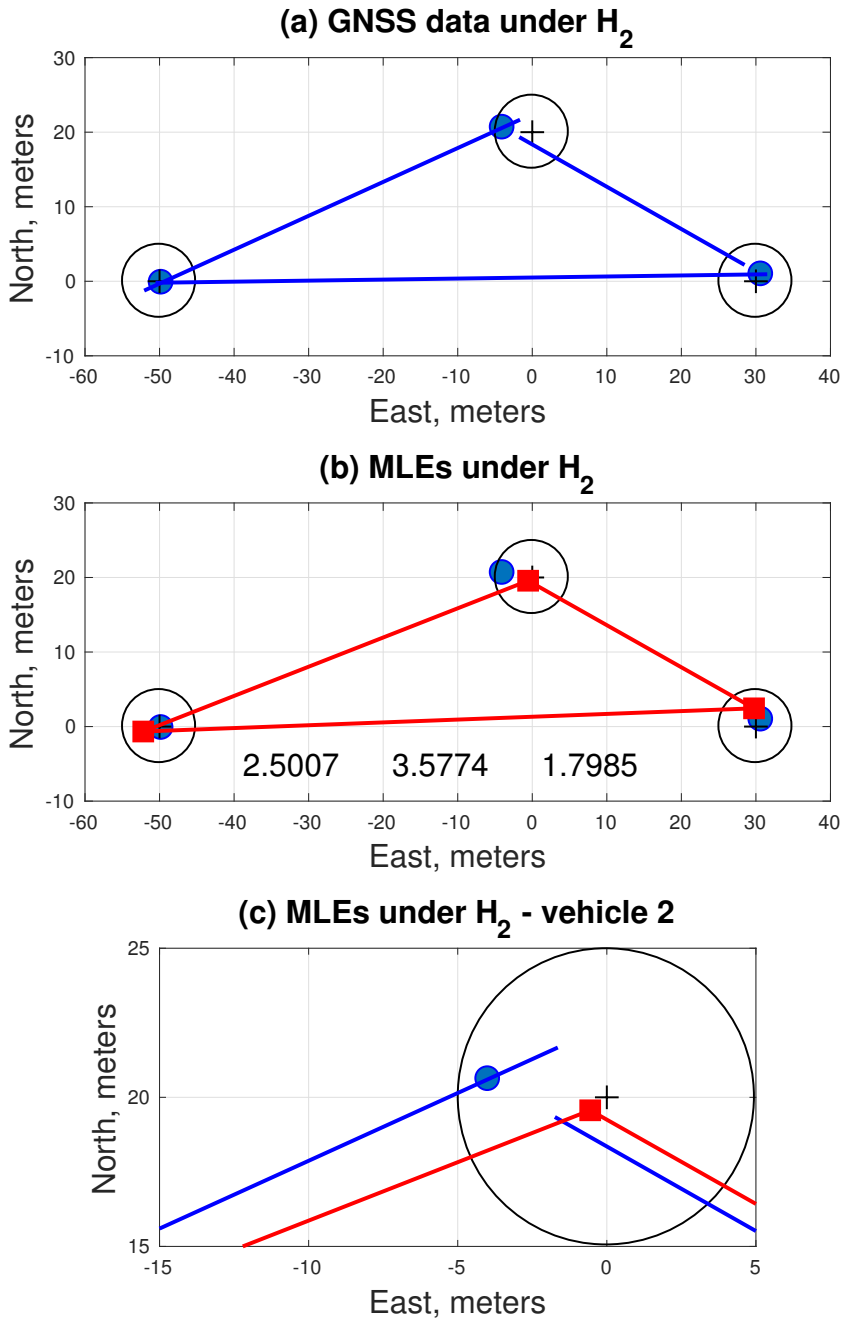

Figure 6: Example 1 – A typical situation under H_2: (a) the GNSS and range measurements; (b) the MLE estimates of position; (c) close-up of the results for vehicle 2.

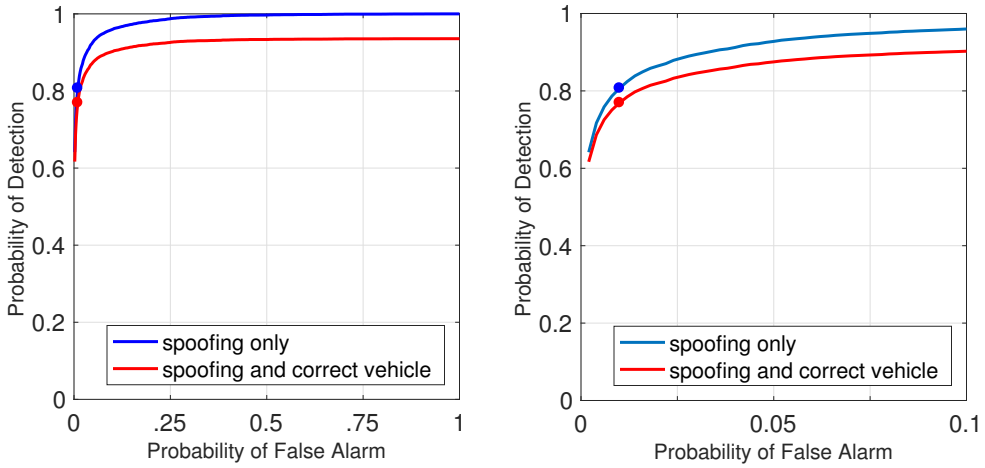

Figure 7: The performance of the $m = 3$ example; spoofing of vehicle 2 by 5 meters to the left.

While these initial results look pretty good, a number of questions come to mind. For example, *Is the performance sensitive to the direction of the spoofing?* or *Is the performance consistent across the three vehicles?* Toward answering these questions we ran a number of Monte Carlo simulations:

- The first, consisting of 100,000 trials, assumed hypothesis H_0 (with $\sigma_G = 1.0$ and $\sigma_R = 0.25$ meters) and its results allow us to accurately select the threshold, λ. Figure 8 shows the result. For example, a 1% probability of false alarm occurs with $\lambda = 2.52$; a 0.1% probability of false alarm occurs with $\lambda = 3.04$.
- Next, we ran a total of 540 (3×180) simulations (10,000 points each), one for each vehicle with spoofer distortion of 5 meters at two degree increments of azimuth (0 degrees being due North and increasing angle in a clockwise direction). Using the threshold for 1% false alarm rate ($\lambda = 2.52$) Figure 9 shows the resulting detection probabilities:
  - Each vehicle has two curves, solid for correctly detecting spoofing, $\mathrm{P_d}(\mathrm{H}_k)$, and dashed for also correcting identifying that vehicle k was spoofed, $\mathrm{P_d}^*(\mathrm{H}_k)$.
  - The performance points marked in Figure 7 are marked on the blue curves (at azimuth 270°).

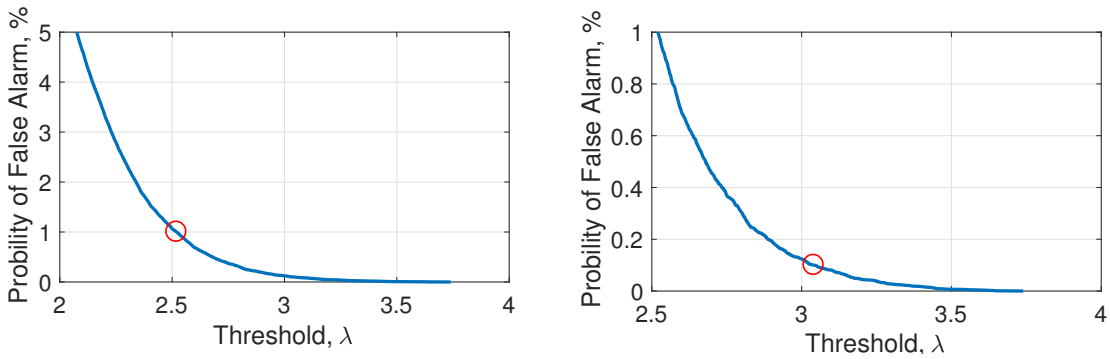

Figure 8: Test thresholds for the $m = 3$ example.

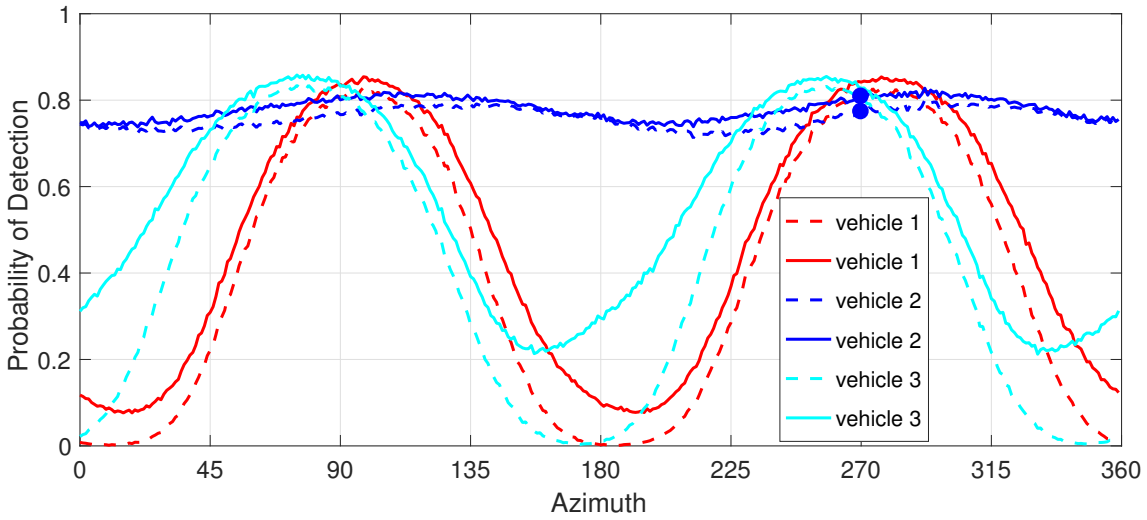

Figure 9: Detection probabilities for the $m = 3$ example, P_{fa} set to 1%. The solid line is the detection of spoofing only, P_d; the dashed line includes correctly identifying the vehicle being spoofed.

– Note that spoof detection of vehicle 2 is well achievable, no matter what direction. Vehicles 1 and 3, however, have situations in which the spoofing is not very detectible, especially at spoofing angles near 0° and 180° at which the performance is astonishing bad(!). These are, of course, situations in which the spoofing direction is nearly orthogonal to the ranges to the other vehicles. And while spoofing is detectible, the correct identification of the vehicle becomes impossible at some angles.

Clearly one could generate an infinity of examples of different configurations; however, our primary interest is to observe how well the proposed detection method works for larger platoon sizes and different range configurations. Toward that end our final example considers a platoon of 8 vehicles and examines two scenarios with different amounts of ranging data. Figure 10 shows the situation of 8 vehicles clustered together; the locations of the vehicles are the intersections of the range lines and are numbered 1 to 8. One of the simulation scenarios includes all $\binom{8}{2} = 28$ possible range measurements (left subfigure – blue) while the second only includes 12 ranges to the "nearest neighbor" vehicles (right subfigure – red). Figure 11 shows the performance results, again for a 1% false alarm rate. The 8 subfigures show the probabilities of detection with the layout matching the underlying platoon configuration. In all

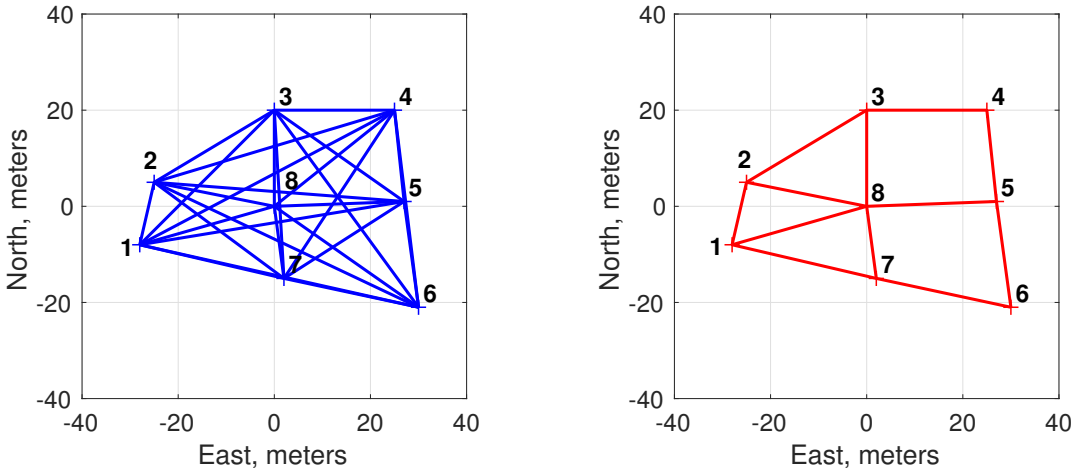

Figure 10: Larger example – two different levels of range measurements.

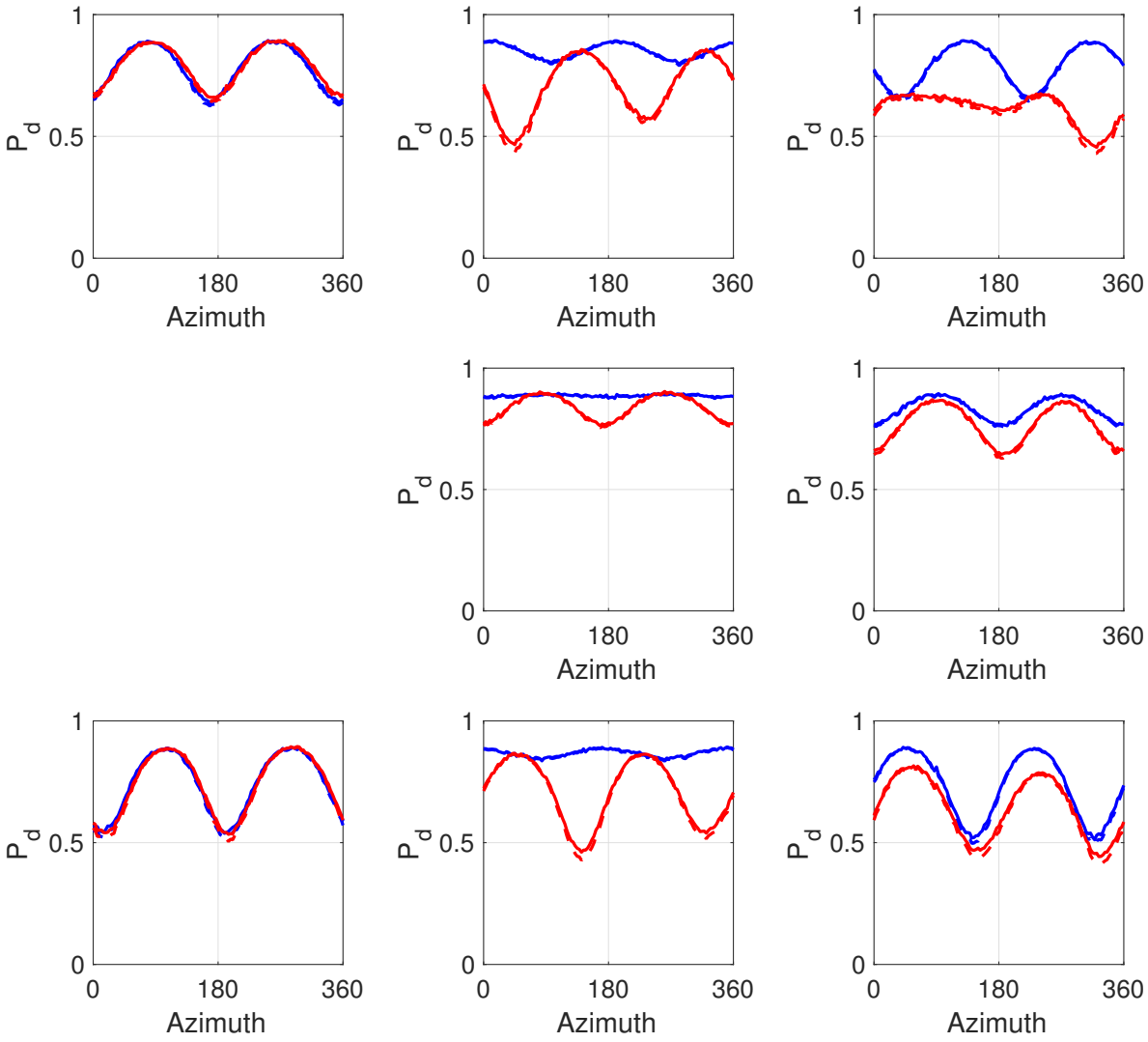

Figure 11: Larger example – results for 1% false alarm rate: blue is all ranges while red is nearby neighbors only; solid is detecting spoofing only while dashed includes correctly identifying which vehicle is being spoofed.

of the subfigures, the blue curves (both solid and dashed) are for the fully connected platoon while the red (also both solid and dashed) are for nearest neighbor ranges only. We note

- The inclusion of more vehicles has brought the ability of correctly identifying which one is being spoofed up to the level of detecting spoofing alone (i.e. the solid and dashed lines are nearly equivalent).
- As expected, the more limited set of ranges has a negative impact on performance; vehicles 3, 4, and 7 have a notable drop in performance with fewer ranges.

EXTENSIONS

The above presented ideas and results can be extended to more realistic statistical models. We continue to assume independence from sensor to sensor, but consider the following:

- Let the range measurements be unbiased Gaussian random variables with $\sigma_{Ri,j}$ representing the standard deviation of $\widehat{r_{i,j}}$. For example, measurements at longer ranges might be of lower precision.
- Let the vectors $\mathbf{x}_k = \begin{bmatrix} e_k \\ n_k \end{bmatrix}$ and $\widehat{\mathbf{x}_k} = \begin{bmatrix} \widehat{e_k} \\ \widehat{n_k} \end{bmatrix}$ represent the position and GNSS measurement for vehicle k, respectively, $k = 1, 2, \ldots m$. Under no spoofing assume that $\widehat{\mathbf{x}_k}$ has mean $\mathbf{x}_k$ and covariance matrix Σ_k, potentially different for each vehicle, while under spoofing $\widehat{\mathbf{x}_k}$ has mean $\mathbf{s} = \begin{bmatrix} e_S \\ n_S \end{bmatrix}$ and covariance matrix Σ_S.

With these more general models the likelihoods are

$$L_0\left(data\right) = \prod_{k=1}^{m} \frac{1}{2\pi|\Sigma_k|^{1/2}} e^{-\frac{1}{2}\left(\widehat{\mathbf{x}_k}-\mathbf{x}_k\right)^T \Sigma_k^{-1}\left(\widehat{\mathbf{x}_k}-\mathbf{x}_k\right)} \prod_{p=1}^{m-1}\prod_{q=p+1}^{m} \frac{1}{\sqrt{2\pi}\sigma_{Rp,q}} e^{-\frac{1}{2\sigma_{Rp,q}^2}\left(\widehat{r_{p,q}}-r_{p,q}\right)^2}$$

and

$$\begin{aligned} L_j\left(data\right) = & \frac{1}{2\pi|\Sigma_S|^{1/2}} e^{-\frac{1}{2}\left(\widehat{\mathbf{x}_j}-\mathbf{s}\right)^T \Sigma_S^{-1}\left(\widehat{\mathbf{x}_j}-\mathbf{s}\right)} \\ & \times \prod_{k=1,k\neq j}^{m} \frac{1}{2\pi|\Sigma_k|^{1/2}} e^{-\frac{1}{2}\left(\widehat{\mathbf{x}_k}-\mathbf{x}_k\right)^T \Sigma_k^{-1}\left(\widehat{\mathbf{x}_k}-\mathbf{x}_k\right)} \prod_{p=1}^{m-1}\prod_{q=p+1}^{m} \frac{1}{\sqrt{2\pi}\sigma_{Rp,q}} e^{-\frac{1}{2\sigma_{Rp,q}^2}\left(\widehat{r_{p,q}}-r_{p,q}\right)^2} \end{aligned}$$

Simplifying as above (i.e. taking logarithms, normalizing by the result under H_0, and dropping unnecessary constants) yields the test metrics

$$T_j = \left(\widehat{\mathbf{x}_j} - \mathbf{x}_j\right)^T \Sigma_j^{-1} \left(\widehat{\mathbf{x}_j} - \mathbf{x}_j\right) - \left(\widehat{\mathbf{x}_j} - \mathbf{s}\right)^T \Sigma_S^{-1} \left(\widehat{\mathbf{x}_j} - \mathbf{s}\right) + \log|\Sigma_j|$$

The MLE of $\mathbf{s}$ is still the GNSS measurement $\widehat{\mathbf{x}_j}$ so the test statistics reduce to

$$T_j = \left(\widehat{\mathbf{x}_j} - \widetilde{\mathbf{x}_j}\right)^T \Sigma_j^{-1} \left(\widehat{\mathbf{x}_j} - \widetilde{\mathbf{x}_j}\right) + \log|\Sigma_j|$$

and do not require knowledge of the spoofer's statistics. The MLE of $\mathbf{x}_j$ must satisfy

$$\widetilde{\mathbf{x}_j} = \widehat{\mathbf{x}_j} + \sum_{p=1,p\neq j}^{m} \frac{\widehat{r_{j,p}} - \widetilde{r_{j,p}}}{\sigma_{Rj,p}^2} \frac{\Sigma_j\left(\widetilde{\mathbf{x}_j} - \widetilde{\mathbf{x}_p}\right)}{\widetilde{r_{j,p}}} I_{j,p} = \widehat{\mathbf{x}_j} + \mathbf{\Delta}_{\mathbf{x}} \tag{8}$$

just a vectorized version of Eqs. (4) and (5) adding the covariances. The iterative method espoused in the Appendix is still useful in solving for the MLEs. Substituting back into the test statistic we have

$$T_j = \mathbf{\Delta}_{\mathbf{x}}^T \Sigma_j^{-1} \mathbf{\Delta}_{\mathbf{x}} + \log|\Sigma_j| \tag{9}$$

We note that the addition of $\log|\Sigma_j|$ effectively makes the threshold in Eq. (7) vehicle dependent.

Further, these spoofing tests can be extended to three dimensions. Rather trivially, the vectors in Eqs. (8) and (9) are expanded as

$$\mathbf{x}_k = \begin{bmatrix} e_k \\ n_k \\ u_k \end{bmatrix} \qquad \text{and} \qquad \widehat{\mathbf{x}_k} = \begin{bmatrix} \widehat{e_k} \\ \widehat{n_k} \\ \widehat{u_k} \end{bmatrix}$$

adding an *up* component, u_k, for each vehicle; the covariance matrices Σ_k grows to 3-by-3.

CONCLUSIONS/FUTURE

We have developed a hypothesis testing procedure to detect spoofing of a platoon of vehicles whose relative locations are only approximately known through inter-vehicle range measurements. From an examination of many examples we note:

- The ability to detect spoofing depends largely on the relative platoon geometry and the direction of spoofing.
- The test not only detects spoofing, but also identifies which vehicle is being spoofed (although with somewhat poorer performance).

One obvious limitation of this work is the assumption that, at most, one vehicle was being spoofed; we respond in two ways:

- If a single spoofer impacts two vehicles then their common spoofed position (recall that a single spoofer can only generate one position solution, no matter where the receiver is [9]) makes the detection problem quite simple (e.g. see the results in [10]).
- The above model can be expanded to multiple spoofers separately impacting multiple vehicles. The solution is a combinatoric extension of the hypothesis tests (i.e. allowing for up to all 2^m possible subsets of vehicles).

Future work could include:

- In the example with $m = 3$ the detection of which vehicle was being spoofed, ${P_d}^*$, was poor for two of the three vehicles, being quite a bit below the P_d result. Conversely, for $m = 8$ the two performance curves (solid and dashed) were uniformly quite close. It might be possible to use some sort of RAIM approach to improve ${P_d}^*$ for smaller m.
- The method presented only detects spoofing, but does not mitigate it. However, the additional range information could be used to estimate the location of the spoofed vehicle via the MLE of positions.
- The examples included GNSS data at each vehicle and many/all of the potential ranges. It would be interesting to consider much sparser sensor suites such as occurs in sensor meshes (i.e. larger m with only a few GNSS "anchors" tying the mesh to the real world).

APPENDIX

The expressions in Eqs. (4) and (5) for the MLEs of the east and north positions for each vehicle are sufficiently complicated that a direct solution for the $\widetilde{e_j}$ and $\widetilde{n_j}$ appears impossible (except for the $m = 2$ case as developed above). These expressions, however, do suggest an iterative technique that converges well in practice. Specifically:

- Let $e_j^{(k)}$ and $n_j^{(k)}$, $j = 1, 2, \ldots m$, represent estimates of the MLEs after the k^{th} iteration which are initiated at the GNSS measurements: $e_j^{(1)} = \widehat{e_j}$ and $n_j^{(1)} = \widehat{n_j}$.
- Using these values, update the ranges from Eq. (1): $r_{i,j}^{(k)} = \sqrt{\left(e_i^{(k)} - e_j^{(k)}\right)^2 + \left(n_j^{(k)} - n_j^{(k)}\right)^2}$
- Define estimates of the MLE offsets at this step:

$$\Delta_{e,j}^{(k)} = \gamma^2 \sum_{i=1, i\neq j}^{m} \left(\widehat{r_{i,j}} - r_{i,j}^{(k)}\right) \frac{e_j^{(k)} - e_i^{(k)}}{r_{i,j}^{(k)}} I_{i,j} \qquad \Delta_{n,j}^{(k)} = \gamma^2 \sum_{i=1, i\neq j}^{m} \left(\widehat{r_{i,j}} - r_{i,j}^{(k)}\right) \frac{n_j^{(k)} - n_i^{(k)}}{r_{i,j}^{(k)}} I_{i,j}$$

- To get the next estimates, perturb the current values by these offsets with a small multiplicative scale factor δ:

$$e_j^{(k+1)} = \widehat{e_j}^{(k)} + \delta\, \Delta_{e,j}^{(k+1)} \qquad n_j^{(k+1)} = \widehat{n_j}^{(0)} + \delta\, \Delta_{n,j}^{(k+1)}$$

For the results above we started δ at 1% of the GNSS standard deviation and decreased it by an order of magnitude whenever the new MLEs did not decrease the log-likelihood value.

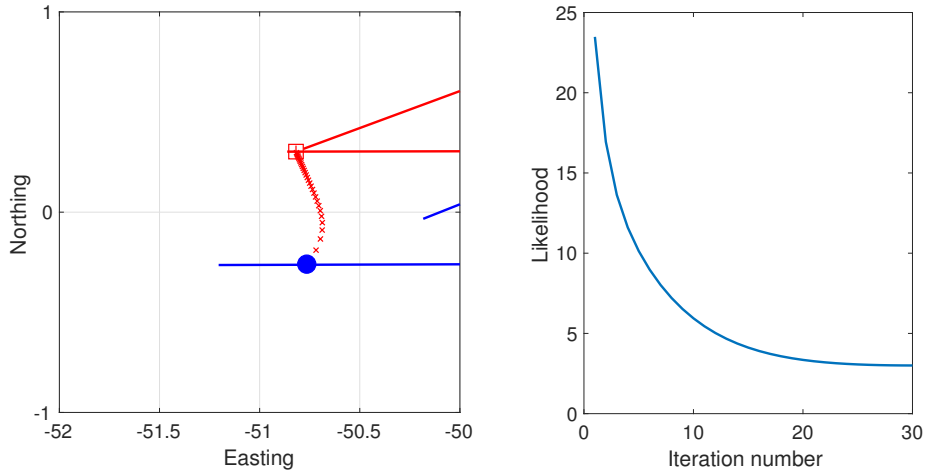

Figure 12: Typical convergence of the MLEs: (left) the location of vehicle 1 (the blue circle is the starting point, the red x's are the results after each iteration, the red square is the MLE), the blue and red lines show the range measurements aligning with each measurement; (right) the value of the log-likelihood after each iteration.

As an example Figure 12 shows the results of the iteration with $m = 3$ vehicles (similar to Figure 6, subfigure c, of the first example in the text). The left subfigure shows the location estimates, starting at the GNSS measurement (the blue circle) and quickly converging (the red x's) to the MLE (the red square). The right subfigure shows the corresponding value of the log-likelihood which is observed to have converged within about 30 cycles of the algorithm.

REFERENCES

[1] J. S. Warner and R. G. Johnston, "GPS spoofing countermeasures," *Homeland Security Jour.*, Dec. 2003.

[2] P. F. Swaszek, S.A. Pratz, B.N. Arocho, K.C. Seals, and R.J. Hartnett, "GNSS spoof detection using shipboard IMU measurements," *Proc. ION GNSS*, Tampa, FL, Sept. 2014.

[3] S. Khanafseh, N. Roshan, S. Langel, F.-C. Chan, M. Joerger, and B. Pervan, "GPS spoofing detection using RAIM with INS coupling," *Proc. ION PLANS*, Monterey CA, May 2014.

[4] C. Tanl, S. Khanafseh, and B. Pervan, "Impact of wind gusts on detectability of GPS spoofing attacks using RAIM with INS coupling," *Proc. ION Pacific PNT*, Honolulu HA, Apr. 2015.

[5] N. Carson and D. Bevly, "A robust method for spoofing prevention and position recovery in attacks against networked GPS receivers." *Proc. ION ITM*, Dana Pt, CA, Jan. 2015.

[6] P. F. Swaszek, R. J. Hartnett, and K. C. Seals, "GNSS spoof detection using range information," *Proc. ION ITM 2016*, Monterey CA, Jan. 2016.

[7] P. F. Swaszek, R. J. Hartnett, and K. C. Seals, "GNSS spoof detection using passive ranges," *Proc. ION GNSS+ 2016*, Portland OR, Sept. 2016.

[8] H. L. Van Trees, **Detection, Estimation, and Modulation Theory, Part I**, New York: Wiley, 1968.

[9] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Čapkun, "On the requirements for successful GPS spoofing attacks," *Proc. ACM CCS 2011*, Chicago, IL, Oct. 2011.

[10] P. F. Swaszek and R. J. Hartnett, "A multiple COTS receiver GNSS spoof detector - extensions," *Proc. ION ITM*, San Diego, CA, Jan. 2014.