Strategies to Design Signals to Spoof Kalman Filter

Zhongshun Zhang, Lifeng Zhou and Pratap Tokekar

***Abstract*— We study the problem of designing spoofing signals to corrupt and mislead the output of a Kalman filter. Unlike existing works that focus on detection and filtering algorithms for the observer, we study the problem from the attacker's point-of-view. In our model, the attacker can corrupt the measurements by adding spoofing signals. The attacker seeks to create a separation between the estimate of the Kalman filter with and without spoofing signals. We present a number of results on how to generate such spoofing signals, while minimizing the signal magnitude. The resulting algorithms are evaluated through simulations along with theoretical proofs.**

I. Introduction

As autonomous systems proliferate, there are growing concerns about their security and safety [1], [2]. Of particular concern is their vulnerability to signal spoofing attacks [3]. As a result, many researchers are designing algorithms that enable an *observer* to detect and mitigate signal spoofing attacks (e.g., [4]–[8]). We study the problem from the opposite (i.e., the attacker's) point-of-view. Our goal is to characterize the capabilities of the attacker that is generating the spoofing signals while assuming that the observer is using a Kalman filter for state estimation.

The problem of generating spoofing attacks has been studied specifically for GPS signals. Tippenhauer et al. [3] describe the requirements as well as present a methodology for generating spoofed GPS signals. Larcom and Liu [9] presented a taxonomy of GPS spoofing attacks.

The typical approach to mitigate sensor spoofing attacks is by designing robust state estimators [10]. Fawzi et al. presented the design of a state estimator for a linear dynamical system when some of the sensor measurements are corrupted by an adversarial attacker [11]. We focus on the scenario where the observer uses a Kalman Filter (KF) for estimating the state using measurements that are corrupted by additive spoofing signals by the attackers. We study the problem of generating spoofing signals of minimum energy that can achieve any desired separation between the KF estimate with spoofing and without spoofing. We show that for many practical cases, the spoofing signals can be generated using linear programming in polynomial time.

The work by Su et al. [12] is most closely related to ours. The authors show how to spoof the GPS signal without triggering a detector that uses the residual in the Kalman filter. They present a 1-step (greedy) online spoofing strategy that solves a linear relaxation of a Quadratically Constrained Quadratic Program (QCQP) at each timestep. We present a strategy that plans for T future timesteps, instead of just the next timestep, while minimizing the spoofing signal energy. Furthermore, we characterize the scenarios under which our strategy finds the optimal solution in polynomial time.

Based on the motion model of the target and the evolution of the KF, three problems for spoofing design are formulated in Section II. Section III shows the approaches to solve these optimization problems. The simulations for verifying spoofing strategies are given in Section IV. Section V summarizes the conclusion and future work.

The authors are with the Department of Electrical & Computer Engineering, Virginia Tech, USA. {`zszhang, lfzhou, tokekar`}`@vt.edu`.

This material is based upon work supported by the National Science Foundation under Grant #1566247.

II. Problem Formulation

Notation: We denote the set of positive real number by $\mathbb{R}^+$, the set of positive integer by $\mathbb{Z}^+$. The set of real vectors with dimension n is denoted by $\mathbb{R}^n,\ n \in \mathbb{Z}^+$, and the set of real matrices with m rows and n columns by $\mathbb{R}^{m\times n},\ m,n \in \mathbb{Z}^+$. We write $\|\cdot\|_p^p,\ p \in \mathbb{Z}^+$ as the p^{th} power of L_p vector norm, $\mathbb{E}(\cdot)$ as the expectation of a random variable, I_n as the identity matrix with size $n,\ n \in \mathbb{Z}^+$, and $\mathcal{N}(\mu, \sigma^2)$ as the normal distribution with mean μ and variance σ^2.

We consider a scenario where an observer estimates the location of a target using a KF in 2D plane. The target misleads the observer by adding spoofing signals to the observer's measurement. We define the target's model as:

$$x_{t+1} = \mathcal{F}x_t + \mathcal{G}u_t + \omega_t, \tag{1}$$

where $\mathcal{F}, \mathcal{G} \in \mathbb{R}^{2\times 2}$, $x_t \in \mathbb{R}^2$ is the position of the target, $u_t \in \mathbb{R}^2$ is the control input and $w_t \sim \mathcal{N}(0, R)$ is the Gaussian distribution, model noise of the motion model with $R \in \mathbb{R}^{2\times 2}$.

The observer estimates the target's position using linear measurement model:

$$z_t = \mathcal{H}x_t + v_t, \tag{2}$$

where $\mathcal{H} \in \mathbb{R}^{2\times 2}$ and $v_t \sim \mathcal{N}(0, Q)$ gives the measurement noise with $Q \in \mathbb{R}^{2\times 2}$.

In order to mislead the observer, the target corrupts the observer's measurement by adding spoofing signal to mislead the observer's estimate. We assume the measurement received by the observer is $\tilde{z}_t \in \mathbb{R}^2$ with spoofing signal (Equation (3)) instead of the true measurement $z_t \in \mathbb{R}^2$ without spoofing signal (Equation (2)). The spoofing signal $\epsilon_t := [\epsilon_{tx}, \epsilon_{ty}]^T \in \mathbb{R}^2$ adds additional measurement error:

$$\tilde{z}_t = z_t + \epsilon_t. \tag{3}$$

The observer uses a KF to estimate target's position with initial distribution $\mathcal{N}(m_0, \Sigma_0)$. Since it receives the spoofing measurement $\tilde{z}_t$ for updating, we denote distributions generated by the evolution of its KF as $\mathcal{N}(\tilde{m}_t, \tilde{\Sigma}_t)$ when step $t \geq$

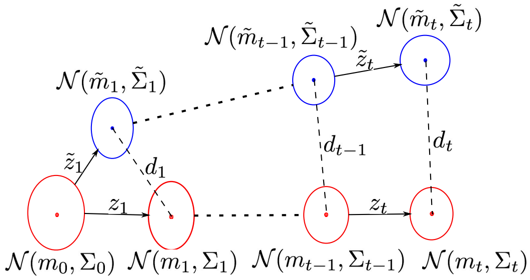


Fig. 1. The evolution of KF estimate by applying z_t and $\tilde{z}_t$, respectively.

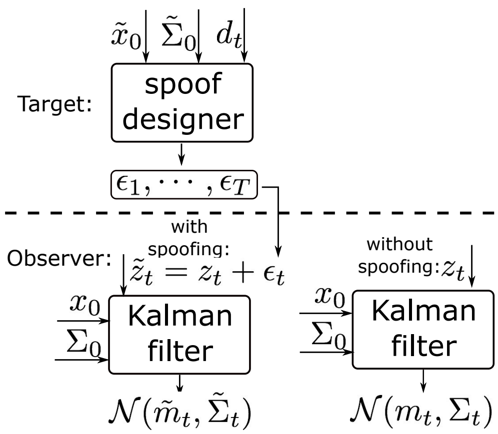


Fig. 2. Signal spoofing process and its effect on the observer's KF estimation.

$1, t \in \mathbb{Z}^+$. We also denote the distributions generated by the evolution of a KF using true measurement z_t as $\mathcal{N}(m_t, \Sigma_t)$. The goal for the target is to set the separation between the mean estimate m_t and $\tilde{m}_t$. The target's spoofing signal is each step within the planning horizon for which some desired separation, $d_t \geq 0$, must be achieved (Figure 1). Figure 2 shows the target's spoofing process where it uses the initial guess of $\mathcal{N}(m_0, \Sigma_0)$ denoted as $\mathcal{N}(\tilde{m}_0, \tilde{\Sigma}_0)$ and desired separation d_t to design spoofing signal ϵ_t. In order to avoid detection, the targets seeks to minimize the magnitude of the spoofing signal. We first propose two problems for offline scenarios as follows.

A. Offline Spoofing Signal Design with Known $\mathcal{N}(m_0, \Sigma_0)$

If the target knows $\mathcal{N}(m_0, \Sigma_0)$ of the KF, then the target can set $\mathcal{N}(\tilde{m}_0, \tilde{\Sigma}_0)$ equal to $\mathcal{N}(m_0, \Sigma_0)$.

Problem 1 (Offline with Known $\mathcal{N}(m_0, \Sigma_0)$) *Consider a target with motion model (Equation* (1)*), measurement model (Equation* (2)*), and spoofing measurement model (Equation* (3)*). Assume target knows $\mathcal{N}(m_0, \Sigma_0)$. Find a sequence of spoofing signal inputs, $\{\epsilon_1, \epsilon_2, \cdots, \epsilon_T\}$ to achieve desired separation d_t between $\tilde{m}_t$ and m_t at step t. Such that,*

$$\textit{minimize} \quad \sum_{t=1}^{T} \gamma_t \cdot \|\epsilon_t\|_p^p$$

subject to,

$$\|m_t - \tilde{m}_t\|_p^p \geq d_t^p, \quad \forall t \tag{4}$$

where $\gamma_t \in \mathbb{R}^+$ is a weighing parameter and $T \in \mathbb{Z}^+$ is the optimization horizon.

B. Offline Spoofing Signal Design with Unknown $\mathcal{N}(m_0, \Sigma_0)$

Next we consider the case where the target does not know the initial condition in the KF. Instead we assume that the initial estimate $\tilde{m}_0$, is not too far away from m_0 (in exception).

Problem 2 (Offline with Unknown $\mathcal{N}(m_0, \Sigma_0)$) *Consider a target with motion model (Equation* (1)*), measurement model (Equation* (2)*), and spoofing measurement model (Equation* (3)*). Assume the target starts spoofing with $\tilde{m}_0$, where $\mathbb{E}(m_0 - \tilde{m}_0) = M_0$ and $\tilde{\Sigma}_0 \neq \Sigma_0$. Find a sequence of spoofing signal inputs, $\{\epsilon_1, \epsilon_2, \cdots, \epsilon_T\}$ to achieve desired separation d_t between $\tilde{m}_t$ and m_t (in expectation) at step t. Such that*

$$\textit{minimize} \quad \sum_{t=1}^{T} \gamma_t \cdot \|\epsilon_t\|_p^p$$

subject to,

$$\|\mathbb{E}(m_t - \tilde{m}_t)\|_p^p \geq d_t^p, \quad \forall t \tag{5}$$

where $\gamma_t \in \mathbb{R}^+$ is a weighing parameters and $T \in \mathbb{Z}^+$ is the optimization horizon.

III. Signal Spoofing Strategies

In this section, we show how to solve Problems 1 and 2 when $p = 1$ and $p = 2$. We first present the relationship between the separation $m_t - \tilde{m}_t$ and the initial bias $m_0 - \tilde{m}_0$.

Theorem 1 *Consider a target with motion model (Equation* (1)*), measurement model (Equation* (2)*), and spoofing measurement model (Equation* (3)*). The evolutions of the KFs by applying z_t and $\tilde{z}_t$ give the distributions $\mathcal{N}(m_t, \Sigma_t)$ and $\mathcal{N}(\tilde{m}_t, \tilde{\Sigma}_t)$, respectively. The difference, $m_t - \tilde{m}_t$ is,*

$$m_t - \tilde{m}_t = \prod_{i=0}^{t-1} A_{t-i} \cdot (m_0 - \tilde{m}_0) + \sum_{i=0}^{t-2} \left(\prod_{j=0}^{i} A_{t-j} (B_{t-1-i} + C_{t-1-i}) \right) + B_t + C_t, \tag{6}$$

where $A_t = \mathcal{F} - \tilde{K}_t \mathcal{H} \mathcal{F}$, $B_t = (K_t - \tilde{K}_t) \left[z_t - \mathcal{H}(\mathcal{F} m_{t-1} + \mathcal{G} u_{t-1}) \right]$, $C_t = -\tilde{K}_t \epsilon_t$.

The proof is given in the appendix.

Corollary 1 *The expected value of the separation is,*

$$\mathbb{E}(m_t - \tilde{m}_t) = \prod_{i=0}^{t-1} A_{t-i} M_0 + \sum_{i=0}^{t-2} \left(\prod_{j=0}^{i} A_{t-j} C_{t-1-i} \right) + C_t. \tag{7}$$

Proof: From Equation 6, $\mathbb{E}\left(m_t - \tilde{m}_t\right)$ follows,

$$\begin{aligned}&\mathbb{E}\left(m_t - \tilde{m}_t\right)\\&= \mathbb{E}\left(\sum_{i=0}^{t-2}\prod_{j=0}^{i} A_{t-j}\cdot B_{t-1-i} + B_t\right) +\\&\prod_{i=0}^{t-1} A_{t-i}\mathbb{E}(m_0 - \tilde{m}_0) + \sum_{i=0}^{t-2}\left(\prod_{j=0}^{i} A_{t-j}C_{t-1-i}\right) + \tilde{K}_t\epsilon_t.\end{aligned}$$

The actual measurement is: $z_i = \mathcal{H}(\mathcal{F}m_{i-1} + \mathcal{G}u_{i-1} + w_i) + v_i$, where w_i and v_i are Gaussian noises with zero mean. The expected measurement value is: $\mathbb{E}(z_i) = \mathcal{H}(\mathcal{F}m_{i-1} + \mathcal{G}u_{i-1})$ for all i, thus $\mathbb{E}[z_i - \mathcal{H}(\mathcal{F}m_{i-1} + \mathcal{G}u_{i-1})] = 0$. Since $\mathbb{E}[B_i] = 0$, we have,

$$\begin{aligned}&\mathbb{E}\left(m_t - \tilde{m}_t\right)\\&= \prod_{i=0}^{t-1} A_{t-i}\mathbb{E}(m_0 - \tilde{m}_0) + \sum_{i=0}^{t-2}\left(\prod_{j=0}^{i} A_{t-j}\tilde{K}_{t-1-i}\epsilon_{t-1-i}\right)\\&\quad + \tilde{K}_t\epsilon_t.\end{aligned} \tag{8}$$

Since we assume $\mathbb{E}(m_0 - \tilde{m}_0) = M_0$ in Problem 2, the claim is guaranteed. ■

Theorem 1 shows the difference between the two estimated means at step t depends on the initial means, m_0 and $\tilde{m}_0$, and the initial covariance matrices Σ_0 and $\tilde{\Sigma}_0$. This is because the Kalman gain K_t depends on the covariance matrix Σ_t. If target sets $m_0 = \tilde{m}_0$ and $\Sigma_0 = \tilde{\Sigma}_0$, it has $\Sigma_t = \tilde{\Sigma}_t$ for all t since the covariance matrix is updated through the same Kalman prediction and update equation (see appendix). Thus, $B_t = 0_{2\times 2}$ and then Equation (6) can be simplified as:

$$m_t - \tilde{m}_t = \sum_{i=0}^{t-2}\left(\prod_{j=0}^{i} A_{t-j}C_{t-1-i}\right) + C_t.$$

As a result, $m_t - \tilde{m}_t$ is independent of the measurements $\{z_1, z_2, \cdots, z_t\}$ when $m_0 = \tilde{m}_0$ and $\Sigma_0 = \tilde{\Sigma}_0$. Thus, the target can generate spoofing signal inputs by solving Problem 1 offline. Similarly following Corollary 1, Problem 2 can be solved offline as well.

Problems 1 and 2 are two nonlinear programming problems for arbitrary vector norms L_p. However, when $p = 1$, they can be formulated as linear programming problems. Linear programming can be solved in polynomial time [13]. When $p = 2$, they become QCQP (Quadratically Constrained Quadratic Program). The following shows the LP and QCQP formulations.

Theorem 2 *If $p = 1$ and the elements in $\mathcal{F}$ and $I - K_t\mathcal{H}$ are all positive, then Problems 1 and 2 can be solved optimally with linear programming. If $p = 1$ and the elements in $\mathcal{F}$ and $I - K_t\mathcal{H}$ are not all positive, then Problems 1 and 2 can be solved optimally with 4^k linear programming instances. If $p = 2$ and $\{\mathcal{H}, \mathcal{F}, Q, R\}$ are diagonal matrices, then Problems 1 and 2 can be solved optimally with linear programming.*

A. Linear Programming Formulation for L_1 Vector Norm

Here, we show how to formulate Problem 1 using linear programming. A similar procedure can be applied to formulate Problem 2 as linear programming.

The constraint in Problem 1 (Equation 4) follows:

$$\begin{aligned}\|m_t - \tilde{m}_t\|_1 &= \left\|\sum_{i=0}^{t-2}\left(\prod_{j=0}^{i} A_{t-j}C_{t-1-i}\right) + C_t\right\|_1\\&= \left\|\sum_{i=0}^{t-2}\left(\prod_{j=0}^{i} A_{j+1}\cdot\tilde{K}_{t-1-i}\cdot\epsilon_{t-1-i}\right) + \tilde{K}_t\epsilon_t\right\|_1\\&\geq d_t,\end{aligned} \tag{9}$$

where $t = 1, 2, \cdots, T$. $\prod_{j=0}^{i} A_{t-j}\cdot\tilde{K}_i \in \mathbb{R}^{2\times 2}$ is a constant matrix for each $i \in \{1, \cdots, t-1\}$ and is calculated from the KF iteration with initial covariance Σ_0 and $\tilde{\Sigma}_0$. Since L_1 vector norm is the sum of the absolute values of the elements for a given vector, Problem 1 can be directly formulated as a linear programming problem when $p = 1$.

Then we show how to transform this constraint to a standard linear constraint form $G_t x_t \geq d_t$ with $x_t := [\epsilon_{1x}, \cdots, \epsilon_{tx}, \epsilon_{1y}, \cdots, \epsilon_{ty}]^T$. The left side of Equation (9) can be formulated as

$$\|m_t - \tilde{m}_t\|_1 = \left\|\begin{array}{c} a_0 + a_1\epsilon_{1x} + \cdots a_t\epsilon_{tx} + \cdots + a_{2t}\epsilon_{ty}\\ b_0 + b_1\epsilon_{1x} + \cdots b_t\epsilon_{tx} + \cdots + b_{2t}\epsilon_{ty}\end{array}\right\|_1 \tag{10}$$

where $a_0, a_1, \cdots, a_{2t}, b_0, b_1, \cdots, b_{2t}$ are corresponding coefficients from Equation 6.

Lemma 1 *If the elements in matrices $\mathcal{F}$ and $I - K_t\mathcal{H}$ are positive, then $\|m_t - \tilde{m}_t\|_1$ is a linear combination of $|\epsilon_{ix}|$ and $|\epsilon_{iy}|$, and Problem 1 can be solved as a single LP instance.*

Proof: According to the proof of Theorem 1 appendix, all the coefficients $\{a_1, ..., a_{2t}, b_1, ..., b_{2t}\}$ are positive if the elements in matrices $\mathcal{F}$ and $I - K_t\mathcal{H}$ are positive. Therefore, the objective function and the constraints are linear in $|\epsilon_{ix}|$ and $|\epsilon_{iy}|$. There always exists an optimal solution where all $\epsilon_{ix} \geq 0$ and $\epsilon_{iy} \geq 0$ or where all $\epsilon_{ix} \leq 0$ and $\epsilon_{iy} \leq 0$. The objective function in both cases will be the same. Without loss of generality, we can assume $\epsilon_{ix} \geq 0$ and $\epsilon_{iy} \geq 0$, which can be solved using a single LP instance. ■

The linear programming strategy containing k constraints is presented in Algorithm 1. G denotes matrix in the linear constraint $Gx \geq D_k$ where $x := [\epsilon_{1x}, \cdots, \epsilon_{Tx}, \epsilon_{1y}, \cdots, \epsilon_{Ty}]^T$ and D_k is the collection of k nonzero separations d_t, $t \in \{1, \cdots, T\}$.

If Lemma 1 does not hold, it is possible that some elements in $a_0, a_1, \cdots, a_{2t}, b_0, b_1, \cdots, b_{2t}$ can be positive and some are negative. In general, there are four different cases depending on the sign of the first row and the second row for considering each constraint $\|m_t - \tilde{m}_t\|_1 \geq d_t$ (Equation 10). Then we can obtain four linear optimization problems along four different sub-constraints of each constraint $\|m_t - \tilde{m}_t\|_1 \geq d_t$. Thus, in the worst case, the optimal solution can be obtained by solving 4^k linear optimization problems. We run Algorithm 1 4^k times by changing the sign

Algorithm 1: Linear Programming Formulation

```
1 Initial ← {(x_o, Σ_0, F, H, G, Q, R, u}
2 G ← 0_{k×2T}
3 Calculate Kalman gain K̃_1, ···, K̃_T
4 for q = 1 : k
5     for i = 1 to the q_th value in D_k //Equation (17)
6         g = ∏_{j=i}^{T-1} A_{j+1} K̃_i
7         G_{q,i} = sum of all rows in g
8 Return G
```

of rows in g (Line 6) appropriately.

B. Quadratically Constrained Quadratic Program Formulation for L_2 Vector Norm

When $p = 2$, Problems 1 and 2 can be formulated as QCQPs [14]:

$$\begin{aligned} &\text{minimize} \quad \frac{1}{2} x_\epsilon^T P_0 x_\epsilon \\ &\text{s.t.} \quad -\frac{1}{2} x_\epsilon^T D_t^T D_t x_\epsilon + d_t^2 \leq 0, \quad \forall t \in \{1, \ldots, T\} \end{aligned} \tag{11}$$

where $x_\epsilon = [\epsilon_{1x}^2, \epsilon_{1y}^2, \cdots, \epsilon_{Tx}^2, \epsilon_{Ty}^2]^T$, $P_0 = I_{2T}$, and $D_t \in \mathbb{R}^{2T \times 2T} :=$

$$\begin{bmatrix} \prod_{j=1}^{t-1} A_{j+1}\tilde{K}_0 & & \cdots & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & \prod_{j=t-1}^{t-1} A_{j+1}\tilde{K}_{t-1} & 0 & 0 & 0 \\ 0 & \cdots & 0 & \tilde{K}_t & 0 & 0 \\ 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & \cdots & 0 & 0 & 0 & \ddots \end{bmatrix}$$

Unfortunately, the QCQP formulations for these three problems are NP-hard since the constraint in each problem is concave. If $\mathcal{F}, \mathcal{G}, \mathcal{H}, \tilde{\Sigma}_0$ are diagonal matrices, it can be shown that D_t is also a diagonal matrix. We can transform the QCQP formulation to a linear programming problem by using change of variables $\{\epsilon_{tx}^2, \epsilon_{ty}^2\}$, $t = \{1, 2, ..., T\}$, and using a procedure similar to $p = 1$.

If D_t is not a diagonal matrix, one solution is to apply the inequality $\sqrt{2}\|x\|_2 \geq \|x\|_1$ between L_1 vector norm and L_2 vector norm. The constraint can be changed to L_1 vector norm, which is a stricter constraint. A sub-optimal solution can be obtained by using the L_1 vector norm.

C. Receding Horizon: Spoofing with online measurement

Problems 1 and 2 describe the offline versions for spoofing. We also extend the offline problems to an online version. The following formulates an online spoofing scenario.

Consider a target with motion model (Equation (1)), measurement model (Equation (2)), and spoofing measurement model (Equation (3)). Assume the target does not know $\mathcal{N}(x_0, \Sigma_0)$. It collects a series of measurements $\{z_1^{real}, z_2^{real}, \cdots, z_{t^o}^{real}\}$ from step 1 to current step t^o. Find a sequence of spoofing signal inputs, $\{\epsilon_{t^o}, \epsilon_{t^o+1}, \cdots, \epsilon_{t^o+H}\}$ to achieve desired separation d_t between $\tilde{m}_t$ and m_t (in expectation) within future H steps. Such that

$$\begin{aligned} &\text{minimize} \quad \sum_{t=t^0}^{t^o+H} \gamma_t \cdot \|\epsilon_t\|_p^p \\ &\text{s.t.} \quad \|\mathbb{E}(m_t - \tilde{m}_t)\|_p^p \geq d_t^p, \quad \forall t \in \{t^o, \cdots, t^o + H\} \end{aligned} \tag{12}$$

where $\gamma_t \in \mathbb{R}^+$ is a weighing parameter, t^o is the current time, and H is the predictive time horizon. The target applies $\epsilon_t = \epsilon_{t^o}$ as spoofing signal input at each step t.

IV. Simulations

In this section, we simulate the effectiveness of spoofing strategies for Problems 1, 2 and online case (Section III-C) where a target designs spoofing signals ϵ_t to mislead an observer by achieving the desired separations d_t between m_t and $\tilde{m}_t$. Our code is available online.[1]

We consider the L_1 vector norm and the following models,

$$\mathcal{F} = I_{2\times2}, \mathcal{G} = I_{2\times2}, u = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, R = 0.5I_{2\times2}, Q = 0.5I_{2\times2}.$$

Set the weight $\gamma_t = 1$ for all t.

For Problem 1, set the initial condition for the KF as,

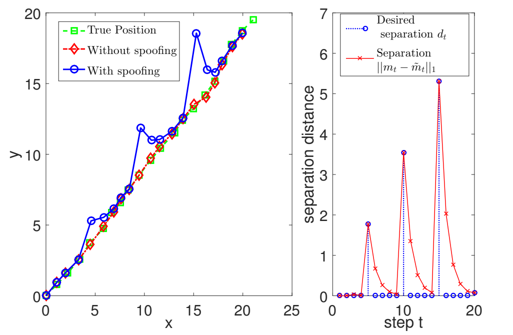
$$\Sigma_0 = I_{2\times2}, \ m_0 = \begin{bmatrix} 0 & 0 \end{bmatrix}^T.$$

Since the target knows $\mathcal{N}(x_0, \Sigma_0)$, it sets $\tilde{m}_0 = m_0$ and $\tilde{\Sigma}_0 = \Sigma_0$. We first consider a scenario where the target wants to achieve the desired separation at steps, $t = 5, 10, 15$, denoted as $d_5 = 1.77$, $d_{10} = 3.54$ and $d_{15} = 5.30$ with the optimization horizon $T = 20$. The target generates a sequence of spoofing signals $\{\epsilon_1, \cdots, \epsilon_{20}\}$ offline by using a linear programming solver. The spoofing performance is shown in Figure 3-(a) where the true separations are the same as the desired separations. Same successful spoofing achieved when the desired separations are chosen as $d_t = 0.25t\|u\|_2$, $t = \{3, ..., 15\}$, as shown in Figure 3-(b).

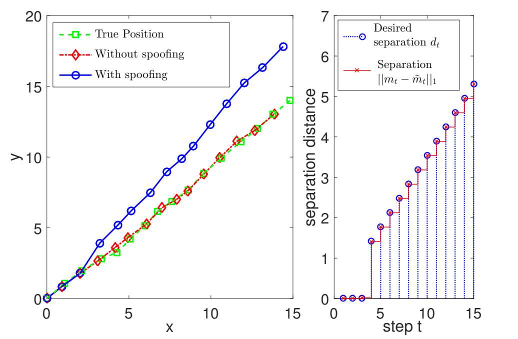
In Problem 2, the target knows $\mathbb{E}(m_0 - \tilde{m}_0) = M_0$ but does not know Σ_0. The spoofing result is no longer deterministic but holds in expectation $\|\mathbb{E}(m_t - \tilde{m}_t)\|_1 \geq d_t$. Figure 4-(a) shows spoofing signals for desired separations as $d_1 = 2$ with $T = 6$ and $M_0 = 1$. Set $\mathcal{N}(\tilde{m}_0, \tilde{\Sigma}_0)$ as $\mathcal{N}(0, 1.5I_2)$, m_0 as a random variable ($m_0 \sim \mathcal{N}(1, 1)$) and $\Sigma_0 = I_2$. In order to see the effectives of the spoofing signals $\{\epsilon_1, \cdots, \epsilon_5\}$, we conduct 100 trials for each desired separation $d_2 \in \{1, 2, 3, 4, 5\}$. Figure 4-(b) shows the $\|m_1 - \tilde{m}_1\|_1$ is no longer deterministic, but $\|\mathbb{E}(m_1 - \tilde{m}_1)\|_1$ is close to the desired value $d_1 = 2$.

For online case, spoofing signals are continuously generated by using receding horizon optimization with new noisy measurements. We set the receding horizon as $H = 15$. Even though offline strategy performs comparatively as online strategy (Figure 5), online spoofing strategy achieves almost the same separation as the desired, while offline strategy has certain divergence (Figure 6). This is because online strategy can update the measurement at each step. Figure 7 shows the online strategy applies less total spoofing magnitude than offline strategy.

[1] https://github.com/raaslab/signal_spoofing.git

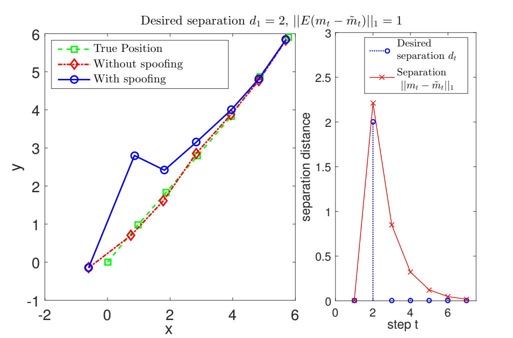


(a) Desired separations, $d_5 = 1.77$ and $d_{10} = 3.54$, with $T = 20$.

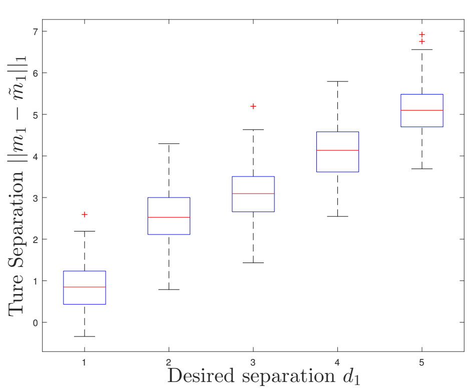


(b) Desired separations, $d_t = 0.25t\|u\|_2$, $t = \{3, ..., 15\}$ with $T = 15$.

Fig. 3. Offline signal spoofing with known (m_0, Σ_0).



(a) One of the simulation results: the desired separations $d_1 = 2$ (left) and actual separation.



(b) Results with $d_1 = \{1, 2, 3, 4, 5\}$ for 100 trials.

Fig. 4. Offline signal spoofing with unknown (m_0, Σ_0).

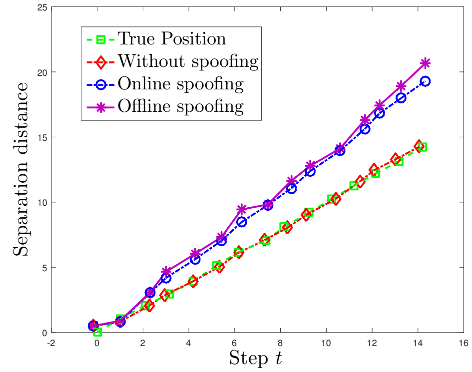


Fig. 5. Online spoofing and offline spoofing with unknown (m_0, Σ_0).

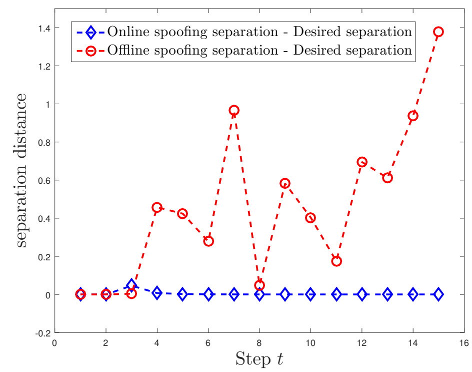


Fig. 6. Divergence caused by online spoofing and offline spoofing. The blue line denotes $(\|\tilde{m}_t - m_t\|_1 - d_t)$ for online spoofing, and red line denotes $(\|\tilde{m}_t - m_t\|_1 - d_t)$ for offline spoofing.

V. Conclusion

We study the problem spoofing signals to achieve any desired separation between a Kalman filter estimate without and with spoofing signals. Our main approach was to formulate the problems as nonlinear, constrained optimization problems in order to minimize the energy of the spoofing signal. We show that under some technical assumptions, the problems can be solved by linear programming optimally. We also present a more computationally expensive approach to solving the problem, without the aforementioned assumptions.

Our immediate future work is to study the game-theoretic aspects of the problem. In this work, we did not consider

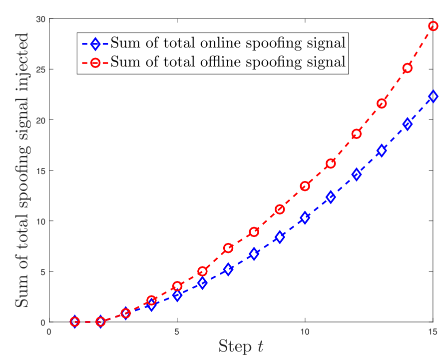


Fig. 7. Total spoofing energy injected by online and offline strategies.

any active strategy being employed by the observer to detect and/or mitigate the attack. In future works, we will consider the case of designing spoofing signals that explicitly take the attack detection and/or mitigation strategies into account. In all the problems considered in this paper, the desired separations are taken as inputs provided by the user. Instead, we can optimize over the desired separation trajectory in order to avoid detection by the observer.

APPENDIX

A. Proof of Theorem 1

Before we prove Theorem 1, we review the Kalman Filter update equations. Suppose the true measurement is z_t, the KF estimation is:

$$x_{t|t-1} = \mathcal{F}x_{t-1|t-1} + \mathcal{G}u_t, \tag{13}$$

$$x_{t|t} = \mathcal{F}x_{t|t-1} + K_t(z_t - \mathcal{H}x_{t|t-1}), \tag{14}$$

where K_t is the Kalman gain and is given by:

$$K_t = (\mathcal{F}\Sigma_{t|t-1}\mathcal{F}' + R_t)\mathcal{H}'(\mathcal{H}\Sigma_{t|t-1}\mathcal{H}' + Q_t)^{-1}. \tag{15}$$

According to the Kalman gain update equation (15), the evolution covariance matrix at step t, Σ_t, only depends on the state model parameters and the initial condition of the covariance matrix Σ_0. The Kalman gain at step t, K_t depends on the covariance matrix Σ_t. Both Σ_t and K_t do not depend on the control input series $\{u_t\}_{t=1,\cdots,k}$, measurement $\{z_t\}_{t=1,\cdots,k}$. Thus, the covariance matrix and the Kalman gain can be predicted from the KF covariance update steps.

$$\begin{aligned} \Sigma_{t+1|t} &= \mathcal{F}\Sigma_{t|t}\mathcal{F}' + R_t, \\ \Sigma_{t+1|t+1} &= (I - K_t\mathcal{H})\Sigma_{t+1|t}. \end{aligned} \tag{16}$$

From Equation (16), the Kalman gain can be predicted from the initial condition Σ_0.

We now prove our main result.

Proof: From the update of KF, we have

$$\begin{aligned} m_t &= m_{t|t-1} + K_t(z_t - \mathcal{H}m_{t|t-1}) \\ &= (I - K_t\mathcal{H})m_{t|t-1} + K_t z_t \\ &= (I - K_t\mathcal{H})(\mathcal{F}m_{t-1} + \mathcal{G}u_{t-1}) + K_t z_t. \end{aligned} \tag{17}$$

and

$$\tilde{m}_t = (I - K_t\mathcal{H})(\mathcal{F}m_{t-1} + \mathcal{G}u_{t-1}) + K_t(z_t + \epsilon_t).$$

Recursively,

$$\begin{aligned} & m_t - \tilde{m}_t \\ =& (I - K_t\mathcal{H})(\mathcal{F}m_{t-1} + \mathcal{G}u_{t-1}) + K_t z_t \\ & - [(I - \tilde{K}_t\mathcal{H})(\mathcal{F}\tilde{m}_{t-1} + \mathcal{G}u_{t-1}) + \tilde{K}_t(z_t + \epsilon_t)] \\ =& (\mathcal{F} - K_t\mathcal{H}\mathcal{F})m_{t-1} - (\mathcal{F} - \tilde{K}_t\mathcal{H}\mathcal{F})\tilde{m}_{t-1} \\ & - (K_t - \tilde{K}_t)\mathcal{H}\mathcal{G}u_{t-1} + [K_t z_t - \tilde{K}_t(z_t + \epsilon_t)] \\ =& (\mathcal{F} - \tilde{K}_t\mathcal{H}\mathcal{F})m_{t-1} - (\mathcal{F} - \tilde{K}_t\mathcal{H}\mathcal{F})m_{t-1} \\ & - (K_t - \tilde{K}_t)\mathcal{H}\mathcal{G}u_{t-1} + (K_t - \tilde{K}_t)z_t - \tilde{K}_t\epsilon_t \\ & - (K_t - \tilde{K}_t)\mathcal{H}\mathcal{F}m_{t-1} \\ =& (\mathcal{F} - \tilde{K}_t\mathcal{H}\mathcal{F})(m_{t-1} - \tilde{m}_{t-1}) \\ & + (K_t - \tilde{K}_t)[z_t - \mathcal{H}(\mathcal{F}m_{t-1} + \mathcal{G}u_{t-1})] - \tilde{K}_t\epsilon_t. \end{aligned} \tag{18}$$

Define, $A_t = \mathcal{F} - \tilde{K}_t\mathcal{H}\mathcal{F}$, $B_t = (K_t - \tilde{K}_t)[z_t - \mathcal{H}(\mathcal{F}m_{t-1} + \mathcal{G}u_{t-1})]$ and $C_t = -\tilde{K}_t\epsilon_t$. Then,

$$\begin{aligned} & m_t - \tilde{m}_t \\ &= A_t(m_{t-1} - \tilde{m}_{t-1}) + B_t + C_t \\ &= A_t[A_{t-1}(m_{t-2} - \tilde{m}_{t-2}) + B_{t-1} + C_{t-1}] + B_t + C_t \\ & \cdots \\ &= \prod_{i=0}^{t-1} A_{t-i} \cdot (m_0 - \tilde{m}_0) + (B_t + C_t) \\ & + A_t(B_{t-1} + C_{t-1}) + \cdots + A_t \cdots A_3 A_2 (B_1 + C_1) \\ &= \prod_{i=0}^{t-1} A_{t-i} \cdot (m_0 - \tilde{m}_0) + B_t + C_t \\ & + \sum_{i=0}^{t-2} \left(\prod_{j=0}^{i} A_{t-j}(B_{t-1-i} + C_{t-1-i}) \right). \end{aligned}$$

■

REFERENCES

[1] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber threats facing autonomous and connected vehicles: Future challenges," *IEEE Transactions on Intelligent Transportation Systems*, 2017.

[2] V. L. Thing and J. Wu, "Autonomous vehicle security: A taxonomy of attacks and defences," in *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016 IEEE International Conference on*. IEEE, 2016, pp. 164–170.

[3] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful gps spoofing attacks," in *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 2011, pp. 75–86.

[4] M. Al Faruque, F. Regazzoni, and M. Pajic, "Design methodologies for securing cyber-physical systems," in *Proceedings of the 10th International Conference on Hardware/Software Codesign and System Synthesis*. IEEE Press, 2015, pp. 30–36.

[5] Y. Chen, W. Trappe, and R. P. Martin, "Detecting and localizing wireless spoofing attacks," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2007. SECON'07. 4th Annual IEEE Communications Society Conference on*. IEEE, 2007, pp. 193–202.

[6] S. Gil, S. Kumar, M. Mazumder, D. Katabi, and D. Rus, "Guaranteeing spoof-resilient multi-robot networks," *Autonomous Robots*, vol. 41, no. 6, pp. 1383–1400, 2017.

[7] J. Zhang, R. S. Blum, L. M. Kaplan, and X. Lu, "Functional forms of optimum spoofing attacks for vector parameter estimation in quantized sensor networks," *IEEE Transactions on Signal Processing*, vol. 65, no. 3, pp. 705–720, 2017.

[8] X. Fan, L. Du, and D. Duan, "Synchrophasor data correction under gps spoofing attack: A state estimation based approach," *IEEE Transactions on Smart Grid*, 2017.

[9] J. A. Larcom and H. Liu, "Modeling and characterization of gps spoofing," in *Technologies for Homeland Security (HST), 2013 IEEE International Conference on*. IEEE, 2013, pp. 729–734.

[10] N. Bezzo, J. Weimer, M. Pajic, O. Sokolsky, G. J. Pappas, and I. Lee, "Attack resilient state estimation for autonomous robotic systems," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 3692–3698.

[11] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure state-estimation for dynamical systems under active adversaries," in *Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on*. IEEE, 2011, pp. 337–344.

[12] J. Su, J. He, P. Cheng, and J. Chen, "A stealthy gps spoofing strategy for manipulating the trajectory of an unmanned aerial vehicle," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 291–296, 2016.

[13] N. Karmarkar, "A new polynomial-time algorithm for linear programming," in *Proceedings of the sixteenth annual ACM symposium on Theory of computing*. ACM, 1984, pp. 302–311.

[14] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.