Distributed Target Tracking under Byzantine Data Attacks

Congqi Shen
College of Information Science and Electronic Engineering
Zhejiang University
Hangzhou 310027, China
shencq@zju.edu.cn

Huifang Chen*
College of Information Science and Electronic Engineering
Zhejiang University
Zhejiang Provincial Key Laboratory of Information Processing, Communication and Networking
Hangzhou 310027, China
chenhf@zju.edu.cn

Lei Xie
College of Information Science and Electronic Engineering
Zhejiang University
Zhejiang Provincial Key Laboratory of Information Processing, Communication and Networking
Hangzhou 310027, China
xiel@zju.edu.cn

ABSTRACT
In this paper, we investigate the problem of distributed target tracking in wireless sensor networks (WSNs) under Byzantine data attacks. The dynamics of the target is defined by an evolution process. An *M*-ary quantizer is used at sensors to obtain local measurement data. With collected local measurement data from sensors in the network, the fusion center (FC) implements the target tracking process by using unscented kalman filter (UKF). For Byzantine nodes, the attack manner is described by the quantization process, the cascade of a normal mapping function and an attack function. We analyze the effect of Byzantine data attack on the performance of the distributed target tracking in terms of posterior Cramer-Rao lower bound (PCRLB). By making the FC obtain no information from both target state evolution model and reported data, we derive the condition to make the FC incapable of estimating the target location correctly and propose the corresponding strategy for the attacker. Numerical results show that the derived condition and proposed strategy can invalidate the distributed target tracking.

CCS Concepts
•Security and privacy → network security → Mobile and wireless security

Keywords
Wireless sensor networks; distributed target tracking; Byzantine data attack.

1. INTRODUCTION
Distributed target localization and tracking are typical applications in the wireless sensor network (WSN). The framework of the distributed target tracking consists of a fusion center (FC) and a number of sensors spatially deployed in the plane of interest (POI). When a target moves within the POI, sensors sense the signal about the target, locally process the signal and send the processed measurement data to the FC. Finally, the FC estimates the location and velocity of the target using the reported data from the network.

ICCNS 2017, November 24–26, 2017, Tokyo, Japan

ACM ISBN 978-1-4503-5349-6/17/11$15.00
https://doi.org/10.1145/3163058.3163078

The survey [1] summarized the target tracking methods, where the target tracking scheme using the filtering algorithms has been widely applied [2-4]. The filtering algorithms being widely used are the extended kalman filter (EKF) [2], the unscented kalman filter (UKF) [3] and the particle filter (PF) [4]. For the UKF, the nonlinear probability density function is approximated using the unscented transform. Compared with the EKF and the PF, the UKF outperforms in terms of the estimation accuracy, and the computational complexity of the UKF is lower than that of the PF. Hence, the UKF has been widely applied in real systems.

For the distributed target tracking, some work focuses on mitigating the imperfections of the network [5-8]. In [5], the impact of measurement noise on the tracking error is analyzed. It is concluded that introducing an adaptive coefficient into the observation model can improve the tracking performance. In [6], authors presented a probabilistic sensor management scheme to maximize the information obtained at the FC. It is shown that this scheme reduces energy consumption at the cost of a little tracking error. Owing to the limited resources of the WSN, specifically the communication bandwidth, authors in [7] proposed a mechanism that the FC schedules the number of bits to be transmitted through the network by each sensor. In [8], a-bias-change-based detection approach is presented to guarantee the robustness of the network in case of the possible changing biases of local sensor data.

Byzantine data attack is a typical attack for the distributed estimation with the WSN. For Byzantine data attack for the distributed target tracking, an attacker captures some sensors and forces these compromised sensors to send falsified local measurement data in order to disrupt the target tracking process. Since the location is estimated sequentially using the distributed measurement data in the target tracking, malicious data introduced by Byzantine nodes deteriorate not only the current location estimation performance, but also the subsequent performance. Therefore, it is essential to study Byzantine data attack problem in the distributed target tracking.

In [9], secure target tracking algorithms are summarized, where the reliable target tracking under the circumstance of possible sensor failure is realized at the price of a little estimation accuracy. In [10], an object tracking scheme with node failure recovery is proposed. Nodes are divided into cells and a node failure is alarmed when an expected reply from a neighbor node is missed. However, the effect of node recovery depends on how the nodes are divided. A reputation-based network protocol is proposed for the target tracking system in [11]. Sensors monitor the action of their neighbors and obtain their trustworthiness. Using the

trustworthiness of node's neighbors in the distributed estimation, the network can tolerate some malicious data. However, all of related works mentioned above focused on bad data tolerance from the point view of the network designer.

The problem of Byzantine data attack in the distributed target tracking is first investigated from the perspective of the attacker in [12], and the impact of Byzantine data attack on the estimation performance is analyzed. By making the information contributed by the reported data from sensors be zero, authors derived the equivalent condition of making the FC incapable of tracking the target location correctly by utilizing one snapshot data collected from sensors. Since the target tracking is a sequential estimation process, except for the reported data from sensors, the state evolution model also provides information. It is concluded that the attacker should compromise at least 50% sensors in order to make the FC incapable of utilizing the reported data for the target tracking (that is, FC is blind). However, the work in [12] did not consider the state evolution model. In addition, the work in [12] studied Byzantine data attack problem in the distributed target tracking with binary quantized data. These are the motivations of our work.

In this paper, we investigate the Byzantine data attack problem for the target tracking with M-ary quantized data from the perspective of the attacker. A general linear state evolution model is used to character the dynamics of the target location. In a distributed target tracking system, sensors obtain the local measurement about the signal radiated from the target and map them into M-ary symbols. The attacker can compromise some sensors to make them send falsified local measurement data to deteriorate the network performance. The contributions of this work are two-fold. First, we analyze the impact of Byzantine data attack on the performance of the target tracking in a sequential manner. By making the FC obtain no information from the state evolution model and the received data at the current time, we derive the equivalent condition of making the FC incapable of tracking the target correctly. Second, we consider a more general quantization scheme, in which sensors transmit M-ary quantized data to the FC. And the optimal Byzantine data attack strategy is proposed for the distributed target tracking.

2. SYSTEM MODEL AND PROBLEM FORMULATION

2.1 System Model

Fig. 1 shows the model of the distributed target tracking system, which consists of N sensors, an FC and a moving target. N sensors are randomly deployed within a two-dimensional plane. Sensors sense the signal about the target continuously, map the local sensing and report the measurement data to the FC. The FC sequentially estimates the target state using the reported data with a filtering algorithm.

In this model, an attacker can capture some sensors deployed in the network and make them become Byzantine nodes. Byzantine data attack is launched by Byzantine nodes, that is, Byzantine nodes may send falsified measurement data to the FC in order to disrupt the normal sequential estimation process and degrade the distributed tracking performance. We assume that α fraction of sensors are malicious.

Since the continuous time scale is impossible in a real system, the target state varies in discrete time sequence. The discrete time dynamic equation is given by $\boldsymbol{\theta}(k)=\boldsymbol{f}(\boldsymbol{\theta}(k-1))+\boldsymbol{q}(k)$.$k$=0 denotes the initial time. Let $\boldsymbol{\theta}(k)=[x_t(k), \dot{x}_t(k), y_t(k), \dot{y}_t(k)]^T$ be the target state at time k, and $\boldsymbol{\theta}(k)\in\mathbb{R}^{4\times1}$, where $x_t(k)$ and $y_t(k)$ are the coordinates

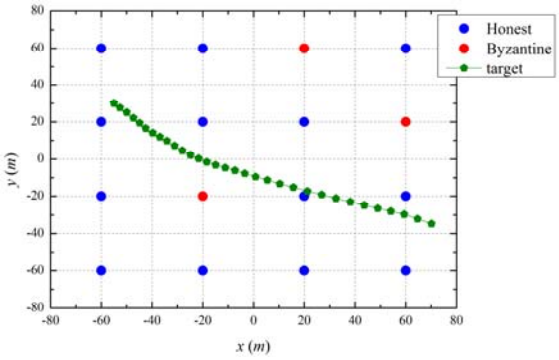


Figure 1. System model of the distributed target tracking.

of the target within the 2-D plane, $\dot{x}_t(k)$ and $\dot{y}_t(k)$ are the velocities of the target in x and y directions. $f: \mathbb{R}^4\rightarrow\mathbb{R}^4$is the model of the target dynamics. $\boldsymbol{q}(k)$ is the process noise.

2.2 Local Measurement Process and Attack Process

We assume that the target moves on the plane with a constant velocity model [13]. Hence, the target state varies as

$$\boldsymbol{\theta}(k)=\mathbf{F}\boldsymbol{\theta}(k-1)+\boldsymbol{q}(k), \tag{1}$$

where $\mathbf{F}=diag(\mathbf{F}_1,\mathbf{F}_1)$, $\mathbf{F}_1=\begin{bmatrix}1 & T\\ 0 & 1\end{bmatrix}$, $\boldsymbol{q}(k)$ is assumed to be an additive white Gaussian process noise with zero-mean and covariance matrix as $\mathbf{Q}=diag(\mathbf{Q}_1,\mathbf{Q}_1)$, and $\mathbf{Q}_1=\varepsilon\begin{bmatrix}T^3/3 & T^2/2\\ T^2/2 & T\end{bmatrix}$, in which ε and $T$are the process noise parameter and the time interval between two continuous local measurements, respectively. The initial target state $\boldsymbol{\theta}(0)$ has a prior distribution $p_0(\boldsymbol{\theta}(0))$ which is the Gaussian distribution, $\boldsymbol{\theta}(0)\sim\mathcal{N}(\bar{\boldsymbol{\theta}}_0, \bar{\Lambda}_0)$, where $\bar{\boldsymbol{\theta}}_0$ and $\bar{\Lambda}_0$ are the mean and covariance of initial target state, respectively. We assume that the FC has the knowledge about the target state evolution model in (1), and the process noise is independent across the time. As mentioned in [13], we also assume that the target motion is independent across x and y directions.

Sensors sense the signal about the target at time k, such as the signal amplitude, the distance value, and so on. Since the signal sensed by a normal sensor is related to the locations of the sensor and the target, the sensing process is denoted by function $h_i(\bullet)$. Let $s_i(k)$ be the signal sensed by sensor i at time k,and $s_i(k)=h_i(\boldsymbol{\theta}(k))+n_i(k)$, where $n_i(k)$ is the measurement noise. It is assumed that the measurement noise is independent identically distributed across the time and among sensors.

Due to the energy and bandwidth limitations in a real WSN, sensors should map the raw signal to quantized measurement data. Suppose that the threshold for M-ary quantization mechanism is given as $\boldsymbol{\eta}=[\eta_0=-\infty,\eta_1,\eta_2,\ldots,\eta_{M-1},\eta_M=+\infty]$, and M quantized data can be chosen by sensors. Let $\mathcal{M}$be the set of M-ary quantized data, and $\mathcal{M}=\{0,1,\ldots,M-1\}$. Let $u_i(k)$ denote the quantized measurement data of normal sensor i at time k, and $u_i(k)\in\mathcal{M}$.

Hence, the mapping process for normal sensors can be represented by a mapping function $g(\bullet)$ as

$$u_i(k)=g(s_i(k))=m,\ \ \eta_m \le s_i(k) < \eta_{m+1},\ m\in\mathcal{M}. \quad (2)$$

Thus, the process of normal sensor i obtaining a local quantized measurement data can be presented as

$$u_i(k)=g(h_i(\boldsymbol{\theta}(k))+n_i(k)),\ u_i(k)\in\mathcal{M}. \quad (3)$$

There are two types of data in the measurement process, namely analog sensing value and quantized data. Byzantine nodes may modify one type of data or both types of data using Byzantine data attack. The measurement process at a sensor is illustrated in Fig. 2, where functions $g(\bullet)$ and $\tilde{g}(\bullet)$ denote the mapping process at a normal sensor and a Byzantine node, respectively. Let $\tilde{u}_i(k)$ be the falsified measurement data for Byzantine node i at time k. From Fig. 2, Byzantine nodes first obtain the normal signal $s_i(k)$ as a normal sensor does, then falsify $s_i(k)$ to the falsified measurement data $\tilde{u}_i(k)$ using a Byzantine mapping process denoted as $\tilde{g}(\bullet)$. So, $\tilde{g}(\bullet)$ can be regarded as the cascade of the normal mapping function $g(\bullet)$ and the attack function $b(\bullet)$.Therefore, the measurement process at Byzantine node i can be denoted by a Byzantine mapping function $\tilde{g}(\bullet)$ as

$$\tilde{u}_i(k)=\tilde{g}(h_i(\boldsymbol{\theta}(k))+n_i(k))=b(g(h_i(\boldsymbol{\theta}(k))+n_i(k))),\ \tilde{u}_i(k)\in\mathcal{M}. \quad (4)$$

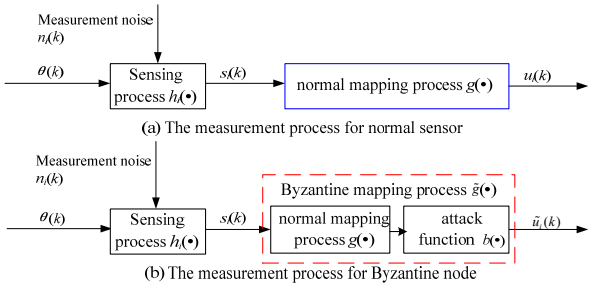


Figure 2. The measurement process for normal sensor and Byzantine node.

Let $P(u_i(k)|\boldsymbol{\theta}(k))$ be the conditional probability of $u_i(k)$ for normal senor i at time k. Then, the probability that normal senor i takes a specific value $m$at time k is

$$P(u_i(k)=m|\boldsymbol{\theta}(k))=\int_{\eta_{m-1}}^{\eta_m} p(s_i(k)|\boldsymbol{\theta}(k))ds_i(k), \quad (5)$$

where $p(s_i(k)|\boldsymbol{\theta}(k))$ is the conditional probability density function of the signal sensed by sensor i at time k.

It is assumed that the attack function $b(\bullet)$ adopted by Byzantine nodes is a probabilistic modification process. Let p_{ml} be the probability that Byzantine node i modifies $u_i(k)=m$ to $\tilde{u}_i(k)=l$, and $p_{ml}=\Pr(\tilde{u}_i(k)=l\,|\,u_i(k)=m)$. In this paper, it is assumed that $b(\bullet)$ is time-invariant. Therefore, the probability that Byzantine node i takes a specific value m is

$$P(\tilde{u}_i(k)=m|\boldsymbol{\theta}(k))=\sum_{l\in\mathcal{M}} p_{lm}P(u_i(k)=l|\boldsymbol{\theta}(k)). \quad (6)$$

2.3 Target Location Estimation Process

Let $v_i(k)$be the data received by the FC from sensor i at time k, and $v_i(k)\in\mathcal{M}$. For analysis simplicity, we assume that the reporting channel between sensors and the FC is error-free. If sensor i is a Byzantine node, $v_i(k)=\tilde{u}_i(k)$; if sensor i is normal, $v_i(k)=u_i(k)$.

Since α fraction of sensors are Byzantines and the FC is not aware of the type of sensors, it is assumed at the FC that sensor i is compromised with probability α. Therefore, the conditional distribution of $v_i(k)$ at the FC is

$$\begin{aligned}P(v_i(k)=m|\boldsymbol{\theta}(k))&=\alpha P(\tilde{u}_i(k)=m|\boldsymbol{\theta}(k))\\&\quad+(1-\alpha)P(u_i(k)=m|\boldsymbol{\theta}(k))\end{aligned} \quad (7)$$

Let $\boldsymbol{v}(k)$ be all the data received by the FC at time k, and $\boldsymbol{v}(k)\in\mathbb{R}^{N\times 1}$. $\hat{\boldsymbol{\theta}}(k)$ is the estimated target state at time k. FC estimates $\hat{\boldsymbol{\theta}}(k)$ using $\boldsymbol{v}(k)$ and $\hat{\boldsymbol{\theta}}(k-1)$ using a filtering algorithm.

Since the measurement process is highly nonlinear and the UKF has a good approximation accuracy dealing with the nonlinear function, the FC uses the UKF to implement the target tracking [3]. Due to the limited space, we omit the detail of the target tracking method here.

2.4 Performance Analysis

Sequential posterior Cramer-Rao lower bound (PCRLB) is regarded as the metric to evaluate the performance of distributed target tracking. The mean square error matrix at time k is bounded below by$\boldsymbol{J}^{-1}(k)$, and $\boldsymbol{J}^{-1}(k)\in\mathbb{R}^{4\times 4}$. Hence, we have $E\left[\left(\hat{\boldsymbol{\theta}}(k)-\boldsymbol{\theta}(k)\right)\left(\hat{\boldsymbol{\theta}}(k)-\boldsymbol{\theta}(k)\right)^T\right]\ge\boldsymbol{J}^{-1}(k)$. A recursive approach is presented in [14] to calculate $\boldsymbol{J}(k)$ as

$$\boldsymbol{J}(k)=\boldsymbol{D}^{22}(k)-\boldsymbol{D}^{21}(k)[\boldsymbol{J}(k-1)+\boldsymbol{D}^{11}(k)]^{-1}\boldsymbol{D}^{12}(k). \quad (8)$$

where,

$$\begin{aligned}\boldsymbol{D}^{11}(k)&=E\left\{-\nabla_{\boldsymbol{\theta}(k-1)}\nabla^T_{\boldsymbol{\theta}(k-1)}\log P(\boldsymbol{\theta}(k)|\boldsymbol{\theta}(k-1))\right\}\\&=E\left\{\left[\nabla_{\boldsymbol{\theta}(k-1)}\boldsymbol{f}^T(\boldsymbol{\theta}(k-1))\right]\mathbf{Q}^{-1}\left[\nabla_{\boldsymbol{\theta}(k-1)}\boldsymbol{f}^T(\boldsymbol{\theta}(k-1))\right]^T\right\},\end{aligned} \quad (9.a)$$

$$\begin{aligned}\boldsymbol{D}^{12}(k)&=E\left\{-\nabla_{\boldsymbol{\theta}(k-1)}\nabla^T_{\boldsymbol{\theta}(k)}\log P(\boldsymbol{\theta}(k)|\boldsymbol{\theta}(k-1))\right\}\\&=-E\left\{\left[\nabla_{\boldsymbol{\theta}(k-1)}\boldsymbol{f}^T(\boldsymbol{\theta}(k-1))\right]\mathbf{Q}^{-1}\right\},\end{aligned} \quad (9.b)$$

$$\begin{aligned}\boldsymbol{D}^{21}(k)&=E\left\{-\nabla_{\boldsymbol{\theta}(k)}\nabla^T_{\boldsymbol{\theta}(k-1)}\log P(\boldsymbol{\theta}(k)|\boldsymbol{\theta}(k-1))\right\}\\&=\left(\boldsymbol{D}^{12}(k)\right)^T,\end{aligned} \quad (9.c)$$

$$\begin{aligned}\boldsymbol{D}^{22}(k)&=E\left\{-\nabla_{\boldsymbol{\theta}(k)}\nabla^T_{\boldsymbol{\theta}(k)}\log P(\boldsymbol{\theta}(k)|\boldsymbol{\theta}(k-1))\right\}\\&\quad+E\left\{-\nabla_{\boldsymbol{\theta}(k)}\nabla^T_{\boldsymbol{\theta}(k)}\log P(\boldsymbol{v}(k)|\boldsymbol{\theta}(k))\right\}\\&=\mathbf{Q}^{-1}+\boldsymbol{D}^{22,\text{data}}(k),\end{aligned} \quad (9.d)$$

Where (9.a)-(9.d), $\nabla_{\boldsymbol{\theta}(k)}$ is a derivative operator, and $\nabla_{\boldsymbol{\theta}(k)}=\left[\frac{\partial}{\partial x_t(k)},\frac{\partial}{\partial y_t(k)},\frac{\partial}{\partial \dot{x}_t(k)},\frac{\partial}{\partial \dot{y}_t(k)}\right]^T$. The initial value of $\boldsymbol{J}(k)$ can be calculated using the prior distribution of the target state as $\boldsymbol{J}(0)=E\left\{-\nabla_{\boldsymbol{\theta}(0)}\nabla^T_{\boldsymbol{\theta}(0)}\log p_0(\boldsymbol{\theta}(0))\right\}=\bar{\boldsymbol{\Lambda}}_0^{-1}$.

From (9.a)-(9.d), one finds that $\boldsymbol{D}^{22,\text{data}}(k)$ is related to the received data at time k, and represents the information provided by the data received at time k. Other items of $\boldsymbol{J}(k)$ except $\boldsymbol{D}^{22,\text{data}}(k)$ are related to the target state evolution model, and represent the information provided by the target state evolution model.

From the view point of the network designer, the information obtained at the FC increases as $\boldsymbol{J}(k)$ increases, which means that the performance of distributed target tracking improves. On the other hand, from the attacker's perspective, the smaller $\boldsymbol{J}(k)$, the larger the damage caused by the Byzantine data attack. Therefore, in a distributed target tracking system under Byzantine data attacks, the network designer should improve $\boldsymbol{J}(k)$ as far as possible, while the attacker will try to make $\boldsymbol{J}(k)$ small.

3. OPTIMAL ATTACK STARTEGY

Since the target tracking is a sequential estimation process, both the target state evolution model and the reported data contribute information to the FC. From the perspective of the attacker, if the contributions of the target state evolution model and the received data are 0, the FC is incapable of tracking the target state correctly using the received data or/and the target state evolution model. That is, $\boldsymbol{J}(k)=\boldsymbol{0}$. This situation is regarded as the FC is perfectly blind here.

According to (8) and (9.a)-(9.d), $\boldsymbol{J}(k)=\boldsymbol{0}$ is equivalent to

$$\boldsymbol{D}^{22}(k)=\boldsymbol{D}^{21}(k)[\boldsymbol{J}(k-1)+\boldsymbol{D}^{11}(k)]^{-1}\boldsymbol{D}^{12}(k). \tag{10}$$

Suppose that Byzantine nodes launch Byzantine data attack in a probabilistic manner and the attack mode is time-invariant. The optimal Byzantine data attack strategy for distributed target tracking system is described in Theorem 1.

Theorem 1. Suppose the target state evolution model adopts the linear model in (1), and the attacker has no *a priori* knowledge about the target state evolution model. Byzantine nodes falsify the quantized measurement data in a probabilistic way in Section 2.3. The quantized data received by the FC is within $\mathcal{M}$, and the size of set $\mathcal{M}$ is M. In order to make FC perfectly blind, the minimum fraction of Byzantine nodes should be $\alpha_{\text{blind}}=\frac{M-1}{M(1-p_{mm})}$ and $p_{mm}<\frac{1}{M}$. Moreover, the condition that $p_{lm}=\frac{1-p_{mm}}{M-1}$, $l\in\mathcal{M}$ and $l\neq m$ should be satisfied as the FC is perfectly blind.

Proof: Since Byzantine nodes have no *a priori* knowledge about the target state evolution model, the attacker cannot calculate $\boldsymbol{J}(k-1)$ or purposely change the contribution of the target state evolution model $\boldsymbol{J}(k-1)$ at time k. Hence, we assume that the attacker always regards $\boldsymbol{J}(k-1)=\bar{\boldsymbol{\Lambda}}_0^{-1}$, and $\bar{\boldsymbol{\Lambda}}_0^{-1}$ is the initial value of $\boldsymbol{J}(k-1)$. Then, the condition $\boldsymbol{J}(k)=\boldsymbol{0}$ can be rewritten as

$$\boldsymbol{D}^{22}(k)=\boldsymbol{D}^{21}(k)[\bar{\boldsymbol{\Lambda}}_0^{-1}+\boldsymbol{D}^{11}(k)]^{-1}\boldsymbol{D}^{12}(k). \tag{11}$$

According to the target state evolution model in (1), (9.a), (9.b) and (9.c) can be rewritten as

$$\boldsymbol{D}^{11}(k)=\mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F}, \tag{12.a}$$

$$\boldsymbol{D}^{12}(k)=(\boldsymbol{D}^{21}(k))^T=-\mathbf{F}^T\mathbf{Q}^{-1}. \tag{12.b}$$

Substituting (12.a) and (12.b) into (11), we have

$$\begin{aligned}\boldsymbol{D}^{22,\text{data}}(k)&=\boldsymbol{D}^{21}(k)\left[\bar{\boldsymbol{\Lambda}}_0^{-1}+\boldsymbol{D}^{11}(k)\right]^{-1}\boldsymbol{D}^{12}(k)-\mathbf{Q}^{-1}\\&=\mathbf{Q}^{-1}\mathbf{F}\left[\mathbf{I}+\bar{\boldsymbol{\Lambda}}_0\mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F}\right]^{-1}\bar{\boldsymbol{\Lambda}}_0\mathbf{F}^T\mathbf{Q}^{-1}-\mathbf{Q}^{-1}.\end{aligned} \tag{13}$$

Since the variance of the initial distribution of the target should be large enough in order that the plane includes 99% confidence region of the target. Thus, we have $\mathbf{I}+\bar{\boldsymbol{\Lambda}}_0\mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F}\approx\bar{\boldsymbol{\Lambda}}_0\mathbf{F}^T\mathbf{Q}^{-1}\mathbf{F}$.

Therefore, (13) can be simplified as $\boldsymbol{D}^{22,\text{data}}(k)=\boldsymbol{0}$, and $\boldsymbol{D}^{22,\text{data}}(k)\in\mathbb{R}^{4\times4}$.

Let $[\boldsymbol{D}^{22,\text{data}}(k)]_{r,c}$ denotes the element in the rth row and the cth column of $\boldsymbol{D}^{22,\text{data}}(k)$, r=0,1,2,3 and c=0,1,2,3. Let $[\boldsymbol{\theta}(k)]_r$ and $[\boldsymbol{\theta}(k)]_c$ be the rth and the cth elements of the vector $\boldsymbol{\theta}(k)$, respectively. Hence, $[\boldsymbol{D}^{22,\text{data}}(k)]_{r,c}$ can be written as

$$\begin{aligned}\left[\boldsymbol{D}^{22,\text{data}}(k)\right]_{rc}=\int p_0(\boldsymbol{\theta}(k))\Bigg(\sum_{i=1}^{N}\sum_{m\in\mathcal{M}}\Bigg(&\frac{-1}{P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))}\\&\frac{\partial P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))}{\partial[\boldsymbol{\theta}(k)]_r}\frac{\partial P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))}{\partial[\boldsymbol{\theta}(k)]_c}\Bigg)\Bigg)d\boldsymbol{\theta}(k),\end{aligned} \tag{14}$$

where $p_0(\boldsymbol{\theta}(k))$ is calculated from $p_0(\boldsymbol{\theta}(0))$ and the target state evolution model. From (14), one finds that when $\boldsymbol{D}^{22,\text{data}}(k)=\boldsymbol{0}$, $\frac{\partial P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))}{\partial\boldsymbol{\theta}(k)}=\boldsymbol{0}$ since $\frac{-1}{P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))}\neq0$. That is, $P(v_i(k)=m|\boldsymbol{\theta}(k))$ is independent of the target state, and $P(v_i(k)=m|\boldsymbol{\theta}(k))$ is a constant for all $m\in\mathcal{M}$.

The conditional probability that the local measurement data for a Byzantine node takes a specific value m can be rewritten as

$$P(\tilde{u}_i(k)=m\,|\,\boldsymbol{\theta}(k))=p_{mm}+\sum_{\substack{l\in\mathcal{M}\\l\neq m}}(p_{lm}-p_{mm})P(u_i(k)=l\,|\,\boldsymbol{\theta}(k)). \tag{15}$$

And the conditional probability of the quantized data received by the FC be m can be represented as

$$\begin{aligned}&P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))\\&=\alpha P(\tilde{u}_i(k)=m\,|\,\boldsymbol{\theta}(k))+(1-\alpha)P(u_i(k)=m\,|\,\boldsymbol{\theta}(k))\\&=1-\alpha+\alpha p_{mm}\\&\quad+\sum_{\substack{l\in\mathcal{M}\\l\neq m}}\left[\alpha(p_{lm}-p_{mm})-(1-\alpha)\right]P(u_i(k)=l\,|\,\boldsymbol{\theta}(k)).\end{aligned} \tag{16}$$

As mentioned above, $P(v_i(k)=m|\boldsymbol{\theta}(k))$ is not related to the target state, and is a constant for all $m\in\mathcal{M}$, while $P(u_i(k)=m|\boldsymbol{\theta}(k))$ depends on the target state. Therefore, we have $\alpha(p_{lm}-p_{mm})-(1-\alpha)=0$ for all $l\in\mathcal{M}$, $l\neq m$, and $P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))=1-\alpha+\alpha p_{mm}$. For $\sum_{m\in\mathcal{M}}P(v_i(k)=m\,|\,\boldsymbol{\theta}(k))=1$, we have $1-\alpha+\alpha p_{mm}=1/M$. Substituting $1-\alpha+\alpha p_{mm}=1/M$ into $\alpha(p_{lm}-p_{mm})-(1-\alpha)=0$, we have $\alpha_{\text{blind}}=\frac{M-1}{M(1-p_{mm})}$, $p_{lm}=\frac{1-p_{mm}}{M-1}$ and $l\neq m$.

In a WSN, it is impossible for the attacker to capture all of the sensors to blind the FC, and $\alpha_{\text{blind}}<1$. Hence, we have $p_{mm}<1/M$.

Therefore, in order to make FC perfectly blind, the minimum

Fraction of Byzantine nodes should be $\alpha_{\text{blind}}=\frac{M-1}{M(1-p_{mm})}$, and the optimal attack strategy is $p_{mm}<\frac{1}{M}$, $p_{lm}=\frac{1-p_{mm}}{M-1}$, $l\in\mathcal{M}$ and $l\neq m$.■

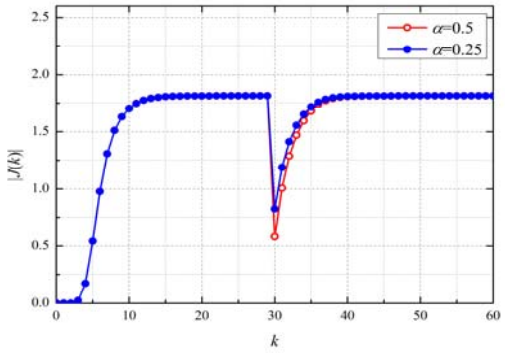
In this way, the FC is incapable of obtaining any information at time k. From (8), one finds that the information contributed by the target state evolution model will become 0 after some iterations.

4. NUMERICAL RESULTS

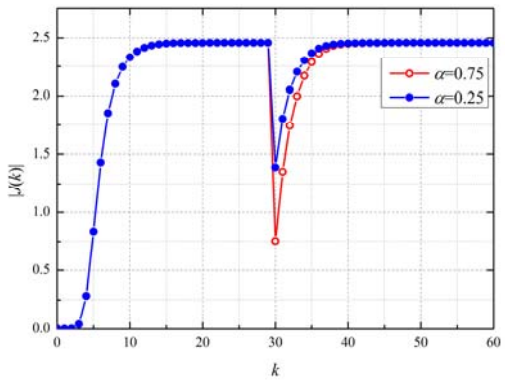
In this section, the impact of Byzantine data attack on the performance of distributed target tracking is evaluated.

Considering a WSN with 36 sensor nodes uniformly deployed in a 200m×200m area. In the measurement process, the received signal strength is used as the sensed signal of the target. The signal power received at sensor i is $(a_i)^2=P_0(d_0/d_i)^2$, where d_i is the distance between sensor i and the target. We assume that d_0=1, P_0=25000, $\bar{\boldsymbol{\theta}}_0$ =[−80 2 −80 2]^T, $\bar{\Lambda}_0$ =*diag*([100 0.25 100 0.25]), and ε=0.16. The measurement noise at each sensor follows Gaussian distribution with zero-mean and variance σ^2 =1.

$|\boldsymbol{J}(k)|$ denotes the information provided by the prior knowledge and the reported data from time 1 to time k. Fig. 3 shows the impact of one-time Byzantine data attack on the performance of distributed target tracking in terms of $|\boldsymbol{J}(k)|$, where Byzantine nodes launch attack at k=30. Fig. 3(a) and 3(b) correspond the value of $|\boldsymbol{J}(k)|$ as M=2 ($\boldsymbol{\eta}$=[−inf, 2.98, inf]) and M=4 ($\boldsymbol{\eta}$=[−inf, 1.61, 2.03, 2.82, inf]), respectively.



(a).*M*=2



(b). *M*=4

Figure 3. The impact of Byzantine data attack launched at *k*=30 on the performance of target tracking.

From Fig. 3, one finds that when Byzantines launch attack at k=30, the information obtained at the FC reduces sharply. If Byzantines do not launch attack from k=31, the information obtained at the FC increases gradually and is back to normal after about 10 iterations (that is the suffering period). Moreover, we observe that the suffering period is not related to α or/and M, while α and M affect the degradation degree of $|\boldsymbol{J}(k)|$ during the suffering period.

The red line in Fig. 3 means that the reported data do not provide any information at k=30. The blue line in Fig. 3 means that the reported data provide some information to the FC at k=30. Therefore, the more sensors captured by the attack, the more damage caused by Byzantine data attack.

Furthermore, from Figs. 3(a) and 3(b), we observe that the FC obtains more information as M=4. Therefore, the network designer can improve the performance of the target tracking by increasing the number of quantization levels.

From Fig. 3, one finds that launching a Byzantine data attack only degrades the performance of distributed target tracking for a while. Without *a priori* knowledge about the target state evolution model, the attacker should continuously launch Byzantine data attack to make the FC blind completely.

Fig. 4 shows the impact of Byzantine data attack on the performance of distributed target tracking in terms of $|\boldsymbol{J}(k)|$, where the attacker tries to blind the FC by adjusting the values of α and p_{mm}. In Fig. 4, the total time duration is 100, which is divided into 5 parts corresponding to 5 cases. Case 1: From k=1 to 25, α=0 and p_{mm}=1. Case 2: From k=26 to 44, α=0.3 and p_{mm}=0.5. Case 3: From k=45 to 63, α=0.5 and p_{mm}=0.5. Case 4: From k=64 to 83, α=0.5 and p_{mm}=0. Case 5: From k=84 to 100, α=0.75 and p_{mm}=0.

From Fig. 4, we observe that the information obtained at the FC reduces as α increases and p_{mm} decreases, which means the damage introduced by the attacker increases. As α reaches $\alpha_{\text{blind}} = \frac{M-1}{M(1-p_{mm})}$, the FC is blind completely and can obtain no information from the reported data or the prior knowledge about the target state evolution model. For example, as M=2, if α=0.3 and p_{mm}=0.5, $|\boldsymbol{J}(k)|$=0.40059; if α=0.5 and p_{mm}=0.5, $|\boldsymbol{J}(k)|$=0.09746; if α=0.5 and p_{mm}=0, $|\boldsymbol{J}(k)|$=4.59E-6. Hence, the FC is made blind in case 4. Therefore, we conclude that the attacker can cause more damage by increasing α and decreasing p_{mm}.

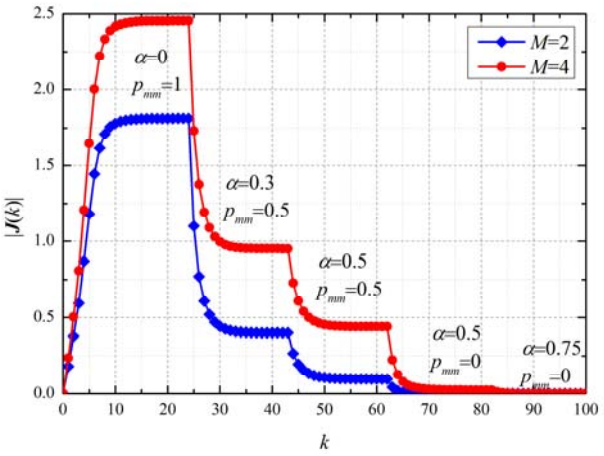


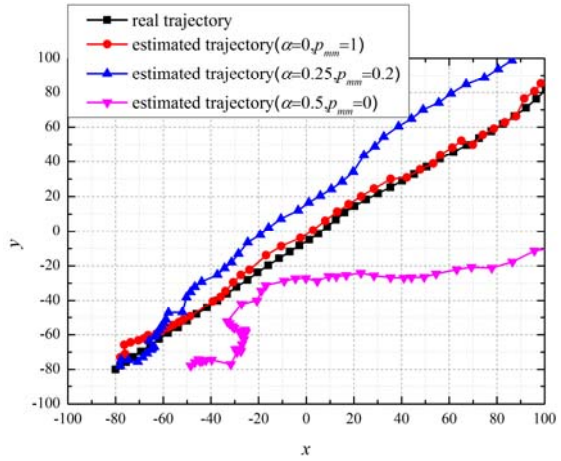
Figure 4. The impact of Byzantine data attack on the performance of distributed target tracking in terms of $|\boldsymbol{J}(k)|$.

Fig. 5 shows the impact of Byzantine data attack on the performance of distributed target tracking in terms of the estimated target trajectory, where Fig. 5(a) and 5(b) correspond the estimated target trajectory as M=2 and M=4, respectively.

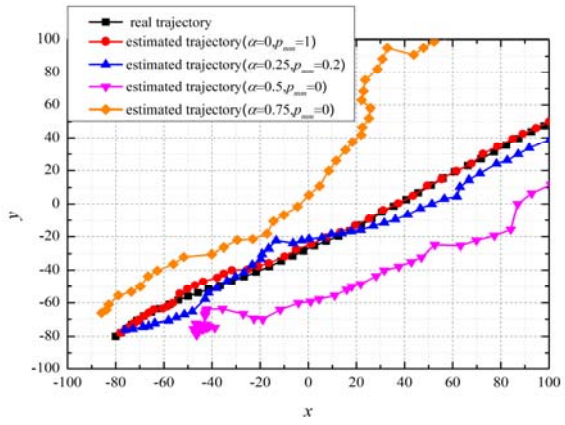
From Fig. 5, we observe that as α increases and p_{mm} decreases, the gap between the estimated target trajectory and the real trajectory broadens. Moreover, when the FC is blind, the estimated target trajectory diverges completely.

Furthermore, from Fig. 5(a) and 5(b), we observe that with same

Byzantine data attack parameters, the estimated target trajectory at M=4 is more precise than that at M=2. Therefore, the network designer can alleviate the impact of Byzantine data attack and improve the estimation performance at the price of the communication overhead of the network.



(a). M=2



(b). M=4

Figure 5.The impact of Byzantine data attack on the performance of distributed target tracking in terms of the estimated target trajectory.

5. CONCLUSION

In this paper, we studied the effect of Byzantine data attack on the performance of distributed target tracking with M-ary quantized data. We define a mapping process implemented at Byzantine nodes and characterize the impact of the Byzantine data attack on the performance of distributed target tracking using sequential PCRLB. By making the FC incapable of obtaining information from the reported data and the target state evolution model, we found the optimal Byzantine data attack strategy for the distributed target tracking. Simulation results validate that the optimal Byzantine data attack strategy can make the FC blind completely. Furthermore, the security performance of the network increases as the number of quantization levels increases. Hence, a trade-off should be achieved by the FC between the security performance and the communication cost.

6. ACKNOWLEDGMENTS

This work was partly supported by National Natural Science Foundation of China (No. 61471318, No. 61671410), the Zhejiang Provincial Natural Science Foundation of China (No. LY14F010014), and the Fundamental Research Funds for the Central Universities.

7. REFERENCES

[1] Souza, E. L., Nakamura, E. F., and Pazzi, R. W., Target Tracking for Sensor Networks: A Survey,*Acm Computing Surveys*, vol. 49, no. 2, pp. 1-31, Nov. 2016.

[2] Huang, X., Zhan, J., Zhang, Y., Technology research of ultra-tightly integration about INS aided tracking loop based on EKF,*International Conference on Intelligent Information Processing*, Wuhan, China ,pp. 1-6, Dec. 2016.

[3] Zhan, R. and Wan, J., Iterated unscented Kalman filter for passive target tracking, *IEEE Trans. on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 1155-1163, Jul. 2007.

[4] Hong, K., Medeiros, H., Shin, P. J., Park, J., Resource-aware distributed particle filtering for cluster-based object tracking in wireless camera networks,*Dissertations & Theses - Gradworks*, vol. 21, no. 3, pp. 137-156, 2016.

[5] Prasov, A. A., and Khalil, H. K.,Tracking performance of a high-gain observer in the presence of measurement noise, *International Journal of Adaptive Control and Signal Processing*, vol. 30, no. 8, pp. 1228-1243, Aug. 2016.

[6] Zheng, Y., Cao, N., Wimalajeewa, T., and Varshney, P. K., Compressive sensing based probabilistic sensor management for target tracking in wireless sensor networks, *IEEE Trans. on Signal Processing*, vol. 63, no. 22, pp. 6049-6060, Nov. 2015.

[7] Cao, N., Brahma, S., and Varshney, P. K.,Target tracking via crowdsourcing: A mechanism design approach, *IEEE Trans. on Signal Processing*,vol.63,no. 6, pp. 1464-1476, Mar. 2015.

[8] Guo, J., Yuan, X., and Han, C., Bias change detection-based sensor selection approach for target tracking in large-scale distributed sensor networks, *IET Radar, Sonar & Navigation*, vol. 11, no. 1, pp. 30-39, Jan. 2015.

[9] Oracevic,A. and Ozdemir, S., Secure and Reliable Prediction Based Target Tracking for Wireless Sensor Networks, *International Conference on Intelligent Systems, Modelling and Simulation*, Taipei, Taiwan, pp. 646-651, Nov. 2015.

[10] Imran, S., Ko, Y. B., A Continuous Object Boundary Detection and Tracking Scheme for Failure-Prone Sensor Networks. *Sensors*, vol. 17, no. 2, pp. 1-17, Feb. 2017.

[11] Oracevic, A., Akbaş, S., Ozdemir, S., and KosSecure, M., Target detection and tracking in mission critical wireless sensor networks, *2014 International Conference on Anti-counterfeiting, Security, and Identification*, Macao, China, pp. 1-5, Dec. 2014.

[12] Vempaty, A., Ozdemir, O., and Varshney, P. K., Target tracking inwireless sensor networks in the presence of Byzantines, *The 16th Internatinoal Conference on Information Fusion*, Istanbul,Turkey, pp. 968-973, Mar. 2013.

[13] Li, X., and Jilkov, V. P., Survey of maneuvering target tracking Part I: Dynamic models, *IEEE Trans. on Aerospace and Electronic Systems*, vol. 39,no. 4, pp. 1333-1364, Oct. 2003.

[14] Tichavsk´y, P., Muravchik, C. H., and Nehorai, A., Posterior Cram´er-Rao bounds for discrete-time nonlinear fltering, *IEEE Trans. on Signal Processing*, vol. 46, pp. 1386-1396, May 1998.