

GPS-based Tracking over WSNs with Delayed and Missing Data using UFIR Filtering

Karen Uribe-Murcia^{1,*}, Yuriy S. Shmaliy^{1,**}, and Amparo Andrade-Lucio^{1,***}

¹Department of Electronics Engineering, Universidad de Guanajuato

Abstract. Vehicles tracking is organized to increase safety in smart cities by localizing cars using the Global Positioning System (GPS). The GPS-based system provides accurate tracking, but is also required to be reliable and robust. As a main estimator, we propose using the unbiased finite impulse response (UFIR) filter, which meets these needs as a more robust alternative to the Kalman filter (KF). The UFIR filter is developed for vehicle tracking in discrete-time state-space over wireless sensor networks (WSNs) with time-stamped discretely delayed on k -step-lags and missing data. The state-space model is represented in a way such that the UFIR filter, KF, and H_∞ filter can be used universally. Applications are given for measurement data, which are cooperatively transferred from a vehicle to a central station through several nodes with k -step-lags. Better tracking performance of the UFIR filter is shown experimentally.

1 Introduction

Accurate target tracking is one of the key problems in urban areas [1]. If a target is equipped with the Global Positioning System (GPS) tracker, then measurement data can be transferred to a central station through one or several nodes of a wireless sensor network (WSN) [2]. It is typically accompanied with information latency and missing data [3] due to many reasons [4, 5]. In networks, communication delays go along with data loss called dropout or intermittence [6]. Also, data delays cause the out-of-sequence problem [4]. Two basic models have been created for delayed data. The delays becomes known when data are time-stamped [7]. Otherwise, the delays are considered to be random [8] and the problem complicates by uncertainties [9].

The Kalman and H_∞ state estimators are most widely used to deal with latency and associated issues [10, 11]. The linear Kalman filter (KF) is optimal when it matches the system perfectly, noise is white Gaussian and uncorrelated, and the noise statistics are known along with the initial values. When such conditions are not obeyed, the KF may demonstrate poor performance. The robust H_∞ filter bounds the mean square error (MSE) for admissible parameter perturbations and delays [9], which allows minimizing errors with less information required than for the noise statistics [10].

Another way to achieve better robustness is to process most recent finite data [12] using finite impulse response (FIR) filters [13]. Such filters have been developed during decades by many authors in signal processing [14–18] and control [19–21]. However, only a few authors pro-

posed FIR solutions to models with delays [22–25]. Note that the unbiased FIR (UFIR) filter [15, 26, 27] is most robust among other FIR solutions owing to an ability to ignore the noise statistics and initial values. This filter is bounded-input bounded-output (BIBO) stable and blind on given horizons of N points, but is still not developed for observations with delayed and missing data. In this paper, we develop the UFIR filter for GPS-based vehicle tracking over WSNs with time-stamped discretely delayed and missing data.

2 Tracking by Kalman and H_∞ Filters

Consider a typical scenario of GPS-based vehicle tracking in WSNs as shown in Fig. 1. The vehicle current coordi-

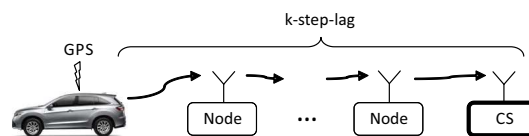


Figure 1. Transferring the time-stamped vehicle coordinates measured by a GPS tracker to a central station (CS) via several nodes of a WSN with a k -step-lag latency.

nates are measured by the GPS tracker. The time-stamped data are transferred to a central station (CS) via one or several nodes of the WSN. Because the nodes may discretely delay data, the vehicle location is observed in CS with a time-varying k_n -step-lag depending on the vehicle location and interaction with the WSN.

* e-mail: karen.uribe15@gmail.com

** e-mail: shmaliy@ugto.mx

*** e-mail: andrade@ugto.mx

For $k_n \geq 0$, the vehicle and its observation can be represented in discrete-time state space as

$$x_n = Fx_{n-1} + w_n, \quad (1)$$

$$y_n = Hx_{n-k_n} + v_n, \quad (2)$$

where n is the discrete-time index, $x_n \in \mathbb{R}^K$, $y_n \in \mathbb{R}^M$, $F \in \mathbb{R}^{K \times K}$, and $H \in \mathbb{R}^{M \times K}$. All data are time-stamped, so that k_n is known at each n and x_{n-1} is also supposed to be known. The uncorrelated noise vectors, $w_n \in \mathbb{R}^K$ and $v_n \in \mathbb{R}^M$, are white Gaussian with known covariances, $Q = E\{w_n w_n^T\}$ and $R = E\{v_n v_n^T\}$, and the property $E\{w_n v_q^T\} = 0$ for all n and q . We will also use the following auxiliary time index

$$p_\nu = \nu - k_\nu + 1, \quad (3)$$

in which ν is integer.

The UFIR filter can be applied if to transform model (1)–(2) to have no latency if to represent x_{n-k_n} using (1) via x_n as

$$x_{n-k_n} = F^{-k_n} \left(x_n - \sum_{i=0}^{k_n-1} F^i w_{n-i} \right) \quad (4)$$

and then substitute (4) into (2) and arrive at

$$y_n = \bar{H}_n x_n + \bar{v}_n \quad (5)$$

where

$$\bar{H}_n = HF^{-k_n}, \quad (6)$$

$$\bar{v}_n = v_n - H \sum_{i=0}^{k_n-1} F^{-k_n+i} w_{n-i}, \quad (7)$$

and the covariance $\bar{R}_n = E\{\bar{v}_n \bar{v}_n^T\}$ of \bar{v}_n is given by

$$\bar{R}_n = R + \bar{H}_n \sum_{i=0}^{k_n-1} F^i Q F^{iT} \bar{H}_n^T. \quad (8)$$

In compact matrix forms, (7) and (8) can be represented as

$$\bar{v}_n = v_n - H \bar{B}_n W_{p_n, n}, \quad (9)$$

$$\bar{R}_n = R + H \bar{B}_n \bar{Q}_n \bar{B}_n^T H^T, \quad (10)$$

where

$$\bar{B}_n = \begin{bmatrix} F^{-1} F^{-2} & \dots & F^{-k_n} \end{bmatrix}, \quad (11)$$

$$W_{p_n, n} = \begin{bmatrix} w_{p_n}^T & w_{p_n+1}^T & \dots & w_n^T \end{bmatrix}^T, \quad (12)$$

and $\bar{Q}_n = \text{diag}[Q \ Q \ \dots \ Q]$ has k_n diagonal components. Here, $\bar{B}_n = 0$ and $W_{p_n, n} = 0$ when $k_n = 0$ and $p_n > n$. Any standard estimation technique can now be applied to model (1) and (5). However, the KF and H_∞ filter were most developed for data with latency.

2.1 Kalman Filtering Estimate

We will exploit an alternative form of the KF algorithm given in [28]. This algorithms starts with the prior error covariance matrix

$$P_n^- = FP_{n-1}^- F^T + Q \quad (13)$$

and then recursively updates the following values as

$$P_n = (P_n^-)^{-1} + \bar{H}_n^T \bar{R}_n^{-1} \bar{H}_n, \quad (14)$$

$$K_n^{\text{KF}} = P_n^{-1} \bar{H}_n^T \bar{R}_n^{-1}, \quad (15)$$

$$\hat{x}_n = F \hat{x}_{n-1} + K_n^{\text{KF}} (y_n - \bar{H}_n F \hat{x}_{n-1}), \quad (16)$$

$$P_{n+1}^- = FP_n^{-1} F^T + Q, \quad (17)$$

where \bar{H}_n is given by (6) and \bar{R}_n by (8) for any $k_n \geq 0$.

2.2 H_∞ Filtering Estimate

The H_∞ filter has been derived in [28] in the form (14)–(17) of the KF using the game theory. For $k_n \geq 0$, the H_∞ filtering algorithm becomes

$$P_n = (\check{P}_n^-)^{-1} - \theta_n S_n + \bar{H}_n^T \check{R}_n^{-1} \bar{H}_n, \quad (18)$$

$$K_n^\infty = P_n^{-1} \bar{H}_n^T \check{R}_n^{-1}, \quad (19)$$

$$\hat{x}_n = F \hat{x}_{n-1} + K_n^\infty (y_n - \bar{H}_n F \hat{x}_{n-1}), \quad (20)$$

$$\check{P}_{n+1}^- = FP_n^{-1} F^T + \check{Q}, \quad (21)$$

where the user-given symmetric positive definite matrices P_0 , \check{Q} , and \check{R} have different meaning than in the KF and \check{P}_n^- can be computed via P_0 using (13) with $Q = \check{Q}$. To keep (18) positive definite, the positive definite matrix $S_n \in \mathbb{R}^{K \times K}$ is subject to

$$(\check{P}_n^-)^{-1} - \theta_n S_n + \bar{H}_n^T \check{R}_n^{-1} \bar{H}_n > 0. \quad (22)$$

If equal weights are required for all errors, matrix S_n must be set identity, $S_n = I$. The performance criterion for this filter is $J_n < 1/\theta_n$, in which a scalar θ_n must be small enough in order for the filter to be efficient. It then follows that the tuning factor θ_n is not allowed to be negative, even though its negative values may reduce errors when the weighting matrices are not maximized. For Gaussian noise, zero θ_n transforms the H_∞ filter to the KF. For any other noise, small $\theta_n > 0$ may result in better robustness.

3 UFIR Filtering for Delayed and Missing Data

To develop the UFIR filter for $k_n > 0$, we extend model (1) and (5) on a horizon $[m, n]$ of N points, from $m = n - N + 1$ to n , that, referring to [27], yields

$$X_{m,n} = A_N x_m + B_N W_{m,n}, \quad (23)$$

$$Y_{m,n} = C_{m,n} x_m + G_{m,n} W_{m,n} + V_{m,n} \quad (24)$$

with the following extended vectors and matrices,

$$X_{m,n} = \begin{bmatrix} x_m^T & x_{m+1}^T & \dots & x_n^T \end{bmatrix}^T, \quad (25)$$

$$Y_{m,n} = \begin{bmatrix} y_m^T & y_{m+1}^T & \dots & y_n^T \end{bmatrix}^T, \quad (26)$$

$$V_{m,n} = \begin{bmatrix} v_m - C \bar{B}_n W_{p_m, m} \\ v_{m+1} - C \bar{B}_n W_{p_{m+1}, m+1} \\ \vdots \\ v_{n-1} - C \bar{B}_n W_{p_{n-1}, n-1} \\ v_n - C \bar{B}_n W_{p_n, n} \end{bmatrix}, \quad (27)$$

$$A_N = \begin{bmatrix} I & F^T & \dots & F^{N-1T} \end{bmatrix}^T, \quad (28)$$

$$B_N = \begin{bmatrix} I & 0 & \dots & 0 & 0 \\ F & I & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ F^{N-2} & F^{N-3} & \dots & I & 0 \\ F^{N-1} & F^{N-2} & \dots & F & I \end{bmatrix}, \quad (29)$$

$$C_{m,n} \triangleq C_{m,n}(k) = \bar{C}_{m,n}(k)A_N, \quad (30)$$

$$G_{m,n} \triangleq G_{m,n}(k) = \bar{C}_{m,n}(k)B_N, \quad (31)$$

$$\bar{C}_{m,n}(k) = \text{diag} \left[\underbrace{CF^{-k_m} \quad CF^{-k_{m+1}} \quad \dots \quad CF^{-k_n}}_N \right], \quad (32)$$

where k represents a set of $\{k_m, k_{m+1}, \dots, k_n\}$.

3.1 Batch UFIR Filter Form

The UFIR estimate $\hat{x}_n \triangleq \hat{x}_{n|n}$ of a vehicle state x_n can be obtained at n in the batch form as [19]

$$\hat{x}_n = \mathcal{H}_{m,n} Y_{m,n}, \quad (33)$$

where $\mathcal{H}_{m,n}$ is the UFIR filter gain and $Y_{m,n}$ is a real data vector (26), if to satisfy the unbiasedness condition

$$E\{x_n\} = E\{\hat{x}_n\}, \quad (34)$$

in which $E\{z\}$ means averaging of z and x_n can be represented with the last row vector in (23) as

$$x_n = F^{N-1} x_m + B_N^{(N)} W_{m,n}, \quad (35)$$

where $B_N^{(N)}$ is the N th row vector in (29) given by

$$B_N^{(N)} = [F^{N-1} \quad F^{N-2} \quad \dots \quad F \quad I]. \quad (36)$$

By combining (33)–(36) following [14], one arrives at the unbiasedness constraint

$$I = \mathcal{H}_{m,n} C_{m,n}, \quad (37)$$

in which

$$C_{m,n} \triangleq C_{m,n}(k) = \begin{bmatrix} CF^{-N+1-k_m} \\ \vdots \\ CF^{-1-k_{n-1}} \\ CF^{-k_n} \end{bmatrix}, \quad (38)$$

multiply the both sides of (37) with $(C_{m,n}^T C_{m,n})^{-1} C_{m,n}^T C_{m,n}$, come up with the UFIR filter gain

$$\mathcal{H}_{m,n} = (C_{m,n}^T C_{m,n})^{-1} C_{m,n}^T, \quad (39)$$

and provide the batch UFIR filtering estimate as

$$\hat{x}_n = (C_{m,n}^T C_{m,n})^{-1} C_{m,n}^T Y_{m,n}. \quad (40a)$$

$$= G_n C_{m,n}^T Y_{m,n}, \quad (40b)$$

where the *generalized noise power gain* (GNPG) [27] is

$$G_n = \mathcal{H}_{m,n} \mathcal{H}_{m,n}^T = (C_{m,n}^T C_{m,n})^{-1}. \quad (41)$$

The batch form (40a) may not suite the real-time tracking problem and we go on with its fast iterative algorithm.

3.2 Iterative UFIR Filter Form

Provided (1) and (5), the iterative UFIR filtering algorithm [13] can be applied straightforwardly, if to substitute matrix H with H given by (6). The UFIR filtering estimate (40a) can then be computed iteratively using recursions

$$G_l = [\bar{H}_l^T \bar{H}_l + (F G_{l-1} F^T)^{-1}]^{-1}, \quad (42)$$

$$\hat{x}_l = F \hat{x}_{l-1} + G_l \bar{H}_l^T (y_l - \bar{H}_l F \hat{x}_{l-1}), \quad (43)$$

beginning with $l = m + K$ and ending when $l = n$. The initial values for (42) and (43) are obtained at $s = m + K - 1$ in the batch forms as

$$G_s = (C_{m,s}^T C_{m,s})^{-1}, \quad (44)$$

$$\hat{x}_s = G_s C_{m,s}^T Y_{m,s}, \quad (45)$$

where $Y_{m,s}$ is a vector (26) of real data. When some data are lost, inaccurate, or unavailable, y_n can be predicted as $y_n = HF \hat{x}_{n-1}$, in which case first data on the horizon $[0, N - 1]$ must be available.

4 Tracking Errors Caused by Latency

Error produced by the UFIR tracker can be defined as $\epsilon_n = x_n - \hat{x}_n$, where x_n is specified by (35). Provided ϵ_n , the error covariance matrix $P_n = E\{\epsilon_n \epsilon_n^T\}$ can also be represented in the batch form as

$$P_n = [B_N^{(N)} - \mathcal{H}_{m,n} G_{m,n}] \bar{Q}_N [B_{m,n}^{(N)} - H_{m,n} G_{m,n}]^T + H_{m,n} \bar{R}_N H_{m,n}^T, \quad (46)$$

where $\bar{Q}_N = \text{diag}[Q \quad Q \quad \dots \quad Q]$ and $\bar{R}_N = \text{diag}[R \quad R \quad \dots \quad R]$ are square matrices with N nonzero diagonal elements. It can be shown that the deterministic case of $\bar{Q}_N = 0$ and $\bar{R}_N = 0$ makes $P_n = 0$ and the UFIR tracker has thus the *deadbeat property*.

4.1 Iterative Computation of P_n

Matrix (46) can be computed iteratively, if to substitute x_n with (1) and \hat{x}_n with (43). Providing the averaging, a recursion for (46) can be found as

$$P_l = (I - G_l \bar{H}_l^T \bar{H}_l) P_l^- (\dots)^T + G_l \bar{H}_l^T \bar{H}_l \times \left(\sum_{i=0}^{k_l-1} F^i Q F^{iT} \right) \bar{H}_l^T \bar{H}_l G_l + G_l \bar{H}_l^T R \bar{H}_l G_l, \quad (47)$$

where P_l^- is given by (13), l ranges as in (42) and (43), and P_n is taken when $l = n$. Recursion (46) suggests that the tracking error grows with k_n , because the sum containing Q grows with k_n . However, the same cannot be said about R , which does not accumulate the effect of k_n .

With no latency, the sum in (47) becomes identically zero and one arrives at the error covariance P_l valid for the standard UFIR filter [27],

$$P_l = (I - G_l H^T H) P_l^- (\dots)^T + G_l H^T R H G_l, \quad (48)$$

which also holds for the KF, if to assign $K_n = G_n H^T$.

Note that the minimization of tracking errors will require an optimal number $N_{\text{opt}} - 1$ of iterations for the UFIR filter, where the optimal horizon N_{opt} can be found as shown in [16].

5 GPS-Based Tracking of a Moving Vehicle

Let us consider the case shown in Fig. 1 when the GPS tracker measures the vehicle coordinates of location at each time-index n and transfer time-stamped data with constant $k = k_n > 0$. We will base our investigations on data obtained in the Cook county of Illinois and available for free use from the University of Illinois at Chicago with the start point in Fig. 2 placed at zero.

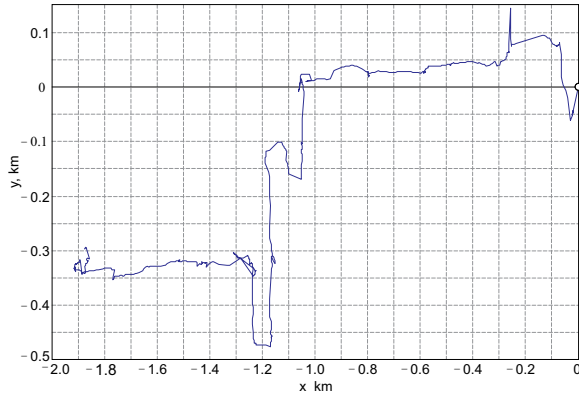


Figure 2. GPS-based vehicle trajectory measured in the north (y) and east (x) coordinates, both in km, with the start point at $(0, 0)$. Measurements are provided each second at 858 data points.

5.1 State-Space Model

We suppose that a vehicle is represented with two states in each directions and assign the state vector, $K = 4$, as $x_n = [x_{1n} \ x_{2n} \ x_{3n} \ x_{4n}]^T$, where $x_{1n} = x_n$, $x_{2n} = \dot{x}_n$, $x_{3n} = y_n$, and $x_{4n} = \dot{y}_n$ that leads to the system matrix of

$$F = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

where $\tau = 1$ s for the considered database. The measurement matrix for the GPS tracker is

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and matrix $C_{m,s}$ required by the UFIR filter to compute the initial values (44) and (45) for $s = m + 3$ becomes

$$C_{s,m} = \begin{bmatrix} 1 & -3\tau & 0 & 0 \\ 0 & 0 & 1 & -3\tau \\ 1 & -2\tau & 0 & 0 \\ 0 & 0 & 1 & -2\tau \\ 1 & -\tau & 0 & 0 \\ 0 & 0 & 1 & -\tau \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} F^{-k}.$$

The only tuning factor $N_{\text{opt}} = 5$ required by the UFIR filter was found for $k = 0$ by minimizing the derivative of the trace of the mean square value of the residual $y_n -$

$H\hat{x}_n(N)$ [16]. Because N_{opt} depends on k , we will apply $N_{\text{opt}} = 5$ in the worst case for the UFIR filter.

Having no information about the process noise, we observe similar trajectories and estimate the average vehicle speed by about 10 m/s or 36 km/hour. Next, accepting the speed standard deviation of about 20%, we set $\sigma_{w2} = 2$ m/s to the second state along each of the coordinates, ignore the unknown noise in the first state, $\sigma_{w1} = 0$, and describe matrix Q as

$$Q = \sigma_{w2}^2 \begin{bmatrix} \tau^2/2 & \tau/2 & 0 & 0 \\ \tau/2 & 1 & 0 & 0 \\ 0 & 0 & \tau^2/2 & \tau/2 \\ 0 & 0 & \tau/2 & 1 \end{bmatrix}.$$

The GPS standard positioning service provides navigation with an error of less than 15 meters with the probability of 95% in the 2-sigma sense. Referring to this value, we assign the standard deviation of the measurement noise in each direction as $\sigma_v = 15/4 = 3.75$ m and obtain

$$R = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_v^2 \end{bmatrix}.$$

Because the above provided matrices Q and R are over-estimated, we set $\tilde{Q} = Q$ and $\tilde{R} = R$ for the H_∞ . It is expected that the tuning factor θ will improve the performance of the H_∞ filter by minimizing the MSE for the maximized errors when the ground truth is available.

5.2 Effect of Latency on the Estimation Accuracy

We start with learning the effect of k on the estimation accuracy, which is illustrated in Fig. 3 with the root MSEs (RMSEs) in the north direction (y -RMSE). The KF is self-

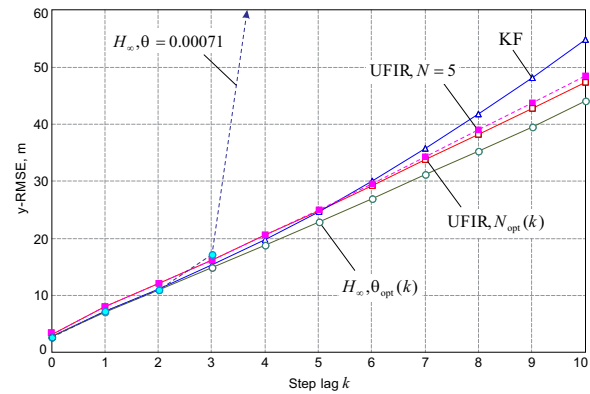


Figure 3. Effect of the k -step-lag on the y -RMSE of the UFIR filter, KF, and H_∞ filter with different tunings.

tuned to k and we consider its RMSE as a benchmark. The RMSE of the KF grows with k nonlinearly and faster than in the UFIR and H_∞ filters. The UFIR filter produces a bit more errors than in the KF with small k and lesser with larger k . A special feature is that the UFIR estimate is low sensitivity to N , which optimal value $N_{\text{opt}} = 5$ holds for $0 \leq k \leq 4$, increases to $N_{\text{opt}} = 6$ for $5 \leq k \leq 6$, and reaches $N_{\text{opt}} = 7$ when $7 \leq k \leq 10$. Note that setting N optimally

for each k does not improve the performance essentially against the worst case when $N_{opt} = 5$ is set for all k .

It θ is set properly for each lag k , the H_∞ filter outperforms both the UFIR filter and KF. However, this filter is highly sensitive to θ , which optimal value θ_{opt} ranges from 1.8×10^{-2} for $k = 0$ to 1.066×10^{-5} for $k = 10$ in a nonlinear way. Unlike in the UFIR filter, $\theta = \text{const}$ is unacceptable for all k . An example is given in Fig. 2, where $\theta_{opt} = 7.1 \times 10^{-4}$ found for $k = 2$ is applied in a wide range of k . It is only when $k = 2$ that the H_∞ filter improves the KF performance. For $k < 2$, there is no improvement and, when $k > 2$, the H_∞ filter rapidly diverges.

5.3 Tracking over Data Delayed on $k = 3$

Suppose that data are transferred with $k = 3$ and tune the filters as follows: UFIR in the worst case of $N_{opt} = 5$ valid for $k = 0$; KF as near optimal; and H_∞ in the best (unfeasible) case of known ground truth.

5.3.1 Tracking in the north direction

The vehicle trajectory (measured and delayed) and estimates provided by the filters in the north direction are sketched in Fig. 4. The trajectory is nonlinear (Fig. 4a) and the vehicle sometimes maneuvers rapidly. All estimates are consistent. Therefore, we show two parts of the trajectory on short time spans of $0 \leq n \leq 200$ in Fig. 4b and $300 \leq n \leq 400$ in Fig. 4c. Several observations can be made from Fig. 4:

- All filters track well the trajectory when a vehicle travels with a near constant velocity in one direction, as on $140 \leq n \leq 180$ in Fig. 4b.
- The filters temporary lose an ability of tracking and go along the delayed data when a vehicle quickly changes the direction, as on $340 \leq n \leq 343$ in Fig. 4c.
- All filters produce dynamic errors: the UFIR filter comes up with larger excursions but shorter transients, KF with shorter excursions but longer transients, and H_∞ is in-between, see on $80 \leq n \leq 100$ in Fig. 4b and $344 \leq n \leq 360$ in Fig. 4c.

Note that tracking in the east direction does not reveal any essential features.

5.4 Tracking with Temporary Lost Data

We next admit that some data points can be lost during the transmission, remove 5 data points at $n = (615...620)$ s and 10 at $n = (665...675)$ s as shown in Fig. 5 for $k = 3$. To predict lost data, we augment each algorithms with the prediction block as mentioned below (45) and run the filters. As can be seen, the filters act consistently with, however, some specifics. The estimates do not get away essentially from each other and the actual trajectory. However, when a vehicle maneuvers during the prediction, all filters diverge and return back to the actual trajectory with similar transients as in Fig. 4. The latter again speaks in favor of the UFIR filter, which has shorter transients.

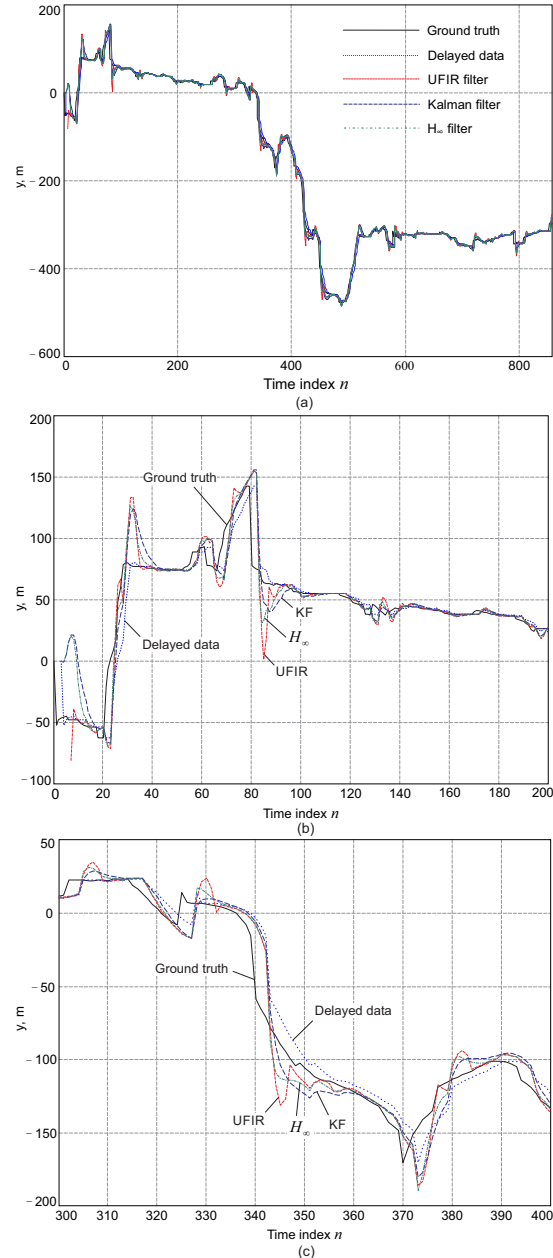


Figure 4. Tracking in the north direction y , m for $k = 3$ with UFIR filter ($N_{opt} = 5$), KF, and H_∞ filter ($\theta = 3.3 \times 10^{-4}$): (a) full time scale, (b) $0 \leq n \leq 200$, and (c) $300 \leq n \leq 400$.

6 Conclusions

The UFIR filter developed in this paper for GPS-based vehicle tracking over WSNs with time-stamped discretely-delayed and missing data has demonstrated better performance than the KF and H_∞ filter. The main benefits of using the UFIR filter are that it 1) does not require any information about noise and initial conditions, 2) becomes blind on given horizons, and 3) has shorter transients. The latter can be considered as an important practical advantage in all situations when the trajectory changes rapidly and estimators temporary loses an ability of tracking. Applications to GPS-based vehicle tracking with known discretely-delayed and missed data have proved a better performance of the UFIR filter.

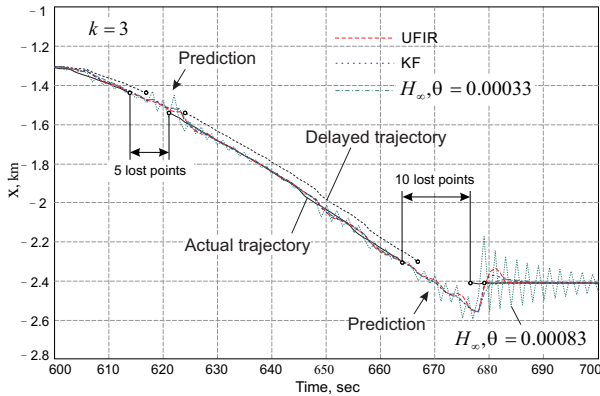


Figure 5. Vehicle tracking with temporary missing data in the east direction x by the UFIR filter, KF, and H_∞ filter, all augmented with the prediction option.

References

- [1] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, Estimation with Applications to Tracking and Navigation, (John Wiley & Sons, Hoboken, NJ, 2001).
- [2] F. Zhao, L. J. Guibas, Wireless Sensor Networks, (Elsevier: Amsterdam, 2004).
- [3] M. Malek-Zavarei, M. Jamshidi, Time-Delay Systems: Analysis, Optimization and Application, (North-Holland: Amsterdam, The Netherlands, 1987).
- [4] Y. Bar-Shalom, Update with out-of-sequence measurements in tracking: exact solution, *IEEE Trans. Aerosp. Electron. Syst.*, **38**, 769–778, (2002).
- [5] A. Ray, Introduction to networking for integrated control systems, *IEEE Contr. Syst. Mag.*, **9**, 76–79, (1989).
- [6] M. Sahebsara, T. Chen, S. L. Shah, Optimal H_∞ filtering with random sensor delay, multiple packet dropout and uncertain observations, *IEEE Trans. Autom. Contr.*, **52**, 1508–1513, (2007).
- [7] H. Zhang, G. Feng, G. Duan, X. Lu, H_∞ filtering for multiple-time-delay measurements, *IEEE Trans. Signal Process.*, **54**, 1681–1688, (2006).
- [8] M. Chitre, S. Shahabudeen, M. Stojanovic, Underwater acoustic communications and networking: recent advances and future challenges, *Marine Technology Society J.*, **42**, 103–116, (2008).
- [9] Z. Wang, D. W. C. Ho, X. Liu, Robust filtering under randomly varying sensor delay with variance constraints, *IEEE Trans. Circuits Systems–II: Express Briefs*, **51**, 320–326, (2004).
- [10] M. Liu, H. Chen, H_∞ state estimation for discrete-time delayed systems of the neural network type with multiple missing measurements, *IEEE Trans. Signal Process.*, **26**, 2987–2998, (2015).
- [11] Y. S. Shmaliy, A. V. Marienko, A. V. Savchuk, GPS-Based Optimal Kalman Estimation of Time Error, Frequency Offset, and Aging, *Proc. 31st PTTI Meeting*, Dana Point, California, USA, 431–440, (1999).
- [12] A. H. Jazwinski, Stochastic Processes and Filtering Theory, (New York: Academic Press, 1970).
- [13] Y. S. Shmaliy, S. Zhao, C. K. Ahn, Unbiased FIR filtering: an iterative alternative to Kalman filtering ignoring noise and initial conditions, *IEEE Contr. Syst. Mag.*, **37**, 70–89, (2017).
- [14] Y. S. Shmaliy, Unbiased FIR filtering of discrete-time polynomial statespace models, *IEEE Trans. Signal Process.*, **57**, 1241–1249, (2009).
- [15] Y. S. Shmaliy, An iterative Kalman-like algorithm ignoring noise and initial conditions, *IEEE Trans. Signal Process.*, **59**, 2465–2473, (2011).
- [16] F. Ramirez-Echeverria, A. Sarr, Y. S. Shmaliy, Optimal memory of discrete-time FIR filters in state-space, *IEEE Trans. Signal Process.*, **62**, 557–561, (2014).
- [17] S. Zhao, Y. S. Shmaliy, F. Liu, Fast Kalman-like optimal unbiased FIR filtering with applications, *IEEE Trans. on Signal Process.*, **64**, 2284–2297, (2016).
- [18] S. Zhao, Y. S. Shmaliy, F. Liu, Fast computation of discrete Optimal FIR estimates in white Gaussian noise, *IEEE Signal Process. Lett.*, **22**, 718–722, (2015).
- [19] W. H. Kwon, S. Han, Receding Horizon Control: Model Predictive Control for State Models, (London: Springer, 2005).
- [20] Z. Quan, A. Han, W. H. Kwon. A robust FIR Filter for linear discrete-time state-space signal models with uncertainties, *IEEE Signal Process. Lett.*, **14**, 553–556, (2007).
- [21] C. K. Ahn, Strictly passive FIR filtering for state-space models with external disturbance, *Int. J. Electron. Commun.*, **66**, 944–948, (2012).
- [22] C. K. Ahn, P. Shi, L. Wu, Receding horizon stabilization and disturbance attenuation for neural networks with time-varying delay, *IEEE Trans. Cybernetics*, **45**, 2680–2692, (2015).
- [23] I. Y. Song, D. Y. Kim, V. Shin, M. Jeon, Receding horizon filtering for discrete-time linear systems with state and observation delays, *IET Radar, Sonar & Navigation*, **6**, 263–271, (2012).
- [24] I. Y. Song, V. Shin, M. Jeon, Distributed fusion receding horizon filtering for uncertain linear stochastic systems with time-delay sensors, *J. Franklin Inst.*, **349**, 928–946, (2012).
- [25] H. D. Choi, C. K. Ahn, M. T. Lim, M. K. Song, Dynamic output-feedback H_∞ control for active half-vehicle suspension systems with time-varying input delay, *Int. J. Contr., Autom., Syst.*, **14**, 59–68, (2016).
- [26] Y. S. Shmaliy, S. Zhao, C. K. Ahn, Unbiased FIR filtering: an iterative alternative to Kalman filtering ignoring noise and initial conditions, *IEEE Control Systems Mag.*, **37** (5) 70–89, (2017).
- [27] Y. S. Shmaliy, D. Simon, Iterative unbiased FIR state estimation: a review of algorithms, *EURASIP J. Advances Signal Process.*, **113**, 1–16, (2013).
- [28] D. Simon, Optimal Estimation: Kalman, H_∞ , and Nonlinear Approaches, (Wiley, Hoboken, NJ, 2006).