RESEARCH　　　　　　　　　　　　　　　　　　　　　**Open Access**

Estimating the number of competing terminals without a state variation detector in wireless LAN

Jaechan Lim[1], Taejin Kim[2] and Daehyoung Hong[2]*

Abstract

Estimating the number of competing terminals n (who wish to transmit a packet at the same time) in the IEEE 802.11 system is important for system throughput performance because optimal back-off window size needs to be selected based on n. Therefore, as a new approach for estimating n, we propose H infinity filter that does not need a state variation detector as opposed to the cases of previously proposed approaches. The state variation detector's flaw is incurring tracking latency in addition to the side effect of increased computational cost. All previously proposed approaches demand the employment of the state variation detector to detect the variation of n in the IEEE 802.11 system. By employing H infinity filter, we show improved throughput performance of the system compared to that of previously proposed approaches (e.g., the Kalman filter and particle filter) based on the improved performance in tracking n. In this paper, we justify the superiority of the proposed approach in the terms of tracking performance, throughput performance, and computational complexity.

Keywords: Competing terminals; Gaussian mixture sigma point particle filter; H infinity filter; IEEE 802.11; Kalman filter

1 Introduction

IEEE 802.11 wireless local area network (WLAN) protocol [1] is commonly employed in response to the increased usage of multi-protocol mobile devices such as smartphones. This protocol shares medium resources based on distributed coordination function (DCF) in its medium access control (MAC) layer. DCF is a random access scheme and employs 'carrier sense multiple access with the collision avoidance (CSMA/CA)' protocol [2,3]. If a terminal detects a collision with other terminals for medium access, it waits for a random time interval and retries to acquire medium resources according to the protocol. The terminal selects random time interval within 'binary exponential back-off window size.' The throughput performance of the system can be maximized by selecting the optimal window size based on the number of competing terminals (n) which try to transmit packets simultaneously [4-6]. According to the DCF protocol, a terminal can choose either the 'basic access mode' or 'request-to-send/

clear-to-send (RTS/CTS) access mode' depending on a predetermined RTS threshold of the packet payload length. It has been shown in [7] that the RTS threshold which maximizes the system throughput performance also depends on n [8]. Therefore, there are a number of motivations for the accurate estimation of n; nonetheless, the task is not easy because n varies without any prior information nor statistical knowledge.

The extended Kalman filter (EKF) approach was proposed in [9] where a 'state variation detector based on cumulative summary (CUSUM) algorithm' is additionally employed to adaptively change the state noise variance when the variation of n is detected. Particle filtering (PF) was also proposed in [10] to combat the non-Gaussian/non-linear problem. Recently, Gaussian mixture sigma point particle filtering (GMSPPF) was proposed in [11] that shows outperforming results over existing methods, e.g., the EKF, unscented Kalman filter (UKF), PF, unscented PF (UPF).

The novel contribution of this paper can be specified in the following. In this paper, we propose a new approach, i.e., H infinity filter (HIF) for estimating n, by which we can obtain enhanced throughput performance compared

*Correspondence: dhong@sogang.ac.kr
[2]Department of Electronic Engineering, Sogang University, Seoul, Korea
Full list of author information is available at the end of the article

to that obtained by previously proposed approaches. The proposed HIF can be employed without the knowledge on noise statistics of the problem (i.e., the state and measurement noise). Consequently, we do not need to know the variance of the state noise as opposed to the cases of the EKF and PF approaches. In the usage of the EKF and PF, we need the knowledge on the mean and variance of the state and measurement noise; furthermore, Gaussian noise is assumed in the applications of the EKF. However, in the investigated problem of this paper, the varying pattern of the state (i.e. *n*) is hard to be known. Therefore, the state variation detector (i.e. CUSUM algorithm) was additionally employed in the application of the EKF [9]. This is the same case for the application of PF to the investigated problem. In other words, additional CUSUM algorithm needs to be employed for the application of PF too for this problem. This state variation detection algorithm also causes the delay in the tracking ability against the prompt detection of the state variation. Consequently, the employment of HIF requires the least computational complexity due to the non-necessity of the additional CUSUM algorithm for the detection of the state variation. The computational complexity of PF is proportional to the number of particles (e.g., if 100 particles are employed, the computational complexity of PF is approximately 100 times as that of the EKF or HIF). Therefore, the advantages of the proposed filter are the superiority in the computational complexity and prompt tracking ability that results in the better mean squared error performance for accurate tracking of *n*. Furthermore, accurate tracking performance affects the throughput performance of the network. These advantages originally result from the feature of the HIF that we can apply this filter with unknown noise statistics (both state and measurement). In this paper, we justify the superiority and advantageous features of the proposed HIF in terms of tracking ability, throughput performance, and computational complexity.

2 The problem formulation

2.1 Estimation in dynamic state system

Many problems in the area of signal processing, a parameter of interest is estimated based on some measurement. Particularly, if the system model can be described by the discrete time-varying states with corresponding measurement, we can estimate the state sequentially by applying dynamic filters. The dynamic state system that **n** and observed measurement **ξ** with zero mean and additive noise processes of **w** and **v** at time step k is expressed as follows:

$$\boldsymbol{n}_k = g(\boldsymbol{n}_{k-1}) + \boldsymbol{w}_k, \tag{1}$$

$$\boldsymbol{\xi}_k = h(\boldsymbol{n}_k) + \boldsymbol{v}_k, \tag{2}$$

where boldface denotes a vector, and $g(\cdot)$ and $h(\cdot)$ are the given state transition and the observation function,

respectively, which are possibly non-linear. According to this system model, we can estimate the time-varying state $\mathbf{n}_k$ sequentially based on the observed measurement $\boldsymbol{\xi}_k$ by dynamic filters, e.g., the Kalman filter, particle filter, and their variants.

In general cases, i.e., apart from the linear and Gaussian case, there exist a number of sub-optimal approaches such as particle filtering (PF) [12,13]. The Kalman filter also can be applied as a sub-optimal approach to non-linear model by extending it using Taylor series (in this case, still Gaussian noise is assumed) [14].

There are also variants of the EKF and those of PF approaches, e.g., UKF, Gaussian PF, auxiliary PF, likelihood PF, etc. Although PF approaches require high computational cost, they outperform EKF and its variants in most highly non-linear problems; furthermore, the Gaussian noise does not have to be assumed.

2.2 Dynamic state system for estimating *n*

All the filtering methods, i.e. the proposed approach in this paper and previously proposed approaches, basically estimate the dynamically varying (in time) state by using a measurement which is a function of the state. Therefore, the problem of estimating *n* needs to be modeled by the 'dynamic state system' and consequent 'measurement equation' first.

Following [7,9], we consider a scenario of the fixed number of *n* competing terminals. Each terminal operates in the saturated network and ideal channel conditions: the saturated network condition means that all terminals always have a packet waiting for transmission, and neither packet corruption nor hidden terminals are assumed in the ideal channel condition. ξ denotes the probability that a packet is transmitted on a collided channel. Then, *n* can be expressed as the function of ξ as follows [9].

$$n = f(\xi) = 1 + \frac{\log(1-\xi)}{\log\left(1 - \frac{2(1-2\xi)}{(1-2\xi)(D_{\min}+1)+\xi D_{\min}(1-(2\xi)^m)}\right)}, \tag{3}$$

where m and $D_{\min}$ denote the maximum back-off stage and the minimum contention window sizes, respectively. After a collision, each terminal doubles its contention window size up to the maximum value $D_{\max} = 2^m D_{\min}$, where $m = \log_2(D_{\max}/D_{\min})$. Maximization of the saturation throughput of IEEE 802.11 systems has been studied [4,7]. Saturation throughput is defined as 'the number of successfully transmitted payload slots by all terminals over observed slot numbers.' It can be thought of as the channel utilization efficiency. In order to maximize the saturation throughput, the minimum back-off window size is properly selected, and the optimal relationship is analytically obtained as follows:

$$D_{\min} = n\sqrt{2T}, \tag{4}$$

where T is the packet transmission time that comprises the header, SIFS, ACK, and DIFS. The size of T depends on the employed PHY and transmitted payload size; therefore, it can be easily obtained to the terminal. D_{min} needs to be selected according to (4) to obtain optimal saturation throughput of the system. From (3), we can estimate the number of competing terminals by measuring collision probability ξ regardless of the packet size. ξ is measured by each terminal at the time step k (each time step comprises B slots) as follows:

$$\xi_k = \frac{1}{B} \sum_{i=(k-1)B}^{kB-1} C_i, \tag{5}$$

where $C_i = 0$ if the corresponding slot is empty or the transmission is successful during the slot i while $C_i = 1$ if the slot is busy or collided.

B is a constant value; however, it does not mean that the true observation time duration is constant [9,15]. As shown in Figure 1, one slot can have the duration of an empty slot (defined by the standard) or last the transmission time of either a successful transmission or a collision. If the packet length or n increases, then the actual observation time increases even if B remains the same. This will degrade the prompt tracking performance.

The discrete time dynamic state system model for the number of competing terminals is defined as follows [9].

$$n_k = n_{k-1} + w_k, \tag{6}$$

where w_k is the state process noise.

The obtained measurement (5) is related with the state n_k based on (3) as follows:

$$\xi_k = f^{-1}(n_k) + v_k = h(n_k) + v_k, \tag{7}$$

where $h(\cdot)$ is the inverse function of (3), v_k is the observation noise that follows the binomial distribution with zero mean, and the variance can be computed as follows [15]:

$$\text{Var}[v_k] = \frac{h(n_k) \cdot [1 - h(n_k)]}{B}. \tag{8}$$

Therefore, the system model of the problem is non-linear and non-Gaussian. Based on the problem formulation, i.e., (6) and (7), we are ready to apply all the filtering approaches now. It is assumed that (6) and (7) are known except for the information of w_k. The state variation detector is employed to obtain even a limited information of the statistics of w_k in the previously proposed approaches, while preliminary training process with unknown statistics is performed in the extended HIF (EHIF) approach. The filtering methods estimate n_k by using the measured information of (7) at every time step. Note that all variables (n_k, ξ_k, w_k, v_k) are expressed in a scalar form because the state and measurement are all scalar in this problem.

3 Filtering methods

In this section, we propose H infinity filter for the problem of estimating the number of competing terminals and also discuss previously proposed approaches.

3.1 Previously proposed approaches

The Kalman filter is the optimal method if the problem is linear and jointly Gaussian. The investigated problem is non-linear and non-Gaussian. The EKF can be employed in non-linear systems using the Taylor expansion. This EKF algorithm was proposed in [9], and additional CUSUM algorithm was employed for the state variation detector. Using the CUSUM algorithm, the EKF changes the state noise variance Q_k adaptively. This
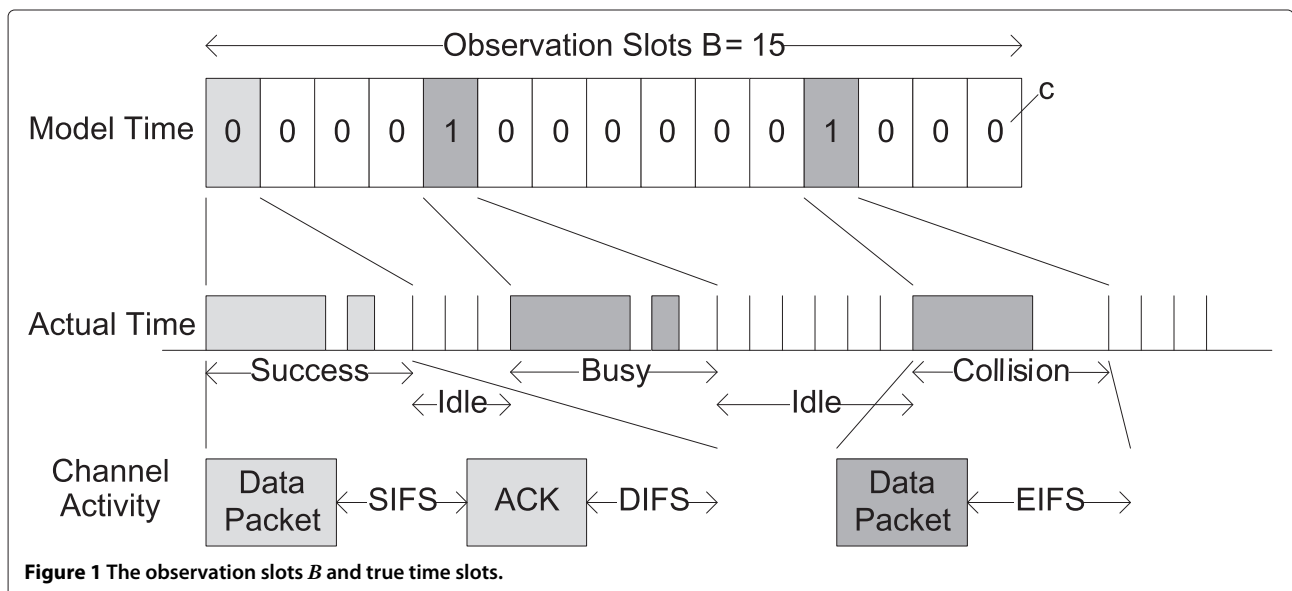

Figure 1 The observation slots B and true time slots.

approach may work well, but it has considerable latency. Another disadvantage for limited performance is that only two deterministic variances are used regardless of statistical variation pattern of n.

GMSPPF [16] adopts an 'UKF-based Gaussian sum filter' in particle filtering framework. GMSPPF employs the Gaussian mixture model (GMM), and GMM has multiple means and variances to represent a complicated density which cannot be represented by a normal Gaussian model. GMSPPF uses $G \cdot I \cdot J$ numbers of parallel UKF banks (where G is the number of GMM components, I is the number of state noise components, and J is the number of measurement noise components). Furthermore, the expectation maximization algorithm is employed to obtain the GMM from particles.

In [11], GMSPPF was proposed for the investigated problem and showed that GMSPPF provides more outstanding performance compared to that of the EKF, UKF, PF, and UPF; however, this paper assessed the performance of the approaches without employing the CUSUM algorithm in any approaches, which is not appropriate. Therefore, in this paper, we apply the CUSUM algorithm to all existing approaches, and compare the performance with that of EHIF which does not need the CUSUM algorithm.

We also consider another PF approach, i.e., Gaussian PF (GPF) [17] in this paper for performance comparison although GPF has not been proposed for the investigated problem in the literature. In GPF, resampling process is replaced by another step of particle generation from a Gaussian density that has the sample mean and the sample variance of generated particles from the prior density. Due to the advantages of simplicity and low computational complexity, it is relatively easy to apply GPF algorithm among particle filtering approaches. We can expect that GPF outperforms the EKF when the CUSUM is employed for both approaches.

3.2 Proposed approach

In this paper, we propose EHIF which does not require the knowledge on noise statistics in its applications [18]. Whereas the EKF minimizes mean squared error (MSE) of the estimate, EHIF is designed to minimize the worst possible error. The original HIF was employed in control areas, and HIF has not been widely employed due to its level of mathematical understanding and the necessity of a good modeling of the system. Discrete time HIF is based on zero-sum game theory, and we minimize the cost function which is designed on the basis of the strategy that the probability of maximum expected error is minimized regardless of the opponent; therefore, the filter is a player prepared for the worst strategy that the other player (the nature) can provide, and the goal is providing a uniformly small estimation error for any processes, measurement

noises, and any initial states. Consequently, we do not need to know the noise statistics of the system for the filtering process. Accordingly, the cost function in discrete time HIF is defined as follows:

$$
J = \frac{\sum_{k=0}^{N-1} \| n_k - \hat{n}_k \|^2_{\chi_k}}{\| n_0 - \hat{n}_0 \|^2_{P_0^{-1}} + \sum_{k=0}^{N-1} \left(\| w_k \|^2_{W_k^{-1}} + \| v_k \|^2_{V_k^{-1}} \right)},
$$
(9)

where N is the number of total time steps, $\hat{n}_k$ is the estimated state at time step k, w_k is the state noise, v_k is the observation noise, and n_0 is the initial state, respectively. $\chi_k, P_k, W_k,$ and V_k are weighting factors, and $\| \cdot \|$ denotes the vector norm $\left(\text{i.e., } \| w_k \|^2_{W_k^{-1}} \text{ implies} w_k^\top W^{-1} w_k \right)$. The way how the weighting factors are determined is that, for instance, if it is known that the second element of $v(k)$ is small, then $V_k(2,2)$ is chosen to be small compared to other elements; otherwise, the weighting factors are determined after preliminary training process. However, the state (n) to be estimated is a scalar in this problem, which results in simplified process of the preliminary tuning process for satisfactory tracking performance.

Direct minimization of J is not tractable; therefore, the performance bound γ is introduced, and it satisfies

$$
J < \gamma^{-1}.
$$
(10)

Then, J' is defined as

$$
J' = -\gamma^{-1} \| n_0 - \hat{n}_0 \|^2_{P_0^{-1}} + \\
\sum_{k=0}^{N-1} \left[\| n_k - \hat{n}_k \|^2_{\chi_k} - \gamma^{-1} \left(\| w_k \|^2_{W_k^{-1}} + \| v_k \|^2_{V_k^{-1}} \right) \right],
$$
(11)

and the problem becomes a matter of solving the following minimax problem:

$$
\min_{\hat{n}_k} \left(\max_{w_k, v_k, n_0} J' \right).
$$
(12)

The EHIF approach which solves above minimax problem is given by [18]

$$
\hat{n}_k = \hat{n}_{k-1} + H_k \left[\xi_k - h(\hat{n}_{k-1}) \right]
$$
(13)

if and only if there exists a stabilizing symmetric solution $P_k > 0$ to the following discrete time *Riccati* equation:

$$
P_k = P_{k-1} S_k + W_k
$$
(14)

where '^' denotes the estimate version, H_k is the H_∞ gain, and given by

$$
H_k = P_{k-1} S_k h'(\hat{n}_{k-1}) V_k^{-1}
$$
(15)

where $h'(\hat{n}_{k-1}) = \frac{\partial h(n_k)}{\partial n_k}\Big|_{n_k = \hat{n}_{k-1}}$, and

$$S_k = \left\{ I - \gamma \chi_k P_{k-1} + \left[h'(\hat{n}_{k-1}) \right]^2 V_k^{-1} P_{k-1} \right\}^{-1}, \quad (16)$$

where I is the identity matrix or 1 in case of a scalar. Note that $h'(\hat{n}_{k-1})$ is employed for an approximated linear form using Taylor expansion due to the non-linearity of the state equation.

Although we need to compute the derivative of the non-linear function $h(n_k)$, it is relatively easy to compute because the state n is a scalar, and consequently, we do not need to compute Jacobian. When the noise attenuation level γ is selected, we have to be very careful, especially it has to satisfy the following condition:

$$\gamma < P_k^{-1} + \left[h'(\hat{n}_{k-1}) \right]^2 V_k^{-1} \quad (17)$$

to maintain $P_k > 0$ [19]. We select γ after preliminary training process for optimal performance. The detailed steps of EHIF algorithm for estimating n are summarized in Algorithm 1.

Algorithm 1 Extended H infinity filtering for estimating *n*

Initialization Initialize the performance bound γ, the estimate $\hat{n}_0$ and weight parameters (χ_k, P_k, W_k, V_k); we select 1 for χ_k.
Recursive update for $k = 1, \ldots, N$

(1) Compute H infinity gain
$H_k = P_{k-1} S_k h'(\hat{n}_{k-1}) V_k^{-1}$.
(2) Update the estimate
$\hat{n}_k = \hat{n}_{k-1} + H_k \left[p_k - h'(\hat{n}_{k-1}) \right]$.
(3) Update the error variance $P_k = P_{k-1} S_k + W_k$.

In case the system model is linear (could be in some other linear problems), i.e., if (1) and (2) can be expressed in linear forms as follows:

$$\boldsymbol{n}_k = \mathcal{G} \boldsymbol{n}_{k-1} + \boldsymbol{w}_k, \quad (18)$$

$$\boldsymbol{\xi}_k = \mathcal{H} \boldsymbol{n}_k + \boldsymbol{v}_k, \quad (19)$$

where $\mathcal{G}$ and $\mathcal{H}$ are matrixes. Non-extended, original H infinity solution can be obtained by the steps in the following:

1. H infinity gain $H_k = \mathcal{G} P_{k-1} S_k \mathcal{H}^\top V_k^{-1}$, where
$S_k = \left\{ I - \gamma \chi_k P_{k-1} + \mathcal{H}^\top V_k^{-1} \mathcal{H} P_{k-1} \right\}^{-1}$.
2. The estimate $\hat{\boldsymbol{n}}_k = \mathcal{G} \hat{\boldsymbol{n}}_{k-1} + H_k \left[\boldsymbol{\xi}_k - \mathcal{H} \hat{\boldsymbol{n}}_{k-1} \right]$.
3. Update $P_k = \mathcal{G} P_{k-1} S_k \mathcal{G}^\top + W_k$.

The proposed EHIF approach just adopts Taylor expansion to this solution for approximate linearization of

non-linear measurement function. In this problem of estimating n, $\mathcal{G}$ equals 1, and furthermore, $\mathcal{H}$ and $\mathcal{H}\hat{n}_{k-1}$ are replaced by $h'(\hat{n}_{k-1})$ and $h(\hat{n}_{k-1})$, respectively, for the application of extended version of H_∞ filtering approach.

3.3 Computational complexity

We also assess the computational complexity of approaches in big O notation in this section. The big O notation concerns only major part of computational complexity for comparison purpose. We assume that 'matrix floating point' multiplication demands the highest computational complexity as assumed in [11]. The computational complexity of the approaches are given in the following, when M is the number of employed particles, the dimension of the state is L, and G is the number of GMM mixtures. The EKF requires $O(L^3)$, and the computation of the Kalman gain spans most of calculation time in the algorithm. The EM algorithm requires most time in the GMSPPF algorithm, and its computational complexity is $O(GL^2 M)$. GPF requires the complexity of $O(L^4 M)$, and the calculation of the particle weight takes most time in the algorithm. The computation of S_k demands most time in EHIF filtering, and whole algorithm requires the complexity of $O(L^4)$.

Furthermore, if the state is a scalar, i.e. $L = 1$ with $\beta = 2$ and $G = 3$, the computational complexity becomes $O(1)$ for the EKF, $O(M)$ for GPF, $O(3M)$ GMSPPF, $O(1)$ for EHIF, respectively. The number of particles are the main factor of the computational complexity for PF approaches. When the number of particles are the same for all PF approaches, GMSPPF requires the highest computational complexity. Although the EKF and EHIF have the same complexity in the big O notation, actually EHIF is much simpler because it does not need the CUSUM algorithm. Due to its low complexity, EHIF has an advantage for implementing in small devices.

4 Performance assessment

By simulations, we assess and compare the performance of the EKF, GPF, GMSPPF, and EHIF. We consider IEEE 802.11 system's DCF basic access mode where the slot period is 20 μs, D_{max} is 1024, and the other simulation parameters are the same as selected in [9]. We assume 2,000 observation slots of B, the initial estimate $\hat{n}_0 = 5$, and the initial error variance $P_0 = 10$ for all filtering methods. The CUSUM algorithm is adopted in the EKF, GPF, and GMSPPF. Three hundred particles are employed for GPF and GMSPPF. The parameters for GMSPPF and EHIF are as follows: $G = 3$, $I = 1$, $J = 2$, $\gamma = 0.001$, $\chi_k = 1$, $W_k = 2$, and $V_k = 0.0001$. The simulations are performed using MATLAB 7.11.0. Some *ReBEL Toolkit* functions (e.g., GMM) are adopted for GMSPPF as employed in [16]. The performance is assessed via 'MSE

of estimated n' and the 'saturation throughput' which has been defined in Section 2.

Figure 2 shows how D_{min} affects the saturation throughput performance. In the middle of simulation, n varies from 10 to 25. At that time, D_{min} is adjusted based on two numbers of terminals according to (4), i.e., the true n and the estimate $\hat{n}^{EKF}$ by the EKF, respectively. Whereas the throughput is degraded as n increases with non-adjusted D_{min}, the throughput degradation is not observed when D_{min} is adjusted based on true n and $\hat{n}^{EKF}$.

In the saturated network condition, all terminals are assumed to always have a packet waiting for transmission in their buffers. We assume that n varies with time as follows: (5, 10, 25, and 15 terminals) and simulation time duration is 350 s. This scenario is similar to that employed in [9]. We run 200 simulations for both MSE and throughput performance assessment. In this scenario, all terminals estimate n and reflect it for adjusting D_{min} to maximize the throughput. The result shows that EHIF outperforms all the other approaches in both MSE and saturation throughput as shown in Table 1. The result shows similar throughput performance for all approaches during the period of non-variation of n whereas EHIF shows the best throughput performance during the period of

Table 1 Results of saturated network condition

| | Mean squared error | | Saturation throughput (150 to 160 s) | |
	Mean	Variance	Mean	Variance (10^{-5})
EKF	1.492829	0.055072	0.634940	4.891636
GPF	1.105716	0.034910	0.635292	3.649367
GMSPPF	1.079393	0.032417	0.635308	3.179673
EHIF	0.894879	0.019242	0.637274	1.853646

variation of n. In Table 1, we highlighted the throughput performance of EHIF during the period of the largest variate of n (i.e., from 10 to 25 at 150 s). This result is from the fact that the approaches that employ the state variation detector based on CUSUM algorithm has considerable latency to make the decision for adjusting the variance in their filtering. On the other hand, EHIF approach does not have any latency during any variation of n because any decision is not needed to be made in its filtering process. This is how EHIF avoids the latency and makes the prompt tracking possible for consequent enhanced performance.
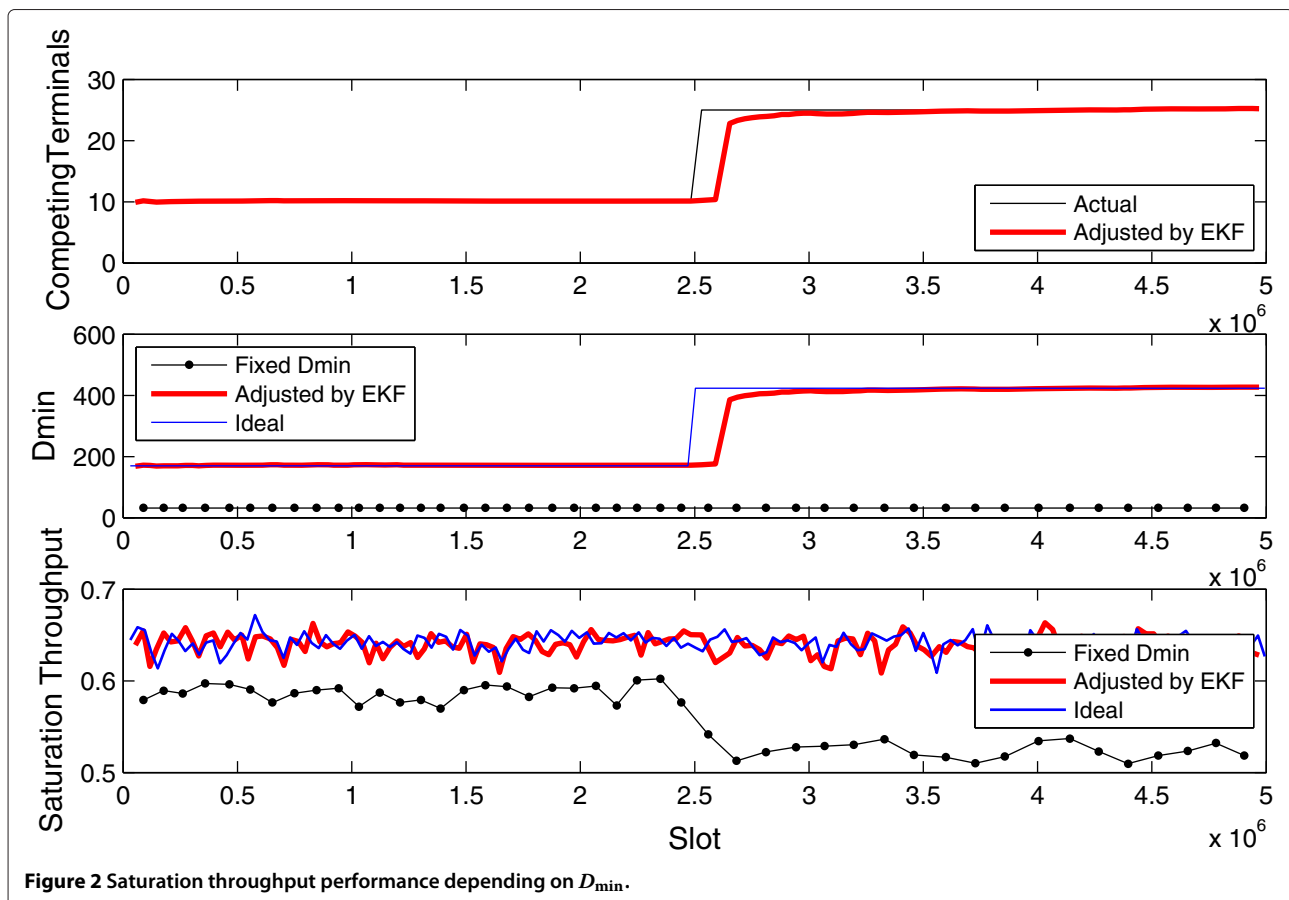

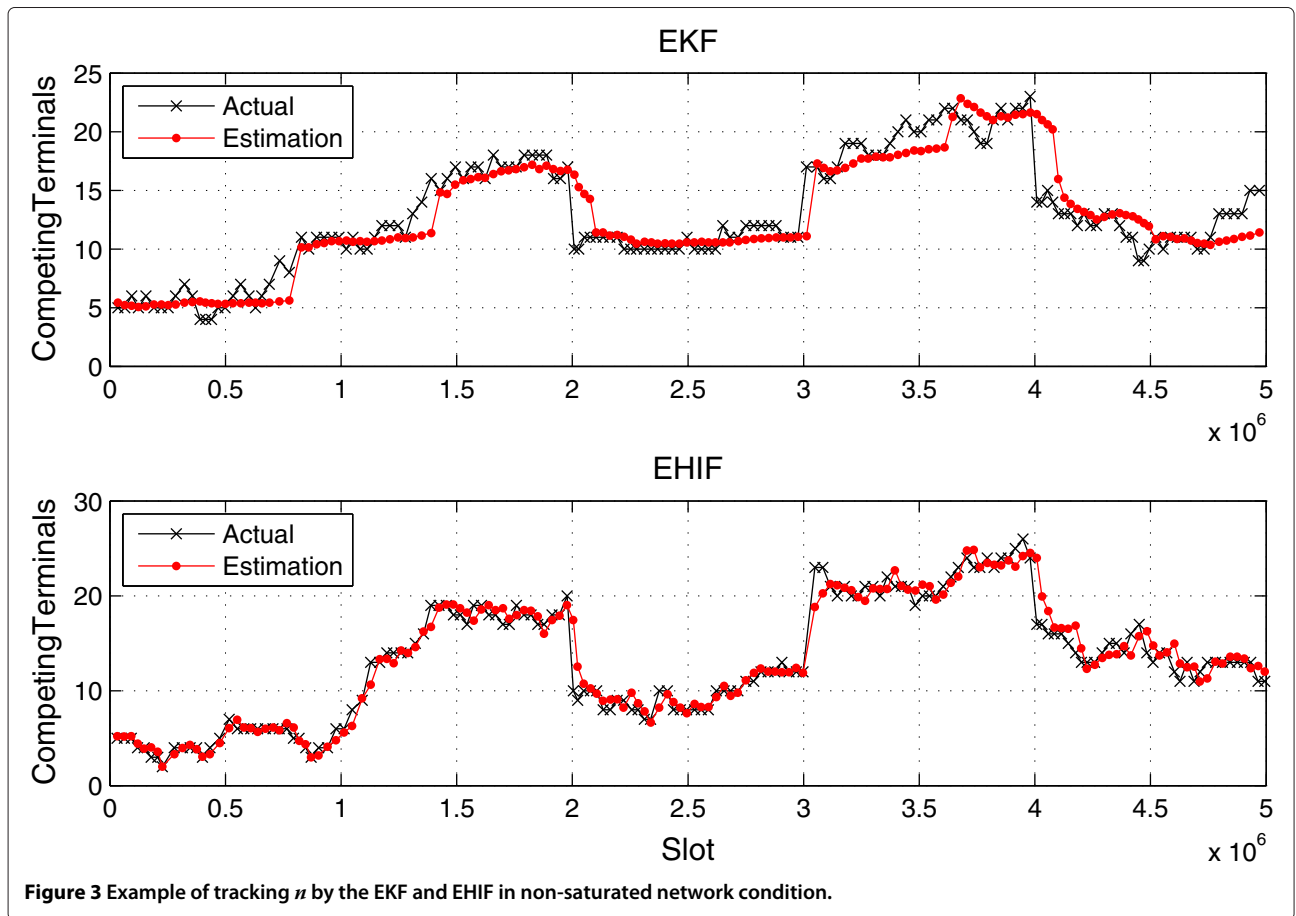**Figure 2 Saturation throughput performance depending on D_{min}.**

Figure 3 Example of tracking n by the EKF and EHIF in non-saturated network condition.
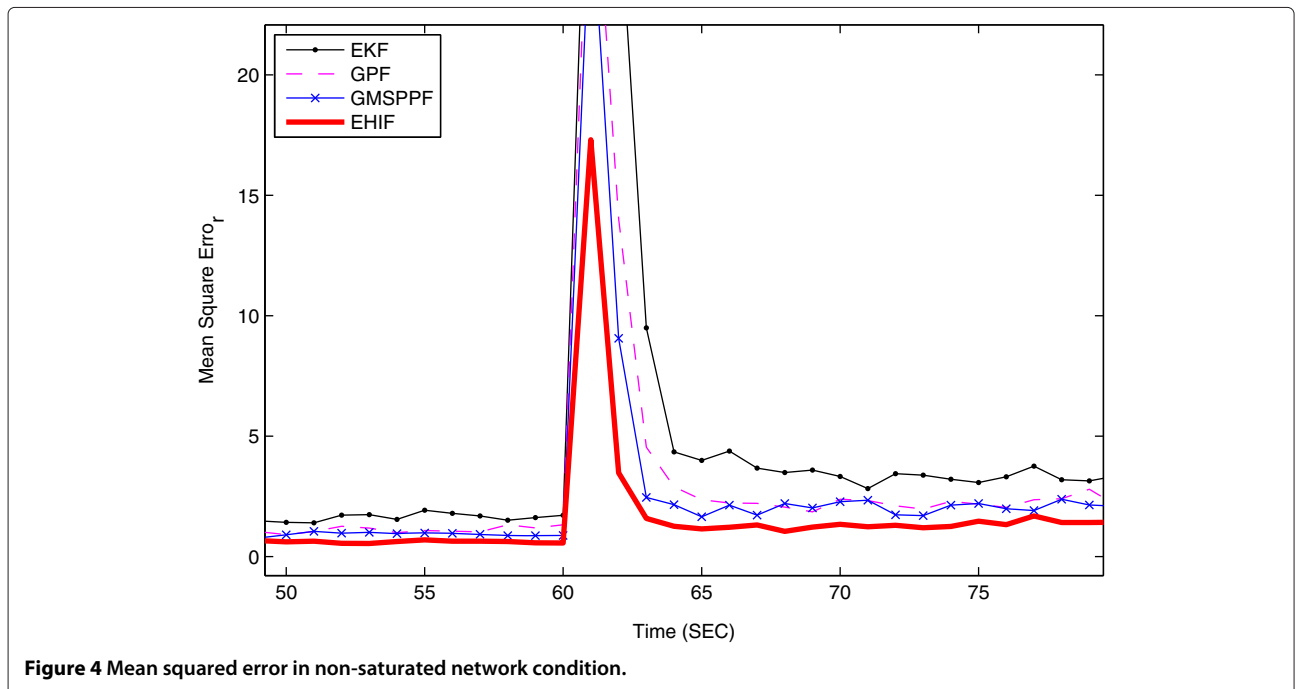

Figure 4 Mean squared error in non-saturated network condition.

In the non-saturated network condition, all terminals send packets randomly with random lengths. The scenario is close to the real internet network condition where users' random requests generate random data traffic. Therefore, prompt tracking and latency avoidance is more critically demanded in this scenario than the case of the non-saturated network condition. We assume that the data traffic length and the transmission period follow the exponential distribution with the rates of 0.11/s and 0.7/s, respectively. To describe large and frequent variations of n, e.g., as in subways or hotspot areas, we change n every 20 s in the pattern of (15, 25, 15, 30, and 20 terminals). Figure 3 shows an example of tracking n by the EKF and EHIF where we can notice that the tracking performance of EHIF is better than that of the EKF. The EKF shows latency clearly in tracking n that is caused by the employment of the state variation detector in the algorithm. On the other hand, EHIF shows prompt tracking in response to the variation of the state n without any latency. We evaluate MSE of estimates and saturation throughput by 200 runs with this scenario. Figures 4 to 5 and Table 2 show that EHIF outperforms all the other filtering approaches in the non-saturated network condition. EHIF shows the best performance, particularly during the periods of the large variate of n due to its prompt tracking performance. Figure 5 shows the throughput performance of the methods, and the performance comparison is noticeable clearly only during the periods of the large variate of n. During the periods of non-varying n, significant difference of the throughput performance is not observed unless the difference between the optimal D_{min} and applied one is greater than a certainly impactive level. Nonetheless,

Table 2 Results of non-saturated network condition

	Mean squared error		Saturation throughput	
	Mean	Variance	Mean	Variance (10^{-7})
EKF	4.363303	0.792361	0.639690	8.408113
GPF	2.818697	0.391450	0.640170	8.687158
GMSPPF	2.432454	0.292157	0.640205	7.770071
EHIF	1.528985	0.110101	0.640418	7.325334

prompt tracking ability is a crucial factor for enhanced throughput performance, particularly for the periods of dynamic variation of n.

5 Conclusions

In this paper, we proposed the EHIF approach for estimating the number of competing terminals to enhance the throughput performance of the IEEE 802.11 system. We showed that EHIF outperformed previously proposed approaches (i.e., EKF and GMSPPF) in both saturated and non-saturated network conditions. We focused on the 'non-saturated condition' more because the problem is barely investigated for the non-saturated condition in the literature. Furthermore, the non-saturated condition is closer to real scenarios because the state n varies more dynamically in the condition than it varies in the saturated condition. In the application of EHIF, the knowledge of noise statistics is not needed; therefore, additional CUSUM algorithm is not needed that incurs the considerable latency in tracking n, which results in significantly degraded tracking performance. Consequently, we
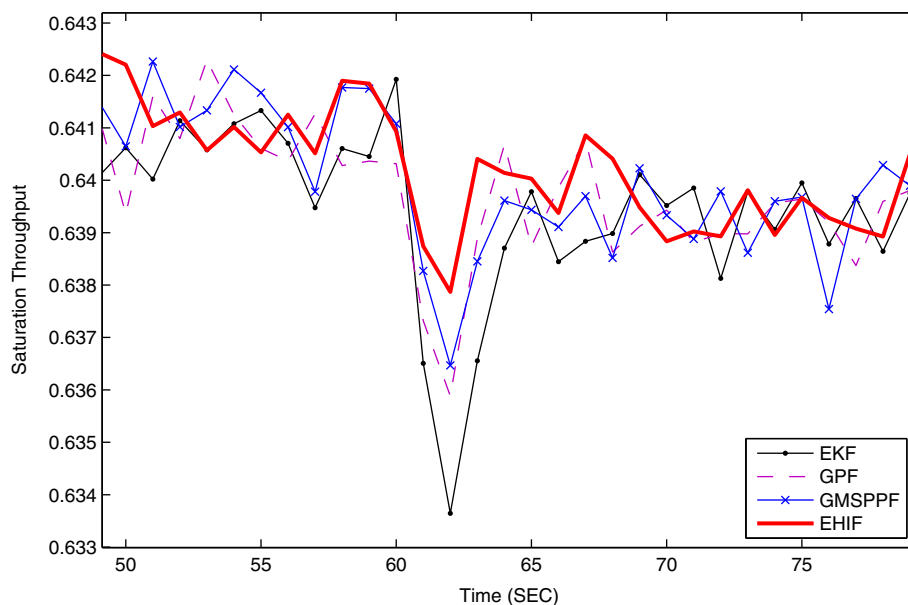

Figure 5 Saturation throughput in non-saturated network condition.

obtained improved saturation throughput performance by employing EHIF based on prompt tracking of n. Furthermore, EHIF can be easily implemented in small devices due to its low computational complexity.

Abbreviations

CSMA/CA: Carrier sense multiple access with the collision avoidance; CTS: Clear-to-send; CUSUM: Cumulative summary; DCF: Distributed coordination function; DIFS: Distributed inter-frame space; EHIF: Extended H infinity filter; EKF: Extended Kalman filter; EM: Expectation maximization; GMM: Gaussian mixture model; GMSPPF: Gaussian mixture sigma point particle filter; GPF: Gaussian particle filter; HIF: H infinity filter; MAC: Medium access control; MSE: Mean squared error; PF: Particle filter(ing); PHY: Physical layer; RTS: Request-to-send; SIFS: Short inter-frame space; UKF: Unscented Kalman filter; WLAN: Wireless local area network.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] Future IT Innovation Laboratory, Pohang University of Science and Technology, Pohang, Korea. [2] Department of Electronic Engineering, Sogang University, Seoul, Korea.

References

1. IEEE, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. (IEEE, Piscataway, 1999)
2. S Romaszko, C Blondia, Cross layer PHY-MAC protocol for wireless static and mobile ad hoc networks. EURASIP J. Adv Signal Processing. **2009**, 278041 (2009)
3. T Hewer, M Nekkovee, P Coveney, Reducing congestion in obstructed highways with traffic data dissemination using ad hoc vehicular. EURASIP J. Adv. Signal Process. **2010**, 169503 (2010)
4. G Bianchi, L Fratta, M Oliveri, Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs. Proc, PIMRC 1996. **2**, 392–396 (1996)
5. G Hwang, D Cho, Performance analysis on coexistence of EDCA and legacy DCF stations in IEEE 802.11 wireless LANs. IEEE Trans. Wireless Commun. **5**(12), 3355–3359 (2006)
6. S Chang, H Wu, Novel adaptive DCF protocol using the computationally-efficient optimization with the feedback network information for wireless local-area networks. IEEE Trans. Wireless Commun. **8**(6), 2827–2830 (2009)
7. G Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function. IEEE J. Select. Areas Telecommun., Wireless Series. **18**(3), 535–547 (2000)
8. D Deng, C Ke, H Chen, Y Huang, Contention window optimization for IEEE 802.11 DCF access control. IEEE Trans. Wireless Commun. **7**(12), 5129–5135 (2008)
9. G Bianchi, I Tinnirello, Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network. Proc. Infocom 2003. **2**, 844–852 (2003)
10. T Vercauteren, AL Toledo, X Wang, Batch and sequential Bayesian, estimators of the number of active terminals in an IEEE 802.11 network. IEEE Trans. Signal Process. **55**(2), 437–450 (2006)
11. J Kim, E Serpedin, D Shin, Improved particle filtering-based estimation of the number of competing stations in IEEE 802.11 networks. IEEE Signal Process. Lett. **15**, 87–90 (2008)
12. M Arulampalam, S Maskell, N Gordon, T Clapp, A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. IEEE Trans. Signal Processing. **50**(2), 174–188 (2002)
13. HAA El-Halym, I Mahmoud, S Habib, Proposed hardware architectures of particle filter for object tracking. EURASIP J. Adv. Signal Process. **2012**, 17 (2012)
14. K Jishy, F Lehmann, A Bayesian track-before-detect procedure for passive radars. EURASIP J. Adv. Signal Process. **2013**, 45 (2013)
15. G Bianchi, I Tinnirello, Interference estimation in IEEE 802.11 networks. IEEE Control Syst. Mag. **30**(2), 30–43 (2010)
16. R van der Merwe, E Wan, Gaussian mixture sigma-point particle filters for sequential probabilistic inference in dynamic state-space models. Proc. Int. Conf. Acoustics, Speech, and Signal, Process. (ICASSP). **6**, 701–704 (2003)
17. JH Kotecha, PM Djuric, Gaussian particle filtering. IEEE Trans. Signal Process. **51**(10), 2592–2601 (2003)
18. X Shen, L Deng, Game theory approach to discrete H_∞ filter design. IEEE Trans. Signal Process. **45**(4), 1092–1095 (1997)
19. Y Jing, F Yin, Z Chen, An H_∞ filter based approach to Coombat inter-carrier interference for OFDM systems. IEEE Commun. Lett. **12**(6), 453–455 (2008)