Hindawi
Wireless Communications and Mobile Computing
Volume 2018, Article ID 9641712, 7 pages
https://doi.org/10.1155/2018/9641712

Research Article

Detection Performance of Packet Arrival under Downclocking for Mobile Edge Computing

Zhimin Wang,[1] Qinglin Zhao,[1] Fangxin Xu,[1] Hongning Dai,[1] and Yujun Zhang[2]

[1]*Faculty of Information Technology, Macau University of Science and Technology, Avenida Wei Long, Taipa, Macau*
[2]*Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China*

Correspondence should be addressed to Qinglin Zhao; zqlict@hotmail.com

Received 8 November 2017; Revised 11 January 2018; Accepted 29 January 2018; Published 28 February 2018

Academic Editor: Anna Kobusinska

Mobile edge computing (MEC) enables battery-powered mobile nodes to acquire information technology services at the network edge. These nodes desire to enjoy their service under power saving. The sampling rate invariant detection (SRID) is the first downclocking WiFi technique that can achieve this objective. With SRID, a node detects one packet arrival at a downclocked rate. Upon a successful detection, the node reverts to a full-clocked rate to receive the packet immediately. To ensure that a node acquires its service immediately, the detection performance (namely, the miss-detection probability and the false-alarm probability) of SRID is of importance. This paper is the first one to theoretically study the crucial impact of SRID attributes (e.g., tolerance threshold, correlation threshold, and energy ratio threshold) on the packet detection performance. Extensive Monte Carlo experiments show that our theoretical model is very accurate. This study can help system developers set reasonable system parameters for WiFi downclocking.

1. Introduction

Mobile edge computing (MEC) [1] aims to provide computing resources and information technology services at the network edge. In MEC, various battery-powered mobile nodes (such as smartphone) will access these resources and services via MEC application servers such as LTE base station and wireless access point (AP). These battery-powered nodes desire to enjoy their service under power saving.

In this paper, we assume that a number of battery-powered nodes access an AP (acting as an MEC application server) via a WiFi network. These devices adopt a novel algorithm called sampling rate invariant detection (SRID) [2] for power saving. SRID is the first downclocking mechanism (adopted in WiFi). With SRID, a node detects one packet arrival at a downclocked rate. Upon a successful detection, the node reverts to a full-clocked rate to receive the packet immediately. For each detection, there are two types of typical errors: miss-detection (i.e., the AP sends a packet but the node does not detect it) and false-alarm (i.e., the AP sends nothing but the node detects a packet mistakenly). To ensure that a node acquires its service immediately, the detection performance (namely, the miss-detection probability and the false-alarm probability) of SRID is of importance. This paper is concerned with the detection performance. Our contributions are summarized as follows:

(i) To the best of our knowledge, this paper is the first one to theoretically analyze the detection performance of WiFi downclocking. Our theoretical model characterizes the crucial impact of SRID attributes (e.g., tolerance threshold, correlation threshold, and energy ratio threshold) on the packet detection performance (i.e., the miss-detection probability and the false-alarm probability).

(ii) We run extensive Monte Carlo experiments to verify that our theoretical model is very accurate. We show that as the downclocked rate decreases, the false-alarm probability increases significantly, which will lead to a serious adverse impact on packet detection.

This study can help system developers set reasonable system parameters for WiFi downclocking.

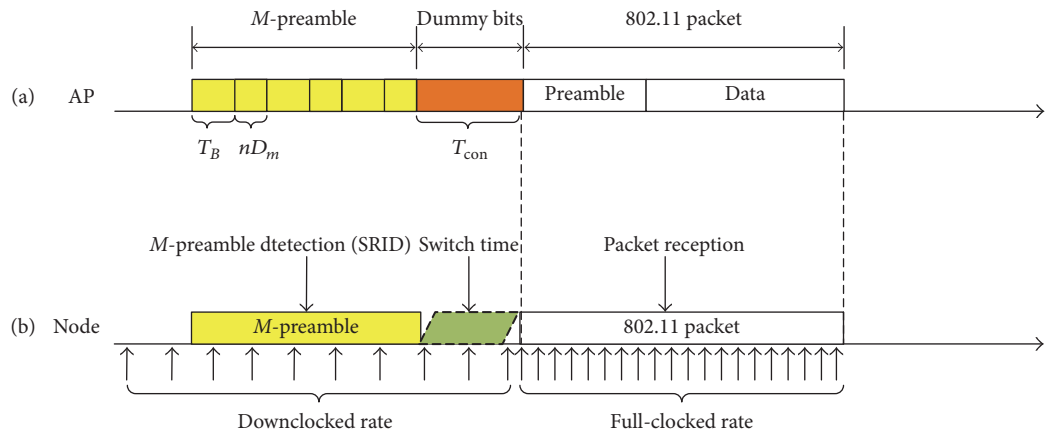

Figure 1: (a) AP's M-preamble transmission and (b) node's M-preamble detection in SRID.

So far, downclocking has received great attention [2–13]. Among the most relevant works, [2] is the first paper that brought downclocking to low-power WiFi networks and proposed the SRID algorithm, which was considered as one of the most classical amendments on power saving of 802.11 protocols [12, 13]. In WiFi, the dominant source of energy consumption is the idle listening operation [7, 8], where a node needs to frequently detect unpredictably arriving packets or assess a clear channel with high power. Therefore, SRID reduced the power consumption by allowing a WiFi node to downclock its sample rate in idle listening mode. SASD [3] was proposed to reduce the power consumption in SRID further by allowing nondestination nodes in idle listening to enter a doze state. AS-MAC [4] was proposed to avoid contention and reduce delay by asynchronously scheduling the wake-up time of neighboring nodes via a downclocking mechanism for wireless sensor networks. SloMo [5] was proposed to allow WiFi nodes to operate their radios at lower clock rates when receiving and transmitting at low bit rates. Sampleless [6] allowed energy-constrained devices to scale down their sampling rates regardless of channel conditions. The above works mainly focused on the hardware implementation of the downclocking mechanism or evaluated its performance via simulation. In contrast, this paper is the first one to model the impact of downclocking on the packet detection performance theoretically.

The rest of this paper is organized as follows. Section 2 gives an overview of SRID. Section 3 theoretically analyzes the detection performance of SRID. Section 4 presents Monte Carlo results that reveal the crucial impact of SRID attributes on the detection performance. Section 5 concludes this paper.

2. Overview of SRID

In the downclocking mechanism, one basic problem is how to detect unpredictably arriving packets at a downclocked rate, so that the node can revert to a full-clocked mode to receive the arriving packets.

The SRID that adopts the downclocking mechanism is designed for WiFi networks. With the help of Figure 1, we specify how SRID works. Assume a WiFi network consisting of one access point (AP) and a number of nodes. The AP is always in the active mode, while each node is in the downclocked mode by default. When the AP has a packet to transmit toward a node, the operations of the AP and the node are as follows.

(i) The AP first transmits an additional preamble called M-preamble, and then a sequence of dummy bits, and finally a conventional 802.11 packet. Here, the M-preamble is used to notify the node of the arrival of an expected packet. The dummy bits are used to provide a guard interval that allows the node to revert to the full-clocked mode from the downclocked mode.

(ii) The node continuously detects its M-preamble via self-correlation and then reverts to the full-clocked mode upon a successful detection.

In the next two subsections, we detail the construction and the detection of an M-preamble.

2.1. Construction of M-Preamble. In SRID, an M-preamble consists of C ($C \geq 2$) duplicated versions of a complex gold sequence (CGS), the length of each CGS sequence being $T_B + nD_m$. Figure 1(a) shows an example of M-preamble, where $C = 3$. Thus the total length of M-preamble T can be expressed as

$$T = C\left(T_B + nD_m\right), \tag{1}$$

where T_B represents the minimum length of the CGS (used for M-preamble). The integer n represents the address of the node in SRID, which is assigned by the AP. $1/D_m$ is the minimum downclocking factor of radio hardware. For example, assume that the full-clocked frequency is 20 MHz. Then the minimum downclocked frequency is $20 * (1/D_m)$ MHz.

2.2. Detection of M-Preamble. In SRID, a node continuously performs self-correlation to detect its M-preamble. Assume that a node operates with a downclocking factor of $1/D \in [1/D_m, 1]$.

Let $z(k)$ denote the sampling value of the node at the sampling point k.

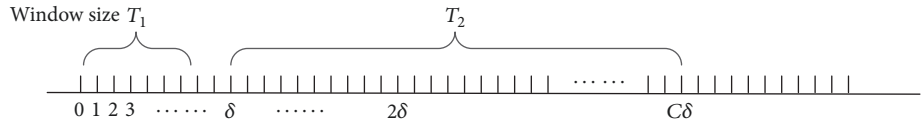

Figure 2: Illustration of C, δ T_1 and T_2.

Let $R(k)$ denote the self-correlation result of the node at the sampling point k. To detect its M-preamble, at each sampling point k, the node with address n performs the self-correlation between the latest T_1 samples and the previous T_1 samples (offset by δ). Therefore, $R(k)$ can be calculated by

$$R(k) = \sum_{i=k}^{k+T_1-1} z(i)\, z(i-\delta), \tag{2}$$

where $T_1 = T_B/D$ is the size of the self-correlation window (in sampling points) and $\delta = (T_B + nD_m)/D$ is the number of sampling points of a CGS when the downclocking factor is $1/D$. Note that T_1 and δ are shown in Figure 2.

Let $E(k)$ denote the energy level at sampling point k, which can be calculated by

$$E(k) = \sum_{i=k}^{k+T_1-1} |z(i)|^2. \tag{3}$$

We say that an M-preamble is successfully detected if the total number of successfully detected points, N_s, is greater than H_1T_2; namely,

$$N_s \geq H_1T_2, \quad H_1 \in (0,1), \tag{4}$$

where H_1 is the tolerance threshold and $T_2 = (C-1)\delta$ is the total number of sampling points (from the 2nd CGS to the C-th CGS), as shown in Figure 2.

We say that a sampling point is detected successfully if the following two conditions are satisfied.

Condition 1. At sampling point k, the correlation result $|R(k)|$ normalized by $E(k)$ is between H and $1/H$; namely,

$$H < \frac{|R(k)|}{E(k)} < H^{-1}, \quad H \in (0,1), \tag{5}$$

where $H \in (0,1)$ is a predefined threshold.

Condition 2. At sampling point k, the energy ratio (in dB) of $E_a(k)$ and $E_a(k - C\delta)$ exceeds a threshold H_s; namely,

$$10 \cdot \log_{10} \frac{E_a(k)}{E_a(k - C\delta)} \geq H_s, \tag{6}$$

where $E_a(k) = T_1^{-1}E(k) + (1 - T_1^{-1})E_a(k-1)$ represents a moving average of energy level, with a window size equal to T_1. The reason of introducing Condition 2 is to reduce the probability that Condition 1 is satisfied but no M-preambles are transmitted.

3. Detection Performance Analysis

In this section, focusing on the downlink traffic from the AP to nodes, we theoretically analyze the crucial impact of SRID attributes (namely, tolerance threshold H_1, correlation threshold H, and energy ratio threshold H_s) on the detection performance.

Due to the downclocked rate and the noise, each SRID detection result is associated with four mutually exclusive minievents: (a) successful detection: AP sends an M-preamble and the node detects it successfully, (b) miss-detection: AP sends an M-preamble but the node does not detect it, (c) false-alarm: AP does not send an M-preamble but the node detects it mistakenly, and (d) Null: AP does not send an M-preamble and the node detects nothing. To study the detection performance, we only need to calculate the successful detection probability P_d = Prob(successful detection) and the false-alarm probability P_{fa} = Prob(false-alarm), because Prob(miss-detection) = 1 − Prob(successful detection) and Prob(Null) = 1 − Prob(false-alarm).

We note that each detection result is determined depending on whether the AP sends an M-preamble. Below, we introduce two competing hypotheses:

$$\begin{aligned} &\mathcal{H}_0 : z(k) = n(k), \quad k = 0, 1, \ldots, C\delta \\ &\mathcal{H}_1 : z(k) = hx(k) + n(k), \quad k = 0, 1, \ldots, C\delta, \end{aligned} \tag{7}$$

where $\mathcal{H}_0$ is referred to as the null hypothesis (i.e., AP does not send an M-preamble to a node) and $\mathcal{H}_1$ as the alternative hypothesis (i.e., AP sends an M-preamble to a node). Under hypothesis $\mathcal{H}_0$, at the sampling point k, the node receives the noise, and therefore its sample value is $z(k) = n(k)$, where $n(k)$ is the Gaussian white noise. Under hypothesis $\mathcal{H}_1$, at the sampling point k, the node receives the M-preamble signal and the noise, and therefore its sampling value is $z(k) = hx(k) + n(k)$, where $x(k)$ represents the sampling value on the M-preamble and h represents the channel coefficient.

3.1. Expression of P_d. We now express P_d. According to (4), we have

$$P_d = P\{N_s \geq H_1T_2 \mid \mathcal{H}_1\}. \tag{8}$$

The sampling process is a Bernoulli process, where a sampling point is marked success if Conditions 1 and 2 (specified in Section 2.2) are satisfied. Therefore, the number of successfully detected points in T_2 trials, N_s, follows a binomial distribution. Thus P_d is expressed by

$$P_d = \sum_{i=H_1T_2}^{T_2} C_{T_2}^{i} \left(P_1P_{\text{ER1}}\right)^i \left(1 - P_1P_{\text{ER1}}\right)^{T_2-i}, \tag{9}$$

//Input: SNR, $C, T_B, nD_m, H_1, H, H_s, h, D, T, T_2$
//Output: P_d
//We run the code below for 100000 times.
(1) $i \longleftarrow 0$
(2) while ($i < 100000$)
(3) Generate a CGS randomly
(4) $x(0, 1, \ldots, T) \longleftarrow [\text{CGS}, \text{CGS}, \ldots, \text{CGS}]_{1\times C}$
(5) $y(0, 1, \ldots, T) \leftarrow \text{awgn}(x(1, \ldots, T), \text{SNR}, \text{“measured”})$
(6) $z(0, 1, \ldots, C\delta) \leftarrow$ Sample y every D points
(7) Calculate $R(0, 1, \ldots, C\delta)$ and $E(0, 1, \ldots, C\delta)$
(8) $N_1 \leftarrow$ the total number of sampling points that satisfy Condition 1.
(9) $P_1(i) \longleftarrow \dfrac{N_1}{C\delta}$
(10) $N_2 \leftarrow$ the total number of sampling points that satisfy Condition 2.
(11) $P_{\text{ER1}}(i) \longleftarrow \dfrac{N_2}{C\delta}$
(12) $i \longleftarrow i + 1$
(13) end
//We first calculate $\text{avg}(P_1)$ and $\text{avg}(P_{\text{ER1}})$, then P_d.
(14) $P_d \longleftarrow \sum_{i=H_1T_2}^{T_2} C_{T_2}^{i} [\text{avg}(P_1)\text{avg}(P_{\text{ER1}})]^i [1 - \text{avg}(P_1)\text{avg}(P_{\text{ER1}})]^{T_2-i}$

Algorithm 1: Calculation of P_d based on the Monte Carlo method.

where P_1 is the probability of Condition 1 being satisfied under $\mathscr{H}_1$ and P_{ER1} is the probability of Condition 2 being satisfied under $\mathscr{H}_1$.

Expression of P_1. According to (5), P_1 can be written as

$$P_1 = P(\mathscr{H}_1; \mathscr{H}_1) = P\left\{H < \frac{|R(k)|}{E(k)} < H^{-1} \mid \mathscr{H}_1\right\}. \quad (10)$$

In (10), $P(\mathscr{H}_i; \mathscr{H}_j)$ represents the probability of deciding $\mathscr{H}_i$ when $\mathscr{H}_j$ is true. Let U_1 denote the normalized correlation result at sampling point k under $\mathscr{H}_1$. Then U_1 can be expressed as $U_1 = \sum_{i=k}^{k+T_1-1} |(hx(i) + n(i))(hx(i-\delta) + n(i-\delta))| / \sum_{i=k}^{k+T_1-1} |hx(i) + n(i)|^2$. Thus P_1 is expressed by

$$P_1 = P\left\{H < U_1 < H^{-1}\right\}. \quad (11)$$

Note that U_1 is complicated, because it is a function of $2T_1$ random variables (i.e., $n(k-\delta), n(k+1-\delta), \ldots, n(k+T_1-1-\delta)$ and $n(k), n(k+1), \ldots, n(k+T_1-1)$). In Section 3.3, we calculate P_1 via the Monte Carlo method [14].

Expression of P_{ER1}. According to (6), P_{ER1} can be written as

$$P_{\text{ER1}} = P\left\{10 \cdot \log_{10} \frac{E_a(k)}{E_a(k - C\delta)} \geq H_s \mid \mathscr{H}_1\right\}. \quad (12)$$

Under $\mathscr{H}_1$, $E_a(k)$ and $E_a(k - C\delta)$ are expressed as follows.

$$\begin{aligned} E_a(k) &= T_1^{-1} E(k) + \left(1 - T_1^{-1}\right) E_a(k-1) \\ &= T_1^{-1} \sum_{i=k}^{k+T_1-1} |hx(i) + n(i)|^2 \\ &\quad + \left(1 - T_1^{-1}\right) E_a(k-1) \\ E_a(k - C\delta) &= T_1^{-1} E(k - C\delta) \\ &\quad + \left(1 - T_1^{-1}\right) E_a(k - C\delta - 1) \\ &= T_1^{-1} \sum_{i=k}^{k+T_1-1} |n(i)|^2 \\ &\quad + \left(1 - T_1^{-1}\right) E_a(k-1). \end{aligned} \quad (13)$$

Note that $E(k - C\delta) = \sum_{i=k}^{k+T_1-1} |n(i)|^2$, because AP transmits one M-preamble (which consists of $C\delta$ sampling points only) for each packet, and thereby the node only receives the noise before the M-preamble. Similar to P_1, we can calculate P_{ER1} via the Monte Carlo method.

3.2. Expression of P_{fa}. We now express P_{fa}. Similar to P_d, P_{fa} can be expressed as follows:

$$\begin{aligned} P_{\text{fa}} &= P\{N_s \geq H_1T_2 \mid \mathscr{H}_0\} \\ &= \sum_{i=H_1T_2}^{T_2} C_{T_2}^{i} (P_2P_{\text{ER2}})^i (1 - P_2P_{\text{ER2}})^{T_2-i}, \end{aligned} \quad (14)$$

where $P_2 = P(\mathscr{H}_1; \mathscr{H}_0) = P\{H < |R(k)|/E(k) < H^{-1} \mid \mathscr{H}_0\}$ is the probability of Condition 1 being satisfied under $\mathscr{H}_0$ and $P_{\text{ER2}} = P\{10 \cdot \log_{10}(E_a(k)/E_a(k - C\delta)) \geq H_s \mid \mathscr{H}_0\}$ is the probability of Condition 2 being satisfied under $\mathscr{H}_0$.

3.3. Calculation of P_d *and* P_{fa} *via Monte Carlo Method.* In the previous two subsections, we give expressions of P_d and P_{fa}. However, they involve $2T_1$ random variables and therefore are hard to solve. Below we adopt the Monte Carlo method [14] to calculate them. Algorithm 1 lists the computation process in

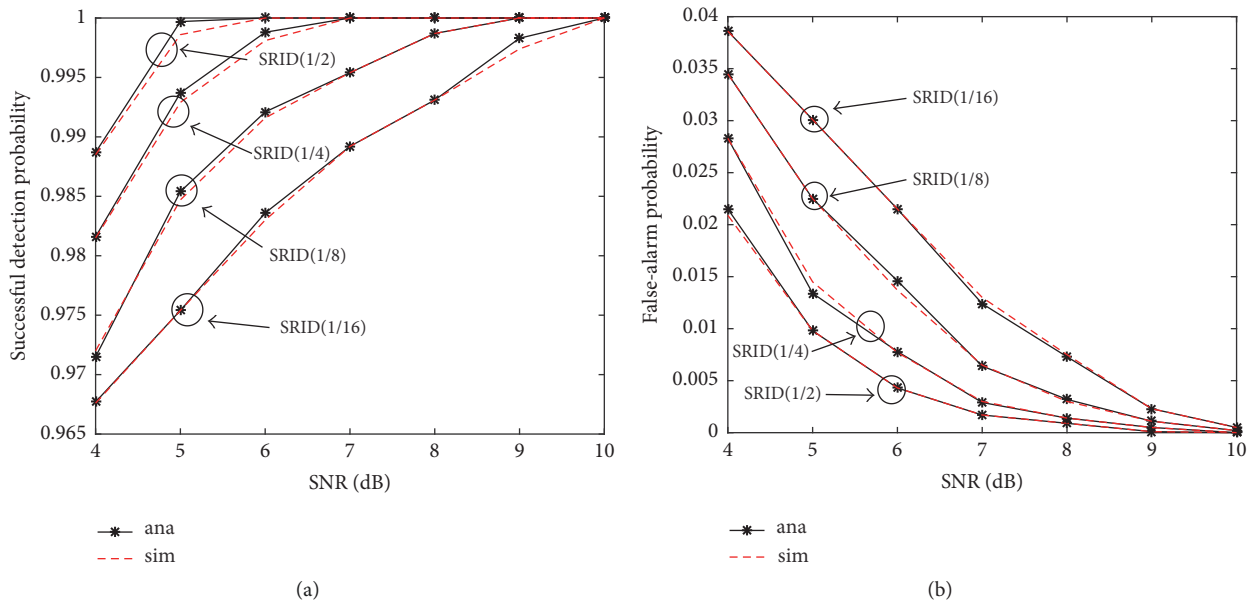

Figure 3: (a) P_d and (b) P_{fa} vary when the SNR varies.

terms of P_d, which is given by (9). Similarly, we can calculate P_{fa}.

In Algorithm 1, we input the SRID parameters and output the value of P_d. In the algorithm, we run Monte Carlo experiment for 100000 times. We now detail each experiment.

(i) In lines (3) to (4), we generate an M-preamble of T samples points, which simulates the AP's M-preamble transmission.

(ii) In line (5), we invoke the Matlab function, awgn(·), to simulate the additive white Gaussian noise (AWGN) channel and then the node's received signal is the result that the AP's M-preamble signal passes through the AWGN channel.

(iii) In line (6), we obtain the downclocked sampling sequence $z(\cdot)$ under the downclocking factor of $1/D$.

(iv) In line (7), we calculate self-correlation result $R(\cdot)$ and the energy level $E(\cdot)$.

(v) In lines (8) to (9), we calculate P_1 in this experiment.

(vi) In lines (10) to (11), we calculate P_{ER1} in this experiment.

Finally, after we finish 100000 runs, we first calculate the average of P_1, avg(P_1), and the average of P_{ER1}, avg(P_{ER1}), and then calculate P_d, as shown in line (14).

4. Model Verification

In this section, we present the Monte Carlo results to illustrate the crucial impact of SRID attributes and SNR on the detection performance (namely, the successful detection probability P_d and the false-alarm probability P_{fa}). The default parameter settings are set by [2] and are shown in Table 1.

Table 1: Parameter settings in simulation.

Parameters	Description	Values
C	Number of CGS	3
T_B	Basic length	64 sampling points
nD_m	Additional length	64 sampling points
H_1	Tolerance threshold	0.6
H	Correlation threshold	0.9
H_s	Energy ratio threshold	4 dB
SNR	Signal-to-noise ratio	9 dB
h	Channel coefficient	1

In Figures 3 and 4, each Monte Carlo result is on average over 100000 runs. In addition, we use "SRID($1/D$)" to denote the SRID detection with the downclocking factor of $1/D$. In all figures, the labels "ana" and "sim", respectively, denote the theoretical and simulation results.

Figures 3(a) and 3(b), respectively, plot P_d and P_{fa} as the SNR varies, when $1/D = 1/2, 1/4, 1/8, 1/16$. From Figure 3, we have the following observations.

(i) Given $1/D$, P_d increases and P_{fa} decreases gradually as the SNR increases.

(ii) Given SNR, as $1/D$ decreases, P_d decreases slightly while P_{fa} increases significantly. For example, for SNR = 5 dB, when $1/D$ decreases from 1/2 to 1/16, P_{fa} grows from 0.0098 to 0.0301 significantly, while P_d just drops from 0.9997 to 0.9754 slightly. This will lead to a serious adverse impact on packet detection.

(iii) The detection performance is almost perfect (i.e., $P_d = 1$ and $P_{\text{fa}} = 0$) when SNR = 10 dB, which is easy to achieve in real environments [15].

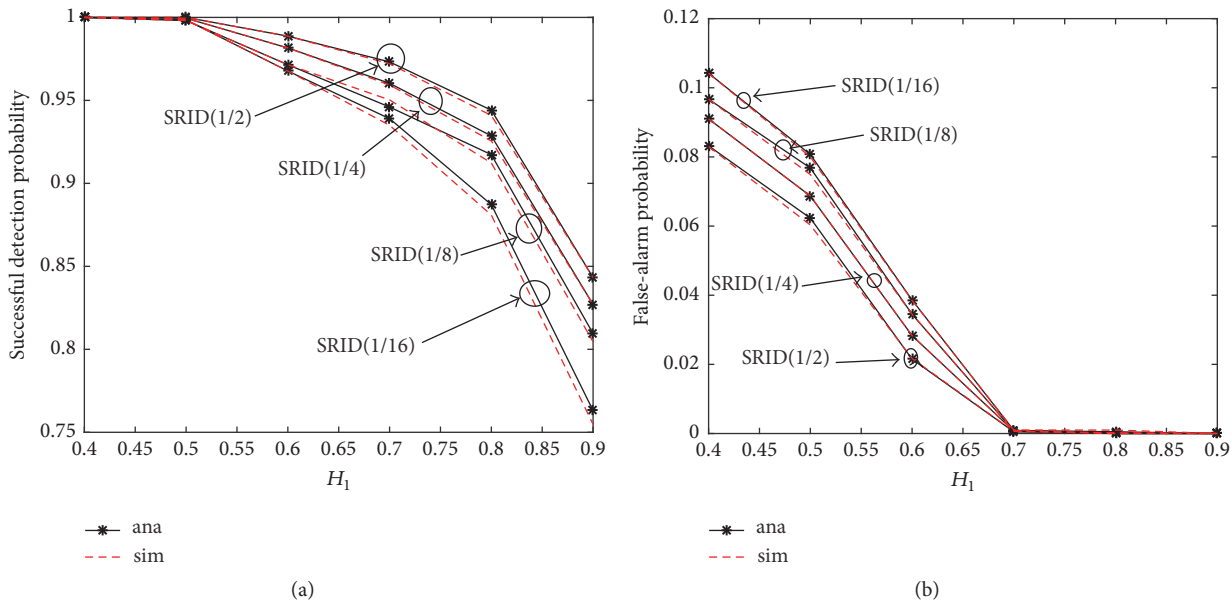

FIGURE 4: (a) P_d and (b) P_{fa} vary when H_1 varies.

Figures 4(a) and 4(b), respectively, plot P_d and P_{fa} as H_1 varies, when $1/D = 1/2, 1/4, 1/8, 1/16$. From Figure 4, we have the following observations.

(i) Given $1/D$, as H_1 increases, P_d always decreases, but P_{fa} first decreases to 0 and then remains unchanged. The reason is as follows: increasing H_1 will decrease the successful detection probability from (9) as well as the false-alarm probability from (14).

(ii) Give H_1, as $1/D$ decreases, P_d decreases significantly, while P_{fa} decreases gradually.

Finally, from these figures, the close match between the theoretical and simulation curves manifests that our performance model is very accurate.

5. Conclusion

In mobile edge computing, various battery-powered mobile nodes desire to acquire information technology services at the network edge under power saving. WiFi downclocking is such a promising technique. In this paper, we investigate a novel WiFi downclocking technique called SRID and first theoretically study the impact of SRID attributes (namely, tolerance threshold, correlation threshold, and energy ratio threshold) on the detection performance of packet arrival. This study is helpful in designing better WiFi downclocking protocols.

Disclosure

Qinglin Zhao is the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the Macao FDCT-MOST Grant 001/2015/AMJ, Macao FDCT Grants 056/2017/A2 and 005/2016/A1, National Science Foundation of China (61672500), and Program of International S&T Cooperation (2016YFE0121500).

References

[1] https://en.wikipedia.org/wiki/Mobile_edge_computing.

[2] X. Zhang and K. G. Shin, "E-mili: energy-minimizing idle listening in wireless networks," *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1441–1454, 2012.

[3] T. Xiong, J. Yao, J. Zhang, and W. Lou, "It Can Drain Out Your Energy: An Energy-Saving Mechanism Against Packet Overhearing in High Traffic Wireless LANs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1911–1925, 2017.

[4] B. Jang, J. B. Lim, and M. L. Sichitiu, "An asynchronous scheduled MAC protocol for wireless sensor networks," *Computer Networks*, vol. 57, no. 1, pp. 85–98, 2013.

[5] Lu F., Voelker G. M., Snoeren A. C. SloMo: downclockingWiFi communication[C]// Usenix Conference on Networked Systems Design and Implementation. 2013:255-268.

[6] W. Wang, Y. Chen, L. Wang, and Q. Zhang, "Sampleless Wi-Fi: Bringing low power to Wi-Fi communications," *IEEE/ACM Transactions on Networking*, vol. 25, no. 3, pp. 1663–1672, 2017.

[7] Y. Agarwal, R. Chandra, A. Wolman, P. Bahl, K. Chin, and R. Gupta, "Wireless wakeups revisited: energy management for VoIP over Wi-Fi smartphones," in *Proceedings of the 5th International Conference on Mobile Systems, Applications and Services (MobiSys '07)*, pp. 179–191, June 2007.

[8] G. Anastasi, M. Conti, E. Gregori, and A. Passarella, "802.11 power-saving mode for mobile computing in Wi-Fi hotspots: limitations, enhancements and open issues," *Wireless Networks*, vol. 14, no. 6, pp. 745–768, 2008.

[9] P. Serrano, A. De La Oliva, P. Patras, V. Mancuso, and A. Banchs, "Greening wireless communications: Status and future directions," *Computer Communications*, vol. 35, no. 14, pp. 1651–1661, 2012.

[10] X. Zhang and K. G. Shin, "Gap Sense: lightweight coordination of heterogeneous wireless devices," in *Proceedings of the 32nd IEEE Conference on Computer Communications (INFOCOM '13)*, pp. 3093–3101, April 2013.

[11] W. R. Dieter, S. Datta, and W. K. Kai, "Power reduction by varying sampling rate," in *Proceedings of the 2005 International Symposium on Low Power Electronics and Design*, pp. 227–232, usa, August 2005.

[12] Y. Cui, X. Ma, H. Wang, I. Stojmenovic, and J. Liu, "A survey of energy efficient wireless transmission and modeling in mobile cloud computing," *Mobile Networks and Applications*, vol. 18, no. 1, pp. 148–155, 2013.

[13] H. A. Omar, K. Abboud, N. Cheng, K. R. Malekshan, A. T. Gamage, and W. Zhuang, "A Survey on High Efficiency Wireless Local Area Networks: Next Generation WiFi," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2315–2344, 2016.

[14] https://en.wikipedia.org/wiki/Monte_Carlo_method.

[15] J. T. Bevan et al., "An Integrated 802.11a Baseband and MAC Processor," in *IEEE ISSCC Digest*, 2002.