Diagnosing reasons of packet loss in 802.11 networks

Yiming Wang[1,a], Zaixue Wei[1,b], Hongwen Yang[1,c]

[1]Beijing University of Posts and Telecommunications, Beijing 100876, China

Email: [a]2012213082@bupt.edu.cn,[b]zaixuew@bupt.edu.cn,[c]yanghong@bupt.edu.cn

Keywords: wireless;802.11; packet loss; collision; interference;

Abstract. Packet loss in the 802.11 network can be attributed to many reasons. Detecting what causes packet loss is important for network optimization and is still a tough question. In recent years, various models have been proposed in literature to deal with this problem. In this paper, we introduce and compare some of these existing methods including their parameter, topology, algorithm, performance and etc.

Introduction

With the wide use of cell phones and wireless end devices all over the world, traditional wired network cannot satisfy people's need. Thus technologies of wireless communication are developed and the standard IEEE 802.11 is established. Nowadays, Wi-Fi of 802.11 is implementedin entire city, home, work place, café shopand etc.[1-4].However, compared to the wired communication systems, thepacket loss in an 802.11 network is more frequent and severe which may degrade the quality of the service considerably.

Causes of packet loss in 802.11 network are diverse and each may require a dedicated solution. The common causes includes collision, congestion, channel fading, interference and etc. To effectively overcome the problem, it is desirable to develop algorithms to accurately distinguish the exact cause of packet loss in an802.11 network.

To this end, many researches have been carried out and many methods have been proposed. In this paper, we introduce some of these methods including TCP E2E [5] which use time interval between sending packet and receiving ACK to distinguish congestion and wireless loss, FGA [6] that use RSSI, LQI PRR to identify interference and attack, COLLIE [7]which separate collision and weak signal using various metrics, ECODA [8]which uses dual threshold method to evaluate the condition of congestion, EVM [9] which is a concept used to classify channel error and collision.

TCP E2E

TCP E2E is proposed by Parag Kulkarni, Mahesh Sooriyabandara, Lu Li from Toshiba Research Europe Ltd. Bristol. It is a modified model of TCP Reno and they try to use this model to distinguish packet losses due to congestion and wireless/link loss. The authors regard duplicate ACK as a signal of packet loss. The topology of this model is shown in Fig. 1 and TCP E2E works entirely at the sender side so it does not need extras support from other nodes. In Fig. 1, d_{p1}, d_{p2}, $d_{p3}$and d_{p4}represent the propagation delay of each links, d_{q1}, $d_{q2}$and d_{q3}denote the queuing delay along end-to-end path. This model only cares about the time interval between sending the packet and receiving it which can be denoted asOWD, so we can find$OWD = \sum d_{pi} + \sum d_{qi}$.

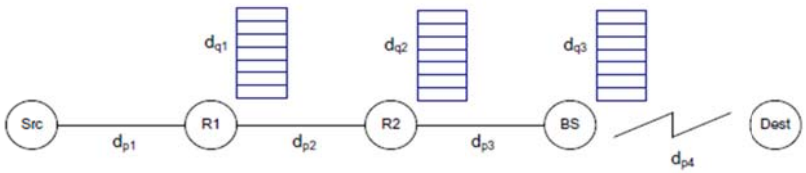

Fig. 1 simple topology of TCP E2E

When there is no congestion $OWD = \sum d_{qi}$. Beside this, TCP E2E sets a small margin called *toleranceMargin* to represent the variability of propagation delay. Therefore, when the time interval of an ACK exceeds $delayThreshold = OWD_{min} + toleranceMargin$, it is suggested that there is congestion in some links. Otherwise, the duplicated ACK is due to wireless/link loss. The algorithm is shown in Fig. 2. In [5], the authors conducted three scenarios to evaluate their model: only wireless loss, no congestion loss; only congestion loss, no wireless loss; both congestion loss and wireless loss. The throughput and delay performance are reproduce in Fig. 3.

Algorithm 1 E2E Algorithm

$OWD_{min} = INFINITY$

On Every ACK Arrival do the following:
measure OWD
if First duplicate ACK **then**
  This means a loss has occurred
  if $measuredOWD < delayThreshold$ **then**
    This loss is likely to be a **wireless** loss. Do not initiate the TCP back-off procedure.
  else
    This loss is likely due to **congestion**. Initiate the standard TCP back-off procedure.
  end if
end if

if $(measuredOWD < OWD_{min})$ **then**
  $OWD_{min} = measuredOWD$
  $toleranceMargin = 0.05 * OWD_{min}$ (/* $toleranceMargin$ chosen to be 5% of OWD_{min}*/)
  $delayThreshold = OWD_{min} + toleranceMargin$
end if

Fig. 2 Algorithm of TCP E2E

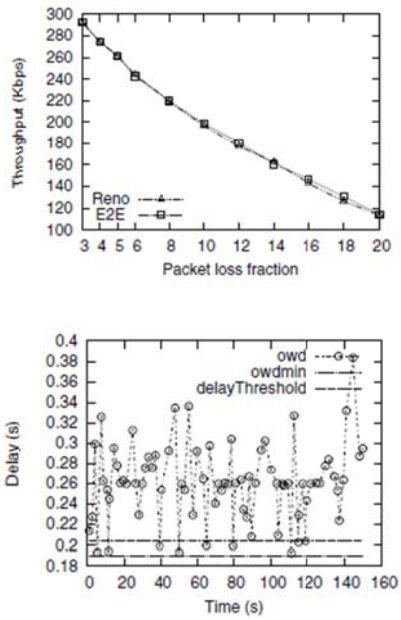

Fig. 3 Performance of TCP E2E

Fine-Grained Analysis (FGA)

This method is proposed by Bilal Shebaro, Daniele Midi, Elisa Bertino from Purdue University in 2014. They use parameters like signal strength indicator (RSSI), link quality indicator (LQI) and packet reception rate (PRR) to classify four causes of packet losses: low radio interference, high radio interference, selective forwarding attack and blackhole attack. The analysis of FGA is event-driven and is carried out simultaneously at every node. The analysis is conducted in stealth so that it gives no chances for malicious nodes to interfere. It is assumed that every node has at least two paths to base station.

At the set-up stage, in order to know all the nodes, BS sends HELLO command to each node and asks them to start link-profiling process. Each node broadcasts M dummy messages to all of its neighbors. Then each node can create a profile consisting of averaged RSSI, LQI and PRR for each neighbor. This process is shown in Fig. 4. The comparison algorithm is shown in Fig. 5. PRR_{curr} is the current PRR of this node and PRR_{thres} is the value that differentiates between partial and total packet losses. $\Delta RSSI$ and ΔLQI is the difference between the initial and current value of RSSI and LQI. $Interf_{thres}^{RSSI}$ and $Interf_{thres}^{LQI}$ are the interference threshold which serve as the minimal difference that can determine the exact cause. When $PRR_{curr} < PRR_{thres}$, the comparison algorithm need re-profile before compare parameters.

When IDS of node n discovering packet loss from n_{bad}, its FGA is triggered. This node will snoop packets from n_{bad} and record corresponding value of RSSI and LQI in an array $R\_bad[]$ in the format of $<n_{bad}, NodeID[], Vote, TC>$. TC is the time when one of the investigating nodes fills up $R\_bad[]$, calculate PRR and make a decision. $Vote$ is the decision made for the cause of packet loss. After one of the investigation node broadcast information with this array, all other nodes stop for stealthy

matter. These other nodes will compute their own votes, aggregate with other IDs' votes received and broadcast their own array. The process will stop when exceeding the preset timeout. The voting aggregation algorithm is based on Table 1.

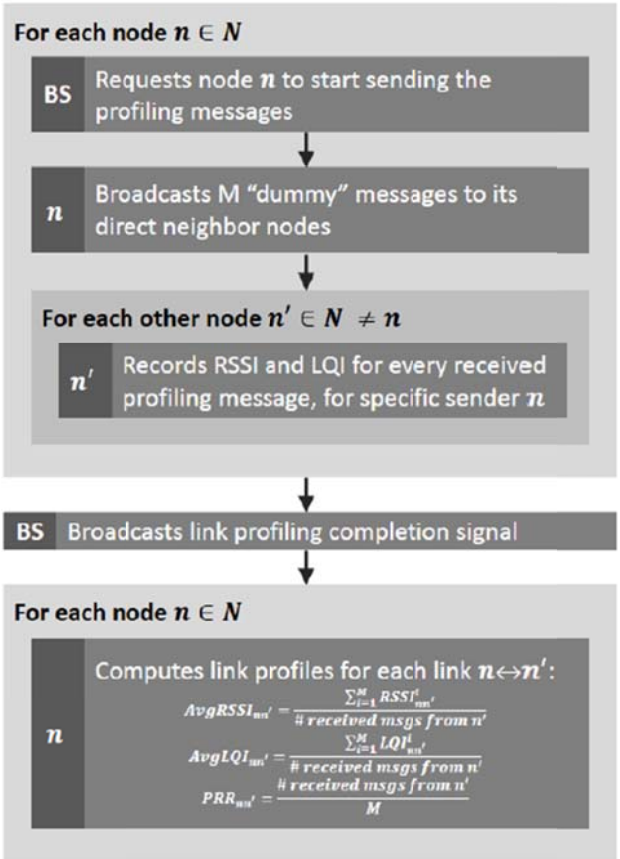

Fig. 4 Profiling steps at set-up stage

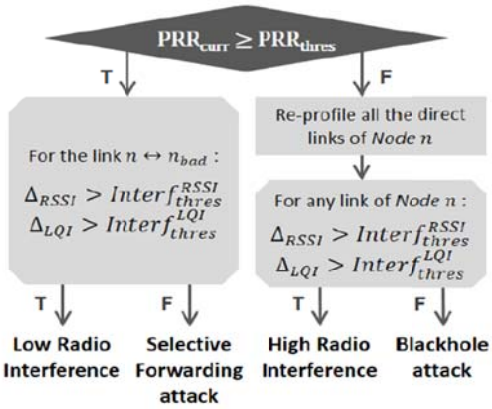

Fig. 5 Vote Aggregation Algorithm

Table 1 Voting Aggregation Principle

Cause of Packet Loss	Voting Aggregation Principle
High Radio Interference	If at least one node votes for High Radio Interference, the aggregated vote is High Radio Interference
Low Radio Interference	If at least one node votes for Low Radio Interference and none of the other nodes votes for High Radio Interference, the aggregated vote is Low Radio Interference
Selective Forwarding Attack	If at least one node votes for Selective Forwarding and none of the other nodes votes for any type of interference in the network medium, the aggregated vote is Selective Forwarding
Blackhole Attack	If at least one node votes for Blackhole when none of other investigating nodes votes for selective forwarding or any type of interference in the network medium, the aggregated vote is Blackhole

Ref. [6] has conducted seven experiments for different cause of packet loss and the setting and process are similar. We will list two of them. The topology is shown in Fig. 6 where node 5 sends packet to node 1, base station, through node 2. Node 3, 4, 6, 7 are direct neighbors of node 2 and thus monitor its behavior.

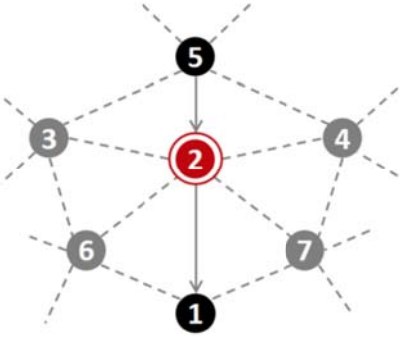

Fig. 6 Topology of FGA

- Experiment 1: Selective forwarding attack is actual cause.

In this experiment, node 2 will drop packets by10% and node 3, 4, 6, 7 successfully detect packet drop and trigger FGA. The final vote aggregation reported it is a "Selective Forwarding" attack.

Vote for node 3:*<2, [3], SEL_FWD, TC>*
Vote for node 4:*<2, [4], SEL_FWD, TC>*
Vote for node 6:*<2, [6], SEL_FWD, TC>*
Vote for node 7:*<2, [7], SEL_FWD, TC>*
Aggregated vote:*<2, [3, 4, 6, 7], SEL_FWD, TC>*

- Experiment 2: Low interference attack is actual cause

In this case, an interferer is placed near node 2 which can isolate node 2 completely. Node 3, 4, 5, 6 and 7 successfully discover the packet loss and the final aggregated result is "Low Interference".

Vote for node 3:*<2, [3], LOW_INTF, TC>*
Vote for node 4:*<2, [4], SEL_FWD, TC>*
Vote for node 5:*<2, [5], LOW_INTF, TC>*
Vote for node 6:*<2, [6], LOW_INTF, TC>*
Vote for node 7:*<2, [7], SEL_FWD, TC>*
Aggregated vote:*<2, [3, 4, 5, 6, 7], SEL_FWD, TC>*

The result of experiments proves FGA can differentiate these four causes of packet loss very well. Other experiments can be found in [6]which show that if the cause of packet loss is interference, FGA is capable to locate the source of interference.

Collision Inferencing Engine (COLLIE)

This model is proposed by Shravan Rayanchu, Arunesh Mishra, Dheeraj Agrawal, Sharad Saha and Suman Banerjee from University of Wisconsin Madison. They use this model to perform diagnosis of collision and weak signal. For collision, Binary-Exponential Backoff (BEB) algorithm will be carried out and for weak signal adaptation of data-rate and transmit power will be utilized. Their model has three components which is shown in Fig 7. (1) Client module. It implements logic to analyze the cause of packet loss. (2) AP. When discerning packet loss, AP will relay the entire packet to client module for analysis. (3) Server. It is optional and it is used to facilitate multi-AP collision detection.They conducted some empirical analysis of a set of metric including received signal strength (RSS), bit-error rate (BER), symbol-error rate (SER), error-per symbol (EPS), symbol error score (S-Score) and joint distribution of SER and EPS. For symbol-error rate, the difference is not obvious, but for other metrics, we take RSS and BER as examples which result are shown in Fig. 8, there is a threshold can be find that can distinguish weak signal and collision.

The basic rule is that if one of the above metrics vote for collision, then the cause will be regarded as collision. The experiment result is shown in Table 2 which indicates that the accuracy is near 60% and false positive rate is about 2%.

The accuracy can be improved by using multi-AP. This is implemented by aggregating result at server. The COLLIE server implements a simple detection algorithm that use information of packets received in error and combine with data-rate of packet received to make the decision.

Four experiments have been carried out in [7] including static scenario, additional collision sources and mobile scenario. The result of static scenario can be seen in Fig. 9. We can observe that throughput increase around 30%.They also conducted an experiment to emulate a voice call, and found that COLLIE can reduce retransmission related costs by 40%.

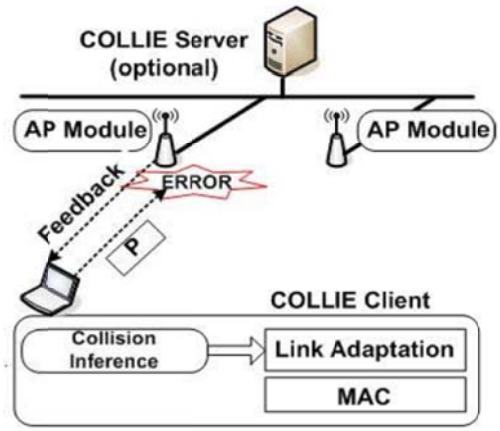

Fig. 7 Components of COLLIE

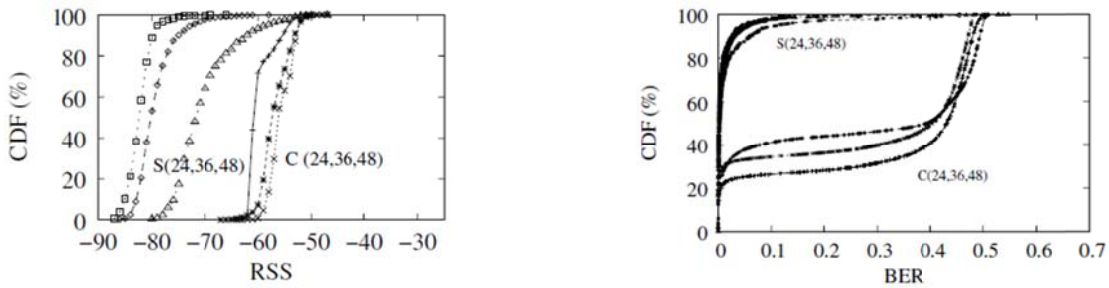

Fig. 8 CDF of various metrics

Table 2 COLLIE accuracy and false positive rates

	BER	EPS	S-Score	Metric-Vote
Accuracy	0.530	0.524	0.441	0.597
False Positives	0.0057	0.022	0.0126	0.024

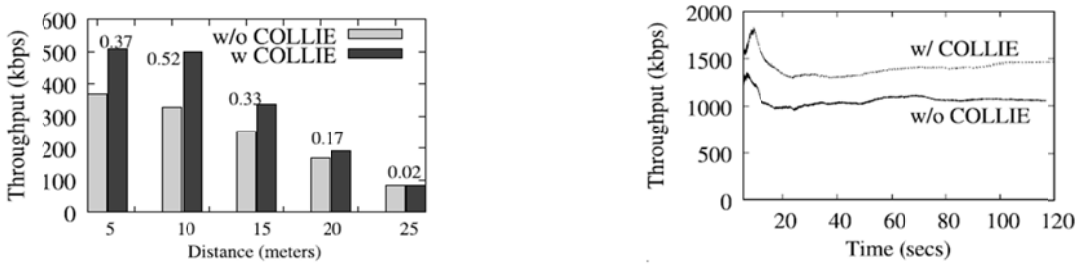

Fig. 9 Experiment Results of Static Scenario of COLLIE

Enhanced Congestion Detection and Avoidance (ECODA)

This method is proposed by Liqiang Tao and Fengqi Yu from the Chinese University of Hong Kong. It uses dual thresholds to weigh difference for congestion detection and ECODA can dynamically estimate channel loading and optimize channel utilization. It defines the congestion into three status: low channel loading and low throughput,S_0, high channel loading and high throughput,S_1, and high channel loading and low throughput which isS_2. Four actions have been defined, namely additively increase data sending rate,B_0, additively decrease data sending rate,B_1, linearly decrease data sending rate,B_2, and medium data sending rate,B_3. The average number of dropped packets because of collision is$\eta_d$and total number of successfully transmitted packets with one or more transmission-attempts isη_s. Throughput$Th(\Delta t)$ equals$\frac{\eta_s(\Delta t)}{\Delta t}$and the number of packet retransmission is$\eta_r(\Delta t) = p*\eta_r(\Delta t)*rl$ whererlequals the number of retransmission for a successfully transmitted packet or the maximum allowed number of retransmission. The algorithm is shown in Fig. 10. In [8], a simulation using NS2 is carried out and theresults are reproduced in Fig.11.

Input: $thresh_1$,$thresh_2$,N

Notation:
$thresh_1$:lower retransmission to determine channel idle;
$thresh_2$:upper retransmission to determine channel congestion;
N:sufficient packets to evaluate channel loading .

Goal: Optimizing channel working in status S_1

1.Check time interval $\triangle t$ and the number of transmissions Tx in $\triangle t$;

2. Compute throughput and the number of retransmissions using formula (5) and (6);

3. If ($\eta_r(\triangle t) \geq 0$ && $\eta_r(\triangle t) < thresh_1$) {
  If ($Tx \geq N$) action='B_0'; status= {S_0, $S_0 \rightarrow S_1$};
  else action='B_3';
}

4. If ($\eta_r(\triangle t) \geqslant thresh_1$ && $\eta_r(\triangle t) < thresh_2$) {
  If ($\eta_r(\triangle t) < \eta_r(t-\triangle t)$ && ($Th(\triangle t) > Th(t-\triangle t)$)
    action='B_0'; status= {$S_0 \rightarrow S_1$};
  If ($\eta_r(\triangle t) \geqslant \eta_r(t-\triangle t)$ && $Th(\triangle t) > Th(t-\triangle t)$)
    action='B_1'; status= {S_1->S_2};
  If ($\eta_r(\triangle t) \geqslant \eta_r(t-\triangle t)$&& $Th(\triangle t) \leq Th(t-\triangle t)$)
    action='B_2'; status= {S_2}; }

5. If ($\eta_r(\triangle t) \geqslant$ thresh$_2$) action='B_2'; status= {S_2};

Fig. 10 Algorithm of ECODA

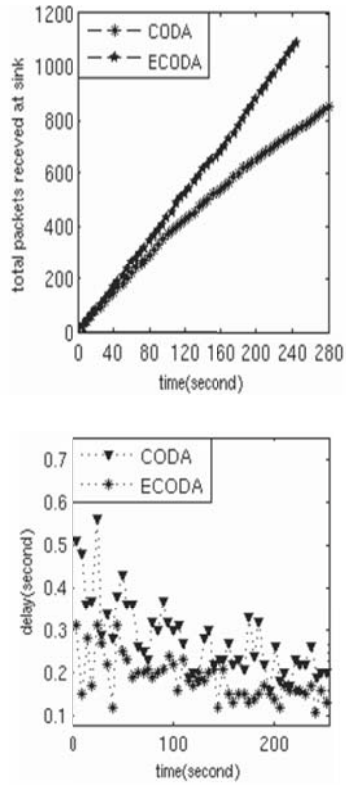

Fig. 11 Performance of ECODA

Error Vector Magnitude (EVM)

This model is proposed by Muhammad Naveed Aman and Biplab Sikdar from Rensselaer Polytechnic Institute. They use this model to distinguish the packet losses due to collision and channel errors. The scenario is simple and is shown in Fig. 12. d_1, $d_2$are distance from T_1, $T_2$to base station and they can be changed. T_1 sends 32 OFDM symbols per frame while the frame size for $T_2$is variable. They use quadrature and in-phase to describe a signal and they define error vector as the difference between complex voltage value of the ideal symbol I_n and the actual received symbolR_n. The root-mean-square of error vector is EVM. Error vector is defined as $E_n = R_n - I_n$ and EVM is defined as:

$$EVM_{RMS} = \sqrt{\frac{\frac{1}{T}\sum_{n=0}^{T-1}|R_n - I_n|^2}{P_0}} = \sqrt{\frac{\frac{1}{T}\sum_{n=0}^{T-1}|E_n|^2}{P_0}} \quad (1)$$

where P_0 is the average power of all symbols for a given modulation and T is the number of received symbols. Through experiments they find that EVM of packets involved in collision will be higher than those due to channel errors, so they decide to collect data to find a threshold of EVM, γ, to differentiate these two causes. They use classification and regression tree (CART) to find this value by placing transmitters at various distance to obtain data and analyze a proper threshold. If EVM exceeds this threshold, then the packet loss will be treated as collision, otherwise packet loss is due to channel errors. The authors carry out two simulations, one under low capture effect and the other under high capture effect. They consider three parameters to evaluate their simulation: (i) false alarm P_{FA}- when channel error miss-matched as collision; (ii) probability of miss-detected P_{MD} - when collision miss-matched as channel errors; (iii) accuracy – the number of cases classified correctly. We will list the simulation result of the case under low capture effect when transmitters are placed at an equal distance from the receiver. By collecting data and training CART model, they decide to use 15.5 dB as the threshold to classify the packet loss. The scatterplot shows that the accuracy is about 97%. Moreover, P_{FA} is 3.1% and P_{MD} is 2.8% proving the correctness of this model.

Comparison and Summary

In this paper, we list four models that have been proposed to differentiate different causes of packet loss. They consider from various aspects and select various metrics to verify their methods. We make summary and comparison of these models in Table 3.

Table 3 Comparison of Different Methods

	TCP-E2E	FGA	COLLIE	ECODA	EVM
Kinds of Causes Considered	Congestion; Wireless or link loss	Low radio interference; High radio interference; Selective forwarding attack; Blackhole attack.	Collision Weak signal	Weigh difference for congestion detection	Channel Error Congestion
Metric	Transmission delay Propagation delay	RSSI LQI PRR	RSSI BER EPS S-Score Joint distribution of SER and EPS	Average number of packet loss Total number of successfully transmitted packet	Error Vector Magnitute
Advantages	Require change only at sender side Remain TCP friendly	Event driven Multiple simultaneous investigation Stealthy manner	Combination of various metrics Without requiring extra transmission from client	Dual buffer Dynamically estimates channel loading Provide fairness to different class of traffic	Direct Real Time data from received packets
Disadvantage	Only when variation is small		Requiring knowing the information of packets		

References

[1] T. Schmidt, and A. Townsend: Wireless networking security: Why Wi-Fi wants to be free. Communications of the ACM, Vol. 46, no. 5(2003), p. 47-52

[2] A. Balachandran, G.M. Voelker, and P. Bahl: Wireless hotspots: currentchallenges and future directions (Kluwer Academic Publishers, 2005).

[3] M. Balazinska , and P. Castro: Characterizing Mobility and NetworkUsage in a Corporate Wireless Local-Area Network (Proceedings of MobiSys 2003: 1st International Conference on Mobile Systems, Applications, and Services).

[4] N. Gupta: Grande Wi-Fi: understanding what Wi-Fi users are doing incoffee shops (M. Sc Comparative Media Studies, Comparative MediaStudies, Massachusetts Institute of Technology, 2004).issues in wireless sensor networks," IEEE Communications Surveys and Tutorials, vol. 8, p. 2–23.

[5] Parag Kulkarni, Mahesh Sooriyabandara and Lu Li: Improving TCP Performance in Wireless Networks by classifying causes of packet losses. IEEE Wireless Communications and Networking Conference (WCNC) 2009, p. 1-6

[6] Bilal Shebaro, Daniele Midi and Elisa Bertino: Fine-Grained Analysis of Packet Losses in Wireless Sensor Networks. 2014 Eleventh Annual IEEE International Conference on Sensing, Communication, and Networking (SECON), p. 320-328

[7] Shravan Rayanchu, Arunesh Mishra, Dheeraj Agrawal, Sharad Saha and Suman Banerjee: Diagnosing Wireless Packet Losses in 802.11: Separating Collision from Weak Signal. IEEE INFOCOM 2008. The 27th Conference on Computer Communications, p. 1409-1417

[8] Liqiang Tao and Fengqi Yu: A Novel Congestion Detection and Avoidance Algorithm for Multiple Class of Traffic in Sensor Network. IEEE Cyber Technology in Automation, Control, and Intelligent Systems (CYBER) 2011, p. 72-77

[9] Muhammad Naveed Aman and Biplab Sikdar: Distinguishing Between Channel Errors and Collisions in IEEE 802.11. IEEE Information Sciences and Systems (CISS) 2012, p. 1-6