Packet Status Prediction Model Using Packet Sniffer Data

Mariana Ávalos Arce
0197495@up.edu.mx
Universidad Panamericana

Daniel Heráclito Pérez Díaz
0190575@up.edu.mx
Universidad Panamericana

***Abstract*—We propose a prediction model for the packet status of a packet that travels through a wireless IEEE 802.15.4 protocol network, obtained from a network exposed to 5 different types of interference in a controlled-experimentation environment. Data is retrieved with a single CC2531 USB Dongle Packet Sniffer, whose information on packets become the features of each packet from which the classifier model will gather the training data with the objective of predicting whether or not a packet will arrive to its destination as an error packet. The classifier model exhibits 76% accuracy measured with the Area Under the Curve (AUC) performance metric. The classifier model is then a good candidate for the prediction of a packet's status which can later be used to assess the analysis of the reasons behind packet loss in wireless networks.**

***Index Terms*—Packet loss, packet sniffer data, binary classification, IEEE 802.15.4 protocol.**

I. Introduction

Nowadays, Wireless Networks are used in industrial automation, vehicles, remote surgery, robots, mobile sensor networks vehicles, and Internet-based applications. A **Wireless Network** is then a connection of two or more nodes where information is sent or received (or both) without the use of a **physical cable connector** between the nodes.

A **packet** is a small segment of a larger message. Data sent over **a network** is divided into packets, and the destination node recombines the packets that form a single data piece when the packets are received. Theoretically, it could be possible to send data over a network without chopping them down into small packets of information. However, such an approach becomes impractical when more than two nodes are involved in the network: whenever any long line of bits gets passed over the network involving two nodes, the other nodes would need to **wait** for this communication to finish so that they can make use of that channel [7]. In other words, packet division for data makes a network able to exchange billions of files instead of just a few. It also means that **packets can take different network paths to the same destination**, as long as they all arrive at their destination.

This management of information as independent pieces that travel in *any order* involves multiple problems when in practice: packets may collide and get lost in the process of arriving to its destination, and this is called **packet loss**. Intermittent data **packet losses** and **network-induced time delay** are known to be two of the main causes for performance deterioration or even instability of any controlled networked system.

II. Hypothesis

Due to the nature of **shared paths** of communication in wireless networks described above, the study of packet loss has attracted considerable research interests. With the purpose of further investigating this problem and with the aim of providing new insights of what are the conditions in which packet loss is presented, the hypothesis of this study is:

> There exists a condition or a set of conditions in a network's environment that, when present, cause packet loss.

In the following pages, a description of the methodology and experimentation applied on a testing network environment will be presented, alongside the prediction model for the Packet Status (OK or ERROR) that was trained from the data collected.

III. Evaluation Hardware & Measurement Setup

The network installation was done using Texas Instruments' Evaluation Boards of models LAUNCHXL-CC1352R1 and LP-CC2652RB as nodes of the network. Both models are evaluation boards with wireless protocol IEEE 802.15.4 support and a built-in temperature sensor[4]. The network had two types of nodes: the **collector** and **sensor** node, so as to implement the simplest form of master-worker architecture used vastly in controlled network systems nowadays[7], which implements **a star network topology**. The collector node was a CC26x Evaluation Board (1) and the sensor nodes where a mixture of CC26x and CC13x Evaluation Boards (8).
The nodes implemented a star network topology since all sensor nodes where in charge of sending periodically the environment temperature in Celsius degrees. The collector node was then in charge of managing the orchestration of the sensor nodes' requests and monitoring the amount of orphan nodes that may disconnect from the network.
In order to capture the packet data across the network, a *sniffer board* was used. The study used Texas Instruments' CC2531 USB Dongle, which supports IEEE 802.15.4 wireless protocol[3].

IV. Experimentation

The nodes of the network were placed in a closed environment consisting of a 10 mts x 30 mts rectangle closed room. The experimentation involved **7 Controlled-environment**

Tests, where each of them exposed the installed network to different conditions of **interference** over **one hour of duration**; that is, the following tests were each performed in a 60 minute time period where the network nodes and sniffer board are working simultaneously to send temperature data to the collector and capture the packets sent, respectively. The tests are:

- Control Test 1: this is the *control* or baseline network data for the experiments. In this test there were **no interference or barrier conditions** at all to stress the network, so that it represented the **ideal conditions** of the system.
- Control Test 2: this is another instance of the previous test type, captured right after to later be averaged with the first control test and form a single Control Test.
- Double Network Test: this involved another star network nearby the analyzed one, becoming a test for IEEE 802.15.4 network interference.
- Radio-frequencies Test: this involved five common-use machines that produce radio-frequencies of different kinds inside a household: a microwave, an air fryer, a blender, a toaster and a sandwich maker where working *at the same time and place* that the network in question was operating in.
- No restriction Test: this involved four people navigating the room and making use of their personal devices (cellphone, laptop) without restrictions, with the purpose being that this test would represent the network and interference traffic in an average household.
- Wind Test: this involved two fans blowing air across the space where the collector node was installed, at a speed of 20 km/h.
- Wireless Test: this involved different machines that used specifically **WiFi network protocol**, which were 1 SmartTV, 3 Smartphones, 1 game console and 2 personal computers, all downloading content from the Internet in the same room and at the same time the network was operating.

Whilst each of the tests was being performed, the sniffer board was capturing in real time the packets travelling the network in question, and after each individual test was finished, the captured packets were stored in a file format named Packet Sniffer Data (PSD). The PSD files where processed and translated into a single tabular file from which the prediction model would gather its training data.

V. Mesaured Data Analysis

A. Measured Data Description

The wireless protocol IEEE 802.15.4 standard suggests[5][8][1] that every packet contains its **Frame Control Type** inside the packet's payload byte block, and this frame type took eight different values[2] in the captured data for the present study: BCN, DATA, CMD, ACK (Acknowledgement), 1_OCT_HEADER (1 Octet MAC Header for Low Latency)[2], CSL_WAKEUP, CSL_SECURE_ACK and RFID_BLINK, all of which are coded to fit in 3 bits inside the payload byte block[8][6] as mentioned. The frame type is crucial since it refers to the type of command that sent a package[2], which in turn specifies if the sender is either the collector/master (Beacon) or one of the sensor nodes. There is a section of the packet's bytes consisting of two bytes called **Status Bytes** destined to store either 0 (ERROR) or 1 (OK)[8]. For all packets captured, this number becomes the **class** of the packet in the present study: OK (1), when a packet was captured as it is meant to be received; and ERROR (0), when a packet is captured after a collision or fragmentation that causes the loss of a packet in the network. Thus, given the classification of every packet captured, a supervised prediction model can be trained.

But first, we assessed to describe *how the signals happen throughout time*, and thus a time series plot of the packets **per frame type** lets the reader visualize the experiment's trendings. For that, a time series using the packet number of every captured packet and the packet's frame type was used. The average number of packets through time per Frame Control Field (FCF) type is shown in Figure 1.

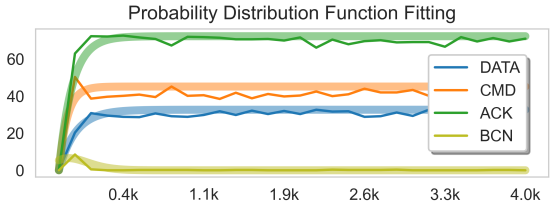

Fig. 1. Average Number of packets' arrival (y axis) through time (x axis) per frame type, and its corresponding parametric distribution fitting.

In Figure 1 a **parametric distribution** fitting is presented, which showcases the tendency of the tests performed: inter-arrival events are the best example of an event that occurs with a periodicity following **exponential distribution**, and such distribution was the one used for the modeling of DATA, CMD and ACK packet arrival behaviour, since those packets are continuously present in the network. BCN frame type packets, as it stands for the Beacon or collector node, only appear in the beginning of the network pipeline and thus its arrival periodicity follow a **poisson distribution**. The exact anaylitical functions are expressed below for DATA, CMD, ACK and BCN frames, respectively.

$$f_{data}(x) = 1 - e^{-\frac{1}{120}x} \tag{1}$$

$$f_{cmd}(x) = 1 - e^{-\frac{1}{75}x} \tag{2}$$

$$f_{ack}(x) = 1 - e^{-\frac{1}{150}x} \tag{3}$$

$$f_{bcn}(x) = \frac{1.15^x}{x!}e^{-1.15} \tag{4}$$

The remaining frame types, that is, 1_OCT_HEADER, CSL_WAKEUP, CSL_SECURE_ACK and RFID_BLINK, were not included in this description since those were barely present in the experiments to have a mathematically defined behaviour.

In order to decide with which model proceed with the classifier, all columns for each captured packet had to be analyzed individually. Most of the columns in the dataset of all captured packets are binary data (except for RSSI, Time and Packet Number), which is explained by the fact that the Packet Sniffer Data (PSD) format stores data by bytes[8] and therefore most fields in the dataset are binary or boolean representatives. Thus, the only continuous fields in the dataset are LENGTH, RSSI and TIME(MS) columns. Thus, a box plot was done per continuous field, but before so the fields were grouped by TEST_TYPE (interference type) in order to output Figure 2.

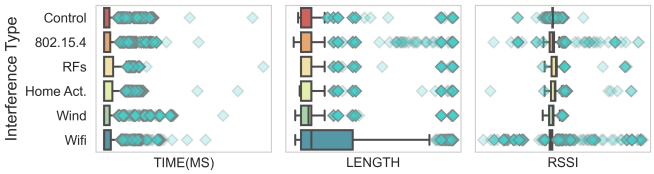

Fig. 2. Box plot of the continuous fields in every captured packet per test type.

From Figure 2 we can conclude that WiFi interference generated the largest fluctuations in all three fields, which is reflected by a larger difference in the range of the box and the outliers with respect to the ranges of the other experiments across the three fields. WiFi interference was also the test type that showed the largest absolute linear correlation to the target field (-0.134) out of all types of interference in the study. The linear correlation between the type of interference with respect to the target field, which is the Packet Status, is shown in Table I. From the correlation values, there is a tendency when there exists the use of WiFi access devices, which is present in the Home Activities and WiFi tests: the correlation with respect to the Packet Status value appears negative in the presence of WiFi interference, suggesting a weak negative association.

TABLE I
LINEAR CORRELATION BETWEEN INTERFERENCE AND PACKET STATUS

Test	Control	802.15.4	RFs	Home Act.	Wind	WiFi
Corr.	0.053	0.023	0.043	-0.027	0.027	-0.134

B. Measured Data Dimension Reduction

Since the majority of packet fields in the dataset are binary (except for the three fields analyzed in the previous section)[8], and the number of fields or features of each packet captured is 23, the visualization of all features and the Packet Status relationship cannot be done for a 24-dimensional Cartesian plane, in this case. The visualization of the Packet Status and its features as a scatter in a Cartesian plane had to be computed by pairs of features (x and y axis) with the Status defining the hue of the scatter, with the objective of distinguishing a visible trend of separation among the two classes. However, the results were the opposite: throughout the different pairs of features, the scattered dots appeared to be inseparable either by a cluterization or by a line that separated the groups of points for class 0 (ERR) and 1 (OK). Figure 3 shows a few of the pair plots performed where no visible separation is perceived.

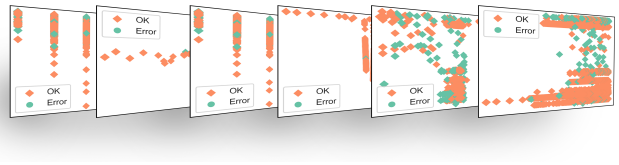

Fig. 3. Pair plots showing inseparable data by classes: OK and Error.

From the pair plots there is really no perceivable separability between classes. Linear Discriminant Analysis was used to reduce the 24-dimensionality in the dataset and to find a possible axis from which the model can separate the two classes with a fairly reduced error. Since we have either class 0 or 1 (c = 2), and LDA reduces dimension to c - 1 dimensions, the model will reduce it to one eigen value that summarizes the whole relationship of the feature fields to the target field. The LDA mentioned results in an axis shown with a dashed line in Figure 4. LDA reduced the dataset to one dimension, and thus the scatter plot ought not have two dimensions, but for visualization purposes, x axis spreads the dots horizontally to showcase the amount of dots per class by each side of the found axis.

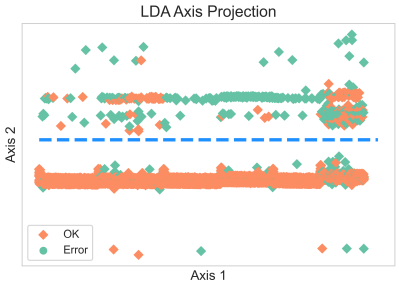

Fig. 4. Separation axis found by LDA reduction model.

VI. MEASURED MODEL PERFORMANCE

To evaluate the classifier that was trained using Linear Discriminant Analysis, we computed the confusion matrix. The advantage of computing the confusion matrix is the fact that we evaluate the model's tendency to predict mistakenly and use that evaluation in combination with the evaluation of the model's tendency to predict correctly, with the objective of computing the AUC metric, which represents how the binary classification by the model tends to give correct answers for

both classes. The confusion matrix computed for the resulting model is shown in Table II.

TABLE II
CONFUSION MATRIX FOR PACKET STATUS CLASSIFIER

	Predicted →	
Real ↓	1	0
1	TP = 10805	FN = 60
0	FP = 97	TN = 108

The number of packets with class 1 (OK) and 0 (Error) was equal, and thus the matrix shows a **high TPR (True Positive Rate** and almost **mid-valued FPR (False Positive Rate)**, which is the desired behaviour for a failure related model: false positives are more dangerous to studies that rely on packet status than False negatives. The ROC Curve gave an AUC = 0.76 , suggesting the classifier is **moderately accurate**. The ROC Curve for the model's behaviour is shown in Figure 5.

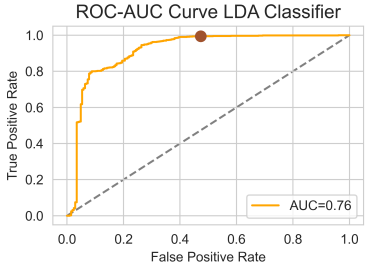

Fig. 5. ROC curve and AUC metric approximation.

VII. CONCLUSIONS

The AUC = 0.76 suggests that there exists in fact a relationship between the *combination of fields* of a sniffed packet and the prediction of its packet status, as it was stated in the hypothesis. There is not, however, a strong relationship - at least linear - between one single field and the packet status, since the correlations of all the fields with the class never goes beyond an absolute value of 0.13. Therefore, the hypothesis is accepted but, nevertheless, it must be noted that the features of a packet *combined* can predict what a handful or one feature cannot.

As a side note, out of all the interference tests performed, WiFi interference seemed to cause the largest fluctuations in data, resulting in larger amounts of outliers as well, shown in Figure 2, which coincides with the highest linear correlation with the class out of all interference types exposed to the network. It is important to note that the frame type 1_OCT_HEADER stands for 1 Octal Header Low Latency [2], and it resulted in 0.44 (highest) linear correlation to the target class, but it is a trivial result since Low Latency packets are sent when the conditions of the network are forcing low response from the devices, and therefore packet loss is expected.

REFERENCES

[1] Vladimir Alemasov. *Whsniff*. 2021 [Online]. URL: https://github.com/homewsn/whsniff/blob/master/src/whsniff.c.

[2] Michael Bahr. *Solution to TG 4e Frame Exhaustion Issue – Frame Type Extensibility Scheme*. Tech. rep. Otto-Hahn-Ring 6, 80200 München, Germany: Siemens AG, Corporate Technology, 2009. URL: https://mentor.ieee.org/802.15/dcn/09/15-09-0825-00-004e-frame-type-extensibility-solution.ppt.

[3] *CC USB Software Examples User's Guide*. Texas Instruments. Dallas, Texas, USA, 2009 [Online]. URL: https://www.ti.com/lit/ug/swru222/swru222.pdf?ts=1654121009066&ref_url=https%5C%253A%5C%252F%5C%252Fwww.google.com%5C%252F.

[4] *CC13x2, CC26x2 SimpleLink™ Wireless MCU*. Texas Instruments. Dallas, Texas, USA, 2019 [Online]. URL: https://dev.ti.com/tirex/explore/node?devtools=LP-CC2652RB&node=AGfdS5ZRIzrrfo.9rKXrnw__coGQ502__LATEST.

[5] *Generation of IEEE 802.15.4 Signals*. Rohde & Schwarz. München, Germany, 2012, pp. 10–21. URL: https://scdn.rohde-schwarz.com/ur/pws/dl_downloads/dl_application/application_notes/1gp105/1GP105_1E_Generation_of_IEEE_802154_Signals.pdf.

[6] Renwei Huang et al. "Analysis and comparison of the IEEE 802.15. 4 and 802.15. 6 wireless standards based on MAC layer". In: *International Conference on Health Information Science*. Springer. 2015, pp. 7–16.

[7] Henning A Sanneck and Georg Carle. "Framework model for packet loss metrics based on loss runlengths". In: *Multimedia Computing and Networking 2000*. Vol. 3969. International Society for Optics and Photonics. 1999, pp. 177–187.

[8] *SmartRF™ Packet Sniffer User's Manual*. Texas Instruments. Dallas, Texas, USA, 2014 [Online]. URL: https://www.ti.com/lit/ug/swru187g/swru187g.pdf?ts=1649867575476&ref_url=https%5C%253A%5C%252F%5C%252Fwww.google.com%5C%252F.