Wireless Networking Technologies Lab 3

Telecommunication Networks (TKN)

Begin: 13.05.2026, 08:00
Submission: **26.05.2026 23:59**

Introduction

In this lab, you will analyze the reception of WiFi beacons over the air. For that, we have collected traces of baseband signal using Software Defined Radio (SDR). These are already converted from raw binary files to an easily readable .mat file and uploaded as a zip on ISIS. In total we recorded channel 1, 5, 9, and 13 in the 2.4 GHz spectrum. We will only examine OFDM modulated frames in this lab. As a baseline you should start with the Matlab beacon receiver example[1] and modify it.

Preface

A WiFi receiver first needs to convert an analog signal into a digital one. This happens by observing an analog radio-frequency waveform, converting it into samples, and then running further processing such as synchronization, demodulation, and MAC-frame parsing. For our traces we used the USRP X310 SDR as receiver. The passband signal is centered around an RF carrier, for example 2.412 GHz for WiFi channel 1. Sampling the waveform directly at RF with an ADC, requires an extremely high sample rate (that of the carrier). Therefore, the received signal is down converted to baseband, such that the signal is centered around 0 Hz. In our traces we use a sample rate of 20 MS/s to receive a 20 MHz complex baseband signal. A complex baseband signal consists of an in-phase and quadrature component, where the in-phase component I is mixed with a cosine reference, while the quadrature component Q is mixed with a sine reference, creating a 90 degree shift. With complex sampling, the negative frequencies are not redundant copies of positive frequencies, but are also carrying information and represent one side of the shifted passband signal. To now analyze the received waveform the power spectral density (PSD) is a way to describe how signal power is distributed over frequency. For a finite trace, the PSD is commonly estimated by FFT-based methods. A spectrogram then repeats a PSD estimate over many overlapping time windows, producing a time-frequency view, where WiFi packets appear as short bursts over their bandwidth. The noise floor

[1] https://nl.mathworks.com/help/wlan/ug/ofdm-beacon-receiver-using-software-defined-radio.html

is the baseline energy level when no transmission is present. Estimating it correctly is important because packet detection, SNR estimation, and occupancy measurements all depend on the difference between burst energy and receiver noise.

For a deeper introduction with interactive figures and additional SDR context, read the PySDR chapter on IQ sampling: `https://pysdr.org/content/sampling.html`. In particular, review its sections on quadrature sampling, receiver-side downconversion, baseband versus bandpass signals, and calculating PSD.

SDR Traces

For this lab, four 20 MHz complex baseband captures recorded around selected 2.4 GHz WiFi channels are provided. Each trace is a `.mat` file containing a complex vector named `iq`. The files and their corresponding frequency and WiFi channels are shown in the following table.

File	**Nominal RF center**	**WiFi channel context**
`sdr_traces/2412mhz.mat`	2412 MHz	Channel 1 center
`sdr_traces/2432mhz.mat`	2432 MHz	Channel 5 center
`sdr_traces/2452mhz.mat`	2452 MHz	Channel 9 center
`sdr_traces/2472mhz.mat`	2472 MHz	Channel 13 center

Task 1: Spectrum

In this task you will examine the spectrum of the received traces.

1. Begin by loading the iq variable from the .mat file. How long is each trace in time?

2. Examine the spectrum of each trace and create a spectrogram for each of them using a 2048 point FFT, an window of 512, and overlap of 64. Create a spectrogram plot of all channels with power in dB. See the below fig. 1 as an example. Use a calibrated color scale across all four plots. A shared color scale is necessary if you want to compare channels fairly.

3. Estimate the noise floor. You can do this by computing a low percentile, for example the 10th percentile, of the spectrogram power values. Report the value in dB.

4. Define a simple occupancy metric. For example, count the fraction of time-frequency bins that exceed the estimated noise floor by 10 dB. Report the channel busy time in % and state the busiest channel.

5. Mark transmissions in the spectrogram (two likely WiFi and two non WiFi). Use rectangles, arrows, or labeled time ranges. Explain your decisions. What could the non WiFi transmission be?

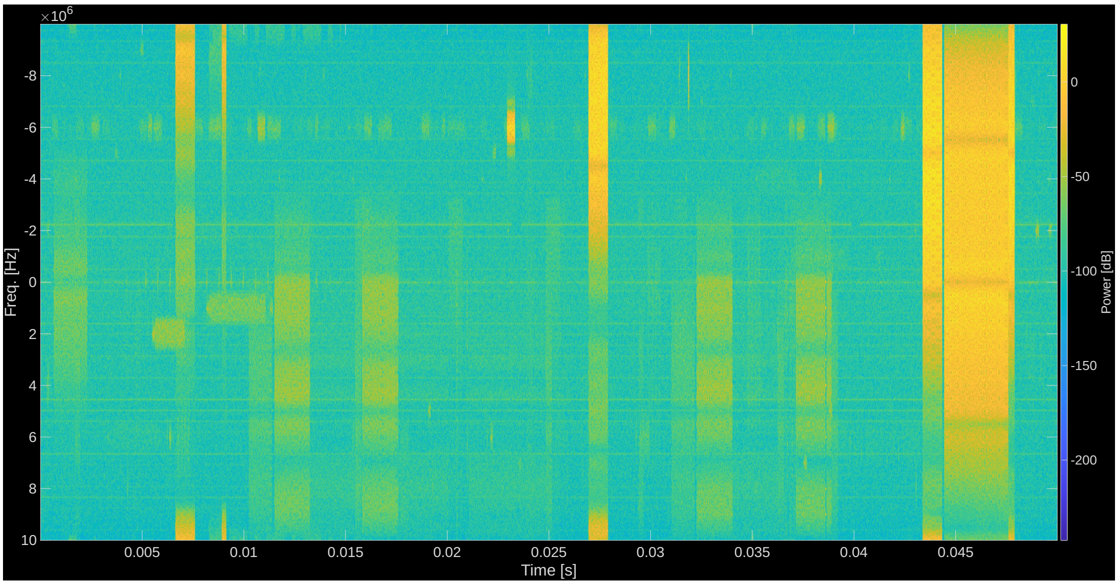

Figure 1: Example spectrogram.

Task 2: WiFi Beacons

In this task you will examine the decoded WiFi beacons.

1. Adapt the given Matlab example and decode the beacons for each channel. State in a table all available APs, SSID, BSSID, beacon interval, their beacon and operating channel, SNR, Vendor (by looking in the MAC address registry), the WiFi standard, and their base rate as well as all their supported rates (mark the base rate in red). An example table is shown below.

2. What are the four address of WiFi used for when transmitting beacons?

3. What are the four address of WiFi used for in a normal data frame?

SSID	BSSID	Interval	B. ch./Op. ch.	SNR	Vendor	Std.	Rates
Fill this table with your decoded results.							

Task 3: APs

WLAN supports virtualization of APs, a physical AP can be virtualized by sending multiple beacons with different BSSIDs operating as multiple APs at once. Does the trace include a virtualized AP? If so, which factors indicate a virtualization of APs in the trace?

Task 4: Generating Beacons

Generate your own OFDM beacons using the Matlab example for OFDM beacon generation. Plot the spectrum of your generated beacon and describe how it changes over time.

Submission

Your submission should be a zip file with the following:

1. A PDF report with all your findings and results to the tasks listed above.
2. All the code needed to reproduce your results.

Do not include a copy of the provided traces in your submission.