I. Approximate Set Identification: PAC Analysis for Group Testing
II. SSR for OFDM channel estimation: Implementation in SDR

Monika Bansal
Advisor: Dr. Chandra R Murthy

Department of ECE, IISc

June 18, 2016

Overview

1. Approximate Set Identification: PAC Analysis for Group Testing
   - Group Testing Framework
   - Function Learning Model
   - PAC Analysis: Approximate Set Identification
     - PAC type bound for CoMa
     - PAC type bound for DD
2. SSR for OFDM channel estimation: Implementation in SDR
   - Sparsity in Channel
   - SBL Framework
   - Implementation in GNU Radio
   - Experiment Results

Group Testing

A set of N items with k defective items (k$\ll$N).

- **Group test:** A group of items is tested in a group test
  - Test outcome **1** indicates presence of **defective** item(s)
  - Outcome **0** indicates all items are **non-defective** in the test
- **Main issues:**
  - Sample Complexity
  - Correctness of solution
  - Pooling Design

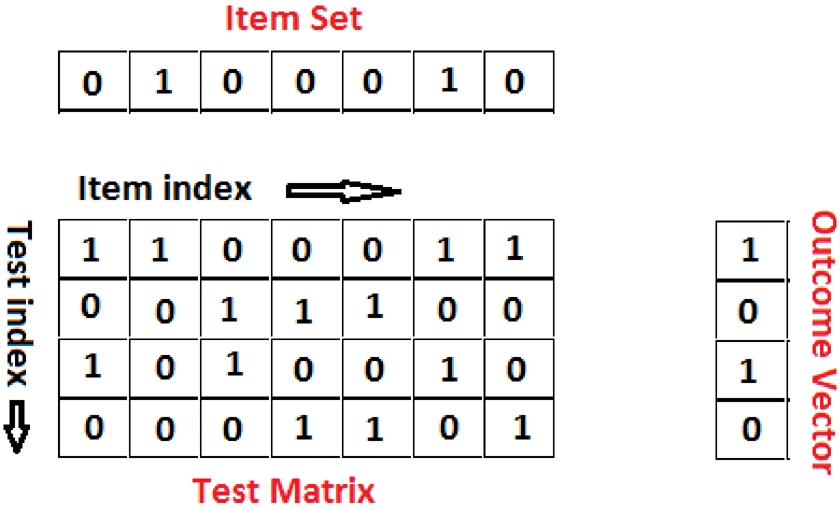


Figure: A toy example for non-adaptive group testing.

Group Testing Model

$$\mathbf{y} = \mathbf{A}\mathbf{x}$$
$$= \bigvee_{j:x_j=1} \mathbf{a}_{c_j}$$

$$\underbrace{\begin{bmatrix}1\\1\\0\end{bmatrix}}_{\mathbf{y}} = \underbrace{\begin{bmatrix}0&1&0&0&1&0&0\\0&0&0&1&1&0&1\\1&0&1&0&0&1&0\end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix}0\\1\\0\\1\\0\\0\\0\end{bmatrix}}_{\mathbf{x}}$$

$\mathbf{y} \in \{0,1\}^M$ is the binary test outcome vector

$\mathbf{a}_{c_j} \in \{0,1\}^M$ is the j^{th} column of $\mathbf{A}$

$\mathbf{a}_{r_i} \in \{0,1\}^n$ is the i^{th} row of $\mathbf{A}$

$\mathbf{A}(i,j) \sim \mathcal{B}(p)$ i.i.d.

$\mathbf{x} \in \{0,1\}^n$ is test item vector

Learning Problem

Learn unknown target function $f(\cdot) \in \mathcal{C}$

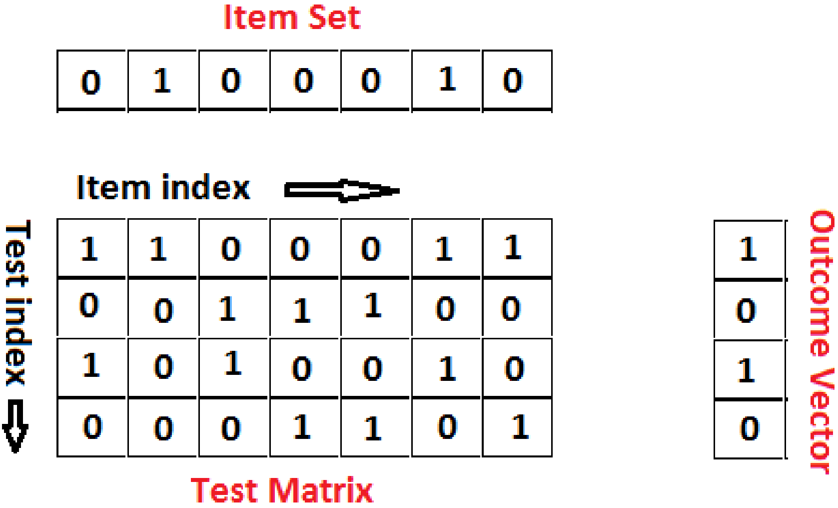
Available items to the **learner:**

- Random examples: $\mathbf{a}_i$
- Corresponding label: y_i

$$\mathbf{a}_i \in \{0,1\}^n$$
$$y_i = f(\mathbf{a}_i) \in \{0,1\}$$
$$f : \{0,1\}^n \rightarrow \{0,1\} \text{ is some boolean function}$$

How many examples do we need to output an hypothesis f^* s.t. maximum error is ϵ with confidence $1-\delta$?

Figure Revisited



Group Testing as a Function Learning Problem

	Group testing	**Learning model**
Target function	$\mathbf{x}$	$f(\cdot)$
Random example	$\mathbf{a}_{r_i}$	$\mathbf{a}_i$
Label	$\mathbf{y}(i)$	y_i
Distribution	$\mathcal{B}(p)$	$\mathcal{D}$
Output hypothesis	$\hat{\mathbf{x}}$	f^*

Goal: Analyze **group testing recovery algorithms** using PAC (Probably Approximately Correct) framework applied to function learning problems.

PAC Analysis

A learning algorithm is said to be PAC-learn $\mathcal{C}$ with *approximation parameter* ϵ *and confidence parameter* δ if $\forall$ distributions $\mathcal{D}$ and all target functions $f \in \mathcal{C}$, the algorithm draws M samples, runs for time at most t and outputs a function f^* s.t.

$$e(f^*, f) = Pr_{\mathbf{a}\sim\mathcal{D}}\left(f^*(\mathbf{a}) \neq f(\mathbf{a})\right)$$

$$Pr\left(e(f^*, f) > \epsilon\right) \leq \delta$$

With prob. $1-\delta$ the output hypothesis f^* will make at most ϵ error.

Defective Set Recovery

$$\underbrace{\begin{bmatrix}1\\1\\0\end{bmatrix}}_{y} = \underbrace{\begin{bmatrix}0&1&0&0&1&0&0\\0&0&0&1&1&0&1\\1&0&1&0&0&1&0\end{bmatrix}}_{A} \underbrace{\begin{bmatrix}0\\1\\0\\1\\0\\0\\0\end{bmatrix}}_{x}$$

- Column Matching (CoMa)[1] Algorithm
- Definite Defective (DD)[2] Algorithm

[1] Non-adaptive Group Testing: Explicit Bounds and Novel Algorithms

[2] Group Testing Algorithms: Bounds and Simulations

ϵ Approximate Set Identification

Allowed hidden non-defective items

$$\mathbb{P}_{\mathbf{a}_i \sim \mathcal{B}}\left(\hat{\mathbf{x}}(\mathbf{a}_i) \neq \mathbf{x}(\mathbf{a}_i)\right) = \left(1 - (1-p)^G\right)(1-p)^k \leq \epsilon$$

$$g_\epsilon = \left\lfloor \frac{\ln\left(1 - \epsilon / (1-p)^k\right)}{\ln(1-p)} \right\rfloor$$

$$\mathbb{P}(G \leq g_\epsilon) \geq 1 - \delta$$

Allowed unidentified defective items

$$\mathbb{P}_{\mathbf{a}_i \sim \mathcal{B}}\left(\hat{\mathbf{x}}(\mathbf{a}_i) \neq \mathbf{x}(\mathbf{a}_i)\right) = (1-(1-p)^D)(1-p)^{k-D} \leq \epsilon$$

$$d_\epsilon = \left\lfloor \frac{\ln(1+\epsilon/(1-p)^k)}{\ln(1/(1-p))} \right\rfloor$$

$$\mathbb{P}(D \leq d_\epsilon) \geq 1-\delta$$

PAC type bound for CoMa

Theorem 1

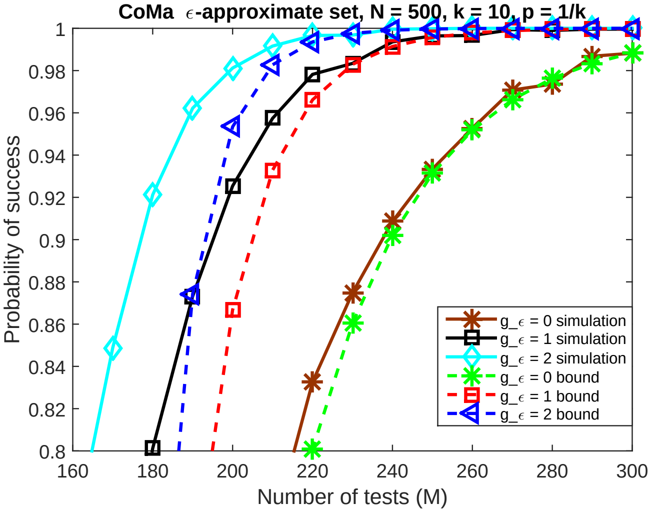
The sufficient number of tests such that estimated set using CoMa does not agree with the true defective set on the future group tests with probability at most ϵ with confidence parameter $1 - \delta$ is given as,

$$M_{g_\epsilon} = \frac{\log \binom{n-k}{g_\epsilon+1} + \log \frac{1}{\delta}}{\log \left(1/(1 - (1-p)^k + (1-p)^{g_\epsilon+k+1})\right)},$$

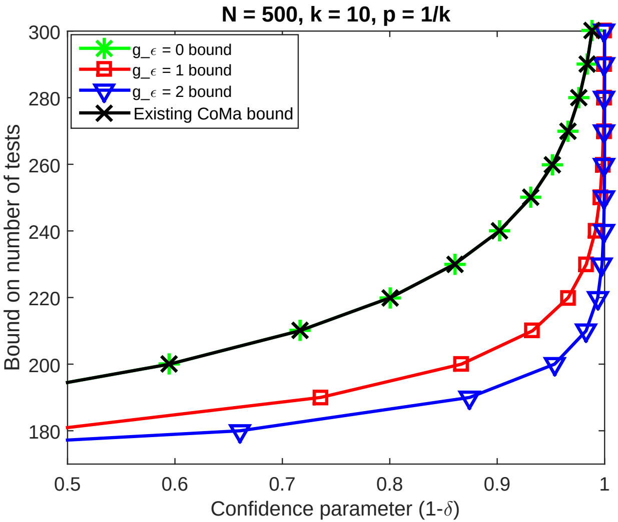
$$\mathbb{P}(e(\hat{\mathbf{x}}, \mathbf{x}) > \epsilon) = \mathbb{P}(G \geq g_\epsilon) \leq \binom{n-k}{g_\epsilon + 1} P^h_{g_\epsilon+1}(M)$$

where, $\quad P^h_{g_\epsilon+1}(M) = \left(1 - (1-p)^k + (1-p)^{g_\epsilon+1+k}\right)^M$

Bound on success probabilities



Bound on number of tests in PAC setting



PAC type bound for DD

Theorem 2

The sufficiency bound on the number of tests such that estimated set using DD do not agree with the true defective set on the future group tests with probability at most ϵ with confidence parameter $1-\delta$ is given as,

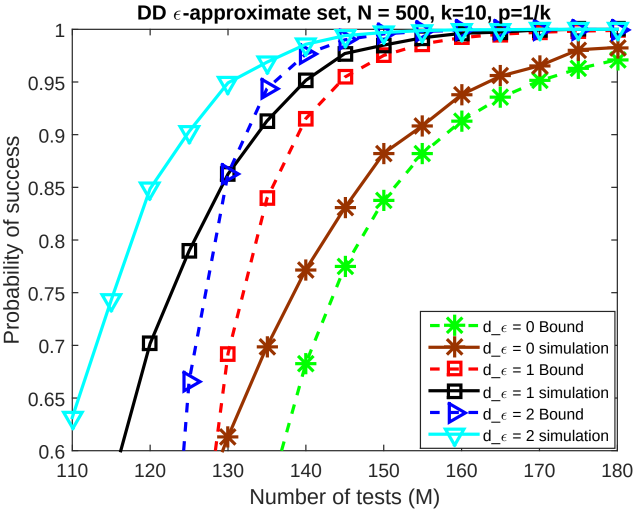
$$\binom{k}{d_\epsilon+1}(1-(d_\epsilon+1)p(1-p)^{k-1+\bar{g}+\tilde{g}})^M \leq \delta,$$

$$\mathbb{P}(e(\hat{\mathbf{x}},\mathbf{x}) > \epsilon / G = g) \leq \binom{k}{d_\epsilon+1}(1-(d_\epsilon+1)p(1-p)^{k-1}(1-p)^g)^M,$$

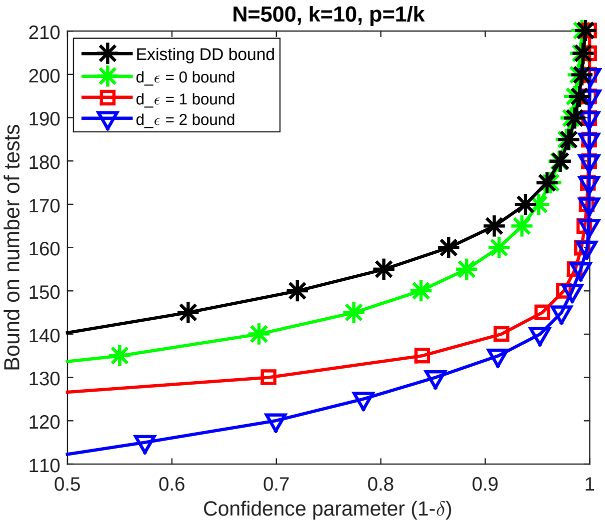
$$\mathbb{P}(e(\hat{\mathbf{x}},\mathbf{x}) > \epsilon) \leq \binom{k}{d_\epsilon+1}\left(1-(d_\epsilon+1)p(1-p)^{k-1}(1-p)^{\bar{g}+\tilde{g}}\right)^M.$$

where $\bar{g} = (N-k)(1-p(1-p)^k)^M$ and $\tilde{g}$ is a tuning parameter which depends on d_ϵ.

Bound on success probabilities



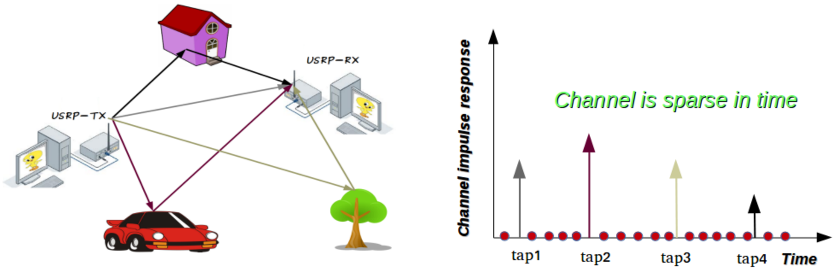
Bound on number of tests in PAC setting



Conclusions

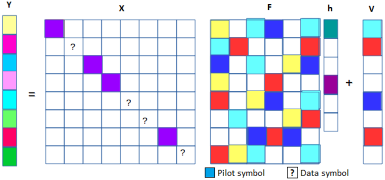
- PAC analysis resulted in approximate set identification analysis
- Full defective set can always be recovered from approximate set
- Two stage procedure has more flexibility
- Once we identify a big number of non-defective items, random pooling does not give much further information

Wireless Communication Channel



Sparse in lag domain

OFDM Channel Model



$$\mathbf{y} = \mathbf{XFh} + \mathbf{v}$$

$\mathbf{y} \in \mathbb{C}^{N\times 1}$ is the received vector after FFT

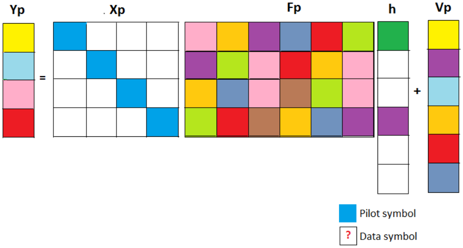
$\mathbf{X} \in \mathbb{C}^{N\times N}$ contains data symbol and pilot symbols along the diagonal

$\mathbf{F} \in \mathbb{C}^{N\times L} (N > L)$ contains the first L columns of $N \times N$ DFT matrix

$\mathbf{v} \in \mathbb{C}^{N\times 1} \sim \mathcal{CN}(0, \sigma^2 \mathrm{I})$ is the AWGN noise

$\mathbf{h} \in \mathbb{C}^{L\times 1}$ is the time domain channel response

Channel Model using *Pilots* only:



$$\mathbf{y}_P = \mathbf{X}_p \mathbf{F}_p \mathbf{h} + \mathbf{v}_p, \qquad (P < L)$$
$$= \phi_p \mathbf{h} + \mathbf{v}_p$$

SBL Framework

$$\mathbf{h} \sim \mathcal{CN}(0, \mathbf{\Gamma}), \qquad \mathbf{\Gamma} = \text{diag}(\gamma(1), ..., \gamma(L))$$

SBL estimation problem:

$$\begin{aligned}\hat{\mathbf{h}} &= \underset{\mathbf{h}, \gamma \in \mathbb{R}_+^{L \times 1}}{\arg\max}\; p(\mathbf{y}_p | \mathbf{h}; \gamma) p(\mathbf{h}; \gamma) \\ &= \underset{\mathbf{h}, \gamma \in \mathbb{R}_+^{L \times 1}}{\arg\min}\; \frac{\|\mathbf{y}_p - \mathbf{X}_p \mathbf{F}_p \mathbf{h}\|_2^2}{\sigma^2} + \log|\mathbf{\Gamma}| + \mathbf{h}^H \mathbf{\Gamma}^{-1} \mathbf{h}\end{aligned}$$

Instead of estimating $\mathbf{h}$ directly, we first estimate γ using type II ML estimate as given below

$$\hat{\gamma}_{ML} = \underset{\gamma \in \mathbb{R}_+^{L \times 1}}{\arg\max}\; p(\mathbf{y}_p; \gamma)$$

EM algorithm

$$p(\mathbf{h};\gamma) = \prod_{i=1}^{L} (\pi\gamma(i))^{-1} \exp\left(-\frac{|h(i)|^2}{\gamma(i)}\right)$$

$$\textbf{E-step:}\ \ Q\left(\gamma|\gamma^{(r)}\right) = \mathbb{E}_{\mathbf{h}|\mathbf{y}_p;\gamma^{(r)}}\left[\log p(\mathbf{y}_p, \mathbf{h};\gamma)\right]$$

$$\textbf{M-step:}\ \ \gamma^{(r+1)} = \underset{\gamma\in\mathbb{R}_+^{L\times 1}}{\arg\max}\ Q\left(\gamma|\gamma^{(r)}\right)$$

$$\gamma^{(r+1)}(i) = \Sigma(i,i) + |\mu(i)|^2$$

Probability densities:

$$p\left(\mathbf{h}|\mathbf{y}_p;\gamma^{(r)}\right) = \mathcal{CN}(\mu,\Sigma)$$

$$\Sigma = \Gamma^{(r)} - \Gamma^{(r)}\phi_p^H(\sigma^2\mathrm{I}_{P_b} + \phi_p\Gamma^{(r)}\phi_p^H)^{-1}\phi_p\Gamma^{(r)},\ \ \mu = \sigma^{-2}\Sigma\phi_p^H\mathbf{y}_p$$

SBL Algorithm for OFDM Channel Estimation

Algorithm 1 SBL for estimating time domain channel taps

Input: $\mathbf{y}_p, \phi_p, r_{max}$ and ϵ.

Initialize $\Gamma^{(0)} = \mathrm{I}_L$, Set difference $= 1$, $r = 0$

while (difference$> \epsilon$ and $r < r_{max}$)

E-step: $\mu = \sigma^{-2}\Sigma\phi_p^H\mathbf{y}_p$

$\Sigma = \Gamma^{(r)} - \Gamma^{(r)}\phi_p^H\left(\sigma^2\mathrm{I}_{P_b} + \phi_p\Gamma^{(r)}\phi_p^H\right)^{-1}\phi_p\Gamma^{(r)}$

M-step: $\gamma^{(r+1)}(i) = \Sigma(i,i) + |\mu|^2$ for $i = 1, 2, \ldots, L$

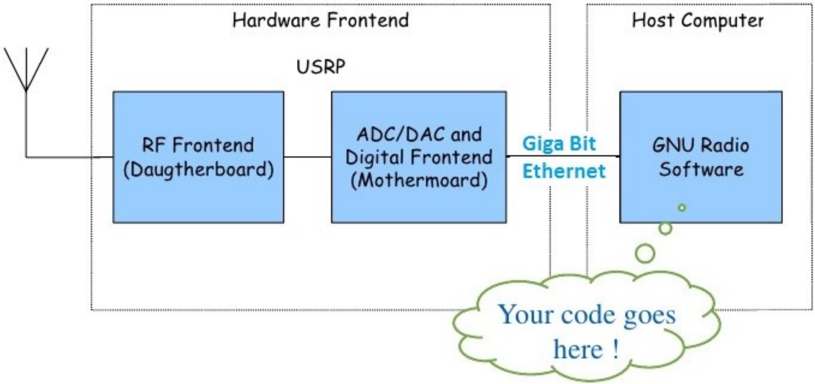
difference $\triangleq \|\gamma^{(r+1)} - \gamma^{(r)}\|_2^2$, $r \leftarrow r + 1$ **end**

output: $\mu, \gamma^{(r)}$

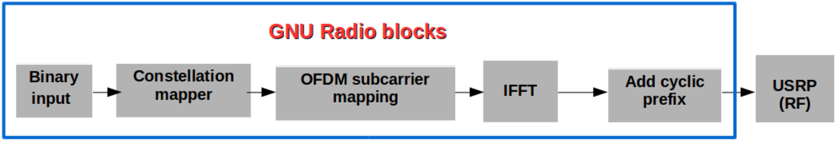
Introduction to GNU Radio

- A software development tool kit with **signal processing** blocks written in C++/Python.
- GRC (GNU Radio Companion) is the user interface for GNU Radio.
- Can be used with external RF hardware (such as USRP N210) to create Software Defined Radio(SDR).

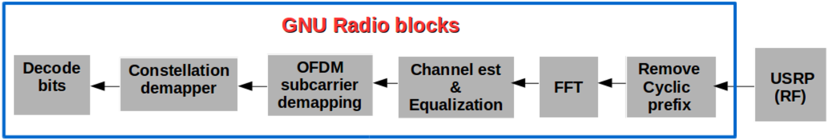
System architecture



OFDM Chain in GNU Radio

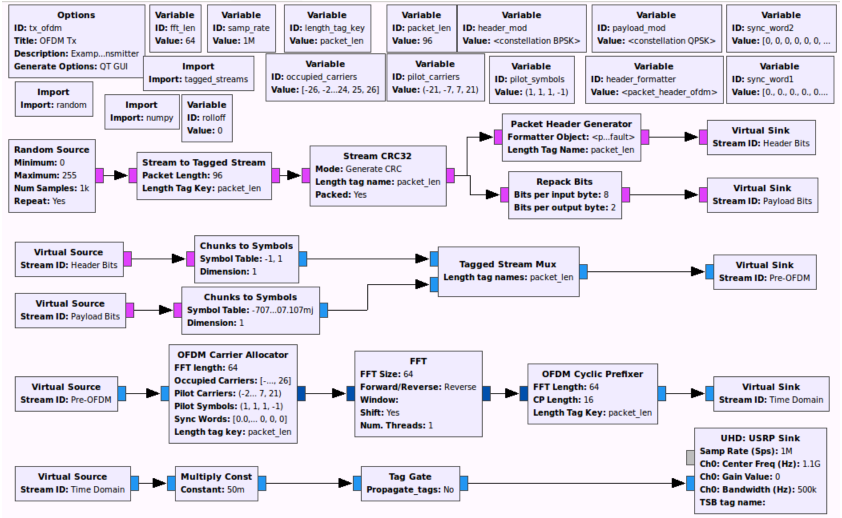


OFDM-Transmitter chain for DATA

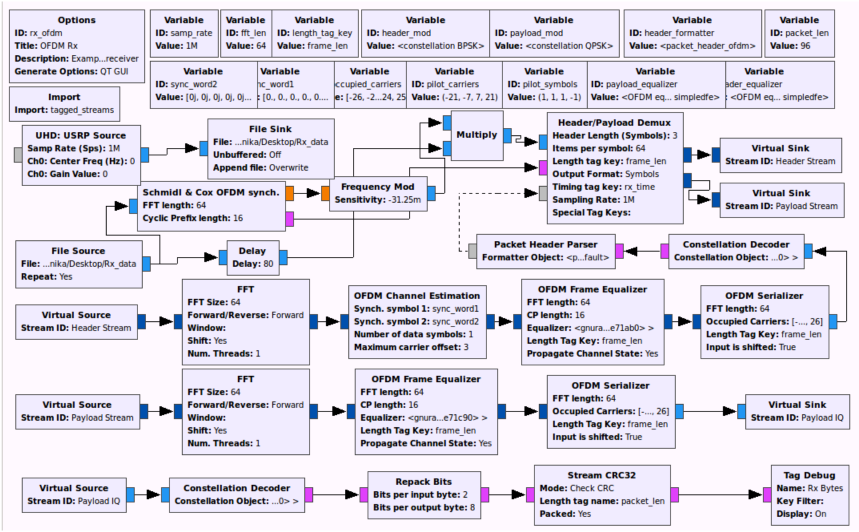


OFDM-Receiver chain for DATA

OFDM Transmit Chain in GNU Radio



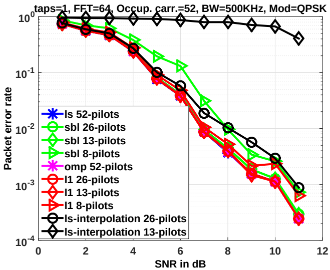
OFDM Receive Chain in GNU Radio

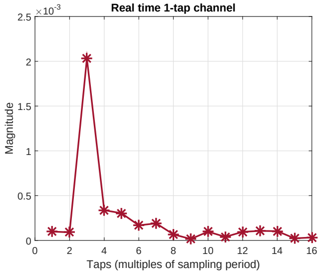


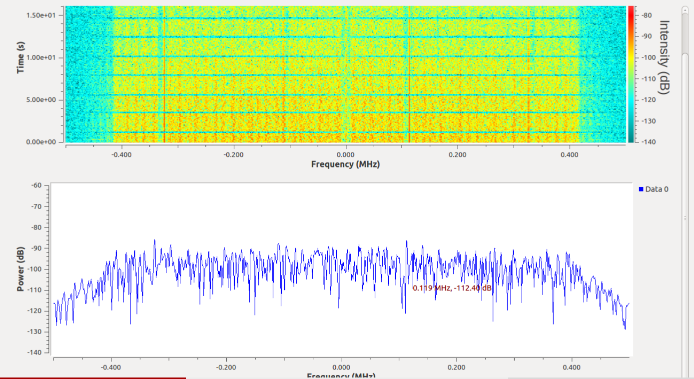
Experiment Results

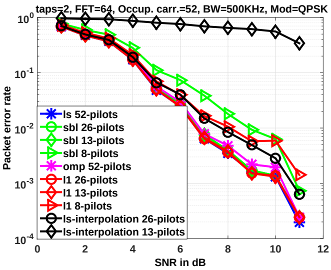
System Parameters:

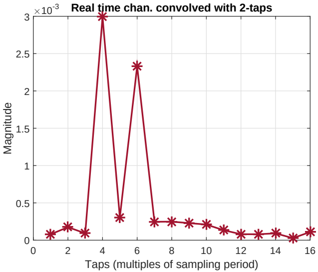
- Number of OFDM sub-carriers (N) = 64
- Cyclic prefix (CP) = 16
- Packet size = 92 Bytes
- Cyclic redundancy check (CRC) = 32 bits = 4 Bytes
- Number of sync words per packet = 2 OFDM symbols
- Header length = 1 OFDM symbol
- Header modulation = BPSK
- Data Modulation = QPSK
- Centre frequency (f_c) = 1.1GHz
- Bandwidth (BW) = 500kHz
- Sampling frequency (f_s) = 1MS/sec

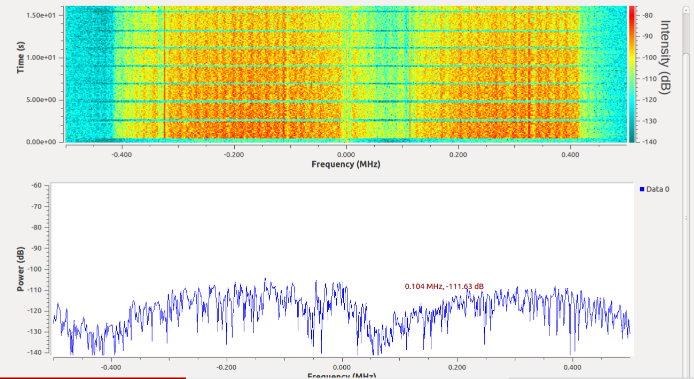

taps=1, FFT=64, Occup. carr.=52, BW=500KHz, Mod=QPSK
Packet error rate
SNR in dB
ls 52-pilots
sbl 26-pilots
sbl 13-pilots
sbl 8-pilots
omp 52-pilots
l1 26-pilots
l1 13-pilots
l1 8-pilots
ls-interpolation 26-pilots
ls-interpolation 13-pilots

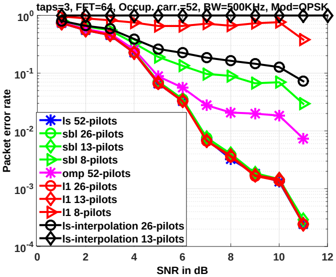

Real time 1-tap channel
Magnitude
Taps (multiples of sampling period)

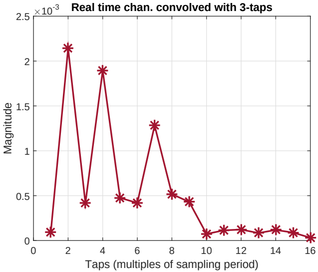

Time (s)
Frequency (MHz)
Intensity (dB)
Power (dB)
Data 0
0.119 MHz, -112.40 dB

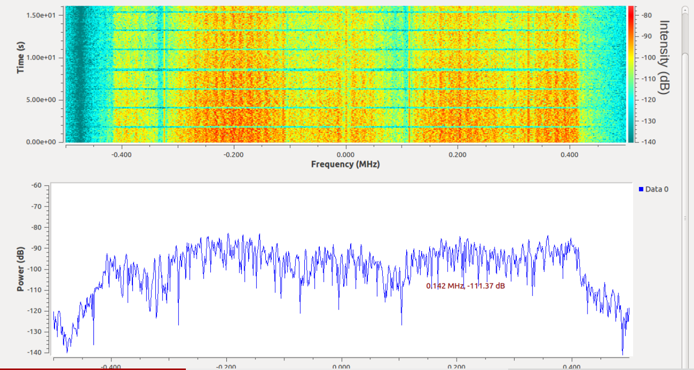

taps=2, FFT=64, Occup. carr.=52, BW=500KHz, Mod=QPSK
Packet error rate
SNR in dB
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
0
2
4
6
8
10
12
ls 52-pilots
sbl 26-pilots
sbl 13-pilots
sbl 8-pilots
omp 52-pilots
l1 26-pilots
l1 13-pilots
l1 8-pilots
ls-interpolation 26-pilots
ls-interpolation 13-pilots


Real time chan. convolved with 2-taps
$\times 10^{-3}$
Magnitude
Taps (multiples of sampling period)
0
0.5
1
1.5
2
2.5
3
0
2
4
6
8
10
12
14
16

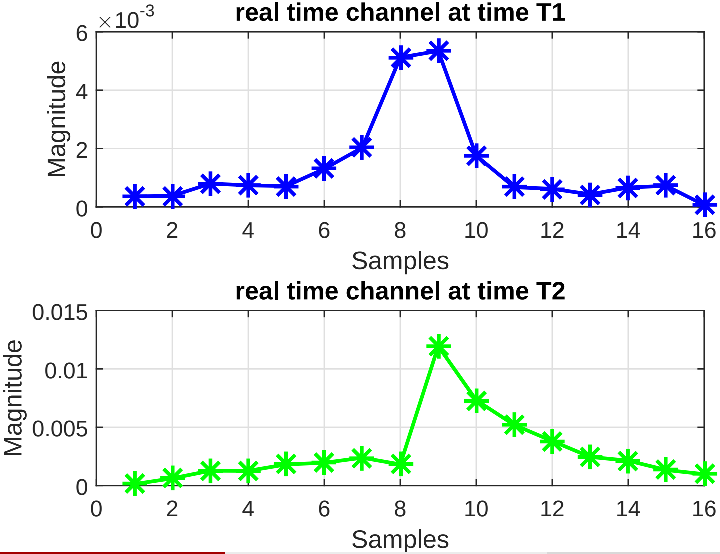

Time (s)
1.50e+01
1.00e+01
5.00e+00
0.00e+00
Frequency (MHz)
-0.400
-0.200
0.000
0.200
0.400
Intensity (dB)
-80
-90
-100
-110
-120
-130
-140
Power (dB)
-60
-70
-80
-90
-100
-110
-120
-130
-140
Data 0
0.104 MHz, -111.63 dB
-0.400
-0.200
0.000
0.200
0.400
Frequency (MHz)


taps=3, FFT=64, Occup. carr.=52, BW=500KHz, Mod=QPSK
Packet error rate
SNR in dB
ls 52-pilots
sbl 26-pilots
sbl 13-pilots
sbl 8-pilots
omp 52-pilots
l1 26-pilots
l1 13-pilots
l1 8-pilots
ls-interpolation 26-pilots
ls-interpolation 13-pilots


Real time chan. convolved with 3-taps
Magnitude
Taps (multiples of sampling period)


Time (s)
Frequency (MHz)
Intensity (dB)
Power (dB)
Data 0
0.142 MHz, -111.37 dB

Real Time Channel



Future Work

- Joint data detection and channel estimation algorithms
- Throughput analysis