Implementation Aspects of the Random Demodulator for Compressive Sensing

UC Louvain, Louvain-la-Neuve - April 26, 2013

Pawel Jerzy Pankiewicz
pjp@es.aau.dk
@pawelpankiewicz

Department of Electronic Systems,
The SparSig project

www.sparsesampling.com

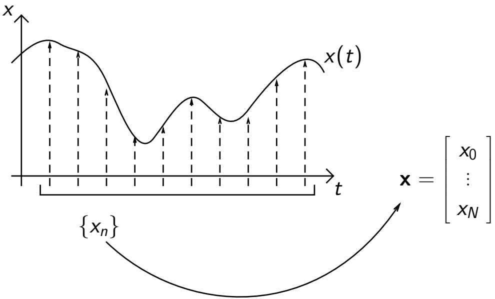
Agenda

Introduction
Compressive Sensing
Motivation
Problem statement
Analog Compressed Sensing
Implementation ideas
Sampling compressed sensing way
Random demodulator
Measurement matrix
Hardware perspective
Random demodulator - influenced architectures
Signal recovery and non-ideal effects
RD with non-ideal filter
Supervised calibration
Quantization

Classic sampling

How do we discretize signals

- Classic Nyquist sampling:



- $T < \frac{1}{2\cdot\mathcal{B}}$
- Nyquist criteria applies into worst case scenarios.
- Usually: we sample a lot of data, but throw most of it away (JPEG, MP3).

Compressed Sensing
What is it?

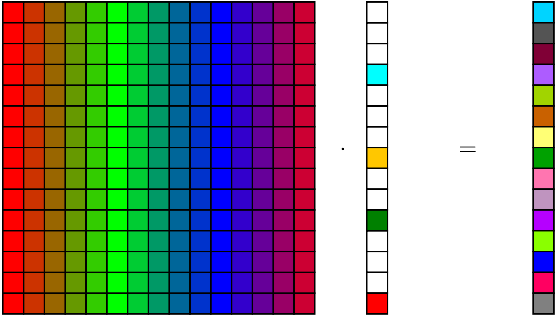
- New signal acquisition/compression theory from around 2004.
- Combines sampling and compression of signals.

"*CS theory asserts that one can recover certain signals and images from far fewer samples or measurements than traditional methods use.*"[1]

[1] Candès and Wakin, "An Introduction To Compressive Sampling", '08'.

Compressed Sensing
Requirements

The signal must be sparse in a *known dictionary*:

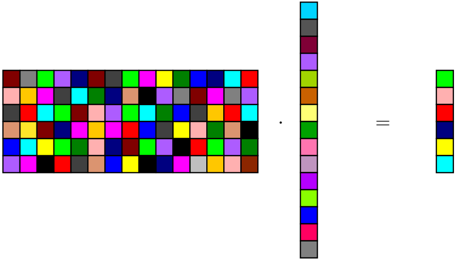


dictionary matrix $\mathbf{\Psi}$

sparse vector α

original signal vector $\mathbf{x}$

Compressed Sensing
Acquisition

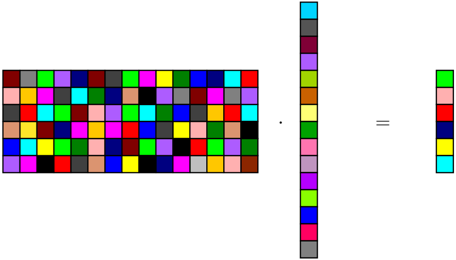


compression matrix $\mathbf{\Phi}$

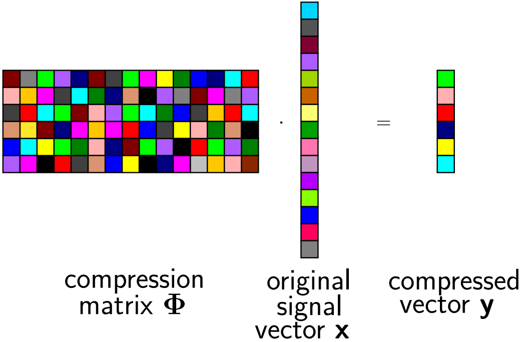
original signal vector $\mathbf{x}$

compressed vector $\mathbf{y}$

AALBORG UNIVERSITY
DENMARK

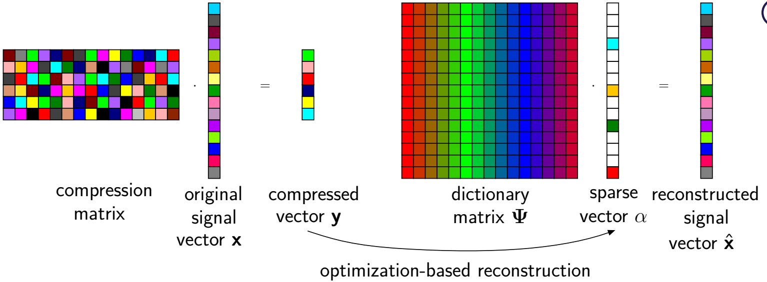


- The signal vector is mixed with a *measurement* matrix before sampling.



- The signal vector is mixed with a *measurement* matrix before sampling.
- Sample the (fewer) mixed "measurements".

Compressed Sensing
The Reconstruction Principle



Dept. of Electronic Systems,
Aalborg University,

Digital device
Option 1

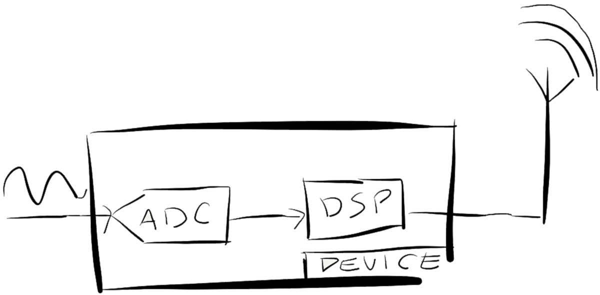


Figure : Traditional (Nyquist) Sampling

Digital device
Option 2

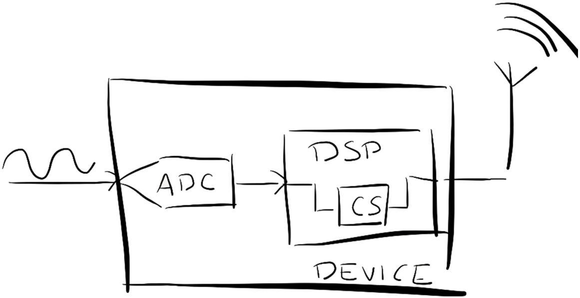


Figure : Digital Compressed Sensing

Digital device
Option 3

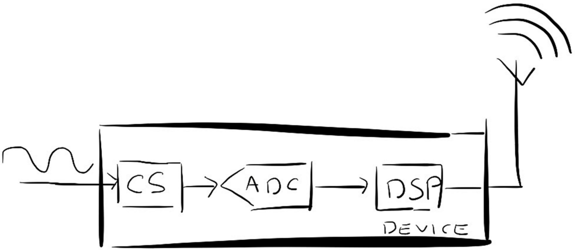


Figure : Analog Compressed Sensing

Motivation
Overview

Analog front-end is an energy bottleneck

- Inevitable analog-to-digital conversion
- Power consumption – mainly dictated by the sampling frequency
- Need to sample according to Nyquist rate

Implementing analog compressed sensing

- Digital Signal Processors – are now highly capable
- Trade analog processing over digital processing

Problem statement
All things digital

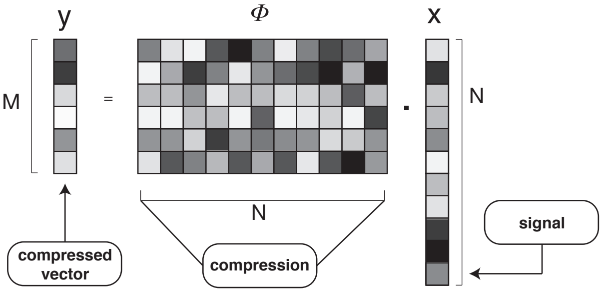


Figure : **Discrete** model of compressed sensing

- But how to deal with analog signals ?
- Analog signal has infinite dimension
- How to obtain compressed vector ?

Problem statement
Does analog CS just work?

- We need a model of an analog front-end in the measurement matrix
- Undesired hardware effects are unavoidable in the implementation.
- Static and dynamic hardware changes.
- Turns out that compressed sensing signal reconstruction methods do not account for many non-idealities.

How to implement compressed sensing
Following the digital model

Sample input and apply CS processing

Initially we can digitalize the signal using an analog-to-digital converter (ADC)

- Sample the signal with the Nyquist rate

Dept. of Electronic Systems, Aalborg University,

How to implement compressed sensing

Following the digital model

Sample input and apply CS processing

Initially we can digitalize the signal using an analog-to-digital converter (ADC)

- Sample the signal with the Nyquist rate
- Apply compressed sensing to a digitized signal

Dept. of Electronic Systems, Aalborg University,

How to implement compressed sensing
Following the digital model

Sample input and apply CS processing

Initially we can digitalize the signal using an analog-to-digital converter (ADC)

- Sample the signal with the Nyquist rate
- Apply compressed sensing to a digitized signal
- Obtain reconstructed signal

Dept. of Electronic Systems, Aalborg University,

How to implement compressed sensing
Following the digital model

Sample input and apply CS processing

Initially we can digitalize the signal using an analog-to-digital converter (ADC)

- Sample the signal with the Nyquist rate
- Apply compressed sensing to a digitized signal
- Obtain reconstructed signal

Why this is a bad approach ?

Dept. of Electronic Systems, Aalborg University,

How to implement compressed sensing
Following the digital model

Sample input and apply CS processing

Initially we can digitalize the signal using an analog-to-digital converter (ADC)

- Sample the signal with the Nyquist rate
- Apply compressed sensing to a digitized signal
- Obtain reconstructed signal

Why this is a bad approach ?

- We sample with the Nyquist frequency
- No power consumption reduction

  There are no benefits
- We perform only unnecessary and energy consuming signal processing

How to implement compressed sensing
Analog-to-information converter - the model we need

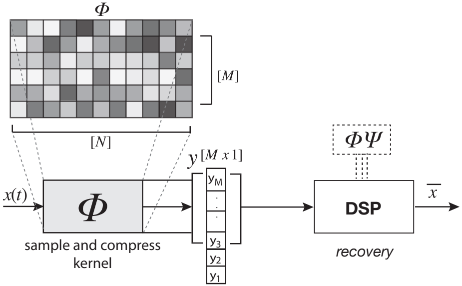


Figure : Compressed sensing scheme

- $\mathbf{y} = \mathbf{\Phi \Psi x}, \qquad \mathbf{y}(M \times 1); \mathbf{\Phi}(M \times N); \mathbf{\Psi}(N \times N)$

We need an analog compression kernel that:

How to implement compressed sensing

Analog-to-information converter - the model we need

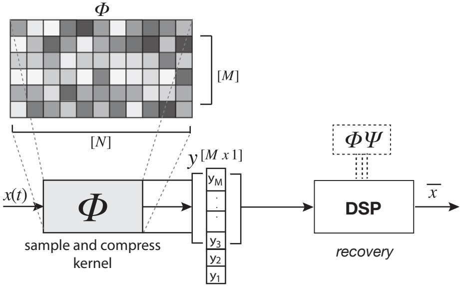


Figure : Compressed sensing scheme

- $\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{x}, \qquad \mathbf{y}(M \times 1); \mathbf{\Phi}(M \times N); \mathbf{\Psi}(N \times N)$

We need an analog compression kernel that:

- provides non-adaptive linear projections of the analog input

Dept. of Electronic Systems, Aalborg University,

How to implement compressed sensing
Analog-to-information converter - the model we need

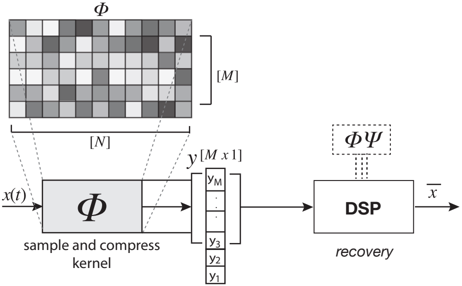


Figure : Compressed sensing scheme

- $\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}\mathbf{x}, \qquad \mathbf{y}(M \times 1); \mathbf{\Phi}(M \times N); \mathbf{\Psi}(N \times N)$

We need an analog compression kernel that:

- provides non-adaptive linear projections of the analog input
- complies with a **RIP** and **incoherence** requirements

Dept. of Electronic Systems, Aalborg University,

Sampling compressed sensing way

Simplest approach

Taking random samples with sub-Nyquist rate

- We decrease sampling frequency

Sampling compressed sensing way
Simplest approach

Taking random samples with sub-Nyquist rate

- We decrease sampling frequency
- Our compression mechanism is incoherent with some of the sparse basis

AALBORG UNIVERSITY
DENMARK

Taking random samples with sub-Nyquist rate

- We decrease sampling frequency
- Our compression mechanism is incoherent with some of the sparse basis
- It is possible to model analog front-end (our encoder) in the DSP and perform reconstruction

Sampling compressed sensing way
Simplest approach

Taking random samples with sub-Nyquist rate

- We decrease sampling frequency
- Our compression mechanism is incoherent with some of the sparse basis
- It is possible to model analog front-end (our encoder) in the DSP and perform reconstruction

Disadvantages ?

Sampling compressed sensing way
Simplest approach

Taking random samples with sub-Nyquist rate

- We decrease sampling frequency
- Our compression mechanism is incoherent with some of the sparse basis
- It is possible to model analog front-end (our encoder) in the DSP and perform reconstruction

Disadvantages ?

- performing random sampling brings a certain amount of imperfections
  - aperture jitter increases due to nonuniform clock usage
  - overall sampling frequency is decreased but we still might need high sampling grid
  - modeling the front-end by measurement matrix might be difficult

AALBORG UNIVERSITY
DENMARK

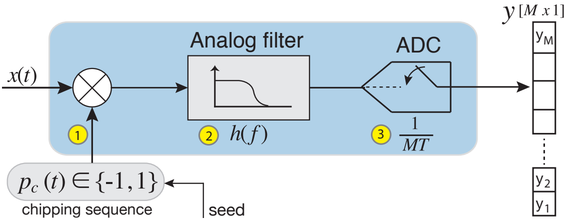


Figure : Random demodulator architecture[2]

1. demodulation
2. low-pass filtering (integration)
3. low-rate sampling

[2]Kirolos et al., "Analog-to-Information Conversion via Random Demodulation", '06.

Dept. of Electronic Systems, Aalborg University,

Random demodulator

Overview

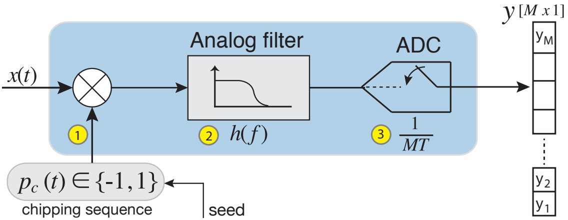


Figure : Random demodulator architecture[2]

1. demodulation
   - mixing the signal with the pseudo-random sequence called the chipping sequence (e.g., [+1/-1])
   - frequency of the sequence > Nyquist frequency of the input

[2]Kirolos et al., "Analog-to-Information Conversion via Random Demodulation", '06.

Dept. of Electronic Systems, Aalborg University,

Random demodulator
Overview

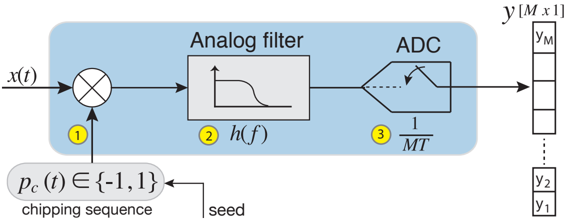


Figure : Random demodulator architecture[2]

2. low-pass filtering
   - anti-aliasing operation prior to the low-rate sampling
   - in respect to sampling frequency we set properly cut-off frequency f_{cut}

[2] Kirolos et al., "Analog-to-Information Conversion via Random Demodulation", '06.

Random demodulator

Overview

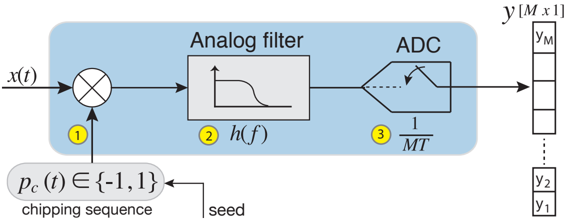


Figure : Random demodulator architecture[2]

3. low-rate sampling
   - sampling the signal using analog-to-digital converter
   - sampling frequency has to be higher than $2 \times f_{cut}$

[2]Kirolos et al., "Analog-to-Information Conversion via Random Demodulation", '06.

Operations
How they might look like

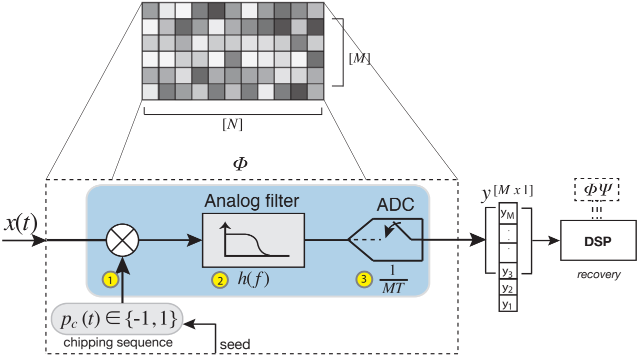
MATLAB DEMO

Source code available at:
`http://sparsesampling.com/pawelpankiewicz/rd`

Dept. of Electronic Systems, Aalborg University,

Random demodulation

Once again - let us recall the model



- $\mathbf{y} = \Phi\Psi\mathbf{x}, \qquad \mathbf{y}(M \times 1); \Phi(M \times N); \Psi(N \times N)$

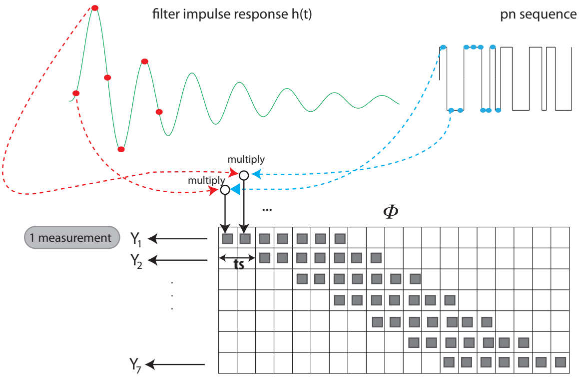
"Analog-to-Information Conversion via Random Demodulation"[3]

[3] Kirolos et al., "Analog-to-Information Conversion via Random Demodulation", '06.

Dept. of Electronic Systems, Aalborg University,

Measurement matrix

How do we model our compressing kernel



We need precise discrete models of:

- impulse response of the low-pass filter
- chipping sequence mixed with the signal

Dept. of Electronic Systems, Aalborg University,

We can construct Φ from 3 sub-matrices

- **impulse response matrix:**

$$
\mathbf{H} = \begin{bmatrix}
h[0] & h[-1] & h[-2] & \dots & \dots & h[-N+1] \\
h[1] & h[0] & h[-1] & \ddots & & \vdots \\
h[2] & h[1] & \ddots & \ddots & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & h[-1] & h[-2] \\
\vdots & & \ddots & h[1] & h[0] & h[-1] \\
h[N-1] & \dots & \dots & h[2] & h[1] & h[0]
\end{bmatrix}, \qquad (1)
$$

- where $\mathbf{h} = [\, h[0], \dots, h[L-2], h[L-1]\,]^{\mathrm{T}} \in \mathbb{R}^{L\times 1}$ represents $L \leq N$ consecutive impulse response samples.

Measurement matrix

How do we model our compressing kernel

- **sampling matrix:**

$$\mathbf{B} = \bigoplus_{m=1}^{M} \boldsymbol{\kappa}, \in \{0,1\}^{M \times N}, \qquad (2)$$

where $\kappa \in \{0,1\}^{1\times R}$ such that:

$$\kappa[n] = \begin{cases} 1, & \text{for } n = 1 \\ 0, & \text{otherwise} \end{cases},$$

$\bigoplus$ denotes direct matrix sum.

- **pseudo-random sequence matrix:**

$$\mathbf{P} = \mathrm{diag}\{p[1], \dots, p[N]\}.$$

Dept. of Electronic Systems, Aalborg University,

Measurement matrix
Reconstruction stage

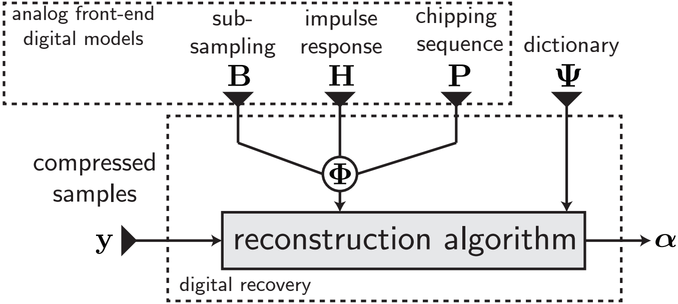


Figure : Information required for successful signal recovery

Existing CS architectures

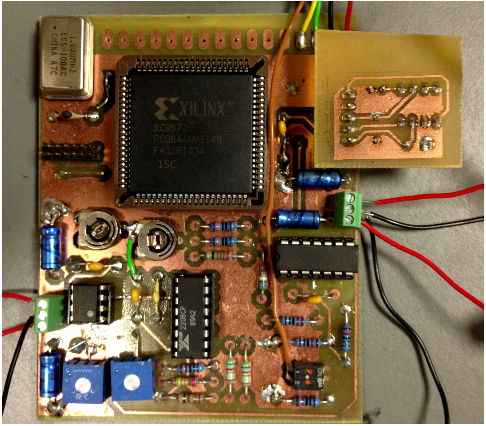
Influenced by random demodulator scheme

A lot of new architectures that share RD-like analog front-end - INSPIRED

- Modulated Wideband Converter[4]
- Random-Modulation Pre-Integrator - RMPI[5]
  - CMOS implementation
  - fully integrated
  - high bandwidth

[4] Mishali et al., "Xampling: Analog to digital at sub-Nyquist rates", '11.
[5] Candes et.al. Caltech, DARPA

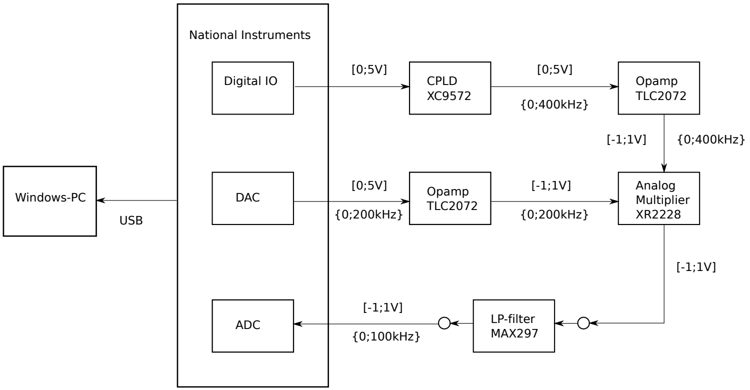
Hardware
RD alpha board at SparSig



- Early prototype, not very reliable
- Low-Pass filter on the module
- External DAQ unit for ADC/DAC (National Instruments)

Hardware

RD alpha board schematic



Dept. of Electronic Systems, Aalborg University,

Single-Pixel Camera
Compressed Sensing in Imaging

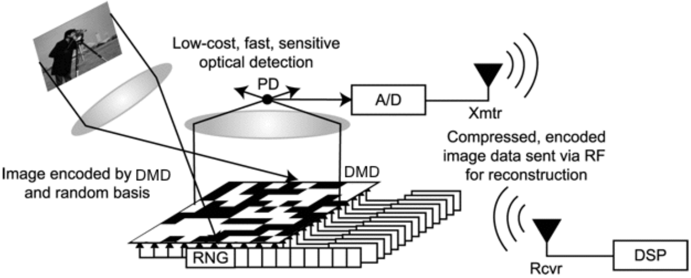


Figure : Compressive Imaging: A New Single-Pixel[6] Camera[7].

Rice University
Digital Signal Processing Group, Kelly Lab Department of Electrical and Computer Engineering.

[6] Duarte et al., "Single-Pixel Imaging via Compressive Sampling", '08.
[7] image from: `http://dsp.rice.edu/cscamera`.

Similarities to random demodulator:

- Micro-mirror array (MMA) – behaves as a chipping sequence.
- Photodiode – integrates the light reflected by MMA.

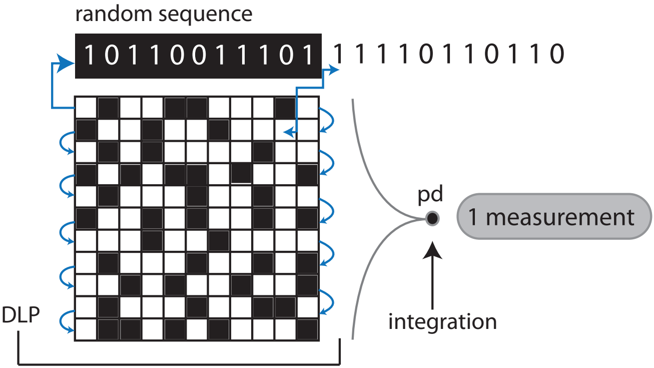


Figure : 1 pixel camera - measurement kernel.

Random demodulator
Non-ideal effects

Signal recovery can be compromised under following effects:

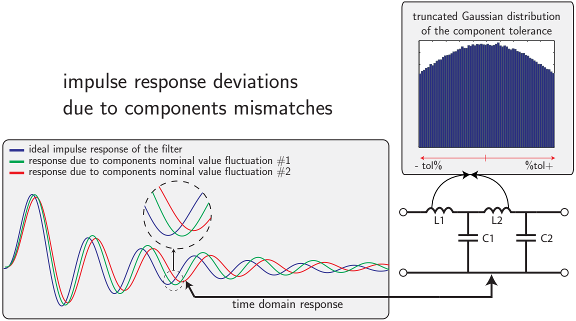
- low-pass filter non ideal response
- quantization error[8]
- clock jitter of the ADC
- mixer distortions
- amplifier nonlinearities (reported in RMPI)
- The algorithms are not designed to consider all hardware non-idealities

[8] Many new algorithms account for some level of quantization noise

Random demodulator

filter tolerances

impulse response deviations
due to components mismatches



$$\theta_{\mu,\sigma}(c) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(c-\mu)^2}{2\sigma^2}\right], & \text{for } |c-\mu| \leq \sigma \\ 0, & \text{otherwise} \end{cases} . \qquad (3)$$

Filter components deviations
Impulse response error bounds

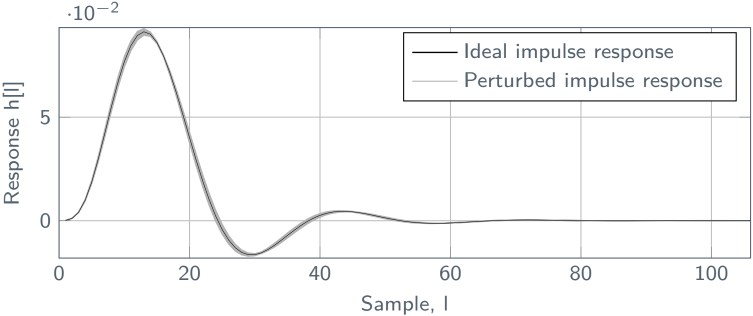


Figure : 2% components deviation in 4th order Butterworth-approximated low-pass filter

Filter components deviations
Impulse response error bounds

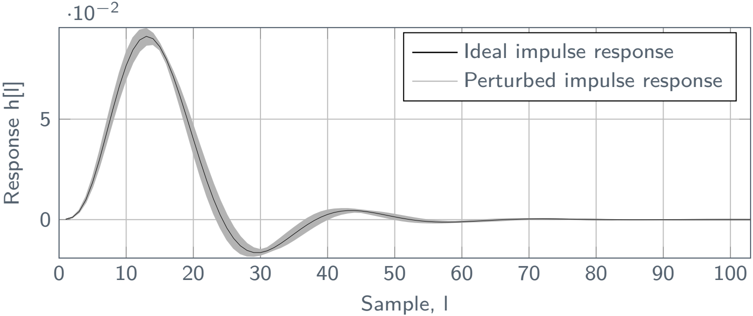


Figure : 5% components deviation in 4th order Butterworth-approximated low-pass filter

Filter components deviations

Impulse response error bounds

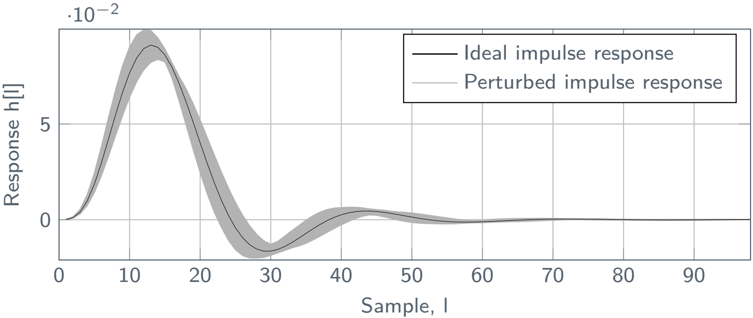


Figure : 10% components deviation in 4th order Butterworth-approximated low-pass filter

Sensitivities of the Random Demodulator

AALBORG UNIVERSITY
DENMARK

MATLAB framework designed to test filter non-idealities

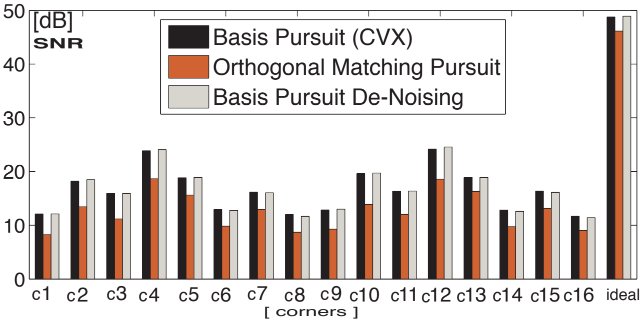
Simulations of the RD reconstruction sensitivity to filter tolerances with fixed input and chipping sequence

- Deterministic
  - Test deviation of one filter component at a time with other components assumed ideal
  - For low filter orders simulate all worst case combinations (*corners*)
- Stochastic
  - Test Monte-Carlo simulations with all filter components varying according to truncated Gaussian distribution of the filter components variations

Dept. of Electronic Systems, Aalborg University,

Sensitivities of the Random Demodulator

Simulation results

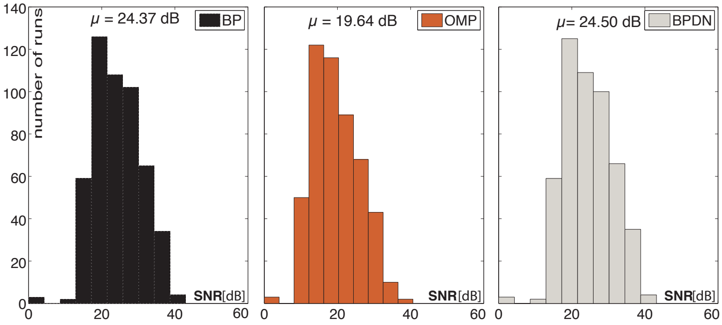


- 16 - corner tolerance values
- 4th order Butterworth filter
- BP, BPDN and OMP benchmarked
- 5% and 10% components deviation

Sensitivities of the Random Demodulator

Monte Carlo with only 1-component deviation

AALBORG UNIVERSITY
DENMARK



- 500 Monte Carlo runs
- 4th order Butterworth filter
- BP, BPDN and OMP benchmarked
- 5% and 10% components deviation

Sensitivities of the Random Demodulator

Single component variation influence

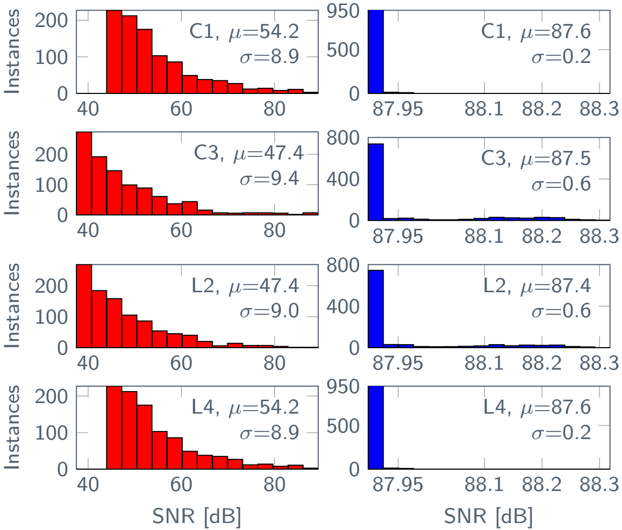
Monte Carlo simulation parameters

- 1000 Monte Carlo runs
- 4th order Butterworth filter
- BPDN reconstruction
- Up to 2% deviation in component

Dept. of Electronic Systems, Aalborg University,

Sensitivities of the Random Demodulator

Single component variation influence



System identification

- discrete Fourier transform trigonometric interpolation (DFTTI)
  - used to calibrate RMPI

Supervised calibration

Black-box approach

System identification

- discrete Fourier transform trigonometric interpolation (DFTTI)
  - used to calibrate RMPI
  - CS sample stream of $N-1$ signals of length N (frequency sweep)

Supervised calibration
Black-box approach

System identification

- discrete Fourier transform trigonometric interpolation (DFTTI)
  - used to calibrate RMPI
  - CS sample stream of $N-1$ signals of length N (frequency sweep)
  - very accurate

Dept. of Electronic Systems, Aalborg University,

System identification

- discrete Fourier transform trigonometric interpolation (DFTTI)
  - used to calibrate RMPI
  - CS sample stream of $N - 1$ signals of length N (frequency sweep)
  - very accurate
  - no initial model needed

Dept. of Electronic Systems, Aalborg University,

Supervised calibration
Black-box approach

System identification

- discrete Fourier transform trigonometric interpolation (DFTTI)
  - used to calibrate RMPI
  - CS sample stream of $N-1$ signals of length N (frequency sweep)
  - very accurate
  - no initial model needed
- requires $M \times (N-1)$ samples!

Dept. of Electronic Systems, Aalborg University,

- Assuming additive error in the discrete impulse response model:

$$\mathbf{H}_{\text{real}} = \mathbf{H} + \mathbf{E} \tag{4}$$

Error matrix

$$\mathbf{E} = \begin{bmatrix} e[0] & 0 & 0 & \dots & \dots & 0 \\ e[1] & e[0] & 0 & \ddots & & \vdots \\ e[2] & e[1] & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 & 0 \\ \vdots & & \ddots & e[1] & e[0] & 0 \\ e[N-1] & \dots & \dots & e[2] & e[1] & e[0] \end{bmatrix}, \in \mathbb{R}^{N\times N} \tag{5}$$

$$\mathbf{e} = \mathbf{h}_{\text{real}} - \mathbf{h} \tag{6}$$

Model-based calibration
Finding modeled impulse response error

- Real compressed measurements

$$\mathbf{y}_{\mathrm{real}} = \Phi_{\mathrm{real}}\mathbf{x}_{\mathrm{in}} \tag{7}$$

$$\mathbf{y}_{real} = \mathbf{y} + \mathbf{BEPx}_{\mathrm{in}} \tag{8}$$

- Ideal measurements (calculated)

$$\mathbf{y} = \Phi\mathbf{x}_{\mathrm{in}} \tag{9}$$

- Model discrepancy

$$\mathbf{y}_{\mathrm{real}} - \mathbf{y} = \mathbf{BEPx}_{\mathrm{in}}. \tag{10}$$

Model-based calibration
Finding modeled impulse response error

- The roles of $\mathbf{E}$ and $\mathbf{x}_{\text{in}}$ can be interchanged as follows:

$$\mathbf{BEPx}_{\text{in}} = \mathbf{De} \tag{11}$$

where:

$$\mathbf{D} = \begin{bmatrix} d[1] & \dots & d[L] \\ d[R+1] & \dots & d[R+L] \\ d[2R+1] & \dots & d[2R+L] \\ \vdots & & \dots \\ d[N-L+1] & \dots & d[N] \end{bmatrix} \in \mathbb{R}^{M \times L}. \tag{12}$$

- consists of the input signal mixed with the chipping sequence
- when $L > R$, $\mathbf{D}$ truncating its first L/R rows and discarding the first L/R measurements of $\mathbf{y}_{\text{real}}$
- $L/R \in \mathbb{Z}$

Calibration principle

Least squares error estimation

- Equation (10) can be further rewritten using (11) to the following form:

$$\mathbf{D}\,\mathbf{e} = \mathbf{y}_{\mathrm{real}} - \mathbf{y}. \tag{13}$$

Least squares estimation

$$\underset{\mathbf{e}\in\mathbb{R}^{L\times 1}}{\text{minimize}} \;\; \|\mathbf{D}\mathbf{e} - \mathbf{y}_{\mathbf{d}}\|_2^2, \tag{14}$$

LS Tikhonov regularization

$$\begin{aligned} \underset{\mathbf{e}\in\mathbb{R}^{L\times 1}}{\text{minimize}} \quad & \|\mathbf{D}\mathbf{e} - \mathbf{y}_{\mathbf{d}}\|_2^2 \\ \text{subject to} \quad & \|\mathbf{G}\mathbf{e}\|_2^2 \le \gamma, \end{aligned} \tag{15}$$

Dept. of Electronic Systems, Aalborg University,

Calibration performance

Monte Carlo simulation - Butterworth

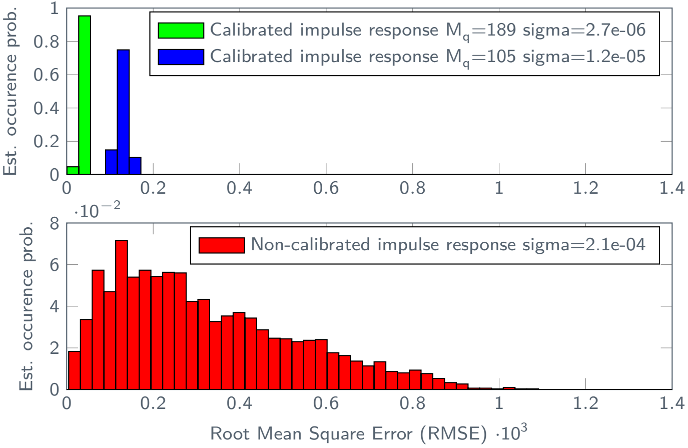


Figure : 2% components deviation, 3000 cases, L = 108, $M_{q1} = 189$ and $M_{q2} = 105$, K = 10

Calibration performance

Monte Carlo simulation - Chebyshev

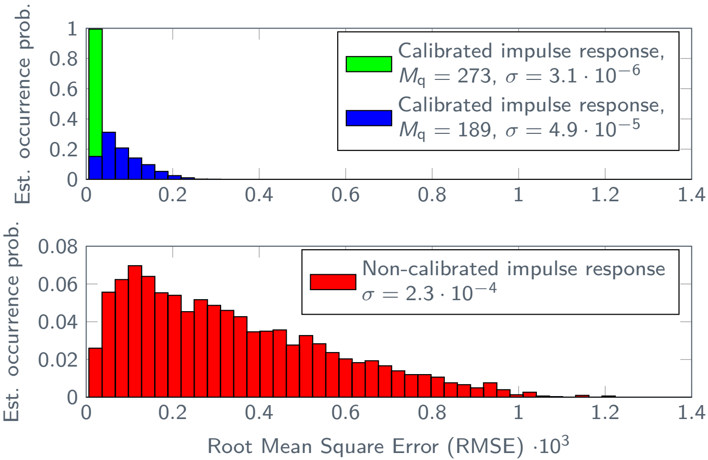


Figure : 2% components deviation, 3000 cases, L = 238, $M_{q1} = 273$ and $M_{q2} = 189$, K = 10

Calibration performance

Monte Carlo simulation - Butterworth

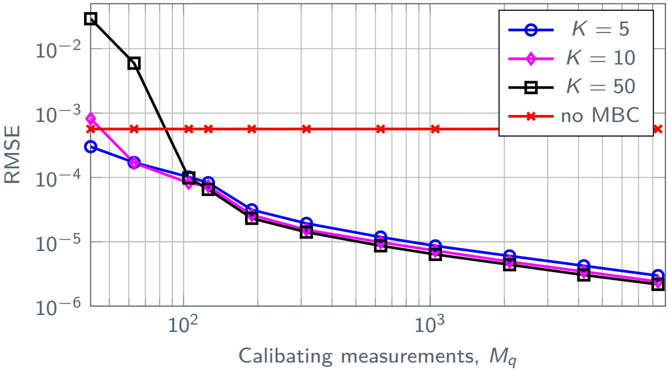
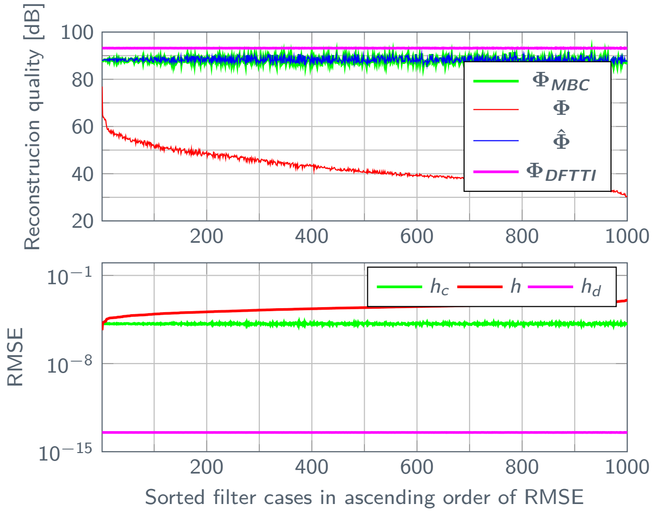


Figure : Calibration performance of the impulse response in RMSE versus the amount of samples used in the least-square estimation (16)

Calibration performance
DFTTI benchmark



Dept. of Electronic Systems, Aalborg University,

LS estimation
Under quantization

More realistic scenario

- What happens when we include quantization ?
- e.g., 4 bit uniform quantizer
- We have only preliminary results

Dept. of Electronic Systems, Aalborg University,

Calibration with quantization included

Results: reconstruction

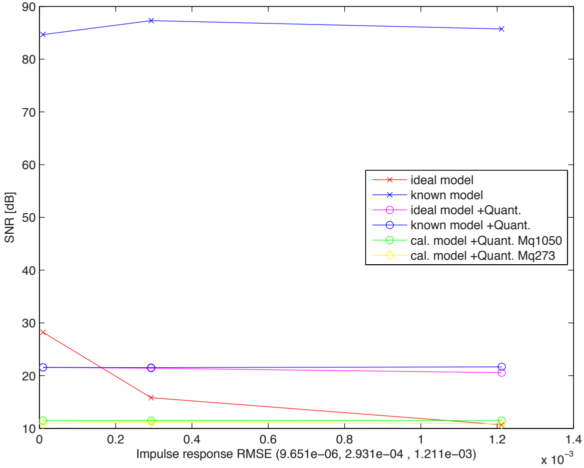


Figure : Reconstruction quality in RD with quantization

Dept. of Electronic Systems, Aalborg University,

Calibration with quantization included

Results: estimated impulse error

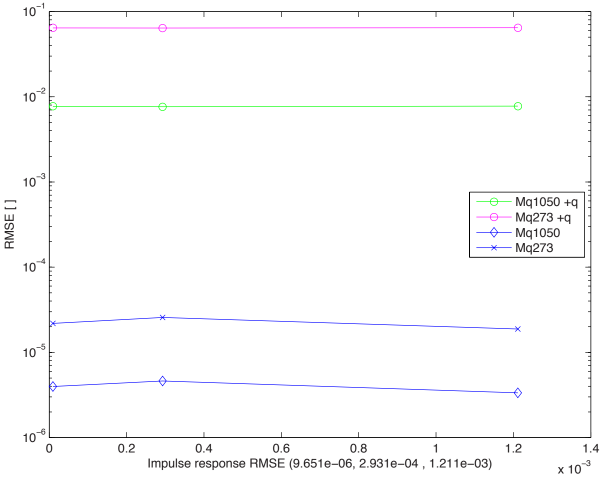


Figure : Calibration error with quantization

Calibration principle and regularization
Impulse response matrix error

AALBORG UNIVERSITY
DENMARK

Least squares estimation

$$\underset{\mathbf{e}\in\mathbb{R}^{L\times 1}}{\text{minimize}} \ \|\mathbf{De}-\mathbf{y_d}\|_2^2, \tag{16}$$

Least squares regularized

$$\begin{aligned} \underset{\mathbf{e}\in\mathbb{R}^{L\times 1}}{\text{minimize}} \quad & \|\mathbf{De}-\mathbf{y_d}\|_2^2 \\ \text{subject to} \quad & \mathbf{e} \leq \text{UpperImpulseErrorBound}, \\ & \mathbf{e} > \text{LowerImpulseErrorBound}, \end{aligned} \tag{17}$$

LS estimation
Under quantization (4 bit)

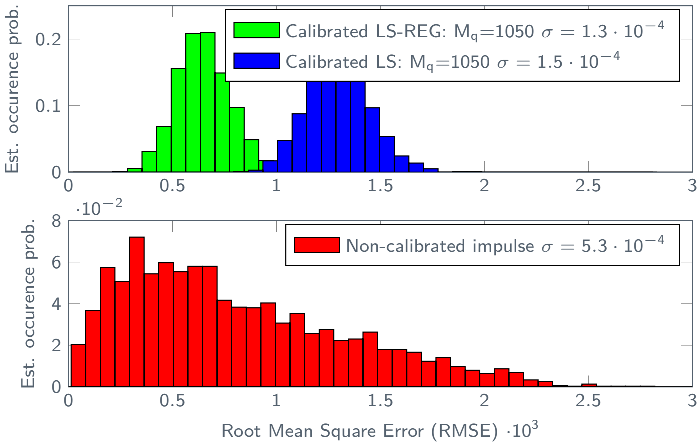


Figure : RMSE histogram - iRMSE (8.086e-04) vs. cRMSER (1.759e-03) vs. cRMSEnR (2.035e-02)

LS estimation
Under quantization (4 bit)

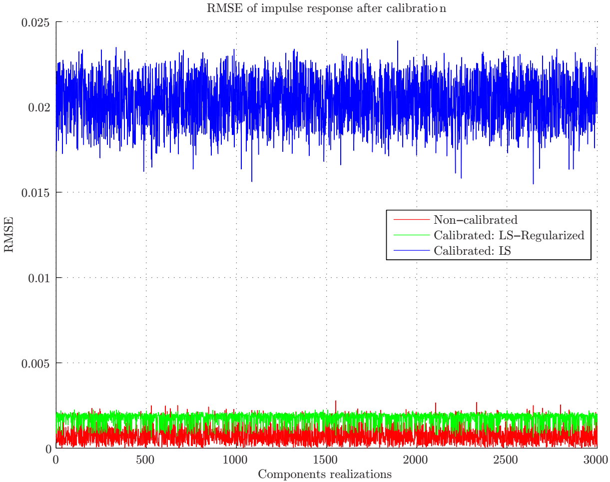


Figure : RMSE case-by-case

Dept. of Electronic Systems, Aalborg University,

LS estimation
Under quantization (4 bit)

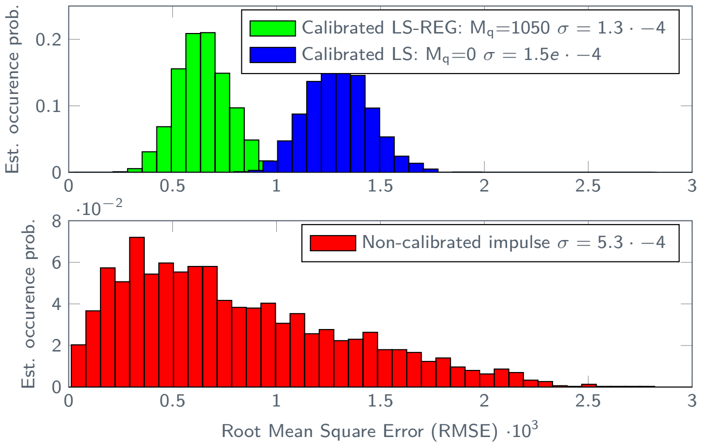


Figure : RMSE histogram - iRMSE (8.086e-04) vs. cRMSER (6.596e-04) vs. cRMSEnR (1.288e-03)

LS estimation
Under quantization (8 bit)

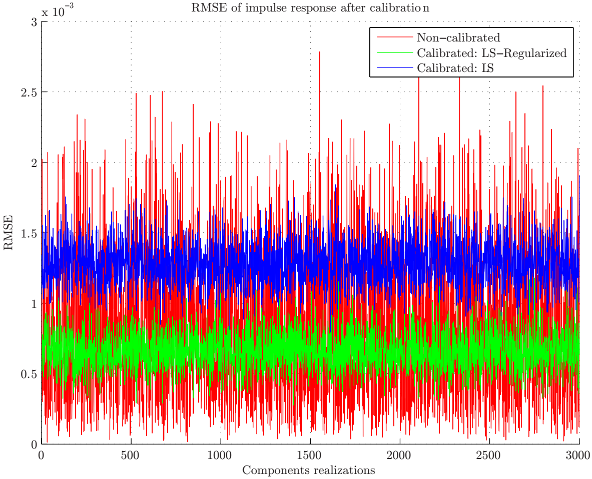


Figure : RMSE case-by-case

Thank you!
Contact Information

Pawel Jerzy Pankiewicz

pjp@es.aau.dk

@pawelpankiewicz

Niels Jernes Vej 12, A6-112

9220 Aalborg Ø

www.sparsesampling.com

Thank you!

Questions?

www.sparsesampling.com

Dept. of Electronic Systems, Aalborg University,