SDR-Based Channel Emulator using a Semi-Stochastic Radio Propagation Model

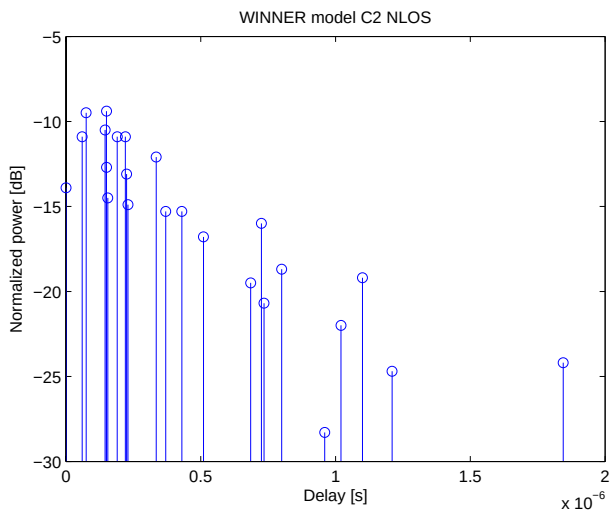
J. Schmitz, X. Xu, F. Schröder, M. Zivkovic, R. Mathar

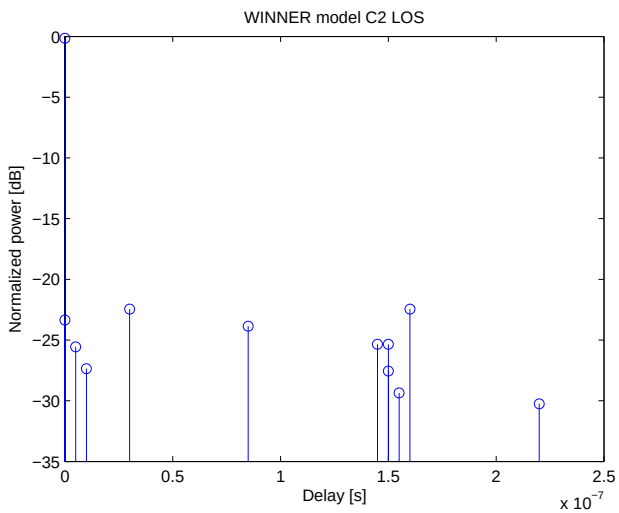
Institute for Theoretical Information Technology
Prof. Dr. Rudolf Mathar
RWTH Aachen University

June 12, 2013

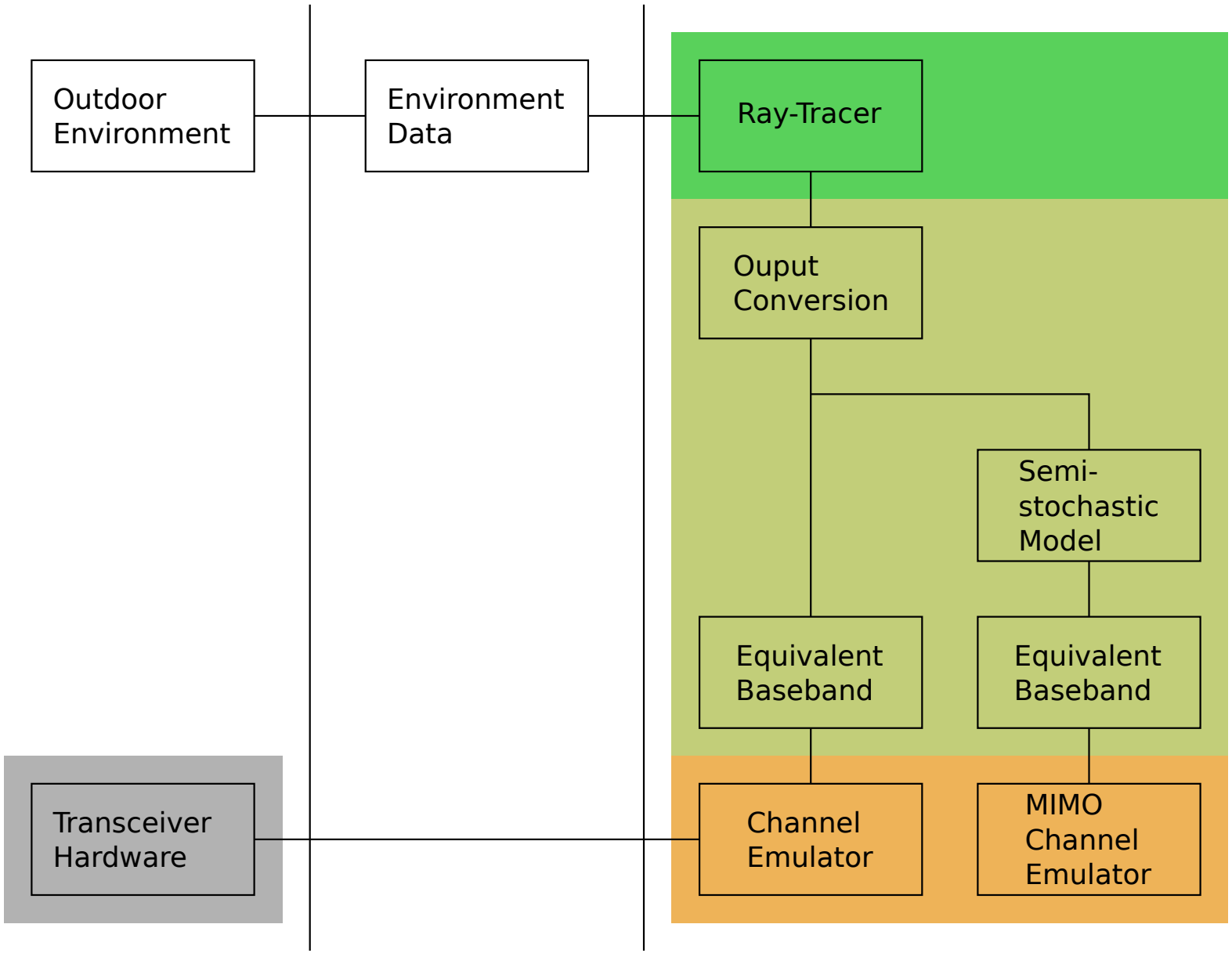
Motivation

- From stochastic channel models to more realistic, site aware models
- Software defined channel emulation
- Hardware in the Loop testing





Workflow Overview

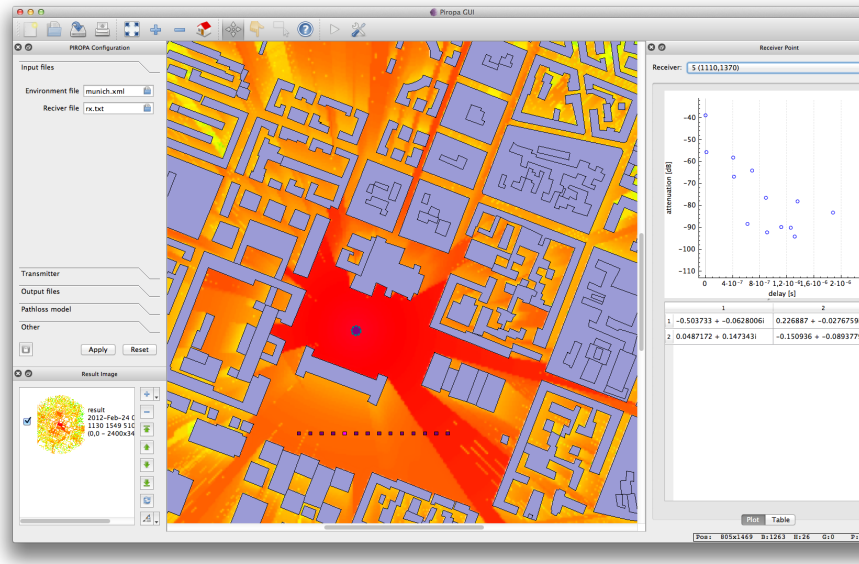


Ray optical path simulation
What is PIROPA?

PIROPA is a non-interactive C++ command line program, a Demonstration GUI facilitates the usage.

Input

- Vector based environment data,
- a transmitter location t, andTo
- receiver locations r_i

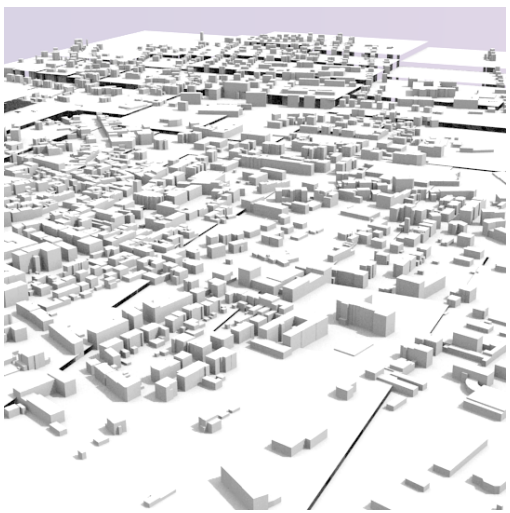


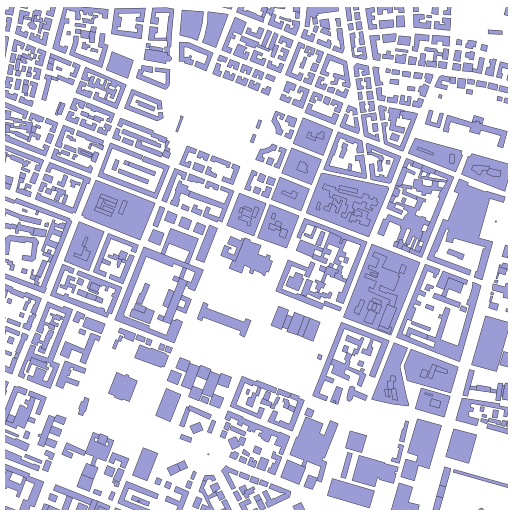
Output

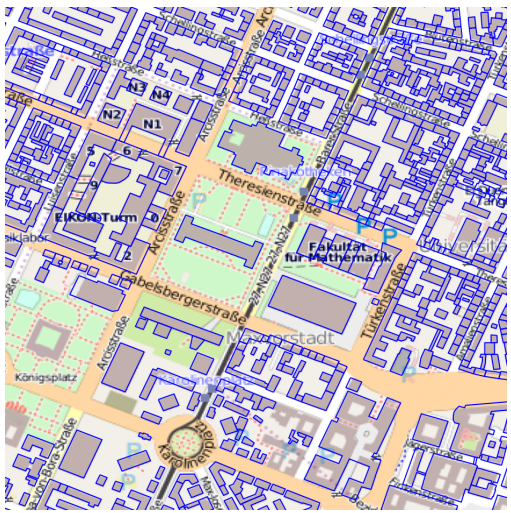
- Sets of paths $\{p|p \text{ is a path from } t \text{ to } r_i\}$, where
- the path p is a sequence $(t, c_1, c_2, \ldots, c_{n-1}, r_i)$ and
- $t, r_i, c_i \in \mathbb{R}^3$ are locations in the environment space.

Ray optical path simulation
Environment data

- Environment data is vector based 2.5D building data
- All walls are upright and roofs are flat
- Data is available from governmental, commercial and non-commercial sources (e.g. OSM)
- Dealing with the data quality, or lack of quality, is a challenge
- There is no unified standard format for such data
- A preprocessing tool creates a custom input format for PIROPA







Basic Path loss model

$$L^{dB}(p) = L_0^{dB}(p) + \sum_{e \in E} \sum_{i=1}^{n_e(p)} L_e^{dB}(\alpha)$$

$$L_0^{dB}(p) = 20 \lg \frac{4\pi}{\lambda_t} - 10 \lg G_t(\phi(r), \psi(r)) + z_A + 10\gamma \lg d(r)$$

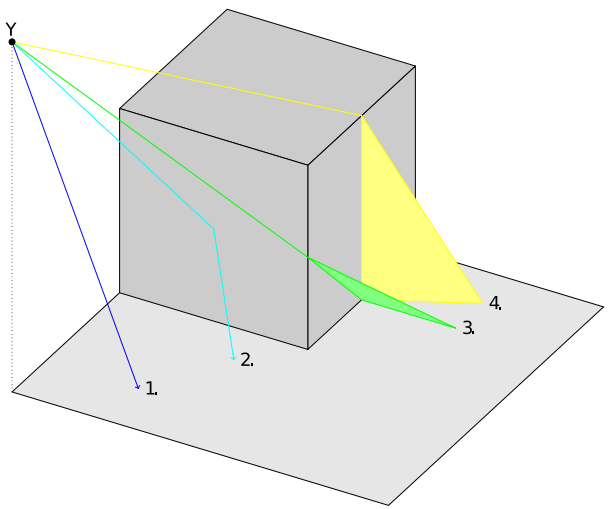
- λ_t signal wavelength
- $G_t(\phi(r), \psi(r))$ antenna gain in the given direction
- z_A calibration coefficient
- γ path loss exponent
- $d(r)$ the passed distance of the path

Ray optical path simulation
From path to color image

Basic Path loss model

$$L^{dB}(p) = L_0^{dB}(p) + \sum_{e \in E} \sum_{i=1}^{n_e(p)} L_e^{dB}(\alpha)$$

- $E = \{\text{Reflection}, \text{Vertical diffraction}, \text{Horizontal diffraction}\}$



- $n_e(p)$ The number of occurrences of an effect $e \in E$

Basic Path loss model

$$L^{dB}(p) = L_0^{dB}(p) + \sum_{e \in E} \sum_{i=1}^{n_e(p)} L_e^{dB}(\alpha)$$

$$L_e^{dB}(\alpha) = \sum_{j=0}^{k} z_{e,j} \alpha_{e,i}^{j}(p)$$

- $\alpha_{e,i}^{j}(p)$ is the angular change at a certain position
- $z_{e,j}$ calibration coefficients for effects

PIROPA Algorithm

1. Shoot rays from the transmitter location into the scene

PIROPA Algorithm

1. Shoot rays from the transmitter location into the scene
2. Intersect the rays with the building surfaces

PIROPA Algorithm

1. Shoot rays from the transmitter location into the scene
2. Intersect the rays with the building surfaces
3. Test the found intersection points for occurrences of physical effects

PIROPA Algorithm

1. Shoot rays from the transmitter location into the scene
2. Intersect the rays with the building surfaces
3. Test the found intersection points for occurrences of physical effects
4. Insert a virtual transmitter according to the effect parameter

PIROPA Algorithm

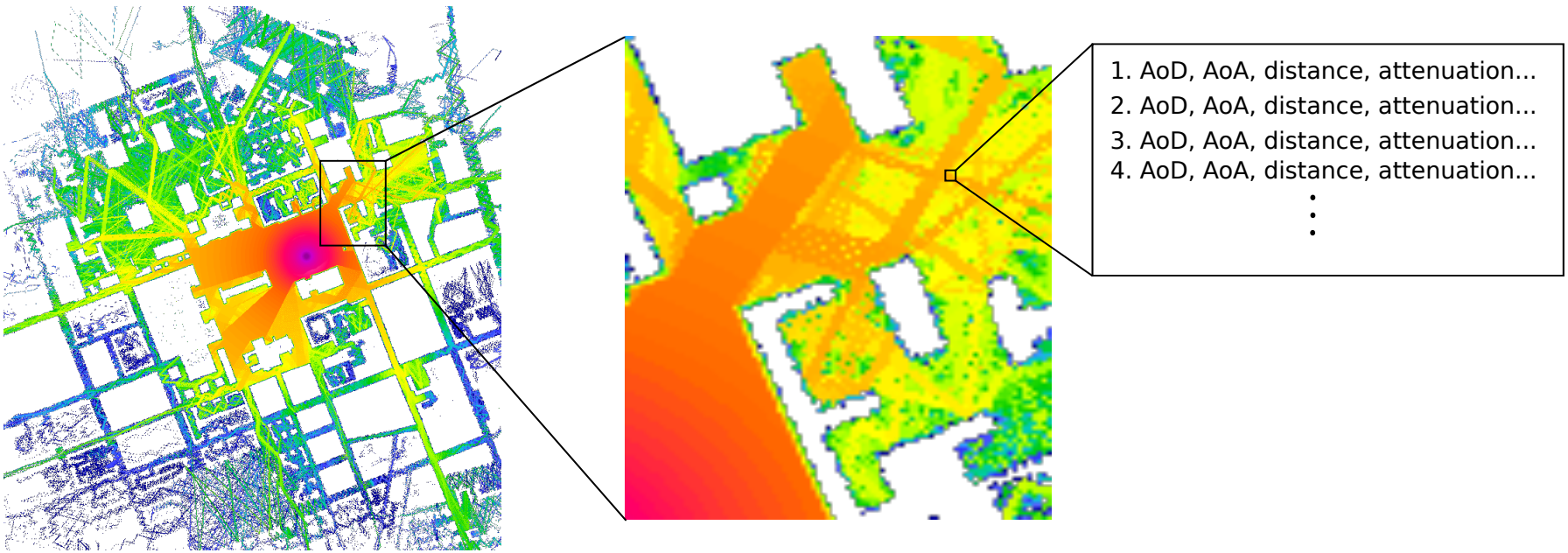
1. Shoot rays from the transmitter location into the scene
2. Intersect the rays with the building surfaces
3. Test the found intersection points for occurrences of physical effects
4. Insert a virtual transmitter according to the effect parameter
5. Start over until termination condition is met

Ray optical path simulation
Simulation Performance

- Many Tweaks to increase performance
- Large environment data is partitioned to increase performance on the intersection test and improve memory handling. This is completely transparent to the simulation.
- The typical building structure allows for some assumptions which leads to a faster calculations. A two times 2D calculation instead of a full 3D calculation is performed.
- OpenCL to employ a multitude of heterogeneous computing hardware

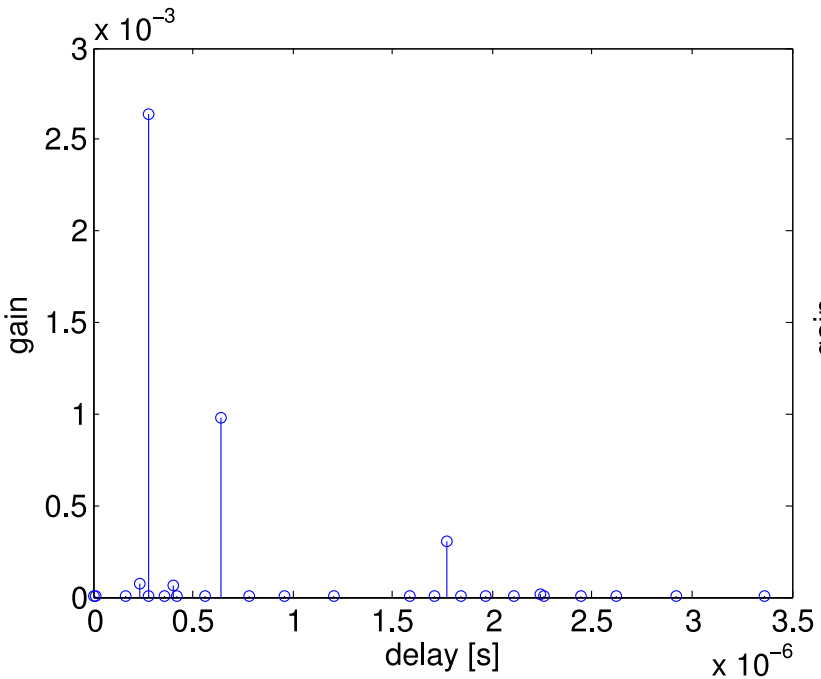
Ray optical path simulation
Output data

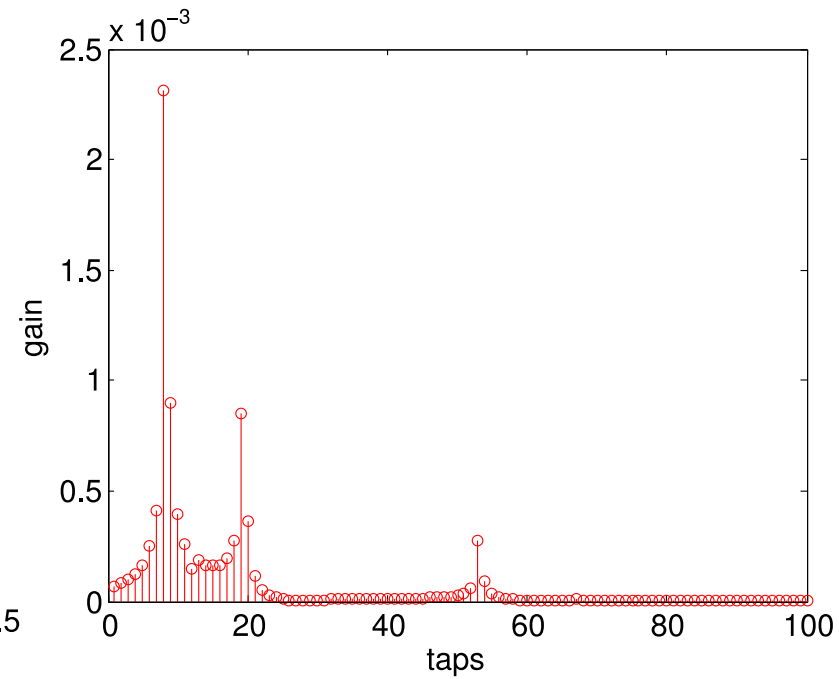
- The output data is a text file or an image representation
- In the image, every pixel is a receiver and the attenuation is displayed by the color
- The text format contains the complete set of information of all paths for a given receiver location
  - extract channel (emulation) data with a conversion tool



Channel Emulator
Equivalent Baseband Conversion

- RWP Simulation outputs discrete "impulse response"/power delay profile (after conversion)
- Pulses not on a uniform sampling grid
- Equivalent baseband conversion is necessary to obtain filter coefficients
- We use an FFT filter design method for the interpolation





Channel Emulator

Equivalent Baseband Conversion

$$h(t) = \sum_m \alpha_m \delta(t - \tau_m T_s) \qquad \tau_m \in \mathbb{R}$$

$$H(f) = \sum_m \alpha_m e^{-j2\pi f \tau_m T_s}$$

$$H_k = \sum_m \alpha_m e^{-j2\pi k f_s \tau_m T_s} \qquad f_s = \frac{1}{N T_s}$$

$$= \sum_m \alpha_m e^{-j2\pi \frac{k}{N} \tau_m}$$

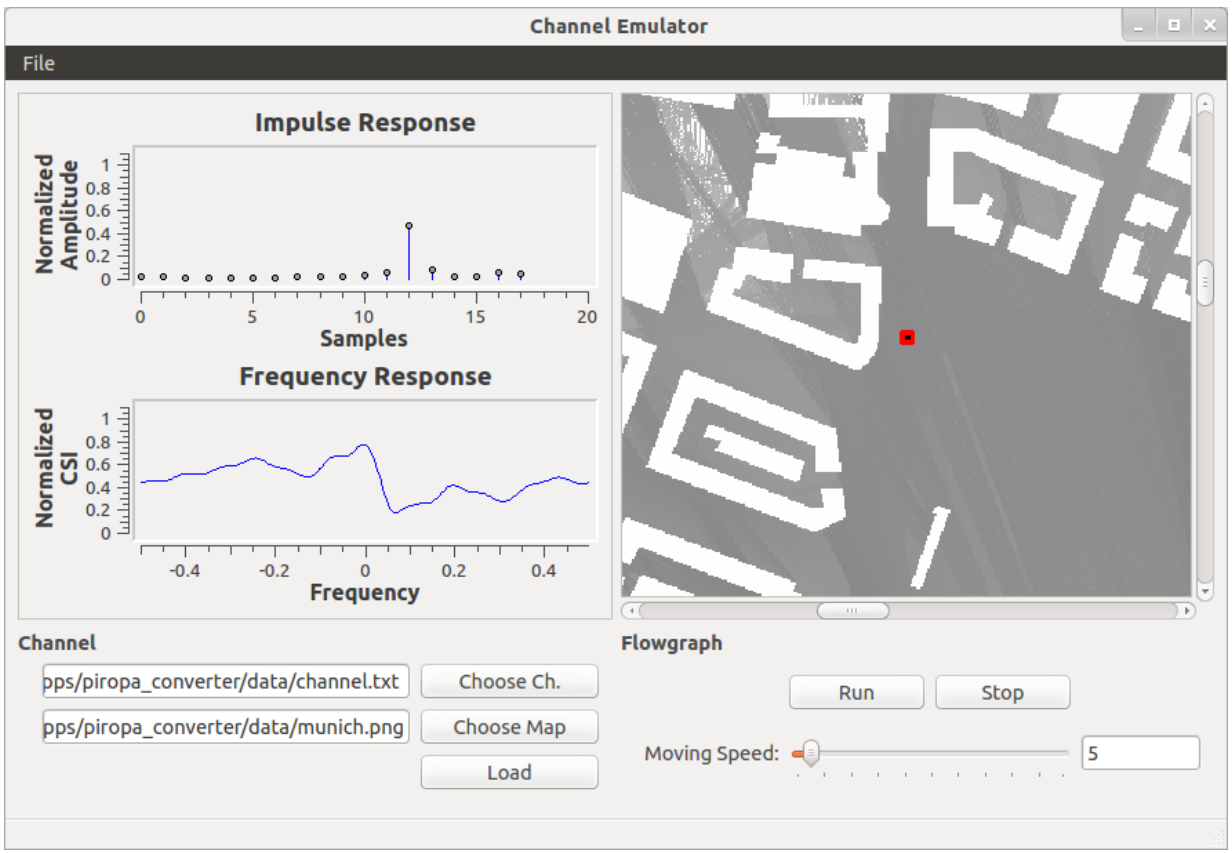
$$h_n = \sum_{k=0}^{N-1} H_k e^{j2\pi \frac{k}{N} n}$$

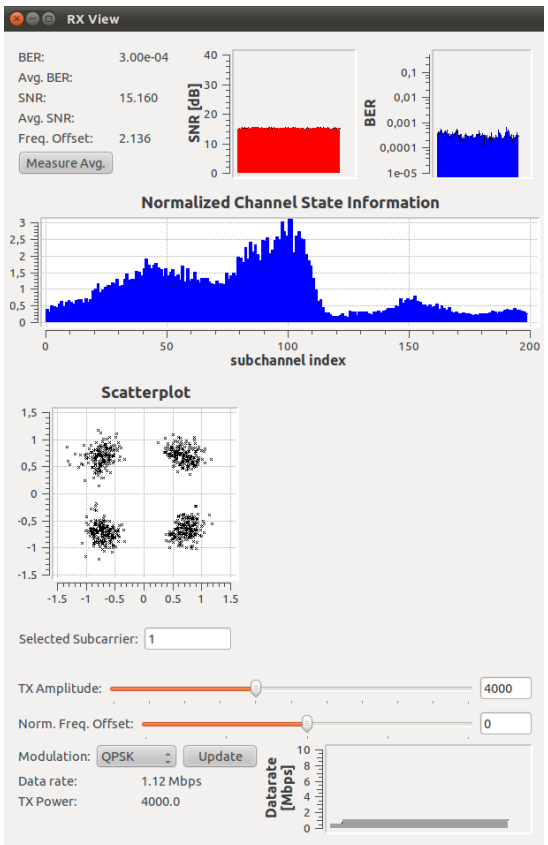
$$\vdots$$

$$= \sum_m \alpha_m e^{-j\frac{\pi}{N}(n+(N-1)\tau_m)} \frac{\sin(\pi\tau_m)}{\sin(\frac{\pi}{N}(\tau_m - n))}$$

Channel Emulator
Software

- Speed of receiver movement defines channel coherence time

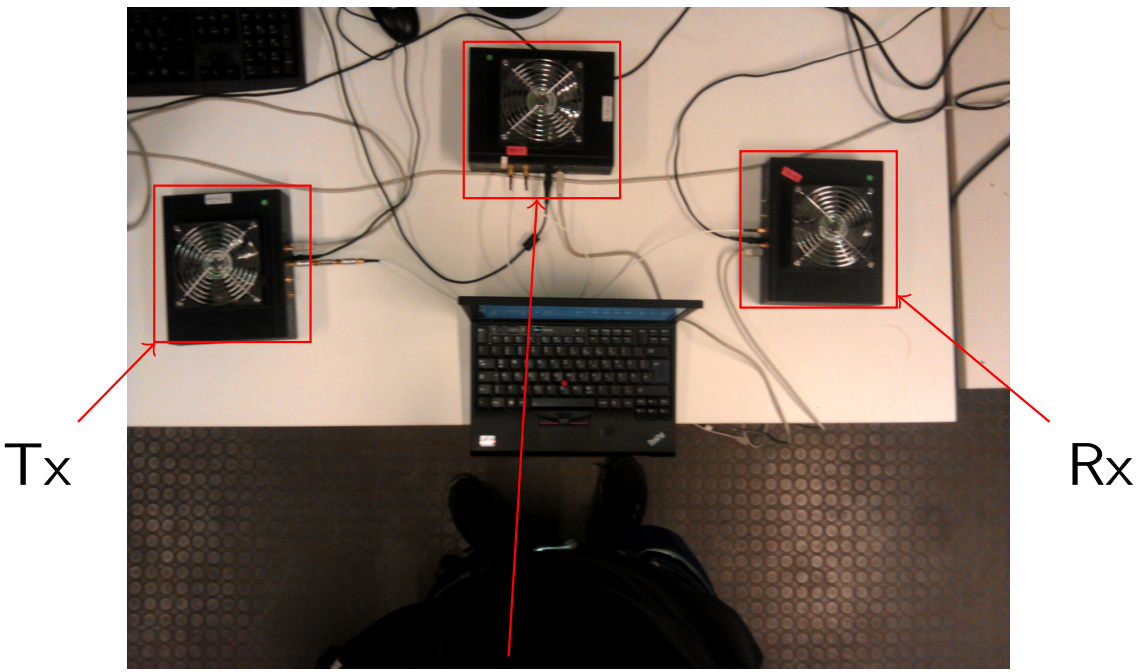




Channel Emulator

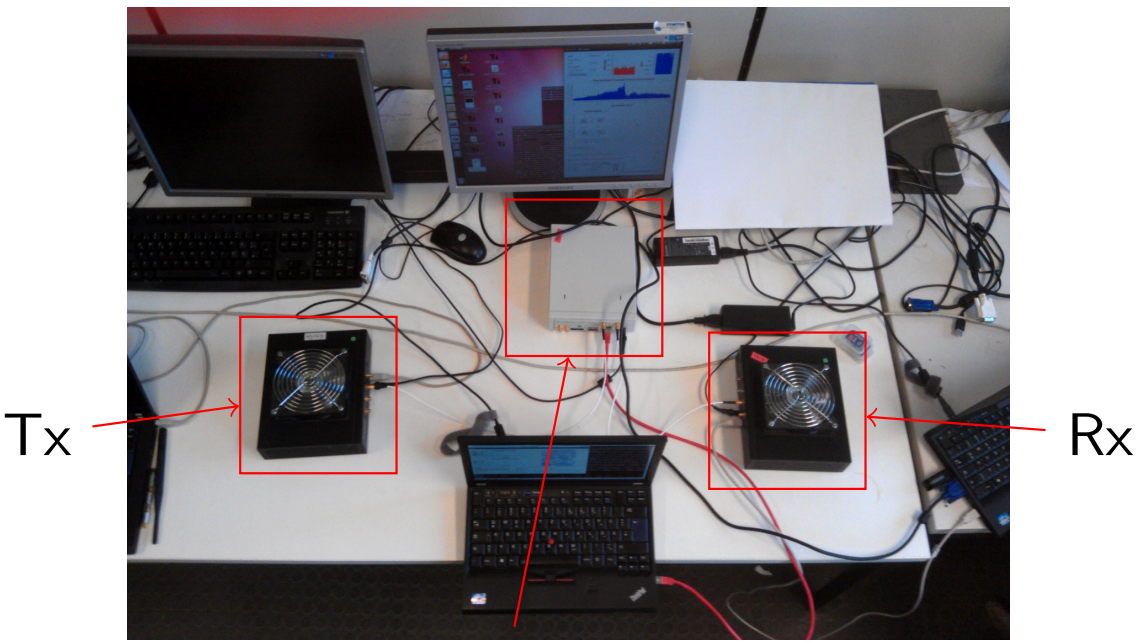
Hardware Testing Configuration

- some testing
- problems with coupling of signals, we used different frequencies for input and output
- problems with additional filtering effect of USRP1, halfband filters etc.

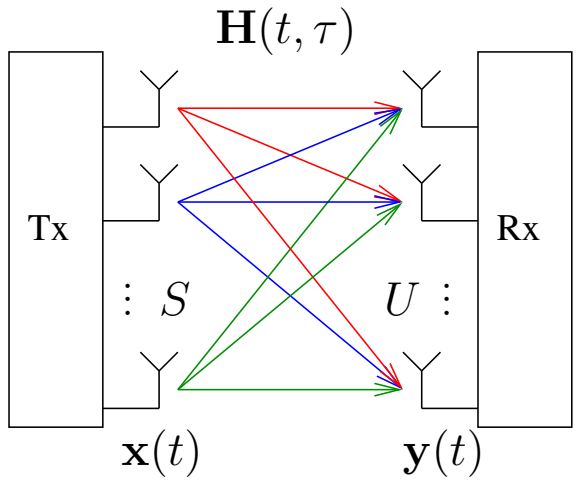


Channel Emulator
Hardware Testing Configuration

- some testing
- problems with coupling of signals, we used different frequencies for input and output
- problems with additional filtering effect of USRP1, halfband filters etc.
- USRP2 works much better



MIMO system model



Received signal:

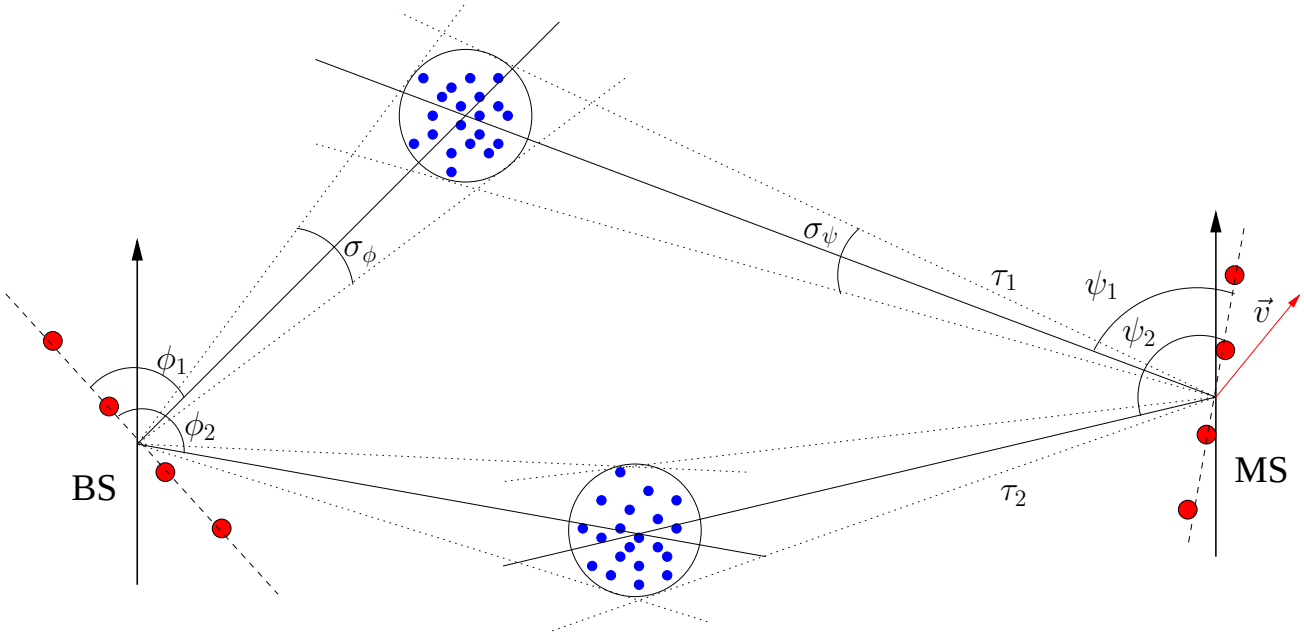
$$\mathbf{y}(t) = \int_{\tau} \mathbf{H}(t;\tau)\mathbf{x}(t-\tau)\mathrm{d}\tau + \mathbf{w}(t)$$

with

$$\mathbf{H}(t;\tau) = \begin{pmatrix} h_{1,1}(t;\tau) & h_{1,2}(t;\tau) & \dots & h_{1,S}(t;\tau) \\ h_{2,1}(t;\tau) & h_{2,2}(t;\tau) & \dots & h_{2,S}(t;\tau) \\ \vdots & \vdots & \ddots & \dots \\ h_{U,1}(t;\tau) & h_{U,2}(t;\tau) & \dots & h_{U,S}(t;\tau) \end{pmatrix}$$

Geometry-based stochastic channel model

Clusterized MIMO channel:



Double directional radio channel:

$$h_{u,s}(t;\tau) = \int_{\phi}\int_{\psi} g_{u,s}(t,\tau,\phi,\psi)\cdot \sqrt{G_{\mathrm{Tx},s}(\phi)}\sqrt{G_{\mathrm{Rx},u}(\psi)}\mathrm{d}\phi\mathrm{d}\psi$$

Output of deterministic step

From Ray-Tracing algorithm:

Parameters	Description
n, N	Path index, total number of paths (clusters)
G_n	Per path gain $\in [0, 1)$
d_n	Propagation distance
ϕ_n, ψ_n	Path AoD and AoA

Normalized power:

$$P_n = \frac{G_n}{\sum_{n=1}^{N} G_n}$$

Delay:

$$\tau_n = d_n / c$$

Generation of clusters

- The N Propagation paths from ray launching algorithm are regarded as representatives of clusters for MIMO channel model
- Generate M rays for each cluster

Parameters	Description
m, M	Ray index, number of rays per cluster
c_{AoA}	Cluster-wise RMS azimuth spread
α_m	Ray offset angles
$\vec{k}_{\mathrm{Tx},n,m}, \vec{k}_{\mathrm{Rx},n,m}$	Ray direction vectors
$\Phi_{n,m}$	Random initial phase $\in [0, 2\pi)$
$\vec{d}_{\mathrm{Tx},s}, \vec{d}_{\mathrm{Rx},u}$	Antenna element position vectors
v, θ	Speed and direction of mobile station

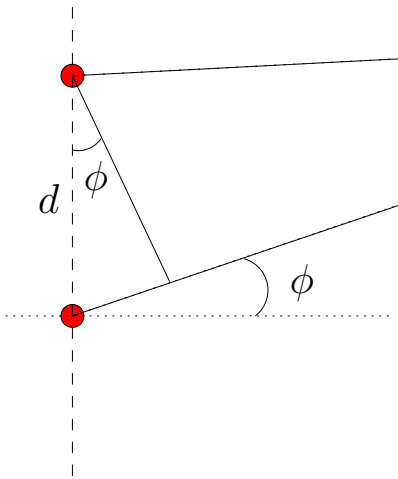
Angle of arrival:

$$\psi_{n,m} = \psi_n + c_{\mathrm{AoA}}\alpha_m$$

Channel coefficients

Channel coefficients based on clusters:

$$h_{u,s}(t,\tau) = \sqrt{\frac{P_n}{M}} \sum_{n=1}^{N} \sum_{m=1}^{M} \sqrt{G_{\mathrm{Tx},s}(\phi_{n,m})} \sqrt{G_{\mathrm{Rx},u}(\psi_{n,m})}$$



$$\exp\left(\jmath\left[\frac{2\pi}{\lambda}\vec{k}_{\mathrm{Tx},n,m}\cdot\vec{d}_{\mathrm{Tx},s}+\Phi_{n,m}\right]\right)$$

$$\exp\left(\jmath\frac{2\pi}{\lambda}\vec{k}_{\mathrm{Rx},n,m}\cdot\vec{d}_{\mathrm{Rx},u}\right)$$

$$\exp\left(\jmath\frac{2\pi}{\lambda}v_{n,m}t\right)\delta(\tau-\tau_{n,m})$$

Doppler frequency component:

$$v_{n,m} = \|v\|\cos(\psi_{n,m}-\theta)$$

Conclusion

- Next: Implement MIMO model into Channel Emulator
- Extensive Testing with 2x2 MIMO OFDM
- Take effects of the hardware into account

Finish

Thank You! Questions?