MULTIDIMENSIONAL PARAMETER ESTIMATION WITH DEEP LEARNING

M.Sc. Steffen Schieler

Thüringer Innovationszentrum MOBILITÄT

Fraunhofer IIS

TECHNISCHE UNIVERSITÄT ILMENAU

Freistaat Thüringen

Ministerium für Wirtschaft, Wissenschaft und Digitale Gesellschaft

OUTLINE

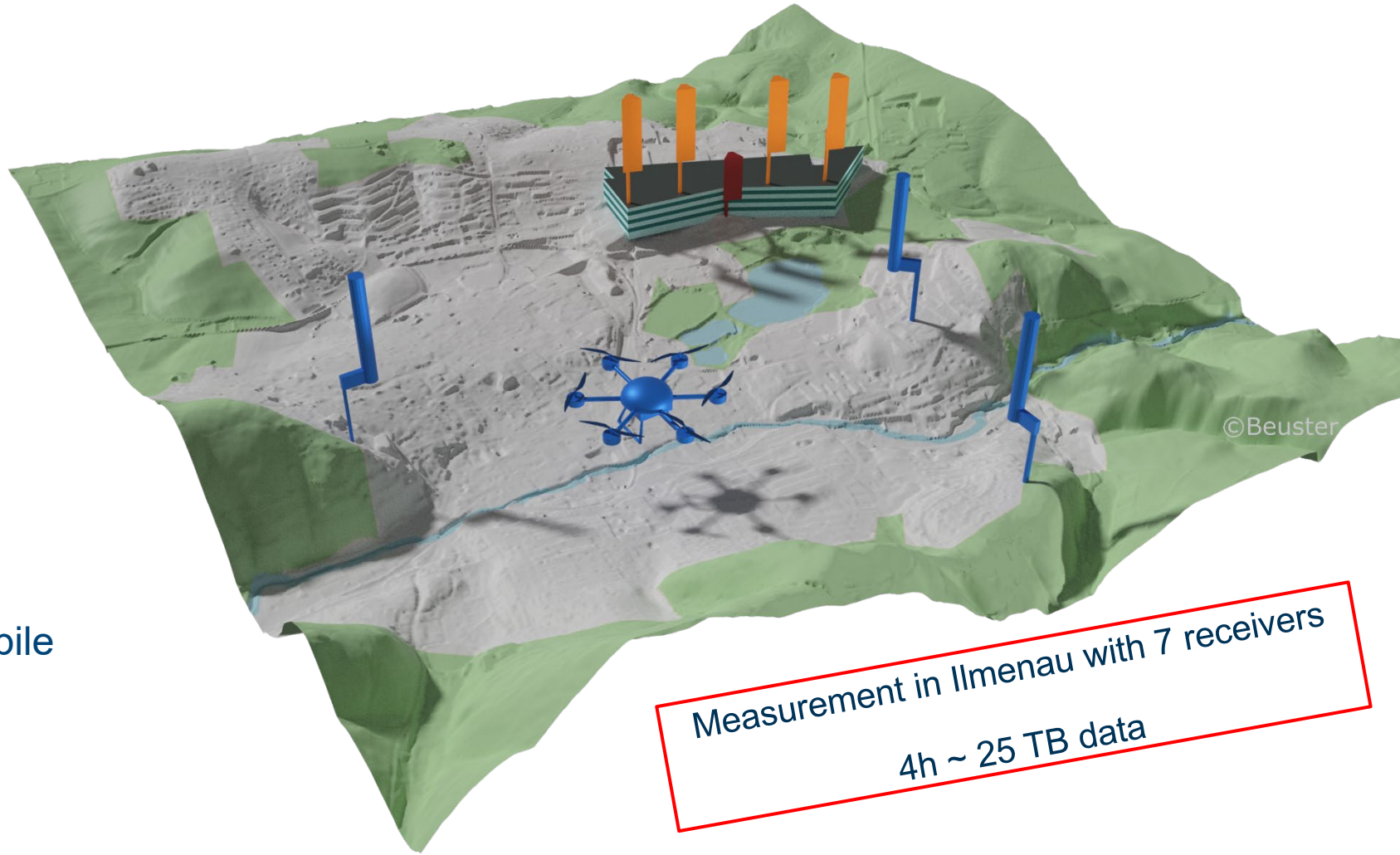
Multidimensional Parameter Estimation with Deep Learning

- Introduction
  - Wireless propagation
  - 1-D parameter estimation
- Evolution from 1-D to N-D
- Preliminary results
- Summary and outlook

Introduction

Drone-Shield

- Use existing wireless communication infrastructure to perform radar detection
- Similar to passive radar
- Uses OFDM(A) waveform
  - LTE, 5G, WiFi 802.11
- E.g. to detect UAVs with an mobile communication network

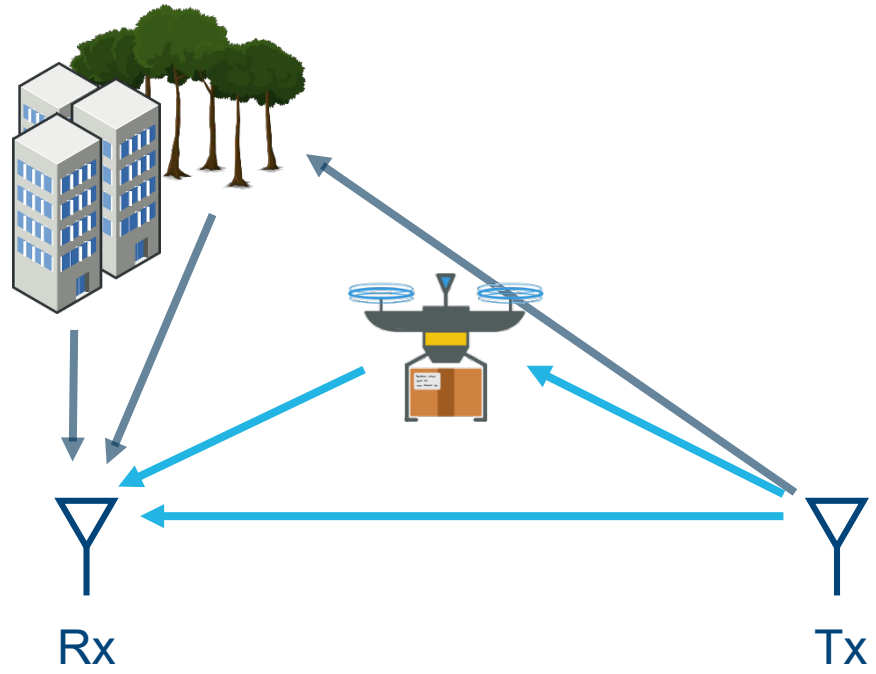


Measurement in Ilmenau with 7 receivers

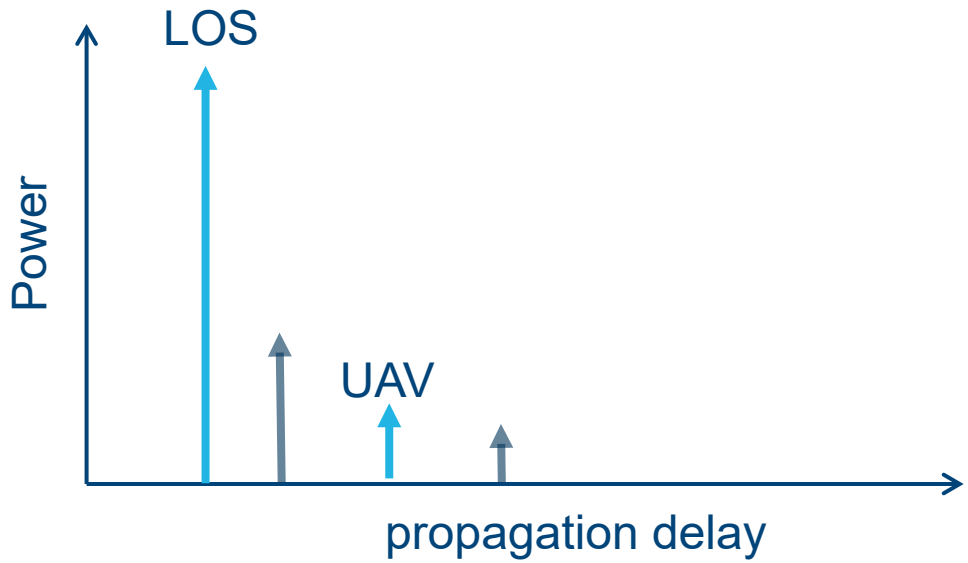
4h ~ 25 TB data

INTRODUCTION

Wireless Propagation and Parameter Estimation



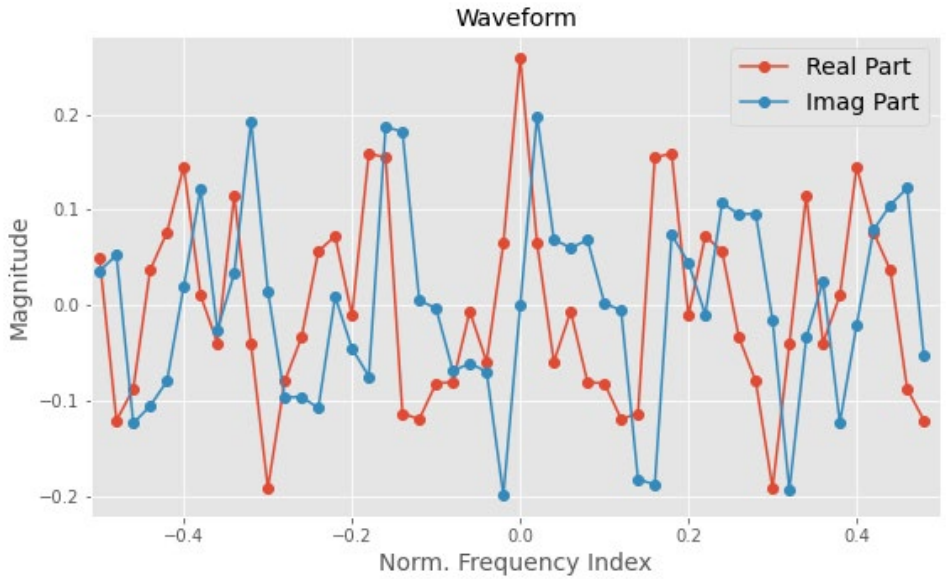
Radar localization

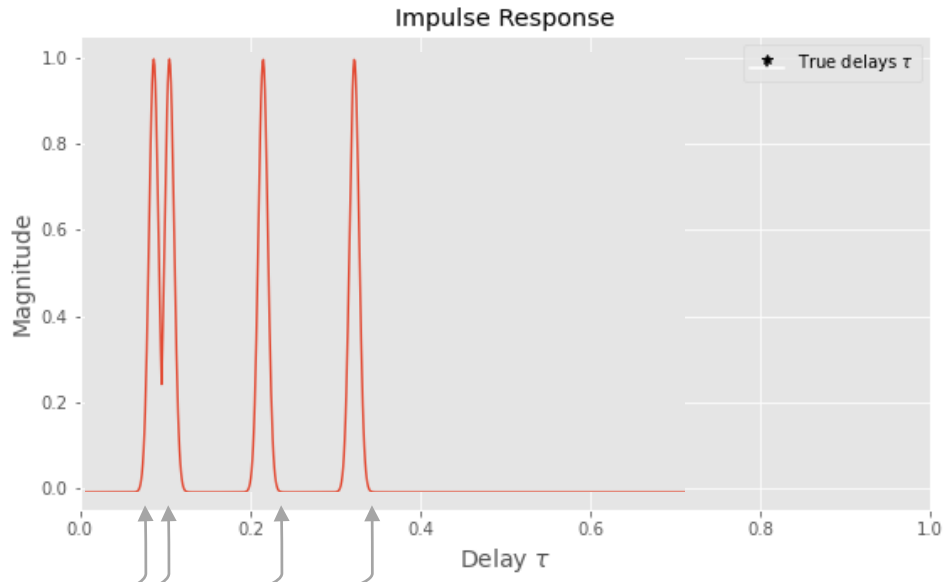


QUICK RECAP

1D parameter estimation

Estimate a pseudo PDF for the parameters space from a channel transfer function



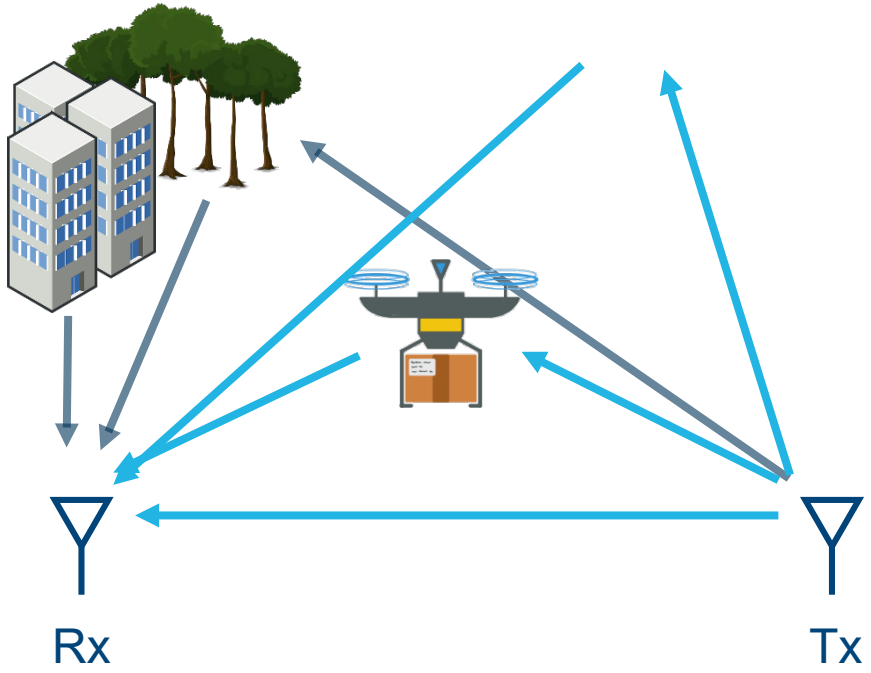


$$\sum_i a_i e^{-j2\pi f \tau_i}$$

Band-unlimited Impulse response

MOTIVATION

Multidimensional Parameter Estimation



- Moving objects cause Doppler shifts in the received signal
  - Each target has a characteristic delay and Doppler
  - Estimation results now has two dimensions (delay, Doppler)
- Why to consider multiple dimensions (e.g. Doppler)?
  - Doppler helps separating static clutter and moving targets
  - Easy filtering and removal of static clutter

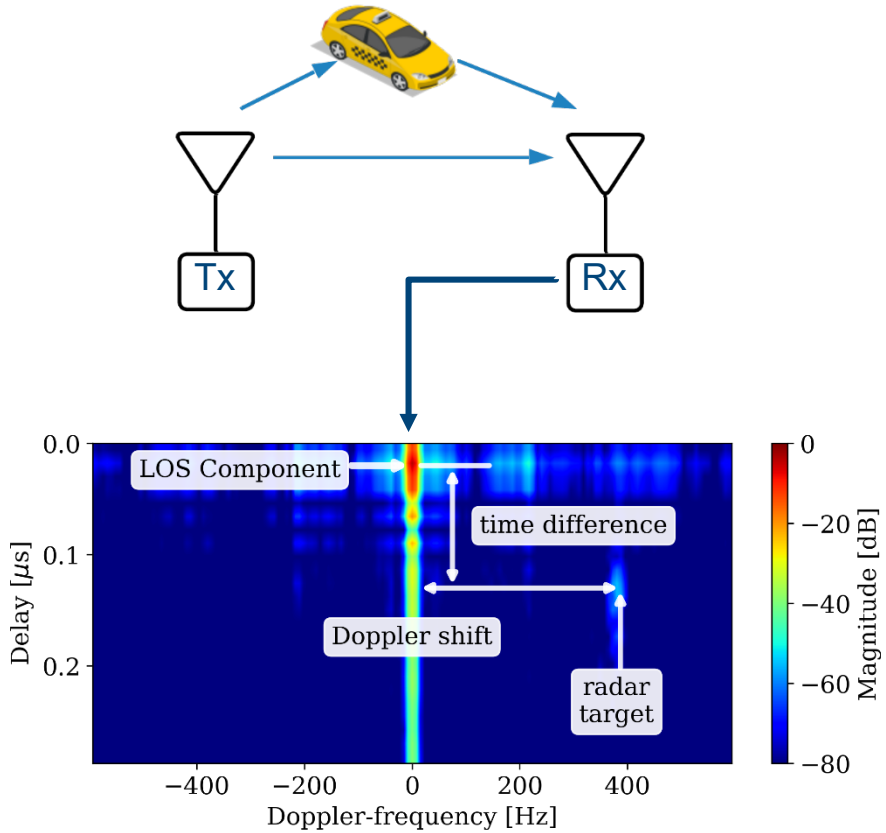
$$y = \sum_i a_i e^{-j2\pi f \tau_i}$$

More dimensions possible!
e.g. spatial, polarimetric

PRACTICLE EXAMPLE

From measurements

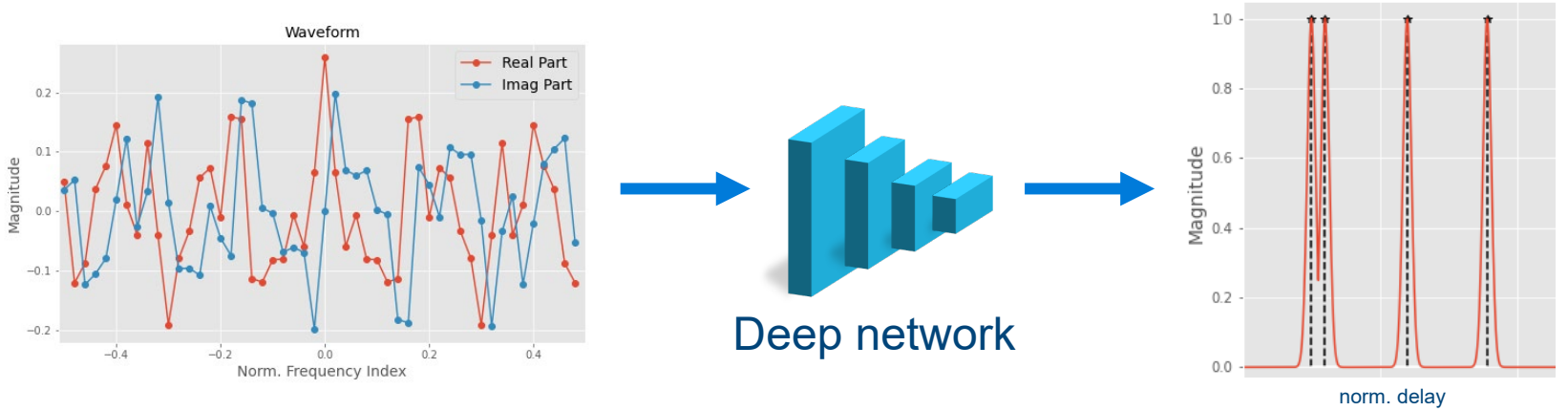
- Use standard OFDM receiver processing to estimate the wireless channel
  - Target can be detected
  - Target can be separated from static clutter
- Also works for UAVs (~~hopefully~~, very likely!)



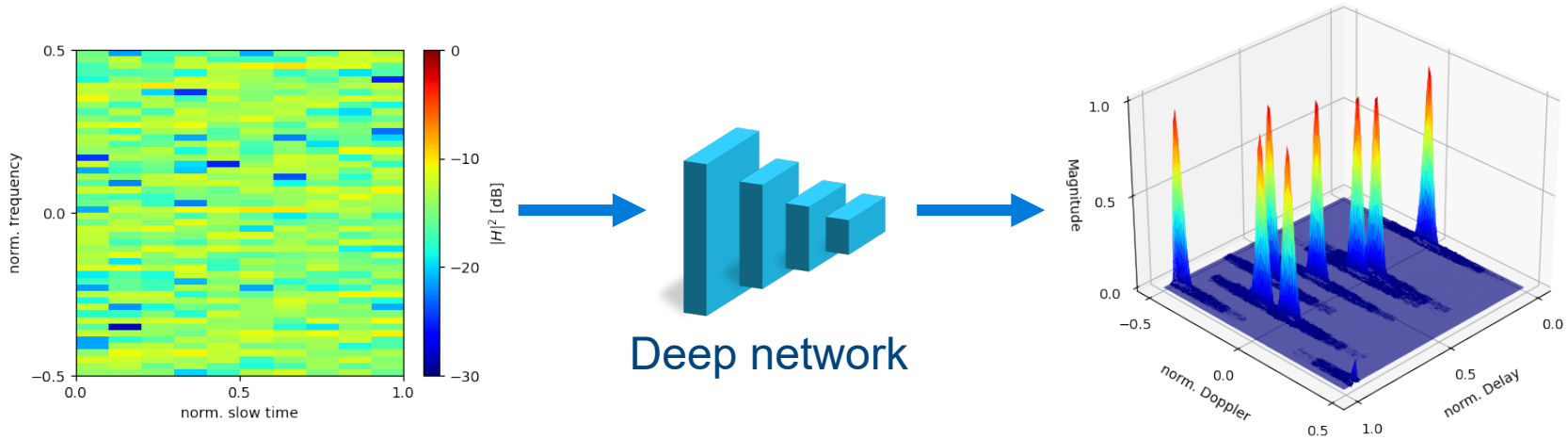
EVOLUTION

Moving from 1-D to 2-D to N-D

1D Task



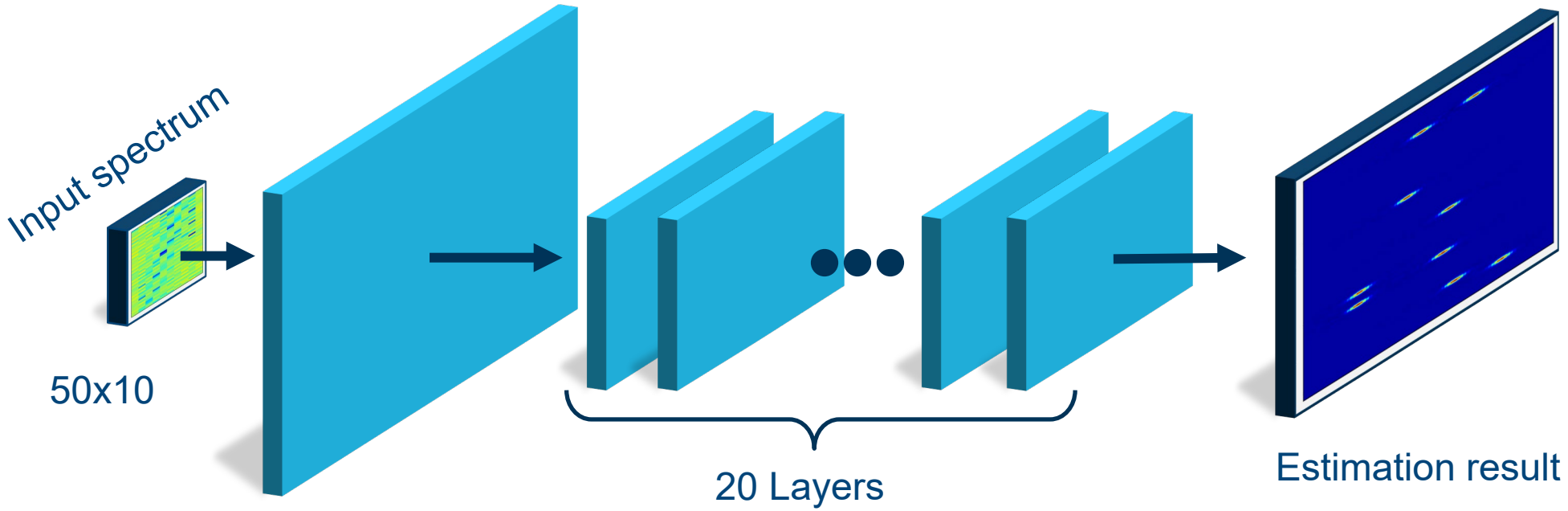
2-D example



NETWORK ARCHITECTURE

Information

Summary: ~ 82,000,000 (trainable) parameters



TRAINING

Information

- The network is trained on 10,000,000 different synthetic examples

- The network is trained for 100 epochs, during each epoch
  - Each sample is passed through the network once
  - Synthetic time-variant channel transfer functions (input) are calculated on the fly
  - Different noise realizations are added

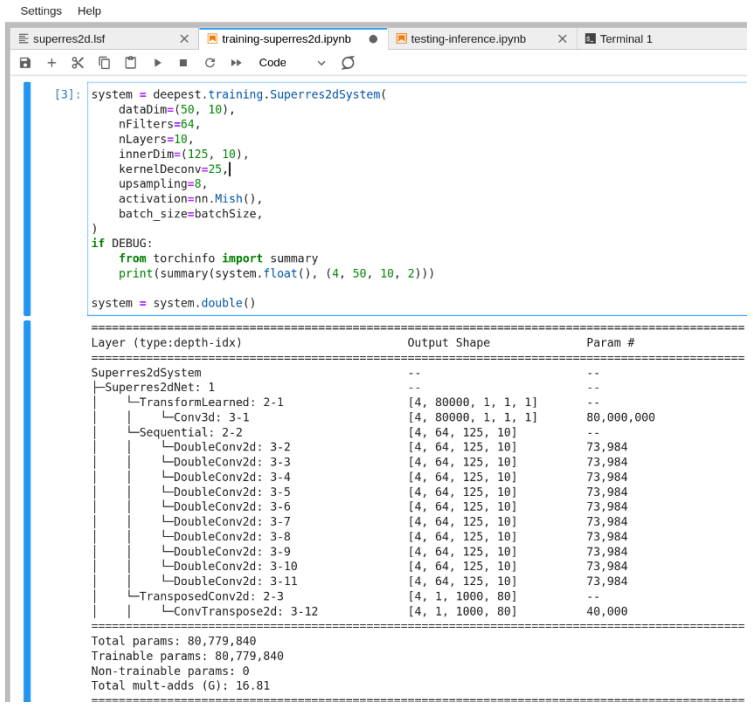
- Training takes ~ 1 days on 8x Nvidia A100 GPUs (training time ~ # GPUs)
- Model is saved and used for testing/evaluation

TRAINING

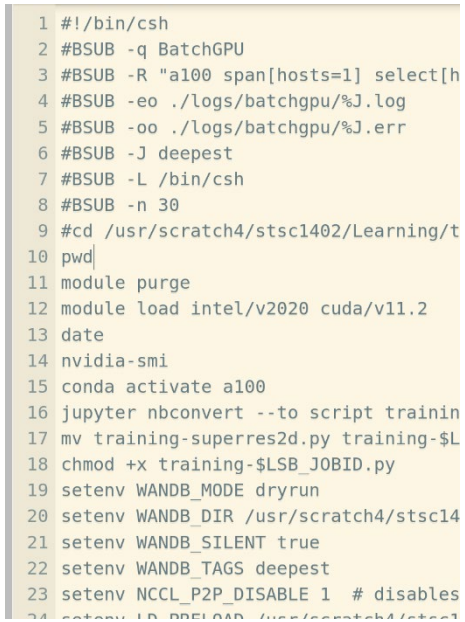
Process on our GPU infrastructure

Code Development

(using Jupyter Notebooks)

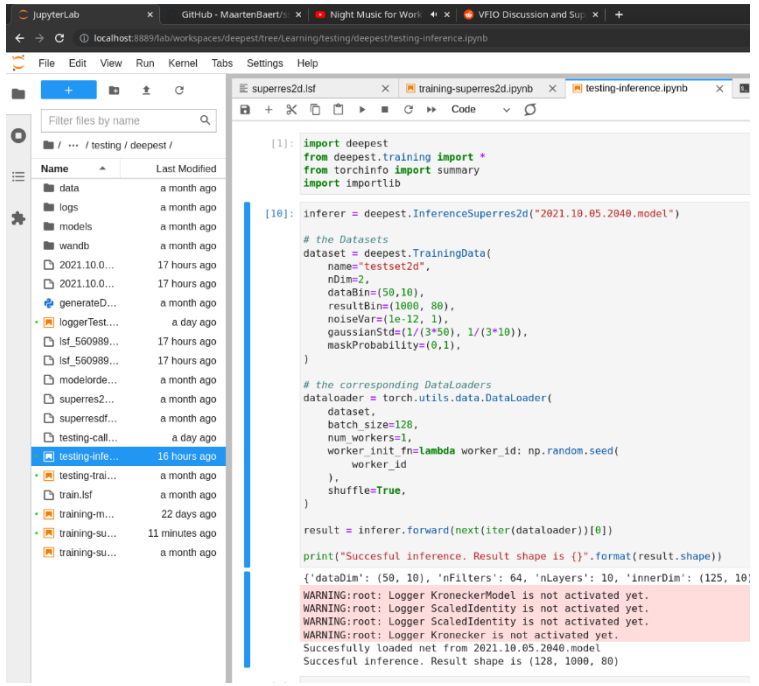


GPU Batchjob



Model Evaluation

(using Jupyter Notebooks)



TRAINING

Statistics on HPC Utilization

Live Demo (lets hope this works)

www.wandb.ai

TRAINING

Progress Visualization