G2NET WG.3

INFN Genova - activity overview

Andrea Chincarini
Alessio Cirone
Stefania Farinon

GE group overview - 2019/2020

A.Chincarini
Senior researcher

VIRGO-GE resp.
AIM-GE resp
CSN5 coord.

G.Gemme
Senior researcher

ET-GE resp.
VIRGO-VEB resp
CSN2 coord.

AIM medical applications

F.Sensi
senior post-doc

Medic

E.Peira
PhD student

Medic

N.Alchera
PhD student

Medic

G.Pedemonte
student

VIRGO noise mitigation

G2NET

S.Farinon
Senior Researcher

VIRGO mechanical modeling

A.Cirone
PhD

VIRGO Noise mitigation

M.Pulze
student

VIRGO controls computing det.char.

D.Bersanetti
Technologist

VIRGO locking optical alignment

L.Rei
Technologist

VIRGO Computing Det. charact.

VIRGO mirror coatings

M.Canepa
Full professor

VIRGO optical coating

M.Magnozzi
post-doc

VIRGO optical coating

VIRGO quantum optics

F.Sorrentino
Researcher

Quantum Optics
VIRGO light squeezing resp.

B.D'Angelo
PhD student

VIRGO light squeezing

B.Garaventa
PhD student

VIRGO light squeezing

G.Gabriella
student

warning: we are not ML experts...

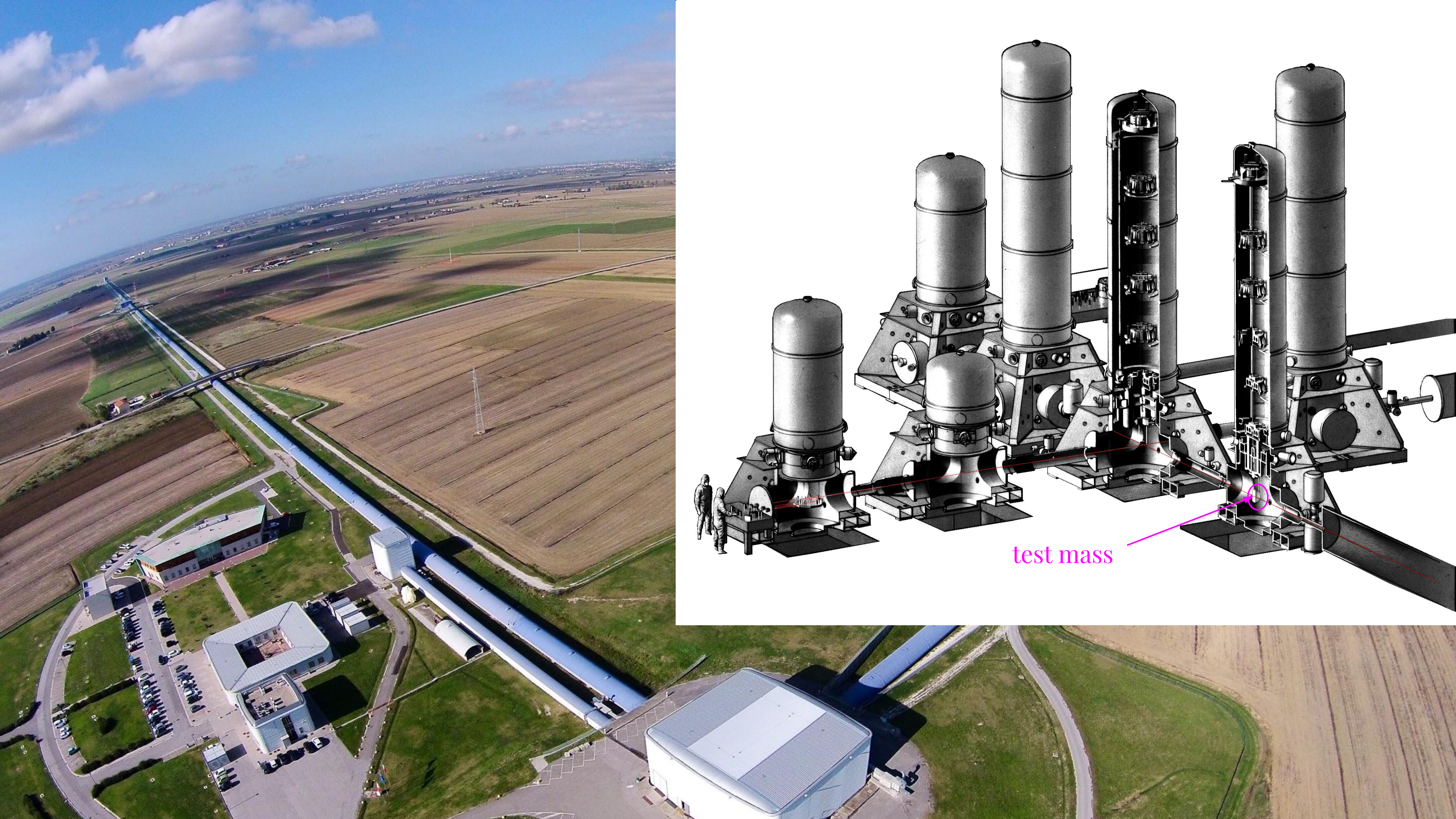
WG.3 activity

- Application of ML to Newtonian Noise reduction for Advanced VIRGO

possible future WG.3

[students wanted!]

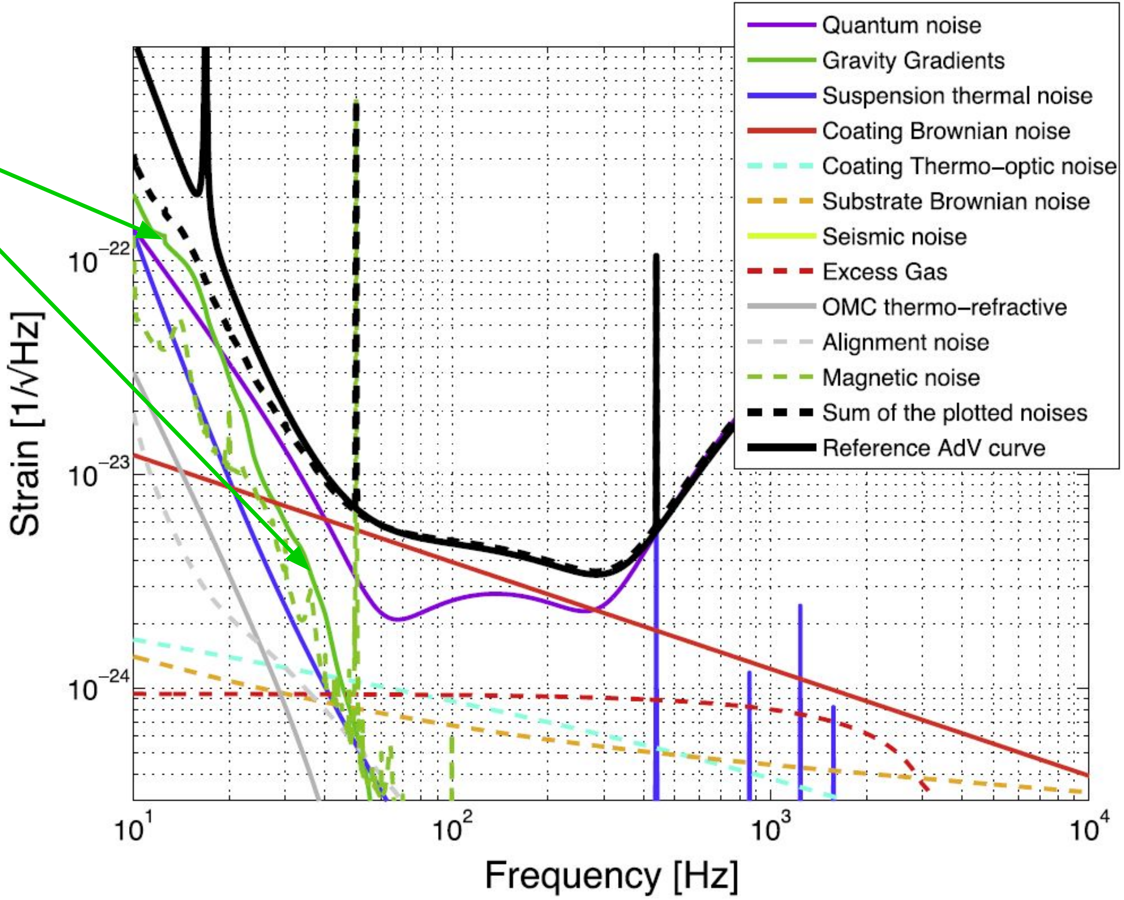
- Detector characterization (noise hunting)
- Controls (interferometer locking procedure)


test mass

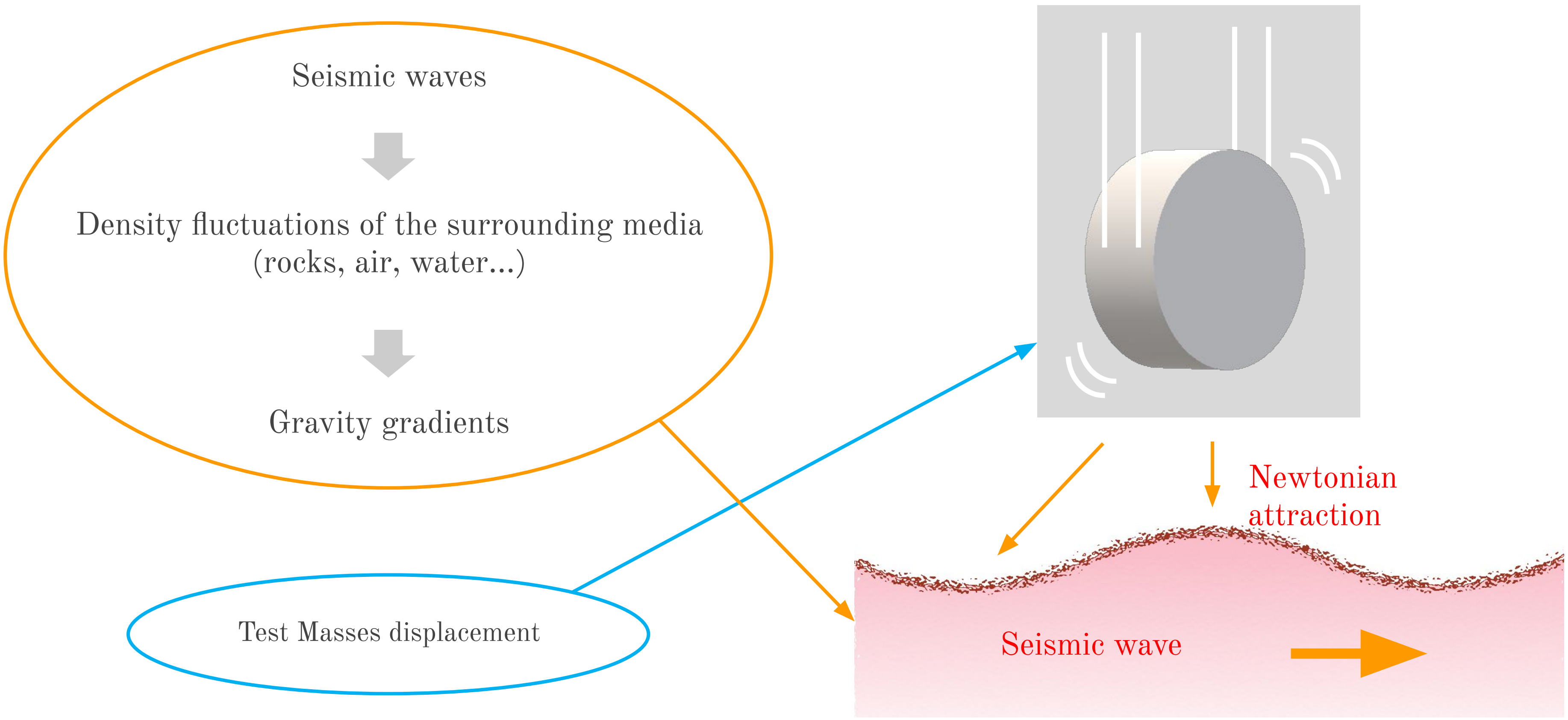
limiting noises in adv. interferometers

Newtonian Noise of seismic origin

It would be a limiting noise source for AdV+ in the 10-40 Hz band

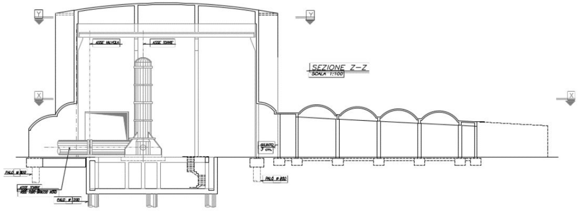


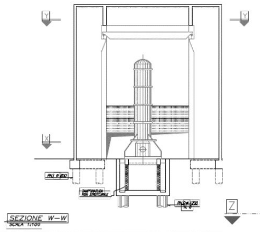
seismic newtonian noise

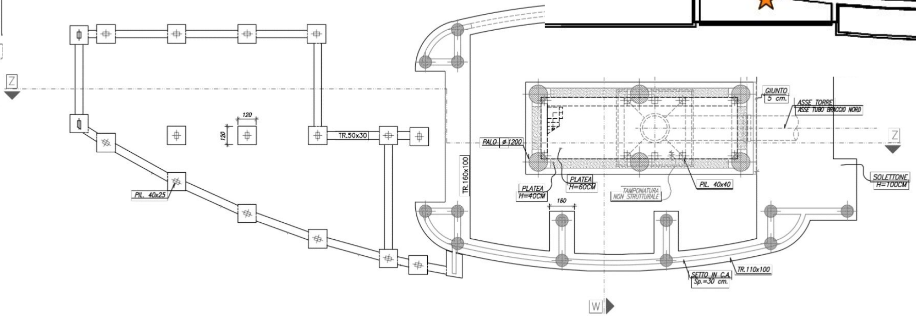


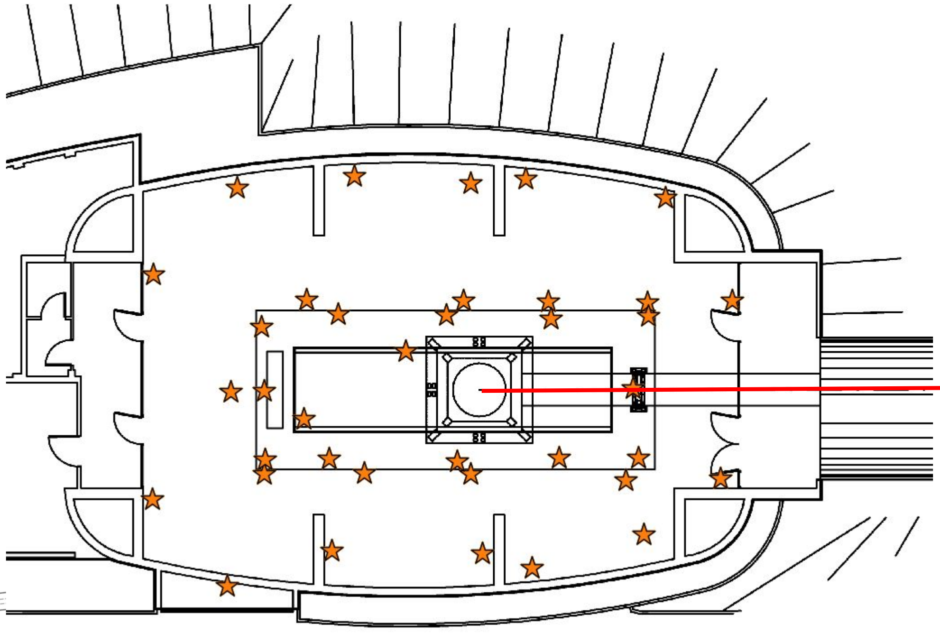
NN cancellation

A Newtonian Noise cancellation scheme can be realized using an array of sensors deployed near the test masses







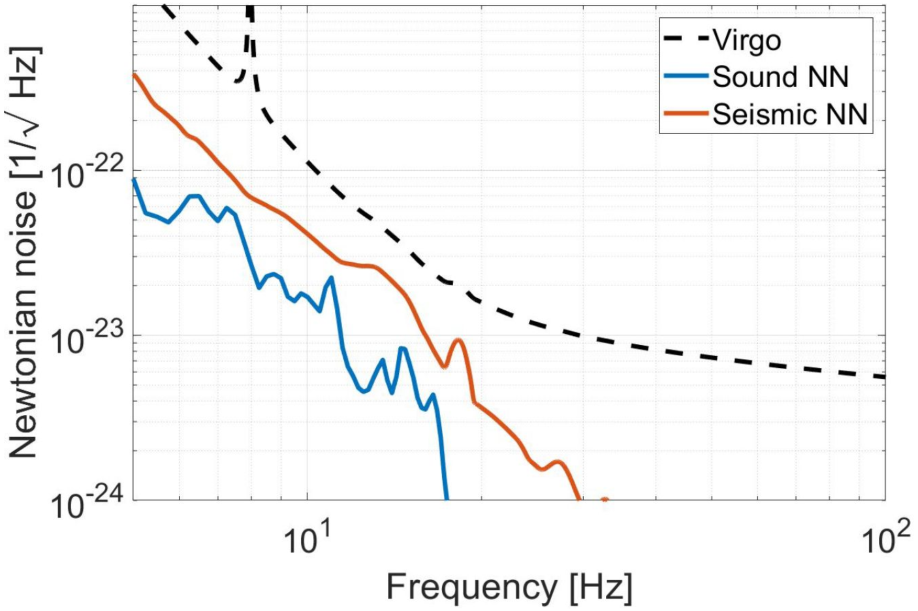


problem unknowns

1. Newtonian noise has never been directly measured
    - Theoretical models
    - semi-realistic FE simulations (including soil properties and the surrounding infrastructure)

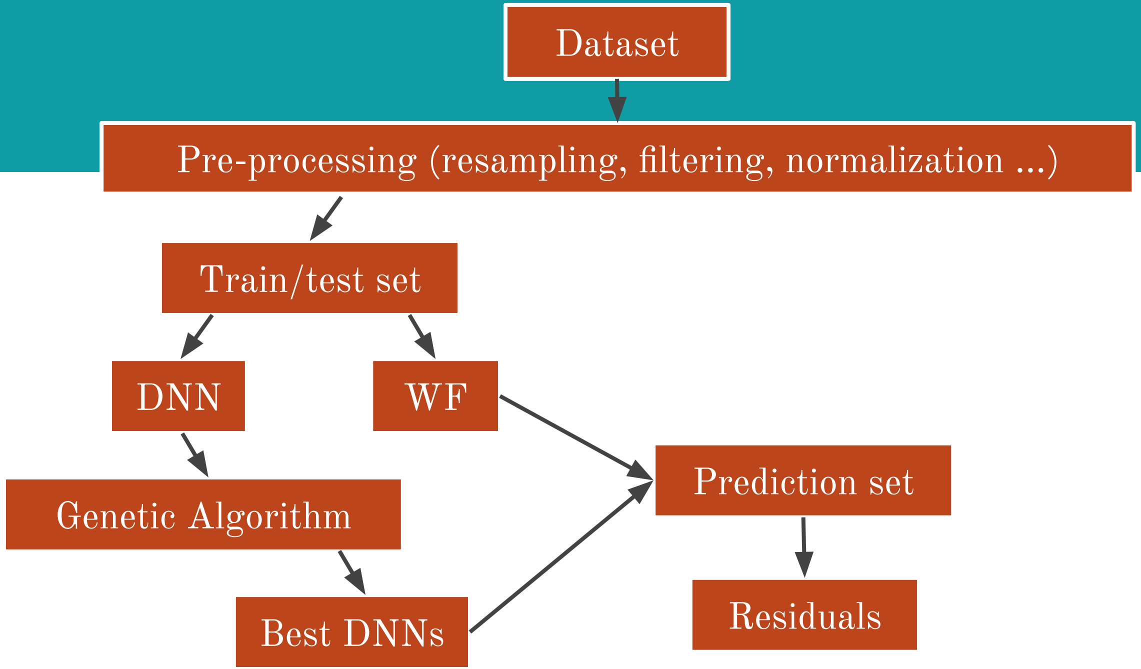
2. Sensors
    - Optimized placement (number and position)
    - Type selection (accelerometers and/or tiltmeters)
    - Underground detectors?

3. Data processing
    - A second feasible subtraction algorithm in addition to Wiener Filter (the gold standard)

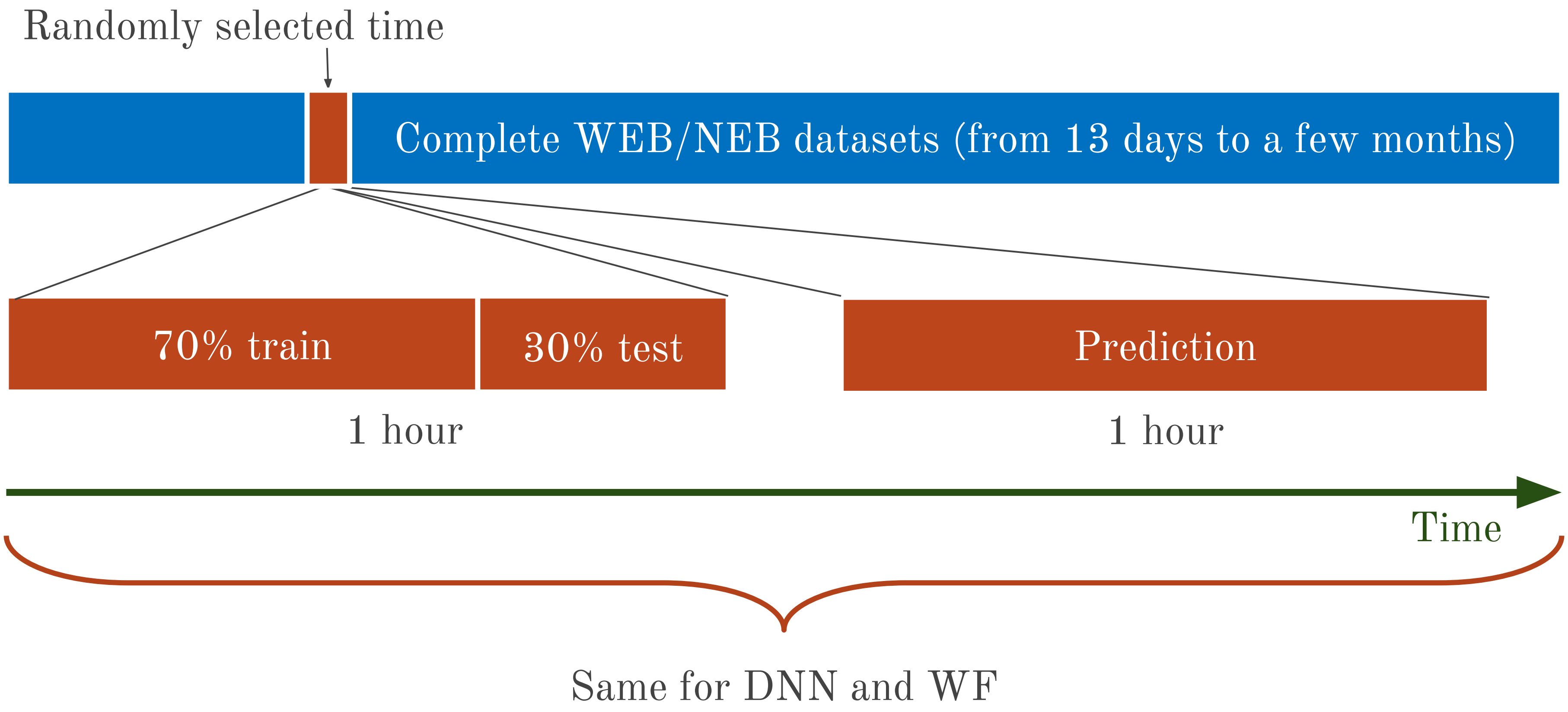


prediction of one sensor displacement from the entire array

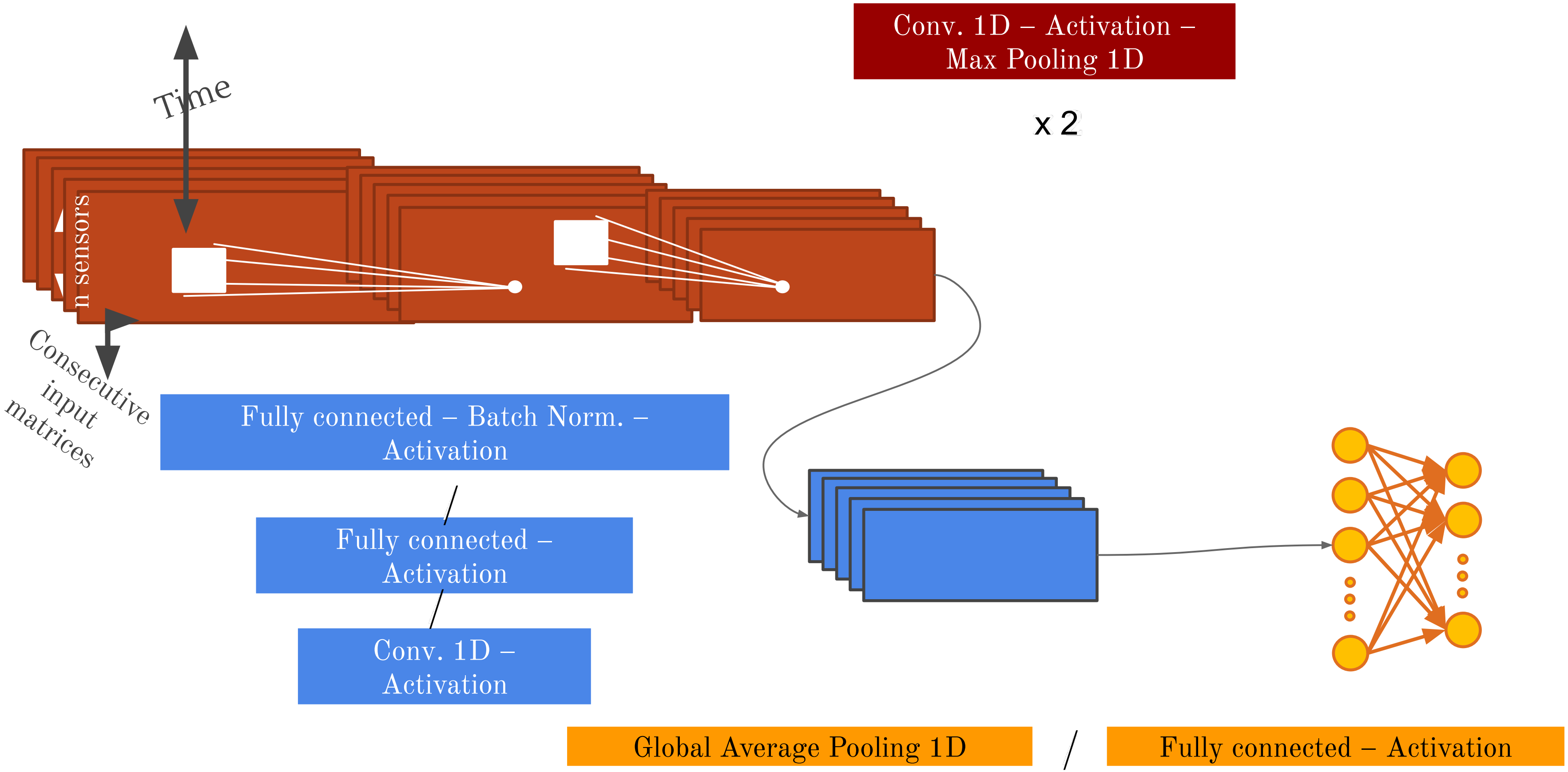
neural network
compared to
Wiener filter



dataset


Randomly selected time
Complete WEB/NEB datasets (from 13 days to a few months)
70% train
30% test
Prediction
1 hour
1 hour
Time
Same for DNN and WF

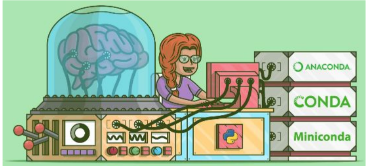
CNN base architecture



computing infrastructure

Training & GA hyperparameters optimized offline on simulated data (parallel processes on CPUs)

Test cluster of 180 cores for distributed processing



Virtual environment with Anaconda python distribution

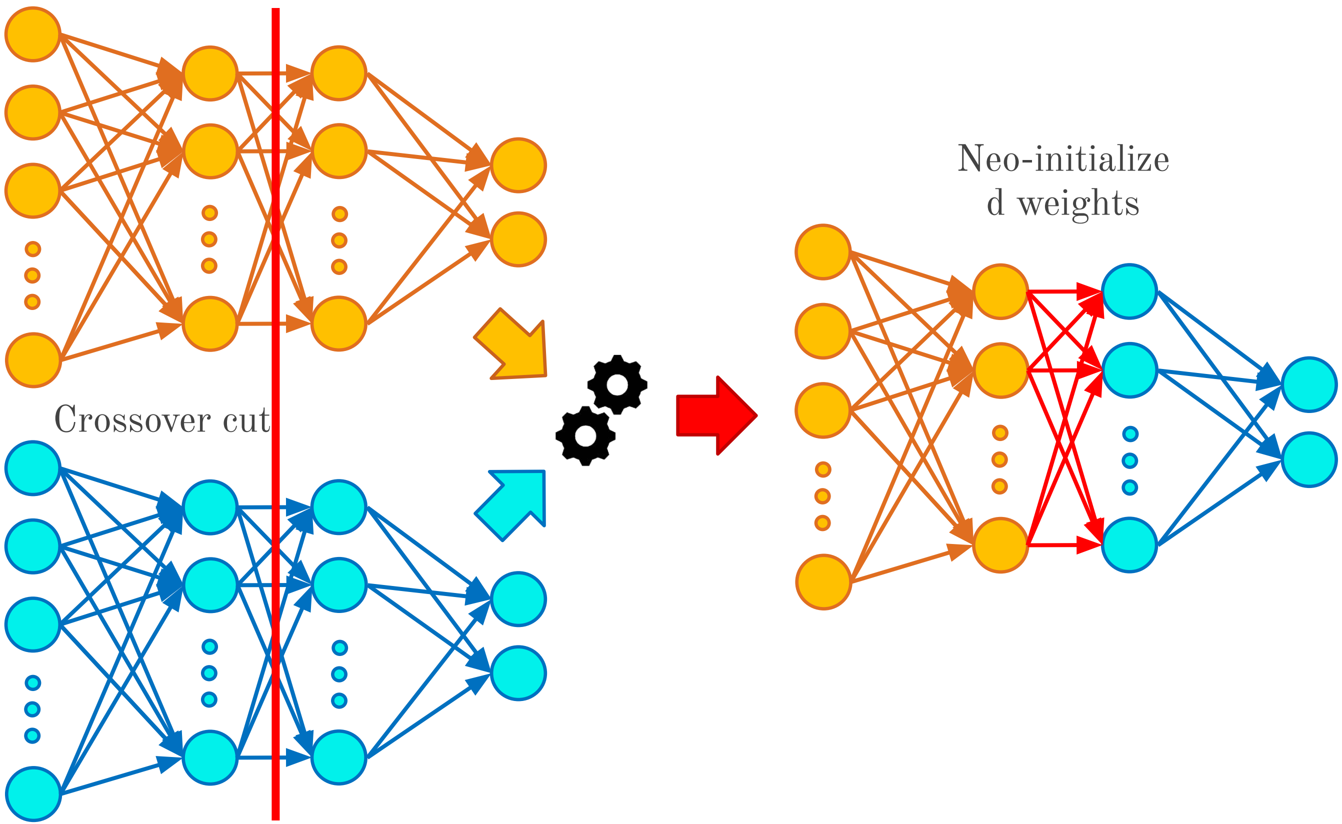
Robust portable environment for Virgo to run on-site and act offline for Newtonian noise subtraction

Good foundations:
python based
open access
CPU & GPU friendly

DISTRIBUTED EVOLUTIONARY ALGORITHMS IN PYTHON

transfer learning to lower the computation time

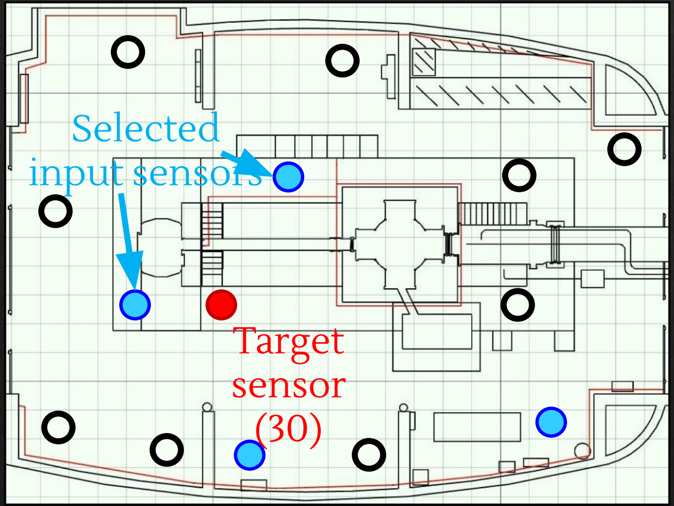
- In GA new networks are created by “mating” two high-performance networks
- The new networks inherit some properties from both parents, in particular their weights, therefore applying an heuristic transfer learning



Evaluate the performance

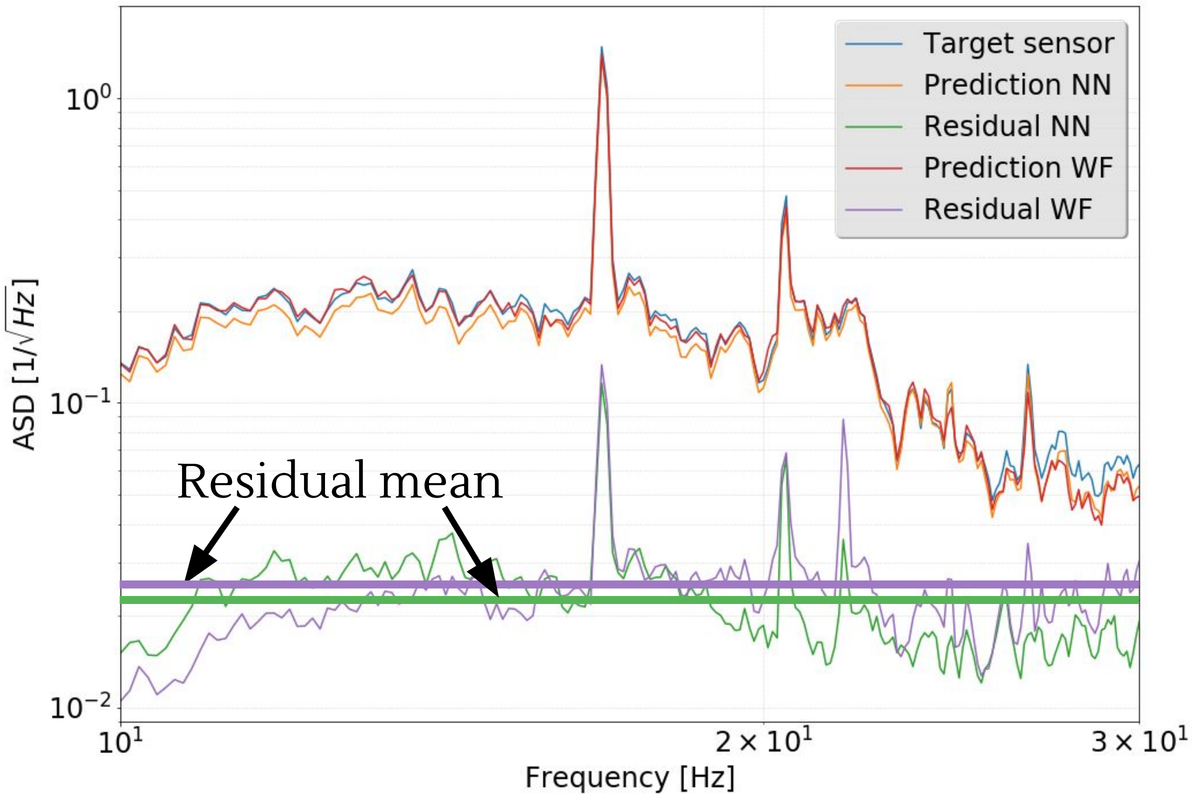
- The DNN takes the sensor array temporal data as input and another single sensor as output

Time	Sensor 1	Sensor 2	…	Sensor k	Target sensor
0.01	1,24E-11	1,26E-13	…	2,72E-14	2,30E-12
0.02	3,16E-10	5,31E-13	…	9,48E-13	5,11E-11
0.03	3,66E-09	4,03E-13	…	1,57E-11	4,68E-12

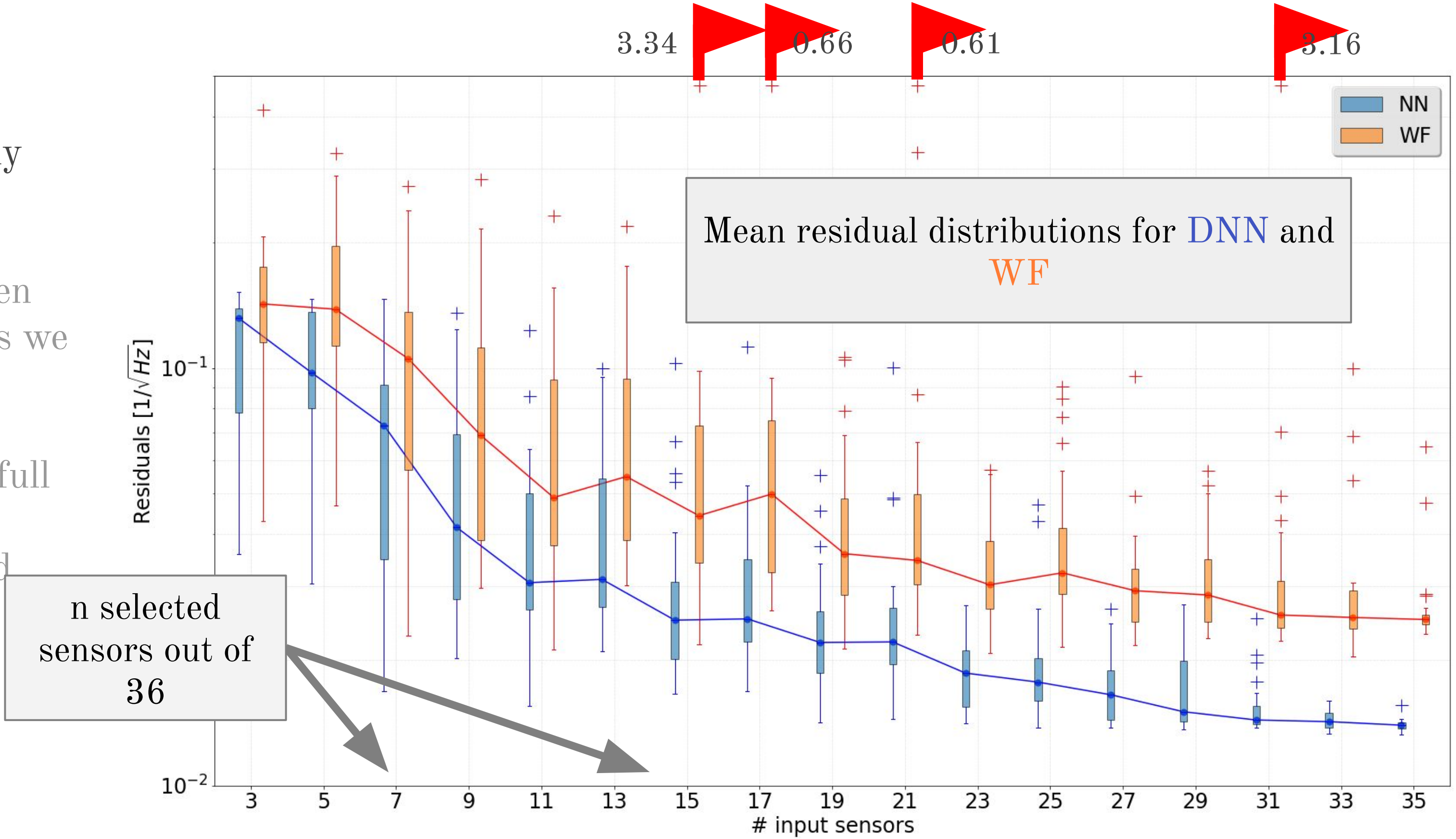


- Montecarlo simulation with n input sensors out of N=36
- Random data selection for each MC choice
- A single predictor for DNN is selected: the mean performance value is taken from the best DNNs

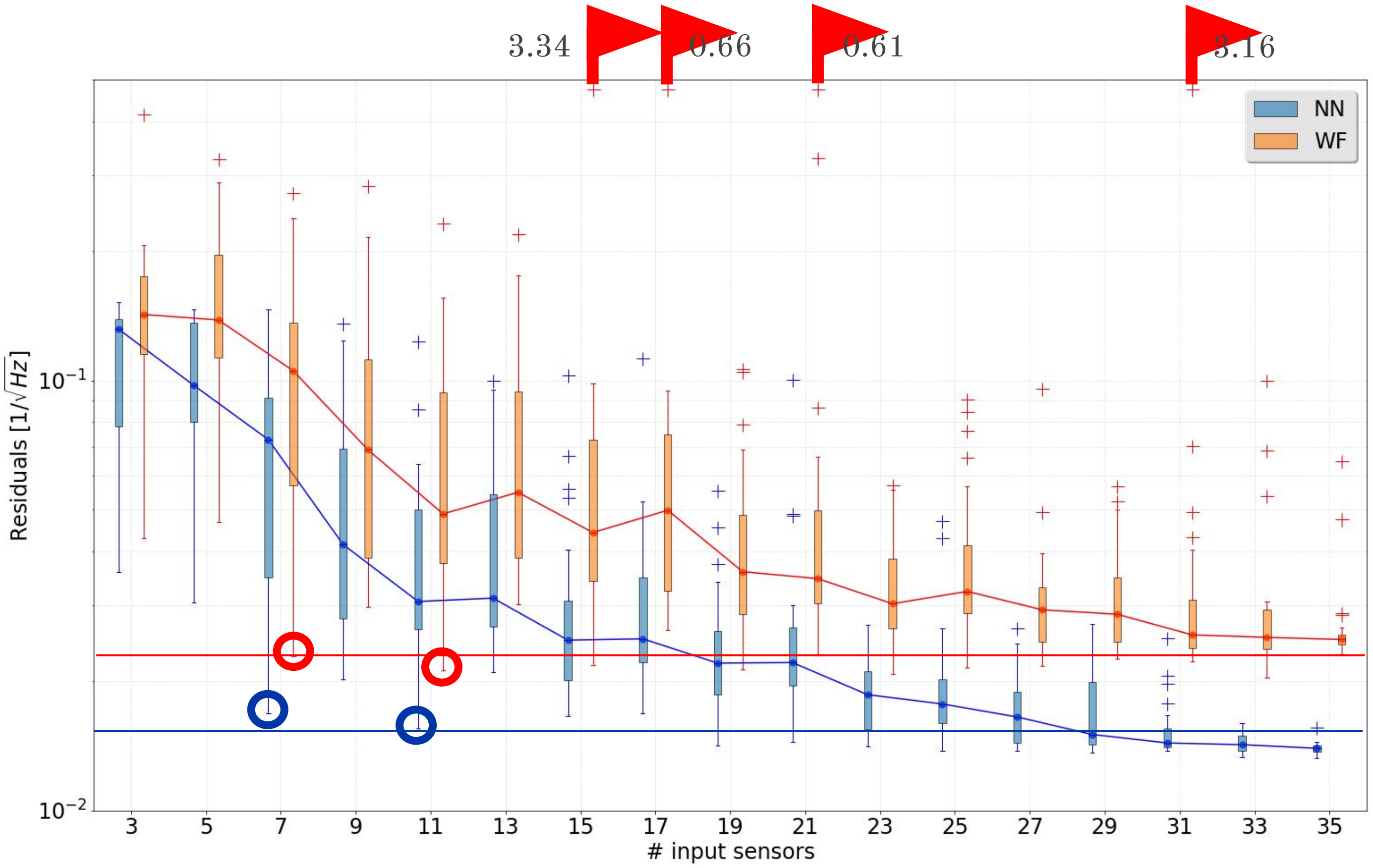
Example of seismic spectra of the target channel and the predicted output with DNN and WF, together with the residuals (entire array as input)

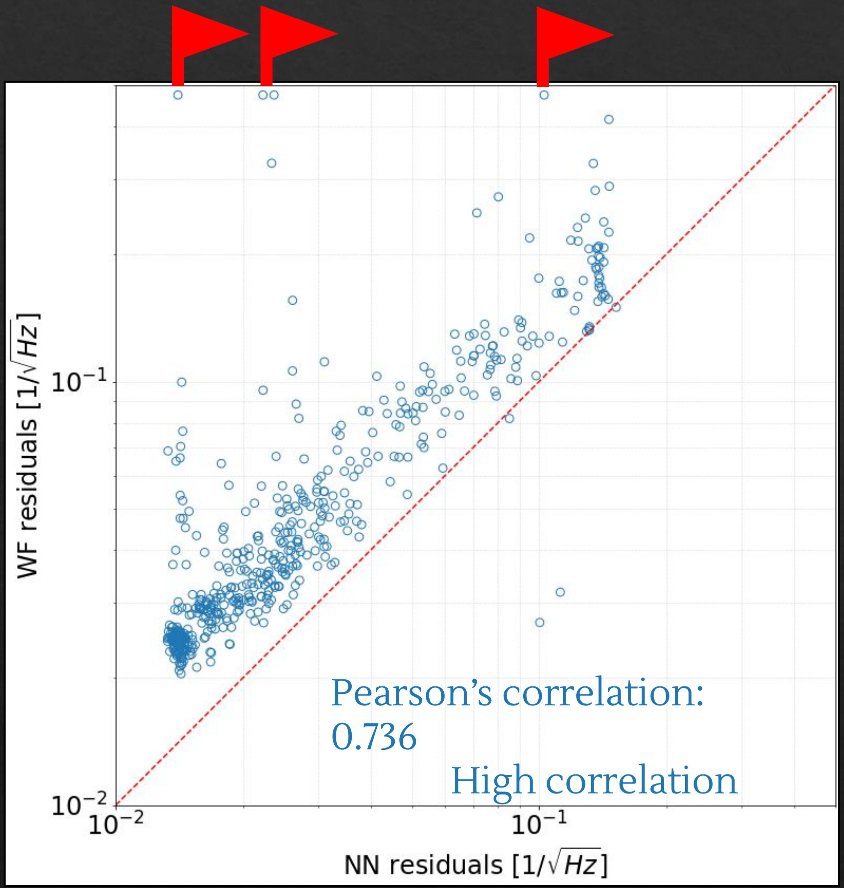


- DNN statistically better than WF
- Redundancy: even with few sensors we can achieve comparable residuals to the full array case, if properly selected
- The higher sensor-target correlation, the better residuals



- DNN statistically better than WF
- Redundancy: even with few sensors we can achieve comparable residuals to the full array case, if properly selected
- The higher sensor-target correlation, the better residuals





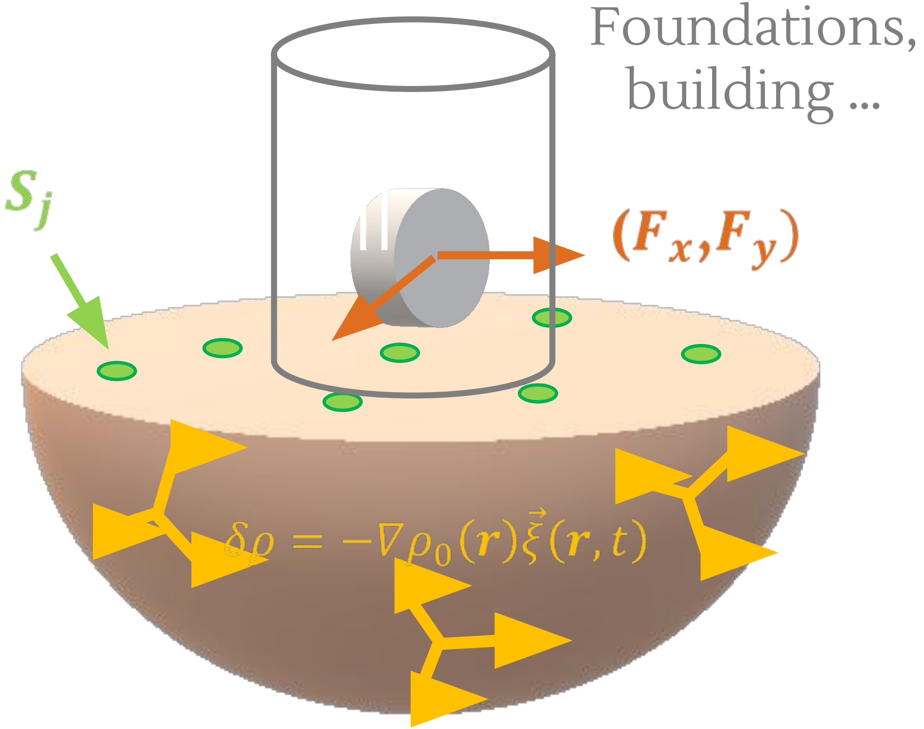
- High correlation between DNN and WF results
- Also some cases in which WF goes so much worse than DNN
- This behaviour is under investigation

$\frac{1}{2}(DNN + WF)$ can be competitive / alternative / complementary to WF alone, even in a sub-optimal sensor configuration

initial approach: time-domain

the idea:

1. model several possible solutions in time-domain with random sources and terrain properties
2. gather covariance/correlation matrix between surface z-displacement and Newt. force
   a. complete dataset consists of N covariance/correlation maps
3. if dataset is big enough and varied, train *neural network* to infer complete covariance map starting from a subset of its points
   a. i.e. use similar technique as in partial image reconstruction
4. apply covariance map reconstruction starting from real data between sensors and mirror displacement
   a. initially do with a proxy, i.e. n sensors vs. one



initial approach: time-domain

1. model building
   a. time-domain solutions require fine time intervals otherwise errors diverge
   b. boundary conditions are paramount. needs extra mesh around Obj of Interest to avoid reflections
   c. time and space constraints on simulation push to higher complexity
2. sources
   a. time-domain solutions require well defined sources
   b. random scattered sources (in space, phase and frequency) model
   c. a number of simulations (>~100) necessary to explore the phase space
3. post-processing

initial approach: time-domain

1. model building
2. sources
3. post-processing
    a. the "true" combination of model + sources not known (even in principle)
    b. a DL approach was devised to learn NN covariance with z-axis displacement (NNmap). Once DL net is trained (on simulations), true covariance can be inferred from partial sensors info (on the real system)