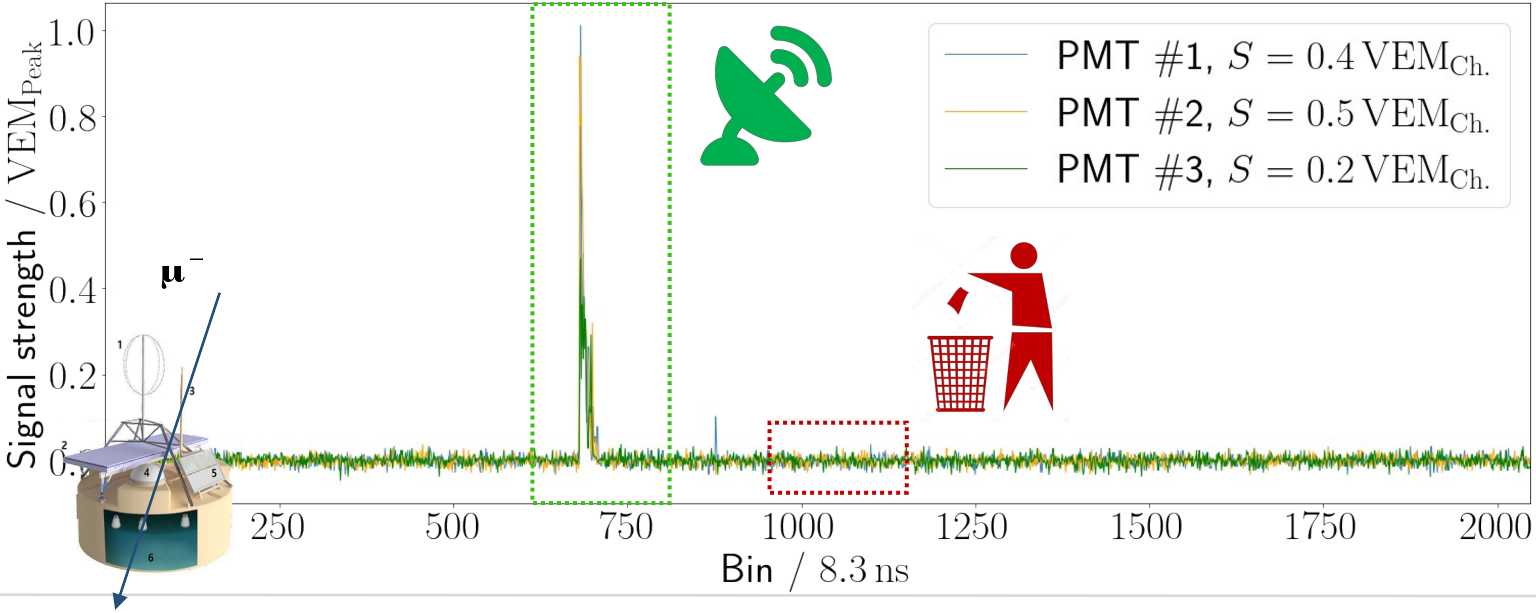
Potential of neural network triggers for the Water-Cherenkov detector array of the Pierre Auger Observatory

Paul Filip – Karlsruhe / Buenos Aires Meeting 14.07.23

Strategy

- Feed labelled subset of trace to neural network architecture
- Teach it to distinguish between Signal / Background



Training / Validation data

- **QGSJET-II.04 protons with** $16.0 \leq \log_{10}\left(E \text{ / eV}\right) < 19.5$
- **UUB Simulation to determine WCD detector response**

- **Background electronic noise measured in field (= Random traces)**
- **Baseline subtraction + Calibration to (true)** $\mathrm{VEM}_{\mathrm{Peak}}$

- **Add both components together to make realistic WCD trace**
- **See previous Ka/BsAs contributions for details [1 / 2]**

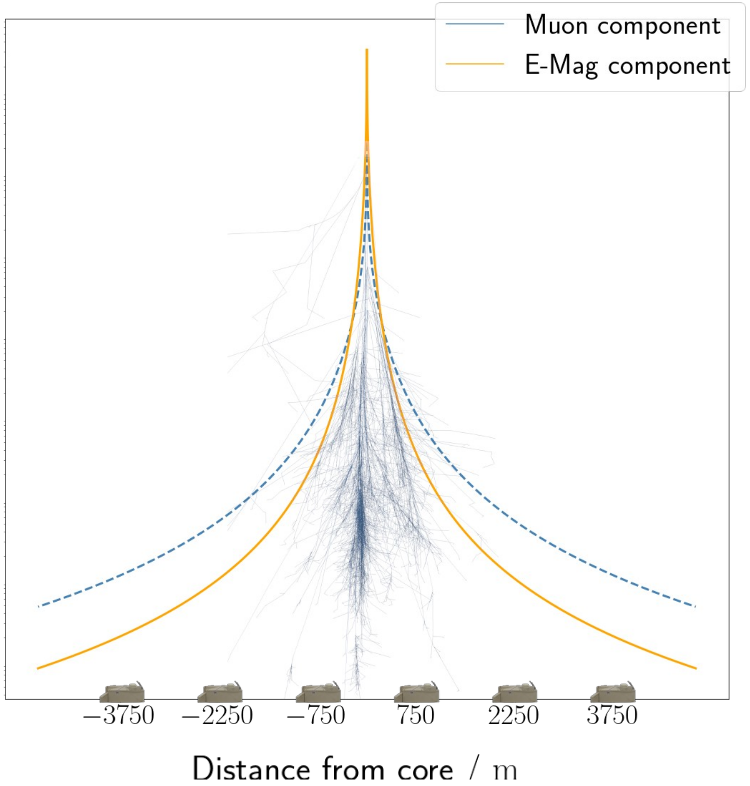
- **Full dataset (hopefully) soon accesible publicly for other uses**

Trigger performance

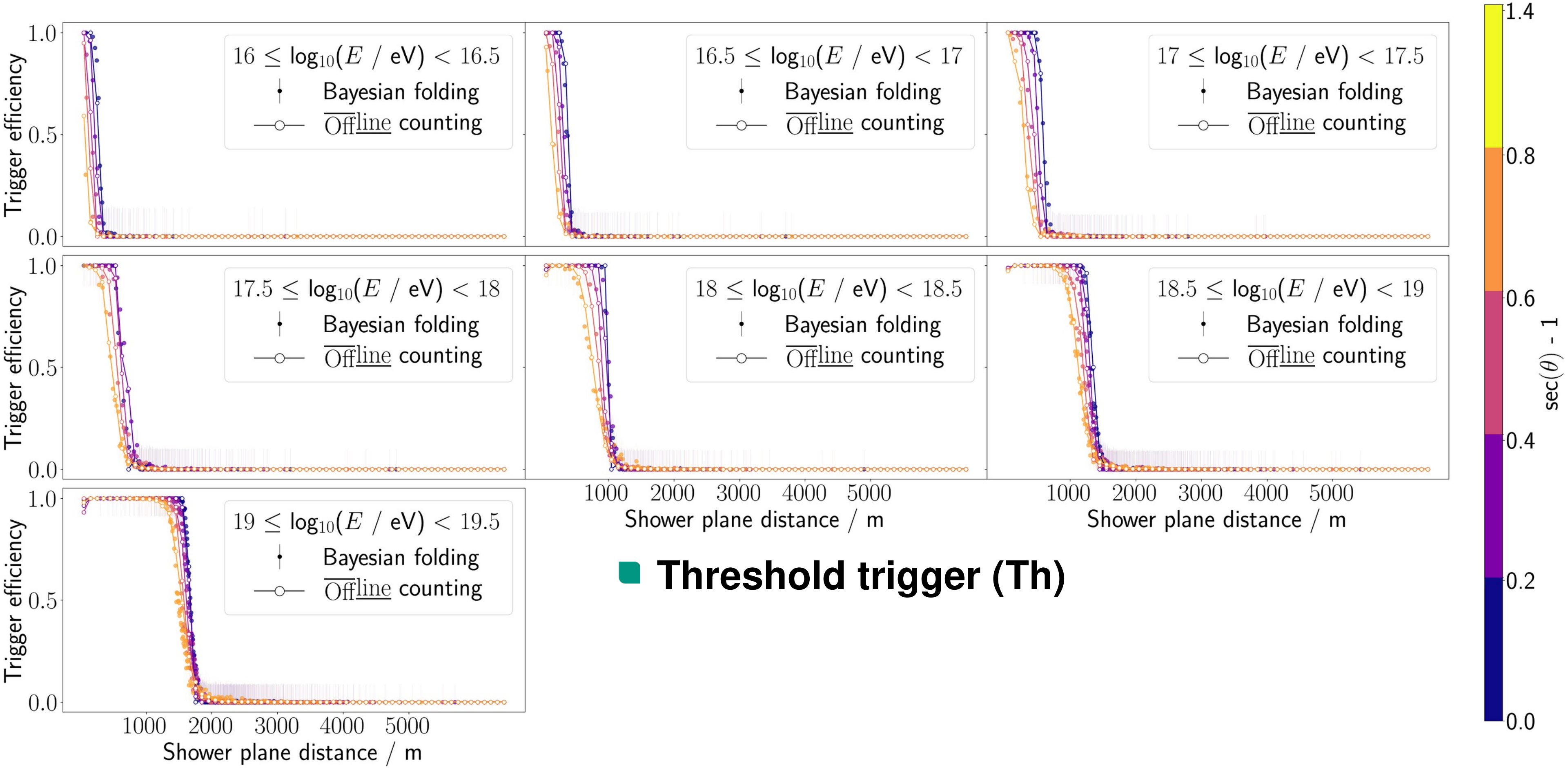
- Classical triggers (Th, ToT, ...) ~100% efficient at $E \approx 3\ \mathrm{EeV}$ in simulations
  - Calculated by running full (Offline) simulation chain
  - Time consuming approach for prototyping new triggers

- Use Bayesian folding to cut down simulation time
  - Assume $P(\mathrm{T2}) = P(\mathrm{T2} \mid \mathrm{Signal}) \times P(\mathrm{Signal})$ on station level
  - Assume $P(\mathrm{T3}) = \prod_{3\,\mathrm{NNs}} P(\mathrm{T2})$ factorizes at event level
  - Test assumptions with classical triggers

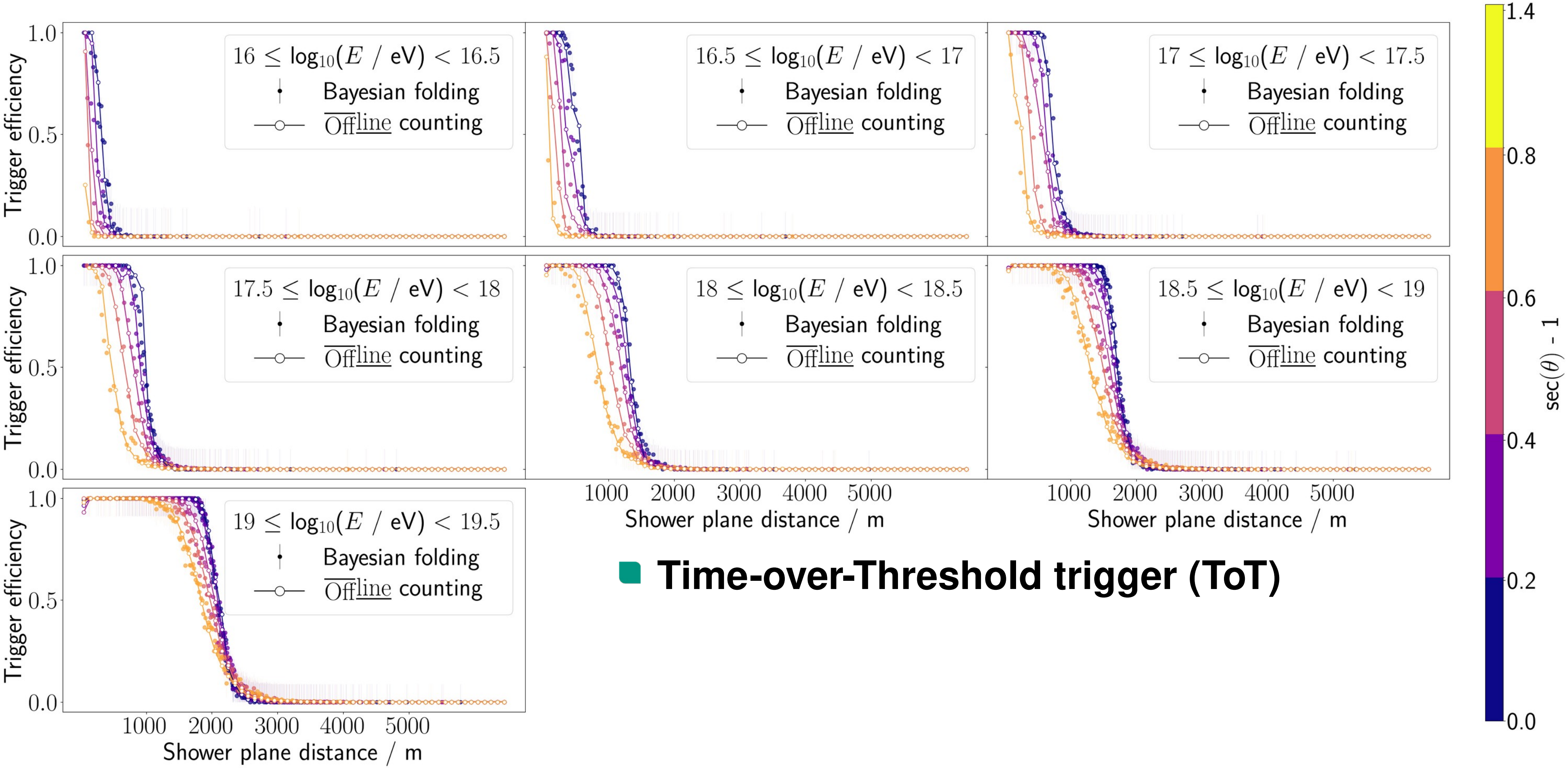
Bayesian folding



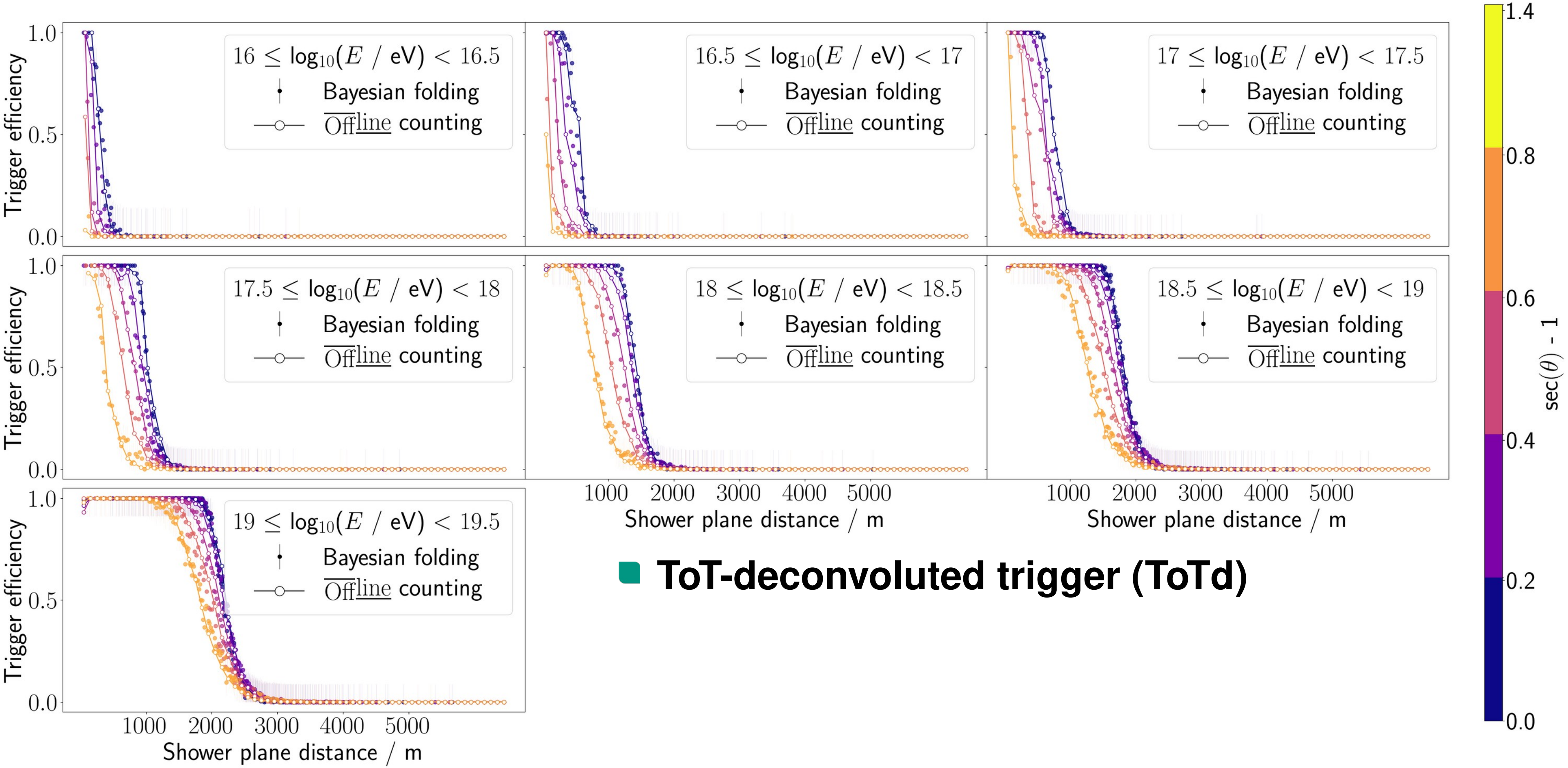
- $P(\mathrm{T2}) = \mathrm{LTP} =$ **lateral trigger probability**
  - **Characterizes chance of trigger w.r.t SPD, Zenith, Energy, etc.**
- $P(\mathrm{Signal}) = \mathrm{LPP} =$ **lateral particle prob.**
  - **Tied to lateral distribution func.**
  - **Ratio of stations that are hit**
- **Independent calculation of** $P(\mathrm{T2} \mid \mathrm{Signal})$
  - **Optimize to specific use-case**
  - **Utilize powerful libraries**



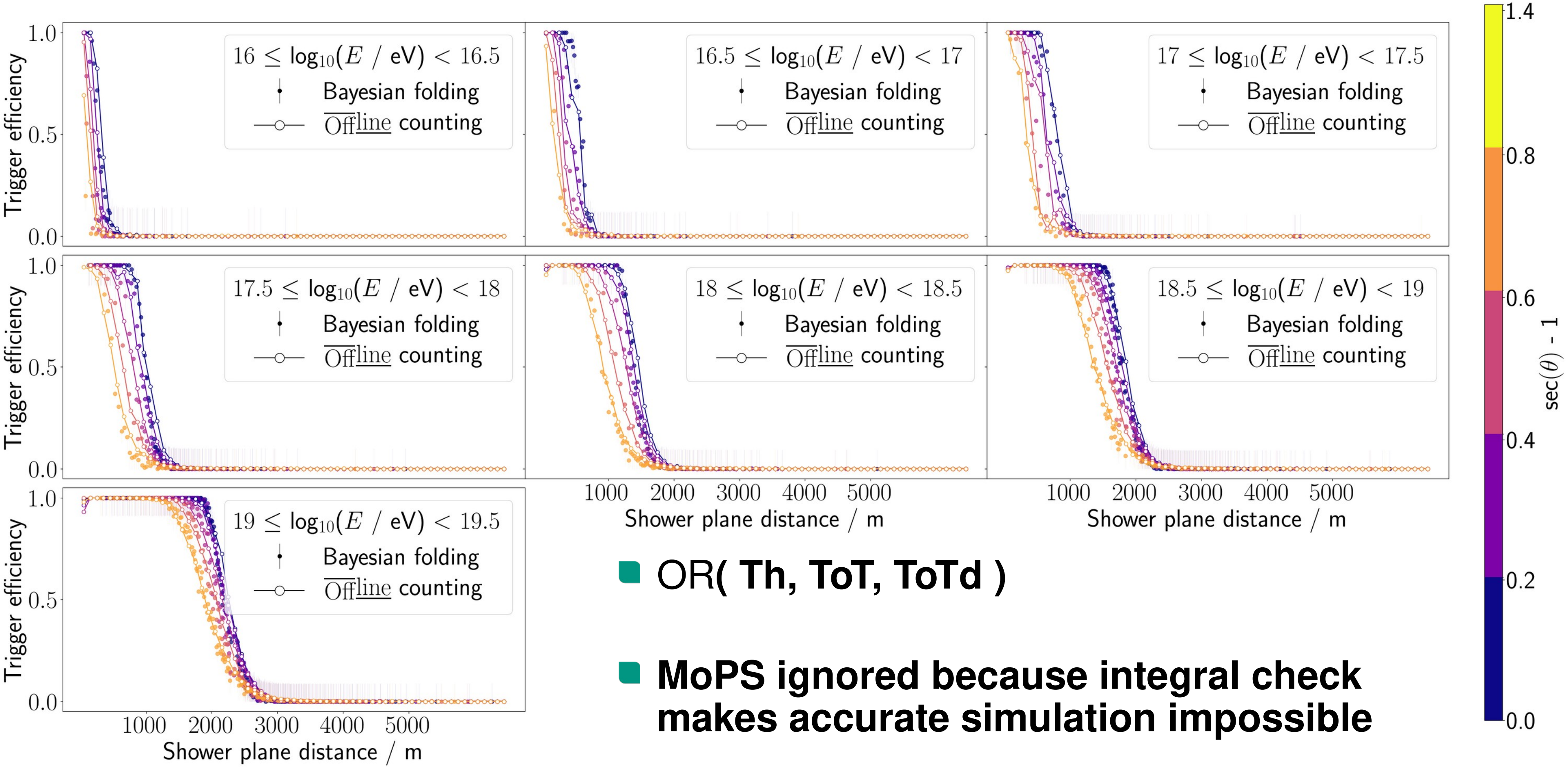
- **Threshold trigger (Th)**



- **Time-over-Threshold trigger (ToT)**



- **ToT-deconvoluted trigger (ToTd)**



- OR(**Th, ToT, ToTd)**
- **MoPS ignored because integral check makes accurate simulation impossible**

Trigger performance

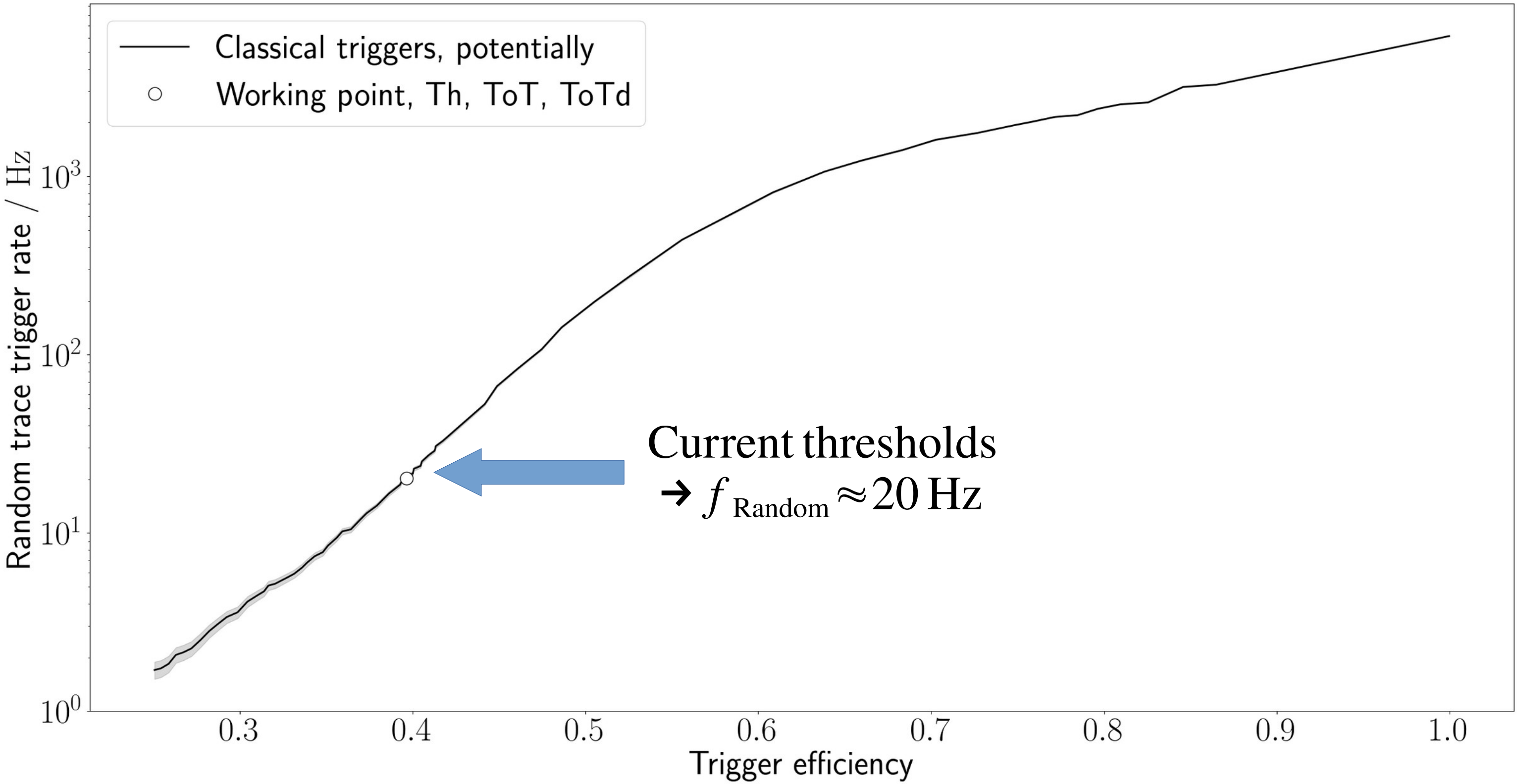
Karlsruhe Institute of Technology

- Base performance on two considerations

- How many showers do we see?
  - conditional trigger efficiency $\epsilon = \frac{1}{W} \sum_{\{\mathrm{TP}\}} \frac{b(E)}{b(E_{\mathrm{ref}})} \frac{E^{-3}}{E_{\mathrm{ref}}^{-3}}$
  - $\epsilon := \frac{n \text{ detected showers}}{n \text{ received showers}}$ for individual station

- How often do we trigger (background events)?
  - Analyse performance on random-trace dataset
  - Trigger rate must not exceed $\approx 20\,\mathrm{Hz}$ (Bandwidth limitations)

Trigger performance – Classical triggers



Trigger performance – Convolutional NNs

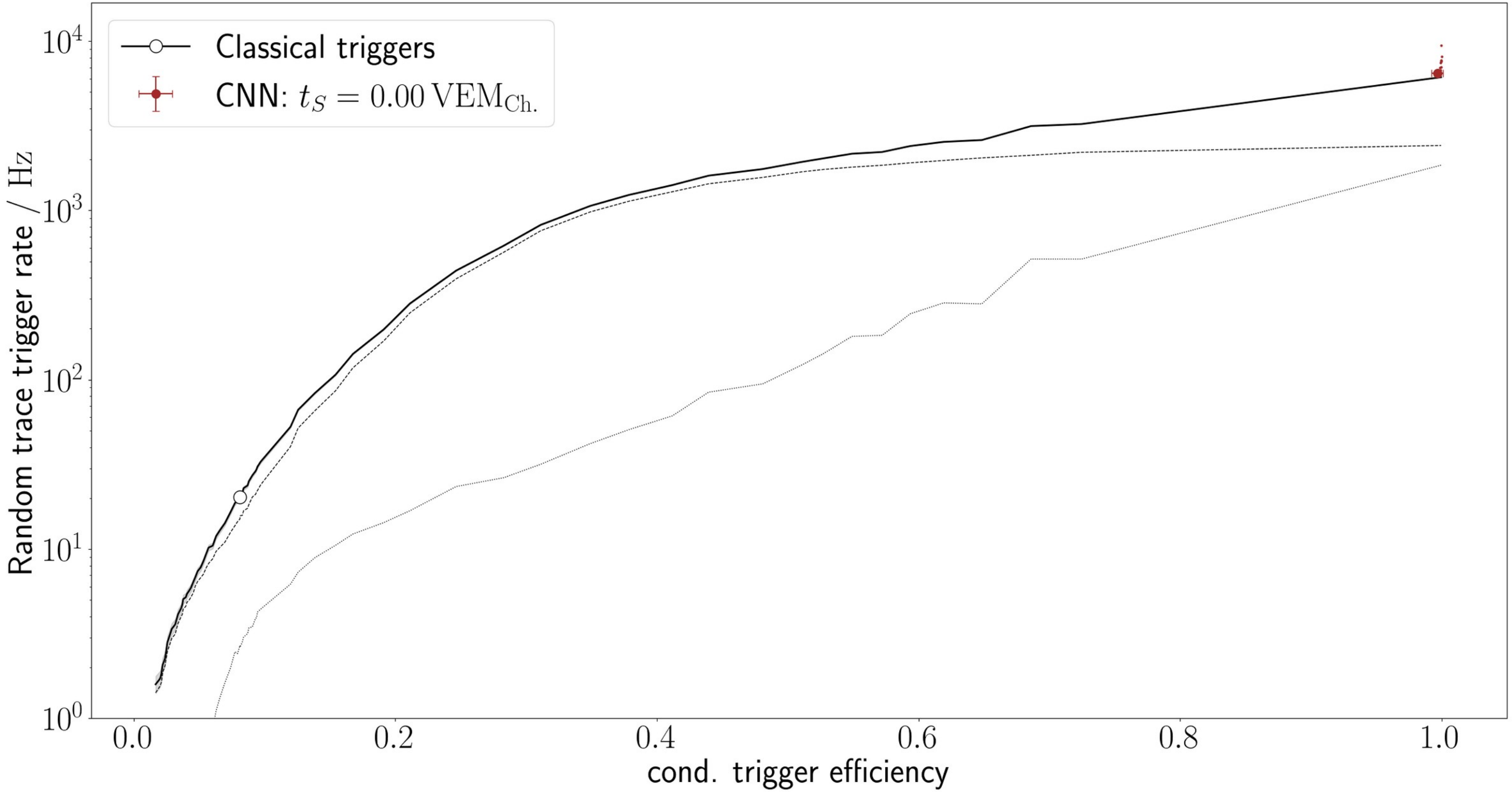
Karlsruhe Institute of Technology

- Opt for 120 bin input size
    - Shown to work for ToT, ToTd, MoPS
    - Compromise network size and input information

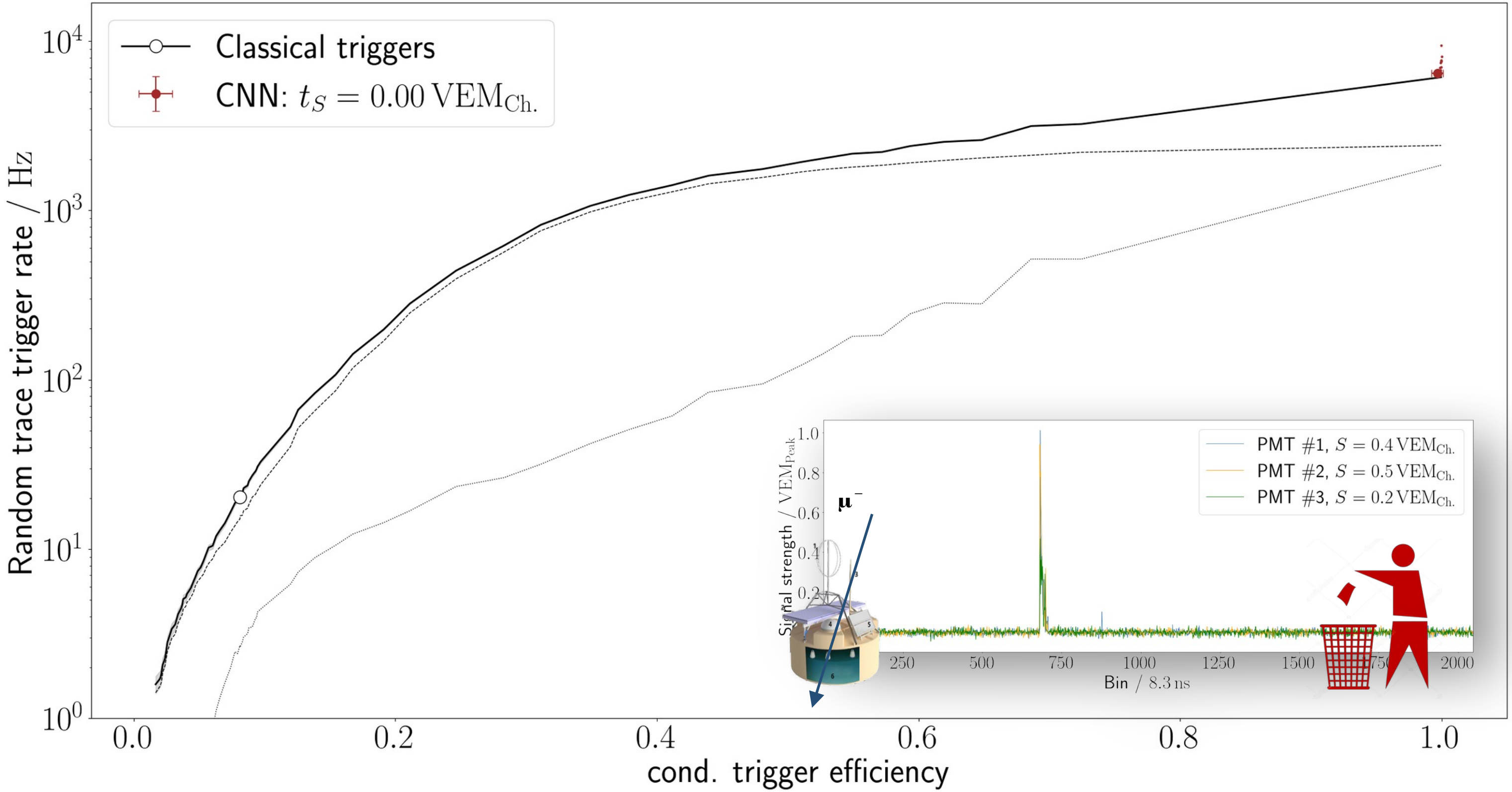
- Motivate network architecture on data structure
    - 2D convolutional layer recognizes correlation between 3 PMTs
    - Dense layer propagates extracted features to binary output

- Evaluate performance after training
    - 32k training events, 8k validation events

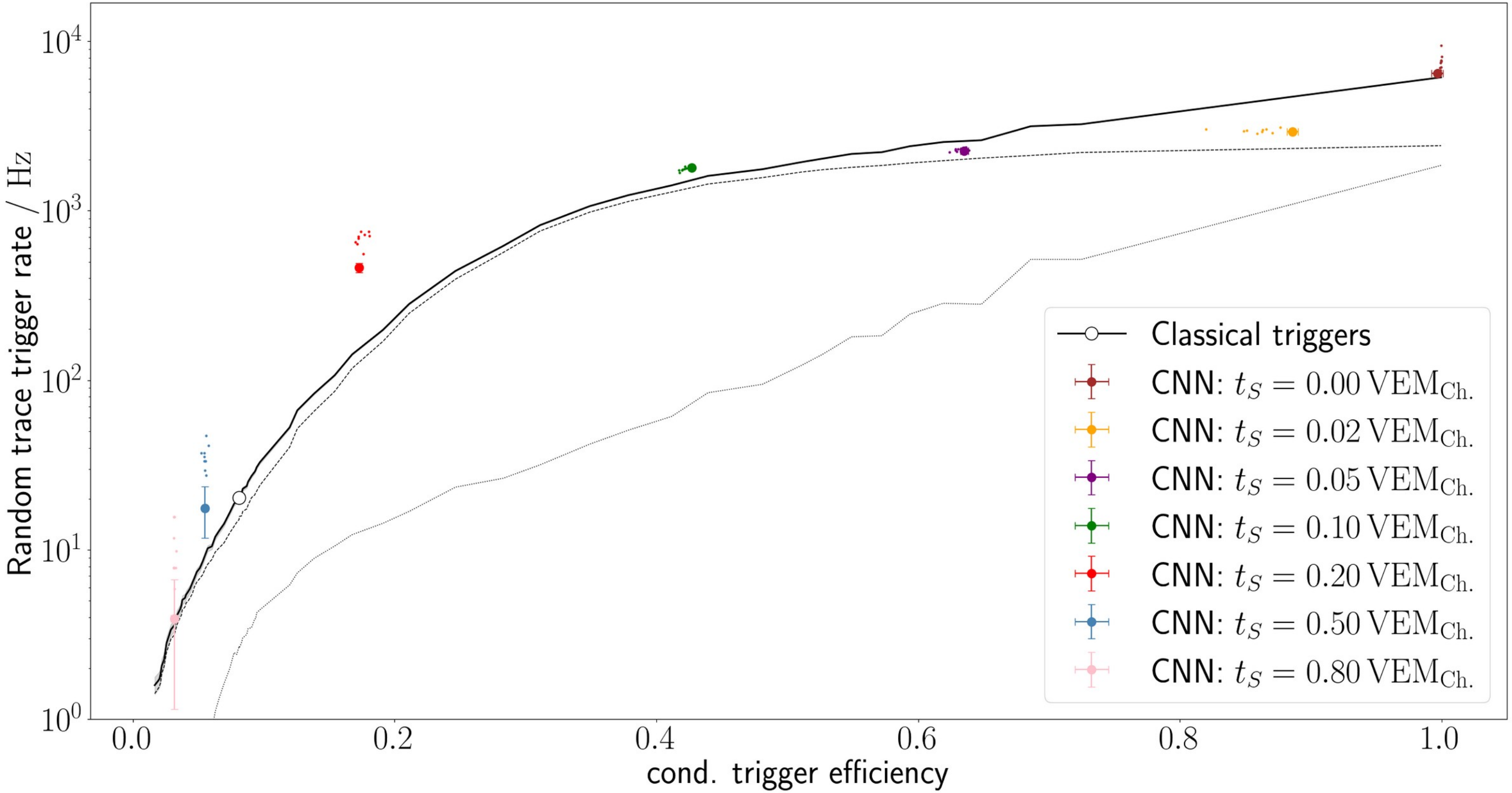
Convolutional neural networks



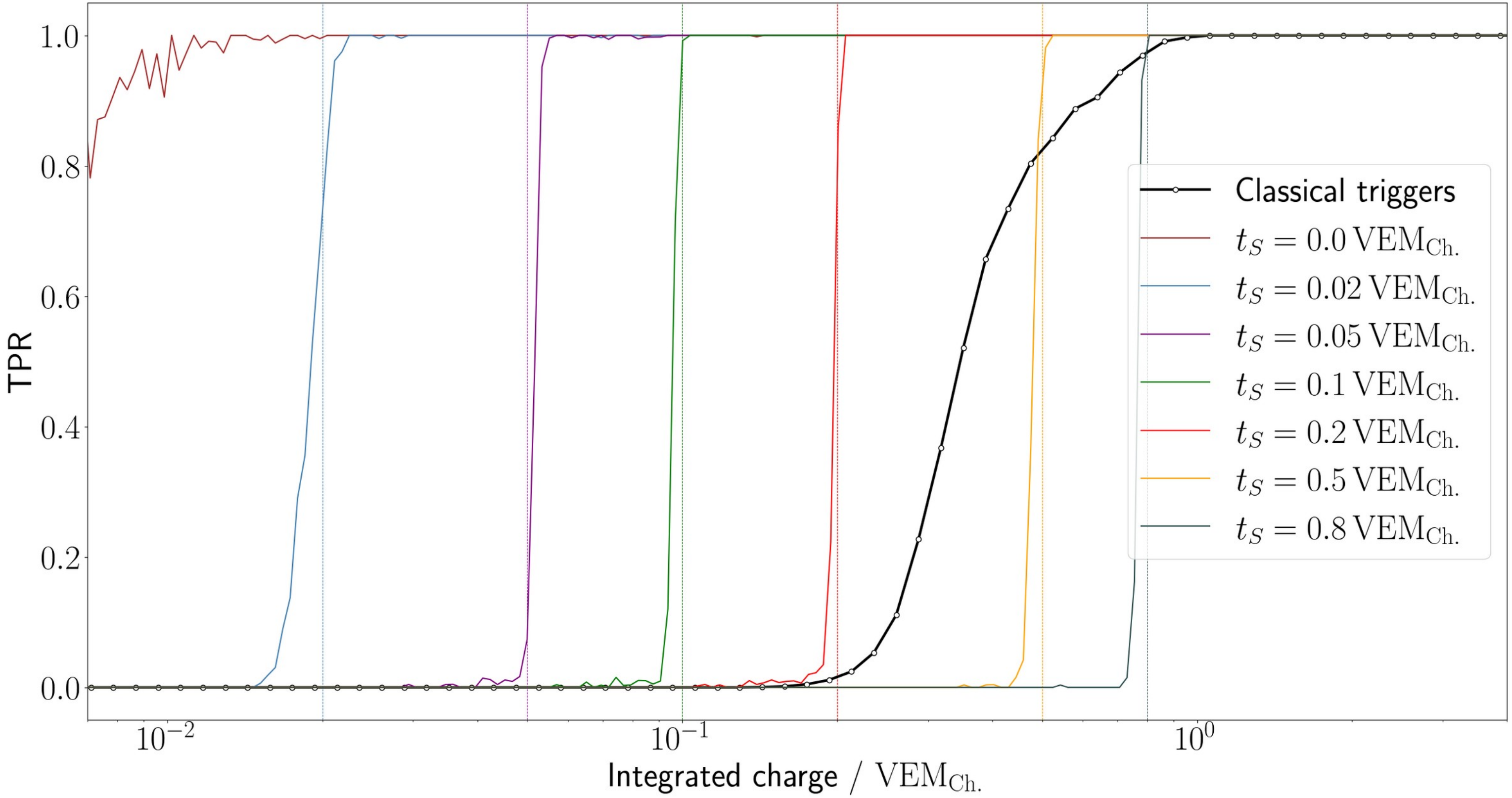
Convolutional neural networks



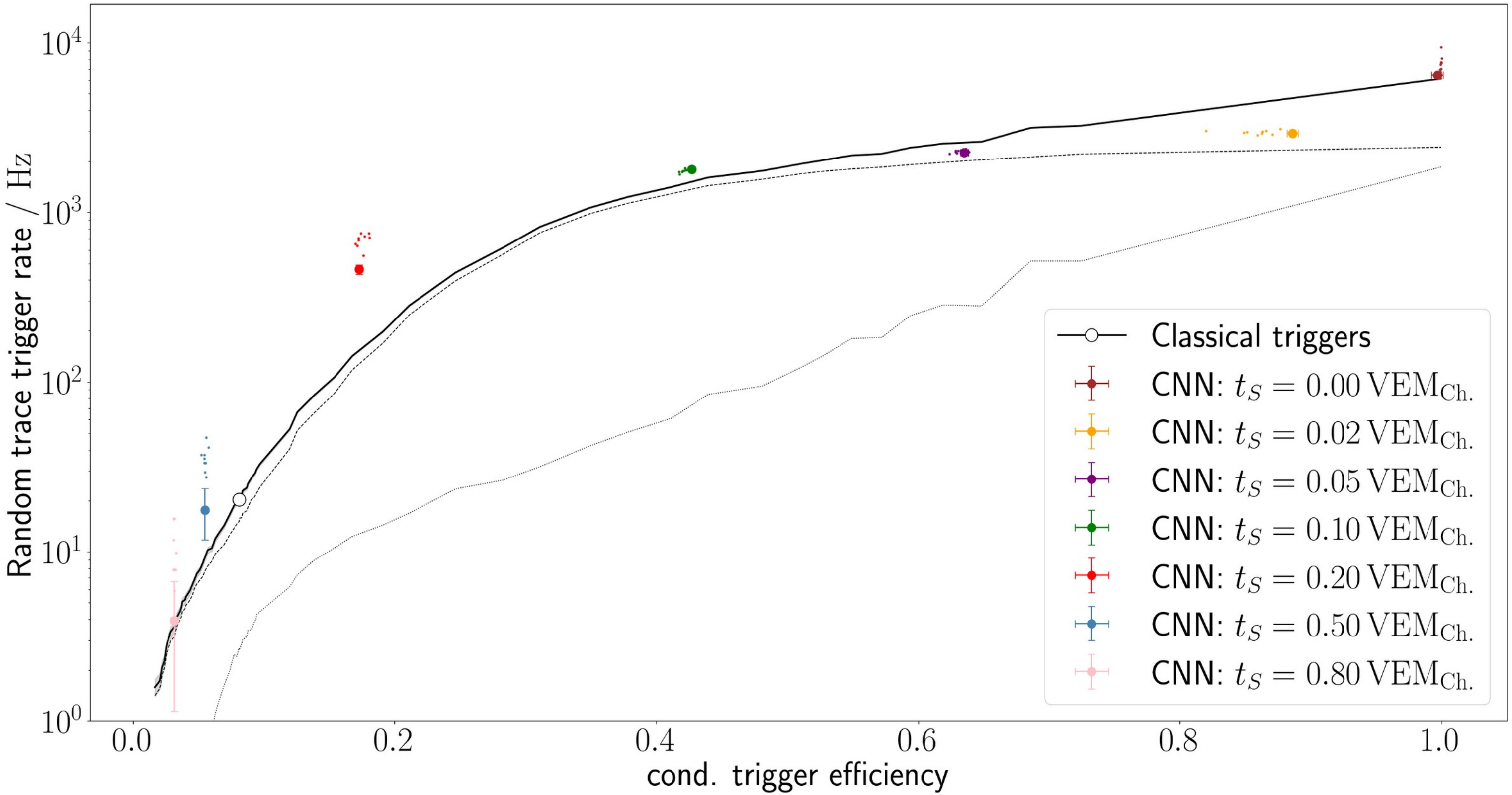
Convolutional neural networks



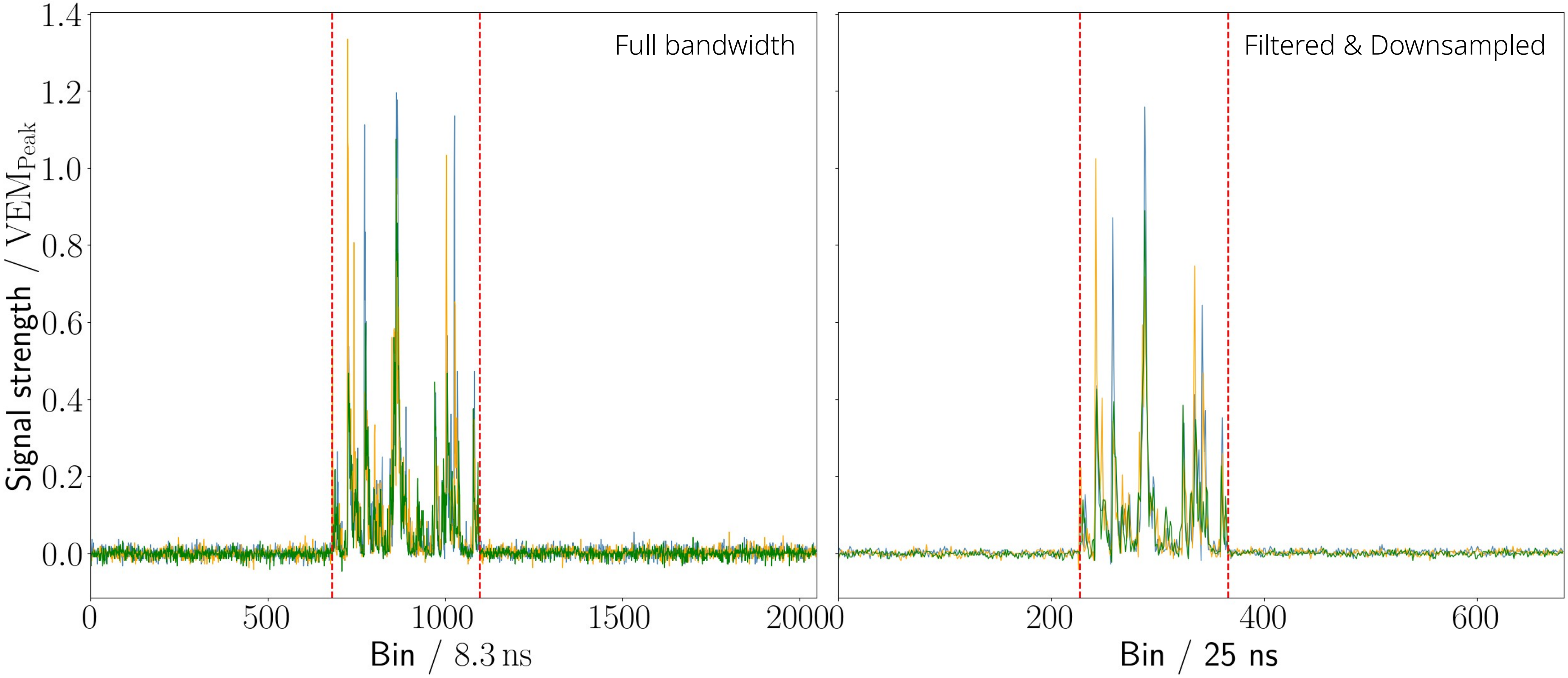
Convolutional neural networks



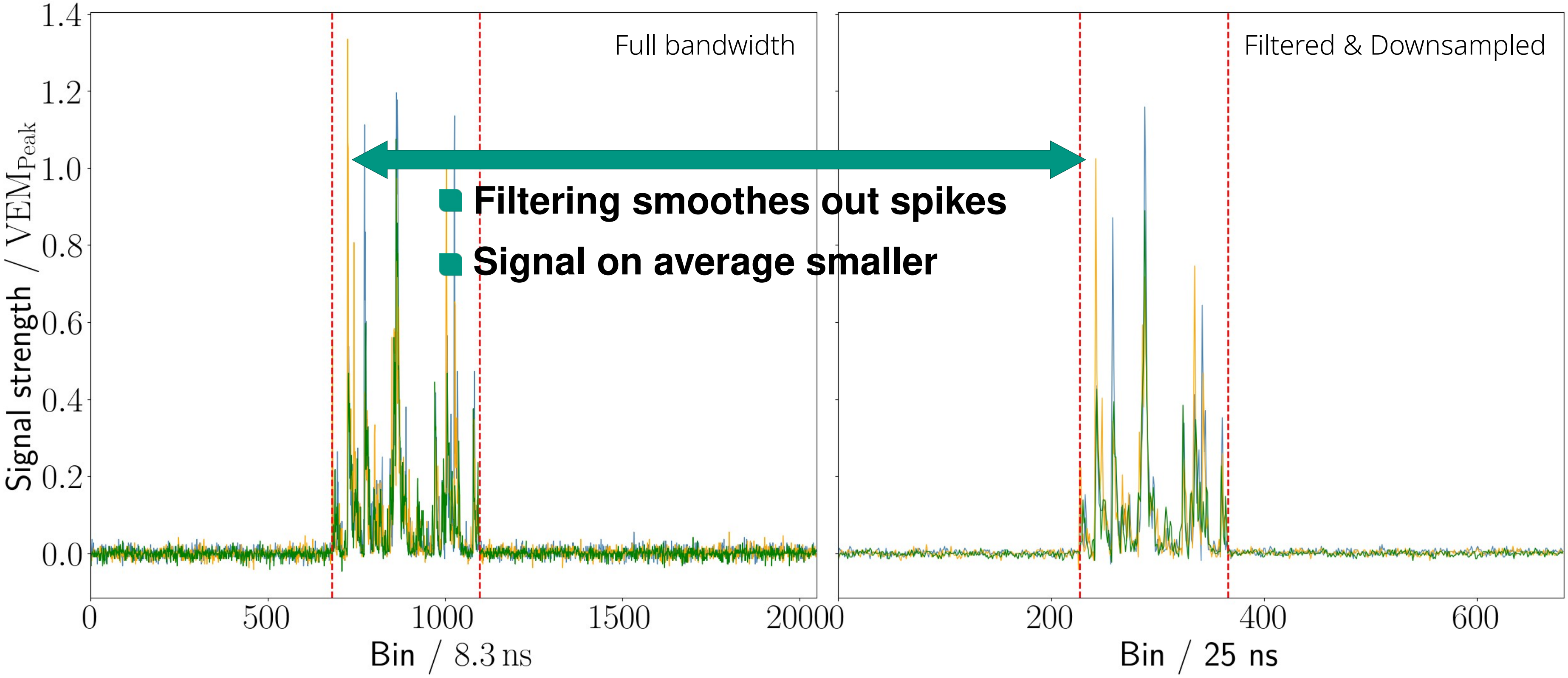
Convolutional neural networks



Filtering and downsampling

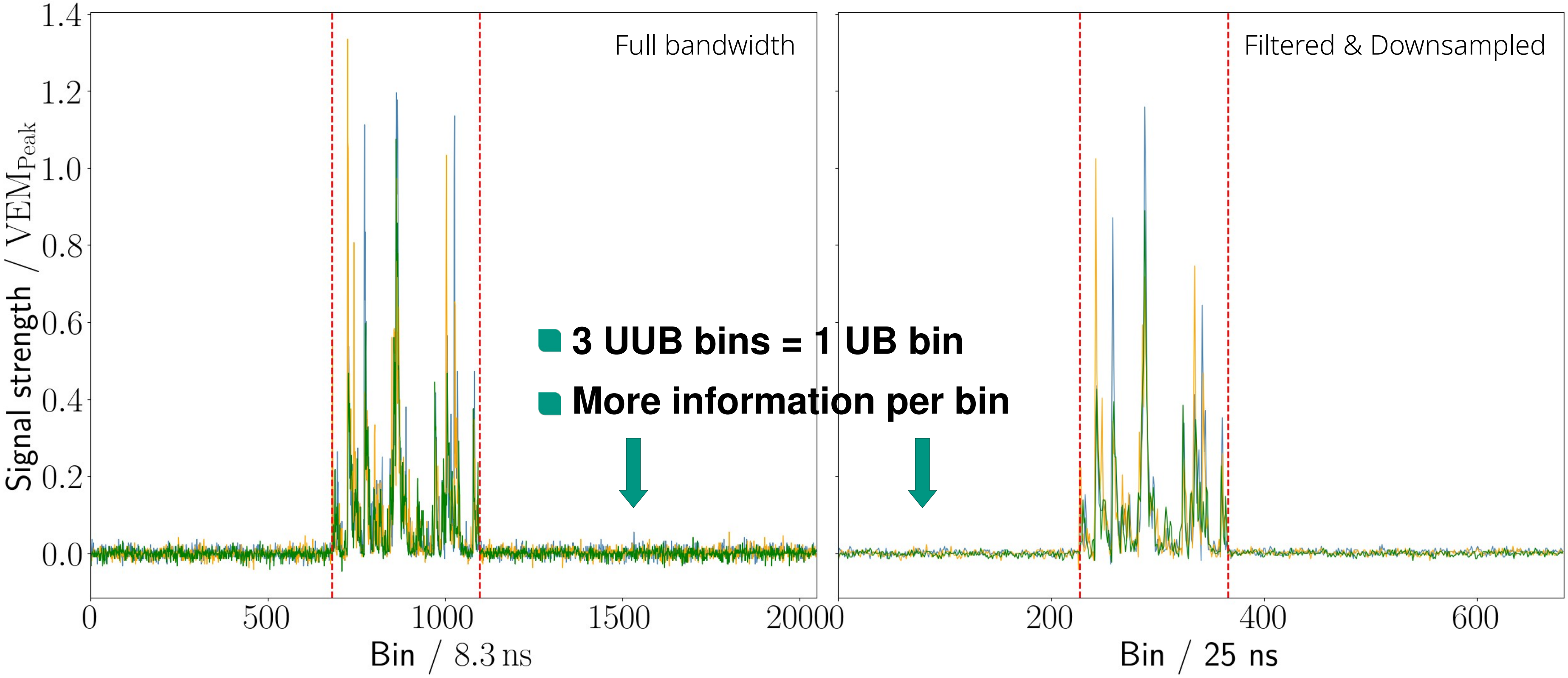


Filtering and downsampling



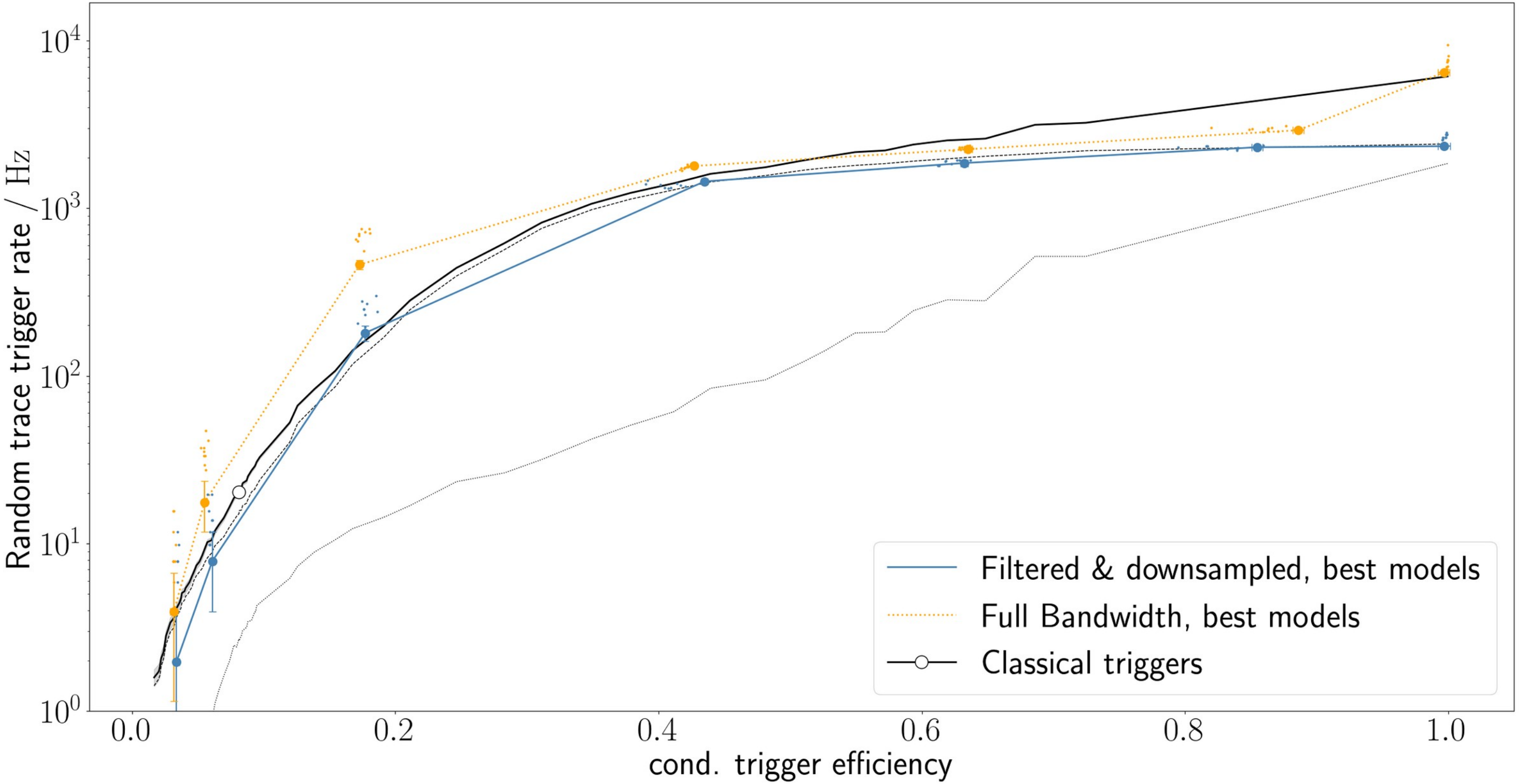
- Filtering smoothes out spikes
- Signal on average smaller

Filtering and downsampling



- 3 UUB bins = 1 UB bin
- More information per bin

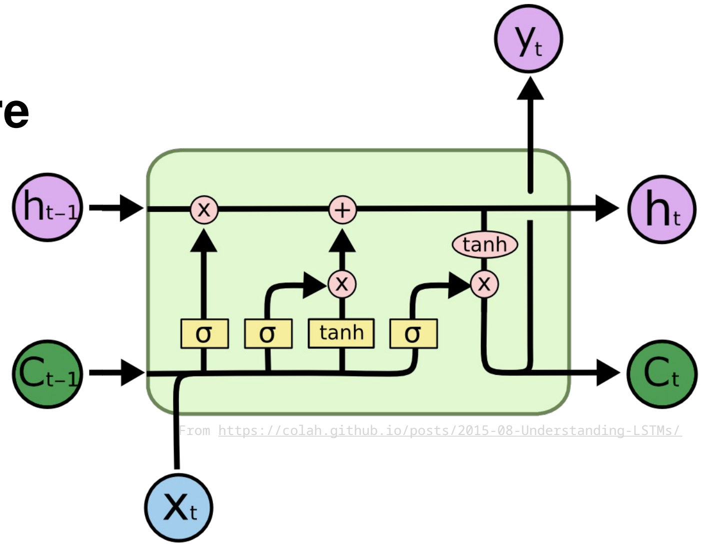
Convolutional neural networks



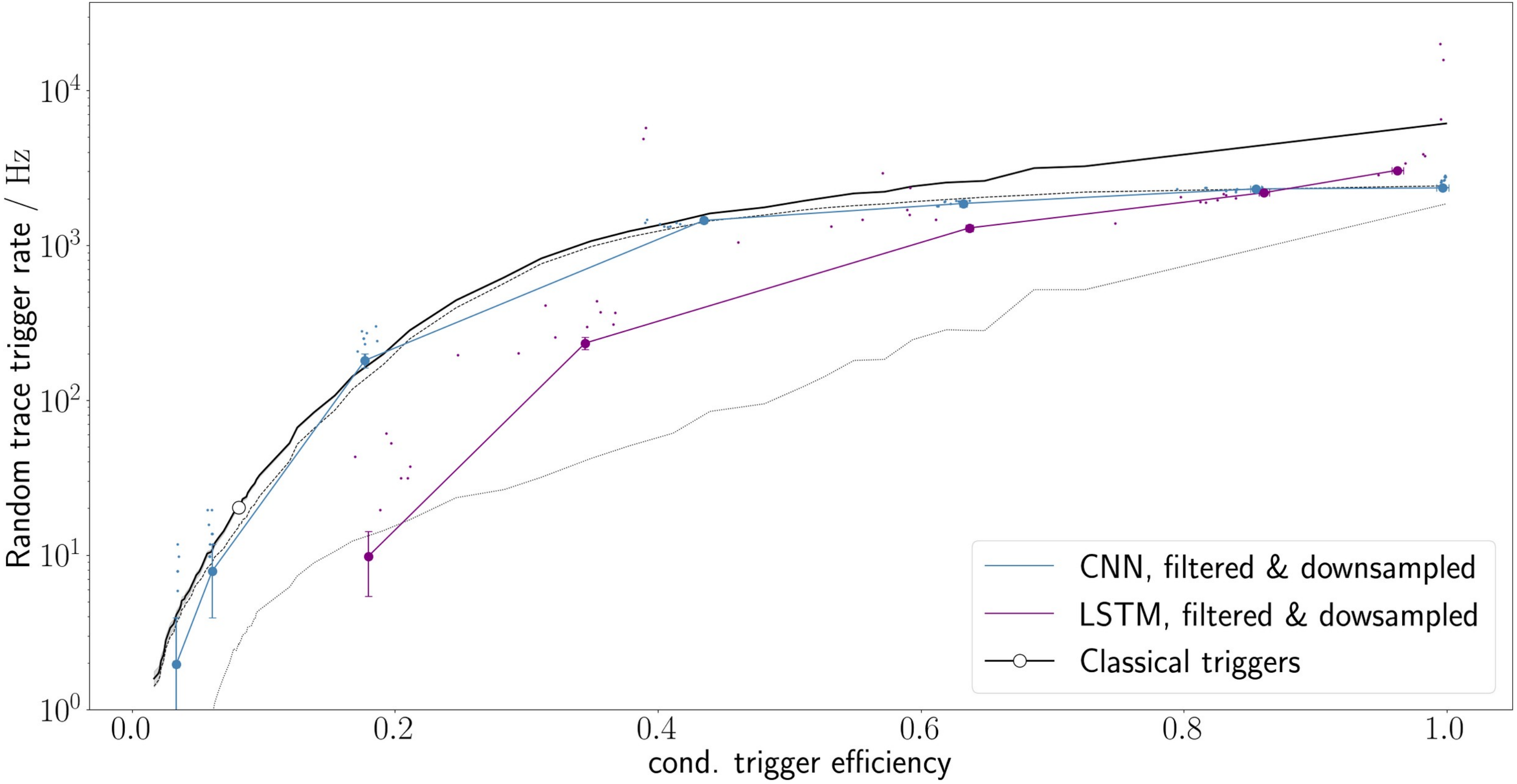
Trigger performance – Recurrent NNs

- Recurrent neural networks capture temporal data
  - Backwards connections make it possible to reevaluate output based on earlier input

- Long-Short-Term-Memory (LSTM) architecture
  - 12 free trainable parameters per cell
  - One cell per PMT = 36 parameters
  - Dense layer for binary output
  - = 44 (possibly 20?) free params

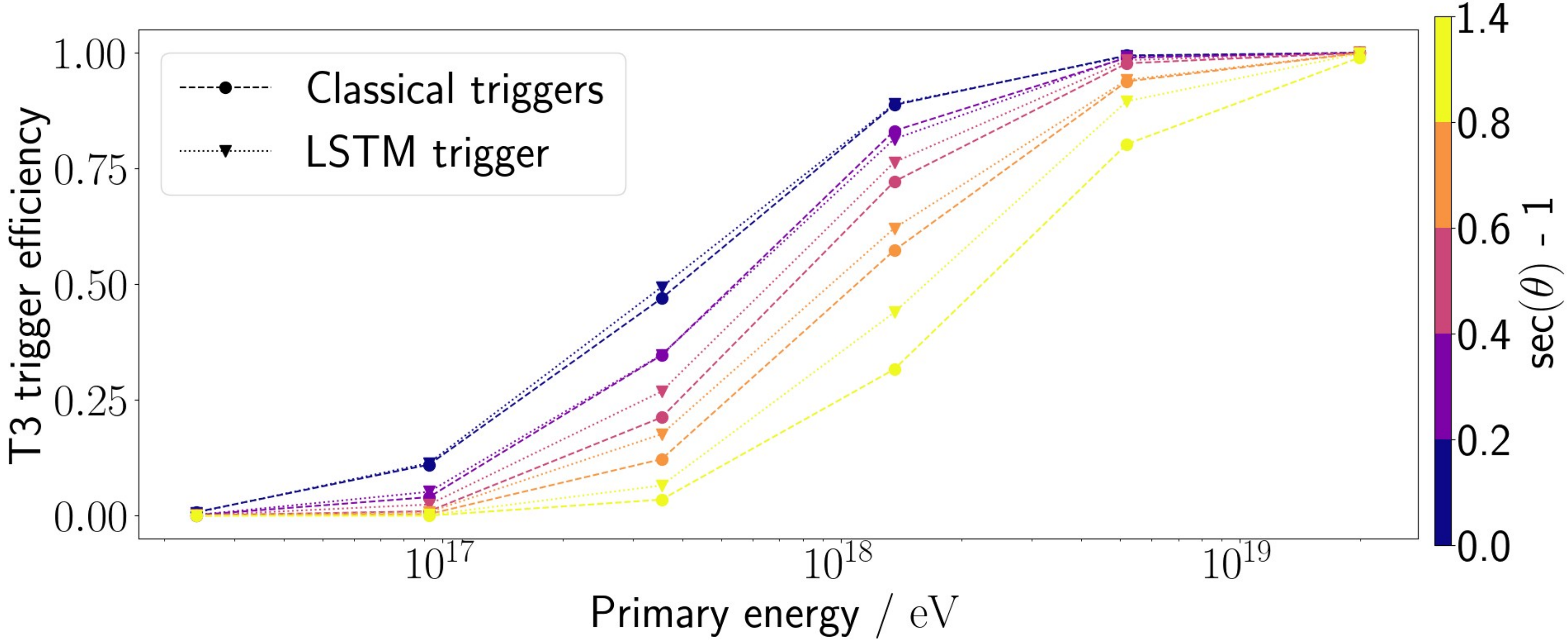


Recurrent neural networks



Recurrent neural networks

- **Most drastic gains at high inclinations, $\theta \approx 60\,°$**
- **Possibly higher gains at $\theta \geq 65\,°$**



Summary / Outlook

- Test data-driven, machine learning concepts
  - Bayesian classifier promising, but needs lots of finetuning
  - Neural networks work out of the box but „too efficient“
  - Control trigger rate by implementing charge cut

- Convolutional neural networks
  - Performance of simple CNN architectures on par with Th-Trigger
  - Filtered & downsampled data preferred over full bandwidth input

- LSTM / recurrent neural networks
  - First results indicate better performance than ToT
  - Large gains in event detection efficiency at high shower angles

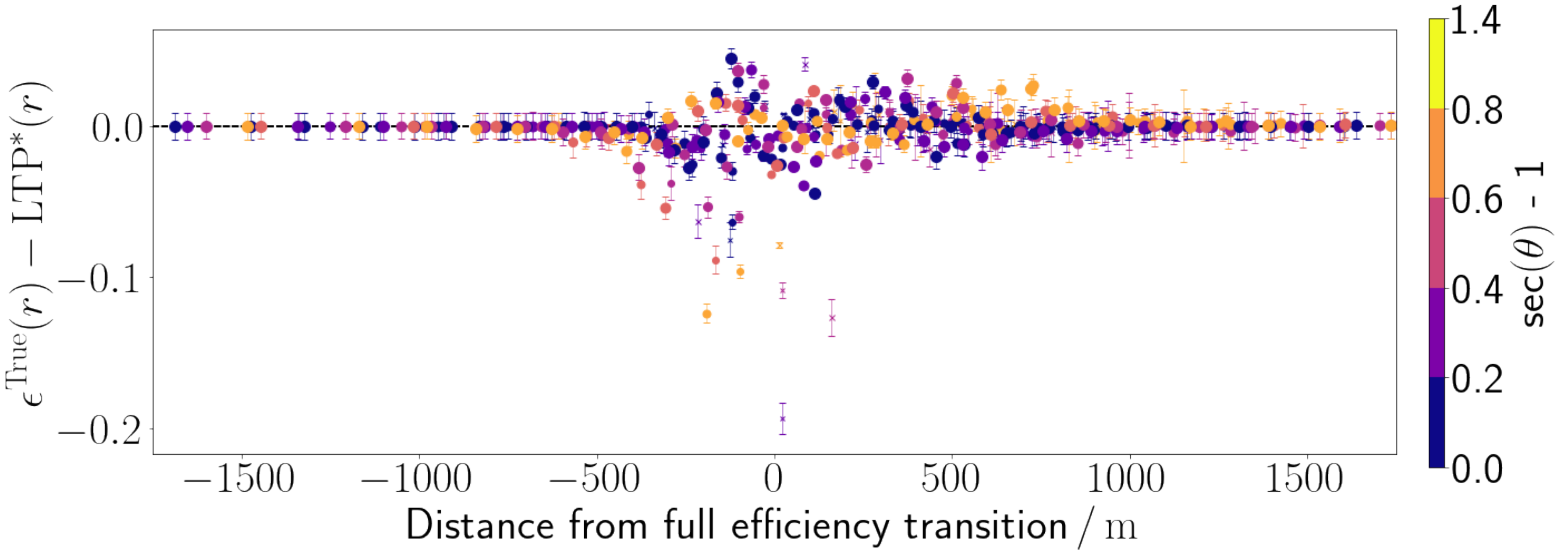
Summary / Outlook

- **Lot of work needed until prototyping stage is left**
  - **Presented results stem from simulations only**
  - **No primary distinction, only data from protons considered**
  - **Only one hadronic interaction model (QGSJET-II.04)**
- **Ground work is completed**
  - **Key assumptions have been tested and verified to hold true**
  - **Analysis chain is implemented and ready to run**
  - **Neural networks show a lot of potential as SD triggers**

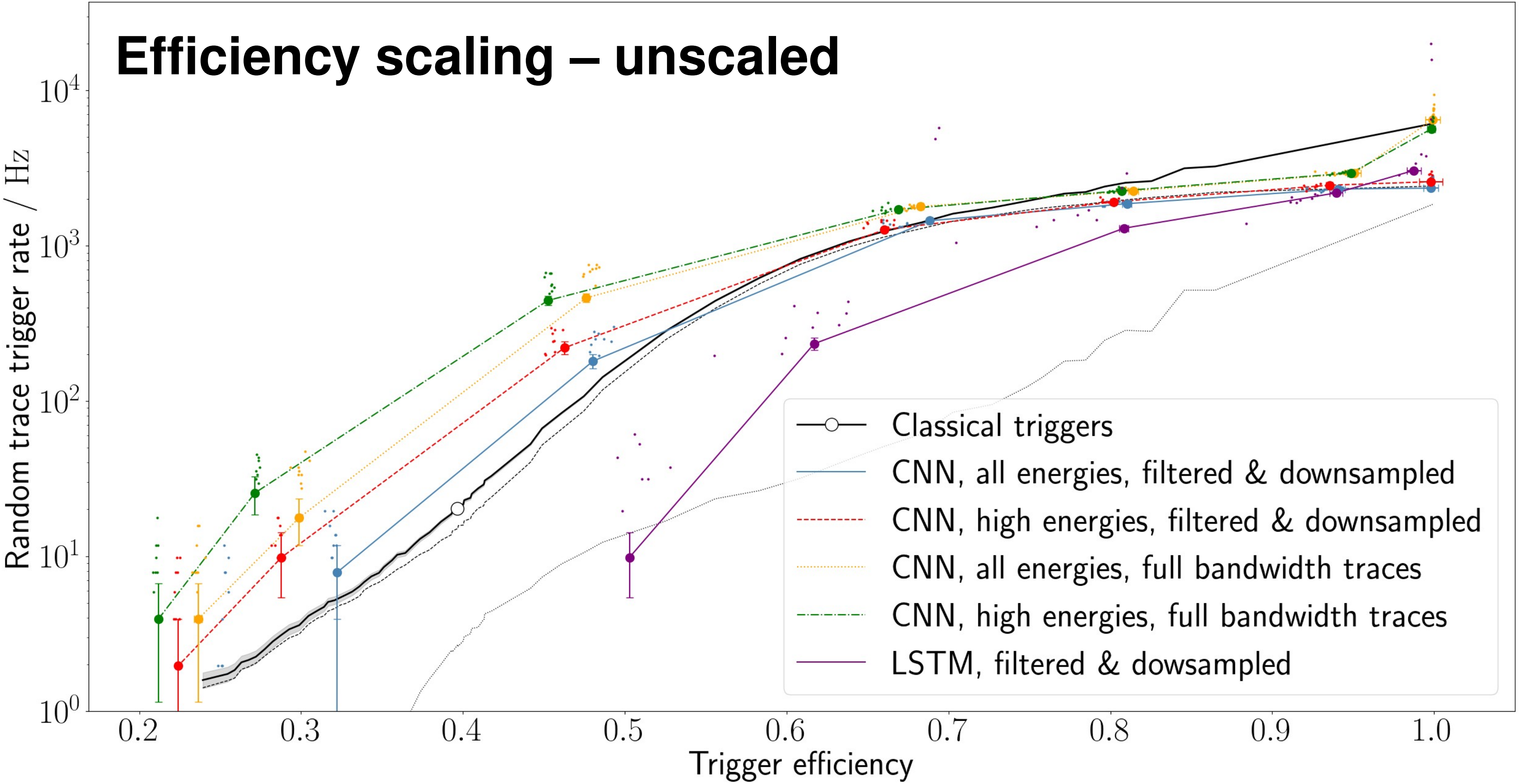
Backup

Residuals – LTP fitfunction

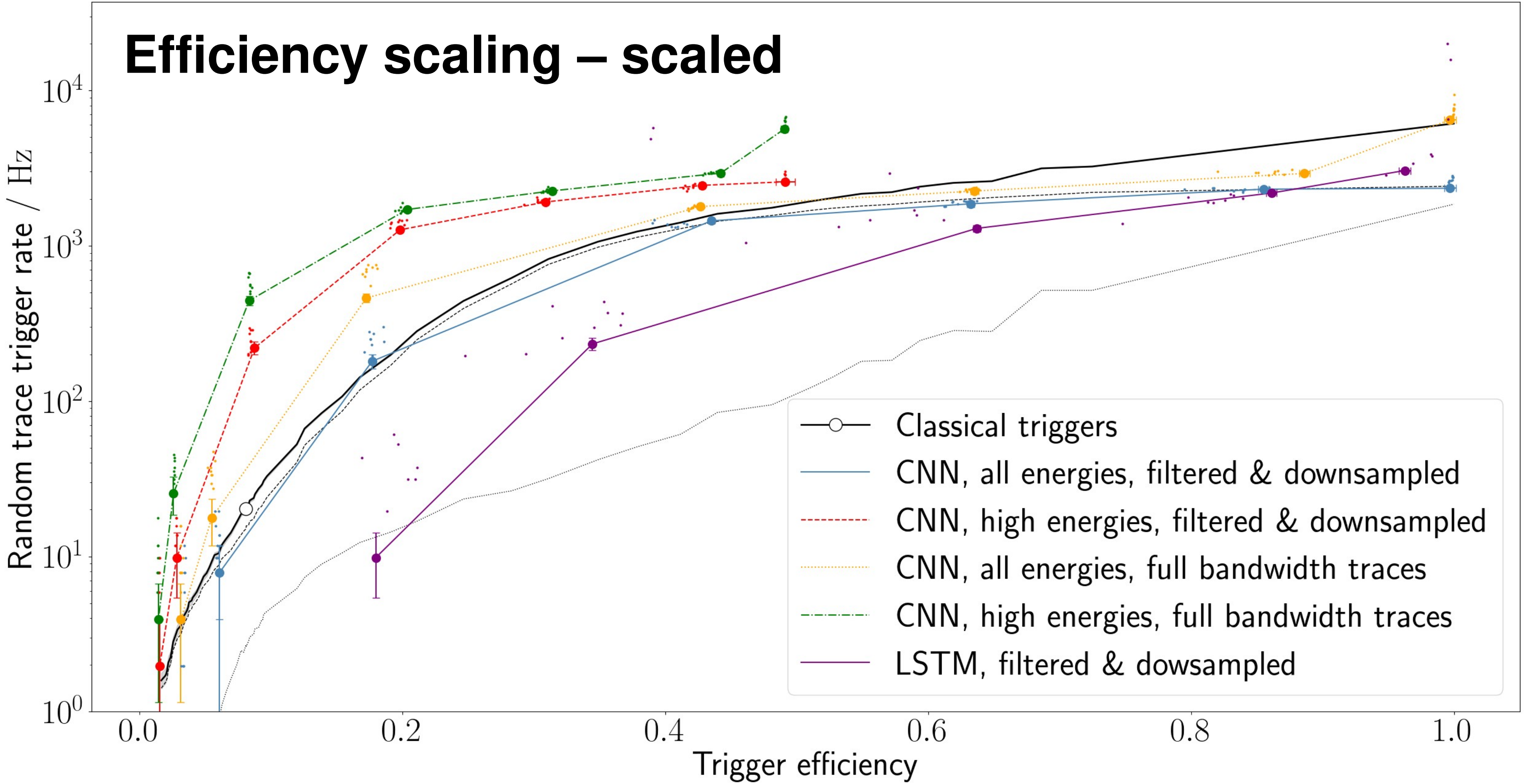
Karlsruhe Institute of Technology



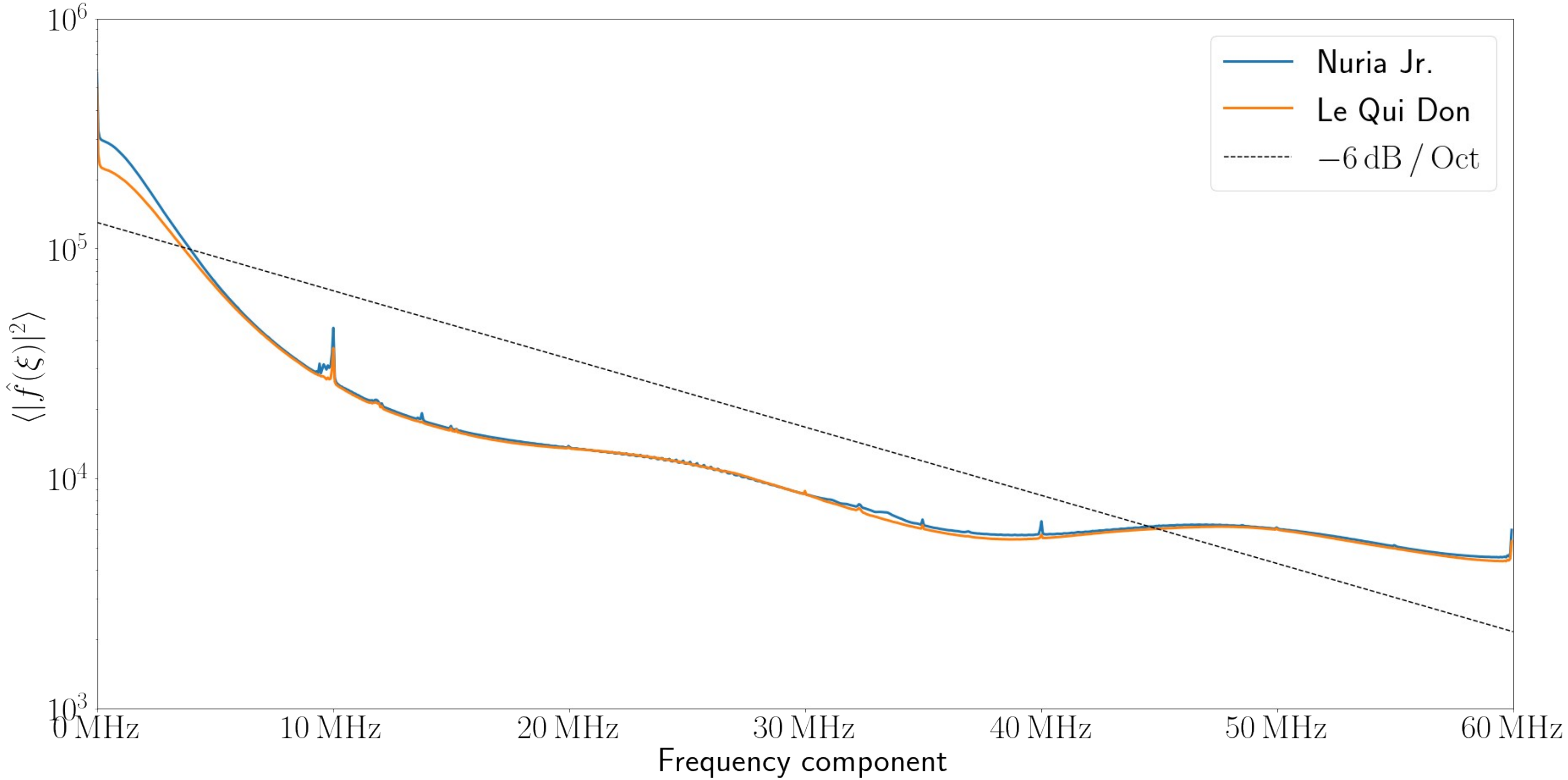
Efficiency scaling – unscaled



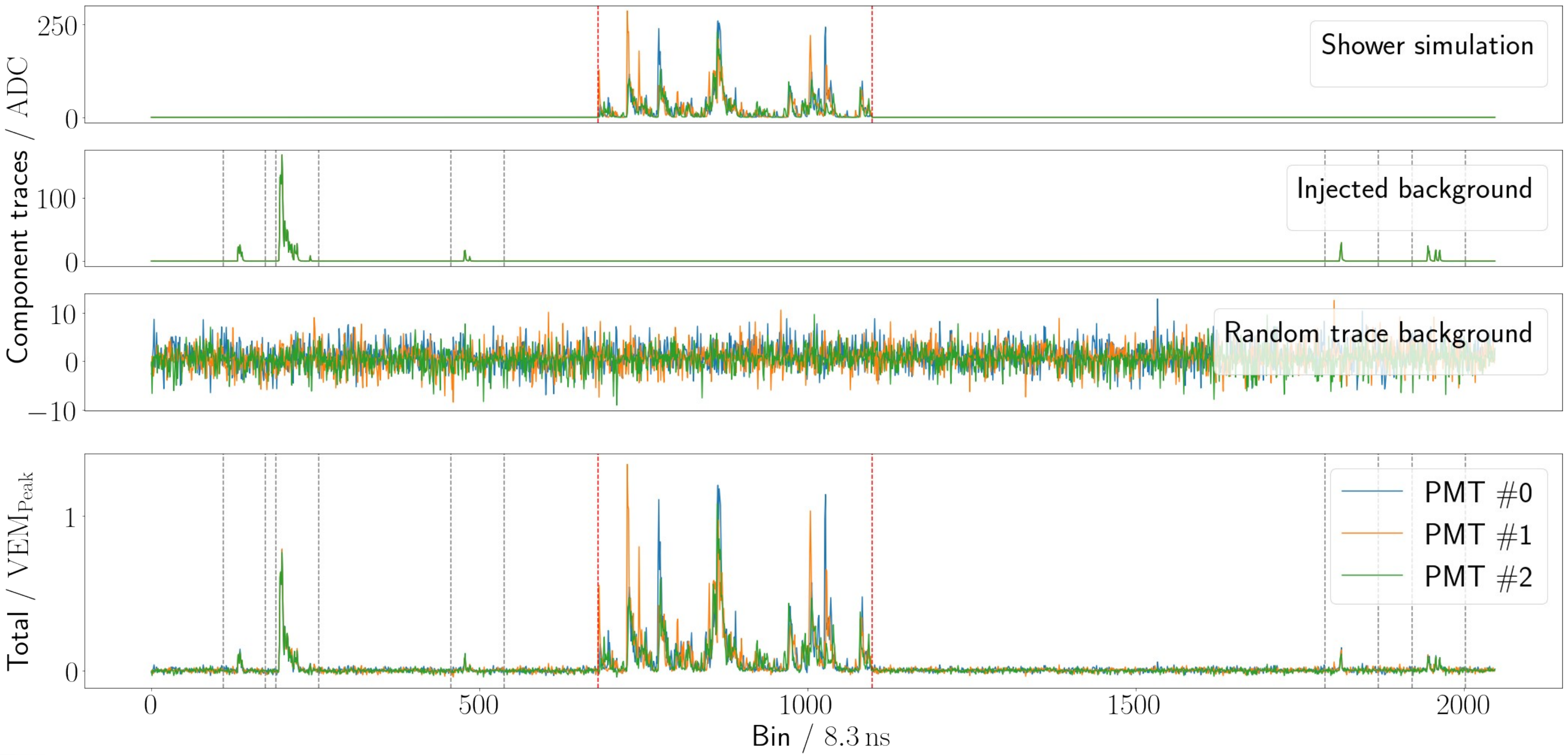
Efficiency scaling – scaled



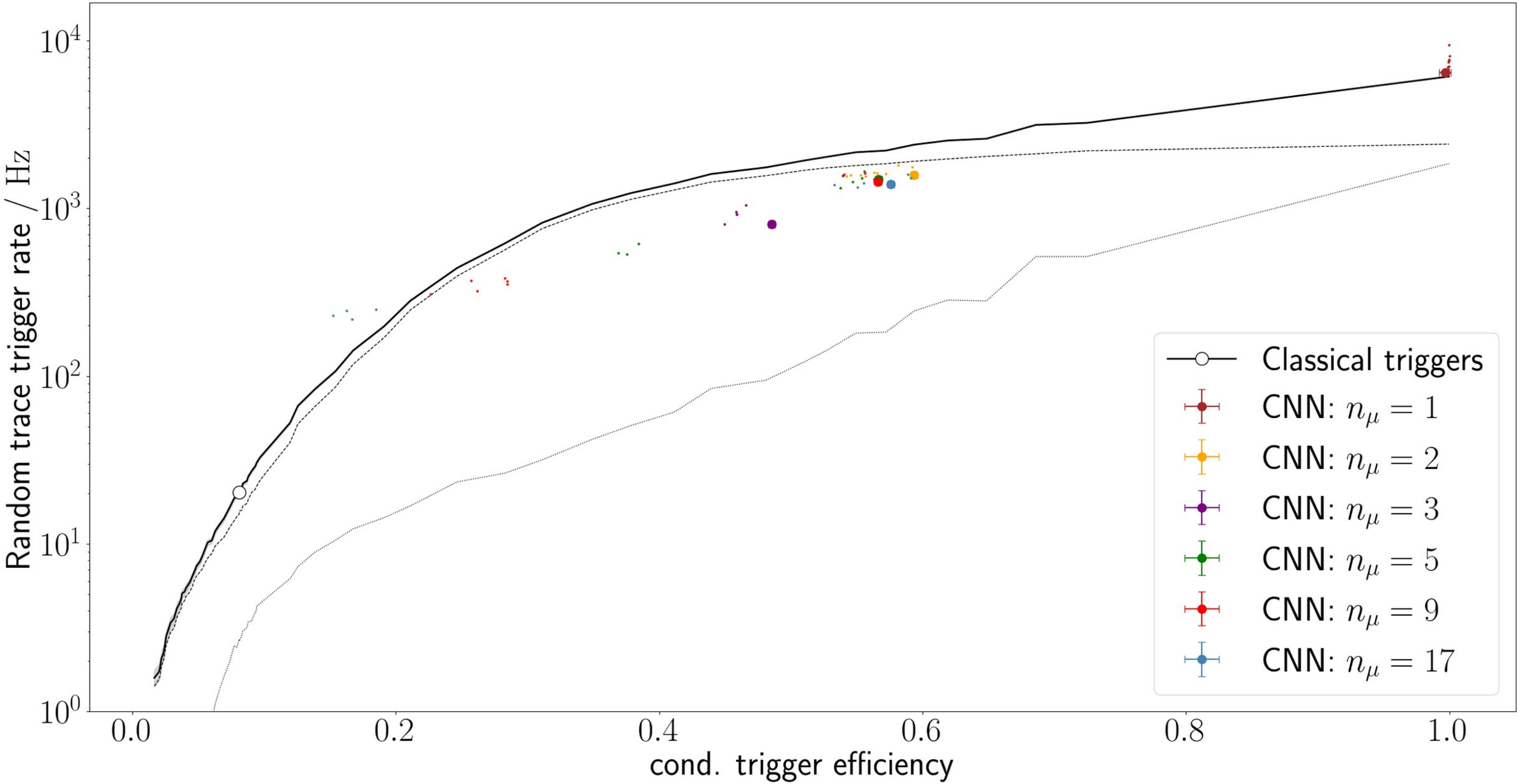
Random traces – Power spectrum



Trace building

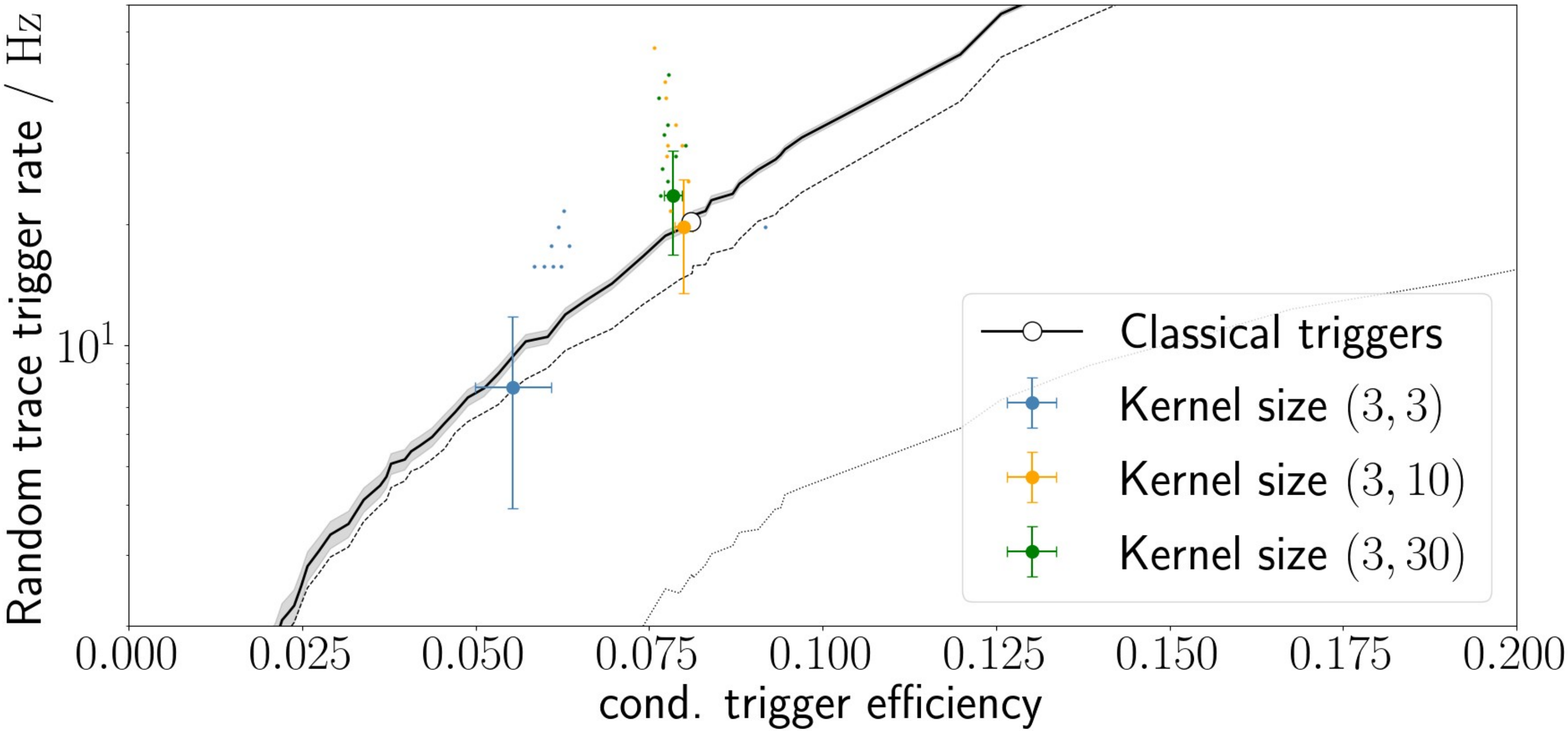


Muon cut



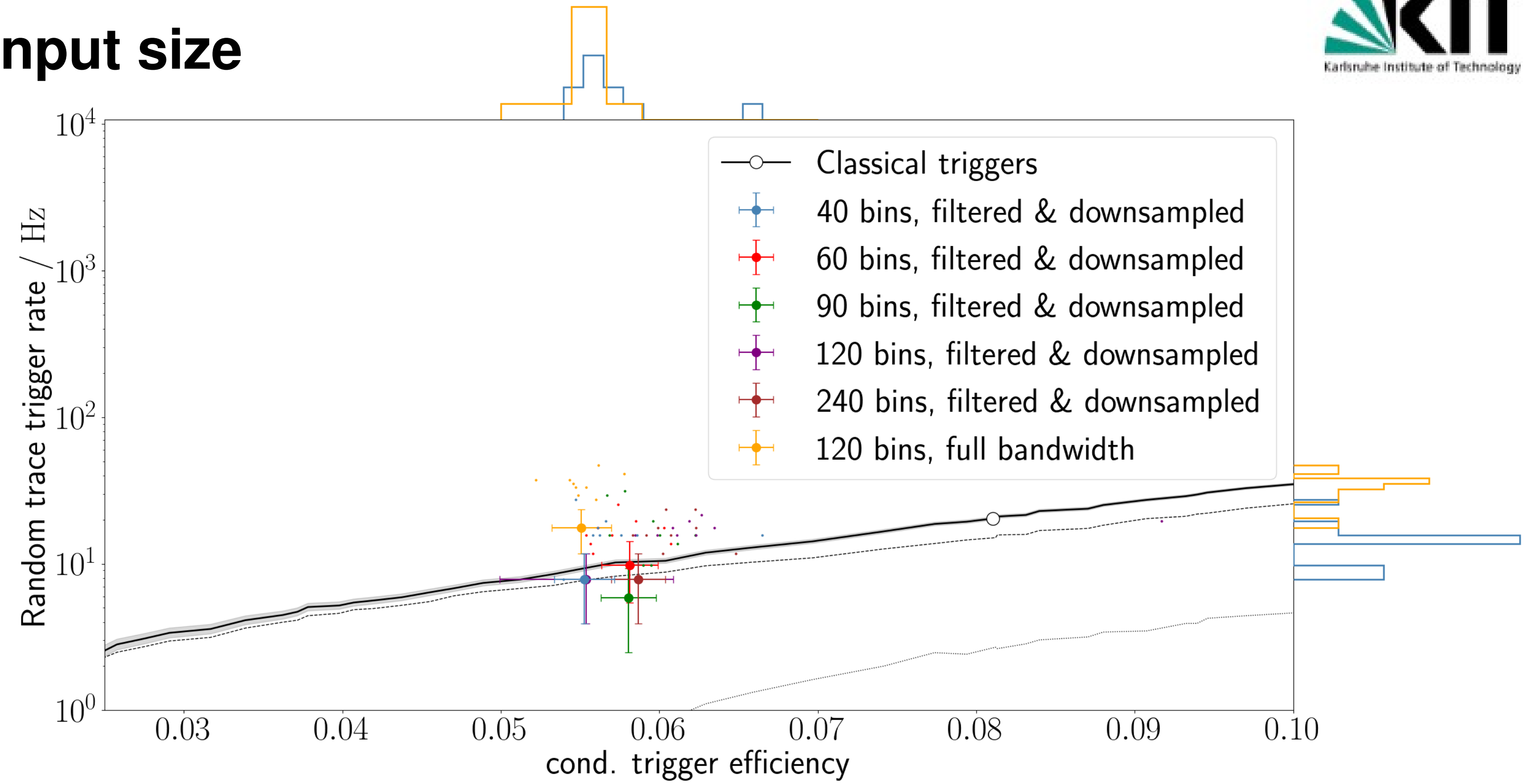
Random trace trigger rate / Hz
10^0
10^1
10^2
10^3
10^4
0.0
0.2
0.4
0.6
0.8
1.0
cond. trigger efficiency
Classical triggers
CNN: $n_\mu = 1$
CNN: $n_\mu = 2$
CNN: $n_\mu = 3$
CNN: $n_\mu = 5$
CNN: $n_\mu = 9$
CNN: $n_\mu = 17$

Kernel size

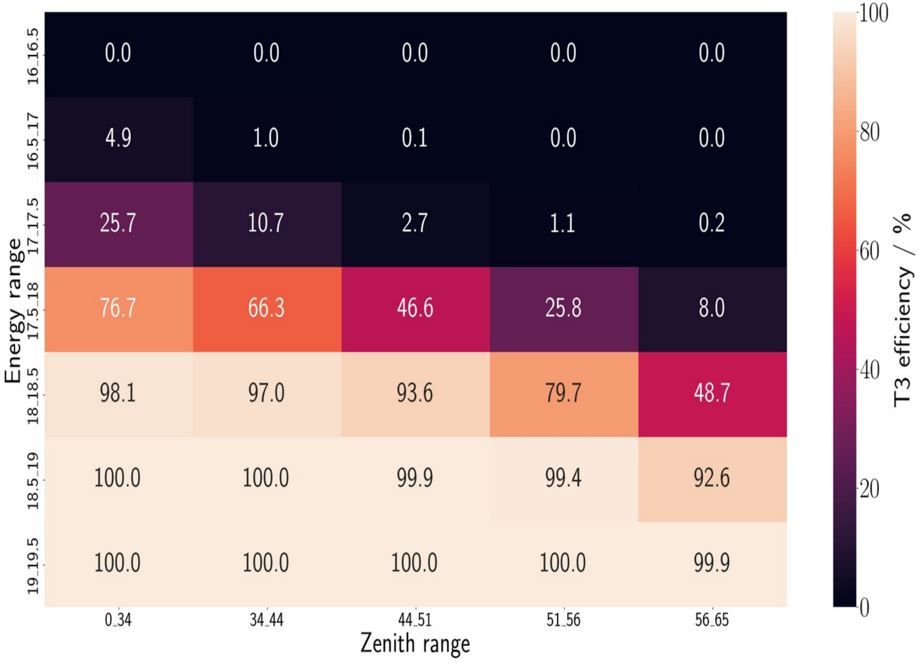


Random trace trigger rate / Hz
10^1
0.000
0.025
0.050
0.075
0.100
0.125
0.150
0.175
0.200
cond. trigger efficiency
Classical triggers
Kernel size $(3, 3)$
Kernel size $(3, 10)$
Kernel size $(3, 30)$

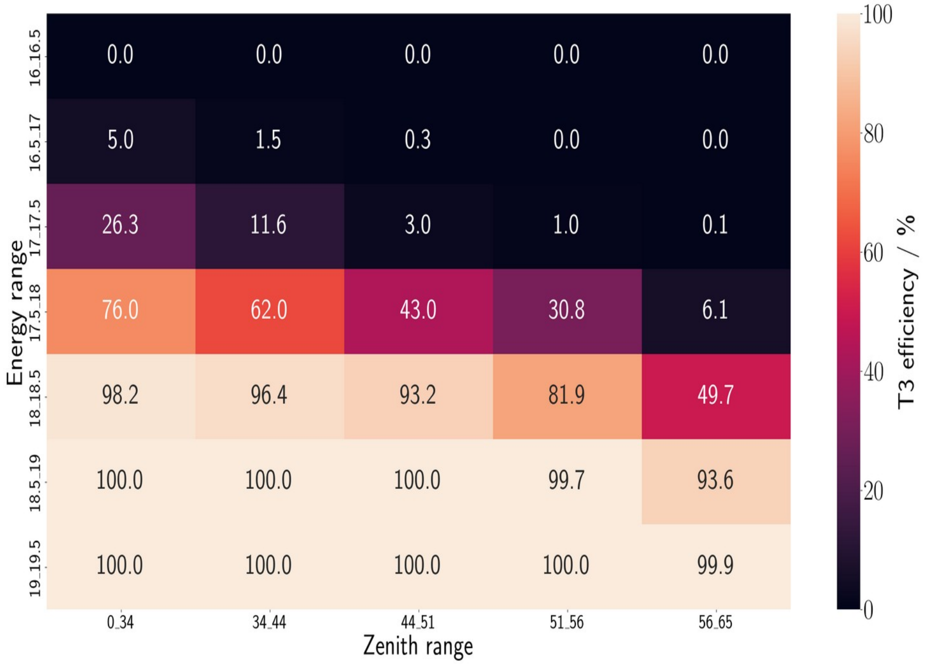
Input size


Random trace trigger rate / Hz
10^0
10^1
10^2
10^3
10^4
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.10
cond. trigger efficiency
Classical triggers
40 bins, filtered & downsampled
60 bins, filtered & downsampled
90 bins, filtered & downsampled
120 bins, filtered & downsampled
240 bins, filtered & downsampled
120 bins, full bandwidth

T3 efficiency calculation

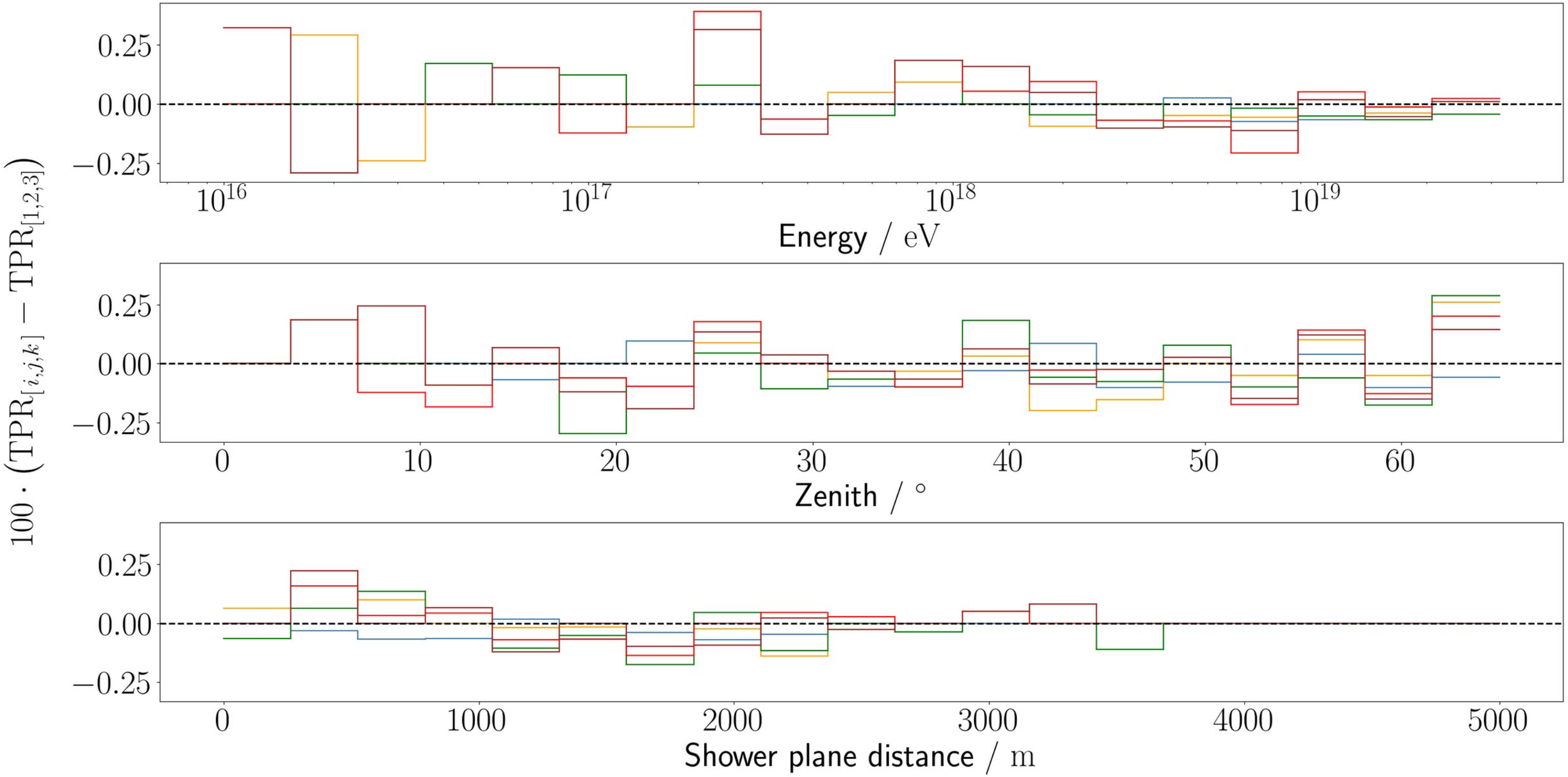


Offline approach



Bayesian folding

LSTM permutations



Network parameters

Type	Input size	Kernel size	n_{train}	w/ dense extension
CNN	$(3, 120)$	$(3, 3)$	**140**	**834**
CNN	$(3, 120)$	$(3, 10)$	**216**	**534**
CNN	$(3, 120)$	$(3, 30)$	**444**	**714**
CNN	$(3, 40)$	$(3, 3)$	**84**	**210**
CNN	$(3, 60)$	$(3, 3)$	**100**	**290**
CNN	$(3, 90)$	$(3, 3)$	**120**	**390**
CNN	$(3, 240)$	$(3, 3)$	**220**	**890**
LSTM	$(3, 120)$	–	**12**	(single layer)
LSTM	$(3, 120)$	–	(three layers)	**44**