Progress Summary

What has been done so far?

CALO5D face-to-face meeting, Bonn, 15 October 2025

Bohdan Dudar

bdudar@uni-mainz.de

How it all started

Continuing studies of software compensation with neural networks and timing

Literature

Timing *has potential to contribute*
to the energy resolution of neutral hadrons

Jack Rolph	2023	PhD thesis
C. Graf and F. Simon	2022	JINST 17 P08027
N. Akchurin et al	2021	JINST 16 P12036

What can we improve?

How much *quantitatively* in a *realistic setting*?
How does it *impact on JER*?

AHCAL prototype	→	full detector (ECAL+HCAL)
single pions	→	physics-like events
TB setup (energies/angle)	→	physics-like events
Briefly touch on timing	→	Focus on timing

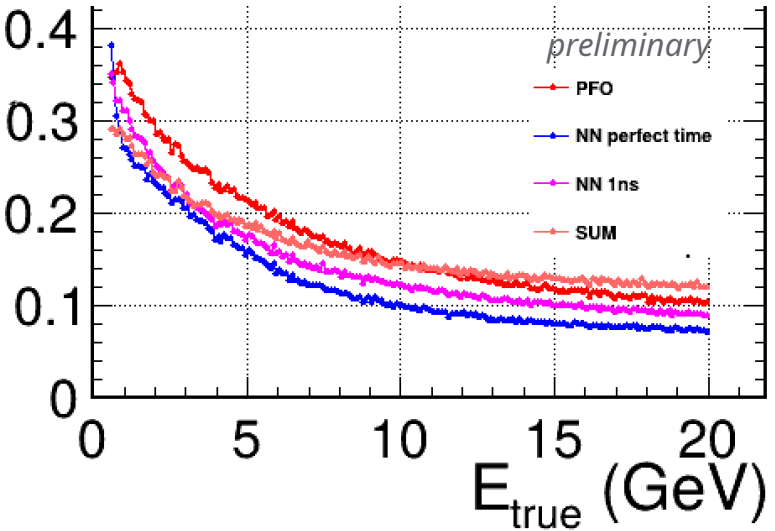
Towards the full detector

Establish workflow with full ILD simulation (still single pions)

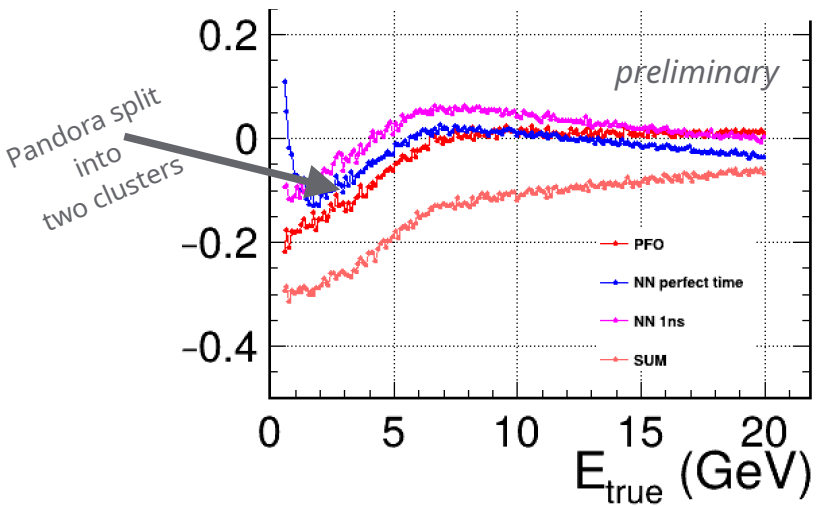
Setup		
Particle	π^-	
Momentum (uniform)	0.5 – 20 GeV/c	Physics-motivated lower energies. No 100 GeV pions (!)
Direction	up	
Gun position	(0, 1794, 0) mm	
Gun position smear	(150, 0, 470) mm	avoid gap in the detector (!)
Neural network	Jack-style (EdgeConv+MLP head)	MSE loss (!) (bad choice at that time)
Reference	PandoraPFA	

First results with π^- gun

π^- Resolution (RMS_{90})



Linearity ($Mean_{90}$)



Results

- NN outperforms Pandora on single pions
- Better timing → better resolution!
- Could be further optimized.

Two big buts…

- Pandora is optimized for physics, not single particles!
- Migrating “single-particle” trained NN on physics events is challenging due to confusions

Towards physics-like events

Establish workflow with Z→qq (uds) in full ILD simulation (only “neutral” hits)

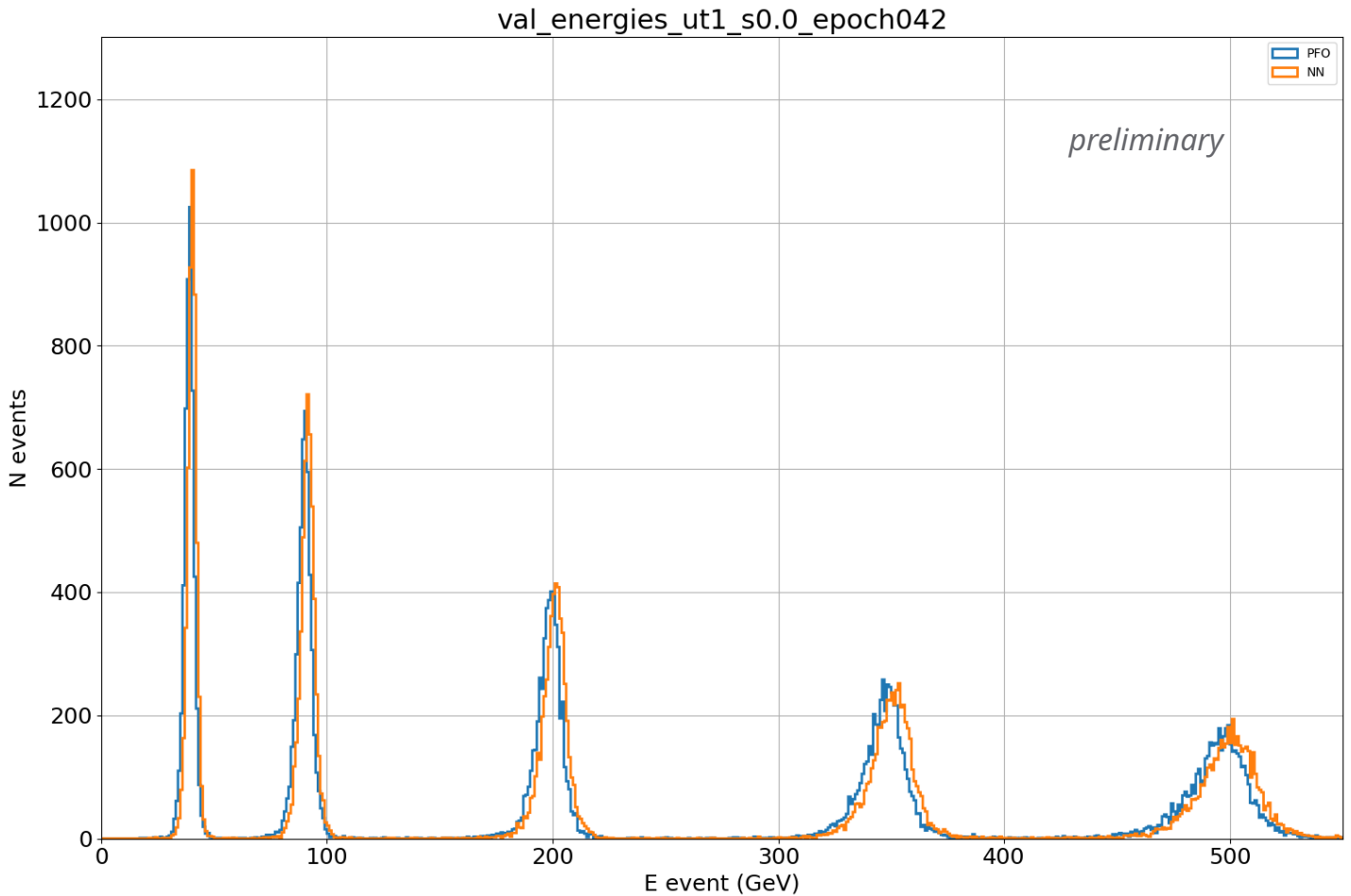
Setup		
Process	Z→ qq (uds)	Significantly more hits than pion gun
E_{CM}	40, 91, 200, 350, 500 GeV	ILD production from 2020. 91 and 350 are not used in training
Input	Only hits from neutral Pandora PFOs (!)	
Neural network	PointNet, JackNet, DGCNN	MSRE loss. $E_{CM,\,pred} = E_{NN,\,neutral} + E_{pandora,\,trk}$
Target	E_{CM}	A lot of effort spent defining the “neutral only” target. Not ideal because of anti-correlation of $E_{neutral}$ and E_{trk}
Reference	PandoraPFA	

First results with Z→qq (uds)

Validation RMS Relative Error (NNs with perfect timing)		Comment
Pandora	4.74 %	Reference
PointNet	4.33 %	No direct local neighborhood features
JackNet	4.15 %	Extremely slow
DGCNN	**4.04 %**	**Nice.**

DGCNN RMS looks much
better than Pandora!

First results with Z→qq (uds)



RMS looks much better:

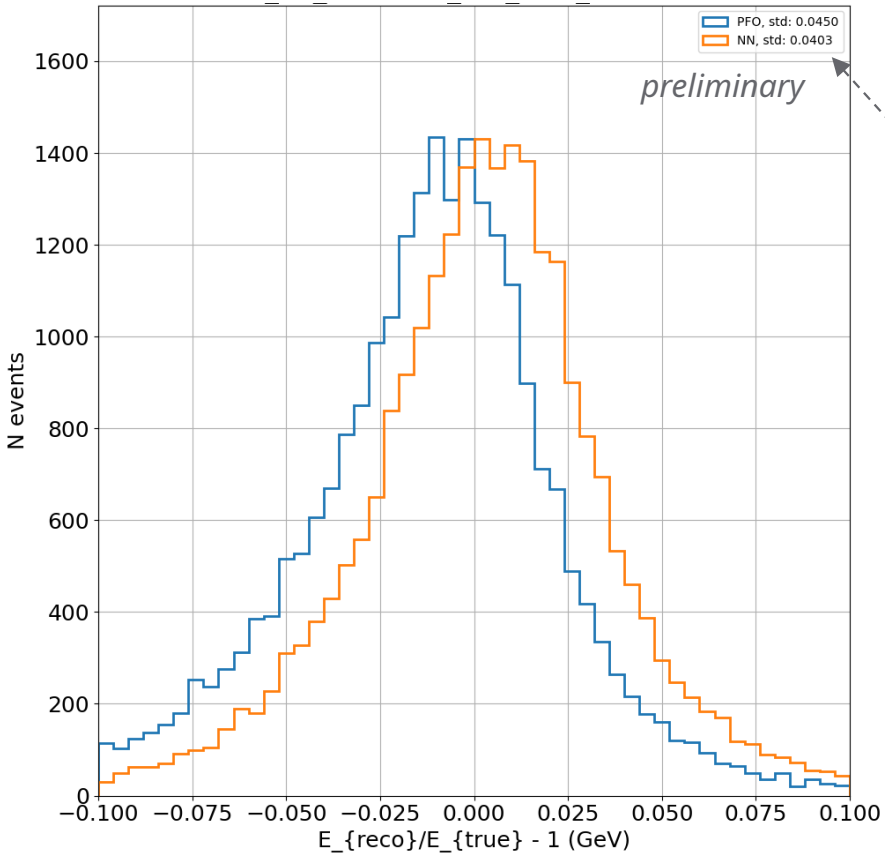
Pandora: 4.74%
DGCNN: 4.04%

But energy plot looks pretty similar

Why?

First results with Z→qq (uds)

Relative residual for all energies combined



Aha! Mainly because of the bias correction.
What if we take a look at *STD instead of RMS*?

STD *still* looks much better:

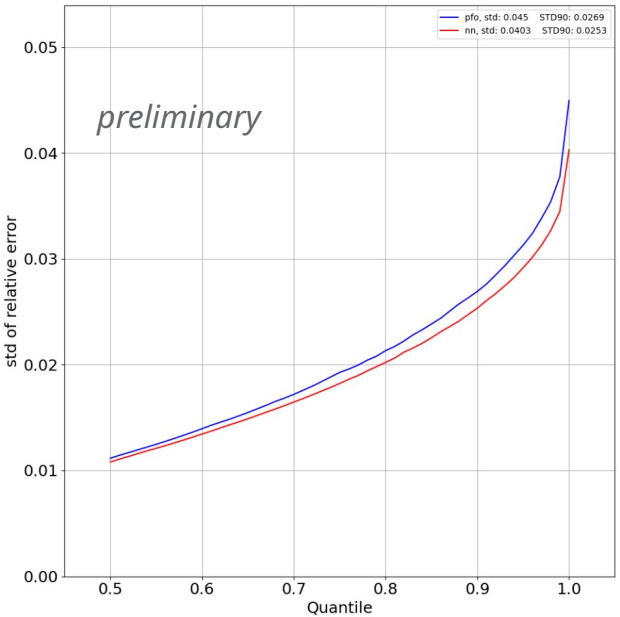
Pandora: 4.5%
DGCNN: 4.0%

But residual plot looks pretty similar again…

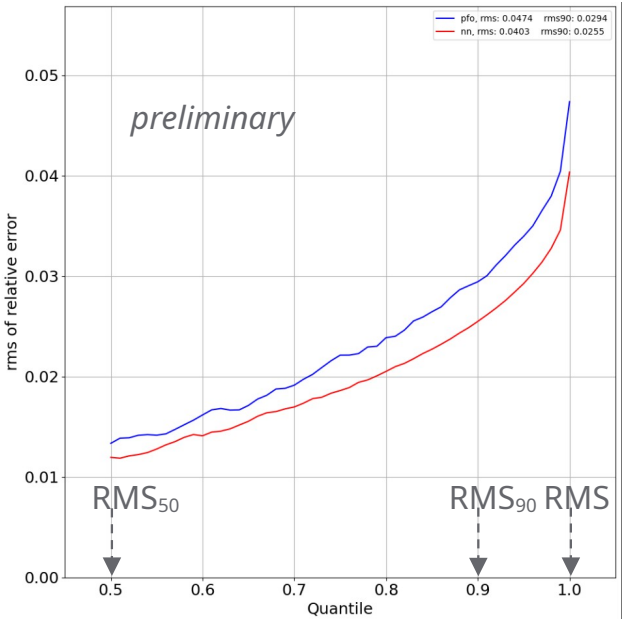
Why?

First results with Z→qq (uds)

STD



RMS



NN mainly improves the outliers, but not as much the core of the distribution

Results

- Nice.
  (slightly better than Pandora)

But…

- we are artificially limiting the performance by throwing away all “charged” hits.
- It might be unfair to make a conclusion for impact of timing at this stage!

Can we improve?

Towards using all hits

Solve the confusion as well. Software compensation → particle flow

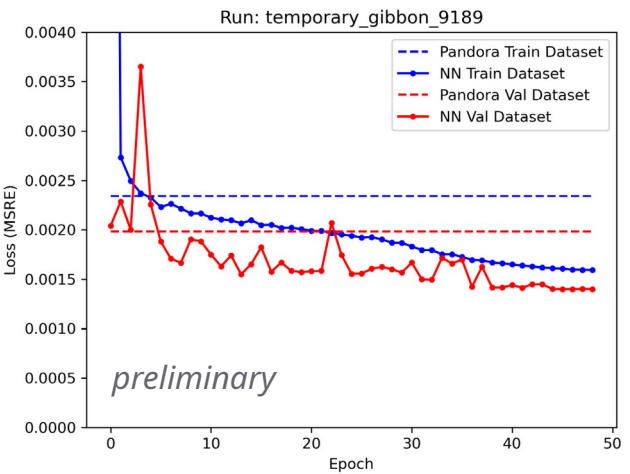
The main addition:

For each “charged” hit use $E_{\text{charged hit}} = E_{\text{track}}/N_{\text{cluster hits}}$

Current approach

Input	Hit features (details in the back-up): positional, energy, time (perfect time res.)
Model	Dynamic Graph CNN (DGCNN)
Output	Corrected per-hit energies (E_{hit})
Target	Generator level $E_{CM} = \Sigma_i E_{\text{hit, i}}$
Loss	MSRE (Mean Squared Relative Error)
Train data	E_{CM}: 40, ~~91~~, 200, ~~350~~, 500 GeV (total 25k evts)
Val data	E_{CM}: 40, 91, 200, 350, 500 GeV (total 25k evts)

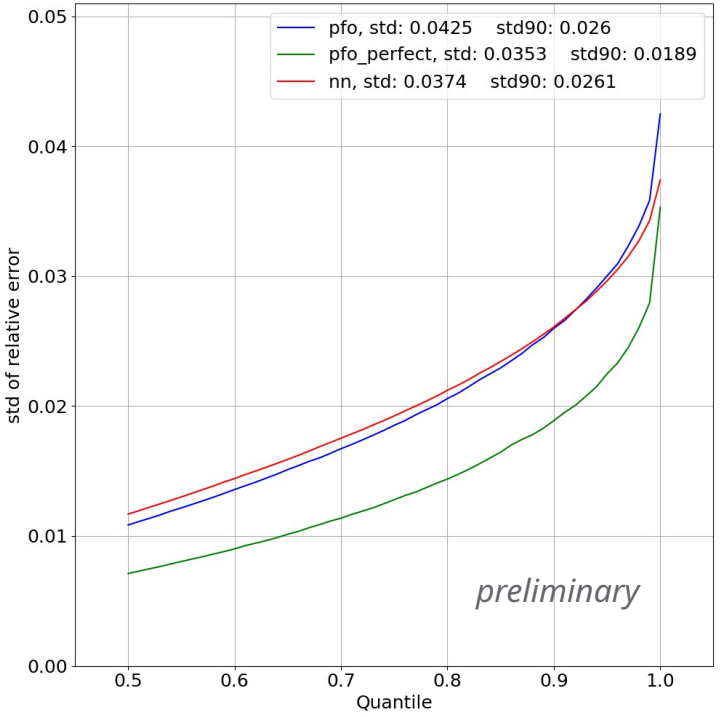
Training…



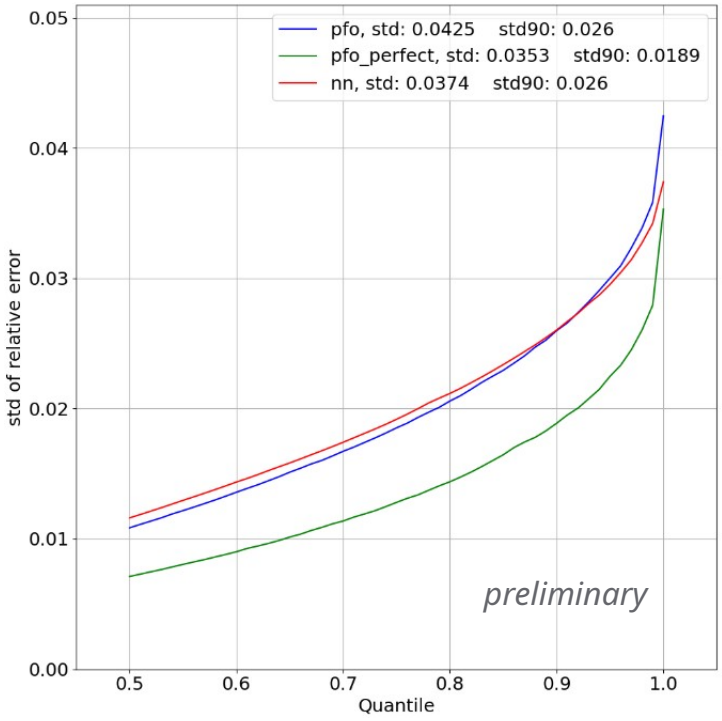
Latest results

Slightly worse performance when used only neutral hits (cheated track energies).
Too early to draw any conclusion on impact of timing. But no effect so far

No timing



Perfect timing



Also tried:
setting k_{NN} = 20, 50, 70
and removing 20% outliers
from training dataset.
No visible difference…

Can we improve?

- *Include track pos/dir*
- *Include charged hit energies*
- *Improve architecture*

Observations and next steps discussion

Many pitfalls are met along the way

- ECAL/HCAL gap in the middle of the detector
- Truth definition (crucial for PF studies!)
- Pandora cluster splitting at low energies
- Absolute vs relative loss
- Good performance mainly due to outliers
- Back-scattering DD4hep bug
- Generator Ghost particles DD4hep bug
- LCIO to Edm4hep migration with subset collections
- And yet many more to come…

Strong communication ensures
we build on past work
instead of revisiting old issues.

Next steps

- Include track directions/position
- Include charge hits energies
- Avoid using Pandora information (charged/neutral)
- And/or advance to PF studies
  (combine efforts with Uli, Dolores, Jan et al)

A call for discussion!
What is the most promising way forward?
What should we focus on? (combine efforts?)
What did we do great?
What did we do bad?
What can we improve?

Conclusions

- We advanced software compensation studies from HCAL prototype to full detector simulation with physics-like events.
- This is a major step toward a realistic evaluation of performance and it led us to even more ambitious goal – study of particle flow reconstruction.
- Our current results show Pandora-like performance, but still not yet optimal indicating
  a big open space for developments, improvements, and optimization.
- At this stage a conclusion on timing is too early to draw. It requires further development of the NN models towards the optimal performance.

Back Up

Back-up: list of all input features for JER with DGCNN study

Position hit features	x, y, z,	*Cartesian detector coordinates*
	rho, r, phi, theta,	*Spherical detector coordinates*
	d_long, d_perp,	*Distance from hit to the shower CoG*
	is_charged, is_neutral, is_undefined,	*Charged of Pandora PFO*
	is_ecal, is_hcal, is_yoke, is_lcal, is_lhcal, is_bcal,	*ILD subdetectors specific*
	is_barrel, is_endcap, is_ring,	
	layer,	
Energy hit features	e,	*Hit's energy (note: for hits associated to tracks, ignore actual hit's energy and split track's energy evenly among all Pandora cluster hits. To be improved.)*
	e_frac	*Hit energy fraction from the total energy of all hits within Pandora cluster*
Time hit features	t,	*Absolute hit time*
	dt,	*Time relative to the earliest time in the cluster*
	t_minus_c	*Absolte hit time – r/c* *(For now perfect time resolution assumed)*