Optimizing the Higgs self-coupling measurement at ILC with Machine Learning

ILD Concept Group Meeting | 2024/01/16

Bryan Bliewert[1,2], Jenny List[1], Julie Munch Torndal[1,3], Junping Tian[4]

[1] DESY Hamburg
[2] Technische Universität München
[3] Universität Hamburg
[4] University of Tokio

HELMHOLTZ

Outline

- Intro: The Higgs self-coupling and the ZHH analysis
- Part I: Jet clustering as a leading source of error
  - Towards a GNN-based jet-clustering algorithm
- Part II: The Matrix-Element-Method (MEM) at ILC
  - Accelerating the MEM using neural importance sampling (NIS)
- Conclusion

Intro: The Higgs self-coupling and the ZHH analysis

Higgs self-coupling λ in the Standard Model (SM)

- Higgs-sector in SM after SSB: only one free parameter

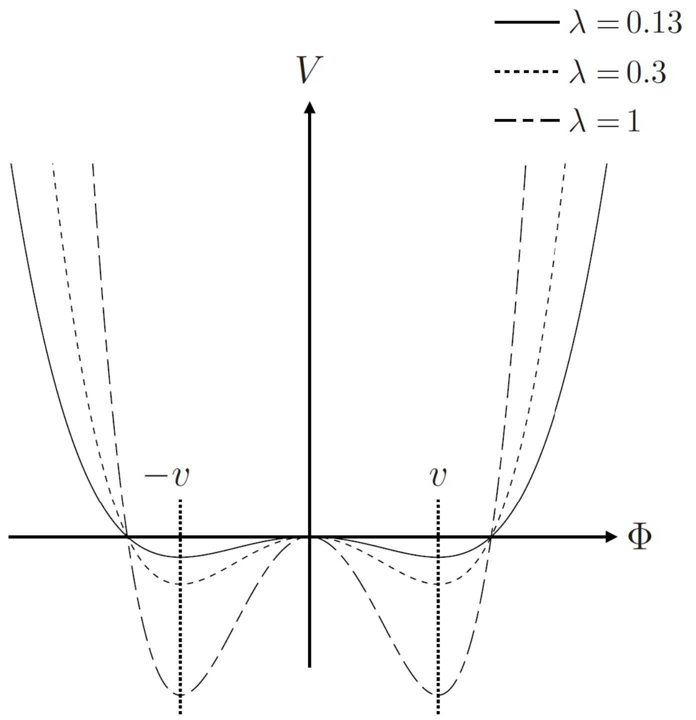
$$V(h) = \frac{1}{2} m_H^2 h^2 + \lambda_3 v h^3 + \frac{1}{4} \lambda_4 h^4$$

$$\frac{m_H^2}{2v^2} = \lambda_3^{SM} = \lambda_4^{SM}$$

- self-coupling λ defines shape of Higgs potential

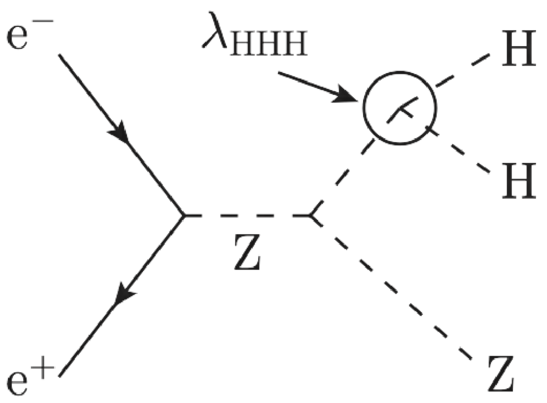
$$\lambda + \delta\lambda = \frac{m_H^2}{2v^2} \pm \frac{\delta m_H}{v^2} m_H \approx 0.13 \pm 10^{-3}$$

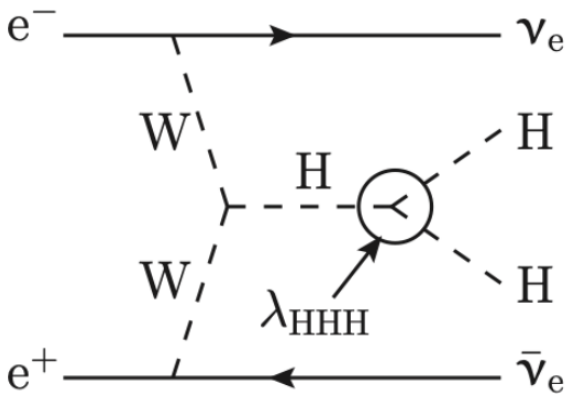
- sensitive to BSM physics by loop corrections

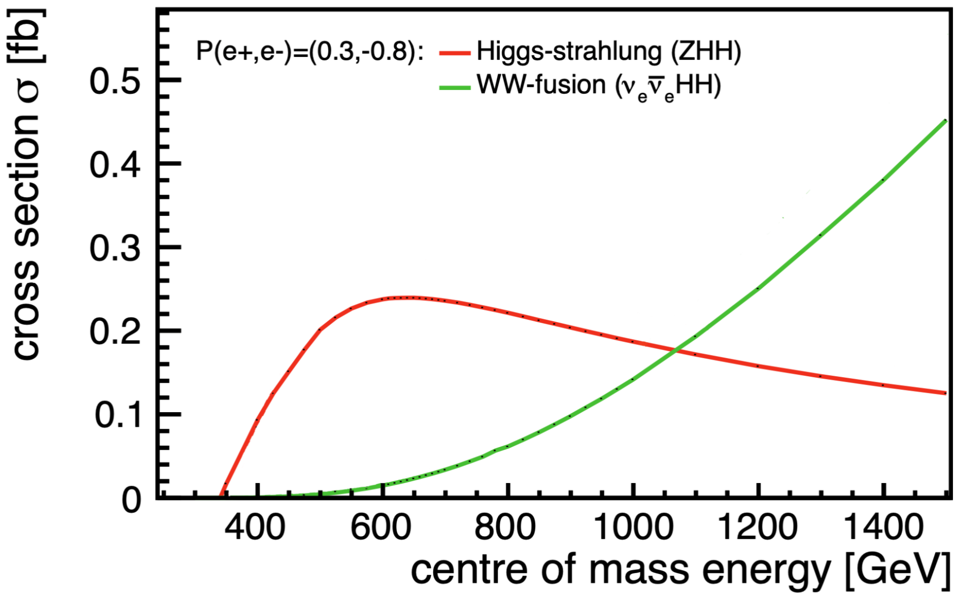


Measuring the Higgs self-coupling at e+e- colliders

- *direct access* to λ possible through double-Higgs production
  - Di-Higgs strahlung (dominant < 1 TeV)
  - vector boson fusion (dominant > 1 TeV)



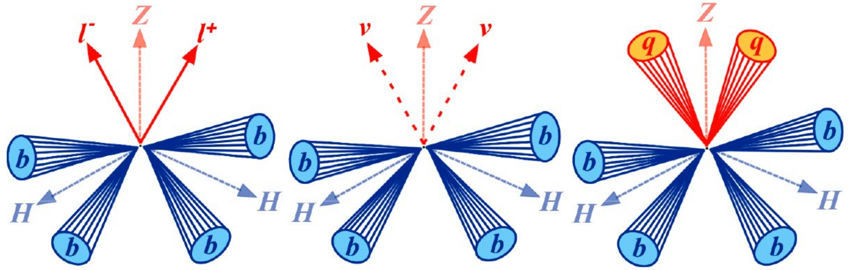




Ph.D. Thesis Dürig, DESY [2016]

The ZHH analysis: Status

- extensive projections at ILC with proposed $\sqrt{s} = 500\ \mathrm{GeV}$ (DESY-Thesis-2016-027)



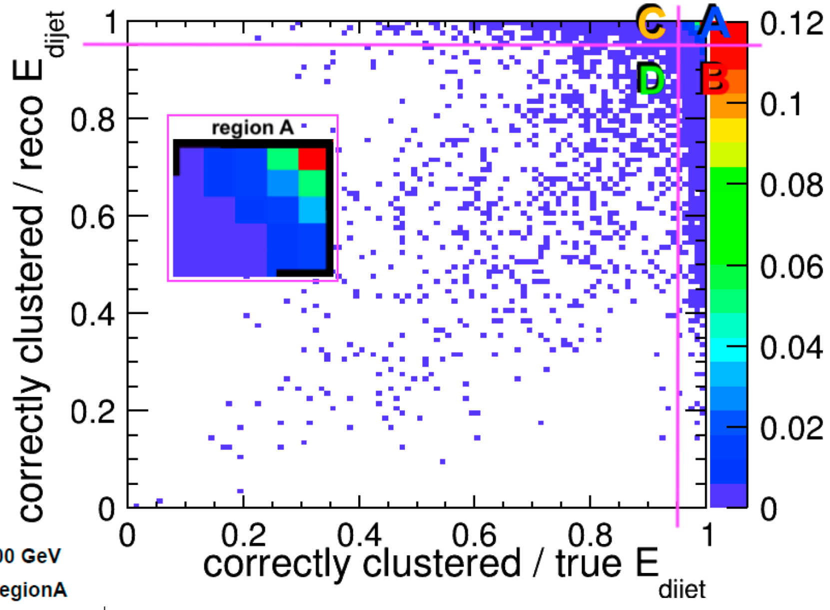
- based on ILD detector concept (DBD2013, IDR2020)
- precision reach after running $4\mathrm{ab}^{-1}$ at $500\ \mathrm{GeV}$ ($\mathrm{HH} \rightarrow \mathrm{b\bar{b}b\bar{b}}$ & $\mathrm{HH} \rightarrow \mathrm{b\bar{b}W^{\pm}W^{\mp}}$)
  - at least **30% precision** on λ **for any value of λ**
  - **20%** precision on λ for $\lambda_{true} = \lambda_{SM}$ expected with state-of-the-art reconstruction tools (kinematic fitting for jet pairing & **hypothesis testing**, flavor tagging)
  - **10%** precision on λ for $\lambda_{true} = \lambda_{SM}$ when combined with additional scenario at 1 TeV

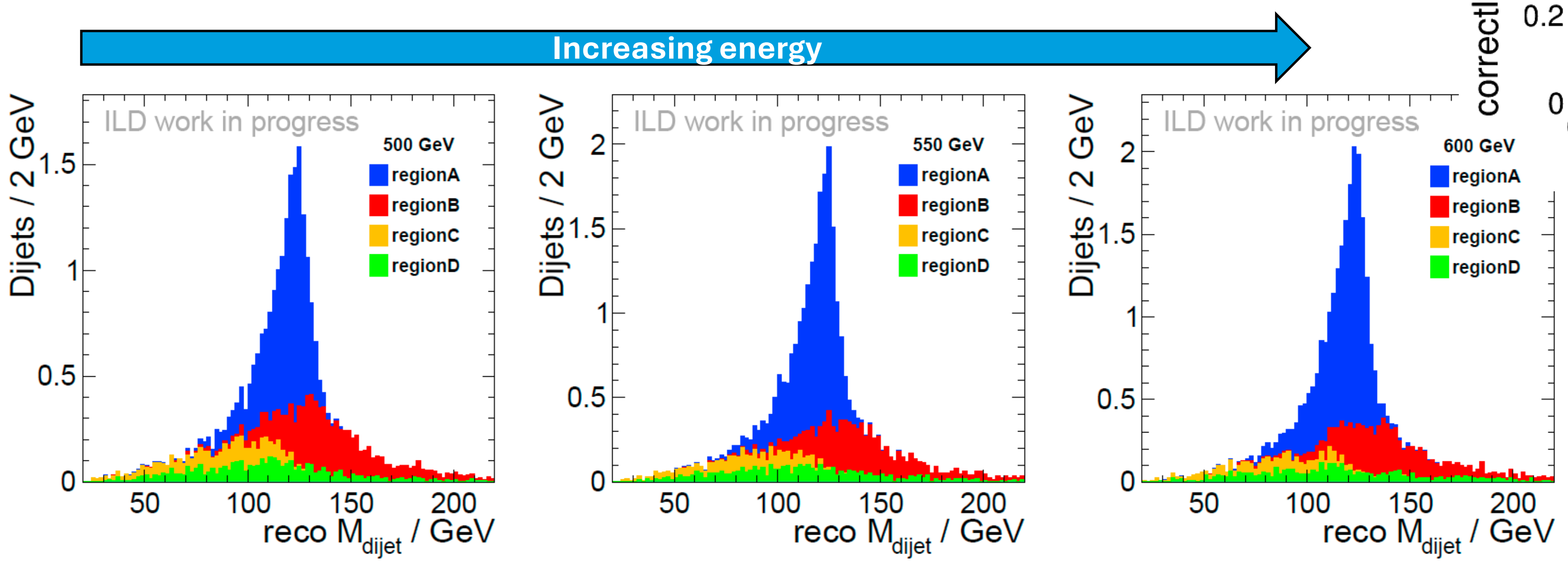
Part I: Jet clustering as a leading source of error

Misclustering of jets with Durham(LCFIPlus)

- ambigious clustering of jets and jet-matching
  → misclustering
- quantify with misclustering categories:
  - overlap fraction between true and reco energy

J. Torndal, J. List [2023]





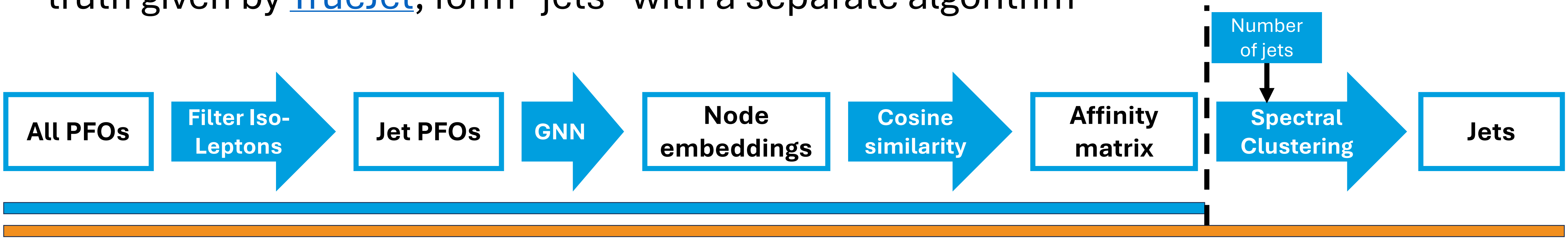
fraction of dijets in category A:

$\sqrt{s}$ [GeV]	A [%]
500	45.5
550	50.5
600	53.7

- WIP: investigate **Graph-Neural-Networks (GNNs)** for improving jet-clustering (next slide)

Supervised jet clustering with GNNs

- **idea**: train a GNN to calculate whether PFOs belong into the same jet with ground-truth given by TrueJet; form “jets“ with a separate algorithm

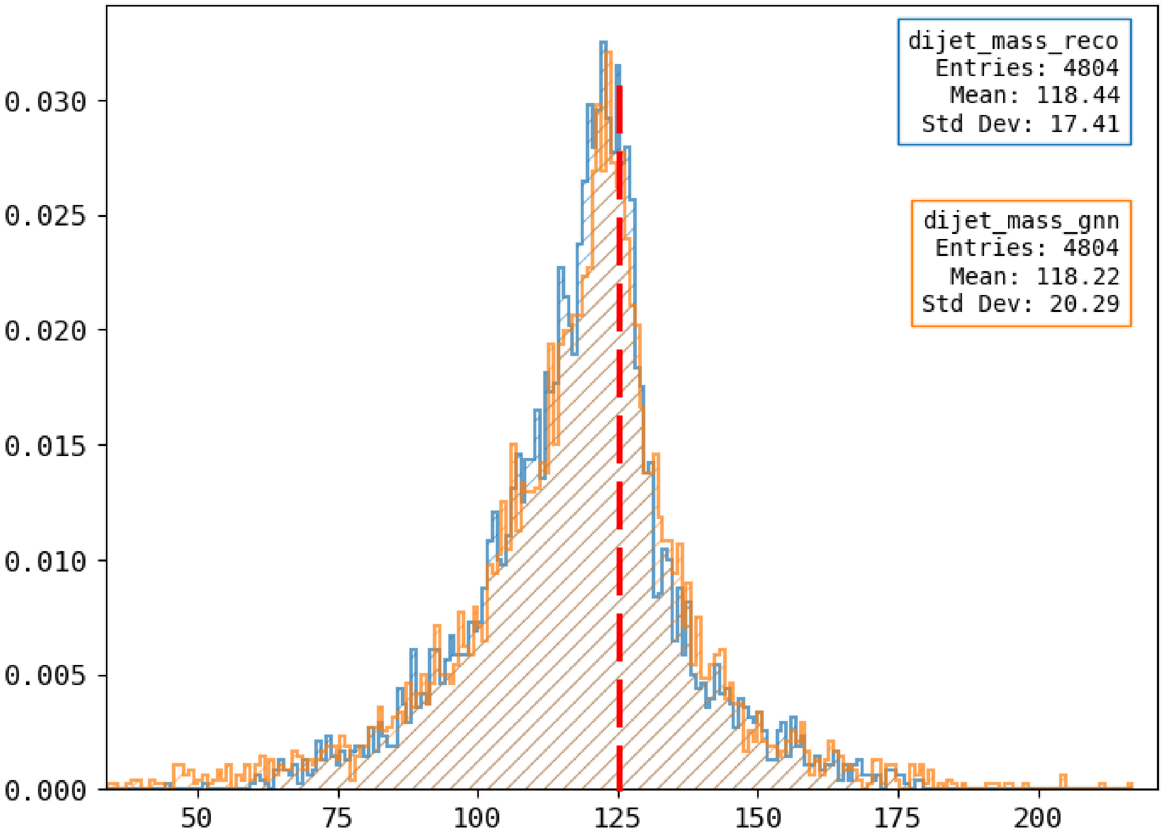


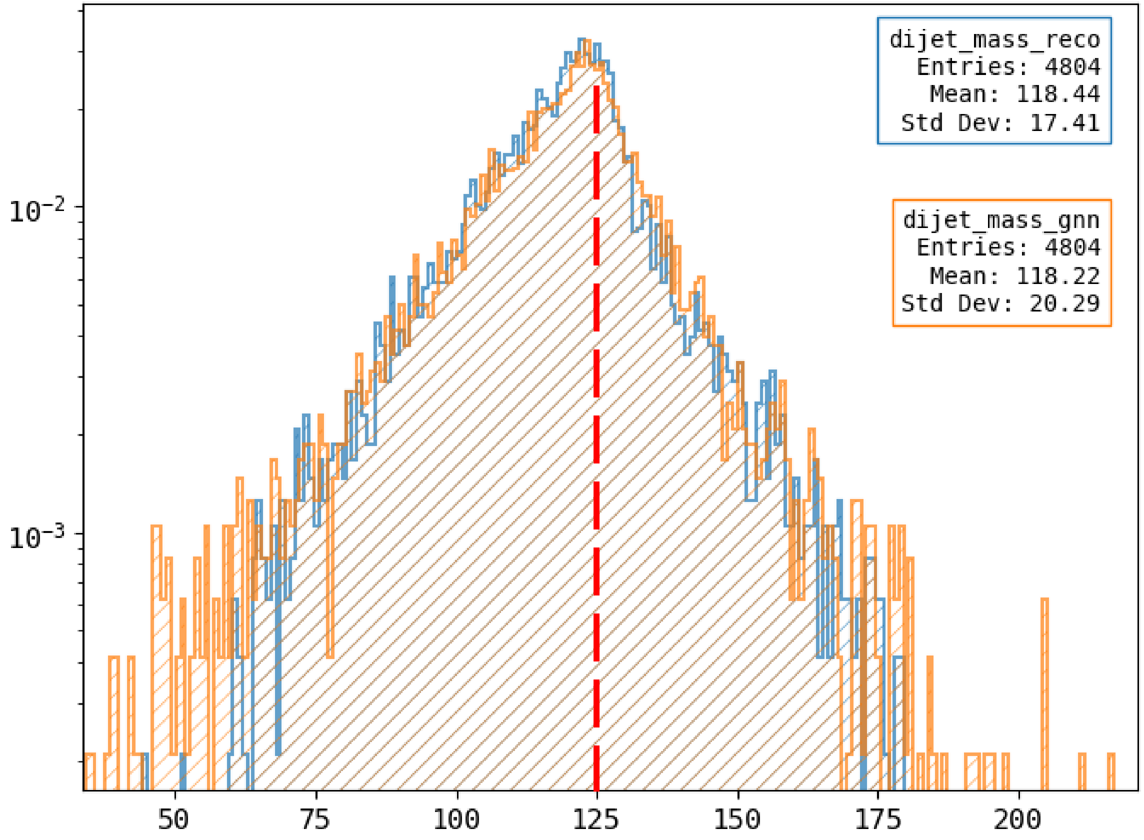
- **design decisions:**
  - **permutation invariance built in**: for each PFO-PFO-pair, calculate score using cosine similarity
  - however, **no IR/C-safety** enforced in model
  - training and hyperparam. optimization in Python, **inference possible in Marlin (JetConvProcessor)**
  - backbone: transformer based architecture

Supervised jet clustering with GNNs

Results: Dijet-mass reconstruction

- training on ca. 100.000 full-simulation ZHH-events (400k jets)



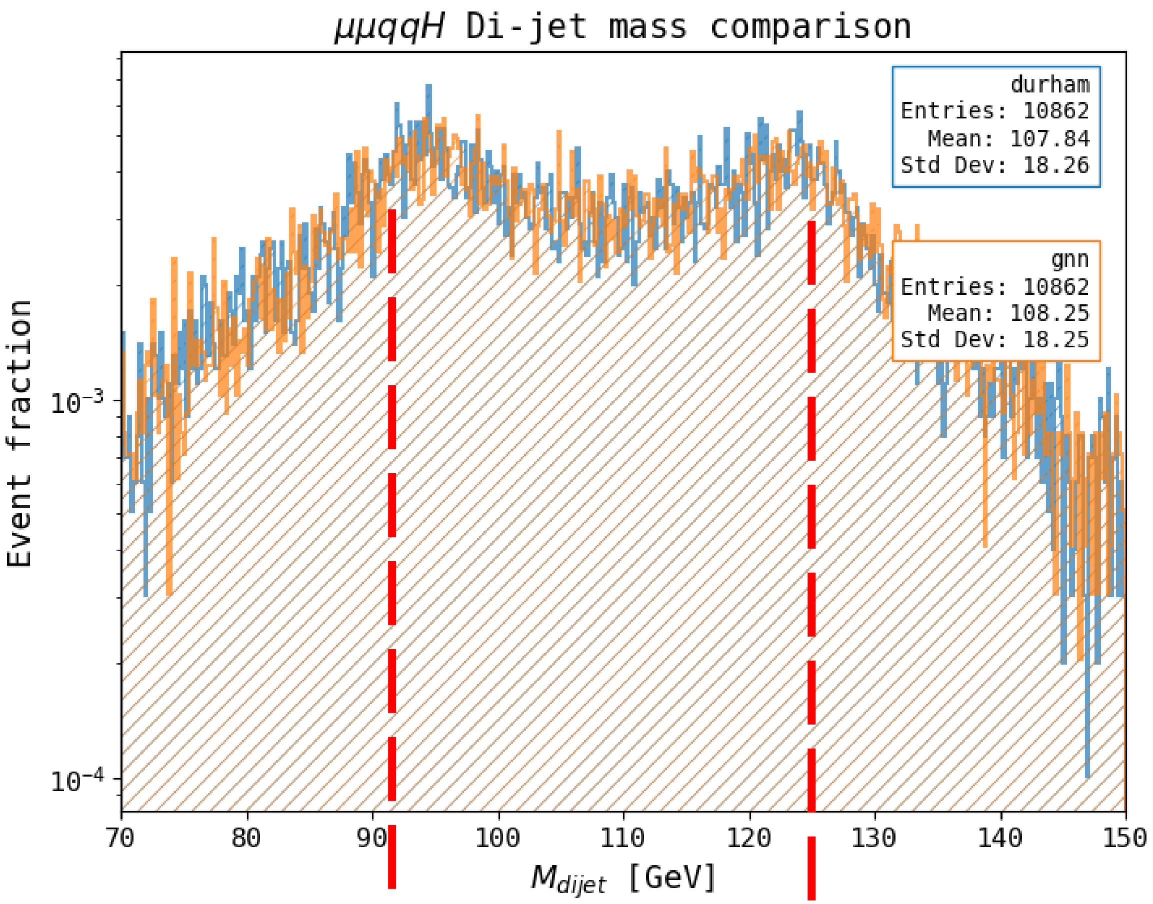


Supervised jet clustering with GNNs

Results: Dijet-mass reconstruction

- training on ca. 100.000 full-simulation ZHH-events (400k jets)

Clustering jets from ZZH (never seen by model)



Part II: The Matrix Element Method (MEM) at ILC

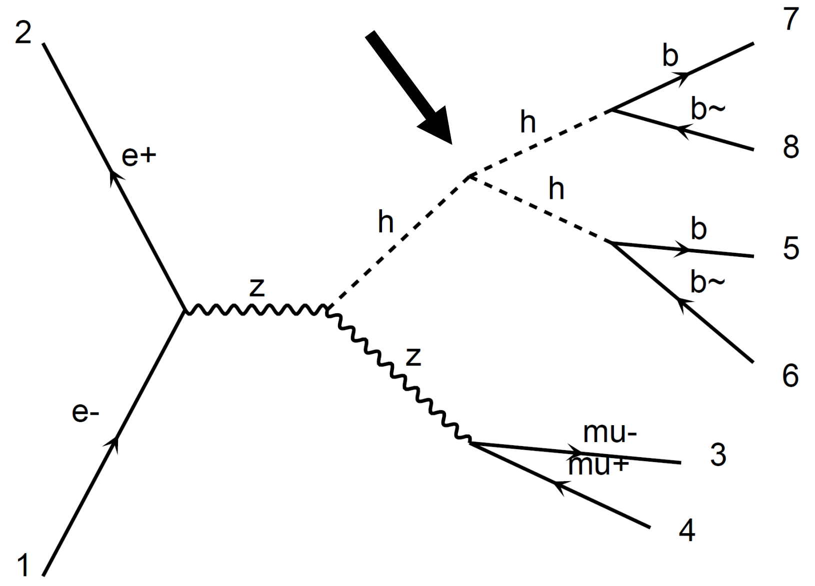
The Matrix Element Method (MEM)

- method for calculating event-likelihoods, use cases:
  - process discrimination (Neyman-Pearsson lemma)
  - parameter estimation (theory parameters; eventually λ)
- goal here: separate ZHH vs. ZZH $\rightarrow \mu^-\mu^+ b\bar{b}b\bar{b}$
- for each event $\boldsymbol{y}$ and process i (ZHH, ZZH), solve

$$P_i\,(\boldsymbol{y}\mid\boldsymbol{a}) = \frac{1}{\sigma_i(\boldsymbol{a})\cdot A_i(\boldsymbol{a})}\int |M_i(\boldsymbol{x},\boldsymbol{a})|^2 W_i(\boldsymbol{y}\mid\boldsymbol{x})\epsilon_i(\boldsymbol{x})d\boldsymbol{\Phi}_n(\boldsymbol{x})$$

  - $M_i(\boldsymbol{x},\boldsymbol{a})$ LO matrix element (HELAS-based Physsim, J. Tian)
  - $W_i(\boldsymbol{y}|\boldsymbol{x})$ detector transfer functions: PDF for measuring $\boldsymbol{y}$ given $\boldsymbol{x}$; fitted from ILD full-simulation
  - phase space integration done using VEGAS+

λ_{HHH}



$\boldsymbol{a}$: theory parameters; e.g. λ_{HHH}
$A_i(\boldsymbol{a})$: signal acceptance
$\epsilon_i(\boldsymbol{x})$: detector efficiency

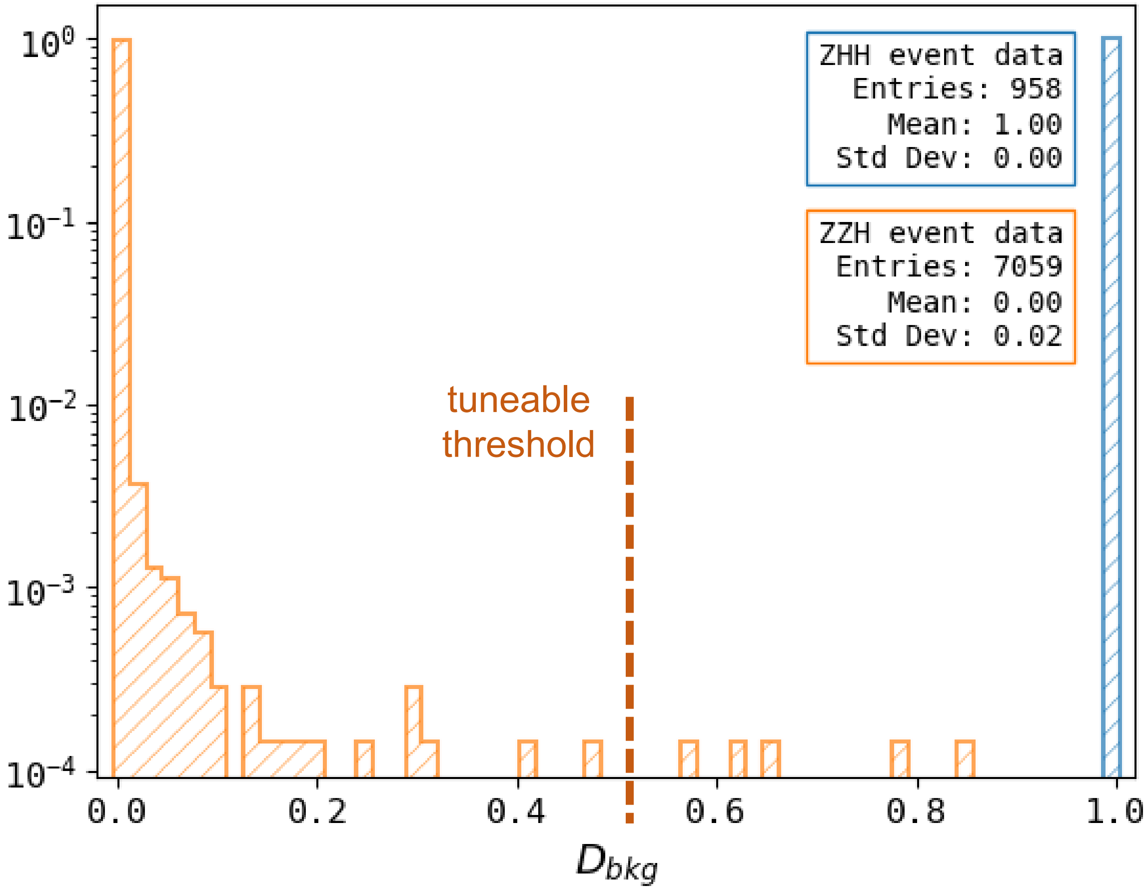
Hypothesis testing with the MEM

MC truth + Matrix Elements (ME) only

- use case: generator-level check
  - calculate discriminator just from $M_i(y_{truth})$ and σ_i
  - no transfer function
- perfect separation, as expected

4-vectors / MEM type	MC truth	Reco
ME only		
ME+DTF	-	

D_{bkg} (MC truth @ ME only)

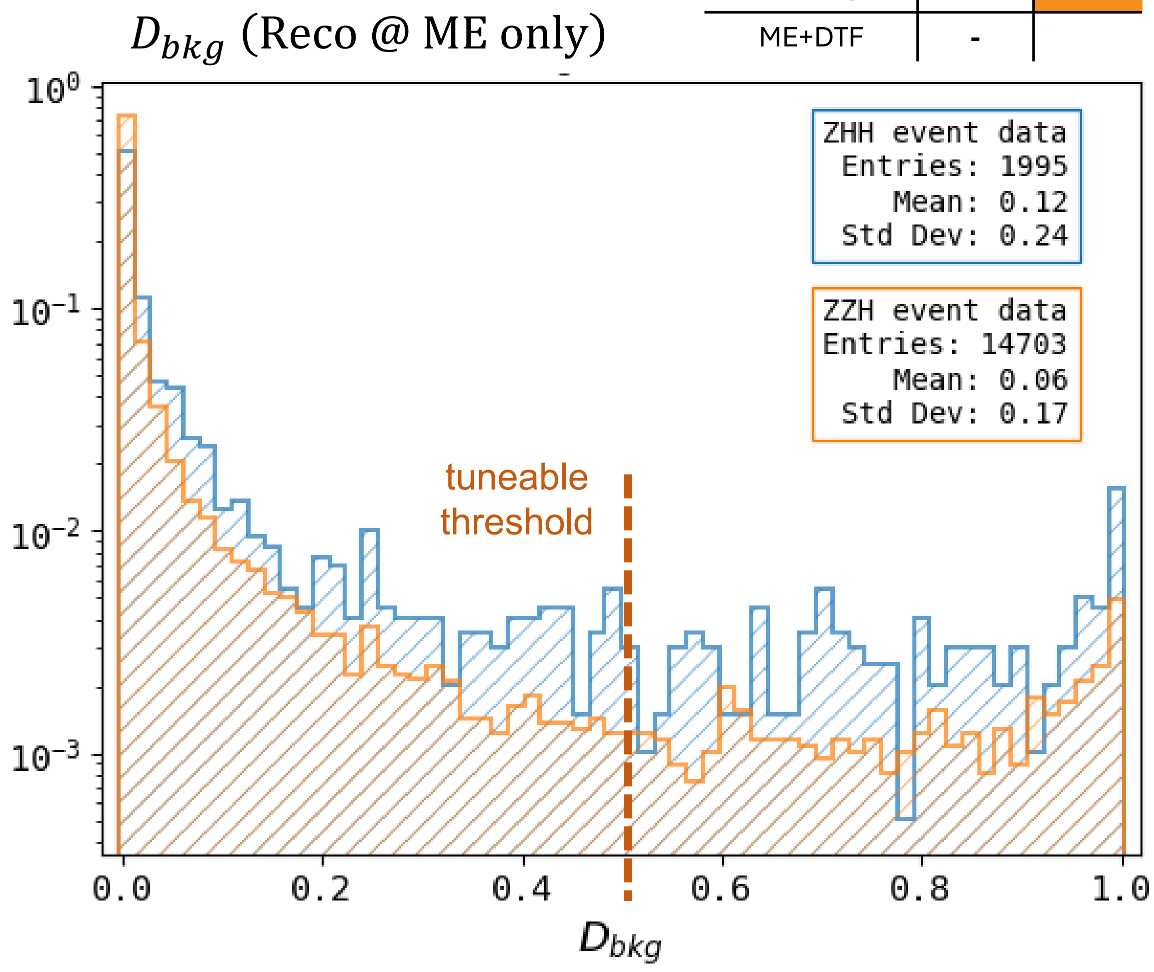


Hypothesis testing with the MEM

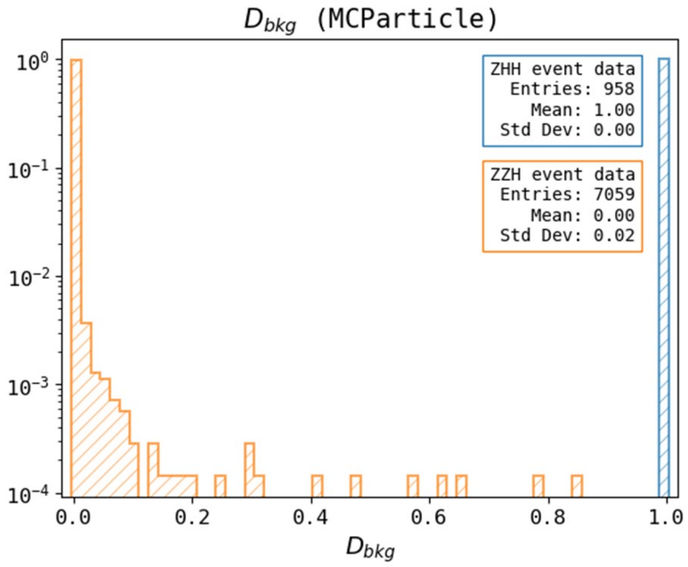
4-vectors / MEM type	MC truth	Reco
ME only		
ME+DTF	-	

Reconstructed data + ME only

- smearing and loss of separation power
- need to account for
  - parton showering, hadronization
  - detector effects, misclustering etc.



For comparison:
ME only with MC data

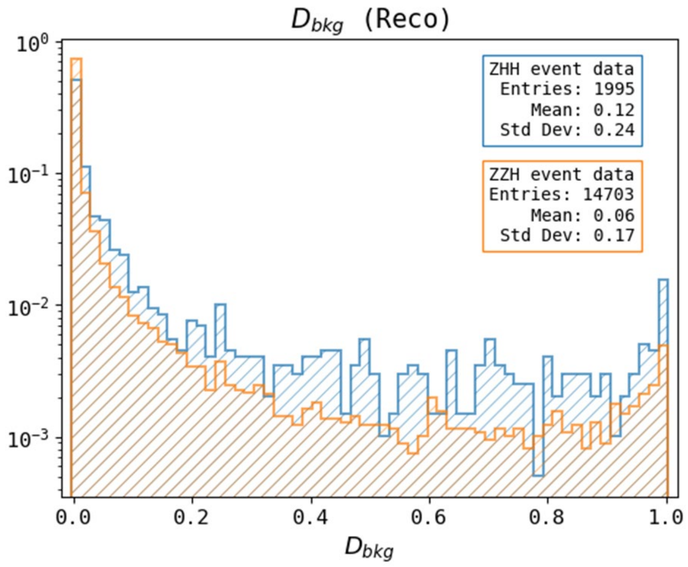


Hypothesis testing with the MEM: Results

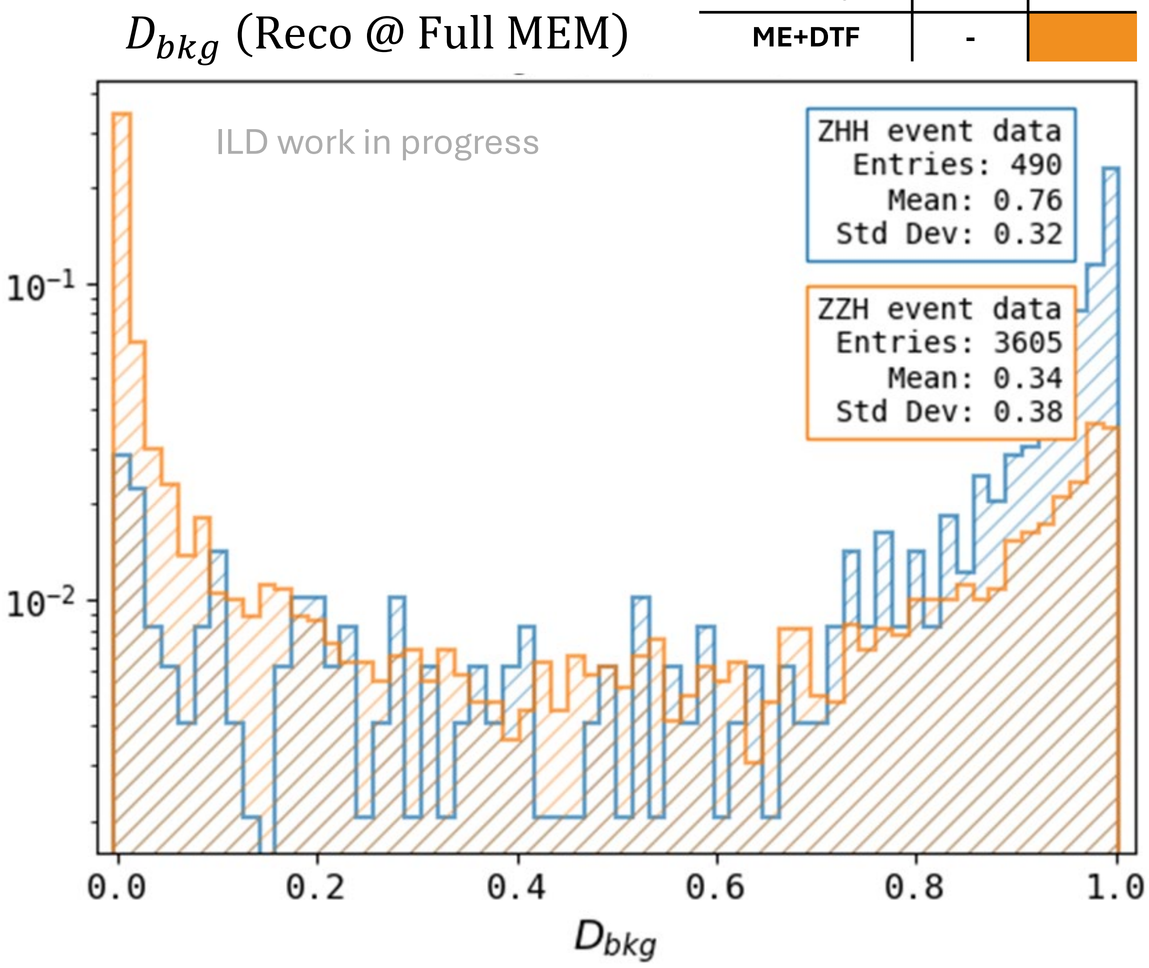
4-vectors / MEM type	MC truth	Reco
ME only		
ME+DTF	-	

Reconstructed data + Full-MEM

- better separation by accounting for detector effects
- possibly: MEM output as input to other MVA



For comparison:
ME only with Reco data

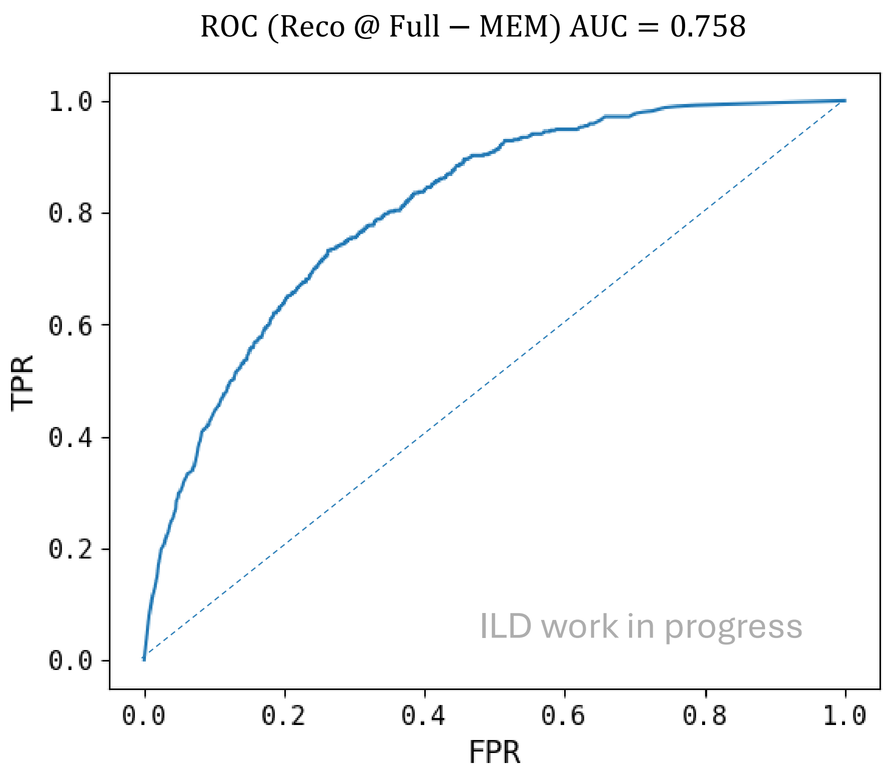


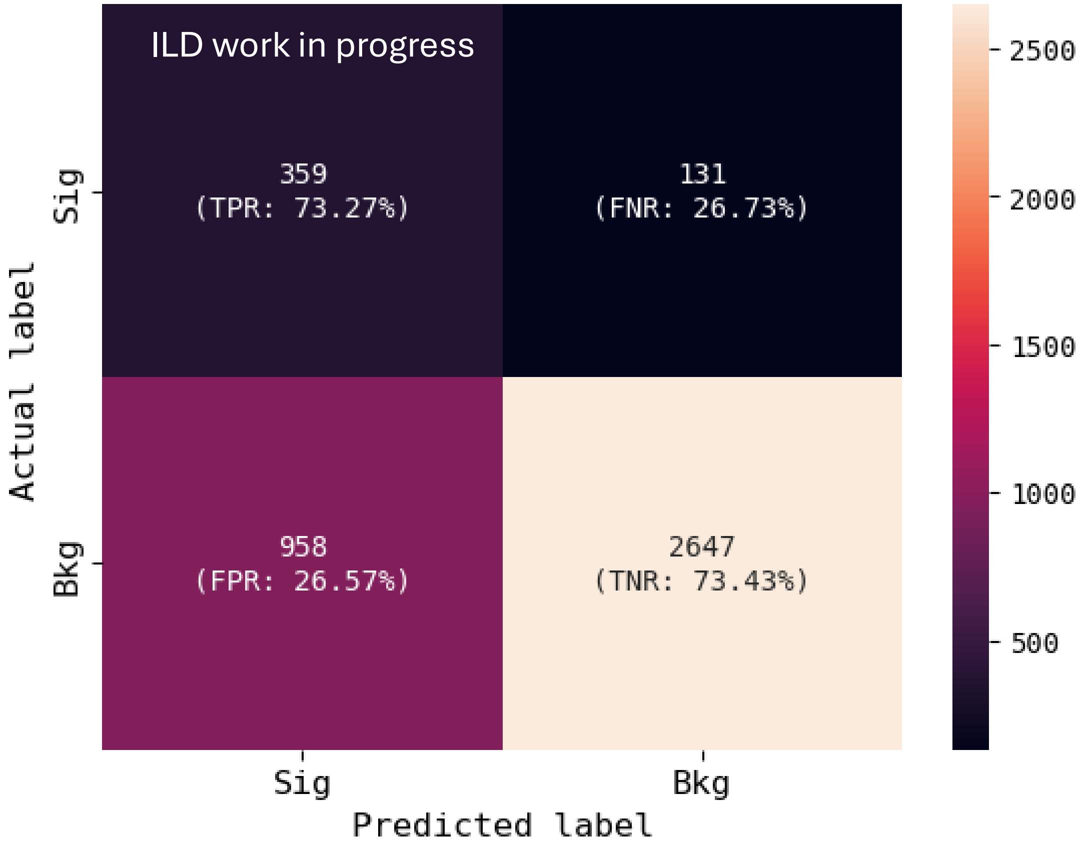
Hypothesis testing with the MEM: Results

MEM type \ 4-vectors	MC truth	Reco
ME only		
ME+DTF	-	

Reconstructed data + Full-MEM

- better separation by accounting for detector effects
- possibly: MEM output as input to other MVA





MEM: Limitations and solutions

- accelerating computation (with VEGAS: ~3min per integration for one parton-jet combo)
  - **Problem:** VEGAS highly-inefficient for non-factorizing integrands
  - **Idea**: use neural importance sampling (see work by Plehn et al)
  - **Implementation**:
    - normalizing flow (NFs) learns regions of integrand yielding highest contributions
    - NFs allow learning probability distributions by distorting a „latent distribution“ by subsequent transformations (jacobian etc. remain tractable)
  - **Status: WIP**
- integrating over full parton-jet-combinatorics
- incorporating b-tagging information
- accounting for detector acceptance

Conclusion

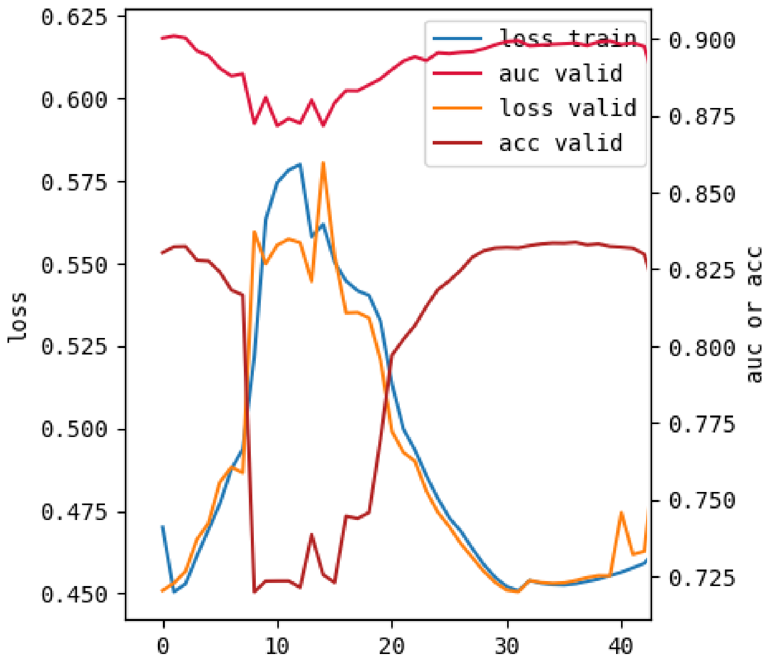
- Matrix Element Method implemented for S/B-separation in Di-Higgs analysis in $ZHH \rightarrow l\bar{l}b\bar{b}b\bar{b}$
  - promising first results, needs final adjustments and higher speed
- ML-based approaches for increasing speed and performance
  - Graph Neural Networks (GNNs): towards a better jet clustering
    - currently sub-Durham performance,
    - **BUT many possibilities for improvement** (end-to-end model, binary loss weighted by PFO energies, IRC safety…)
    - Marlin processor available for analysis
  - invertible neural network (INNs) for more efficient importance sampling and faster MEM computation

Thank you!

Backup

Supervised jet clustering with GNNs

- **idea**: using ML, train a GNN to calculate whether PFOs belong into the same jets based on truth-information (TrueJet) and form clusters with a separate algorithm
- **training** (for each event):
  - filter out isolated PFOs (IsolatedLeptons)
  - calculate node embeddings with GNN
  - for each PFO-PFO-pair, calculate an *edge-score* using cosine similarity (→ **permutation invariance**)
- **inference**: require n jets
  - calculate affinity matrix (i.e. all edge scores)
  - **spectral clustering** estimates optimal decision boundaries and groups the PFOs into n clusters



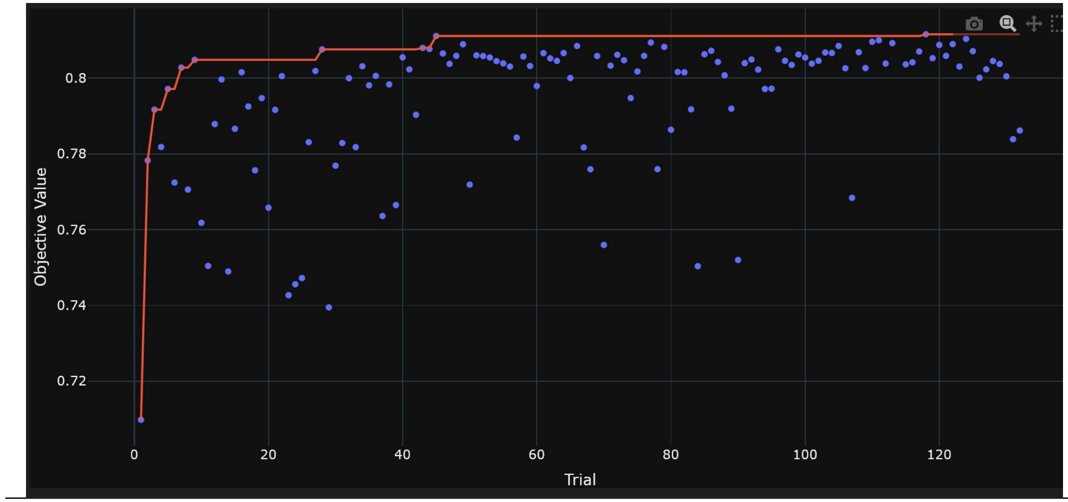
Training loss and performance metrics

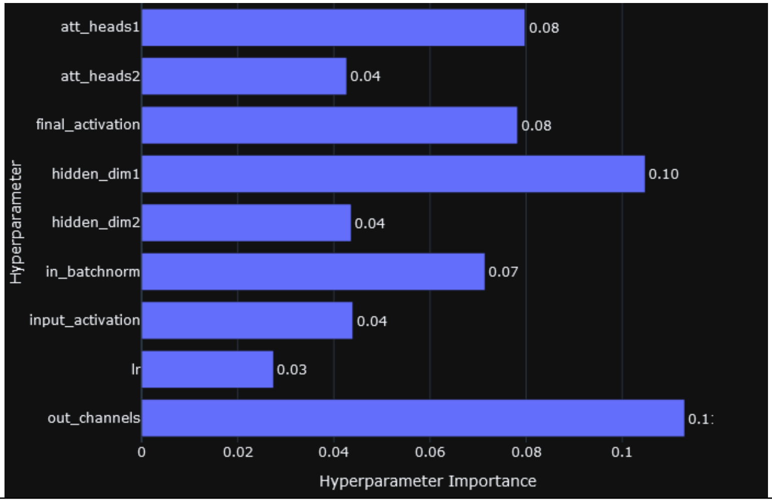
GNN hyperparameter optimization

9 GNN parameters automatically optimized (using 5% of all events, i.e. ca. 5000)

```
trainer_args = {
    'hidden_dim1': trial.suggest_int("hidden_dim1", 8, 256, log=True),
    'hidden_dim2': trial.suggest_int("hidden_dim2", 8, 384, log=True),
    'out_channels': trial.suggest_int("out_channels", 8, 128, log=True),
    'in_batchnorm': trial.suggest_int("in_batchnorm", 0, 1),
    'att_heads1': trial.suggest_int("att_heads1", 1, 8),
    'att_heads2': trial.suggest_int("att_heads2", 1, 8),
    'lr': trial.suggest_float("lr", 0.0001, 0.004, log=True),
    'input_activation': trial.suggest_categorical("input_activation", ['LeakyReLU', 'ReLU', 'Tanh', 'ELU', 'PReLU']),
    'final_activation': trial.suggest_categorical("final_activation", ['LeakyReLU', 'ReLU', 'Tanh', 'ELU', 'PReLU']),
}
```

Result: 5%-increase of model accuracy to 84%, after full training





MEM Monte-Carlo integration

$$P_i\,(\boldsymbol{y}\,|\,\boldsymbol{a}) = \frac{1}{\sigma_i(\boldsymbol{a})\cdot A_i(\boldsymbol{a})}\int W_i(\boldsymbol{y}\,|\,\boldsymbol{x},\boldsymbol{a})|M_i(\boldsymbol{x},\boldsymbol{a})|^2 T_i(\boldsymbol{x},\boldsymbol{a})d\boldsymbol{\Phi}_n$$

$$d\boldsymbol{\Phi}_n = \prod_i^{\mu^-,\mu^+,b_1,\overline{b_1},b_2,\overline{b_2}} \frac{d^3\boldsymbol{p_i}}{(2\pi)^3 2E_i}$$

- leptons well measured ➔ no integration for μ^-, μ^+
- conservation of four momentum and narrow-width-approximation ➔ reduction of integration to 7 dimensions
- integration variables: $\Theta_{b1}, \phi_{b1}, \rho_{b1}, \theta_{b1b}, \phi_{b1b}, \rho_{b2}, \Theta_{b2}$
- with VEGAS+ and integrand in C++, computation time 1-2 minutes per process (including setup of integration grid)
- „hit-or-miss“ MC (unphysical integration variables → 0)

itn	integral	wgt average	chi2/dof	Q
1	4.2(3.6)e-09	4.2(3.6)e-09	0.00	1.00
2	6.7(2.7)e-10	6.9(2.7)e-10	0.94	0.33
3	6.0(2.1)e-10	6.4(1.7)e-10	0.50	0.60
4	2.69(55)e-10	3.05(52)e-10	1.81	0.14
5	3.49(58)e-10	3.24(39)e-10	1.44	0.22
6	2.96(43)e-10	3.12(29)e-10	1.20	0.31
7	5.0(1.2)e-10	3.23(28)e-10	1.42	0.20
8	4.78(94)e-10	3.35(27)e-10	1.58	0.14
9	8.6(2.2)e-10	3.43(27)e-10	2.11	0.03
10	5.9(1.8)e-10	3.48(26)e-10	2.07	0.03

result = 3.48(26)e-10   Q = 0.03

itn	integral	wgt average	chi2/dof	Q
1	1.58(18)e-09	1.58(18)e-09	0.00	1.00
2	1.68(19)e-09	1.63(13)e-09	0.13	0.72
3	1.94(19)e-09	1.72(11)e-09	0.96	0.38
4	1.91(13)e-09	1.800(82)e-09	1.04	0.37
5	1.98(27)e-09	1.815(79)e-09	0.88	0.48
6	2.73(99)e-09	1.821(78)e-09	0.88	0.50
7	1.78(10)e-09	1.807(62)e-09	0.74	0.61
8	2.03(17)e-09	1.834(59)e-09	0.86	0.54
9	1.72(13)e-09	1.816(54)e-09	0.82	0.58
10	1.813(83)e-09	1.815(45)e-09	0.73	0.68

result = 1.815(45)e-09   Q = 0.68

MEM results for example ZHH (top) and ZZH (bottom) event

MEM detector transfer functions

- PDF for energies/angles between reconstructed and parton-level particles
- „conventional approach“: fitting transfer functions explicitly
- separate transfer functions possible for signal/background hypothesis