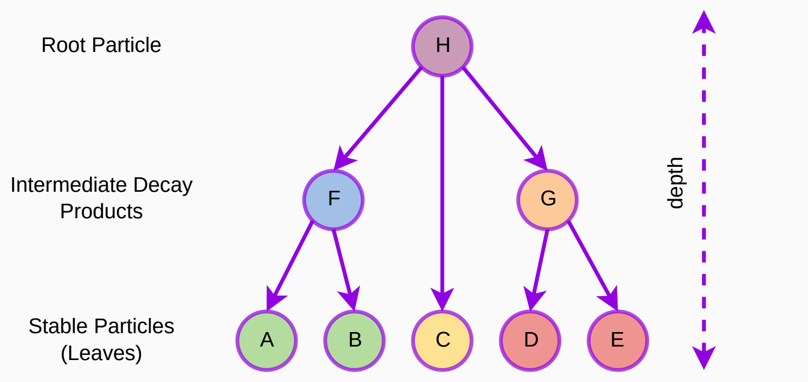
ACAT 2022

Reconstructing Particle Decay Trees with Quantum Graph Neural Networks in High Energy Physics

Melvin Strobl, Eileen Kühn, Max Fischer, Achim Streit

Particle Decay Trees



An exemplary decay tree with two intermediate decay products and five stable particles which originates from a root particle.
Each decay is assumed to result in at least two children and the number of detected particles is exact.
Furthermore no measurement errors are assumed.

Input: Feature matrix with the four-momentum E, p_x, p_y and p_z for each Final State Particle (FSP) $[0, \ldots, l, \ldots, L-1]$.

Target: Lowest Common Ancestor Generation (LCAG) [1] matrix. Each entry represents the number of generations to go back in the decay tree until a common node is present

Feature Matrix

	E	p_x	p_y	p_z
A	E^A	p_x^A	p_y^A	p_z^A
B	E^B	p_x^B	p_y^B	p_z^B
C	E^C	p_x^C	p_y^C	p_z^C
D	E^D	p_x^D	p_y^D	p_z^D
E	E^E	p_x^E	p_y^E	p_z^E

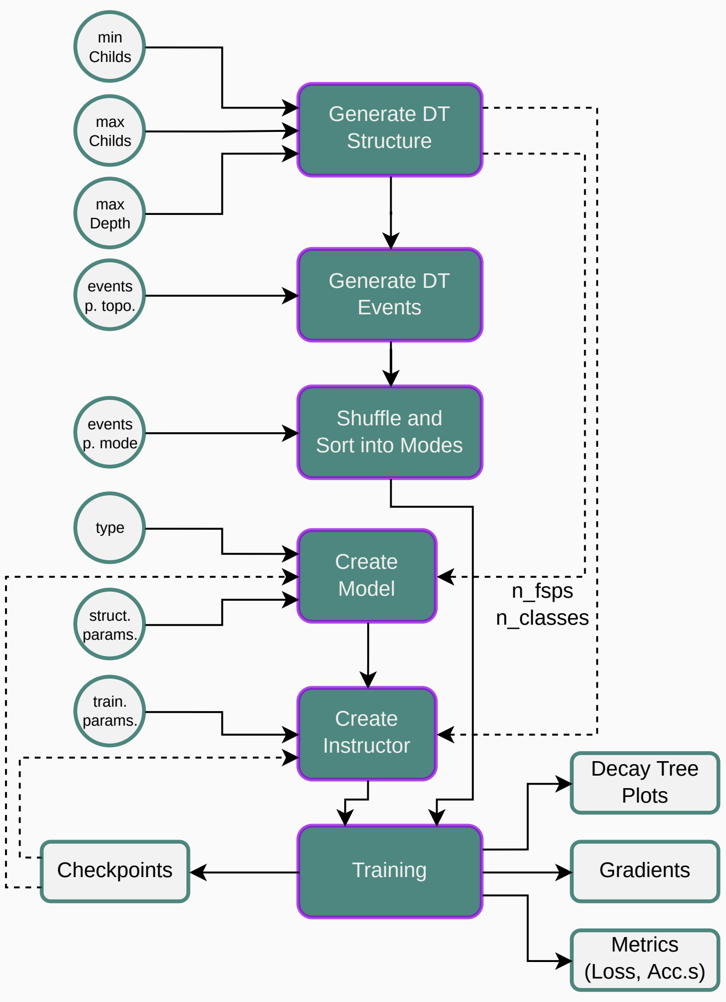
Lowest Common Ancestor Generation Matrix

	A	B	C	D	E
A	0	1	2	2	2
B	1	0	2	2	2
C	2	2	0	2	2
D	2	2	2	0	1
E	2	2	2	1	0

The synthetic dataset consists of decay events generated using the phasespace library [6].
Each event is represented by the four-momenta of the FSPs (input) and the structural properties of the decay tree (label).
The complexity of the dataset is controlled by the following parameters as introduced in [1]:

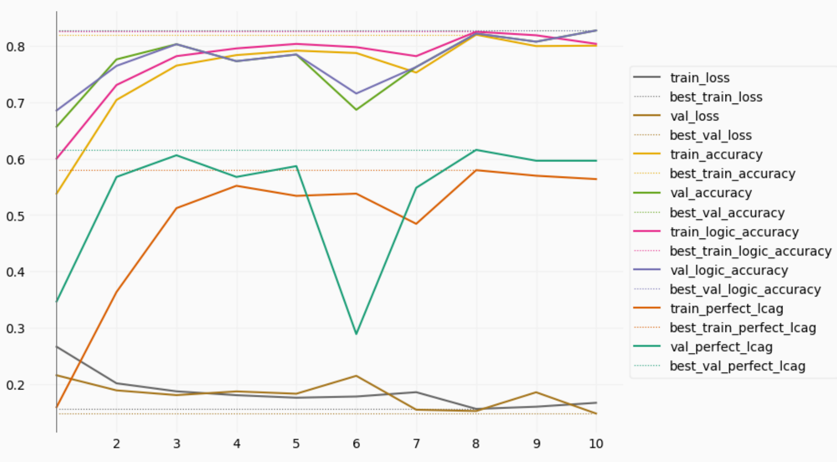
- MAX_CHILDREN: Maximum number of childs for each node
- MIN_CHILDREN: Minimum number of childs for each node
- MAX_DEPTH: Maximum number of generations within the tree
- N_TOPOLOGIES: Number of different topologies (constraint by the above parameters)
- N_EVENTS_PER_TOP_[mode]: Number of events generated for each topology and for each mode (training, validation and test)

The total number of FSPs and the number of available classes are being used to create the model.
The latter is subject to structural parameters including classical ones (e.g. feedforward dimension and dropout rate) but also quantum exclusive parameters (e.g. data reupload and measurement interpretation).



Loss: Cross-Entropy with a mask for "-1" classes representing the diagonal entries and invalid leaves.
Accuracy: Three different accuracy metrics are applied.
1. Counting only perfect reconstructions of the LCAG matrix ("perfect_lcag_accuracy")
2. Counting true and false classifications of edges/ parent generations ("accuracy")
3. Same as 2. but with a processing where logical mistakes of the tree are corrected ("logic_accuracy")

Experiment Results



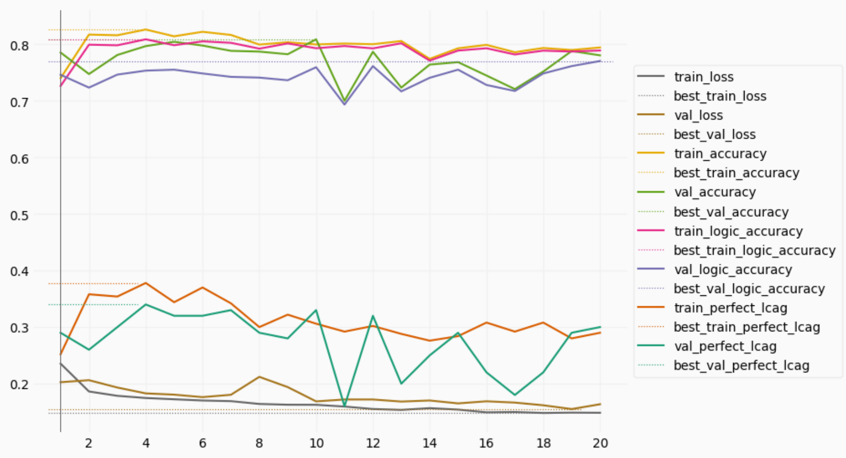
The model achieves a validation loss of 0.147 and 0.827 validation accuracy.
On the validation data, the perfect_lcag score is 0.596 and the logic_accuracy 0.827.
It was observed that the model is very sensitive to the learning rate.
The VQC with "Circuit 19" showed a slightly lower performance.
Data Reuploading improved the performance significantly.

Validation on real Quantum Devices

After training the model in simulation using classical devices, validation was carried out on a 7-qubit Noisy Intermediate Scale Quantum (NISQ) device (*ibm_perth*) from IBM. This experiment should verify if an application on real quantum devices is already feasible and should give a hint on the generalization capability of the trained model. The following metrics have been measured:
val_loss:0.235, val_accuracy:0.655, val_logic_accuracy:0.655, val_perfect_lcag:0.404.
As expected, the scores are much lower in comparison to the ideal simulated circuit, which is mainly caused by the inherent noise, but also due to the transpilation process, where the circuit is being translated into a smaller set of gates and adapted to the device specific topology.

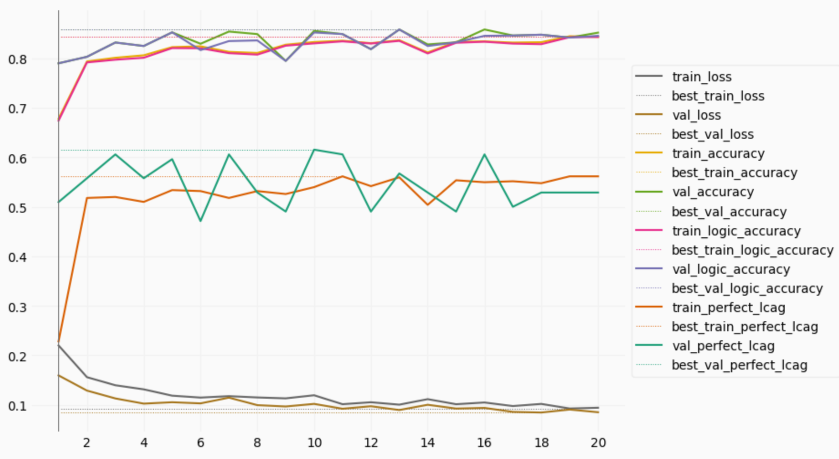
Comparison: QMLP + MLPs

In a different approach, the quantum circuit was evaluated with series of simple MLP layers instead of the GNN [1].
It was found that despite the competitive accuracy and loss, the generalization is not as good as in the GNN. Both approaches are set to have a similar number of parameters, while the QMLP has more parameters within the PQC.



Baseline Comparison

The approach presented in [1] and the results of the classical training serve as a baseline. The GNN is re-implemented and trained on the same amount of data as the hybrid approaches above. The model has 147395 parameters and is manually tuned on the reduced dataset.



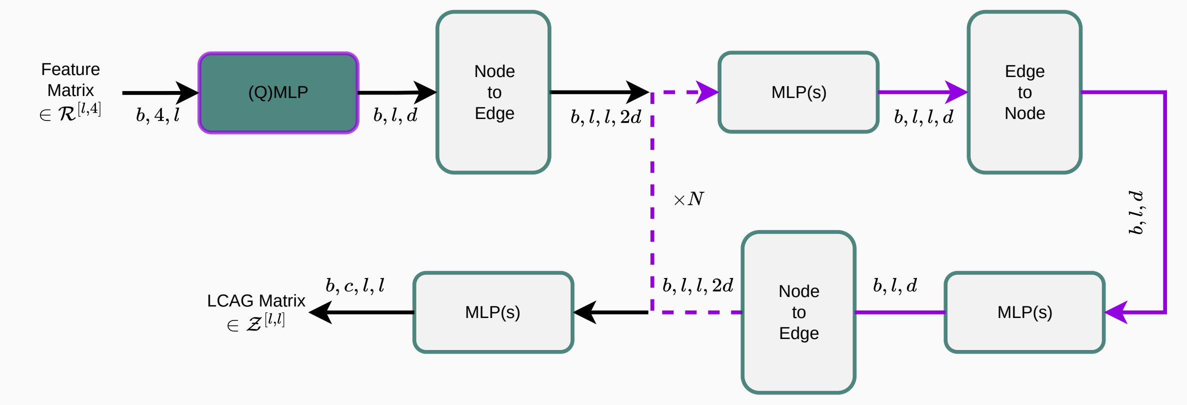
The slightly better validation metrics indicate a good generalization capability.
On the validation data, the perfect LCAG score is 0.529 and the logic accuracy 0.845.

On an extended dataset, a logic accuracy of 0.985 and a validation loss of 0.008 is achieved.

When the number of parameters of the GNN is reduced to a similar size as in the hybrid approach, the metrics drop to 0.809 accuracy, 0.796 logic accuracy, a perfect LCAG score of 0.385 and loss of 0.129 on the validation dataset

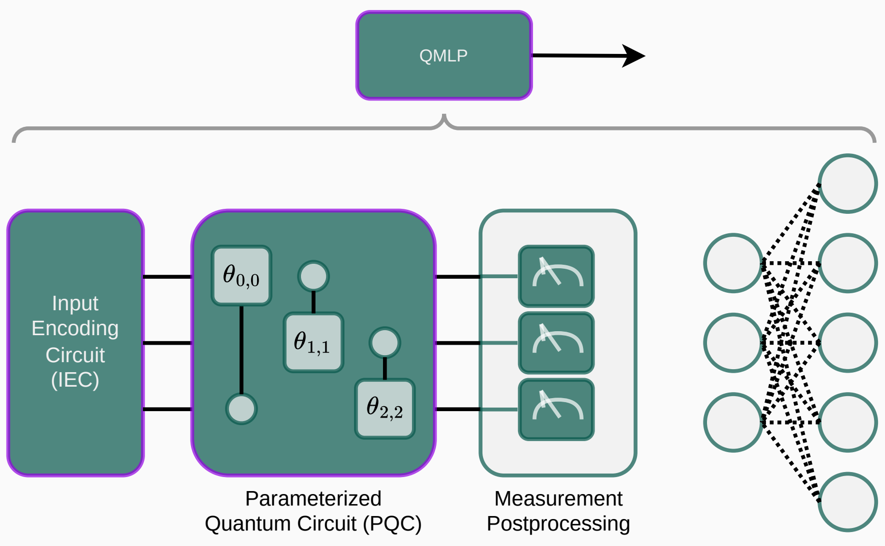
Ansatz

Neural Relational Inference (NRI) Encoder in form of a Graph Neural Network Neural Network (GNN) from [2] where the classical input layer is being extended by the quantum equivalent, the so called Quantum Multi Layer Perceptrons (QMLP).



The overall circuit in training iteration i is described as a composition of the Input Encoding Circuit (IEC) and the Parameterized Quantum Circuit (PQC) as:
$\langle\phi| = U^i_{PQC} U^i_{IEC} |0\rangle$

Learnable parameters θ in a PQC control the rotation of spin and phase of a qubit as well as the entanglement between them.
Each layer within a QMLP represents a new PQC and, in case of data reuploading also an IEC.

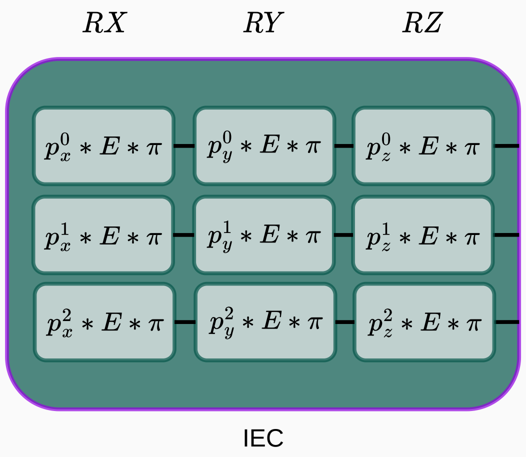


Parameter updates within the PQC are received by the gradient calculation according to the parameter shift rule $\frac{d}{d\theta} f(\theta) = r\left[f\left(\theta + \frac{\pi}{4r}\right) - f\left(\theta - \frac{\pi}{4r}\right)\right]$ [5] and backpropagation. The PQC itself can take various forms and is described explicitly in the following sections.

IEC and PQC in Detail

The FSPs are split across the qubits while their features are embedded in different type of gates as described in the following equation for a single qubit in the IEC.
$U^i_{IEC} = \bigotimes_l^L RX^i_l(p^l_x * E^l * \pi) RY^i_l(p^l_y * E^l * \pi) RZ^i_l(p_z * E^l * \pi), \qquad i \in \mathcal{N}$
The underscore _ is indicating one of the three rotational axis X, Y and Z.

The PQC follows a an entangling approach as in [7]. Additional rotational gates can be added optionally to increase the number of parameters scaling so that it scales with $2^L + 2L$.
$U^{i,l}_{PQC} = RX^i_l(\theta_{i,l,x}) RY^i_l(\theta_{i,l,y}) CRX^i_{l,L-1}(\theta_{i,l,cx}) CRY^i_{l,L-1}(\theta_{i,l,y}), \qquad i \in \mathcal{N}$
where the parameterizable matrices $CR\_(\cdot)$ describing controlled gates with the sames properties as the solely rotational gates. They are assumed to match the number of qubits L, i.e. are tensored with the identity matrix of an appropriate size.
Only X and Y types are being as they are sufficient to cover the area of the Bloch Sphere.
Alternatively, "Circuit 19" [7] is evaluated as PQC. In comparison, the latter has fewer parameters as the RY and CRY gates are omitted.



Interpretation of the measurement outcome:
Measuring a L qubit system yields 2^L values for the pseudo-probabilities of each combination of the qubits being either in the $|0\rangle$ or $|1\rangle$ state. While using the pseudo-probabilities of single qubits is usefull for classification tasks, it disregards the entanglement properties of the states.
Therefore pseudo-probabilities for the individual qubits are filtered and used to build up and $l \times 2^{L-1} + 1$ matrix which is used as input to the GNN.

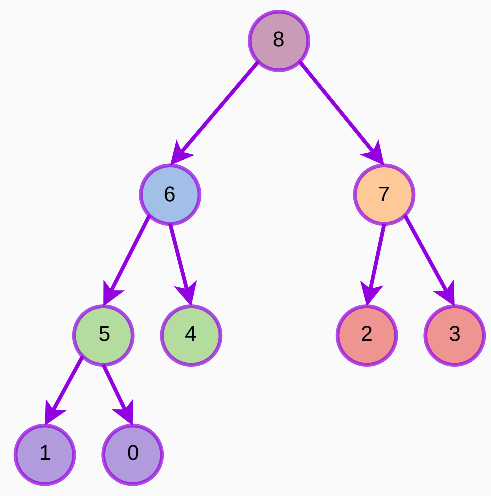
QMLPs within the GNN

This approach was evaluated as it appeared intuitive to replace the classical MLP layers with quantum equivalents. Hereby, the QMLP shown above takes the same input as the classical MLPs in the GNN. However, due to the size of the GNN, the number of parameters of sufficiently large QMLPs (e.g. "Circuit 19" [7]) exceeds that which can be optimized during a reasonably long training.
Therefore it was decided to not further investigate in this approach.

Dataset and Model Parameters

Due to the significantly longer evaluation time, experiments were conducted on a smaller dataset [4] with the parameters: MAX_CHILDREN=3, MIN_CHILDREN=2, MAX_DEPTH=3, N_TOPOLOGIES=5, N_EVENTS_PER_TOP_[train., val.]=[100,30]. This setting yields a maximum number of FSPs of $2^{3-1} = 8$ (exponential tree), although the constraints regarding the masses yields an effective number of 5.
Gradients were tuned using the ADAM optimizer in conjunction with learning rate scheduling.

The model is implemented in PyTorch and Qiskit [3].
It features 3 VQC layers which contribute to the total 38317 parameters of the hybrid model.
Mini-Batch training with a batch size of 8 in the classical- and 4 in the hybrid model training.



Quintessence

The approach in which the GNN is enhanced by a QMLP is able to surpass the purely classical experiment on the perfect LCAG score.
Furthermore, the accuracy and logic accuracy score as well as the loss can be considered being very competitive. Especially in regards on the parameters, the quantum part seems to approve the overall perfomance significantly.
This can also be seen when comparing the hybrid approach to a classical one where the number of parameters is similar. In this scenario, all metrics are significantly surpassed by the hybrid approach.
However, validation on a real quantum computer decreases e.g. the perfect LCAG score by almost 20%.
This drawback motivates further research towards noise resilient approaches which hopefully also improve the accuracy on noise prone input data.
It should be noted that the much longer training time, access limitations to real quantum devices, and limited scalability of this approach will not make real-world application feasible in the near future, but rather help build an understanding and intuition of quantum technologies.

References

- [1] Learning tree structures from leaves for particle decay reconstruction - Kahn et al. - 2022
- [2] Neural Relational Inference for Interacting Systems - Kipf et al. - 2018
- [3] Qiskit: An Open-source Framework for Quantum Computing --Md Sajid et al. - 2021
- [4] Predict better with less training data using a QNN - Reese et al. - 2022
- [5] Evaluating analytic gradients on quantum hardware - Schuld et al. - 2019
- [6] phasespace: n-body phase space generation in Python - Navarro and Eschle - 2019
- [7] Expressibility and Entangling Capability of Parameterized Quantum Circuits for Hybrid Quantum-Classical Algorithms - Sukin Sim

D3A | Steinbuch Centre For Computing | Karlsruhe Institute of Technology
Contribution #184 | Contact: melvin.strobl@kit.edu