STEREOGRAPH

Application of GNNs to the stereoscopic reconstruction of events from the Cherenkov Telescope Array (CTAO)

Machine learning workshop for Analysis of high energy cosmic particles
University of Delaware
January 30, 2025

Hana Ali Messaoud,CDD, LAPP, Annecy (France)
hana.alimessaoud@lapp.in2p3.fr

CTAO
Cherenkov Telescope Array Observatory

learn

LAPP LISTIC UNIVERSITÉ SAVOIE MONT BLANC cnrs

Introduction

CONTEXT

- **Gamma Learn :** launched in 2017 through a collaboration between LAPP and LISTIC.
- **Goal :** to improve CTAO event reconstruction using AI methods, deep learning
- **Stereograph** (started in 2024)
- **Objective :** to explore stereoscopic reconstruction of gamma events using graph neural networks

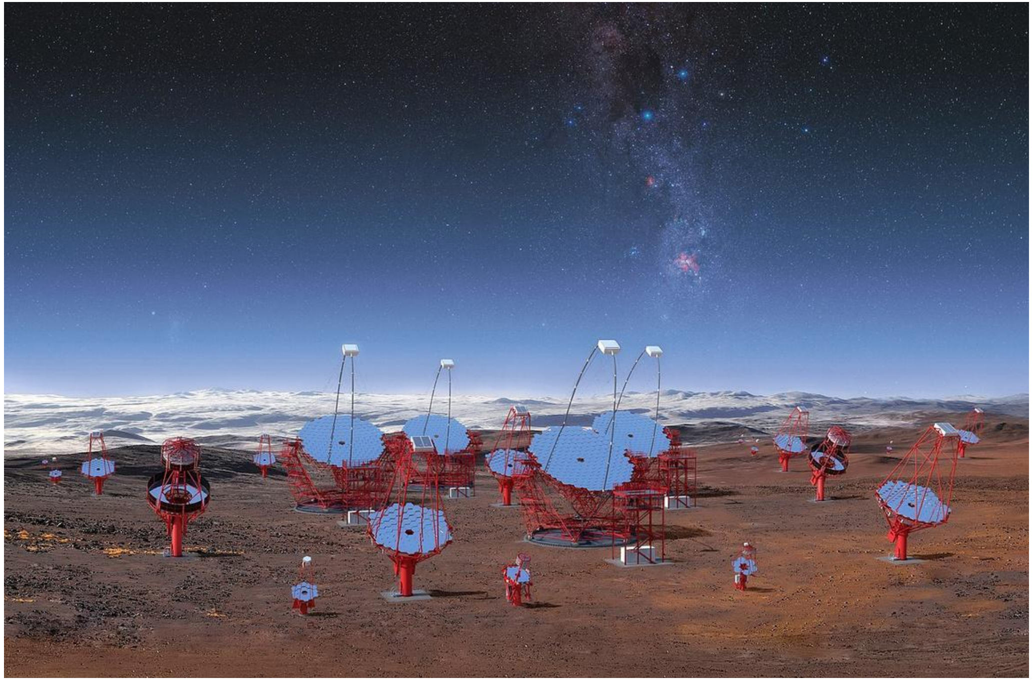
Contents

Introduction

CTAO

The Cherenkov Telescope Array Observatory

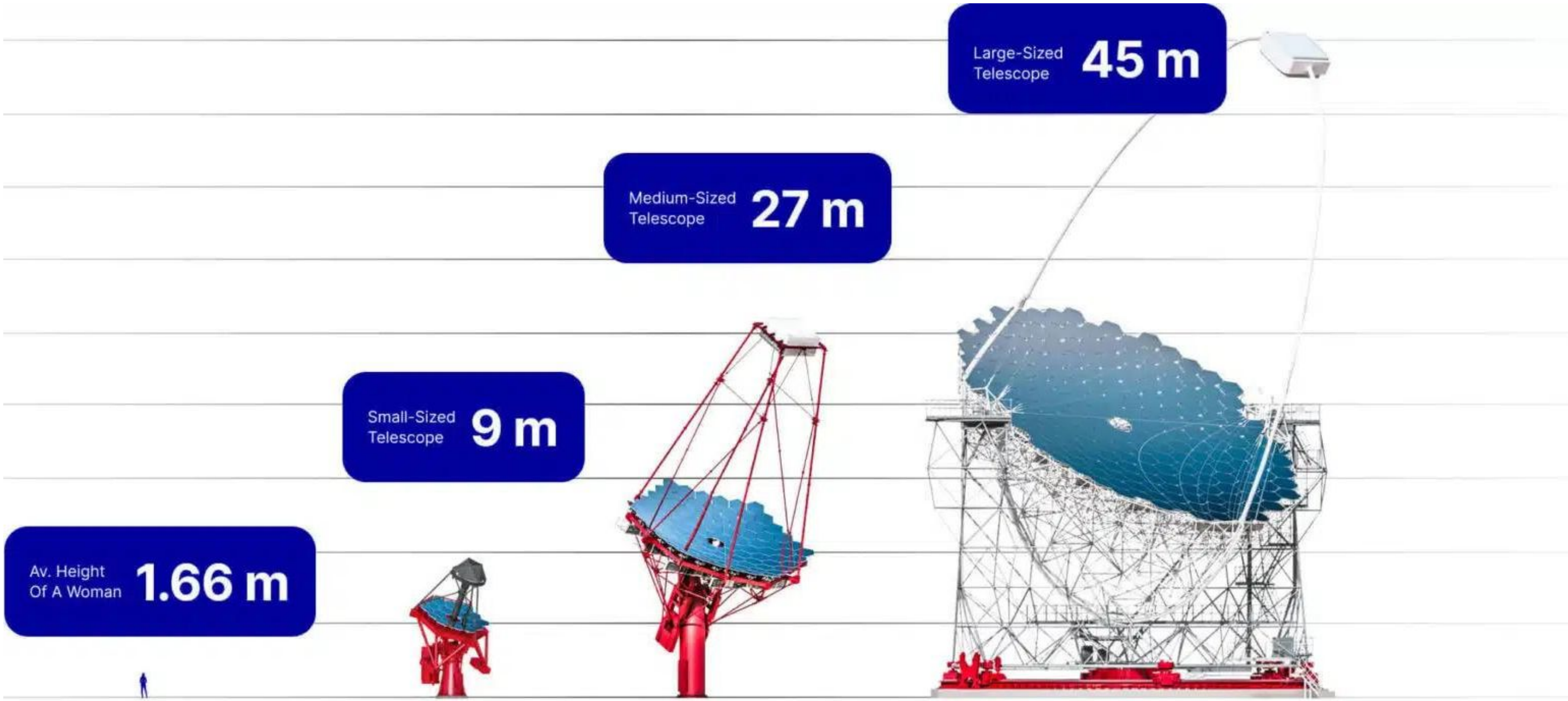
- The project will feature about sixty telescopes of various sizes.
- Which will be installed in the Northern Hemisphere on the island of La Palma and in the Southern Hemisphere in Chile.
- They will cover an energy range of approximately 20 GeV to 300 TeV.



Artist's view of the three types of telescopes deployed in Chile

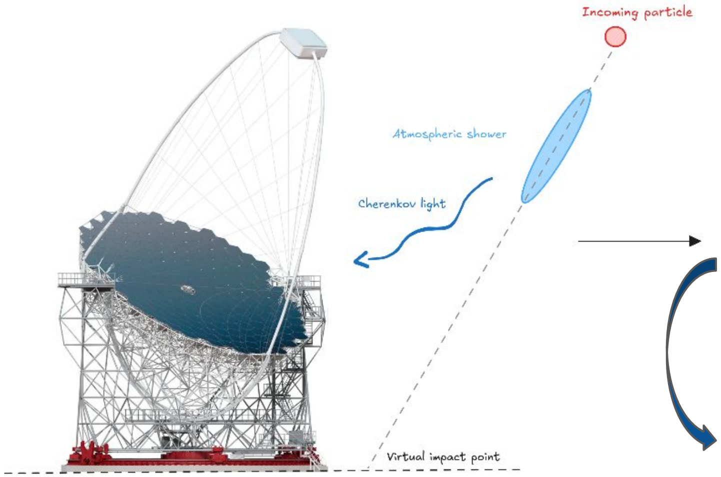
3 TYPES OF TELESCOPES

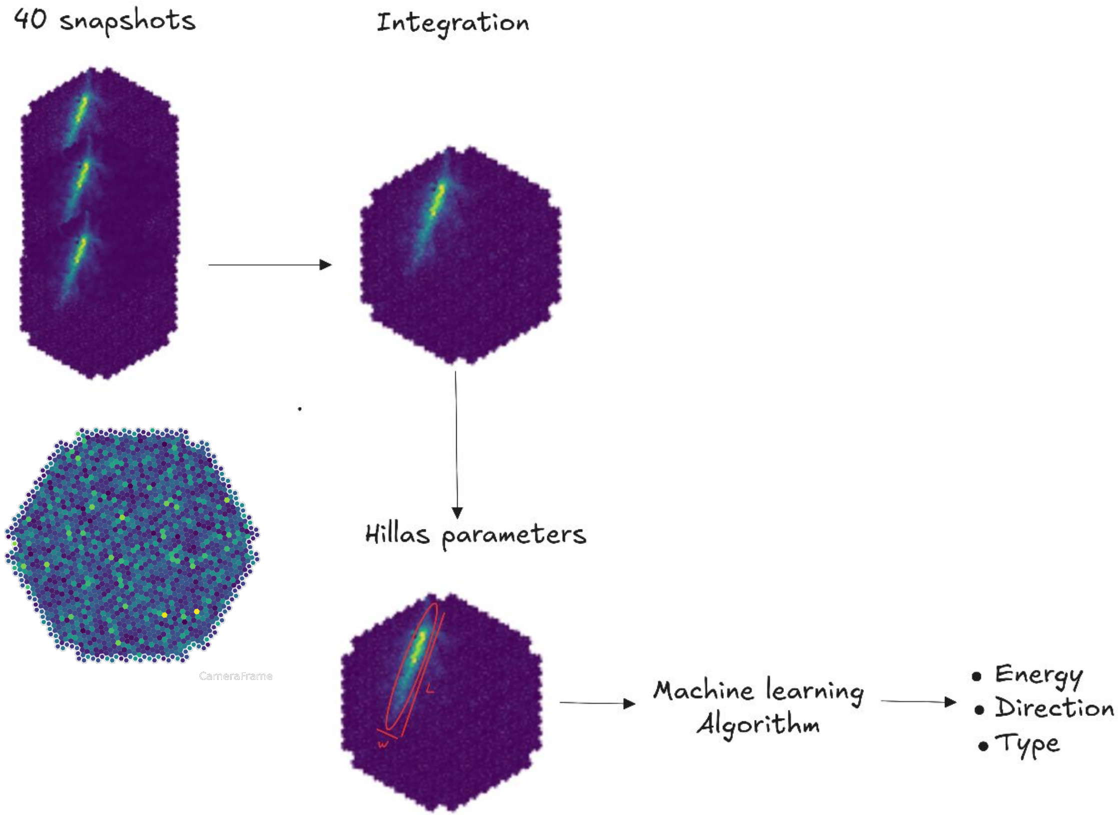
- Ongoing project under construction
- Currently, only one telescope is in operation (LST-1).



Source:https://www.ctao.org/emission-to-discovery/telescopes/

Event Detection





- Gamma photons interact with the atmosphere and generate particle cascades (atmospheric shower).
- The charged secondary particles travel faster than light in the atmosphere and emit blue light (Cherenkov radiation).
- This light is detected by ground-based telescopes.

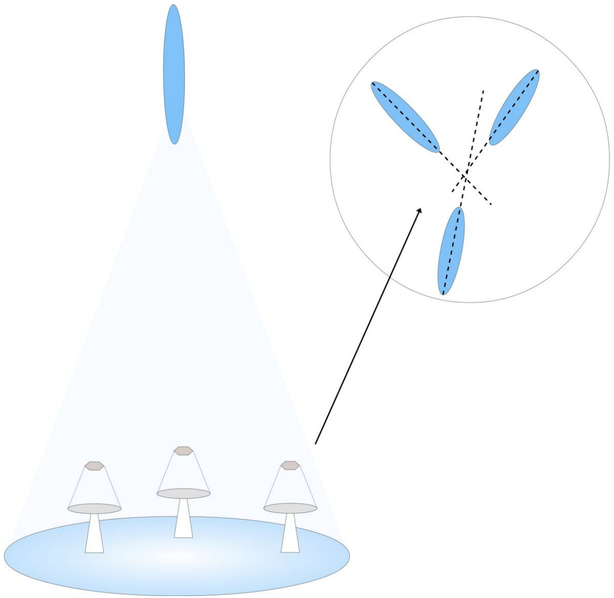
The stereoscopic event reconstruction

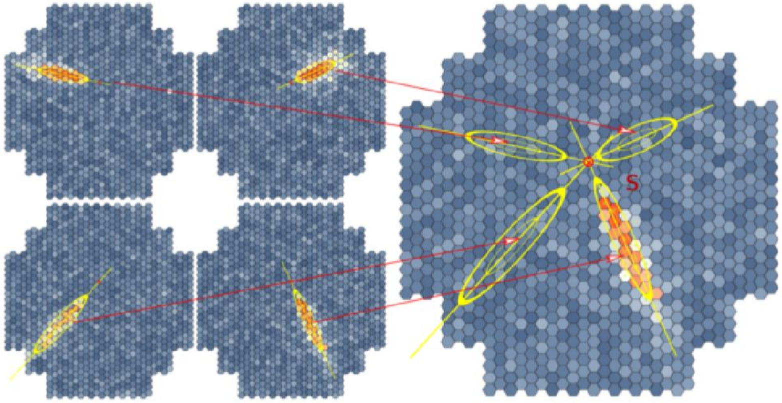
- By machine learning methods
- **Standard method**: Random Forest + Hillas
    - Training one Random Forest per telescope and per task (Energy and class).
    - Fast and robust

- **Stereoscopy :**
    - The direction is reconstructed geometrically.
    - By combining the observations of all telescopes to give a common estimation of a recorded shower

The **stereoscopic** event reconstruction

Stereoscopy : By combining the observations of all telescopes to give a common estimation of a recorded shower





The **stereoscopic** event reconstruction

Random Forest

Weighted average of the predictions by telescope (for the energy and class)

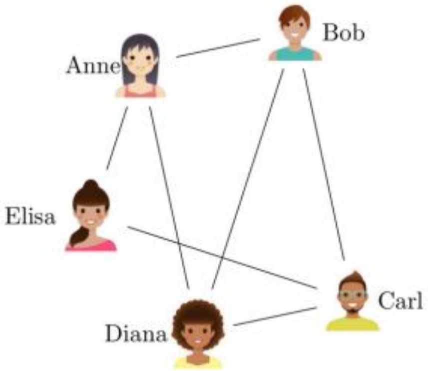
Deep Learning

A deep learning approach based on graphs.

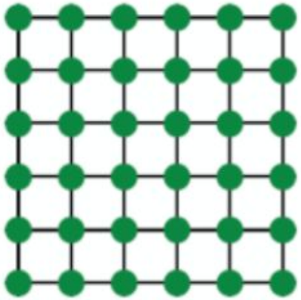
GRAPHS

Graphs

- **G=(V,E)** : A pair consisting of two sets
- **V :** vertices (also called nodes)
- **E :** edges, each associated with a pair {u, v} of vertices, where (u, v) ∈ V
- Graphs are everywhere around us, and they can be found in various fields and applications, such as

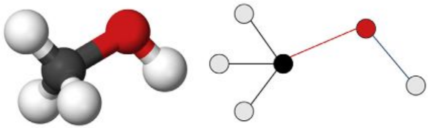


Social networks



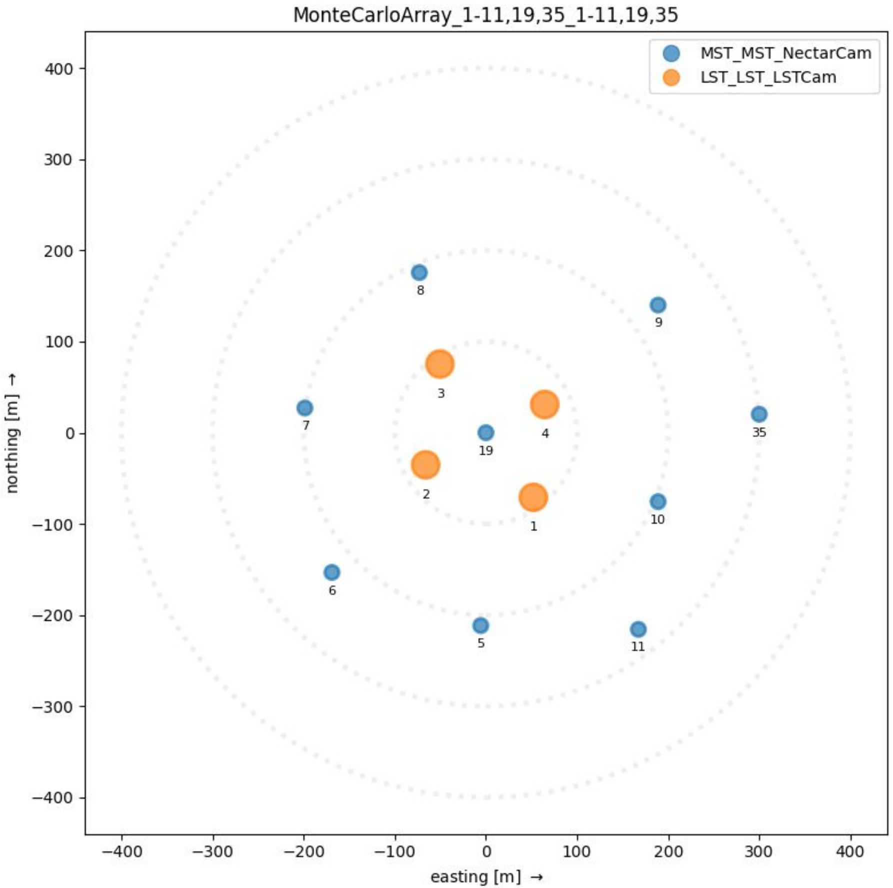
Images

Text

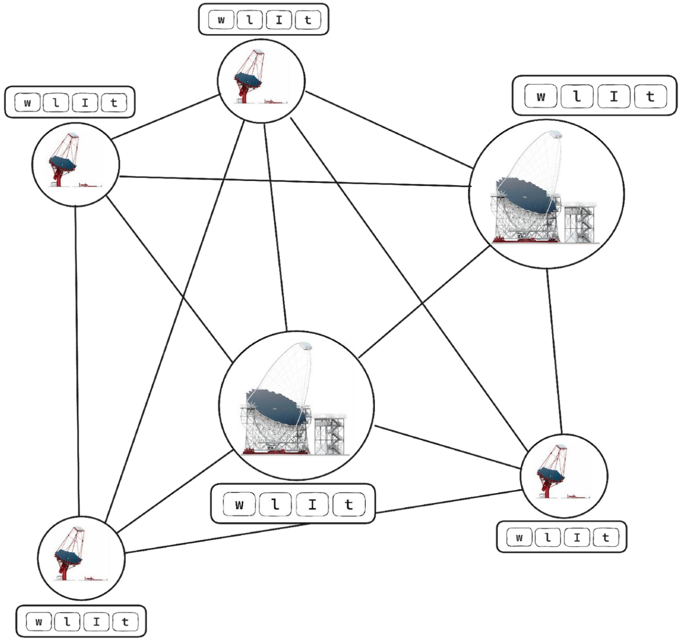


Molecule structures

- ***Why graphs?***



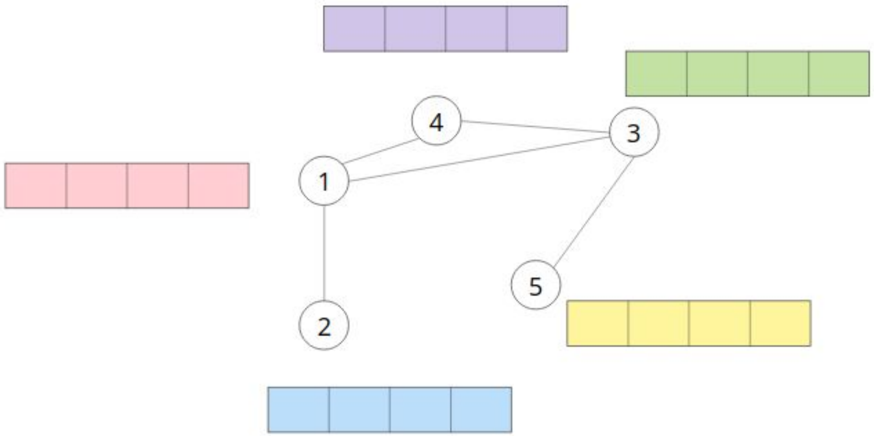
Array display (North site- La Palma)



Schematic representation of a graph. Each node of the graph is a telescope of
the array.

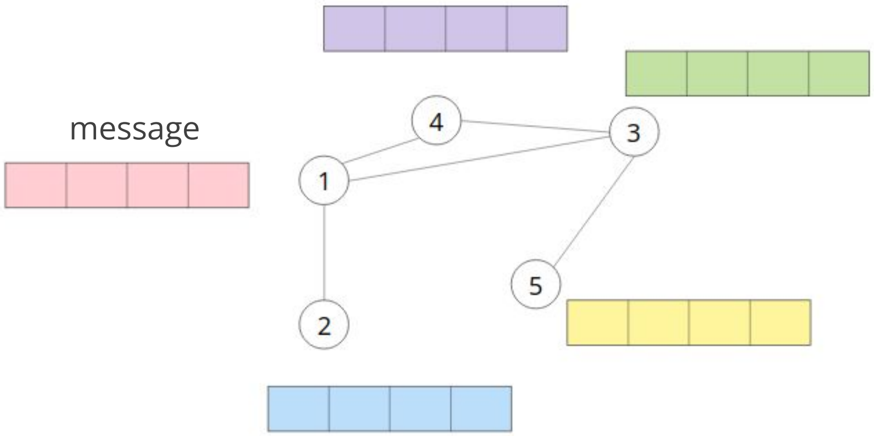
GRAPH NEURAL NETWORKS

LEARNING ON GRAPHS



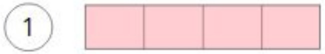
LEARNING ON GRAPHS

- **Message passing:** Each node in the graph collects the embeddings of its neighbors (messages).
  - **Message passing function**: Affine, or a neural network (MLP).

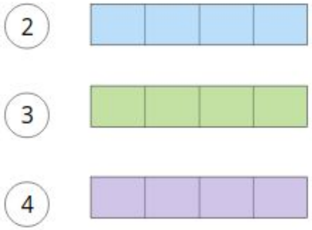


LEARNING ON GRAPHS

- **Message passing:** Each node in the graph collects the embeddings of its neighbors (messages).
    - **Message passing function**: Affine, or a neural network (MLP).



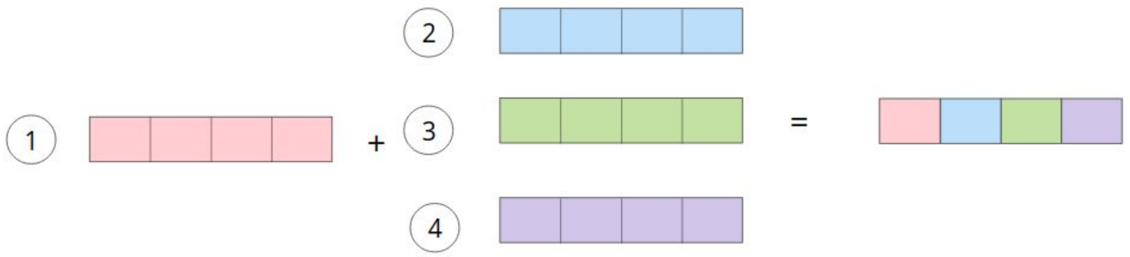
- **Aggregation:** The neighbors' messages are aggregated using an aggregation function (sum, max, average).



LEARNING ON GRAPHS

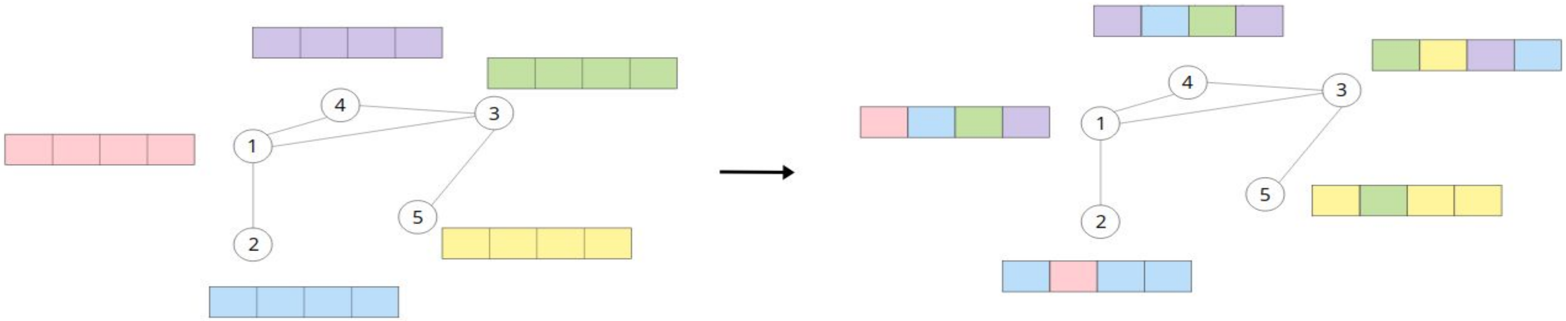
- **Update of embeddings :** All the aggregated messages are then passed through an update function to produce the new node embeddings.

$$h_i = \sigma\left(\gamma(\phi(x_i)) + \bar{m}_i\right)$$



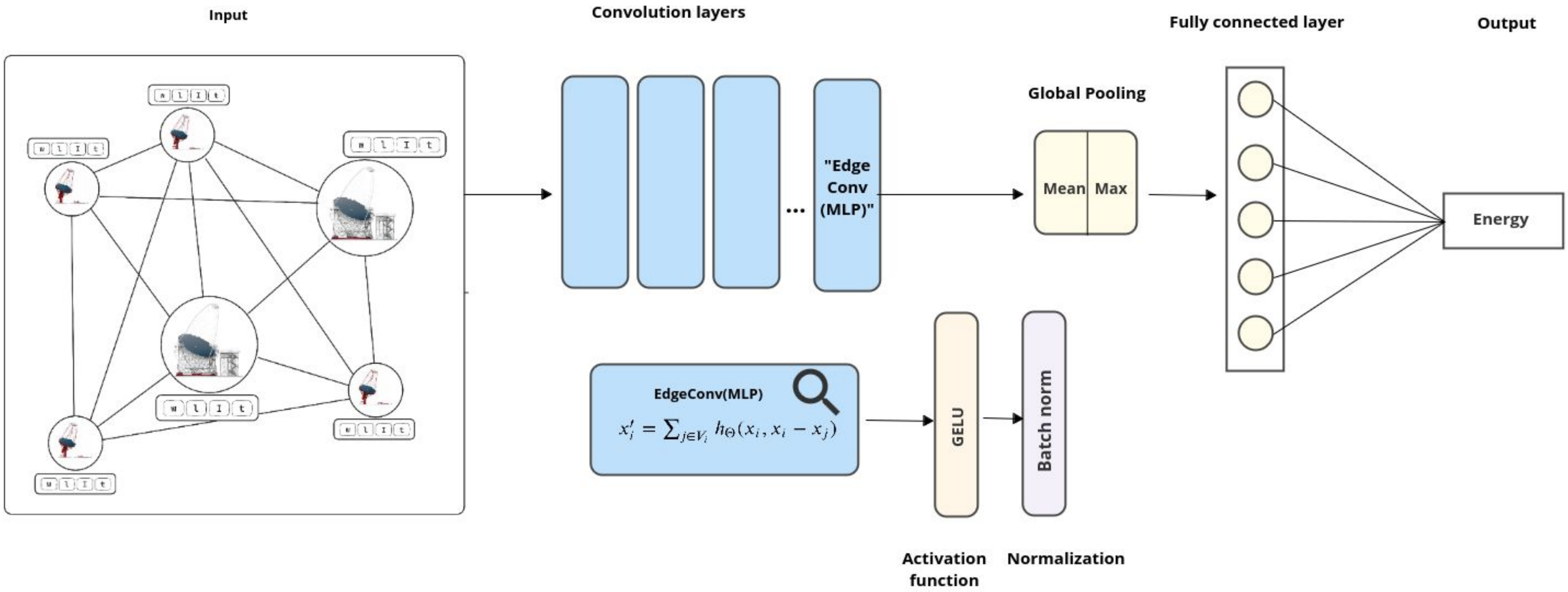
LEARNING ON GRAPHS

- Each step is repeated for every node in the graph to update the embeddings.



The **stereoscopic** event reconstruction

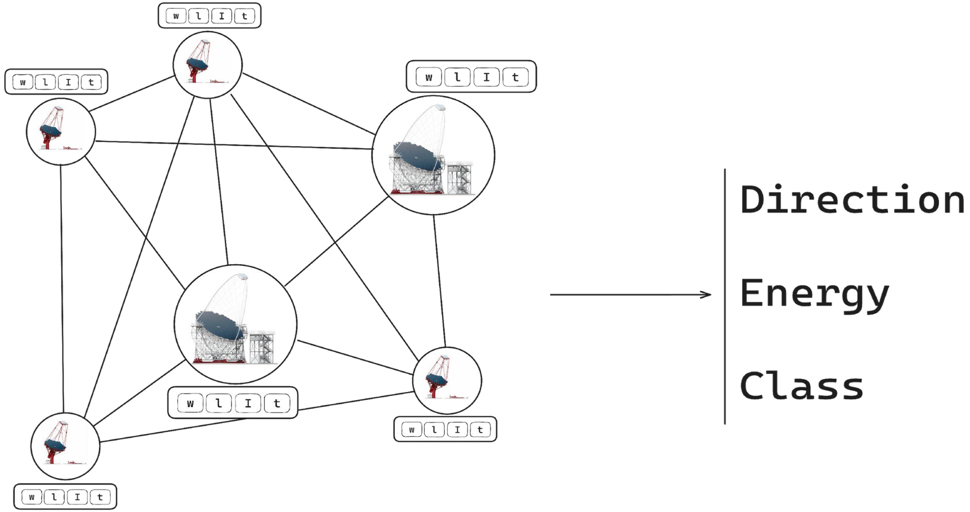
MODEL ARCHITECTURE



The **stereoscopic** event reconstruction

PRODUCING GRAPHS

- Each event will be represented by one graph.
- 42 features per node
- Hillas parameters on nodes (same parameters used to train RFs).
- Monte Carlo (MC) simulations from ctapipe v19 (prod5_ctapipe_v0.19).
- Non triggered-telescopes aren't included

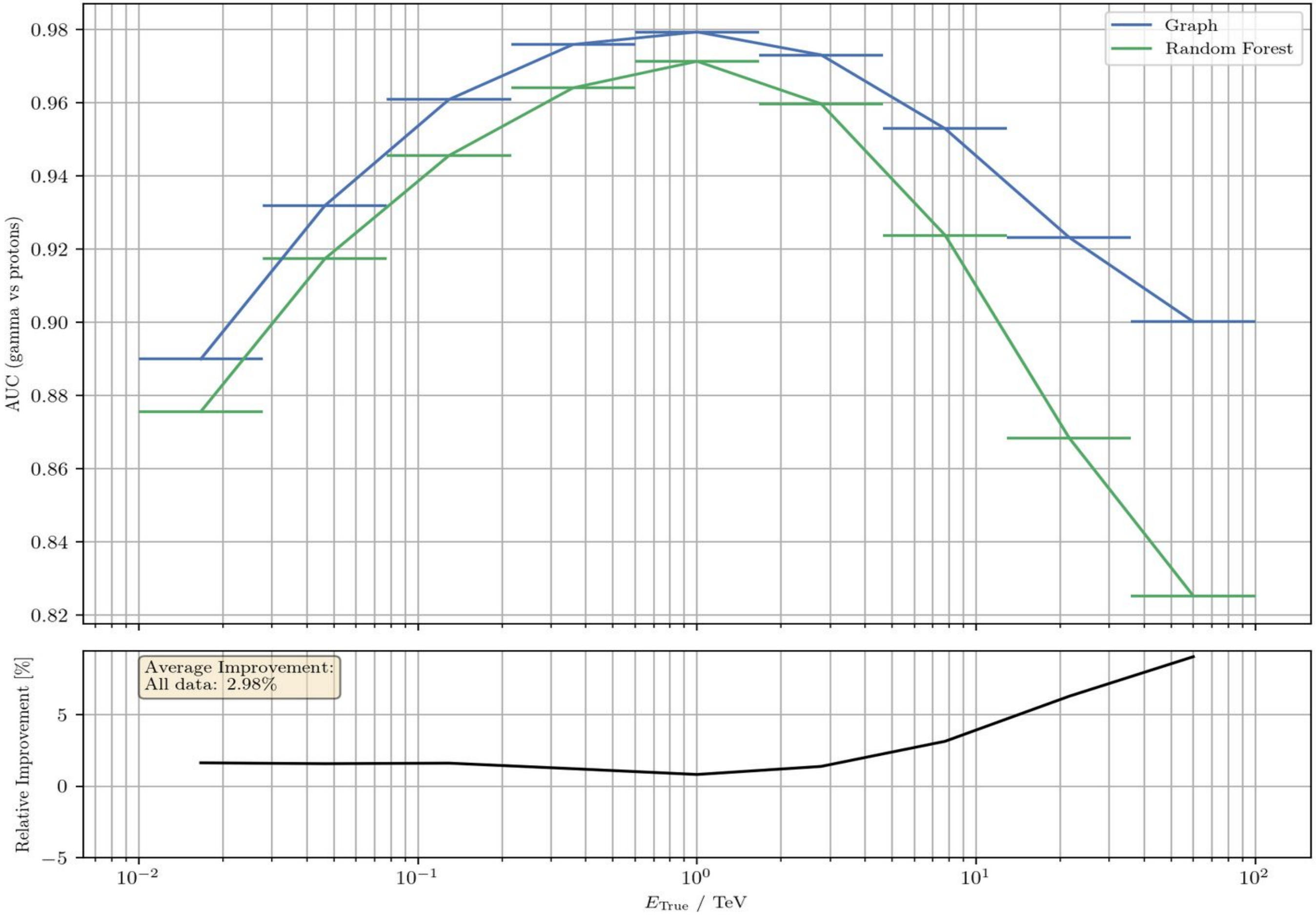


Each node has the Hillas parameters as node features.
For each graph, we predict a global graph reconstruction:
the direction, energy, and class probability.

RESULTS

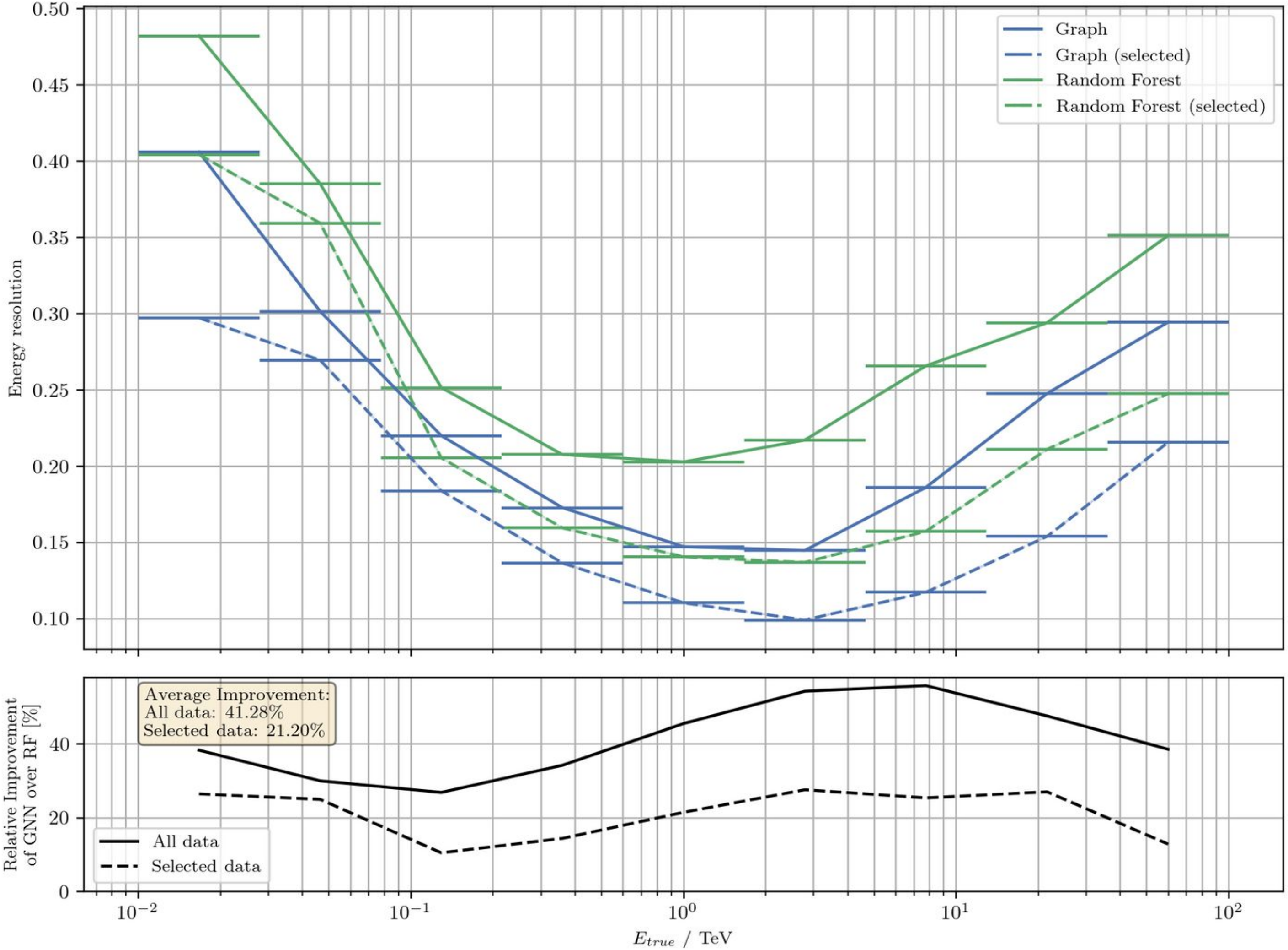
Results

ROC/AUC

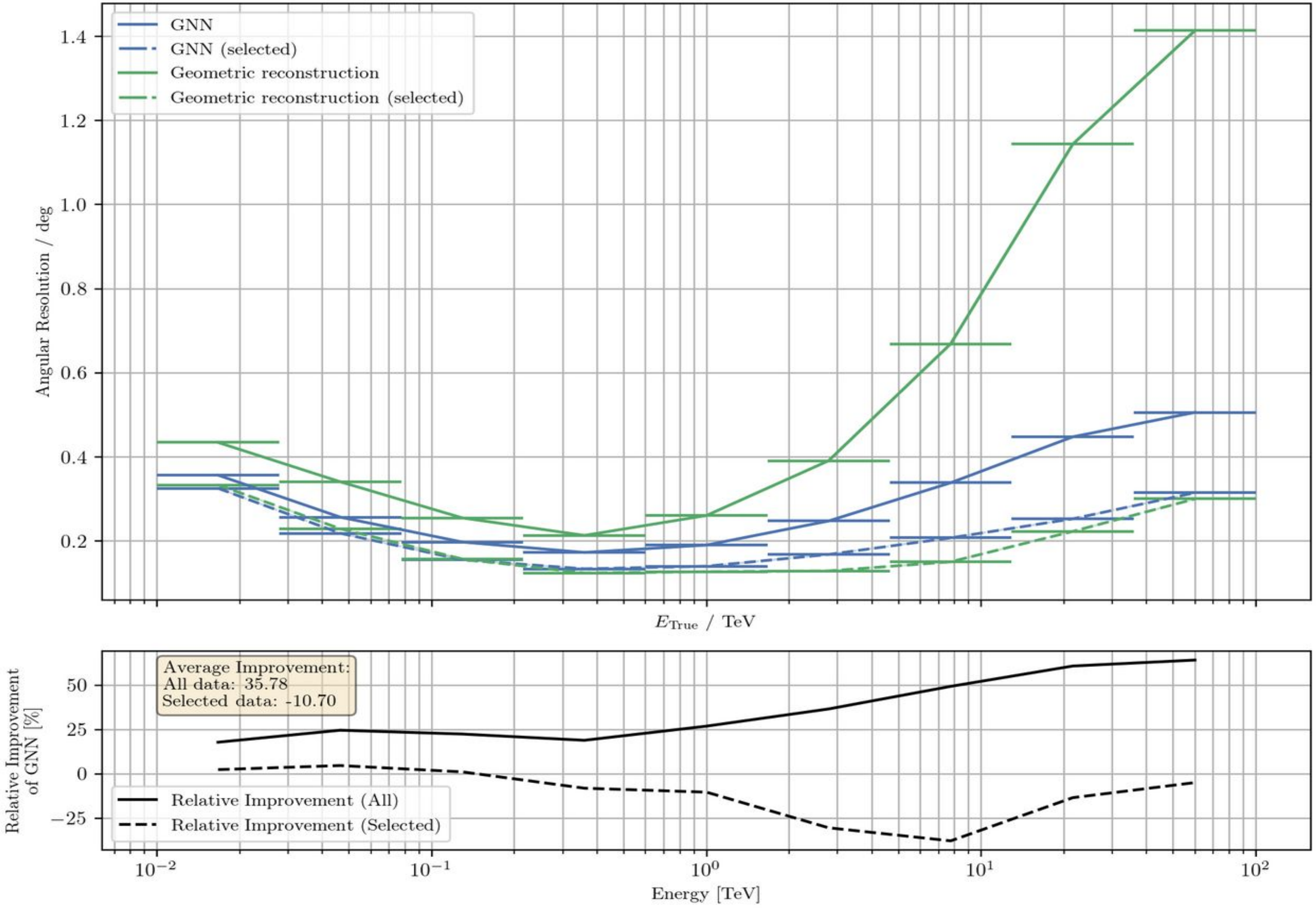


Energy resolution

Event selection:
We apply a gammaness cut to retain 70% of the events most likely to be gamma rays.



Angular resolution



Conclusion

- Graphs : **A promising approach** for the stereoscopic reconstruction of gamma events.
- Significant improvement observed in the reconstruction, with better energy and direction resolution (before event selection), as well as improved separation between gamma photons and protons.
- Not more complicated than RFs (same inputs, restructured as node features).
- Relatively fast to train (a few hours on a small GPU, or CPU)
- **Drawbacks :**
- Lack of transparency in the functioning of GNNs
- Memory greedy (graph production).

Perspectives

- Potential integration into the GammaLearn project, which is currently focused on monoscopic reconstruction.
- Open science

Links

- Reproducible analysis (Open data)
- Gitlab repository (Open source) :

https://gitlab.in2p3.fr/gammalearn/stereograph/stereograph

Thank you !

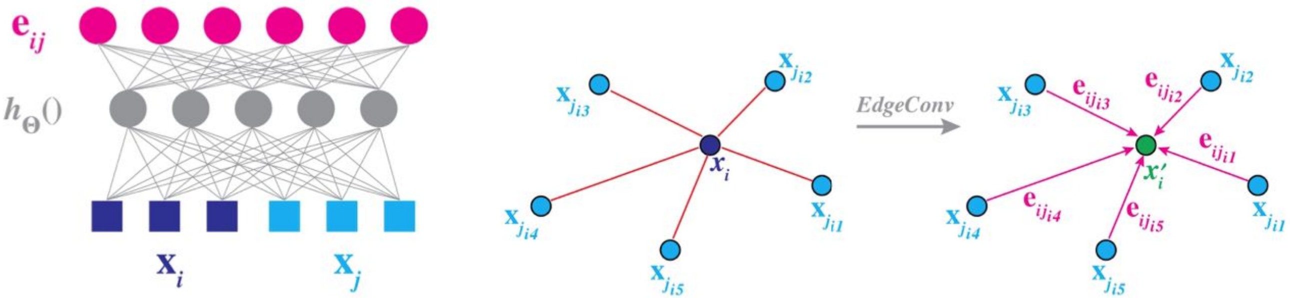
Backup

Input features : Hillas parameters

tel_id	timing_intercept	morphology_n_pixels	intensity_max
hillas_intensity	timing_deviation	morphology_n_islands	intensity_min
hillas_skewness	timing_slope	morphology_n_small_islands	intensity_mean
hillas_kurtosis	leakage_pixels_width_1	morphology_n_medium_islands	intensity_std
hillas_fov_lon	leakage_pixels_width_2	morphology_n_large_islands	intensity_skewness
hillas_fov_lat	leakage_intensity_width_1	intensity_kurtosis	intensity_kurtosis
hillas_r	leakage_intensity_width_2	peak_time_max	peak_time_min
hillas_phi	concentration_cog	peak_time_mean	peak_time_std
hillas_length	concentration_core	peak_time_skewness	peak_time_kurtosis
hillas_length_uncertainty	concentration_pixel	core_psi	type
hillas_width			
hillas_width_uncertainty			
hillas_psi			

EDGE CONV

- **Edge conv** : graph convolution operator used for point cloud learning and graph-based tasks.
- **Idea :** The feature of each node is updated by aggregating information from its neighbors, taking into account both the characteristics of the node itself and the relative difference between the node and its neighbors.
- The output of EdgeConv at the i-th node $x_i' = \sum_{j \in V_i} mlp(x_i, x_i - x_j)$

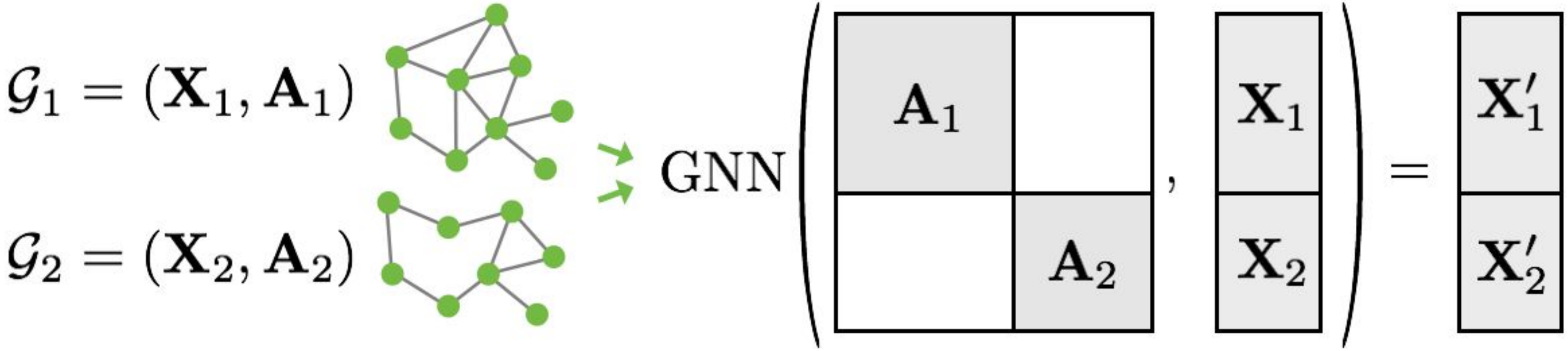


Left : Computing the features of an edge

Right : EdgeConv operation.

https://arxiv.org/pdf/1801.07829:

Batch processing



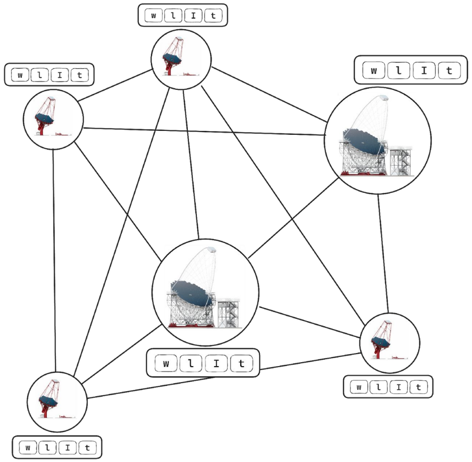
- Each graph is represented by a data object:

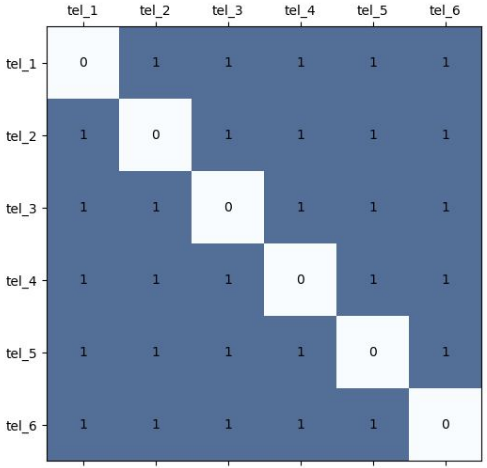
  data= Data(x=x, edge_index=edge_index)
- Data is grouped into batches
- Each Batch object has a batch vector, which associates each node with its corresponding graph in the batch

$$\text{batch} = [0 \quad \cdots \quad 0 \quad 1 \quad \cdots \quad n-2 \quad n-1 \quad \cdots \quad n-1]^\top$$

GRAPH REPRESENTATION

- The characteristics of the nodes are stored in a matrix.
- The edge weights are stored, depending on the case, either in the adjacency matrix or in an independent vector.





Boolean matrix describing the edges of the graph

Nœud	Adjacence
1	$\{2,3,4,5,6\}$
2	$\{1,3,4,5,6\}$
3	$\{1,2,4,5,6\}$
4	$\{1,2,3,5,6\}$
5	$\{1,2,3,4,6\}$
6	$\{1,2,3,4,5\}$

A list containing the list of adjacent vertices.