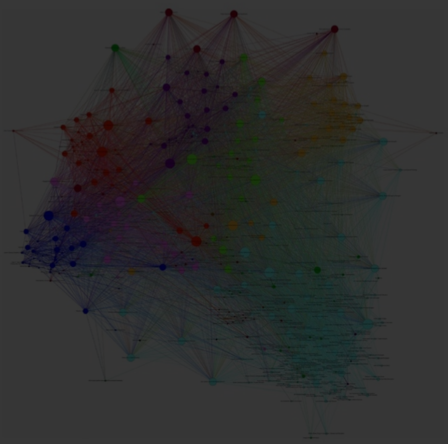
Graph Neural Networks (GNNs) & their relevance in particle physics

Tobias Golling

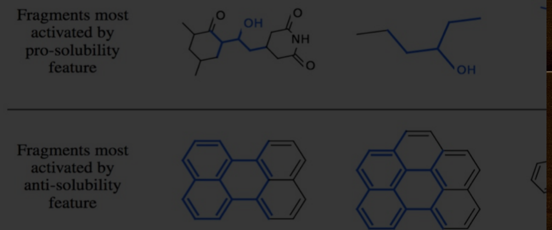
UNIVERSITÉ DE GENÈVE

FACULTY OF SCIENCE

Graph-st … ywhere…

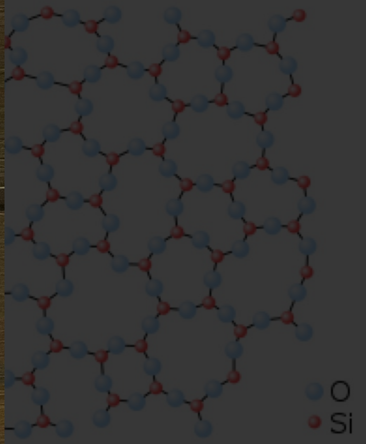


Social networks



Drug molecules

wledge graphs



Glass structure

This is a graph

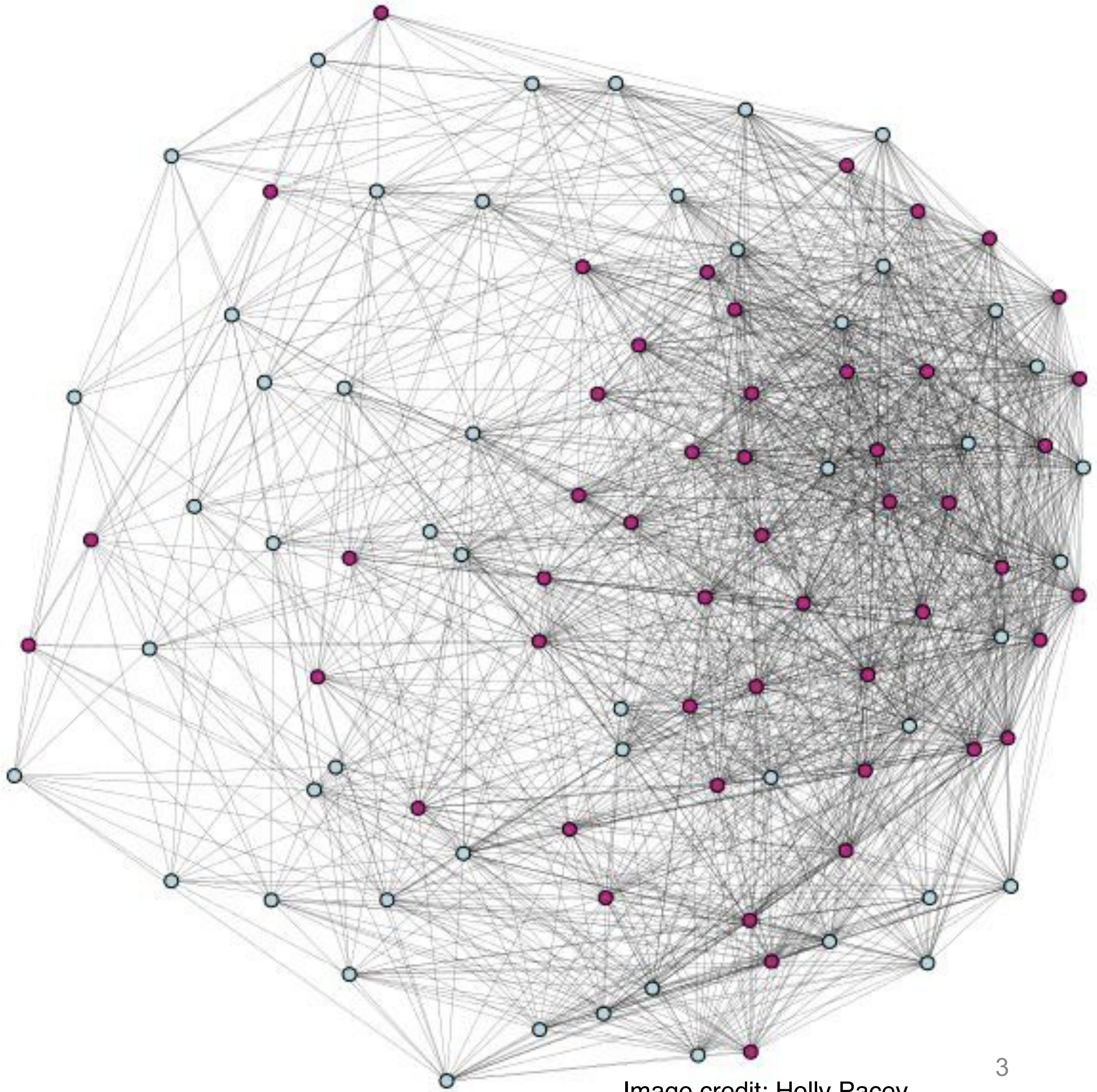


Image credit: Holly Pacey

It has nodes

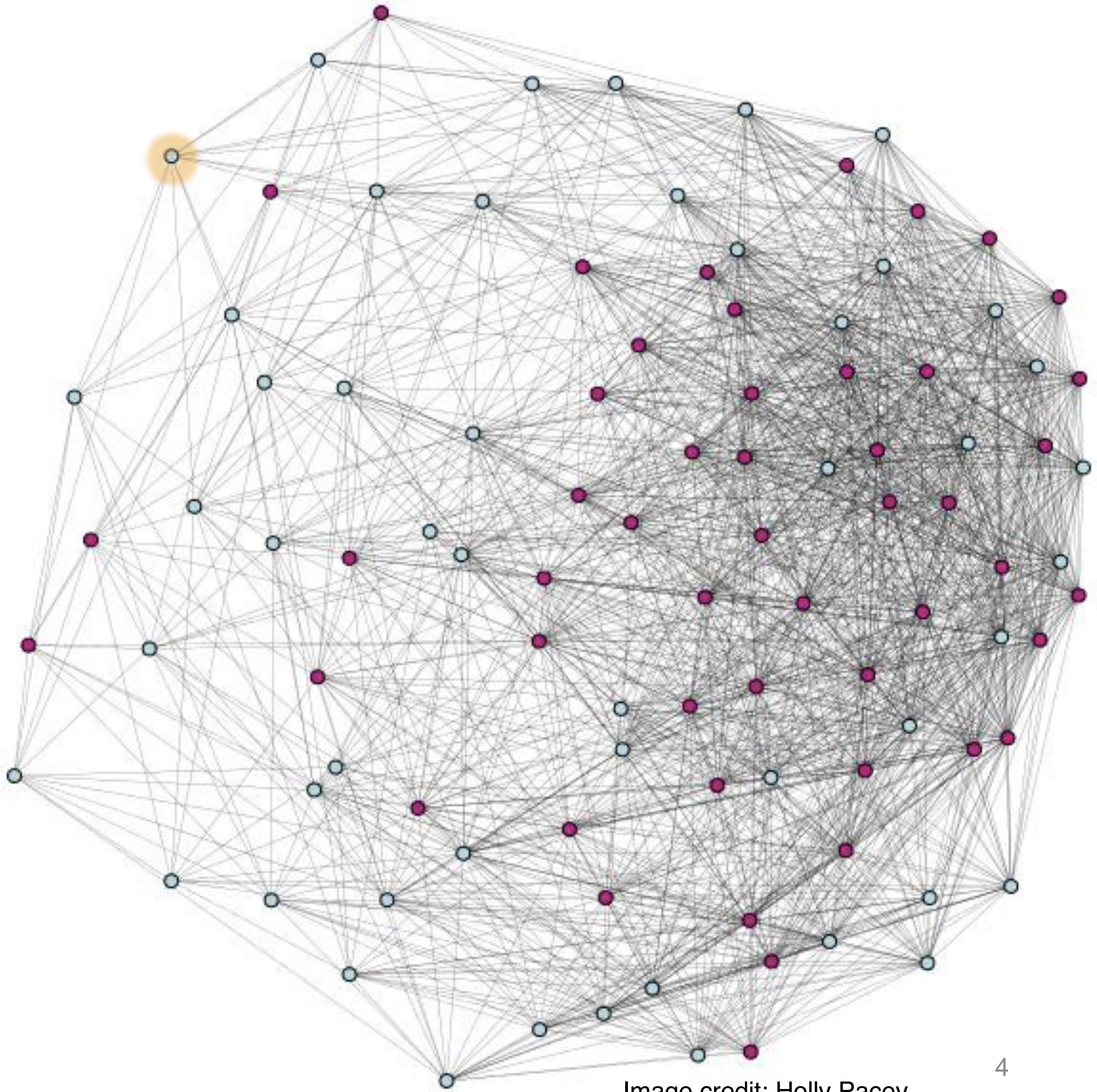


Image credit: Holly Pacey

Connected by edges

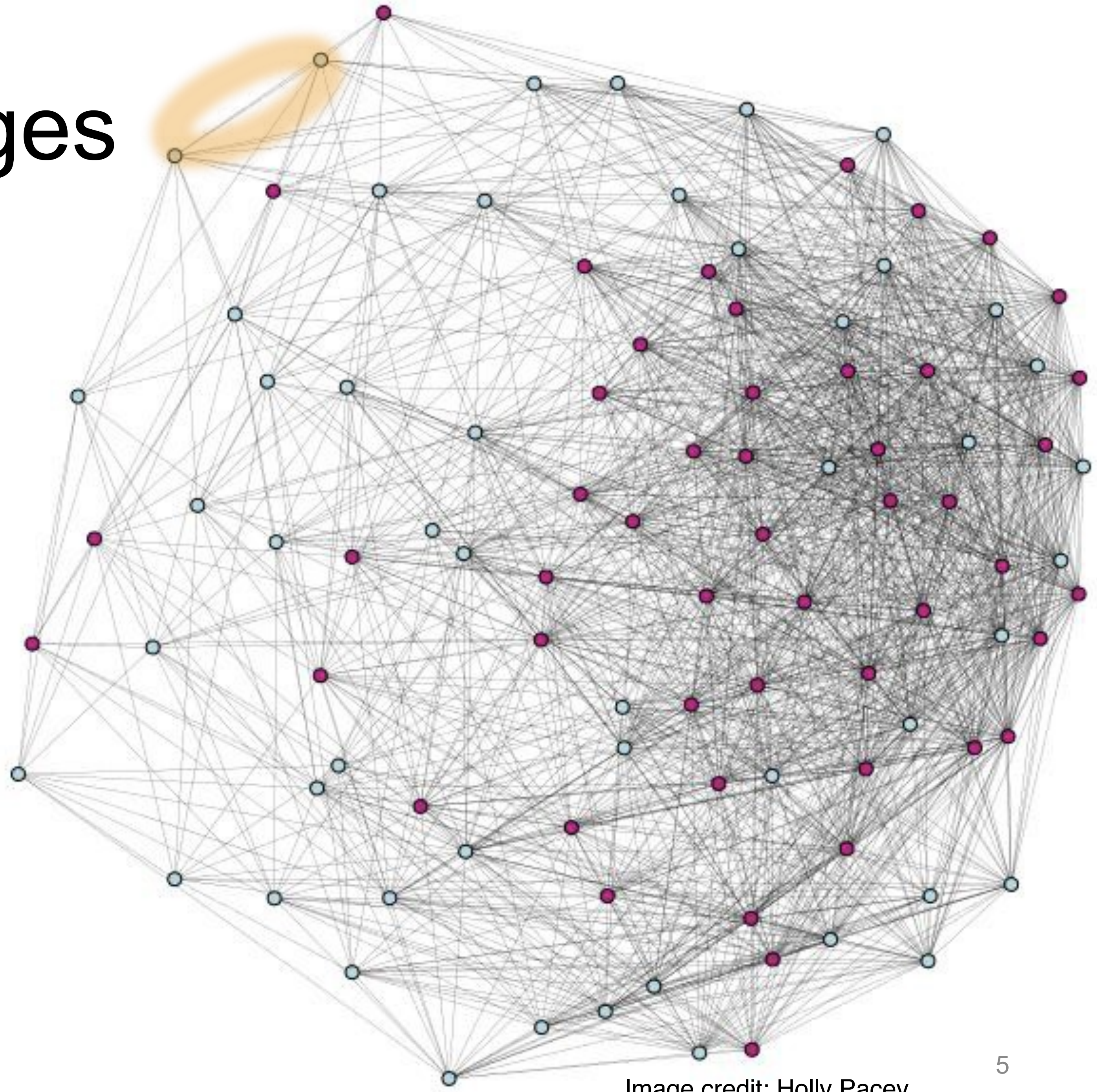
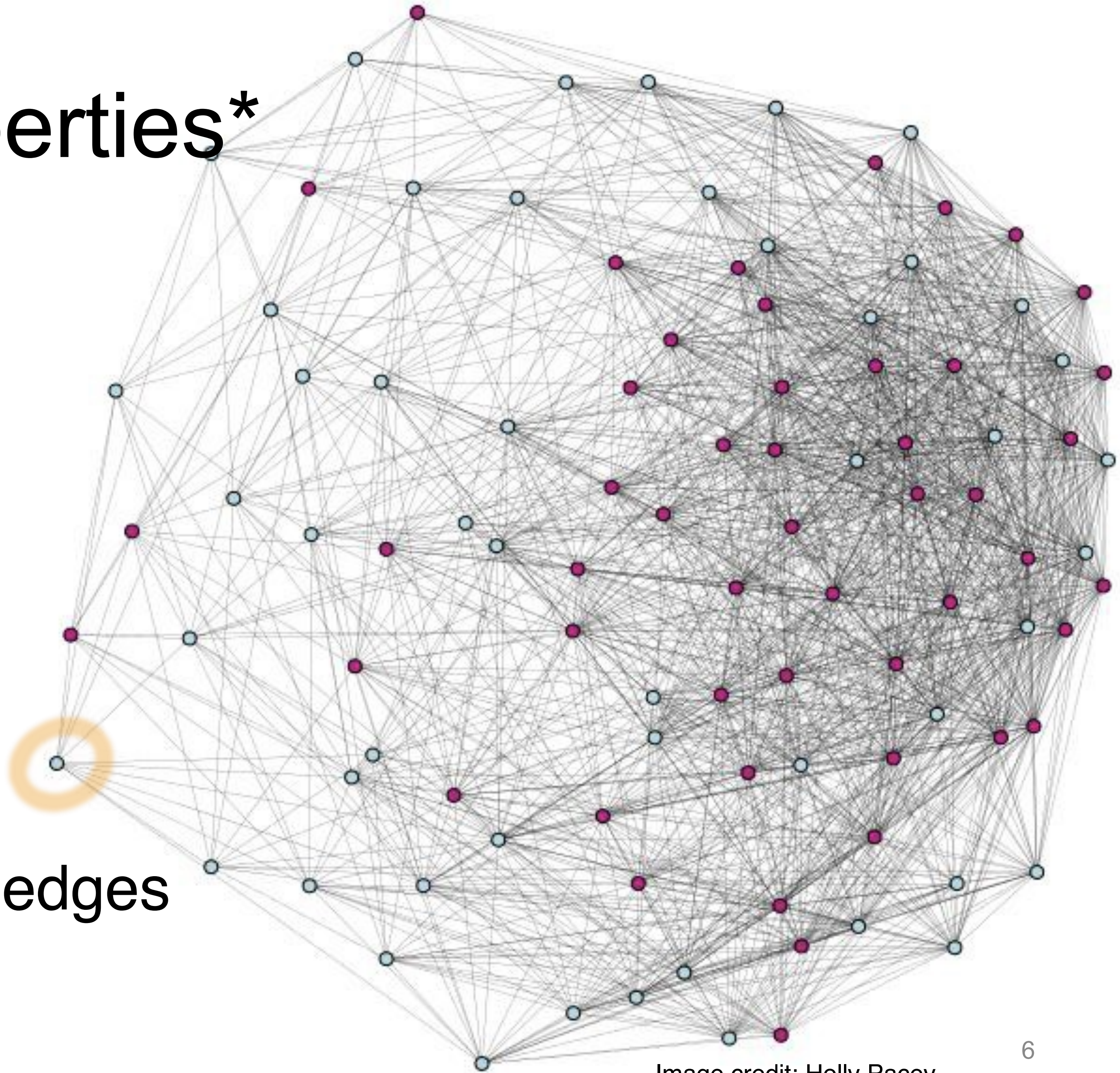


Image credit: Holly Pacey

Nodes have properties*



e.g. how many connected edges

*Properties = features = attributes

Image credit: Holly Pacey

Properties at…

Node level

Edge level

Graph level

e.g. number of edges
in connected nodes

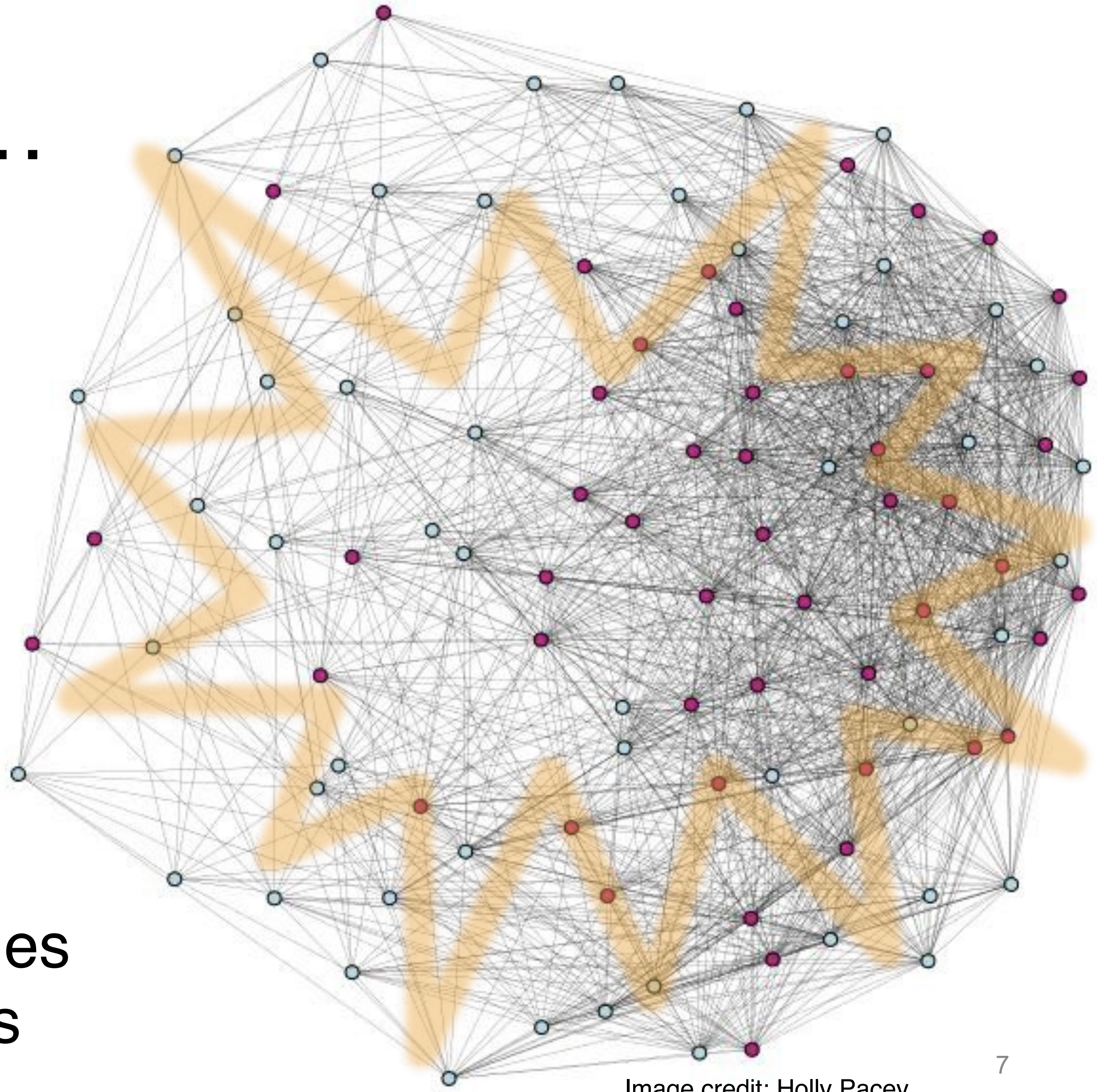


Image credit: Holly Pacey

Many GNN success stories in past few years

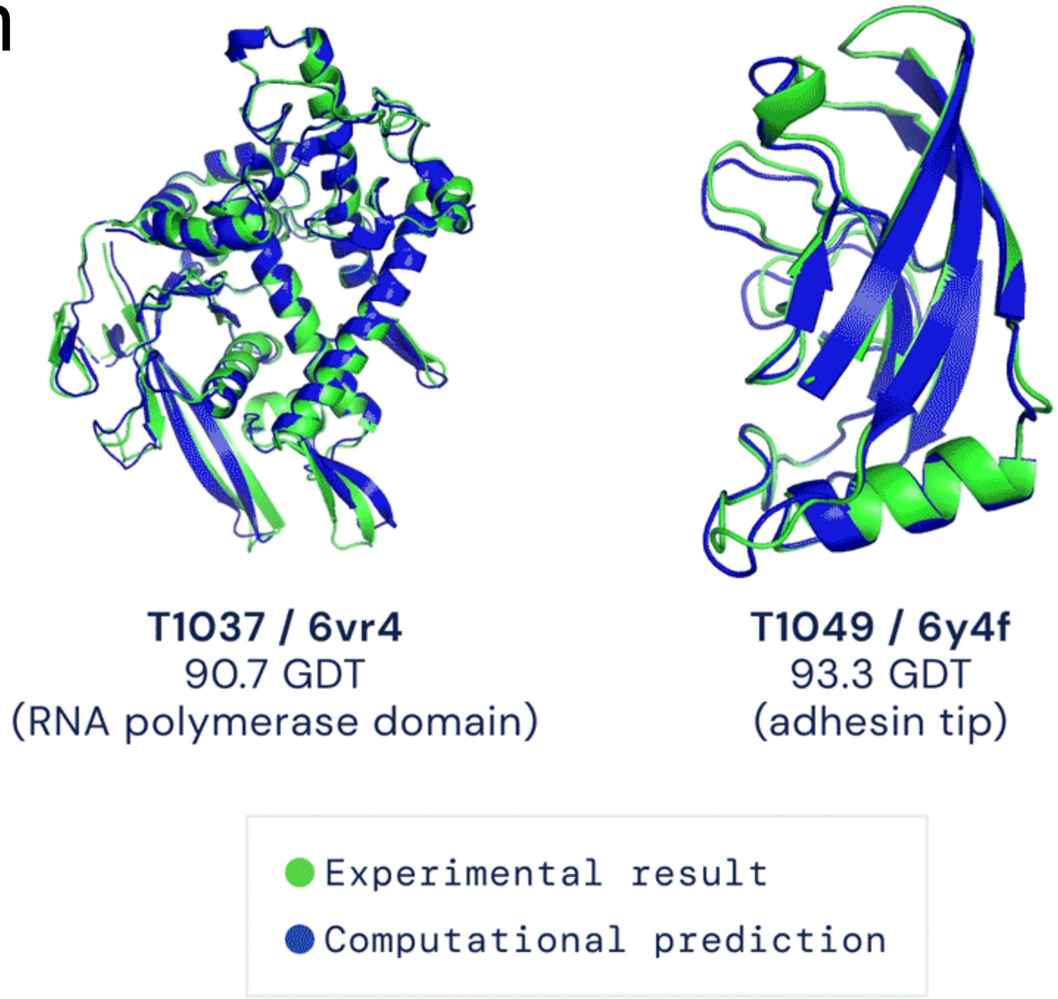
In Science and beyond…

AlphaFold (DeepMind)

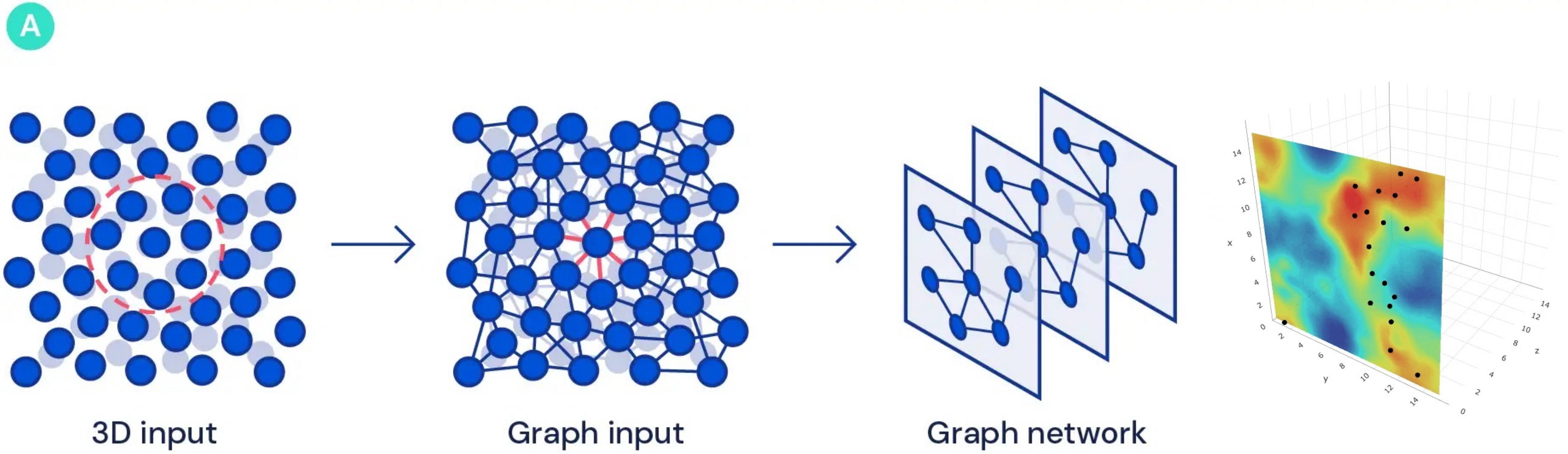
Predict 3D protein shape from sequence of amino acids with GNNs

Applications:

- Drug discovery
- Engineer enzymes
- ...

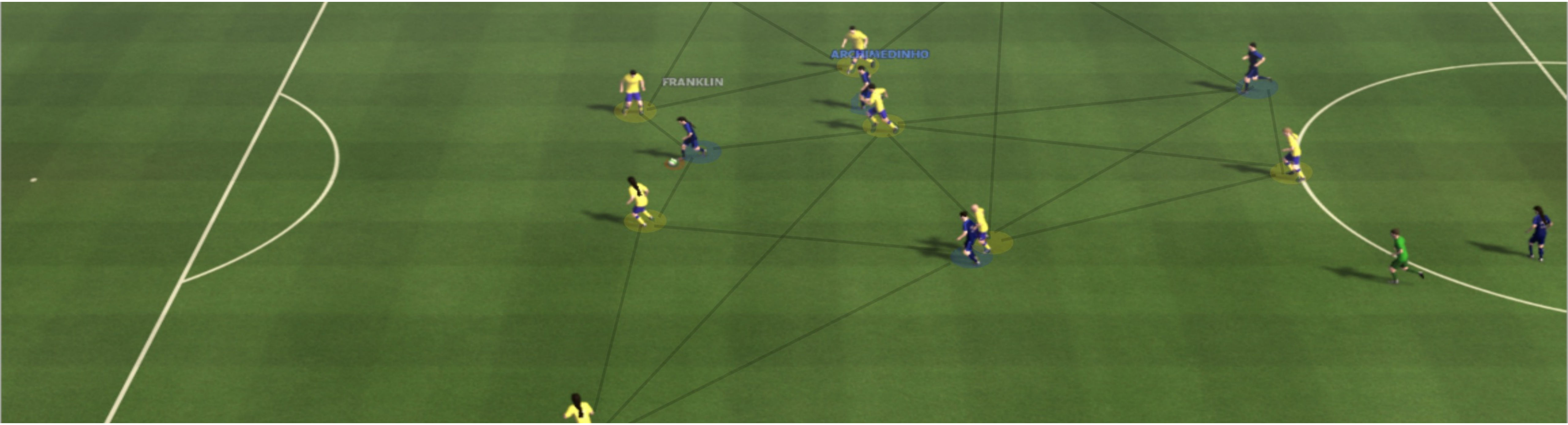


Glass dynamics simulation (DeepMind)



On current occasion…

https://www.kaggle.com/c/google-football

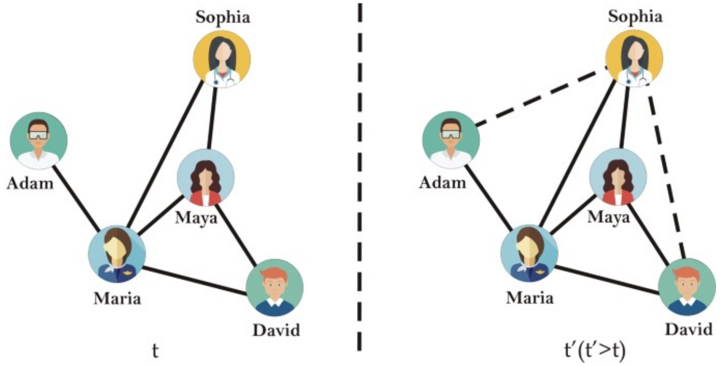


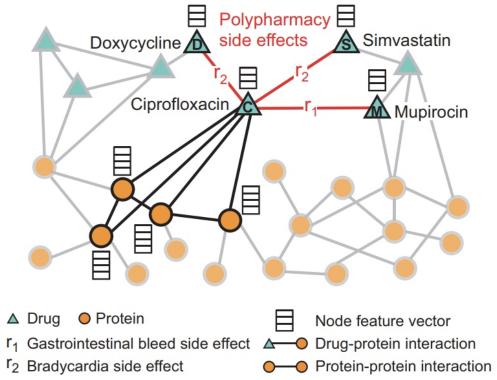
- Objective: build an autonomous football-team agent
  - Players = nodes
  - Player interactions = edges

[Learning to Play Football with Graphs and Reinforcement Learning/]

Discover relational structure

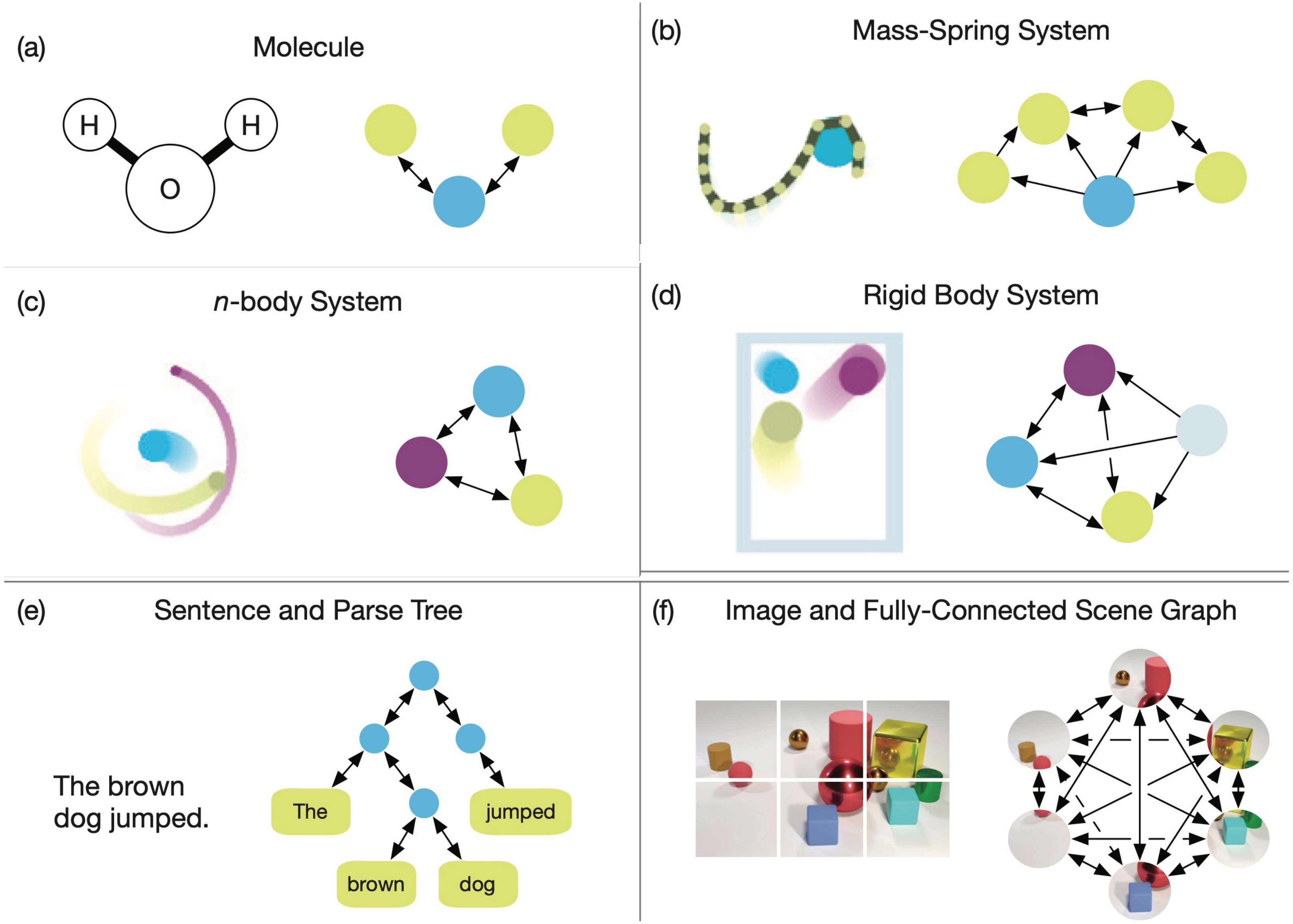
- Node/edge attribute prediction (e.g. drug side effects)
- Link prediction





- Node prediction (Object discovery)
- Learn optimal graph structure for downstream task

Introduction to graph networks



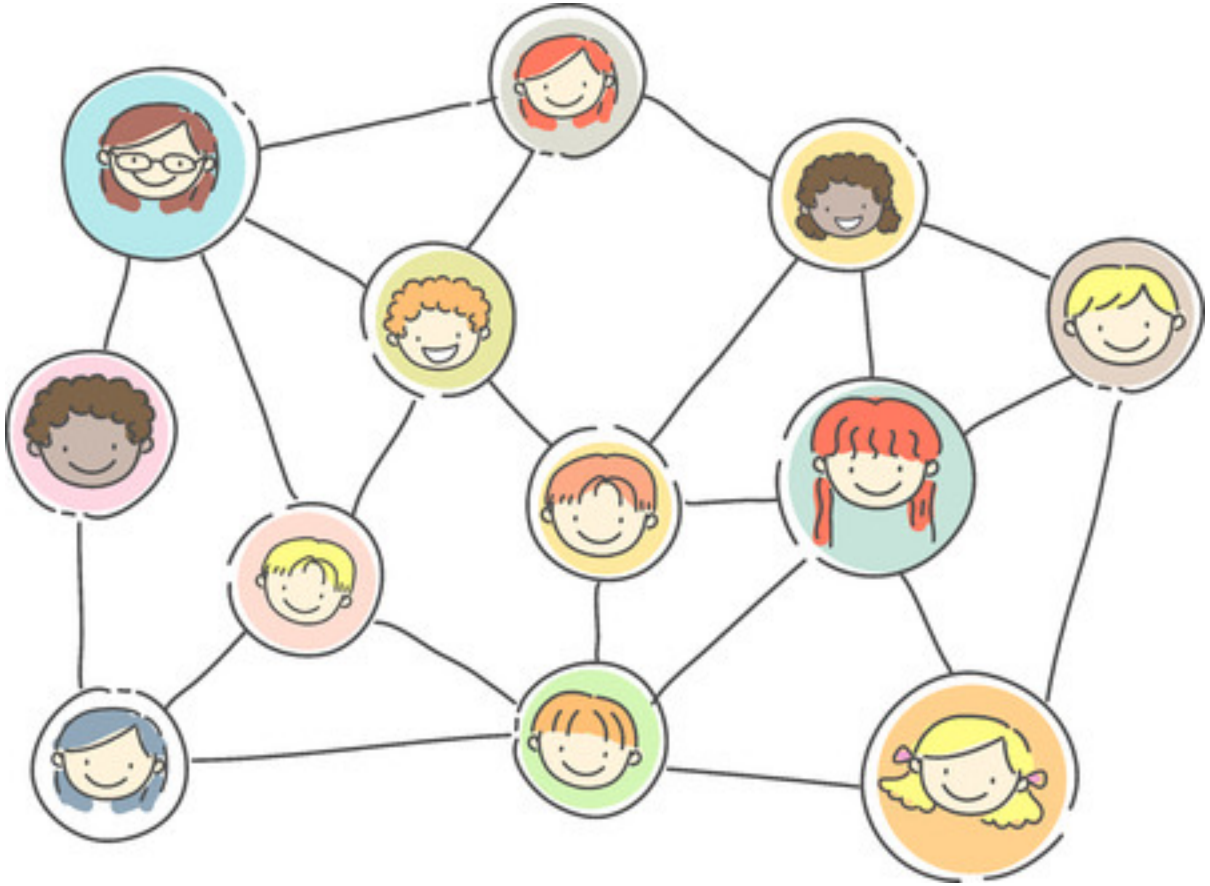
"Relational inductive biases, deep learning, and graph networks"

[1806.01261]

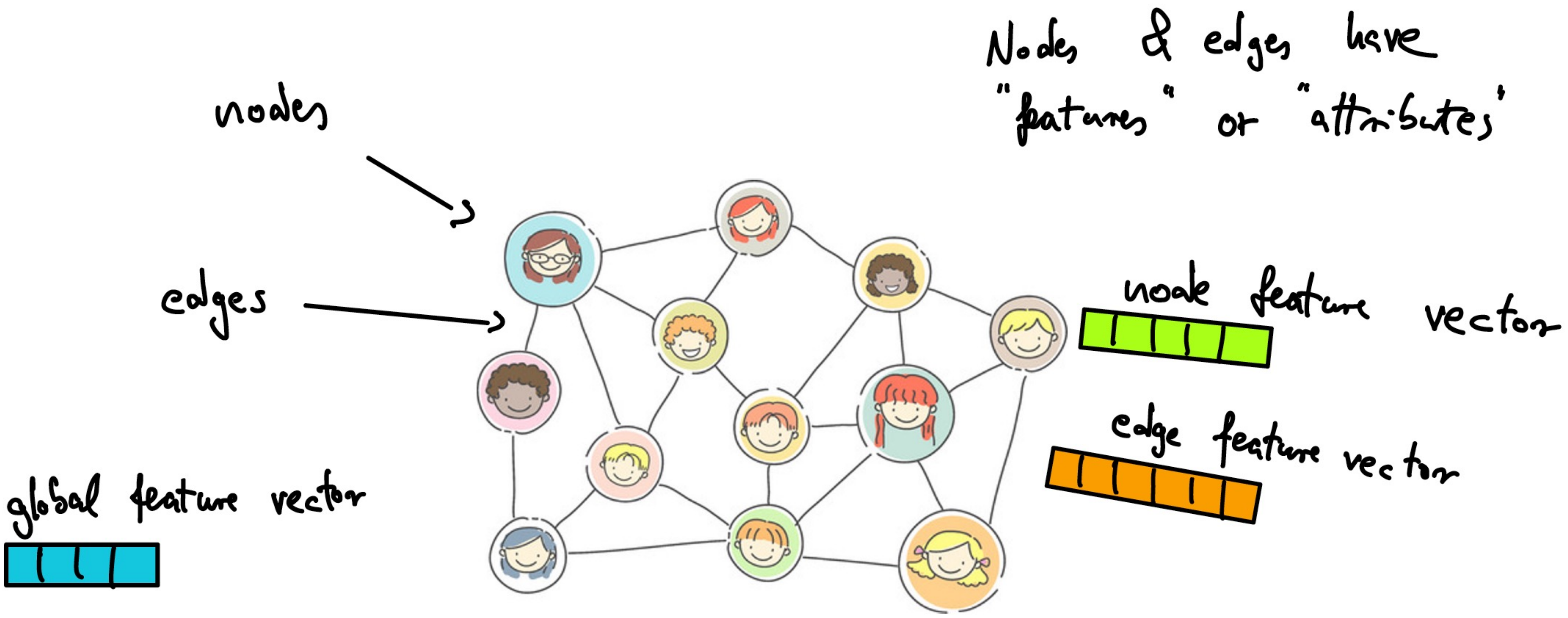
Graphs as real-world data representation

- Most real-world data is
  - Unordered
  - Variable-size

- Examples
  - Social networks
  - Molecules
  - Particles in a LHC collision
  - Planets in a solar system
  - Transportation networks
  - Covid-19 patients
  - …



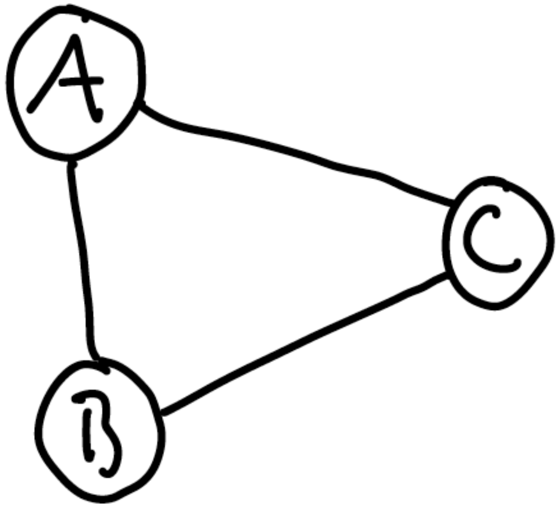
Graphs: very flexible data format



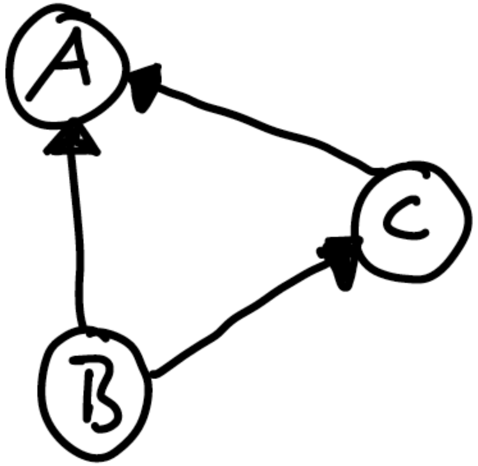
Order of nodes / edges is arbitrary

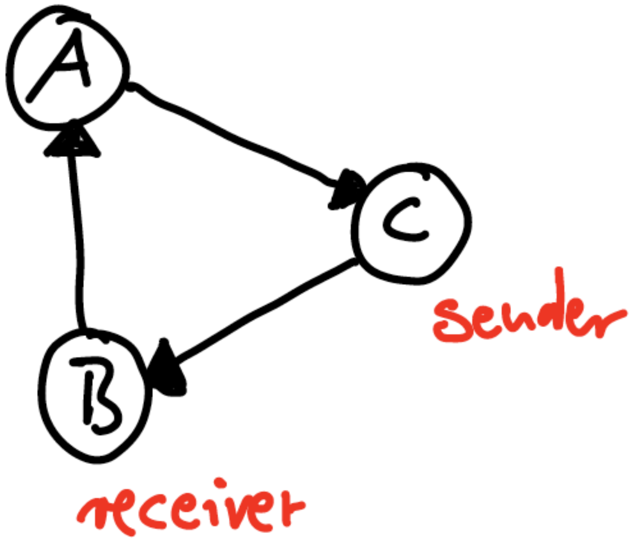
Directed and undirected graphs

undirected edges



directed edges



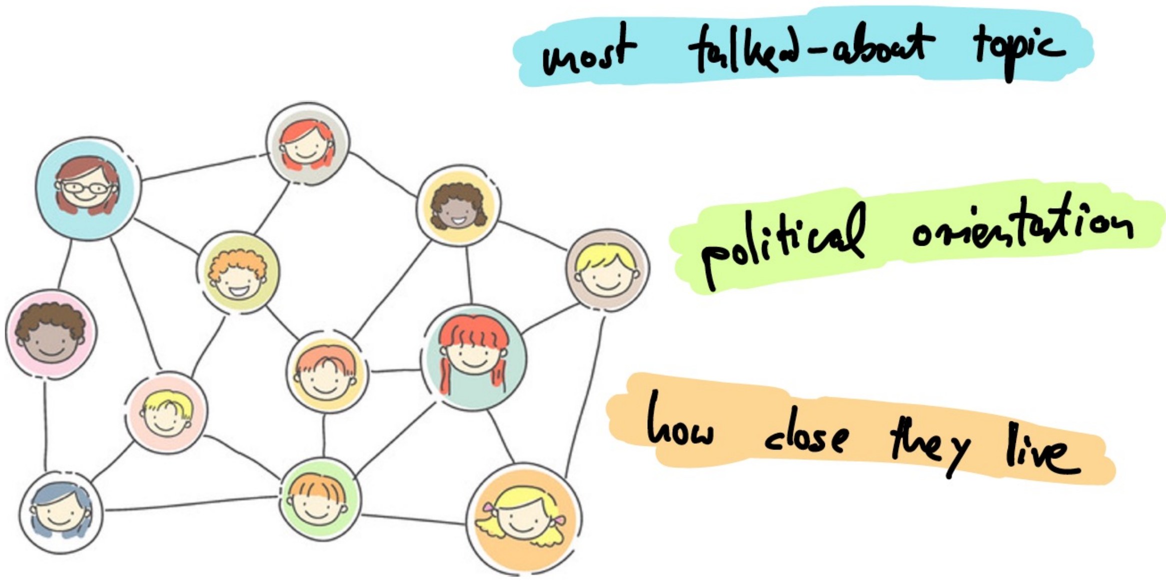


Example: social network

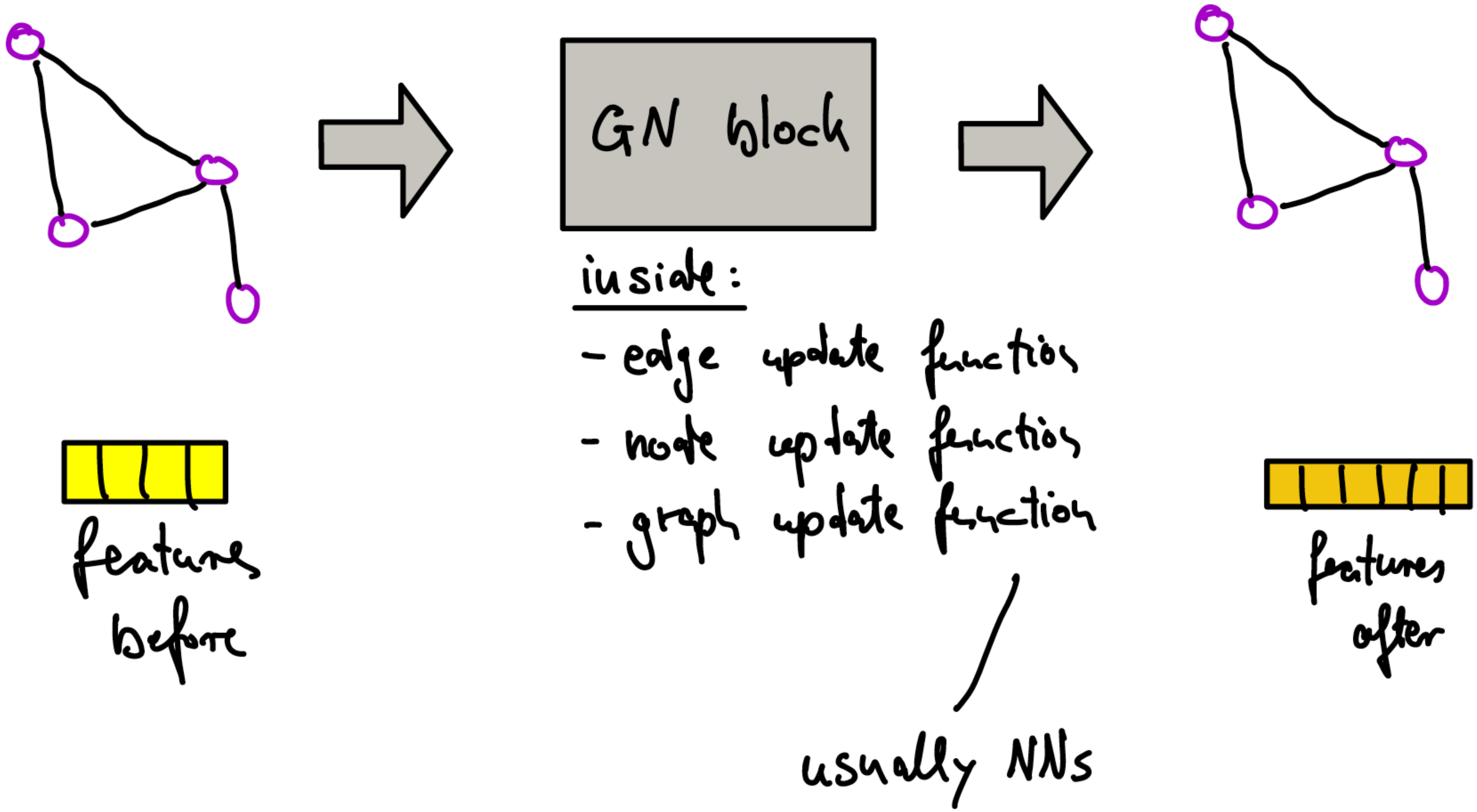
Example: flow of information

Graphs as inputs to a NN

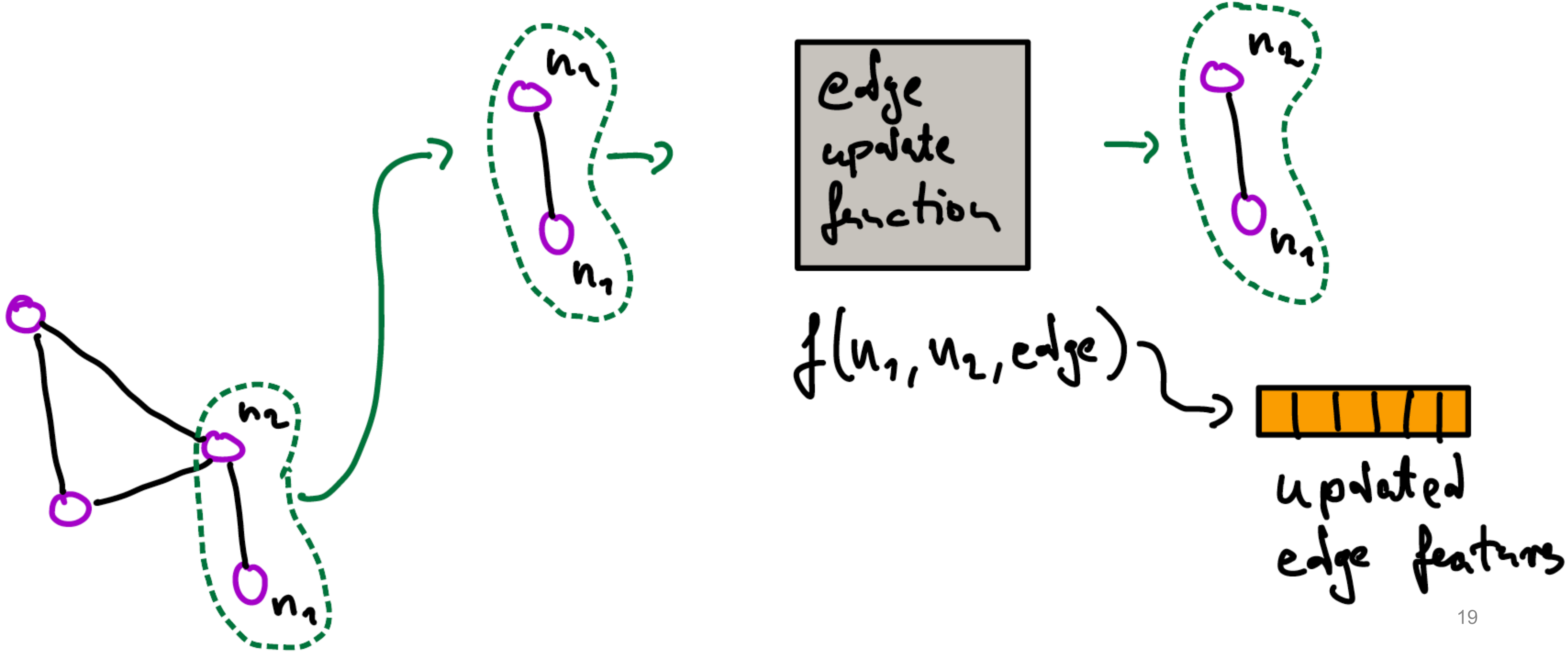
- Neural net: f(x) → y'
  - Minimize cost J(y,y') with gradient descent and back-prop,…
- Tasks at level of:
  - Whole graph
  - Each node
  - Each edge
  - Or any combination of those
- Classification, regression, segmentation,…



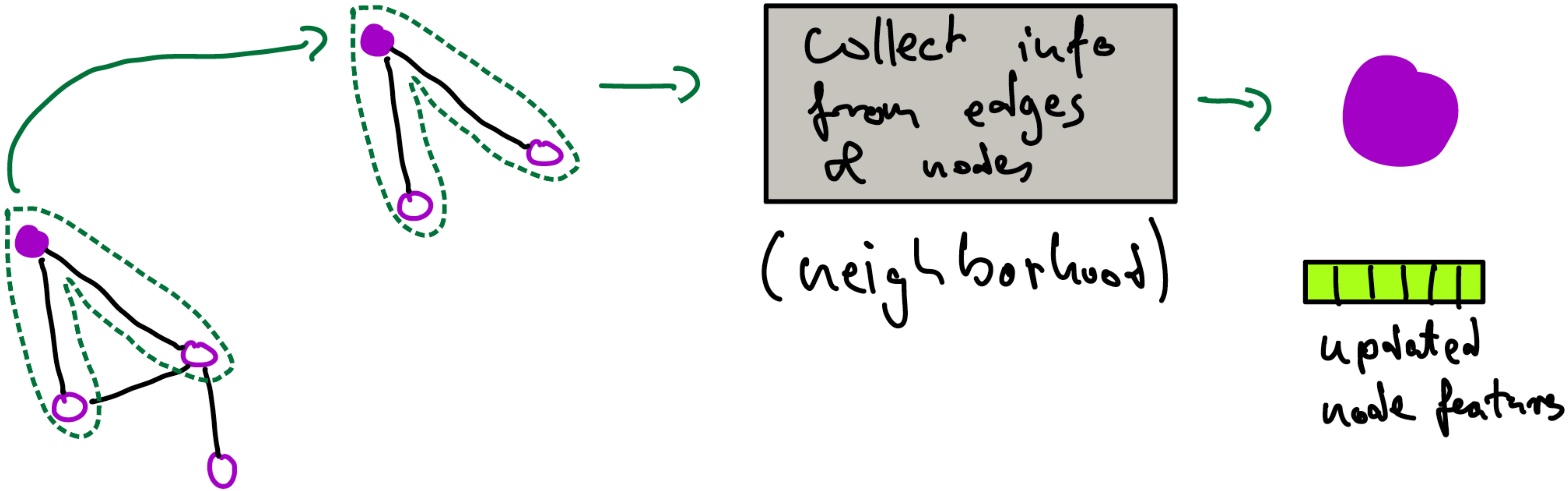
The GN block


GN block
inside:
- edge update function
- node update function
- graph update function
usually NNs
features before
features after

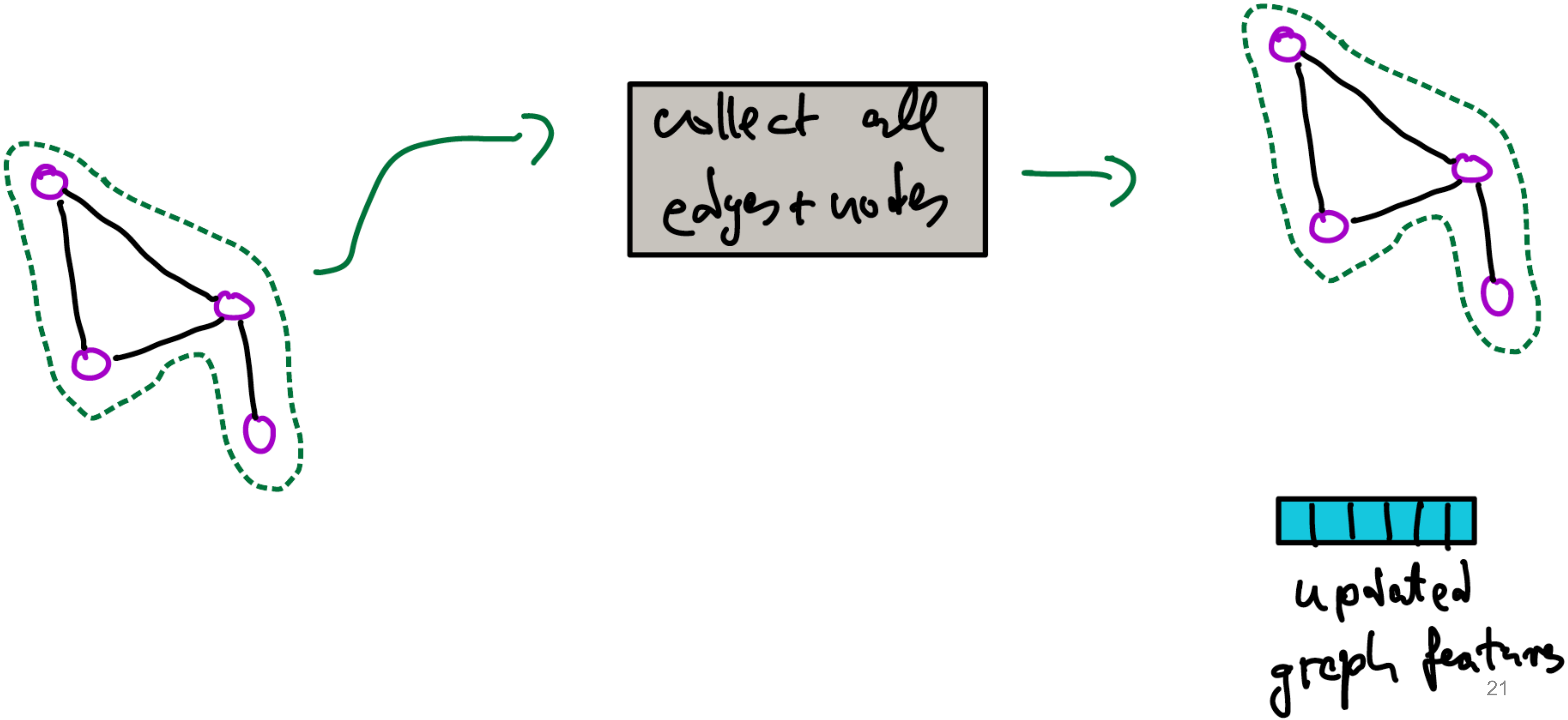
Edge update function in GN block



Node update function in GN block



Graph update function in GN block

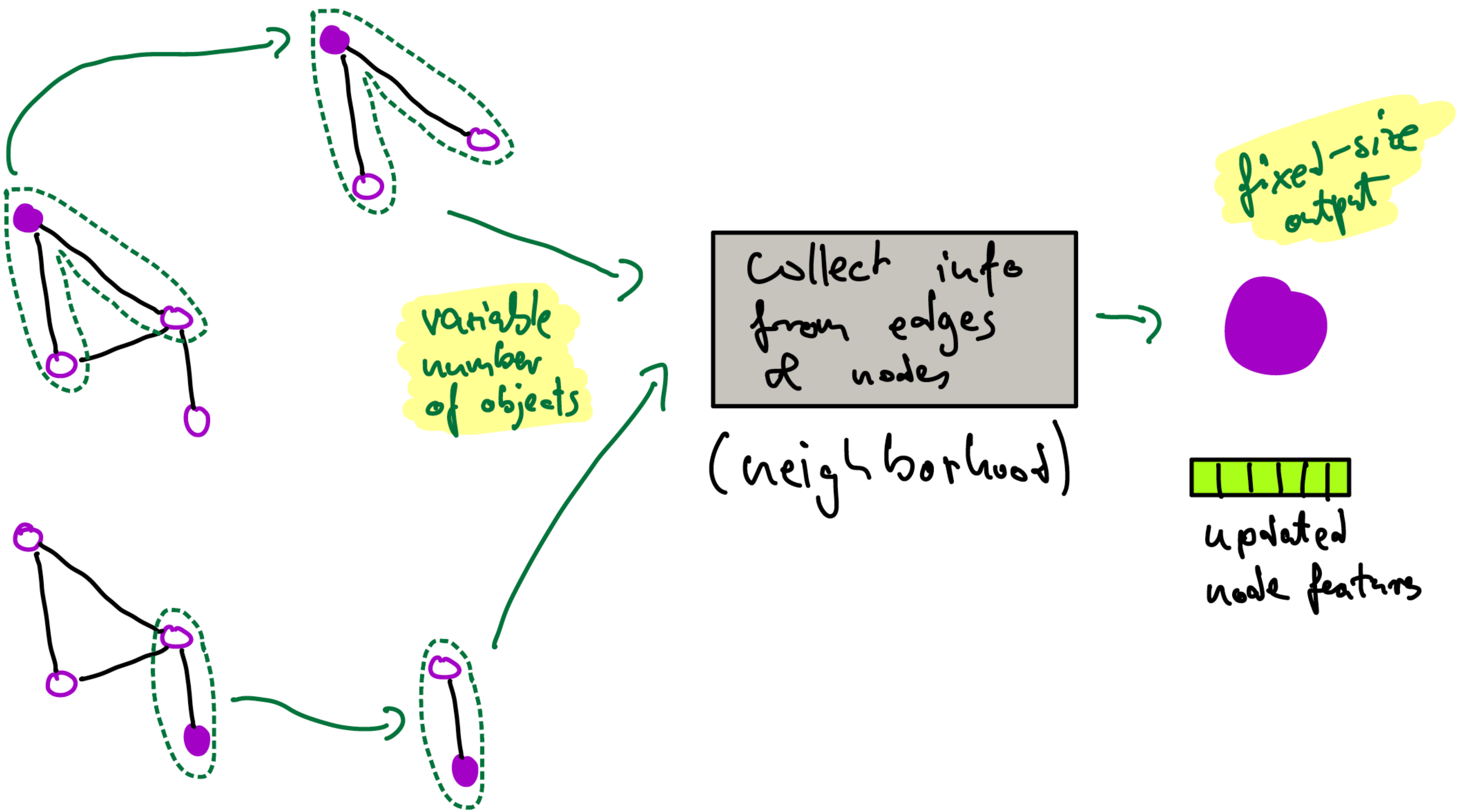


Building block: aggregate information

Variable-size inputs → **Collect or aggregate information** → Fixed-size output

Independent of order of inputs

Aggregate information: node update



What function does the trick?

Desired properties

- Variable-size input & fixed-size output
- Independent of order of inputs

Sum

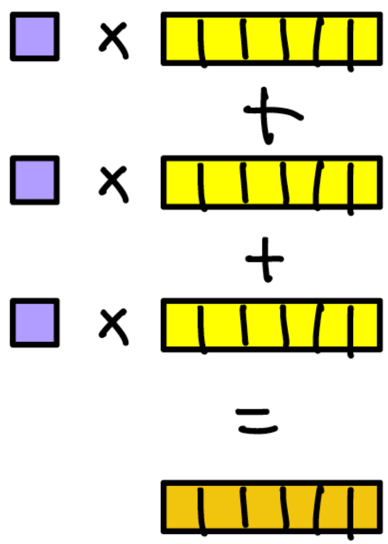
Mean

Max/Min

Most advanced: weighted sum

Attention weights, i.e. importance to task at hand

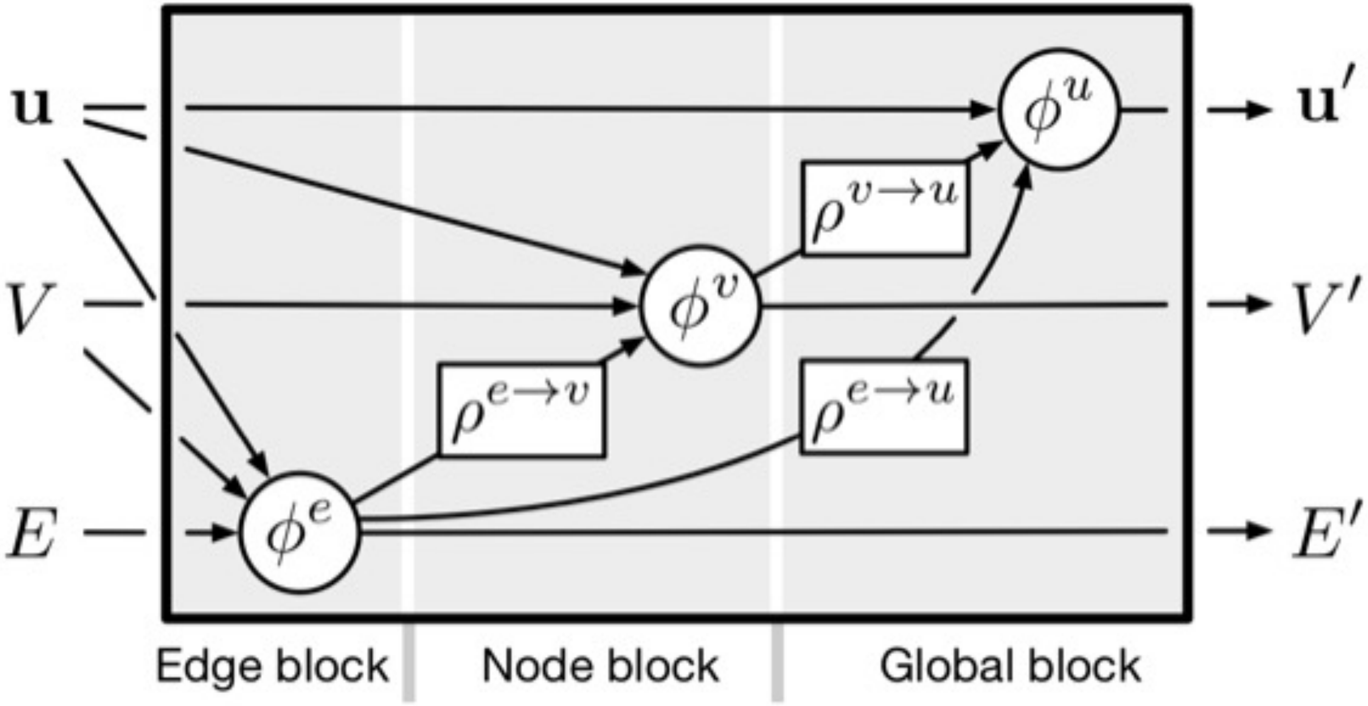
→ Transformers



The complete GN block

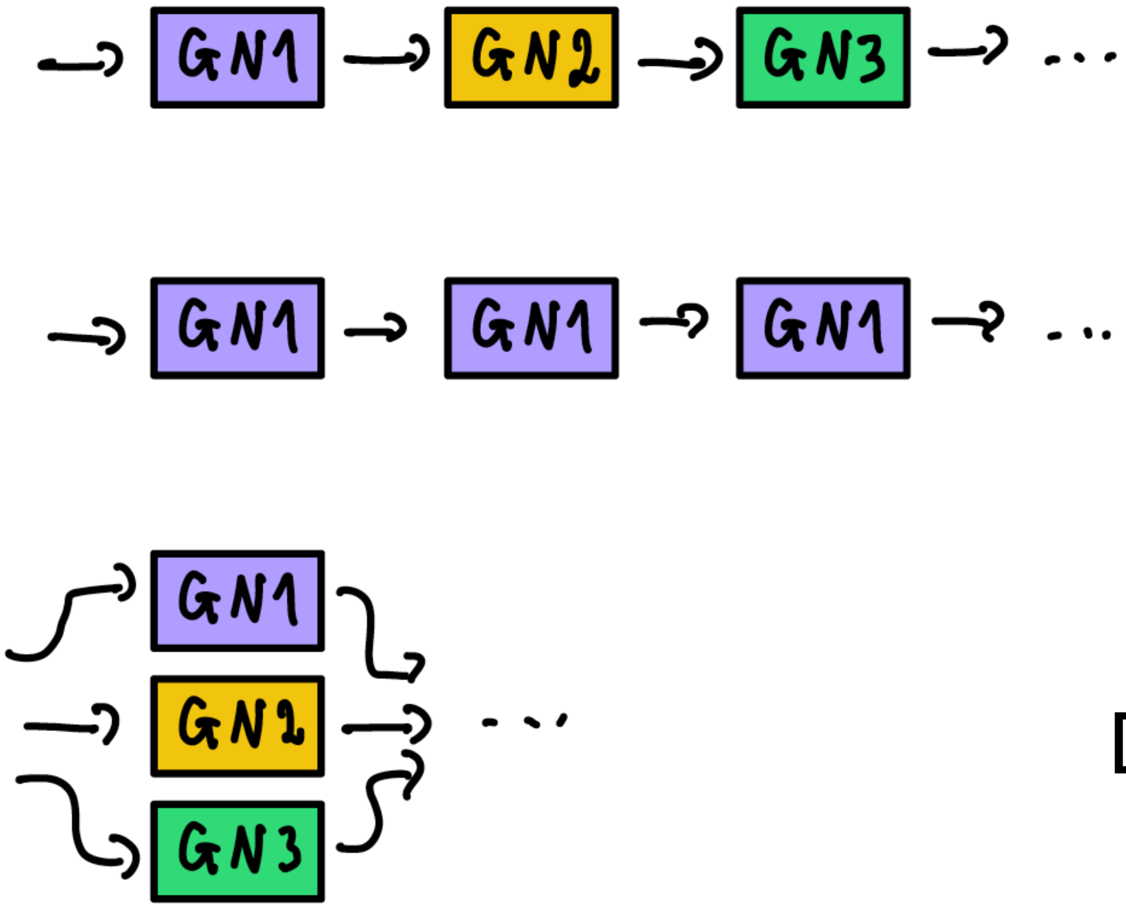
Variable-size inputs

Fixed-size output,
invariant to order of inputs



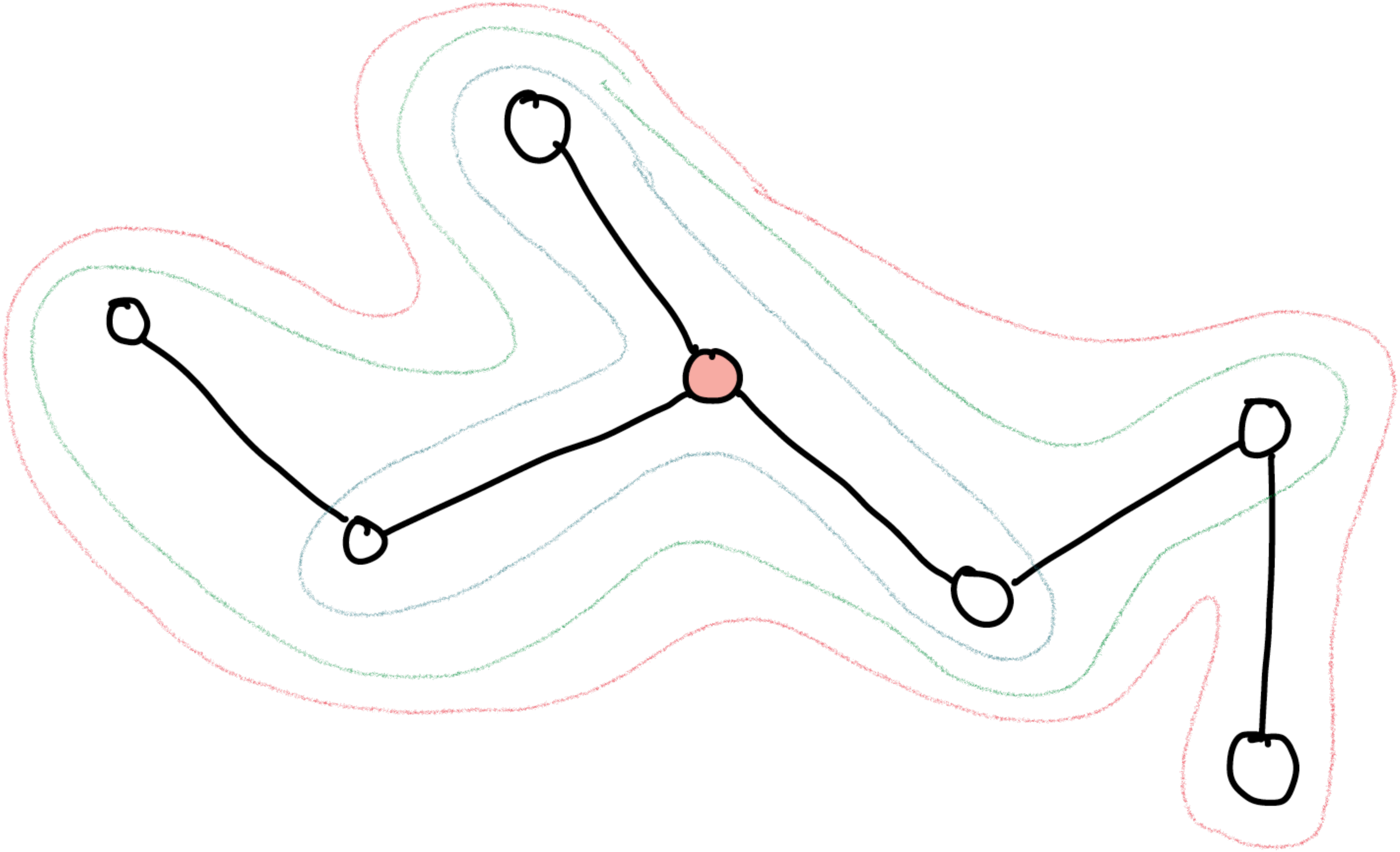
[https://arxiv.org/abs/1806.01261]

Stacking GN blocks

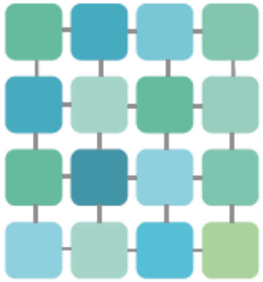


Depending on task at hand

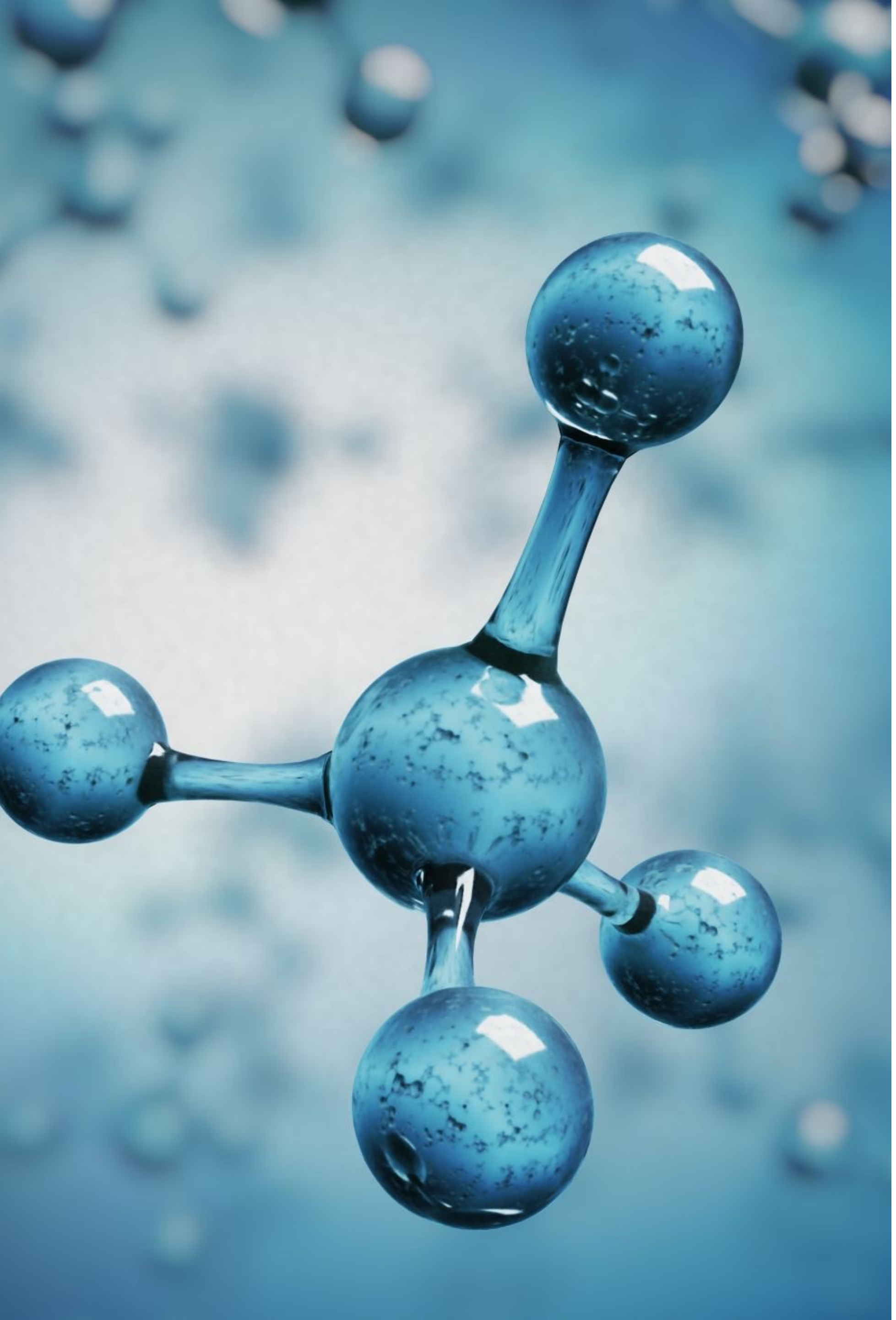
Stacking GN blocks to increase receptive field



Compare with other data representations

	UNSTRUCTURED	SEQUENTIAL	GRID	COMPLEX RELATIONAL STRUCTURE
Structure				
Model	**MLP**	**RNN**	**CNN**	**GNN**
				Arbitrary order Variable-size inputs
Inductive Bias *	**Weak**	**Sequentiality**	**Locality**	**Strong relational bias**

* Beliefs about the model and data properties. Right initial beliefs ⇒ better generalization with less data



Example applications

Properties of molecules – **graph-level regression**

Movement of N-body system – **node-level regression**

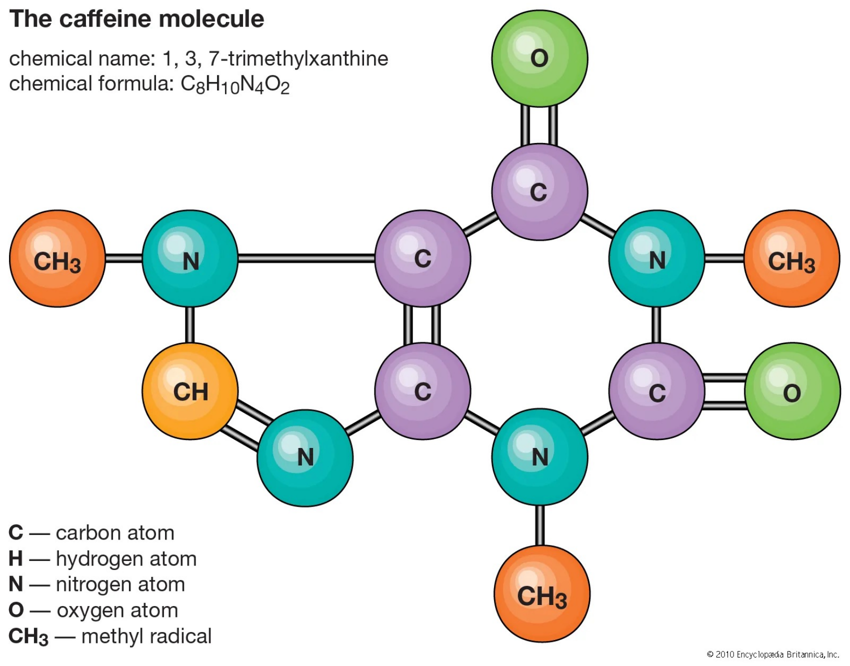
Query a knowledge graph – **more complex architecture**

Applications in particle physics

Example 1: molecules

Nodes = atoms

Edges = chemical bonds



Neural Message Passing for Quantum Chemistry

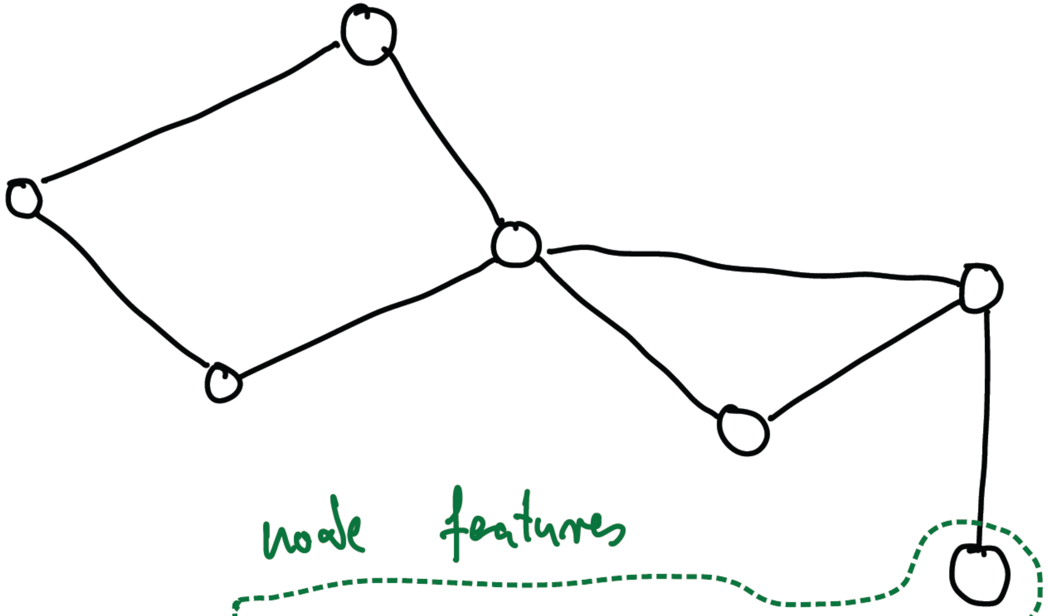


Table 1. Atom Features

Feature	Description
Atom type	H, C, N, O, F (one-hot)
Atomic number	Number of protons (integer)
Acceptor	Accepts electrons (binary)
Donor	Donates electrons (binary)
Aromatic	In an aromatic system (binary)
Hybridization	sp, sp2, sp3 (one-hot or null)
Number of Hydrogens	(integer)

Training data:

134k drug-like organic molecules that span a wide range of chemistry

[https://arxiv.org/abs/1704.01212]

Graph-level regression task

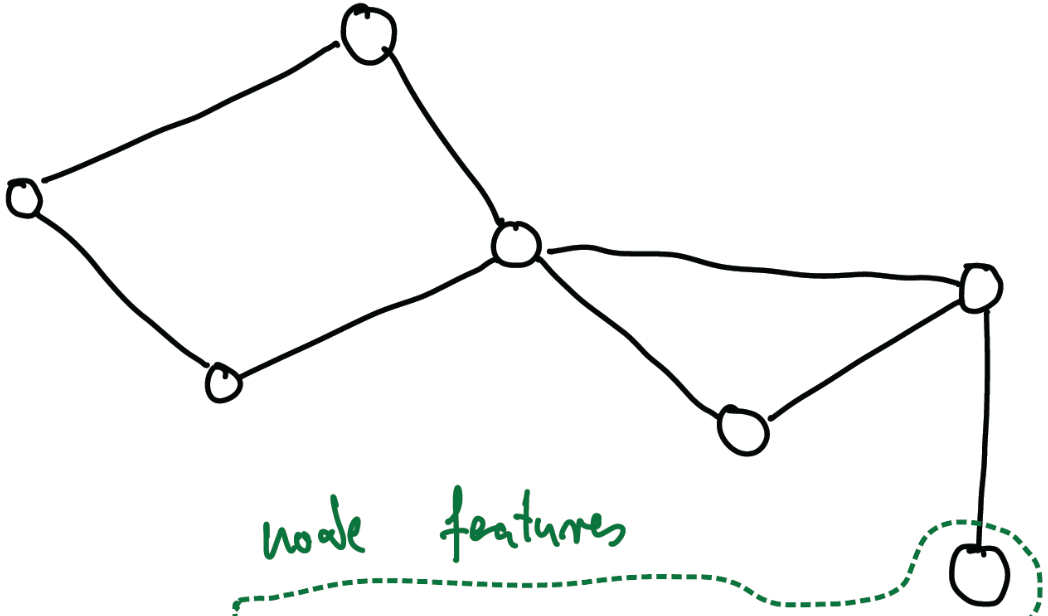


Table 1. Atom Features

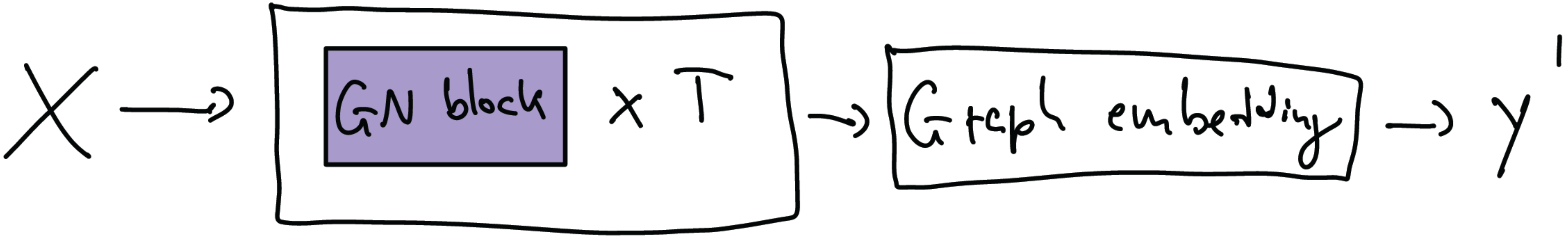
Feature	Description
Atom type	H, C, N, O, F (one-hot)
Atomic number	Number of protons (integer)
Acceptor	Accepts electrons (binary)
Donor	Donates electrons (binary)
Aromatic	In an aromatic system (binary)
Hybridization	sp, sp2, sp3 (one-hot or null)
Number of Hydrogens	(integer)

Target:

- Atomisation energy
- Vibrational frequencies
- Etc.

[https://arxiv.org/abs/1704.01212]

Architecture used



$3 \leq T \leq 8$

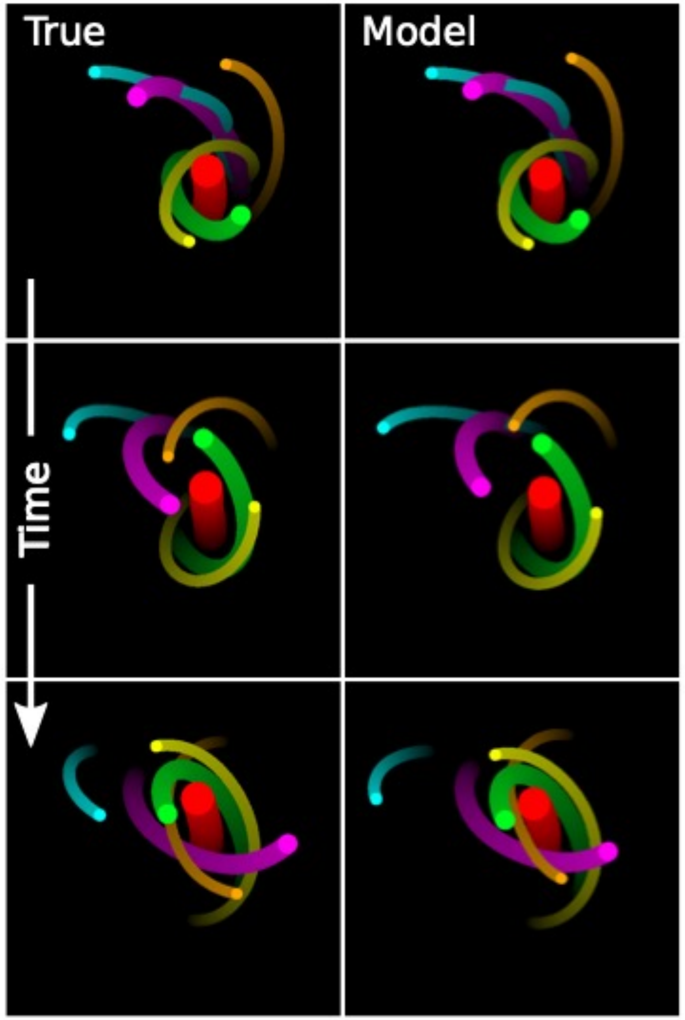
number of iterations T
is a hyperparameter

[https://arxiv.org/abs/1704.01212]

Example 2: learn dynamics of physical systems

- Nodes = planets
  - Position
  - Velocity vector
  - Mass

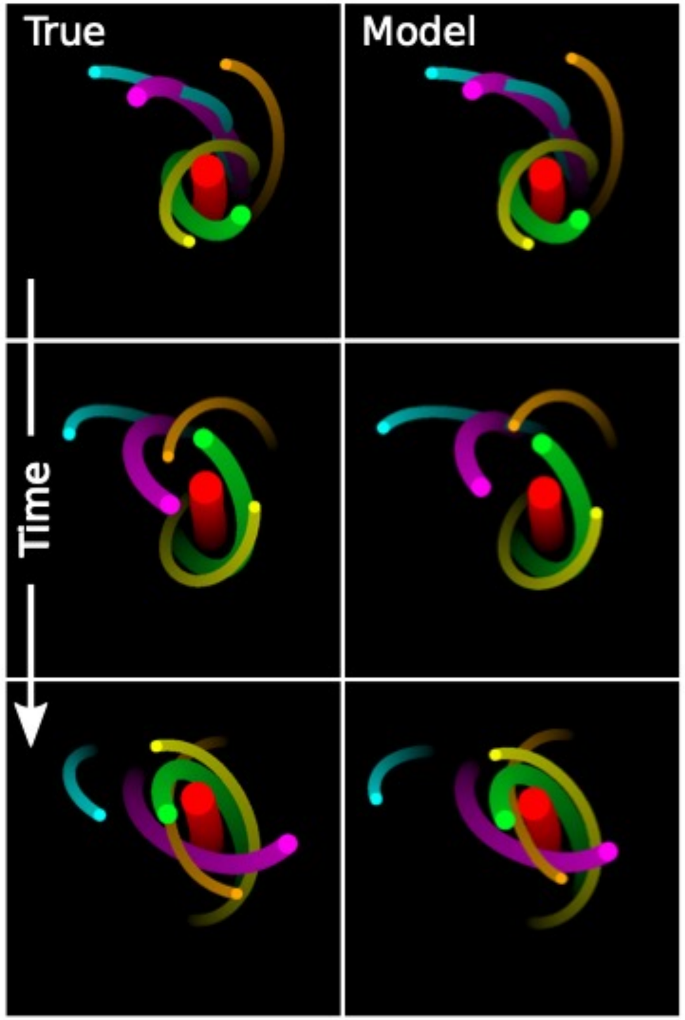
- Edges = “fully connected”
  - Gravitational force between any 2 planets



[https://arxiv.org/abs/1612.00222]

Node-level regression task

- Predict nodes features after time t
  - Dynamics of system
- Train on N=6 planets
  - Generalize to e.g. N=12 planets
- Edge function:
  - Compute distance of planets
- Node function:
  - Masses + distances ⇒ force
  - Force + current velocity + mass ⇒ future position



[https://arxiv.org/abs/1612.00222]

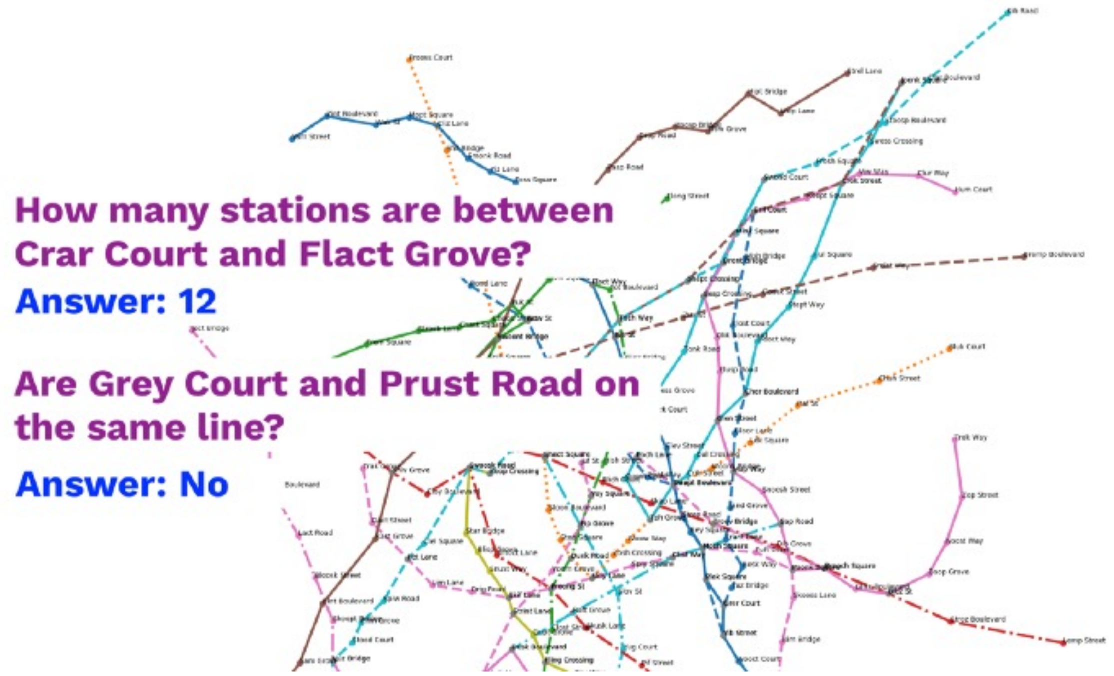
Example 3: knowledge graph

Transport network

Node = metro station

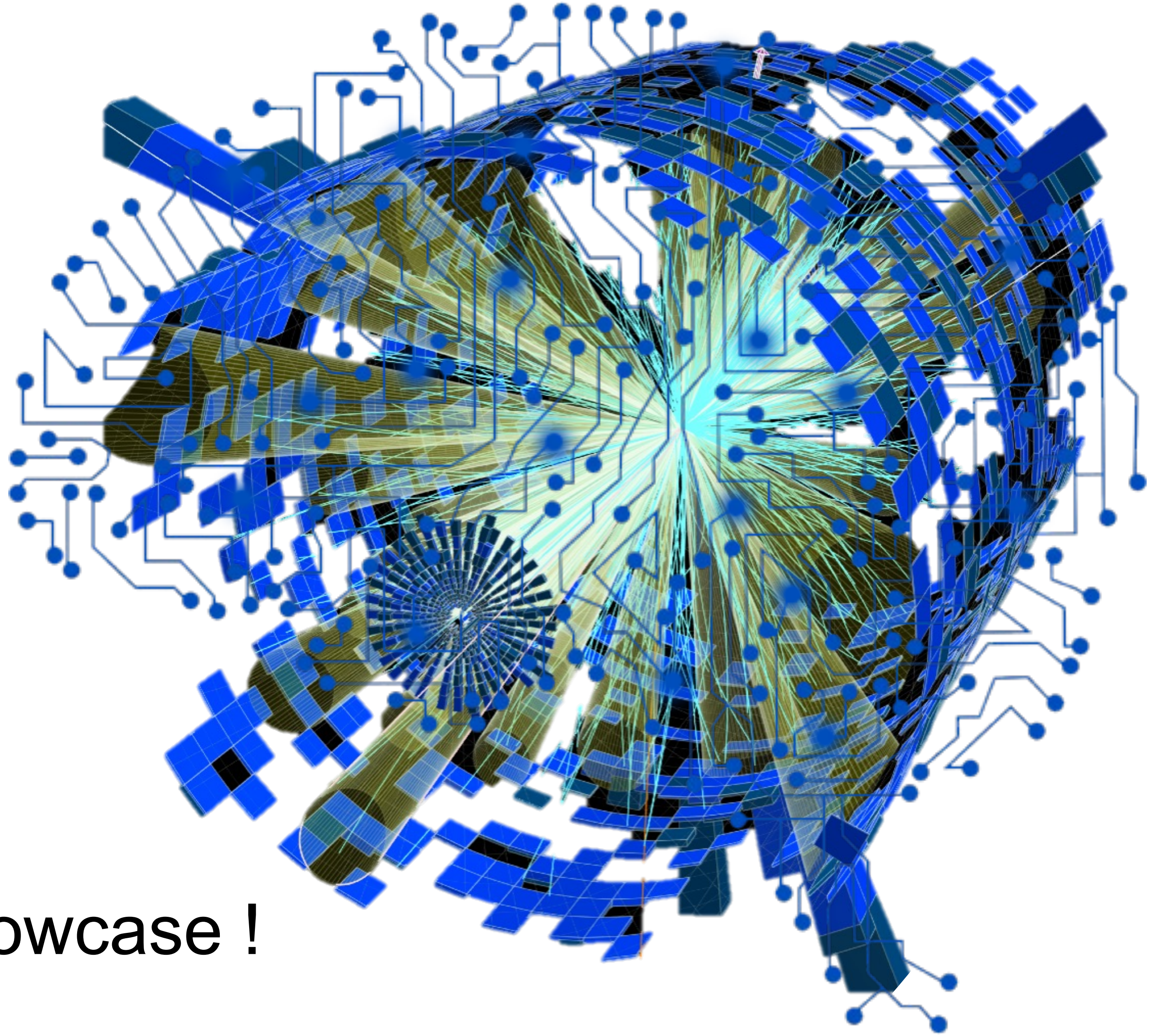
Edge = connecting line

Embed graph in natural language processing



[https://medium.com/octavian-ai/clevr-graph-a-dataset-for-graph-based-reasoning-5e4e64f28ffb]

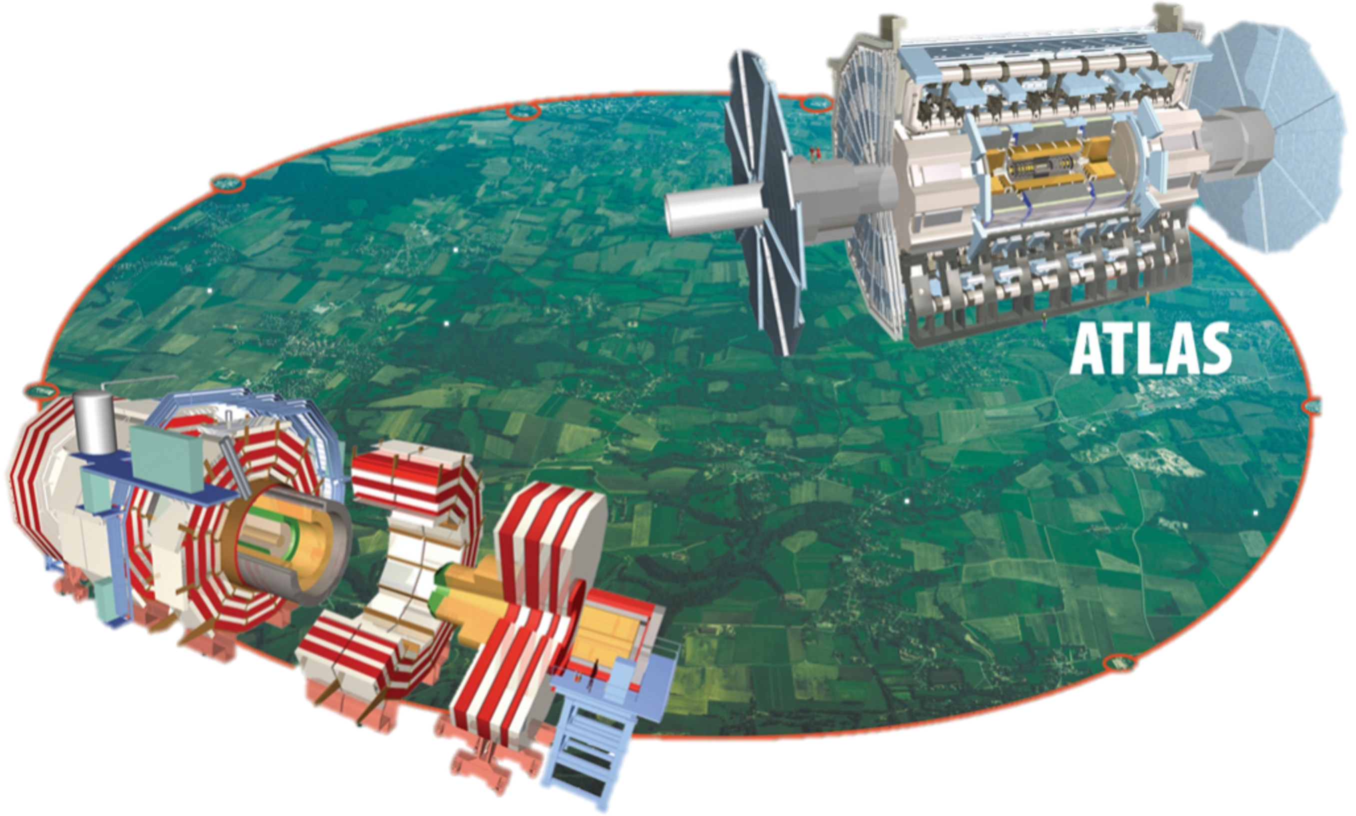
GNNs in particle physics: the LHC



Your lecturer's expertise

And it's a great GNN showcase !

LHC in a nutshell



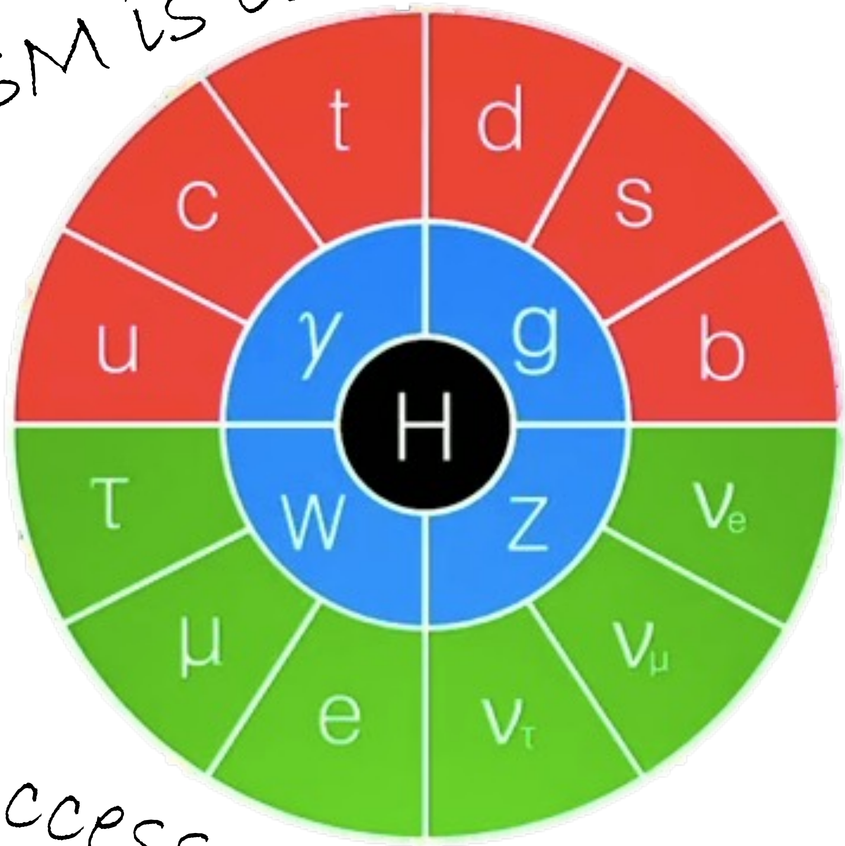
Collide high-energy (HE) particles in controlled environment

Hermetic detectors around collision points to measure all produced particles

Answer fundamental questions

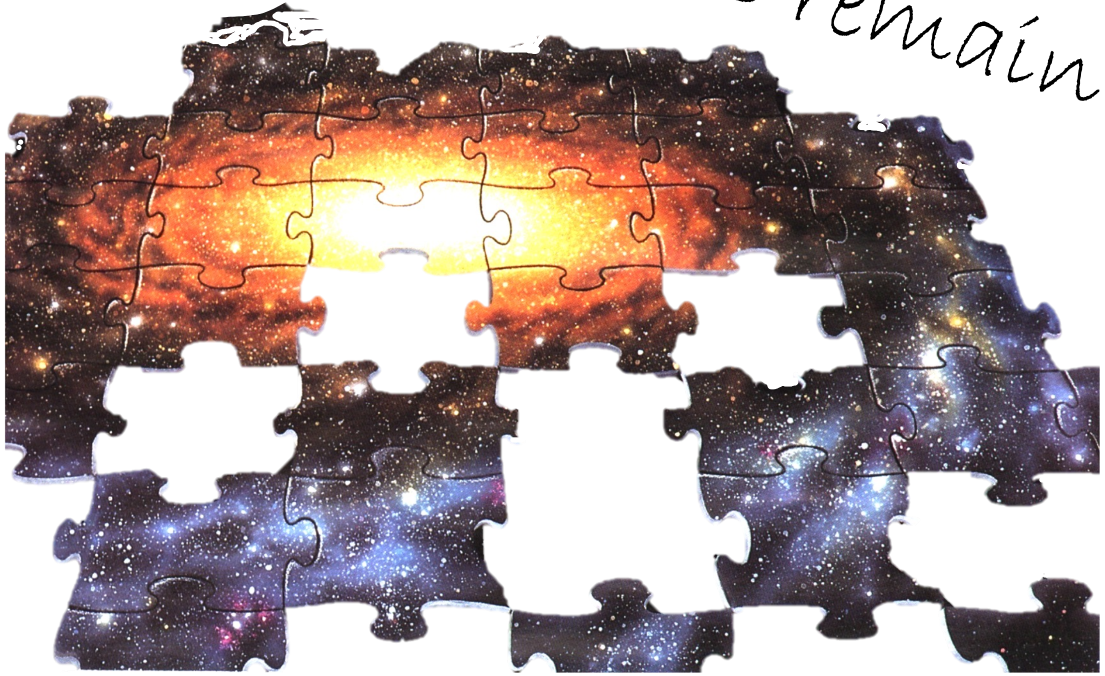
The motivation

The SM is complete



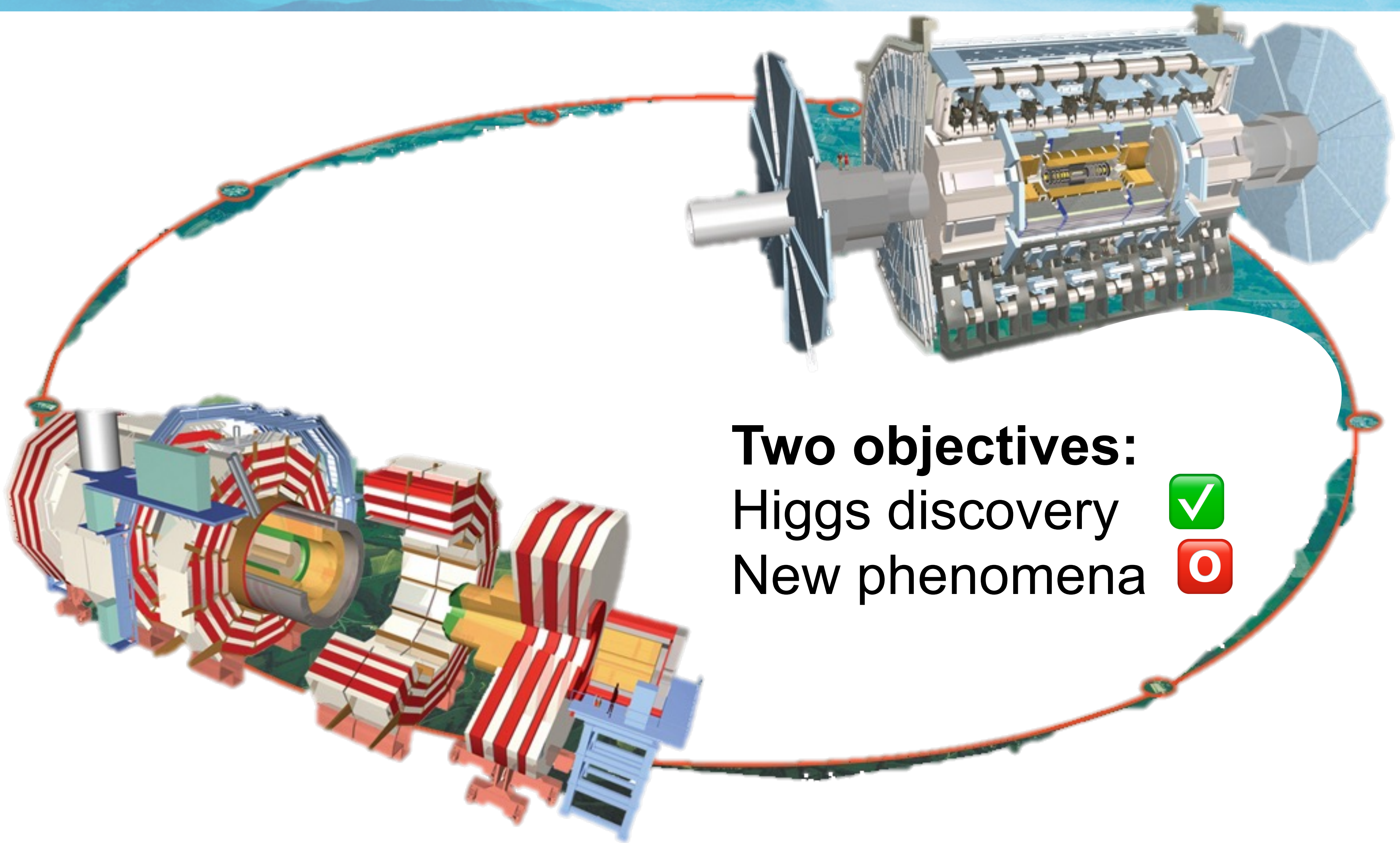
Success story!

Open mysteries remain



Dark matter, dark energy, quantum gravity,...

The Large Hadron Collider (LHC)

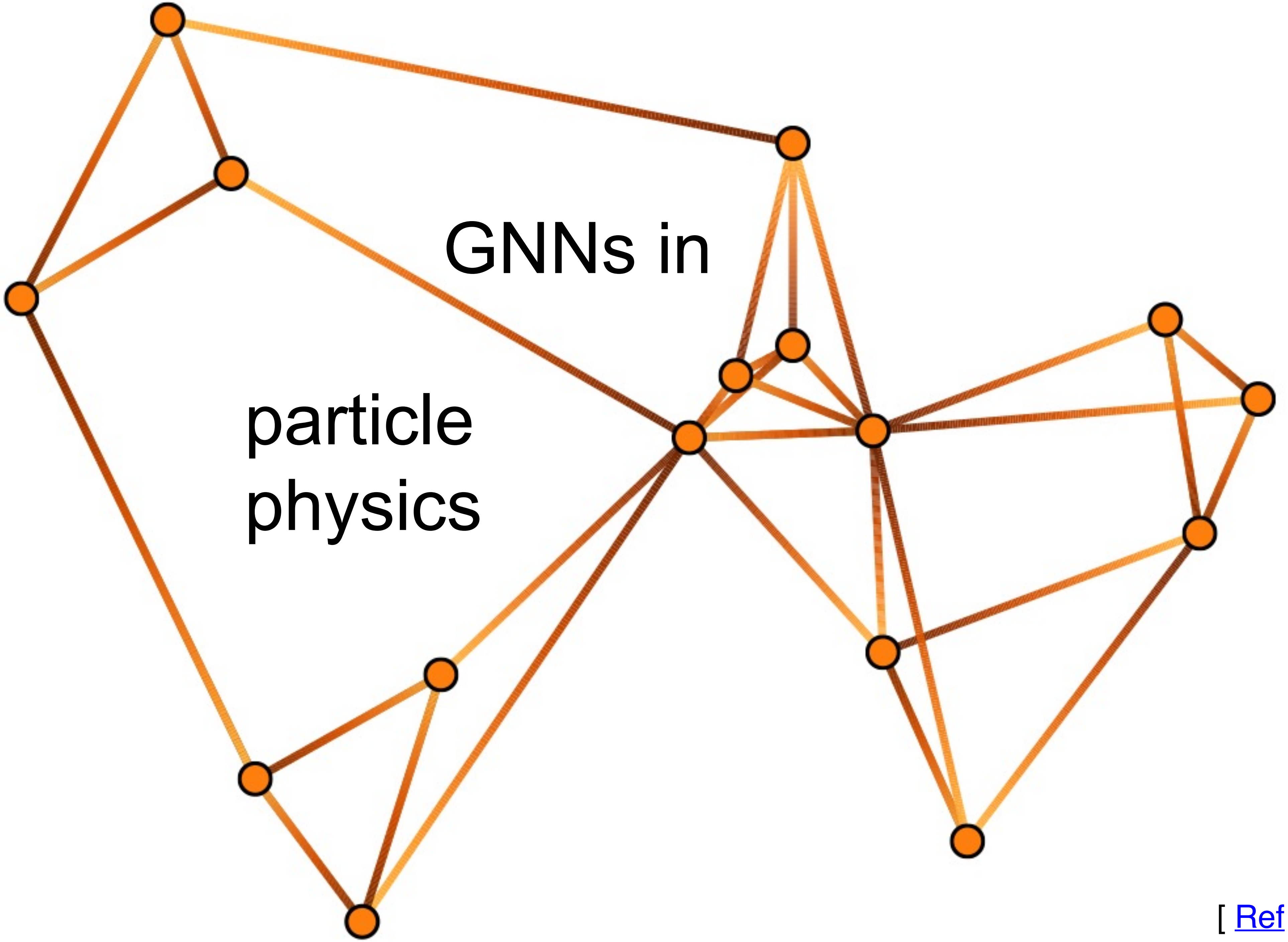


Two objectives:

Higgs discovery ✅

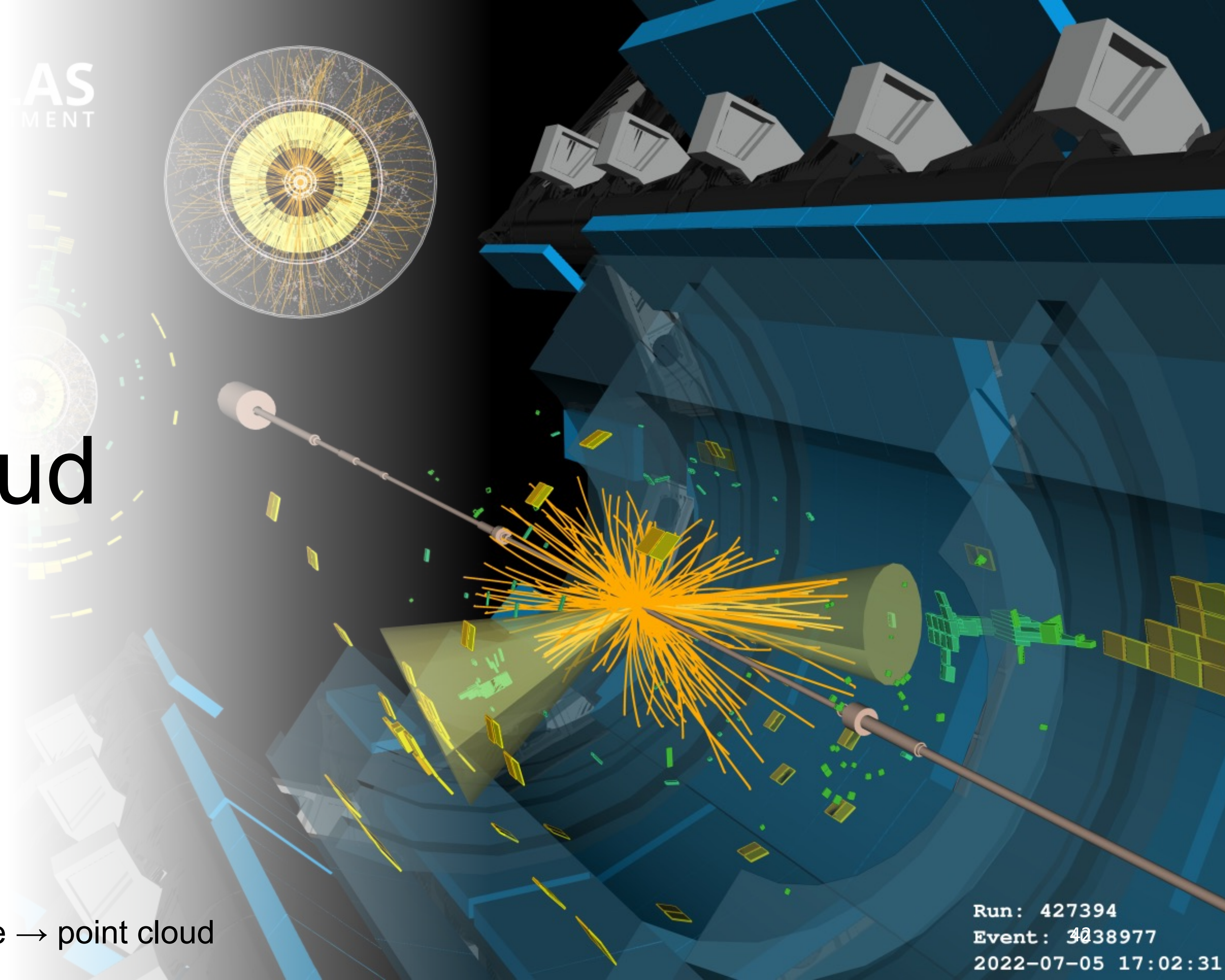
New phenomena ⭕

GNNs in particle physics



[Ref]

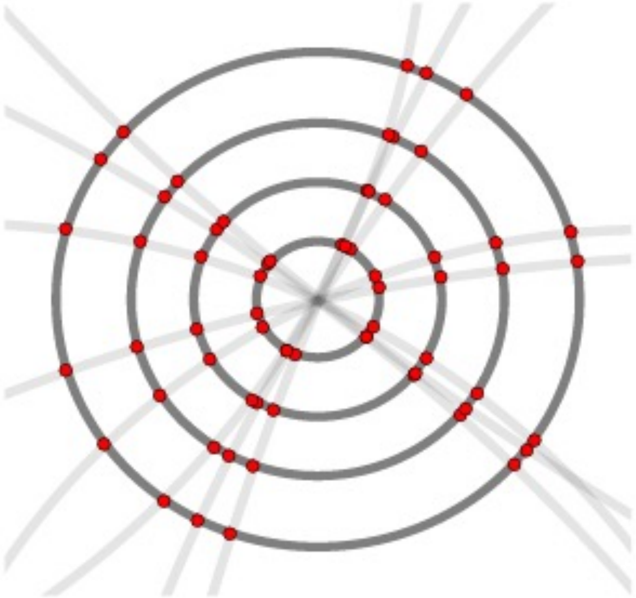
Point cloud data

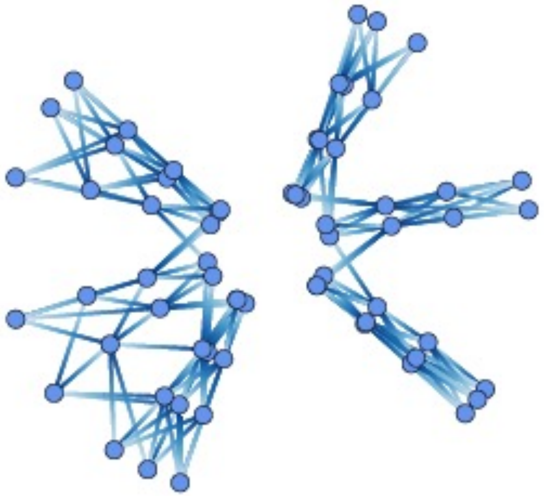


Sparse inhomogeneous 3D image → point cloud

Track reconstruction = edge prediction

Connecting the dots

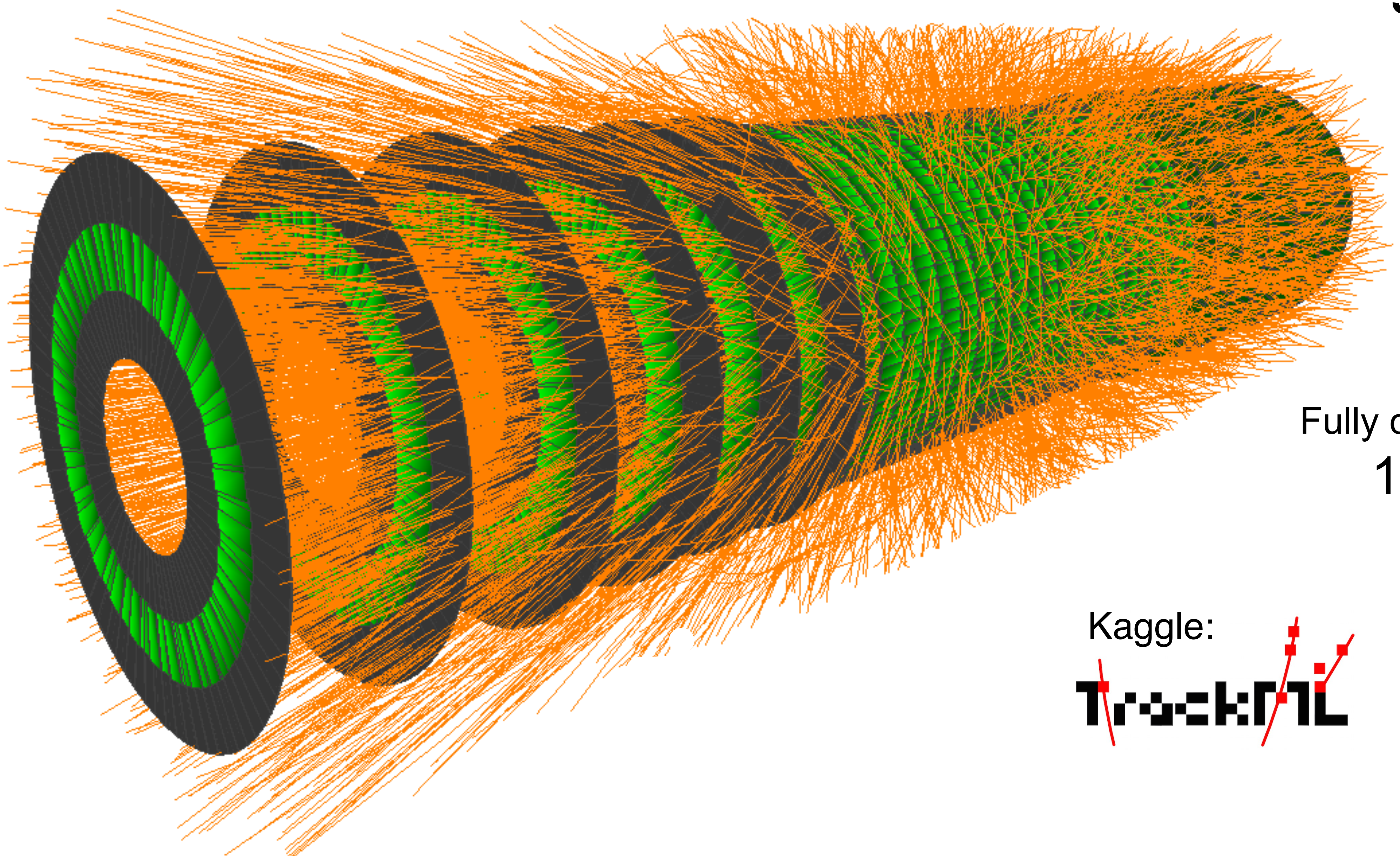




Traditional solution: Kalman filter
but too slow for HL-LHC !

[Ref, Exa.TrkX & L2IT]

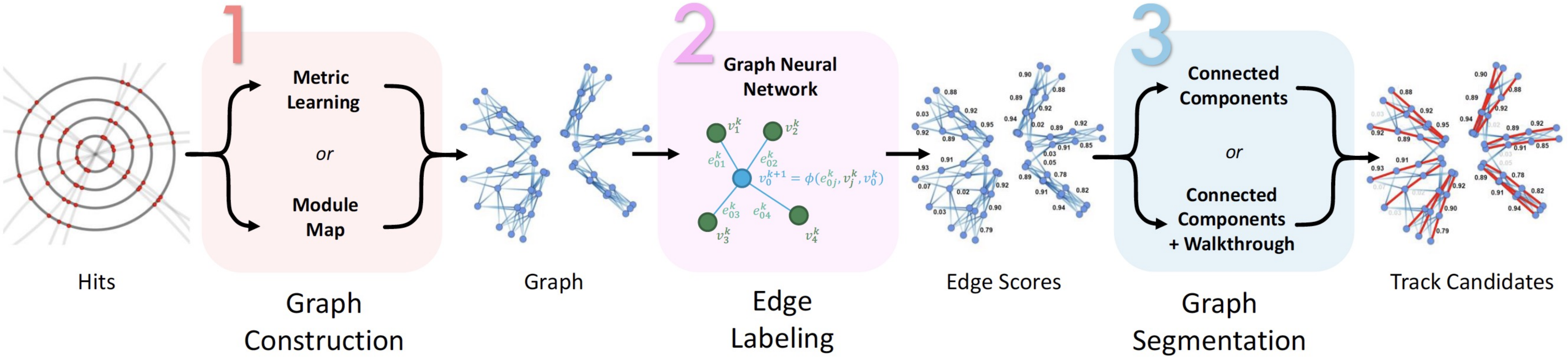
Connect 300k dots to 10k tracks using ML



Fully connected graph:
10^{10} edges

Kaggle:

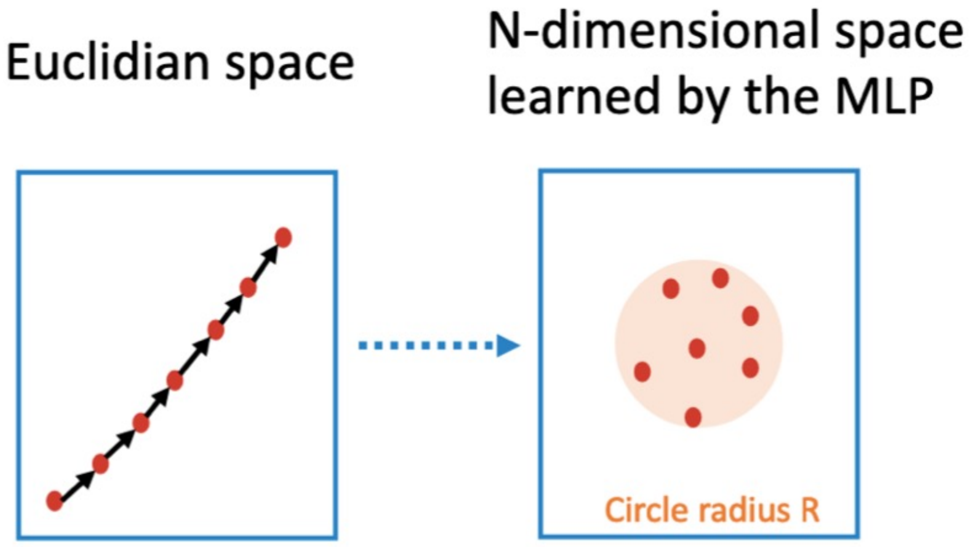
Track reco with GNN in 3 steps



[IDTR-2022-01]

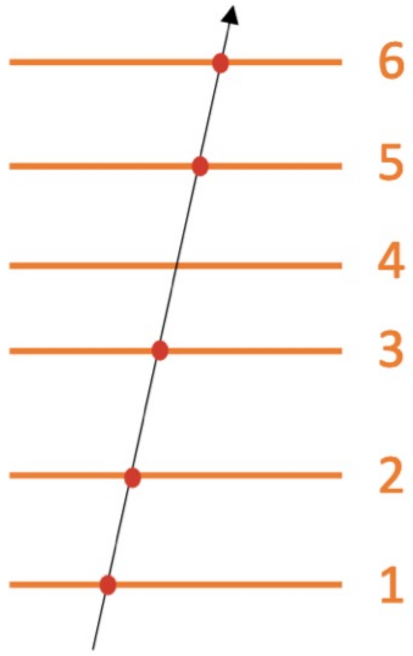
1. Graph construction

Metric learning



[TG et al., 2101.06428]

Module map

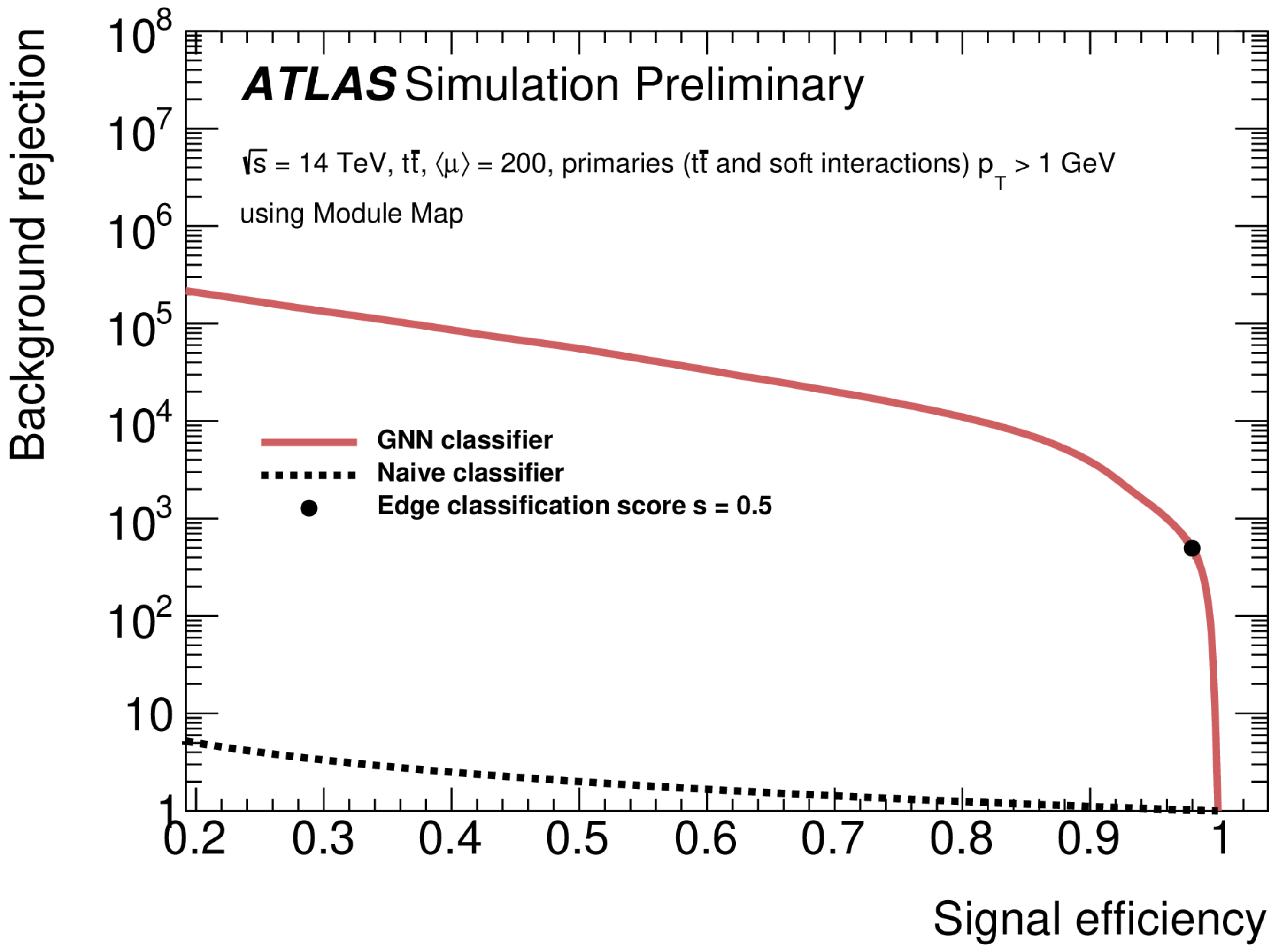


Lookup table:
possible module groups
traversed by a particle

10^{10} edges → 10^{6} edges (10^{4} true edges)

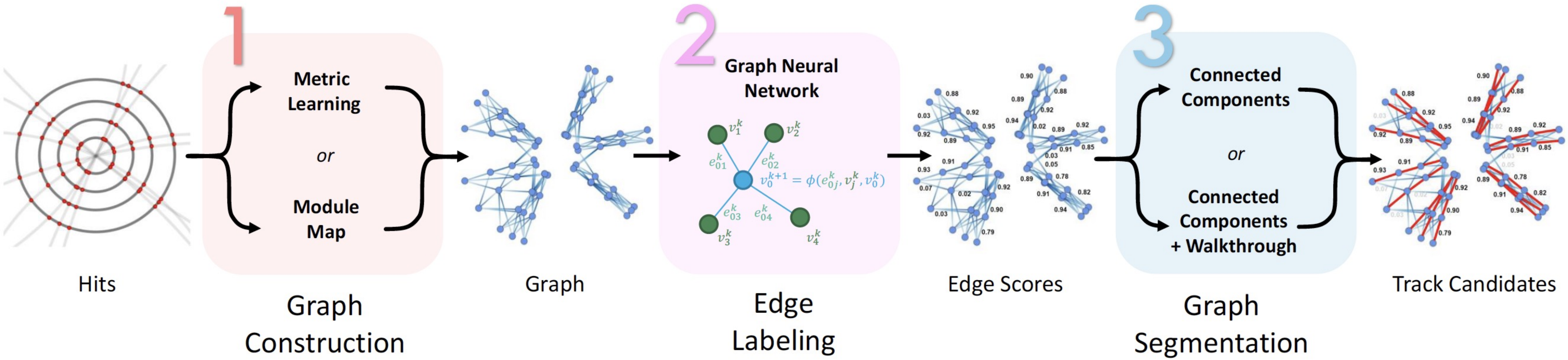
[IDTR-2022-01]

2. Edge classification



ROC curve

↗efficiency, ↗ purity, ↘ compute time



- Edge lost can't be retrieved later
- Dominated by fake edges
- Dominates compute time (0.5 s)

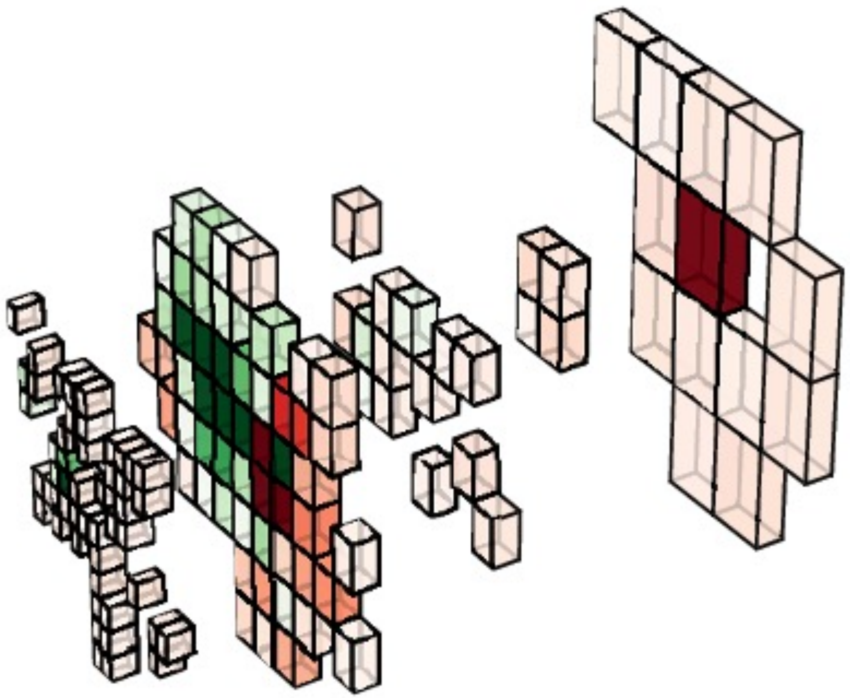
Efficiency vs. purity

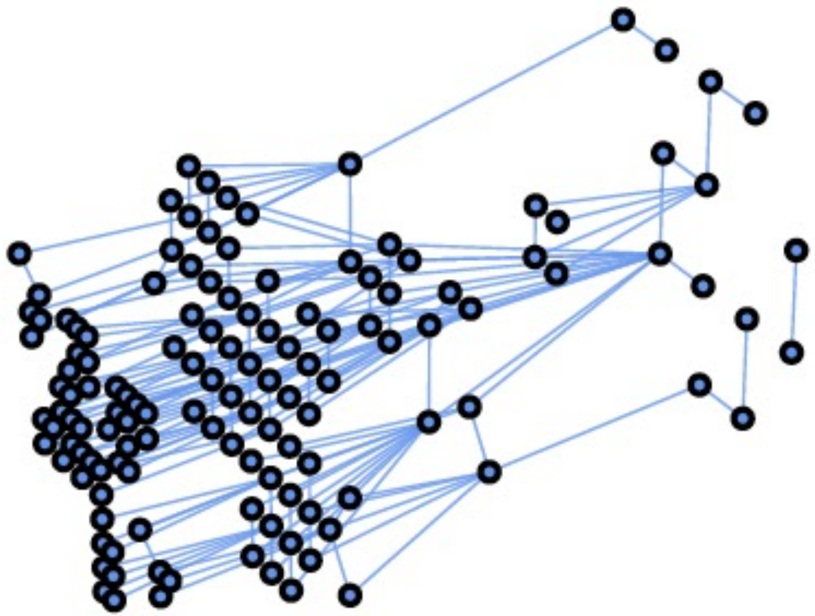
Work in progress

[IDTR-2022-01]

Jet reconstruction based on energy depositions

A *cloud of particles*



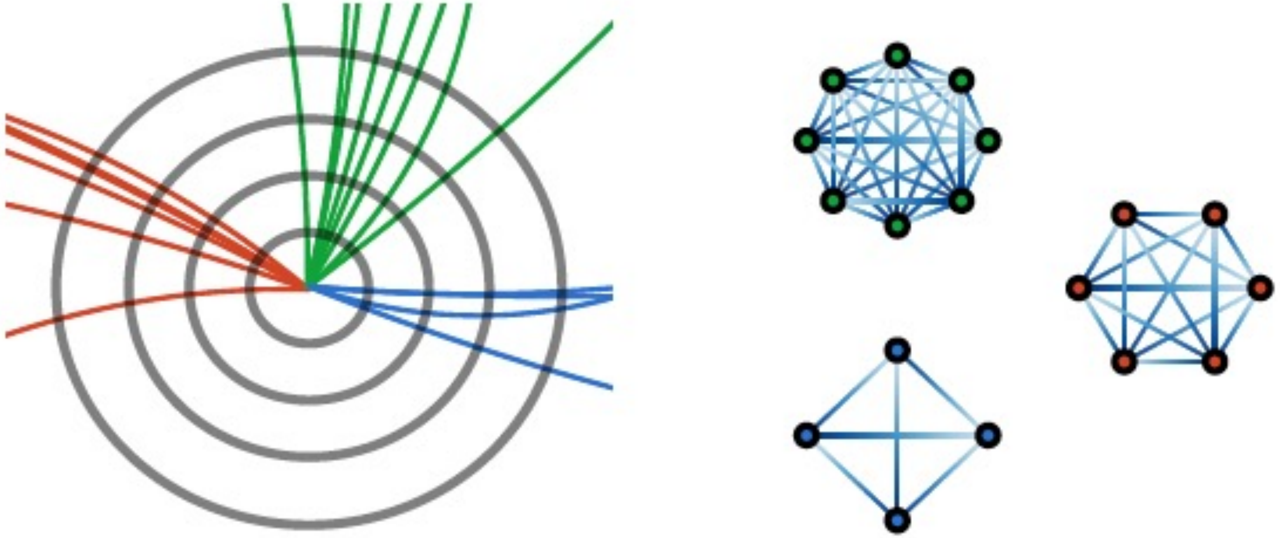


- Graph-level classification

[ParticleNet]

[Ref]

Interpreting jets based on associated particles



→ Flavor tagging – **graph-level classification**

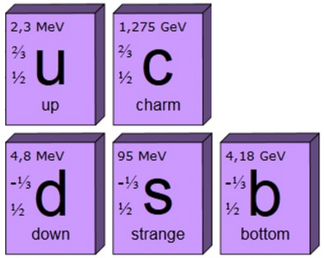
[Ref]

Flavor tagging domain:
longstanding & very active history of ML usage

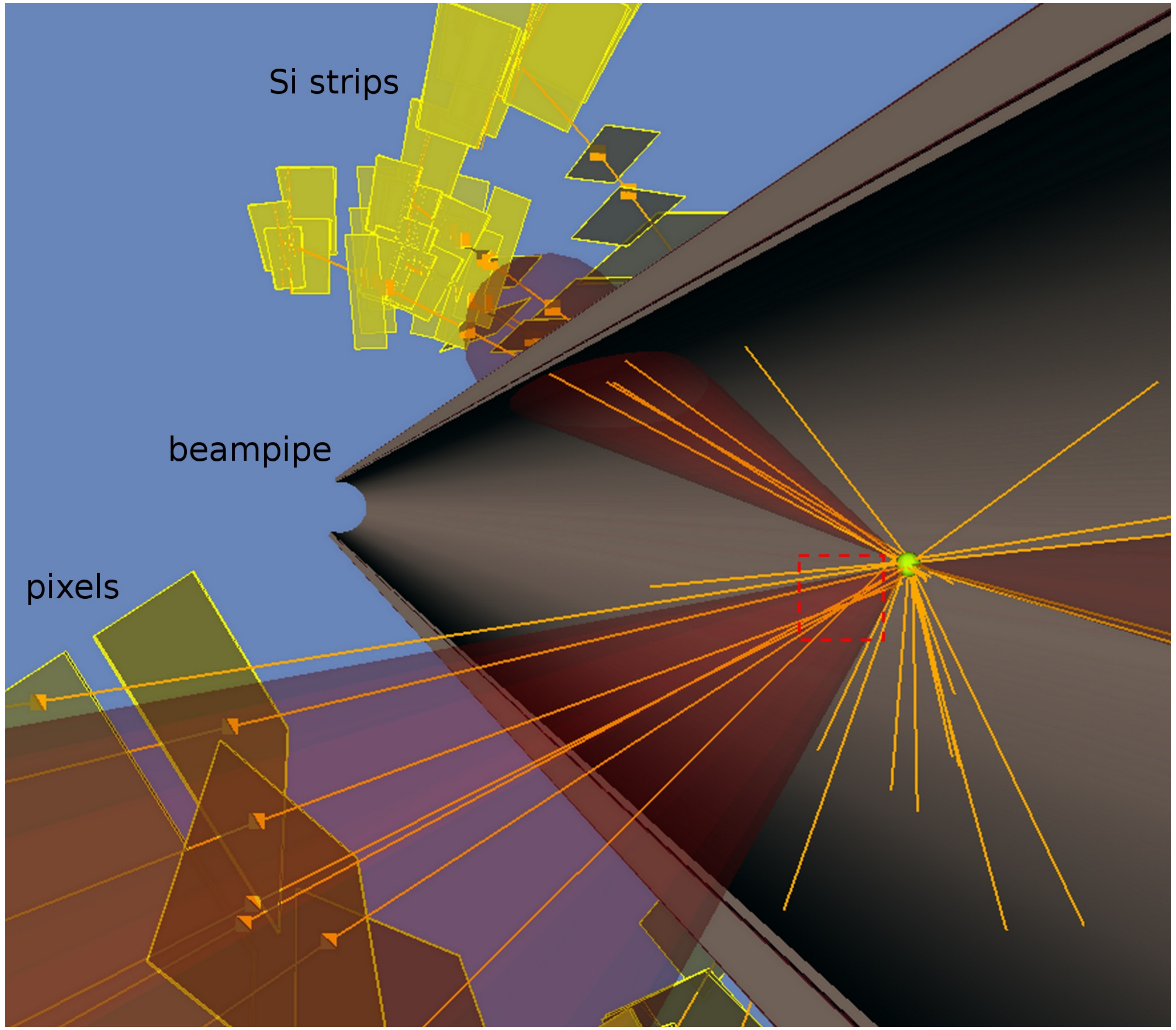
We will discuss:
Data representations
Learning algorithms

Mini-intro to flavor tagging

- Quark hadronizes to collimated bunch of hadrons = *jet*
- They come in flavors
  - *c*-jet
  - *b*-jet
  - light-jet
- Interesting physics: *b*, *c*
- Task: identify jet flavor
- Train on truth-labelled simulation data



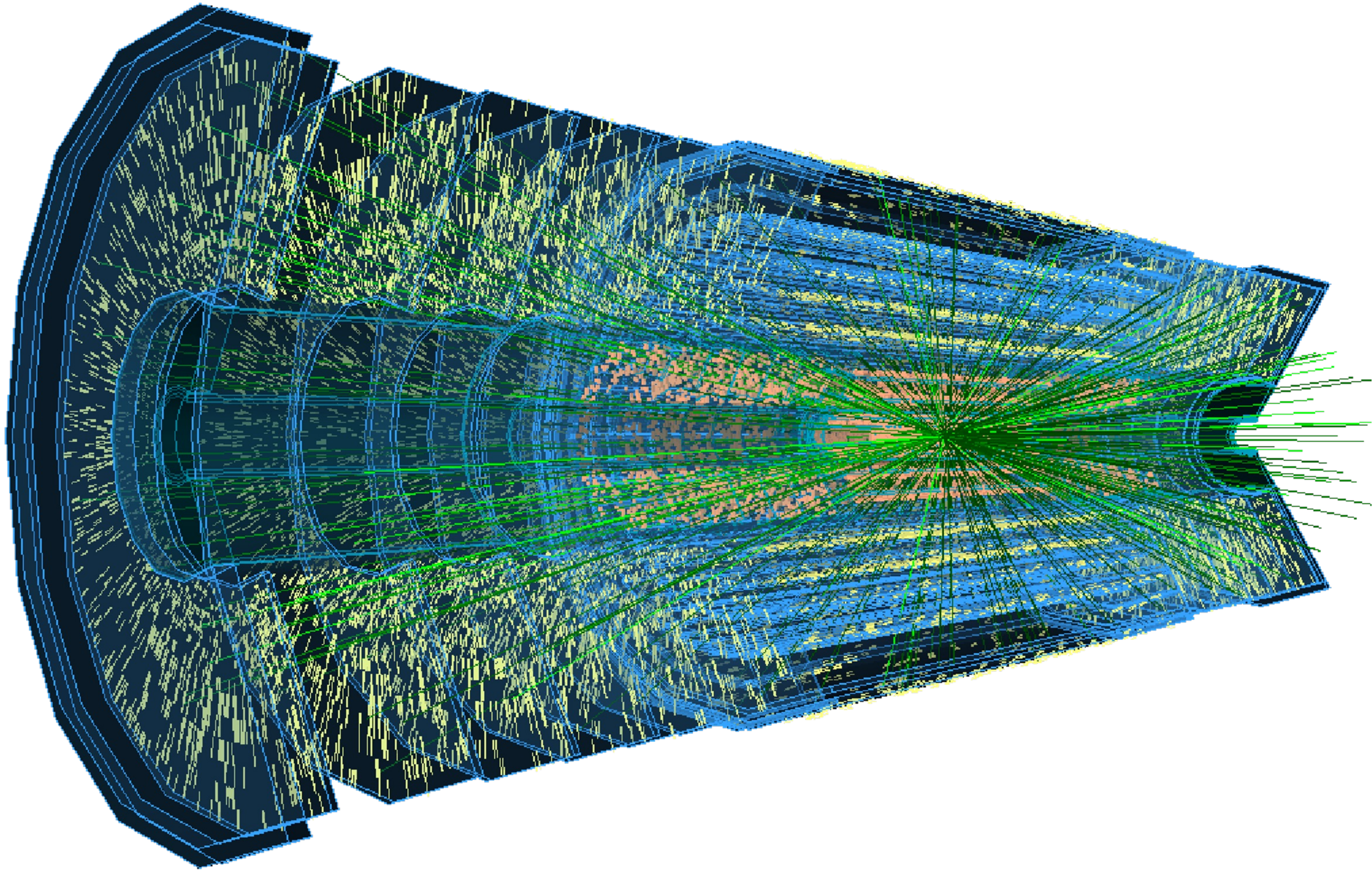
Visualizing a *jet* in a collision



[ATLAS experiment]

MC simulation

Too complex to predict
experiment outcomes
from first principles

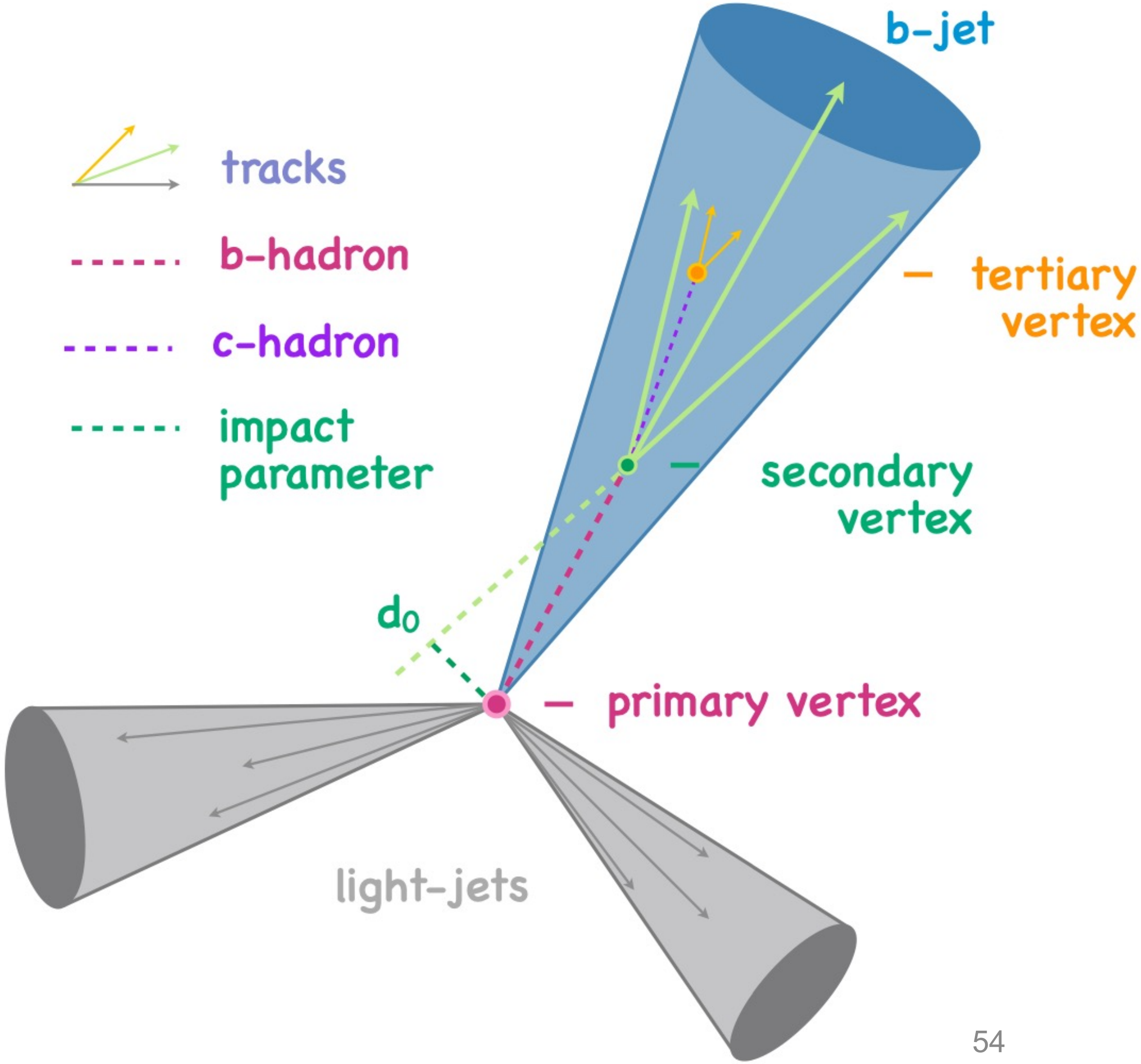


- High-fidelity simulation engines (synthetic data) to describe
  - Physics processes in a LHC collision
  - Passage of particles through the (ATLAS) detector

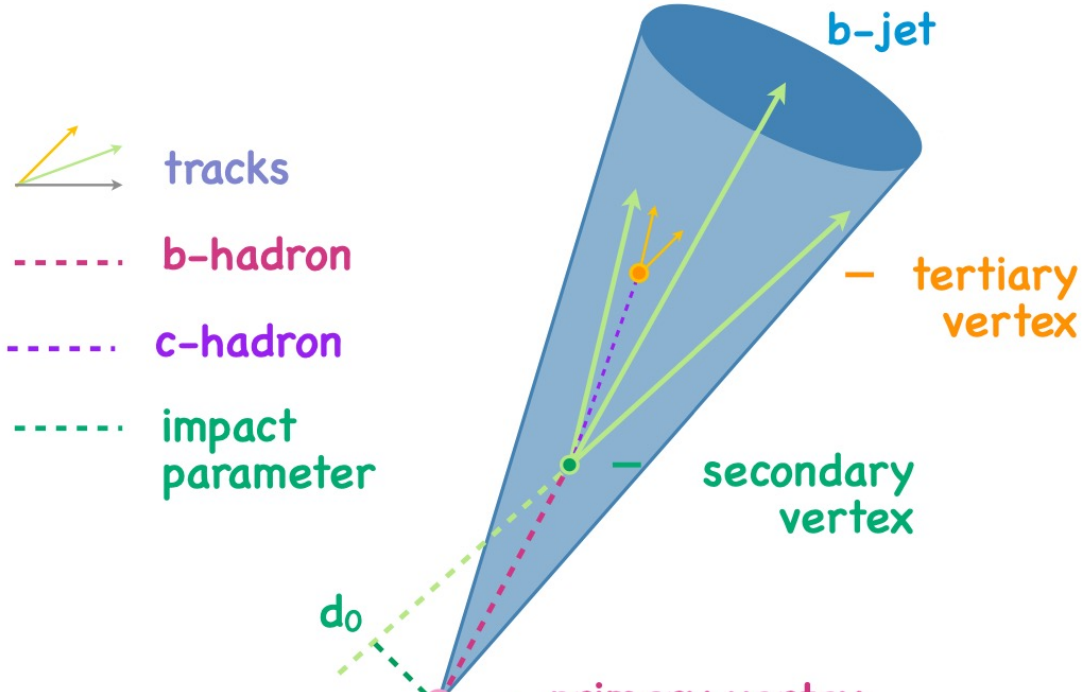
B and C hadron features

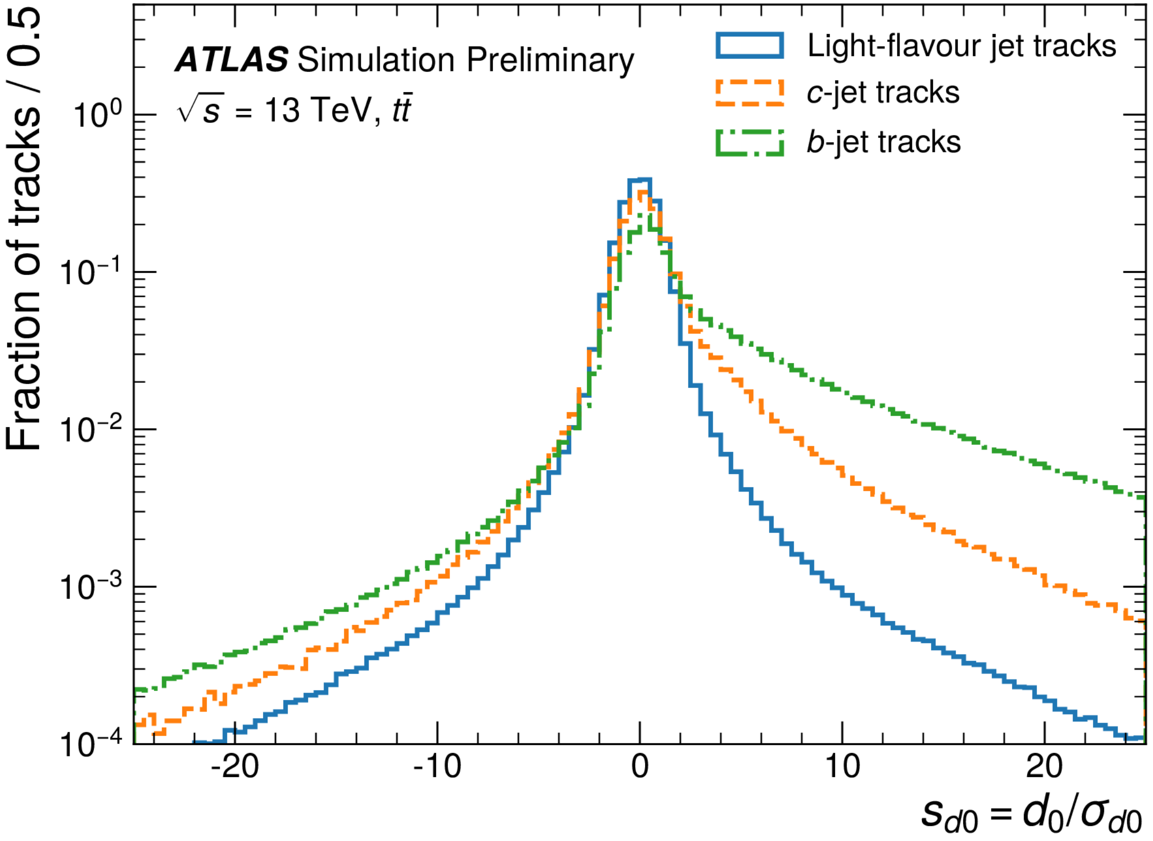
- Long lifetime
- High mass
- High decay product multiplicity
- B hadron often decays to C hadron

- What we measure
  - Reconstruct **tracks** (from hits)
  - Extrapolate tracks to **vertices**



Track features: *signed IP significance*





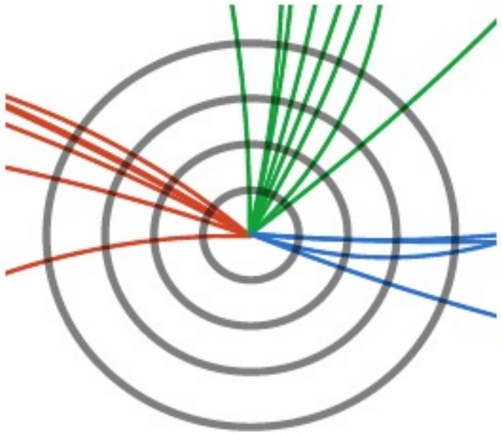
Can interpret as probability density functions p_b, p_c, p_l

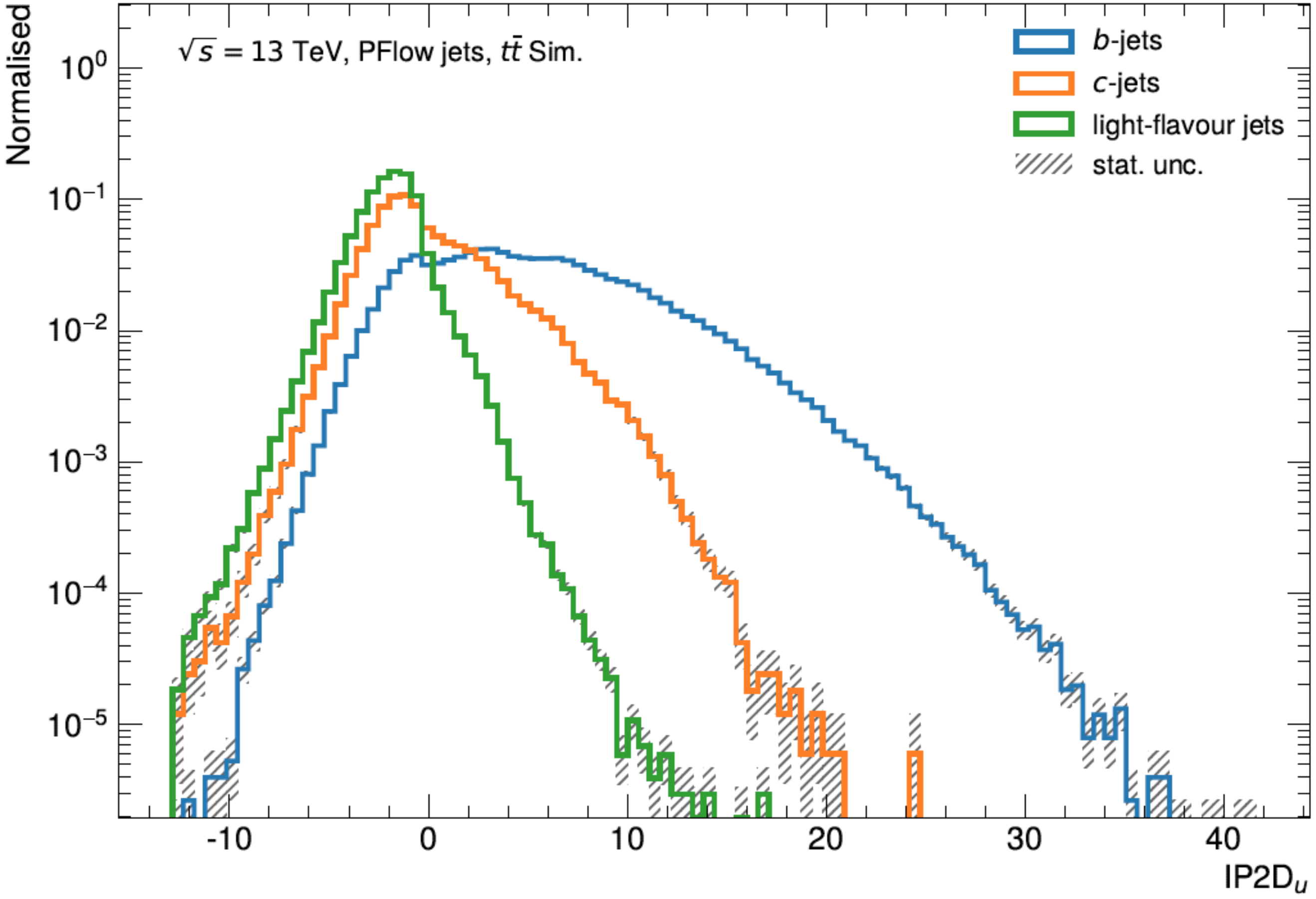
https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PUBNOTES/ATL-PHYS-PUB-2020-014/

Hand-designed **jet feature**: *IP2D*

Sum log-likelihood-ratio
(Neyman–Pearson lemma)

$$\mathrm{IPxD}_{l,c,cl} = \sum_{i \in \mathrm{tracks}} \log\left(\frac{p^{i}_{b,b,c}}{p^{i}_{l,c,l}}\right)$$

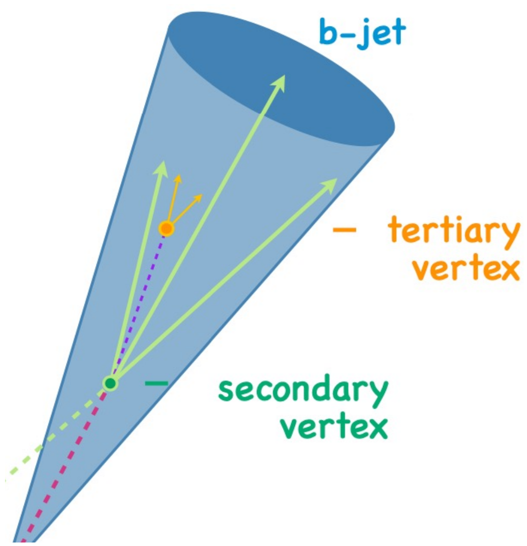


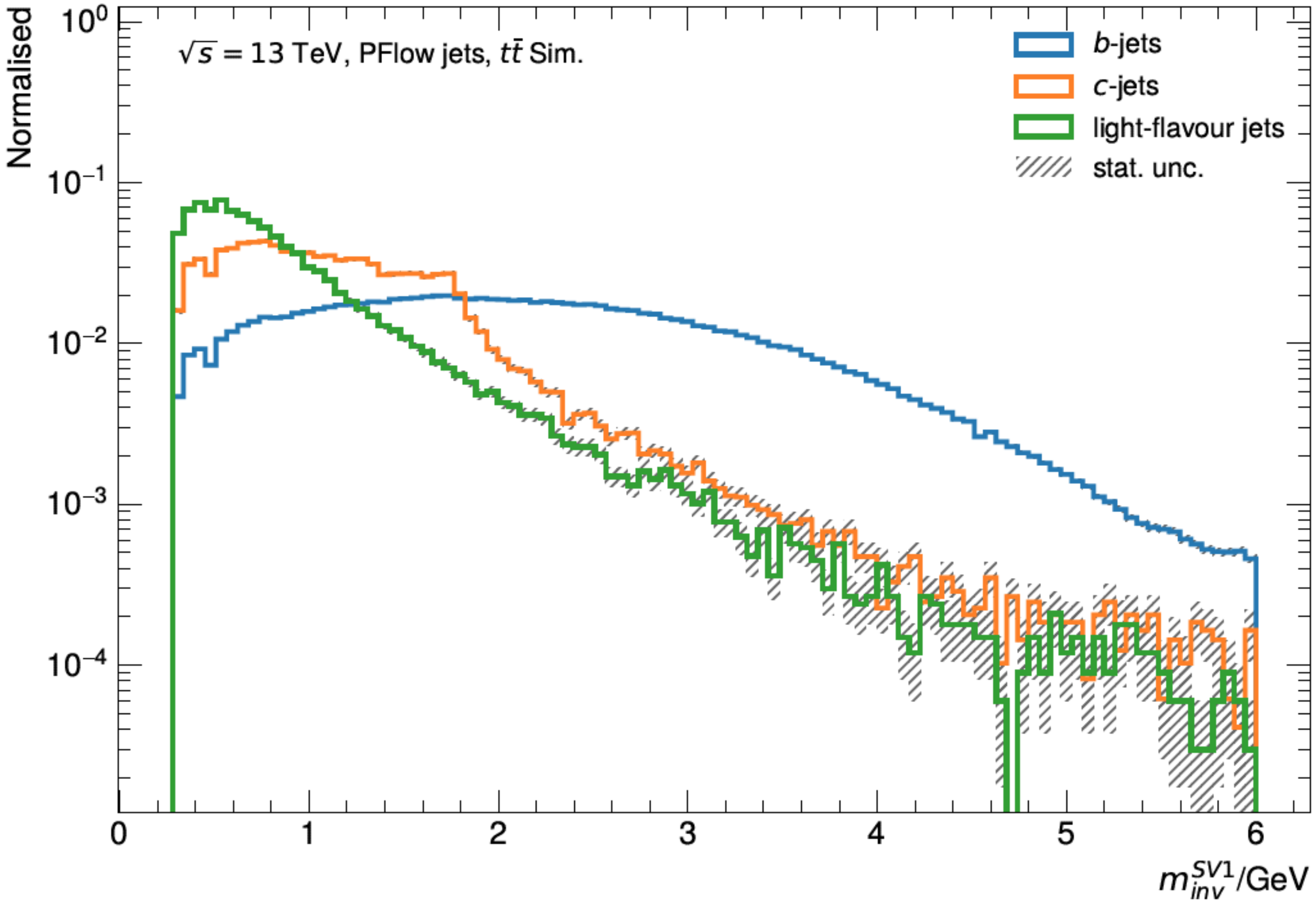


https://cds.cern.ch/record/2765038?ln=en

Vertex feature: *SV1*

Secondary vertex features

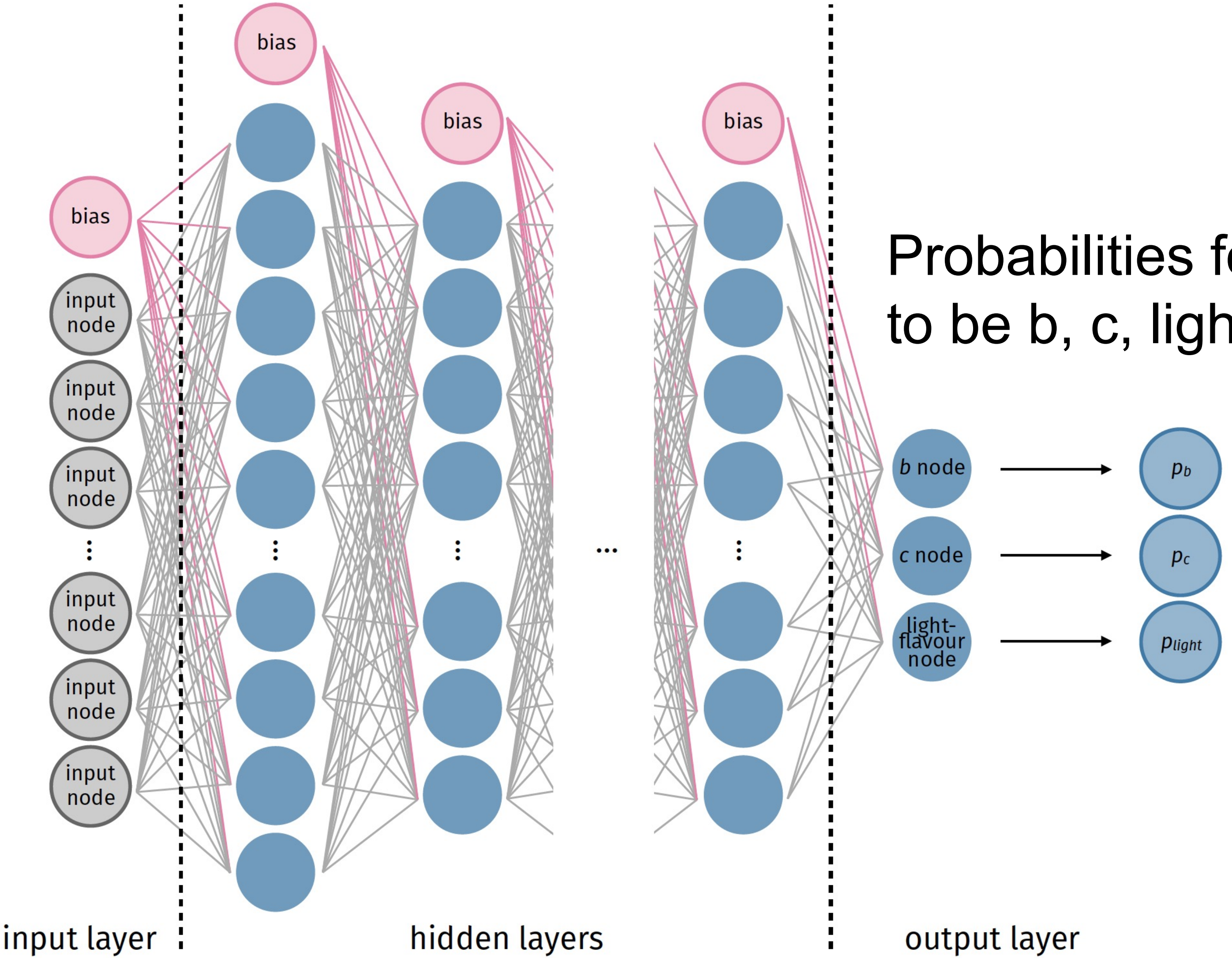




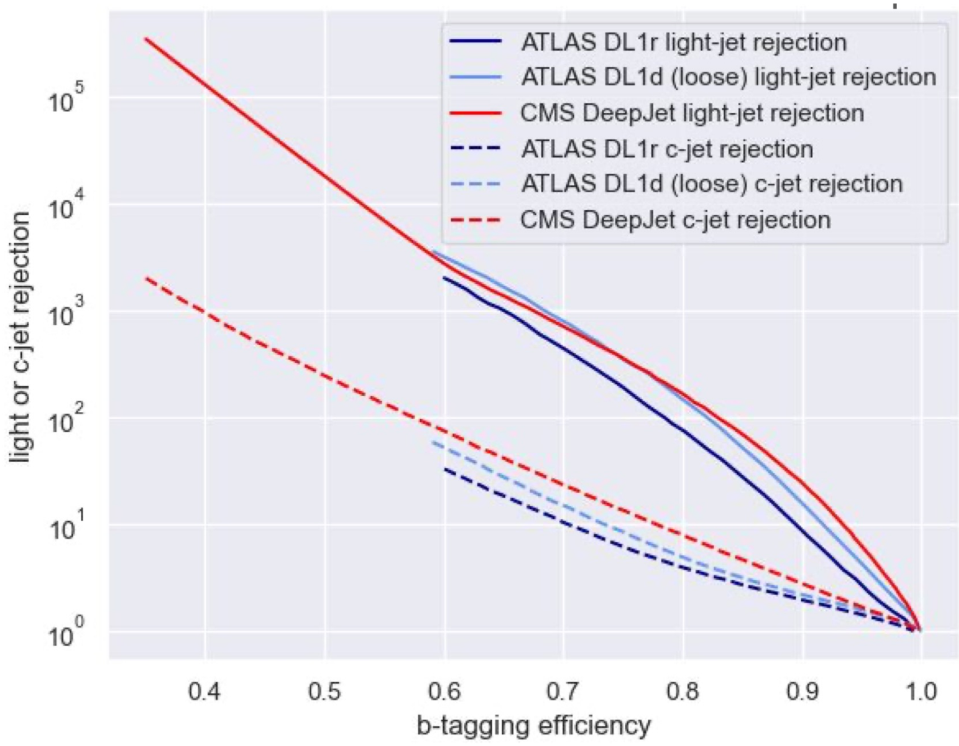
https://cds.cern.ch/record/2765038?ln=en

Weak *inductive bias*: MLP (*DL1*)

Probabilities for jet to be b, c, light



ROC curve:



https://cds.cern.ch/record/2765038?ln=en

Recap

Inputs = **variable** number of **unordered** tracks (& vertices)

MLP = **fixed-size** number of **ordered** inputs

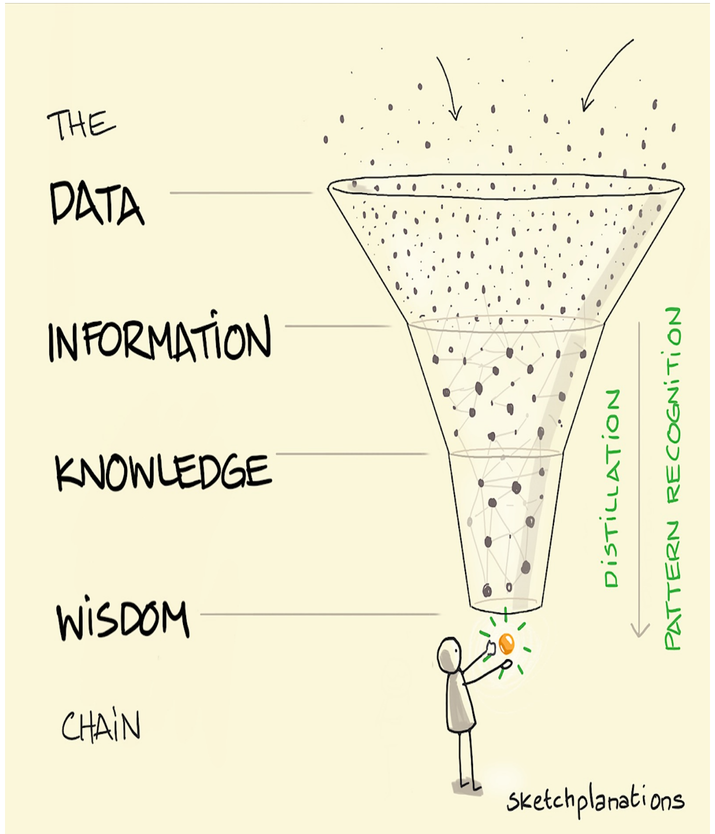
- Ad-hoc workaround:
  - **Fixed-size**: zero-pad/truncate variable-size or sum
  - **Ordered**: leading N tracks
- NOT ideal – why?

Universal approximation theorem

MLP (FF NN) can represent a good function for a task,
but the problem is how to construct it

Might require:

- *Infinite* data
- *Infinite* network size
- *Infinite* compute



Add inductive bias: Recurrent NNs (RNNs)

- Handle **variable-length ordered** sequences (e.g. NLP)
  - The food was **good**, not *bad* at all
  - The food was *bad*, not **good** at all

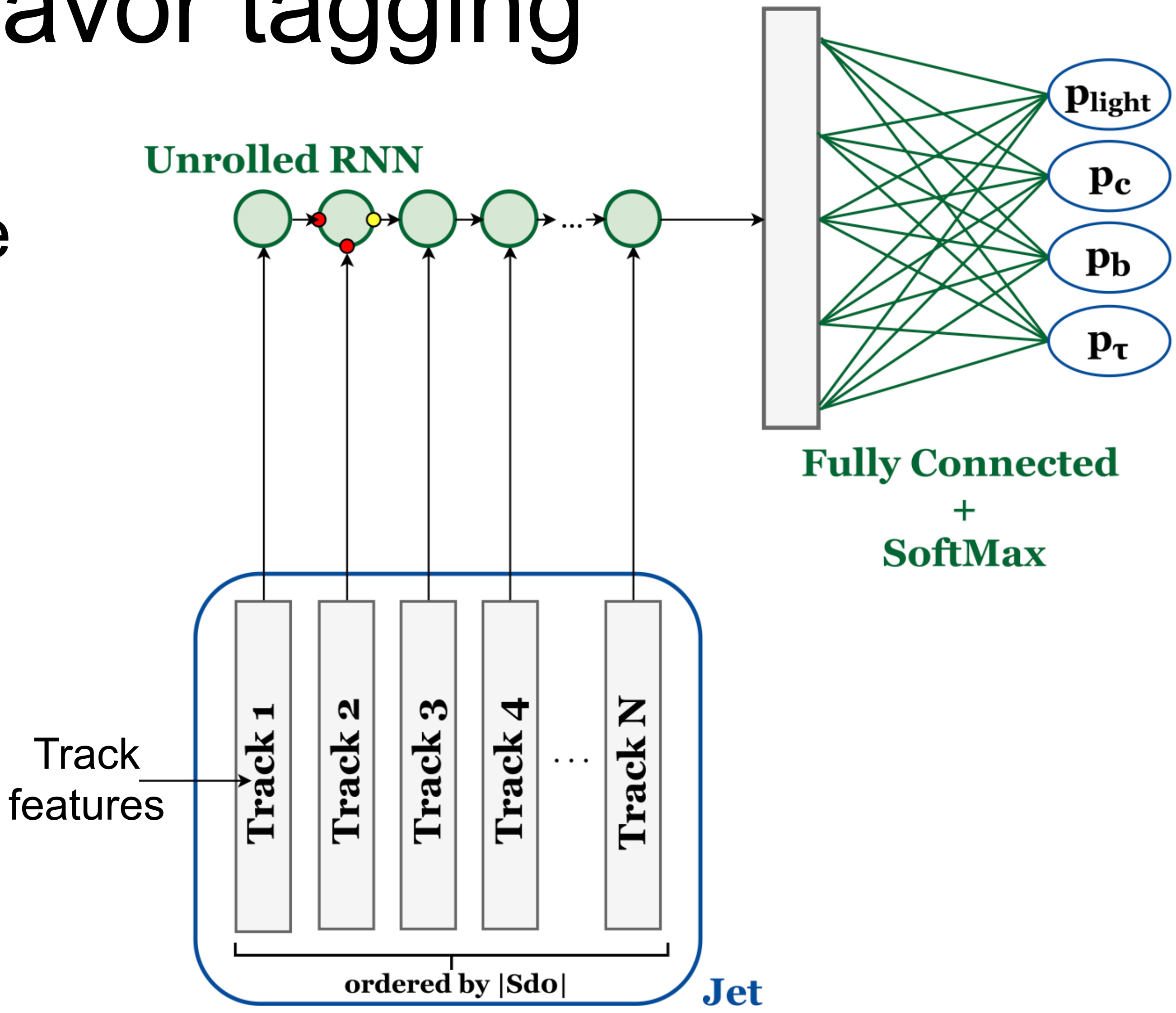
- Share parameters across the sequence

RNNs for flavor tagging

Tracks = variable-sized sequence

Better than summing tracks

But order is *arbitrarily* imposed



[ATL-PHYS-PUB-2017-003]

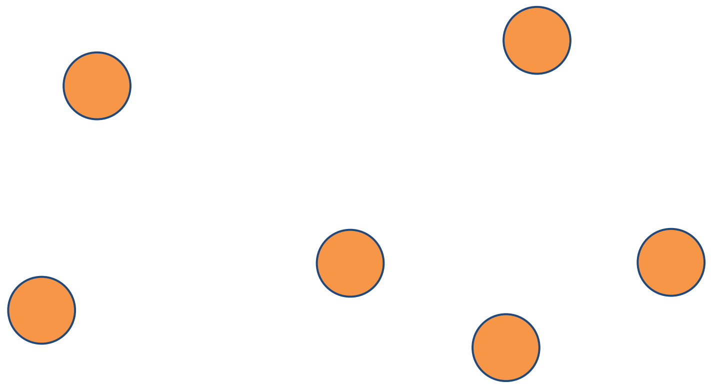
Data representation *matters*

Ideal representation: GNNs

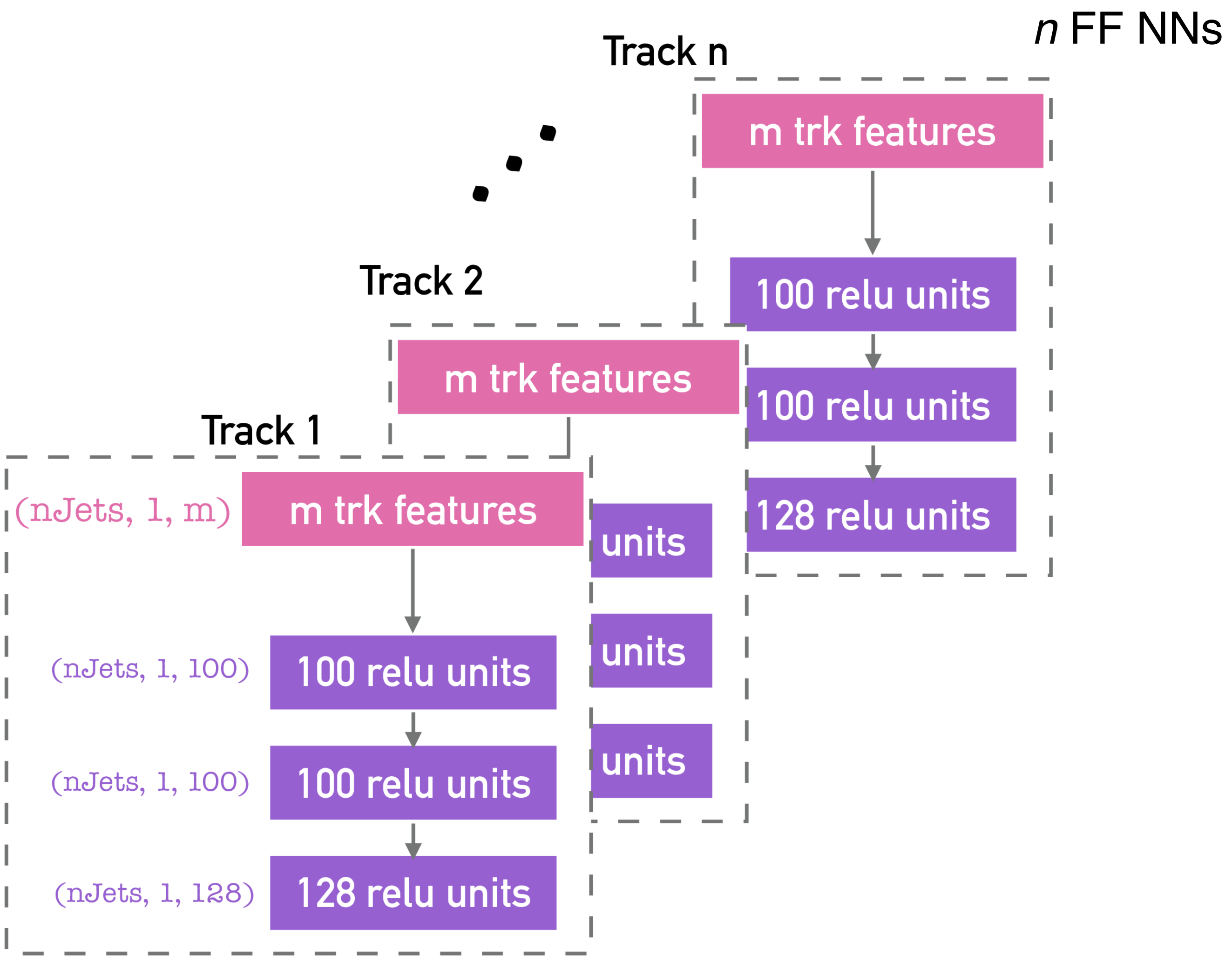
- Variable-sized inputs
- Unordered

Flavor tagging with Deep Sets

Deep Set
=
Graph without edges

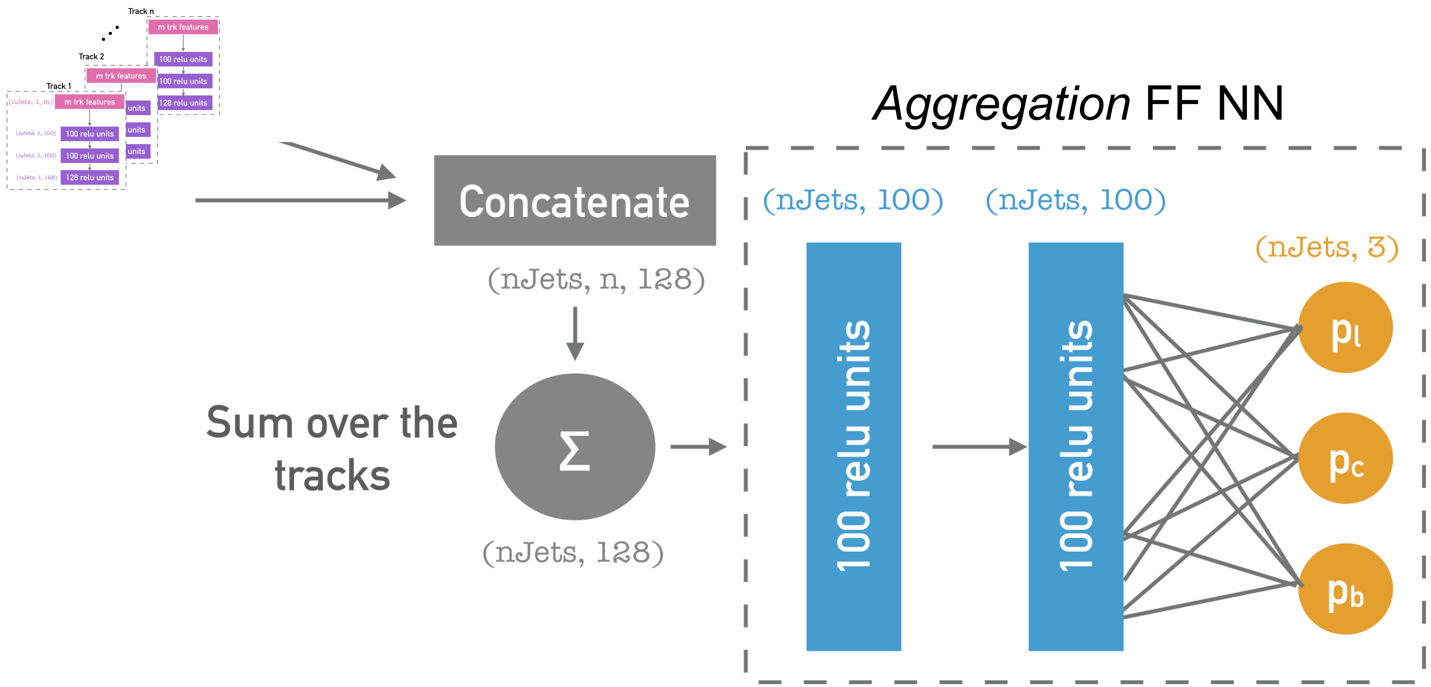


[DIPS]



Embed in higher-dimensional latent space

Flavor tagging with Deep Sets (step 2)



Permutation invariant

Arbitrary input size

Account for track correlations

[DIPS]

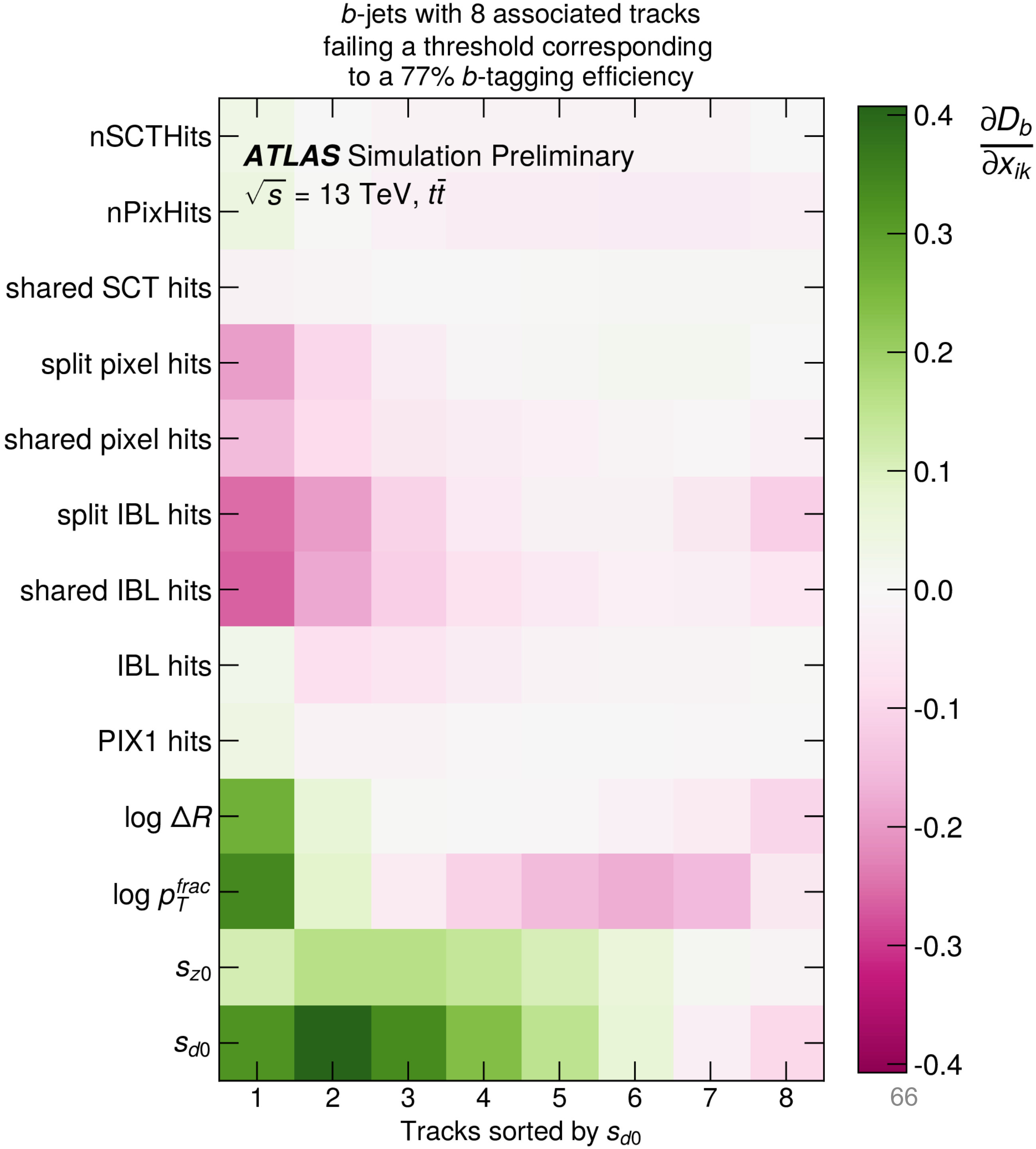
Interpretability

Saliency map:

how sensitive is discriminant to input changes

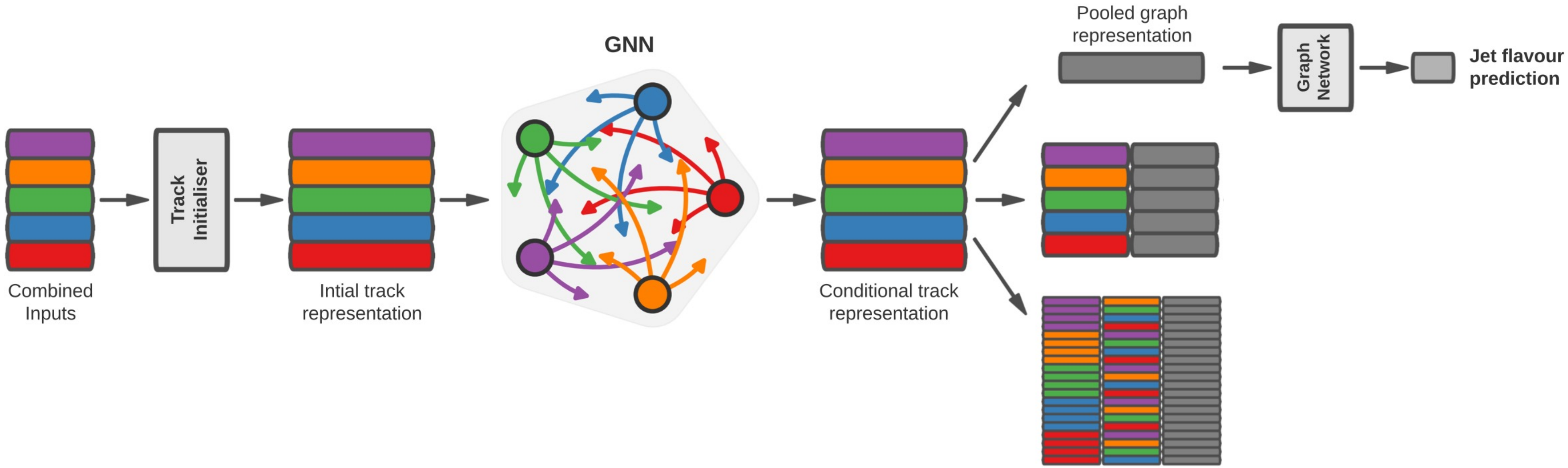
$$\frac{\partial D_b}{\partial x_{ik}} = \frac{1}{N}\sum_{j=1}^{N} \frac{\partial D_b^{(j)}}{\partial x_{ik}^{(j)}},$$

[DIPS]



Deep Set → GNN

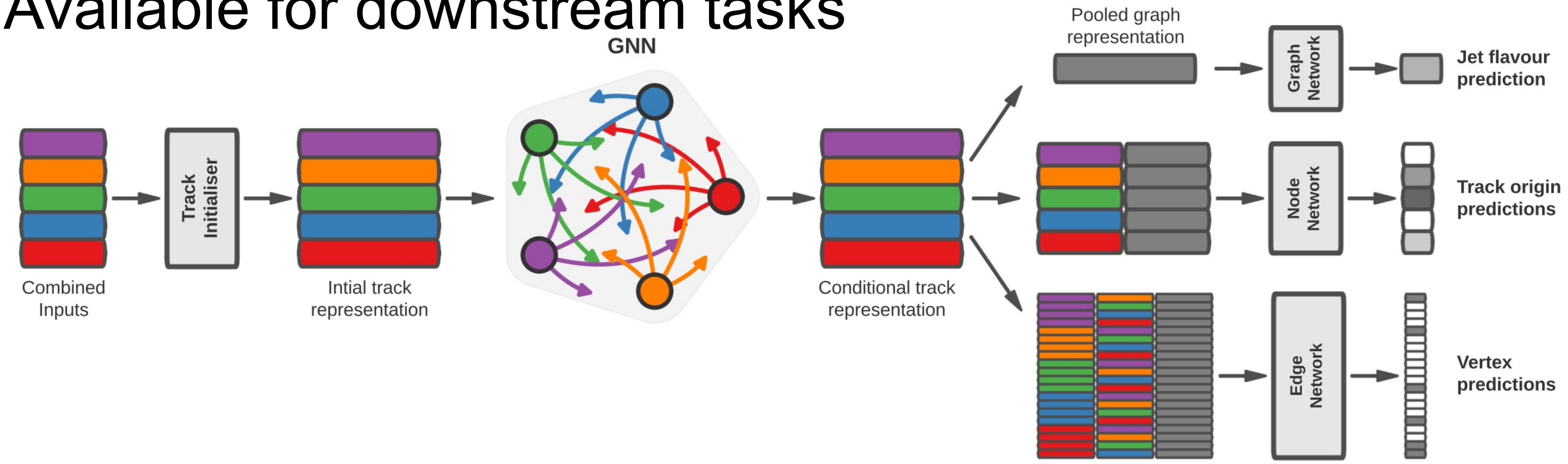
- Fully connected graph
- Edges → information exchange



[ATL-PHYS-PUB-2022-027]

The importance of *auxiliary tasks*

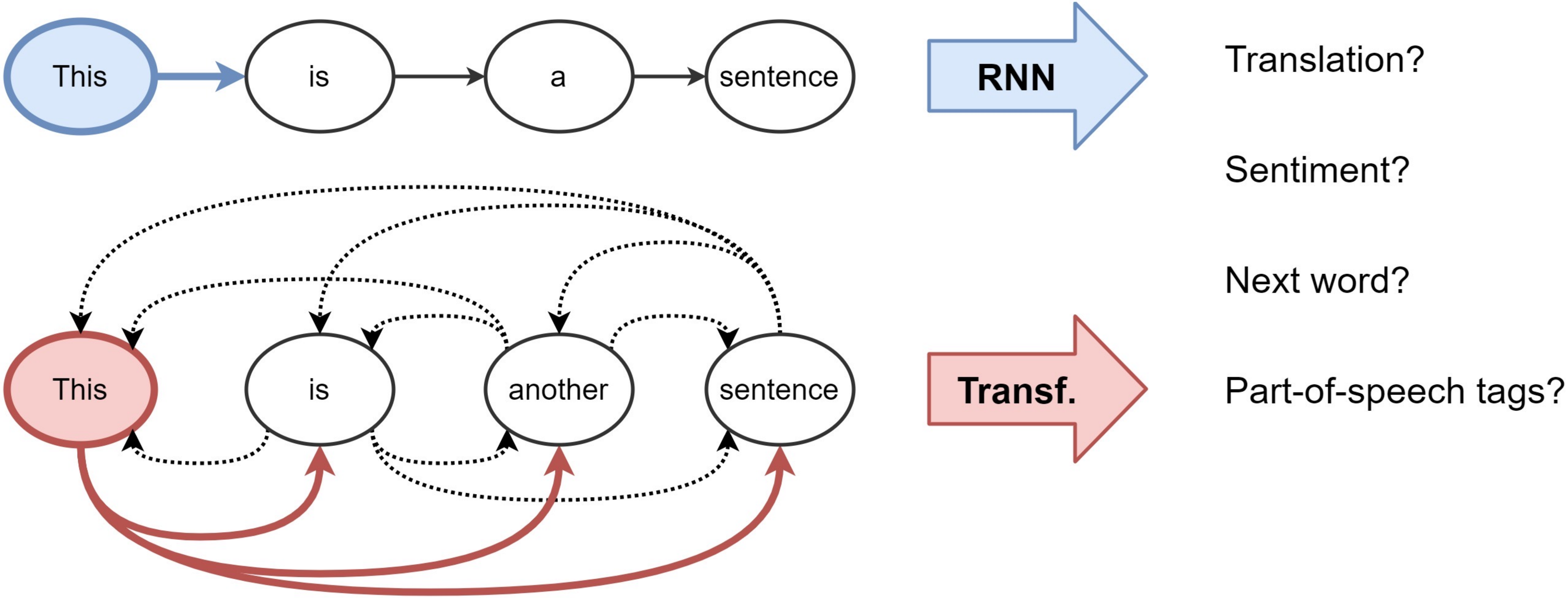
- Improves performance
- *Stepping stone*
- Available for downstream tasks



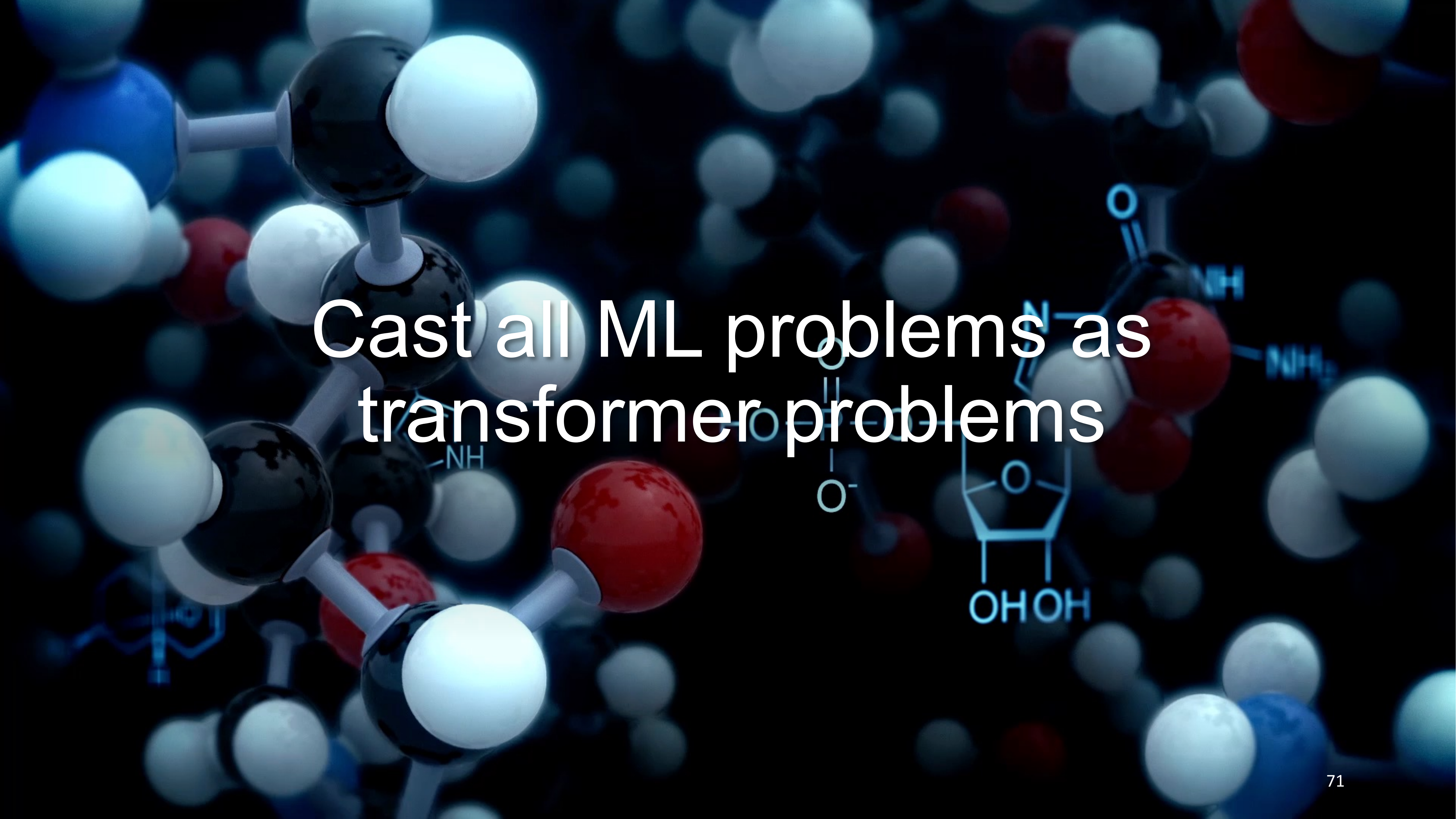
[ATL-PHYS-PUB-2022-027]

Transformers Are Taking the AI World by Storm

Representation Learning for NLP

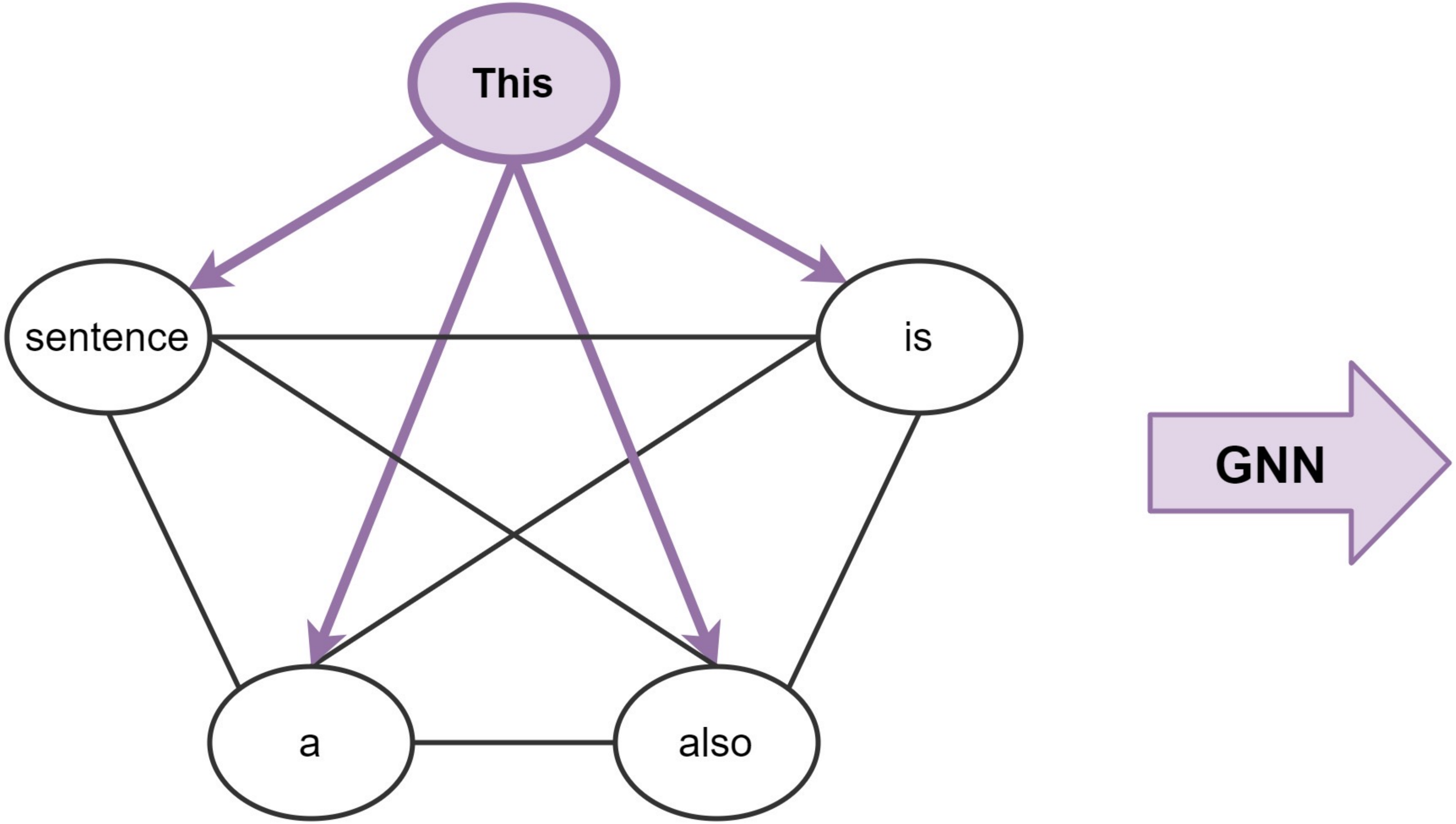


Transformers completely superseded RNNs



Cast all ML problems as transformer problems

Sentences are fully-connected word graphs



Translation?

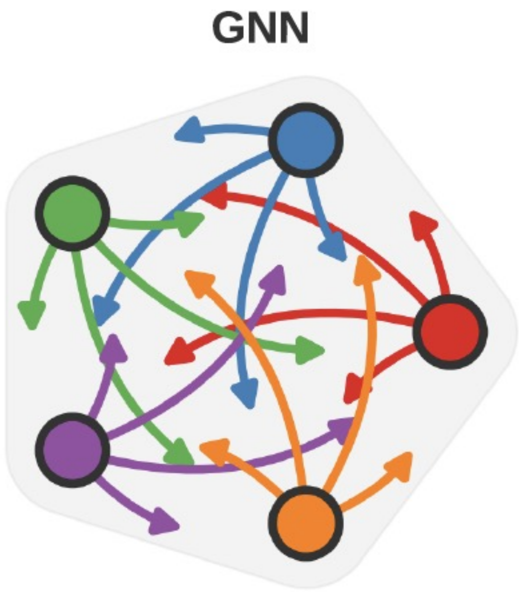
Sentiment?

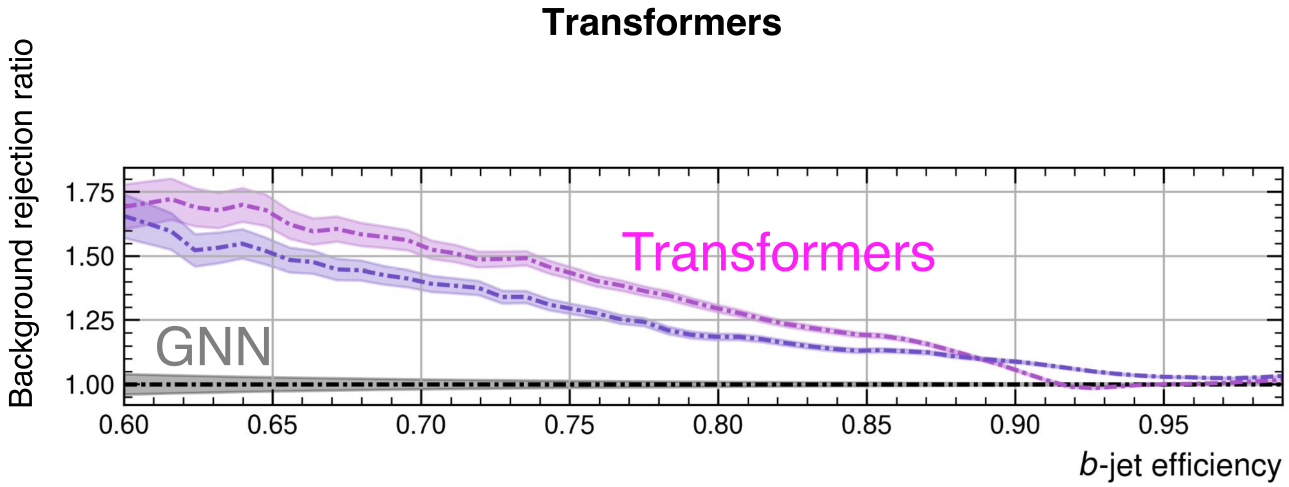
Next word?

Part-of-speech tags?

Transformers are graph neural networks

GNN → Transformer

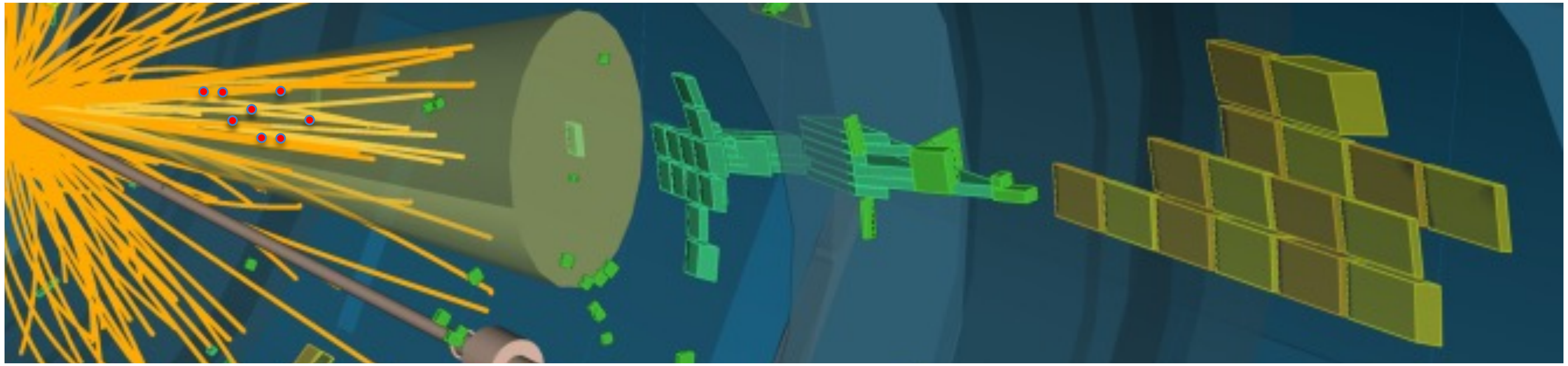




Faster to train
Scale better with more data & more parameters
Potential for *pre-trained backbone + fine-tuning*

Is there more unused information?

- Tracks
- Hits (leftover)
- Neutrals



- Heterogenous graph
  - Different node & edge types (track, hit, neutral)
  - Cannot apply same GN block

- Way out: embed in a common latent space

Have we really used all information now?

The difference between face recognition & PP?

We have a model

The holy grail

Encode model in graph structure

Back to the *inductive bias* story

Learning generic functions = curse of dimensionality

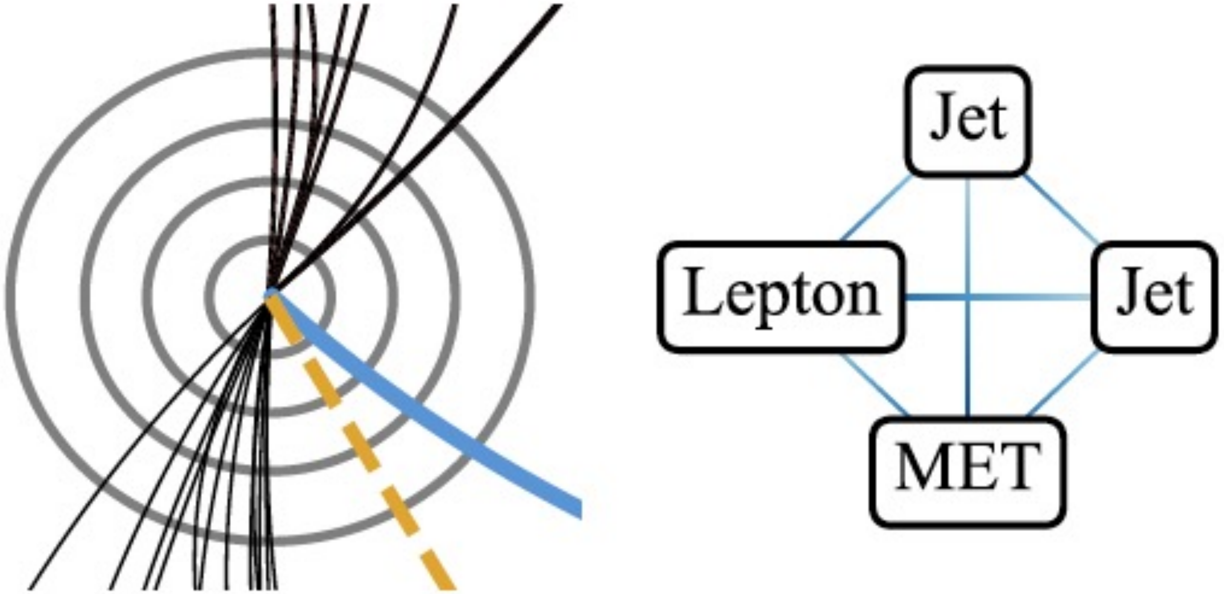
Encode physics priors to reduce dimensionality

Graphs separate **data representation** from **learning**

Encode structure: *leaving edges out or adding nodes*

Geometric deep learning

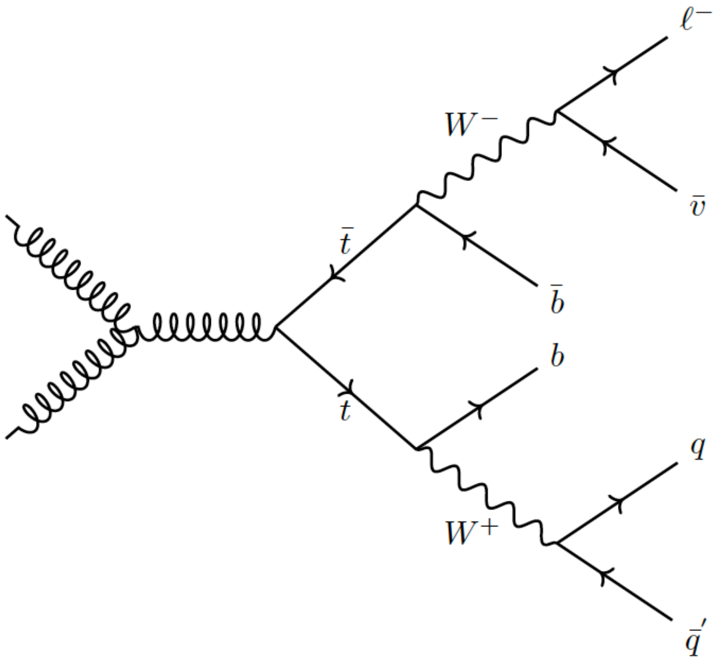
Interpreting particles in a collision



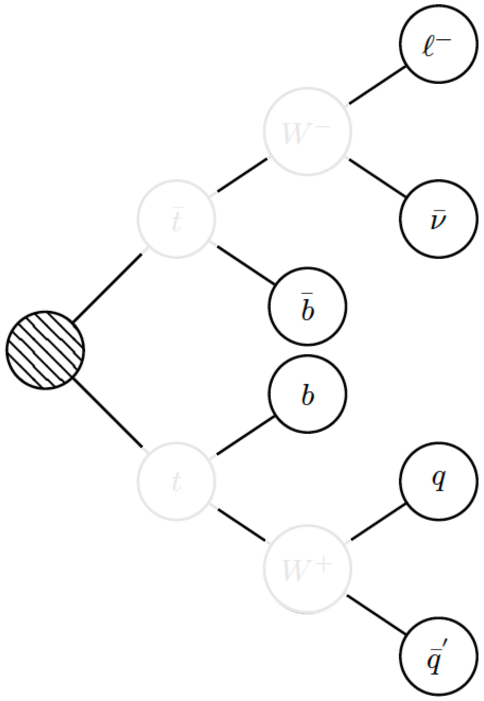
[Ref]

Encoding model in GNN

Unseen nodes



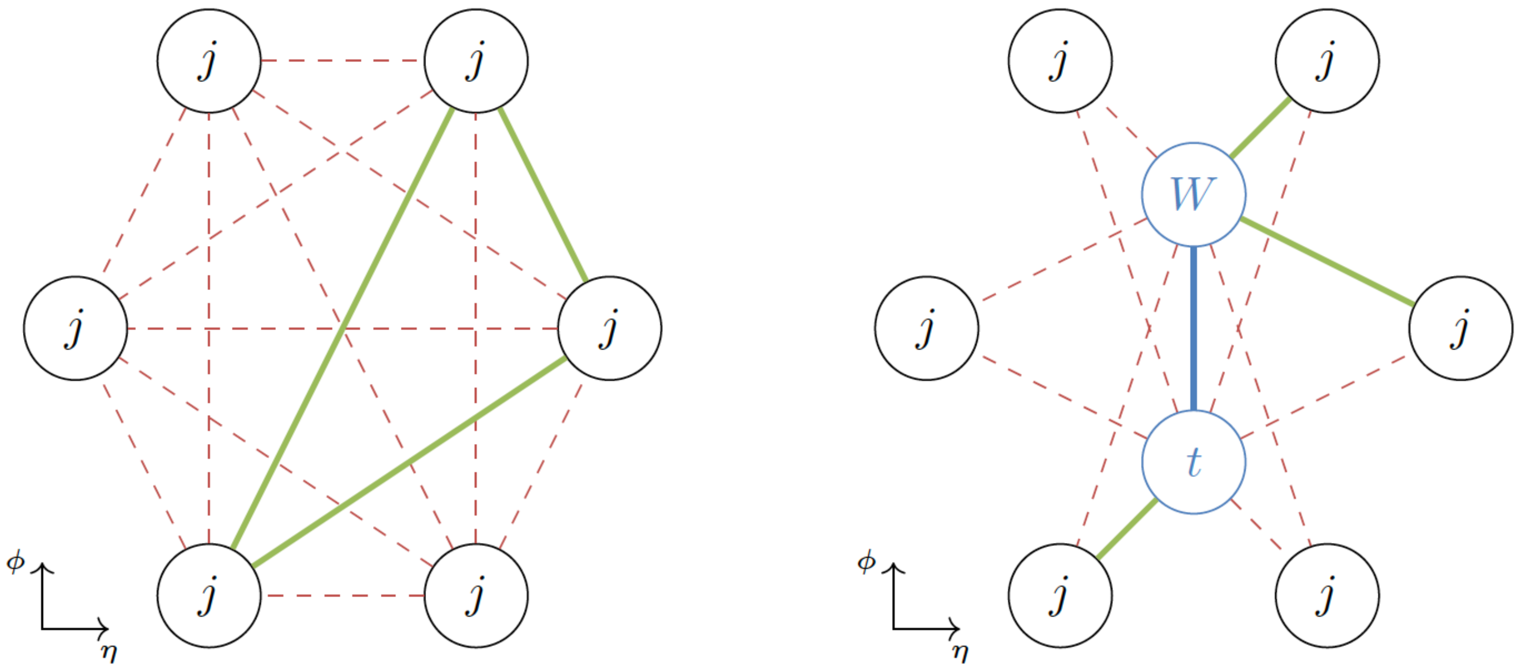
(a) Feynman diagram



(b) Node and edge graph

Topograph idea

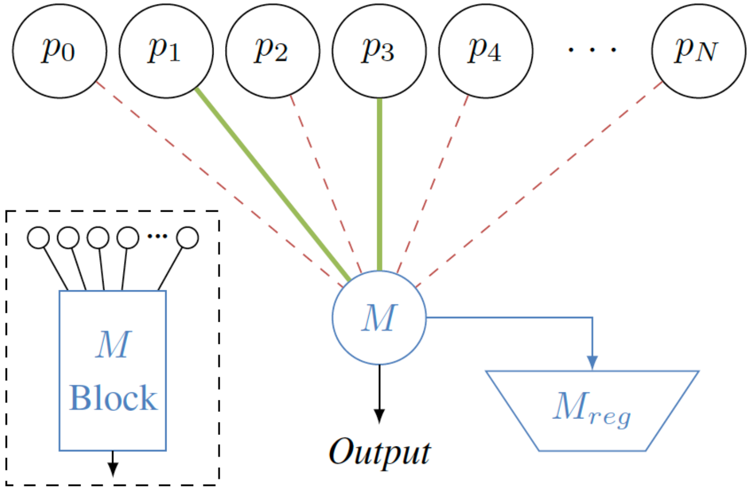
- Encode Feynman diagram in GNN
- Add virtual W & top nodes
- Combinatorics
  - Fully connected $O(n^2)$
  - Topograph $O(n)$
- Predict kinematics

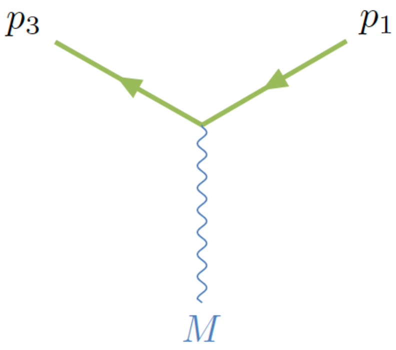


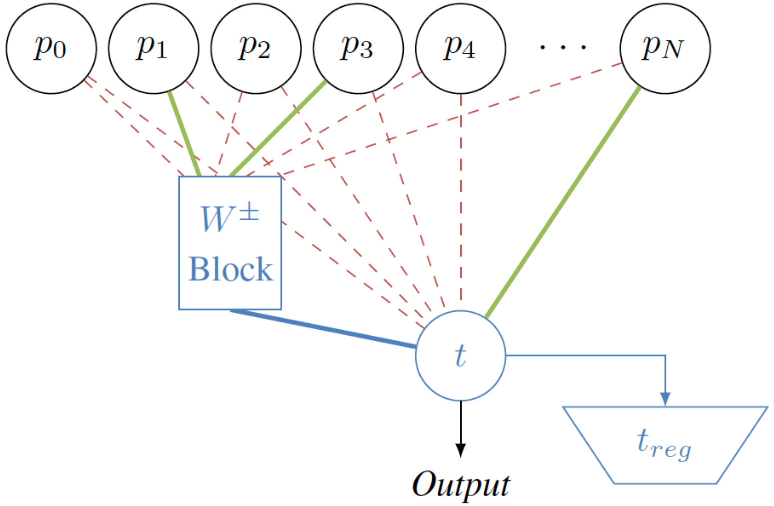
(a) Fully connected graph (b) Topograph

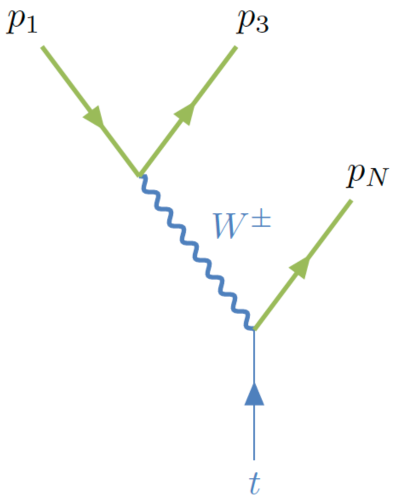
[arxiv submission in preparation]

Particle Blocks



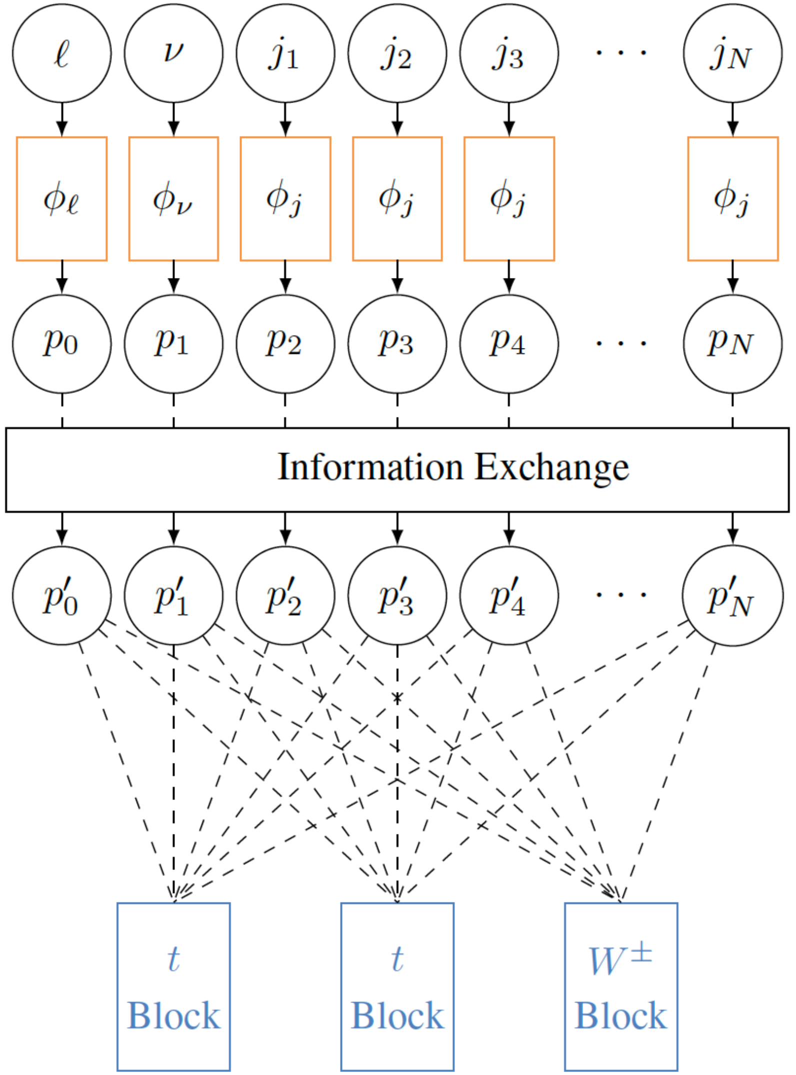






- Auxiliary tasks
  - Virtual node regression
  - Link prediction

Modular network structure

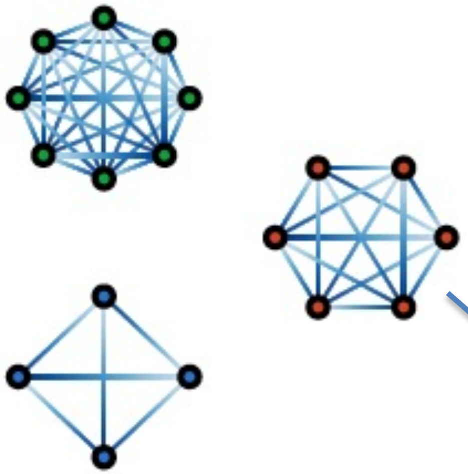


Network ϕ to embed particles in same space p

GNN layer to exchange information between particles

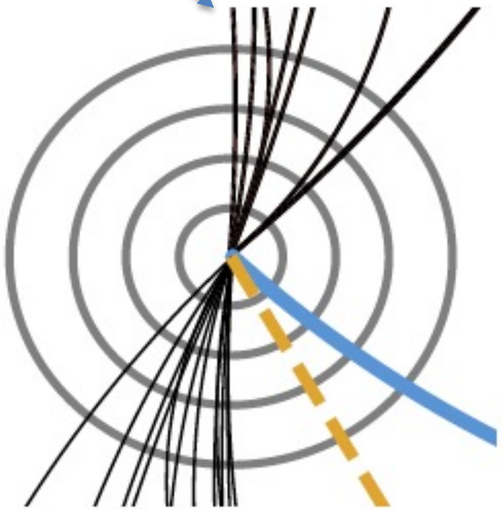
Initialize virtual nodes with attention pooling

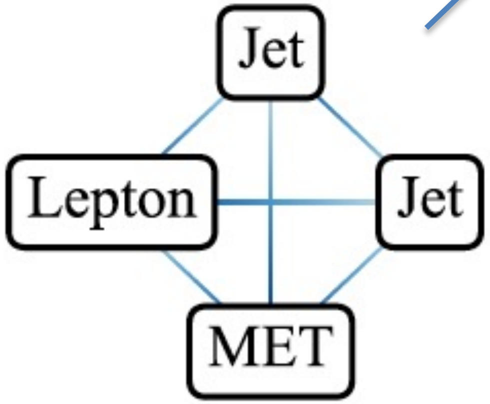
Graph of graphs of graphs…

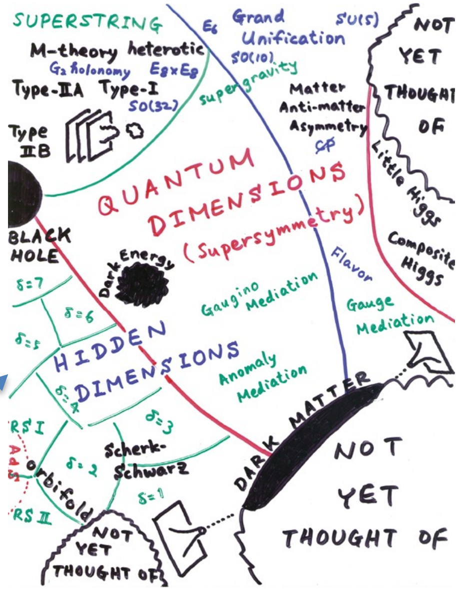


Space of constituents of a jet

Space of objects in a collision event







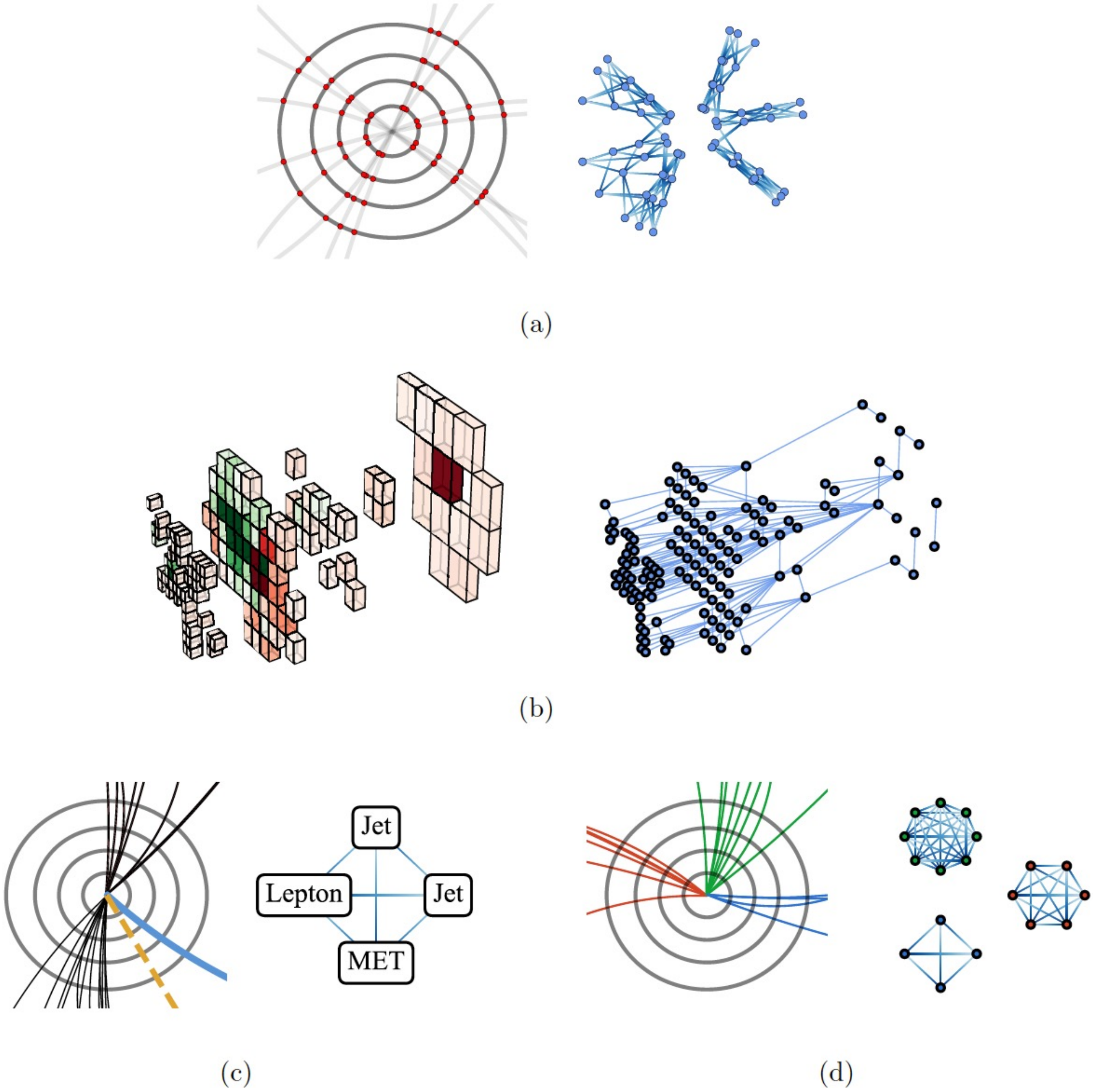
Collision events = nodes in theory space

Raw data =

point cloud

~

particle cloud

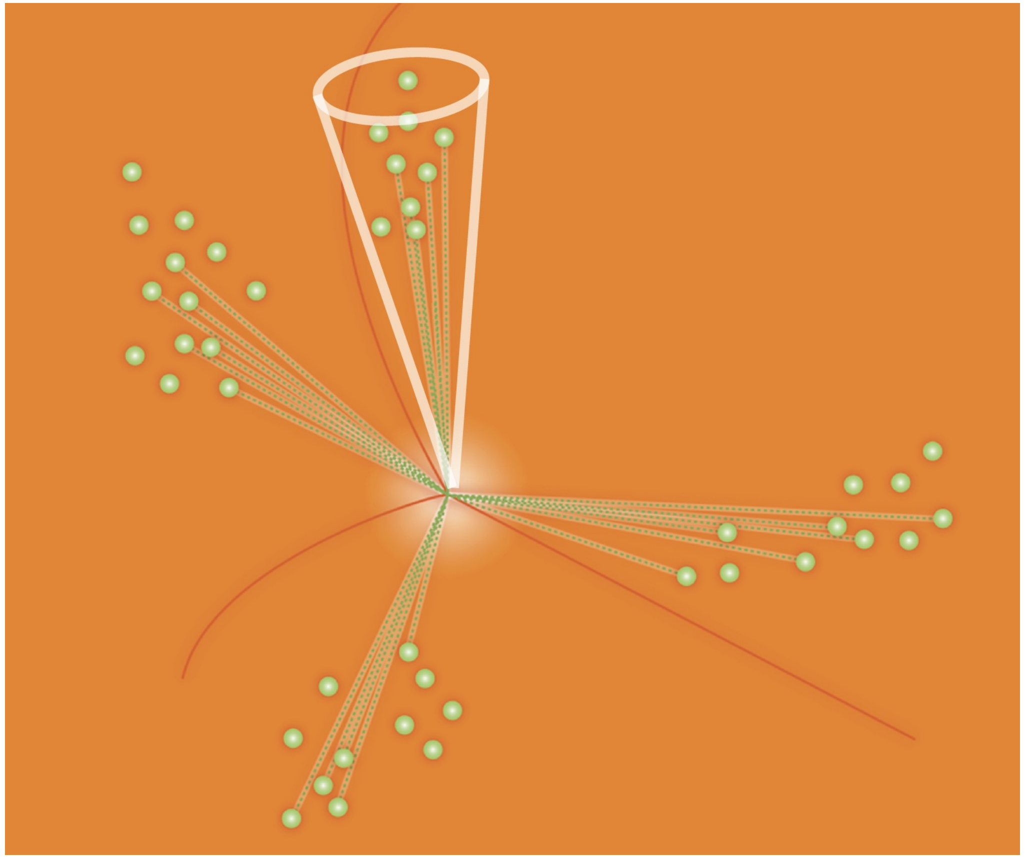


[Ref]

Generating images ⇔ generating point clouds

[Karras et al., 2018]

⇔



Based on fully-connected GNN [2106.11535]

No AI lecture w/o these 2 topics

Quick intermezzo: going beyond GNNs…

1. Ethical AI

- *What you see is what you get*

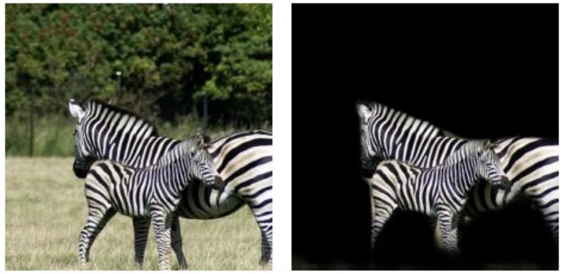
- Trained models reflect the training data
  - Existing biases are kept!
  - Famous *cow-on-the-beach* issue
    - Universal cow features
    - Spurious patterns

- Effort needed to unbias
  - Augment
  - Decorrelate
    - Support issue

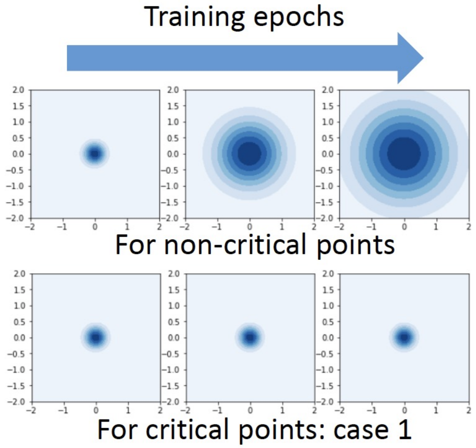
2. Trustworthy AI

Explain to human how the verdict was reached

1. *Post-hoc explainer* NN applied to trained model
   – Perturbation-based [SHAP, LIME]
   – Gradient-based [Saliency map, see b-tag example]



2. *Self-explainer:* learn like a human during training
   – Inject stochasticity & learn the noise
   – For GNNs: node Gaussian noise, edge Bernoulli noise
      • Resonates with a physicist's notion of *uncertainty*



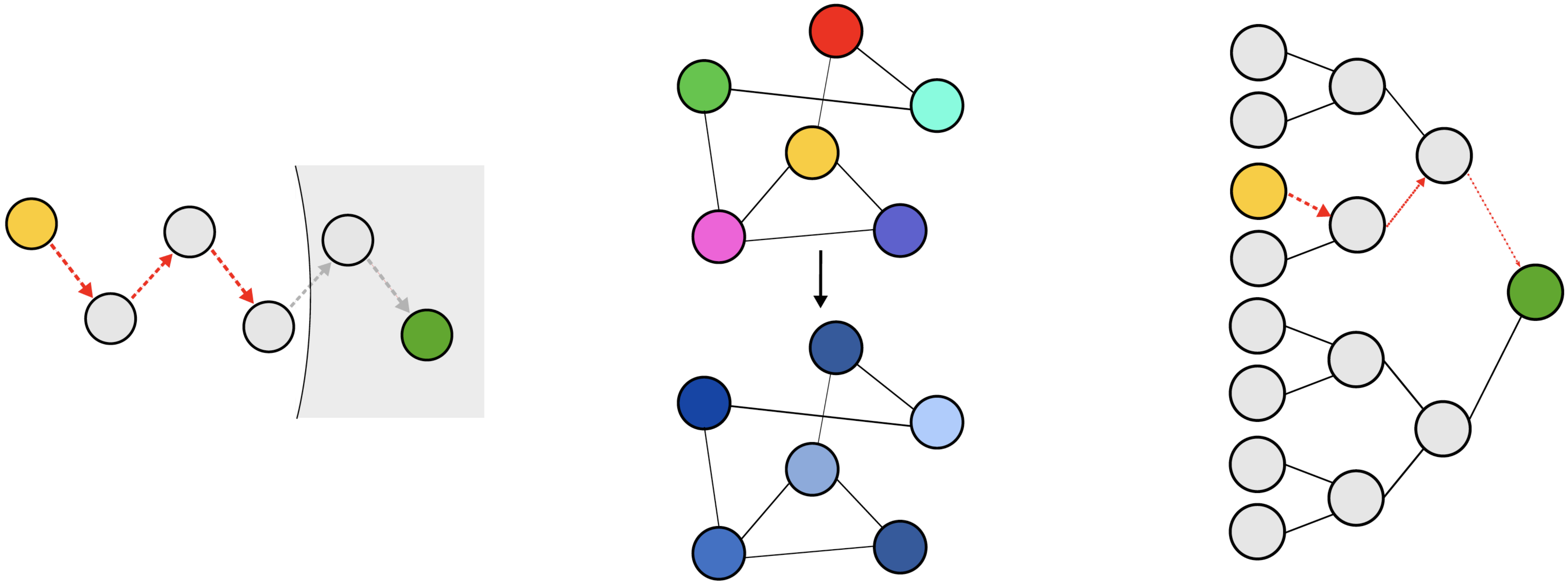
[2210.16966]

Graphs for experts

(that you are now!)

What could go wrong?

Common problems with GNNs

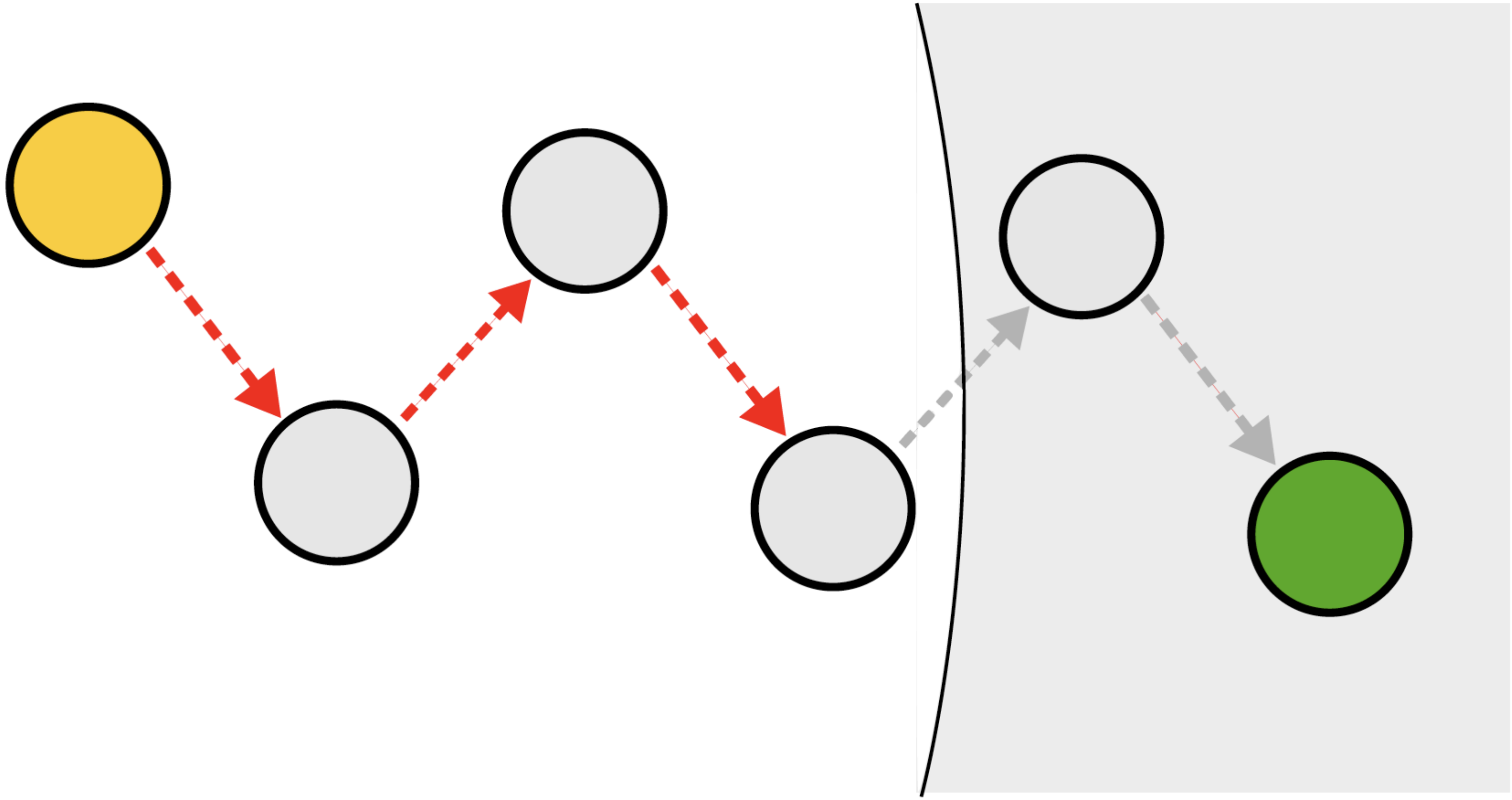


Under-reaching Over-smoothing Over-squashing

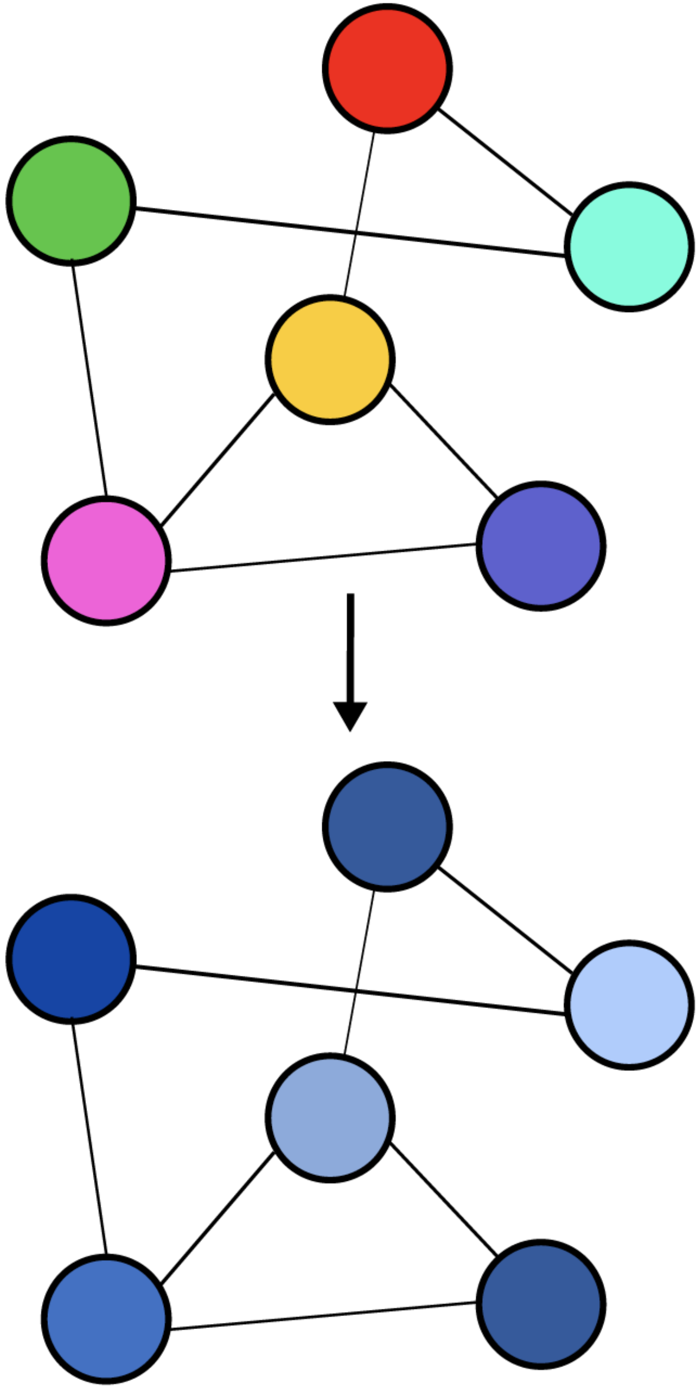
[Slides from Hammers & Nails talk by Francesco Di Giovanni, Twitter]

Under-reaching

Yellow cannot *reach*
green in 3 updates



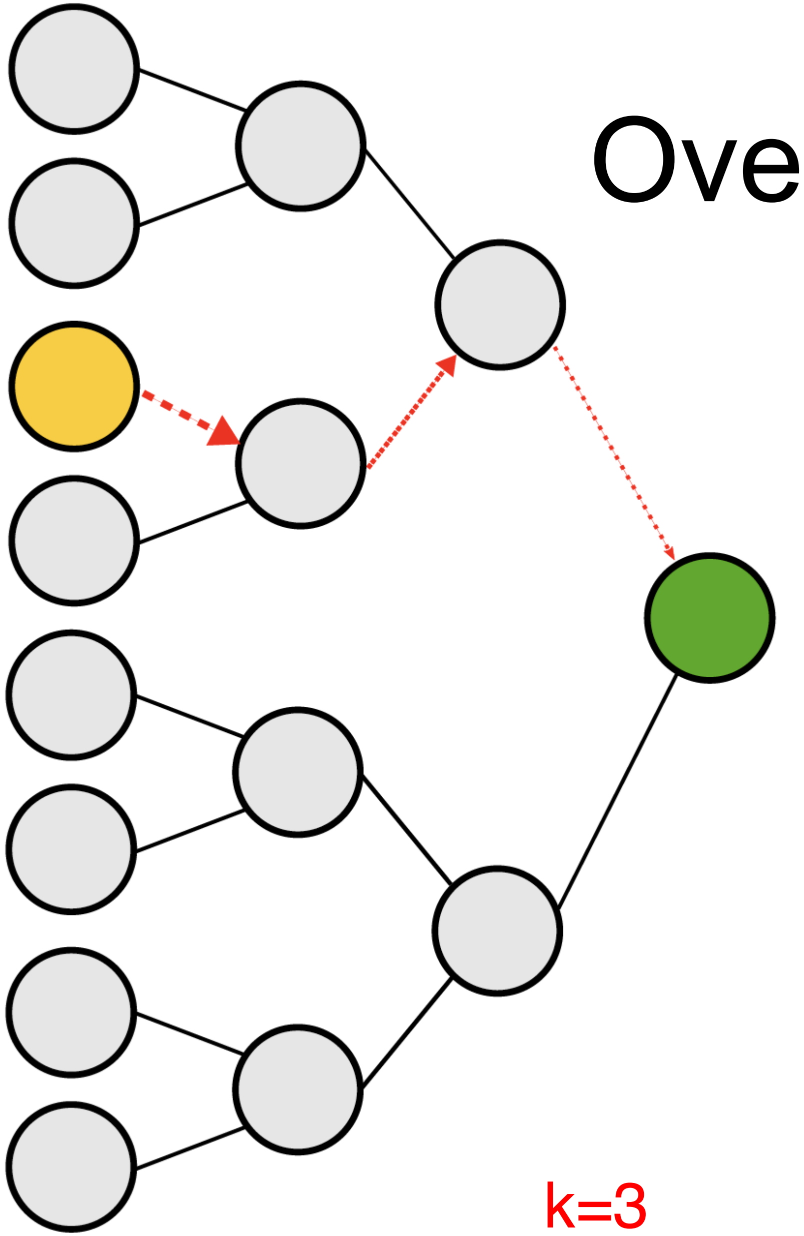
Fix: increase depth



Over-smoothing

Too deep: node representations can become similar (*smoothed out*) and weaken influence of graph structure

Fix: decrease depth / sharper update functions



Over-squashing

Size of k-hop neighborhoods grows substantially with k

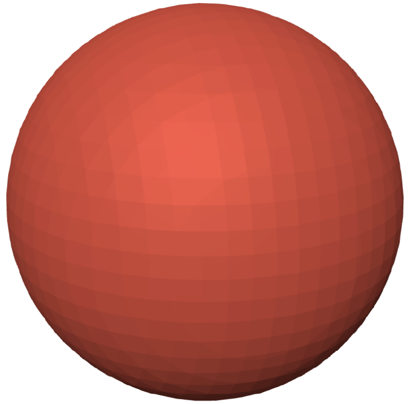
Squashing more and more information into a node

Congestion / bottleneck issue

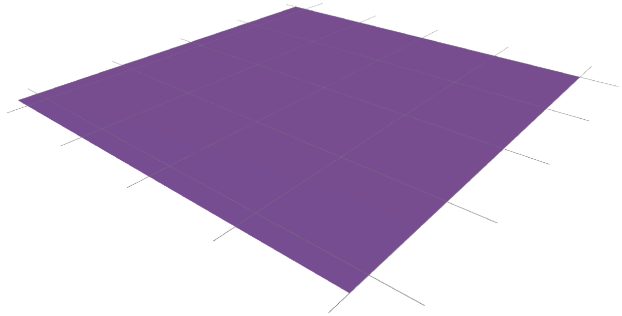
Fix: add *short-cuts* based on *curvature*

[1] Alon and Yahav (2021)

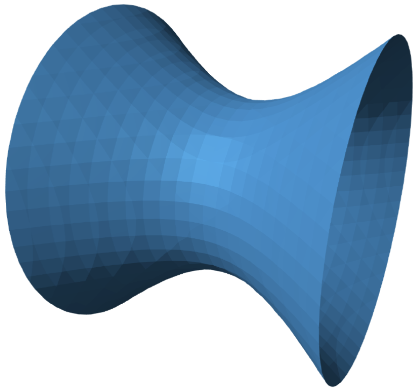
Ricci curvature – intuition



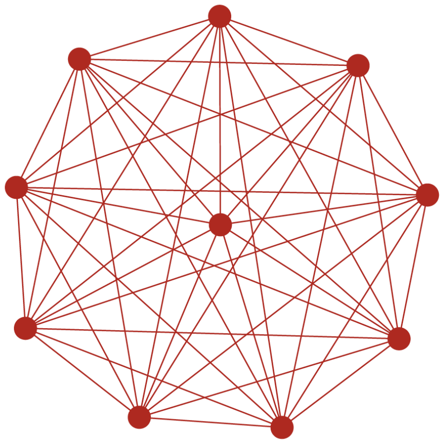
Spherical (> 0)



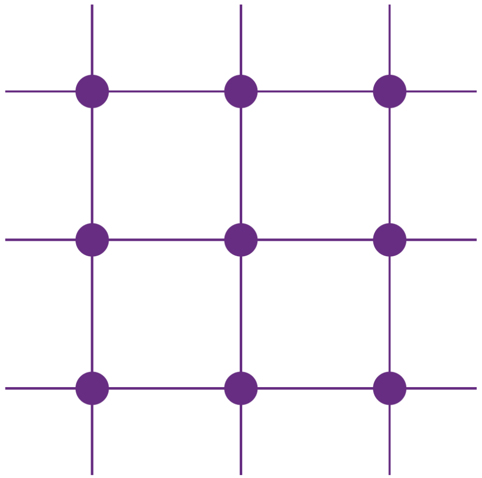
Euclidean ($= 0$)



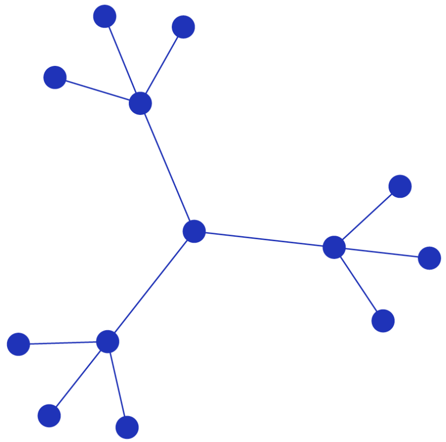
Hyperbolic (< 0)



Clique (> 0)

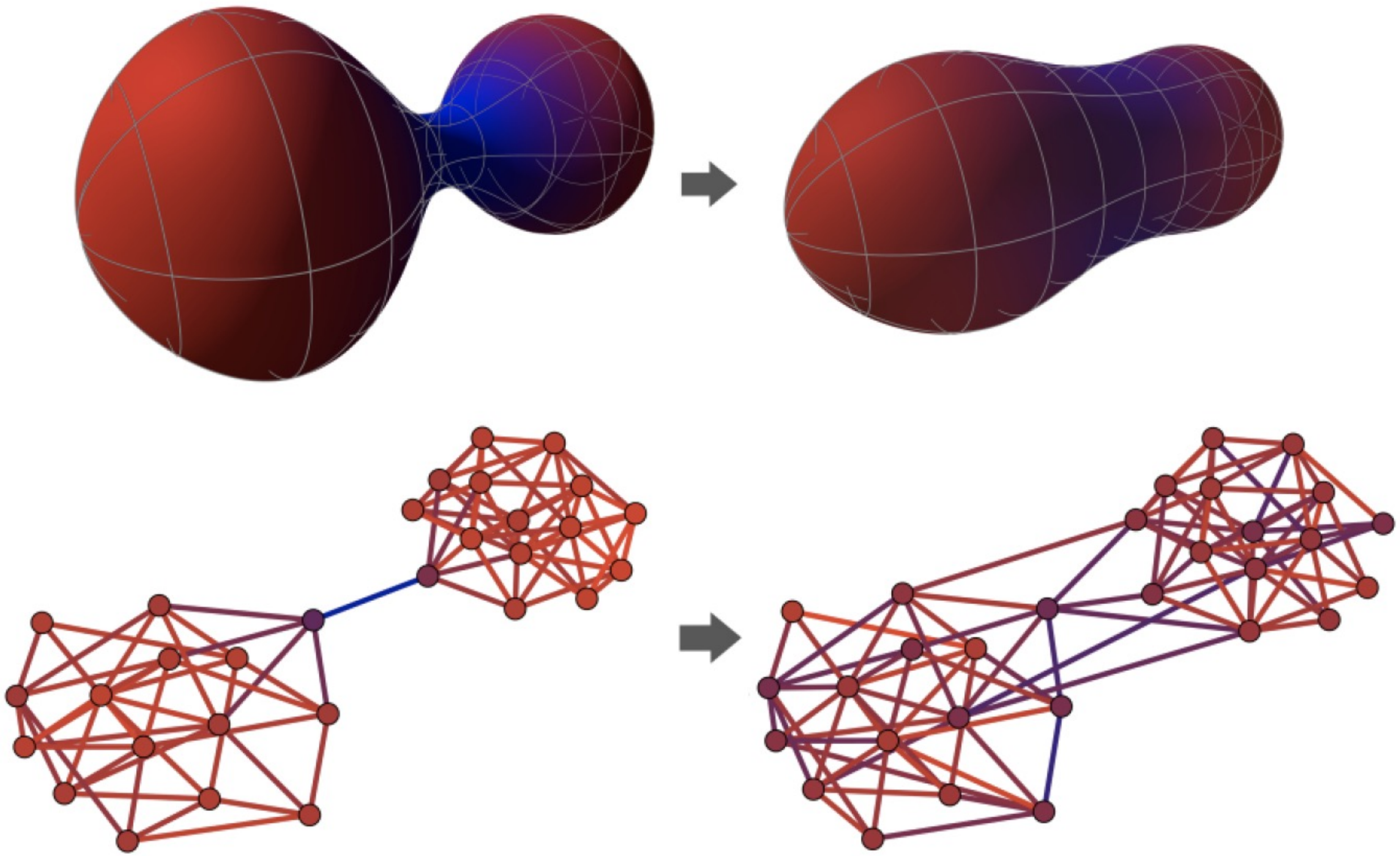


Grid ($= 0$)

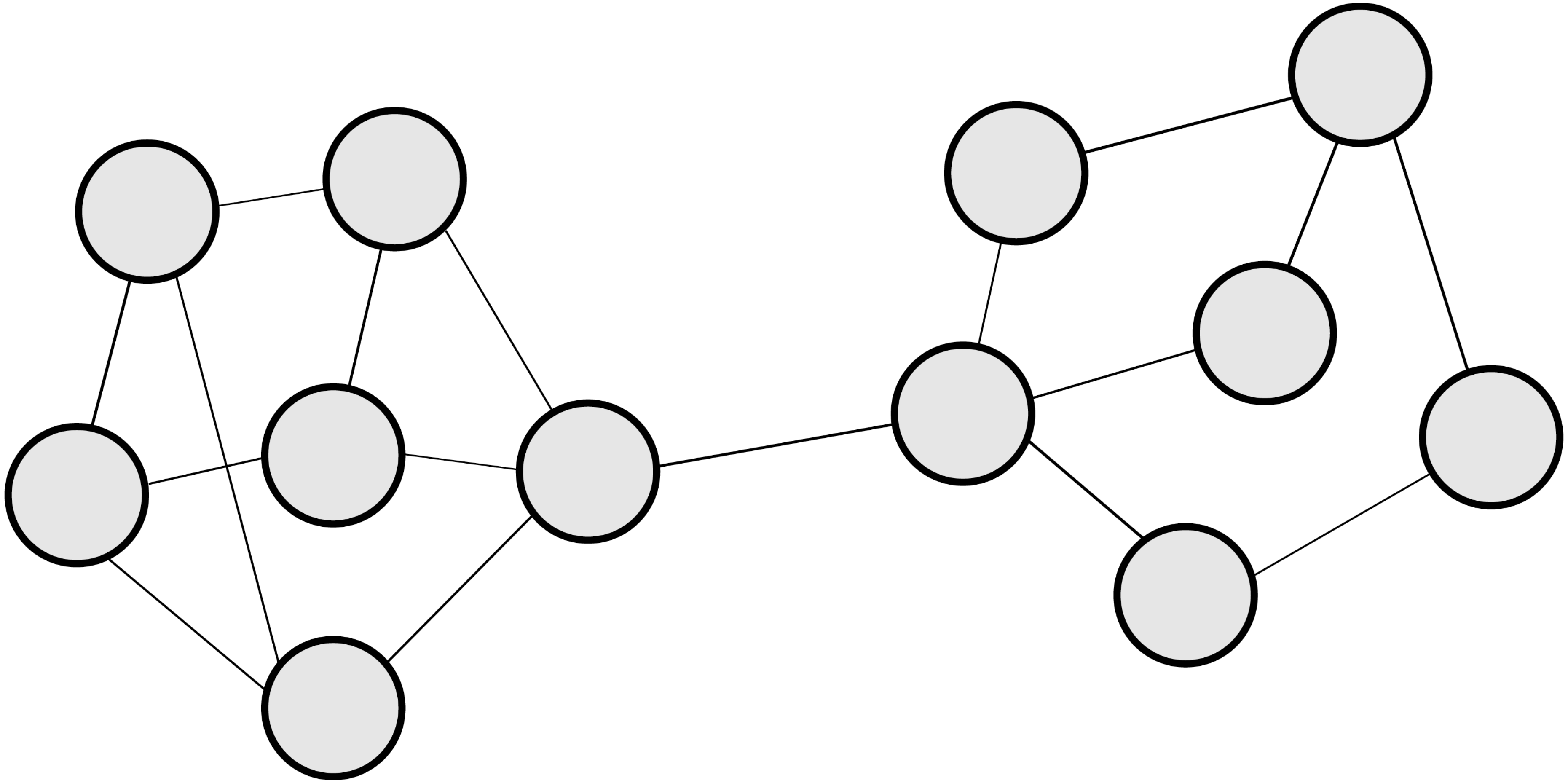


Tree (< 0)

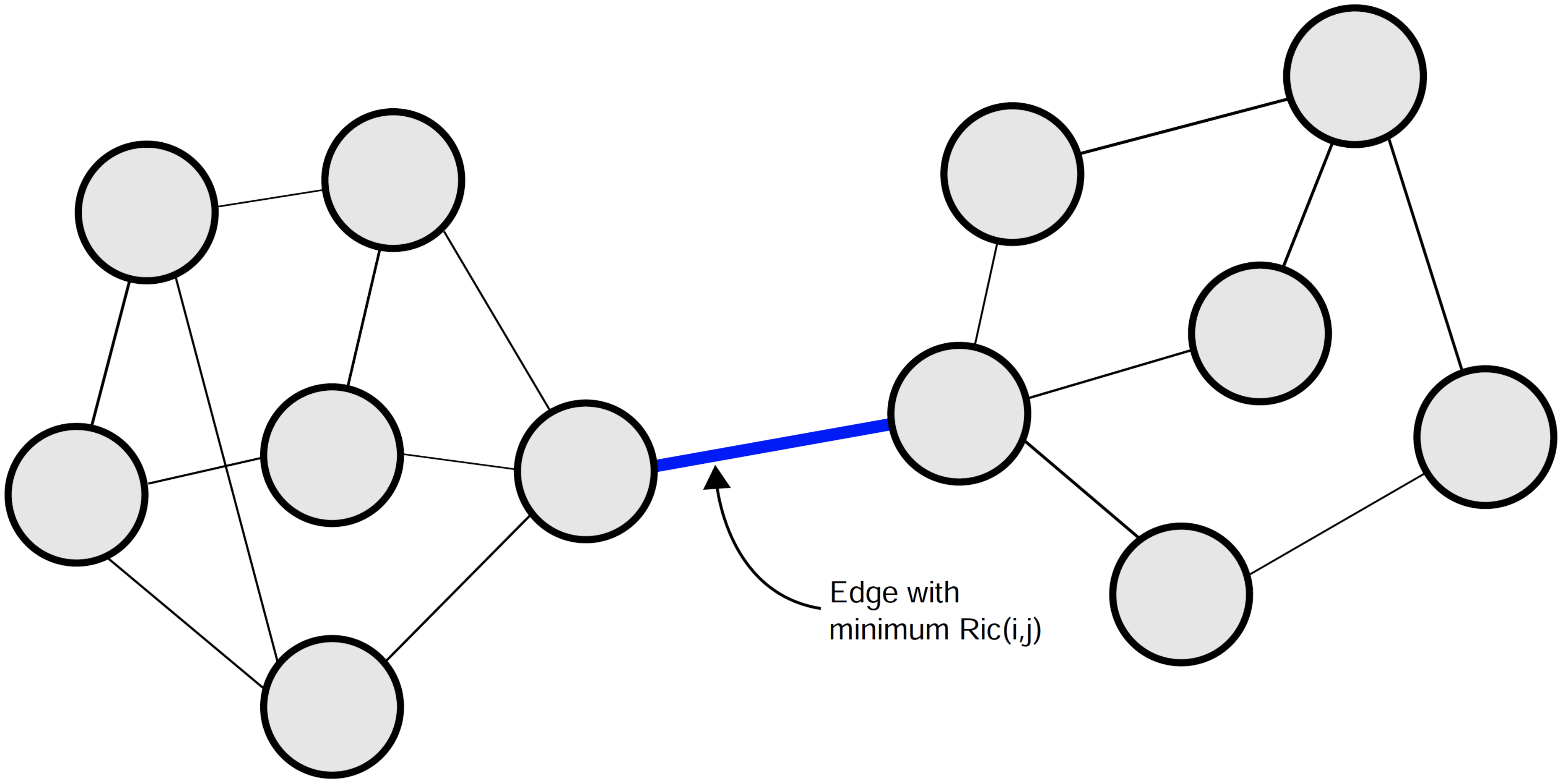
Curvature-inspired alleviation of over-squashing



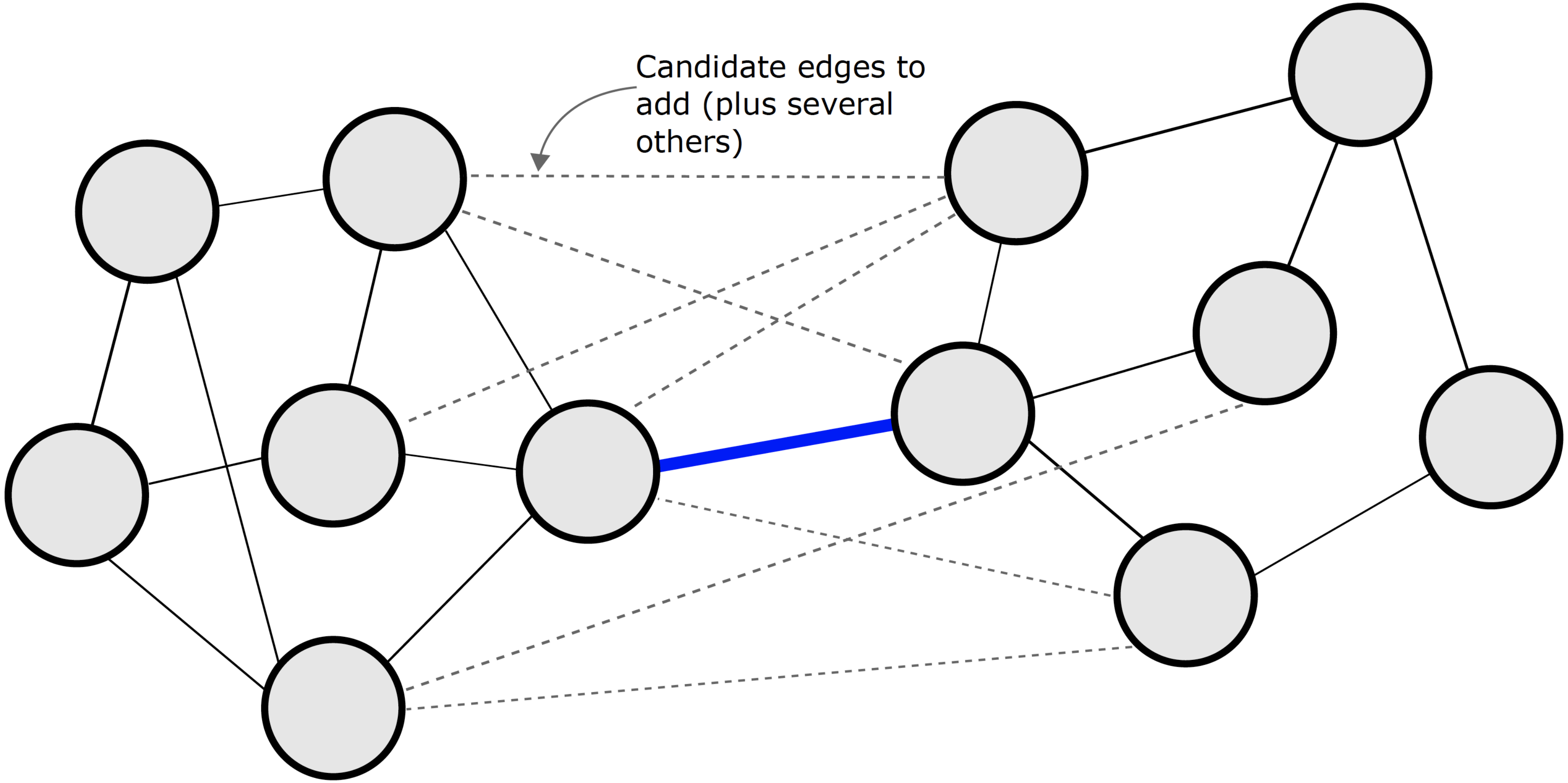
Example



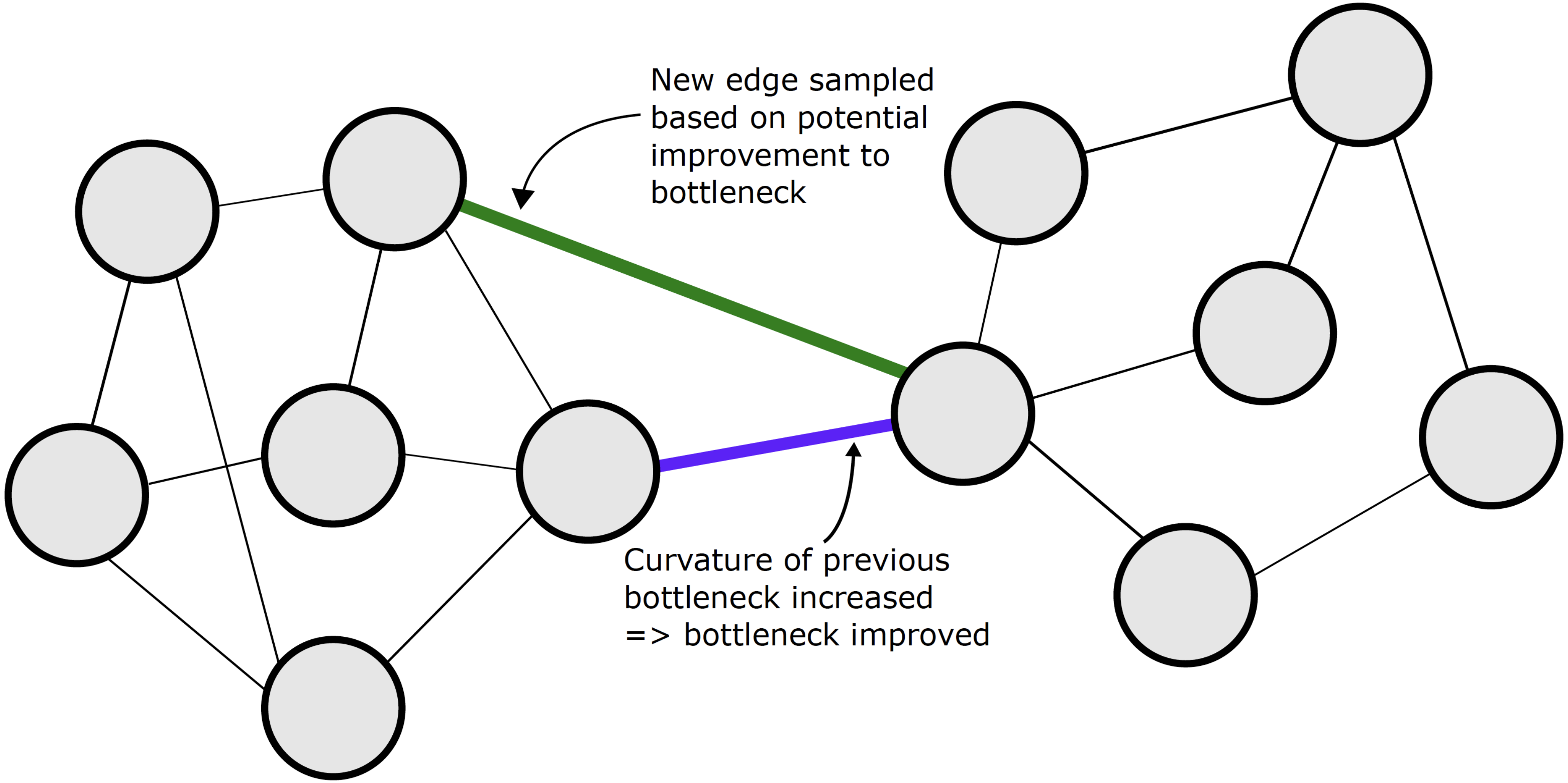
Example



Example

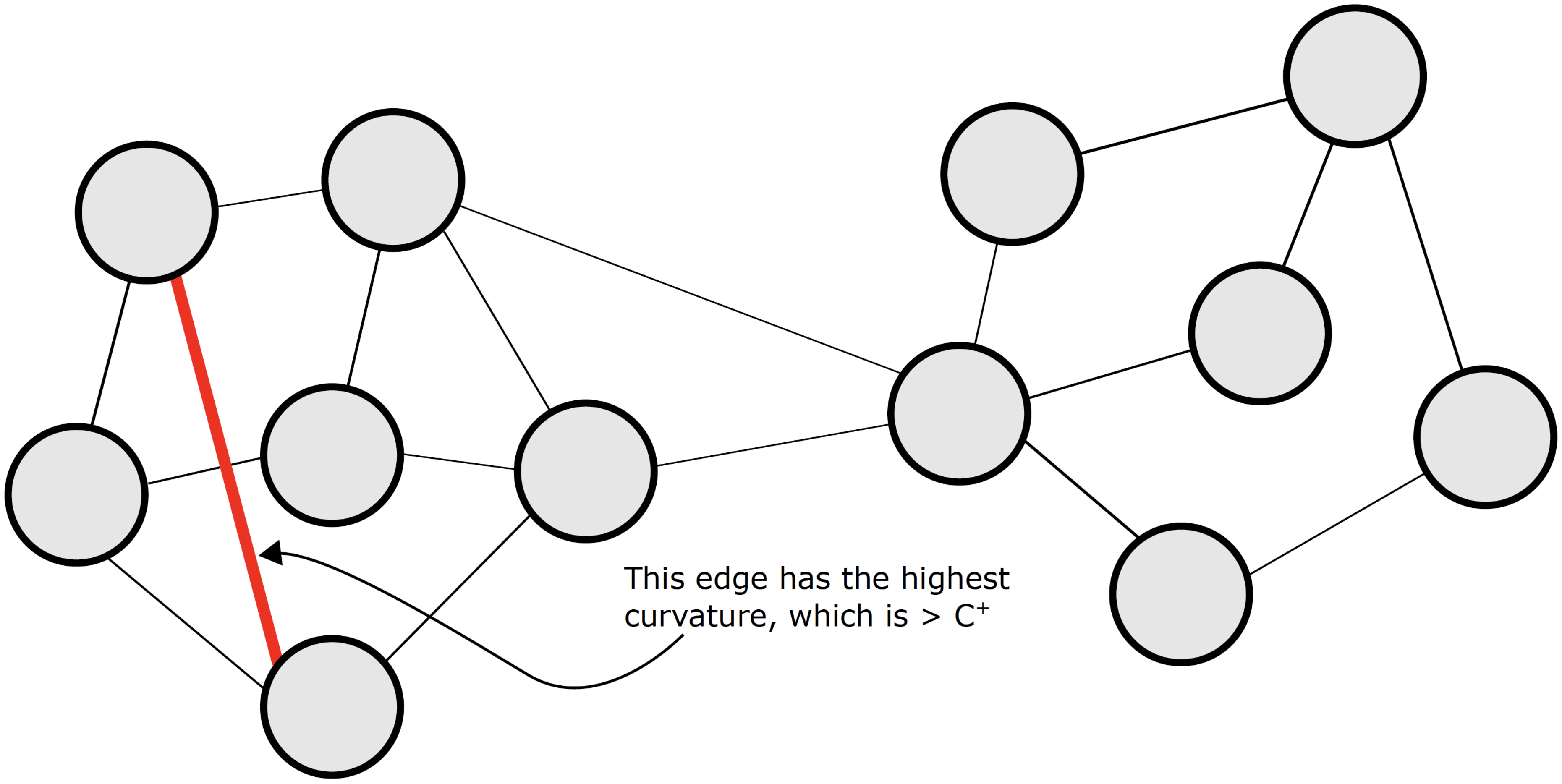


Example

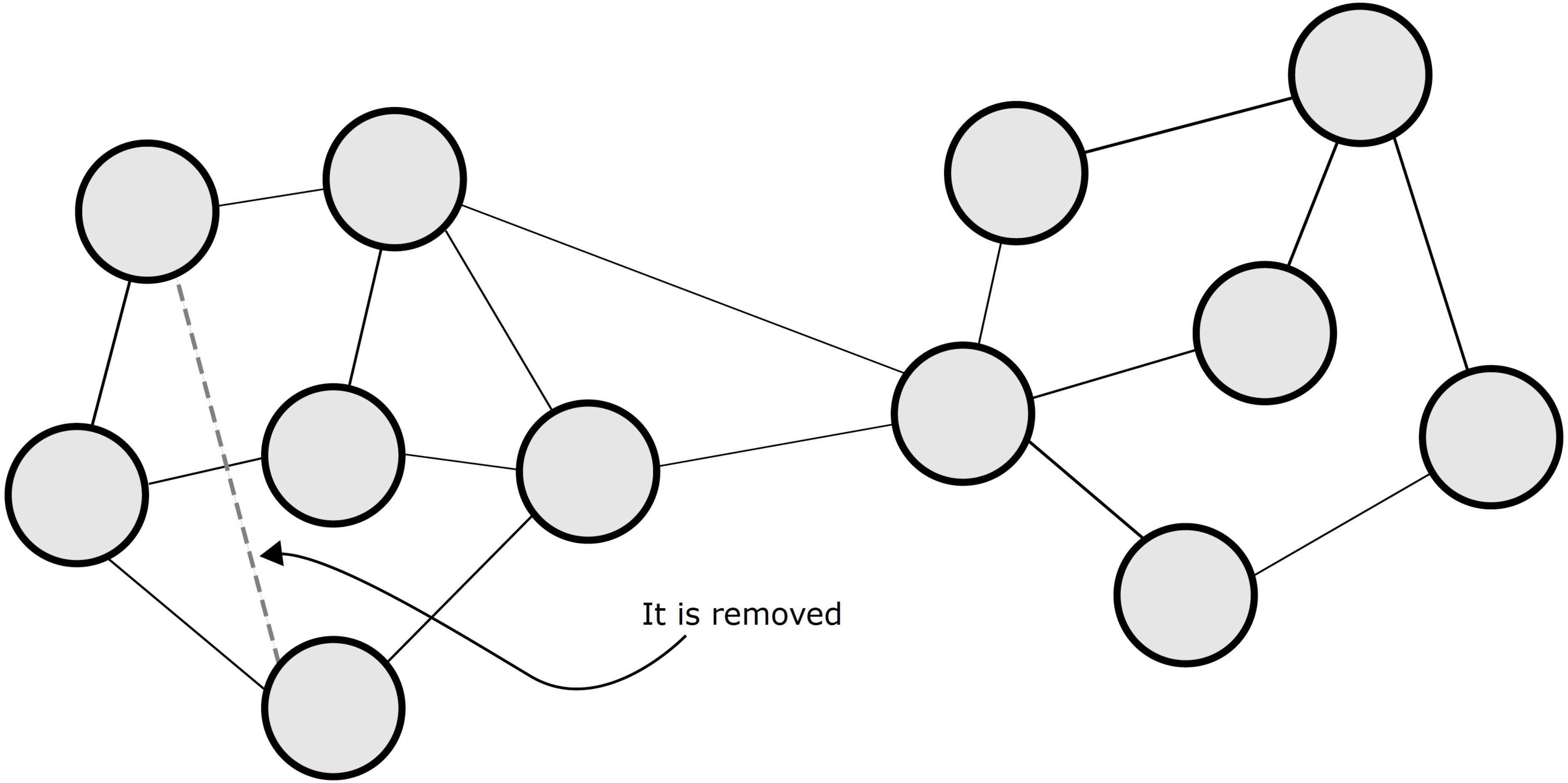


New edge sampled
based on potential
improvement to
bottleneck
Curvature of previous
bottleneck increased
=> bottleneck improved

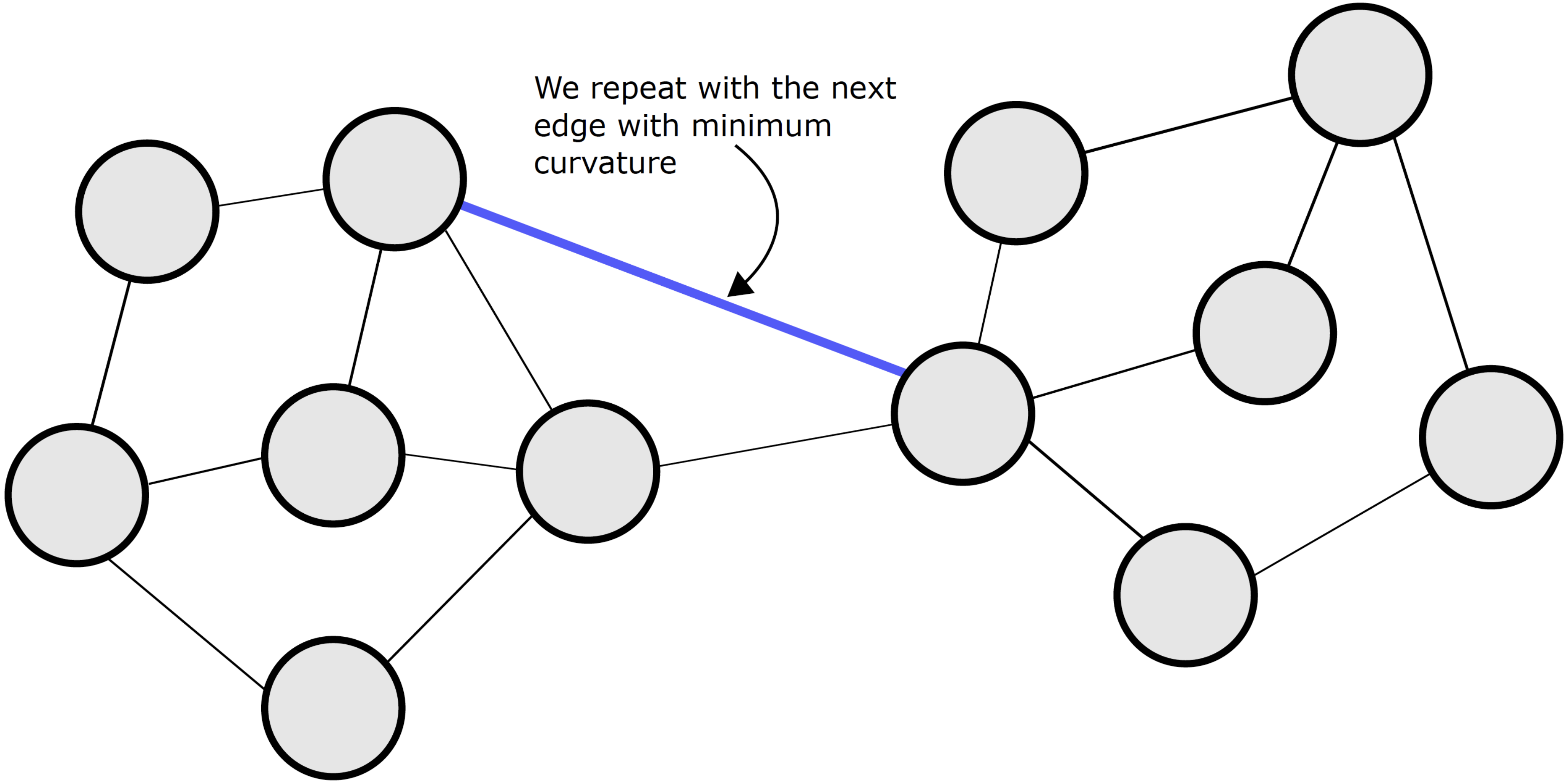
Example



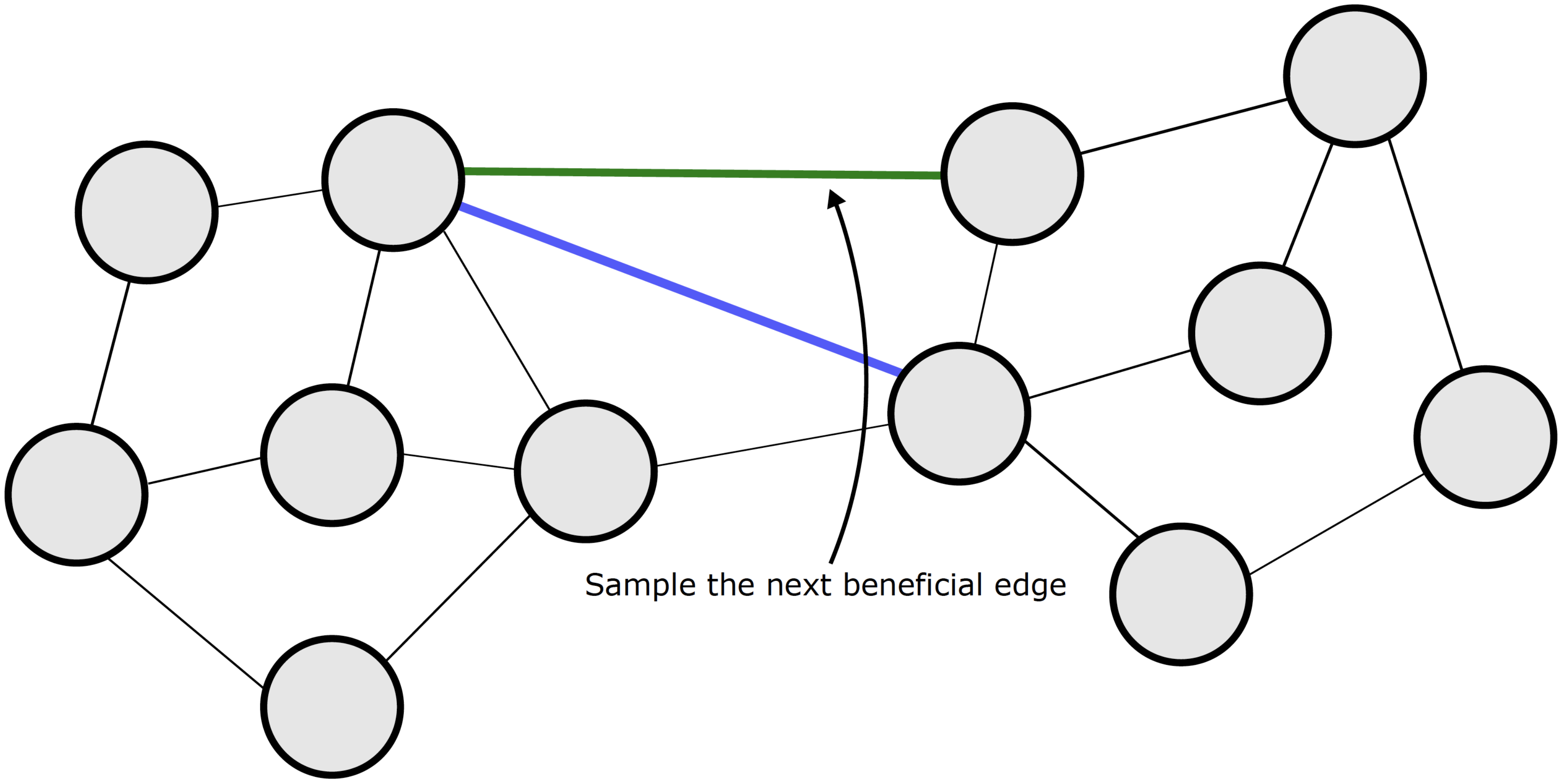
Example



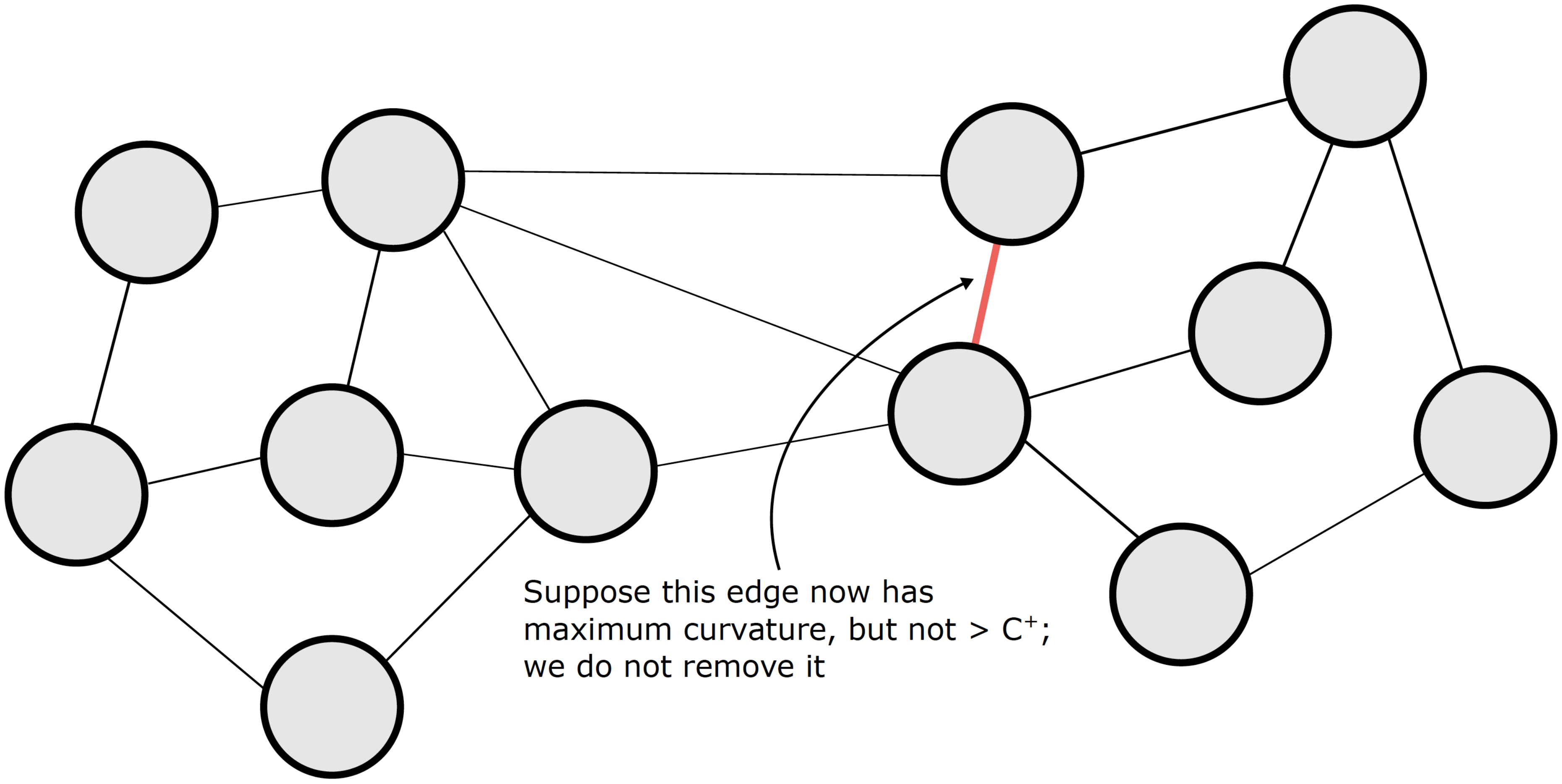
Example



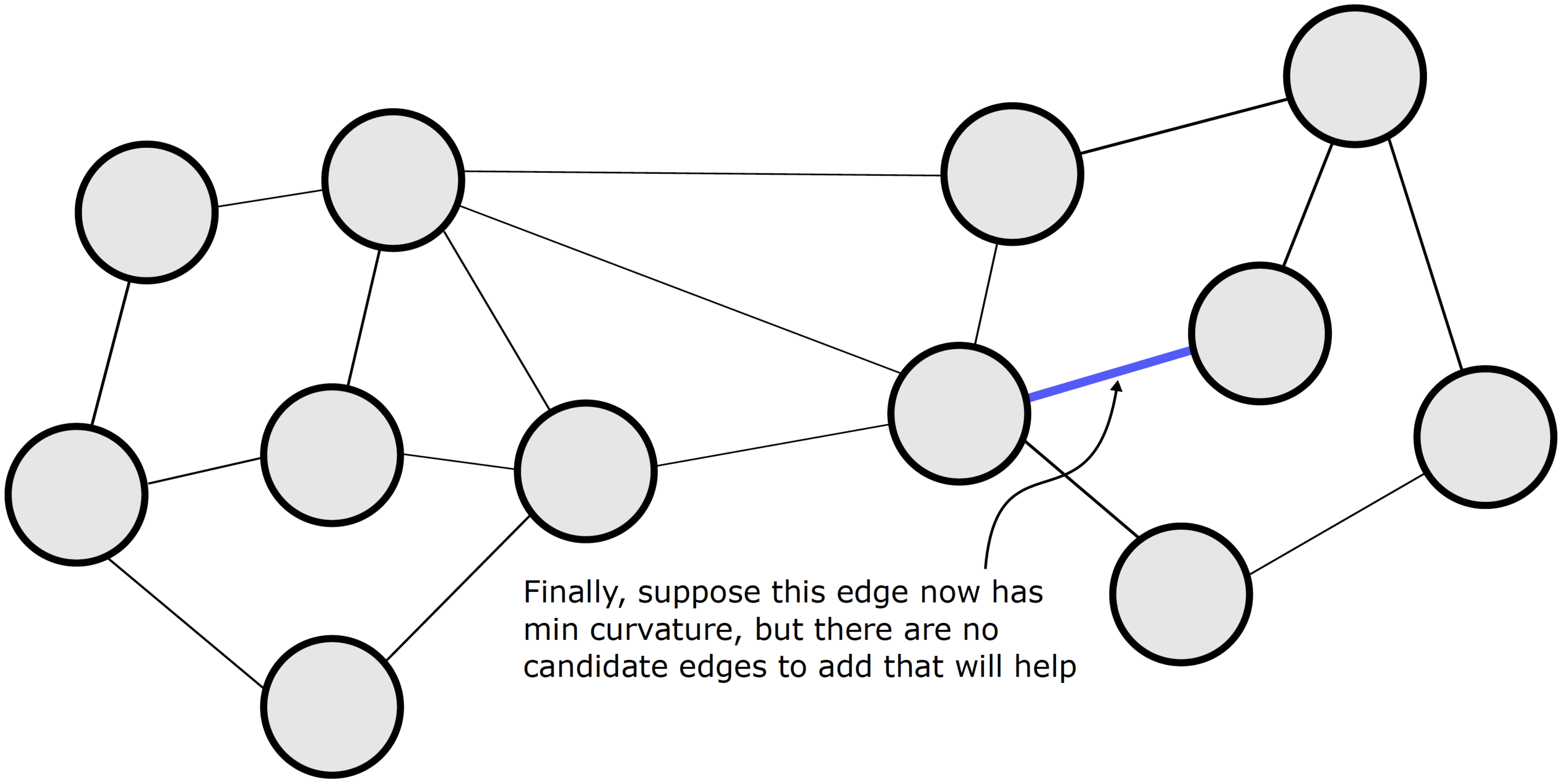
Example



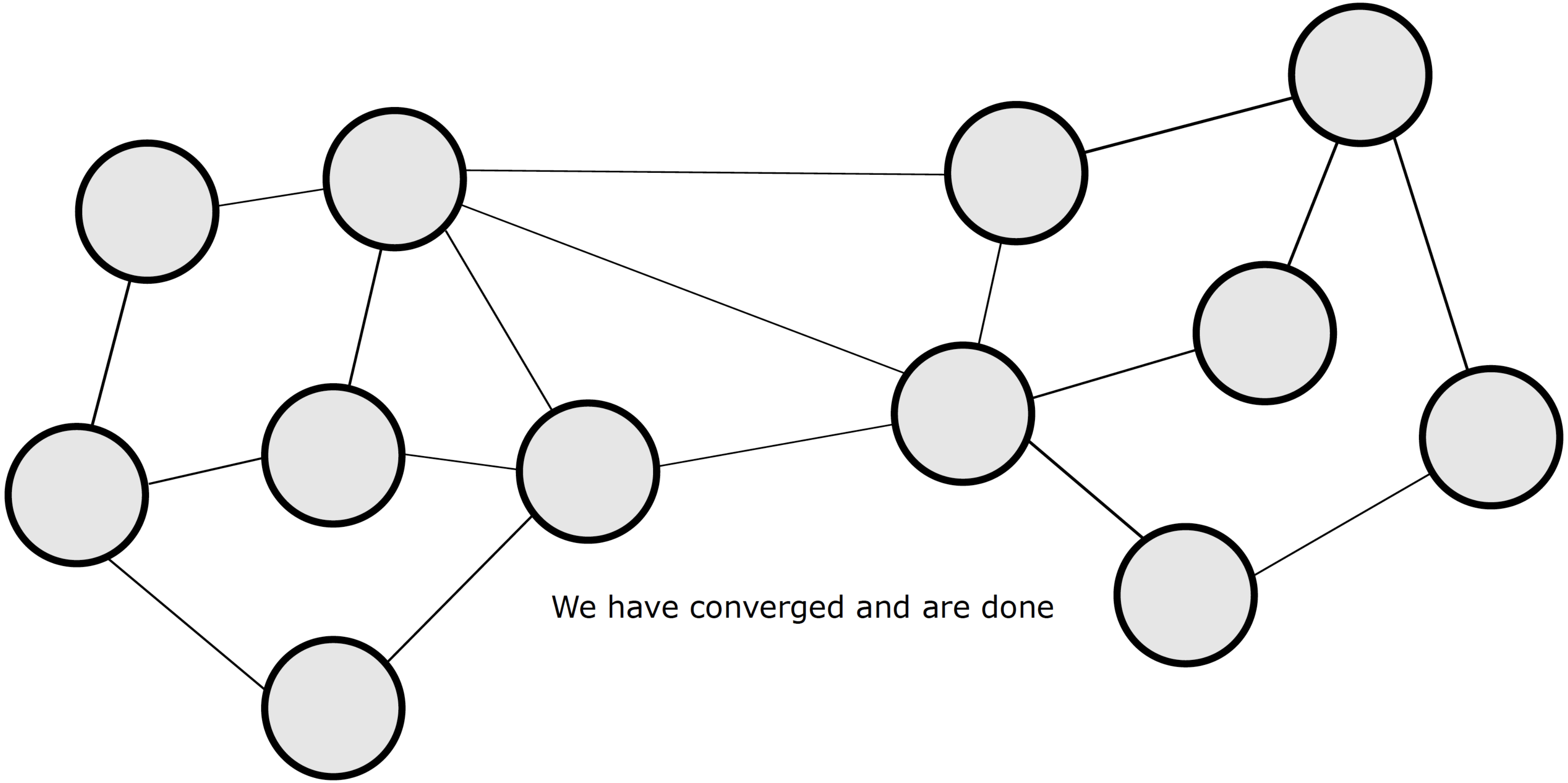
Example


Suppose this edge now has
maximum curvature, but not > C^+;
we do not remove it

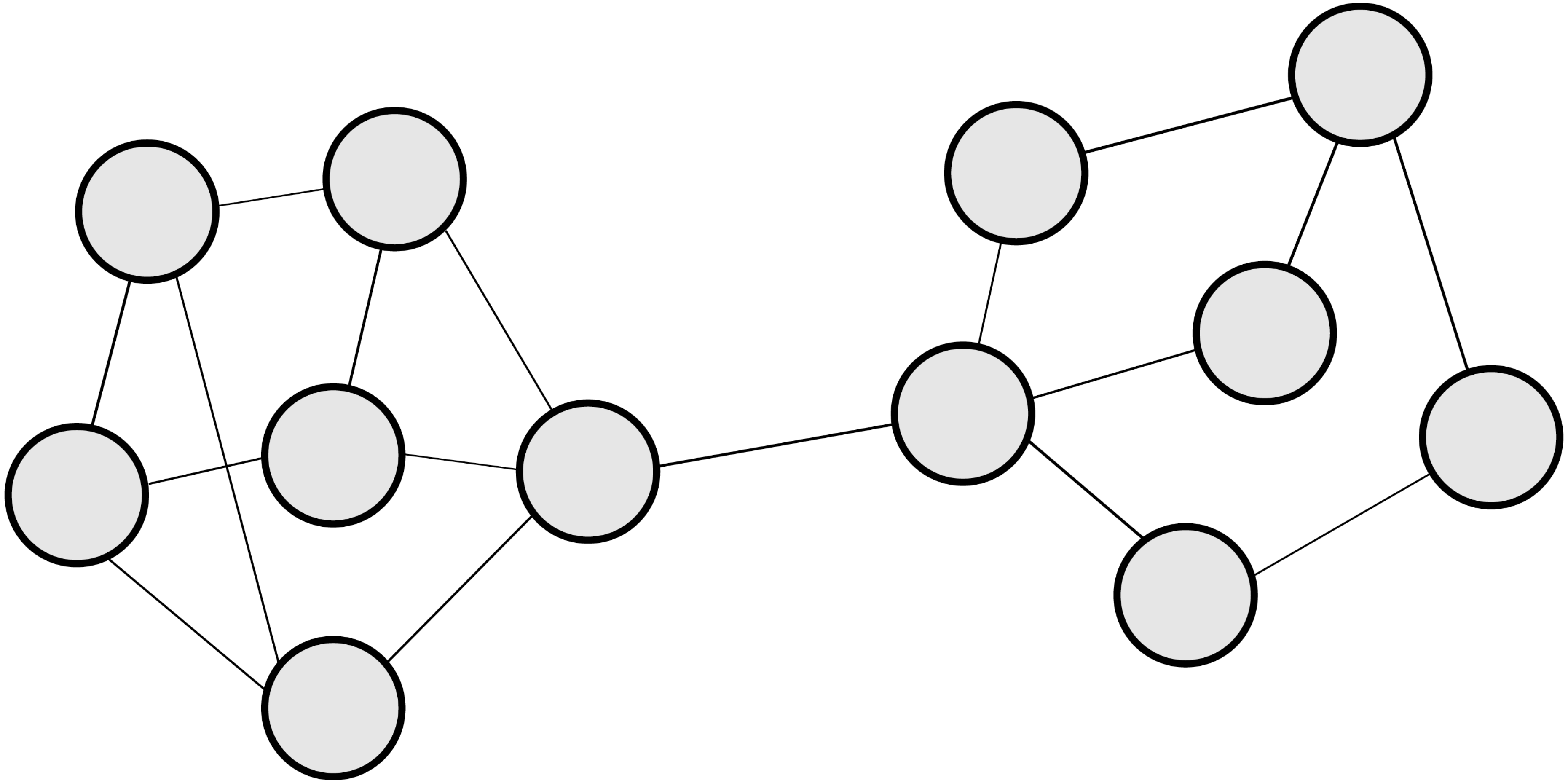
Example


Finally, suppose this edge now has
min curvature, but there are no
candidate edges to add that will help

Example



This is what we started from



Discussion

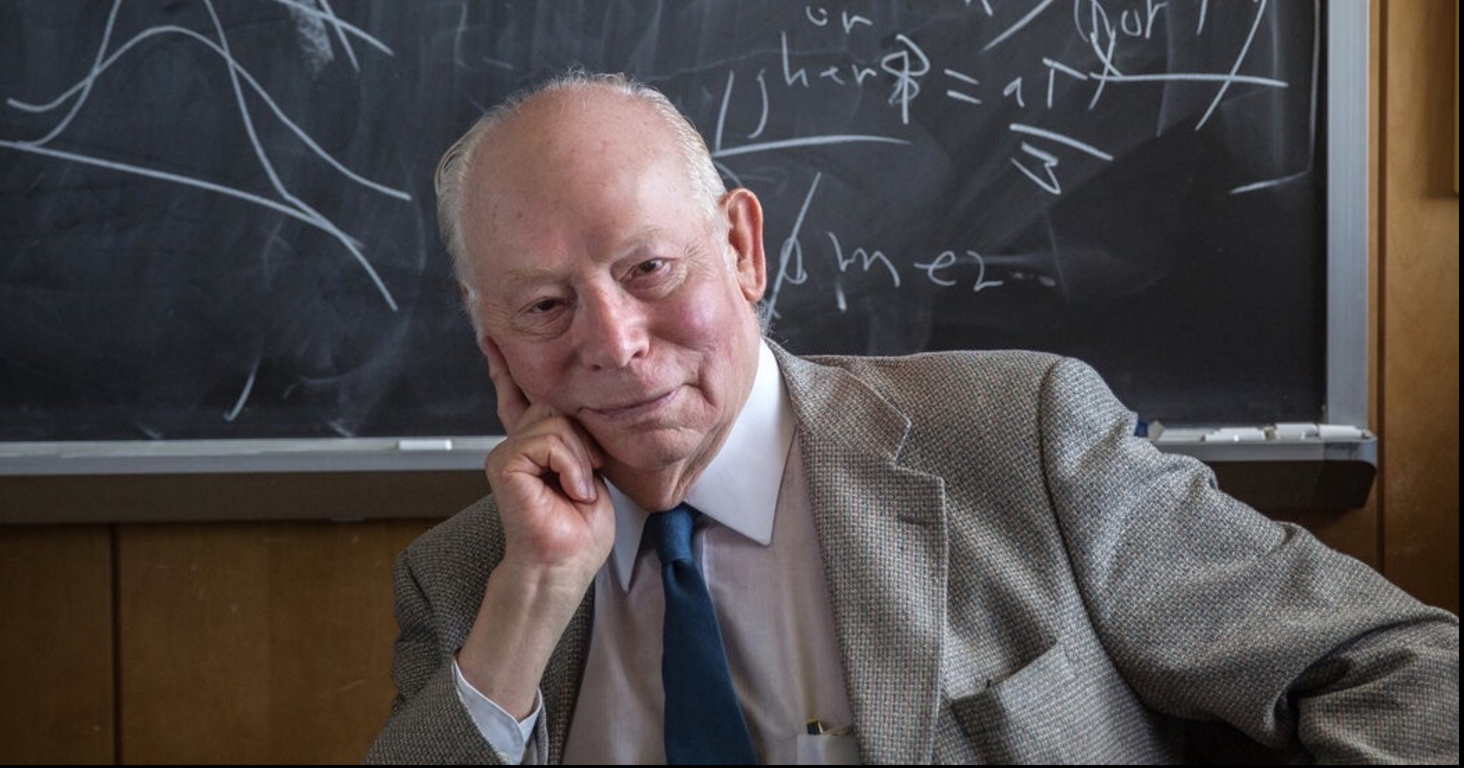
- Improved flow of messages in graph
- But what if structure matters? (which was changed)

“New directions in science are launched by new tools much more often than by new concepts.”

- Freeman Dyson

“Solving intelligence, and then using that to solve everything else.”

- Demis Hassabis, Google DeepMind



"Go for the messes – that's where the action is."

- Steven Weinberg

"Deep Learning today reminiscent of the field of particle physics before the Standard Model: veritable zoo of ~~particles~~ but few unifying principles."

NN architectures

- Michael Bronstein on geometric deep learning (freely quoted)

Summary

- Graph-structured data is everywhere
- Encode & discover relational inductive bias
- Any domain & downstream task
  - Huge impact in particle physics
- Transformers are GNNs

- Very active field of research
- More innovation to come