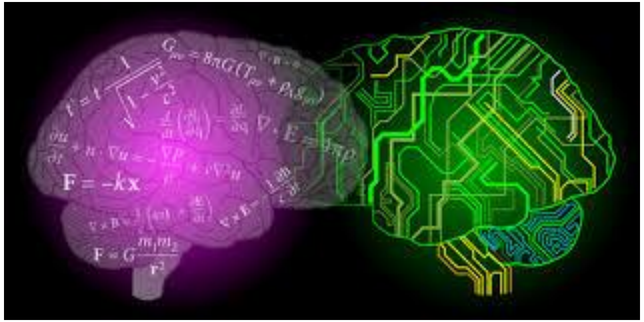
Leveraging Reinforcement Learning, Genetic Algorithms and Transformers for background determination in particle physics

G. Hijano, D. Lancierini, A. Marshall, A. Mauri, P. Owen, M. Patel, K. Petridis, S. R. Qasim, N. Serra, W. Sutcliffe, H. Tilquin

31/03/2026

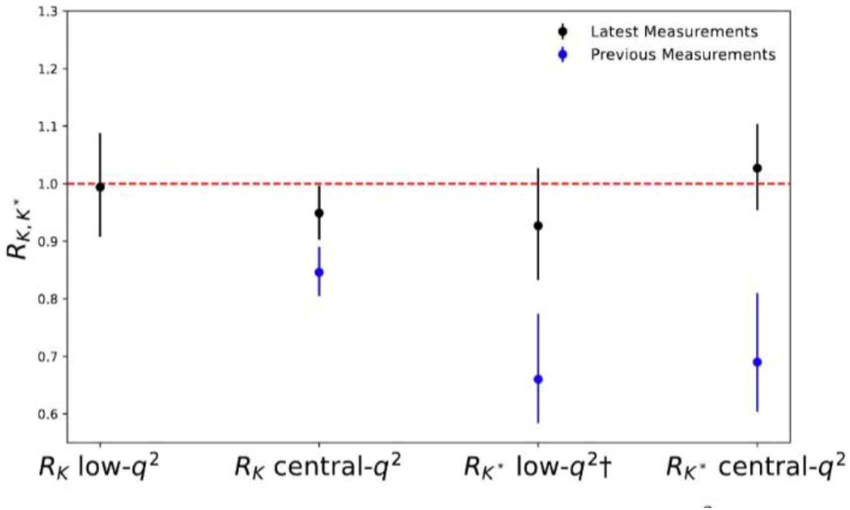
Overview

- Motivation
- Current goals
  - Branching Ratio prediction
  - Momentum distribution prediction
  - Background finder
    - RL



Motivation

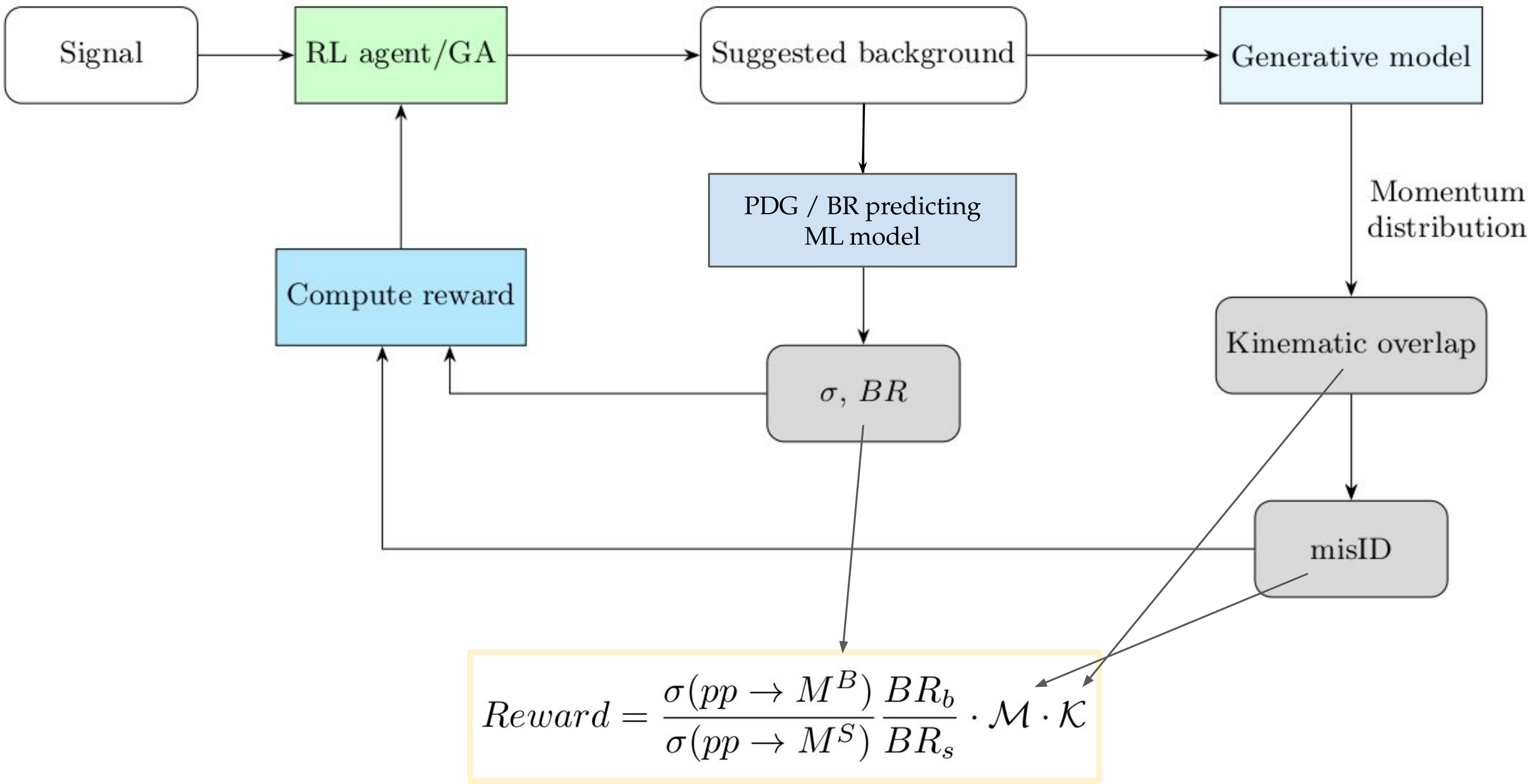
- Design an algorithm to automatize the procedure of finding the **most relevant backgrounds** (events that mimic the response of the signal in the detector) for a signal specified by the user
- Missing backgrounds have led to confusion in analyses performed in the past (LHCb)
  - Example: [2, 3]



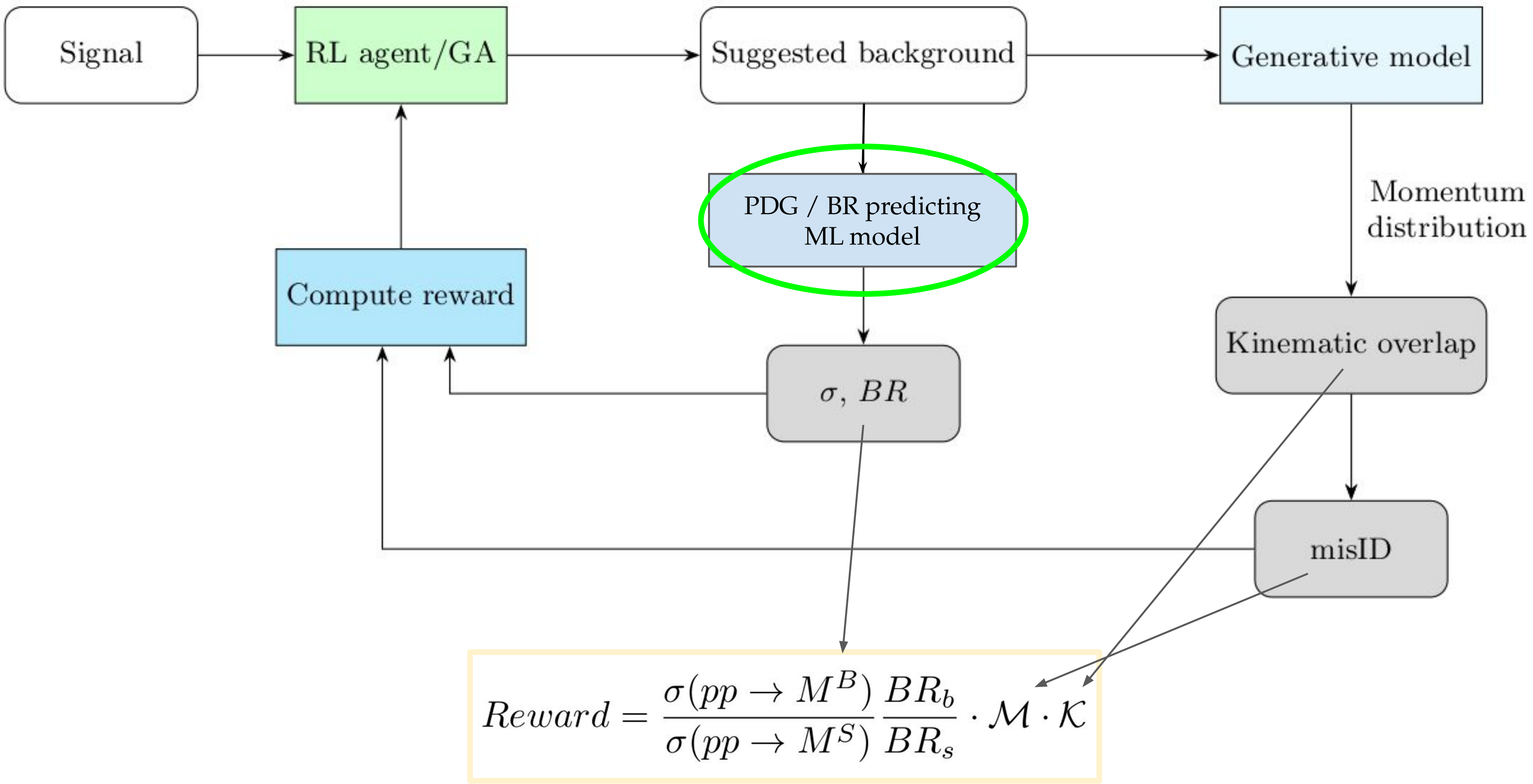
- R_{K,K^*} measurements initially showed **discrepancies** w.r.t. SM predictions (the expected value was 1)
- Statistical fluctuations? Detector or physics mismodellings? New physics?
  - Reason for discrepancy: **missing background**

Workflow

Particle Data Group (PDG): [4]



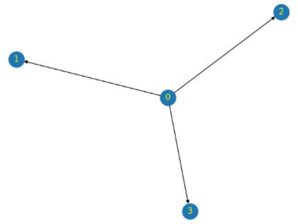
Workflow

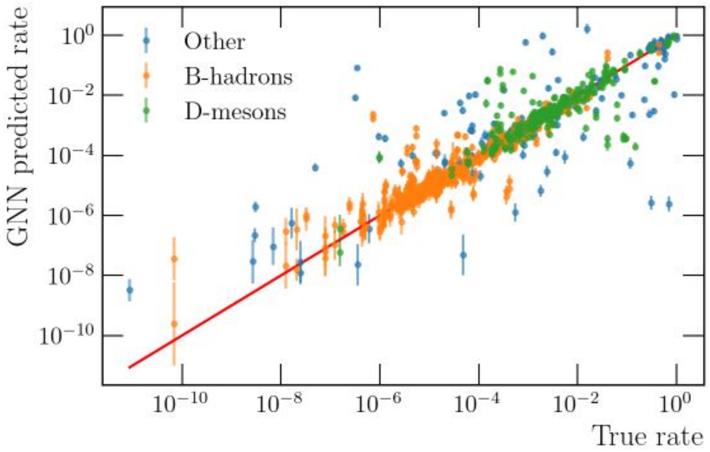


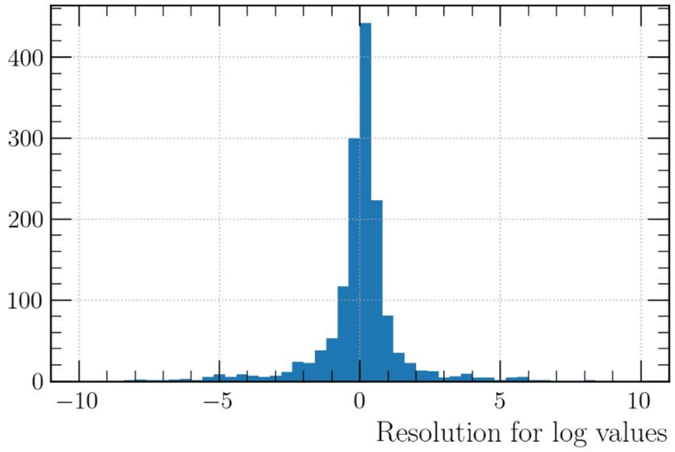
Signal
RL agent/GA
Suggested background
Generative model
PDG / BR predicting ML model
Momentum distribution
Compute reward
Kinematic overlap
σ, BR
misID
$Reward = \frac{\sigma(pp \to M^B)}{\sigma(pp \to M^S)} \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}$

BR predictions

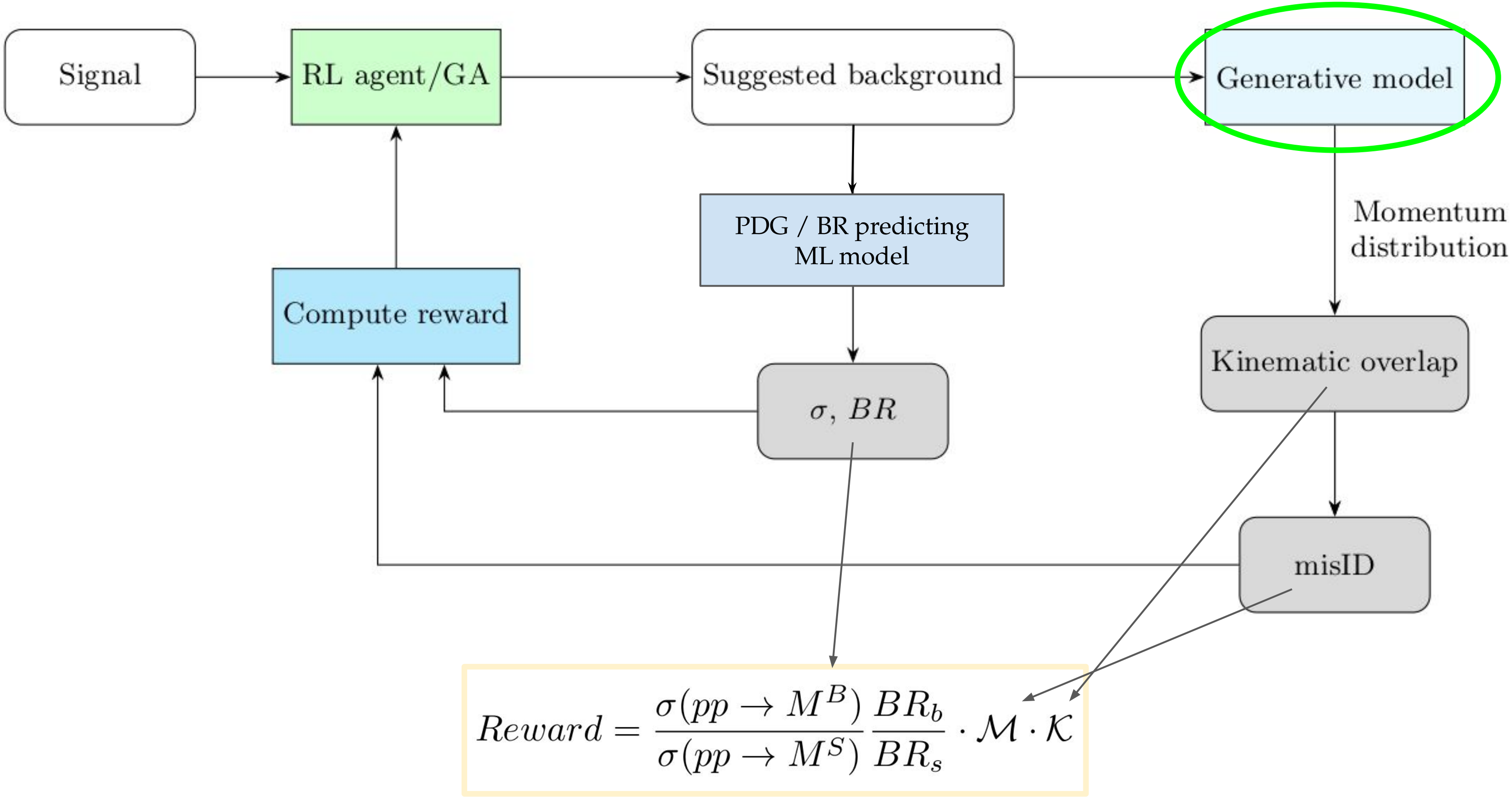
- **Branching Ratio** (BR) predictions for decays not present in the Particle Data Group (PDG)
  - Graph Neural Network (**GNN**) model trained on the PDG data
  - **Graphs** to represent decays
    - Graph is invariant under particle ordering
  - Using a Bayesian architecture to obtain uncertainty on predictions
  - First results seem promising
    - We just need a rough estimate in most cases







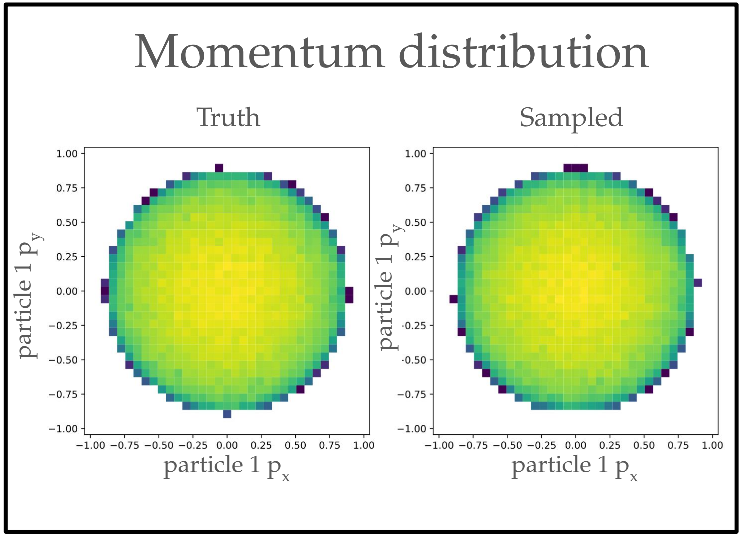
Workflow


Signal
RL agent/GA
Suggested background
Generative model
PDG / BR predicting ML model
Momentum distribution
Compute reward
σ, BR
Kinematic overlap
misID
$Reward = \frac{\sigma(pp \to M^B)}{\sigma(pp \to M^S)} \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}$

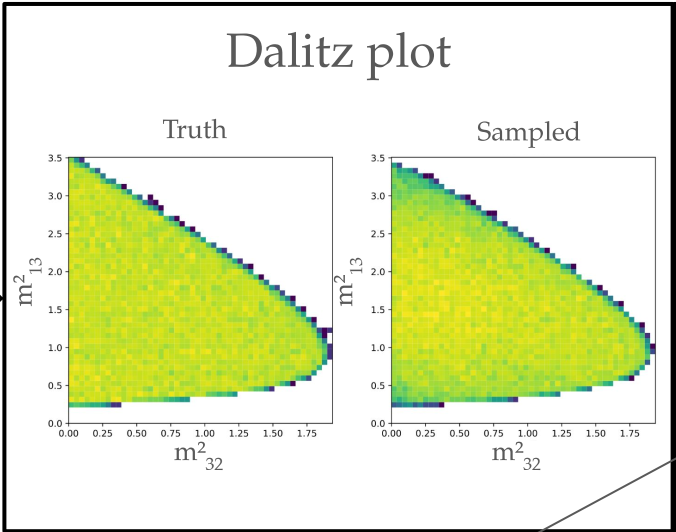
Generative model: generator level variables

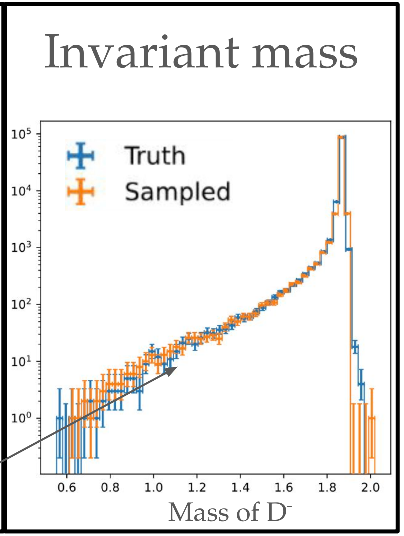
- Given masses of daughter particles, their **momentum distribution** can be estimated to obtain the **kinematic overlap** between signal and background.
- ML is already being used for fast simulation in particle physics, for example, in: [5]
- Using Monte Carlo as truth.
- Example: $D^- \rightarrow K^0 e^-$ anti-ν_e

Variational Autoencoder (**VAE**) model



⇒



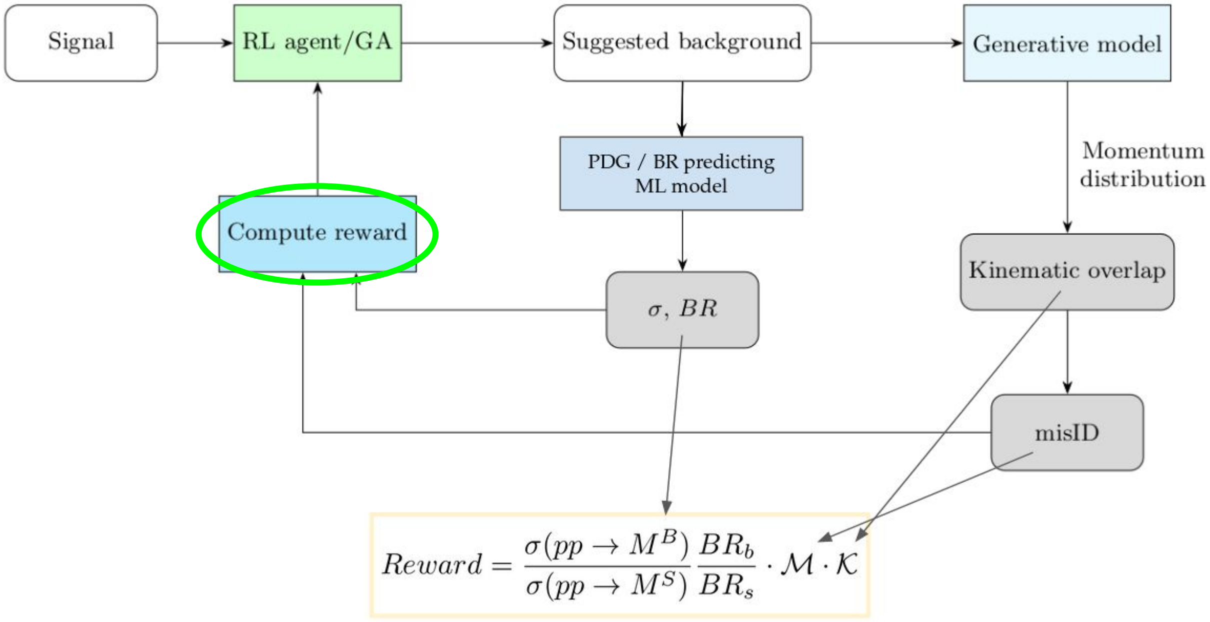


Radiative tail correctly modelled!

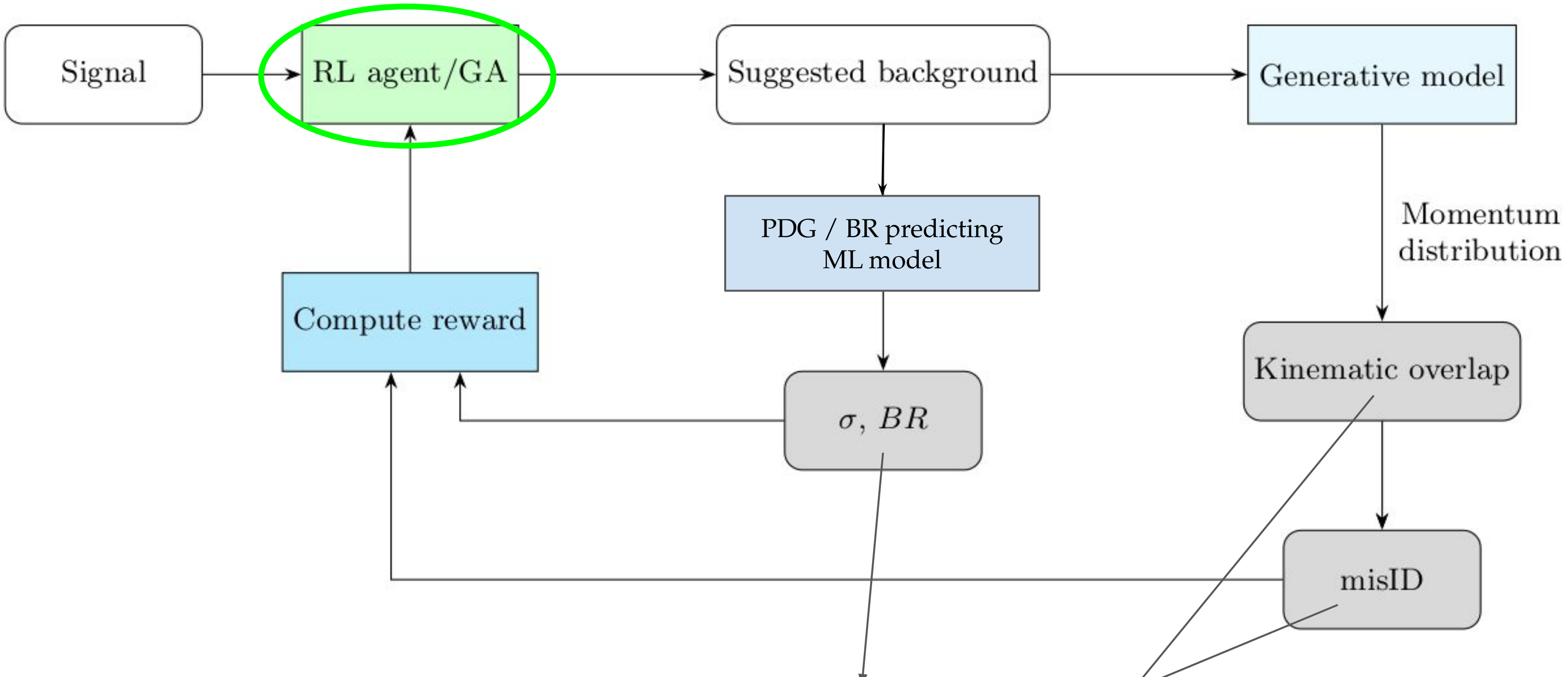
Work published in: [6]

Reward

- **Reward** describes how relevant a background is w.r.t. a signal
- Currently using a toy model for the reward. In the future, ML models previously mentioned will provide input to the agent.



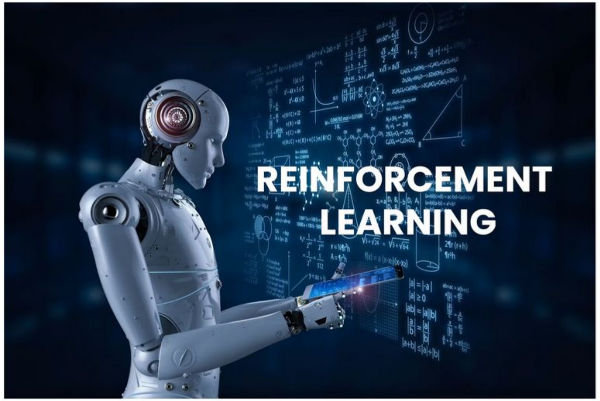
Workflow

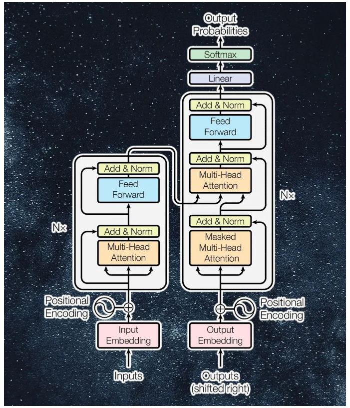


$$Reward = \frac{\sigma(pp \to M^B)}{\sigma(pp \to M^S)} \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}$$

Uncovering of hidden backgrounds

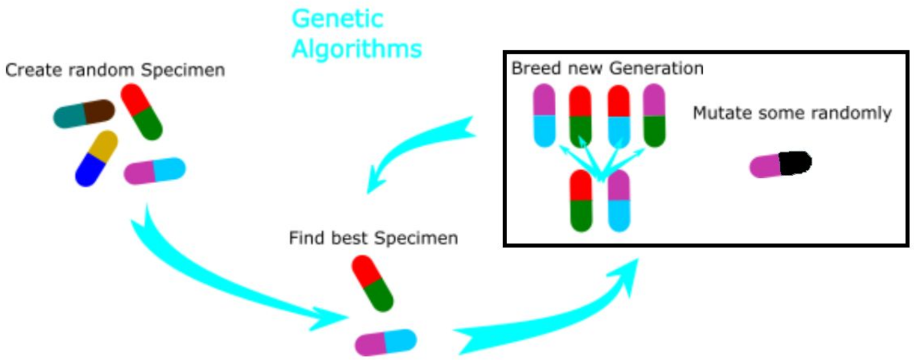
- A nested **for loop** to iterate over the PDG recursively to build all possible decay chains?
  - Impractical. Need a smarter approach.
- Approach combines **Genetic Algorithms (GAs)**, **Reinforcement Learning (RL)** and **transformers**
  - GAs perform an efficient exploration: identify promising trajectories
    - High quality training data in RL training
  - RL agent further explores, learns and generalizes
  - Transformer architecture for agent to deal with token sequences representing decays





Background finder: GAs

- Genetic Algorithms (GAs):
    - **AI** technique to solve **optimization** problems. Inspired by evolution process and natural selection theory.
        - Population of individual solutions represented by its **genes**. Selection of best individuals. **Crossover** and **mutation** to obtain offspring. Iterative process towards an optimal solution.
        - Advantages: Robustness with respect to local maxima or minima. Better results in problems that do not adapt properly to traditional optimization techniques
    - Goal: optimize a fitness function that describes how relevant a background is w.r.t. a signal
    - Individuals are backgrounds

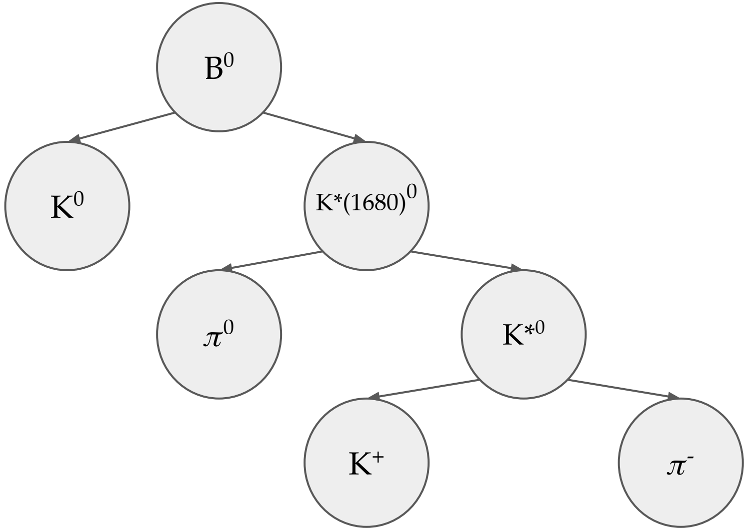


GAs for background finder

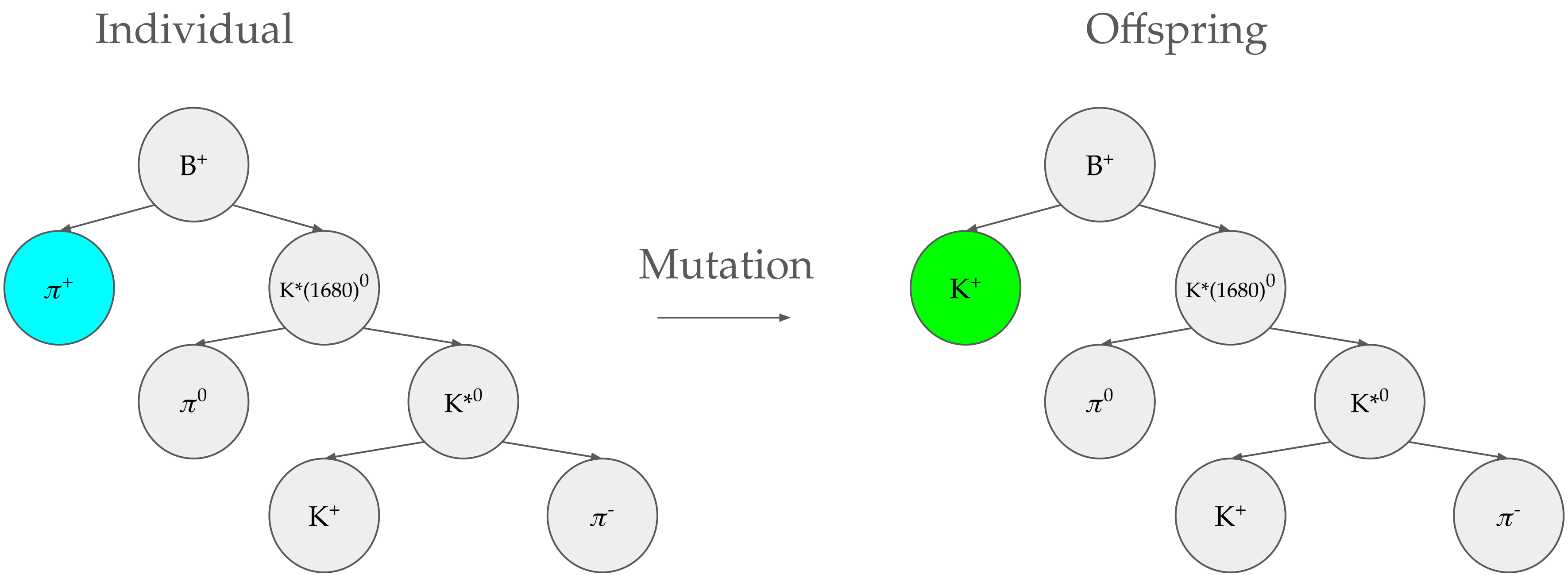
- **Genes** of individuals are represented by a **tree structure** that describes the decay chain
- Apart from the traditional **variation processes** (crossover and mutation) we introduced custom variation processes:
  - To build **intermediate resonances**
  - To make the search more efficiently based on the **physical properties**
  - To be able to use **learnt information** in future optimization problems

Signal: $B^0 \rightarrow K^{*+}(K^+ \pi^0) \pi^-$

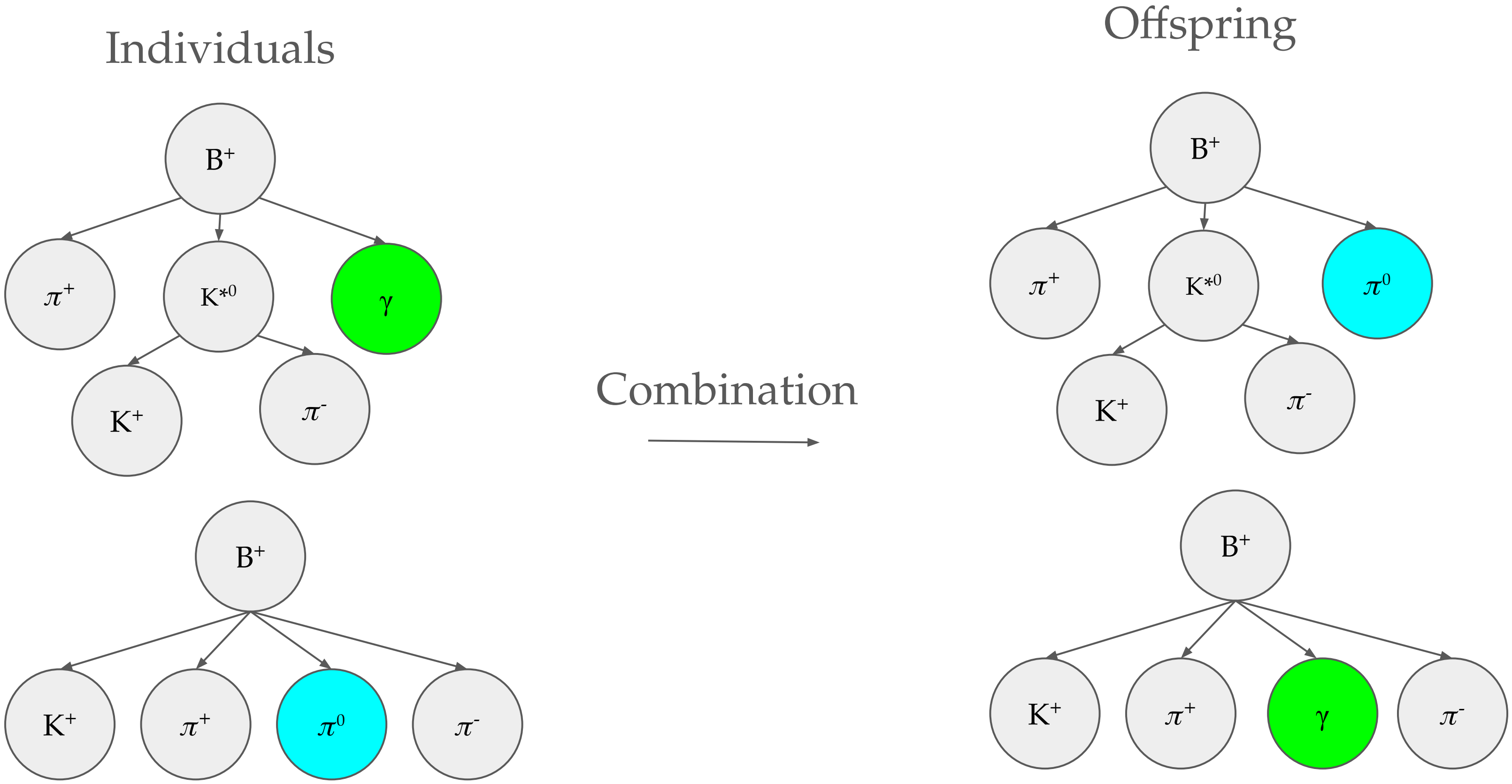
Background candidate:



Mutation

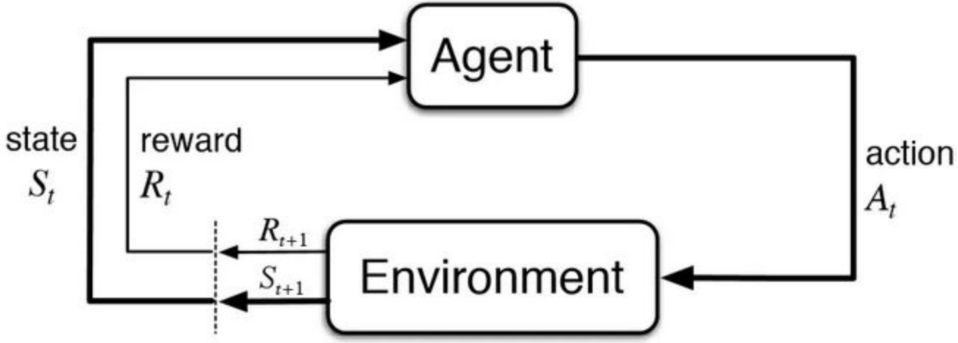

Individual
Offspring
Mutation
B+
π+
K*(1680)0
π0
K*0
K+
π-
B+
K+
K*(1680)0
π0
K*0
K+
π-

Crossover


Individuals
B+
π+
K*0
γ
K+
π-
B+
K+
π+
π0
π-
Combination
Offspring
B+
π+
K*0
π0
K+
π-
B+
K+
π+
γ
π-

Background finder: RL

- Reinforcement Learning (**RL**):
  - Goal: **train a ML model** that can successfully **predict** the most relevant backgrounds for new signals. Agent will learn the decay modes.



RL approach

Token: element of a sequence that needs to be converted into numerical data to provide input to a language model

- **State** will be described by a sequence of tokens
  - State is initialized with the information of the signal
- **Agent** will predict one token in each step, filling up the information of the background
- We can deal with **intermediate resonances**, **misidentifications** and **partially reconstructed backgrounds** with a few tokens.
- Example:

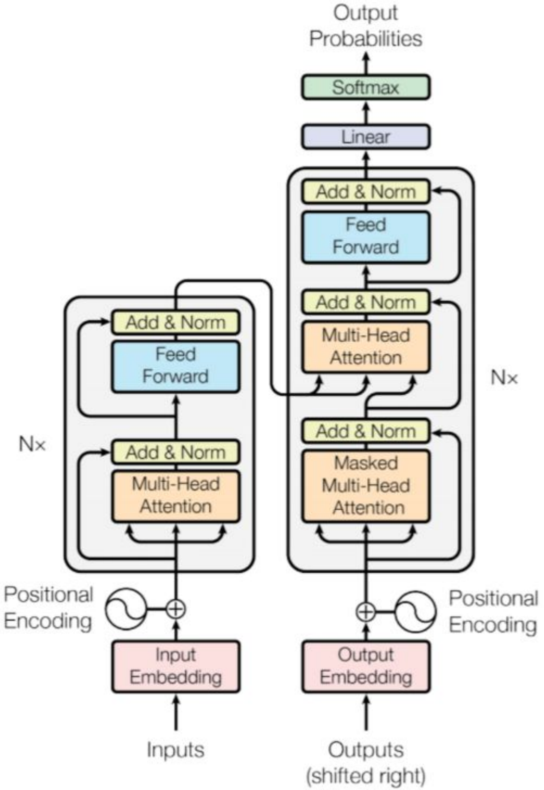
- Signal: B^0-> 0 K^+ 1 π^0 2 π^-
- Background: B^+-> K^{*+}(0 K^+ 1 π^0) LOST π^+ 2 K^- END

K^- and π^- are misidentified

π^+ is not detected

RL strategy

- **Agent**: **transformers** are an ideal architecture for dealing with tokenized sequences
- **Action masking** is applied to mask the tokens that do not make sense for the current state
- **AlphaZero** algorithm (for single player games) to deal with:
  - Large trajectory space
  - Purely terminal rewards
- We use the **GA-discovered solutions** as **expert demonstrations** to guide the RL training for facing the challenge of a highly sparse reward configuration
- Using a **reward shaping** strategy to avoid agent focusing only on the most relevant backgrounds
- Incorporated a **supervised fine tuning** of the agent at the end of its training to improve performance
- Once agent is trained, backgrounds are predicted by sampling actions from the probability distribution provided by the agent.



How we use expert demonstrations

- We use GA-solutions as expert demonstrations during RL training. We explored 2 different strategies:
  - Periodic Expert Guidance: Force the agent to play certain configurations on the expert demonstrations during the training of the agent.
  - Policy Gradient with Supervised Updates: Extend AlphaZero loss by incorporating a supervised term where trajectory losses are weighted by reward.

$$\mathcal{L} = \mathcal{L}_{\text{AZ}} + \lambda \cdot \mathcal{L}_{\text{supervised}}$$

$$= \mathbb{E}_{s\sim\mathcal{D}}\left[(R - v_\theta(s))^2 - \pi^\intercal \log p_\theta(\cdot \mid s)\right] + c||\theta||^2 + \lambda \cdot \mathbb{E}_{\tau\sim\mathcal{D}}\left[\sum_{i=1}^{N_\tau} \frac{R_i}{T_i} \sum_{t=1}^{T_i} - \log p_\theta(a_t^{(i)}|s_t)\right]$$

Reward shaping strategy

- Raw reward:

$$Reward = \frac{\sigma(pp \to M^B)}{\sigma(pp \to M^S)} \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}$$

- Compressed reward:

$$r = \ln\left(1 + k \cdot \frac{\sigma(pp \to M^B)}{\sigma(pp \to M^S)} \cdot \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}\right)$$

Scaling factor

- New reward:

$$R = \alpha r + (1-\alpha)\theta(r - r_\varepsilon) \quad 0 \leq \alpha \leq 1$$

Heaviside function

Threshold to consider a decay is a background

- 2 interesting limits:
  - α=0 ⇒ Equally reward only the relevant backgrounds
  - α=1 ⇒ Previously used reward

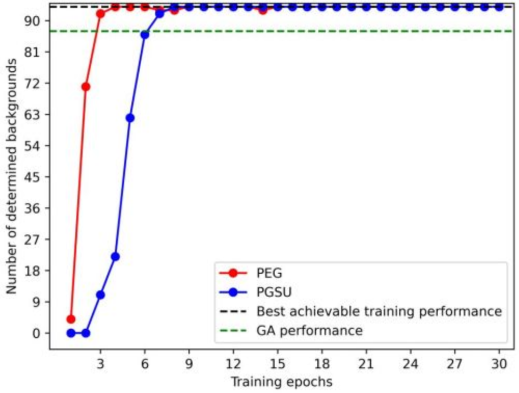
We also penalise the agent reaching invalid states

Experiment

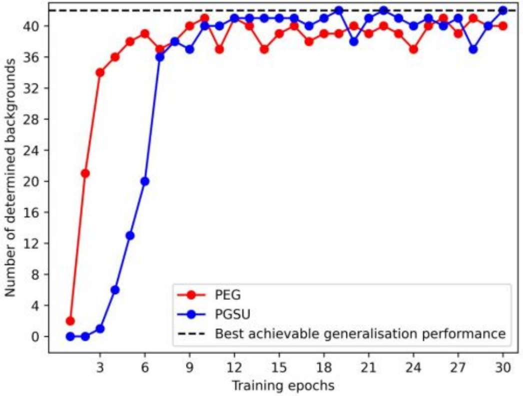
- Trained the agent with **17 training signals** (94 backgrounds)
- Checked its **generalization** ability with **8 new signals** (42 backgrounds). Chosen to be similar to the training signals.
- **Performance**:
  - GA performance (using a population of 6,000 individuals and 40 generations):
    - Found 87/94 training backgrounds
  - RL agent performance (measured building 100,000 sequences per signal):
    - 94/94 training backgrounds learnt by agent
    - 42/42 generalization backgrounds found by agent
- Example:
  - Signal used to check generalization ability (CP conjugate of one of the training signals):
    - B^+-> $\pi^+ \pi^+ \pi^-$ anti-D^0($K^+ \pi^-$)
  - Relevant backgrounds (according to toy model reward):
    - B^+-> $\pi^+ \pi^+ D^-$($K^+ \pi^- \pi^-$)
    - B^+-> $\pi^+ \pi^+ \pi^-$ anti-D^0($K^+ \pi^- \pi^0$)
    - B^+-> π^+ anti-D^0($\pi^+ \pi^- K^{*0}$($K^+ \pi^-$))
    - B^+-> π^+ anti-D^0($\pi^+ \pi^- K^+ \pi^-$)
    - B^+-> $\pi^+ \pi^+ \pi^0 D^{*-}$(π^- anti-D^0($K^+ \pi^-$))
    - B^+-> $\pi^+ \pi^+ D^{*-}$(π^- anti-D^0($K^+ \pi^-$))
    - B^0-> $\pi^+ \pi^+ \pi^- D^-$($K^+ \pi^- \pi^-$)
    - B^+-> $\pi^+ \pi^+ \pi^0 D^-$($K^+ \pi^- \pi^-$)

Experiment: learning curve

- Performance (100.000 games) before fine-tuning:
  - Training: 94/94
  - Generalization: ≃40.5/42

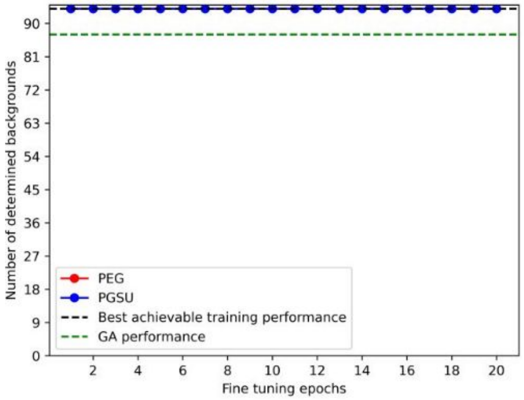


(a) Training performance

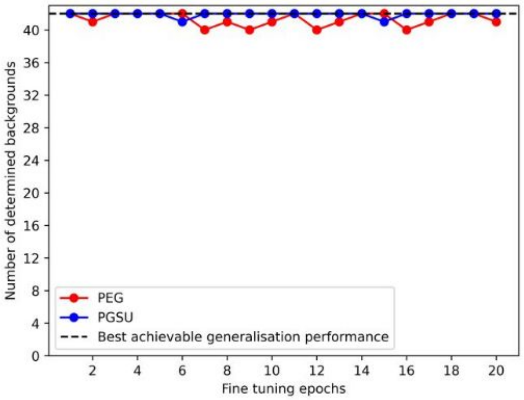


(b) Generalisation performance

- Performance (100.000 games) after fine-tuning:
  - Training: 94/94
  - Generalization: 42/42



(a) Training performance



(b) Generalisation performance

Conclusions

- Goal:
    - Design an algorithm to determine the most relevant backgrounds to:
        - **accelerate the workflow** in particle physics analyses
        - **mitigate human errors**
- Publication status:
    - Momentum distribution prediction studies published [6]
    - Background finder strategy submitted to a journal and available as preprint on arXiv [7]
- **What is next?**
    - The **performance** of each of the models involved seems to be very **promising**
    - Wrap all 3 subprojects together in order to apply this tool to **real case scenarios**

References

[1] A. Golutvin, A. Iniukhin, A. Mauri, P. Owen, N. Serra, and A. Ustyuzhanin, Eur. Phys. J. C 83, 779 (2023)

[2] R. Aaij and others (LHCb Collaboration), Nat. Phys. 18, 277–282 (2022)

[3] R. Aaij and others (LHCb Collaboration), Phys. Rev. D 108, 032002 (2023)

[4] S. Navas et al., Phys. Rev. D 110, 030001 (2024)

[5] G. R. Khattak, S. Vallecorsa, F. Carminati, and G. M. Khan, Eur. Phys. J. C 82, 386 (2022)

[6] G. Hijano, D. Lancierini, A. M. Marshall, A. Mauri, P. Owen, M. Patel, K. Petridis, S. R. Qasim, N. Serra, W. Sutcliffe, and H. Tilquin, Phys. Rev. D 113, 012005 (2026)

[7] G. Hijano, D. Lancierini, A. M. Marshall, A. Mauri, P. Owen, M. Patel, K. Petridis, S. R. Qasim, N. Serra, W. Sutcliffe, and H. Tilquin, Leveraging Reinforcement Learning, Genetic Algorithms and Transformers for background determination in particle physics (2025), arXiv:2509.14894v2

Thank you very much for your attention

Auxiliar slides

Results

3 Hall of fame:

3.1 B0-> pi+ D*-(pi- anti-D0(K+ pi- pi0))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B0-> pi+ pi0 D*-(pi- anti-D0(K+ pi-))	0.605	✓	✓
No. 2	B0-> pi+ pi- anti-D0(K+ pi- pi0)	0.371	✓	✓
No. 3	B+-> pi+ pi+ pi- anti-D0(K+ pi- pi0)	0.31	✓	✓
No. 4	B0-> pi+ D-(pi- pi0 K*0(K+ pi-))	0.281	✓	✓
No. 5	B0-> pi+ D*-(pi0 D-(K+ pi- pi-))	0.267	✓	✓
No. 6	B0-> pi+ pi0 D*-(pi- anti-D0(K+ pi- pi0))	0.26	✓	✓
No. 7	B0-> pi+ pi0 D-(K+ pi- pi-)	0.166	✓	✓
No. 8	B+-> pi+ pi+ pi0 D*-(pi- anti-D0(K+ pi-))	0.142	×	✓
No. 9	B0-> pi+ pi- pi0 anti-D0(K+ pi-)	0.14	✓	✓
No. 10	B0-> pi+ pi0 D*-(pi0 D-(K+ pi- pi-))	0.087	✓	✓
No. 11	B0-> pi+ D-(K+ pi- pi- pi0)	0.084	✓	✓
No. 12	B+-> pi+ pi+ pi0 D-(K+ pi- pi-)	0.07	✓	✓
No. 13	B0-> pi+ pi- anti-D*0(pi0 anti-D0(K+ pi-))	0.056	✓	✓
No. 14	B0-> pi+ pi- pi0 anti-D0(K+ pi- pi0)	0.052	✓	✓

3.2 anti-B0-> mu- anti-nu_mu D+(K+ pi+ K-)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	anti-B0-> mu- anti-nu_mu D*+(pi+ D0(K+ K-))	1.139	✓	✓
No. 2	anti-B0-> mu- anti-nu_mu D+(K+ anti-K*0(pi+ K-))	0.414	✓	✓
No. 3	anti-B0-> mu- anti-nu_mu D+(pi+ pi+ K-)	0.287	✓	✓
No. 4	anti-B0-> pi+ mu- anti-nu_mu D0(K+ K-)	0.242	✓	✓
No. 5	anti-B0-> mu- anti-nu_mu D*+(pi+ D0(pi+ K-))	0.19	✓	✓

3.3 anti-B0-> pi- pi0 D*+(pi+ D0(pi+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	anti-B0-> pi- D*+(pi+ D0(pi+ K- pi0))	0.79	✓	✓
No. 2	anti-B0-> pi+ pi- D0(pi+ K- pi0)	0.432	✓	✓
No. 3	B–> pi+ pi- pi- D0(pi+ K- pi0)	0.363	✓	✓
No. 4	anti-B0-> pi- D+(pi+ pi0 anti-K*0(pi+ K-))	0.329	✓	✓
No. 5	anti-B0-> pi- D*+(pi0 D+(pi+ pi+ K-))	0.314	✓	✓
No. 6	anti-B0-> pi- pi0 D*+(pi+ D0(pi+ K- pi0))	0.306	✓	✓
No. 7	anti-B0-> pi- pi0 D+(pi+ pi+ K-)	0.197	✓	✓
No. 8	B–> pi- pi- pi0 D*+(pi+ D0(pi+ K-))	0.168	✓	✓
No. 9	anti-B0-> pi+ pi- pi0 D0(pi+ K-)	0.166	✓	✓
No. 10	anti-B0-> pi- pi0 D*+(pi0 D+(pi+ pi+ K-))	0.104	✓	✓
No. 11	anti-B0-> pi- D+(pi+ pi+ K- pi0)	0.1	✓	✓
No. 12	B–> pi- pi- pi0 D+(pi+ pi+ K-)	0.083	✓	✓
No. 13	anti-B0-> pi+ pi- D*0(pi0 D0(pi+ K-))	0.067	✓	✓
No. 14	anti-B0-> pi+ pi- pi0 D0(pi+ K- pi0)	0.062	✓	✓
No. 15	B–> pi- pi- D*+(pi+ D0(pi+ K- pi0))	0.057	✓	✓

3.4 anti-B0-> pi- D*+(pi+ D0(pi+ K- pi0))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	anti-B0-> pi- pi0 D*+(pi+ D0(pi+ K-))	0.605	✓	✓
No. 2	anti-B0-> pi+ pi- D0(pi+ K- pi0)	0.371	✓	✓
No. 3	B–> pi+ pi- pi- D0(pi+ K- pi0)	0.31	✓	✓
No. 4	anti-B0-> pi- D+(pi+ pi0 anti-K*0(pi+ K-))	0.281	✓	✓
No. 5	anti-B0-> pi- D*+(pi0 D+(pi+ pi+ K-))	0.267	✓	✓
No. 6	anti-B0-> pi- pi0 D*+(pi+ D0(pi+ K- pi0))	0.26	✓	✓
No. 7	anti-B0-> pi- pi0 D+(pi+ pi+ K-)	0.166	✓	✓
No. 8	B–> pi- pi- pi0 D*+(pi+ D0(pi+ K-))	0.142	✓	✓
No. 9	anti-B0-> pi+ pi- pi0 D0(pi+ K-)	0.14	✓	✓
No. 10	anti-B0-> pi- pi0 D*+(pi0 D+(pi+ pi+ K-))	0.087	✓	✓
No. 11	anti-B0-> pi- D+(pi+ pi+ K- pi0)	0.084	✓	✓
No. 12	B–> pi- pi- pi0 D+(pi+ pi+ K-)	0.07	✓	✓
No. 13	anti-B0-> pi+ pi- D*0(pi0 D0(pi+ K-))	0.056	✓	✓
No. 14	anti-B0-> pi+ pi- pi0 D0(pi+ K- pi0)	0.052	✓	✓

3.5 B+-> mu+ nu_mu anti-D0(K+ pi- pi0)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B+-> mu+ nu_mu anti-D*0(pi0 anti-D0(K+ pi-))	0.361	✓	✓

3.6 B--> e- anti-nu_e D0(pi+ K- pi0)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> e- anti-nu_e D*0(pi0 D0(pi+ K-))	0.361	✓	✓

3.7 B0-> mu+ nu_mu D*-(pi- anti-D0(K+ pi-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B0-> mu+ nu_mu D-(K+ pi- pi-)	0.95	✓	✓
No. 2	B0-> mu+ pi- nu_mu anti-D0(K+ pi-)	0.121	✓	✓

3.8 B--> pi+ pi- pi- D0(pi+ K-)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> pi- pi- D+(pi+ pi+ K-)	0.322	✓	✓
No. 2	B–> pi+ pi- pi- D0(pi+ K- pi0)	0.306	✓	✓
No. 3	B–> pi- D0(pi+ pi- anti-K*0(pi+ K-))	0.256	✓	✓
No. 4	B–> pi- D0(pi+ pi+ K- pi-)	0.218	✓	✓
No. 5	B–> pi- pi- pi0 D*+(pi+ D0(pi+ K-))	0.139	✓	✓
No. 6	B–> pi- pi- D*+(pi+ D0(pi+ K-))	0.126	✓	✓
No. 7	anti-B0-> pi+ pi- pi- D+(pi+ pi+ K-)	0.069	✓	✓
No. 8	B–> pi- pi- pi0 D+(pi+ pi+ K-)	0.069	✓	✓

3.9 anti-B0-> mu- anti-nu_mu D*+(pi+ D0(pi+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	anti-B0-> mu- anti-nu_mu D+(pi+ pi+ K-)	0.95	✓	✓
No. 2	anti-B0-> pi+ mu- anti-nu_mu D0(pi+ K-)	0.121	✓	✓

3.10 B--> e- anti-nu_e D*0(pi0 D0(pi+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> e- anti-nu_e D0(pi+ K- pi0)	1.195	✓	✓

3.11 B+-> pi+ pi+ D-(K+ pi- pi-)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B+-> pi+ pi+ pi- anti-D0(K+ pi-)	1.289	✓	✓
No. 2	B+-> pi+ pi+ pi- anti-D0(K+ pi- pi0)	0.663	✓	✓
No. 3	B+-> pi+ anti-D0(pi+ pi- K*0(K+ pi-))	0.57	✓	✓
No. 4	B+-> pi+ anti-D0(K+ pi+ pi- pi-)	0.495	✓	✓
No. 5	B+-> pi+ pi+ pi0 D*-(pi- anti-D0(K+ pi-))	0.331	×	✓
No. 6	B+-> pi+ pi+ D*-(pi- anti-D0(K+ pi-))	0.303	✓	✓
No. 7	B0-> pi+ pi+ pi- D-(K+ pi- pi-)	0.171	✓	✓
No. 8	B+-> pi+ pi+ pi0 D-(K+ pi- pi-)	0.171	✓	✓
No. 9	B+-> pi+ pi+ D*-(pi- anti-D0(K+ pi- pi0))	0.119	✓	✓
No. 10	B+-> pi+ pi+ pi- K_0*0(K+ pi-)	0.064	✓	✓
No. 11	B+-> pi+ pi+ pi- K*0(K+ pi-)	0.064	✓	✓
No. 12	B+-> pi+ anti-D0(pi+ pi- pi0 K*0(K+ pi-))	0.059	×	✓

3.12 B--> mu- anti-nu_mu D*0(pi0 D0(K+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> mu- anti-nu_mu D0(pi+ K- pi0)	0.205	✓	✓
No. 2	B–> mu- anti-nu_mu D0(K+ K- pi0)	0.1	✓	✓
No. 3	B–> mu- anti-nu_mu D*0(pi0 D0(pi+ K-))	0.094	×	✓
No. 4	B–> mu- anti-nu_mu D0(K- K*+(K+ pi0))	0.072	✓	✓

3.13 B+-> e+ nu_e anti-D*0(pi0 anti-D0(K+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B+-> e+ nu_e anti-D0(K+ pi- pi0)	0.205	✓	✓
No. 2	B+-> e+ nu_e anti-D0(K+ K- pi0)	0.1	×	✓
No. 3	B+-> e+ nu_e anti-D*0(pi0 anti-D0(K+ pi-))	0.094	×	✓
No. 4	B+-> e+ nu_e anti-D0(K+ K*-(K- pi0))	0.072	×	✓
No. 5	B+-> e+ pi0 nu_e anti-D0(K+ K-)	0.055	✓	✓

3.14 B--> mu- anti-nu_mu D0(pi+ K- pi0)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> mu- anti-nu_mu D*0(pi0 D0(pi+ K-))	0.361	✓	✓

3.15 B+-> e+ nu_e anti-D0(K+ pi- pi0)

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B+-> e+ nu_e anti-D*0(pi0 anti-D0(K+ pi-))	0.361	✓	✓

3.16 B+-> pi+ anti-D*0(pi0 anti-D0(K+ pi- pi0))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B0-> pi+ pi0 D*-(pi- anti-D0(K+ pi- pi0))	0.159	✓	✓
No. 2	B+-> pi+ pi0 anti-D0(K+ pi- pi0)	0.145	✓	✓
No. 3	B+-> pi+ anti-D0(pi0 pi0 K*0(K+ pi-))	0.074	✓	✓
No. 4	B0-> pi+ pi0 D*-(pi0 D-(K+ pi- pi-))	0.052	✓	✓

3.17 B--> e- anti-nu_e D*0(pi0 D0(K+ K-))

	True		Found by GAs	Found by RL
	Background	Reward		
No. 1	B–> e- anti-nu_e D0(pi+ K- pi0)	0.205	✓	✓
No. 2	B–> e- anti-nu_e D0(K+ K- pi0)	0.1	✓	✓
No. 3	B–> e- anti-nu_e D*0(pi0 D0(pi+ K-))	0.094	✓	✓
No. 4	B–> e- anti-nu_e D0(K- K*+(K+ pi0))	0.072	✓	✓

4 Generalization hall of fame:

4.1 B+-> pi+ pi+ pi- anti-D0(K+ pi-)

	True		Found by RL
	Background	Reward	
No. 1	B+-> pi+ pi+ D-(K+ pi- pi-)	0.322	✓
No. 2	B+-> pi+ pi+ pi- anti-D0(K+ pi- pi0)	0.306	✓
No. 3	B+-> pi+ anti-D0(pi+ pi- K*0(K+ pi-))	0.256	✓
No. 4	B+-> pi+ anti-D0(K+ pi+ pi- pi-)	0.218	✓
No. 5	B+-> pi+ pi+ pi0 D*-(pi- anti-D0(K+ pi-))	0.139	✓
No. 6	B+-> pi+ pi+ D*-(pi- anti-D0(K+ pi-))	0.126	✓
No. 7	B0-> pi+ pi+ pi- D-(K+ pi- pi-)	0.069	✓
No. 8	B+-> pi+ pi+ pi0 D-(K+ pi- pi-)	0.069	✓

4.2 B0-> pi+ pi0 D*-(pi- anti-D0(K+ pi-))

	True		Found by RL
	Background	Reward	
No. 1	B0-> pi+ D*-(pi- anti-D0(K+ pi- pi0))	0.79	✓
No. 2	B0-> pi+ pi- anti-D0(K+ pi- pi0)	0.432	✓
No. 3	B+-> pi+ pi+ pi- anti-D0(K+ pi- pi0)	0.363	✓
No. 4	B0-> pi+ D-(pi- pi0 K*0(K+ pi-))	0.329	✓
No. 5	B0-> pi+ D*-(pi0 D-(K+ pi- pi-))	0.314	✓
No. 6	B0-> pi+ pi0 D*-(pi- anti-D0(K+ pi- pi0))	0.306	✓
No. 7	B0-> pi+ pi0 D-(K+ pi- pi-)	0.197	✓
No. 8	B+-> pi+ pi+ pi0 D*-(pi- anti-D0(K+ pi-))	0.168	✓
No. 9	B0-> pi+ pi- pi0 anti-D0(K+ pi-)	0.166	✓
No. 10	B0-> pi+ pi0 D*-(pi0 D-(K+ pi- pi-))	0.104	✓
No. 11	B0-> pi+ D-(K+ pi- pi- pi0)	0.1	✓
No. 12	B+-> pi+ pi+ pi0 D-(K+ pi- pi-)	0.083	✓
No. 13	B0-> pi+ pi- anti-D*0(pi0 anti-D0(K+ pi-))	0.067	✓
No. 14	B0-> pi+ pi- pi0 anti-D0(K+ pi- pi0)	0.062	✓
No. 15	B+-> pi+ pi+ D*-(pi- anti-D0(K+ pi- pi0))	0.057	✓

4.3 B--> mu- anti-nu_mu D*0(pi0 D0(pi+ K-))

	True		Found by RL
	Background	Reward	
No. 1	B–> mu- anti-nu_mu D0(pi+ K- pi0)	1.195	✓

4.4 B0-> e+ nu_e D-(K+ pi- pi-)

	True		Found by RL
	Background	Reward	
No. 1	B0-> e+ nu_e D*-(pi- anti-D0(K+ pi-))	0.489	✓
No. 2	B0-> e+ pi- nu_e anti-D0(K+ pi-)	0.078	✓

4.5 anti-B0-> mu- anti-nu_mu D+(pi+ pi+ K-)

	True		Found by RL
	Background	Reward	
No. 1	anti-B0-> mu- anti-nu_mu D*+(pi+ D0(pi+ K-))	0.489	✓
No. 2	anti-B0-> pi+ mu- anti-nu_mu D0(pi+ K-)	0.078	✓

4.6 B+-> mu+ nu_mu anti-D*0(pi0 anti-D0(K+ K-))

	True		Found by RL
	Background	Reward	
No. 1	B+-> mu+ nu_mu anti-D0(K+ pi- pi0)	0.205	✓
No. 2	B+-> mu+ nu_mu anti-D0(K+ K- pi0)	0.1	✓
No. 3	B+-> mu+ nu_mu anti-D*0(pi0 anti-D0(K+ pi-))	0.094	✓
No. 4	B+-> mu+ nu_mu anti-D0(K+ K*-(K- pi0))	0.072	✓
No. 5	B+-> mu+ pi0 nu_mu anti-D0(K+ K-)	0.055	✓

4.7 B0-> e+ pi- nu_e anti-D0(K+ pi-)

	True		Found by RL
	Background	Reward	
No. 1	B0-> e+ nu_e D-(K+ pi- pi-)	2.588	✓
No. 2	B0-> e+ nu_e D*-(pi- anti-D0(K+ pi-))	2.17	✓
No. 3	B0-> mu+ nu_mu D-(K+ pi- pi-)	0.116	✓
No. 4	B0-> mu+ nu_mu D*-(pi- anti-D0(K+ pi-))	0.075	✓

4.8 B0-> mu+ nu_mu D*-(pi- anti-D0(K+ K-))

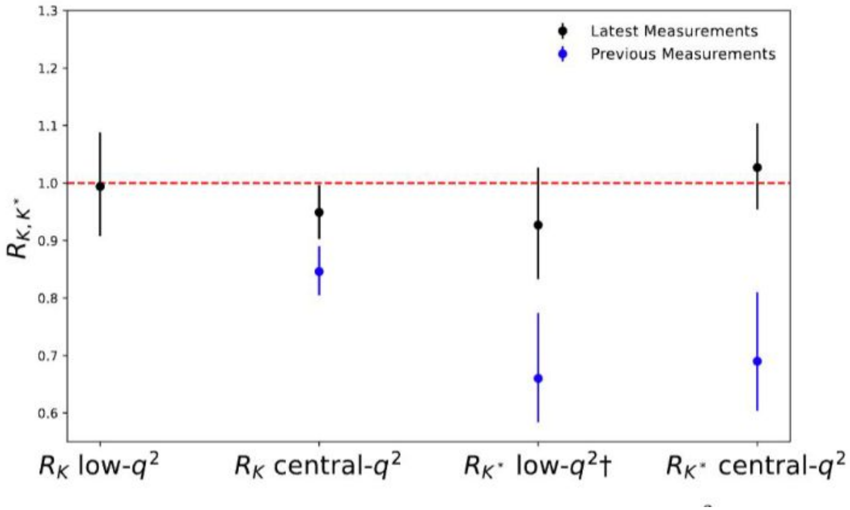
	True		Found by RL
	Background	Reward	
No. 1	B0-> mu+ nu_mu D-(K+ K- pi-)	0.386	✓
No. 2	B0-> mu+ nu_mu D-(K- K*0(K+ pi-))	0.216	✓
No. 3	B0-> mu+ nu_mu D-(K+ pi- pi-)	0.145	✓
No. 4	B0-> mu+ pi- nu_mu anti-D0(K+ K-)	0.121	✓
No. 5	B0-> mu+ nu_mu D*-(pi- anti-D0(K+ pi-))	0.094	✓

Motivation

- Performing a particle physics analysis involves:
  - simulation
  - thinking about all the backgrounds
  - checking possible mismodellings
  - …
- This can lead to **human errors**.
- Our goal is to use **Machine Learning** (ML) to:
  - automatize the workflow
  - improve the accuracy

Motivation

https://arxiv.org/pdf/2212.09153



- Lepton Flavour Universality (LFU) test initially showed discrepancies w.r.t. SM predictions
- Statistical fluctuations? Detector or physics mismodellings? New physics?

$$R_{K^{(*)}} = \frac{\mathcal{B}(B \to K^{(*)}\mu\mu)}{\mathcal{B}(B \to K^{(*)}ee)}$$

Toy model for the reward function

- We consider only signals and backgrounds originating from: B^+, B^-, B^0 and $\bar{B}^0$:

$$Reward = \frac{BR_b}{BR_s} \cdot \mathcal{M} \cdot \mathcal{K}$$

- We assume the BRs from the PDG to represent the true values
- We assume a global misidentification rate of 1%:

$$\mathcal{M} = 0.01^{N_{misID}}$$

Number of misidentified particles

- For the kinematic overlap factor, we don't consider the momentum distribution of the missing particles, and we assign a multiplicative penalty for each unit of the absolute difference between the number of non-reconstructed particles in signal and background:

$$\mathcal{K} = 0.1^{|N_{miss} - N_\nu^s|}$$

Number of neutrinos in signal

Number of non-reconstructed particles in background

Tokenization

- Tokens for representing particles
- Index tokens: $\{1, 2, \ldots, N_S\} \rightarrow N_S$ tokens (N_S = number of final state particles in signal)
- Token for)
- Token for "END"
- Token for "LOST"
- Token for separating signal and background (not an action)
- Token to represent elements with a not assigned token (not an action)

"LOST" and "index tokens" are related to the next token in the sequence and allow to describe misID and Part. Reco.

Example:

K^- and π^- are misidentified

- Signal: $B^0 \to K^+ \pi^- \pi^0$
- Background: $B^+ \to K^{*+}($ 1 K^+ 3 π^0) LOST π^+ 2 K^- END

π^+ is not detected

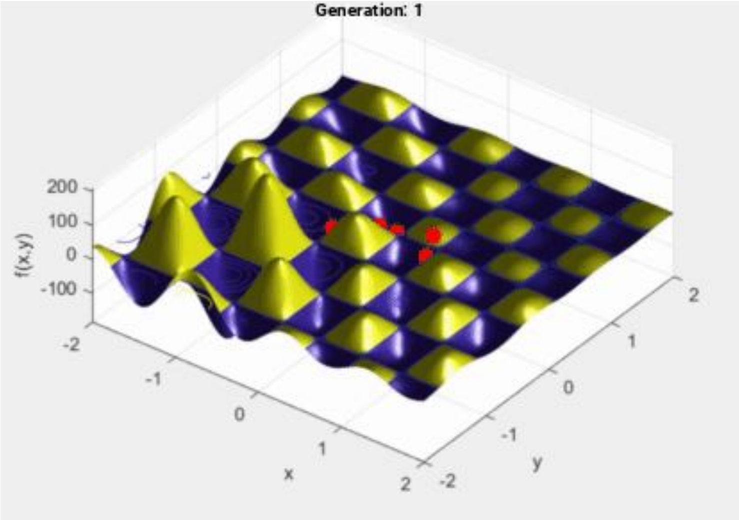
Interesting points:

- We can deal with **intermediate resonances**, **misidentifications** and **partially reconstructed backgrounds** with a few tokens.
- Tokens (and thus actions in RL) are O(N). For example, actions like $pi^+ \leftrightarrow K^+$ would be $O(N^2)$.
- We can exploit a lot the **masking** of actions/tokens. For example, if last predicted token was "LOST", then we can mask all other actions that are not final state particles

Genetic Algorithms (GAs)

Example:

$$f(x,y) = \left(\sum_{n=1}^{5} n \ \cos[n + x(n+1)]\right)\left(\sum_{n=1}^{5} n \ \cos[n + y(n+1)]\right)$$



https://drive.google.com/file/d/1OWyJNH0tnq4um7fEkUvOwZIG2oyjckRg/view?usp=sharing

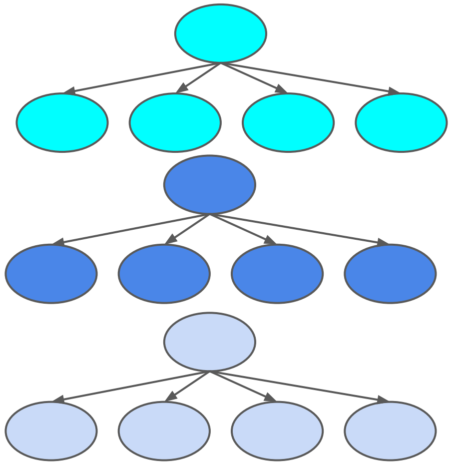
Resonance creation

- Some genes are randomly selected to create an intermediate resonance
- The charge of the intermediate particle is computed from the selected genes, and an intermediate particle of this charge is randomly suggested
- Example:

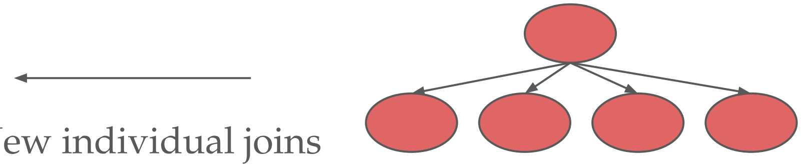
Random immigration

- Introduce into the population random individuals with a certain probability
  - The purpose is to be able to explore more, and to not depend that much on the initialization of the algorithm
  - Arrivals are only considered in case an individual did not suffer crossover, mutations or a resonance creation ⇒ We are not increasing the population size

Current population genes:

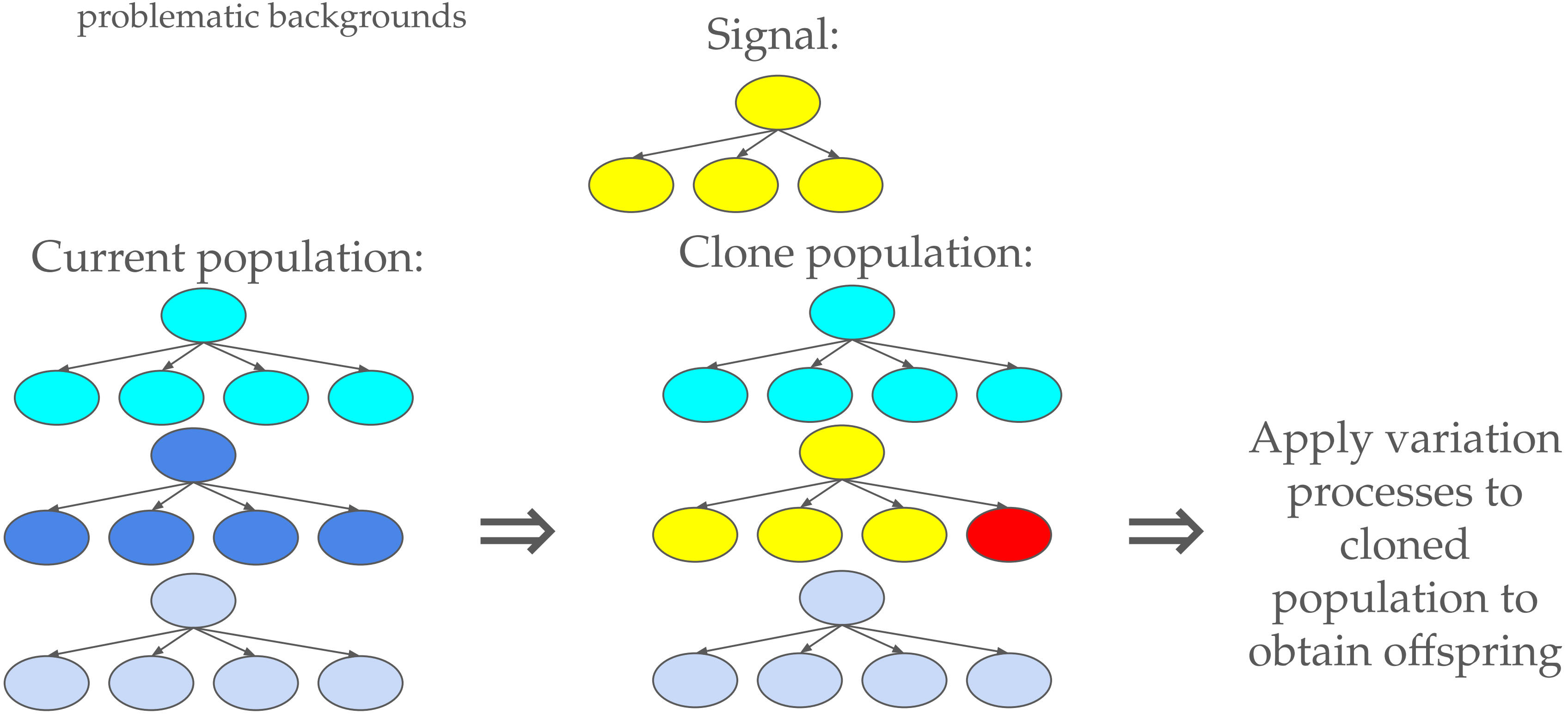


New individual joins



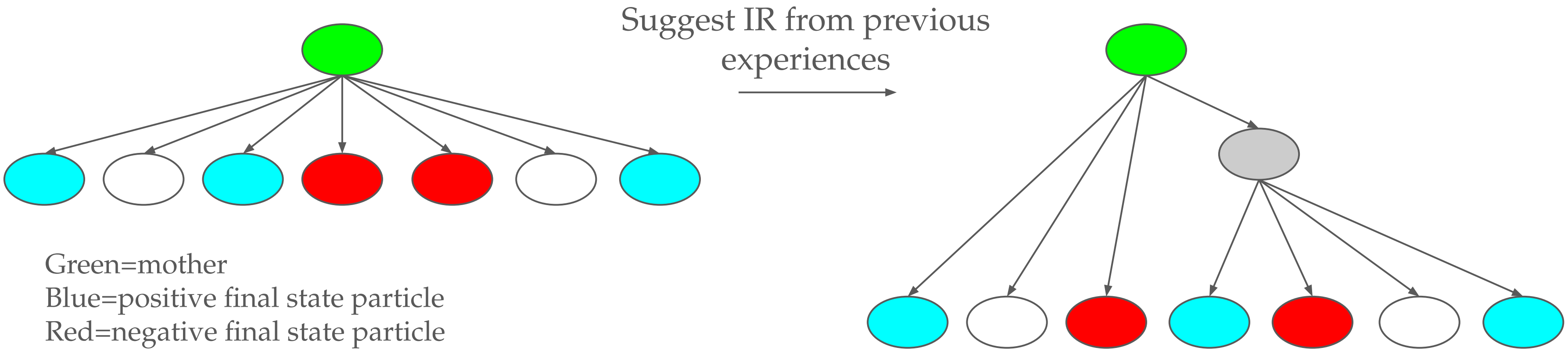
Inheriting from the signal

- With a certain probability the genes of the signal will be cloned instead of cloning an individual from the current population
- This allows to explore more the region of the space near the signal, which is more likely to have problematic backgrounds



Previous experience learning

- When an individual has a non zero fitness value, and has an intermediate resonance in its genes that was never seen before, the GA “learns” this is a valid intermediate resonance for future optimizations.
- With a certain probability, the genes of the individual are analyzed, and from all intermediate resonances that are possible for these genes one is randomly suggested

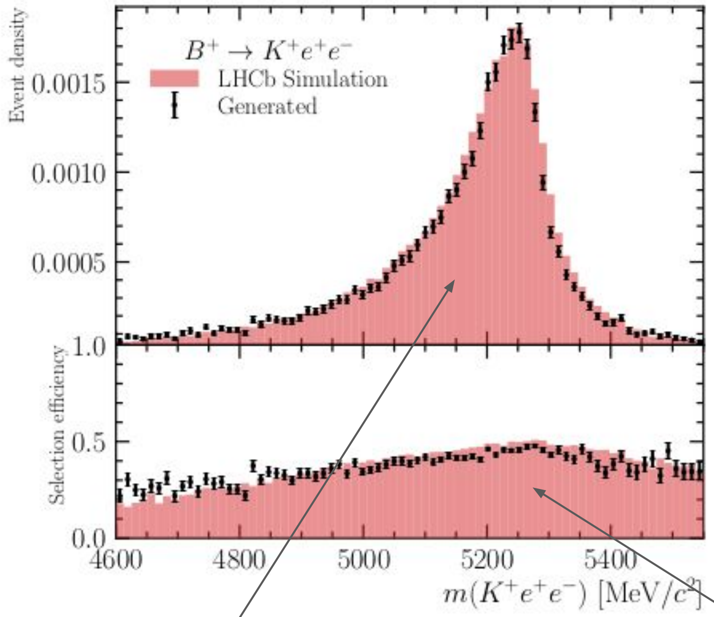


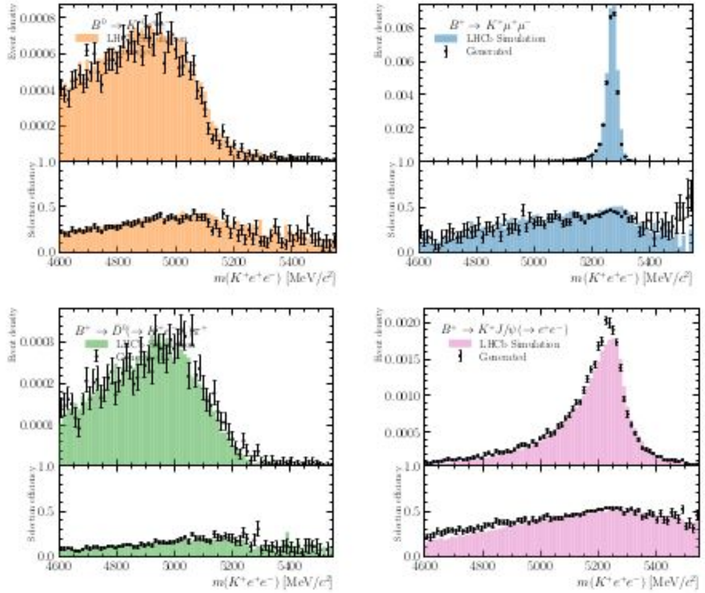
Green=mother
Blue=positive final state particle
Red=negative final state particle
White=neutral final state particle
Gray=intermediate particle

Work published in: [6]

Signal

Backgrounds





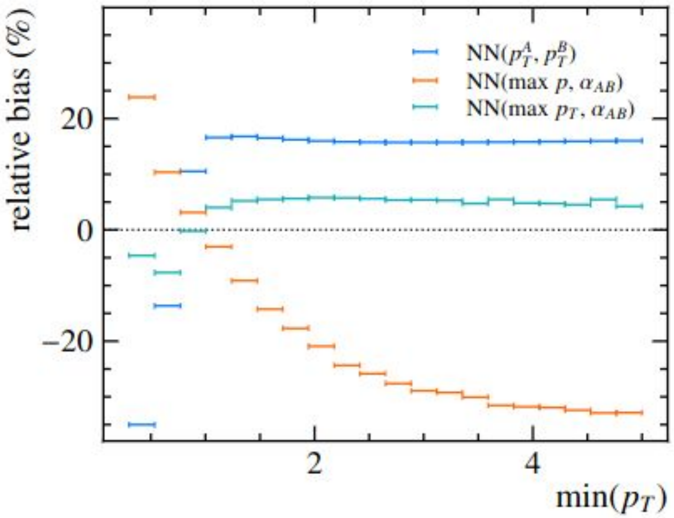
Generated mass distribution

Selection efficiency

Previous works: efficiency mismodelling

Previous works on the Devil's Advocate project: [1]

- Proposal of a new method based on **machine learning** to play the devil's advocate and investigate the **impact of** detector or physics **mismodellings** in a quantitative way
  - Focused on the signal efficiency



Efficiency mismodelling introduced up to a 30% relative bias (on differential Branching Ratio (BR))