

Hierarchical clustering in particle physics through reinforcement learning

Johann Brehmer¹, Sebastian Macaluso¹, Duccio Pappadopulo², and Kyle Cranmer¹

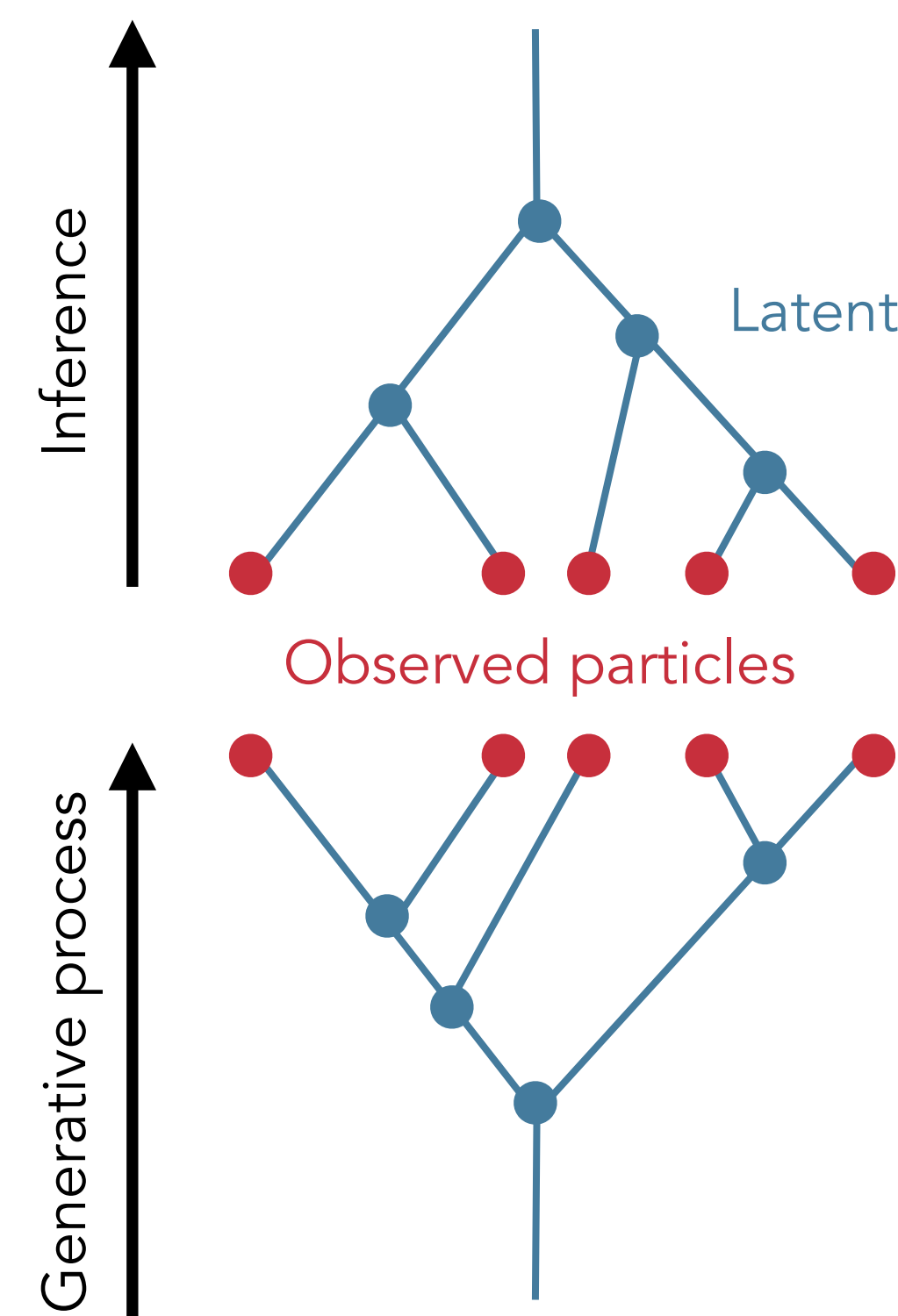
¹ NYU ² Bloomberg (work done before joining Bloomberg)

arXiv:2011.08191



Hierarchical clusterings in particle physics

- Jets – sprays of hadronic particles – are the most common object at the Large Hadron Collider experiments
- They are produced when a quark or gluon repeatedly radiates more quarks and gluons. These repeated binary splittings result in a tree structure. We have reliable simulators for this generative process
- A key problem for many LHC analyses is to invert this generative process, a hierarchical clustering problem: given observed particles (the leaves of the binary tree), the goal is to infer the tree structure that maximizes the likelihood
- Unfortunately, the number of clusterings grows as $(2N - 3)!!$ with the number of leaves N : we can't search exhaustively
- Particle physicists usually resort to greedy algorithms based on heuristics



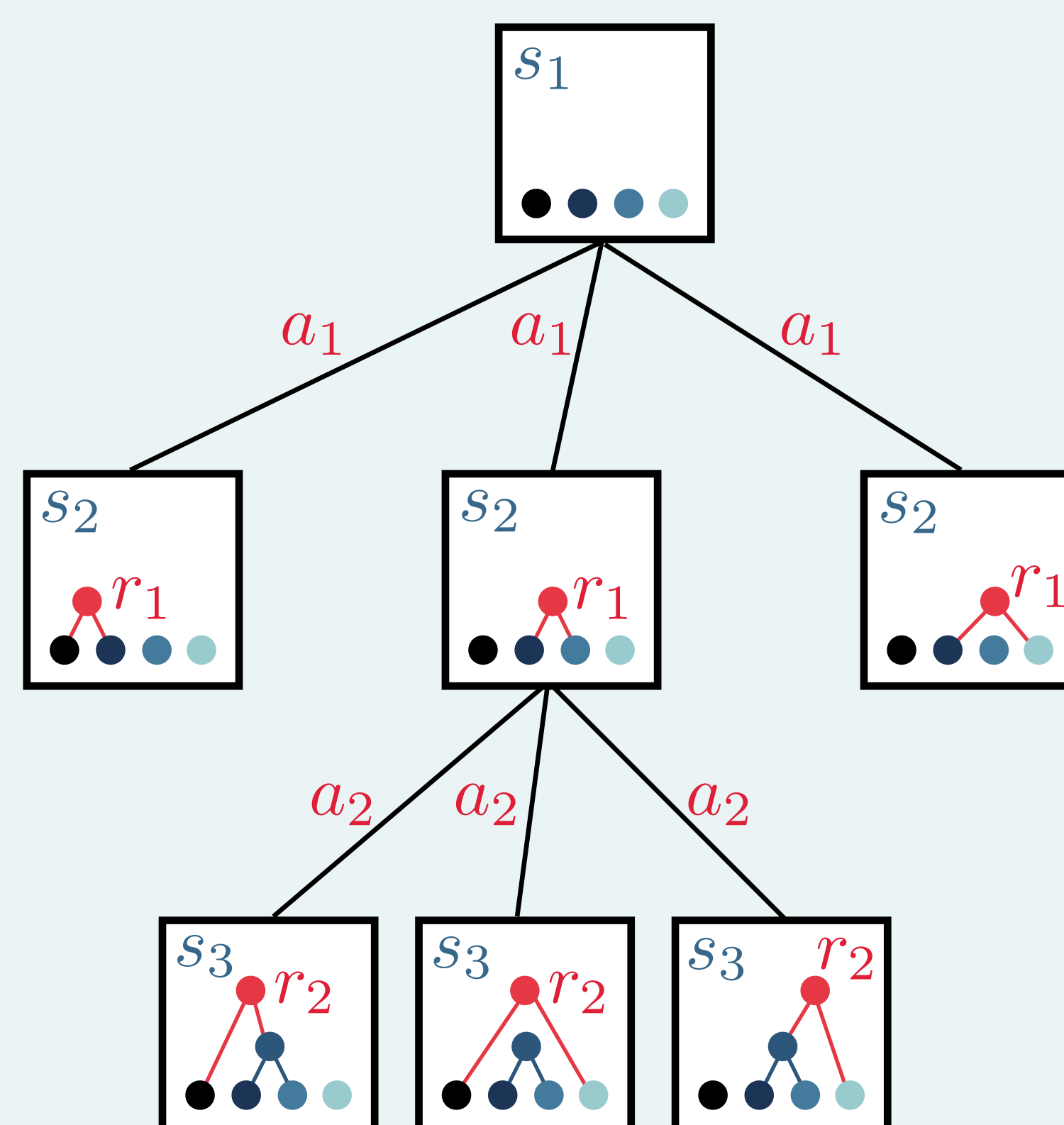
Algorithms

- Any off-the-shelf RL agent can now solve the hierarchical clustering problem in particle physics
- We adapt **Monte-Carlo Tree Search** (MCTS) guided by a neural policy (similar to AlphaZero), which explores the search tree of possible clusterings by choosing actions that maximize

$$\text{PUCT}_{s,a} = Q_{s,a} + c \pi(s,a) \frac{\sqrt{N_s}}{1 + N_{s,a}}$$

Annotations: $Q_{s,a}$ is mean reward in past roll-outs; $\pi(s,a)$ is neural policy; $N_{s,a}$ is visit count.
- We also explore **Behavioral Cloning** (BC), training a policy to mimic the ground-truth trees in simulation data
- As baselines, we use a greedy algorithm, beam search, and a random policy. For small number of leaves we also compare to the exact maximum likelihood tree, which we compute with the algorithm proposed in [Greenberg et al, 2002.11661](#)

Particle clusterings as a Markov Decision Process



- Each **state** is a set of particles: the initial state is the set of leaf nodes, other states represent partial clusterings
- An **action** selects two particles to be merged next
- The **reward** is the log likelihood of the corresponding binary splitting
- The **state transition** is deterministic and clusters the two selected particles, following energy and momentum conservation
- The MDP is **episodic** and terminates when a single particle is left

Experiments

- We test these algorithms on data from [Ginkgo](#), a toy generative model with a tractable likelihood
- MCTS produces higher-quality clusterings than the baselines, though it requires more resources

