MACHINE LEARNING
Science and Technology

PURPOSE-LED PUBLISHING™

Part of the MACHINE LEARNING Series

PAPER • OPEN ACCESS

Constrained collaborative optimization of charged particle tracking with multi-agent reinforcement learning

To cite this article: Tobias Kortus *et al* 2026 *Mach. Learn.: Sci. Technol.* **7** 015021

MACHINE LEARNING Science and Technology

OPEN ACCESS

RECEIVED
27 October 2025

REVISED
12 December 2025

ACCEPTED FOR PUBLICATION
7 January 2026

PUBLISHED
29 January 2026

PAPER

Constrained collaborative optimization of charged particle tracking with multi-agent reinforcement learning

Tobias Kortus[1,*], Ralf Keidel[1], Nicolas R Gauger[1] and Jan Kieseler[2], on behalf of the Bergen pCT Collaboration

[1] Chair for Scientific Computing, RPTU University Kaiserslautern-Landau, Kaiserslautern, Germany
[2] Institute of Experimental Particle Physics (ETP), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
[*] Author to whom any correspondence should be addressed.

E-mail: tobias.kortus@rptu.de

Keywords: multi-agent reinforcement learning, combinatorial optimization, safety layer, charged particle tracking, end-to-end optimization, high-energy physics

Abstract

Reinforcement learning (RL) demonstrated immense success in modeling complex physics-driven systems, providing end-to-end trainable solutions by interacting with a simulated or real environment, maximizing a scalar reward signal. In this work, we propose, building upon previous work, an end-to-end multi-agent RL approach with assignment constraints for reconstructing particle tracks in pixelated particle detectors. Our approach optimizes collaboratively a parameterized policy, functioning as a heuristic to a multidimensional assignment problem, by jointly minimizing the total amount of particle scattering over the reconstructed tracks in a readout frame. To satisfy constraints, guaranteeing a unique assignment of particle hits, we propose a safety layer solving a linear assignment problem for every joint action. Further, to enforce cost margins, increasing the distance of the local policies predictions to the decision boundaries of the optimizer mappings, we recommend the use of an additional component in the blackbox gradient estimation, forcing the policy to solutions with lower total assignment costs. We empirically show on simulated data, generated for a particle detector developed for proton imaging, the effectiveness of our approach, compared to multiple single- and multi-agent baselines. We further demonstrate the effectiveness of constraints with cost margins for both optimization and generalization, introduced by wider regions with high reconstruction performance as well as reduced predictive instabilities. Our results form the basis for further developments in RL-based tracking, offering both enhanced performance with constrained policies and greater flexibility in optimizing tracking algorithms through the option for individual and team rewards.

1. Introduction

Reinforcement learning (RL) and multi-agent RL (MARL) are promising paradigms for constructing and optimizing autonomous agents that can compete in a wide variety of complex sequential decision problems, such as games (Mnih *et al* 2013, Silver *et al* 2018), robotics (Gu *et al* 2017, Andrychowicz *et al* 2020) or autonomous driving (Kendall *et al* 2019) by discovering complex interaction mechanisms in the underlying environment. Coupled with the tremendous success in the aforementioned fields, RL has recently demonstrated great potential in optimizing and controlling physics processes (Kain *et al* 2020, Degrave *et al* 2022, Vage 2022, Kortus *et al* 2023), by maximizing a scalar reward signal using trial and error (Littman 1994, Sutton and Barto 2018). Especially for combinatorial problems, RL has been shown to learn generalizable policies that can even outperform supervised learning approaches, despite the lack of ground truth information (Joshi *et al* 2021). Kortus *et al* (2023) and Vage (2022) have shown for charged particle tracking used in high-energy physics reconstruction the potential of deep RL for optimizing over otherwise non-differentiable discrete assignment operations. The presented approaches aim to construct discrete sets of particle tracks over subsequent layers under the influence of particle interaction mechanisms. Modeling each track independently, as proposed in previous work, however, reduces the total system's observability as the interactions of neighboring tracks are neglected. Further, the simpler problem formulation of independent interacting tracks fails to constrain the solution set to globally feasible solutions with exclusive hit assignments. Extending previous work, we further investigate the concept of MARL-based charged particle tracking as a combinatorial optimization problem to tackle the aforementioned limitations of single-agent systems. We therefore propose a collaborative MARL approach with assignment constraints, iteratively optimizing a joint policy of multiple track followers. We represent the stepwise agent constraints as a centralized safety layer, ensuring unique hit assignment across all agents, both during training and inference. The assignment constraints are realized by solving for every reconstruction step a linear sum assignment problem (LSAP) projecting the unsafe local agent policies to a global safe policy. Our main contributions and findings in this paper summarize as follows:

- Building upon previous work in Kortus *et al* (2023), we propose multiple multi-agent extensions of RL-based particle tracking. To maximize both information availability during training and restrict global information during execution, we utilize decentralized agents with optional safety layer, satisfying assignment constraints, trained in a centralized manner using centralized critic architectures.
- We extend the blackbox (BB) differentiation technique by Vlastelica *et al* (2020) by adding an extra simple gradient component, effectively increasing the cost margins between predictions and decision boundaries, which leads to significantly better training and generalization capabilities. We provide a supplementary ablation study, highlighting the robustness of the biased gradient estimator to the exact choice of cost margin gradient weighting.
- We demonstrate excellent empirical performance of our method compared to a conventional sequential track follower (Pettersen *et al* 2020) as well as single-agent (Kortus *et al* 2023) and multi-agent baselines and highlight the advantage of constrained policy optimization for high particle densities.
- Finally, we validate the benefit of the designed multi-agent architecture and the necessity of the adapted gradient through the safety layer by examining reconstruction performance, reward surfaces (Sullivan *et al* 2022), prediction instabilities (Fard *et al* 2016), and policy entropy.

2. Theory and background

Throughout this work, we focus on particle data generated by the digital tracking calorimeter (DTC) prototype developed by the *Bergen pCT Collaboration* (Alme *et al* 2020, Aehle *et al* 2023) for proton computed tomography. While the general methodology of the proposed MARL-based tracking is in principle extendable to particle detectors of arbitrary geometry (subject to detector-specific adaptations in, e.g. reward design), we leave a more general application for different particle detectors for further work. In the following section, we describe both the detector and the basic particle interaction mechanisms expected at relevant particle energies of $\mathcal{O}(230\text{ MeV})$.

Bergen pCT detector prototype: The Bergen pCT DTC is a multi-layer high-granular pixelated tracking calorimeter, consisting in total of 43 sensitive layers with two tracking layers followed by 41 detector-absorber sandwich calorimeter layers. The high granularity of the detector prototype enables the simultaneous tracking of multiple particles for improved time efficiency. Each sensitive layer, spanning an area of 27× 16.6 mm, is composed using multiple strips of ALPIDE pixel sensors (Mager 2016, Aglieri Rinella 2017) with additional 3.5 mm aluminum absorbers in each calorimeter layer, functioning both as absorber and carrier. To reduce scattering of particles, allowing for accurate directional measurements of particles entering the detector, both tracking layers are separated by 57.8 mm air gaps, while the carrier material is significantly reduced. Further details and a fine-grained decomposition of the detector material are described in Alme *et al* (2020). While the proposed MARL formulation itself is independent of the exact detector composition, the unique material budgets of the tracking and calorimeter layers lead to distinct particle interaction behaviors. Since the learned policy must account for these layer-specific interactions to accurately estimate particle direction and energy, these differences introduce additional complexity to the learning task.

Particle interactions and tracking Accelerated charged particles undergo numerous complex interactions with the traversed matter (Groom and Klein 2000), each contributing uniquely to changes in their trajectories. In proton imaging, charged particles are mainly influenced by electromagnetic Coulomb interactions with atomic electrons and nuclei (Groom and Klein 2000, Gottschalk 2018).

As charged particles pass through matter, they lose small amounts of energy through interactions with atomic electrons, resulting in the particle incrementally slowing down until it eventually stops around its projected range. The process itself is stochastic but captured in terms of mean energy loss or linear stopping power as a function of particle energy (Bethe 1932). The stopping power increases sharply near the end of the particle's path, forming a distinct Bragg peak, providing its beneficial characteristic for proton therapy. Despite the electromagnetic force acting on the charged particle, its path remains unchanged due to the relatively low mass of the atomic electron.

When interacting with atomic nuclei, charged particles are randomly deflected from their straight path. Integrated over thin slabs of material (e.g. the aluminum absorbers or separating air gaps), the particle deflection angles are characterized by an approximately Gaussian shape (Highland 1975, Groom and Klein 2000). Multiple scattering constitutes the primary driver of complexity in reconstructing the original particle trajectory in a readout frame of multiple particles, as it causes the path to deviate unpredictably from a straight line.

Additionally, on some occasions, particles undergo complex inelastic interactions with the atomic nucleus in a destructive process where the original primary particle is absorbed, and new particles originate from this process. Due to its highly stochastic nature, secondary tracks originating from the new particles cause additional complexities during reconstruction and are unusable for imaging.

To recover usable characteristic properties of the particles, tracking algorithms aim to model or learn the pattern of the particle in the detector readouts under the influence of the inherent interaction mechanisms, aiming to reconstruct complete and coherent particle trajectories.

3. Related work

Particle tracking: While early particle tracking algorithms heavily relied on conventional algorithms such as iterative (Frühwirth 1987, Pettersen *et al* 2020), evolutionary (Mankel 1997) or combinatorial (Pusztaszeri *et al* 1996) approaches, modern tracking solutions heavily utilize machine learning to tackle the increasing combinatorial explosion due to increasing particle counts. Especially geometric deep learning, operating either on node (Kieseler 2020, Lieret *et al* 2023) or edge level (DeZoort *et al* 2021, Kortus *et al* 2025) of graph representations, demonstrated to be highly effective, while maximizing computational efficiency. Aiming to combine advantages from conventional tracking and deep learning, recent work on RL-based tracking demonstrated, both on discrete- (Kortus *et al* 2023) and continuous action spaces (Vage 2022), the ability to learn reconstruction policies in an end-to-end fashion by interacting with an environment. Our work extends the mechanisms in Kortus *et al* (2023) to a multi-agent setting, reducing partial observability and enforcing assignment constraints on the learned policy during both training and inference.

Safe/constrained RL: Learning safe policies, operating under safety or functional constraints, is an emerging research field, both in single- and multi-agent RL. For this work, we focus on state-wise safety by constraining the set of feasible policies. Our work is closely related to the idea of safety layers and shielding. Pham *et al* (2018) proposed OptLayer for robotic control, embedding robot constraints as a quadratic program that is solved end-to-end differentiable with an interior point method. Dalal *et al* (2018), Sheebaelhamd *et al* (2021), proposed the usage of an implicit safety layer that performs action correction of the policy using a linearized version of the constraint function. Similarly, Alshiekh *et al* (2017), ElSayed-Aly *et al* (2021)proposed the usage of safety editors, restricting the agent to safe actions by either reducing the safe action space or correcting unsafe actions of the policy. Vinod *et al* (2022, 2024) decouples agent constraints and optimization by training individual agents and restricting the use of the constraint layer to inference.

4. Methodology

In the following, we outline a general notion of constrained and unconstrained collaborative charged particle tracking, extending existing work described in Kortus *et al* (2023), and propose multiple agent architectures for the centralized training for decentralized execution (CTDE) paradigm (Oliehoek *et al* 2008). We specifically chose this unique scheme to

- Restrict the usage of global information and costly communication protocols in the policy parameterization, keeping decentralized policies more tractable compared to a centralized approach, where the joint state and action spaces grow exponentially with the number of agents (Oliehoek and Amato

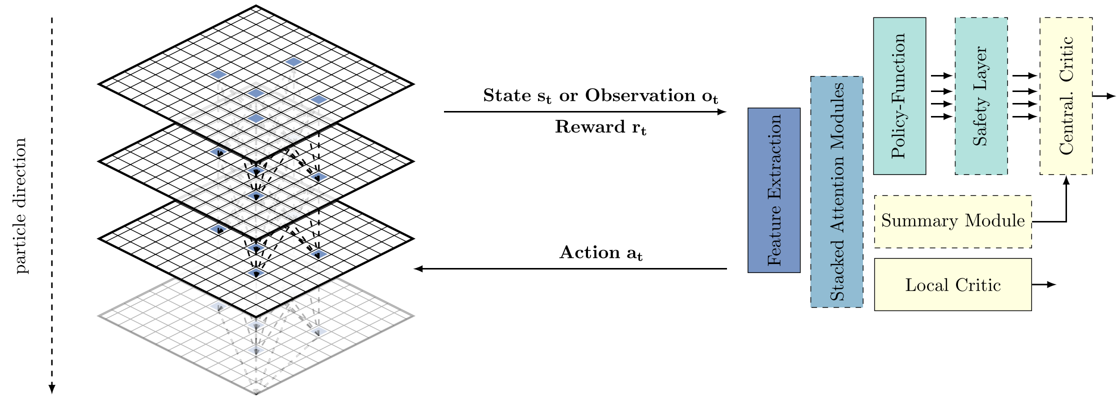

Figure 1. General description of charged particle tracking framework for single- or multi-agent reinforcement learning. The agent (right) learns by iterated interaction with the environment, represented as a directed acyclic graph (left), reconstruction policies that maximize the obtained rewards. Agent components marked with dashed lines are optional and are only used for some agent configurations.

2016, Gronauer and Diepold 2022). This further contributes to minimizing negative adverse effects on the inference performance compared to the single-agent baseline[3].

- Yet, we aim to propagate and exploit globally available information shared between all agents during training in a centralized critic. By doing so, we ensure and foster collaborative behavior between agents, contributing to reducing ambiguities at higher particle densities.

Finally, we describe different training schemes for both unconstrained and constrained MARL with CTDE, highlighting the task-specific modifications required to tackle the specific challenges introduced by both multi-agent and constrained formulation of the reconstruction task. The extended RL-based tracking framework together with the integration of new components is outlined in figure 1.

4.1. Problem statement

We formulate sequential multi-agent particle tracking over multiple layers of discrete particle readout data as a *decentralized partially observable Markov decision process* (Dec-POMDP) (Bernstein *et al* 2009) on a directed acyclic graph structure. The Dec-POMDP framework is an extension of the standard partially observable Markov decision process (POMDP) (Kaelbling *et al* 1998) to a multi-agent setting, enabling a principled way of dealing with the effect of uncertainties with respect to other agents (Oliehoek and Amato 2016).

Let $\mathcal{S}$ be a set of global (unobservable) environment states describing all partial track candidates in a readout frame up to a specific layer. Given any set of partial track candidates, the aim is to iteratively extend them by a set of connections to hits in the subsequent layer (in the following also referred to as track segments), such that a shared reward signal is optimized. However, instead of perceiving the global environment state, each agent $i \in N$ can only draw individual local observations $\boldsymbol{o}_t^{(i)} \sim \mathcal{O}_i$ from the joint observation space $\mathcal{O} = \{\mathcal{O}_i\}_{i=1}^N$. To maintain compatibility with the single-agent framework while minimizing global and shared information, local agent observations follow the state definition of the single-agent MDP in Kortus *et al* (2023). Each observation is composed of the last reconstructed track segment and all possible next track segments of the track candidate associated with agent i according to

$$\boldsymbol{o}_{t,\text{Dec-POMDP}}^{(i)} = s_{t,\text{MDP}} = \left\{v_t^{(i)}, e_{t-1,t}^{(i)}\right\} \cup \bigcup_{j \in \mathcal{N}(v_t)} \left\{v_{t+1,j}^{(i)}, e_{t,t+1,j}^{(i)}\right\}. \tag{1}$$

Here $\{v_t^{(i)}, e_{t-1,t}^{(i)}\}$ defines the vertex and edge of the last reconstructed track segment, defined by a connection between two subsequent layers, and $\bigcup_{j\in\mathcal{N}(v_t)}\{v_{t+1,j}^{(i)}, e_{t,t+1,j}^{(i)}\}$ is the set of next candidates for extending the current track candidate.

Each agent, parameterized by either a deterministic $\mu_\theta^{(i)}$ or stochastic policy $\pi_\theta^{(i)}$, can select, based on its perceived observation, from a set of local actions $a_t^{(i)} \in \mathcal{A}_i$ defined by the set of possible next segments (see figure 2). To enforce unique hit assignment, the factored joint action space $\mathcal{A} = \{\mathcal{A}_i\}_{i=1}^N$ over

[3] To enforce a richer representation of the environment state, communication schemes (Foerster *et al* 2016, Jiang and Lu 2018) can be optionally integrated at the expense of increased cost of evaluating the agent's policies.

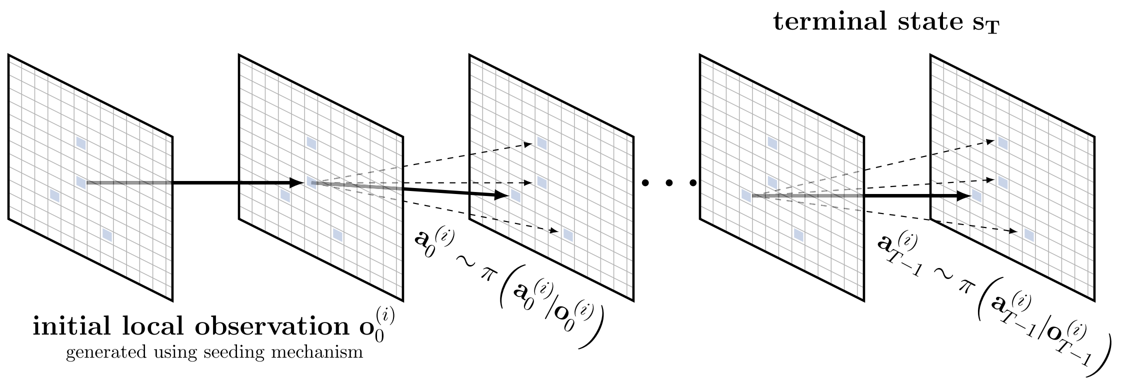
terminal state $\mathbf{s_T}$
$\mathbf{a}_0^{(i)} \sim \pi\left(\mathbf{a}_0^{(i)}/\mathbf{o}_0^{(i)}\right)$
$\mathbf{a}_{T-1}^{(i)} \sim \pi\left(\mathbf{a}_{T-1}^{(i)}/\mathbf{o}_{T-1}^{(i)}\right)$
initial local observation $\mathbf{o}_0^{(i)}$
generated using seeding mechanism

Figure 2. Schematic representation of the sequential particle tracking procedure for a single agent (i) with a stochastic policy $\pi^{(i)}$ over the set of detector layers. The transition dynamic between layers is described as edges in a directed acyclic graph (hit graph).

all agents is optionally constrained to feasible actions. For each interaction, all agents receive a single shared scalar reward signal $r_t \in \mathcal{R}$, accumulated until a terminal state s_T. This terminal state is induced by the absence of further valid actions, signifying a complete reconstruction of the readout frame.

Graph construction: Following the parameterization of particle readouts described in Kortus *et al* (2023), we model the particle data as a directed acyclic graph (*hit graph*), where each hit represents a vertex in the graph. Edges are generated between hits of adjacent layers, opposite to the direction of the particle, directing a reconstruction from less occupied area in the detector to the first tracking layer (see figure 2). Both vertices and edges are parameterized by a set of features $\boldsymbol{v}_i = [\Delta E, x, y, \mathbb{1}_z]$ and $\boldsymbol{e}_{ij} = [r_{ij}, \theta_{ij}, \phi_{ij}]$, defining the energy deposition and position of the hit with a one-hot encoded layer index as well as the spherical coordinates of the edge connections. Finally, we employ the feature normalization scheme described in Kortus *et al* (2023) with beam spot centering, compensating for the beam position in the detector, providing translation invariant features.

Sampling of track candidates: Track candidates are constructed for any given hit graph, starting from all initial unoccupied graph vertices in the last detector layer, by iteratively adding new vertices in subsequent layers until a terminal state in the first detector layer is reached (see figure 2). Unassigned vertices in subsequent layers are incrementally added to the list of track candidates to ensure a full reconstruction of the readout frame[4]. To provide a starting track segment, functioning as an initial local observation, we rely in this work on ground-truth seeding (Kortus *et al* 2023), avoiding unwanted dependencies of seeding algorithms on the performance of the proposed algorithms and providing an agnostic performance upper bound for MARL-based tracking. This mechanism is selected specifically to avoid complex interactions of both mechanisms during evaluation. When moving beyond this controlled experimental setting, ground-truth seeding has to be replaced with a suitable (ground-truth-free) seeding mechanism, either from literature or specifically designed for MARL-based tracking. For independently reconstructed particle tracks, an imperfect seeding with error rate ϵ_s reduces the upper-bounded success probability by a factor $(1 - \epsilon_s)$. Under strictly enforced assignment constraints as proposed, additional cascading errors can further influence performance. Therefore, both error rates and the interaction with the tracking algorithm are paramount to ensure efficient operations. This matter and possible implementations are further discussed in section 6. To further quantify the effects of imperfect sampling as a function of the seeding error rate ϵ_S, we perform a supplementary ablation study (appendix C) by analyzing a synthetically corrupted seeding mechanism that enables a fine-tunable control over seeding quality. A detailed interpretation of the results is performed in the relevant sections.

Objective: We attempt to find, by repeatedly interacting in the described environment, a joint policy that collaboratively maximizes the gathered expected discounted return of track candidates under a shared team reward. Similar to Kortus *et al* (2023), we aim to optimize the reconstruction policy by minimizing the average amount of particle scattering in a readout frame over all agents $i \in N$. We thus define the shared reward signal as the negative average scatter angle obtained for each segment transition defined

[4] Adding vertices as additional track candidates dynamically increases the number of agents in the next reconstruction step.

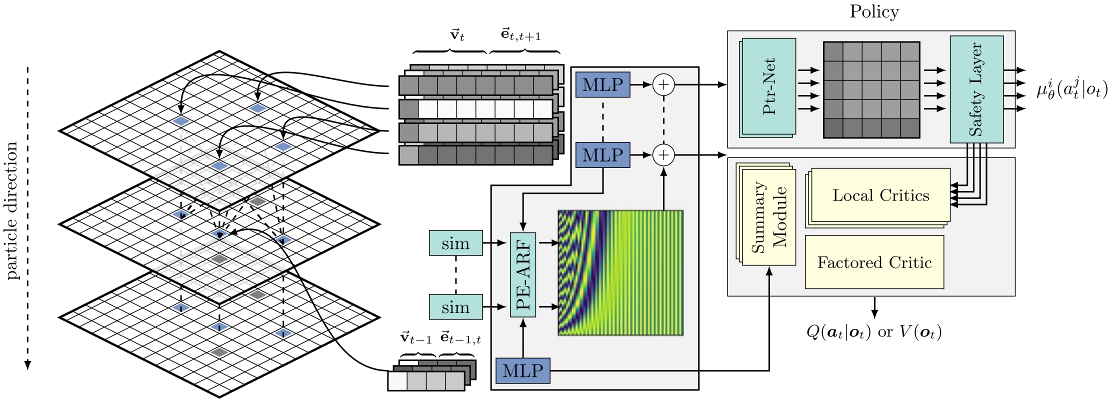

Figure 3. Interaction loop between environment description containing particle readouts in the form of a directed acyclic graph based on. The agent (network architecture on the right) observes a state, describing the current particle trajectory, and chooses a next particle hit in the subsequent layer. The reward is defined based on the physical likelihood of the undertaken transition. Reproduced from Kortus *et al* (2023). CC BY 4.0.

by a hit triplet (v_{t-1}, v_t, v_{t+1}) over three subsequent layers as

$$\boldsymbol{r}_t = -\frac{1}{N}\sum_{i=1}^{N} \arccos\left(\frac{\boldsymbol{p}_{t-1:t}^{(i)} \cdot \boldsymbol{p}_{t:t+1}^{(i)}}{\|\boldsymbol{p}_{t-1:t}^{(i)}\| \|\boldsymbol{p}_{t:t+1}^{(i)}\|}\right), \tag{2}$$

Here $\boldsymbol{p}_{t-1:t} = (\boldsymbol{p}_t - \boldsymbol{p}_{t-1})$ and $\boldsymbol{p}_{t:t+1} = (\boldsymbol{p}_{t+1} - \boldsymbol{p}_t)$ denote the path segments for previously reconstructed and current action selection. In the multi-agent case, we rely on this naive description over the more detailed modeling of the energy-dependent scattering behavior (Highland 1975), described in Kortus *et al* (2023), to remove the dependence of the reward signal on full track candidates, making it more suitable for both on- and off-policy algorithms. Despite its practical effectiveness, the simplified reward lacks the physics-informed weighting of scattering events, limiting its fidelity to realistic track behavior.

4.2. Architecture and implementation

In this section, we describe extensions to the existing attention-based agent parameterization (Kortus *et al* 2023) for multi-agent RL, providing both a permutation-invariant and action-size-independent processing for collaborative charged particle tracking (see figure 3). Our main focus lies on the design of centralized critic components that can be seamlessly integrated into the existing framework for particle tracking (Kortus *et al* 2023). To improve over the existing architecture, we simplify the policy by moving computationally intensive layers from the policy to the centralized critic, limiting the availability of this information to the training phase while reducing inference overhead. Finally, we propose the use of a differentiable safety layer inspired by Dalal *et al* (2018), Sheebaelhamd *et al* (2021) for constrained particle tracking, guaranteeing unique assignments of particle hits. We further provide useful gradient information, building upon existing work in decision-focused learning by Vlastelica *et al* (2020), Sahoo *et al* (2023).

Feature preparation: Following the description of local observations in section 4.1, we extract edge- and node-level features for both last reconstructed $(v_{t-1} \rightarrow v_t)$ and possible next track segments $(v_t \rightarrow v_{t+1,j})$ from the hit graph. Both are projected by separate multi-layer perceptrons (Ψ_1 and Ψ_2) into an equally sized higher dimensional embedding space ($\boldsymbol{h}^{\text{emb}}$) according to

$$\begin{aligned} \boldsymbol{h}_{\text{obs}}^{\text{emb},(i)} &= \Psi_1\left([\boldsymbol{v}_t, \boldsymbol{e}_{t-1,t}]\right) \quad \text{and} \\ \boldsymbol{h}_{\text{act},j}^{\text{emb},(i)} &= \Psi_2\left(\left[\boldsymbol{v}_{t+1}, \boldsymbol{e}_{t,t+1,j}\right]\right) + \text{PE-ARF}\left(s_{\cos}\left(e_{t-1,t}, e_{t,t+1,j}\right)\right) \end{aligned}. \tag{3}$$

For performance reasons, we omit the additional feature vector generated by a graph neural network as proposed in Kortus *et al* (2023), as we found the simple feature description to be sufficient in combination with the use of a safety layer. The *positional encoding with adaptive receptive field* (PE-ARF) mechanism, proposed in Kortus *et al* (2023), is used to provide relative information of segment candidates. PE-ARF utilizes cosine similarity information to create sinusoidal embedding vectors similar to Vaswani *et al* (2017), providing a strong inductive bias for tracking. This mechanism is augmented by an adaptive rescaling mechanism, dynamically reallocating the resolution of the positional encoding based on the current graph topology.

Local agent policies: We parameterize a local deterministic policy $\mu_\theta^{(i)}$ (or equivalently $\pi_\theta^{(i)}$ for stochastic policies) of each agent using a pointer mechanism (Vinyals *et al* 2015) (Ptr-Net), with parameter sharing between agents. The policy is designed to predict the conditional probability of the local action $a_j^{(i)}$ conditioned on *observation-* and *action features*. Therefore, the policy utilizes additive attention (Bahdanau *et al* 2015) according to

$$\alpha_j^{(i)} = \boldsymbol{v}^T \tanh\left(\boldsymbol{W}_1 \boldsymbol{h}_{\text{act},j}^{\text{emb},(i)} + \boldsymbol{W}_2 \boldsymbol{h}_{\text{obs}}^{\text{emb},(i)}\right), \tag{4}$$

where $\boldsymbol{W}_1$, $\boldsymbol{W}_2$ and $\boldsymbol{v}$ are learnable parameter matrices/vectors. The output scorings are normalized over all possible segments using a Softmax activation. By computing individual scores for each *observation-* and *action feature* combination, the resulting policy is invariant to both permutations of the graph edges and the total number of available action candidates.

Communication: We focus in this work on decentralized actor architectures, requiring no or minimal global communication during inference, thus minimizing the computational overhead of communication protocols. While Kortus *et al* (2023) uses multi-head attention (MHA) to learn an agreement between segment candidates, we consider this mechanism as a form of communication and thus reallocate it for all multi-agent architectures from the actor to the centralized critic, reducing the computational cost of evaluating the policy.

Safety policy layer: To correct the predicted local policies for duplicate assignments, we propose, similar to Kortus *et al* (2025), the usage of a centralized safety layer (Dalal *et al* 2018, Sheebaelhamd *et al* 2021), performing for every reconstruction step an action correction for the learned joint policy by solving a LSAP. The safety layer ensures during both training and inference a full or partial unique matching between source ($\mathcal{V}_\text{S}$) and target vertices ($\mathcal{V}_\text{T}$) defined by

$$\begin{aligned} \min \quad & \sum_{(i,j)\in\mathcal{E}} \widehat{\mu}_{ij} c_{ij} \\ \text{s.t.} \quad & \sum_{i\in\mathcal{V}_\text{S}} \widehat{\mu}_{ij} = 1, \quad j \in \mathcal{V}_\text{T}, \\ & \sum_{j\in\mathcal{V}_\text{T}} \widehat{\mu}_{ij} \leqslant 1, \quad i \in \mathcal{V}_\text{S} \end{aligned} \tag{5}$$

that minimizes the required cost of deviating from the proposed local policies. Here $c_{ij} \in \hat{\boldsymbol{\mathcal{C}}}$ are the individual elements of a $|\mathcal{V}_T| \times |\mathcal{V}_S|$ cost matrix, defined either by infinite cost for assignments already occupied by another track due to its initial seeding mechanism or by the L2-norm of the local policy to the one-hot encoding of the corresponding target vertex, according to

$$c_{ij} = \begin{cases} \|\boldsymbol{\mu}^i\left(a_j|\boldsymbol{o}\right) - \mathbb{1}\left(a_j\right)\|_2^2 & \text{if not used for seeding} \\ \infty & \text{otherwise.} \end{cases} \tag{6}$$

Solving an LSAP for each interaction in the environment scales with $\mathcal{O}(n^3)$ as the number of hits n in the detector layer grows, therefore adding additional cost to the agent evaluation. For the experiments conducted in this work, solving the LSAP for a dense cost matrix posed no unsustainable cost (see appendix A), yet scaling the approach to larger detectors might require additional considerations. The impact of the safety layer and possible relaxations of the problems for scaling the solution to larger detector setups are therefore discussed in section 6.

By projecting the unsafe action, the action-corrected policy becomes inherently deterministic, requiring off-policy optimization and an exploratory policy for generating training samples. We sample track candidates with random exploration using parameter noise (Fortunato *et al* 2018, Plappert *et al* 2018). We therefore replace the linear layers of the pointer mechanism with noisy linear layers (Fortunato *et al* 2018).

BB differentiation: To provide gradient information for the combinatorial solver integrated into the safety layer we follow the BB scheme by Vlastelica *et al* (2020). The authors proposed for differentiating through combinatorial solvers of the general form $y(\boldsymbol{\mathcal{C}}) = \arg\min_{y\in\mathcal{Y}} c(\boldsymbol{\mathcal{C}}, y)$, substituting the piecewise constant solvers mapping of combinatorial solvers at the point $\hat{\boldsymbol{\mathcal{C}}}$ by a linear interpolation between the points $\hat{\boldsymbol{\mathcal{C}}}$ and $\boldsymbol{\mathcal{C}}'$ according to

$$\nabla^{\mathrm{BB}}_{\boldsymbol{\mathcal{C}}} f_\lambda\left(\hat{\boldsymbol{\mathcal{C}}}\right) := -\frac{1}{\lambda}\left[y\left(\hat{\boldsymbol{\mathcal{C}}}\right) - y_\lambda\left(\boldsymbol{\mathcal{C}}'\right)\right], \quad \text{where} \tag{7}$$

$$\boldsymbol{\mathcal{C}}' = \mathrm{clip}\left(\hat{\boldsymbol{\mathcal{C}}} + \lambda\frac{\mathrm{d}L}{\mathrm{d}y}\left(y\left(\hat{\boldsymbol{\mathcal{C}}}\right)\right), 0, \infty\right). \tag{8}$$

Here, $y(\hat{\boldsymbol{\mathcal{C}}})$ and $y(\boldsymbol{\mathcal{C}}')$ are solutions generated by predicted and perturbed costs. Further, $\lambda \in \mathbb{R}^+$ functions as a tunable hyperparameter, interpolating between truthfulness and informativeness of the gradients (Vlastelica *et al* 2020). The usefulness of the gradient information for particle tracking has already been demonstrated in Kortus *et al* (2025).

Cost margins: With an increasing number of solution sets, the policy becomes prone to settle changes in the cost matrix, limiting generalization. Sahoo *et al* (2023) proposed adding random noise to the predicted cost, increasing the margin to the decision boundaries of the predictive output[5]. As we found this mechanism to be highly unstable for our use case, we instead propose including an additional component $\nabla^{\leftrightarrow}_{\boldsymbol{\mathcal{C}}}$ to the BB-scheme, with

$$\nabla^{\mathrm{BB}}_{\boldsymbol{\mathcal{C}}} f_\lambda\left(\hat{\boldsymbol{\mathcal{C}}}\right) + \nu\nabla^{\leftrightarrow}_{\boldsymbol{\mathcal{C}}} f\left(\hat{\boldsymbol{\mathcal{C}}}\right), \quad \text{where} \quad \nabla^{\leftrightarrow}_{\boldsymbol{\mathcal{C}}} f\left(\hat{\boldsymbol{\mathcal{C}}}\right) = y\left(\hat{\boldsymbol{\mathcal{C}}}\right), \tag{9}$$

forcing the assignments of the joint policy $\boldsymbol{\mu}$ in the direction of lower assignment costs. The influence of $\nabla^{\leftrightarrow}_{\boldsymbol{\mathcal{C}}}$ can be controlled using the hyperparameter $\nu \in \mathbb{R}^+$, where larger values of ν enforce stronger cost margins at the cost of introducing increased bias to the policy gradients. We find that the cost margin gradient component is only nominally sensitive to the exact choice of weighting constant ν (see appendix B for more details), thus requiring only a coarse-grained tuning of the parameter.

Centralized critic: To mitigate instationarity introduced by the otherwise independent learners (Tan 1993, Sunehag *et al* 2018), we propose centralized factored critic functions for state- $V^\theta(\boldsymbol{o}_t)$ and action-value function $Q^\theta(\boldsymbol{a}_t|\boldsymbol{o}_t)$, decomposing the global value function into agent-wise values (Sunehag *et al* 2018) according to

$$Q\left(\boldsymbol{a}_t, \boldsymbol{o}_t\right) \approx \frac{1}{N}\sum_{i=1}^{N} Q^{(i)}_\theta\left(a^{(i)}_t, o^{(i)}_t, \phi\left(\boldsymbol{o}_t, \boldsymbol{o}_t\right)\right) \tag{10}$$

$$V\left(\boldsymbol{o}_t\right) \approx \frac{1}{N}\sum_{i=1}^{N} V^{(i)}_\theta\left(o^{(i)}_t, \phi\left(\boldsymbol{o}_t\right)\right). \tag{11}$$

Each agent-wise value is composed using local $(a^{(i)}_t, o^{(i)}_t)$ and global information ($\phi(\cdot)$, where $\phi(\cdot)$ is a shared communication channel), utilizing a mixture of additive (Bahdanau *et al* 2015) and self-attention (Iqbal and Sha 2019). To provide for each agent a single feature, we compress the set of agent observations $\langle \boldsymbol{h}_{\mathrm{obs}}, \boldsymbol{h}^{(1)}_{\mathrm{act}}, \ldots, \boldsymbol{h}^{(N)}_{\mathrm{act}} \rangle$ for both V^θ and Q^θ. For the action dependent Q-function, we model the compressed representation $\boldsymbol{h}^{(i)}_Q$ by a joint policy-weighted function of observation- action features according to

$$\boldsymbol{h}^{(i)}_Q = \left[\sum_{j=1}^{M} \boldsymbol{\mu}^i\left(\boldsymbol{a}_{t,j}, \boldsymbol{o}_t\right)\left(\boldsymbol{h}^{\mathrm{emb},(i)}_{\mathrm{obs,i}} + \overline{\boldsymbol{h}}^{\mathrm{emb},(i)}_{\mathrm{act},j}\right)\right]. \tag{12}$$

Here, $\overline{h}$ is an assembled feature over true and uncorrelated reference action features aggregated as a weighted sum over multiple random samples from a replay buffer $\mathcal{D}$ following

$$\overline{\boldsymbol{h}}^{\mathrm{emb},(i)}_{\mathrm{act},j} = \boldsymbol{h}^{\mathrm{emb},(i)}_{\mathrm{act},j} + \gamma \sum_{t',i',j'\sim\mathcal{D}} \boldsymbol{h}^{\mathrm{emb},\left(i'\right)}_{\mathrm{act},j',t'}, \tag{13}$$

where γ is a hyperparameter. This expression functions as a smoothing and regularization term with contextual information, allowing for reduced variance during training, improving convergence. For the action-independent state-value function V^μ_θ, the weighting of the action features is replaced by a learnable weighting, modeled using an additive attention mechanism (Bahdanau *et al* 2015) according to

[5] Additionally Rolínek *et al* (2020a, 2020b) proposed a cost margin mechanism enforcing increased margins by penalizing individual predictions based on ground truth information; this mechanism is due to the usage of ground-truth however, incompatible with our MARL approach.

$$\boldsymbol{h}_V^{(i)} = \left[\boldsymbol{h}_{\text{obs}}^{(i)} + \sum_{j=1}^{M} \alpha_j \boldsymbol{h}_{\text{act},j}^{(i)}\right], \quad \text{with} \tag{14}$$

$$\alpha_j^{(i)} = \boldsymbol{v}^T \tanh\left(\boldsymbol{W}_1 \boldsymbol{h}_{\text{act},j}^{\text{emb},(i)} + \boldsymbol{W}_2 \boldsymbol{h}_{\text{obs}}^{\text{emb},(i)}\right). \tag{15}$$

The soft weighting makes the cross-state regularization for variance reduction obsolete. Further, we encourage global communication between agents in form of two stacked self-attention blocks with layer normalization (Ba *et al* 2016) and skip connections (He *et al* 2016), each defined as

$$\boldsymbol{h}_{Q/V}^{(i,l)} = \text{LN}\left(\boldsymbol{h}_{Q/V}^{(i,l-1)} + \text{ReLU}\left(\text{MHA}\left(\boldsymbol{h}_{Q/V}^{(1:N,l-1)}\right)\right)\right). \tag{16}$$

Finally, factored values are obtained as the average agent-wise estimate conditioned on $\boldsymbol{h}_Q^{(i)}/\boldsymbol{h}_V^{(i)}$ using an MLP. The value range for Q and V is restricted for either raw- (sigmoid) or normalized rewards (tanh) accordingly (additional details in section 4.3) and scaled by the learnable parameter s.

$$Q(s,a) = -\frac{1}{N}\sum_{i=1}^{N} s \cdot \sigma\left(\Phi_Q\left(\boldsymbol{h}_Q^{(i)}\right)\right) \tag{17}$$

$$V(s) = -\frac{1}{N}\sum_{i=1}^{N} s \cdot \tanh\left(\Phi_V\left(\boldsymbol{h}_V^{(i)}\right)\right). \tag{18}$$

For completed particle tracks without valid assignments (early termination), we employ a value masking, where the relevant local agent-wise value estimates are excluded from the global value estimate. This representation prevents the observation of rewards obtained after early termination, posing additional complexity to the credit assignments (Cohen *et al* 2021), however, we choose the masking mechanism in favor of simplicity of the overall architecture[6].

4.3. Optimization of agents

The following section outlines the various optimization schemes used to optimize both unconstrained and constrained agents. Special emphasis is placed on the details and necessary modifications when applying these schemes to particle tracking.

Unconstrained on-policy baseline: We optimize an unconstrained joint policy using the multi-agent proximal policy optimization algorithm (MAPPO) (Ma and Luo 2022, Yu *et al* 2022), providing an extrapolation of the learning abilities of Kortus *et al* (2023) to a collaborative multi-agent setting. We use the architecture described in section 4.2, replacing the deterministic joint policy μ_θ by an unconstrained stochastic policy π_θ and a centralized state-value estimator $V_{\boldsymbol{\pi}}^{\theta}$. We estimate team advantages using the generalized advantage estimator (Schulman *et al* 2016) and employ independent reward normalization for calorimeter and tracker layer, following the normalization scheme in Kortus *et al* (2023).

Off-policy optimization: To cope with the deterministic safety-layer corrected policies, we optimize it similarly to Sheebaelhamd *et al* (2021), using a multi-agent variant of the *deep deterministic policy gradient* (DDPG) algorithm (Lillicrap *et al* 2016). However, while Sheebaelhamd *et al* (2021) uses the multi-agent DDPG algorithm (Lowe *et al* 2017), we found the MATD3 (Ackermann *et al* 2019) algorithm with two critic networks, mitigating overestimation bias, together with periodical hard critic updates, worked superior for our use case. We found the independent reward normalization mechanism to have a negative impact on the policy updates and thus only perform equal weighting of tracking and calorimeter transitions in the critic loss. Finally, we use a replay buffer with a small buffer size, owed to the quickly changing distribution of samples of the large joint action space (Hu *et al* 2021).

5. Experiments

For the studies reported in this work, we rely on Monte-Carlo simulations of detector readout data (Kortus *et al* 2022), generated using the GATE toolkit (Jan *et al* 2004, 2011) based on the Geant4

[6] While we did not witness significant issues in credit assignment, incremental updates of the architecture could introduce absorbing states for agents with early termination (Cohen *et al* 2021), potentially further improving the learning abilities.

Table 1. Overview of all considered RL and MARL particle tracking schemes evaluated in section 5.

Name	Algorithm	Centr. V/Q	γ	SL(T)	SL(E)	SL-grad.
PPO	Schulman *et al* (2017)					
PPO+LSA	Schulman *et al* (2017)				✓	
MAPPO	Yu *et al* (2022)	✓				
MATD3+LSA (BB)	Ackermann *et al* (2019)	✓	0.75	✓	✓	BB Vlastelica *et al* (2020)
MATD3+LSA (BB$_\nu^{\leftrightarrow}$)	Ackermann *et al* (2019)	✓	0.25	✓	✓	BB + ours

simulation framework (Agostinelli *et al* 2003, Allison *et al* 2006, 2016). The dataset consists of multiple simulations of a scanning pencil beam with and without water phantom (100 mm, 150 mm and 200 mm), positioned between the particle beam and detector. For the pencil beam source, a Gaussian beam with $\sigma_x = \sigma_y = 5$ mm, an angular divergence $\sigma_\theta = \sigma_\phi = 2.8$ mrad and 3 mradmm beam emittance is modeled. The data is further diversified by manually splitting the data into synthetic readout frames of different particle densities (p^+/F) of 50, 100, 150, and 200, which covers a range of particle counts expected for a real beam-detector setup used in proton computed tomography. Each simulation consists of 10 000 simulated primary particles. All data is publicly available on Zenodo (Kortus *et al* 2022). As the detector prototype in Alme *et al* (2020) is currently still under construction, no additional results on the physical-world optimization or simulation-to-reality gaps are provided and left for future work.

Configurations: To explore the performance of single- and multi-agent systems of various degrees of complexity, we construct variations of the agent described in the previous sections, summarized in table 1. Each variant is constructed based on the selected optimization algorithm, the usage of a safety layer (during training SL(T) and execution SL(E)) as well as the differentiation scheme. We could not find a stable MATD3 configuration without a safety layer that consistently converged to low-reward solutions, and thus excluded it from the results. The single agent results for PPO and PPO+LSA are based on the trained models in Kortus *et al* (2023).

Training procedure: We use particle simulations without absorber material between the beam source and detector for optimization, providing a worst-case scenario in terms of secondary production and track length. We then train, for each configuration in table 1, five independent policies on sampled track candidates for 3000 iterations with a particle density of 50 primary particles per readout frame to obtain robust results with confidence intervals.

Baselines: In addition to the multi-agent schemes listed in table 1, we compare the reconstruction performance with both two single-agent variants of particle tracking described in Kortus *et al* (2023) (with an additional centralized version using the proposed safety layer during inference) and a sequential track follower searching for solutions that minimize the total amount of scattering (Pettersen *et al* 2020). To obtain comparable results, all techniques construct the initial seed used for tracking also using ground-truth information. To situate our approach within current state-of-the-art tracking strategies, we also report supplementary pilot results in appendix C, comparing it with a global GNN-based edge-classification model developed with similar layerwise constraints (Kortus *et al* 2025), illustrating how MARL performance compares under increasing seeding error rates.

Performance metrics: We assess and compare the performance of the proposed tracking algorithms using track purity (p) and efficiency (ϵ), estimated after prior rejecting partial or implausible tracks using simple cuts for scattering angle and energy deposition according to Pettersen *et al* (2021). We select according to Kortus *et al* (2023) an angular cut of $\Delta\theta_{\max} = 271$ mrad, corresponding to an 2σ upper bound for particles in the last layer before stopping using extrapolated values from the PSTAR database (Berger *et al* 1999) and an energy cut of $\Delta E_{\min} = 2.5\,\mathrm{keV}\,\mu\mathrm{m}^{-1}$ (Pettersen *et al* 2020, 2021). Additionally, a minimum track length of 4 layers is enforced. For assessing the correctness of a track, we rely on a *perfect matching* criterion, where all hits in a track need to be correctly assigned.

5.1. Optimization and tracking performance

We examine and compare the performance for all configurations in table 1 to identify and quantify the necessary factors for multi-agent-based particle tracking using MARL. Figure 4 shows the average reward obtained during training as a function of network updates and sampled track segments. Here, we find similar training performance for the on-policy MAPPO and off-policy MATD3 approaches for an equal number of training iterations. However, due to the on-policy nature of MAPPO, requiring data generated from the current policy, this approach requires significantly more transitions to converge and is

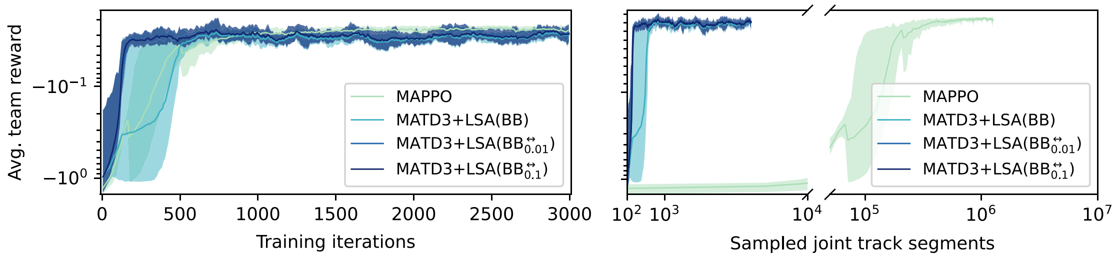

Figure 4. Average return obtained by the agents over time during training, plotted as a function of performed updates (for MAPPO: iteration over all epochs are counted as a single update) and sampled track transitions.

thus significantly more sample inefficient than the off-policy MATD3 algorithm, utilizing a replay buffer. Further, while all multi-agent variants except for the unconstrained MATD3 approach, which we excluded from the experiments, converge to high average team rewards, MAPPO converges consistently to the highest average reward, suggesting the best optimization behavior of all. Finally, we find that both constrained agents with cost margins show significantly faster convergence to high rewards, requiring approximately 300 training iterations less than the other agents.

Table 2 extends the previous results by summarizing the reconstruction performance (purity p and efficiency ϵ) of all MARL and baseline algorithms. We find that, while achieving lower average rewards compared to MAPPO, MATD3+LSA ($BB^{\leftrightarrow}_{\nu}$) outperforms all baseline and MARL variants in both configurations of ν by a significant margin. Especially for higher particle densities, the constrained policy with cost margins can benefit from the increased assignment complexity, outperforming the single-agent and unconstrained algorithms. We find the safety layer to be a critical component in multi-agent tracking, allowing for efficient sampling during training and inference, simplifying spatial credit assignment across agents, while avoiding duplicate assignment of particle hits. As previously outlined and supplemented by appendix B, we find the performance of MATD3+LSA($BB^{\leftrightarrow}_{\nu}$) to be robust to the exact choice of ν, producing similar results for both selected configurations. Therefore no extensive optimization of this parameter is needed, and only a coarse-grained optimization might be necessary to transfer the result to different detector setups.

To quantify the impact of the multi-agent optimization, we compare the performance of MATD3+LSA($BB^{\leftrightarrow}_{\nu=0.1}$) with a post-training centralized version of the single-agent PPO algorithm (PPO+LSA). Table 3 shows that PPO+LSA achieves similar performance, with only slight improvements in performance for the multi-agent approach. We find that the overall difference in performance is statistically not or only marginally significant (avg. p-values obtained by one-sided t-test (Welch 1947): p: 0.19, ϵ: 0.12)), demonstrating the strong ability of single-agent RL to efficiently learn reasonable conditional probabilities usable to resolve assignment conflicts during inference. Similar results are presented in Kortus *et al* (2025) for supervised learning. However, for large particle multiplicities (e.g. 200 p^{+}/F) we find the constrained multi-agent approach to outperform the single-agent approach by 0.75 percentage points (pp) (p-value: 0.03) in purity and 1.12 pp (p-value: 0.02) in efficiency, while only using limited information of the single-agent reward, indicating the usefulness of constrained multi-agent optimization to account for high amounts of ambiguities in hit candidates. Supplementary comparisons with a supervised edge-classification approach (Kortus *et al* 2025) in appendix C show that the proposed MARL approach achieves comparable performance for low to moderate synthetic seeding error rates (see figure 12). Especially for high particle densities and residual energies, MARL proves to be a strong competitor; however, its effectiveness diminishes as readout frames become increasingly sparse.

5.2. Effectiveness of cost margins

We verify the effectiveness of the enforced cost margins, described in section 4.2, by analyzing the predictive entropy of the learned policies. We rely on this measure as a proxy for quantifying the distance to the closest decision boundary, where lower predictive entropies suggest larger margins due to the lower ambiguity in the assignment probabilities. While this does not guarantee a monotonous dependency between entropy and cost margins, large discrepancies are a strong indication of differences in cost margins. Figure 5 shows the distribution of the agents' local policies estimated over all decisions generated over a subset of the first five environments in the dataset for multiple particle density and phantom configurations. We find that local agent policies trained without enforced cost margins show the highest predictive uncertainties (Avg. entropy $\overline{H}(\mu) = 4.099 \pm 0.221$), indicating only minimal separation or spread of assignment cost. For both parameter values of ν, weighing the cost-margin gradient,

Table 2. Reconstruction performance for water phantoms of 100, 150 and 200 mm thickness and 100, 150 and 200 primaries per frame p^+/F. The highest scores are highlighted in bold. Results for Track follower and PPO are reproduced from Kortus *et al* (2023). CC BY 4.0.

		100 mm Water Phant		150 mm Water Phant		200 mm Water Phant	
p^+/F	Algorithm	p [%] (↑)	ϵ [%] (↑)	p [%] (↑)	ϵ [%] (↑)	p [%] (↑)	ϵ [%] (↑)
50	Track follower	88.1 ± 0.0	79.7 ± 0.0	90.3 ± 0.0	82.7 ± 0.0	91.2 ± 0.0	83.8 ± 0.0
	PPO	92.5 ± 0.2	81.5 ± 0.3	93.8 ± 0.1	84.0 ± 0.4	94.5 ± 0.1	85.5 ± 0.2
	MAPPO	80.1 ± 21.7	70.3 ± 19.5	82.7 ± 19.5	73.6 ± 18.0	83.9 ± 19.7	75.8 ± 18.0
	MATD3+LSA (BB)	56.6 ± 21.5	48.6 ± 19.3	63.5 ± 22.4	55.2 ± 21.1	68.8 ± 22.9	60.1 ± 22.9
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.01}$)	96.2 ± 0.1	**84.0 ± 0.1**	**97.0 ± 0.1**	**85.9 ± 0.1**	**97.3 ± 0.1**	**87.3 ± 0.0**
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**96.3 ± 0.2**	**84.0 ± 0.2**	96.9 ± 0.1	85.7 ± 0.1	**97.3 ± 0.1**	87.2 ± 0.2
100	Track follower	83.0 ± 0.0	74.6 ± 0.0	86.6 ± 0.0	79.0 ± 0.0	87.4 ± 0.0	80.3 ± 0.0
	PPO	85.7 ± 0.2	75.1 ± 0.5	89.0 ± 0.2	79.1 ± 0.5	89.5 ± 0.1	80.9 ± 0.3
	MAPPO	71.3 ± 24.1	62.4 ± 21.5	75.1 ± 23.0	66.3 ± 21.3	76.3 ± 23.3	68.8 ± 21.2
	MATD3+LSA (BB)	40.2 ± 20.4	34.2 ± 17.6	48.6 ± 23.2	42.0 ± 20.8	55.0 ± 25.3	48.1 ± 23.4
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.01}$)	**91.9 ± 0.2**	**79.5 ± 0.2**	**94.1 ± 0.1**	**82.5 ± 0.2**	93.6 ± 0.1	**83.5 ± 0.1**
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**91.9 ± 0.2**	**79.5 ± 0.2**	94.0 ± 0.2	82.4 ± 0.2	**93.7 ± 0.1**	**83.5 ± 0.2**
150	Track follower	79.1 ± 0.0	70.9 ± 0.0	83.2 ± 0.0	75.7 ± 0.0	84.7 ± 0.0	77.7 ± 0.0
	PPO	80.6 ± 0.3	70.8 ± 0.6	84.0 ± 0.1	74.5 ± 0.6	85.5 ± 0.2	77.1 ± 0.3
	MAPPO	65.0 ± 24.4	57.2 ± 21.8	69.4 ± 23.4	61.3 ± 21.9	71.3 ± 24.4	64.3 ± 22.2
	MATD3+LSA (BB)	31.4 ± 18.0	26.6 ± 15.3	39.6 ± 21.5	34.0 ± 18.9	46.4 ± 24.4	40.6 ± 22.0
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.01}$)	**88.8 ± 0.2**	**76.8 ± 0.3**	90.9 ± 0.2	**79.2 ± 0.2**	91.2 ± 0.2	81.1 ± 0.2
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**88.8 ± 0.4**	**76.7 ± 0.**4	**91.1 ± 0.3**	**79.2 ± 0.3**	**91.4 ± 0.2**	**81.2 ± 0.3**
200	Track follower	75.4 ± 0.0	67.4 ± 0.0	80.1 ± 0.0	72.9 ± 0.0	81.6 ± 0.0	75.0 ± 0.0
	PPO	75.5 ± 0.3	66.6 ± 0.6	80.3 ± 0.4	71.1 ± 0.6	81.9 ± 0.3	73.9 ± 0.4
	MAPPO	59.6 ± 23.6	52.8 ± 21.2	65.2 ± 23.5	57.6 ± 22.0	66.9 ± 24.8	60.5 ± 22.6
	MATD3+LSA (BB)	25.8 ± 15.8	21.8 ± 13.3	33.7 ± 19.4	28.9 ± 16.8	40.7 ± 22.7	35.6 ± 20.3
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.01}$)	84.7 ± 0.3	73.0 ± 0.3	88.2 ± 0.2	76.6 ± 0.3	88.2 ± 0.2	78.2 ± 0.2
	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**84.9 ± 0.3**	**73.3 ± 0.3**	**88.6 ± 0.3**	**76.7 ± 0.3**	**88.4 ± 0.3**	**78.3 ± 0.4**

Table 3. Reconstruction performance, measured in terms of purity p and efficiency ϵ for water phantoms of 100, 150 and 200 mm thickness and 100, 150 and 200 p^+/F. The highest scores are highlighted in bold. Results for PPO+LSA are generated with the models reproduced from Kortus *et al* (2023). CC BY 4.0.

		100 mm Water Phant		150 mm Water Phant		200 mm Water Phant	
p^+/F	Algorithm	p [%] (↑)	ϵ [%] (↑)	p [%] (↑)	ϵ [%] (↑)	p [%] (↑)	ϵ [%] (↑)
50	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**96.3 ± 0.2**	**84.0 ± 0.2**	96.9 ± 0.1	85.7 ± 0.1	**97.3 ± 0.1**	**87.2 ± 0.2**
	PPO+LSA	95.9 ± 0.2	83.3 ± 0.6	**97.0 ± 0.1**	**85.7 ± 0.4**	97.2 ± 0.3	**87.2 ± 0.4**
100	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**91.9 ± 0.2**	**79.5 ± 0.2**	**94.0 ± 0.2**	**82.4 ± 0.2**	**93.7 ± 0.1**	**83.5 ± 0.2**
	PPO+LSA	**91.5 ± 0.4**	79.0 ± 0.5	**94.0 ± 0.2**	82.3 ± 0.3	93.6 ± 0.4	83.3 ± 0.4
150	MATD3+LSA($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**88.8 ± 0.4**	**76.7 ± 0.4**	**91.1 ± 0.3**	**79.2 ± 0.3**	**91.4 ± 0.2**	**81.2 ± 0.3**
	PPO+LSA	88.4 ± 0.4	75.9 ± 0.9	90.5 ± 0.4	78.6 ± 0.6	90.8 ± 0.5	80.2 ± 0.5
200	MATD3+LSA ($\text{BB}^{\leftrightarrow}_{\nu=0.1}$)	**84.9 ± 0.3**	**73.3 ± 0.3**	**88.6 ± 0.3**	**76.7 ± 0.3**	**88.4 ± 0.3**	**78.3 ± 0.4**
	PPO+LSA	84.0 ± 0.5	72.0 ± 0.9	87.9 ± 0.4	75.8 ± 0.7	87.7 ± 0.8	77.1 ± 0.6

the long tail of the distribution is reduced significantly, lowering the average entropy by multiple orders of magnitude ($\overline{H}(\mu_{\nu=0.01}) = 0.241 \pm 0.002$ and $\overline{H}(\mu_{\nu=0.1}) = 0.022 \pm 0.003$). The steep reduction in average entropy, indicates that the optimization process benefits from the additional gradient component, effectively increasing the separation from decision boundaries. We find, similar to the results in table 2, that the reduction in uncertainty is robust to the exact choice of ν, showing only marginal different values that are likely due to random mechanisms during training.

5.3. Analysis of policy constraints and cost margins

To understand why certain agents, despite achieving similar rewards during training, exhibit vastly different outcomes in terms of reconstruction quality, the following section presents analyses of reward surfaces for different agents, together with their corresponding surfaces of reconstruction performance. By

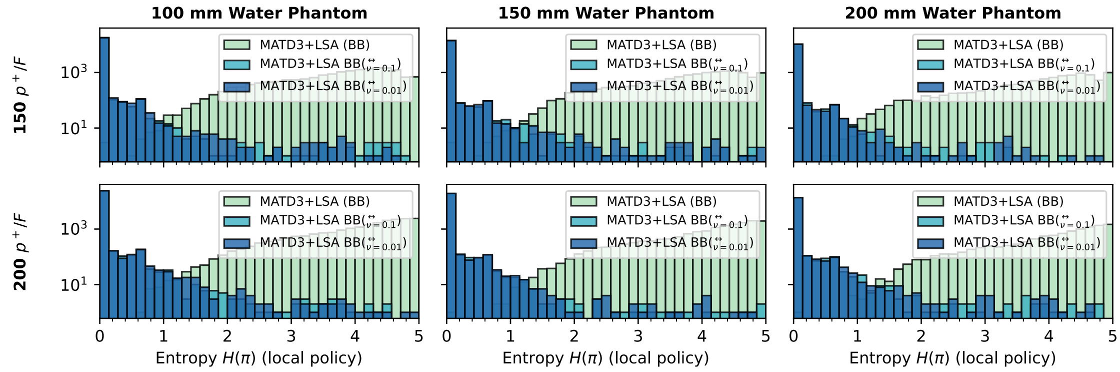

Figure 5. Distributions of the uncertainties in local policy predictions, measured as the predictive entropy for various water phantoms and particle densities. Techniques with enforced cost margins demonstrate significantly reduced uncertainties.

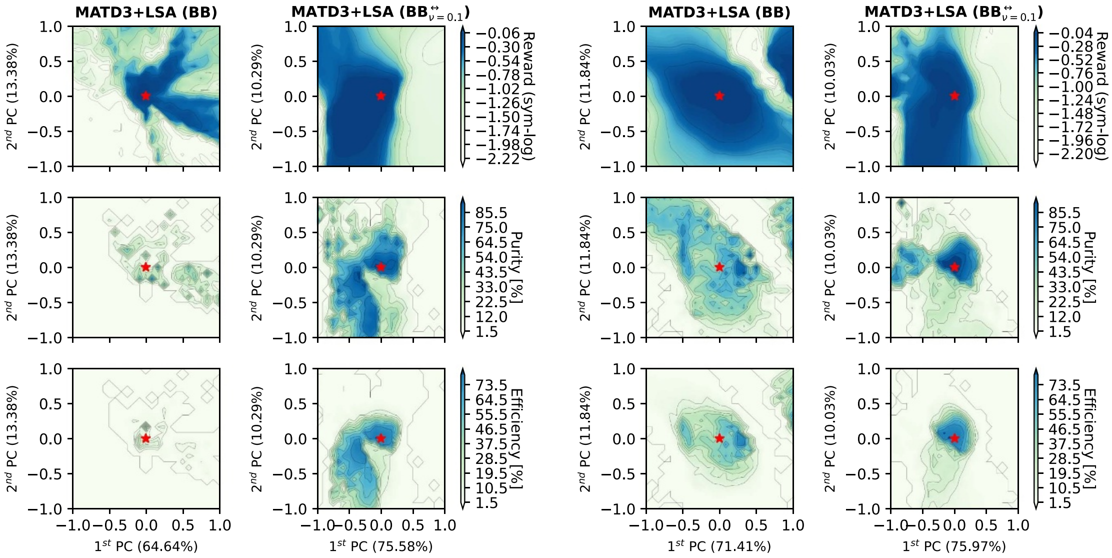

Figure 6. Two-dimensional reward and performance surfaces of multi-agent framework with (MATD3+LSA(BB$^{\leftrightarrow}_{\nu=0.1}$)) and without cost margins (MATD3+LSA(BB)) generated along the first two principal directions, calculated over the intermediate training checkpoints. Marked with ⋆ are the trained network parameters.

comparing the reward surfaces with the track reconstruction performance, we aim to compare and highlight discrepancies in optimization and generalization and highlight the importance of policy constraints as well as cost margins. Sullivan *et al* (2022) previously demonstrated the benefit of visualizing loss landscape for characterizing the complexity of learning tasks in RL, providing a compelling empirical tool for analyzing the otherwise complex optimization behavior of MARL. We generate all surfaces, based on the technique described in Li *et al* (2018), Sullivan *et al* (2022), as two-dimensional slices through the high-dimensional landscapes along two directions defined by $\boldsymbol{\nu}$ and $\boldsymbol{\eta}$ according to

$$f(\alpha,\beta) = \mathcal{L}\left(\boldsymbol{\theta}^* + \alpha\boldsymbol{\nu} + \beta\boldsymbol{\eta}\right). \tag{19}$$

We parameterize $\boldsymbol{\nu}$ and $\boldsymbol{\eta}$ as the first two principal components over saved training checkpoints, capturing the most informative directions of the training trajectory through the parameter space (Li *et al* 2018). All figures are generated for the $100\,p^+/F$, 100 mm phantom dataset with a resolution of 25×25 uniformly sampled parameter configurations in a region of $[-1,1]\times[-1,1]$ for cost margins and $[-3,3]\times[-3,3]$ for constrained and unconstrained policies. In the latter, we experienced multiple configurations where the policy showed numerical issues, resulting in the prediction of `nan` values, marked in black.

Cost margins: Analyzing the characteristic structure of reward and performance surfaces for agents with and without enforced cost margins, displayed in figure 6, we confirm the initial finding in section 5.1, that enforcing cost margins with the additional cost margin term is paramount, significantly improving

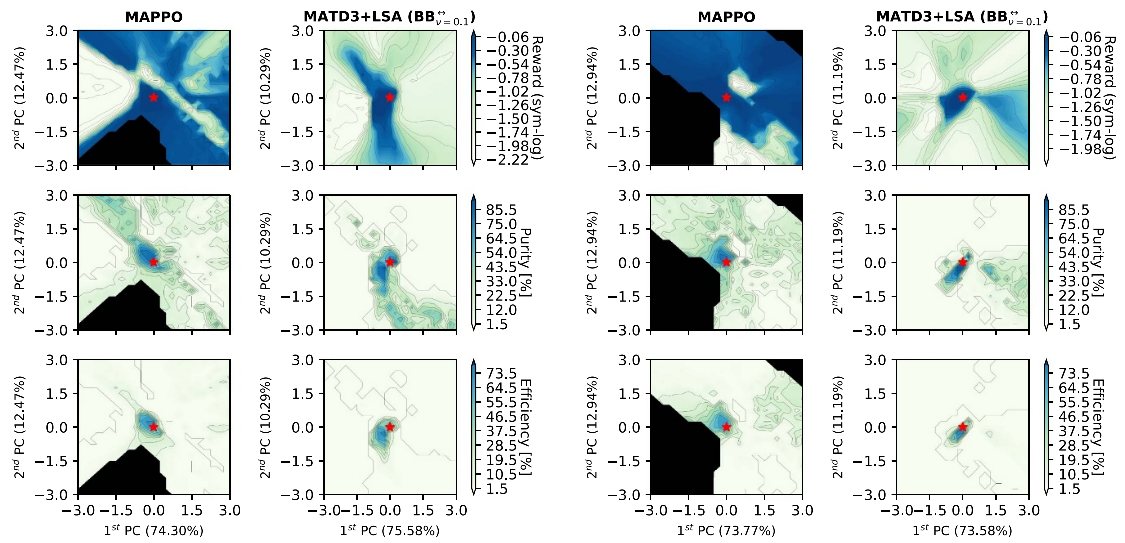

Figure 7. Two-dimensional reward and performance (purity and efficiency) surfaces of multi-agent framework with (MATD3+LSA($BB^{\leftrightarrow}_{\nu=0.1}$)) and without policy constraints (MAPPO) generated along the first two principal components, calculated over the intermediate training checkpoints. Marked with ⋆ are the trained network parameters.

both optimization and generalization performance. Although the reward surfaces for policies with and without cost margins exhibit a similar shape and therefore indicate a similar complexity of the learning task (Sullivan *et al* 2022) which is in agreement with figure 6, we observe a substantial disparity in the surfaces for reconstruction purity and efficiency. We find that the agents with cost margins (right) converge to regions characterized by relatively wide, smooth, and connected maxima, while the surfaces without cost margins (left) are dominated by multiple distinct, narrow minima in the loss landscape. We argue that the smooth, connected shapes in the loss landscape suggest both improved generalization performance (Hochreiter and Schmidhuber 1994) and increased robustness to perturbations, enabled by adequate separation from decision boundaries, while reducing the complexity during training. As a result, agents without cost margins fail to efficiently learn robust and generalizable patterns, therefore yielding subpar results on test data.

Policy constraints: Figure 7 visualizes the differences in learning abilities for the unconstrained MAPPO and constrained MATD3+LSA architecture with cost margins. Here, we find, similarly to figure 6, good agreement of the reward surfaces with wide and smooth regions of high reward, while the unconstrained policy shows extended regions of high reward. However, the reward surfaces of the unconstrained MAPPO correlate only moderately with the reconstruction performance, demonstrating a strong degeneracy of the reward surface introduced by the larger combinatorial space caused by unconstrained assignments. Therefore, a larger set of agent parameterizations reflects high-reward policies with small average scattering angles. Due to misaligned reward signals, allowing for high-reward solutions without constraint satisfaction, the unconstrained agents demonstrate a significant decline in performance with significant fluctuations between runs, governed by random effects during training (see table 2). The strong degeneracy of the reward surfaces demonstrates the necessity of policy constraints for collaborative multi-agent optimization. Alternatively, additional cost terms, representing soft constrained solutions, might be beneficial to improve the performance of unconstrained MARL for charged particle tracking. However, the additional complexity of balancing reward and cost terms most likely outweighs the possible improvements.

5.4. Functional similarities and prediction instabilities

While post-training centralized single-agent (PPO+LSA) and per-design centralized multi-agent policies (MATD3+LSA) achieve strong empirical reconstruction performances at low particle densities, MARL shows a clear and growing advantage as particle counts increase. Therefore, a remaining key question is whether the two approaches learn similar reconstruction policies, why centralized agents perform superior in high-particle scenarios, and how stable the optimization and final learned policies are, e.g. across random initializations. Especially, ensuring consistency in tracking performance with minimal variations in prediction is paramount for the unsupervised optimization of the approach and its application in potential safety-critical applications.

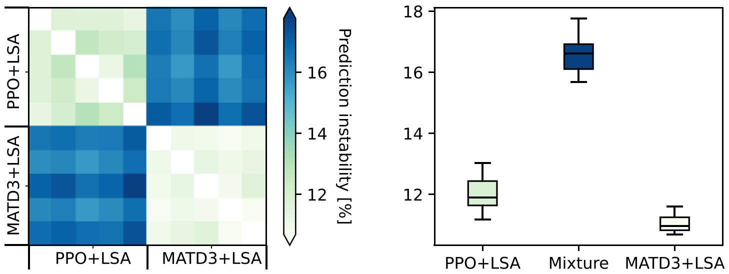

Figure 8. Prediction instabilities of trained reconstruction policies generated for different combinations of optimization algorithms and random initializations.

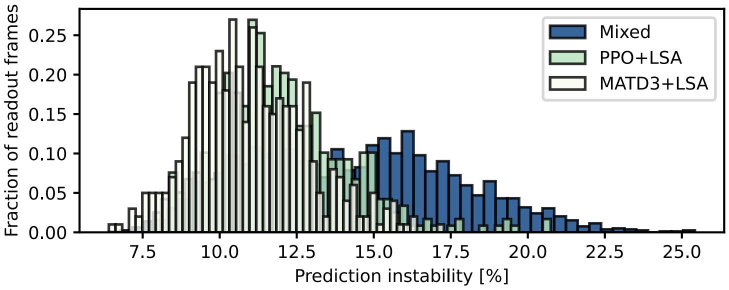

Figure 9. Distributions of prediction instabilities on a readout frame level generated for all combinations presented in figure 8.

To quantify potential prediction instabilities (Fard *et al* 2016, Klabunde *et al* 2023), we closely follow the techniques in Fard *et al* (2016), Klabunde and Lemmerich (2023), where the amount of disagreement between two predictors f_1 and f_2 is quantified as the average fraction of classification errors, defined as

$$d = \mathop{\mathbb{E}}_{x,f_{1,2}} \left[\mathbb{1}\left\{\arg\max f_1(x) \neq \arg\max f_2(x)\right\}\right]. \tag{20}$$

Klabunde and Lemmerich (2023) proposes an additional extension (min-max normalized disagreement), mapping the raw disagreement rates to a value range of [0, 1] providing better interpretability over the initial approach in Fard *et al* (2016). Following this definition, $d_{\text{norm}}(f_1,f_2)$ is calculated according to

$$d_{\text{norm}}(f_1,f_2) = \frac{d(f_1,f_2) - \min d(f_1,f_2)}{\max d(f_1,f_2) - \min d(f_1,f_2)}, \tag{21}$$

with $\min d(f_1,f_2) = |q_{\text{Err}}(f_1) - q_{\text{Err}}(f_2)|$ and $\max d(f_1,f_2) = \min(q_{\text{Err}}(f_1) + q_{\text{Err}}(f_2), 1)$, where q_{Err} is the error rate of a model. However, due to the sequential nature of RL, the presented concept of quantifying prediction instabilities is not directly applicable, as different predictions lead to changing track candidates. We thus calculate the prediction instability for all manually constructed correctly assigned states, avoiding the propagation of errors throughout the whole detector.

Figure 8 shows both the full correlation-like instability matrix for all combinations of trained agents across agent type and random initializations, as well as the grouped distribution of values. We find that PPO and MATD3+LSA show pronounced differences in training behavior, resulting in substantial prediction instabilities, with a median of approximately 16.5%. Across different random initializations of the same agent type, we find that the instabilities are reduced.

Our analysis reveals that the centralized multi-agent approach (MATD3+LSA) exhibits lower prediction instabilities compared to the single-agent method (PPO+LSA), with an average reduction of 0.98 pp (p-value: 0.01). We further find that while for both approaches average prediction instability is considerably low, outliers on a frame-by-frame level, in the form of a long tail of the otherwise Gaussian distribution (see figure 9), demonstrate more pronounced instabilities for complex readout frames, posing additional risk for the reconstruction of complex readout frames. Here, we find that our multi-agent approach is able to reduce the number of outliers more effectively compared to the single-agent approach. We argue that the improved stability for complex readout frames with a high amount of ambiguities is closely related to the previous results presented in table 3, highlighting the importance of multi-agent optimization for charged particle tracking.

6. Discussion

Our MARL framework with assignment constraints introduces a novel ground-truth-free particle tracking scheme, bridging the gap between iterative reconstruction algorithms and deep learning methods, and extending previous work presented in Kortus *et al* (2023). While Kortus *et al* (2023) demonstrated the feasibility of RL for charged particle tracking, we show that collaborative optimization with shared information and assignment constraints further improves performance. Nonetheless, several limitations remain or arise from the use of assignment constraints, which are discussed in the rest of this section.

Safety layer scalability and approximate solutions: For the experiments conducted in this work, we find that solving the safety layer at every interaction only introduces a marginal computational cost that does not significantly improve reconstruction. Yet, scaling this mechanism to larger detector systems requires additional considerations, as solving LSAP scales cubically w.r.t. particles in the readout frame. To ensure sufficient scaling, multiple optimizations can be integrated, including:

- **Parallelization:** To improve scaling of the LSAP solver, GPU-accelerated implementations (Date and Nagi 2016, Kawtikwar and Nagi 2024) provide significant performance gains for large-scale LSAP problems, improving scalability for large detector systems. Additional parallelization over multiple readout frames can further reduce the effective cost per particle track.
- **Sparsity:** As detector scale increases, the cost matrix can be increasingly sparsified due to structural constraints in the detector geometry (e.g. distant or opposite layers in barrel detectors). Further, a divide-and-conquer-type reduction of complexity can be achieved by partitioning the detector into feasible subproblems with independent cost matrices. Optimized solvers for sparse LSAP (e.g. LAPMOD (Volgenant 1996)) can exploit sparsity in cost matrices, reducing both memory usage and computational overhead.
- **Approximate solver:** Finally, if none of the previous optimizations for exact solvers can scale sufficiently, approximate alternatives, such as Sinkhorn (Brun *et al* 2022) or auction-based algorithms (Bertsekas 1988), provide scalable alternatives to exact LSAP solutions. To remain with feasible gradient information, adaptations to the gradient estimation scheme might be necessary. Soft approximations, however, often provide out-of-the-box gradients, removing the necessity for dedicated gradient estimators.

Track seeding: In this work the isolated performance of MARL-based tracking is investigated to limit coupling effects between both tracking and seeding procedures. As a replacement, conventional seeding mechanisms such as doublet and triplet finding (Mankel and Spiridonov 1999) can be integrated isolated from the main tracking procedure. Applying unique assignment constraints (equation (5)) further enables finding locally consistent seeds; however, interactions with the reconstruction algorithm are neglected. Interpreting the process of seeding as tracking under perceptually aliased observations (track history is aliased) opens a framework to directly integrate the seeding procedure into the agent architecture, limiting error cascades caused by isolated seeding. Initial studies showed large potential; further work is, however, still required to optimize both performance and integration of this mechanism.

Parameter efficiency: Compared to SOTA GNN-based algorithms, our reconstruction policy requires considerably more parameters to ensure robust convergence during optimization. This, however, comes at the cost of increased runtime during inference. Therefore, incremental updates require increased parameter efficiency to optimize runtimes. This can be achieved either by optimizing the proposed architecture or by using alternative optimization techniques such as parameter pruning (Han *et al* 2015). Initial results utilizing gradual parameter pruning during training are already promising; additional work is yet required.

Simulation to reality transfer: Due to the lack of a physical detector setup, we were not able to verify on-device training abilities of the proposed MARL reconstruction scheme. Therefore, further work is still required to demonstrate and quantify training performance on real detector systems and to quantify the simulation-to-reality gap introduced by utilizing simulated data for optimization.

7. Conclusion

In this paper, we introduce multiple extensions to an existing single-agent RL scheme for charged particle tracking, enabling the joint reconstruction of particle tracks in a multi-agent setting with additional (optional) assignment constraints. We realize the assignment constraints by an implicit, centralized safety layer, projecting the local unsafe actions onto global safe actions. We demonstrate the robust empirical performance of our approach on simulated data for a detector prototype designed for proton computed tomography. Our findings demonstrate that constrained optimization offers a significant advantage over its unconstrained MARL counterpart. We attribute the subpar convergence of unconstrained approaches to the high degeneracy of solutions that maximize the team reward signal while producing a significant amount of incorrect tracks and the increased complexity of spatial credit assignment in the unconstrained action space. While we were able to achieve similar performance for a post-hoc centralized agent at low to moderate particle densities, we find that learning particle tracking with constraints both improves reconstruction at high particle densities and reduces predictive instability across random initializations. This suggests that while single-agent training can suffice under simpler conditions, it struggles to generalize in more complex regimes where ambiguity is prevalent. Supplementary results analyzing the sensitivity of the proposed approach to increasingly imperfect seeding highlight the competitive performance of our approach beyond sequential track follower architectures. When compared to a state-of-the-art edge-classification architecture, we find significant potential, particularly for high particle multiplicities and high residual energy corresponding to longer tracks, encouraging further research. Using multi-agent techniques for optimization provides more flexibility than single-agent RL enabling the design of more sophisticated reward functions utilizing information that can only be obtained collaboratively for an aggregate over multiple particle tracks in a readout frame. Further modeling all tracks in a readout frame provides the potential to resolve aliased local observations, e.g. partial description of seeds, enabling enhanced architectures incorporating complex processes such as seeding. With the results presented, we aim to extend this work in the future to a generalized and adaptive particle tracking framework that can learn policies for different particle/tracking detectors with additional components, e.g. magnetic fields, and is also able to adapt to dynamic changes introduced by, e.g. aging of the detector components. We further aim to further reduce the computational complexity of our approach to enable reliable scaling to arbitrary detector designs.

Members of the Bergen pCT Collaboration

Max Aehle[a], Johan Alme[b], Gergely Gabor Barnaföldi[c], Tea Bodova[b], Vyacheslav Borshchov[d], Anthony van den Brink[e], Mamdouh Chaar[b], Viljar Eikeland[b], Gregory Feofilov[f], Christoph Garth[g], Nicolas R. Gauger[a], Georgi Genov[b], Ola Grøttvik[b], Håvard Helstrup[h], Sergey Igolkin[f], Ralf Keidel[i], Chinorat Kobdaj[j], Tobias Kortus[a], Viktor Leonhardt[g], Shruti Mehendale[b], Raju Ningappa Mulawade[i], Odd Harald Odland[k,b], George O'Neill[b], Gabor Papp[l], Thomas Peitzmann[e], Helge Egil Seime Pettersen[k], Pierluigi Piersimoni[b,m], Maksym Protsenko[d], Max Rauch[b], Attiq Ur Rehman[b], Matthias Richter[n], Dieter Röhrich[b], Joshua Santana[i], Alexander Schilling[i], Joao Seco[o,p], Arnon Songmoolnak[b,j], Akos Sudar[c,q], Jarle Rambo Sølie[r], Ganesh Tambave[s], Ihor Tymchuk[d], Kjetil Ullaland[b], Monika Varga-Kofarago[c], Boris Wagner[b], RenZheng Xiao[b,v], Shiming Yang[b], Hiroki Yokoyama[e]

a) Chair for Scientific Computing, TU Kaiserslautern, 67663 Kaiserslautern, Germany; b) Department of Physics and Technology, University of Bergen, 5007 Bergen, Norway; c) Wigner Research Centre for Physics, Budapest, Hungary; d) Research and Production Enterprise “LTU” (RPELTU), Kharkiv, Ukraine; e) Institute for Subatomic Physics, Utrecht University/Nikhef, Utrecht, Netherlands; f) St. Petersburg University, St. Petersburg, Russia; g) Scientific Visualization Lab, TU Kaiserslautern, 67663 Kaiserslautern, Germany; h) Department of Computer Science, Electrical Engineering and Mathematical Sciences, Western Norway University of Applied Sciences, 5020 Bergen, Norway; i) Center for Technology and Transfer (ZTT), University of Applied Sciences Worms, Worms, Germany; j) Institute of Science, Suranaree University of Technology, Nakhon Ratchasima, Thailand; k) Cancer Clinic, Haukeland University Hospital, 5021 Bergen, Norway; l) Institute for Physics, Eötvös Lorand University, 1/A Pazmany P. Setany, H-1117 Budapest, Hungary; m) UniCamillus – Saint Camillus International University of Health Sciences, Rome, Italy; n) Department of Physics, University of Oslo, 0371 Oslo, Norway; o) Department of Biomedical Physics in Radiation Oncology, DKFZ—German Cancer Research Center, Heidelberg, Germany; p) Department of Physics and Astronomy, Heidelberg University, Heidelberg, Germany; q) Budapest University of Technology and Economics, Budapest, Hungary; r) Department of Diagnostic Physics, Division of Radiology and Nuclear Medicine, Oslo University Hospital, Oslo, Norway; s) Center for Medical and Radiation Physics (CMRP), National Institute of Science Education and Research (NISER), Bhubaneswar, India; t) Biophysics, GSI Helmholtz Center for Heavy Ion Research GmbH, Darmstadt, Germany; u) Department of Medical Physics and Biomedical Engineering, University College London, London, UK; v) College of Mechanical & Power Engineering, China Three Gorges University, Yichang, People's Republic of China.

Data availability statement

The source code supporting the findings of this paper are available together with hyperparameters, data, and models on GitHub (https://github.com/SIVERT-pCT/marl-tracking) and Zenodo (https://doi.org/10.5281/zenodo.7426388).

Acknowledgment

We kindly acknowledge the support of the regional university computing center (RHRK). The ALPIDE chip was developed by the ALICE collaboration at CERN

Funding

This work was supported by the German federal state Rhineland-Palatinate (Forschungskolleg SIVERT), by the Research Council of Norway (Norges forskningsrad) and the University of Bergen, Grant Number 250 858. TK and NRG gratefully acknowledge the funding of the German National High-Performance Computing (NHR) association for the Center NHR South-West. JK is supported by the Alexander-von-Humboldt-Stiftung. The simulations and computations were executed on the high performance cluster 'Elwetritsch' at the University of Kaiserslautern-Landau (RPTU), which is part of the 'Alliance of High Performance Computing Rhineland-Palatinate' (AHRP).

Appendix A. Computational demands for solving an LSAP for the safety layer

Solving the LSAP, defined by the safety layer in section 4.2, comes with a worst-case runtime complexity of $\mathcal{O}(n^3)$ w.r.t. the number of hits in the detector layer, posing a potential bottleneck for reconstruction. To guarantee an efficient execution of the reconstruction algorithm, it is paramount to quantify the cost of this operation. We therefore provide general benchmark results for solving the LSAP of the action constraint layer using py-lapsolver[7] implementing the LAPJV algorithm (Jonker and Volgenant 1987). All following results were obtained on an AMD EPYC 9135 16-core processor using the 100 mm water phantom, which yields the highest residual energy and thus track length of all test data.

Figure 10 presents the runtime results of the LSAP solver as a function of particle density. Both single LSAP execution times for an entire detector layer (left) and approximate average LSAP costs for reconstructing a whole particle track are included. We include both randomly generated and true cost matrices for comparison:

- **Random cost matrix:** The cost matrix is generated for a fixed size $N \times N$, with each $c_{ij} \in \boldsymbol{C}$ sampled from a normal distribution with zero mean and unit variance. Solving the LSAP for the random cost matrix functions as an upper bound in complexity, as no structured sparsity, resulting in beneficial initial conditions, can be exploited by the solver.
- **True cost matrix:** As a realistic comparison, true cost matrices of reconstructed trajectories are used. To remove non-representative cost matrices that are significantly smaller than N, an additional filter, requiring the size of $\boldsymbol{C}$ to be at least 0.95% in both rows and columns is employed.

For simplicity, the approximate runtime per track is estimated for both random and true cost matrices by multiplying the average reconstruction time times the average of occupied detector layers (30), providing an upper bound on the total cost.

We find that for all tested configurations, no disproportionate overhead is introduced by the LSAP solver. The additional structure in the cost matrix introduced by the trained local policies further improves the performance significantly. Yet, with increasing particle count, the runtime scales polynomially, potentially requiring further optimization (see section 6).

[7] py-lapsolver, developed by Christoph Heindl, is available under https://github.com/cheind/py-lapsolver.

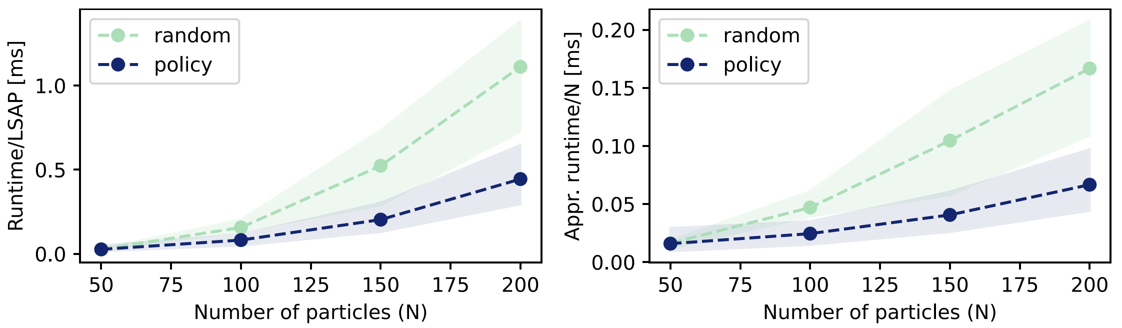

Figure 10. Average runtimes for evaluating the cost matrix and approximated worst-case runtime per track for a randomly generated cost matrix and actual cost matrix for the reconstruction policy $\boldsymbol{\mu}_\theta$.

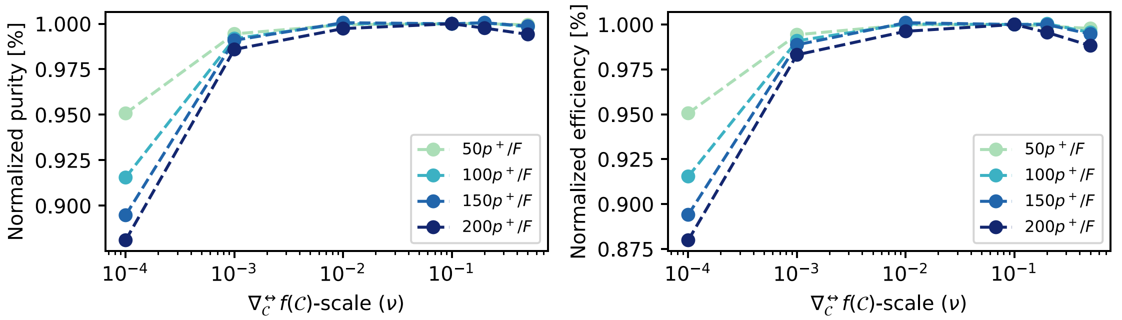

Figure 11. Ablation results, quantifying the sensitivity of the cost margin gradient $\nabla_{\boldsymbol{C}}^{\leftrightarrow} f(\boldsymbol{C})$ scaling factor ν, with respect to normalized reconstruction purity and efficiency. For each particle density (50, 100, 150 and $200p^+/F$), purity and efficiency are normalized w.r.t. $\nu = 0.1$.

Appendix B. Ablation studies of cost margin scaling

Section 4.2 introduces a cost margin term to the blackbox gradient scheme by Vlastelica *et al* (2020) that depends on the hyperparameter λ for scaling. While this mechanism proves to be effective in the presented empirical evaluation, low sensitivity of this constant is paramount to make it functional beyond this work. Figure 11 presents ablation results, demonstrating the sensitivity of the hyperparameter selection on the reconstruction performance. We therefore include normalized purity and efficiency scores (normalized w.r.t. $\nu = 0.1$) for the 100 mm water phantom at various particle counts.

As outlined in section 5.1, we find the performance to be robust to the exact parameter choice. Yet, it requires some degree of tuning to achieve good results. Values outside the range of $[0.01, 0.1]$ result in an increasing degradation of performance. With increasing particle density, this effect becomes more pronounced as the combinatorial complexity and, therefore, ambiguity increase. Therefore, careful tuning of this hyperparameter becomes increasingly important. As the approach is itself independent of ground-truth information during training, initial tuning of this value on simulated data might become necessary to obtain optimal results.

Appendix C. Imperfect seeding and global particle tracking

Ground truth seeding is implemented in this work to isolate the performance of the agents from unrelated processing steps to assess and contrast the reconstruction quality of proposed multi-agent reconstruction schemes with various sequential reconstruction algorithms. While this ensures controlled testing of the algorithms, relying on ground truth seeding is infeasible for real detector operation. Relying on ground truth seeding further prevents comparisons with global reconstruction schemes that operate independent of track seeds, as it systematically overestimates its realistic performance.

To examine the sensitivity of the proposed reconstruction scheme to imperfect seeding and provide a method-agnostic comparison with global reconstruction techniques, we derive a synthetic corrupted seeding mechanism that enables fine-grained control of error rates while ensuring geometrically coherent error patterns. With this mechanism, we extend our analysis in section 5.1 by an edge classification

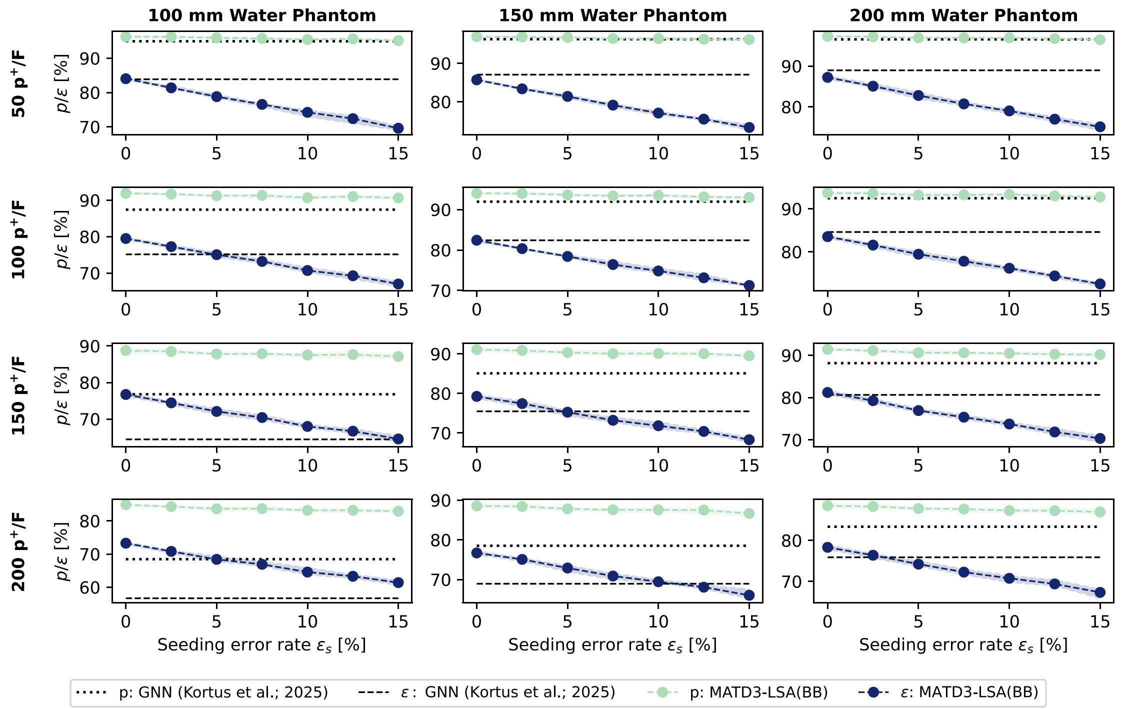

Figure 12. Sensitivity of the multi-agent MATD3-LSA tracking scheme to imperfect seeding introduced by synthetic errors for different particle counts and phantom geometries. Additional baseline results for a global edge-classification baseline are reproduced from Kortus *et al* (2025). CC BY 4.0.

GNN architecture developed for the investigated detector (Kortus *et al* 2025). This enables a direct comparison with state-of-the-art global reconstruction methods while also incorporating assignment constraints similar to the described safety layer.

Corrupted seeding: Synthetically injecting incorrect seeds into the ground truth seeding mechanism requires a geometrically plausible perturbation of assignment indices subject to unique assignment constraints. To ensure that the seeds do not conflict with the reconstruction policy, seeding is performed prior to evaluating the policy[8]. For each layer with N seeds, we draw a random number of corrupted seeds with

$$n \sim \text{Binomial}\left(N, \epsilon_s\right). \tag{22}$$

Subsequently, we generate n corrupted seeds by replacing the true successor $v_i \rightarrow v_j$ with an alternative neighbor $v_{j'}$ from the $k = 8$ nearest neighbors of v_j. To enforce geometric coherence, each neighbor is weighted proportional to the Euclidean distance to the true seed with

$$w_{jj'} = \frac{\exp\left(-\|\boldsymbol{p}_j - \boldsymbol{p}_{j'}\|\right)}{\sum_{k\in\mathcal{N}_{knn}} \exp\left(-\|\boldsymbol{p}_j - \boldsymbol{p}_k\|\right)}. \tag{23}$$

Random errors are constructed as a corruption set $\mathcal{C} = \{(v_j, v_{j'})\}$, where for every error (1) both v_j and $v_{j'}$ are distinct, and (2) no v_j appears in any $v_{j'}$. Selecting $v_{j'}$ that already appears as a ground-truth seed can invalidate two assignments simultaneously. To keep the total number of corrupted seeds consistent with the sampled budget n, we associate each pair with a cost $c_{jj'}$ and restrict the total cost of the corruption set $\mathcal{C}$ to n. Selecting a $v_{j'}$ that already appears as a ground-truth seed effectively affects two seeds at once, coming at a cost $c_{jj'} = 2$; otherwise the cost $c_{jj'}$ defaults to one.

Imperfect seeding and global particle tracking: Figure 12 visualizes the sensitivity of the multi-agent reconstruction scheme MATD3-LSA to imperfect seeding as a function of ϵ_s. Given the robustness to the

[8] In a realistic deployment, where the seeding algorithm is replaced by a conventional geometric seeding algorithm (e.g. doublet or triplet seeding), the sequence of seeding and policy evaluation can be modified to take advantage of the high quality of decisions obtained from the MARL policy.

parameter choice of ν (appendix B), we restrict the evaluation $\nu = 0.1$. Further, the reconstruction performance of the global edge-classification scheme in Kortus *et al* (2025) is included, enabling a detailed extrapolated comparison with state-of-the-art techniques. For both approaches, the average performance over five independent training runs is reported.

Across all particle densities and phantom geometries, we observe that the reconstruction performance degrades approximately linearly with the seeding error rate. Importantly, the degradation in reconstruction performance comes primarily at the cost of reconstruction efficiency, while the track purity remains largely unaffected. This behavior highlights the robustness of the MARL policy to compensate for invalid initial local observations obtained from seeding. By enforcing assignment constraints, the incorrect seeds are consistently guided to continue the correct track associated with the vertex $v_{j'}$. The resulting inconsistencies cause the track candidates to be removed by the employed track filter with only minimal impact on reconstruction purity.

Comparing the reconstruction performance to the GNN approach, we find that MARL-based tracking is generally outperformed for low particle density even with no or marginal seeding errors. Yet, we find improved generalization ability for densely occupied readout frames and high residual energies (reduced material in the phantom geometry). For $200p^+/F$ MARL attains higher reconstruction efficiency and track purity and continues to perform well even in the presence of considerable seeding errors. As a result, the sequential MAR method attains a lower per-edge error rate than the global edge-classification model. To maximize the overall performance of our proposed approach, the choice of the seeding mechanism is therefore paramount to ensure efficient operation and competitiveness with state-of-the-art reconstruction algorithms.

ORCID iDs

Tobias Kortus 0000-0002-0987-8544
Ralf Keidel 0000-0002-1474-6191
Nicolas R Gauger 0000-0002-5863-7384
Jan Kieseler 0000-0003-1644-7678

References

Ackermann J, Gabler V, Osa T and Sugiyama M 2019 Reducing overestimation bias in multi-agent domains using double centralized critics (arXiv:1910.01465v2)

Aehle M *et al* 2023 The Bergen proton CT system *J. Instrum.* **18** C02051

Aglieri Rinella G 2017 The ALPIDE pixel sensor chip for the upgrade of the ALICE inner tracking system *Nucl. Instrum. Methods Phys. Res. A* **845** 583–7

Agostinelli S *et al* 2003 GEANT4 - A simulation toolkit *Nucl. Instrum. Methods Phys. Res. A* **506** 250–303

Allison J *et al* 2006 GEANT4 developments and applications *IEEE Trans. Nucl. Sci.* **53** 270–8

Allison J *et al* 2016 Recent developments in GEANT4 *Nucl. Instrum. Methods Phys. Res. A* **835** 186–225

Alme J *et al* 2020 A high-granularity digital tracking calorimeter optimized for proton CT *Front. Phys.* **8** 1–20

Alshiekh M, Bloem R, Ehlers R, Könighofer B, Niekum S and Topcu U 2017 Safe Reinforcement Learning via Shielding *32nd AAAI Conf. on Artificial Intelligence, AAAI 2018* (AAAI press) pp 2669–78

Andrychowicz O A I M *et al* 2020 Learning dexterous in-hand manipulation *Int. J. Robot. Res.* **39** 3–20

Ba J L, Kiros J R and Hinton G E 2016 Layer normalization (arXiv:1607.06450)

Bahdanau D, Cho K H and Bengio Y 2015 Neural machine translation by jointly learning to align and translate *3rd Int. Conf. on Learning Representations, ICLR 2015 - Conf. Track Proc.* pp 1–15

Berger M, Coursey J and Zucker M 1999 ESTAR, PSTAR, and ASTAR: computer programs for calculating stopping-power and range tables for electrons, protons, and helium ions (version 1.21) (available at: http://physics.nist.gov/Star)

Bernstein D S, Amato C, Hansen E A and Zilberstein S 2009 Policy iteration for decentralized control of Markov decision processes *J. Artif. Intell. Res.* **34** 89–132

Bertsekas D P 1988 The auction algorithm: a distributed relaxation method for the assignment problem *Ann. Oper. Res.* **14** 105–23

Bethe H 1932 Bremsformel für Elektronen relativistischer Geschwindigkeit *Z. Phys.* **76** 293–9

Brun L, Gaüzére B, Renton G, Bougleux S and Yger F 2022 A differentiable approximation for the Linear Sum Assignment Problem with edn *2022 26th Int. Conf. on Pattern Recognition (ICPR)* pp 3822–8

Cohen A, Teng E, Berges V P, Dong R P, Henry H, Mattar M, Zook A and Ganguly S 2021 On the use and misuse of absorbing states in multi-agent reinforcement learning (arXiv:2111.05992v2)

Dalal G, Dvijotham K, Vecerik M, Hester T, Paduraru C and Tassa Y 2018 Safe exploration in continuous action spaces (arXiv:1801.08757)

Date K and Nagi R 2016 GPU-accelerated Hungarian algorithms for the Linear Assignment Problem *Parallel Comput.* **57** 52–72

Degrave J *et al* 2022 Magnetic control of tokamak plasmas through deep reinforcement learning *Nature* **602** 414–9

DeZoort G, Thais S, Duarte J, Razavimaleki V, Atkinson M, Ojalvo I, Neubauer M and Elmer P 2021 Charged particle tracking via edge-classifying interaction networks *Comput. Softw. Big Sci.* **5** 1–13

ElSayed-Aly I, Bharadwaj S, Amato C, Ehlers R, Topcu U and Feng L 2021 Safe multi-agent reinforcement learning via shielding *Proc. Int. Joint Conf. on Autonomous Agents and Multiagent Systems, AAMAS* vol 1 (International Foundation for Autonomous Agents) pp 483–91

Fard M M, Cormier Q, Canini K and Gupta M 2016 Launch and Iterate: Reducing Prediction Churn *Advances in Neural Information Processing Systems* vol 29

Foerster J N, Assael Y M, Freitas N and Whiteson S 2016 Learning to communicate with Deep multi-agent reinforcement learning *Proc. 30th Int. Conf. on Neural Information Processing Systems* (Curran Associates Inc.) pp 2145–53

Fortunato M *et al* 2018 Noisy networks for exploration *6th Int. Conf. on Learning Representations, ICLR 2018 - Conf. Track Proc.* pp 1–21

Frühwirth R 1987 Application of Kalman filtering to track and vertex fitting *Nucl. Instrum. Methods Phys. Res. A* **262** 444–50

Gottschalk B 2018 Radiotherapy proton interactions in matter (arXiv: 1804.00022)

Gronauer S and Diepold K 2022 Multi-agent deep reinforcement learning: a survey *Artif. Intell. Rev.* **55** 895–943

Groom D and Klein S 2000 Passage of particles through matter *Eur. Phys. J. C* **15** 163–73

Gu S, Holly E, Lillicrap T and Levine S 2017 Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates *Proc. - IEEE Int. Conf. on Robotics and Automation* pp 3389–96

Han S, Pool J, Tran J and Dally W J 2015 Learning both weights and connections for efficient neural networks *Proc. 29th Int. Conf. on Neural Information Processing Systems - Volume 1* (MIT Press) pp 1135–43

He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* pp 770–8

Highland V L 1975 Some practical remarks on multiple scattering *Nucl. Instrum. Methods* **129** 497–9

Hochreiter S and Schmidhuber J J 1994 Simplifying neural nets by discovering flat minima *Advances in Neural Information Processing Systems* (MIT Press) (https://proceedings.neurips.cc/paper_files/paper/1994/file/01882513d5fa7c329e940dda99b12147-paper.pdf)

Hu J, Jiang S, Harding S A, Wu H and Liao S 2021 Rethinking the implementation tricks and monotonicity constraint in cooperative multi-agent reinforcement learning (arXiv:2102.03479)

Iqbal S and Sha F 2019 Actor-attention-critic for multi-agent reinforcement learning *36th Int. Conf. on Machine Learning, ICML 2019* pp 5261–70

Jan S *et al* 2004 GATE -Geant4 Application for Tomographic Emission: a simulation toolkit for PET and SPECT *Phys. Med. Biol.* **49** 4543–61

Jan S *et al* 2011 GATE V6: a major enhancement of the GATE simulation platform enabling modelling of CT and radiotherapy *Phys. Med. Biol.* **56** 881–901

Jiang J and Lu Z 2018 Learning attentional communication for multi-agent cooperation *Proc. of the 32nd Int. Conf. on Neural Information Processing Systems* (Curran Associates Inc.) pp 7265–75

Jonker R and Volgenant A 1987 A shortest augmenting path algorithm for dense and sparse linear assignment problems *Computing* **38** 325–40

Joshi C K, Cappart Q, Rousseau L M and Laurent T 2021 Learning TSP requires rethinking generalization (available at: https://drops.dagstuhl.dedocumentLIPIcs.CP.2021.33)

Kaelbling L P, Littman M L and Cassandra A R 1998 Planning and acting in partially observable stochastic domains *Artif. Intell.* **101** 1–36

Kain V, Hirlander S, Goddard B, Velotti F M, Della Porta G Z, Bruchon N and Valentino G 2020 Sample-efficient reinforcement learning for CERN accelerator control *Phys. Rev. Accel. Beams* **23** 124801

Kawtikwar S and Nagi R 2024 HyLAC: Hybrid linear assignment solver in CUDA *J. Parallel Distrib. Comput.* **187** 104838

Kendall A, Hawke J, Janz D, Mazur P, Reda D, Allen J M, Lam V D, Bewley A and Shah A 2019 Learning to drive in a day *Proc. - IEEE Int. Conf. on Robotics and Automation* pp 8248–54

Kieseler J 2020 Object condensation: one-stage grid-free multi-object reconstruction in physics detectors, graph and image data *Eur. Phys. J. C* **80** 1–12

Klabunde M and Lemmerich F 2023 On the prediction instability of graph neural networks *Machine Learning and Knowledge Discovery in Databases* (*Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*) vol 13715 (Springer) pp 187–202

Klabunde M, Schumacher T, Strohmaier M and Lemmerich F 2023 Similarity of neural network models: a survey of functional and representational measures (arXiv:2305.06329v2)

Kortus T, Keidel R and Gauger N R 2023 Towards neural charged particle tracking in digital tracking calorimeters with reinforcement learning *IEEE Trans. Pattern Anal. Mach. Intell.* **45** 15820–33

Kortus T, Keidel R and Gauger N R 2025 Exploring end-to-end differentiable neural charged particle tracking - a loss landscape perspective *Trans. Mach. Learn. Res.* 1–34 (available at: https://openreview.net/forum?id=1Pi2GwduEz)

Kortus T, Schilling A, Keidel R and Gauger N R 2022 Particle tracking data: Bergen DTC prototype *Zenodo* (available at: https://doi.org/10.5281/zenodo.7426388)

Li H, Xu Z, Taylor G, Studer C and Goldstein T 2018 Visualizing the loss landscape of neural nets *Advances in Neural Information Processing Systems* pp 6389–99

Lieret K, DeZoort G, Chatterjee D, Park J, Miao S and Li P 2023 High pileup particle tracking with object condensation (arXiv:2312.03823v1)

Lillicrap T P, Hunt J J, Pritzel A, Heess N, Erez T, Tassa Y, Silver D and Wierstra D 2016 Continuous control with deep reinforcement learning *4th Int. Conf. on Learning Representations, ICLR 2016 - Conf. Track Proc.*

Littman M L 1994 Markov games as a framework for multi-agent reinforcement learning *Machine Learning Proc.* pp 157–63

Lowe R, Wu Y, Tamar A, Harb J, Abbeel P and Mordatch I 2017 Multi-agent actor-critic for mixed cooperative-competitive environments *Advances in Neural Information Processing Systems* pp 6380–91

Ma Y and Luo J 2022 Value-decomposition multi-agent proximal policy optimization *Proc. - 2022 Chinese Automation Congress, CAC 2022* (Institute of Electrical) pp 3460–4

Mager M 2016 ALPIDE, the monolithic active pixel sensor for the ALICE ITS upgrade *Nucl. Instrum. Methods Phys. Res. A* **824** 434–8

Mankel R 1997 A concurrent track evolution algorithm for pattern recognition in the HERA-B main tracking system *Nucl. Instrum. Methods Phys. Res. A* **395** 169–84

Mankel R and Spiridonov A 1999 The concurrent track evolution algorithm: extension for track finding in the inhomogeneous magnetic field of the HERA-B spectrometer *Nucl. Instrum. Methods Phys. Res. A* **426** 268–82

Mnih V, Kavukcuoglu K, Silver D, Graves A, Antonoglou I, Wierstra D and Riedmiller M 2013 Playing atari with deep reinforcement learning (arXiv:1312.5602)

Oliehoek F A and Amato C 2016 *A Concise Introduction to Decentralized Pomdps* ed G Tesauro, D Touretzky and T Leen (Springer)

Oliehoek F A, Spaan M T J and Vlassis N 2008 Optimal and approximate Q-value functions for decentralized POMDPs *J. Artif. Intell. Res.* **32** 289–353

Pettersen H E S *et al* 2021 Investigating particle track topology for range telescopes in particle radiography using convolutional neural networks *Acta Oncol.* **60** 1413–8

Pettersen H E S, Meric I, Odland O H, Shafiee H, Sølie J R and Röhrich D 2020 Proton tracking algorithm in a pixel-based range telescope for proton computed tomography (arXiv:2006.09751)

Pham T H, De Magistris G and Tachibana R 2018 OptLayer - practical constrained optimization for deep reinforcement learning in the real world *Proc. - IEEE Int. Conf. on Robotics and Automation* pp 6236–43

Plappert M, Houthooft R, Dhariwal P, Sidor S, Chen R Y, Chen X, Asfour T, Abbeel P and Andrychowicz M 2018 Parameter space noise for exploration *6th Int. Conf. on Learning Representations, ICLR 2018 - Conf. Track Proc.* pp 1–18

Pusztaszeri J-F, Rensing P E and Liebling T M 1996 Tracking elementary particles near their primary vertex: a combinatorial approach *J. Glob. Optim.* **9** 41–64

Rolínek M, Musil V, Paulus A, Vlastelica M, Michaelis C and Martius G 2020 Optimizing rank-based metrics with blackbox differentiation *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* pp 7617–27

Rolínek M, Swoboda P, Zietlow D, Paulus A, Musil V and Martius G 2020 Deep graph matching via blackbox differentiation of combinatorial solvers *Deep Graph Matching via Blackbox Differentiation of Combinatorial Solvers* (*Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*) pp 407–24

Sahoo S, Paulus A, Vlastelica M, Musil V, Kuleshov V and Martius G 2023 Backpropagation through combinatorial algorithms: identity with projection works *Proc. 11th Int. Conf. on Learning Representations*

Schulman J, Moritz P, Levine S, Jordan M I and Abbeel P 2016 High-dimensional continuous control using generalized advantage estimation *4th Int. Conf. on Learning Representations, ICLR 2016 - Conf. Track Proc.* pp 1–14

Schulman J, Wolski F, Dhariwal P, Radford A and Klimov O 2017 Proximal policy optimization algorithms (arXiv:1707.06347)

Sheebaelhamd Z, Zisis K, Nisioti A, Gkouletsos D, Pavllo D and Kohler J 2021 Safe deep reinforcement learning for multi-agent systems with continuous action spaces (arXiv:2108.03952)

Silver D *et al* 2018 A general reinforcement learning algorithm that masters chess, shogi and Go through self-play *Science* **362** 1140–4

Sullivan R, Terry J K, Black B and Dickerson J P 2022 Cliff diving: exploring reward surfaces in reinforcement learning environments *Proc. 39th Int. Conf. on Machine Learning* pp 20744–76

Sunehag P *et al* 2018 Value-decomposition networks for cooperative multi-agent learning based on team reward *Proc. Int. Joint Conf. on Autonomous Agents and Multiagent Systems, AAMAS* vol 3 pp 2085–7

Sutton R S and Barto A G 2018 *Reinforcement Learning: An Introduction* (A Bradford Book)

Tan M 1993 Multi-agent reinforcement learning: independent vs. cooperative agents *Machine Learning Proc.* pp 330–7

Vage L H 2022 Reinforcement learning for charged-particle tracking reinforcement learning *Proc. of the CTD 2022*

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N and Polosukhin I 2017 Attention is all you need *Advances in Neural Information Processing Systems* pp 5999–6009

Vinod A P, Safaoui S, Chakrabarty A, Quirynen R, Yoshikawa N and Di Cairano S 2022 Safe multi-agent motion planning via filtered reinforcement learning *2022 Int. Conf. on Robotics and Automation (ICRA)* pp 7270–6

Vinod A P, Safaoui S, Summers T H, Yoshikawa N and Di Cairano S 2024 Decentralized, safe, multiagent motion planning for drones under uncertainty via filtered reinforcement learning *IEEE Trans. Control Syst. Technol.* **32** 2492–9

Vinyals O, Fortunato M and Jaitly N 2015 Pointer networks *Advances in Neural Information Processing Systems* pp 2692–700

Vlastelica M, Paulus A, Musil V, Martius G and Rolínek M 2020 Differentiation of blackbox combinatorial solvers *8th Int. Conf. on Learning Representations, ICLR 2020* pp 1–19

Volgenant A 1996 Linear and semi-assignment problems: a core oriented approach *Comput. Oper. Res.* **23** 917–32

Welch B L 1947 The generalisation of student's problems when several different population variances are involved *Biometrika* **34** 28–35

Yu C, Velu A, Vinitsky E, Gao J, Wang Y, Bayen A and Wu Y 2022 The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games *Advances in Neural Information Processing Systems* vol 35, ed S Koyejo, S Mohamed, A Agarwal, D Belgrave, K Cho and A Oh (Curran Associates, Inc.) pp 24611–24