Reinforcement Learning Algorithms for Charged Particle Tracking with Applications in Proton Computed Tomography

Tobias Kortus [1] Ralf Keidel [1] Nicolas R. Gauger [2]
on behalf of the *Bergen pCT Collaboration*
[1]University of Applied Sciences Worms [2]University of Kaiserslautern-Landau (RPTU)

Proton Computed Tomography and Particle Tracking

- **Photon-based radiation**: High entrance doses with exponential decreasing doses with increasing penetrating depth.
- **Proton-based radiation**: Lower entrance doses with high energy deposition at *Bragg peak* → minimize damage to healthy tissue.
- Accurate treatment planning is essential to avoid damaging healthy tissue.
- **Proton Computed Tomography**: Direct estimation of relative stopping power using protons (higher initial energy) → requires measurements of proton trajectories and energy (estimated using energy depositions of trajectory over detector layers).
- **Particle tracking:** Identification of distinct set of hits in discrete detector readouts over multiple layers corresponding to the same particle → in the following: simulations of the *Bergen pCT DTC prototype* [1].

Reinforcement Learning for Particle Tracking [2]

- **Goal**: Find an optimal (or nearly-optimal) reconstruction policy π^* by repeatedly interacting with the constructed environment (maximizing long-term rewards).
- **Policy**: Decision strategy of the agent for each given state $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$.
- **Reward**: Likelihood of track segment under the theory of multiple Coulomb scattering (MCS).
- **Value-function**: Estimate of the expected future reward of a current state → used for bootstrapping to reduce variance of sampled track candidates.

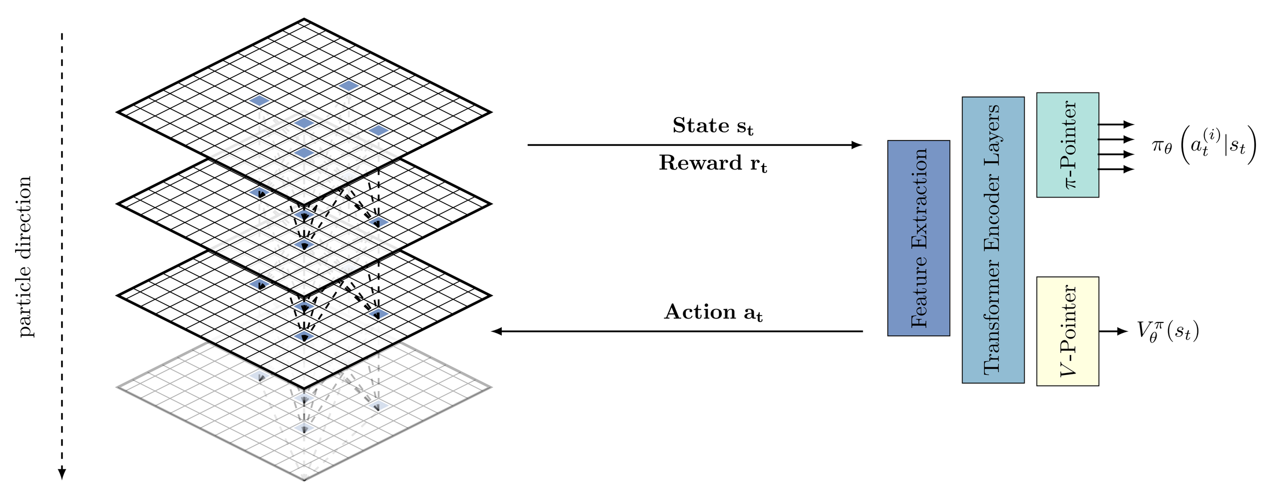

Figure 1. Interaction loop between environment containing particle readouts and a reconstruction agent.

Graph Construction

- Track hypotheses as a directed acyclic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with:
  - $v \in \mathcal{V}$: Particle hits in the detector.
  - $e \in \mathcal{E}$ Possible segments connecting two hits in adjacent layers (reversed → backward tracking).
- Edge and node features $(\vec{v}, \vec{e})$: $\vec{v}_i = (\Delta E, x_i, y_i, z_i)$ $\vec{e_{ij}} = (r_{ij}, \theta_{ij}, \phi_{ij})$

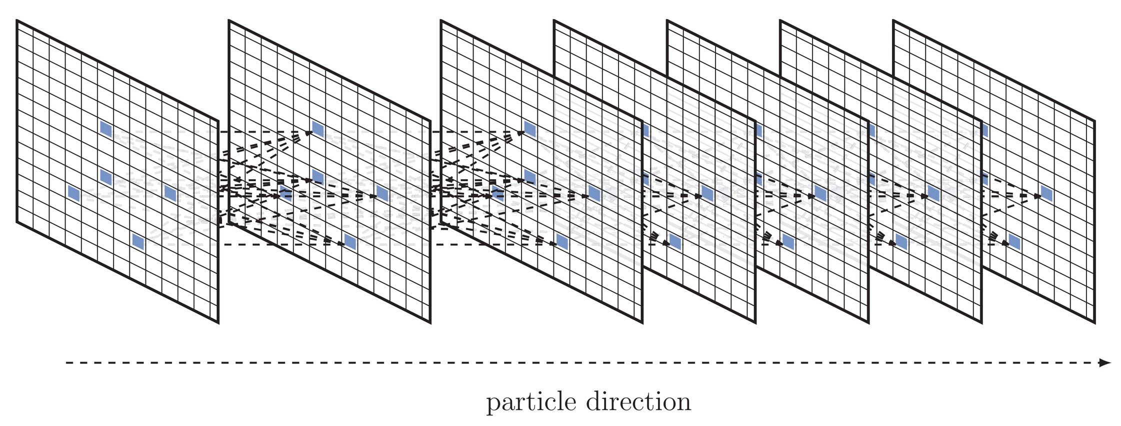

Figure 2. Schematic representation of a fully connected hit graph with possible track segments.

Network Architecture

- **Node embedding** Updated node representation by aggregating information over a multi-hop neighborhood using a Graph Neural Network (GNN).
- **Dynamic positional encoding:** Encoding of positional information using cosine-similarities (last → current segments) with dynamic focusing of area of attention.
- **Action encoder**: Exchange of information between action candidates.
- **Action decoder**: Estimation of likelihood of taking an action/ value of a state by correlating preprocessed action representation and observation representation [3].

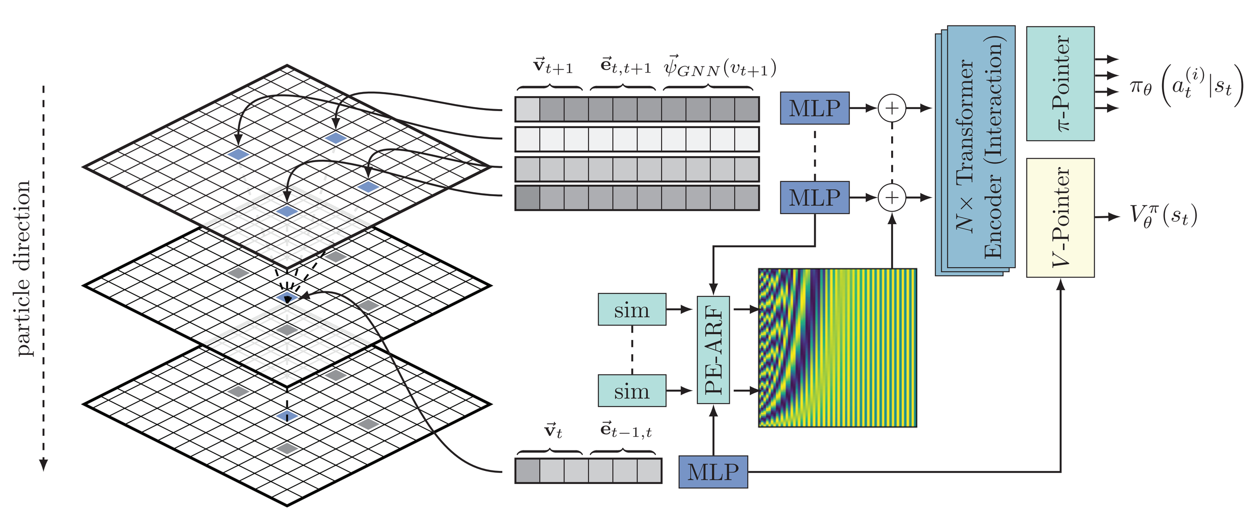

Figure 3. Shared network architecture for policy and state-value estimates.

- **Optimization:** Iterative optimization of policy/value using Proximal Policy Optimization (PPO) [4] → reward for each interaction based on likelihood of observed scattering angle.

Limitations of Single-Agent Reinforcement Learning

- **Partial observability:** Reconstruction w.r.t. entire readout frame remains still partial observable (other tracks are not taken into consideration).
- **Ambiguities in assignments:** Conflicts in reconstruction can assign the same particle to multiple tracks → generation of implausible tracks.

Multi-Agent Reinforcement Learning (Work in Progress)

Design considerations for training Multi-agent RL (MARL) agents for particle tracking:

1. **Dec-POMDP:** Consider multiple decentralized agents (similar to single-agent) with only local observations per agent and limited communication → minimal performance impact by avoiding global information or complex communication protocols.
2. **CTDE:** Use additional information during training that is unavailable during inference (centralized critic) → better training performance (reduced instationarity of the whole system).
3. **Constraints**. Enforce agreement between agents → unique particle assignment (constraint satisfaction by designing a safety layer [5]). → deterministic output policy μ' motivates use of Multi-agent Deep Deterministic Policy Gradient (MADDPG) [6].

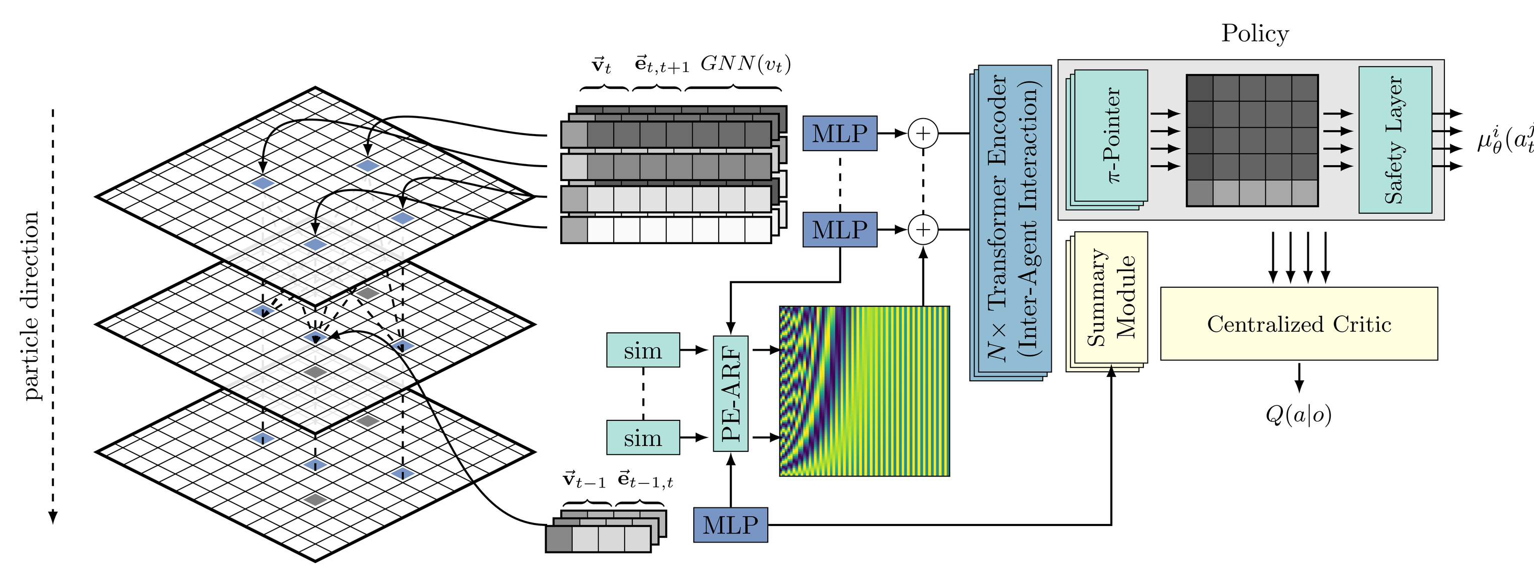

Figure 4. MARL architecture for training of decentralized agents with safety layer using centralized critic.

LSA Safety Policy Layer

Unique particle hit assignment (with constraint satisfaction guarantee) using linear sum assignment (LSA) with imperfect matching:

$$\min_{M'} \sum_{(i,j)\in\mathcal{E}} \mu'_{ij} w_{ij}, \quad \text{where} \quad \mu'_{ij} \in \{0, 1\} \tag{1}$$

where $\mu' = \{\mu'_1, \dots \mu'_N\}$ is the inferred safe policy by solving the LSA problem using the weight matrix $\mathcal{W} = {w_{ij}}^{M\times N}$ (distance between local policy and possible action).

Blackbox Differentiation: Gradients are either zero or infinity (piece-wise constant function of linear sum assignment solver) → implicitly interpolated gradients using [7].

(Preliminary) Results

- Evaluation of reconstruction performance on Monte-Carlo simulations using GATE software toolkit [8].
- Simulation of multiple phantom geometries using water cubes with various thicknesses (100–200 mm).

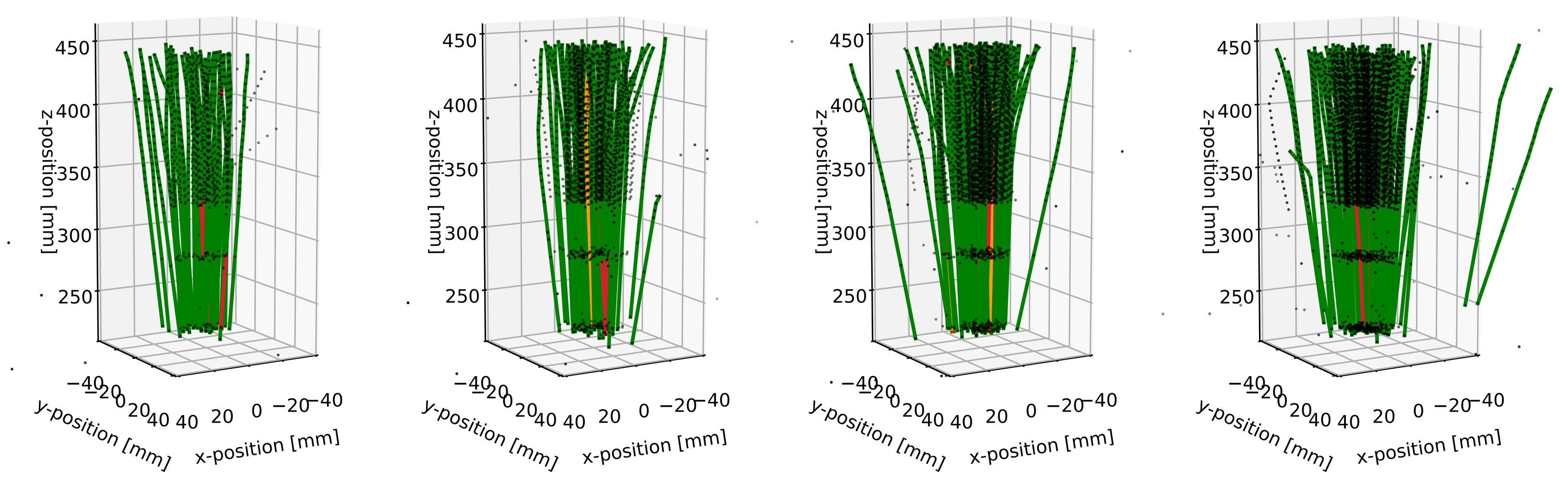

Figure 5. Reconstructed tracks using single-agent RL algorithm with 50, 100, 150 and 200 tracks per frame.

- Purities (p) and efficiencies (ϵ) (after filtering tracks with high angle scattering and low energies in final layer) → in depth evaluation of single-agent RL available in [2].

Table 1. Reconstruction performance, measured in terms of purity p and efficiency ϵ for different configurations. Results marked with * are cherry-picked runs due to instability of training. Results for *Search* and *PPO* are taken from [2].

		100 mm Water		150 mm Water		200 mm Water	
Density	Algorithm	p [%]	ϵ [%]	p [%]	ϵ [%]	p [%]	ϵ [%]
100	Search [9]	83.0±0.0	74.6±0.0	86.5±0.0	79.0±0.0	87.4±0.0	80.3±0.0
	PPO [2]	85.6±0.3	75.2±0.5	88.8±0.5	79.0±0.5	89.5±0.4	80.8±0.5
	MADDPG+LSA*	**90.8**±-.-	**75.7**±-.-	**92.8**±-.-	**79.1**±-.-	**93.1**±-.-	**81.0**±-.-
150	Search [9]	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 [2]	80.5±0.4	70.8±0.3	83.8±0.7	74.4±0.6	85.3±0.6	76.9±0.5
	MADDPG+LSA*	**87.3**±-.-	**71.0**±-.-	**89.1**±-.-	74.2±-.-	**90.7**±-.-	73.6±-.-
200	Search [9]	75.4±0.0	**67.4±0.0**	80.1±0.0	**72.9±0.5**	81.6±0.4	**75.0±0.0**
	PPO [2]	75.3±0.6	66.6±0.6	80.0±0.8	70.9±0.6	81.7±0.6	73.8±0.5
	MADDPG+LSA*	**83.0**±-.-	65.6±-.-	**86.9**±-.-	71.1±-.-	**87.7**±-.-	73.6±-.-

References

[1] J. Alme *et al.*, "A High-Granularity Digital Tracking Calorimeter Optimized for Proton CT," *Frontiers in Physics*, vol. 8, no. October, pp. 1–20, 2020.
[2] T. Kortus *et al.*, "Towards neural charged particle tracking in digital tracking calorimeters with reinforcement learning," *IEEE TPAMI*, pp. 1–14, 2023.
[3] O. Vinyals *et al.*, "Pointer networks," *Advances in Neural Information Processing Systems*, vol. 2015-Janua, pp. 2692–2700, 2015.
[4] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," pp. 1–12, 2017.
[5] G. Dalal *et al.*, "Safe Exploration in Continuous Action Spaces," 2018.
[6] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 6380–6391, 2017.
[7] M. Vlastelica *et al.*, "Differentiation of Blackbox Combinatorial Solvers," *8th International Conference on Learning Representations, ICLR 2020*, pp. 1–19, 2020.
[8] S. Jan *et al.*, "GATE -Geant4 Application for Tomographic Emission: a simulation toolkit for PET and SPECT," *Phys Med Biol. Phys Med Biol*, vol. 49, no. 19, pp. 4543–4561, 2004.
[9] H. E. Pettersen *et al.*, "Proton tracking algorithm in a pixel-based range telescope for proton computed tomography," *arXiv*, 2020.
[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 2017-Decem, pp. 5999–6009, 2017.