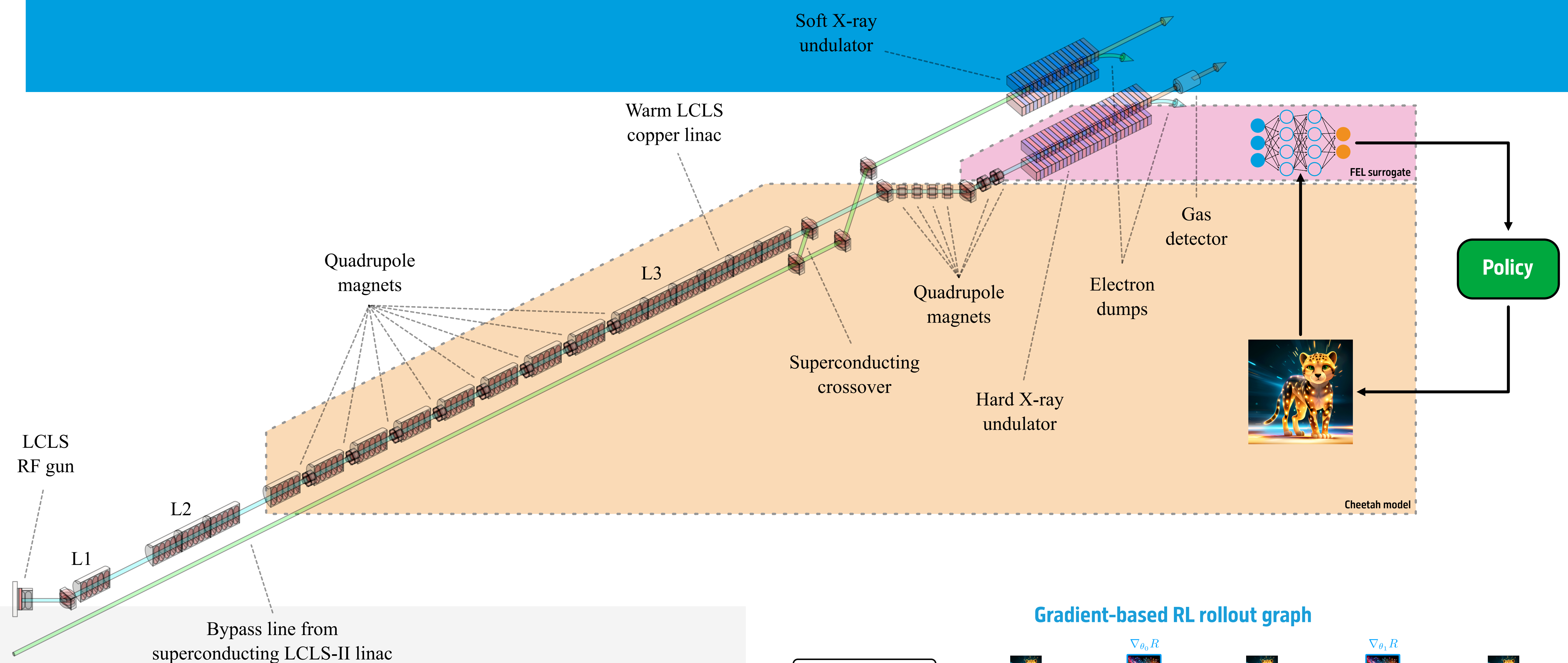


# The fast differentiable beam dynamics simulator **Cheetah** coupled with neural network surrogate models enables **gradient-based RL** in **minutes** on complex high-dimensional tasks like FEL tuning.



## High-Speed Differentiable Simulations as an Enabler for Reinforcement Learning-based FEL Tuning Agents.

J. Kaiser\*, A. Eichler, J. Morgan, Z. Zhu, R. Roussel, A. Edelen, D. Ratner, M. Schram, K. Rajput

### Motivation

- Tuning and control tasks at accelerators can be very complex, with high-dimensional input spaces, complex physics and sparse objective signals, which make them difficult to solve with conventional methods.
- Reinforcement learning (RL) can train policies capable of solving such difficult tasks, but **RL training is notoriously sample-inefficient**, which can result in intractable training times.
  - We consider as one such task a setup, where **14 quadrupoles are tuned to maximise the FEL pulse intensity at LCLS**.
  - This task is performed frequently as the FEL has to switch between energies and phase space shapes.
  - Similar tasks need to be solved at **other facilities like European XFEL and FLASH**.
  - Training **on the real machine or with conventional simulators takes on the order of weeks to months**, and even with a high-speed simulator (not using gradients) it requires two days to train a good policy using the PPO algorithm.

### Gradient-based reinforcement learning using Cheetah

- The sample-inefficiency of most RL algorithms results from the need to estimate stable gradients from large sample sizes because environments are considered to be black boxes with unknown gradients.
- We developed **Cheetah**, a **high-speed differentiable beam dynamics simulator**. In addition to **orders of magnitude faster compute**, Cheetah can provide gradients of beam dynamics with **automatic differentiation**.
- Cheetah enables us to compute the exact gradient of the episodic return with respect to the policy parameters from just a few samples instead of having to collect an enormous amount of samples to smooth out a noisy estimate.

### AI/ML coupling to include complex dynamics

- Some physical effects like the FEL process are **inherently expensive** to compute, and remain too time-consuming even when implementations are extensively optimised.
- Here, **neural network surrogate models** and **Cheetah's native integration with PyTorch** enable the intuitive inclusion of such effects in gradient-based RL training, while **preserving both the computational speed and the ability to compute gradients**.

### Results

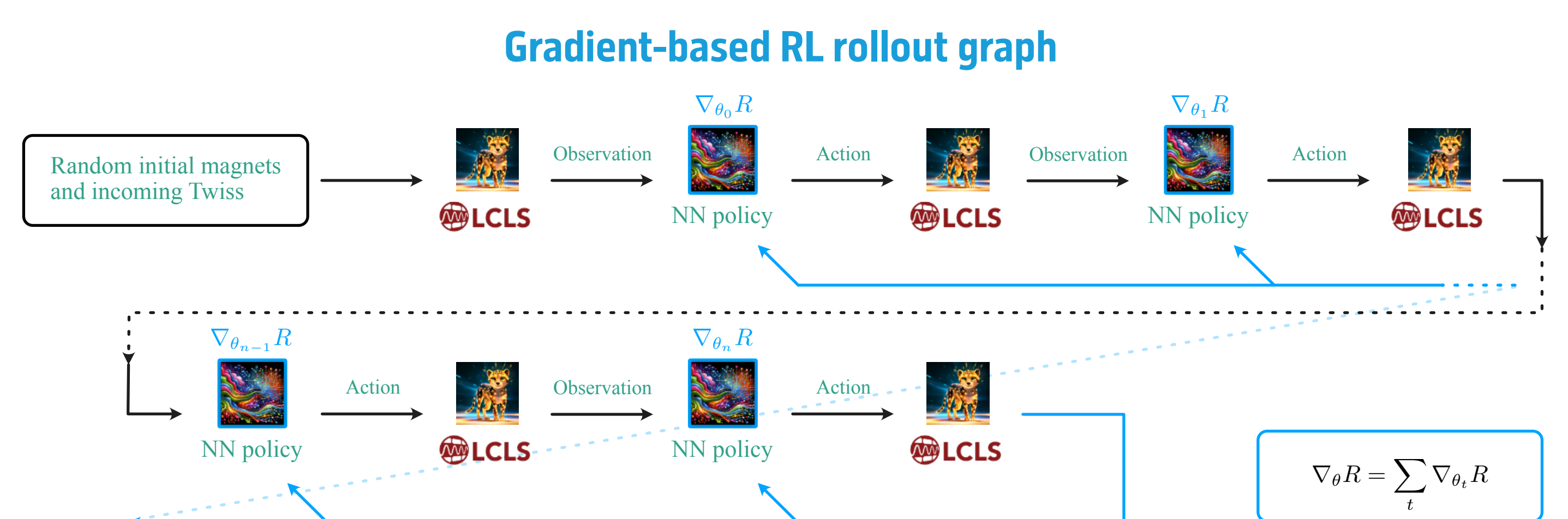
- Our implementation of gradient-based policy optimisation using Cheetah and *PyTorch Lightning* reaches the same reward threshold as PPO in at least **45x fewer samples**. The resulting policy performs just as well.
- The result is the ability to train an RL policy on a complex high-dimensional FEL intensity tuning task **in a matter of minutes**.

\*Corresponding author: jan.kaiser@desy.de

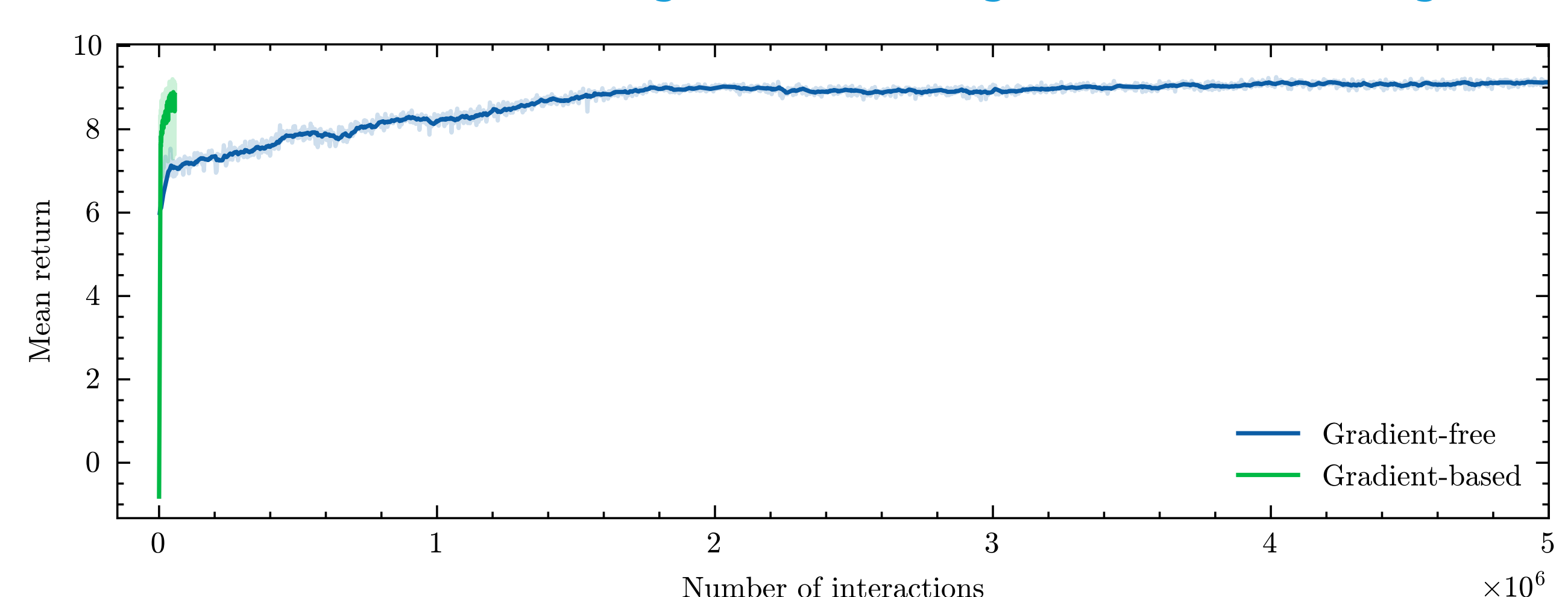
Deutsches Elektronen-Synchrotron DESY  
A Research Centre of the Helmholtz Association



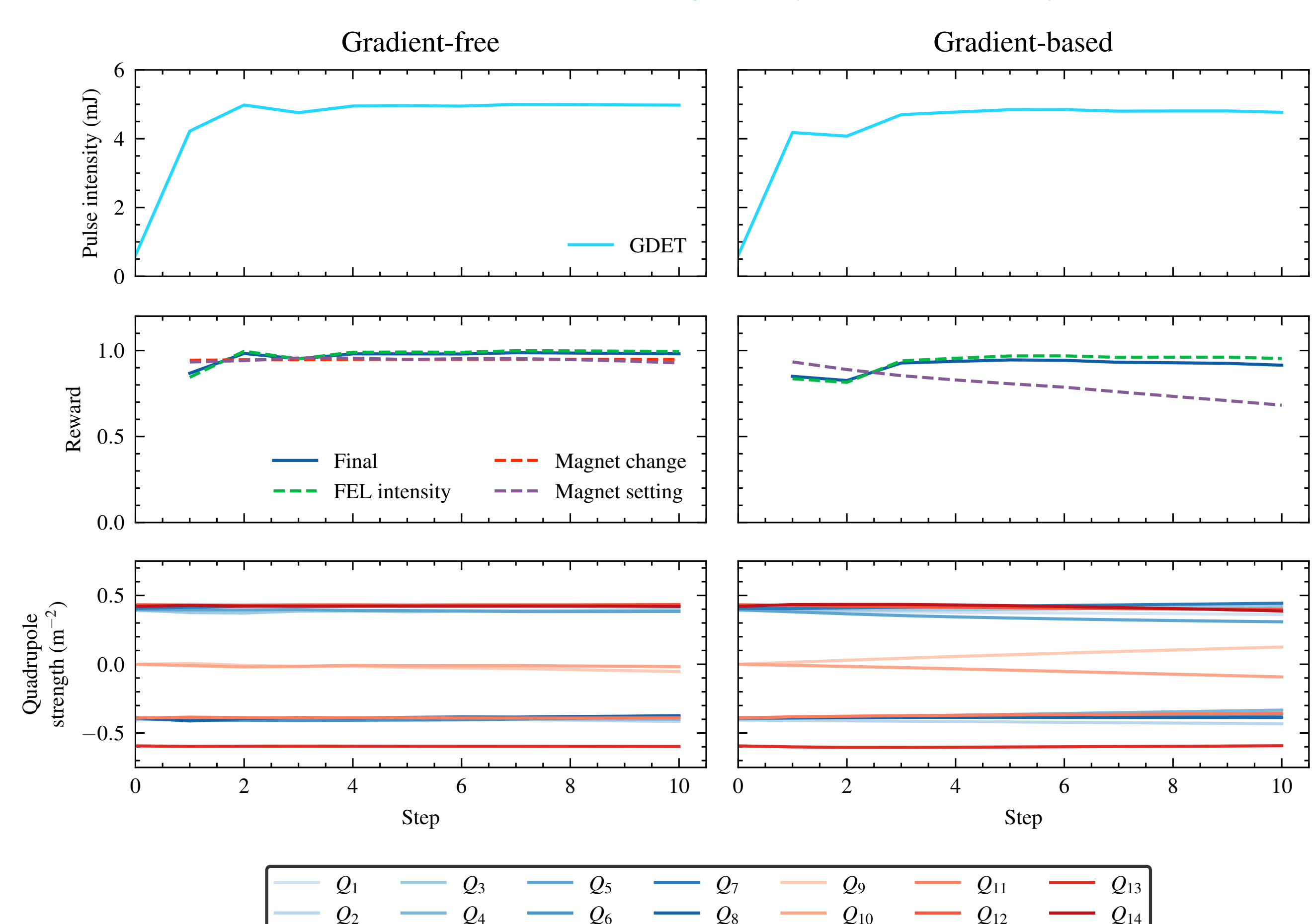
Scan me to download the poster!



### Mean return curves of gradient-free vs. gradient-based training



### Example FEL tuning run by a trained policy



HIR<sup>3</sup>X

Jefferson Lab

SLAC

NATIONAL ACCELERATOR LABORATORY

