Karlsruhe Institute of Technology

Institute for Data Processing and Electronics

Real-time Reinforcement Learning on FPGA with Online Training for Autonomous Accelerators

Luca Scomparin, Jürgen Becker, Edmund Blomley, Erik Bründermann, Michele Caselle, Timo Dritschler, Andreas Kopmann, Anke-Susanne Müller, Andrea Santamaria Garcia, Johannes L. Steinmann, Chenran Xu | 5-8 March 2024

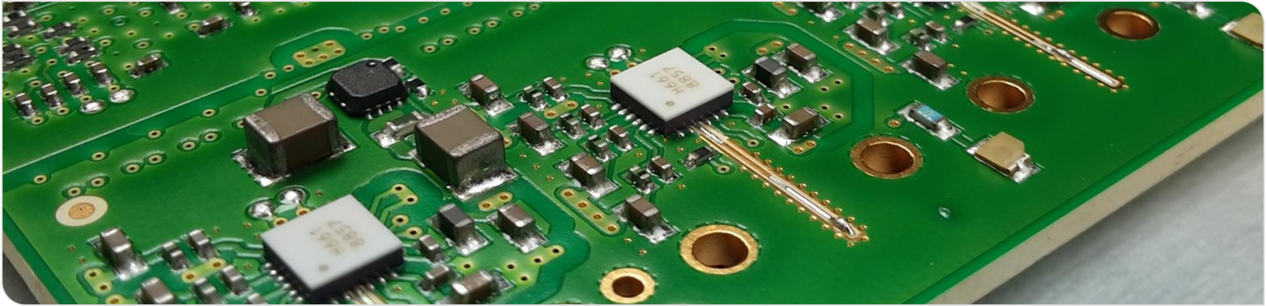
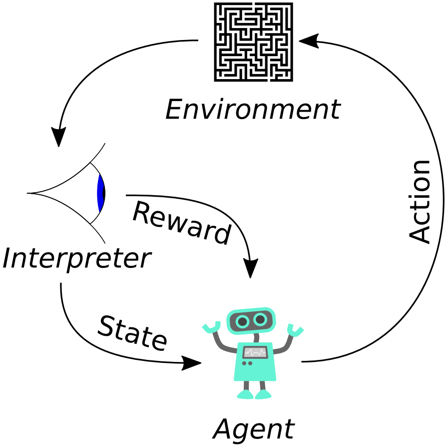
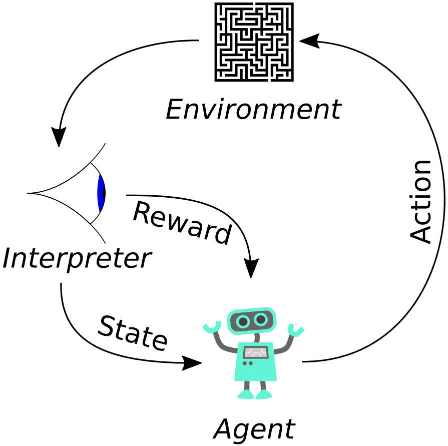


Table of contents

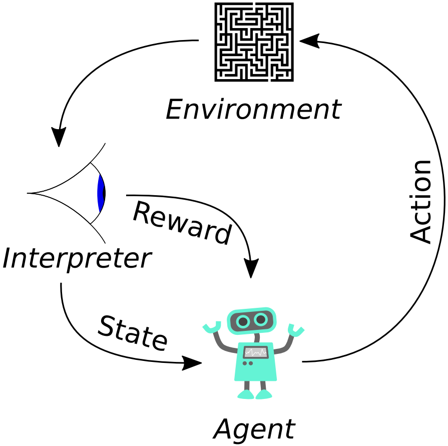

Environment
Interpreter
Reward
Action
State
Agent

Motivation



Usually data-hungry

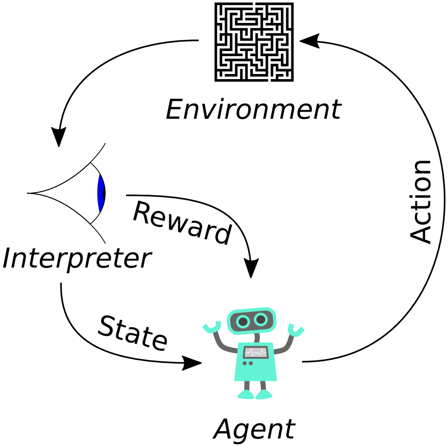
Motivation



Usually data-hungry

Great data rate $\rightarrow$ lot of training data

Motivation

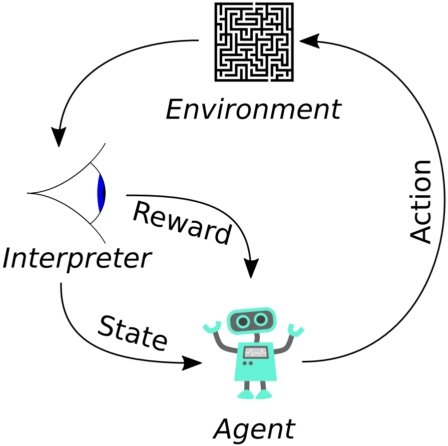


Usually data-hungry

Great data rate $\rightarrow$ lot of training data

Possibility of **training online**

Motivation



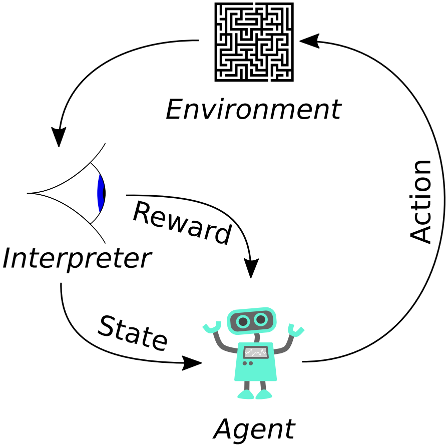
Usually data-hungry

Great data rate $\rightarrow$ lot of training data

Possibility of **training online**

No simulation required!

Motivation



Usually data-hungry

Great data rate $\rightarrow$ lot of training data

Possibility of **training online**

No simulation required!

Timing constrains become relevant!

What is Real-Time?

Shin and Ramanathan (1994) identify major components:

> *Correctness of a computation depends not only on the logical correctness but also on the time at which the results are produced.*

1. "time" is the most precious resource;
2. reliability is crucial;
3. **environment of operation** is an active component.

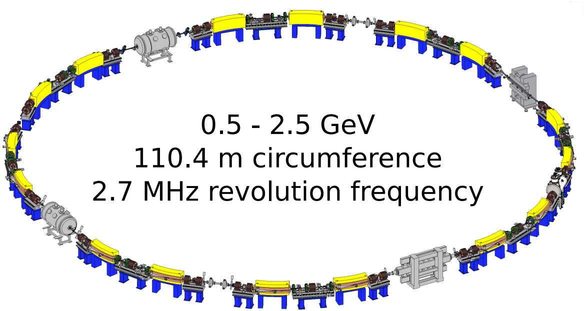
Predictability is fundamental!

Three possible levels/categories:

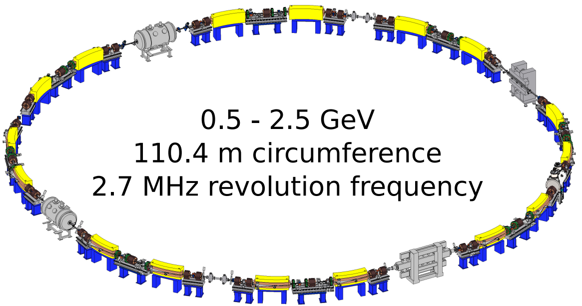
1. *hard*, catastrophic consequences;
2. *firm*, results produced late not useful;
3. *soft*, later means decreasing usefulness.

Depending on environment, RL can be either one of these!

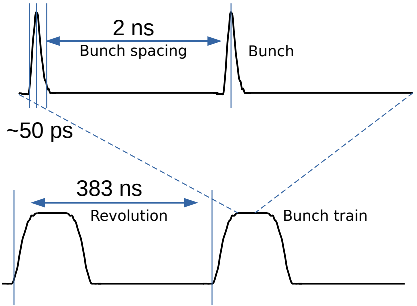
Case study: KARA



Case study: KARA

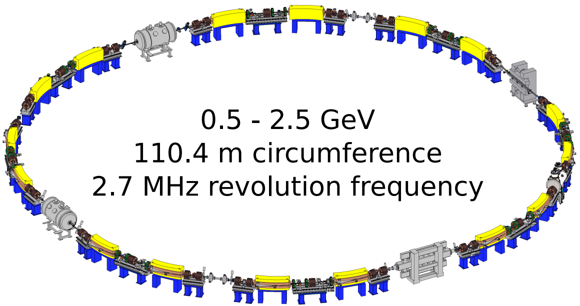


Example signal

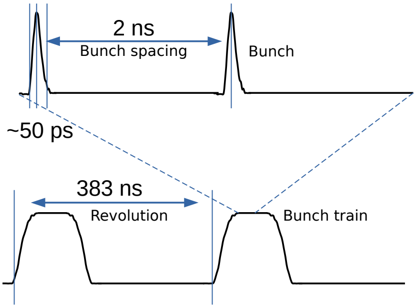


Synchrotron frequency/period $O(10\,\text{kHz} \leftrightarrow 100\,\mu\text{s})$

Case study: KARA



Example signal

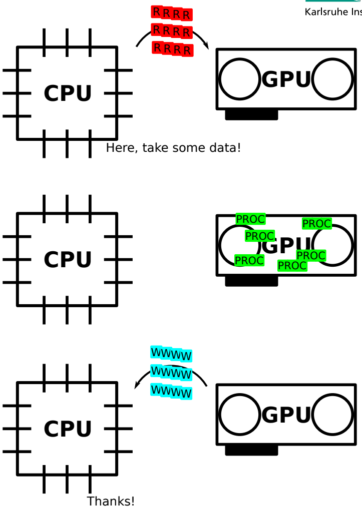


Synchrotron frequency/period $O(10\,\mathrm{kHz} \leftrightarrow 100\,\mu s)$

Huge variety of time scales!

Issues of Real-Time AI

- Current ML frameworks have mainly throughput in mind → no/little real-time optimization;
- use of batched execution on GPU → not optimal for latency;
- conventional computing hardware not meant for low-latency real-time;
- it still works great for latency in the millisecond range!



Heterogeneous platforms

Karlsruhe Institute of Technology

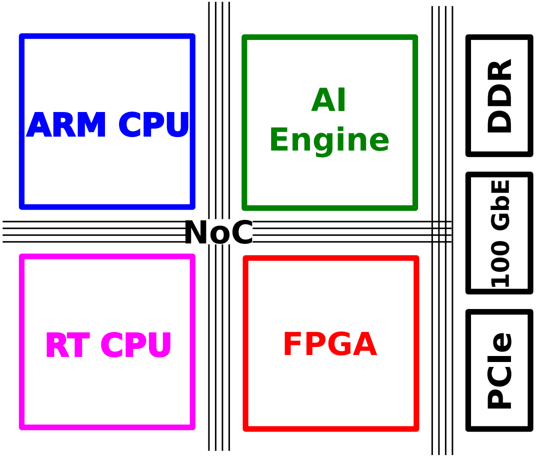
Different computing platforms → different benefits

Heterogeneous combine CPUs, FPGAs and "GPUs"

An example, AMD Versal:

- combines FPGAs and ARM CPUs;
- AI Engine array for heavy multiplication workloads;
- Network-on-Chip interconnect;
- high-speed interfaces.

These computation unit work in synergy and share memory!



The KINGFISHER RL platform

Experience accumulator

Real-Time inference BUT Offline/Batched training

The KINGFISHER RL platform

Experience accumulator

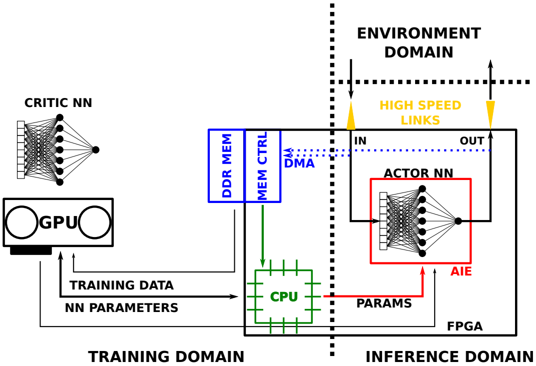
Real-Time inference BUT Offline/Batched training

Pros:

+ "easy" real-time;
+ can use complex training algorithms;
+ can use GPUs and other accelerators;
+ training time reward definition™.

Cons:

- data inefficient;
- actor design is critical;
- training overhead.



First test: betatron oscillations

Idea

Start simple $\rightarrow$ test all components together

First test: betatron oscillations

Idea

Start simple $\rightarrow$ test all components together

- Betatron oscillations ($\approx 700\,\mathrm{kHz}$)are well understood
- Easy to frame as Markov Decision Process
- Classical control for comparison

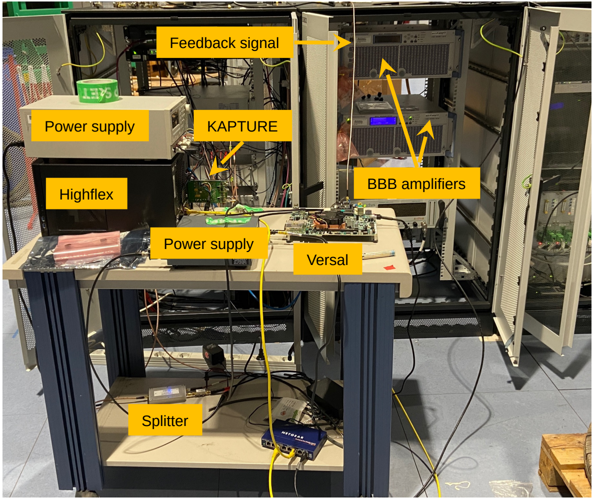
First test: betatron oscillations

Idea

Start simple → test all components together

- Betatron oscillations (≈ 700 kHz)are well understood
- Easy to frame as Markov Decision Process
- Classical control for comparison

Excite oscillation with kicker, damp it with RL!



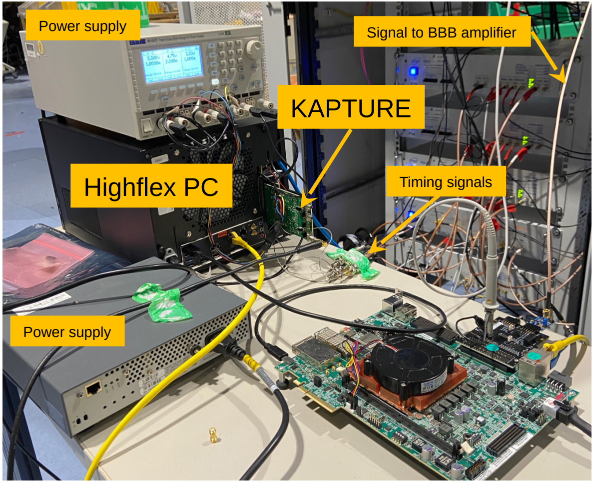
First test: betatron oscillations

Idea

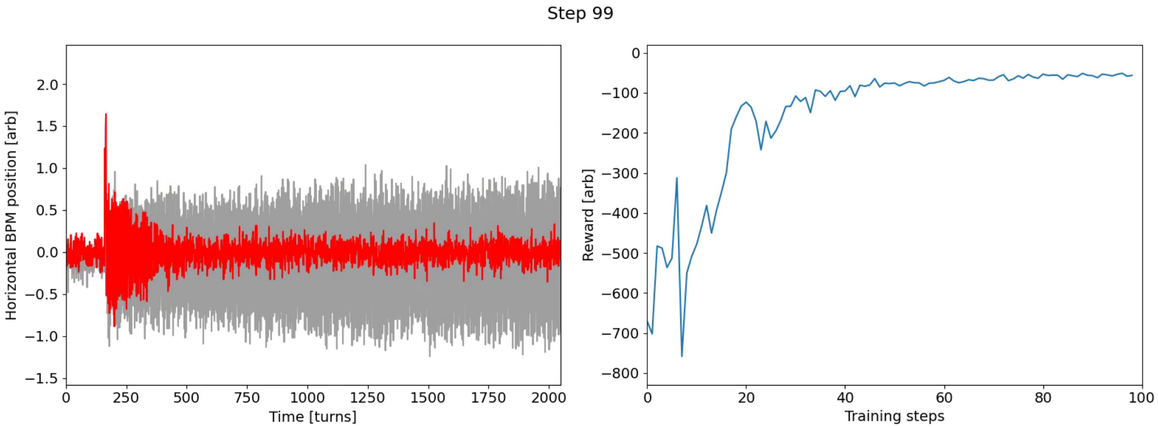
Start simple $\rightarrow$ test all components together

- Betatron oscillations ($\approx$ 700 kHz)are well understood
- Easy to frame as Markov Decision Process
- Classical control for comparison

Excite oscillation with kicker, damp it with RL!

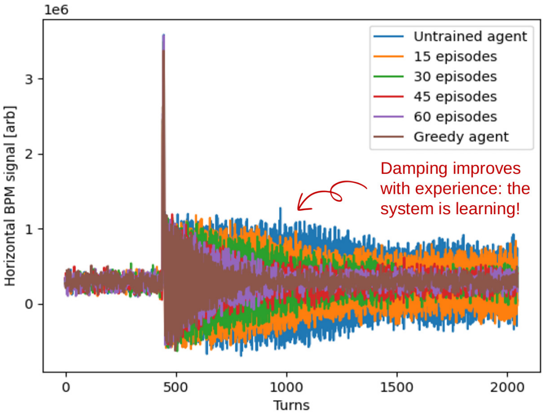


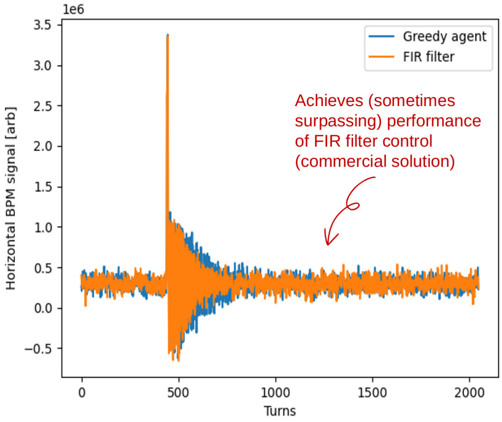
First test: betatron oscillations



Reward with L1 norm, L2 norm and tanh. 2.7 MHz action rate!

First test: betatron oscillations

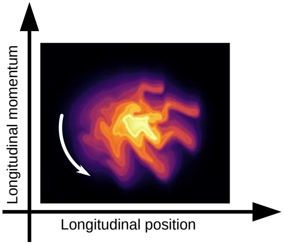




Hard problem: Microbunching Instability

Unstable coherent synchrotron radition (THz) production

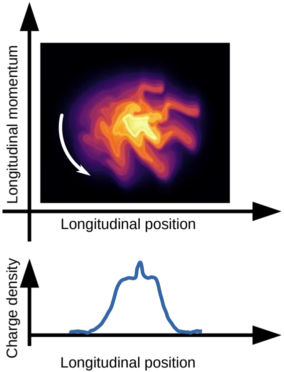
- Self-interaction of bunch with emitted radiation
- Nonlinear dynamics, several timescales/frequency components
- Main timescales: $O(10\,\mu s)$, $O(10\,\mathrm{ms})$, with $T_s = O(100\,\mu s)$
- Expensive to simulate!



Hard problem: Microbunching Instability

Unstable coherent synchrotron radition (THz) production

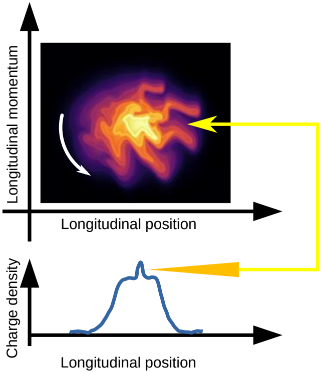
- Self-interaction of bunch with emitted radiation
- Nonlinear dynamics, several timescales/frequency components
- Main timescales: $O(10\,\mu s)$, $O(10\,ms)$, with $T_s = O(100\,\mu s)$
- Expensive to simulate!

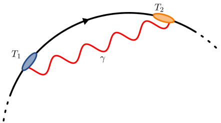


Hard problem: Microbunching Instability

Unstable coherent synchrotron radition (THz) production

- Self-interaction of bunch with emitted radiation
- Nonlinear dynamics, several timescales/frequency components
- Main timescales: $O(10\,\mu\mathrm{s})$, $O(10\,\mathrm{ms})$, with $T_s = O(100\,\mu\mathrm{s})$
- Expensive to simulate!

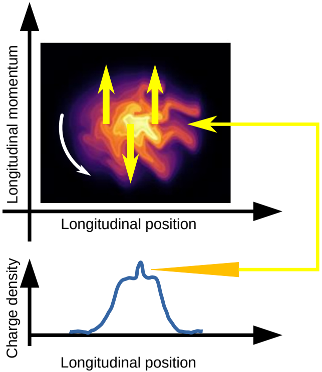


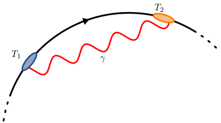


Hard problem: Microbunching Instability

Unstable coherent synchrotron radition (THz) production

- Self-interaction of bunch with emitted radiation
- Nonlinear dynamics, several timescales/frequency components
- Main timescales: $O(10\,\mu\text{s})$, $O(10\,\text{ms})$, with $T_s = O(100\,\mu\text{s})$
- Expensive to simulate!



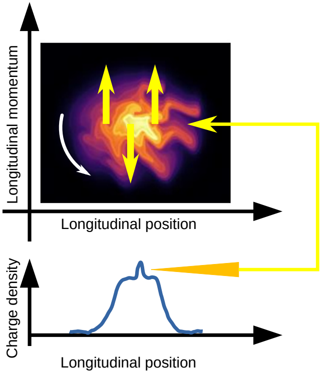


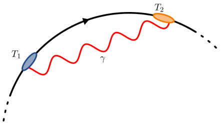
Hard problem: Microbunching Instability

Unstable coherent synchrotron radition (THz) production

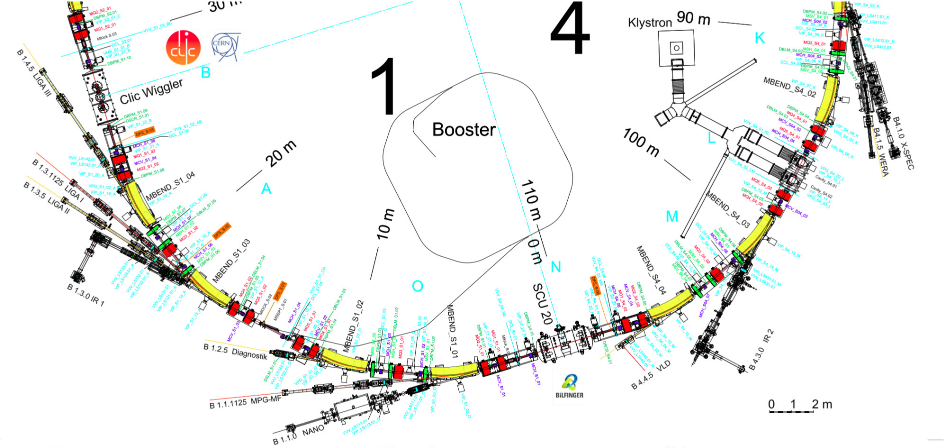
- Self-interaction of bunch with emitted radiation
- Nonlinear dynamics, several timescales/frequency components
- Main timescales: $O(10\,\mu s)$, $O(10\,ms)$, with $T_s = O(100\,\mu s)$
- Expensive to simulate!

Perfect candidate for real time RL!

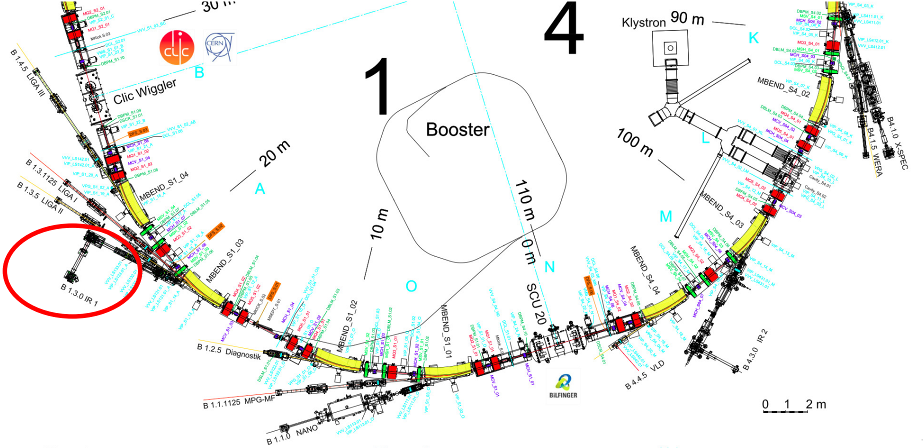




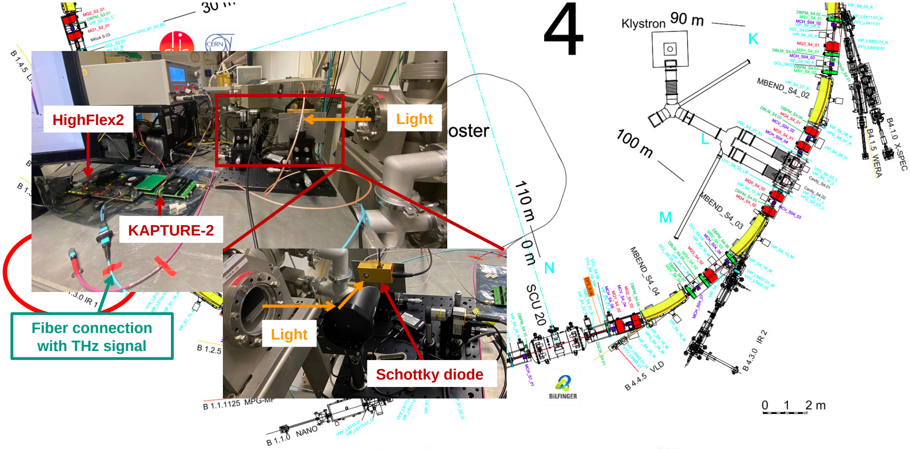
Hardware implementation



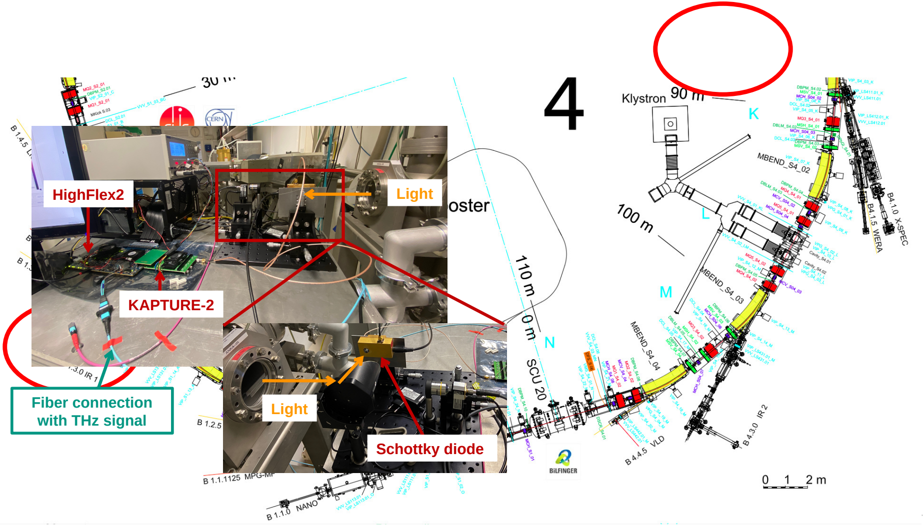
Hardware implementation



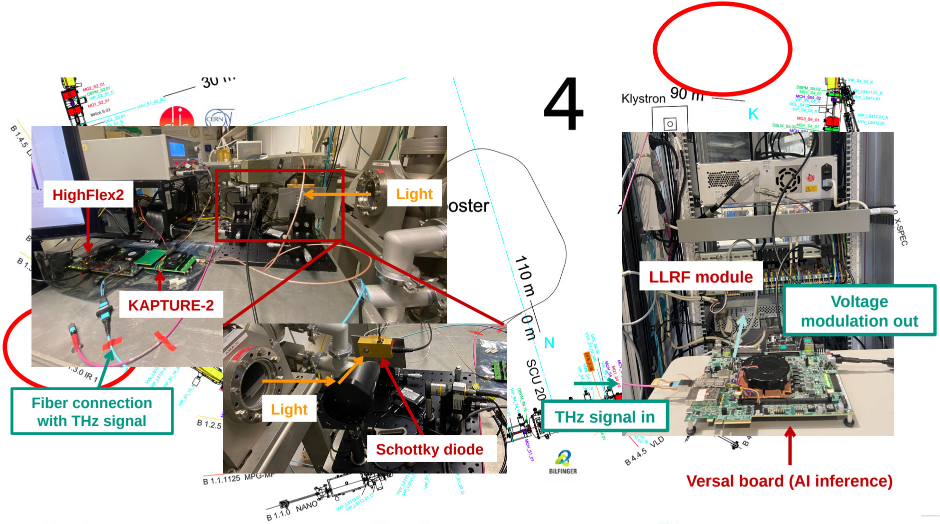
Hardware implementation



Hardware implementation



Hardware implementation



RL problem definition

Environment $\rightarrow x_i$ Coherent Sychrotron Radiation power each turn

Initial approach

$$O = \{\mu_{\text{CSR}}, \sigma_{\text{CSR}}, m_{\text{trend}}, A_{\text{FFT max}}, f_{\text{FFT max}}, \Delta_\theta\}$$

$$A = \{A_{\text{mod}}, f_{\text{mod}}\}$$

RL problem definition

Environment $\rightarrow$ x_i Coherent Sychrotron Radiation power each turn

Initial approach

$$O = \{\mu_{\text{CSR}}, \sigma_{\text{CSR}}, m_{\text{trend}}, A_{\text{FFT max}}, f_{\text{FFT max}}, \Delta_\theta\}$$

$$A = \{A_{\text{mod}}, f_{\text{mod}}\}$$

FFT and cross-correlation with $O(10\ \mu s)$ latency $\rightarrow$ hard on FPGA

RL problem definition

Karlsruhe Institute of Technology

Environment $\rightarrow x_i$ Coherent Sychrotron Radiation power each turn

Initial approach

$$O = \{\mu_{\text{CSR}}, \sigma_{\text{CSR}}, m_{\text{trend}}, A_{\text{FFT max}}, f_{\text{FFT max}}, \Delta_\theta\}$$

$$A = \{A_{\text{mod}}, f_{\text{mod}}\}$$

FFT and cross-correlation with $O(10\ \mu s)$ latency $\rightarrow$ hard on FPGA

New approach

$$y_i = \{\text{filtered and decimated } x_i\}$$

$$O = \{N \text{ latest } x_i\}$$

$$A = \text{action or delta-action}$$

RL problem definition

Environment $\rightarrow x_i$ Coherent Sychrotron Radiation power each turn

Initial approach

$$O = \{\mu_{\text{CSR}}, \sigma_{\text{CSR}}, m_{\text{trend}}, A_{\text{FFT max}}, f_{\text{FFT max}}, \Delta_\theta\}$$

$$A = \{A_{\text{mod}}, f_{\text{mod}}\}$$

FFT and cross-correlation with $O(10\ \mu s)$ latency $\rightarrow$ hard on FPGA

New approach

$$y_i = \{\text{filtered and decimated } x_i\}$$

$$O = \{N \text{ latest } x_i\}$$

$$A = \text{action or delta-action}$$

Hardware friendly! Decimation controls time perception

RL problem definition

Environment $\rightarrow x_i$ Coherent Sychrotron Radiation power each turn

Initial approach

$$O = \{\mu_{\text{CSR}}, \sigma_{\text{CSR}}, m_{\text{trend}}, A_{\text{FFT max}}, f_{\text{FFT max}}, \Delta_\theta\}$$

$$A = \{A_{\text{mod}}, f_{\text{mod}}\}$$

FFT and cross-correlation with $O(10\ \mu s)$ latency $\rightarrow$ hard on FPGA

New approach

$$y_i = \{\text{filtered and decimated } x_i\}$$

$$O = \{N \text{ latest } x_i\}$$

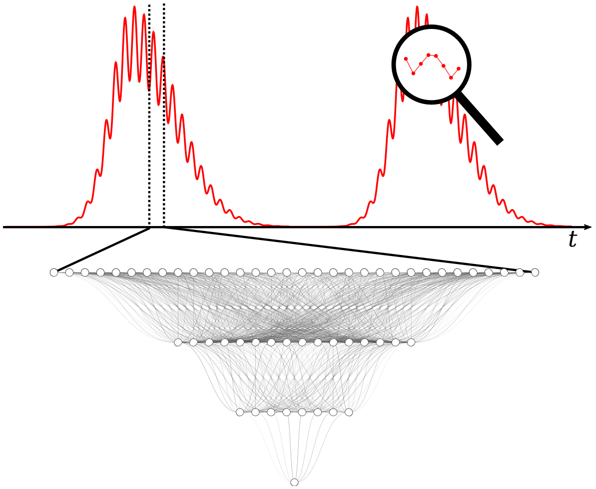
$$A = \text{action or delta-action}$$

Hardware friendly! Decimation controls time perception

Reward is observation based and varies at runtime

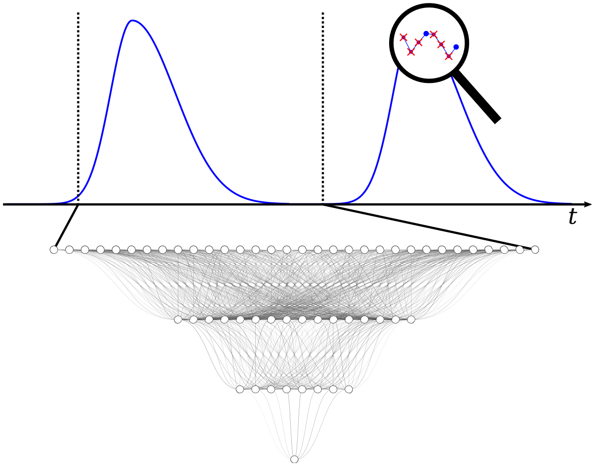
Agent design

- 64 samples input into network
- Sampling rate affects network time-span
- Filter signal $\rightarrow$ remove high-frequency components
- Drop samples $\rightarrow$ network "sees" more time



Agent design

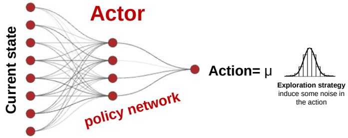
- 64 samples input into network
- Sampling rate affects network time-span
- Filter signal → remove high-frequency components
- Drop samples → network "sees" more time

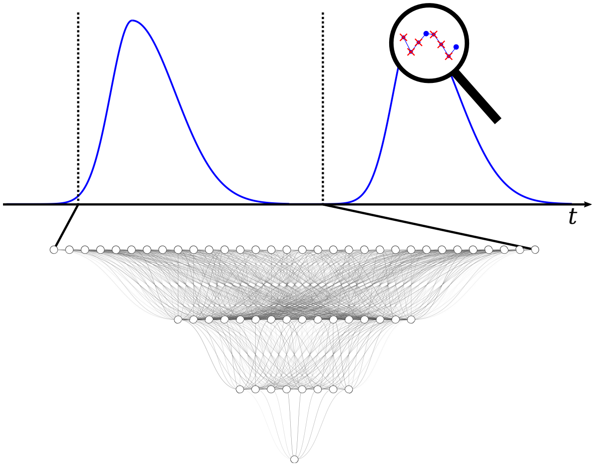


Agent design

- 64 samples input into network
- Sampling rate affects network time-span
- Filter signal $\rightarrow$ remove high-frequency components
- Drop samples $\rightarrow$ network "sees" more time

Agent output chooses mean of gauss distribution

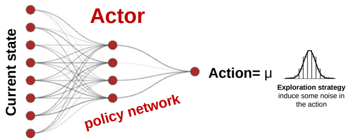


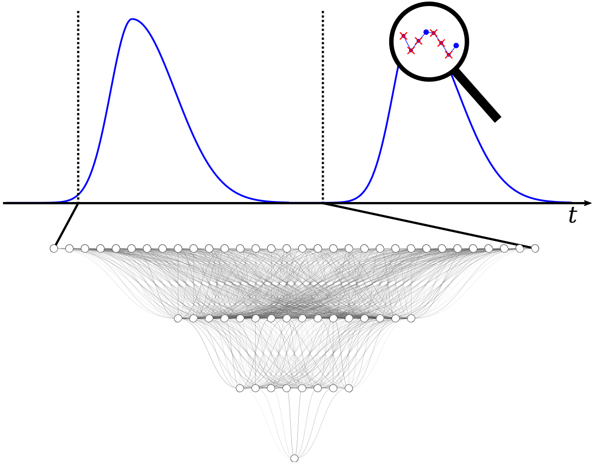


Agent design

- 64 samples input into network
- Sampling rate affects network time-span
- Filter signal → remove high-frequency components
- Drop samples → network "sees" more time

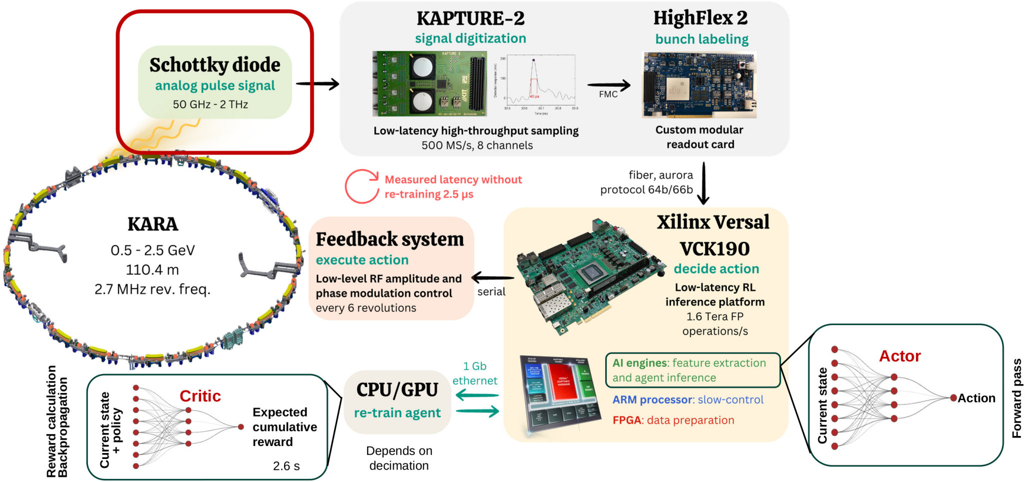
Agent output chooses mean of gauss distribution





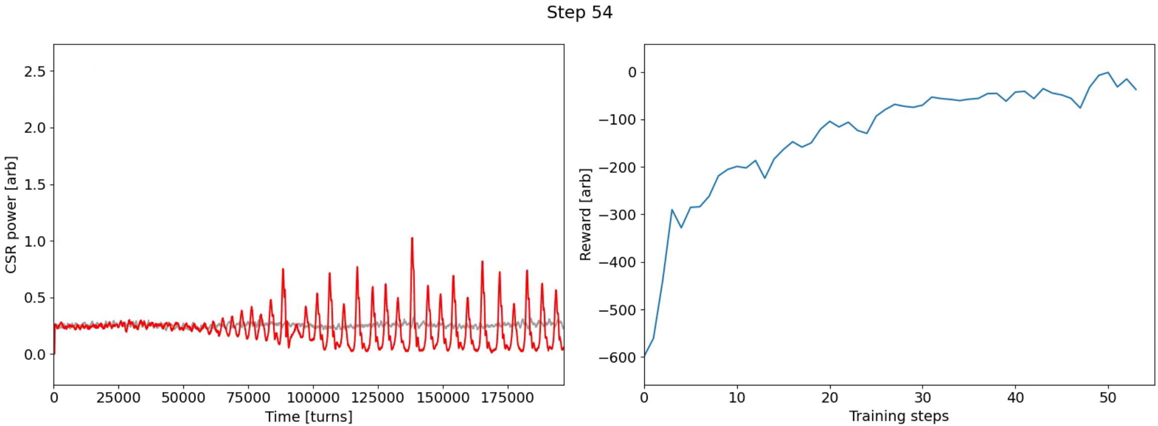
Possibility of cumulative action

System schematic



Courtesy Andrea Santamaria Garcia

In action



Reward with L2 norm from average

Conclusion

Karlsruhe Institute of Technology

- First online training purely on accelerator
- μs Real-Time RL is a viable option
- Its performance is problem dependent
- FPGAs and Heterogeneous platforms are the key
- Hardware aware problem design is fundamental

Sounds interesting? Let’s find more applications!