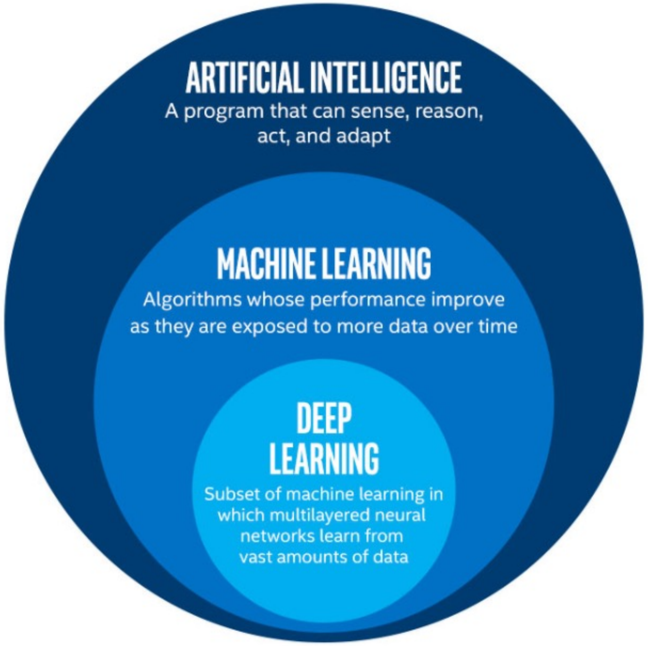
AI4EIC: Digital Twin and AI/ML for complex systems


ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

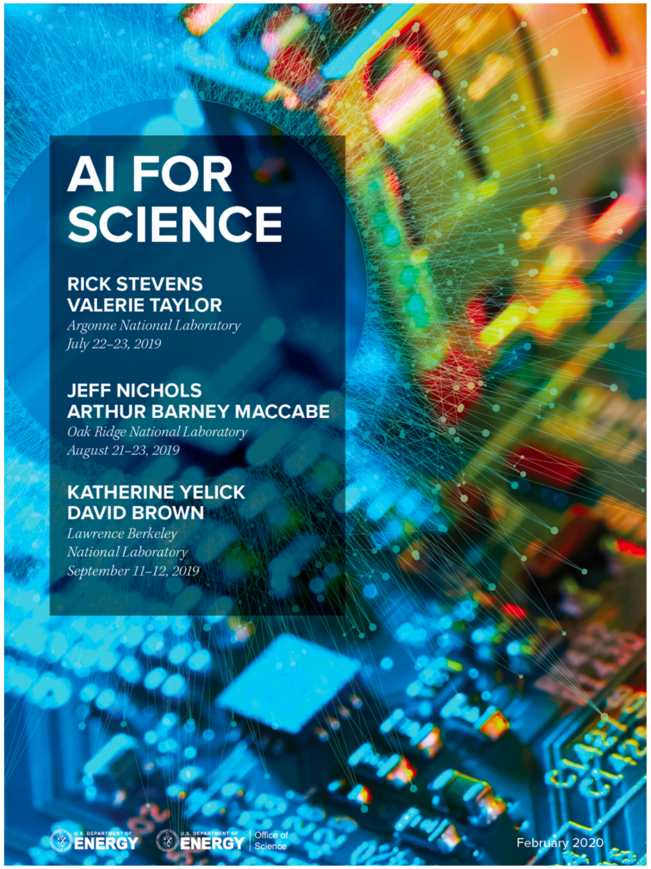
Malachi Schram, Ph.D.
Department of Data Science
Thomas Jefferson National Accelerator Facility

AI for Science Report

“New Deep Learning methods are required to ***detect anomalies*** and ***optimize operating parameters***…”

“... move from ***human-in-the-loop to AI-driven*** design, discovery, and evaluation also manifests across the ***design of scientific workflows***, *optimization of large-scale simulation codes*, and ***operation of next generation instruments***.”

- Excerpts from the Executive Summary



AI FOR SCIENCE

RICK STEVENS
VALERIE TAYLOR
Argonne National Laboratory
July 22–23, 2019

JEFF NICHOLS
ARTHUR BARNEY MACCABE
Oak Ridge National Laboratory
August 21–23, 2019

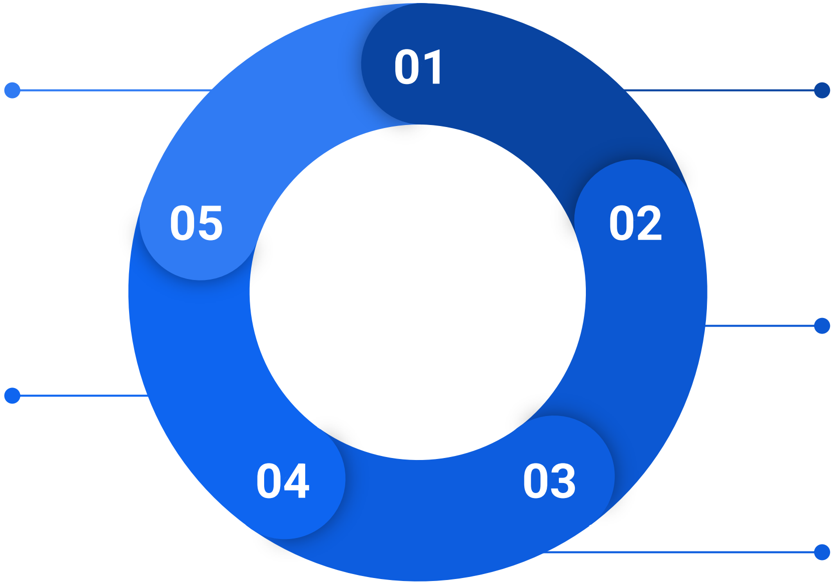
KATHERINE YELICK
DAVID BROWN
Lawrence Berkeley National Laboratory
September 11–12, 2019

U.S. DEPARTMENT OF ENERGY
U.S. DEPARTMENT OF ENERGY | Office of Science

February 2020

Baseline design/control workflows

- Leverage AI/ML components to steer workflow
- No longer requires a human-in-the-loop or prescriptive rules, however, you might need multiple models



Next actions

Provide new actions for the system to perform

AI-based policy model

Continuous training/validating policy model

Data generation

Simulation and/or sensor data
Initial state or updated state based on new actions

Data processing

Data cleaning, re-structuring, etc.

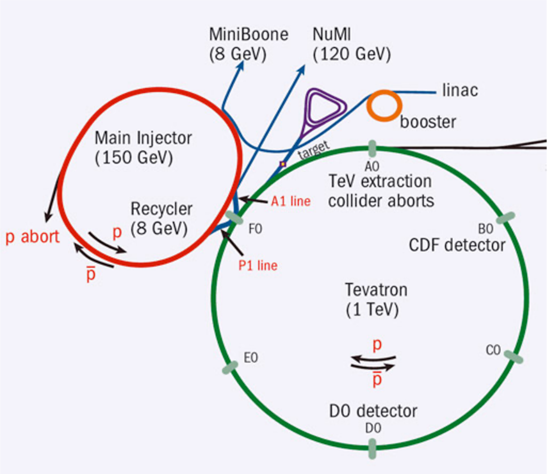
ML-based augmented state representation (optional)

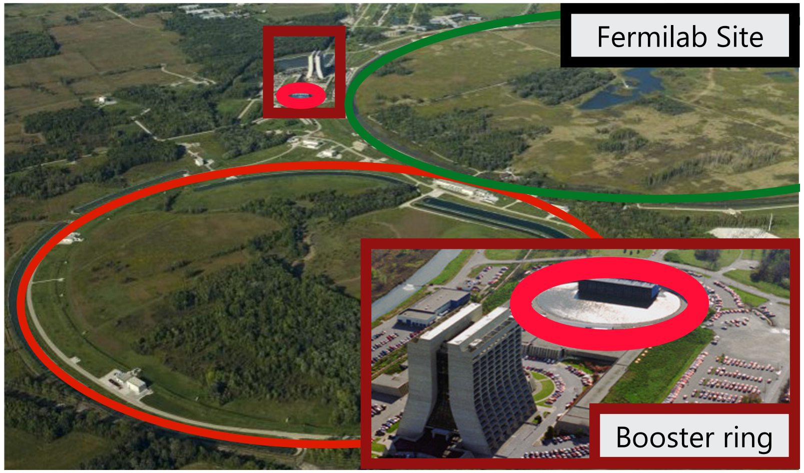
Enhance the state space representation using ML encoding, e.g. anomaly detection input, domain-aware models, etc.)

Workflow using RL for FNAL Booster control policy

Problem definition:
Reduce beam losses in the FNAL Booster by developing a Machine Learning (ML) model that provides an optimal set of actions for accelerator controls

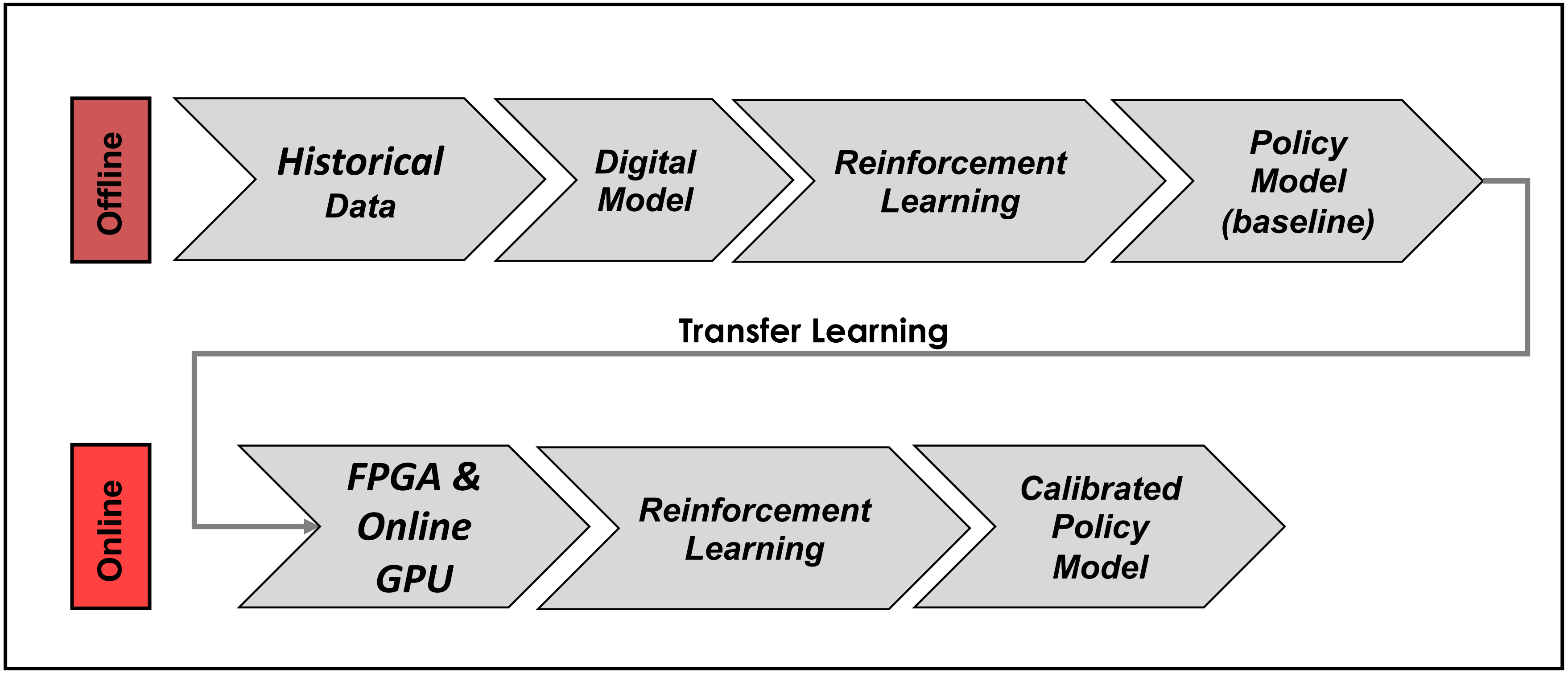
FNAL Accelerator Complex:





Courtesy: Christian Herwig

AI/ML workflow



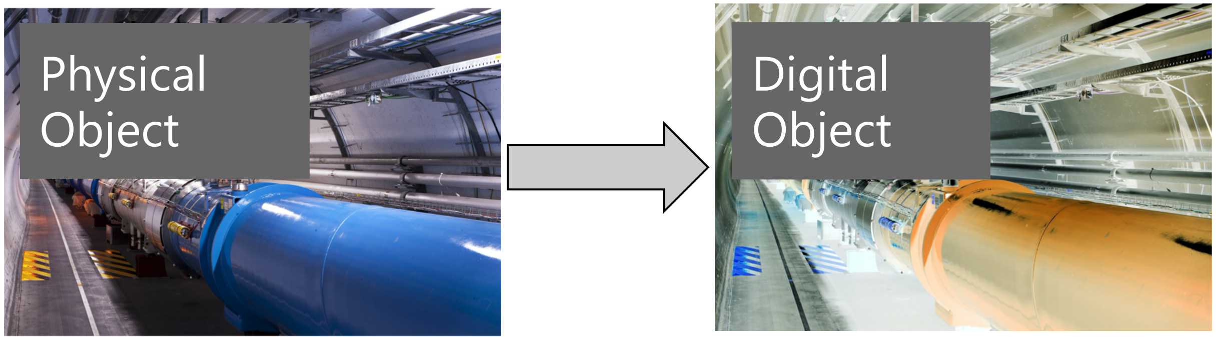
Data-driven ML-based digital model

Scope and usage for digital twin (digital model for now):

- Provide accurate predictions of future time for key variables to be used by the reinforcement learning framework

Dataset provided:

- Historical temporal information from key variables was available based on input from subject matter expert
- Caution:
  - Data did not include detailed history on commissioning, maintenance, etc.
  - Should conduct a full data inventory assessment



Reinforcement Learning

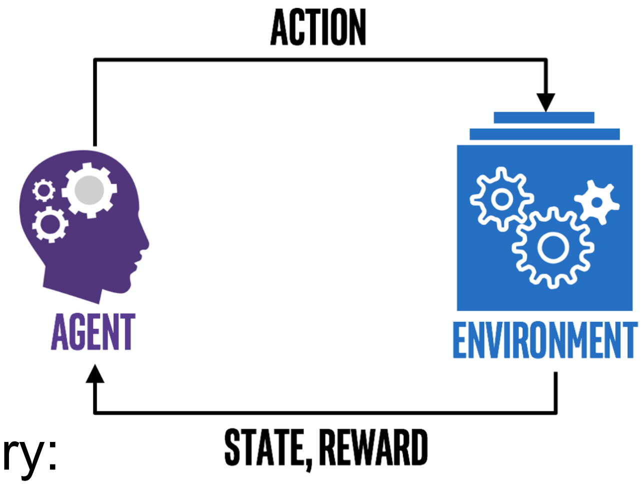
“Reinforcement Learning is learning what to do — how to map situations to actions—so as to maximize a numerical reward signal. The learner is not told which actions to take, but instead must discover which actions yield the most reward by trying them.” - Barto & Sutton

Key concepts to Reinforcement Learning:

- Agent (controller – policy and sampling)
  - *Action* (control signal)
- Environment (controlled system)
  - *State* (representation of environment)
  - *Reward* (numerical consequence of action)

- Sequence of experience and agent forms trajectory:

  Example RL Trace: (S_0, A_0, R_0), (S_1, A_1, R_1), …

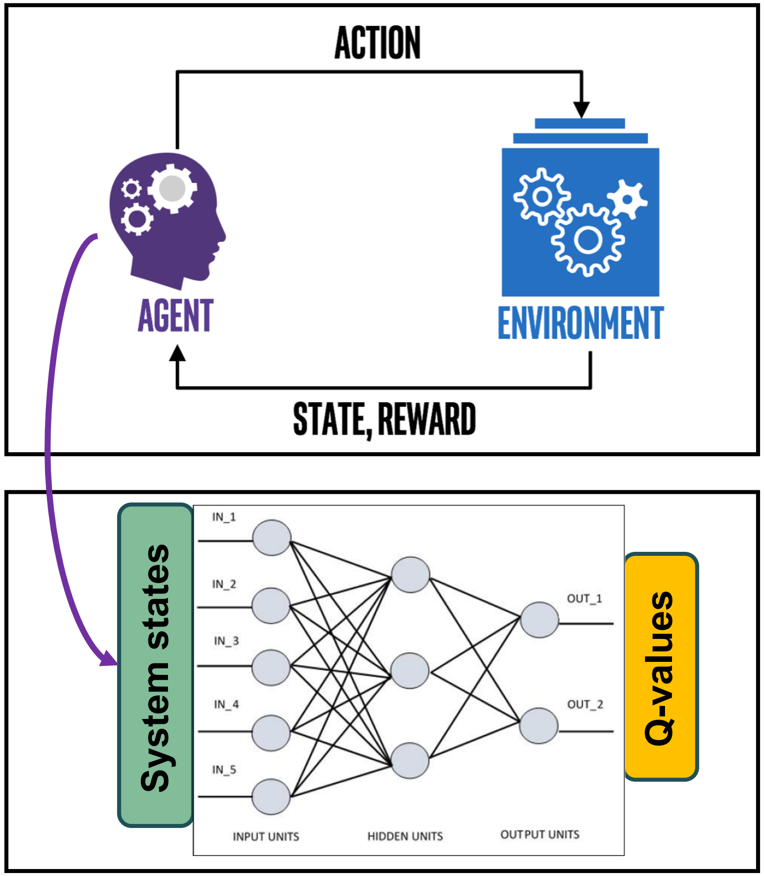


Deep Q-Networks algorithm

DQN uses a deep neural network to estimate the value of taking a specific action at a certain state, also called the state-action value or Q-value.

The DQN agent, once trained properly, suggests the action with the highest Q-value as its policy, and maximizes the total reward over the episode.

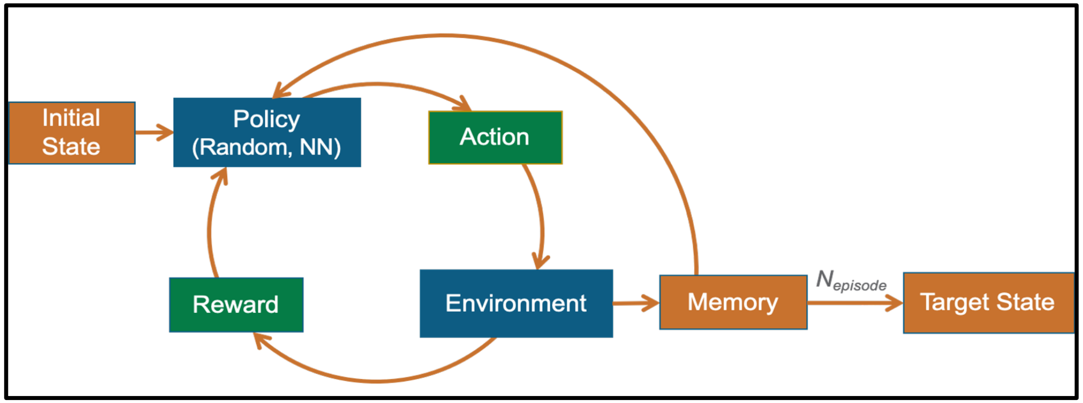
Five discrete actions were defined as possible control changes to the regulator.



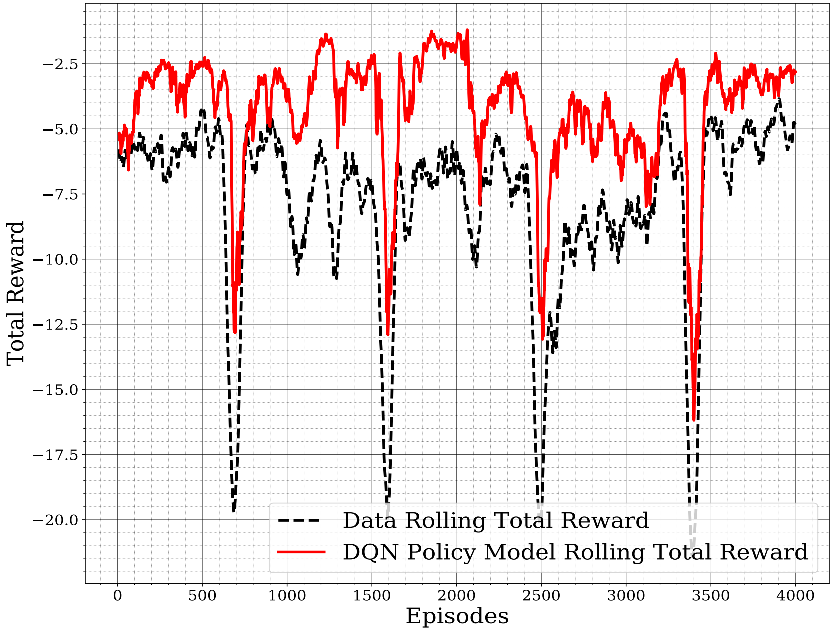
Reinforcement learning FNAL Booster workflow

The optimization was formulated as an episodic problem:

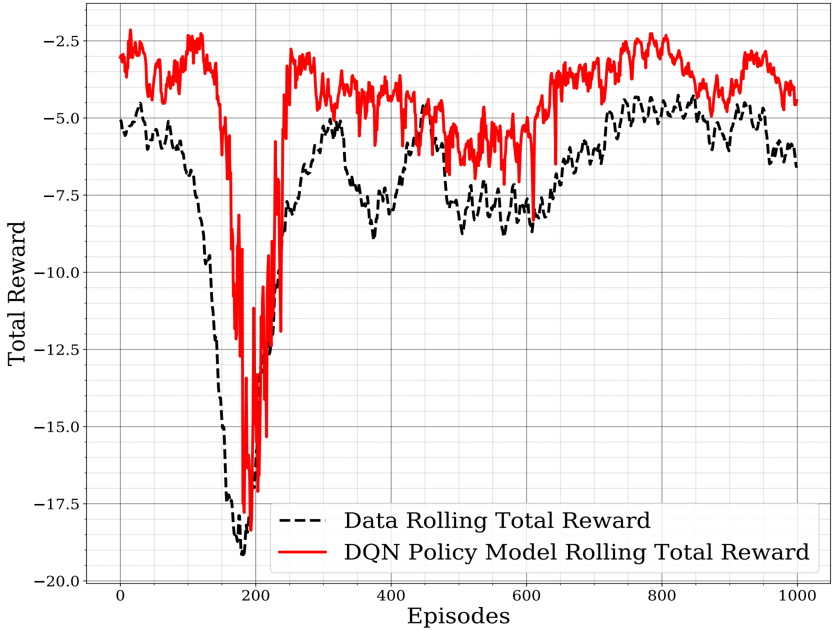
- For each episode the environment was reset to a random initial time in the historical data
- Training and testing data was orthogonal



Train

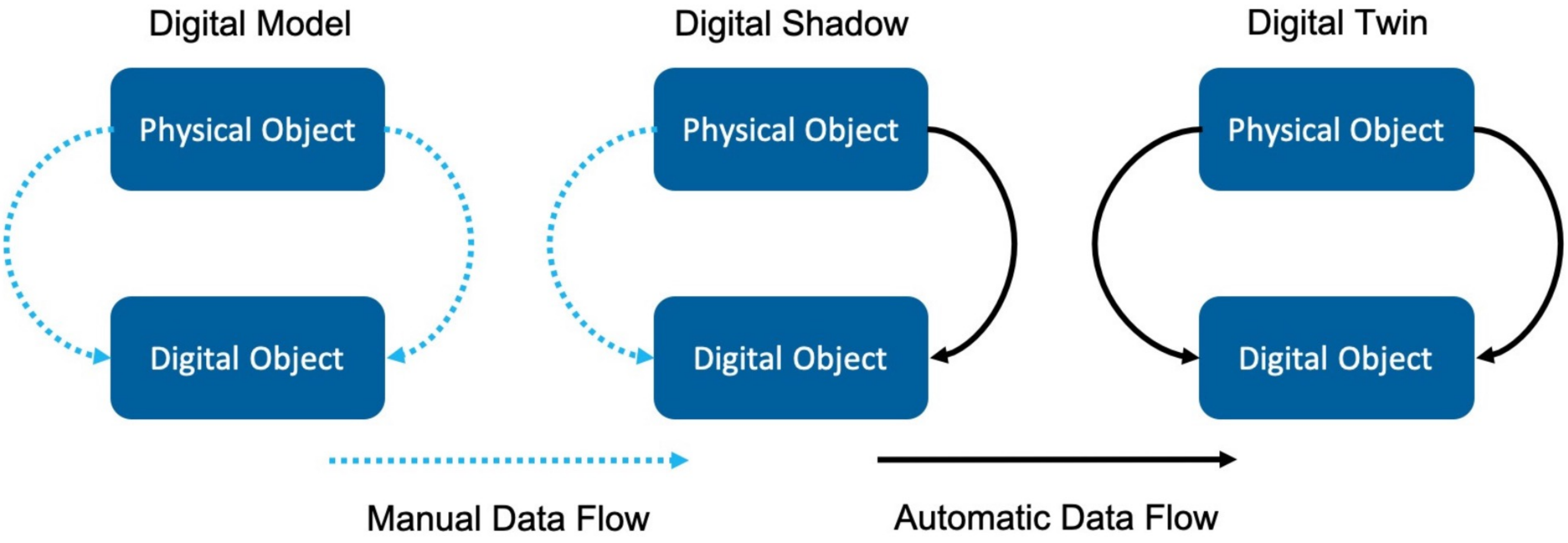


Test



Digital twin definitions

- **Digital Model:** a digital version of a pre-existing or planned physical object
- **Digital Shadow:** digital representation of a physical object with a one-way data flow from the physical to digital object
- **Digital Twin:** data flows between a physical object and a digital object are fully integrated and bilateral

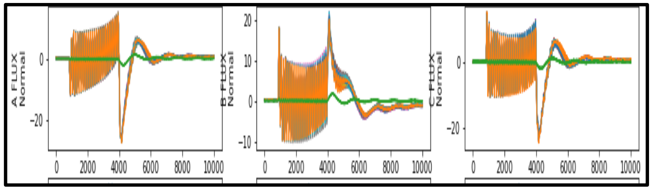


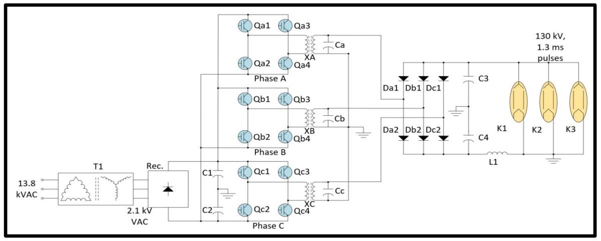
Digital twin application examples

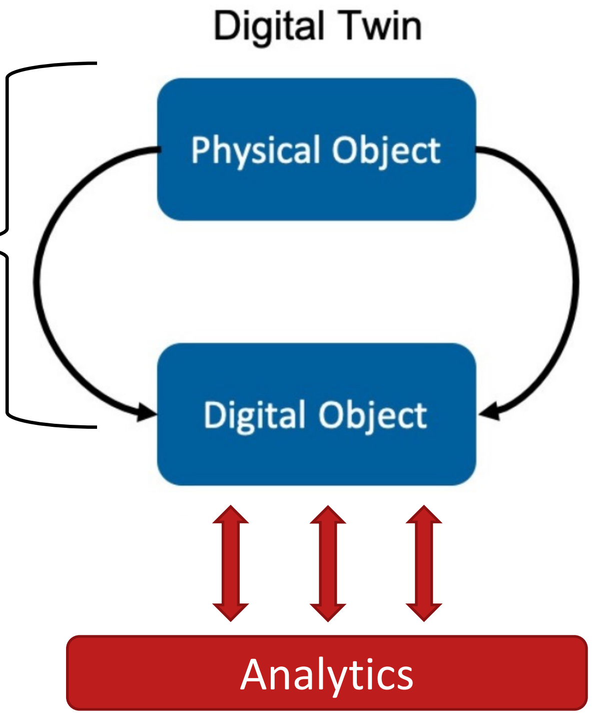
- Digital Twin provides the ability to conduct analytical studies without impacting the physical system, for example:
  - Statistical analysis:
    - Box plots (mean, median, quantiles, etc.)
    - Threads
  - Time series forecasting:
    - Gaussian Processes
    - Quantile Models
    - Recurrent Neural Networks
  - Anomaly detection and classification
    - Random Forest
    - Deep Neural Network
    - Siamese Networks
  - Forecasting component fatigue and failures
    - Physics based models

- Depending on time budget for actionable responses, these studies can be performed on the edge or on HPC systems

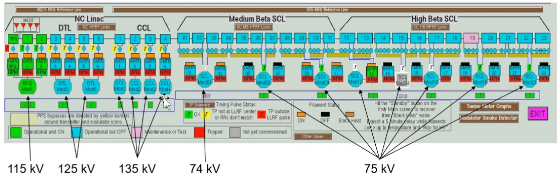
Integrating digital twin into the analytics workflow



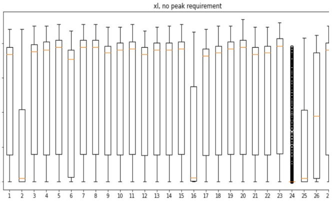


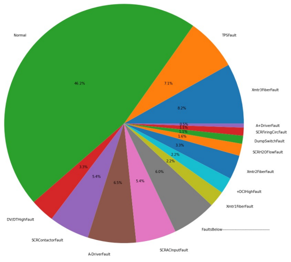


- SNS has 15 HVCM (RFQ, DTL, CCL, SCL) with 3 different designs.
- Each HVCM has multiple faults, and not all the faults have recorded waveforms.
- The waveforms for each HVCM design may be different

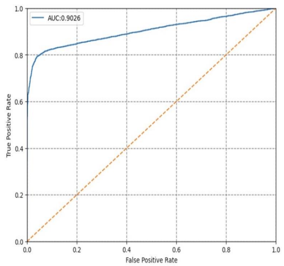


Statistical Analysis

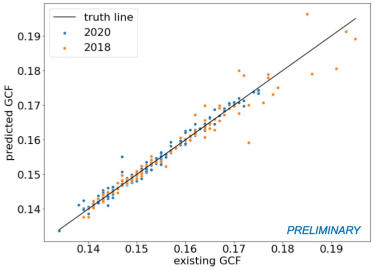




Monitoring & Anomaly Detection*



Calibration*

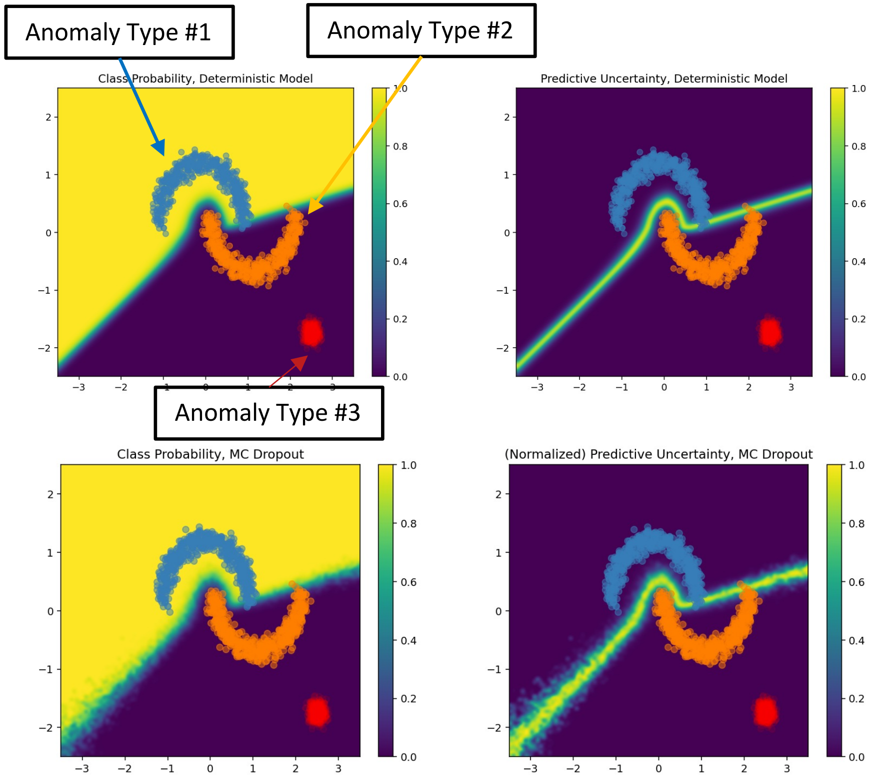


Understand what your model knows and doesn’t know

Different method yield vastly different classification predictions, some examples:

- Deterministic
- MC Dropout
- Deep Ensemble
- Gaussian Processes
- Bayesian Neural Networks

- Different models architectures can yield better results if you do not know all classifications



Anomaly Type #1

Anomaly Type #2

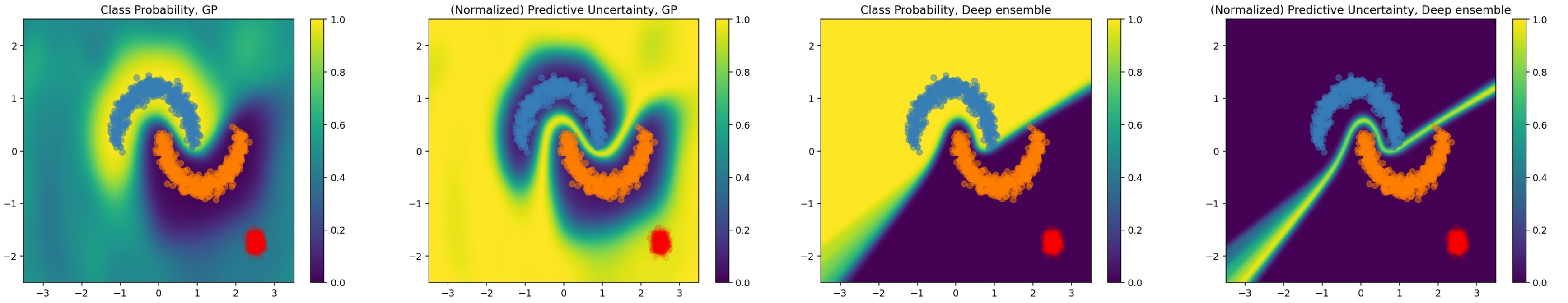
Anomaly Type #3

Class Probability, Deterministic Model

Predictive Uncertainty, Deterministic Model

Class Probability, MC Dropout

(Normalized) Predictive Uncertainty, MC Dropout



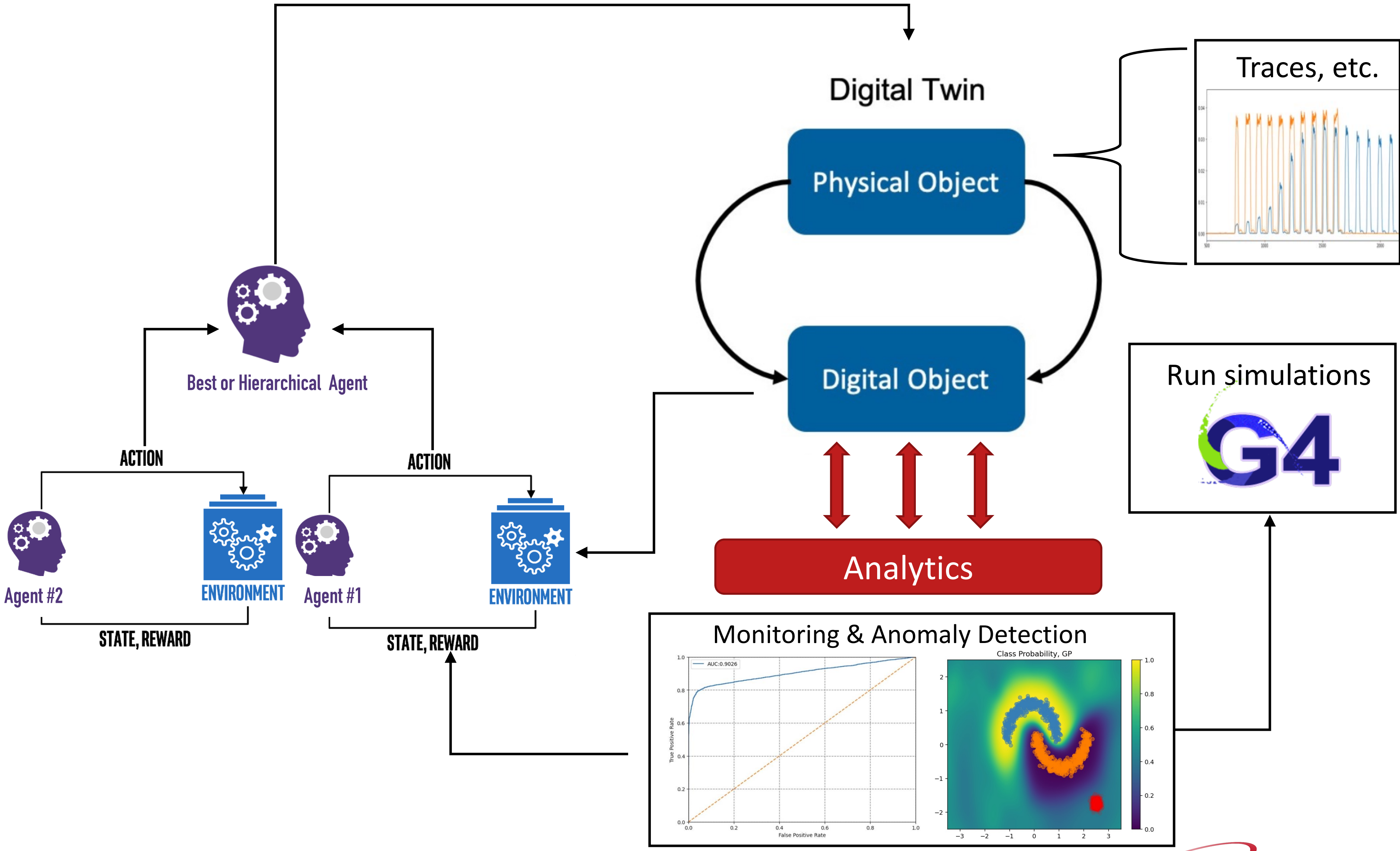
Class Probability, GP

(Normalized) Predictive Uncertainty, GP

Class Probability, Deep ensemble

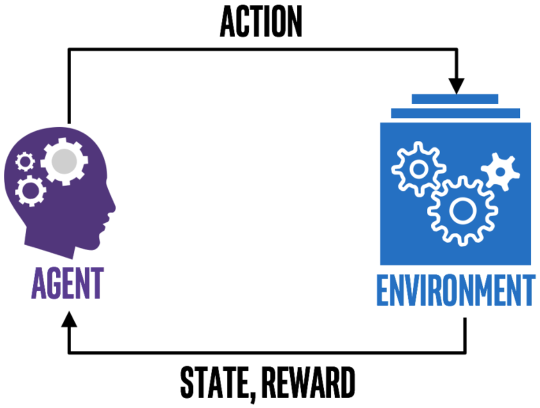
(Normalized) Predictive Uncertainty, Deep ensemble

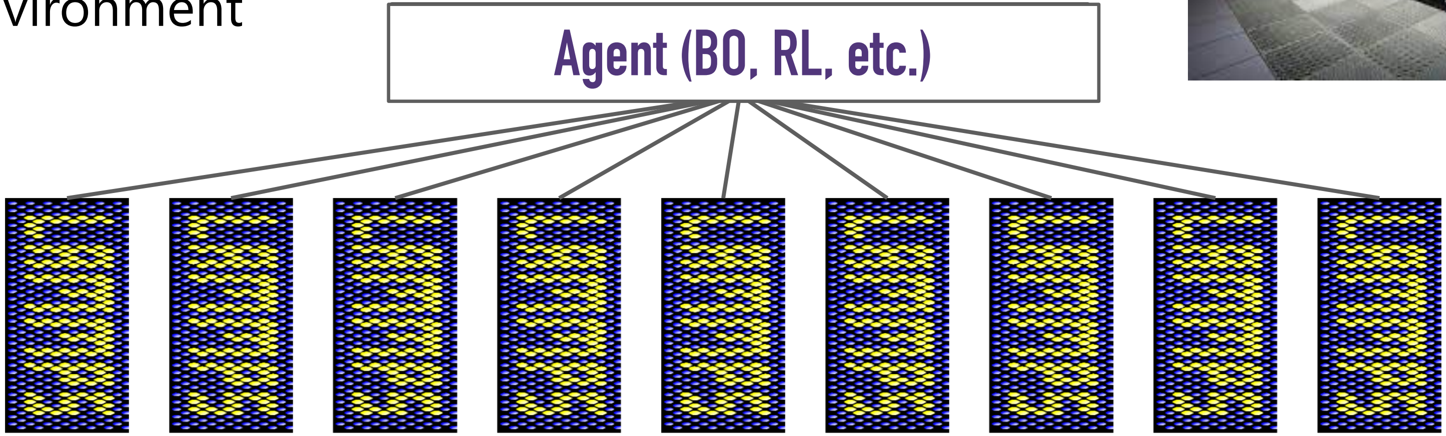
Extending digital twin and analytics for control workflow



Scaling workflow on HPC system

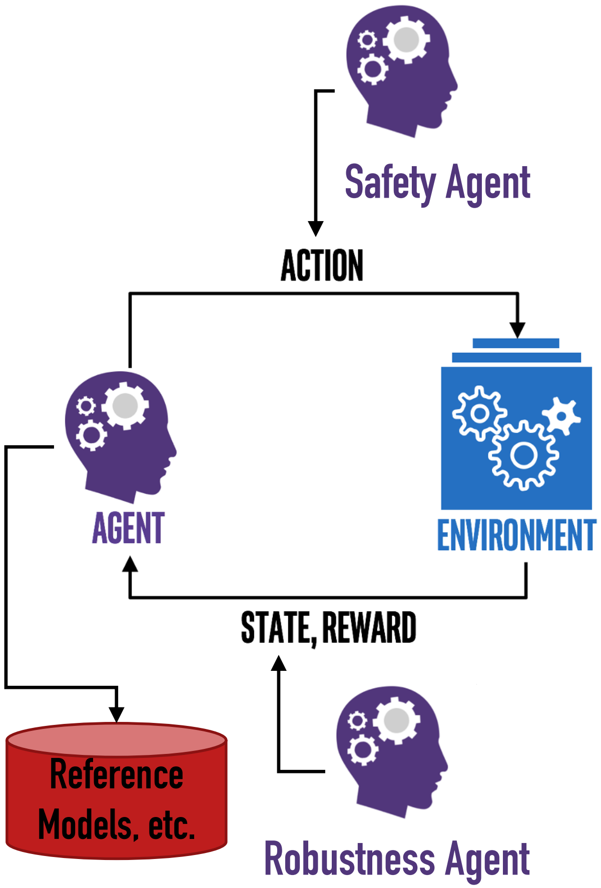
- Digital Object is interfaced with industry standard OpenAI gym environment
- To accelerate the data generation we developed a MPI-based framework
- We created an agent that maps the action-reward for all simulations
- A production job split the MPI communications between the agent and each Digital Object environment





Additional considerations

- **Safety:** We need to ensure that the actions provided by the AI-based controller are within “safe” parameters.
- **Robustness:** Understanding how the AI-based control behaves in the presence of unexpected changes in the input state
  - Models robustness - loss landscape, etc.
  - Impact from noisy and/or dead sensors
  - Adversarial techniques
- **Explainability:** With all AI/ML models we need to understand why the model made a given prediction:
  - Saliency maps
  - Hierarchical models
- **Continuous learning:** The underlying system dynamics can change over time. We need to evaluate the current states to previous states to determine if there has been any notable change that would require the model to be updated
- **Computing infrastructure:** Data-intensive workflow, data movement (DTNs, Wired/Wireless), processing architecture (FPGA, GPU, etc.), etc


Safety Agent
ACTION
AGENT
ENVIRONMENT
STATE, REWARD
Reference Models, etc.
Robustness Agent