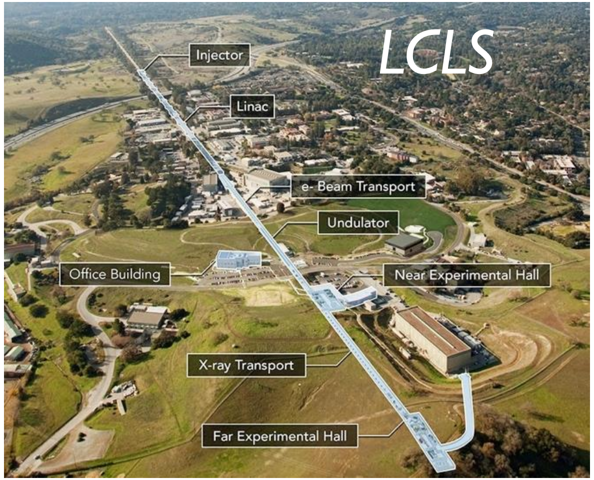
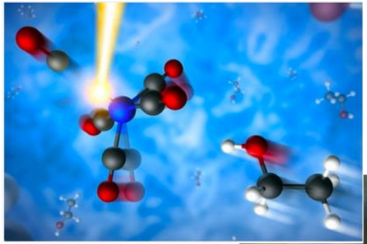
Combining system models and ML-based control to enable new capabilities in beam tuning for accelerators

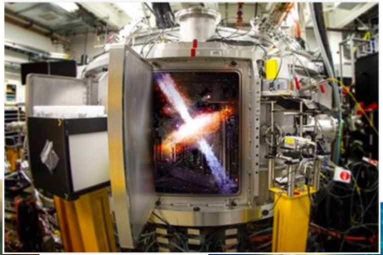
ML4FE
May 20, 2025

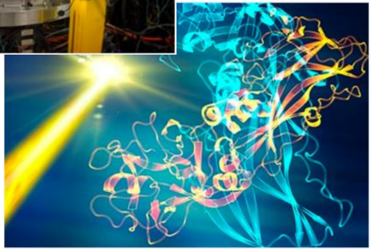
Auralee Edelen
Machine Learning Department Head, Accelerator Research Division
SLAC National Accelerator Laboratory
edelen@slac.stanford.edu

R. Roussel, D. Ratner, D. Kennedy, Y. Yazar, E. Cropp, C. Mayes, J. Bellister, Z. Zhu, Z. Zhang, C. Emma, S. Miskovich, W. Neiswanger, C. Xu, T. Boltz, J.P. Gonzalez-Aguilera, and many other collaborators





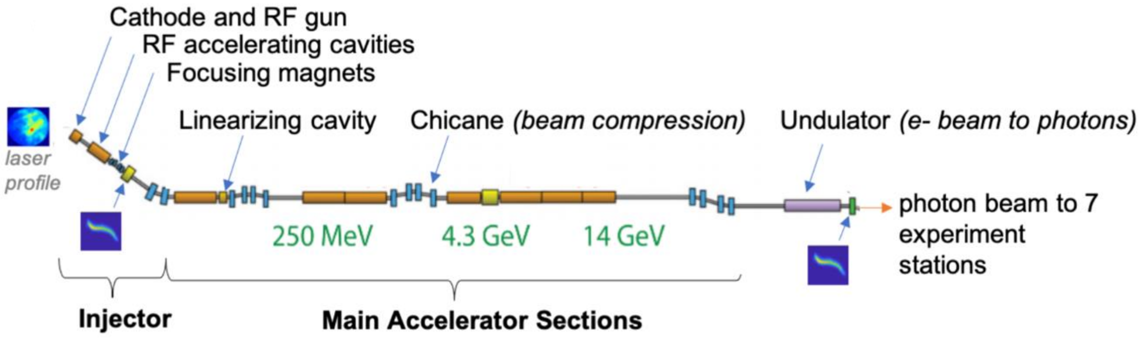


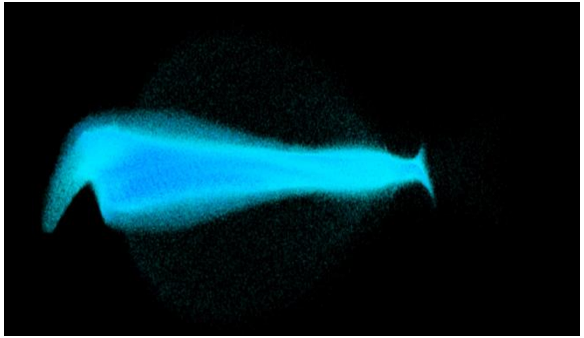


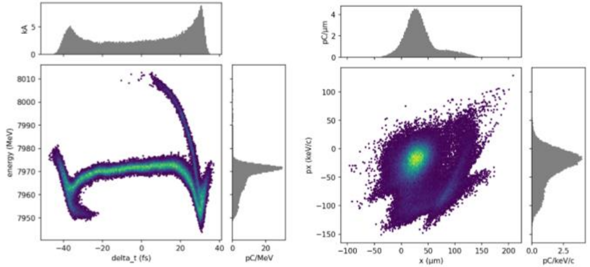
beam duration, average energy, energy spread, pulse intensity adjusted for each experiment

https://lcls.slac.stanford.edu

Detailed beam phase space customization required for different experiments





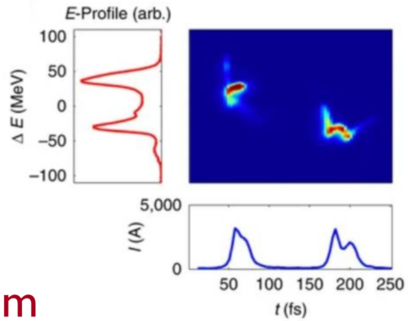


Beam exists in 6-D position-momentum phase space

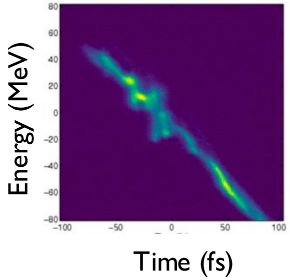
Incomplete information: measure 2-D projections or reconstruct based on perturbations of upstream controls (e.g. tomography)

Dozens-to-hundreds of controllable variables and hundreds-of-thousands to monitor

Increasingly dynamic control needed during experiments



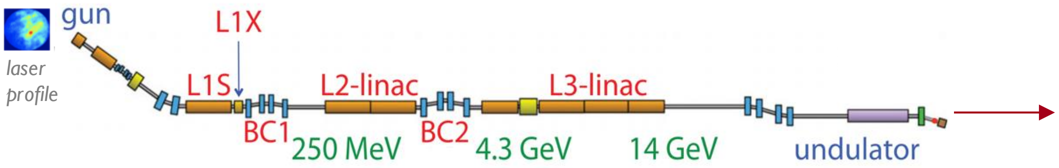
A. Marinelli, et al., Nat. Commun. **6**, 6369 (2015)



A. Marinelli, IPAC'18

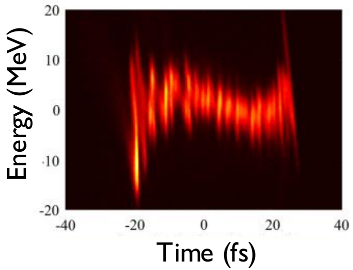
Nonlinear, high-dimensional optimization/control problem

wide spectrum of tuning needs

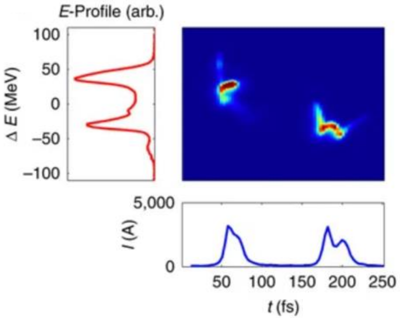


photon beam to 7 experiment stations

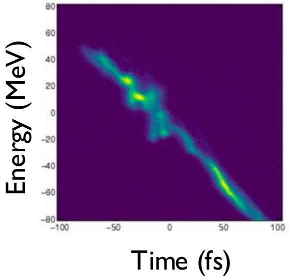
J. Qiang et al, PRAB (2017)



A. Marinelli, et al., Nat. Commun. 6, 6369 (2015)



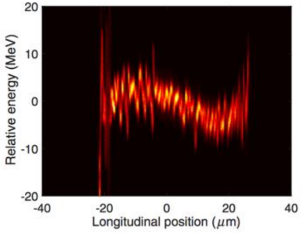
A. Marinelli, IPAC'18



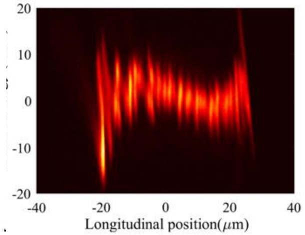
Rapid beam customization	Achieve new configurations + unprecedented beam parameters	Fine control to maintain stability within tolerances

computationally expensive simulations

Simulation



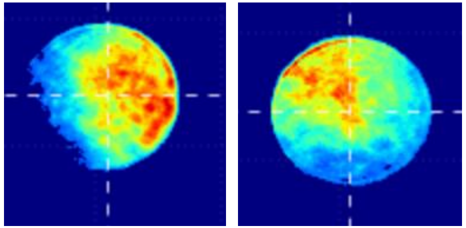
Measurement



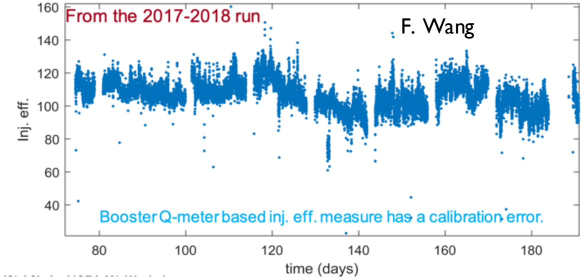
"10 hours on thousands of cores at the NERSC"

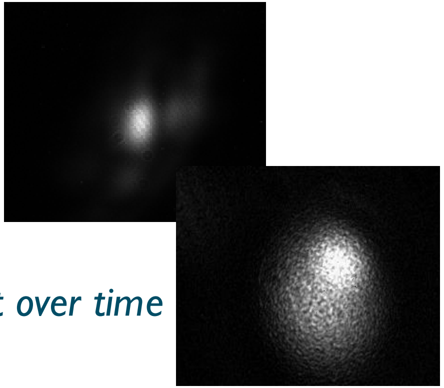
J. Qiang, et al., PRSTAB30, 054402, 2017

fluctuations/noise (e.g. initial beam conditions)

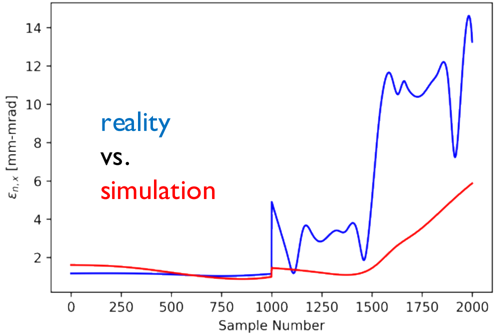


hidden variables / sensitivities

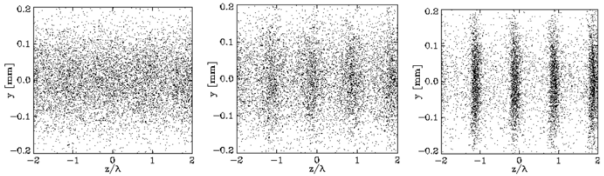




drift over time



many small, compounding sources of uncertainty

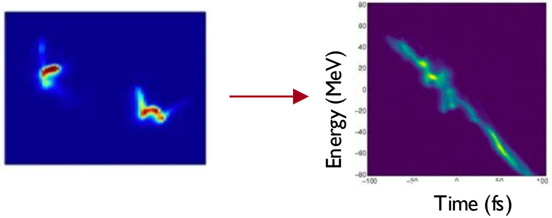


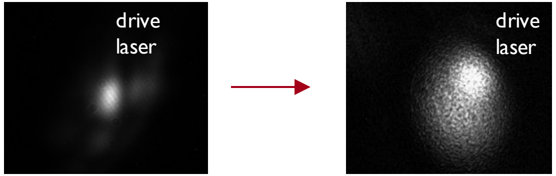
nonlinear effects / instabilities

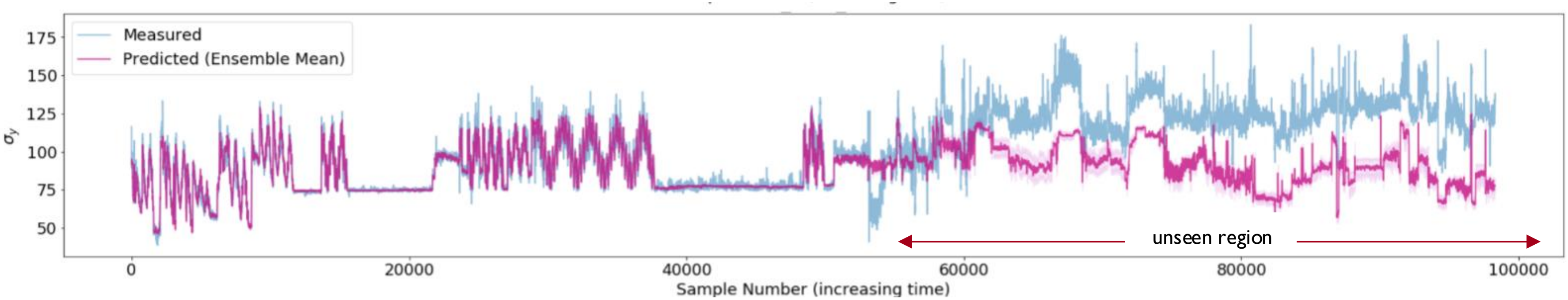
Distribution Shift is a Major Challenge in Particle Accelerators

Many sources of change over time:

- Deliberate changes in beam configuration (e.g. beam charge)
- Unintended drift in initial conditions (including in unobservable variables), diurnal temperature/humidity changes, etc
- Time-dependent action of feedback systems





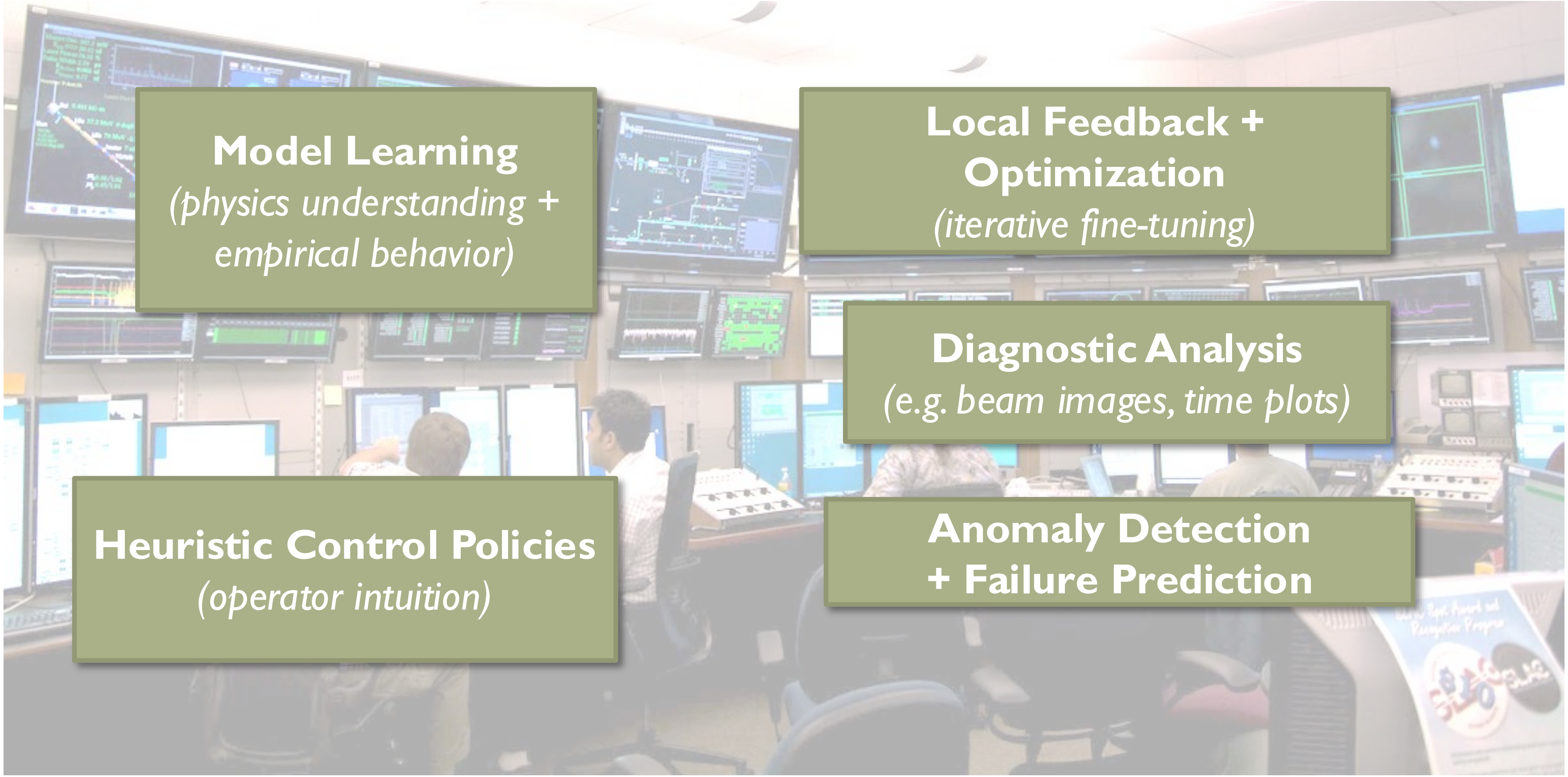


Example: beam size prediction and uncertainty estimates under drift from a neural network
Uncertainty estimate from neural network ensemble does not cover prediction error, but does give a qualitative metric for uncertainty

Reliable uncertainty estimates and model adaptation methods are key for putting online models to use operationally
Need fast ways of obtaining characterization data from accelerator

We rely heavily on human operators for day-to-day control tasks . . .

We rely heavily on operators for day-to-day control tasks . . .



Model Learning
(physics understanding + empirical behavior)

Local Feedback + Optimization
(iterative fine-tuning)

Diagnostic Analysis
(e.g. beam images, time plots)

Heuristic Control Policies
(operator intuition)

Anomaly Detection + Failure Prediction

…many analogous techniques in optimization, machine learning, computer vision, etc.

Tuning approaches leverage different amounts of data / previous knowledge → suitable under different circumstances

less ← assumed knowledge of machine → more

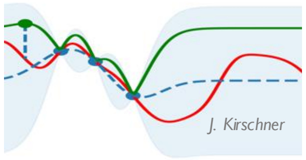
Model-Free Optimization

Observe performance change after a setting adjustment

→ estimate direction or apply heuristics toward improvement

gradient descent
simplex
ES

Model-guided Optimization

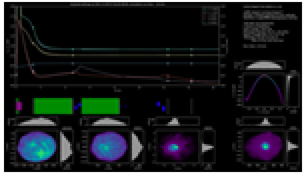


Update a model at each step

→ use model to help select the next point

Bayesian optimization
reinforcement learning

Global Modeling + Feed-forward Corrections



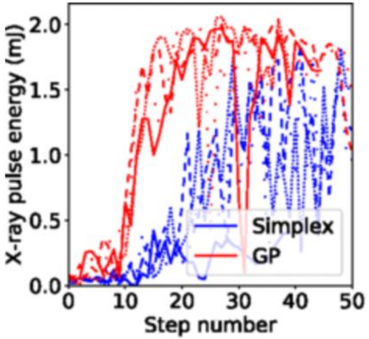
→ provide initial guess (i.e. warm start)
→ provide insight to operators
→ model-based control

ML system models +
inverse models

General strategy: start with sample-efficient methods that do well on new systems, then build up to more data-intensive and heavily model-informed approaches.

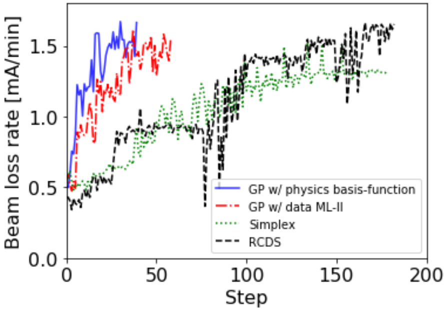
Many successes with Bayesian Optimization *(+ algorithmic improvements)*

FEL pulse energy tuning at LCLS



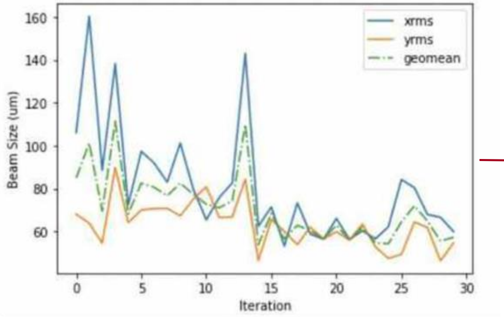
Duris et. al. PRL , **2020**

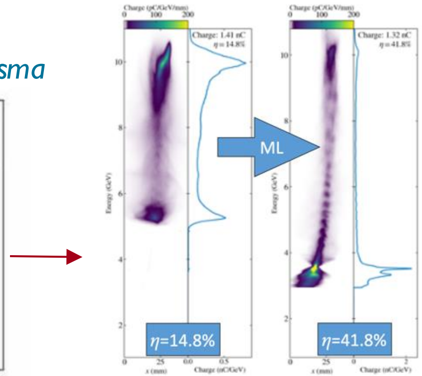
Loss rate tuning at SPEAR3



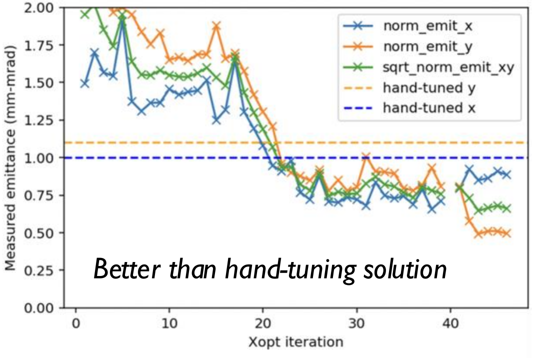
Hanuka et. al. PRAB , **2021**

Sextupole tuning at FACET-II
2x efficiency of acceleration in plasma

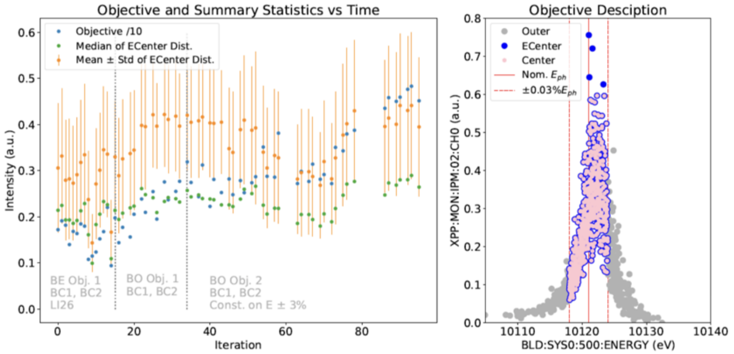




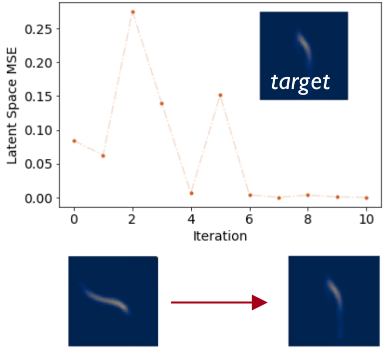
Beam emittance tuning for LCLS-II injector



Tuning on monochrometer signal



Longitudinal phase space tuning on LCLS



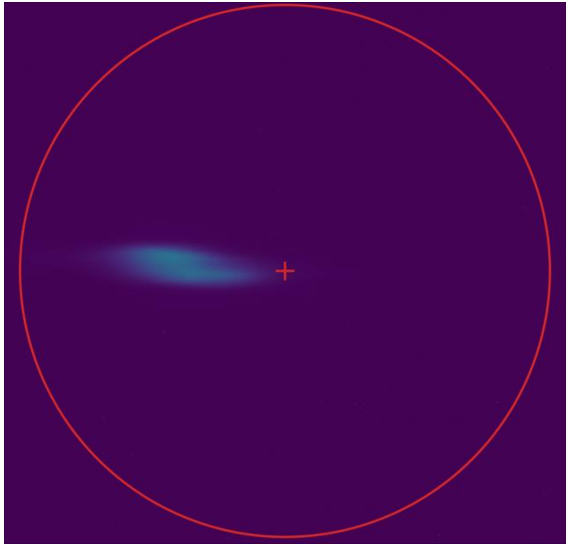
Algorithms being implemented/distributed in Xopt: https://github.com/xopt-org/Xopt

Comprehensive review of advanced BO for particle accelerators: https://doi.org/10.1103/PhysRevAccelBeams.27.084801

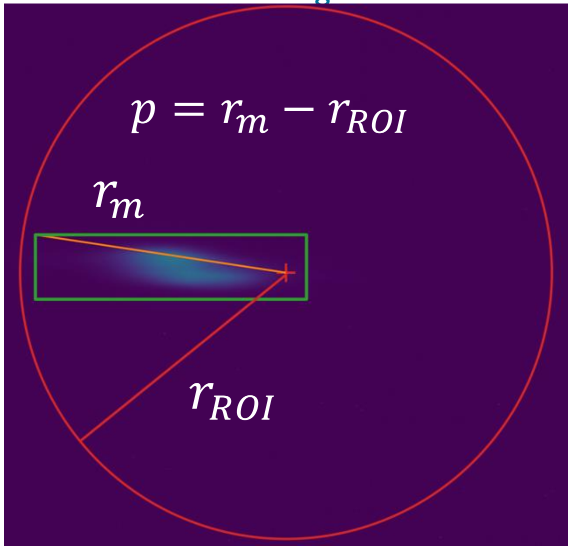
Incorporating Constraints

We want to ensure during measurements that the beam stays on screen
→ Define a **smoothly varying** penalty function to act as a constraint

Define a circular ROI

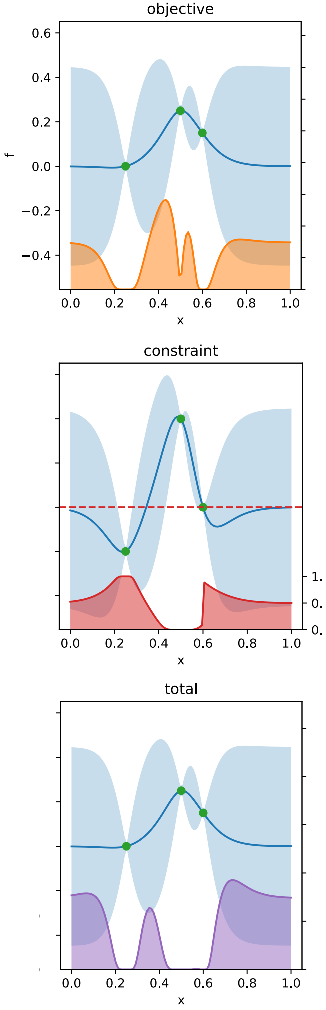


Measure maximum distance from the ROI center to bounding box corners



Constraint: $p \leq 0$

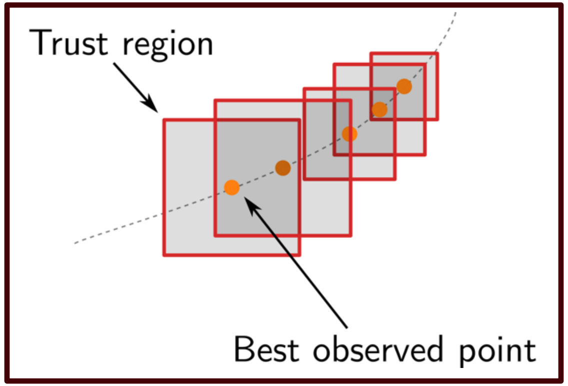
Other examples: Beam losses, dark current production, emittance, etc.

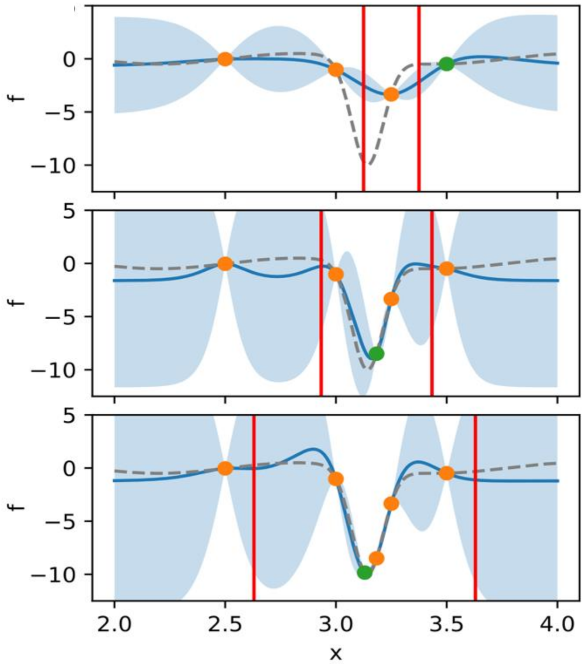


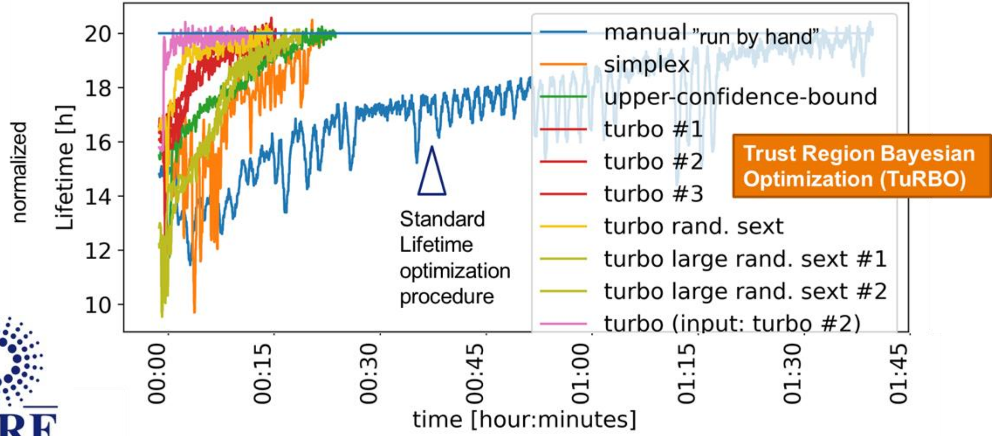
Gardner et. al. ICML 2014

See R. Roussel et al., PRAB (2024) https://journals.aps.org/prab/abstract/10.1103/PhysRevAccelBeams.27.084801

Trust Region Bayesian Optimization







S. Maria Liuzzo, ICALEPCS 2023, MO3AO01

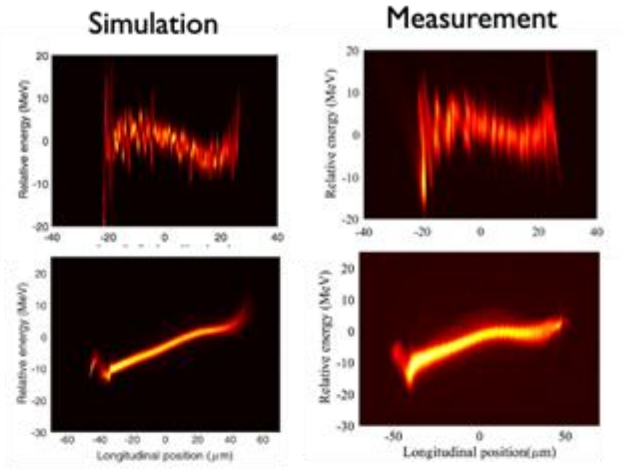
ESRF for lifetime optimization:

- 50x faster than human operator
- Achieved best lifetime yet observed at ESRF (41 hours)
- Now used in regular operation

Trust region BO enables efficient extension to very high dimensional problems with narrow ranges of stability

Fast-Executing, Accurate System Models

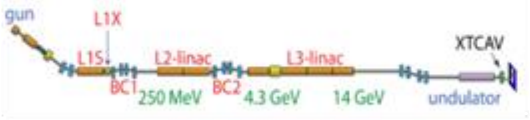
Accelerator simulations that include nonlinear and collective effects are powerful tools, but they can be computationally expensive



10 hours on thousands of cores at NERSC!

J. Qiang, et al., PRSTAB30, 054402, 2017

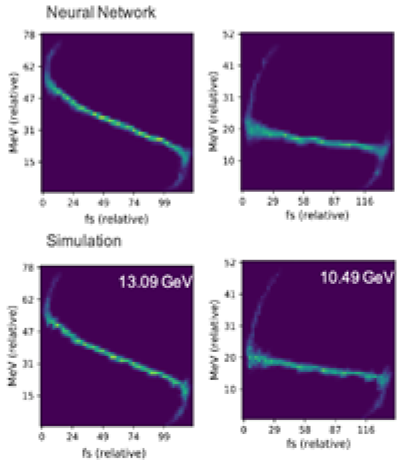
ML models are able to provide fast approximations to simulations (“surrogate models”)



Linac sim in Bmad with collective beam effects

Scan of 6 settings in simulation

Variable	Min	Max	Nominal	Unit
L1 Phase	-40	-20	-25.1	deg
L2 Phase	-50	0	-41.4	deg
L3 Phase	-10	10	0	deg
L1 Voltage	50	110	100	percent
L2 Voltage	50	110	100	percent
L3 Voltage	50	110	100	percent



< ms execution speed

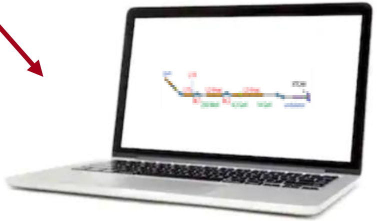
10^6 times speedup

Edelen et al., NeurIPS 2019

Long history now of using ML modeling to enable accurate predictions of accelerator system responses with unprecedented speeds

Fast-Executing, Accurate System Models

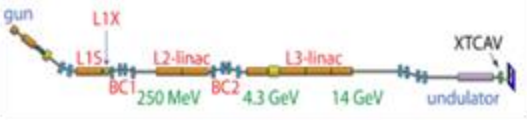
Bringing simulation tools from HPC systems to online/local compute



Control prototyping
Experiment planning

Online prediction
Model-based control

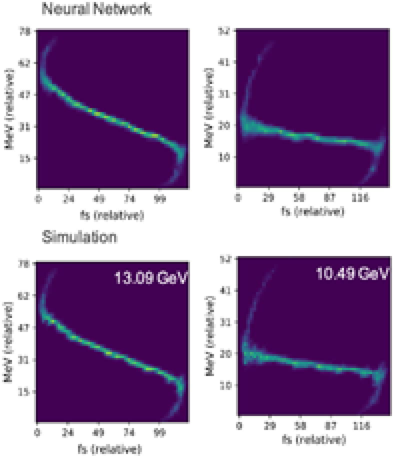
ML models are able to provide fast approximations to simulations ("surrogate models")



Linac sim in Bmad with collective beam effects

Scan of 6 settings in simulation

Variable	Min	Max	Nominal	Unit
L1 Phase	-40	-20	-25.1	deg
L2 Phase	-50	0	-41.4	deg
L3 Phase	-10	10	0	deg
L1 Voltage	50	110	100	percent
L2 Voltage	50	110	100	percent
L3 Voltage	50	110	100	percent



< ms execution speed

10^6 times speedup

Edelen et al., NeurIPS 2019

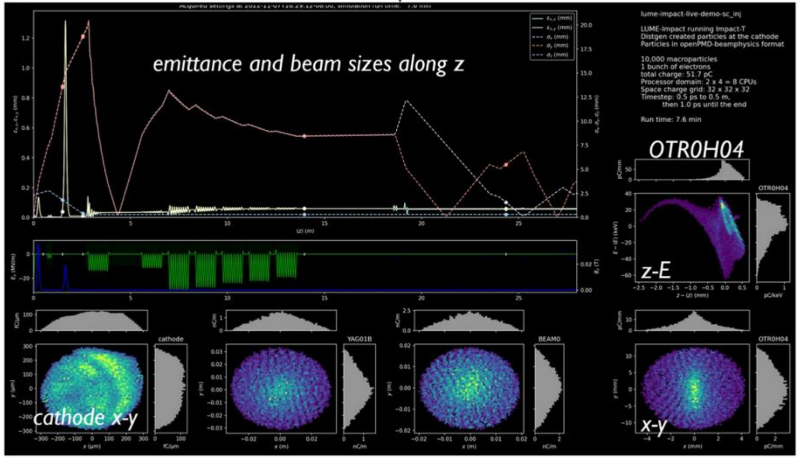
Long history now of using ML modeling to enable accurate predictions of accelerator system responses with unprecedented speeds

Combining BO with Warm Starts from Online Physics Models

Used combination of online physics simulation and Bayesian optimization algorithms to aid LCLS-II injector commissioning

Readings from machine via EPICS

injector settings, laser profile from VCC image

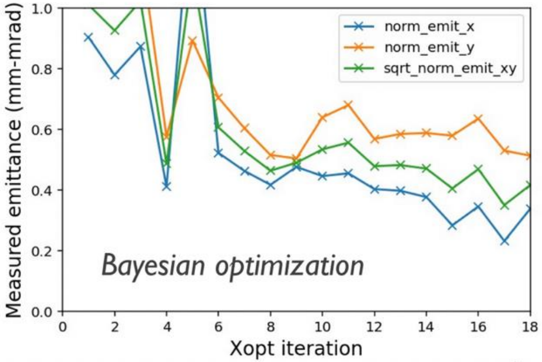


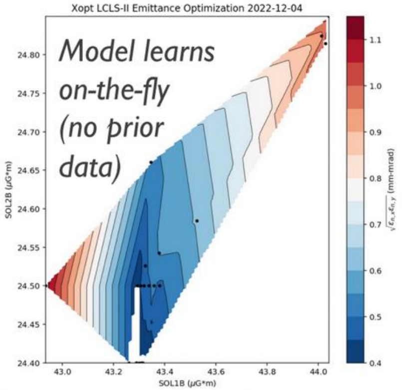
LCLS-II live sim: run on HPC and display in control room

Updates every 3-8 mins, space charge included, uses LUME-IMPACT

Adjust settings / ranges with insight from predictions

Hand over to ML-based optimization for fine tuning





Best emittance yet obtained during LCLS-II injector commissioning

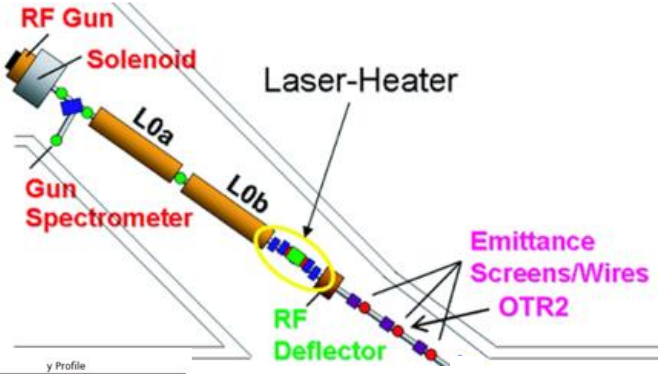
despite extensive previous hand-tuning

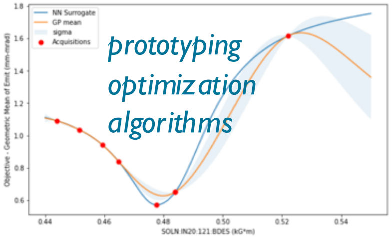
Physicists' intuition aided by detailed online physics model → simple example of how a "virtual accelerator" can aid tuning

HPC enables fundamentally new capabilities in what can be realistically simulated online

In Regular Use: Injector Surrogate Model at LCLS

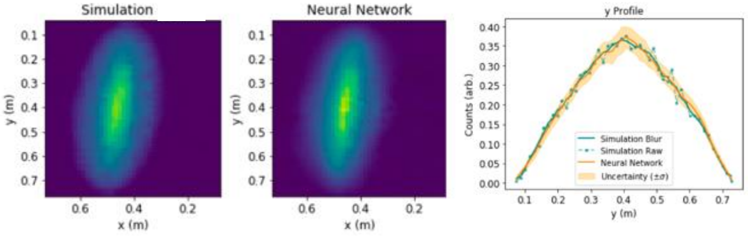
- ML models trained on detailed physics simulations with nonlinear collective effects
- Accurate over a wide range of settings → calibrate to match machine measurements
- Provide initial parameters for downstream model



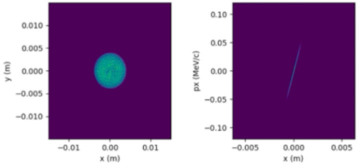


prototyping optimization algorithms

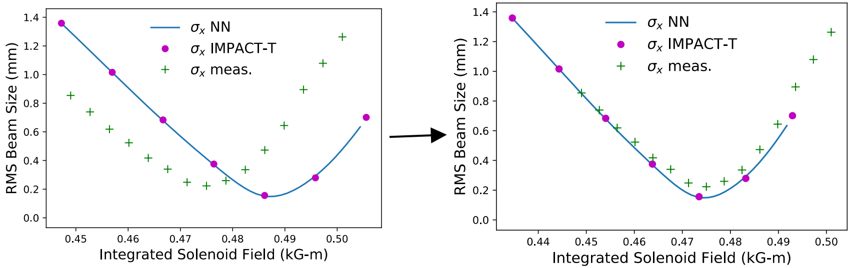
ML model matches simulation under interpolation



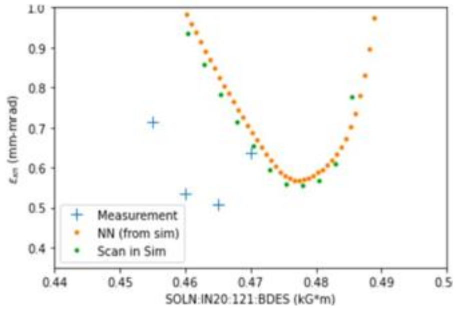
Simulation and ML model trained on it are qualitatively similar to measurements under interpolation (setting combinations reasonable distance from training set)



interactive model widget and visualization tools



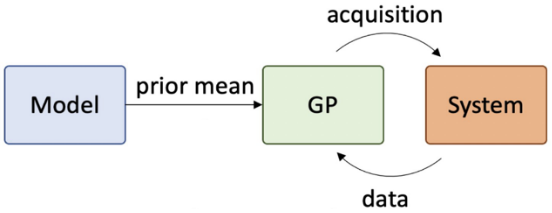
Automatic adaptation of models and identification of sources of deviation between simulations and as-built machine

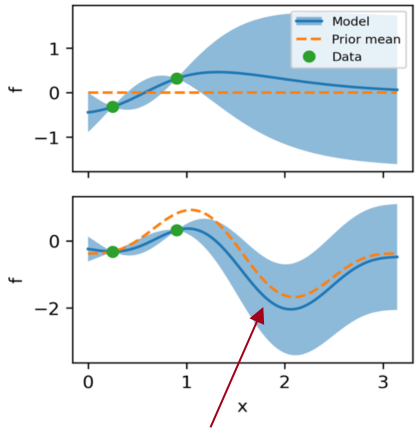


ML models trained on simulations and measurements have enabled fast prototyping of new optimization algorithms, facilitated rapid model adaptation under new conditions, and can directly aid online tuning and operator decision making

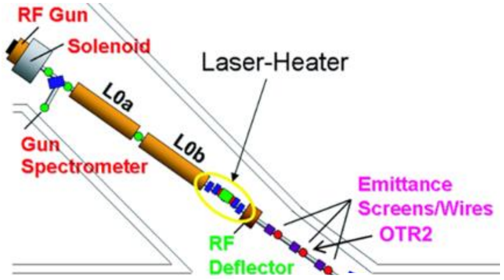
Leveraging Online Models for Faster Optimization

Combining existing models with BO
→ **important for scaling up to higher dimension**

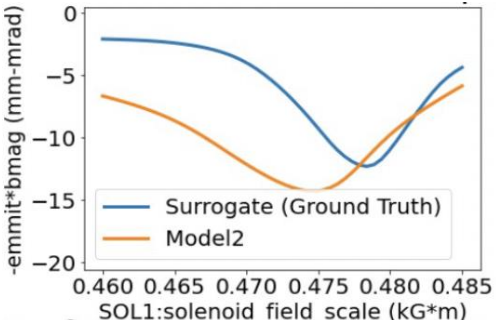




Prototyped on LCLS injector
variables: *solenoid, 2 corrector quads, 6 matching quads*
objective: *minimize emittance and matching parameter*

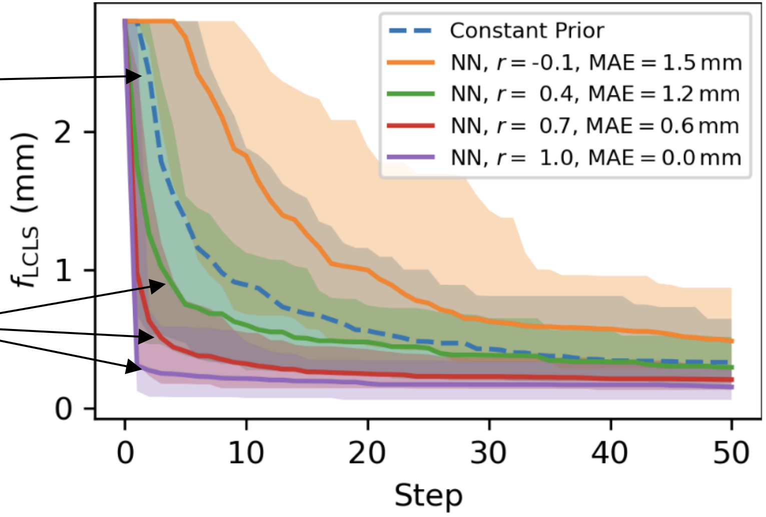


model prediction returns to prior



regular Bayesian optimization

prior mean from models with different fidelity

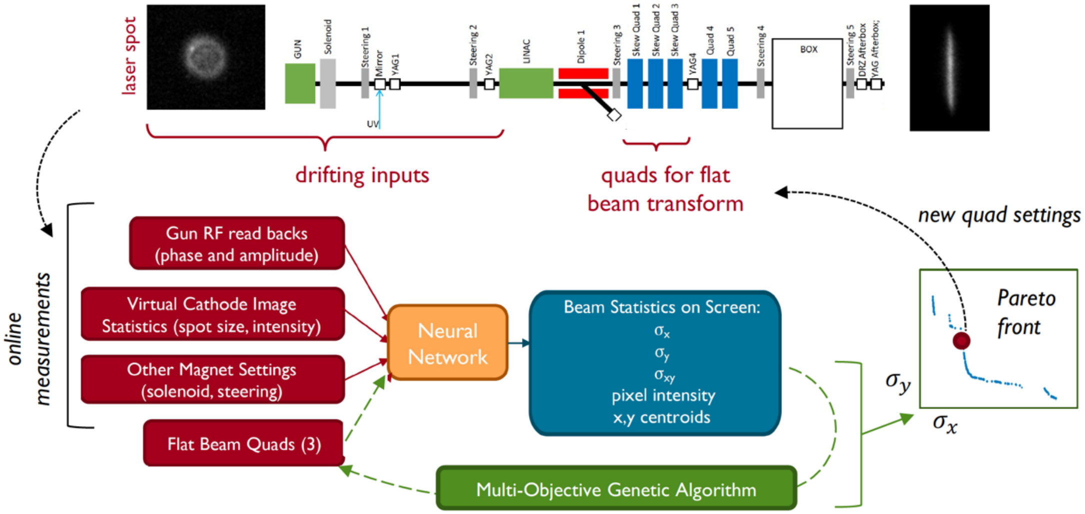


Even prior mean models with substantial inaccuracies provide a boost in optimization speed

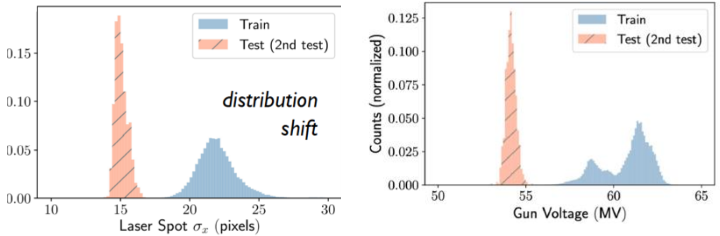
https://arxiv.org/abs/2403.03225 https://arxiv.org/abs/2211.09028

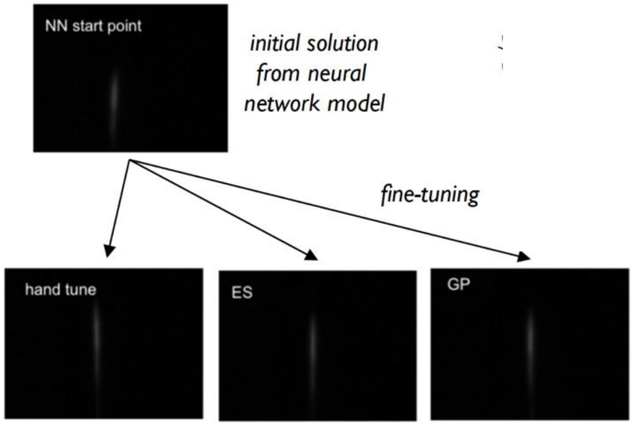
Example: Compensate for Upstream Drift in Fast Setup

E. Cropp et al., in preparation

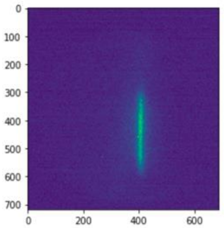


Can work even under distribution shift





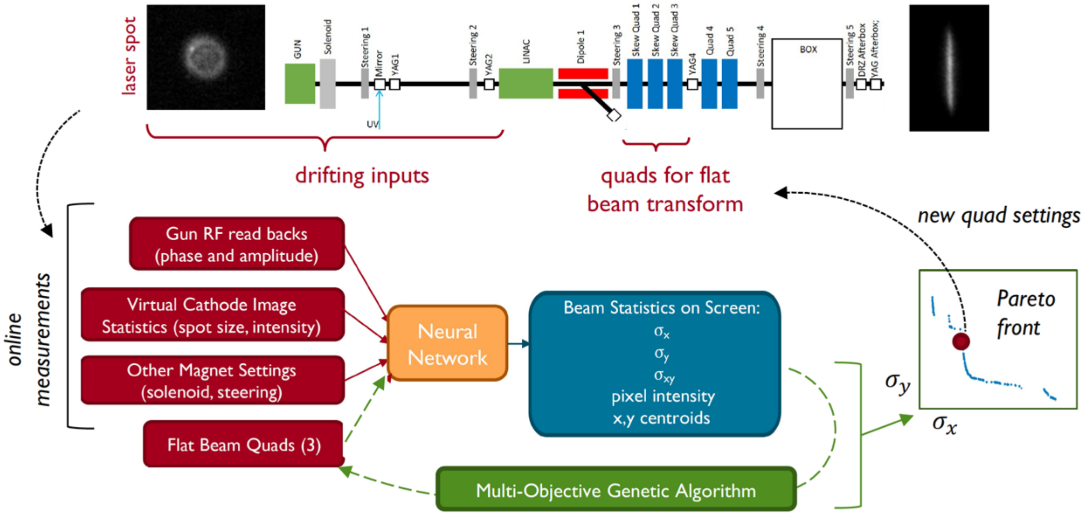
- Round-to-flat beam transforms are challenging to optimize → 2019 study explored ability of a learned model to help
- Trained neural network model to predict fits to beam image, based on archived data
- Tested online multi-objective optimization over model (3 quad settings) given present readings of other inputs
- Used as warm start for other optimizers
- Trained DDPG Reinforcement Learning agent and tested on machine under different conditions than training



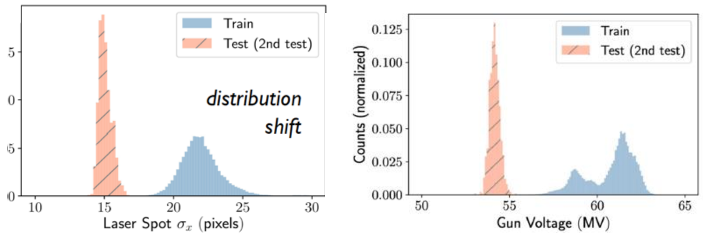
Hand-tuning in seconds vs. tens of minutes

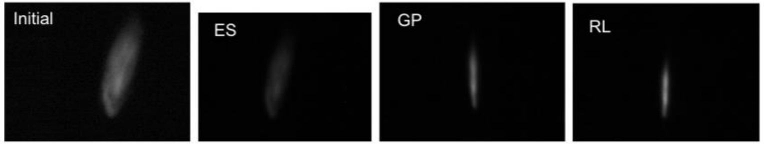
Boost in convergence speed for other algorithms

Example: Compensate for Upstream Drift in Fast Setup



Can work even under distribution shift





- Round-to-flat beam (RTFB) transforms are challenging to optimize; sensitive to upstream drift (e.g. in laser, rf systems)
  → *want to be able to set up RTFB quickly despite drift*

- 2019 study explored ability of a learned model and tuning algorithms to help

- NN model used as warm start for BO, extremum seeking, hand-tuning

- Trained DDPG Reinforcement Learning agent on NN model and tested on machine under different conditions

RL agent converged faster/more smoothly than BO

→ *Broadly similar problem (at different scale) for LCLS/FACET-II switching between setups*

Reinforcement Learning

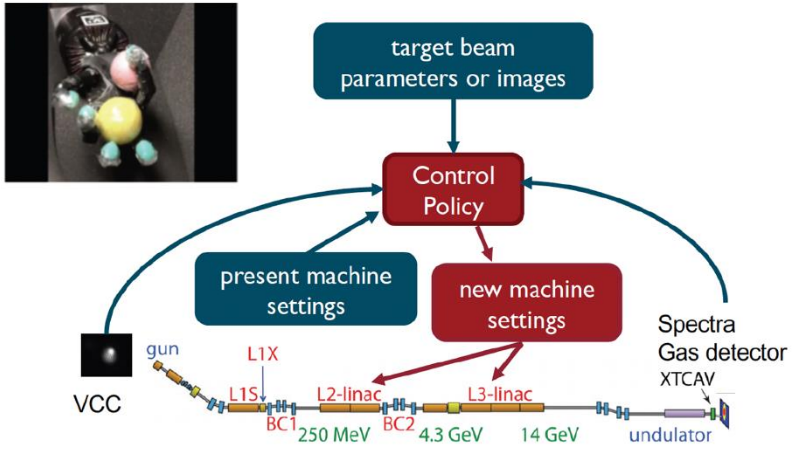
RL can help address a different set of needs than BO:

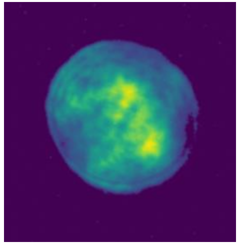
- Use global machine information, more historical data
- Treat as a dynamical system *(many time-dependent processes/feedbacks + drift)*
- Address demands for fast dynamic control from users

Suitability of accelerator tuning problems for RL:

- Many variables, multi-modal signals (images, scalars, time series)
- Continuous state/action spaces (similar to robotics)
- Have physics models/simulators for many problems

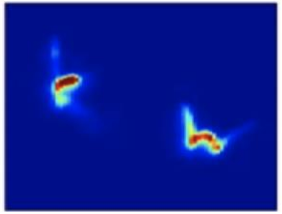
Nagabandi, et al., 2019



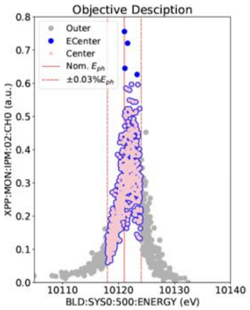


x-y laser

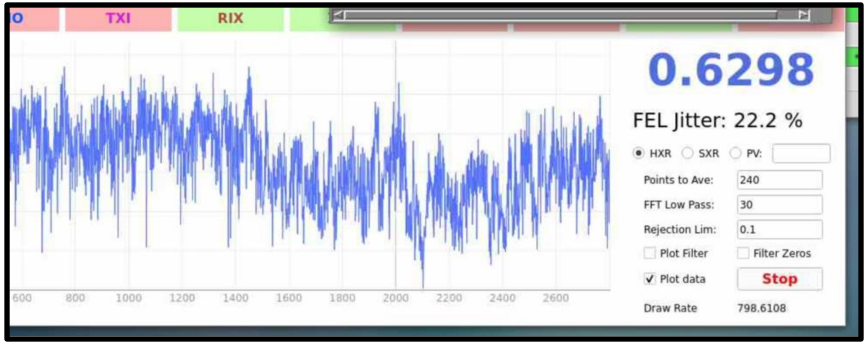
energy



time



Variety of high dimensional signals for states, objectives



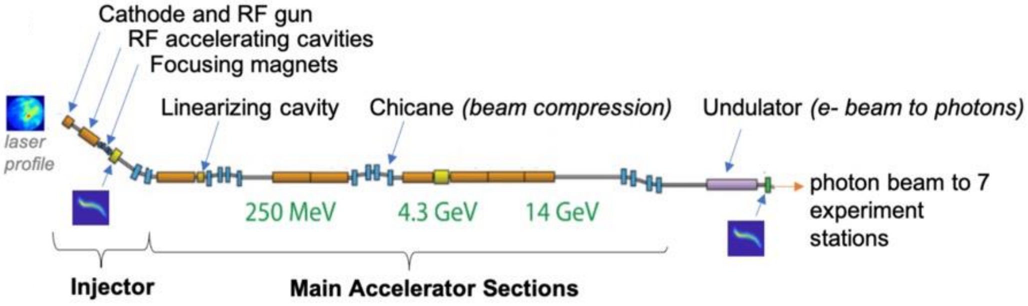
120 Hz FEL pulse intensity

Nonlinear instability → sensitive to dynamic processes

(e.g. trajectory feedback, cooling, LLRF control)

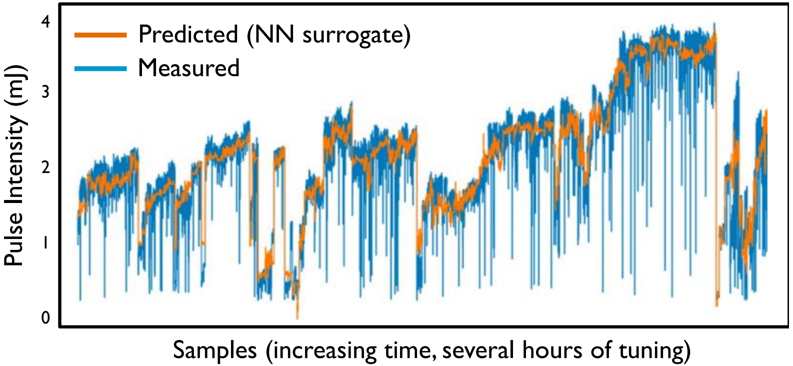
Reinforcement Learning

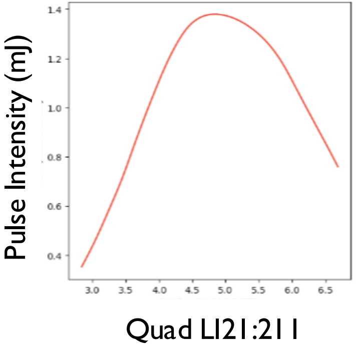
- FEL is sensitive to focusing, trajectory; perturbing beam/feedbacks too much results in beam losses
- Using data-driven surrogates and differentiable sims to train agents
- Iteratively add more data, targets and variables:
  - Photon pulse intensity
  - Beam phase space images, spectra
  - Focusing magnets, RF cavities, undulator



~28 focusing magnets for FEL pulse intensity

(many more variables to include: steering, rf cavities, undulator, drive laser)





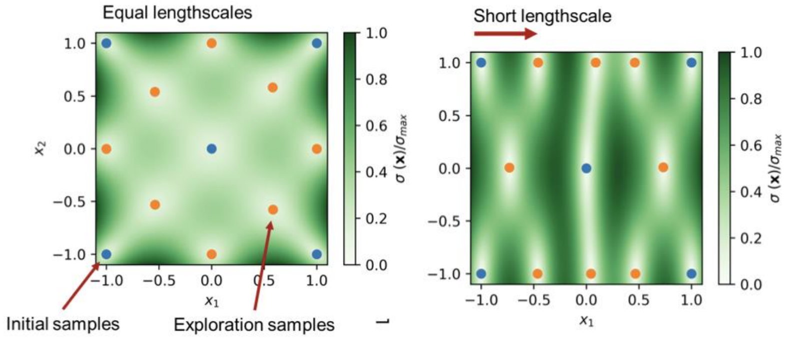
Efficient Characterization with Bayesian Exploration

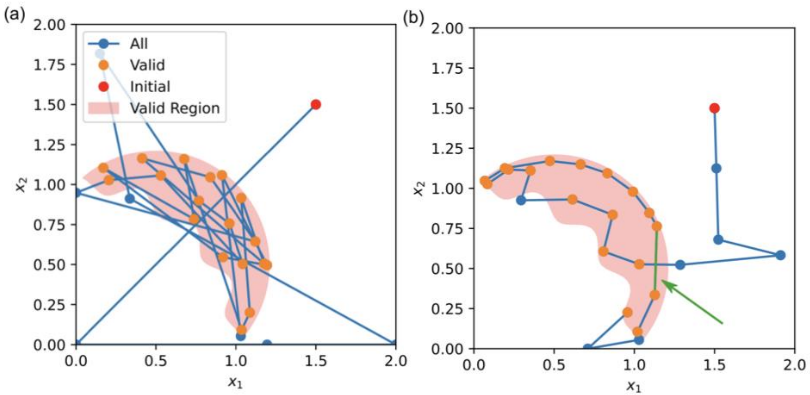
$$\alpha(\boldsymbol{x}) = \sigma(\boldsymbol{x}) \prod_{i=1}^{N} p_i(g_i(\boldsymbol{x}) \geq h_i)\, \Psi(\boldsymbol{x}, \boldsymbol{x_0})$$

R. Roussel et. al.
Nat. Comm. **2021**

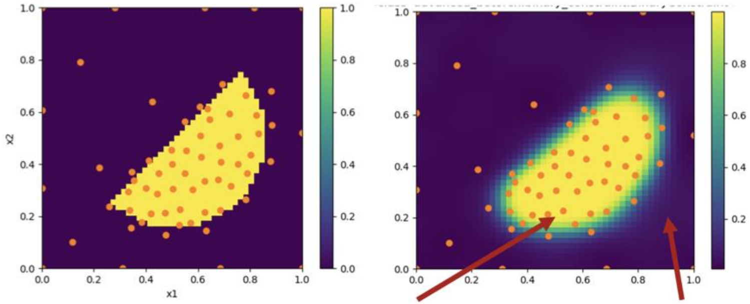
proximal biasing

adaptive sampling





learning constraints



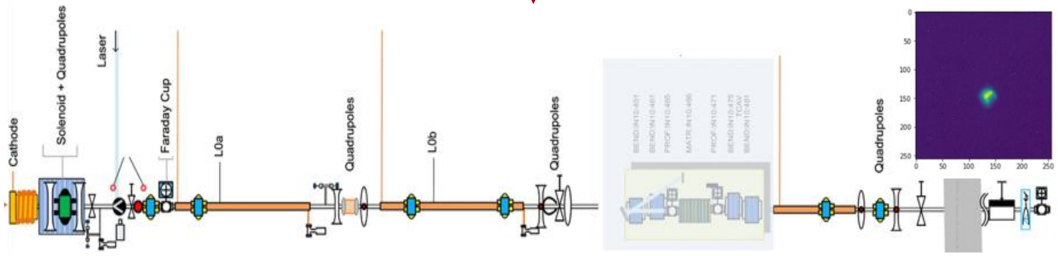
Enables sample-efficient characterization of high-dimensional spaces, while respecting both input and output constraints

Bayesian Exploration for Efficient Characterization

Automatic Exploration
(constrained to useful values of emittance and match)

Comprehensive ML Models of Injector

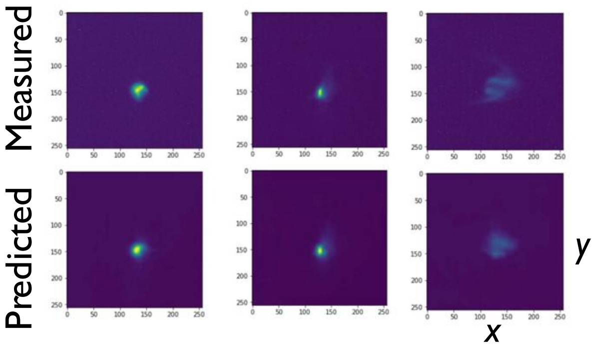
Setting changes on 10 variables (solenoid, bucking coil, corrector quads and matching quads)



FACET-II Injector

x-y emit, match, and beam images

transverse phase space



https://www.nature.com/articles/s41467-021-25757-3

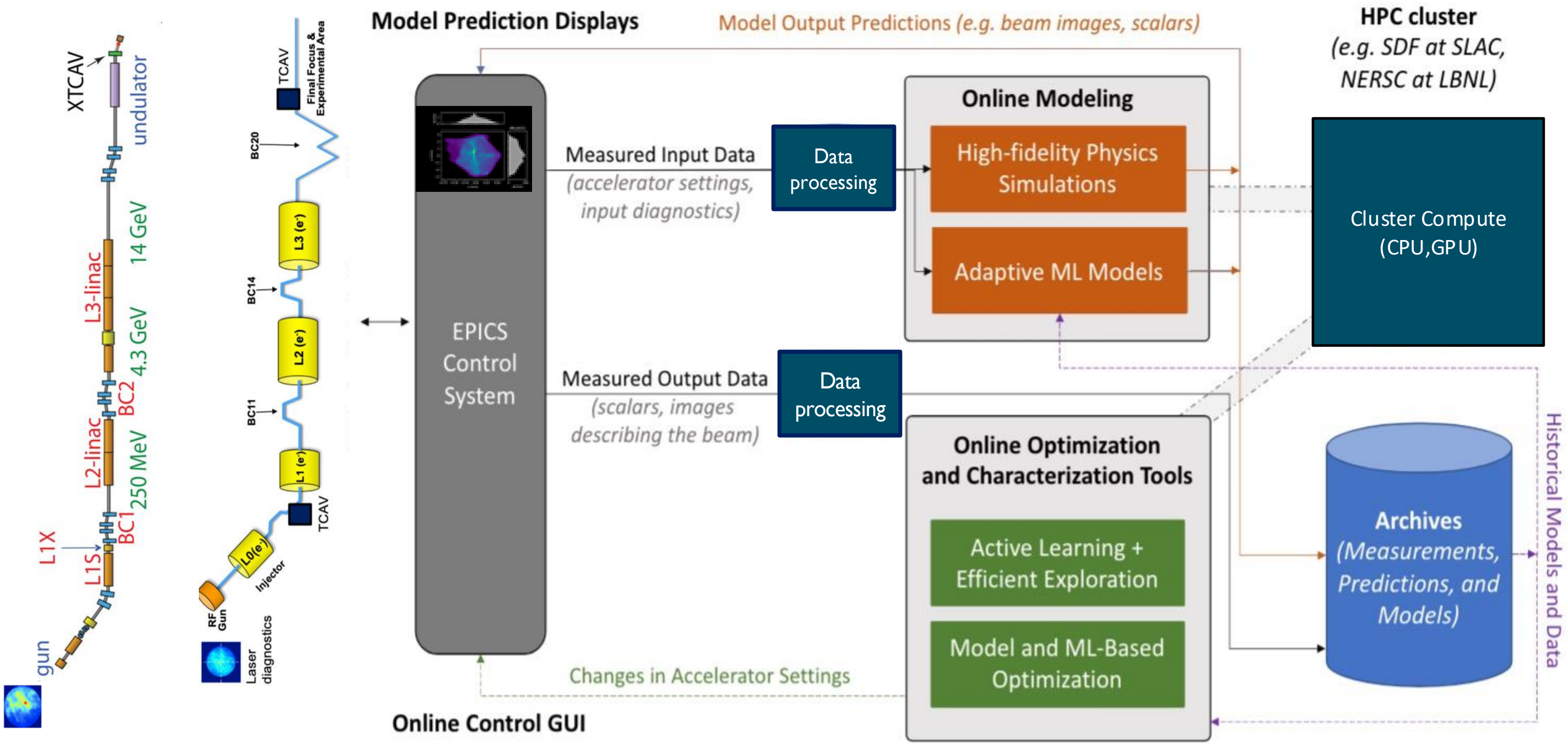
- Used Bayesian Exploration for efficient high-dimensional characterization (10 variables) of emittance and match at 700pC: **2 hrs for 10 variables compared to 5 hrs for 4 variables with N-D parameter scan (~8x faster)**
- Data was used to train neural network model of injector response predicting x-y beam images. GP ML model from exploration predicts emittance and match.
- **Example of integrated cycle between characterization, modeling, and optimization → now want to extend to larger system sections and new setups**

Use of Bayesian exploration to generate training data was sample-efficient, reduced burden of data cleaning, and resulted in a well-balanced distribution for the training data set over the input space. ML models were immediately useful for optimization.

Goal: Full Integration of AI/ML Optimization, Data-Driven Modeling, and Physics Simulations

*Working on a **facility-agnostic*** ecosystem for online simulation, ML modeling, and AI/ML driven characterization/optimization

Will enable system-wide application to aid operations, and help drive AI/ML development *(e.g. higher dimensionality, robustness, combining algorithms efficiently)*



Making good progress toward this vision with open-source, modular software tools

Argonne NATIONAL LABORATORY

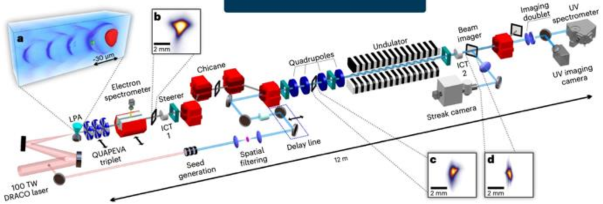
SLAC NATIONAL ACCELERATOR LABORATORY

Fermilab

Xopt

Badger GUI interface

BERKELEY LAB



Brookhaven National Laboratory

ESRF The European Synchrotron

DESY.

Will work closely with UH and Prof. Siqi Li to adapt to UH machine and explore new algorithmic approaches!

Roussel, et al. IPAC 2023 THPL164

https://github.com/xopt-org/Xopt

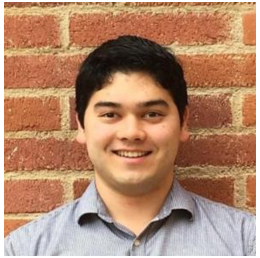
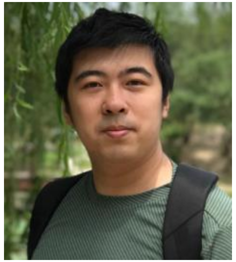
https://github.com/xopt-org/Badger

Common software tools (Xopt, Badger) enables rapid transfer between facilities and algorithmic progress
Also working to link accelerator and photon beamline tuning (e.g. BlueSky integration)

Thanks for your attention!
Any questions?










Thanks to the core team at SLAC working on various AIML technologies and infrastructure!

Thanks to many other collaborators not shown!

Backups

Existing Capabilities and Software

Many capabilities can be readily adapted to new cases

ML-based tuning (Xopt)

- *Learned output constraints*
- *Information-based sampling (characterization)*
- *Trust region optimization*
- *Multi-objective optimization*
- *Beam alignment through optics components*
- *Hysteresis-aware tuning*
- *Physics/ML system models to speed up ML-based tuning (Priors, expected correlations, etc)*

Graphical User Interface (Badger)

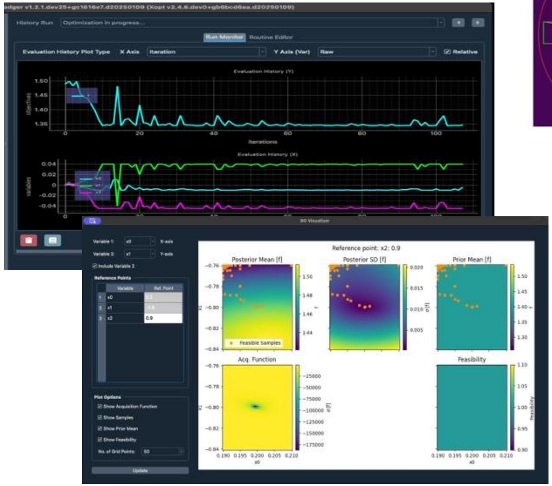
- *Modular backend*
- *Easy to select variables, objectives, constraints and algorithm*
- *Algorithm progress and model visualization*

Digital twin infrastructure

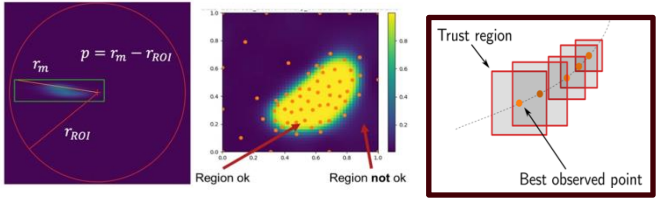
- *Adaptive ML model wrapping and deployment (lume-model)*
- *Physics and ML model deployment workflow using Kubernetes and Prefect (includes S3DF deployment)*

S3DF integration with control system

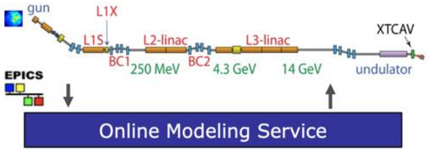
- *Simple I/O from batch jobs*
- *Kubernetes for long-running jobs*

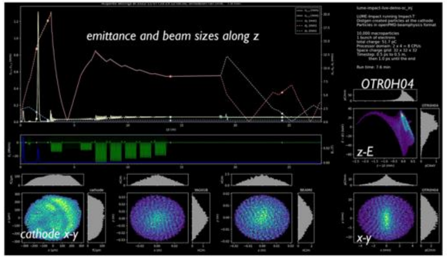


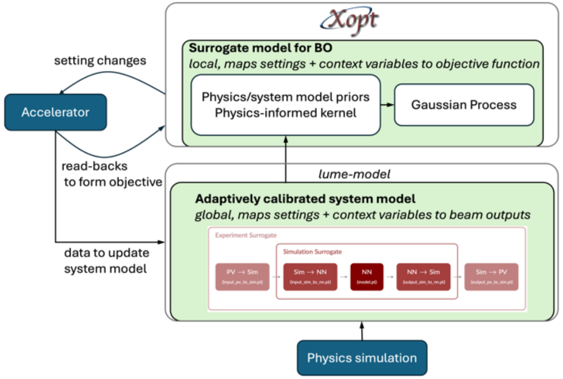
Xopt *Open-source software for AI/ML tuning*



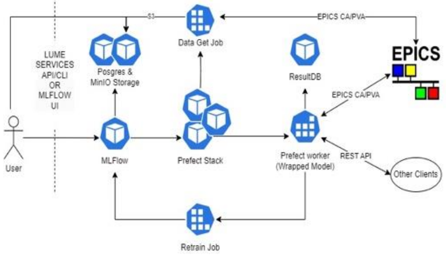
Robust algorithms for commonly-encountered issues







Integration of adaptive system models with ML-based control

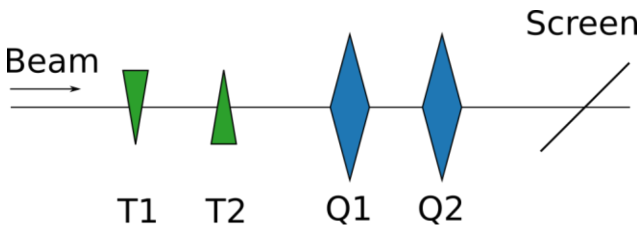


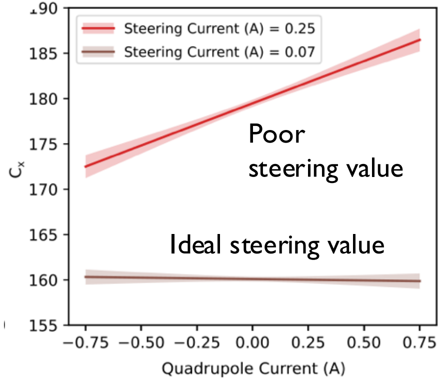
Digital twin infrastructure (local and S3DF)

https://github.com/xopt-org/
https://www.lume.science/

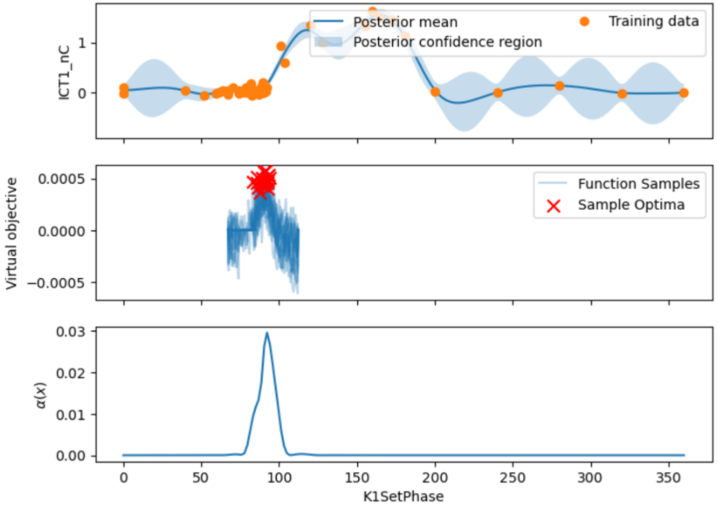
Further Automation

- Chaining together automation of sub-tasks and measurements
- RF /laser timing scans, beamline alignment, smart sampling for measurements

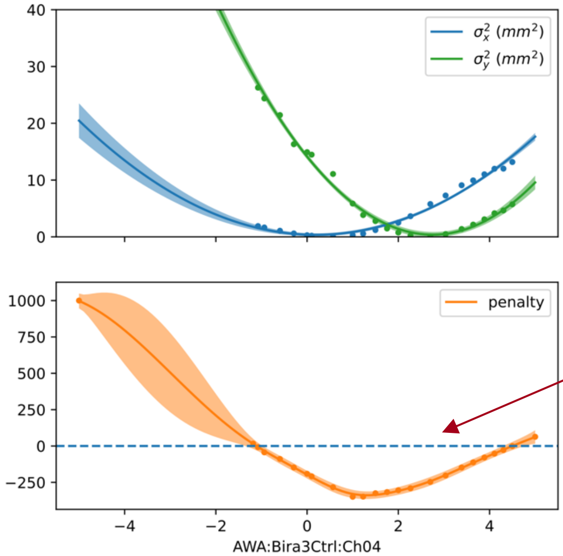




Automated beam alignment
→ 20-30 minutes by hand
→ 5 minutes with BAX



Automated determination of gun phase with BAX



Smart sampling for emittance measurements with Bayesian Exploration

Beam bounding box penalty

R. Roussel, D. Kennedy

Deployment: Xopt and Badger

Xopt: houses optimization algorithms

```yaml
xopt:
    max_evaluations: 6400

generator:
    name: cnsga
    population_size: 64
    population_file: test.csv
    output_path: .

evaluator:
    function: xopt.resources.test_functions.tnk.evaluate_TNK
    function_kwargs:
      raise_probability: 0.1

vocs:
    variables:
        x1: [0, 3.14159]
        x2: [0, 3.14159]
    objectives: {y1: MINIMIZE, y2: MINIMIZE}
    constraints:
        c1: [GREATER_THAN, 0]
        c2: [LESS_THAN, 0.5]
    linked_variables: {x9: x1}
    constants: {a: dummy_constant}
```

```python
# create Xopt object.
X = Xopt(YAML)

# take 10 steps and view data
for _ in range(10):
    X.step()

X.data
```

Python interface

Many optimization algorithms

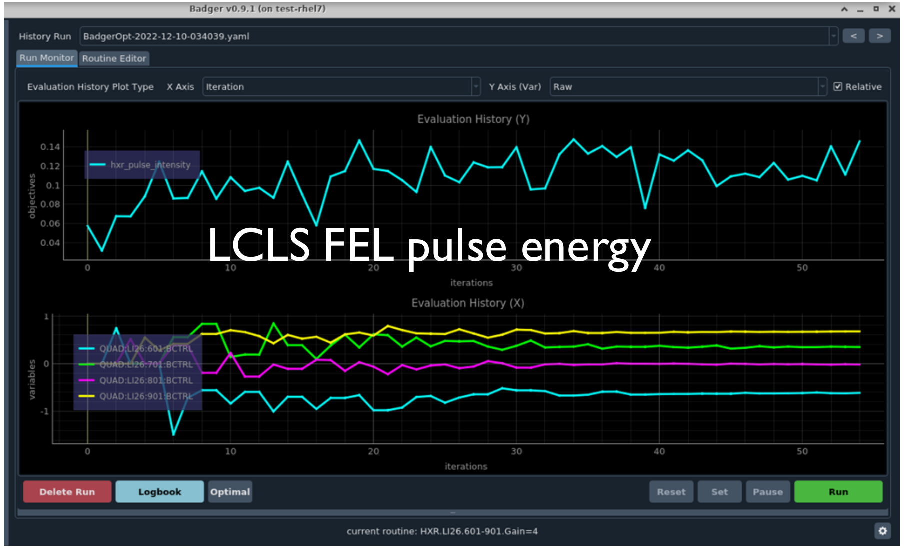
- *Genetic algorithms (NSGA-II, etc.)*
- *Nelder-Mead Simplex*
- *Bayesian Optimization*
- *Bayesian Exploration*
- *Trust-region BO*
- *Learned output constrained BO*
- *Interpolating BO*

Badger GUI interface

User interface, I/O with machine

https://github.com/xopt-org/Xopt
https://github.com/xopt-org/Badger

→ Has been used for online optimization at numerous facilities (LCLS/LCLS2, FACET-II, ESRF, AWA, NSLS-II, FLASHForward)
→ Working to make interoperable with other software (e.g. Gymnasium)



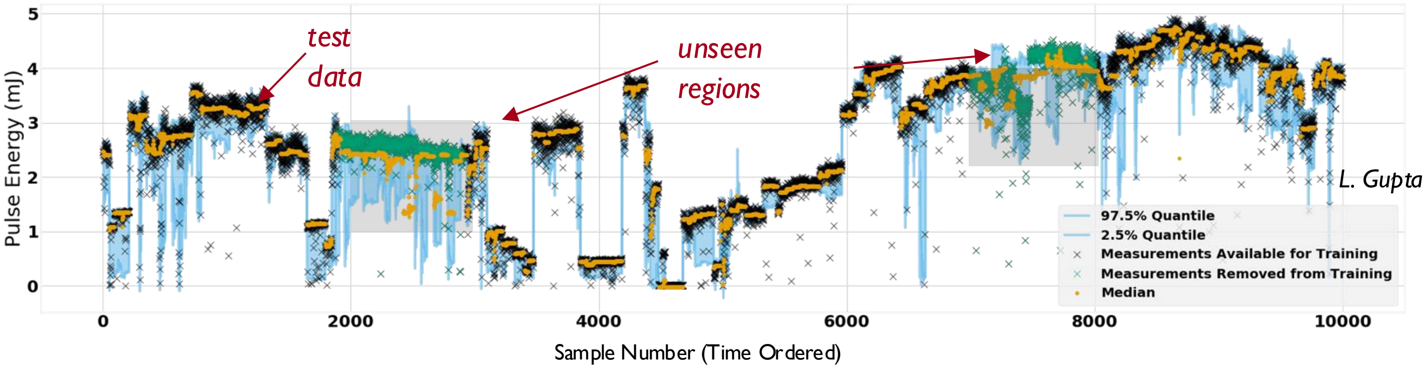
0.04 to 0.14 mJ in SXR → 15% better than hand-tuning

41hr → best lifetime observed ever (in record speed of 15 minutes)
injection efficiency improved by 5%

- Can specify constraints on settings and outputs (e.g. avoid dark current, beam losses, etc)
- Trust-region method allows conservative high-dimensional tuning (e.g. used >100 sextupoles at ESRF)
- Working on integrating global model priors → not learning from scratch each time
- Working to make compatible with RL problems + gymnasium

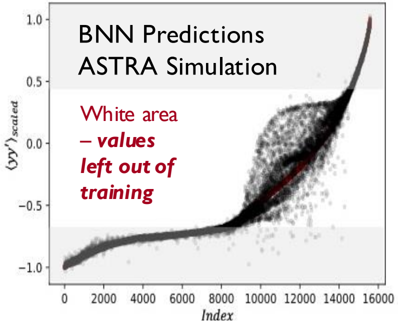
Uncertainty Quantification / Robust Modeling

Essential for decision making under uncertainty (e.g. safe opt., intelligent sampling, virtual diagnostics)



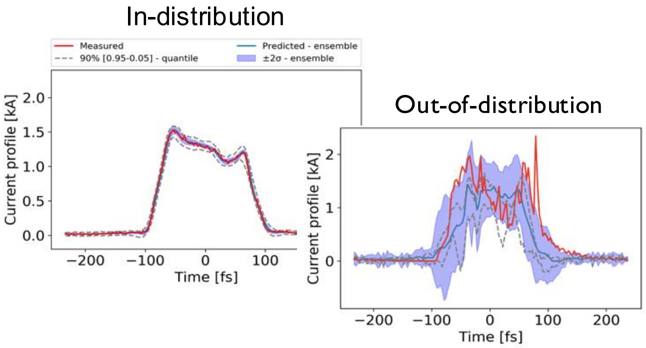
Current approaches
- Ensembles
- Gaussian Processes
- Bayesian NNs
- Quantile Regression

Neural network with quantile regression predicting FEL pulse energy at LCLS



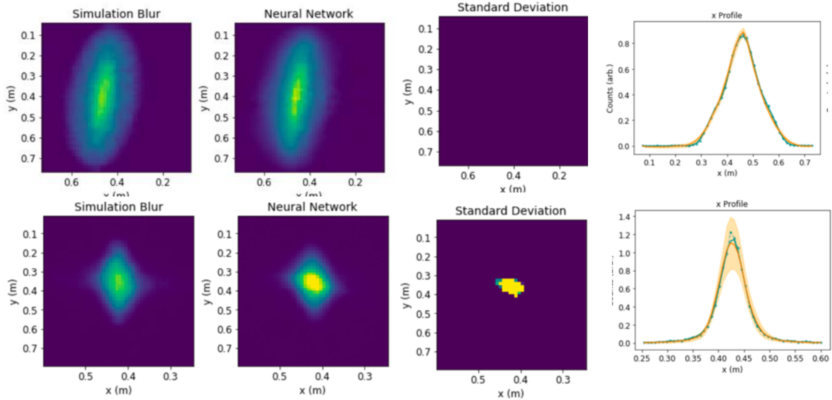
Scalar parameters for the LCLS-II injector (Bayesian neural network)

A. Mishra et. al., PRAB, 2021



longitudinal phase space (quantile regression + ensemble)

O. Convery, et al., PRAB, 2021

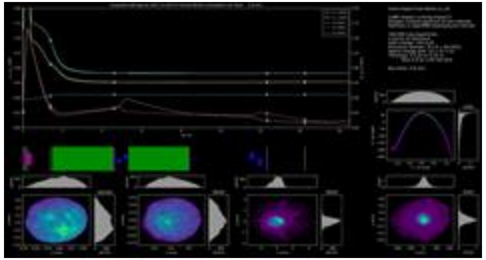


LCLS injector transverse phase space (ensemble)

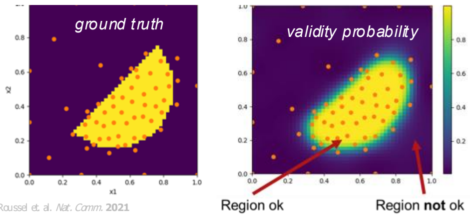
Broad Research Program at SLAC in AI/ML for Accelerators

(1) Developing new approaches for accelerator optimization/characterization and faster higher-fidelity system modeling, (2) developing portable software tools to support end-to-end AI/ML workflows, (3) helping integrating these into regular use

Online prediction with physics sims and **fast/accurate ML system models**



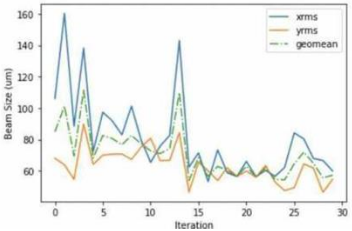
Efficient, safe optimization algorithms



Roussel et. al. *Nat. Comm.* 2021

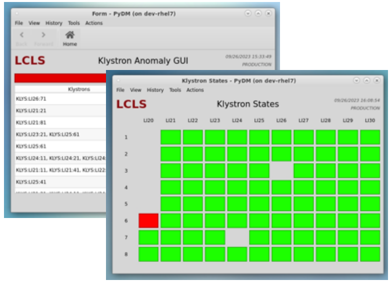
Output constraints learned on-the-fly

Adhere to constraints and balance multiple targets

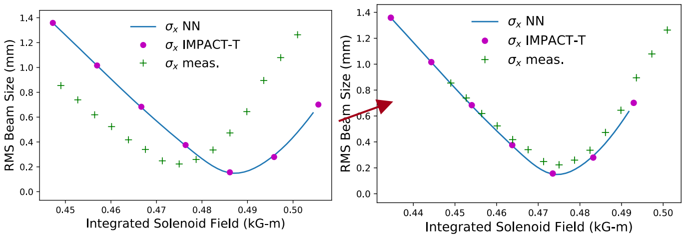


Challenging problems: e.g. sextupole tuning

Anomaly detection

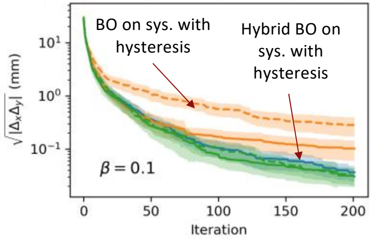


Adaptation of models and **identification of sources of deviation** between simulations and as-built machine



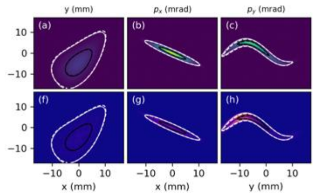
Combining physics and ML for better performance

Hysteresis-aware optimization



Roussel et. al. PRL. 2022

Differentiable simulations + ML for 6D phase space reconstruction

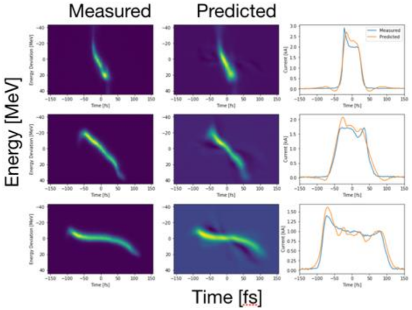


Roussel et. al. PRL. 2023

ML-enhanced diagnostics

Rapid analysis/virtual diagnostics

Shot-to-shot predictions at beam rate



C. Emma, et al. – PRAB **21**, *112802 (2018)*

Many solutions put into reusable open-source software (e.g. Xopt/Badger) demoed at many facilities

AI/ML enables fundamentally new capabilities across a broad range of applications → highly promising from initial demos.

Digital Twin Infrastructure

Ecosystem of modular tools (can use independently)

LUME – simulation interfaces/wrappers in Python

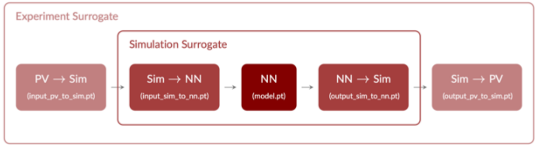
lume-model – wraps ML models, facilitates calibration

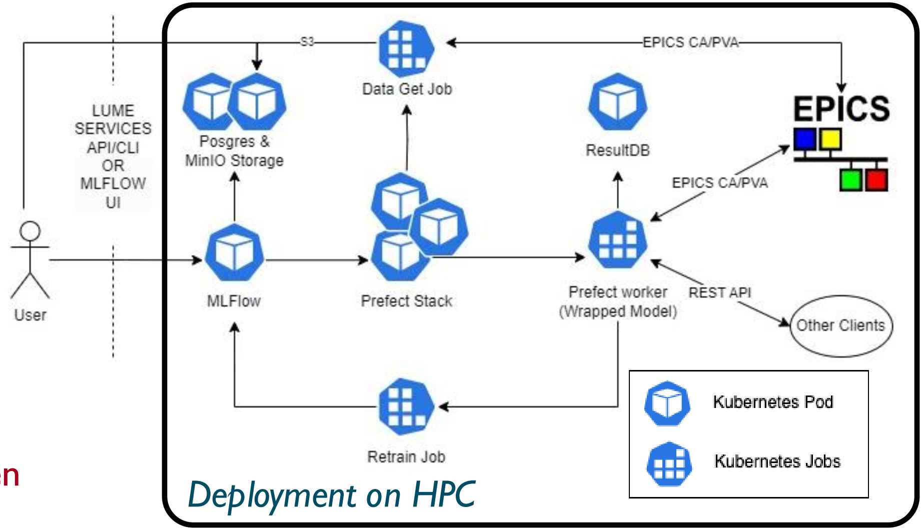
lume-services – online model deployment and orchestration

distgen – flexible creation of beam distributions

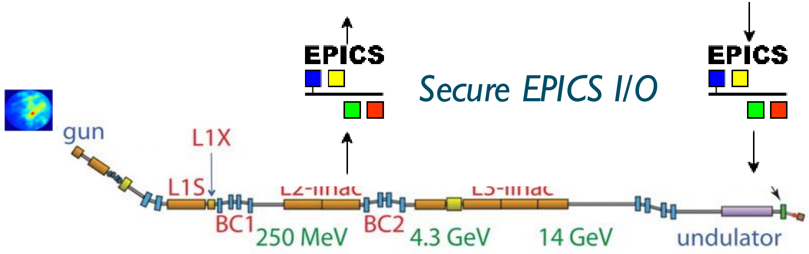
Integration with MLFlow for MLOps

https://www.lume.science/





- Live physics simulations and ML models now linked between SLAC's HPC system (S3DF) and control system
  → *run with Kubernetes and Prefect*
- Working with NERSC to swap between S3DF/NERSC resources
- Beginning work on MLOps aspects that will be used in continual learning research



Substantial progress on deploying ML and Physics-based models and integrating with HPC in a portable way

Modular, Open-Source Software Development

Community development of **re-usable, reliable, flexible software tools** for AI/ML workflows has been essential to maximize return on investment and ensure transferability between systems

Modularity has been key: separating different parts of the workflow + using shared standards

Different software for different tasks:

Optimization algorithm driver *(e.g. Xopt)*

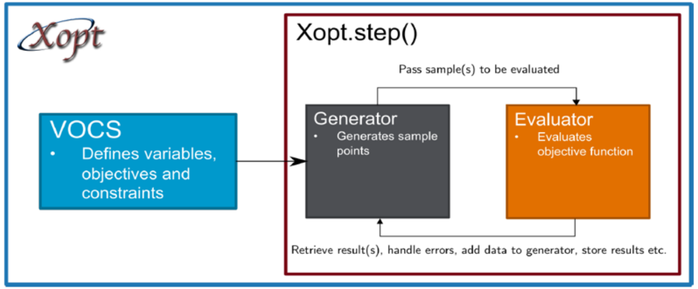
Visual control room interface *(e.g. Badger)*

Simulation drivers *(e.g. LUME)*

Standards model descriptions, data formats, and software interfaces *(e.g. openPMD)*

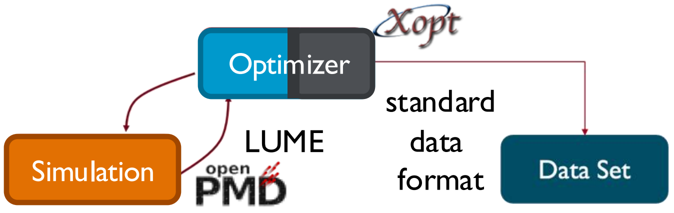
Online model deployment *(LUME-services)*

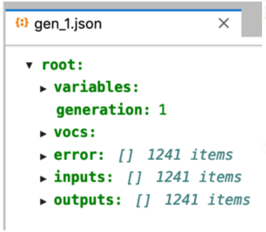
More details at https://www.lume.science/

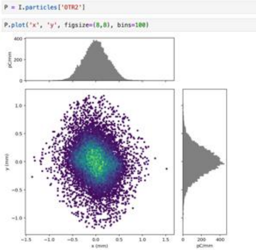


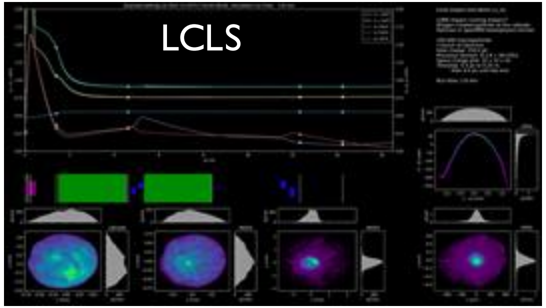

```
vocs:
  name: TNK_test
  variables:
    x1: [0, 3.14159]
    x2: [0, 3.14159]
  objectives: {y1: MINIMIZE}
  constraints:
    c1: [GREATER_THAN, 0]
    c2: ['LESS_THAN', 0.5]
```

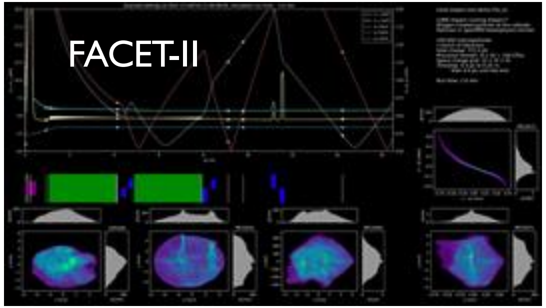
```
algorithm:
  name: bayesian_exploration
  options:
      n_initial_samples: 5
      n_steps: 25
      generator_options:
          batch_size: 1
          #sigma: [[0.01, 0.0],
          use_gpu: False
```











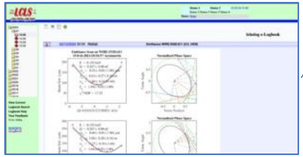
Online Impact-T simulation and live display; trivial to get running on FACET-II using same software tools as the LCLS injector

Modular open-source software has been essential for our work.

SLAC Pursuing AIML for Accelerators Very Broadly

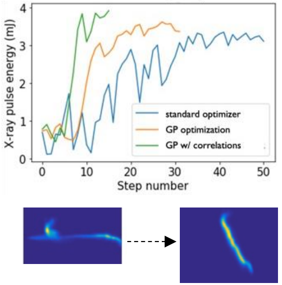
Human-computer interaction

Language modeling / multi-modal data
(e.g. electronic logbook)

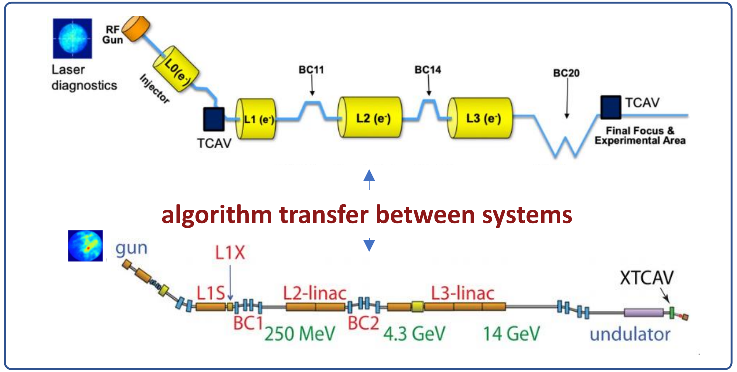


Data reduction/rejection *(kHz/MHz data streams)*
Event triggering

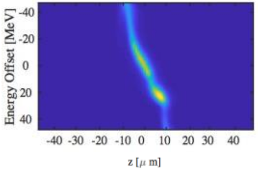
Automated control + optimization



J. Duris et al., PRL, 2020

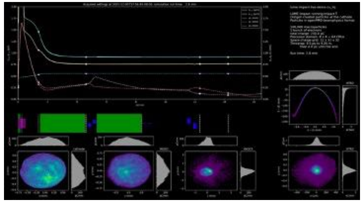


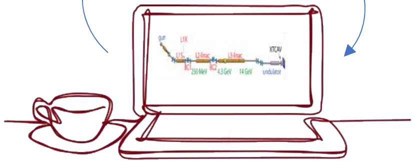
ML-enhanced diagnostics
(provide insight at faster rate, at higher resolution, non-invasively)



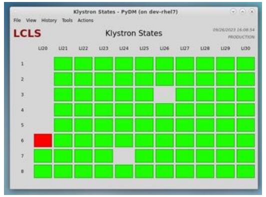
C. Emma et al., PRAB, 2018

Anomaly detection failure prediction
(plan maintenance; alert to changes in machine; alert to interesting science)





Extract unknown relationships + correlations
(feed into future control / design)

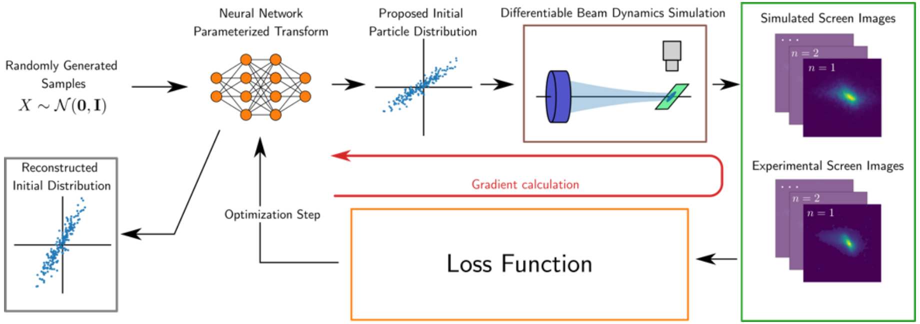


Digital twins + online modeling
(fast sims, differentiable sims, model calibration, model adaptation)

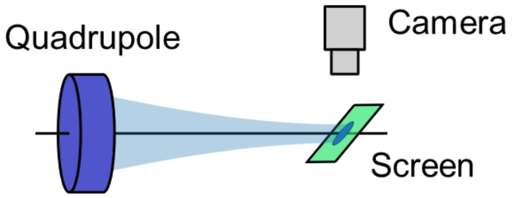
+ need uncertainty quantification for all
+ can incorporate physics information in all

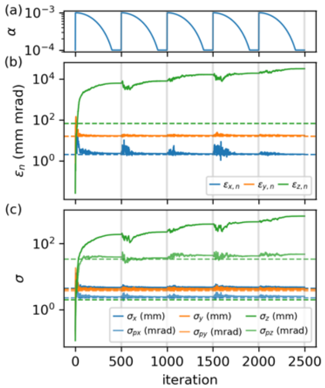
Phase Space Reconstruction with Differentiable Tracking Simulations

Differentiable pipeline for reconstructing 6D phase space distribution using neural network parameterization

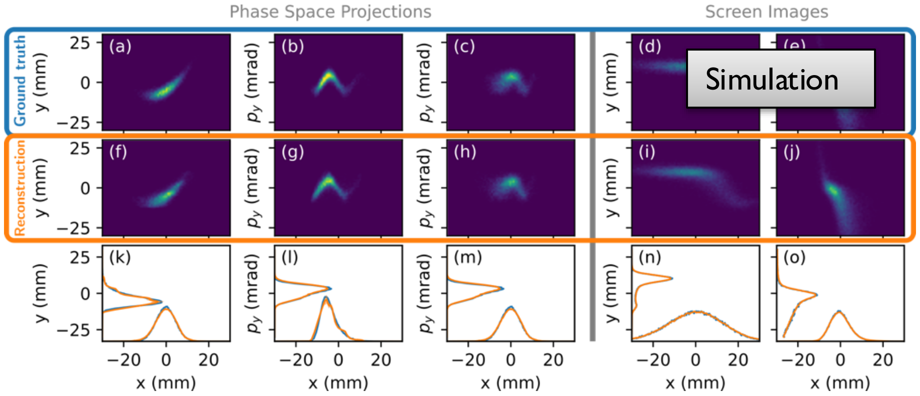


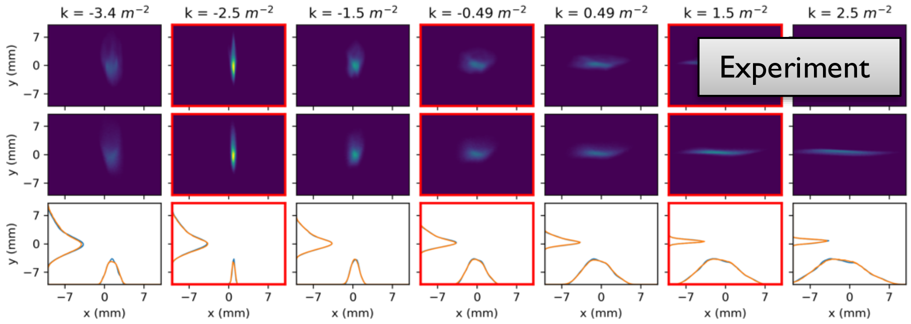
Reconstruct 4D phase space distribution + approx. energy spread from simple beamline diagnostic and 10 measurements





Confidence estimates

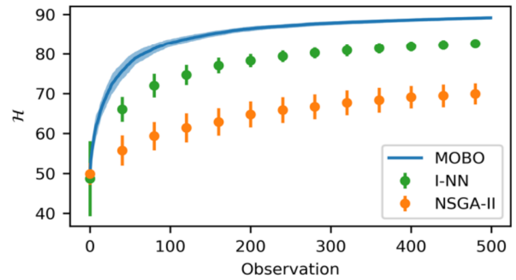


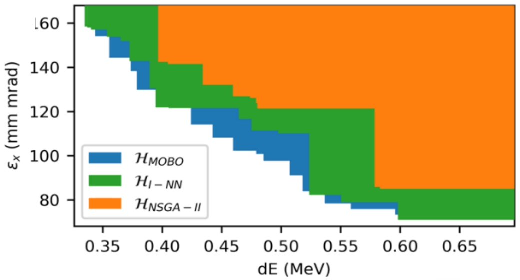


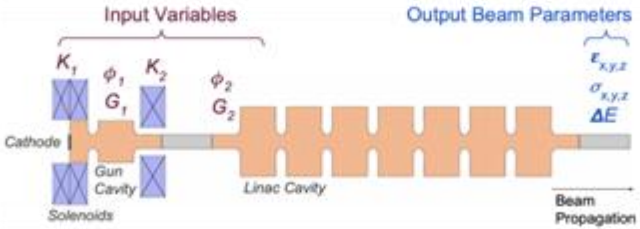
ML combined with differentiable simulations opens up a new paradigm for constructing detailed phase space diagnostics in a way that is computationally-efficient and sample-efficient

Multi-Objective Bayesian Optimization

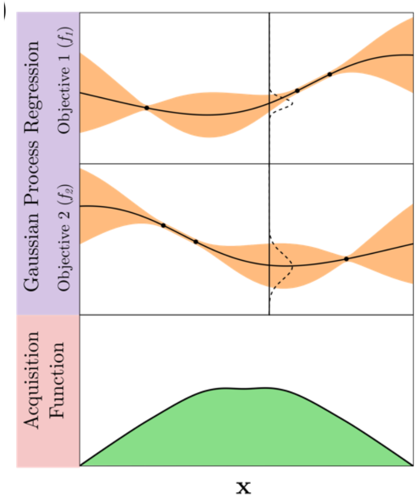
Roussel et. al. PRAB **2021**

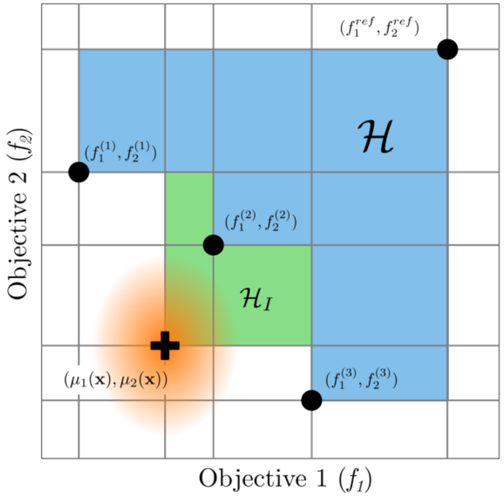


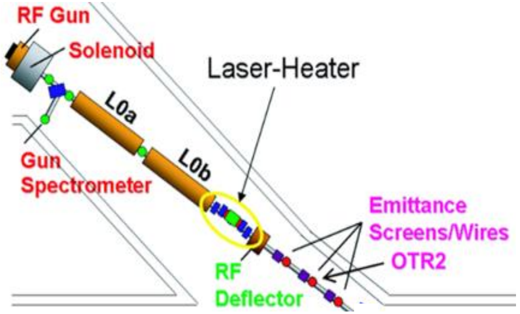


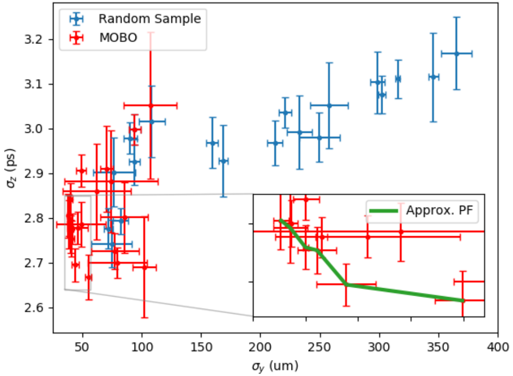


Simulation study with the AWA injector







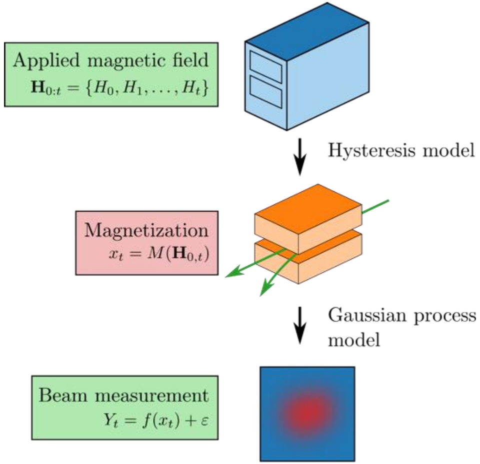


Experimental demo with the LCLS injector

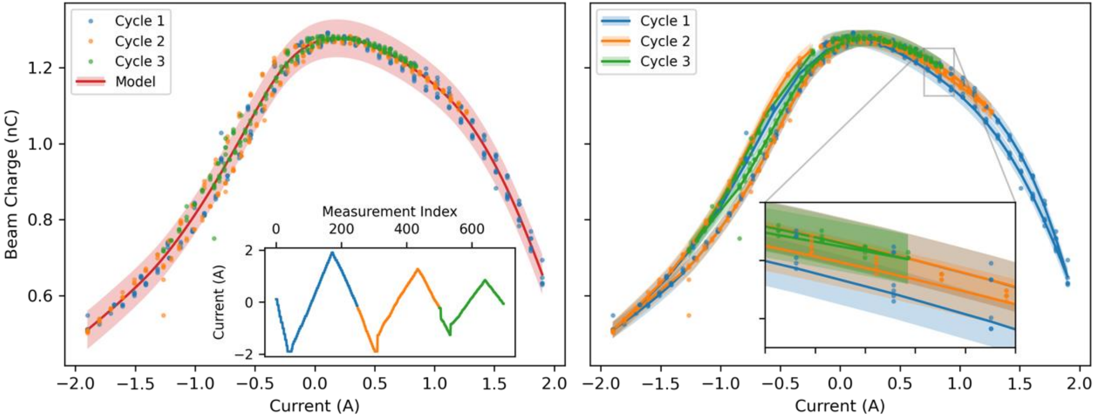
Multi-objective Bayesian optimization enables efficient, direct examination of experimental tradeoffs

Addressing Magnetic Hysteresis with Differentiable Physics Models

Learn both hysteresis properties and beam response simultaneously using two step modeling

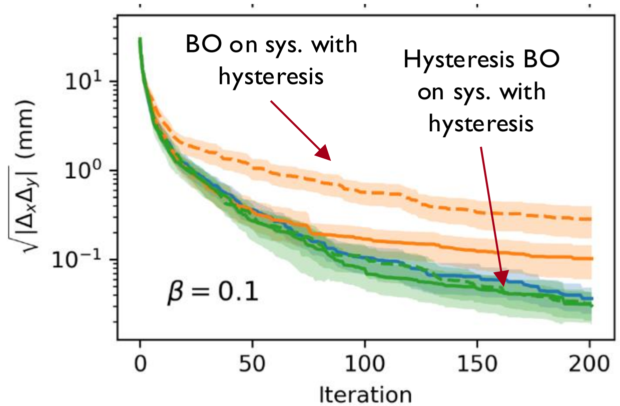


R. Roussel, et. al. Phys. Rev. Lett. **128**, 204801



Modeling accuracy increases

Optimization performance increases

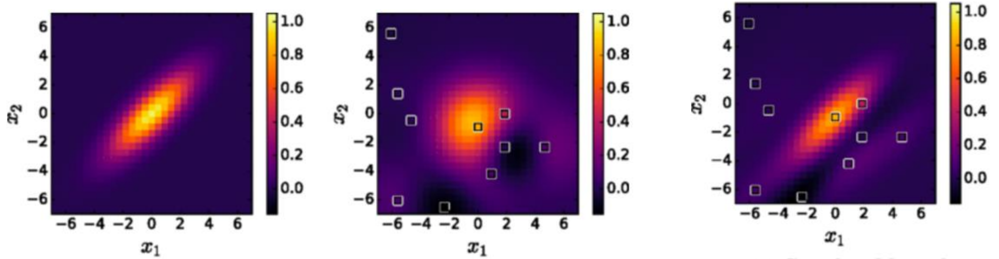


Hysteresis-aware BO efficiently solves long-standing issues with accelerator optics tuning

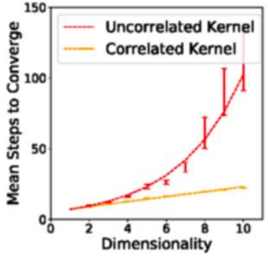
Physics-Aware Bayesian Optimization: Correlated Kernel

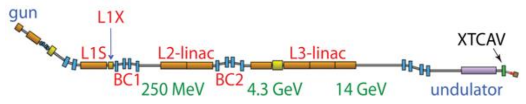
J. Duris et al., PRL, 2020
A. Hanuka, et al., PRAB, 2021

→ Design Gaussian Process kernel from expected correlations between inputs (e.g. quadrupole magnets)



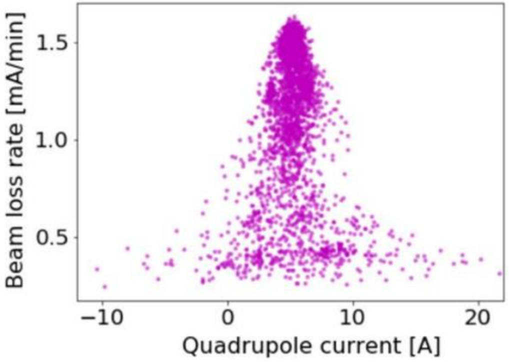
(a) Ground truth (b) Isotropic kernel (c) Correlated kernel

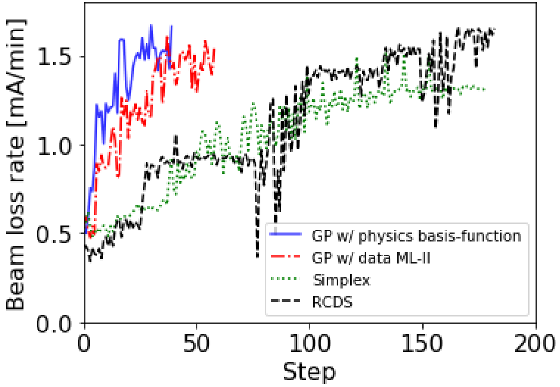


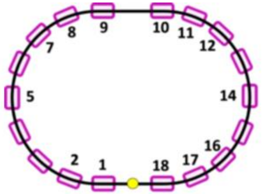


FEL tuning @LCLS

→ Take the Hessian of model at expected optimum to get the correlations







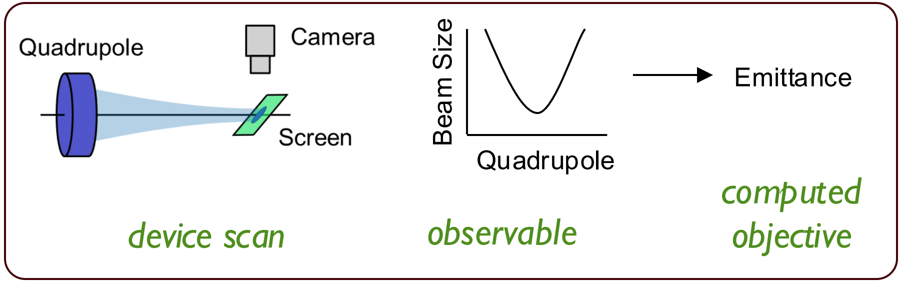
vertical emittance tuning @SPEAR3

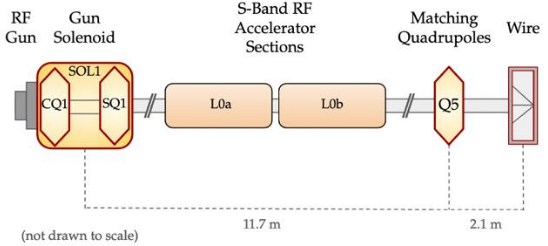
No measured data needed ahead of time, just a physics model of system

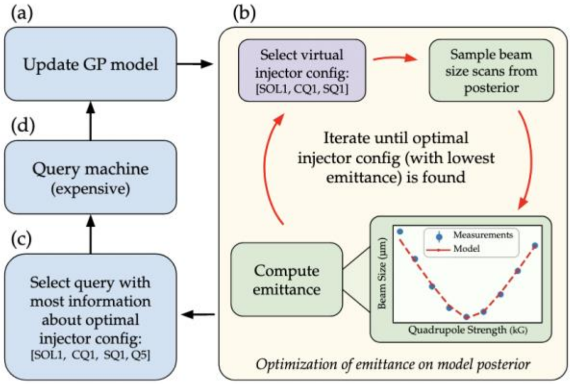
Including correlation between inputs enables increased sample-efficiency and results in faster optimization
→ kernel-from-Hessian enables easy computation of correlations even in high dimension

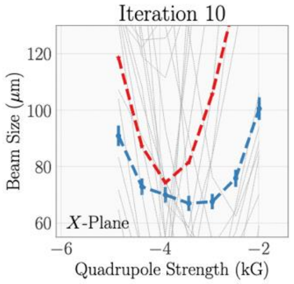
Optimization with Virtual Objectives

- Many objectives require layered scans or optimization problems
- Instead learn model from scratch online and do scan on model
- Bayesian Algorithm Execution (BAX) → 20x speedup in tuning

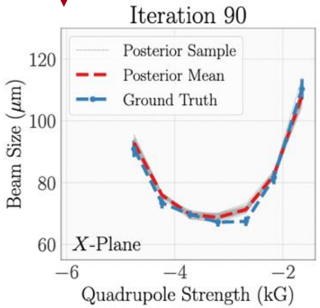


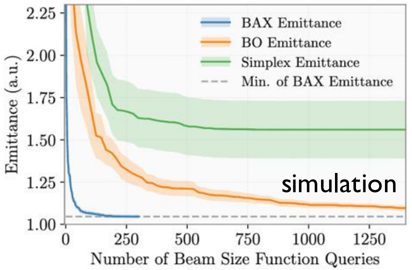


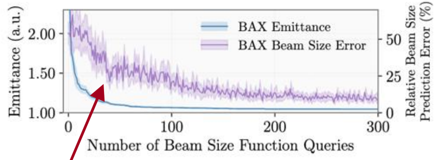




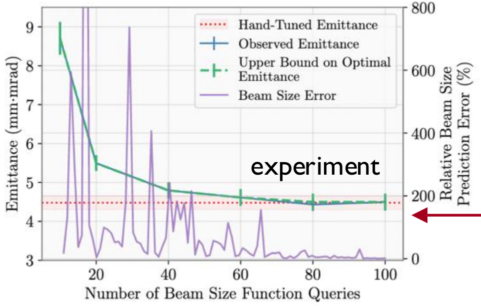
model is learned on-the-fly







Convergence of beam size prediction error gives practical indicator of convergence



20x faster tuning than standard BO, equivalent or better solution than hand-tuning

S. Miskovich, MLST, 2024

BAX enables a paradigm shift in how optimization problems with complicated scans or other indirect measurements are handled