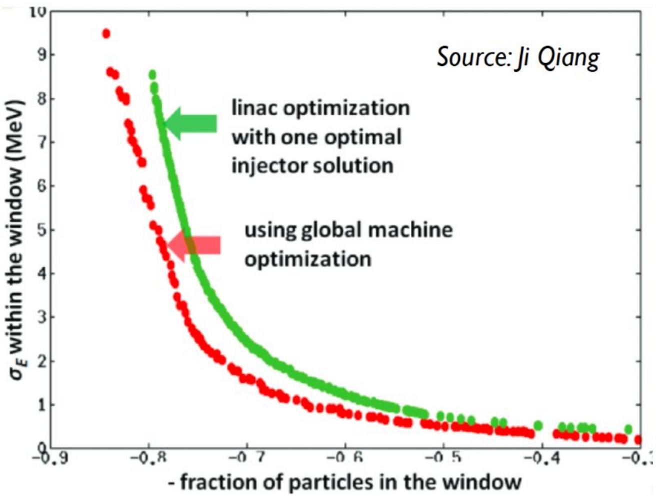
E331 - Neural network based tuning to exploit machine-wide sensitivities in pursuit of high beam quality

FACET-II AARD Long Term Planning Meeting
August 22, 2024

Major limitations in the way accelerator tuning is done:

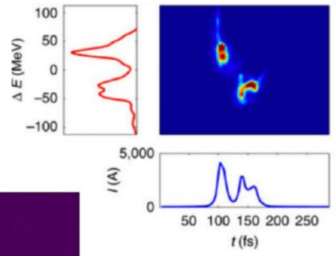
- Piecemeal tuning of subsystems *(known to be sub-optimal)*
- Indirect use of high-dimensional diagnostics *(e.g. images)*
- Often a lack of accurate online models

  → *Potentially limiting factors in control of extreme beams*

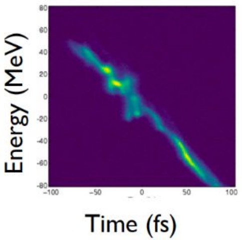


More global view can enable better control:

- Fully exploit unknown system-wide sensitivities + nonlinearities
- Faster switching between setups *(if using global representation of machine)*
- Better handling of parameter tradeoffs *(e.g. jitter, matching, longitudinal phase space)*

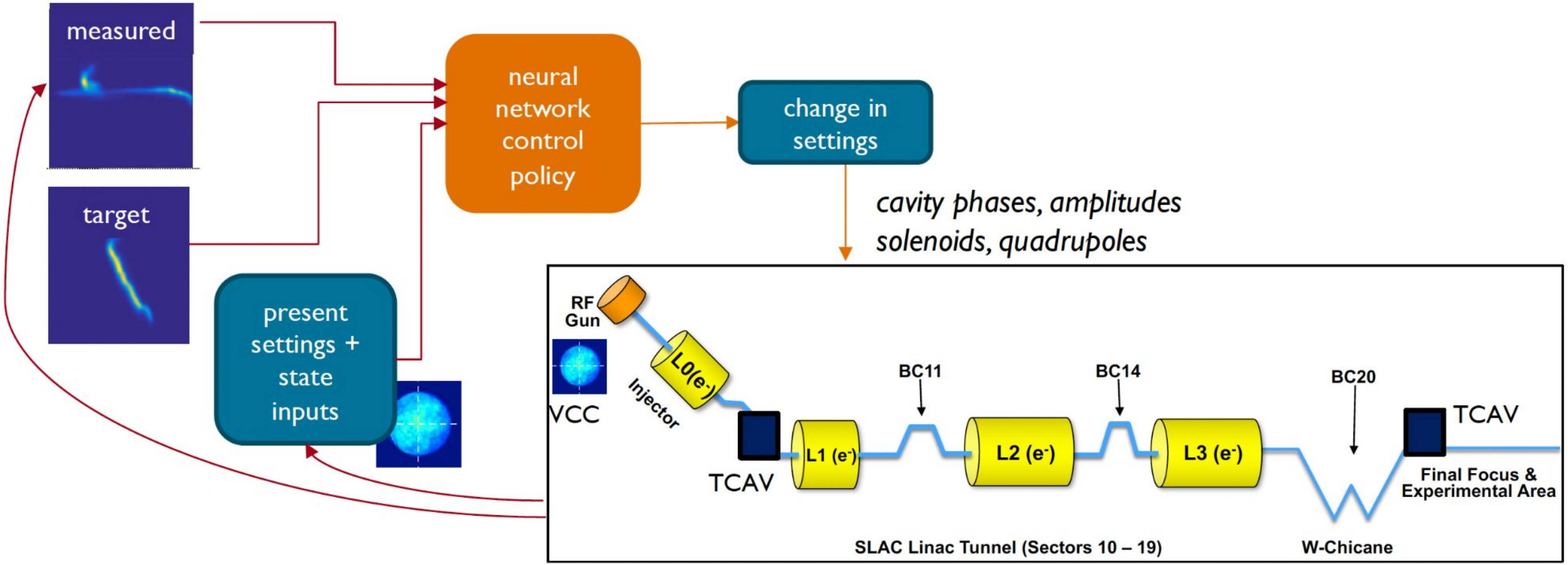


A. Marinelli, et al., Nat. Commun. **6**, 6369 (2015)



A. Marinelli, IPAC'18

Comprehensive, system-wide control is likely to be a key factor in improving custom control of extreme beams, but this is a difficult task



Build out on sample-efficient methods on subsystems first (e.g. Bayesian approaches), then transition to more comprehensive approach (reinforcement learning, neural networks leveraging learned system model information)

Incorporate ML-based tuning into FACET-II operation to aid experiment goals along the way

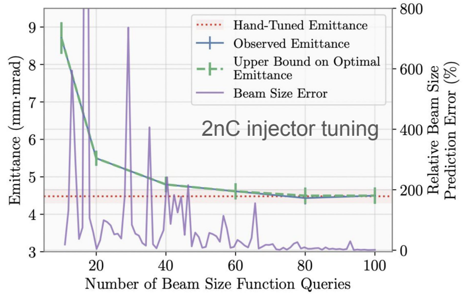
ML Experiments - E331

What worked (since last run)

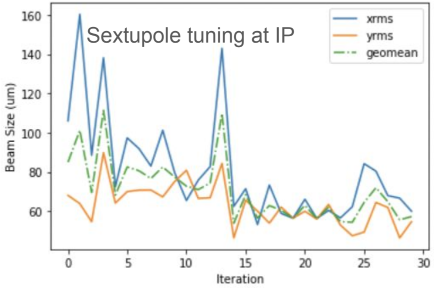
- Emittance tuning demo in injector (BAX - 20x faster than vanilla BO)
- Sextupole tuning demonstrated repeatedly and began integration into E300 *→ improved plasma performance (and only just scratching surface of possibilities!)*
- Smart data sampling for characterization / system model calibration - Bayesian Exploration (to gather data), multi-fidelity model calibration

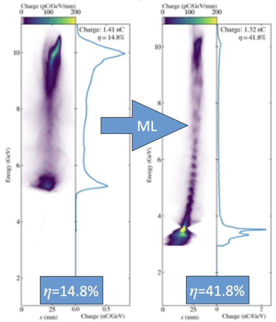
What didn't work

- Compute limitations: long inference times for BO *→ GPU for control system ordered and on its way (expect several orders of magnitude faster)*
- Challenges with automated data acquisition (e.g. wirescan GUI need server mode – human in the loop to take measurement; error prone / need to identify by eye) *→ need to think about for future observables we want to include in automated tuning*
- Challenges with writing/reading settings in SCP through python *→ need to set up ahead of time for controllable variables/read-backs we + users may want*
- For E300 tuning, simple metrics worked but need refinement (algo. will do exactly what its told to do….) *→ examples of measurements + working with plasma side ahead of time will help to set these up*
- Need way of triggering the DAQ from python

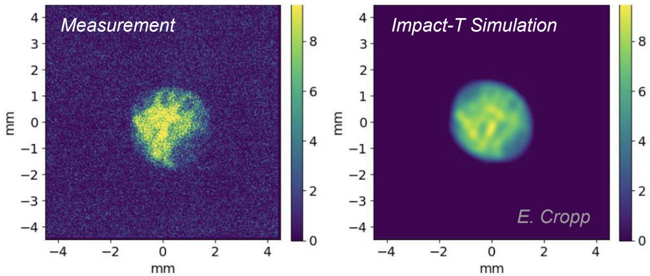


20x faster than regular BO

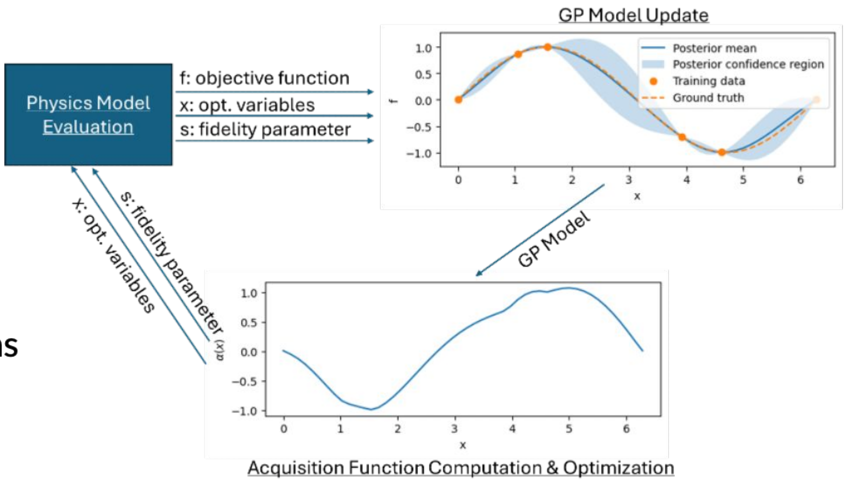




R. Ariniello

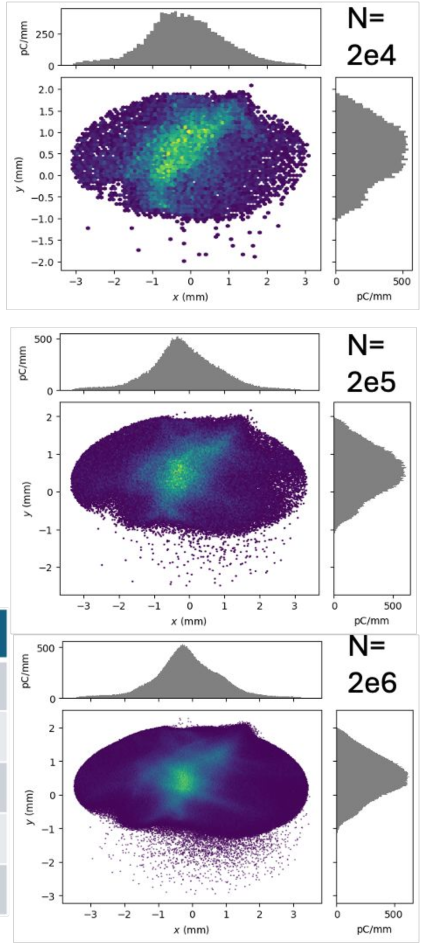


Multifidelity Optimization



- Information theoretic approach to simulations
- Learn correlations between different model fidelities
- Use multi-fidelity Bayesian optimization to select model fidelity and next optimization variables

Number of Particles (N)	2e4	2e5	2e6
Space Charge Grid Size	16	32	64
Execution time	~1 min	~2.5 min	~25 min
σ_x (um)	1026	1018	1017
σ_y (um)	654	623	614
Norm x emit (um)	9.26	8.87	8.77



E. Cropp

Finding Sources of Error Between Simulations and Measurements

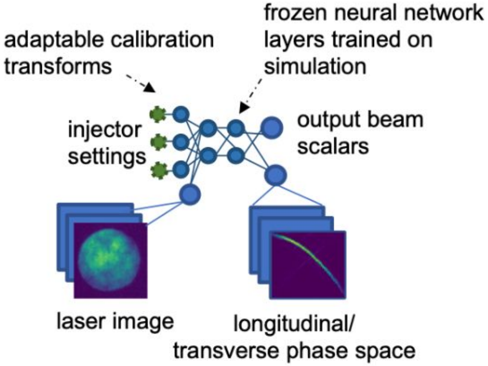
Many non-idealities not included in physics simulations:

- **static error sources** (e.g. magnetic field nonlinearities, physical offsets)
- **time-varying changes** (e.g. temperature-induced phase calibrations)

Want to identify these to get better understanding of machine performance

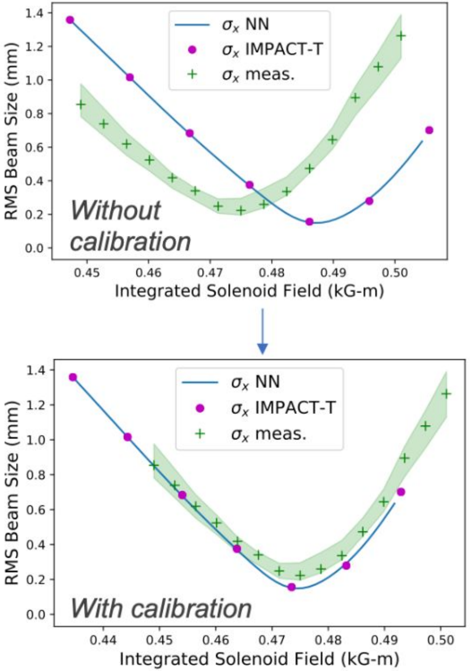
→ ***ML model allows fast / automatic exploration of error sources in high dimension***

Example: calibration offset in injector solenoid strength found automatically with neural network model (trained first in simulation, then calibrated to machine)



Inputs
Laser radius
Laser spot sizes
Pulse length
Charge
Solenoid
L0A phase
L0B phase
SQ quad
CQ quad
6 matching quads

Outputs
Beam size (x,y)
Emittance (x,y)
Bunch length



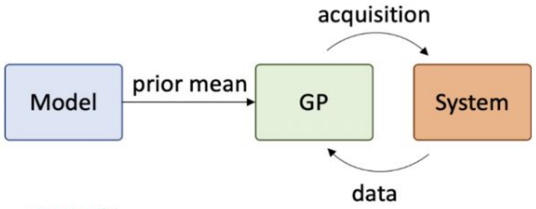
Speed and differentiability of ML models enables rapid identification of error sources between idealized physics simulations and real machine

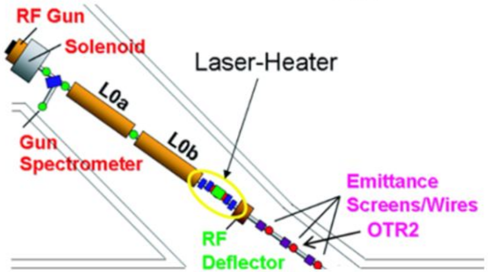
Background

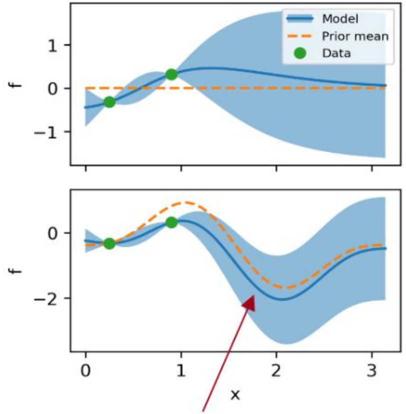
Leveraging Online Models for Faster Optimization

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

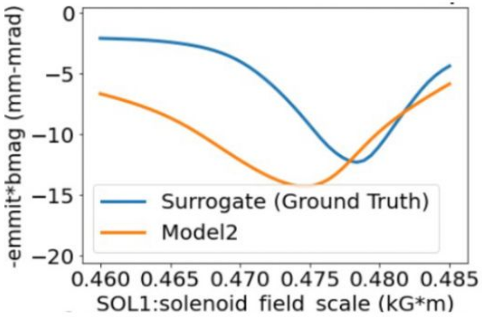
Prototyped on LCLS injector
variables: *solenoid, 2 corrector quads, 6 matching quads*
objective: *minimize emittance and matching parameter*





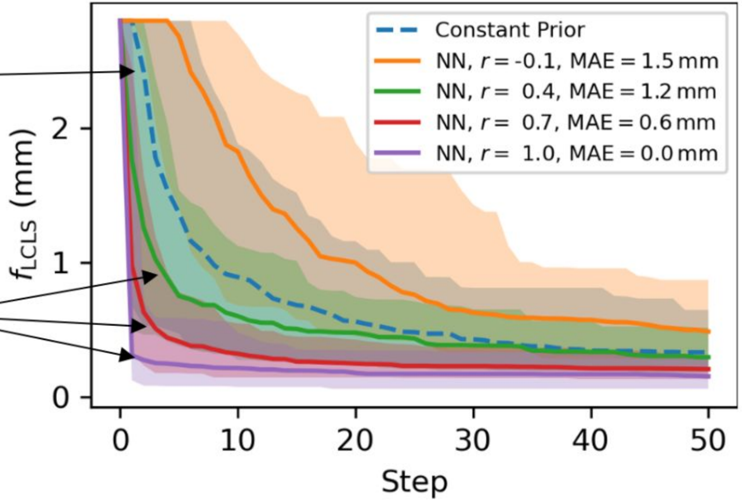


model prediction returns to prior



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

Digital Twin Infrastructure

Ecosystem of modular tools (can use independently)

LUME – simulation interfaces/wrappers in Python

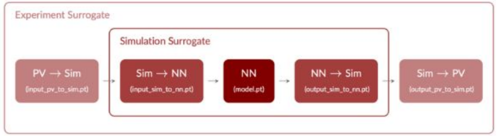
lume-model – wraps ML models, facilitates calibration

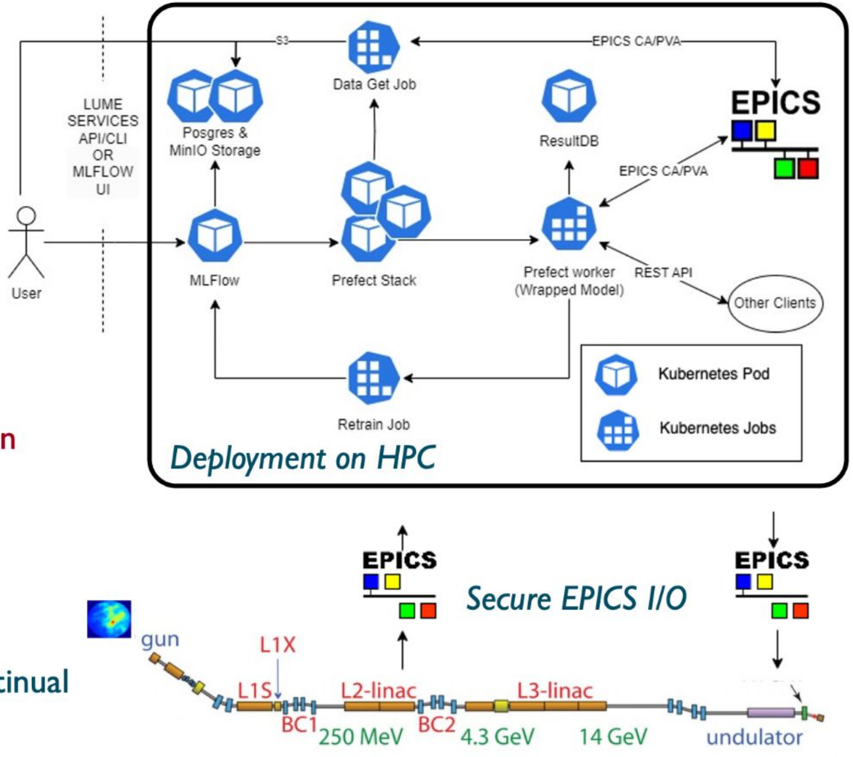
lume-services – online model deployment and orchestration

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

Background

Reinforcement Learning

Appealing for moving toward large-scale, comprehensive control of accelerators

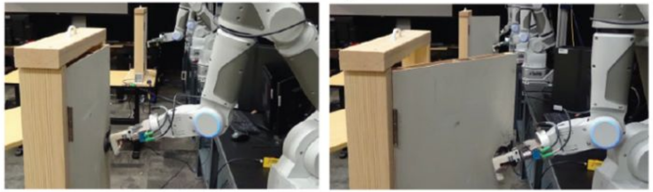
→ Many similarities to robotics applications

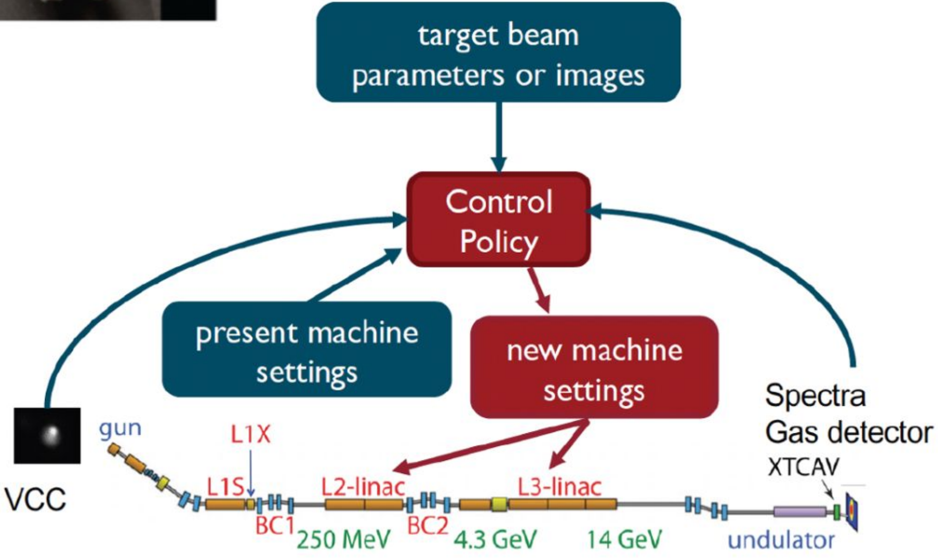
→ Ability to learn from many observations

→ Multi-modal, high-dimensional data

Nagabandi, et al., 2019

Gu, et al., 2016

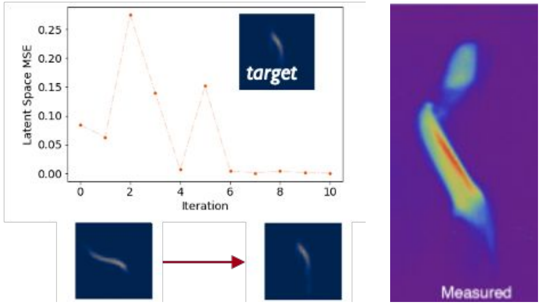




Goals For The Coming Run

Two themes: AI/ML R&D items (purple) and facility/experiment impact items (orange)

- Two-bunch tuning / LPS tuning ML development
  - Have algorithms to try for this→ need to set up with diagnostics/PVs to adjust
  - Prototype w/ previous data (e.g. image analysis) and simulations
  - Need XTCAV or other diagnostics we want to use for metrics ready
  - Incorporate additional diagnostics / objectives / constraints (e.g. LPS plus keep losses low, examine spectra?)
- Sextupole tuning
  - Use priors / correlations from previous runs (form model based on data or sim)
  - Improve integration with plasma metrics
  - Refine diagnostic analysis/setup for objectives/constraints
  - Deliver to ops + in Badger (AD PD funding to support)
- Model-based ML tuning – model dev + use as priors for BO and model-based RL
  - Path to faster/higher-precision tuning by adjusting more variables together across machine
  - Need model + tackling in stages: injector, linac, plasma
    - Incorporate calibrated injector system model into tuning
    - Extend model calibration downstream (e.g. up to IP) – LPS then transverse
    - Incorporate downstream system model into tuning
- Expand tuning scope (driven by operations need)
  - Emittance tuning to downstream (emittance preservation)
    - Would need some help in getting 3-wire measurement set up etc
  - Multiple objectives /constraints in tuning (e.g. emittance / losses, LPS, plasma) → want suggestions on what would be highest impact for operation and experiments



N. Majernik

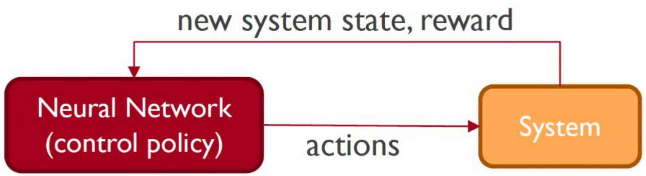
Time to data / pub

Publications

	High level science goals	First high impact publication	How to get from here to there
E-331	Comprehensive ML-based control for new capabilities (higher quality beams faster) Two-bunch / LPS customization	• BAX paper already published (mid impact) • Contribute to E300 paper (impact of ML on E300 tuning – sextupole tuning) (now - ?) • Model calibration for injector (mid impact, now - 0.25 yr) and linac (now - 0.5 yr?) • LPS single and two-bunch customization (0.5 - 1 yr? sooner?) • Model calibration for injector + linac (0.25 - 0.5 yr?)	**Need reliable diagnostics + analysis for LPS (XTCAV, others?)** **Need to set up fast analysis from BSA data or epics pulls (+ launch BSA data taking from python)** **Need to sort out setting PVs from python in SCP** **Need to gather historical/new data + latest physics models for two-bunch, single bunch for injector + linac**

Backups

Deep Reinforcement Learning

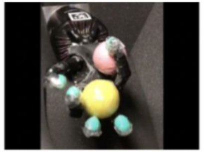


- Control policy maps states to actions
- Policy is learned over time based on performance *(quantified by the "reward")*
- Neural network enables use of diverse signal types *(e.g. scalars, images, time series)*
- Often learns a system model simultaneously *(map states + actions to expected reward)*

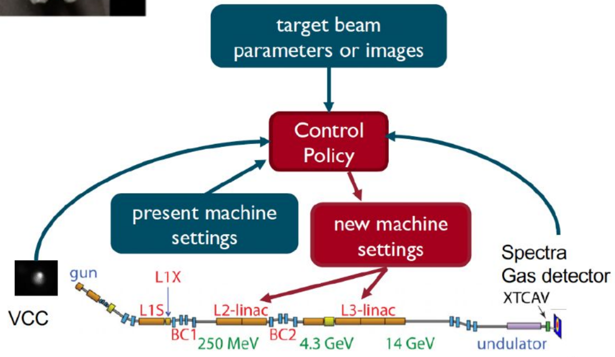
Appeal for accelerator control:

- Suitable for large, nonlinear systems
- Exploit machine-wide sensitivities + directly use complicated diagnostic information
- Leverage information from past observations
- Transfer between similar designs
- Well-established in other fields (e.g. robotic control)
  → but accelerators have unique challenges

Nagabandi, et al., 2019



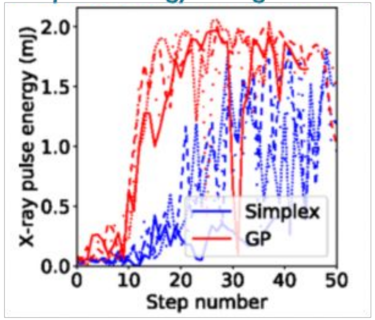
Gu, et al., 2016



Deep RL is well-suited to accelerator control, but dedicated R&D is needed to bring it to full fruition

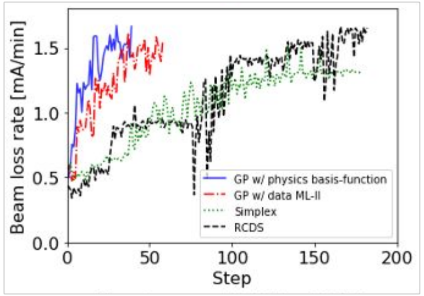
Many successes with Bayesian Optimization *(+ algorithmic improvements)*

FEL pulse energy tuning at LCLS



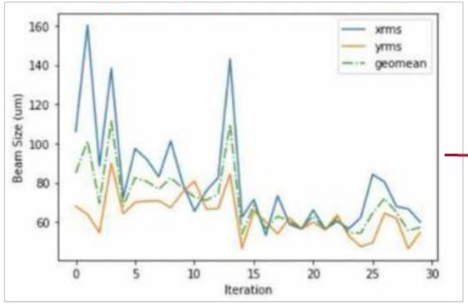
Duris et. al. PRL , **2020**

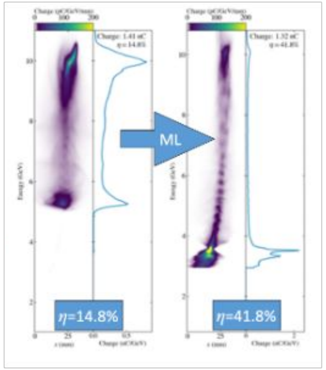
Loss rate tuning at SPEAR3



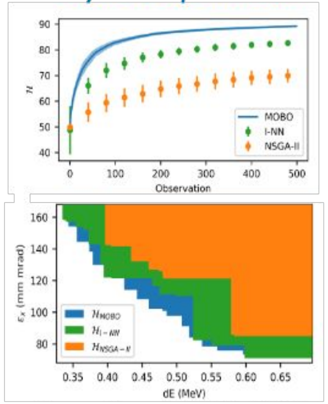
Hanuka et. al. PRAB , **2021**

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

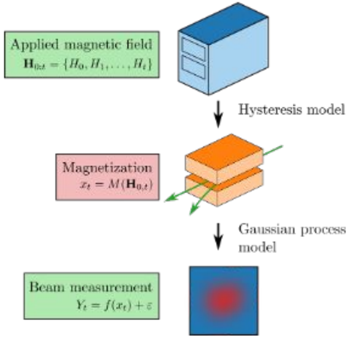




Multi-objective Bayesian Optimization

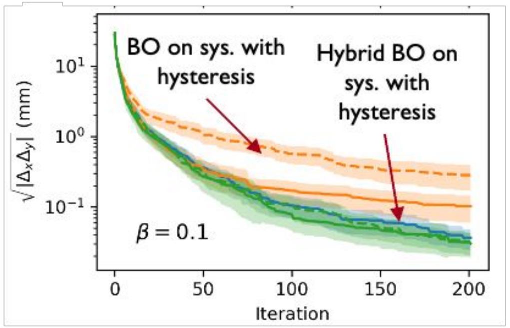


Roussel et. al. PRAB , **2021**

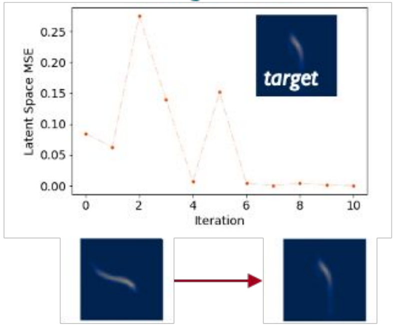


Roussel et. al. PRL , **2022**

Higher-precision optimization possible when including hysteresis effects in model



Longitudinal phase space tuning on LCLS



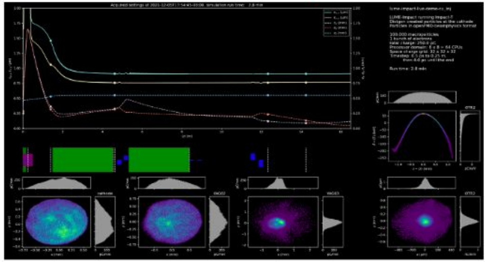
Algorithms being implemented/distributed in Xopt: https://github.com/ChristopherMayes/Xopt
Comprehensive review of advanced BO for particle accelerators: https://arxiv.org/html/2312.05667v2

Xopt

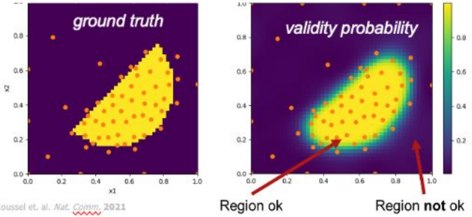
Broad Research Program at SLAC in AI/ML for Accelerators

(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



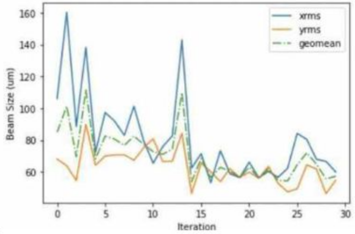
Efficient, safe optimization algorithms



Roussel et. al. Nat. Comm. 2021

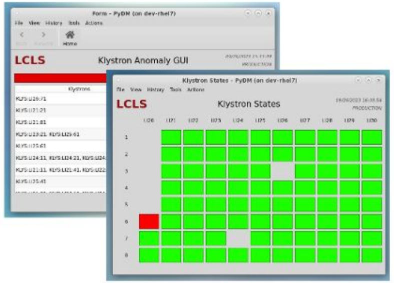
Output constraints learned on-the-fly

Adhere to constraints and balance multiple targets

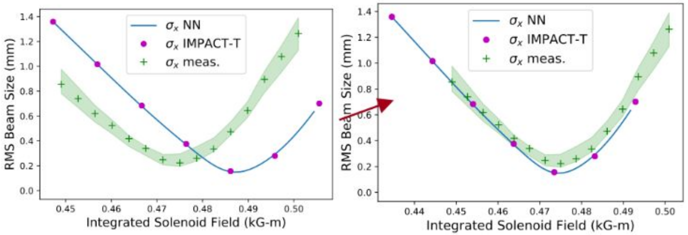


Challenging problems: e.g. sextupole tuning

Anomaly detection

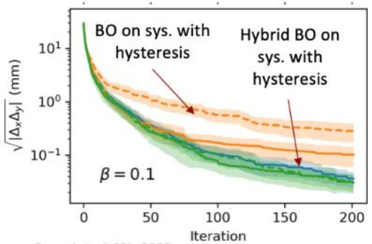


Adaptation of models and identification of sources of deviation between simulations and as-built machine



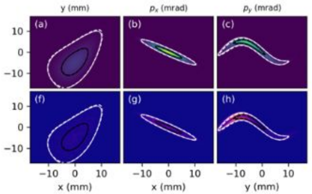
Combining physics and ML for better performance

Hysteresis-aware optimization



Roussel et. al. PRL. 2022

Differentiable simulations + ML for 6D phase space reconstruction

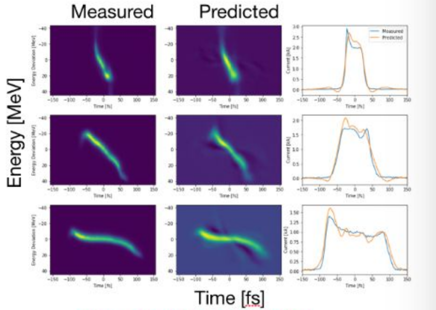


Roussel et. al. PRL. 2023

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.