ACCELERATOR SYSTEMS CYBER SECURITY ACTIVITIES AT SLAC*

G. White†, A. Edelen, SLAC National Accelerator Laboratory, California, U.S.A

Abstract

We present the idea that the needs of accelerator beam tuning by the emerging methods of Machine Learning and multi-particle modeling, will disrupt the norms of computer networking and cyber-security architectures of large accelerator control systems. We review present SLAC activities in solving beam problems using techniques such as Bayesian optimization, surrogate and inverse Neural Network model inferencing. These are frequently trained on multi-particle simulations, the training itself is computationally expensive, and increasingly on very large stored datasets of beam synchronous observables' past values. High Performance Computing (HPC) and Big Data will therefore take a central role in the accelerator control system. SLAC is building a "digital twin" framework in which to run and manage these models in the HPC cluster. Increasingly, the results of the modeling in HPC, will be deployed into the running accelerator, implying that AIs, outside the classical secure control network, can and will deploy setpoints autonomously. Finally, we review new controls protocol architecture and technology being developed at SLAC, in advance of these realities to come.

HISTORICAL CONTEXT AND CONTROLS ARCHITECTURE

Historically, low order models such as transfer matrix and Courant-Snyder parameters, have been central to "online" beam optimization, being the basis of beam orbit correction, bumps, and basic feedback. That is, beam tuning as carried out on the running accelerator controls, has been in opposition to offline lattice design and beam dynamics study, which have classically been done offline, using model methods able to investigate beam phase space, but that require High Performance Computing (HPC) and runtime periods of hours or days. Furthermore, our use of these methods for online tuning, has come with some assumptions that have become coded into norms. First, that we know *a priori* the basic the relation between actuators and sensors – and we approximate it largely linearly (plus some 2nd order). Second, that for global optimization, minimizing the orbit RMS, or timing, will optimize the true objective – minimize emittance or maximize luminosity. Third, that direct tuning 6D phase space, or beam time structure, was out of reach for the turn around time of accelerator operations.

Over the last years, model methods have evolved. Directly tuning the injector requires modeling space charge. To understand true linac optics, RF kicks, magnet errors, and dynamic initial conditions, must be included. These imply multi-particle codes and Machine Learning brought to bear together online. ML is used to compute solutions under uncertainty, or to learn the dynamics for fast online execution, or simply to give empirical insight where the available physics or simulation have not accessed the parameter space with enough precision.

MODELING ACTIVITIES AT SLAC

Many tuning problems at LCLS/LCLS-II and FACET-II at SLAC and elsewhere, now require detailed phase space customization for different experiments. The beam exists in 6-D position-momentum phase space. We measure 2-D projections and reconstruct based on perturbations of upstream controls (e.g. tomography, quad scans). However, we have incomplete empirical information and we have dozens-to-hundreds of controllable variables and hundreds-of-thousands (up to millions for LCLS-II). Beam optimization then, is a nonlinear, high-dimensional problem. We also have a wide variety of tuning needs, such as rapid FEL beam pulse energy spectra construction, beam parameters such as two-beam, or pulse-probe, and maintaining time-energy stability. To these problems we bring a collection of approaches; from model-free estimation like gradient Decent, to model guided optimization and Physics Informed Neural Networks, and inverse models for feed-forward corrections. Our strategy is to start with sample-efficient methods that do well on new systems, then build up to more data-intensive and heavily model-informed approaches.

The long term requirement then is for fast, accurate, system models of the beam and experiment dynamics. Accelerator simulations that include nonlinear and collective effects are powerful tools, but they can be computationally expensive. To some extent, ML models are able to provide fast approximations to simulations (the so-called "surrogate models") so in-situ optimization is orders of magnitude faster than problem error minimization by multi-particle code iteration. However, the surrogate models must themselves be trained on simulations, which must be computed, or on large, typically long baseline, data sets - which take a lot of storage space.

DIGITAL TWIN FRAMEWORK

Tuning then depends on High Performance Computing, either for running the multi-particle codes or for training the ML. For cost reasons, HPC is typically a shared resource outside the production control system, so our codes and models often run "offline" in HPC facilities such as NERSC and, recently, the new Stanford Research Computing Facility (SRCF).

As the first, large step then, we're developing a "Digital Twin" framework for accelerator modelling, which specifically includes modeling and optimizing the extant accelera-

[*] Work supported by US Dept of Energy DE-AC02-76SF00515, and grant from Executive Order 14028

[†] greg@slac.stanford.edu

tor in "real-time". The models of accelerator twins will be housed mostly in (SRCF), though our models are open and the paradigm of LUME (see Fig. 1) micro services should allow some facility neutrality. The new modelling system includes databases of models (multi-particle, envelope or "ML") plus the results of model runs. At SLAC, models in execution in SRCF will have access to the extant accelerators' Process Variable values.

Notably, the solutions computed by the modeling HPC systems, practically speaking, are sets of Process Variables values, that the model optimizer in question recommends. To be efficient, ideally the deployment is a write operation to the relevant PVs. And that of course, is a cyber-security matter - a system or person outside the secure perimeter writes to a PV on the accelerator. To address that cyber issue at SLAC, whose accelerators use EPICS, we propose to use the new Transport Layer Security (TLS) additions being developed by SLAC in collaboration with Osprey and ORNL. In short, TLS is being added to pvAccess (PVA)- the new PV communications protocol of EPICS. This secure PVA will mediate communication from the compute facility, to the accelerators.

BIG DATA FOR ACCELERATORS

In Free-Electron Laser Accelerators, a key objective is control and tuning the energy spectra of delivered X-ray pulses. This requires us to build an invertible model of the accelerated electron beam and photon energy spectra. However, that model requires reliable long baseline electron and photon pulse synchronicity, and additionally beam pulse synchronous machine meta data- such as kicker activity, cathode parameters, etc. The controls requirement would then be is simple but difficult to achieve: save all beam synchronous data (beam monitors, toroids, fast actuators, klystrons, etc), for every pulse, all the time.

Further, a large data store of pulse synchronous accelerator data, was identified as *the* major unrealized required resource for machine learning for beam dynamics, at the ICFA Beam Dynamics Workshop on Machine Learning in 2018.

SLAC set out then to read and store all pulse synchronous data for every beam in the LCLS (then a warm-copper only, 120 Hz machine), 24x7. This "Beam Synchronous Data Service" (BSA Service) effort has been successful, all beams are now recorded to h5 files for use in ML training continuously. The contemporaneously read beam data is also made available to EPICS listeners as pvAccess NTTables.

Since then also LCLS has been upgraded with a second, superconducting linac, with a peak repetition rate of 1 MHz (!). A new version of the BSA Service now being commissioned with the superconducting linac, appears capable of reading all BSA signals up to 1 MHz, averaging down to 1 kHz (while also, importantly maintaining at least all signals unreduced at 1 kHz. IOCs stream pvAccess NTTables and they're assembled into h5 in the SRCF mass storage facility.

None of these high performance EPICS services could have been possible with EPICS version 7, in which pvAccess is approx. x2.5 faster than it's predecessor CA for bulk data transfer, and which contains the Normative Types dataset PV primitives with which we intuitively transfer all signals on a pulse.

SUMMARY ML REQUIREMENTS OF CONTROLS

Historically our online models have been constant, and largely linear. It was then satisfactory that the model lattice was assembled offline, and only recomputed online with known RF and focusing. Tuning was based on linear invertible submodels. However, now, differentiable multiparametric and Neural Network models are in continuous use online, and big-data acquisition is informing online operations and beam tuning in real time. Online models are no longer linear only. Space charge, RF kicks, magnet errors, dynamic initial conditions, are included.

These is still though a gap between the classical and emerging use cases. MLs are naturally compute hungry, so they are trained offline. It's hard to deploy and retrain online due to computational limitations. In practice, the tools of ML analysis, like pytorch and Jupyter, aren't a natural fit for the conservative production environment.

The summary requirement is to train and run models in the High Performance Compute environment, and allow them to tune the machine directly. But that implies PV writes from outside the protected accelerator network to inside. What would be the requirements to make MLs, running in a supercompute facility, part of the operational tuning of a secure accelerator?

- MUST guarantee that the computed PV solution (or more likely collection of PVs) is not detrimental to operations or dangerous to the machine. E.g. Synchrotron orbit correction deployed right at the time of ring injection. Solutions within magnet ranges. At least minimize subject to constraints
- MUST be protocol secure. enable a computed recommendation for a machine PV value to be deployed into the accelerator without risk that PV's value has been trivially hacked - compromised by a malign actor. E.g. someone on HPC just writes to an objective PV, that then is reflected into the accelerator
- MUST/SHOULD guarantee the code of the model and optimizer framework is secure. That the PV processing itself has not been hacked. For instance, the model itself, hosted in HPC, maybe an EPICS IOC, which must not be compromised if its able to write to the accelerator.

SECURE TUNING SOLUTION DEPLOYMENT

The key problem is that solutions will be computed outside the classical secure network of the accelerator. One solution that presents itself is to run a so-called mailbox server outside the secure perimeter, to which the optimizer

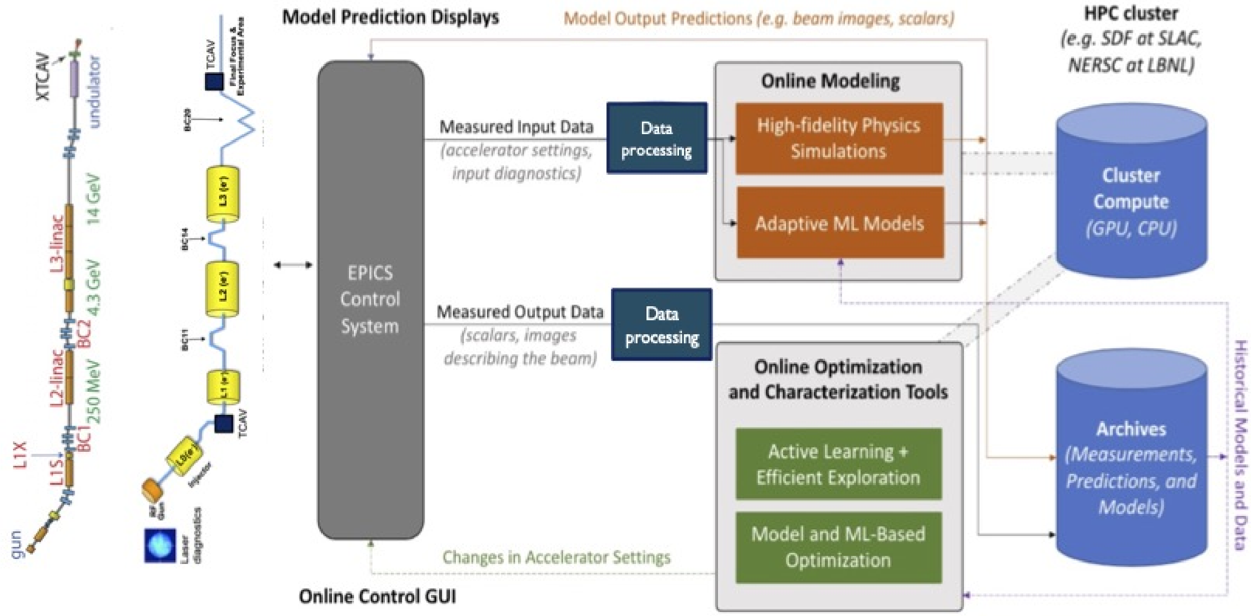
XTCAV
undulator
14 GeV
L3-linac
4.3 GeV
BC2
L2-linac
250 MeV
BC1
L1X
L1S
gun
TCAV
Final Focus & Experimental Area
BC20
L3 (e⁻)
BC14
L2 (e⁻)
BC11
L1 (e⁻)
TCAV
L0(e⁻)
Injector
RF Gun
Laser diagnostics
Model Prediction Displays
Model Output Predictions (e.g. beam images, scalars)
HPC cluster
(e.g. SDF at SLAC, NERSC at LBNL)
EPICS Control System
Measured Input Data
(accelerator settings, input diagnostics)
Data processing
Online Modeling
High-fidelity Physics Simulations
Adaptive ML Models
Cluster Compute
(GPU, CPU)
Measured Output Data
(scalars, images describing the beam)
Data processing
Online Optimization and Characterization Tools
Active Learning + Efficient Exploration
Model and ML-Based Optimization
Archives
(Measurements, Predictions, and Models)
Historical Models and Data
Changes in Accelerator Settings
Online Control GUI

Figure 1: The LUME Architecture; live process variable I/O will be via secure EPICS pvAccess (see below), distributed in some cases via kafka to simulations and ML models running in high performance compute. The computation may be interactive ad-hoc in some cases, or kubernetes pods for long running jobs such as multi-modal start-to-end simulation. Existing models (eg weights or IMPACT-T configurations), plus past model runs and their results are archived for reuse.

writes, and the production network monitors. When the optimizer has written a new value to a publish-subscribe endpoint like an EPICS record, a callback issued to a trusted server inside the production would pick up the new value, check its validity, and write it to its true destination. This obeys the common secure control edict - no writes allowed from outside production to inside. In the Kubernetes context common in cluster computing, the write to the mailbox may be by a kafka channel, particularly if reads from the control system are by also mediated by kafka.

We posit a more direct solution; use the new secure pvAccess protocol. Consider the objective of control network; Authentication, Authorization, Audit (AAA). The use of a private network is to assure that only authorized people can enter the network in which the control protocol allows write - that is, it controls authentication. Secondarily, the control protocol, in our case those of EPICS (CA and pvAccess), can be configured to allow writes (or reads) only to a set of people in an access control list. That is, the protocols do include authorization- but only in so far as the user is truely authenticated (not an imposter). (Parenthetically, EPICS servers also include Audit via logging.)

So, in practice, large facilities secure the control network by using login based restrictions - one can only change a setpoint if logged into the production network. And ones username is then, in EPICS, used to check the access control list.

In EPICS, clients make no explicit check that server they're talking to the true server of the signal (PV) they intend. Servers make no check that the client is who they claim to be. A man-in-the-middle attack requires only that the imposter respond first to a pv name search request. More bluntly, an imposter can simply usurp the true server with their own.

SECURE EPICS PVACCESS

Recently, SLAC has started a program to address the fundamental security of EPICS. Based on early work of Michael Davidsaver, in articulating the problem and outline solution with TLS, and George McIntyre, in designing how TLS would really be added to pvAccess, we have started the development effort. At the time of writing very early prototypes are already available - though with the major caveat of rudimentary certificate management. We have a 2 year plan ahead of us, funded specifically by a grant from the U.S. Dept of Energy under an application to the Executive Office.

CONCLUSION

AI, ML, and multi-particle simulation will change how we do online beam tuning, and in turn change computing infrastructure and its design for cyber security.

ACKNOWLEDGMENTS

Many extraordinary groups and individuals have contributed to SLAC's outstanding success in beam tuning and in the computer science and engineering that made it possible. Also, our success couldn't have come without the vision of the EPICS core group in working towards EPICS 7, and Bob Dalesio in particular. Additionally, we would like to express great appreciation to individuals in the U.S. Dept. of Energy, Office of the CISO, in particular Dale Leschnitzer and Mark Hahnert, without whose support we would not have received funding to add TLS to EPICS pvAccess, on which the network vision is predicated.