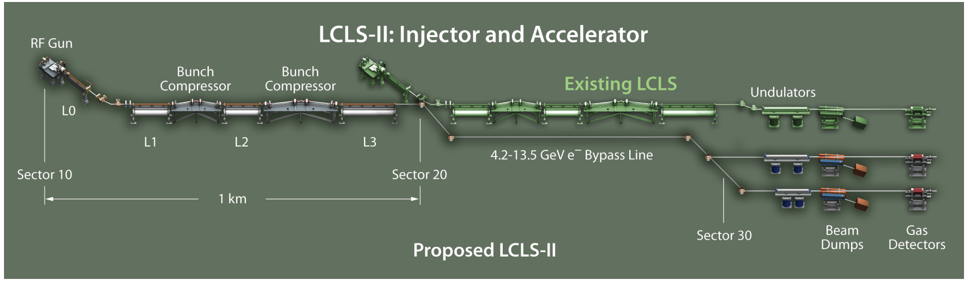
Adaptive Feedback and Machine Learning for Time-Varying Particle Accelerator Systems

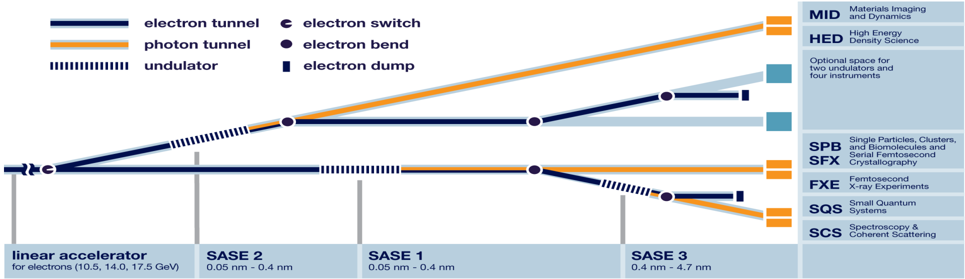
Alexander Scheinker

FRIB ACCELERATOR PHYSICS AND ENGINEERING SEMINARS
FEBRUARY 12, 2021

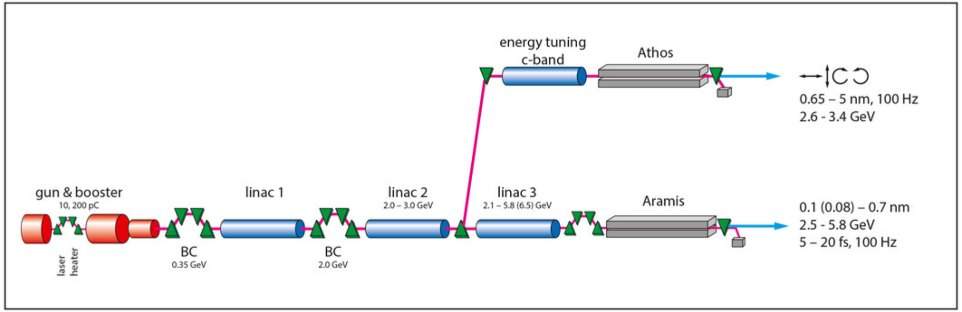
Los Alamos
NATIONAL LABORATORY
EST. 1943



LCLS/LCLS-II



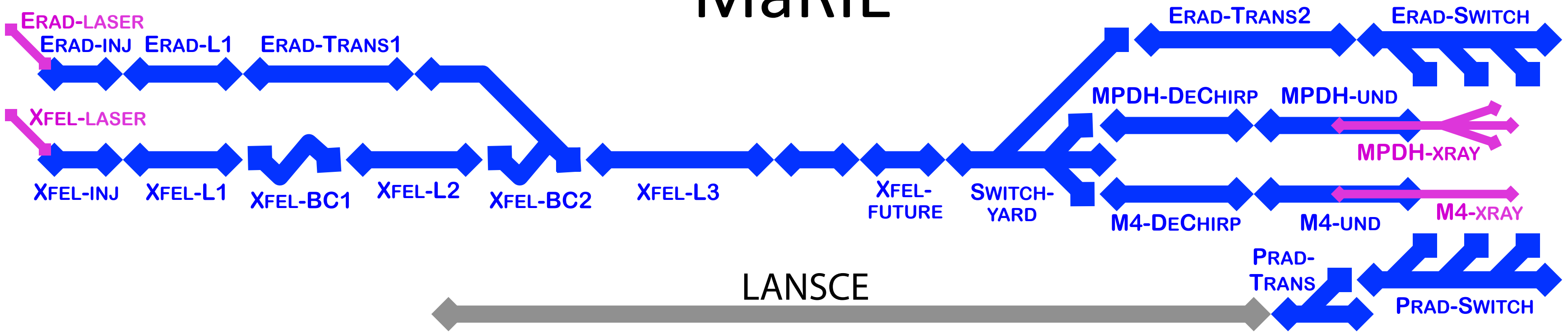
EuXFEL

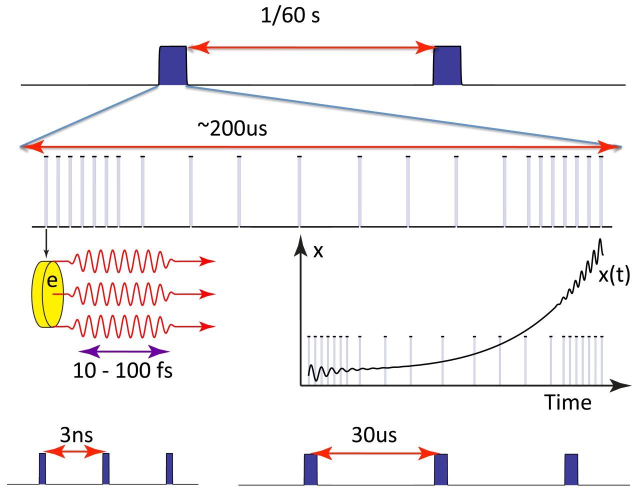


Swiss XFEL

0.6 fs pulses!

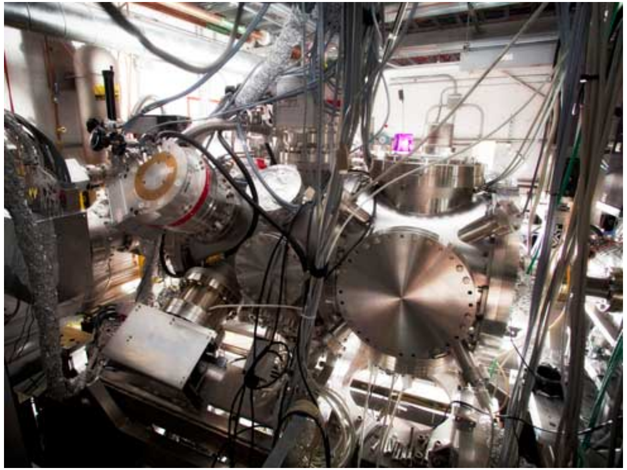
MaRIE





Photon energy: 4– 42 keV

AMO



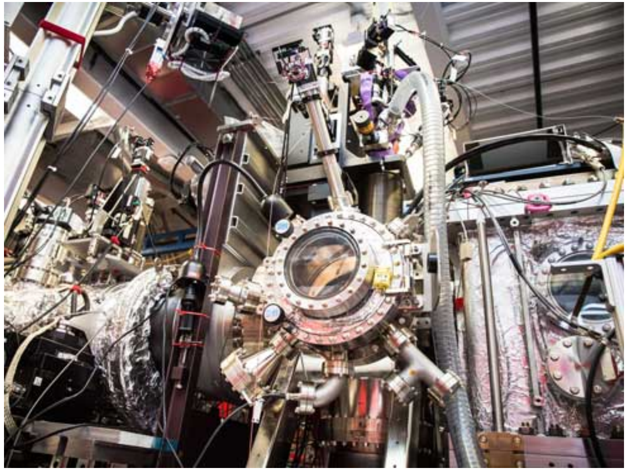
Atomic, Molecular & Optical Science

Soft X-rays for intense ultra short pulses. Gaseous targets of atoms, molecules, and nanoscale objects: protein crystals or viruses.

Photon energy: 0.48 – 2 keV
Pulse duration: 35 – 300 fs
Low charge mode pulse duration: No
Pulse energy: 1 – 20 mJ @ 266 - 800 nm

Max energy adjustment factor: 4.2
Low charge mode: No

CXI



Coherent X-ray imaging

Brilliant hard X-ray pulses for coherent diffractive imaging (CDI). Ultra short pulses for “Diffraction-Before-Destruction” experiments.

Photon energy: 5 – 12 keV
Pulse duration: 40 – 300 fs
Low charge mode pulse duration: <10 fs
Pulse energy: 1 – 3 mJ

Max energy adjustment factor: 2.4
Low charge mode: Yes

MEC

Matter in Extreme Conditions

High peak brightness, ultra short pulses of tunable energy X-rays for studying the transient behavior of matter in extreme conditions.

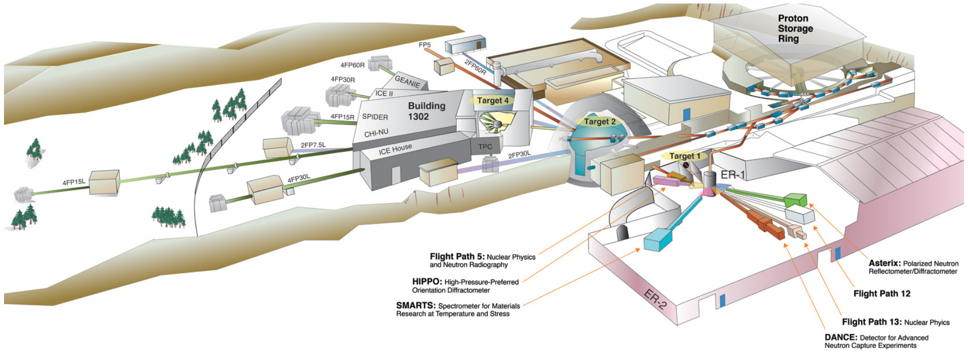
Photon energy: 2.5 – 12 keV
Pulse duration: 10 – 300 fs
Low charge mode pulse duration: <10 fs
Pulse energy: 1 – 3 mJ

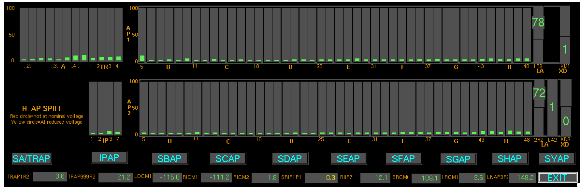
Max energy adjustment factor: 4.8
Low charge mode: Yes

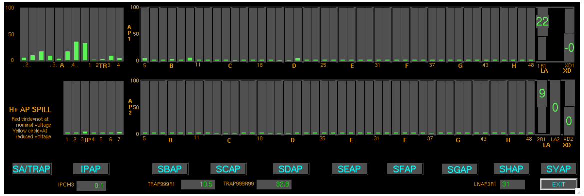
Low charge mode: Lower charge per bunch allows for tighter compression without destroying the electron beam’s phase space. Originally studying for accelerating 0.02 nC bunches instead of 1 nC.

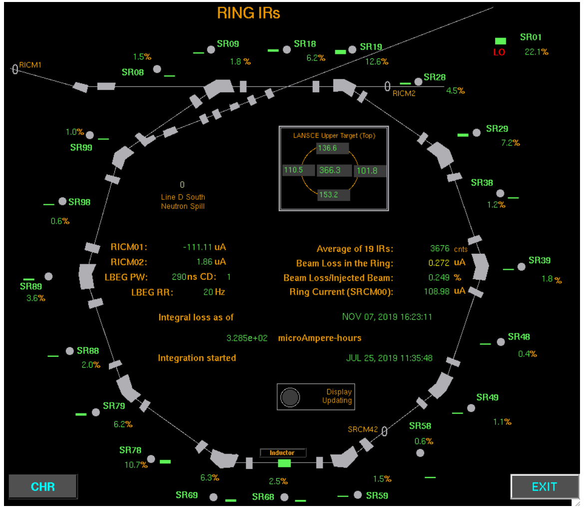
LANSCE

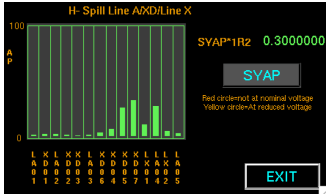
~5-6 weeks of tune up time

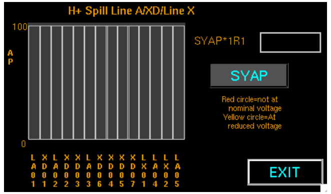






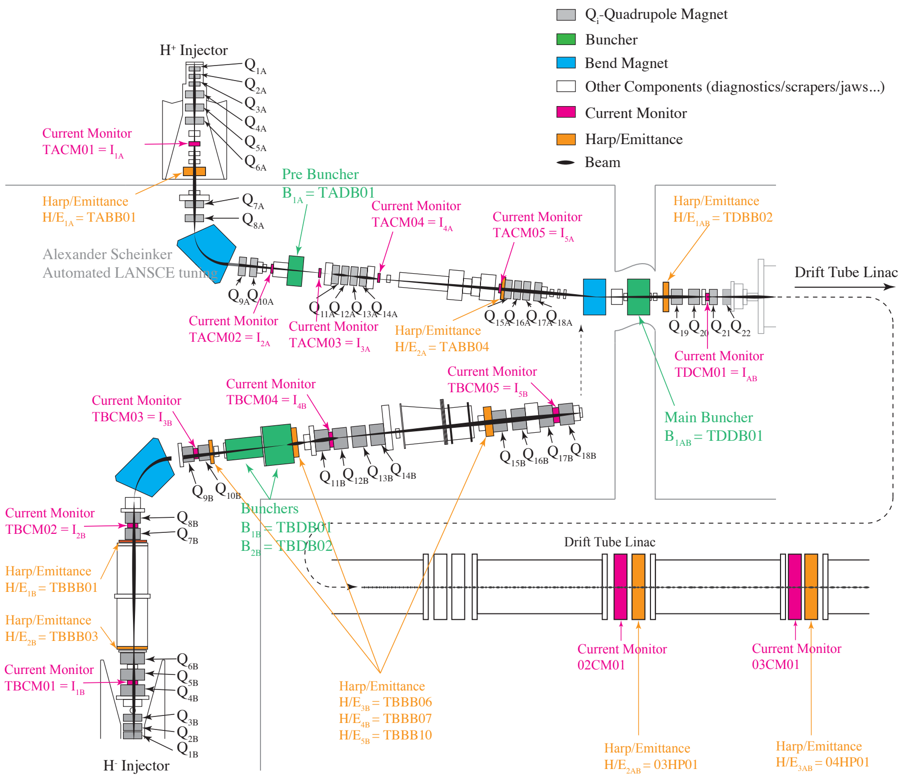


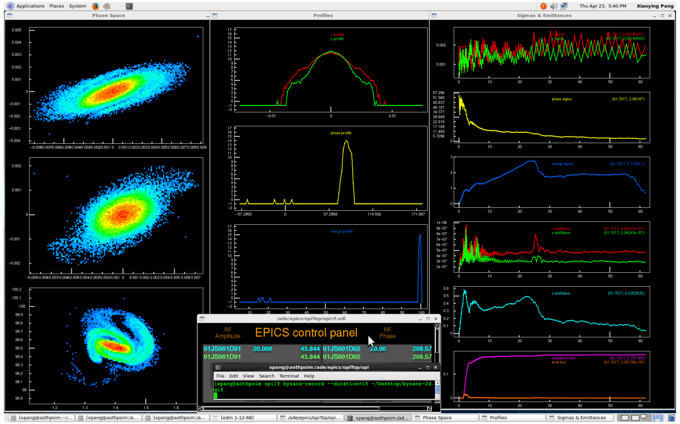


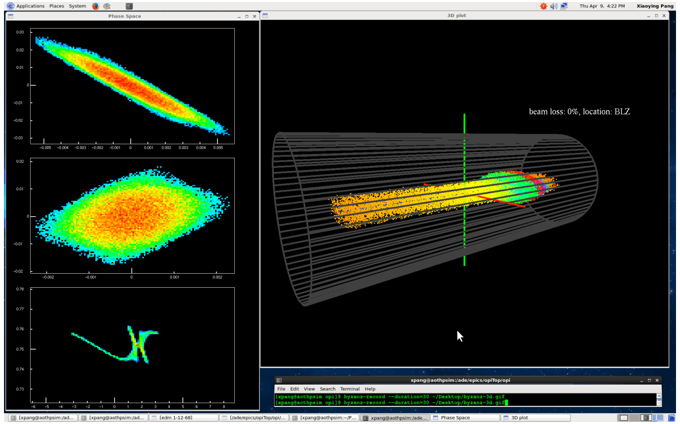


LANSCE Codes do not yet match the accelerator

Closely enough to serve as diagnostics





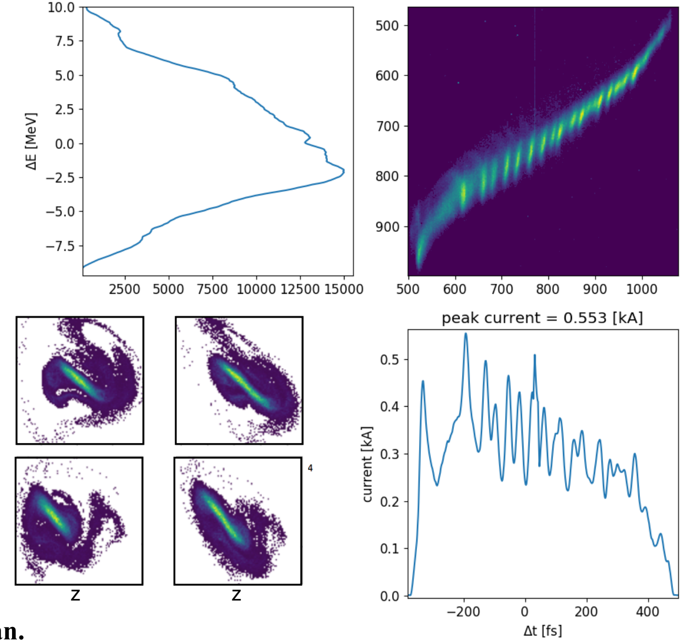


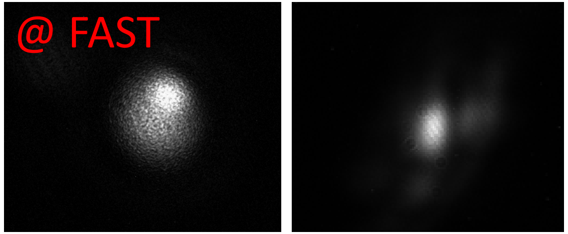
X. Pang, *HPSim*

Accelerator Tuning Challenges

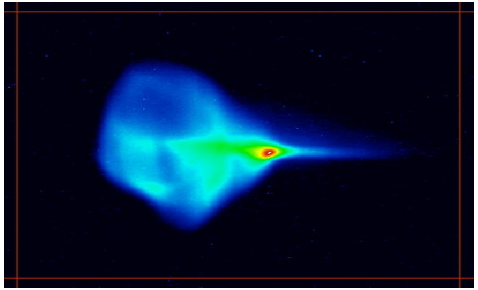
- Dynamics of intense charged particle bunches dominated by:
  - **Components drift unpredictably with time, misalignments**
    - **Uncertain and time varying beam distributions**
  - **Complex collective effects:**
    - **Wakefields**
    - **Space charge**
    - **Coherent synchrotron radiation**
  - **Limited non-invasive diagnostics**

EuXFEL: µBunch Instabilities



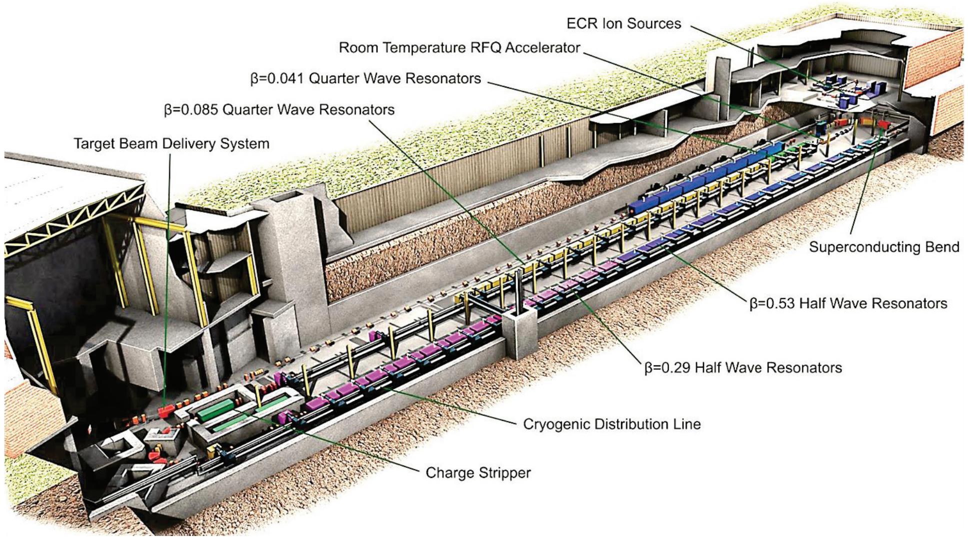


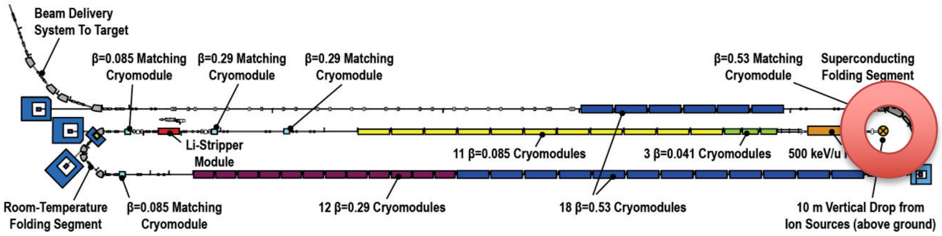
Example images of laser spot (10 Aug. 2016, 11 Nov. 2017)



Typical 2D (x,y) beam profile, not a simple Gaussian.

Facility for Rare Isotope Production (FRIB) at MSU



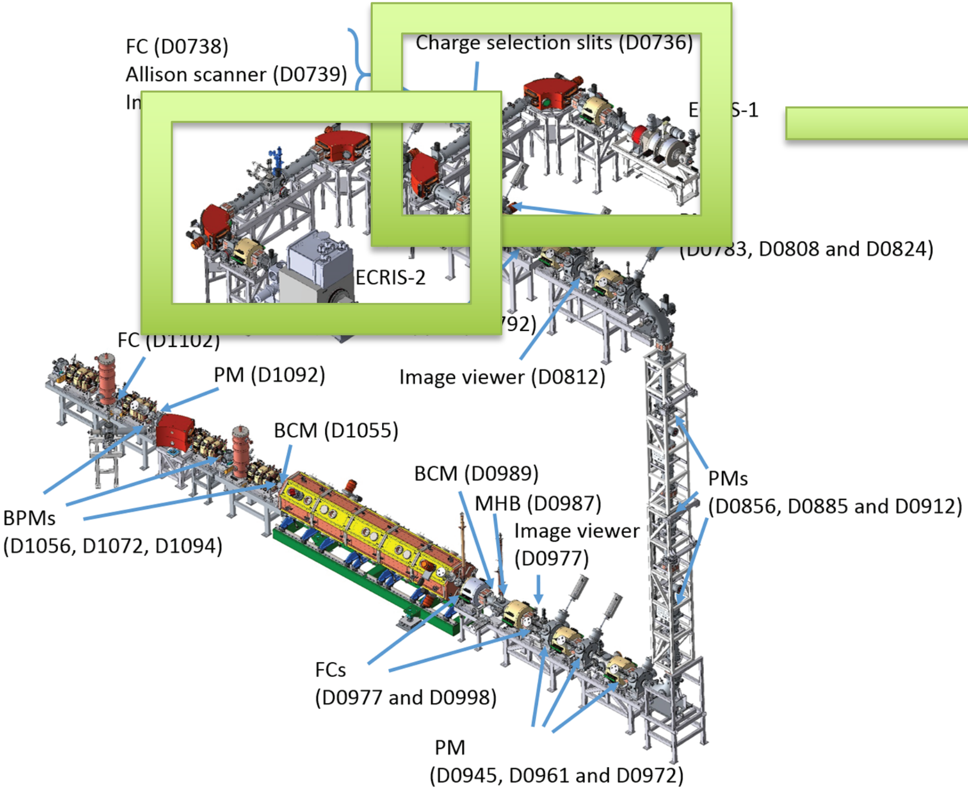


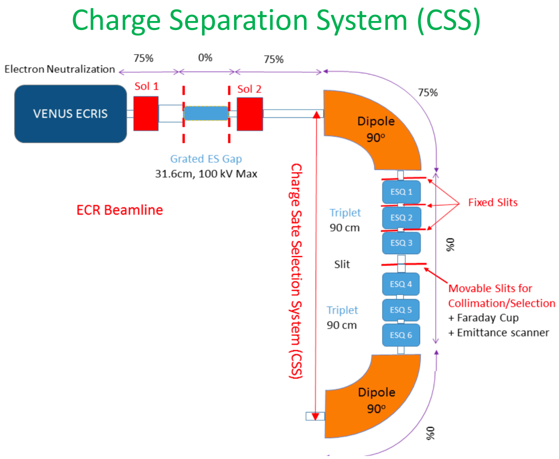
- Acceleration of 2 beam species simultaneously
- Low energy beam transport:
  - 35 keV, 12 keV/u
- Strong collective effects (space charge forces)

FRIB injector

M. Leitner, et al. "The FRIB Project at MSU." in Proceedings of SRF2013, Paris, France MOIOA01

FRIB injector





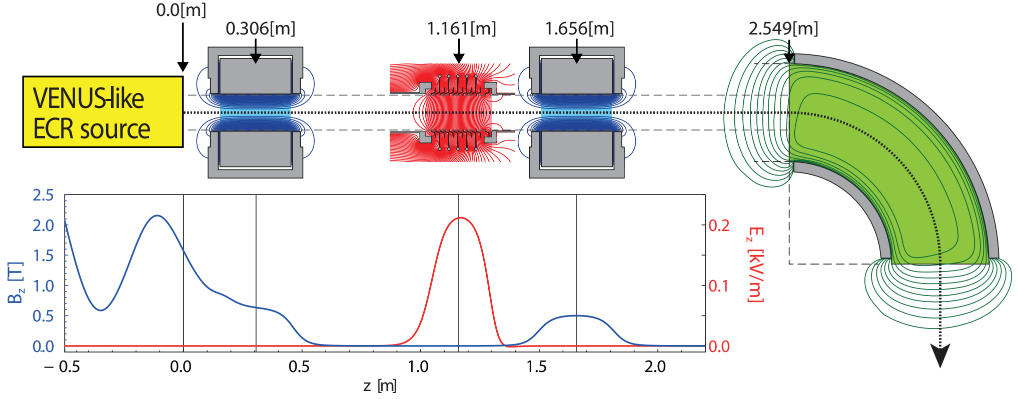
$$\text{Rigidity} \quad [B\rho] = \frac{p}{q} = \frac{\gamma m v}{q}$$

$$\delta = \left(\frac{\delta p}{p}\right)_{\text{eff}} = \frac{\Delta[B\rho]}{[B\rho]_0}$$

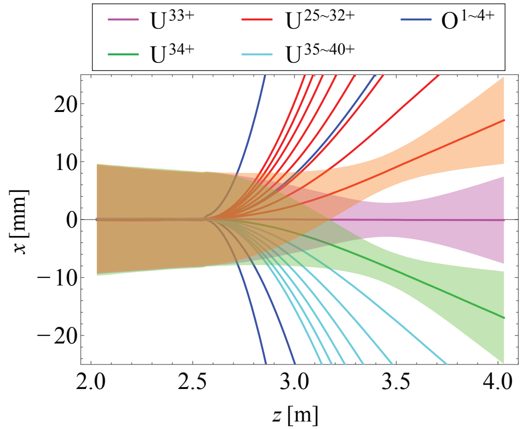
SM Lund and C. Y. Wong "09. Momentum Spread Effects in Bending and Focusing." US Particle Accelerator School, 2018

P. N. Ostroumov, et al. "Heavy ion beam acceleration in the first three cryomodules at the Facility for Rare Isotope Beams at Michigan State University." *Physical Review Accelerators and Beams*, 22, 040101, 2019

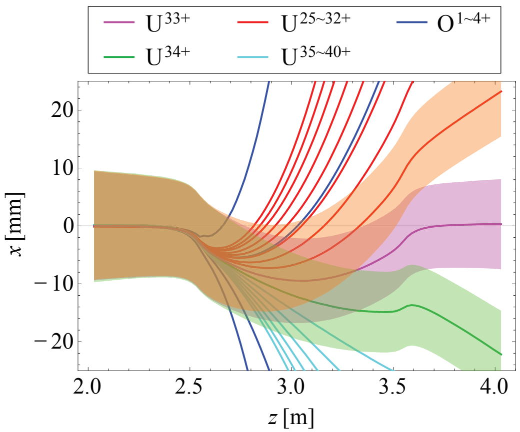
FRIB CSS



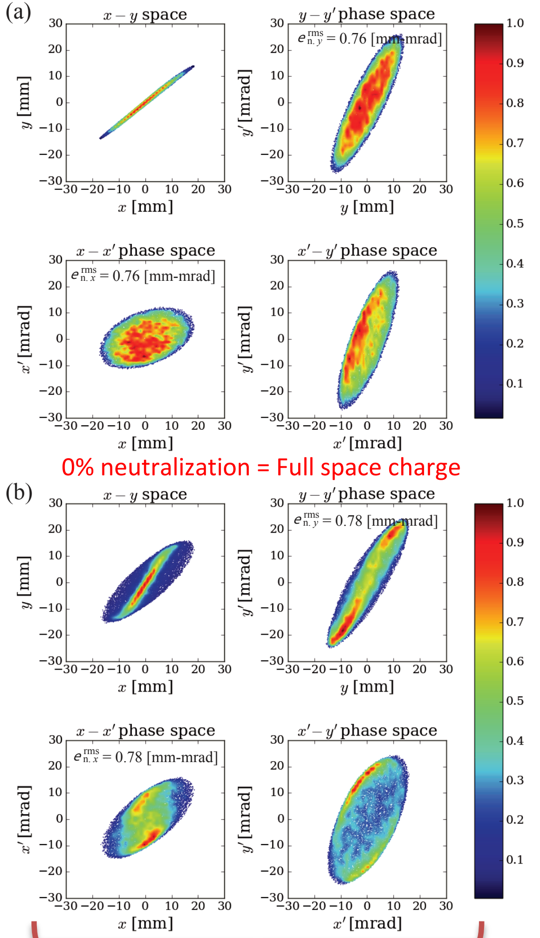
Perfect Dipole



Fringe Fields



75% neutralization



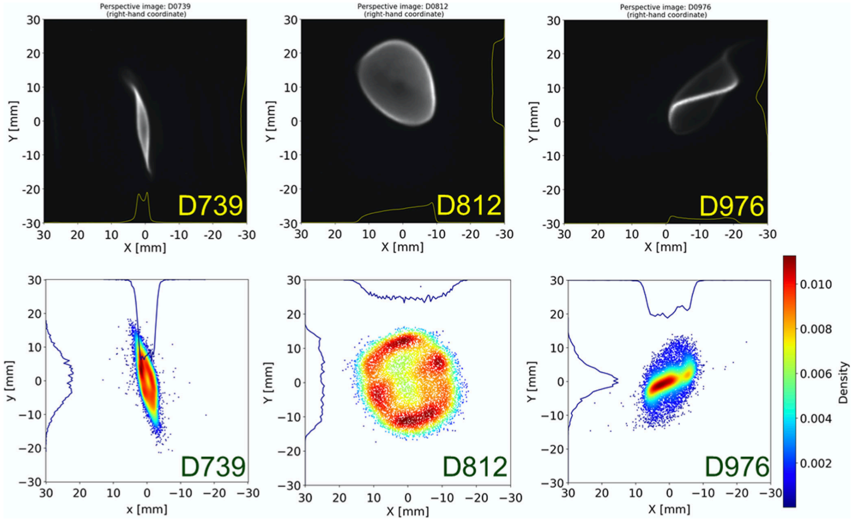
0% neutralization = Full space charge

Unknown & time-varying

K. Fukushima, S. M. Lund and C. Y. Wong "Multispecies simulation of the FRIB frontend near the ECR sources with the WARP code." in Proceedings of LINAC2016, East Lansing, MI, USA 2016

FRIB beams

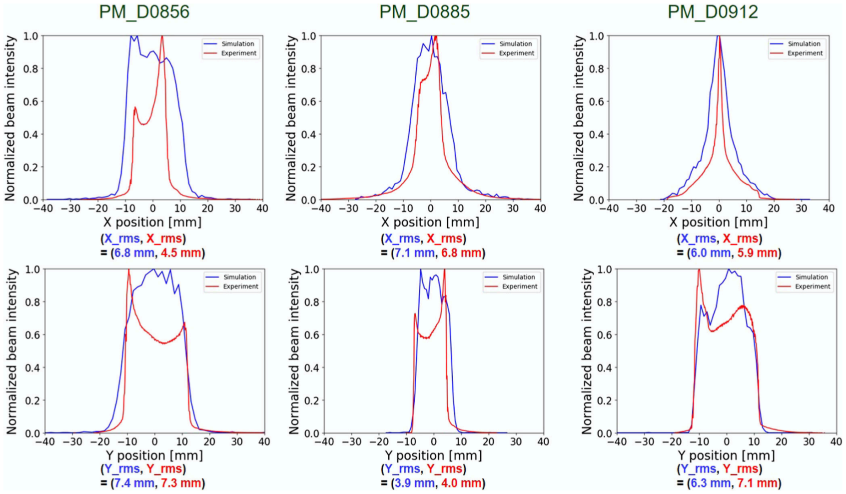
measurements



simulations

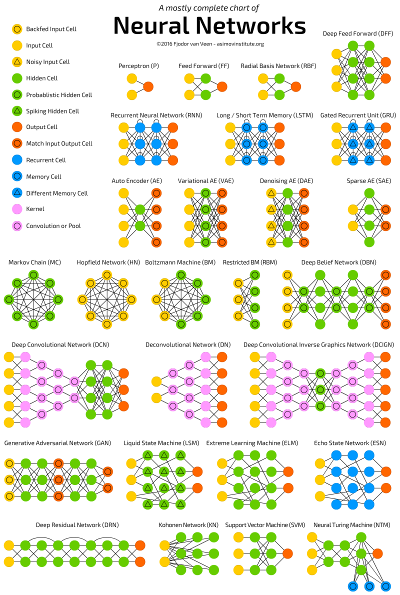
measurements

simulations



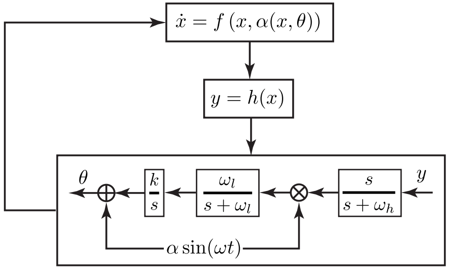
P. N. Ostroumov, et al. "Heavy ion beam acceleration in the first three cryomodules at the Facility for Rare Isotope Beams at Michigan State University." *Physical Review Accelerators and Beams*, 22, 040101, 2019

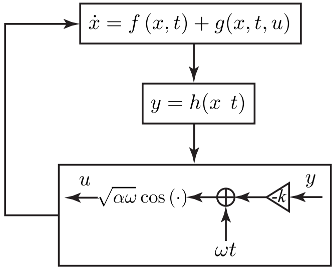
Machine Learning and Adaptive Feedback



Surrogate models
Big data
Global tuning
Anomaly detection

Adapative Feedback





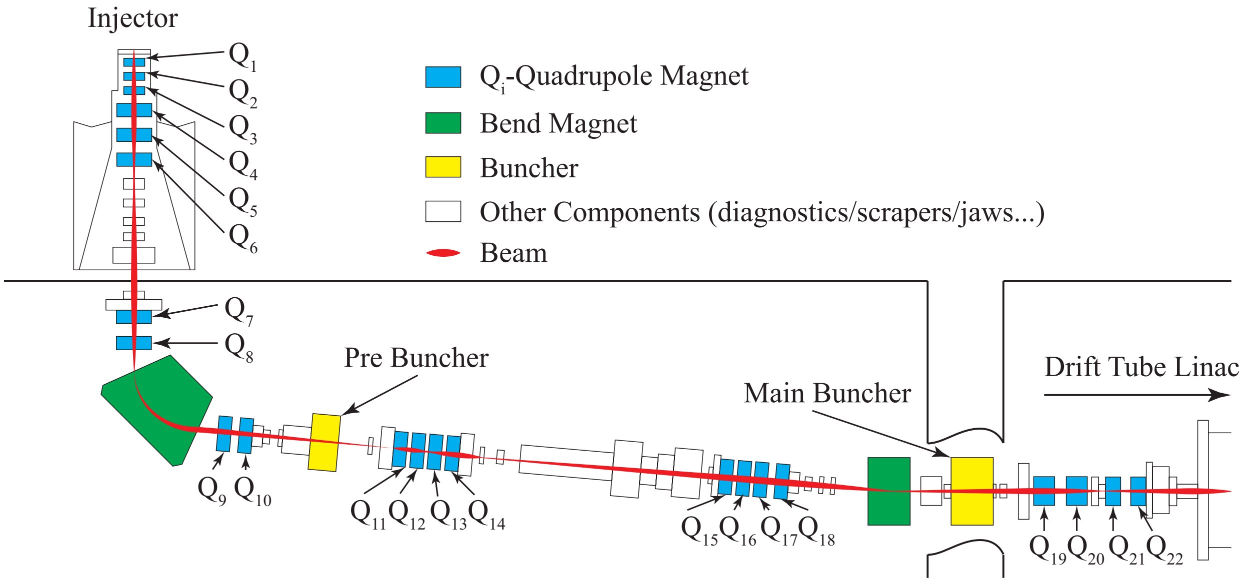
Virtual diagnostics
Real time feedback
Optimization
Phase space tuning

Model Independent Adaptive Feedback

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

Bounded Extremum Seeking: Model-Independent Tuning and Optimization

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$



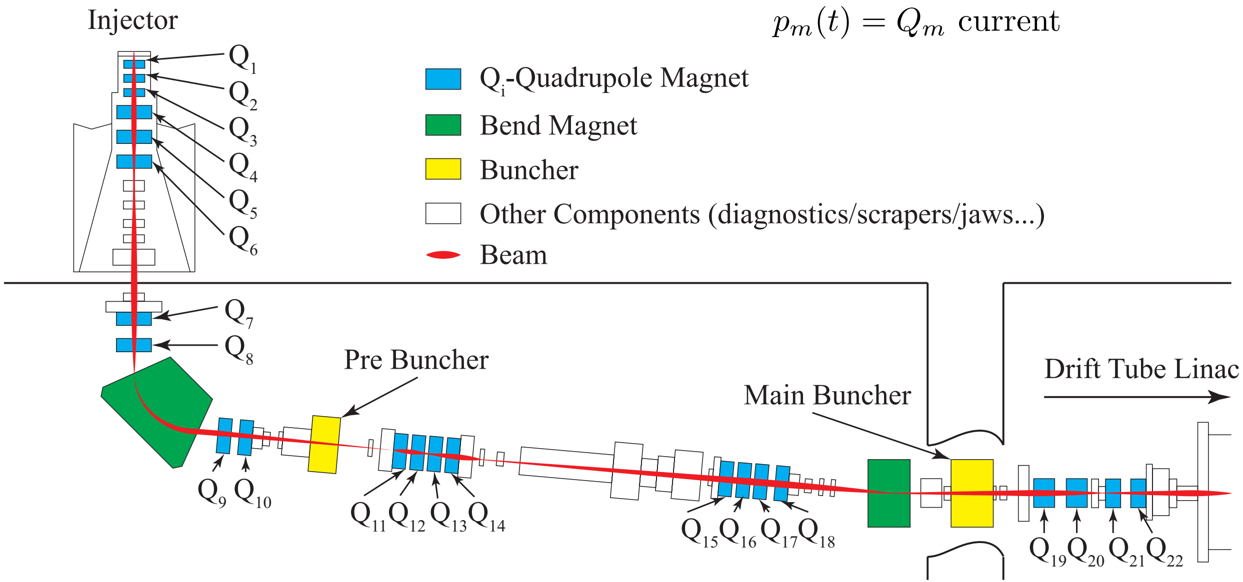
Bounded Extremum Seeking: Model-Independent Tuning and Optimization

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

$$p_1(t) = Q_1 \text{ current}$$

$$\vdots$$

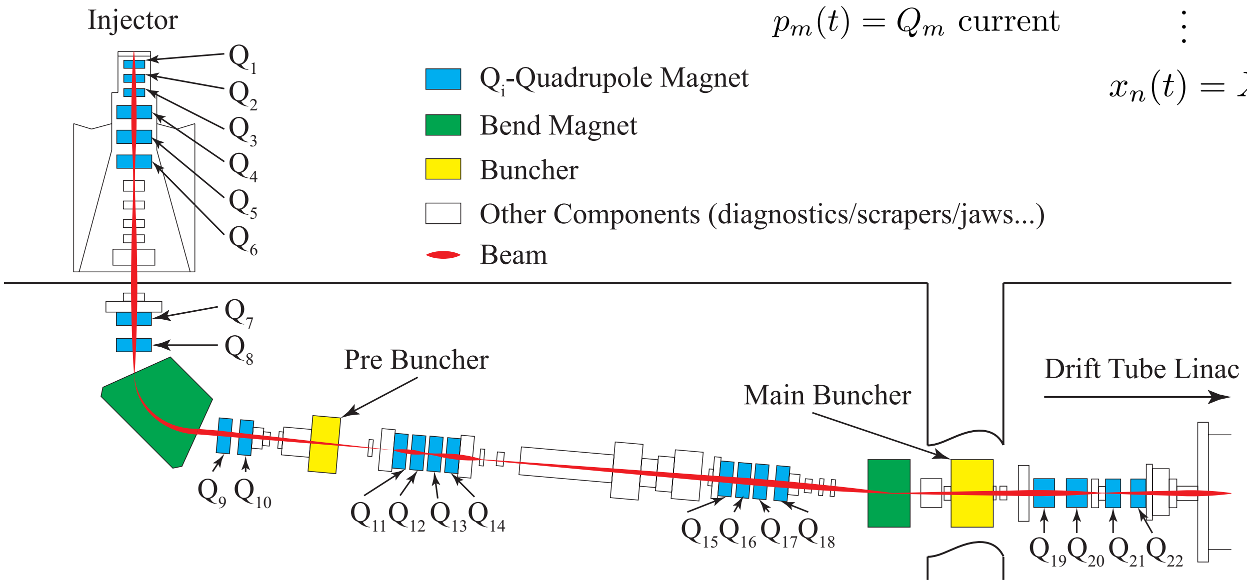
$$p_m(t) = Q_m \text{ current}$$



$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

$$p_1(t) = Q_1 \text{ current} \qquad \vdots \qquad p_m(t) = Q_m \text{ current}$$

$$x_1(t) = X_{\text{rms}}(\text{position } 1)$$
$$x_2(t) = Y_{\text{rms}}(\text{position } 1)$$
$$\vdots$$
$$x_n(t) = X_{\text{rms}}(\text{position n})$$



Bounded Extremum Seeking: Model-Independent Tuning and Optimization

A. Scheinker and D. Scheinker, "Extremum Seeking with Discontinuous Dithers," *Automatica*, vol. 69, pp. 250-257, 2016.

A. Scheinker and D. Scheinker, "Extremum Seeking for Stabilization of Systems not Affine in Control," 2017.

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

$$y = V(\mathbf{x}, t) + n(t)$$

noise

$$y = (I(t) - I_0)^2 + n(t)$$

Surviving beam current at end.

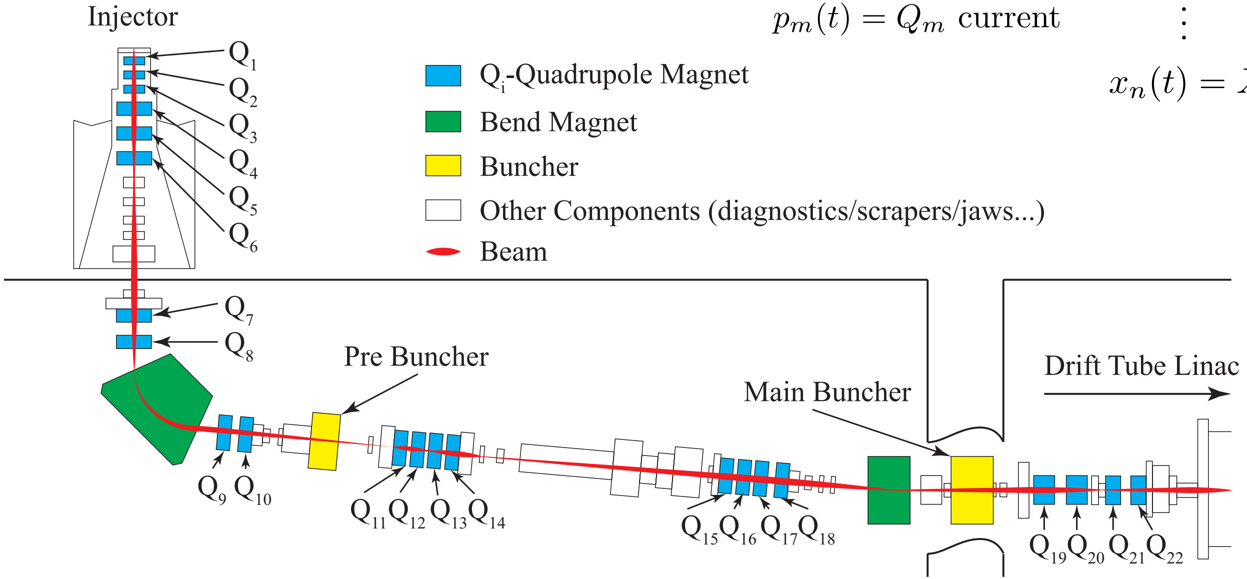
$$p_1(t) = Q_1 \text{ current} \quad \ldots \quad p_m(t) = Q_m \text{ current}$$

$$x_1(t) = X_{\text{rms}}(\text{position } 1)$$

$$x_2(t) = Y_{\text{rms}}(\text{position } 1)$$

$$\vdots$$

$$x_n(t) = X_{\text{rms}}(\text{position n})$$



$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

$$y = V(\mathbf{x}, t) + n(t)$$

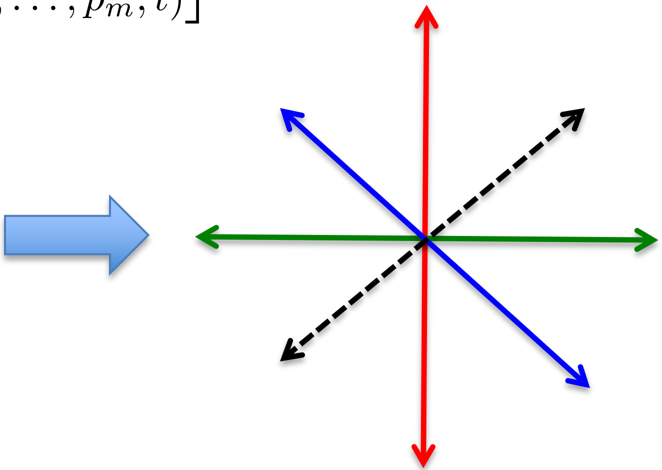
$$\frac{dp_1}{dt} = \sqrt{\alpha\omega_1} \cos(\omega_1 t + ky)$$

$$\frac{dp_2}{dt} = \sqrt{\alpha\omega_2} \cos(\omega_2 t + ky)$$

$$\frac{dp_3}{dt} = \sqrt{\alpha\omega_3} \cos(\omega_3 t + ky)$$

$$\vdots$$

$$\frac{dp_m}{dt} = \sqrt{\alpha\omega_m} \cos(\omega_m t + ky)$$



Dithering with different frequencies makes the parameters “perpendicular” in Hilbert space.

$$\omega_i = \omega r_i, \; r_i \neq r_j \implies \text{for any } t > 0$$

$$\lim_{\omega \to \infty} \langle \cos(\omega_i t), \cos(\omega_j t) \rangle = \lim_{\omega \to \infty} \int_0^t \cos(\omega_i \tau) \cos(\omega_j \tau) d\tau = 0$$

$$\begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{p}, t) = \begin{bmatrix} f_1(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \\ \vdots \\ f_n(x_1, \ldots, x_n, p_1, \ldots, p_m, t) \end{bmatrix}$$

$$y = V(\mathbf{x}, t) + n(t)$$

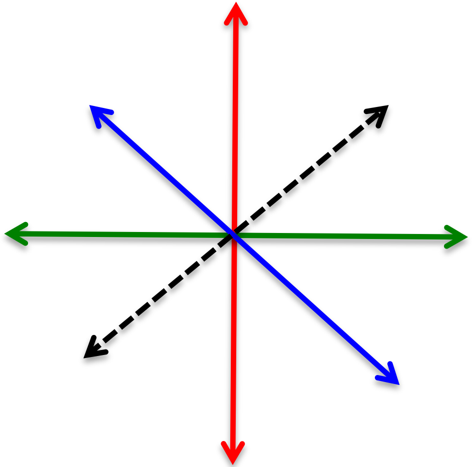
$$\frac{dp_1}{dt} = \sqrt{\alpha\omega_1}\cos(\omega_1 t + ky)$$

$$\frac{dp_2}{dt} = \sqrt{\alpha\omega_2}\cos(\omega_2 t + ky)$$

$$\frac{dp_3}{dt} = \sqrt{\alpha\omega_3}\cos(\omega_3 t + ky)$$

$$\vdots$$

$$\frac{dp_m}{dt} = \sqrt{\alpha\omega_m}\cos(\omega_m t + ky)$$



$$\frac{d\mathbf{p}}{dt} = -\frac{k\alpha}{2}\left(\nabla_{\mathbf{p}} V(\mathbf{x}, t)\right)^T$$

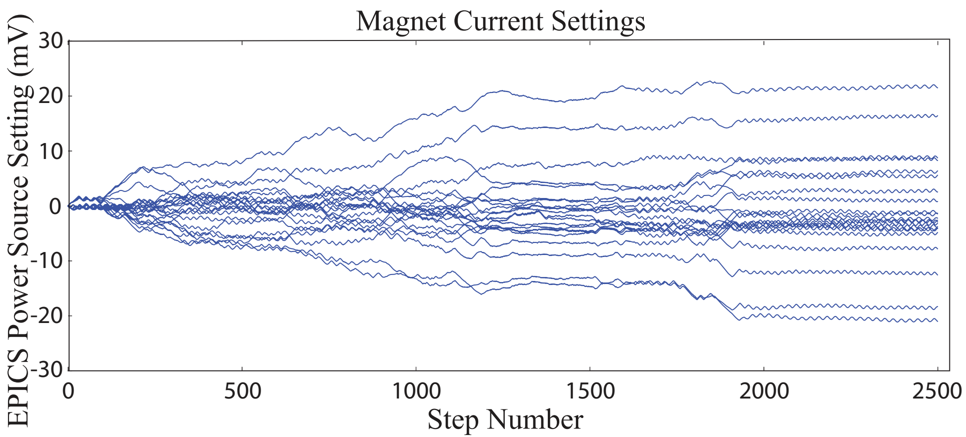
On average, the system performs minimizes the **unknown, time-varying** function V(x,t)

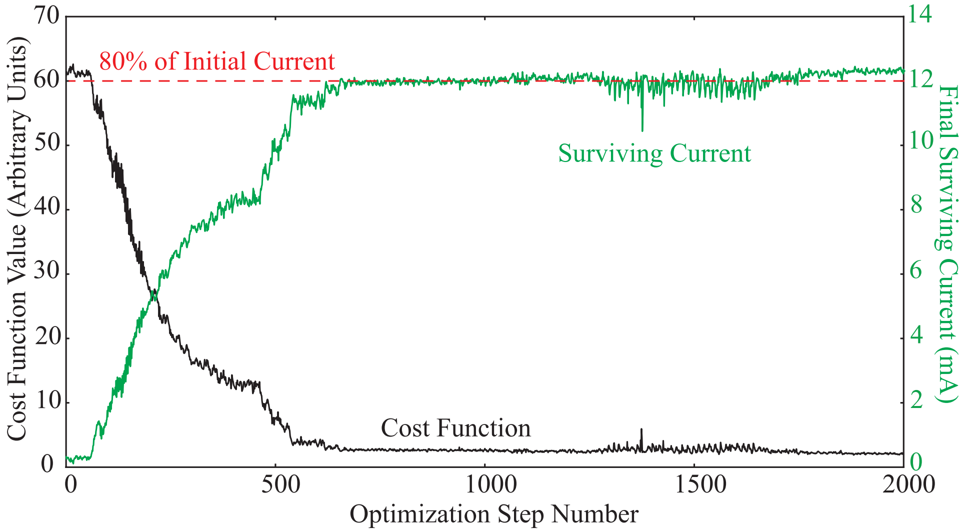
Allows simultaneous tuning of ALL parameters in parallel.

$$\omega_i = \omega r_i, \; r_i \neq r_j \implies \text{for any } t > 0$$

$$\lim_{\omega \to \infty} \langle \cos(\omega_i t), \cos(\omega_j t) \rangle = \lim_{\omega \to \infty} \int_0^t \cos(\omega_i \tau)\cos(\omega_j \tau) d\tau = 0$$

Accelerator Tuning and Control



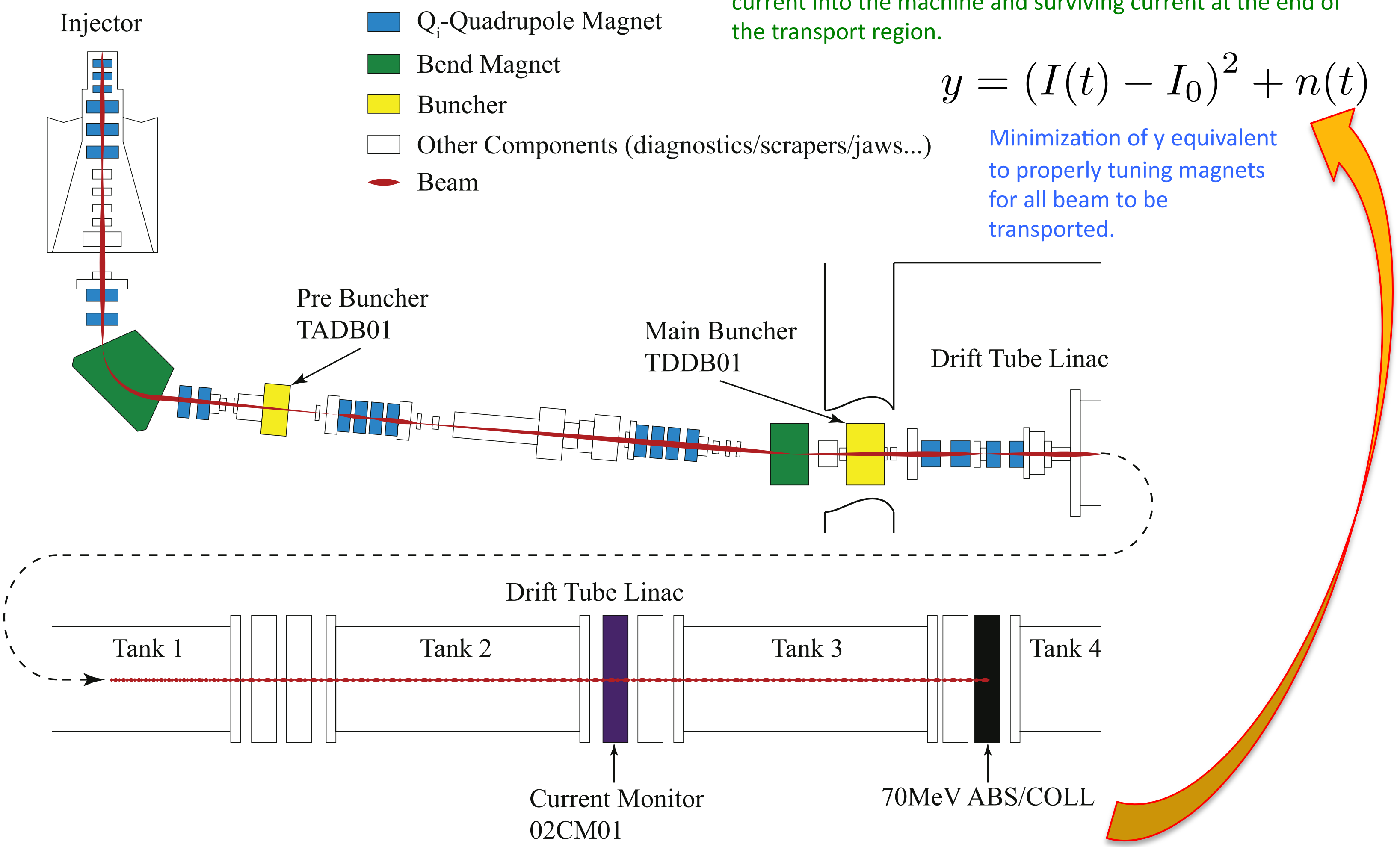


A. Scheinker et al, *2013*.

Cost is noisy measurement of the difference between initial current into the machine and surviving current at the end of the transport region.

$$y = (I(t) - I_0)^2 + n(t)$$

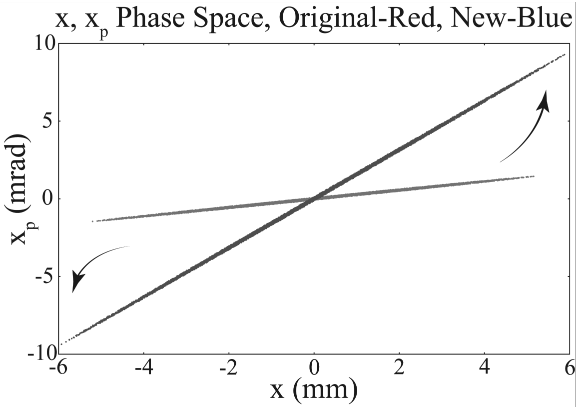
Minimization of y equivalent to properly tuning magnets for all beam to be transported.

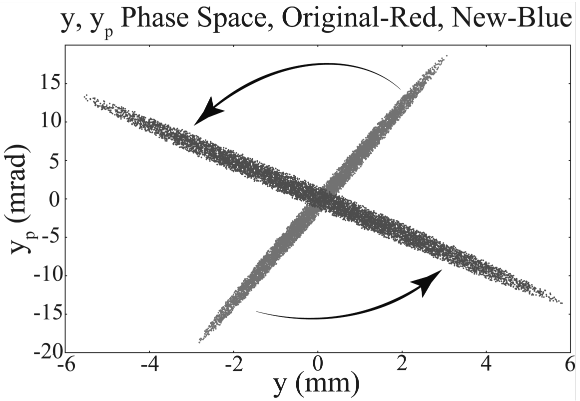


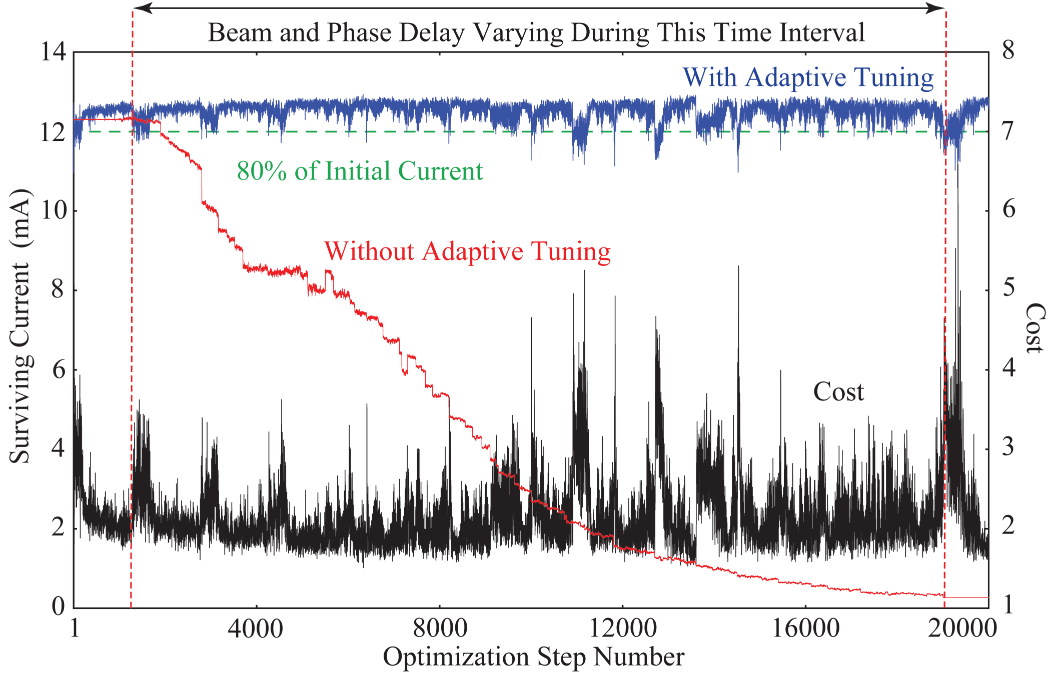
After the magnetic lattice was matched to transport the beam, beam phase space was continuously varied, and arbitrary phase drifts were introduced into the RF buncher cavities.

Without adaptive feedback all beam is quickly lost (red line in figure below).

With adaptive tuning the 22 quad magnetic lattice and 2 RF buncher cavities are continuously re-tuned to maintain maximal beam transmission and acceleration.



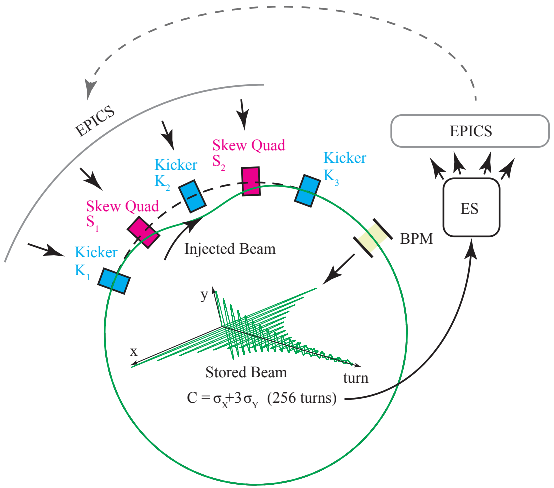


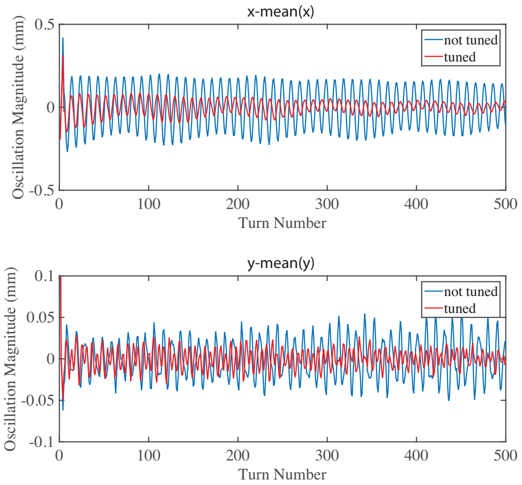


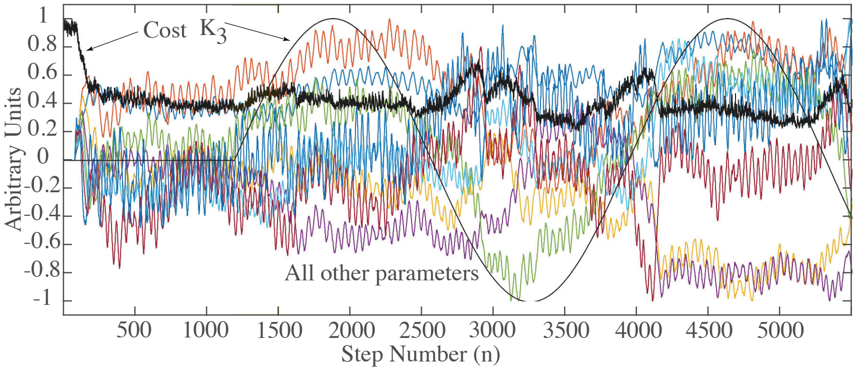
A. Scheinker, X. Pang, and L. Rybarcyk. "Model-independent particle accelerator tuning." *Physical Review Special Topics-Accelerators and Beams* 16.10 (2013): 102803.

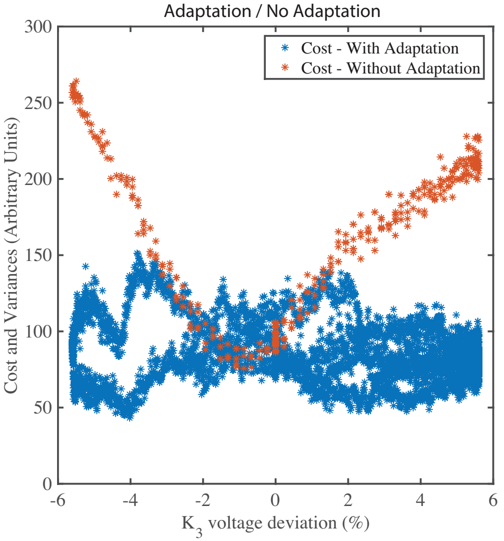
A. Scheinker et al., "Minimization of Betatron oscillations of electron beam injected into a time-varying lattice via extremum seeking," *IEEE Transactions on Control Systems Technology*, 2016.

SPEAR3 time-varying magnetic lattice

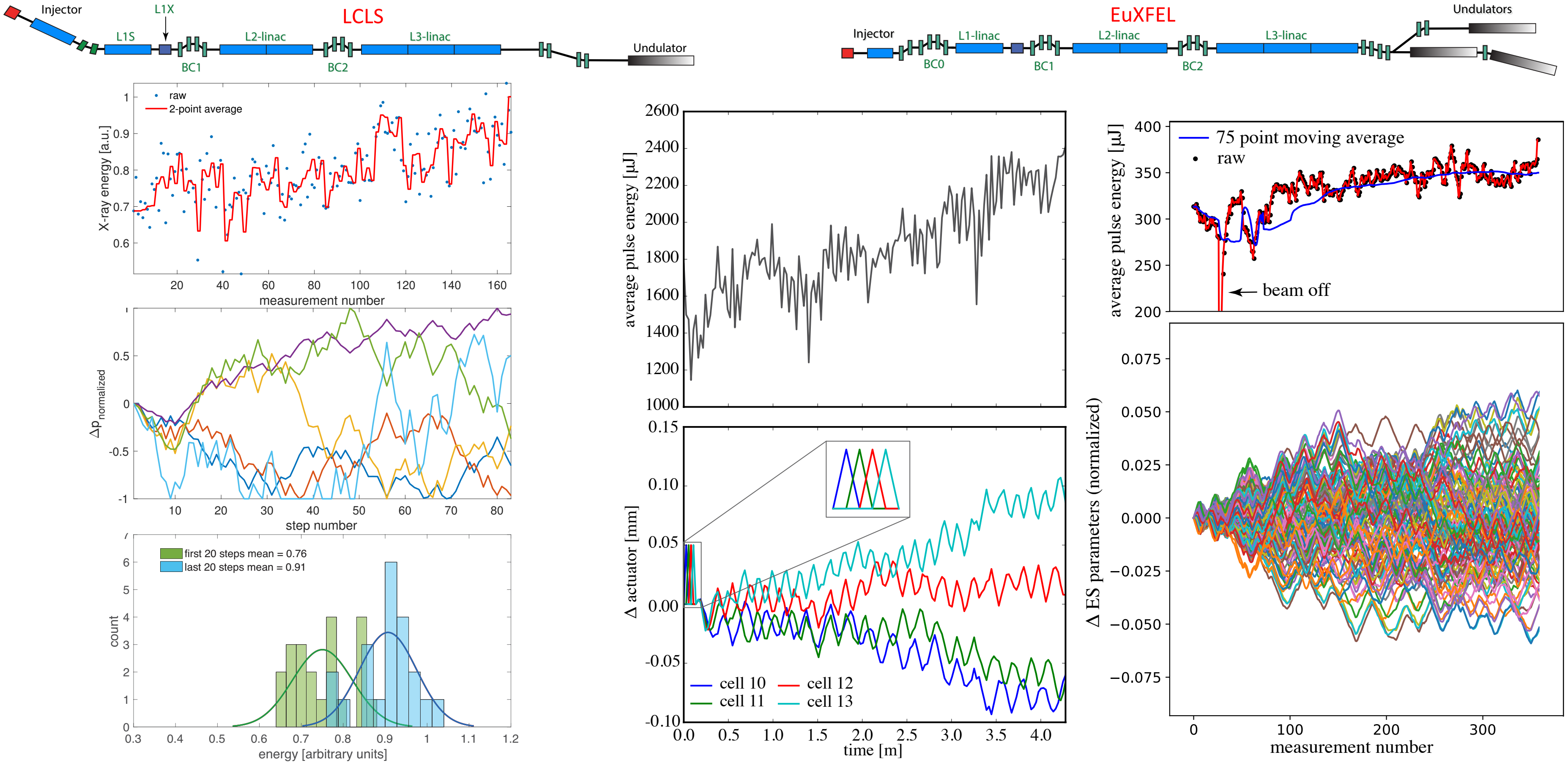








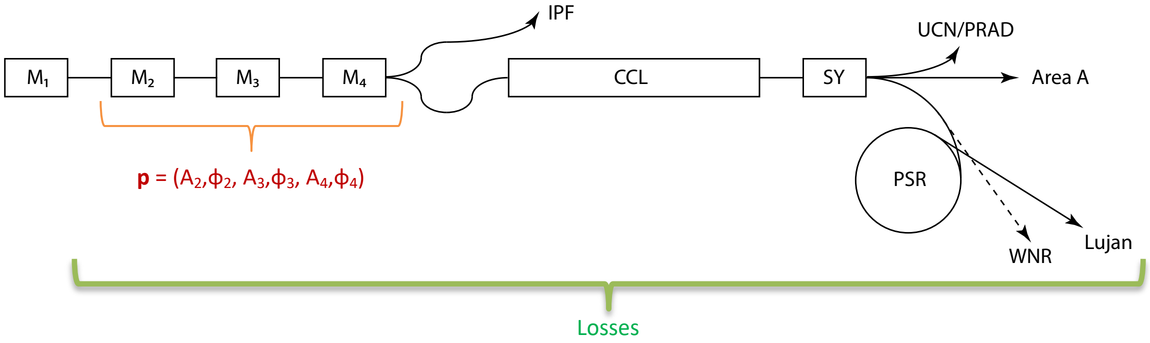
Model-independent Adaptive Feedback for pulse energy maximization at **LCLS & EuXFEL**

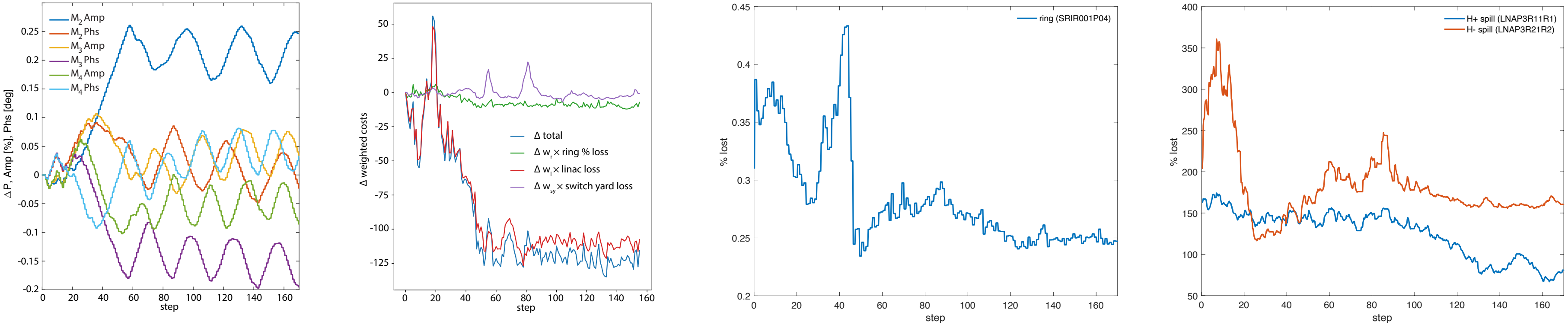


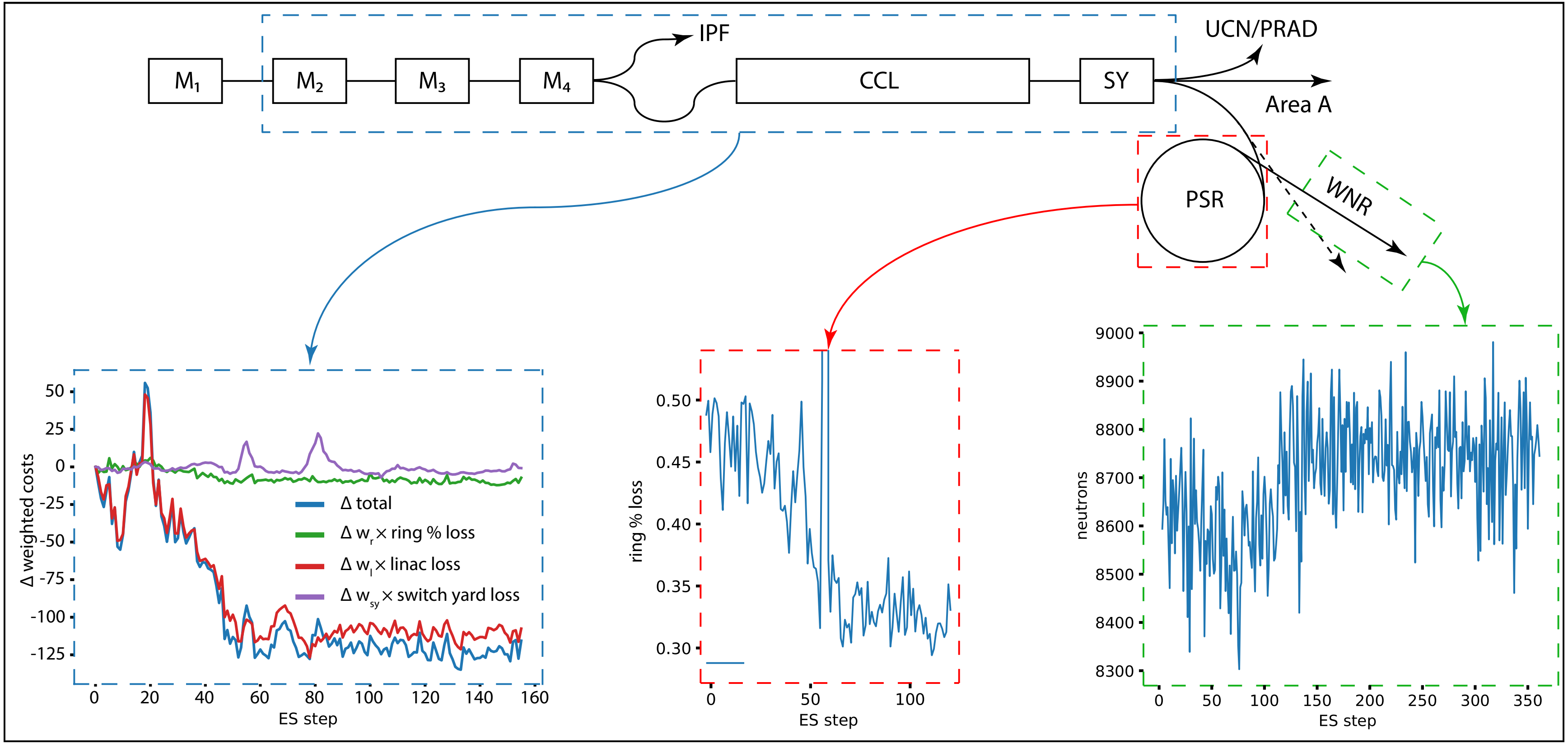
A. Scheinker, et al. "Model-independent tuning for maximizing free electron laser pulse energy." Physical Review Accelerators and Beams, 22.8, 082802, 2019. https://doi.org/10.1103/PhysRevAccelBeams.22.082802

LANSCE tuning, first results

Alexander Scheinker, Peter Naffziger, Antonio Garcia, Matt Hardy







Minimizing LINAC losses by tuning RF cavities

Minimizing PSR losses by tuning steering magnets

Minimizing WNR losses by tuning steering magnets and quads

A. Scheinker, P. Naffziger, and A. Garcia. "Extremum Seeking for Minimization of Beam Loss in the LANSCE Linear Accelerator by Tuning RF Cavities." *2020 American Control Conference (ACC)*. IEEE, 2020.

GUI running in the control room as a tuning tool

Quad ramp tuning

$\mathbf{p} = (p_1, p_2, \ldots, p_{10})$

p_1 = 05QD002P01

p_2 = 05QD003P01

p_3 = 06QD001P01

p_4 = 06QD003P01

p_5 = 07QD001P01

p_6 = 07QD003P01

p_7 = 08QD001P01

p_8 = 08QD003P01

p_9 = 09QD001P01

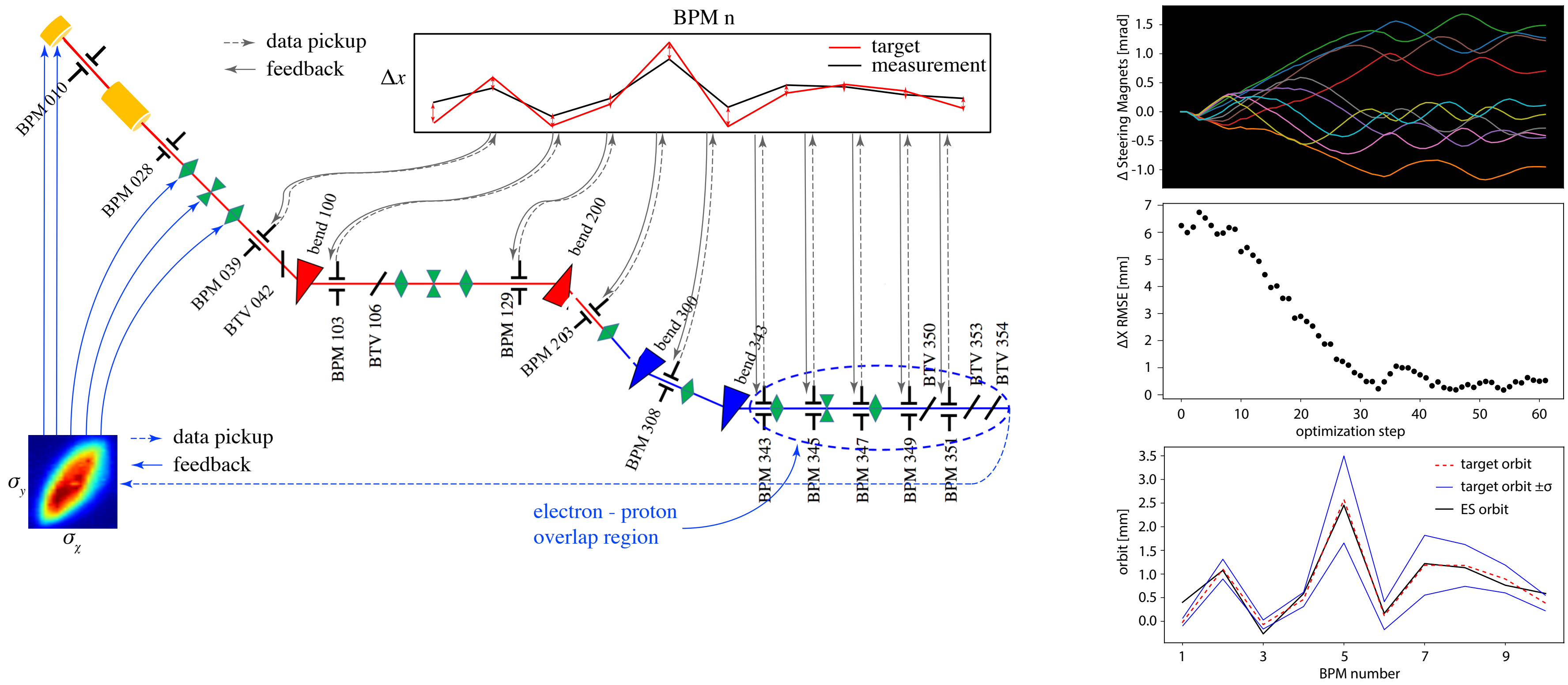
p_{10} = 09QD003P01

$C(\mathbf{p},t) = w_1 C_1(\mathbf{p},t) + w_2 C_2(\mathbf{p},t)$

$C_1(\mathbf{p},t)$ = LNAP3R2

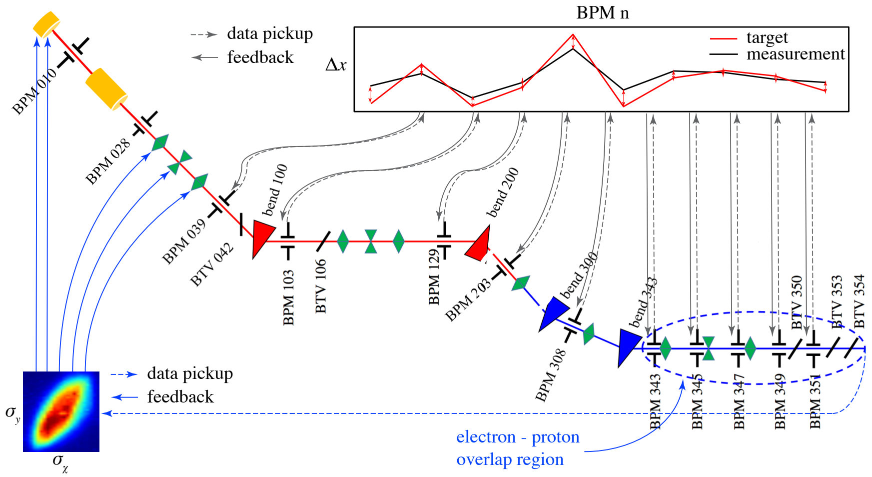
$C_2(\mathbf{p},t)$ = 15AP002R02

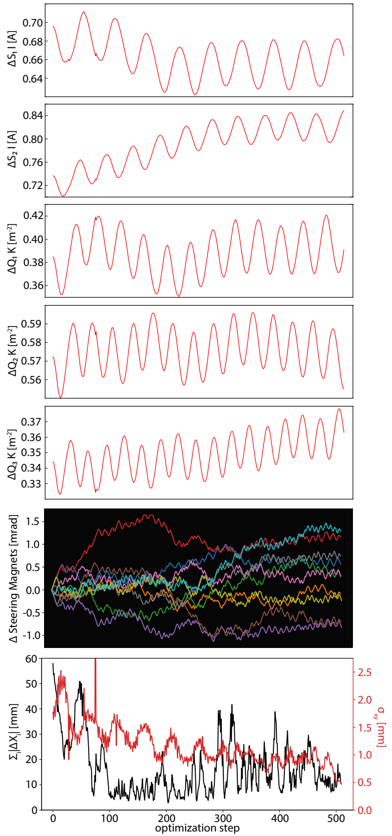
Multi-objective Optimization at **AWAKE**



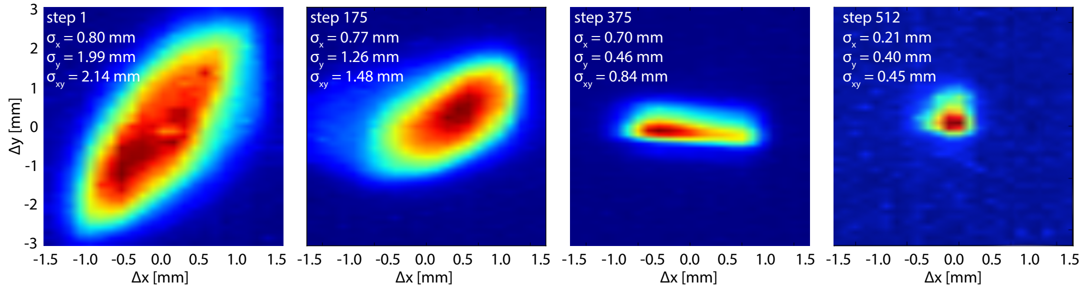
A. Scheinker, et al. "Online Mulit-Objective Particle Accelerator Optimization of the AWAKE Electron Beam Line for Simultaneous Emittance and Orbit Control." *AIP Advances* 10.5 (2020): 055320 https://doi.org/10.1063/5.0003423

Multi-objective Optimization at **AWAKE**





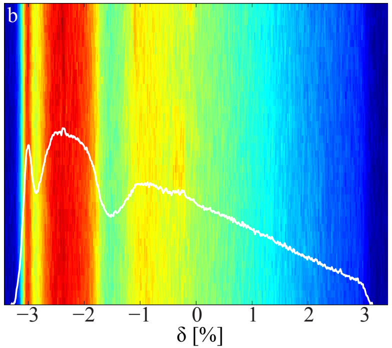
Tuning 15 components simultaneously: 2 solenoids, 3 quads, 10 steering magnets to simultaneously maintain the desired orbit and minimize emittance growth.

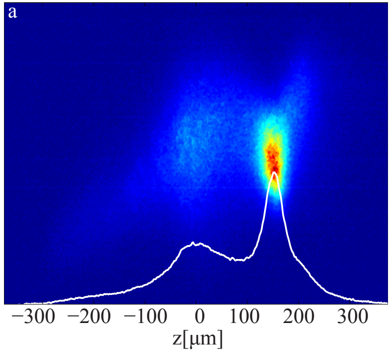


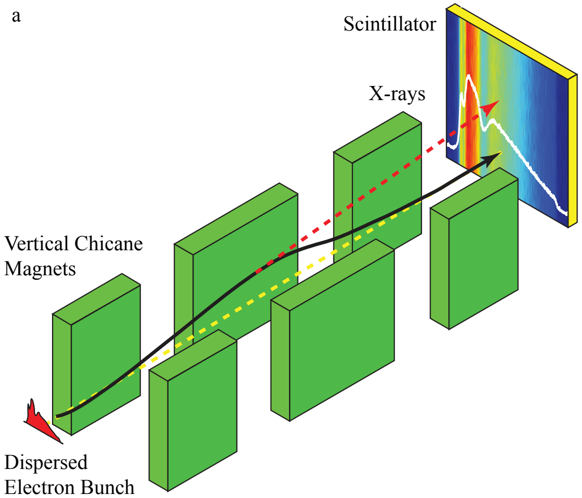
A. Scheinker, et al. "Online Mulit-Objective Particle Accelerator Optimization of the AWAKE Electron Beam Line for Simultaneous Emittance and Orbit Control." *AIP Advances* 10.5 (2020): 055320 https://doi.org/10.1063/5.0003423

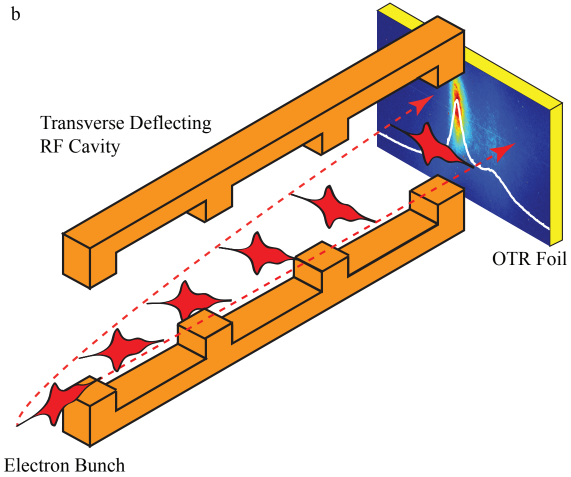
Non-Invasive Adaptive Diagnostics

A. Scheinker and S. Gessner, PRAB, 2015.



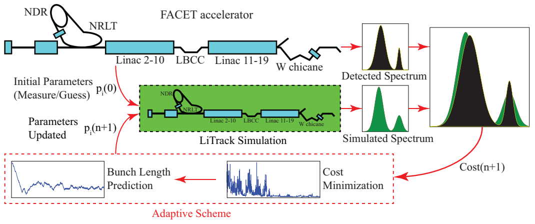


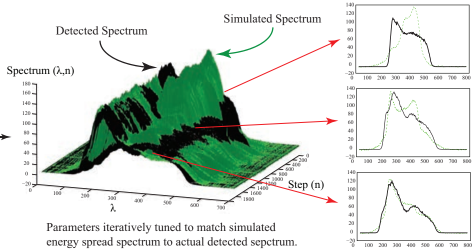




Adaptively tuned models for XTCAV longitudinal phase space predictions at **FACET**

Spectrum-based online model tuning.

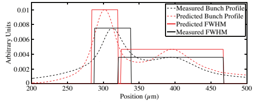


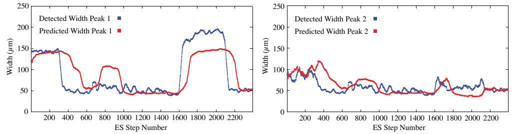


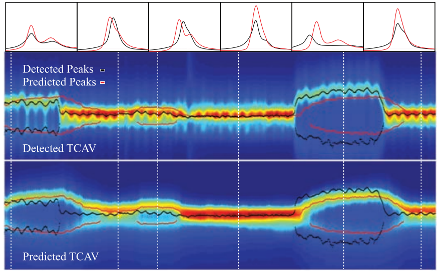
Parameters iteratively tuned to match simulated energy spread spectrum to actual detected sepctrum.

Energy spread spectrum matching leads to longitudinal bunch density prediction, as confirmed by comparison to detected TCAV measurements.

Time-varying bunch length predictions



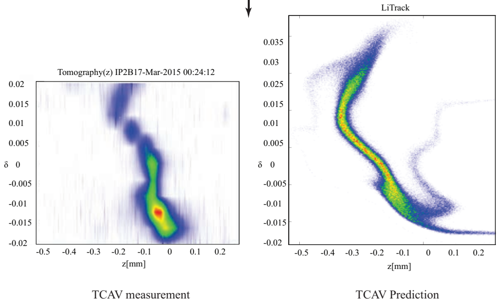


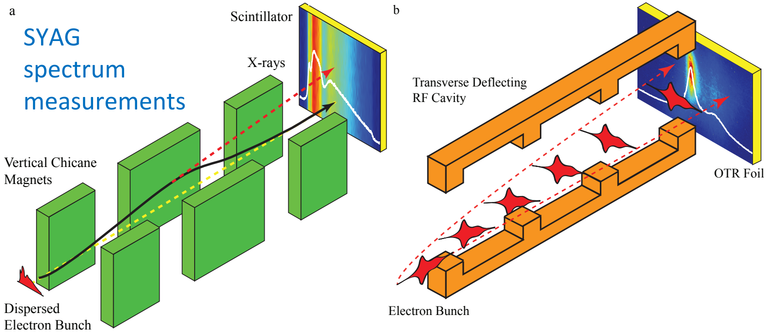


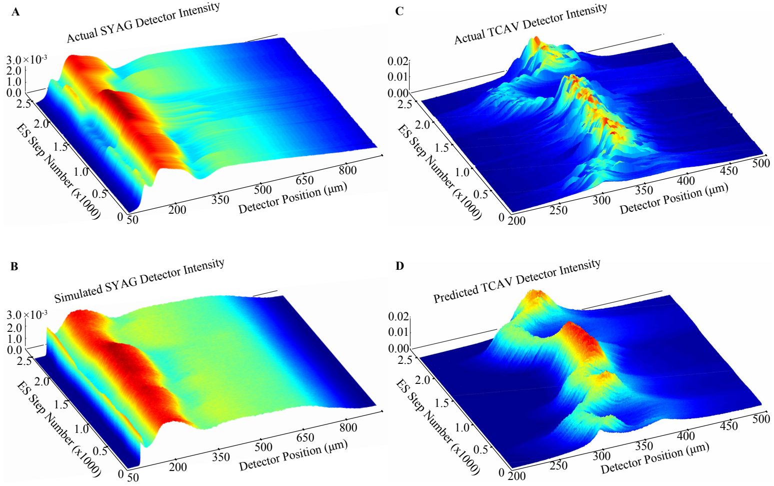
Bunch length and bunch-to-bunch separation tracking.

Time-varying phase space predictions

Longitudinal phase space prediction of XTCAV measurement.

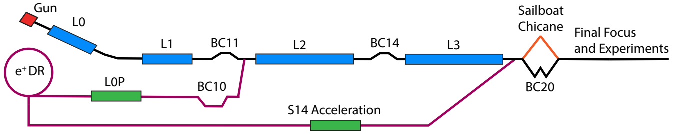






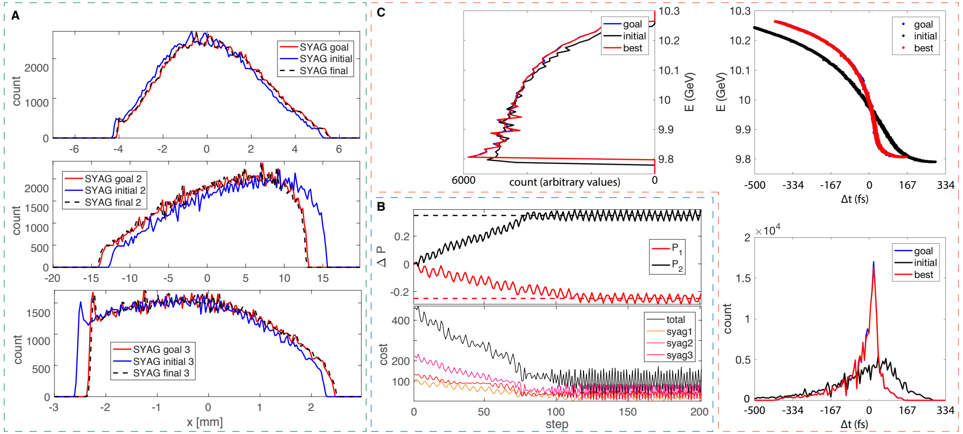
A. Scheinker and S. Gessner, "Adaptive method for electron bunch profile prediction." Physical Review Accelerators and Beams, 18(10), 102801, 2015. https://doi.org/10.1103/PhysRevSTAB.18.102801

Adaptively tuned models for XTCAV longitudinal phase space control & predictions at **FACET-II**



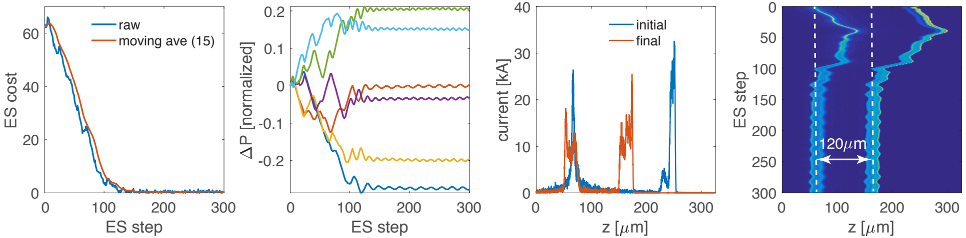
1. P1: L1 phase [deg]
2. P2: L1 voltage offset [dV/V]
3. P3: L2 phase [deg]
4. P4: L2 voltage offset [dV/V]
5. P5: x-offset
6. P6: y-offset

SYAG spectrum measurements



Longitudinal phase space XTCAV predictions

Adaptive model tuning



Non-invasive phase space diagnostics-based adaptive tuning of the longitudinal phase space distribution of the FACET-II beam. Automatic current profile tuning.

A. Scheinker, et al. "Adaptive model tuning studies for non-invasive diagnostics and feedback control of plasma wakefield acceleration at FACET-II." NIMA, 163902, 2020. https://doi.org/10.1016/j.nima.2020.163902

Reinforcement Learning / Optimal Adaptive Control

Bellman, Dynamic Programming

$$\dot{x} = f(x(t), u(t)), \quad J = h(x(t_f)) + \int_0^{t_f} g(x(t), u(t))dt$$

$$\frac{x(t+\Delta) - x(t)}{\Delta} \approx f(x(t), u(t)), \quad J \approx h(x(N)) + \Delta \sum_{n=1}^{N} g(x(n), u(n))$$

$$x(t+\Delta) = x(t) + \Delta f(x(t), u(t))$$

$$x(n+1) = f_d(x(n), u(n)), \quad J = h(x(N)) + \sum_{n=1}^{N} g_D(x(n), u(n))$$

$$J^*_{N-K,N}(x(N-K)) = \min_{u(N-K)} \left\{ g_D(x(N-K), u(N-K)) + J^*_{N-(K-1),N}(f_d(x(N-K), u(N-K)) \right\}$$

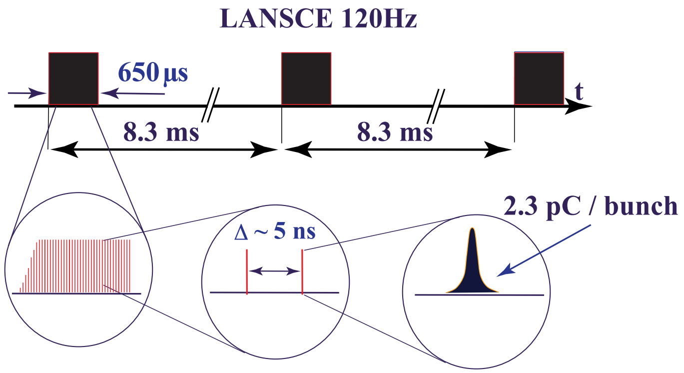
Markov Chain Decision Processes, Agents, Policies, etc .. = Optimal Adaptive Control with Uncertainties

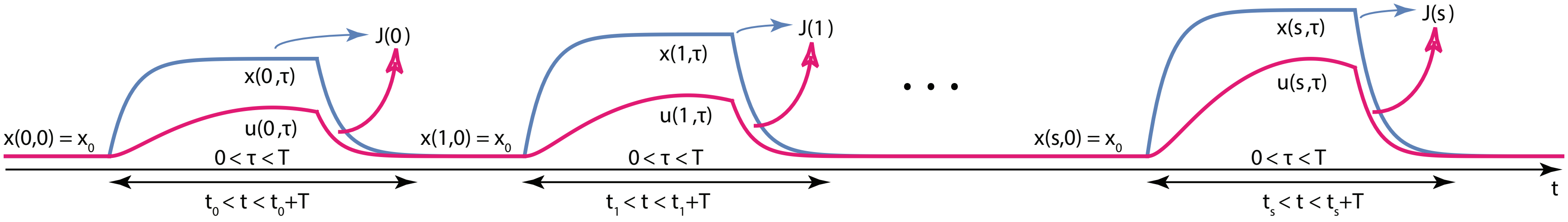
1. R. Bellman, "Dynamic programming and Lagrange multipliers." *Proceedings of the National Academy of Sciences of the United States of America* 42.10 (1956): 767.
2. D. E. Kirk, *Optimal Control Theory, An Introduction,* Dover Publications, 1970
3. F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, Optimal Control, Wiley, 2012
4. R. S. Sutton, and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Optimal Control of Unknown Time-Varying Systems

$$\dot{x}(\tau) = f(x,u,\tau), \quad \tau \in [t_0, t_0+T]$$

$$J(t_0, x(t_0), u) = F(x(T)) + \int_{t_0}^{t_0+T} G\left(x(\tau), u\left(\tau, x\right)\right) d\tau$$





Vector Valued Quadratic Tracker

$$\frac{d\mathbf{x}(t,\tau)}{d\tau} = A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), \quad (1)$$

$$\mathbf{y} = C(t)\mathbf{x}(t,\tau), \quad (2)$$

Vector Valued Quadratic Tracker

$$\frac{d\mathbf{x}(t,\tau)}{d\tau} = A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), \quad (1)$$

$$\mathbf{y} = C(t)\mathbf{x}(t,\tau), \quad (2)$$

$$\begin{aligned} J = {} & \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right) \\ & + \frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau \\ & + \frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau, \end{aligned} \quad (3)$$

Vector Valued Quadratic Tracker

$$
\begin{aligned}
\frac{d\mathbf{x}(t,\tau)}{d\tau} &= A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), && (1)\\
\mathbf{y} &= C(t)\mathbf{x}(t,\tau), && (2)
\end{aligned}
$$

$$
\begin{aligned}
J &= \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)\\
&\quad + \frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau\\
&\quad + \frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau, && (3)
\end{aligned}
$$

$$
\mathbf{u} = -K(\tau)\mathbf{x} + R^{-1}B^T\mathbf{v}(\tau), \quad K(t) = R^{-1}B^T S(\tau), \qquad (4)
$$

Vector Valued Quadratic Tracker

$$\frac{d\mathbf{x}(t,\tau)}{d\tau} = A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), \tag{1}$$

$$\mathbf{y} = C(t)\mathbf{x}(t,\tau), \tag{2}$$

$$\begin{aligned} J &= \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right) \\ &+\frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau \\ &+\frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau, \end{aligned} \tag{3}$$

$$\mathbf{u} = -K(\tau)\mathbf{x} + R^{-1}B^T\mathbf{v}(\tau), \quad K(t) = R^{-1}B^T S(\tau), \tag{4}$$

$$-\dot{S} = A^T S + SA - SBR^{-1}B^T S + C^T QC, \quad S(T) = C^T PC \tag{5}$$

Vector Valued Quadratic Tracker

$$\begin{aligned} \frac{d\mathbf{x}(t,\tau)}{d\tau} &= A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), \quad (1)\\ \mathbf{y} &= C(t)\mathbf{x}(t,\tau), \quad (2) \end{aligned}$$

$$\begin{aligned} J &= \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right) \\ &+ \frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau \\ &+ \frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau, \quad (3) \end{aligned}$$

$$\mathbf{u} = -K(\tau)\mathbf{x} + R^{-1}B^T\mathbf{v}(\tau), \quad K(t) = R^{-1}B^T S(\tau), \quad (4)$$

$$-\dot{S} = A^T S + SA - SBR^{-1}B^T S + C^T QC, \quad S(T) = C^T PC \quad (5)$$

$$-\dot{\mathbf{v}} = \left(A - BK(\tau)\right)^T \mathbf{v} + C^T Q\mathbf{r}(\tau), \quad \mathbf{v}(T) = C^T P\mathbf{r}(T). \quad (6)$$

- Linear
- Known dynamics
- Known trajectory
- Analytically known cost function
- Exact global optimal solution

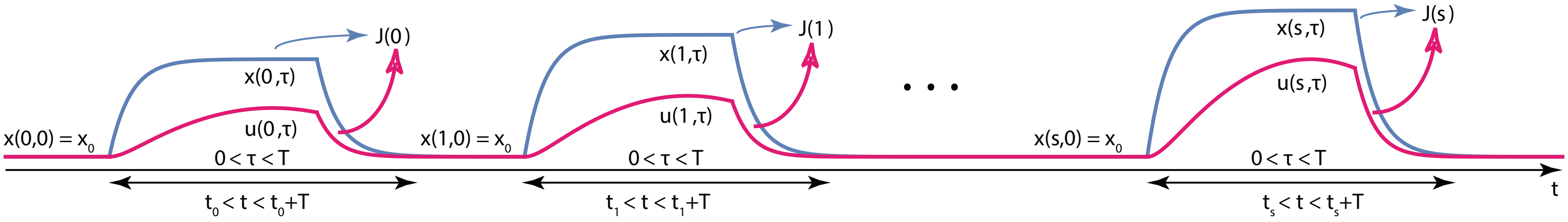
$$\begin{aligned} \frac{d\mathbf{x}(t,\tau)}{d\tau} &= A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau), \\ \mathbf{y} &= C(t)\mathbf{x}(t,\tau), \end{aligned}$$

$$\begin{aligned} J &= \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right) \\ &+ \frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau \\ &+ \frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau, \end{aligned}$$

$$\mathbf{u} = -K(\tau)\mathbf{x} + R^{-1}B^T\mathbf{v}(\tau), \quad K(t) = R^{-1}B^T S(\tau),$$

$$-\dot{S} = A^T S + SA - SBR^{-1}B^T S + C^T QC, \quad S(T) = C^T PC$$

$$-\dot{\mathbf{v}} = \left(A - BK(\tau)\right)^T \mathbf{v} + C^T Q\mathbf{r}(\tau), \quad \mathbf{v}(T) = C^T P\mathbf{r}(T).$$



$$\frac{d\mathbf{x}(t,\tau)}{d\tau} = \mathbf{f}(\mathbf{x}(t,\tau),\mathbf{u}(t,\tau),t), \quad \mathbf{x}(t,0)=\mathbf{x}_0 \in \mathbb{R}^{n_x}, \mathbf{u}\in\mathbb{R}^{n_u}, \quad \tau\in[0,T], \quad u_i(\cdot,\tau)\in L^2[0,T]$$

$$\mathbf{u}_i(t,\tau) = \sum_{j=1}^{m} a_{ij}(t)\phi_j(\tau) = \sum_{j=1}^{m/2} a_{ij}(t)\cos\left(\frac{2\pi jt}{T}\right) + \sum_{j=m/2+1}^{m} a_{ij}(t)\sin\left(\frac{2\pi jt}{T}\right)$$

$$\frac{da_{ij}(t)}{dt} = \sqrt{\alpha\omega_{ij}}\cos\left(\omega_{ij}t + kJ(\mathbf{a},t)\right), \quad \mathbf{a} = \begin{bmatrix} a_{11}(t) & \dots & a_{1m}(t) \\ \vdots & \ddots & \vdots \\ a_{n_u1}(t) & \dots & a_{n_um}(t) \end{bmatrix}$$

$$J(\mathbf{a},t) = F(\mathbf{x}(t,T)) + \int_0^T G(\mathbf{x}(t,\mathbf{u},\tau),\mathbf{u}(\mathbf{a},\tau))d\tau$$

$$\frac{d\bar{a}_{ij}}{dt} = -\frac{k\alpha}{2}\frac{\partial J(\bar{\mathbf{a}},t)}{\partial \bar{a}_{ij}}$$

- Linear
- Known dynamics
- Known trajectory
- Analytically known cost function
- Exact global optimal solution

$$\frac{d\mathbf{x}(t,\tau)}{d\tau} = A(t)\mathbf{x}(t,\tau) + B(t)\mathbf{u}(c(t),\tau),$$
$$\mathbf{y} = C(t)\mathbf{x}(t,\tau),$$

$$J = \frac{1}{2}\left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right)^T P \left(C\mathbf{x}(t,T) - \mathbf{r}(T)\right) + \frac{1}{2}\int_0^T \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right)^T Q \left(C\mathbf{x}(t,\tau) - \mathbf{r}(\tau)\right) d\tau + \frac{1}{2}\int_0^T \mathbf{u}(t,\tau)^T R\mathbf{u}(t,\tau) d\tau,$$

$$\mathbf{u} = -K(\tau)\mathbf{x} + R^{-1}B^T\mathbf{v}(\tau), \quad K(t) = R^{-1}B^T S(\tau),$$

$$-\dot{S} = A^T S + SA - SBR^{-1}B^T S + C^T QC, \quad S(T) = C^T PC$$

$$-\dot{\mathbf{v}} = \left(A - BK(\tau)\right)^T \mathbf{v} + C^T Q\mathbf{r}(\tau), \quad \mathbf{v}(T) = C^T P\mathbf{r}(T).$$

- Linear or Nonlinear
- Unknown dynamics
- Unknown trajectory
- Analytically unknown cost function
- Arbitrarily accurate globally optimal solution for convex costs
- Iterative approximation
- Repeatable systems

$$\mathbf{a} = \begin{bmatrix} a_{11}(t) & \dots & a_{1m}(t) \\ \vdots & \ddots & \vdots \\ a_{n_u 1}(t) & \dots & a_{n_u m}(t) \end{bmatrix}$$

$$\mathbf{u}_i(t,\tau) = \sum_{j=1}^{m} a_{ij}(t)\phi_j(\tau), \quad \phi_j(\tau) \in L^2[0,T]$$

$$\hat{J} = J + n(t)$$

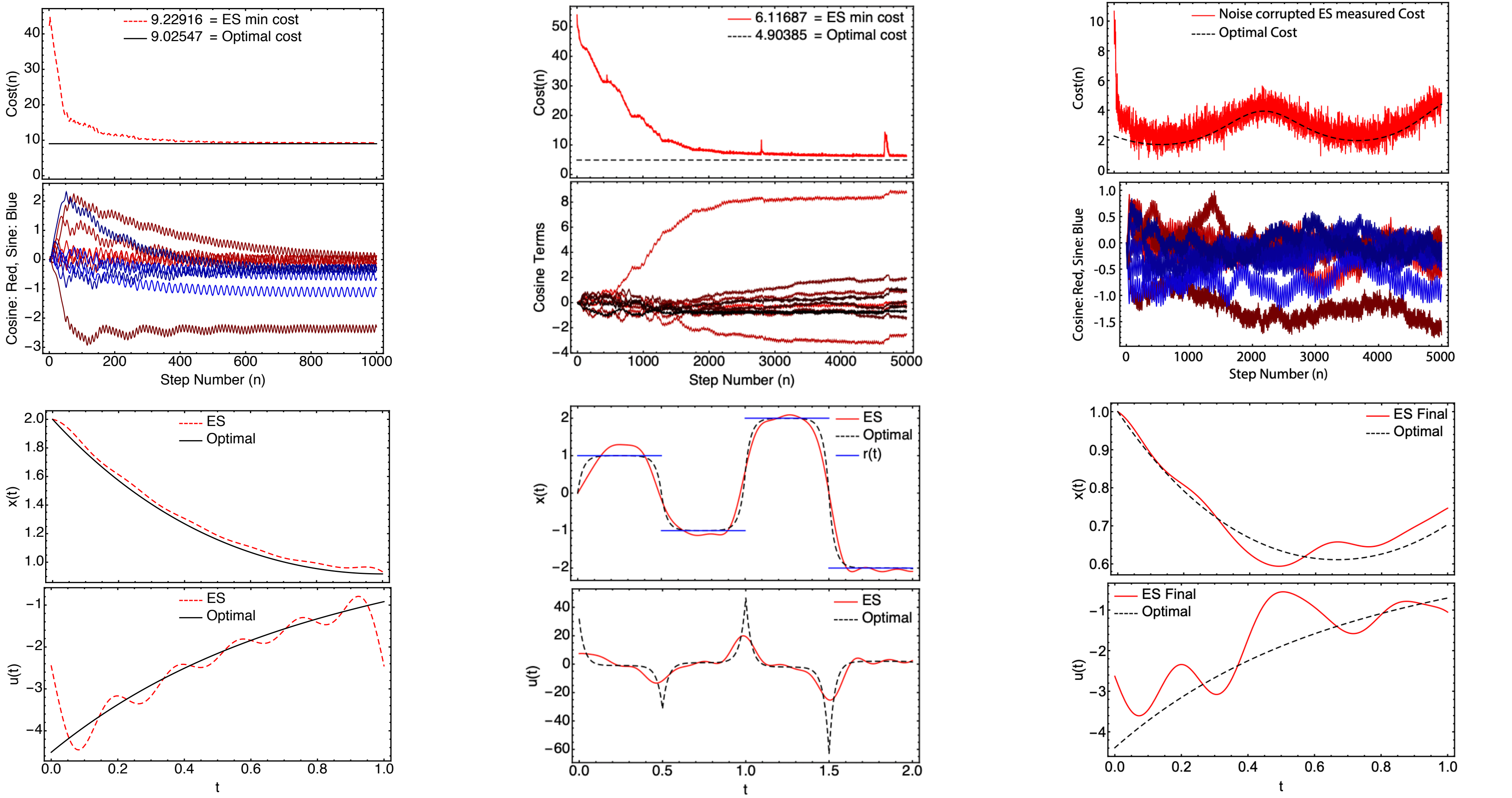
$$\frac{da_{ij}(t)}{dt} = \sqrt{\alpha\omega_{ij}}\cos\left(\omega_{ij}t + k\hat{J}\right)$$

$$\frac{d\bar{a}_{ij}}{dt} = -\frac{k\alpha}{2}\frac{\partial J(\bar{\mathbf{a}},t)}{\partial \bar{a}_{ij}}$$

$$\dot{x} = x + u, \quad J = x^2(1) + \int_0^1 \left(x^2(t) + u^2(t)\right) dt$$

$$e(t) = x(t) - r(t), \quad J = e(1)^2 + \int_0^1 \left(10e^2(t) + \frac{1}{50}u^2(t)\right) dt$$

$$\dot{x} = a(t)x + b(t)u, \quad \hat{J}(t) = n(t) + x^2(1) + \int_0^1 \left(x^2(t) + u^2(t)\right) dt$$



$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{x1} \\ u_{x2} \end{bmatrix}}_{\mathbf{u}_x}$$

$$\underbrace{\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix}}_{\dot{\mathbf{y}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{y1} \\ u_{y2} \end{bmatrix}}_{\mathbf{u}_y}$$

$$\mathbf{u}_x = -K_{ES}(t,\tau)\mathbf{x}, \quad \mathbf{x}(0) = (x_{10}, x_{20})$$
$$\mathbf{u}_y = -K_{ES}(t,\tau)\mathbf{y}, \quad \mathbf{y}(0) = (y_{10}, y_{20})$$

$$K_{ES}(t,\tau) = \begin{bmatrix} k_{11}(t,\tau) & k_{12}(t,\tau)] \\ k_{21}(t,\tau) & k_{22}(t,\tau) \end{bmatrix}$$

$$k_{lp}(t,\tau) = \sum_{j=1}^{m} \left[a_j^{lp}(t)\cos(\nu\tau) + b_j^{lp}(t)\sin(\nu_j\tau) \right]$$

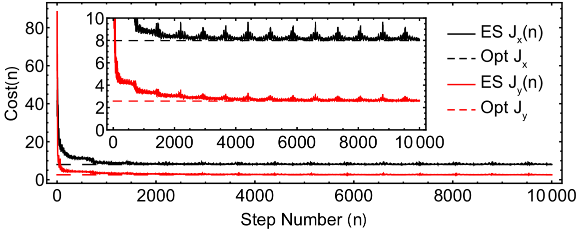
$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{x1} \\ u_{x2} \end{bmatrix}}_{\mathbf{u}_x}$$

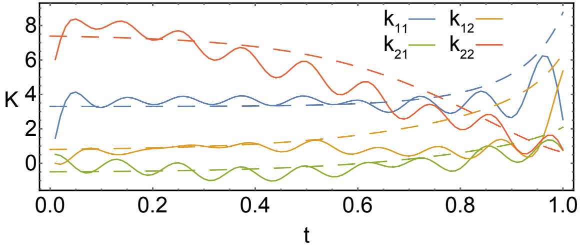
$$\underbrace{\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix}}_{\dot{\mathbf{y}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{y1} \\ u_{y2} \end{bmatrix}}_{\mathbf{u}_y}$$

$$\mathbf{u}_x = -K_{ES}(t,\tau)\mathbf{x}, \quad \mathbf{x}(0) = (x_{10}, x_{20})$$
$$\mathbf{u}_y = -K_{ES}(t,\tau)\mathbf{y}, \quad \mathbf{y}(0) = (y_{10}, y_{20})$$

$$K_{ES}(t,\tau) = \begin{bmatrix} k_{11}(t,\tau) & k_{12}(t,\tau)] \\ k_{21}(t,\tau) & k_{22}(t,\tau) \end{bmatrix}$$

$$k_{lp}(t,\tau) = \sum_{j=1}^{m} \left[a_j^{lp}(t)\cos(\nu\tau) + b_j^{lp}(t)\sin(\nu_j\tau) \right]$$





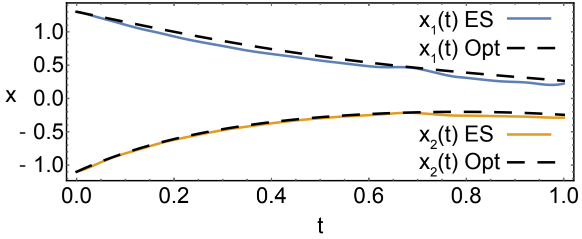
$$\underbrace{\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{x1} \\ u_{x2} \end{bmatrix}}_{\mathbf{u}_x}$$

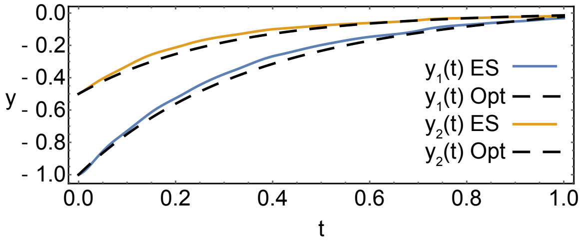
$$\underbrace{\begin{bmatrix} \dot{y}_1 \\ \dot{y}_2 \end{bmatrix}}_{\dot{\mathbf{y}}} = \underbrace{\begin{bmatrix} 1.0 & 0.25 \\ 0.3 & 0.7 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}}_{\mathbf{y}} + \underbrace{\begin{bmatrix} 1.0 & 0.1 \\ 0.2 & 0.5 \end{bmatrix}}_{B} \underbrace{\begin{bmatrix} u_{y1} \\ u_{y2} \end{bmatrix}}_{\mathbf{u}_y}$$

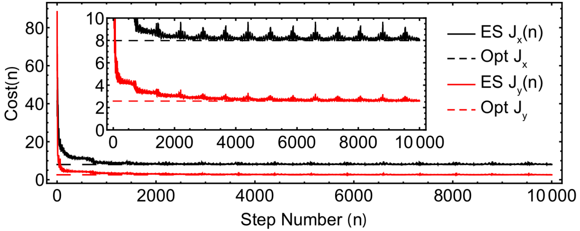
$$\mathbf{u}_x = -K_{ES}(t,\tau)\mathbf{x}, \quad \mathbf{x}(0) = (x_{10}, x_{20})$$
$$\mathbf{u}_y = -K_{ES}(t,\tau)\mathbf{y}, \quad \mathbf{y}(0) = (y_{10}, y_{20})$$

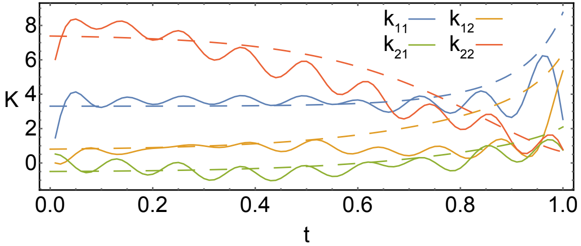
$$K_{ES}(t,\tau) = \begin{bmatrix} k_{11}(t,\tau) & k_{12}(t,\tau) \\ k_{21}(t,\tau) & k_{22}(t,\tau) \end{bmatrix}$$

$$k_{lp}(t,\tau) = \sum_{j=1}^{m} \left[a_j^{lp}(t)\cos(\nu\tau) + b_j^{lp}(t)\sin(\nu_j\tau) \right]$$









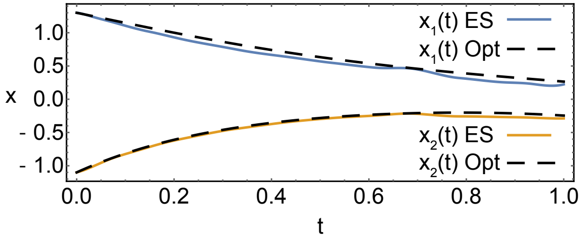
$$\underbrace{\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}x_1\\ x_2\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{x1}\\ u_{x2}\end{bmatrix}}_{\mathbf{u}_x}$$

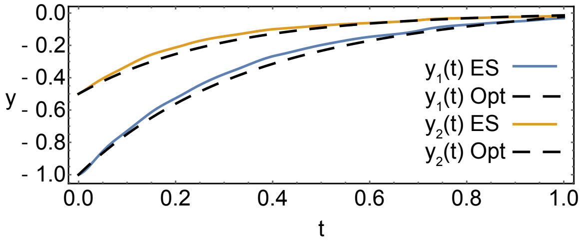
$$\underbrace{\begin{bmatrix}\dot{y}_1\\ \dot{y}_2\end{bmatrix}}_{\dot{\mathbf{y}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}y_1\\ y_2\end{bmatrix}}_{\mathbf{y}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{y1}\\ u_{y2}\end{bmatrix}}_{\mathbf{u}_y}$$

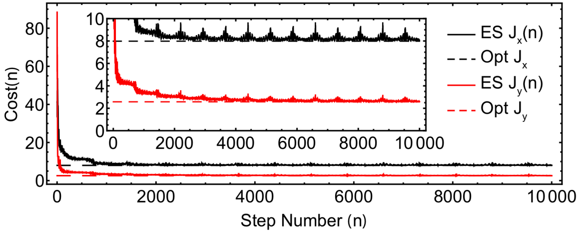
$$\mathbf{u}_x = -K_{ES}(t,\tau)\mathbf{x}, \quad \mathbf{x}(0) = (x_{10}, x_{20})$$
$$\mathbf{u}_y = -K_{ES}(t,\tau)\mathbf{y}, \quad \mathbf{y}(0) = (y_{10}, y_{20})$$

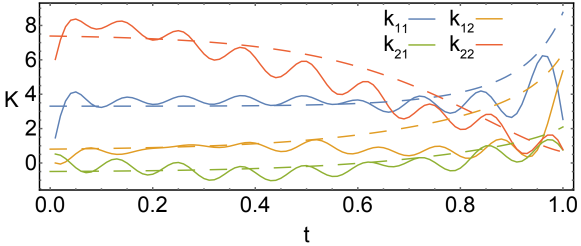
$$K_{ES}(t,\tau) = \begin{bmatrix} k_{11}(t,\tau) & k_{12}(t,\tau)] \\ k_{21}(t,\tau) & k_{22}(t,\tau)\end{bmatrix}$$

$$k_{lp}(t,\tau) = \sum_{j=1}^{m}\left[a_j^{lp}(t)\cos(\nu\tau) + b_j^{lp}(t)\sin(\nu_j\tau)\right]$$









$$\underbrace{\begin{bmatrix}\dot{z}_1\\ \dot{z}_2\end{bmatrix}}_{\dot{\mathbf{z}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}z_1\\ z_2\end{bmatrix}}_{\mathbf{z}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{z1}\\ u_{z2}\end{bmatrix}}_{\mathbf{u}_z}$$

$$\mathbf{u}_z = -K_{ES}(t,\tau)\mathbf{z}, \quad \mathbf{z}(0) = (z_{10}, z_{20})$$

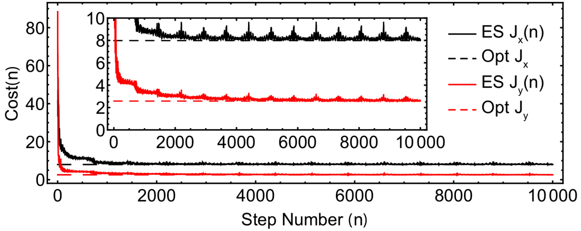
$$\underbrace{\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}x_1\\ x_2\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{x1}\\ u_{x2}\end{bmatrix}}_{\mathbf{u}_x}$$

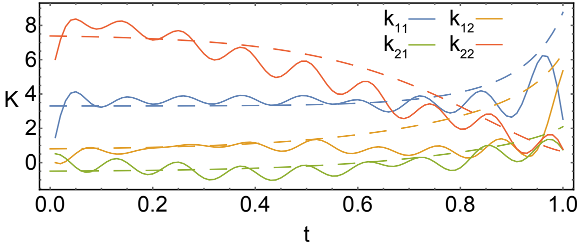
$$\underbrace{\begin{bmatrix}\dot{y}_1\\ \dot{y}_2\end{bmatrix}}_{\dot{\mathbf{y}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}y_1\\ y_2\end{bmatrix}}_{\mathbf{y}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{y1}\\ u_{y2}\end{bmatrix}}_{\mathbf{u}_y}$$

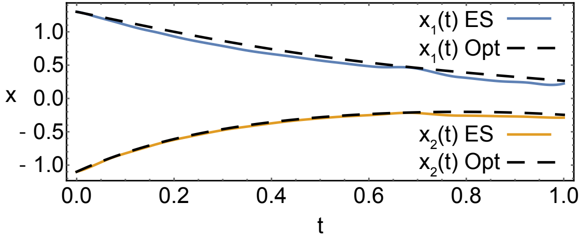
$$\mathbf{u}_x = -K_{ES}(t,\tau)\mathbf{x}, \quad \mathbf{x}(0) = (x_{10}, x_{20})$$
$$\mathbf{u}_y = -K_{ES}(t,\tau)\mathbf{y}, \quad \mathbf{y}(0) = (y_{10}, y_{20})$$

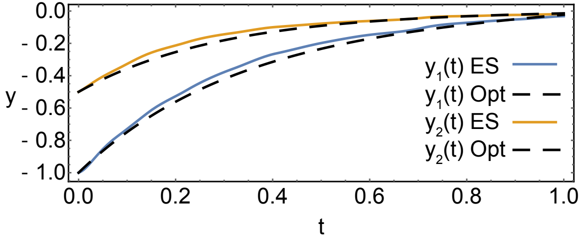
$$K_{ES}(t,\tau) = \begin{bmatrix} k_{11}(t,\tau) & k_{12}(t,\tau)] \\ k_{21}(t,\tau) & k_{22}(t,\tau)\end{bmatrix}$$

$$k_{lp}(t,\tau) = \sum_{j=1}^{m}\left[a_j^{lp}(t)\cos(\nu\tau) + b_j^{lp}(t)\sin(\nu_j\tau)\right]$$



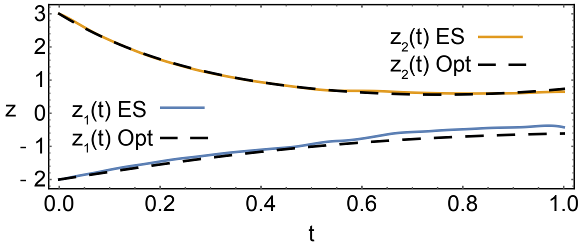






$$\underbrace{\begin{bmatrix}\dot{z}_1\\ \dot{z}_2\end{bmatrix}}_{\dot{\mathbf{z}}} = \underbrace{\begin{bmatrix}1.0 & 0.25\\ 0.3 & 0.7\end{bmatrix}}_{A}\underbrace{\begin{bmatrix}z_1\\ z_2\end{bmatrix}}_{\mathbf{z}} + \underbrace{\begin{bmatrix}1.0 & 0.1\\ 0.2 & 0.5\end{bmatrix}}_{B}\underbrace{\begin{bmatrix}u_{z1}\\ u_{z2}\end{bmatrix}}_{\mathbf{u}_z}$$

$$\mathbf{u}_z = -K_{ES}(t,\tau)\mathbf{z}, \quad \mathbf{z}(0) = (z_{10}, z_{20})$$



$$V_{RF}(t) = A(t)\cos(\omega_{RF}t + \theta_{RF}(t))$$

$$V_{LO}(t) = \cos(\omega_{LO}t + \theta_{LO}(t)), \quad \omega_{LO} = \omega_{RF} - \omega_{IF}$$

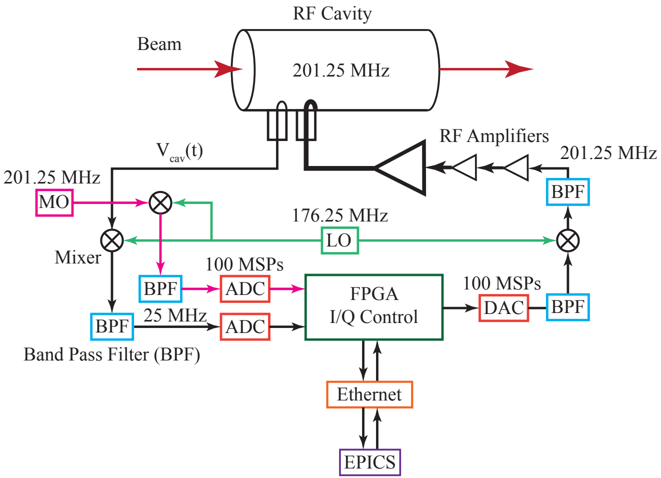
$$V_{RF}(t) \times V_{LO}(t) = \frac{A(t)}{2}\cos((\omega_{RF} + \omega_{LO})t + \theta_{RF}(t) + \theta_{LO}(t)) - \frac{A(t)}{2}\sin((\omega_{RF} - \omega_{LO})t + \theta_{RF}(t) - \theta_{LO}(t))$$

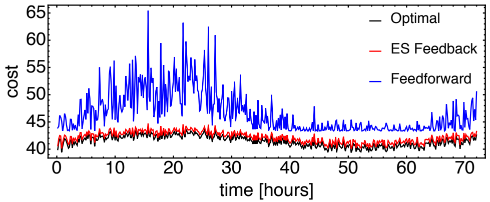
$$F,\ G \quad \Longrightarrow \quad V_{IF}(t) = A(t)\cos(\omega_{IF}t + \theta_{RF}(t) - \theta_{LO}(t))$$

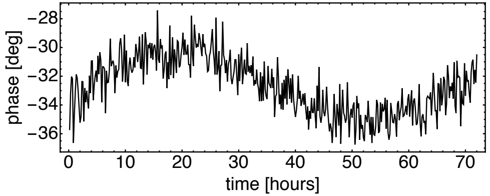
$$= A(t)\cos(\theta(t))\cos(\omega_{IF}t) - A(t)\sin(\theta(t))\sin(\omega_{IF}t)$$

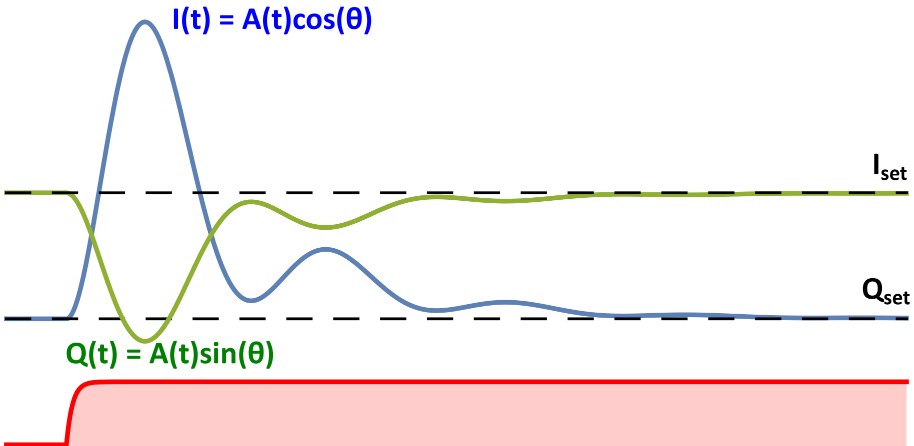
$$T_s = \frac{2\pi n}{4\omega_{IF}} \quad \Longrightarrow \quad I(0), -Q(T_s), -I(2T_s), Q(3T_s), \dots$$

$$A(nT_s) \approx \sqrt{I\left((n-1)T_s\right)^2 + Q(nT_s)^2}, \quad \theta(nT_s) \approx \arctan\left(\frac{Q(nT_s)}{I((n-1)T_s)}\right)$$



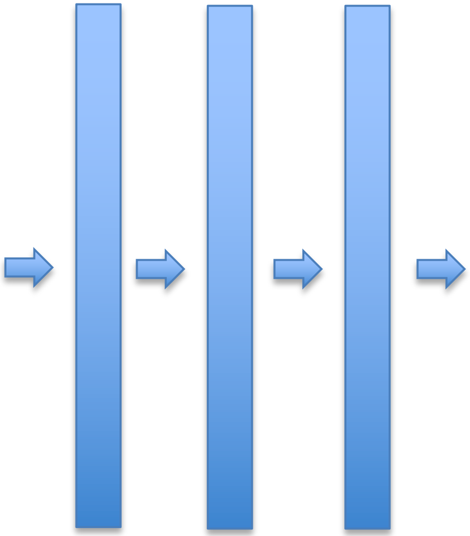




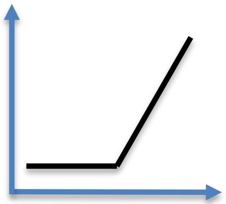


A. Scheinker and D. Scheinker. "Extremum seeking for optimal control problems with unknown time-varying systems and unknown objective functions." *International Journal of Adaptive Control and Signal Processing* (2020). https://doi.org/10.1002/acs.3097

Machine Learning



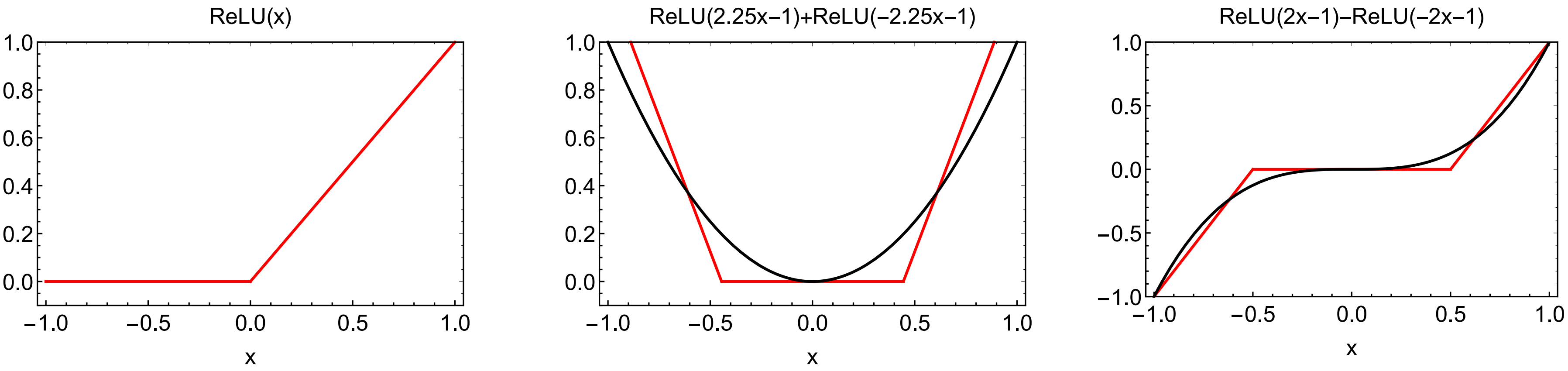
$$\begin{bmatrix} x_1^i \\ x_2^i \end{bmatrix} \begin{bmatrix} w_{11}^1 & w_{12}^1 \end{bmatrix} + b_1^1 = x_1^i w_{11}^1 + x_2^i w_{12}^1 + b_1^1$$



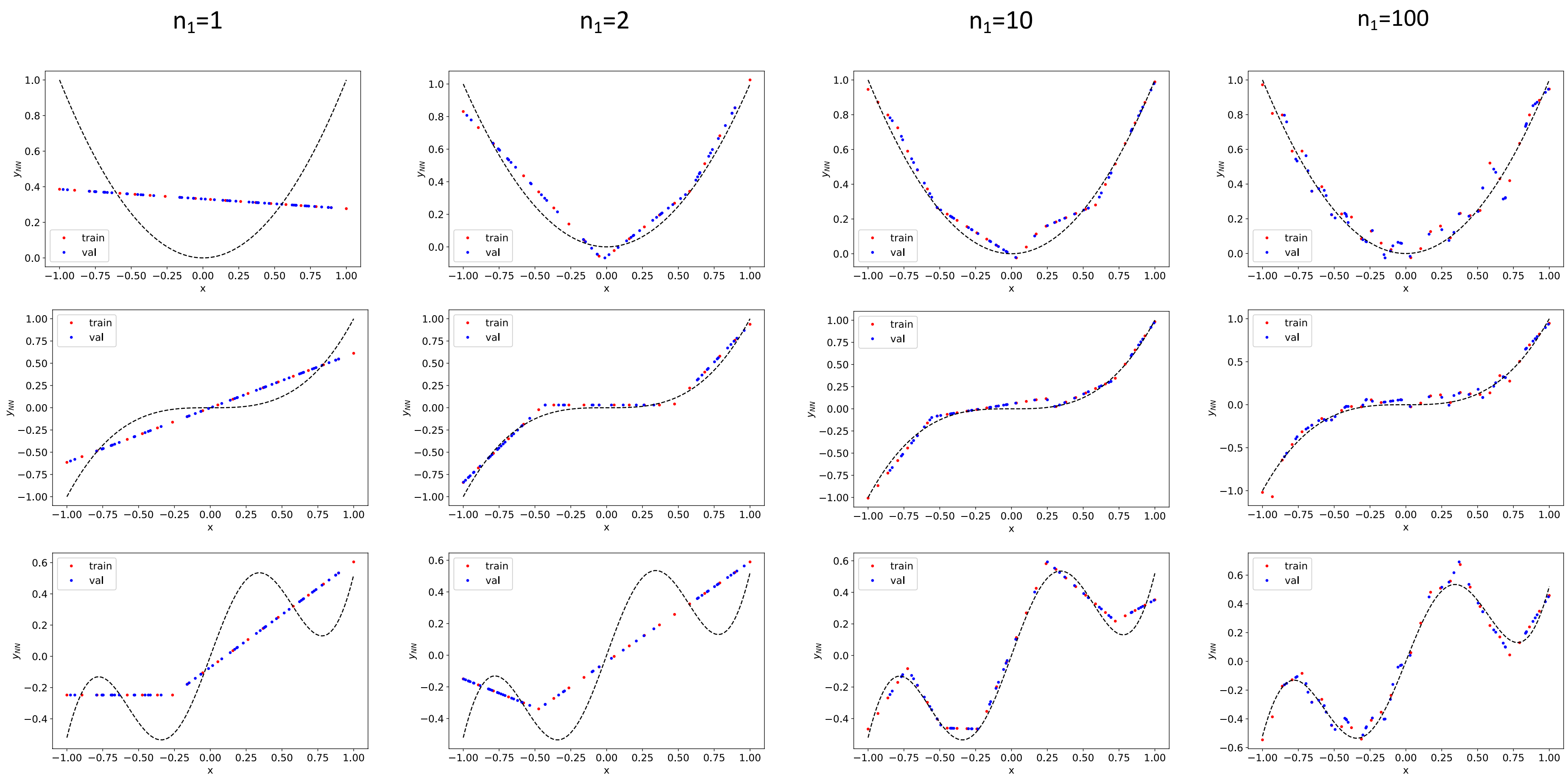
ReLU

$$\mathrm{ReLU}(x) = \max\{0, x\}$$

$$\mathrm{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{12}^1 + b_1^1\right)$$



ReLU(x)
ReLU(2.25x−1)+ReLU(−2.25x−1)
ReLU(2x−1)−ReLU(−2x−1)
1.0
0.8
0.6
0.4
0.2
0.0
0.5
−0.5
−1.0
x

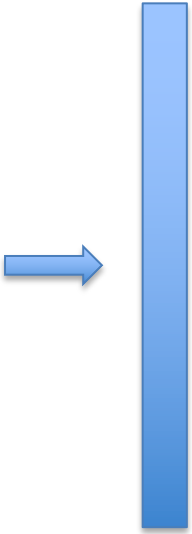


n1=1
n1=2
n1=10
n1=100
train
val
x
yNN

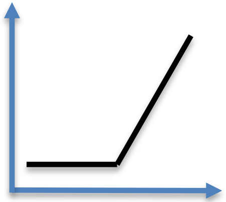
$$\begin{bmatrix} x_1^i \\ x_2^i \end{bmatrix} \begin{bmatrix} w_{11}^1 & w_{12}^1 \end{bmatrix} + b_1^1 = x_1^i w_{11}^1 + x_2^i w_{12}^1 + b_1^1$$

$$\mathbf{x}^i = \begin{bmatrix} x_1^i & x_2^i \end{bmatrix}$$

Input data



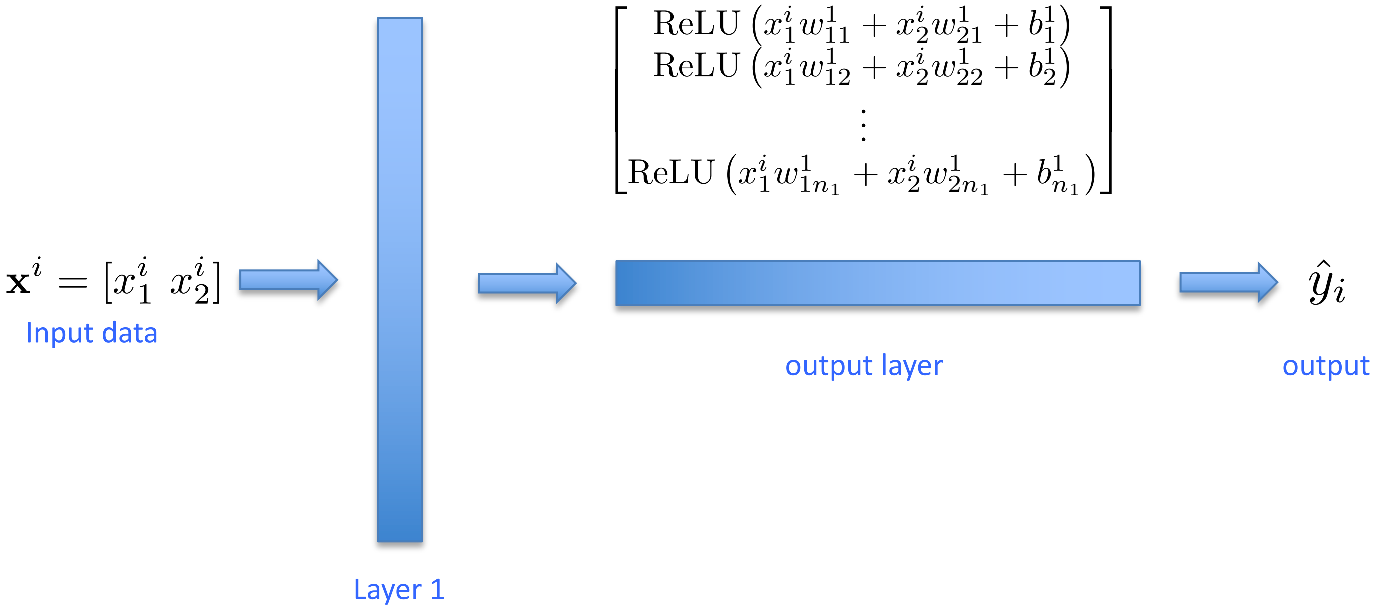
Layer 1



ReLU

$$\text{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{12}^1 + b_1^1\right)$$

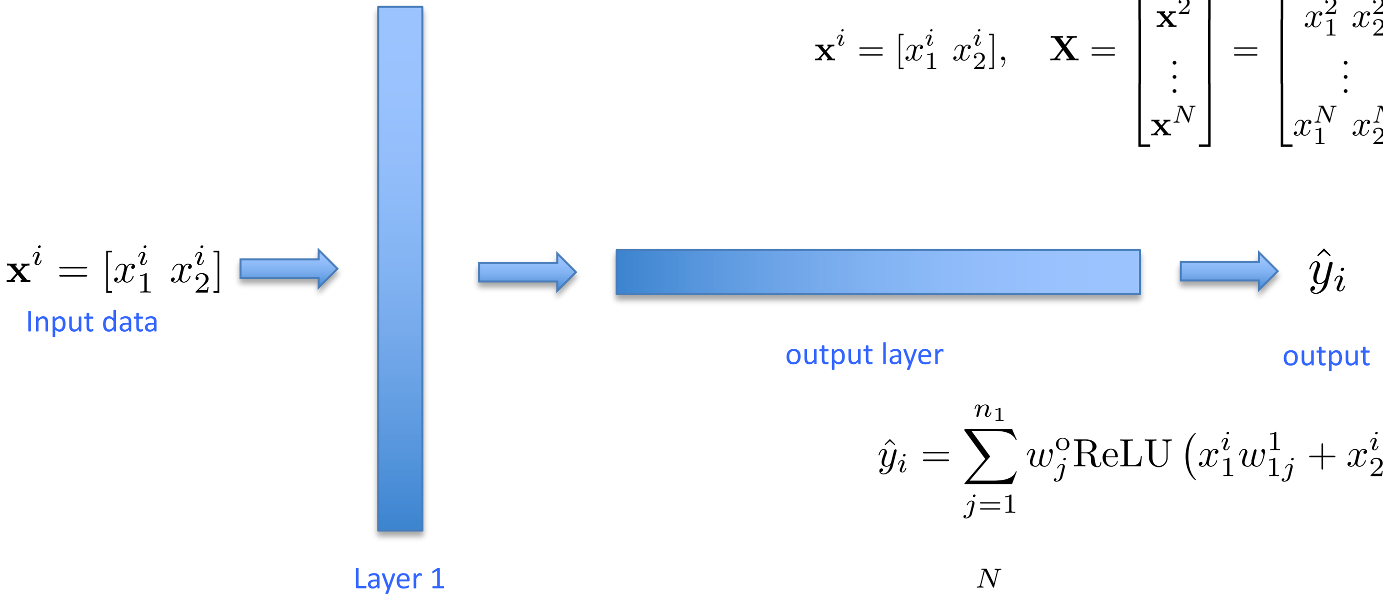
$$\begin{bmatrix} w_{11}^1 & w_{12}^1 \\ w_{21}^1 & w_{22}^1 \\ \vdots & \\ w_{1n_1}^1 & w_{2n_1}^1 \end{bmatrix} \begin{bmatrix} x_1^i \\ x_2^i \end{bmatrix} + \begin{bmatrix} b_1^1 \\ b_2^1 \\ \vdots \\ b_{n_1}^1 \end{bmatrix} = \begin{bmatrix} x_1^i w_{11}^1 + x_2^i w_{21}^1 + b_1^1 \\ x_1^i w_{12}^1 + x_2^i w_{22}^1 + b_2^1 \\ \vdots \\ x_1^i w_{1n_1}^1 + x_2^i w_{2n_1}^1 + b_{n_1}^1 \end{bmatrix} \Longrightarrow \begin{bmatrix} \text{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{21}^1 + b_1^1\right) \\ \text{ReLU}\left(x_1^i w_{12}^1 + x_2^i w_{22}^1 + b_2^1\right) \\ \vdots \\ \text{ReLU}\left(x_1^i w_{1n_1}^1 + x_2^i w_{2n_1}^1 + b_{n_1}^1\right) \end{bmatrix}$$



$$[w_1^{\text{o}} \; w_2^{\text{o}} \; \ldots \; w_{n_1}^{\text{o}}] \begin{bmatrix} \text{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{21}^1 + b_1^1\right) \\ \text{ReLU}\left(x_1^i w_{12}^1 + x_2^i w_{22}^1 + b_2^1\right) \\ \vdots \\ \text{ReLU}\left(x_1^i w_{1n_1}^1 + x_2^i w_{2n_1}^1 + b_{n_1}^1\right) \end{bmatrix} + b^{\text{o}}$$

$$\hat{y}_i = \sum_{j=1}^{n_1} w_j^{\text{o}} \text{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right) + b^{\text{o}}$$

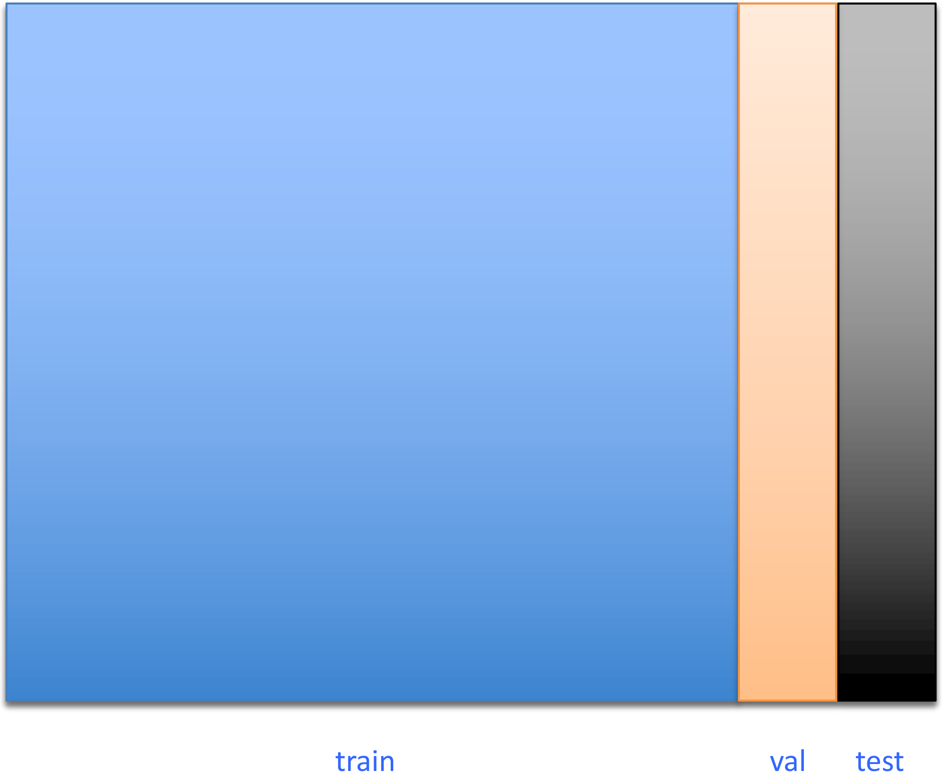
$$\mathbf{x}^i = [x_1^i \ x_2^i], \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}^1 \\ \mathbf{x}^2 \\ \vdots \\ \mathbf{x}^N \end{bmatrix} = \begin{bmatrix} x_1^1 & x_2^1 \\ x_1^2 & x_2^2 \\ & \vdots & \\ x_1^N & x_2^N \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

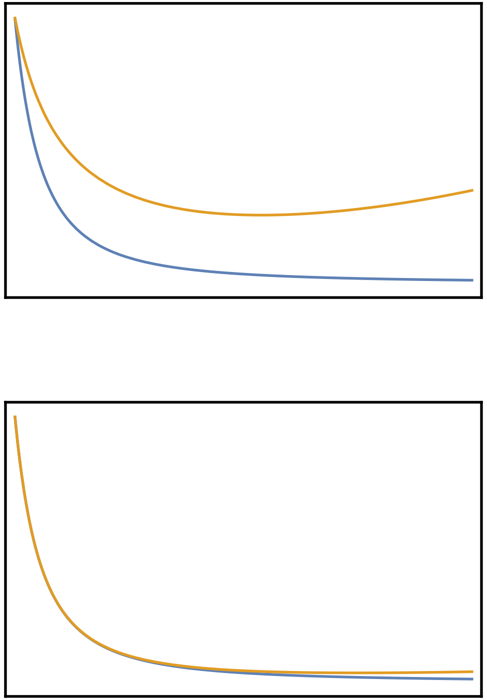


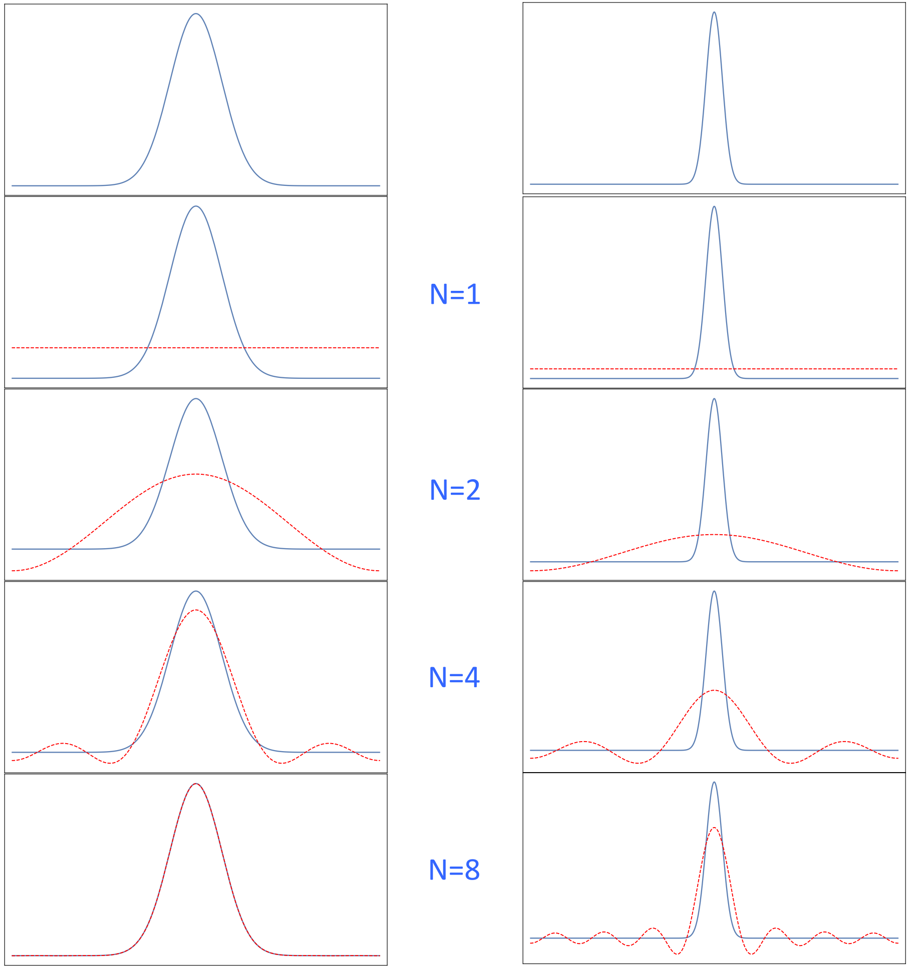
$$\hat{y}_i = \sum_{j=1}^{n_1} w_j^{\mathrm{o}} \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right) + b^{\mathrm{o}}$$

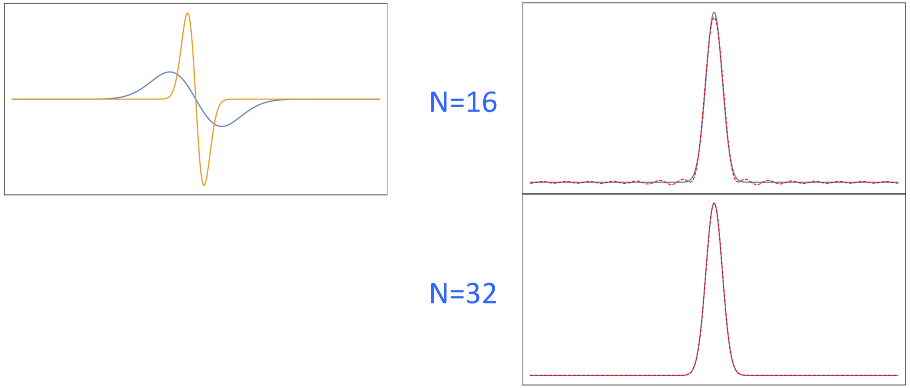
$$C = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2, \quad p \Longrightarrow p - \delta \nabla_p C$$

$$\nabla_{w_{kj}^1} C = 2 \sum_{i=1}^{N} (y_i - \hat{y}_i) \frac{\partial \hat{y}_i}{\partial w_{kj}^1} = 2 w_j^{\mathrm{o}} \sum_{i=1}^{N} (y_i - \hat{y}_i) \begin{cases} x_k^i, \text{ if } \ \mathrm{ReLU}\left(\sum_{k=1}^{2} x_k^i w_{kj}^1 + b_j^1\right) > 0 \\ 0, \text{ if } \ \mathrm{ReLU}\left(\sum_{k=1}^{2} x_k^i w_{kj}^1 + b_j^1\right) < 0 \end{cases}$$


train
val
test

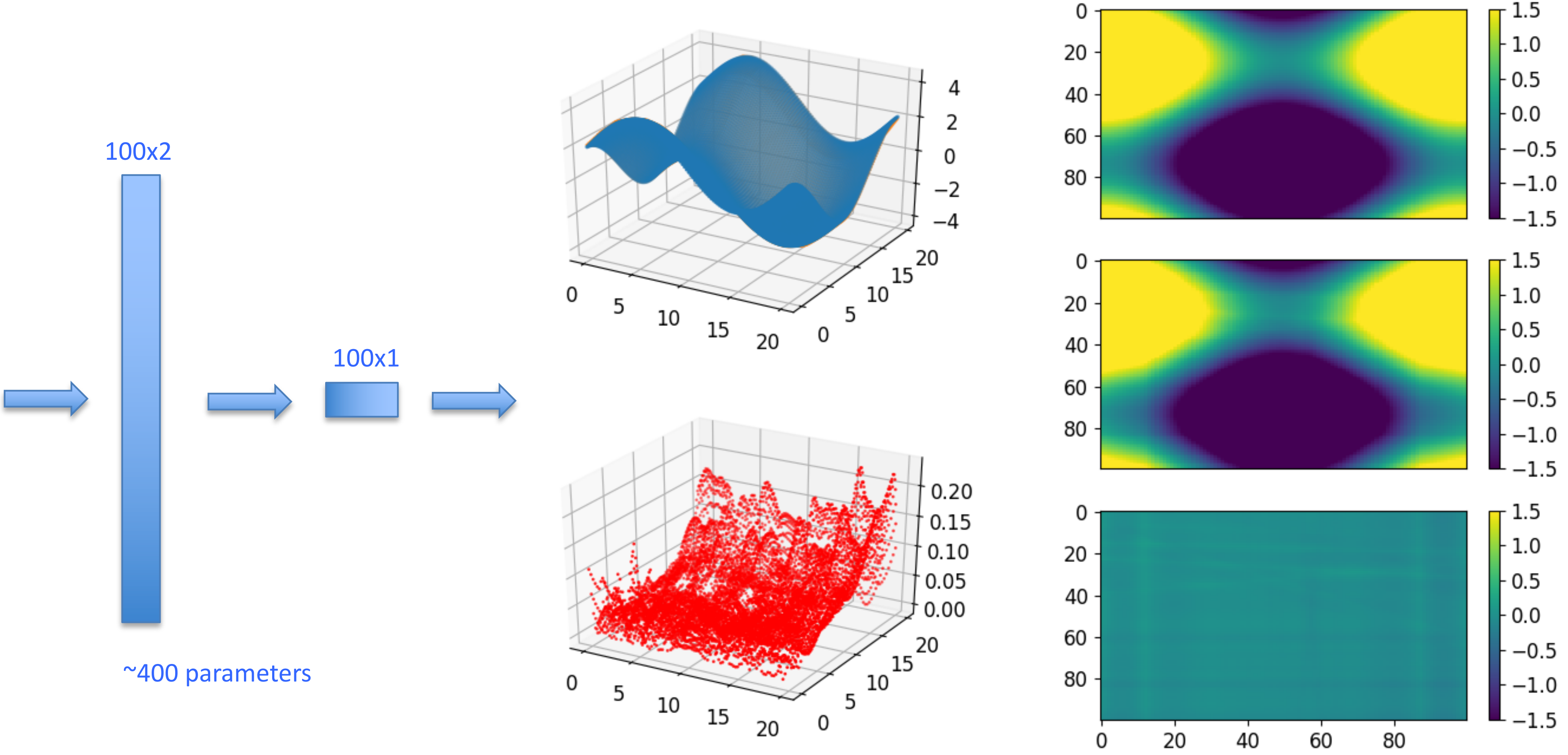




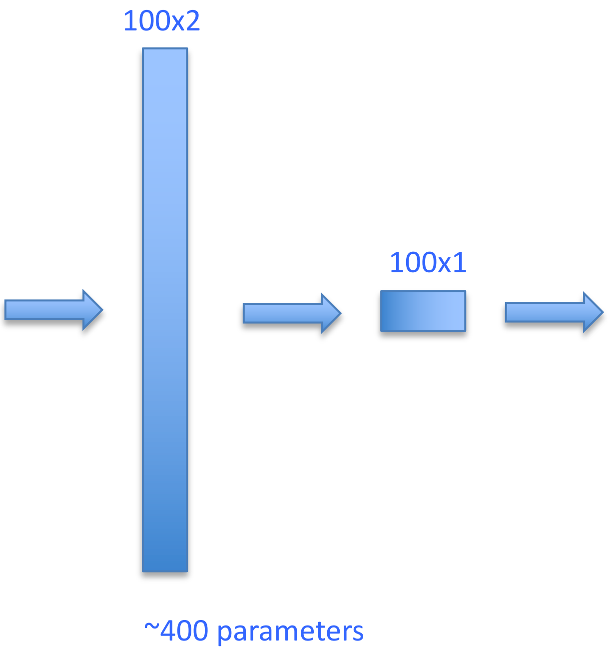


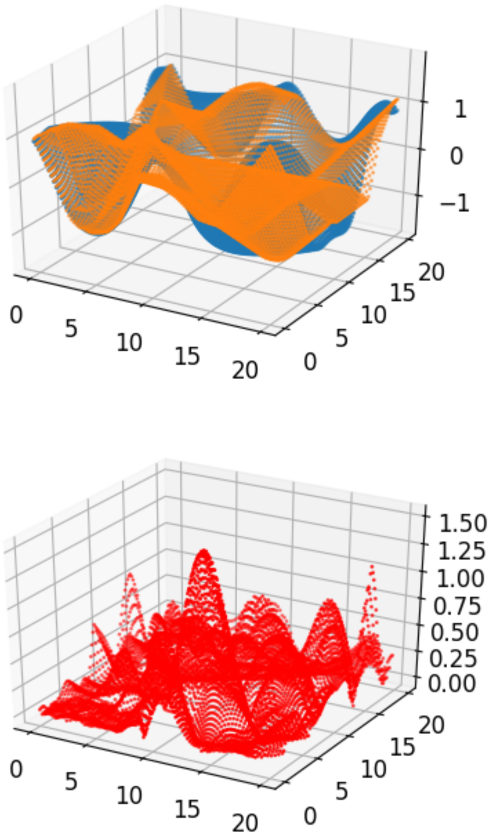
$$
\begin{aligned}
f(x) &\approx a_0 + \sum_{n=1}^{N} \left[a_n \cos\left(\frac{2\pi n x}{L}\right) + b_n \sin\left(\frac{2\pi n x}{L}\right) \right] \\
a_0 &= \frac{1}{L} \int_0^L f(x) dx \\
a_n &= \frac{2}{L} \int_0^L f(x) \cos\left(\frac{2\pi n x}{L}\right) dx \\
b_n &= \frac{2}{L} \int_0^L f(x) \sin\left(\frac{2\pi n x}{L}\right) dx
\end{aligned}
$$

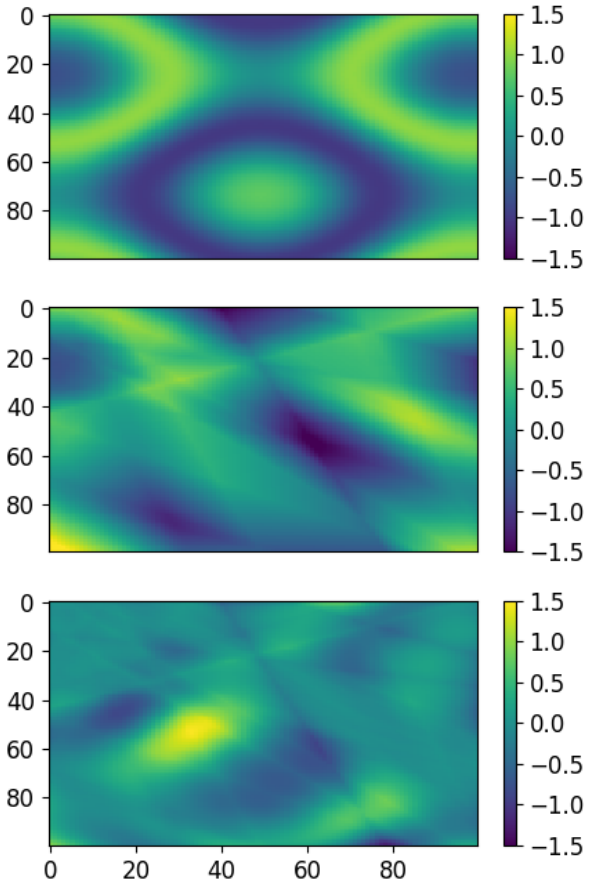
$$f(x, y) = 2\left[\sin(0.6x) + \cos(0.6y)\right]$$



$$f(x,y) = \sin\left(2\left[\sin(0.6x) + \cos(0.6y)\right]\right)$$

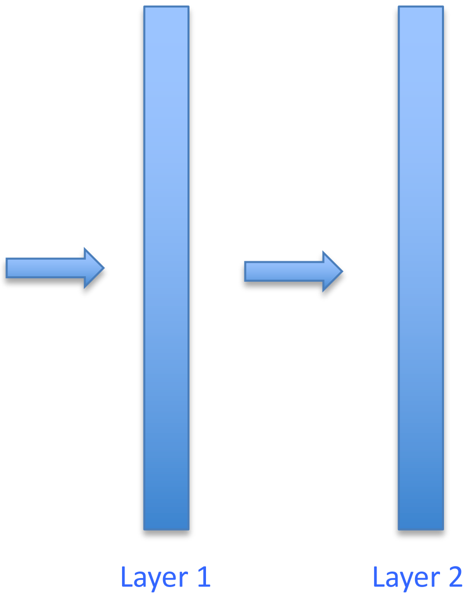






$$\mathbf{x}^i = [x_1^i \;\; x_2^i]$$

Input data



$$\begin{bmatrix} w_{11}^2 & w_{12}^2 & \dots & w_{1n_1}^2 \\ w_{21}^2 & w_{22}^2 & \dots & w_{2n_1}^2 \\ & & \vdots & \\ w_{n_21}^2 & w_{n_22}^2 & \dots & w_{n_2n_1}^2 \end{bmatrix} \begin{bmatrix} \mathrm{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{21}^1 + b_1^1\right) \\ \mathrm{ReLU}\left(x_1^i w_{12}^1 + x_2^i w_{22}^1 + b_2^1\right) \\ \vdots \\ \mathrm{ReLU}\left(x_1^i w_{1n_1}^1 + x_2^i w_{2n_1}^1 + b_{n_1}^1\right) \end{bmatrix} + \begin{bmatrix} b_1^2 \\ b_2^2 \\ \vdots \\ b_{n_2}^2 \end{bmatrix}$$

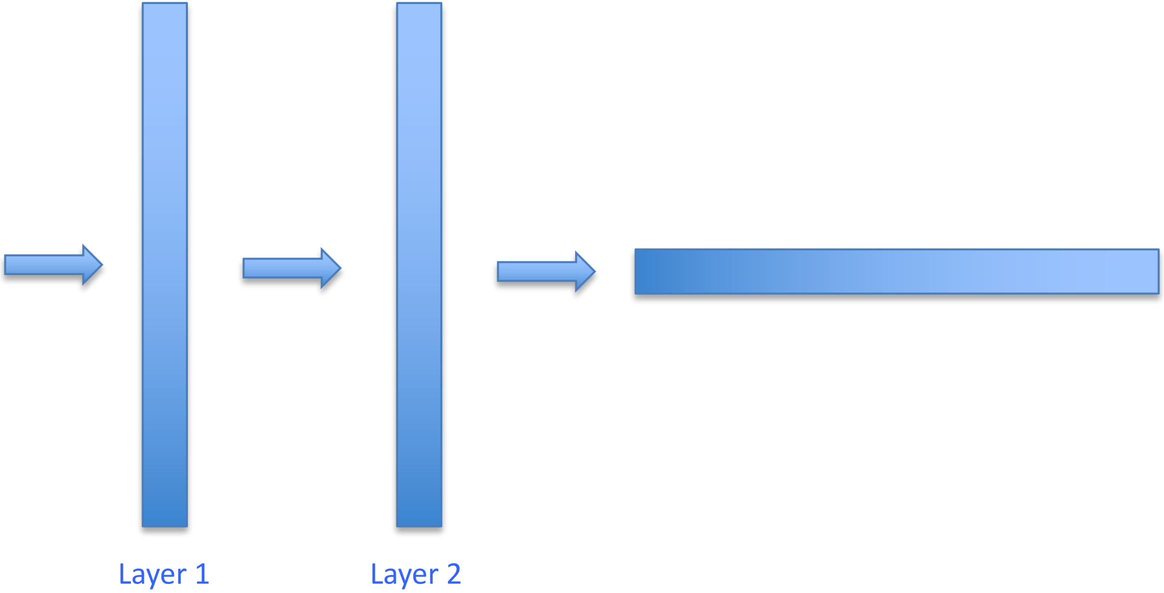
$$\begin{bmatrix} b_1^2 + \sum_{j=1}^{n_1} w_{1j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right) \\ b_2^2 + \sum_{j=1}^{n_1} w_{2j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i x_{2j}^1 + b_j^1\right) \\ \vdots \\ b_{n_2}^2 + \sum_{j=1}^{n_1} w_{n_2j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right) \end{bmatrix}$$

$$\begin{bmatrix} \mathrm{ReLU}\left(b_1^2 + \sum_{j=1}^{n_1} w_{1j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right)\right) \\ \mathrm{ReLU}\left(b_2^2 + \sum_{j=1}^{n_1} w_{2j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i x_{2j}^1 + b_j^1\right)\right) \\ \vdots \\ \mathrm{ReLU}\left(b_{n_2}^2 + \sum_{j=1}^{n_1} w_{n_2j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right)\right) \end{bmatrix}$$

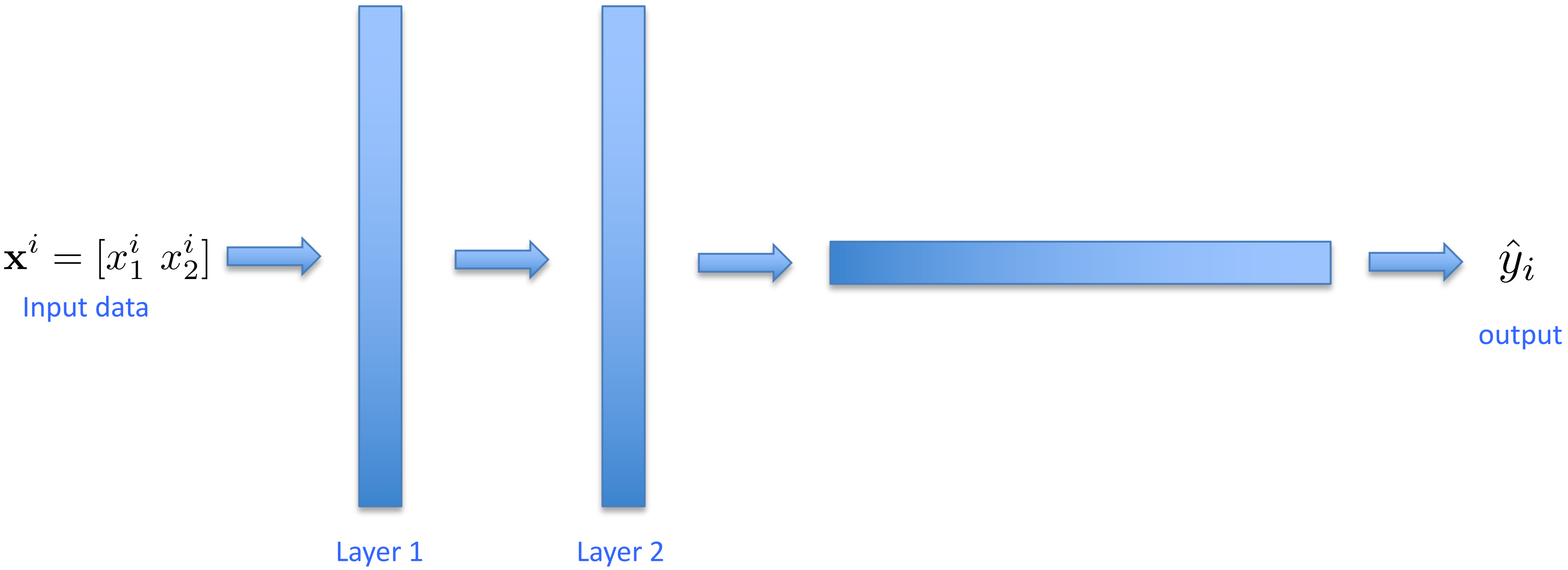
$$\begin{bmatrix} \mathrm{ReLU}\left(x_1^i w_{11}^1 + x_2^i w_{21}^1 + b_1^1\right) \\ \mathrm{ReLU}\left(x_1^i w_{12}^1 + x_2^i w_{22}^1 + b_2^1\right) \\ \vdots \\ \mathrm{ReLU}\left(x_1^i w_{1n_1}^1 + x_2^i w_{2n_1}^1 + b_{n_1}^1\right) \end{bmatrix}$$

$\mathbf{x}^i = [x_1^i \ x_2^i]$

Input data

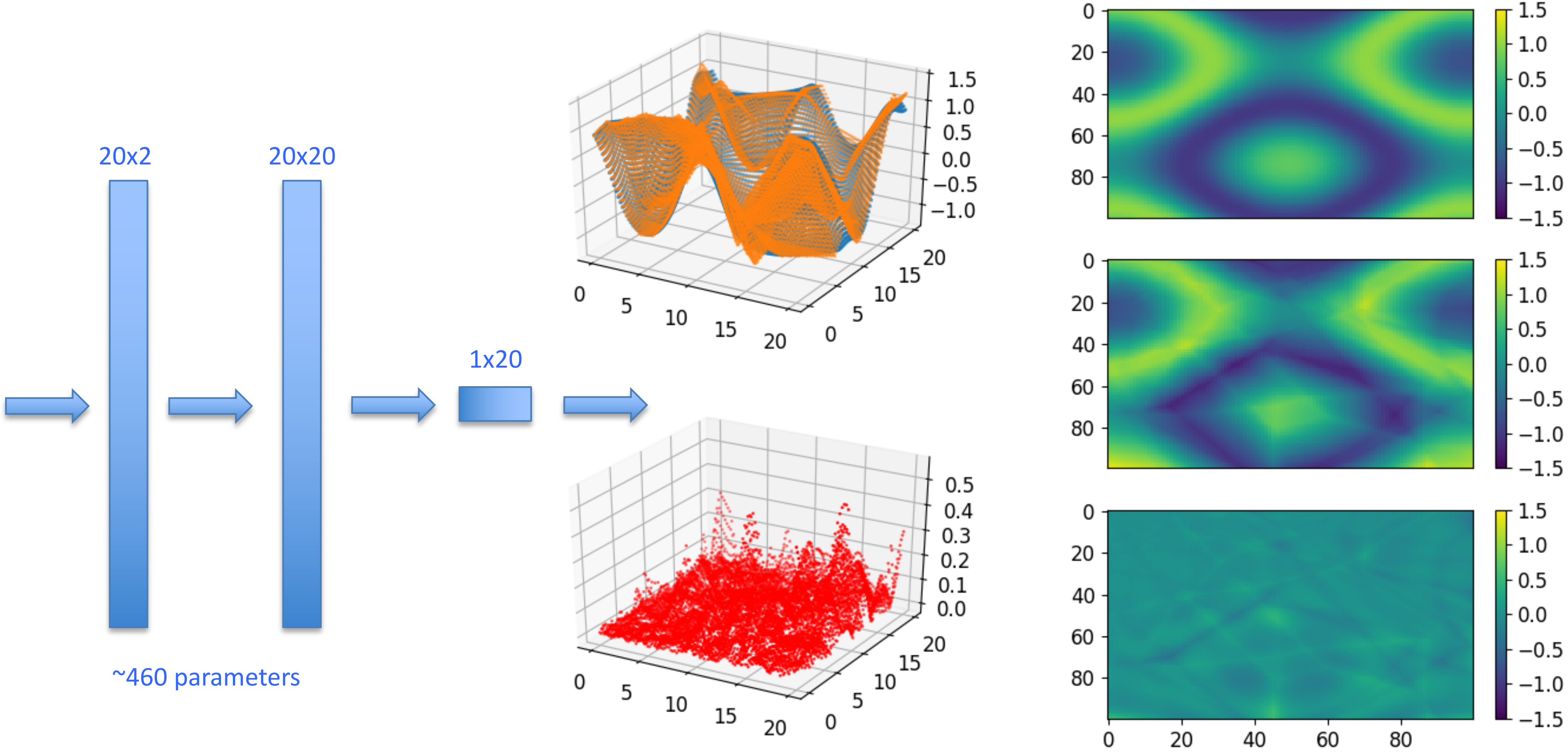


$$
\begin{bmatrix} w_1^{\mathrm{o}} & w_2^{\mathrm{o}} & \ldots & w_{n_2}^{\mathrm{o}} \end{bmatrix}
\begin{bmatrix}
\mathrm{ReLU}\left(b_1^2 + \sum_{j=1}^{n_1} w_{1j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right)\right) \\
\mathrm{ReLU}\left(b_2^2 + \sum_{j=1}^{n_1} w_{2j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i x_{2j}^1 + b_j^1\right)\right) \\
\vdots \\
\mathrm{ReLU}\left(b_{n_2}^2 + \sum_{j=1}^{n_1} w_{n_2 j}^2 \mathrm{ReLU}\left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1\right)\right)
\end{bmatrix}
+ b^{\mathrm{o}}
$$

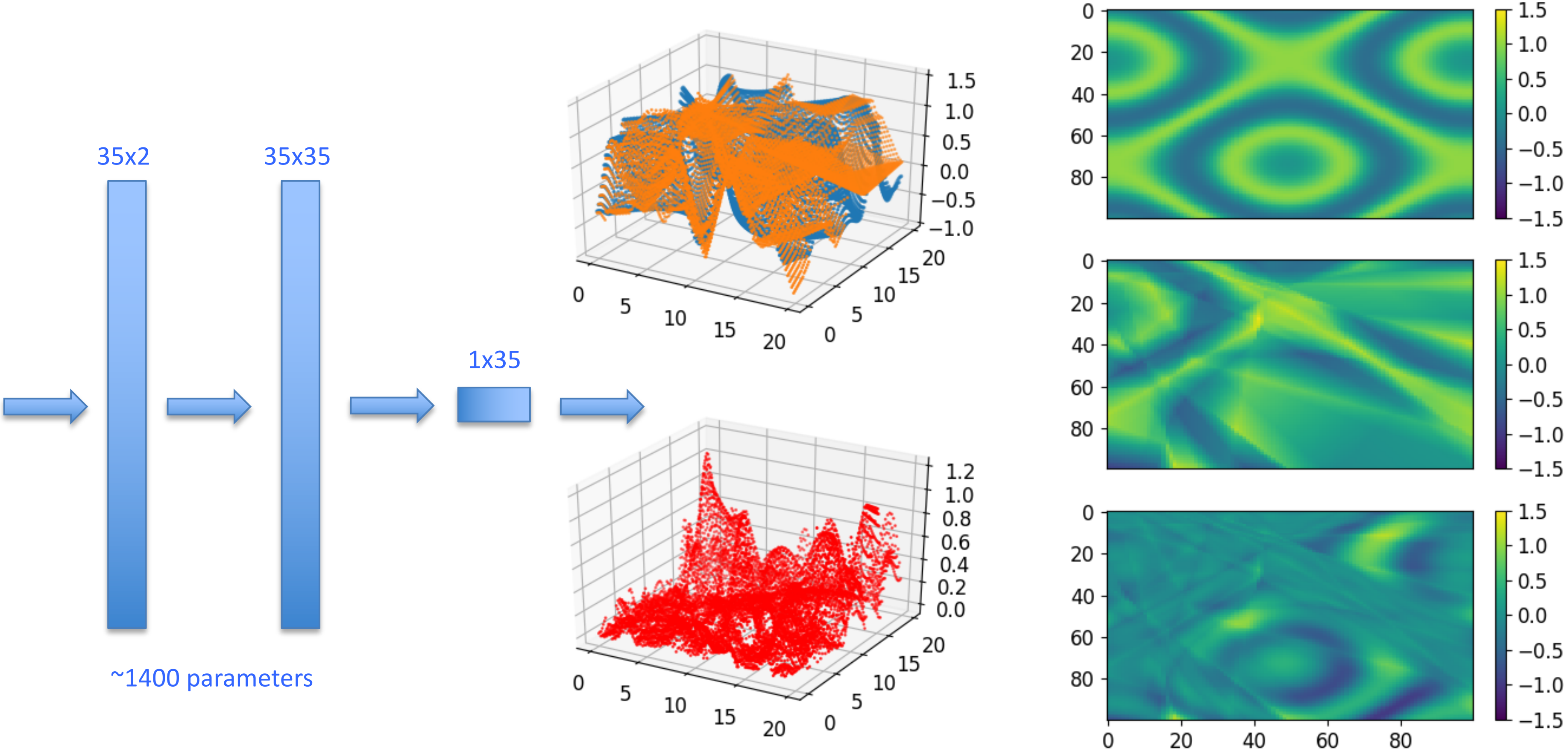


$$\hat{y}_i = \sum_{k=1}^{n_2} \left[w_k^{\text{o}} \text{ReLU} \left(b_k^2 + \sum_{j=1}^{n_1} w_{kj}^2 \text{ReLU} \left(x_1^i w_{1j}^1 + x_2^i w_{2j}^1 + b_j^1 \right) \right) \right] + b^{\text{o}}$$

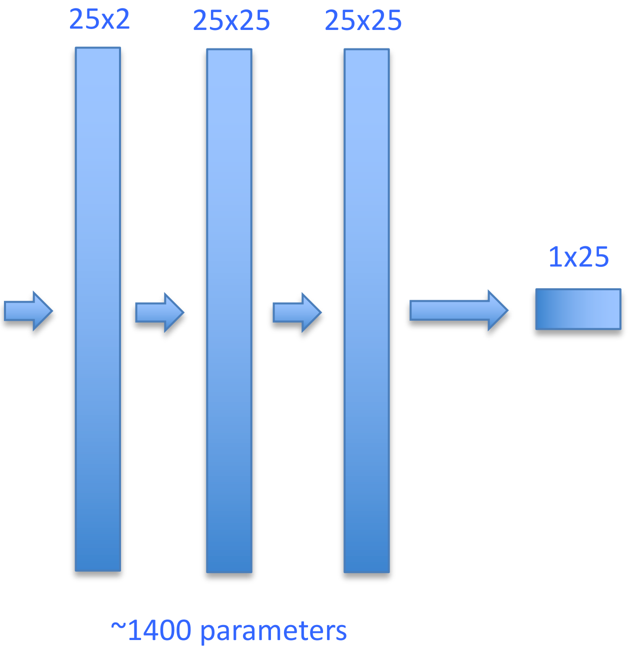
$$f(x,y) = \sin\left(2\left[\sin(0.6x) + \cos(0.6y)\right]\right)$$

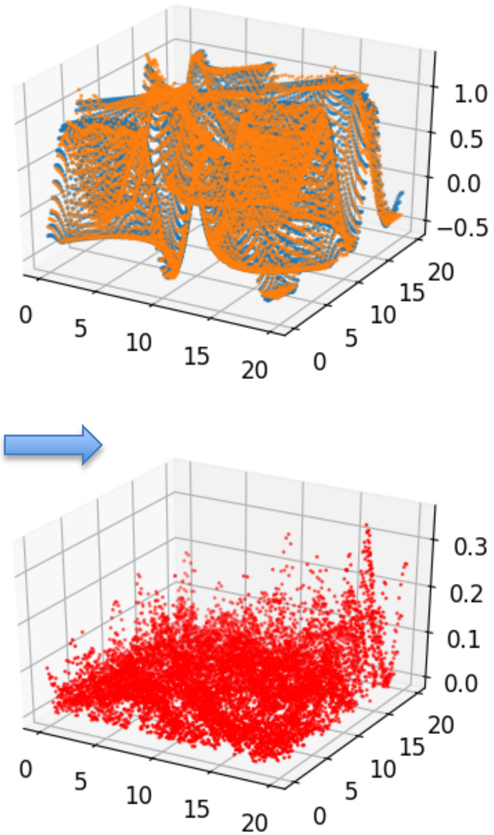


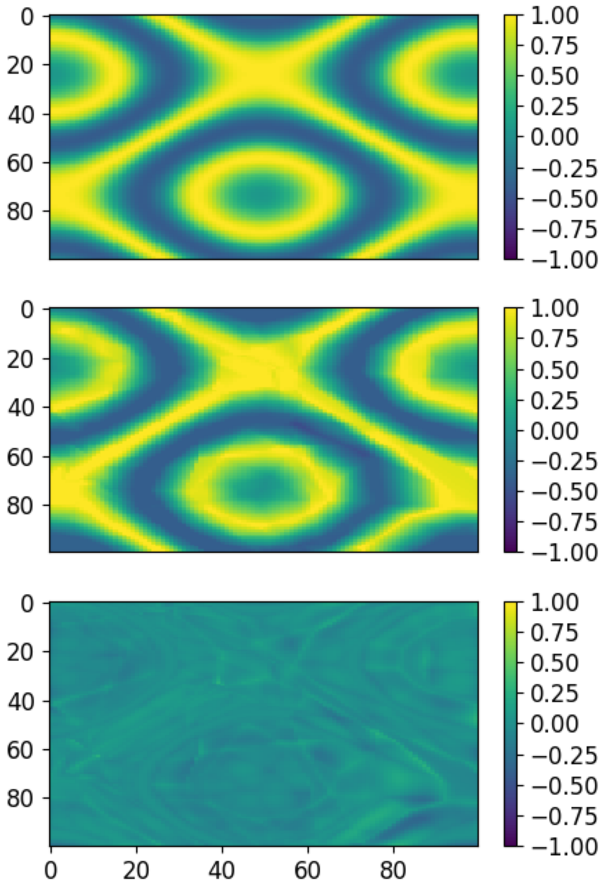
$$f(x, y) = \cos\left[\sin\left(2\left[\sin(0.6x) + \cos(0.6y)\right]\right)\right]$$



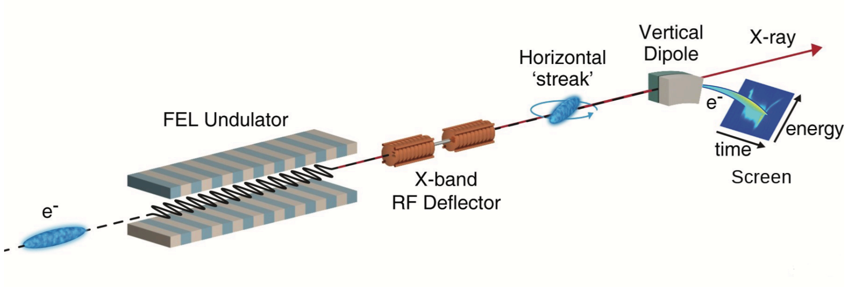
$$f(x,y) = \cos\left[\sin\left(2\left[\sin(0.6x) + \cos(0.6y)\right]\right)\right]$$

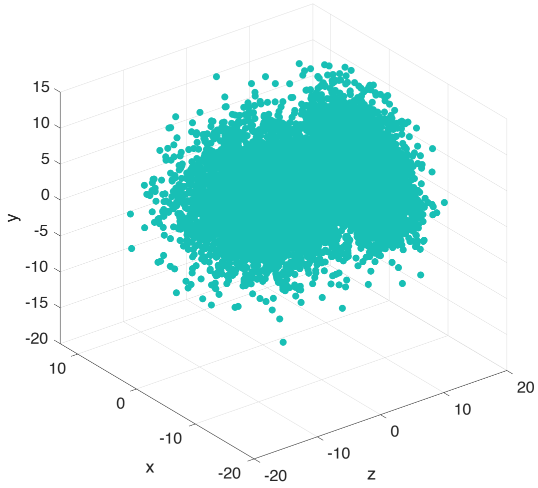




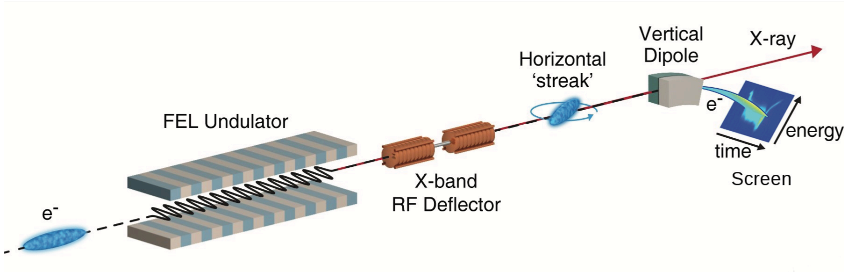


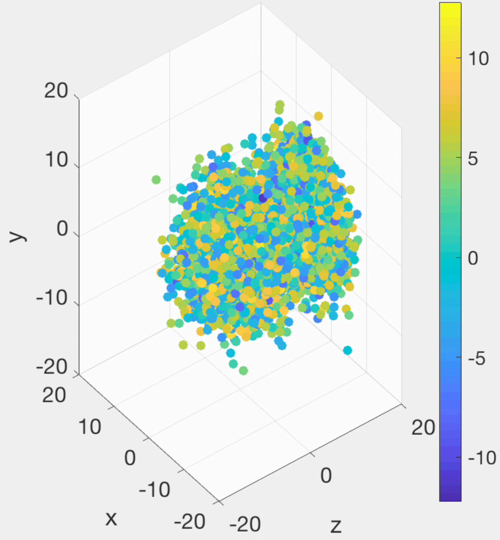
Transverse Deflecting Cavity



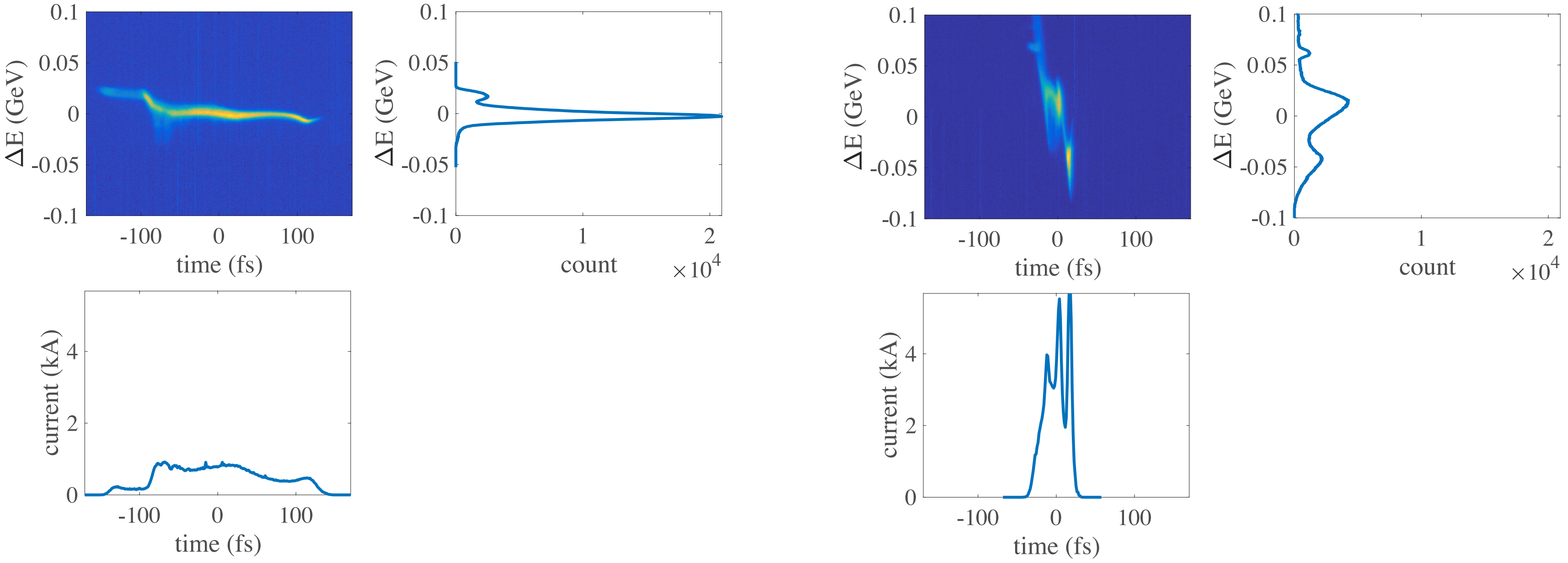


Transverse Deflecting Cavity

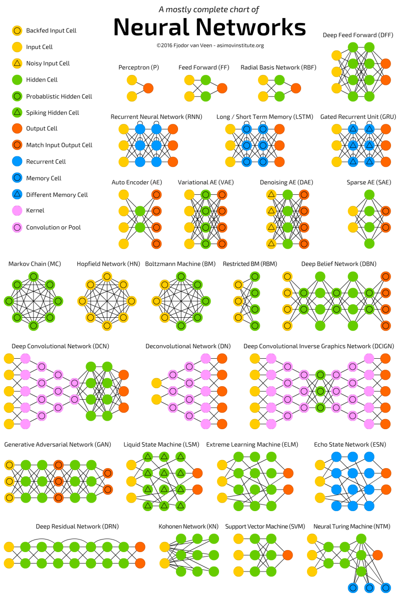


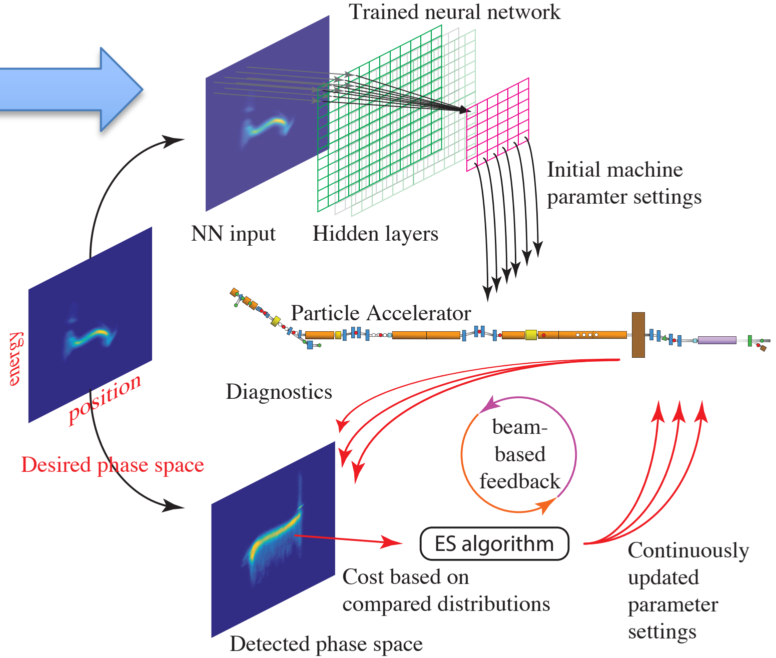


XTCAV Data

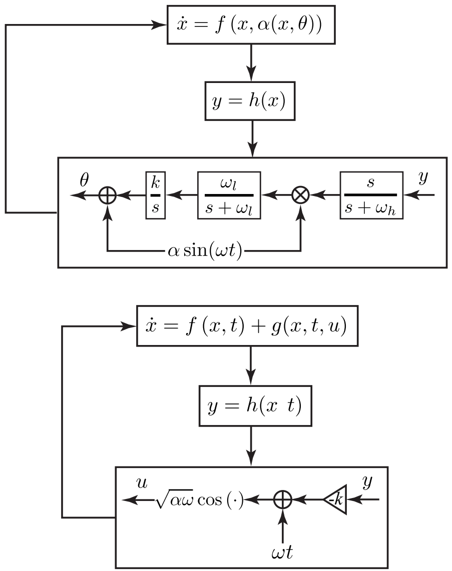

ΔE (GeV)
0.1
0.05
0
-0.05
-0.1
-100
0
100
time (fs)
count
1
2
×10^4
current (kA)
4
2
0

Adaptive Machine Learning for Time Varying Systems



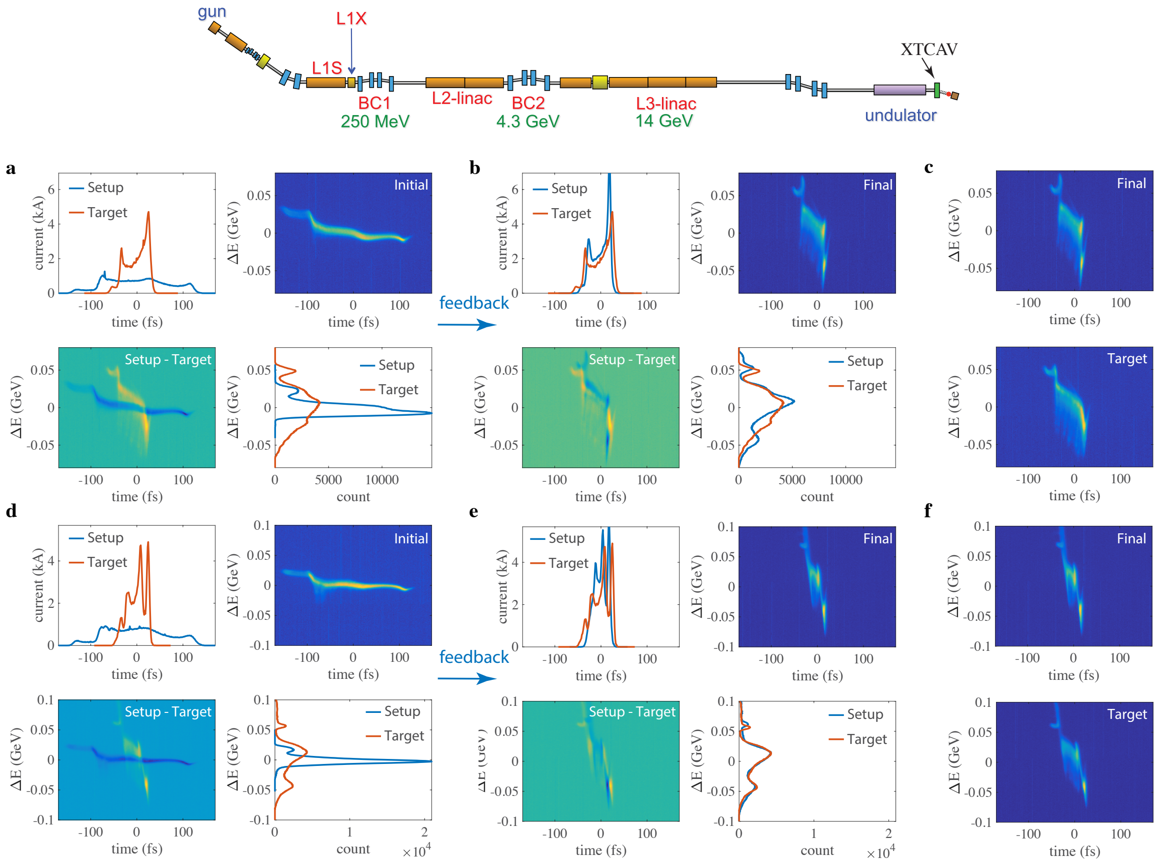


Adapative Feedback Control



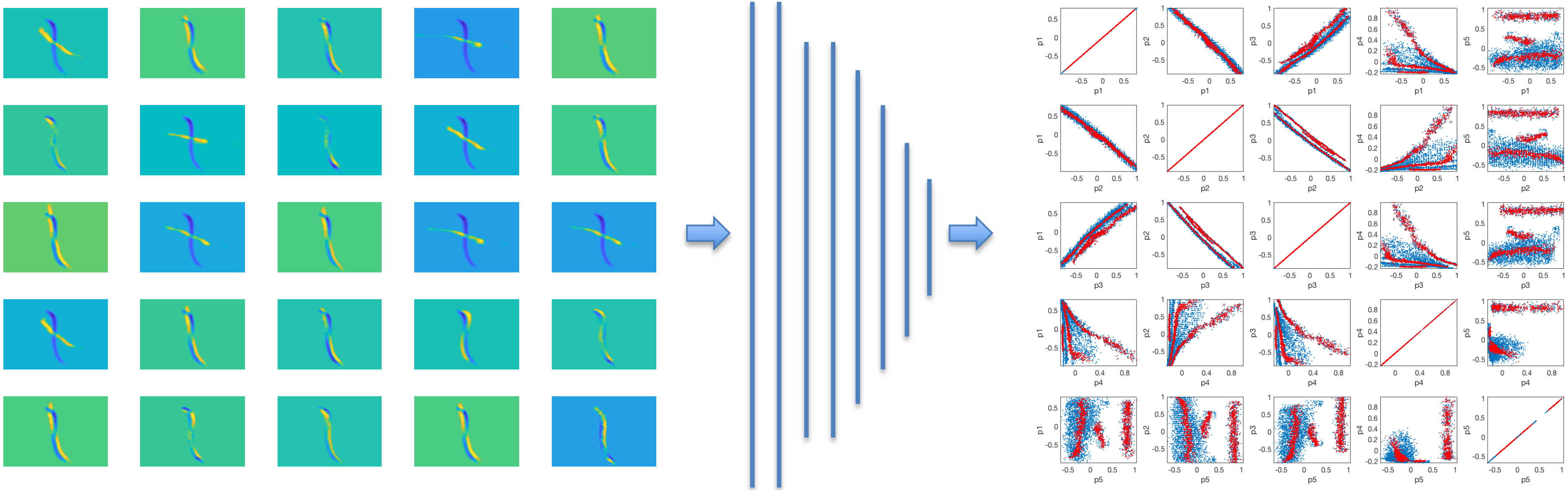
A. Scheinker, et al. “Demonstration of model-independent control of the longitudinal phase space of electron beams in the Linac-coherent light source with Femtosecond resolution.” Physical Review Letters, 121.4, 044801, 2018.
https://doi.org/10.1103/PhysRevLett.121.044801

Adaptive ML for automatic longitudinal phase space control at the **LCLS**

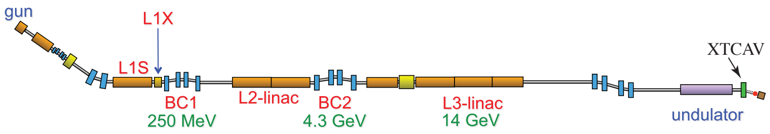


A. Scheinker, et al. "Demonstration of model-independent control of the longitudinal phase space of electron beams in the Linac-coherent light source with Femtosecond resolution." Physical Review Letters, 121.4, 044801, 2018. https://doi.org/10.1103/PhysRevLett.121.044801

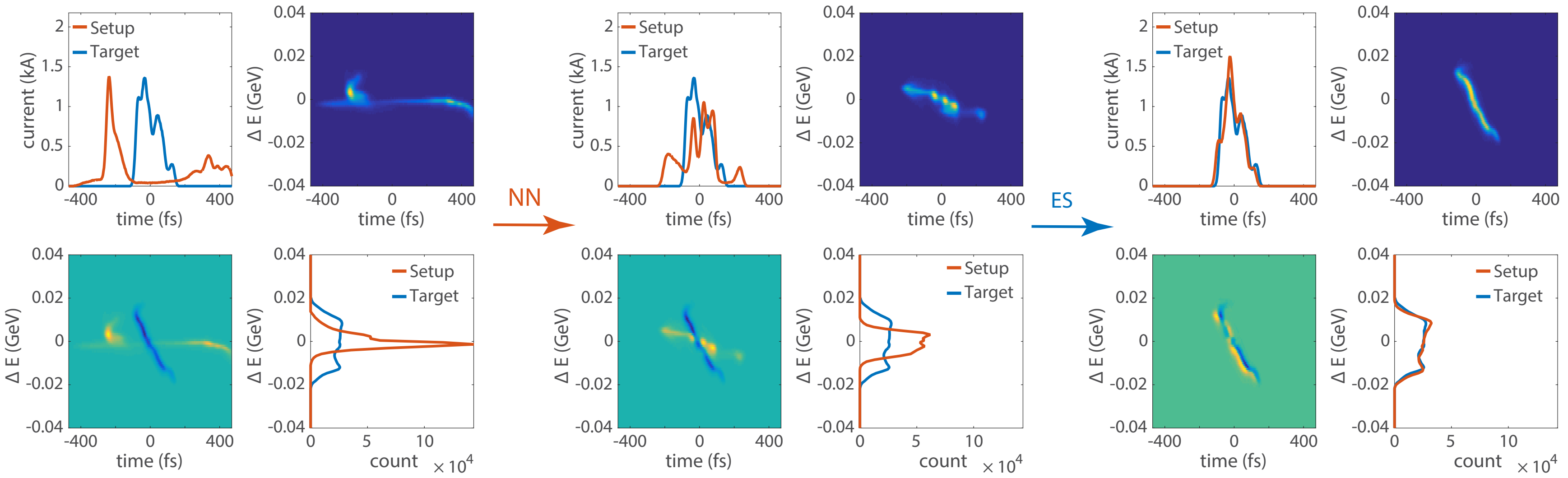
XTCAV NN



Adaptive ML for automatic longitudinal phase space control at the **LCLS**



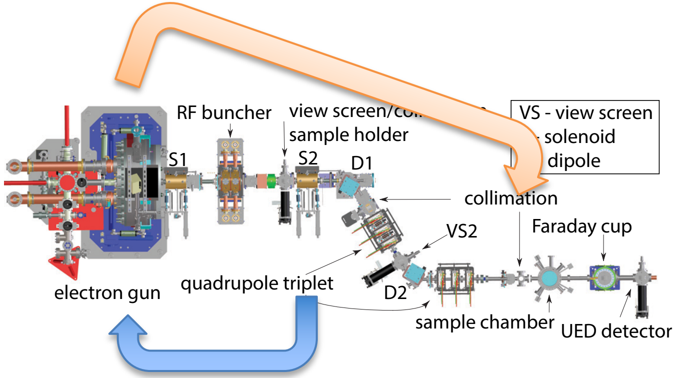
$$C = \int_{-\Delta L}^{\Delta L}\int_{-\Delta E}^{\Delta E} \left|\hat{\rho}(z,E) - \rho(z,E)\right| dE dz$$

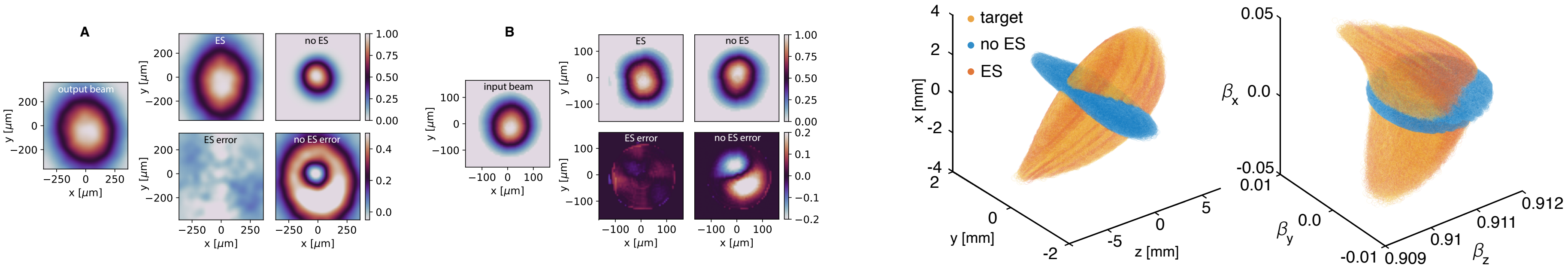


A. Scheinker, et al. "Demonstration of model-independent control of the longitudinal phase space of electron beams in the Linac-coherent light source with Femtosecond resolution." Physical Review Letters, 121.4, 044801, 2018. https://doi.org/10.1103/PhysRevLett.121.044801

Adaptive Machine Learning for Predicting Beam Distributions

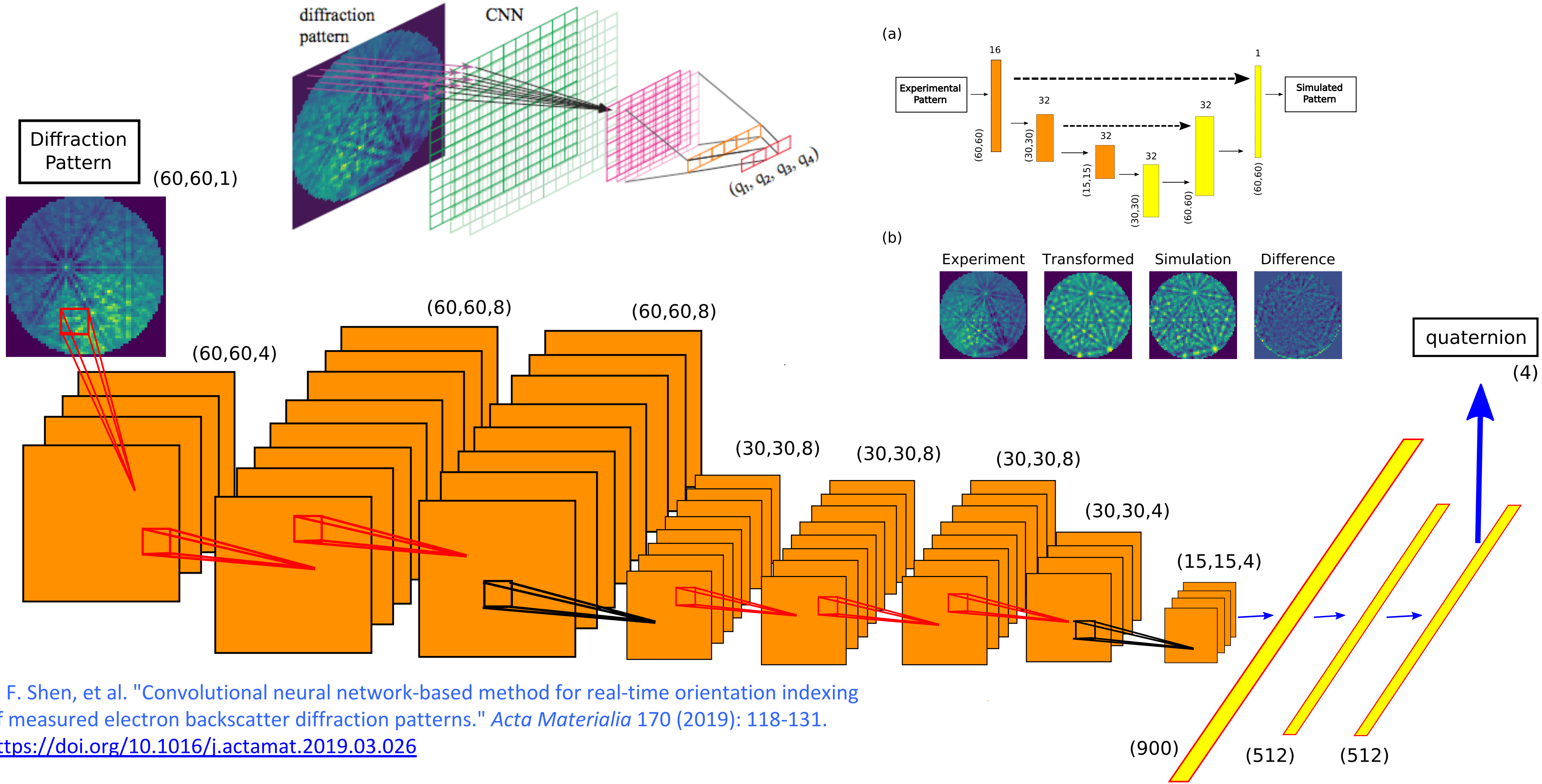
A. Scheinker, F. Cropp, S. Paiagua, D. Filippetto, "Demonstration of adaptive machine learning-based distribution tracking on a compact accelerator: Towards enabling model-based 6D non-invasive beam diagnostics", 2021, *under review.*





Adjusting Neural Networks for Time-Varying Systems

Re-Training and Domain Transfer for Convolutional Neural Networks



Y. F. Shen, et al. "Convolutional neural network-based method for real-time orientation indexing of measured electron backscatter diffraction patterns." *Acta Materialia* 170 (2019): 118-131.
https://doi.org/10.1016/j.actamat.2019.03.026