Current status and future improvements of machine learning implementation for the beam control at SACLA

Kenji Yasutome, Hirokazu Maesaka, Eito Iwai
RIKEN SPring-8 Center

加速器・ビーム物理の機械学習ワークショップ2023

RIKEN, Wako Campus
November 27-29th, 2023

Contents

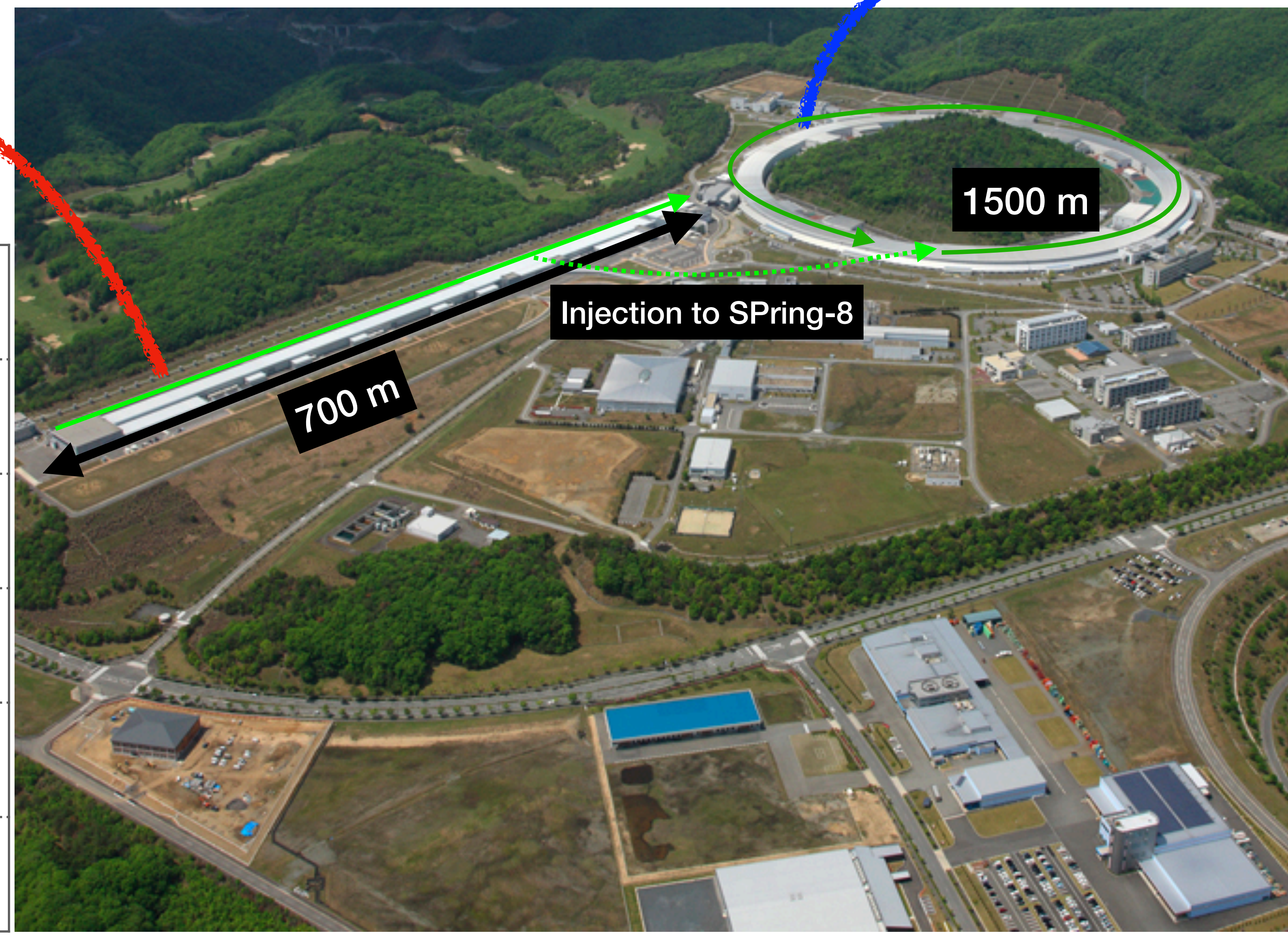
- Introduction to an XFEL facility, SACLA
- Current status
  - Introduction to the GPR Optimizer
  - Recent achievements
- Future improvements
  - Idea to use a deep learning algorithm for the beam control
- Summary

Introduction to SACLA

SPring-8

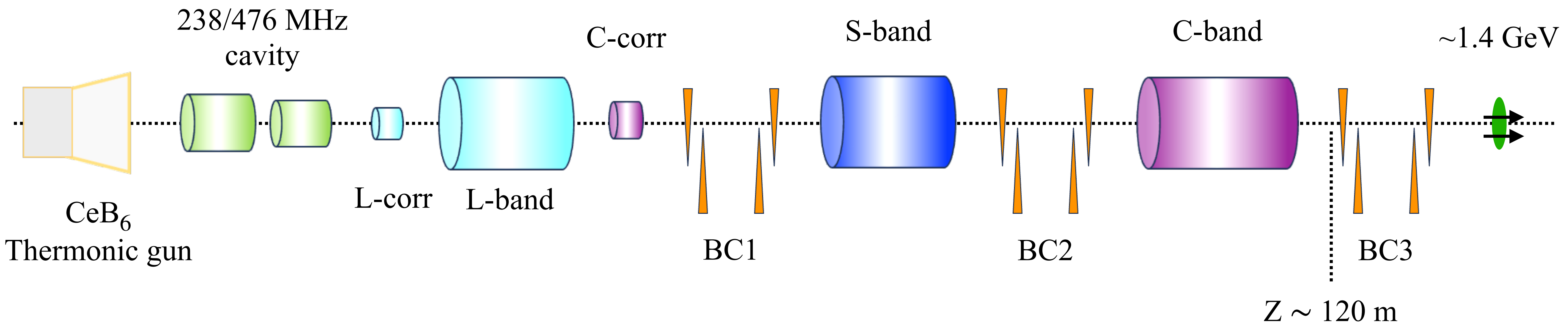
SACLA
(The SPring-8 Angstrom Compact free electron LAser)

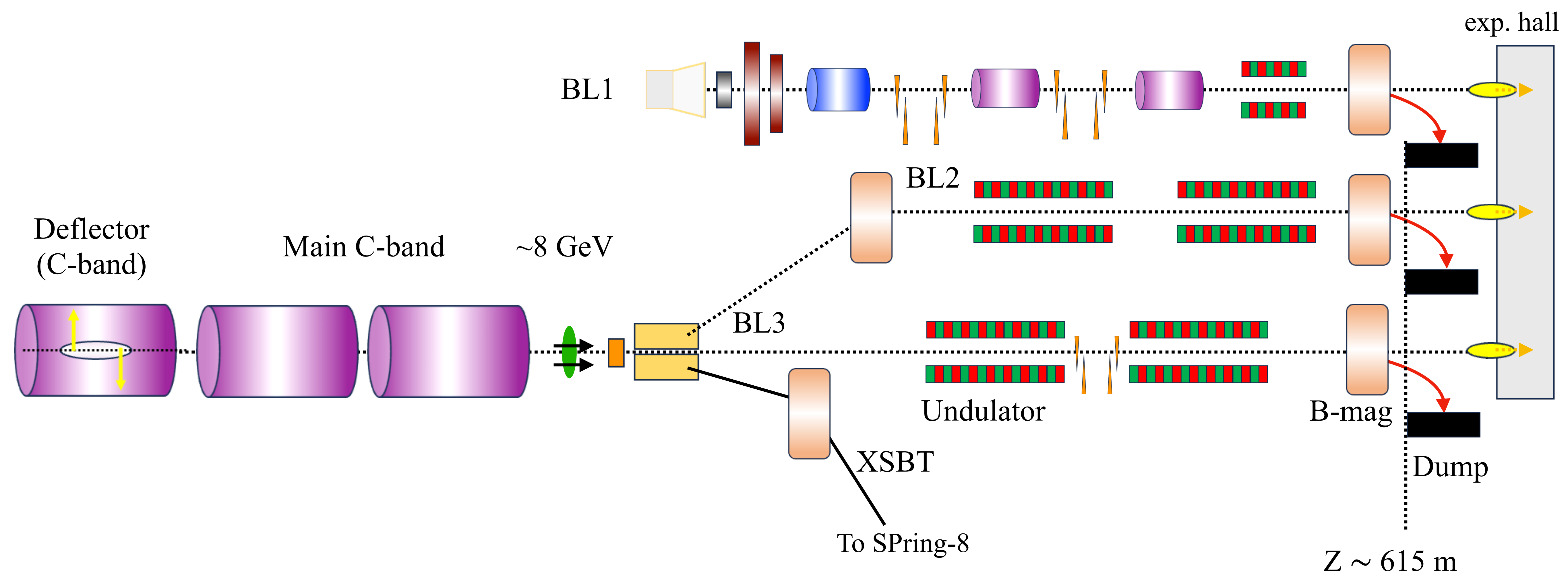
Param	BL2, BL3 Xray-FEL	BL1 EUV-FEL
Beam energy	6-8 GeV	~800 MeV
Photon energy	4-22 keV	40-150 eV
Pulse Energy	~700 uJ	~50 uJ
Pulse width	< 10 fs	~ 30 fs
Rep. rate	60 Hz	60 Hz



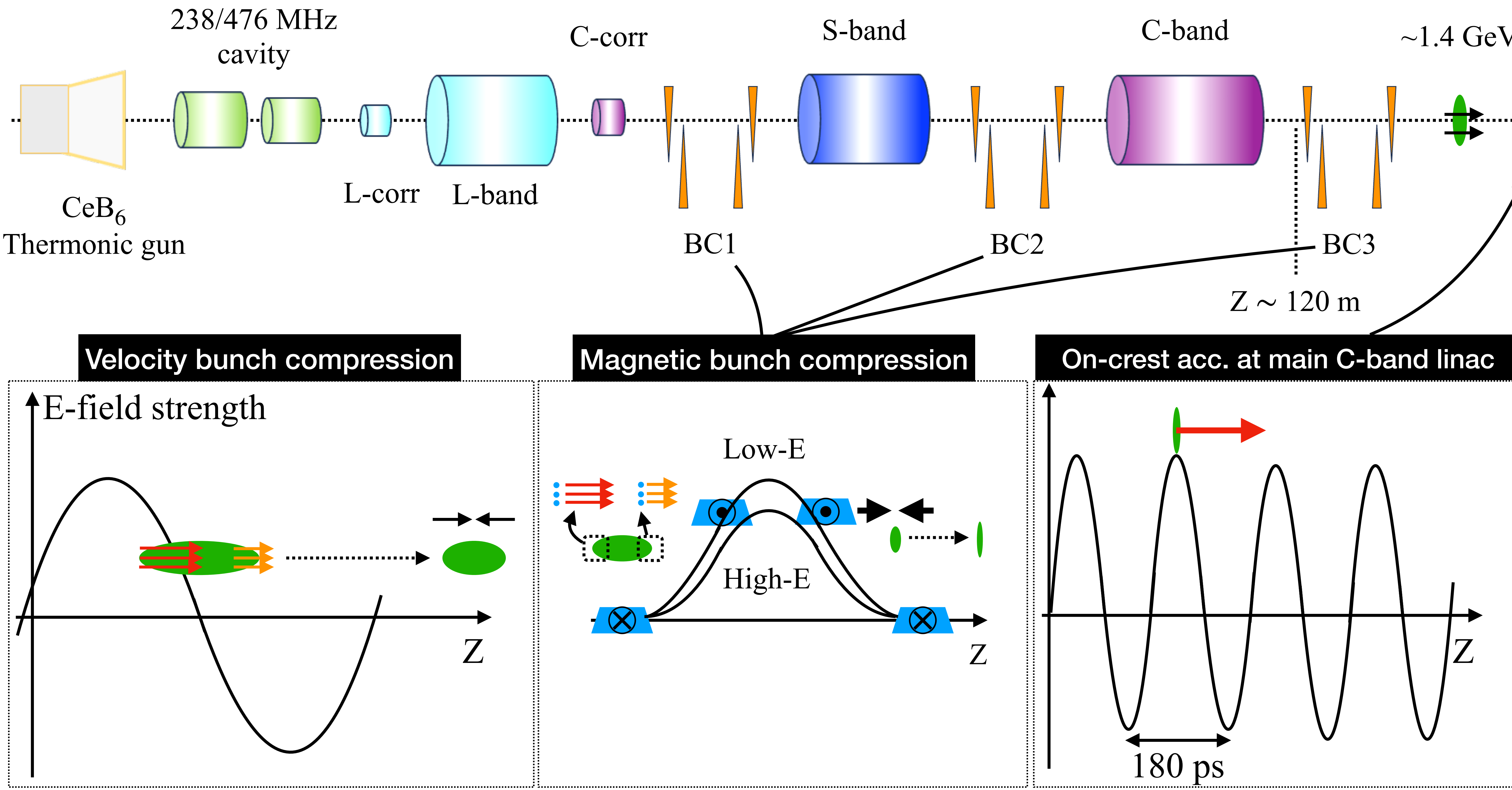
- SACLA: LINAC-based XFEL facility, Injector to SPring-8
  - Simultaneous operations of three beamlines (BL1, BL2, and BL3)
  - Operate 6000 hours/year, with high availability (~100 users/year)

Introduction to SACLA





Introduction to SACLA



Initial electron bunches (1 A, 1 ns) are highly compressed to short bunches (10 kA, 10 fs)

Difficulties in the beam tuning at SACLA

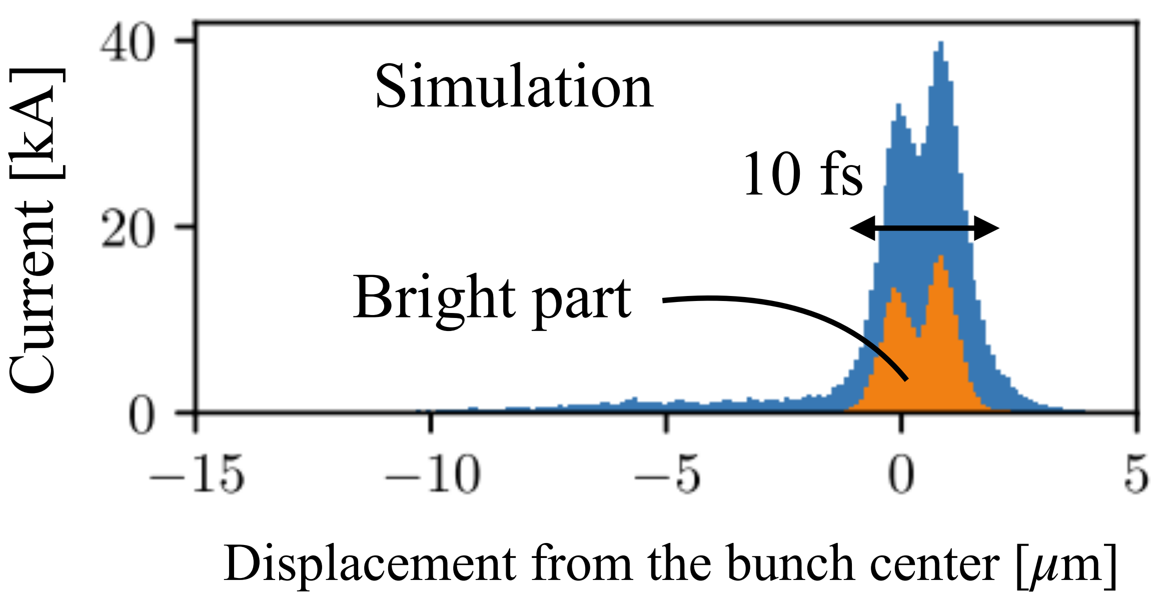
Electron beam

- Highly compressed beam due to non-linear and complex bunch compression process
- Non-gaussian beam (Typically two peaks)

Accelerator

- Sensitive to a slight environmental change (temperature, humidity, etc…)
- Simultaneous operations of BL1, BL2, BL3, and SR injection by pulse-by-pulse distribution
- Tuning qualities depending on operators' experiences

Time structure of e-beam at SACLA (bunch compressed to 40 kA)

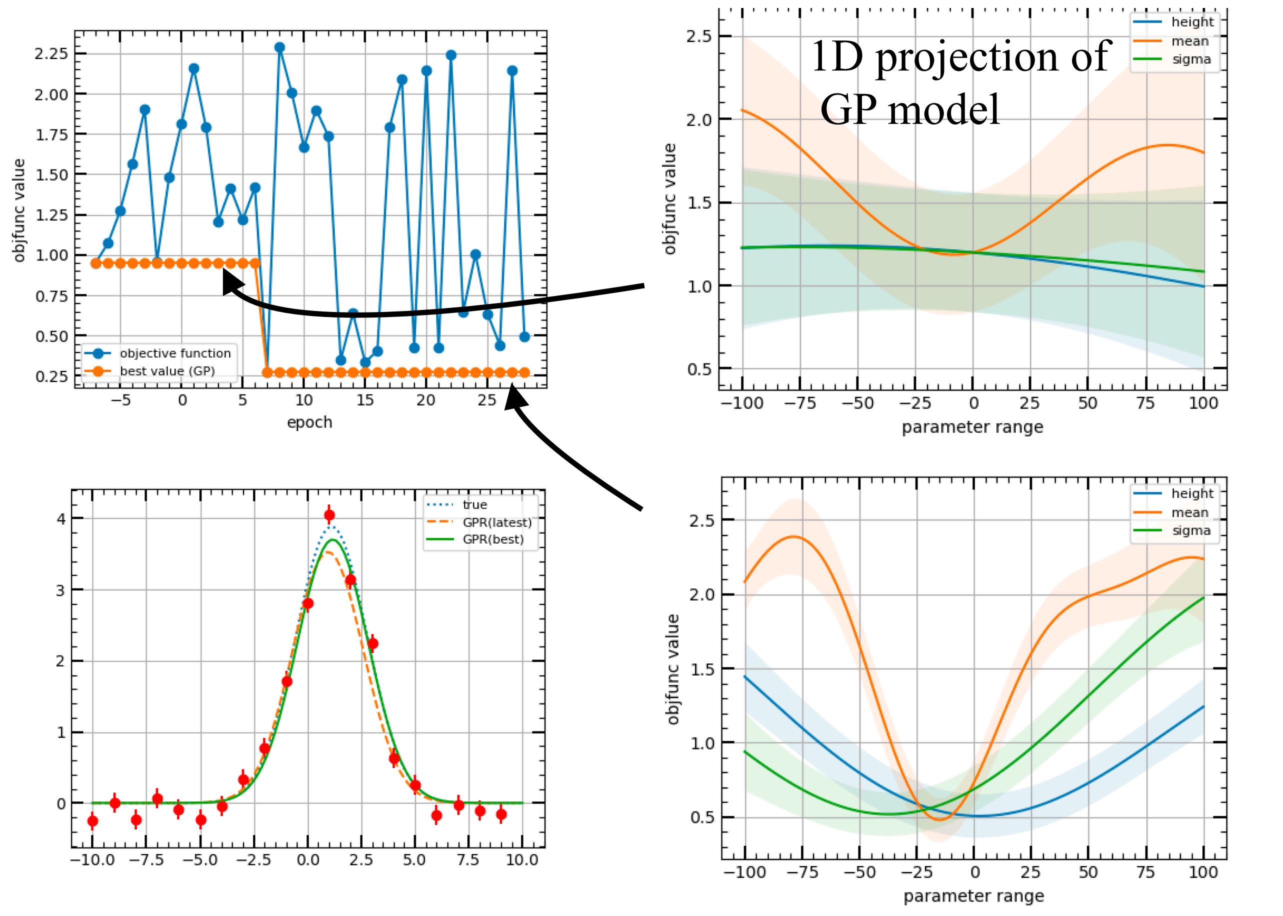


To overcome them, we've established an optimizer based on the Gaussian Process Regressor, called GPR optimizer.

GPR was a good starting point
ex) easy to understand and control the behavior
ex) easy to use libraries shared in a market

Introduction to the GPR optimizer

Gaussian fit



Core calculations

$$y = f(x, \theta) + \epsilon$$

y: Response function, ϵ: Gaussian noise

Bayes theorem

Prior

$$p(y^* \,|\, \mathbf{x}^*, \mathbf{y}, X, \theta) = \frac{p(\mathbf{y} \,|\, X, \mathbf{y}^*, \mathbf{x}^*, \theta) \times p(y^* \,|\, \mathbf{x}^*, \theta)}{p(\mathbf{y} \,|\, X, \theta)}$$

Posterior

Hyper parameters

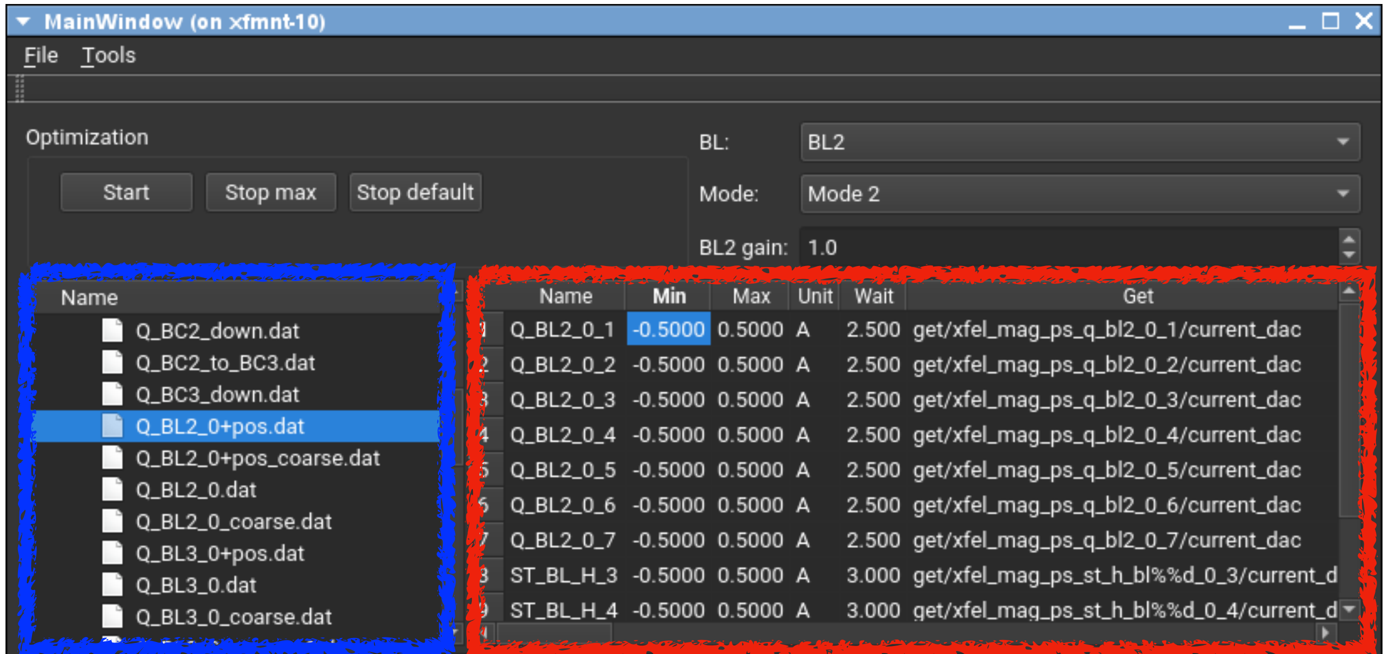
Expected improvements

$$\alpha_{\mathrm{EI}}(\mathbf{x}^*) = \int_{y_{\max}}^{\infty} (y^* - y_{\max}) p(y^* \,|\, \mathbf{x}^*, \mathbf{y}, X, \theta) dy^*$$

- Optimizer routine
  - Create a GP model with a set of initial data
  - Calculate the maxima for the expected improvements (EI)
  - Test the parameter set providing the maximum EI
  - Update the GP model

Usual operations of the beam tuning

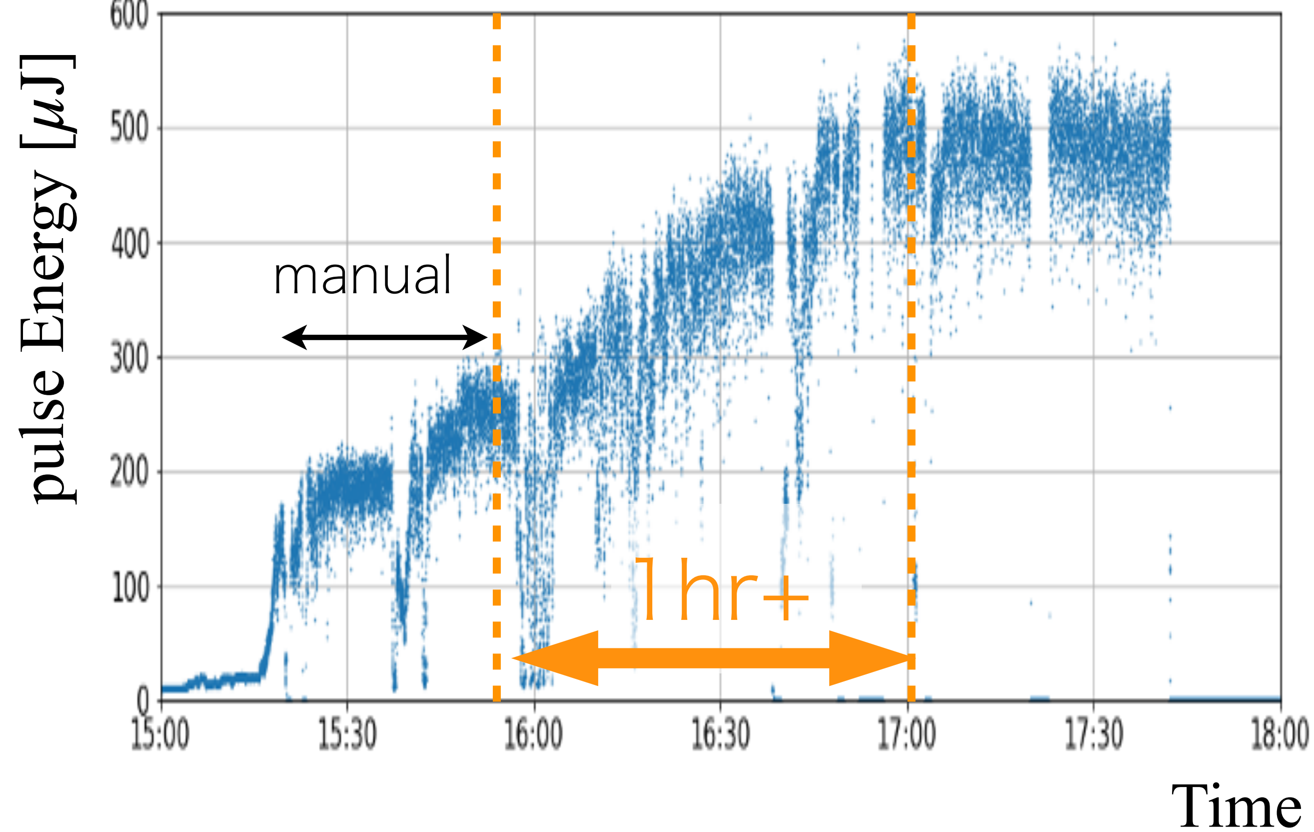
GUI window for the GPR tuning



Select parameter properties

Select beam configuration

BL3 XFEL intensity



- Weekly-based beam tuning with the GPR optimizer (python: PyTorch/BoTorch lib.)
  - Easy to use for every operator with the GUI
  - Save tuning time for operators
- Considering stability and reproducibility, 10 to 15 parameters are tuned simultaneously.

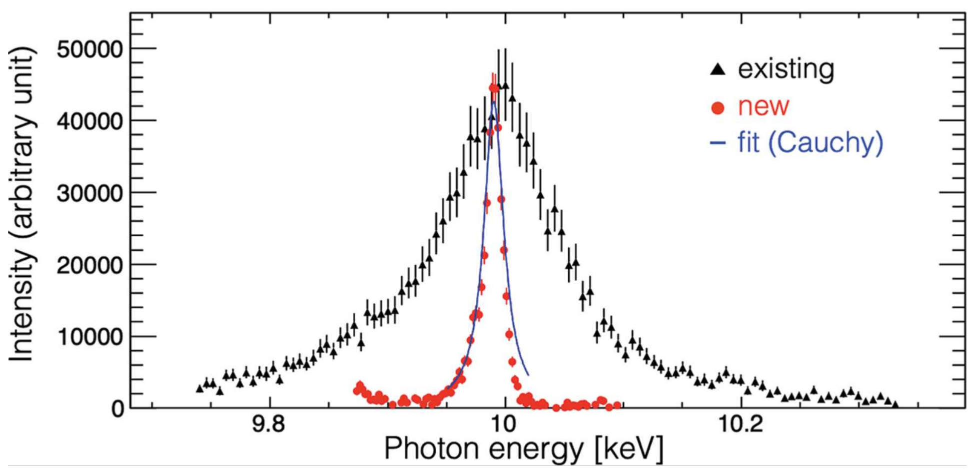
Recent achievements with the GPR optimizer

- Optimization of spectral brightness
- Suppression of side-band peak contributions in XFEL spectrum
- Tuning of beam profiles at the SACLA injector section

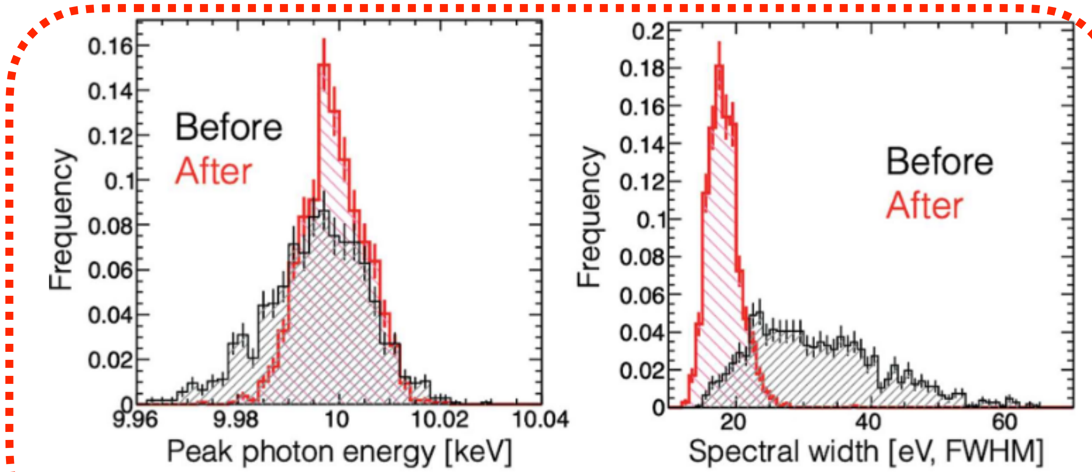
Spectral-brightness optimization of an X-ray free-electron laser by machine-learning-based tuning,
Eito Iwai et al. , JSR 30, 1048-1053, 2023

Spectral width reduced by half and improved spectral brightness by 1.7 times

Inline spectrometer with improved resolution



GPR optimizer



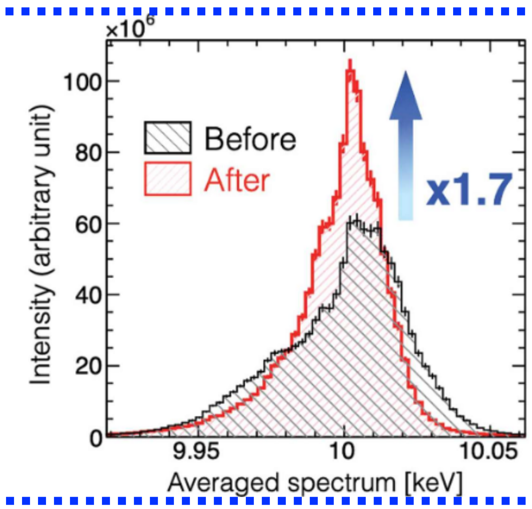
10 → 6 eV   20~60 → 20 eV

$$\text{Object Function} = \frac{\text{PulseEnergy } [\mu J]}{\sigma \ [\text{eV}]}$$

$$\sigma^2 = \sigma^2_{\text{mean}} + \sigma^2_{\text{width}}$$

σ_{mean} : uncertainty of the mean wavelength

σ_{width} : uncertainty of shot-by-shot pulse width



Recent achievements with the GPR optimizer

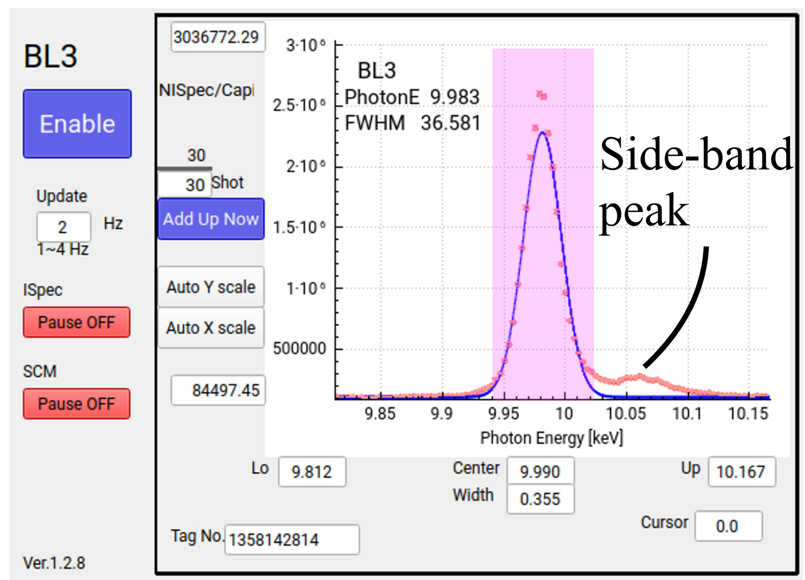
- Optimization of spectral brightness
- Suppression of side-band peak contributions in XFEL spectrum
- Tuning of beam profiles at the SACLA injector section

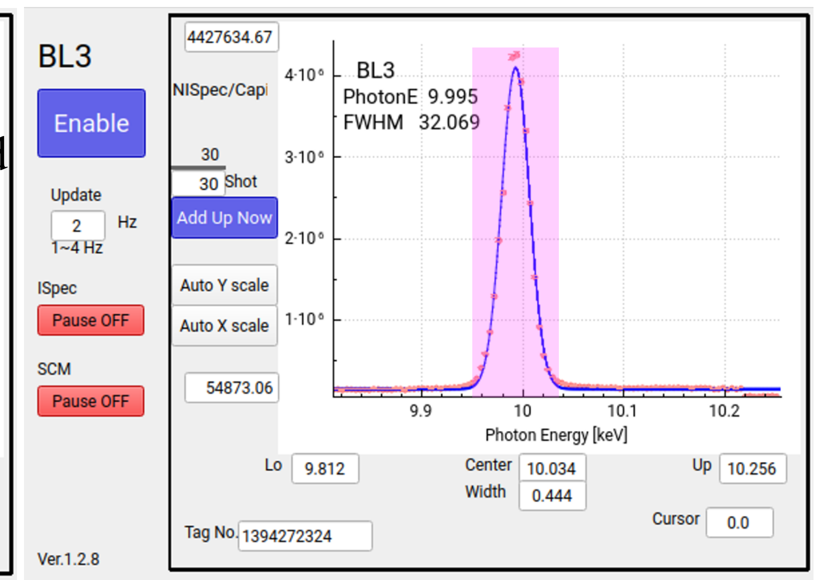
Realize the user request to suppress the side-band peak in XFEL spectrum

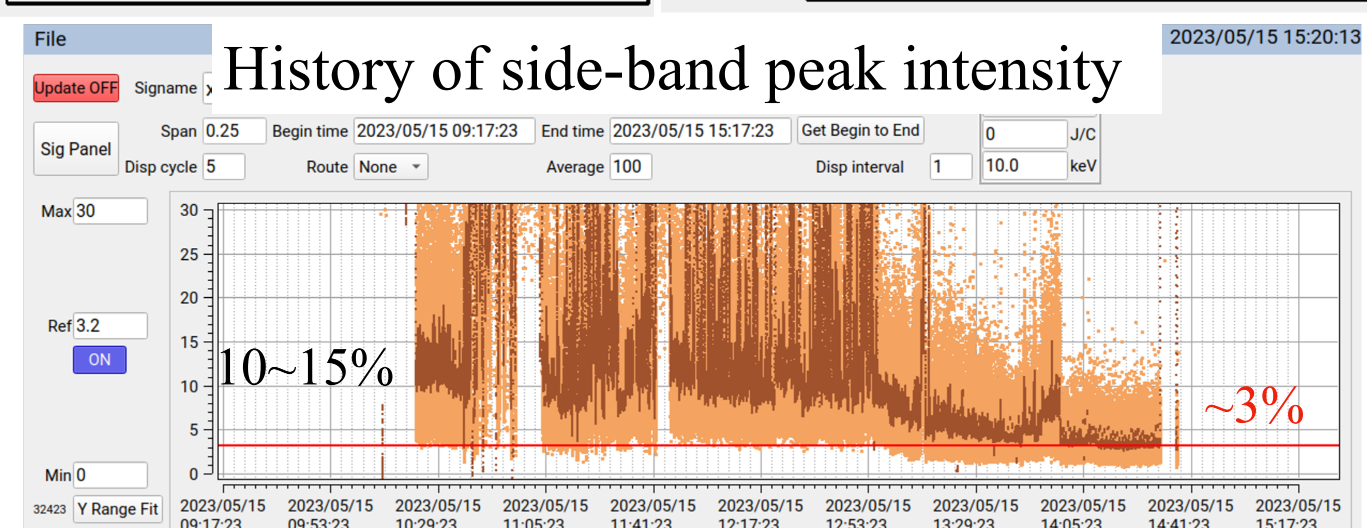
Background
Some side-band peak contributions cannot be suppressed by the optimization of the spectral brightness.

Inputs
Maximize the mean spectral brightness weighted by the side-band peak intensity.
(Lower weight for a large side-band peak intensity)

$$\text{Obj. func.} = \sum_{i}^{N} w_i \frac{\text{pulseEnergy}_i\ [\mu\text{J}]}{\sigma_i\ [\text{eV}]}$$







Recent achievements with the GPR optimizer

- Optimization of spectral brightness
- Suppression of side-band peak contributions in XFEL spectrum
- Tuning of beam profiles at the SACLA injector section

Realize an automatic tuning of 2D beam profiles at the injector section

Background

Difficult to tune 2D profiles or their 1D projections

Biased by the dynamic range or gain settings

Inputs

Use reduced χ^2 for the residual between input pixel values and the reference pixel values

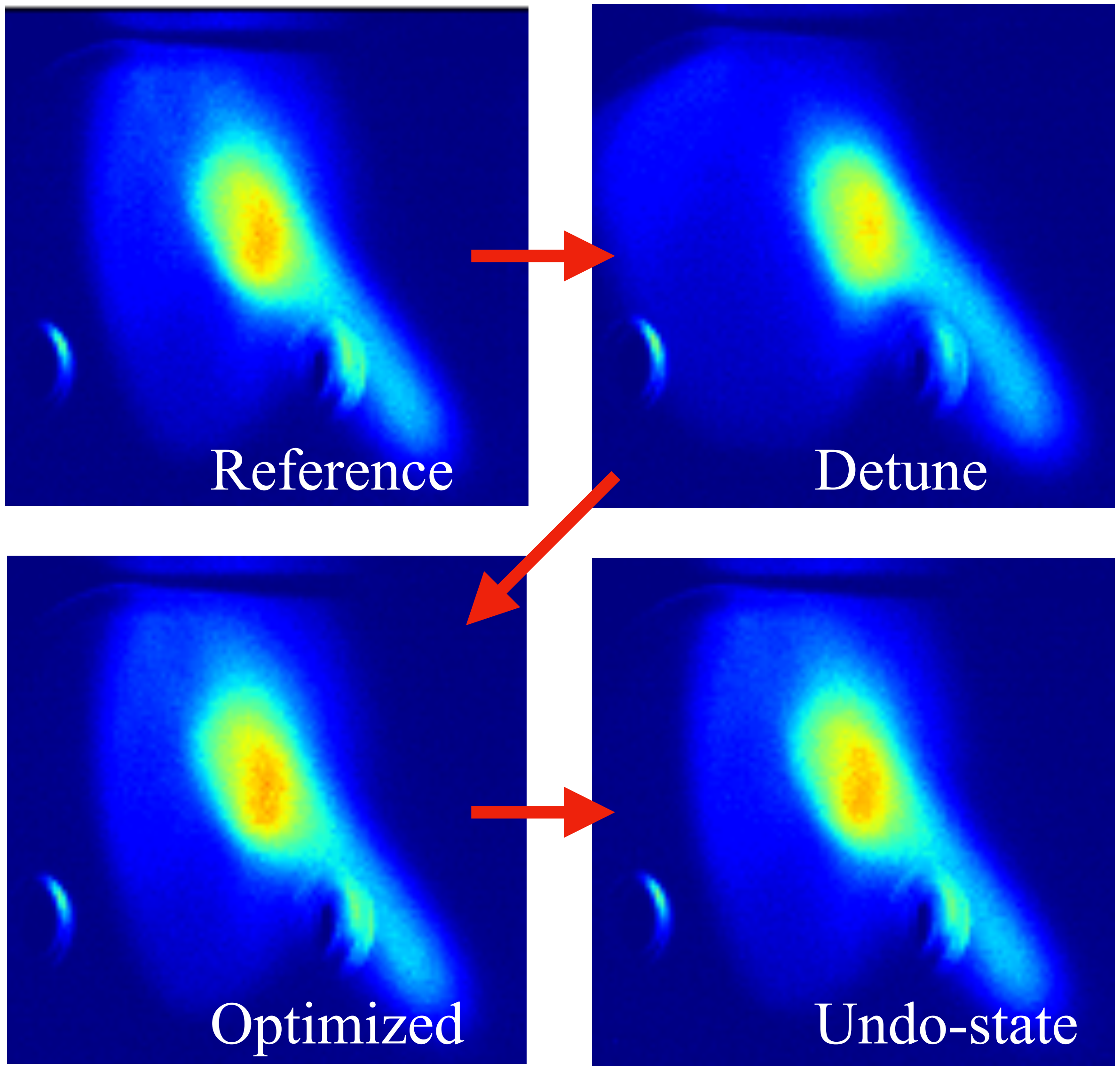
$$\chi^2 = \sum_i \left(\frac{x_i - k x_i^{\mathrm{ref}}}{\sigma(x_i)} \right)^2, \; k = \sum_i \frac{x_i}{x_i^{\mathrm{ref}}}$$

$$\sigma(x_i)^2 = \sigma_{\mathrm{p-stat}}^2(x_i) + \sigma_0^2$$

$\sigma_{\mathrm{p-stat}}(x_i)$: uncertainty of photon statistics

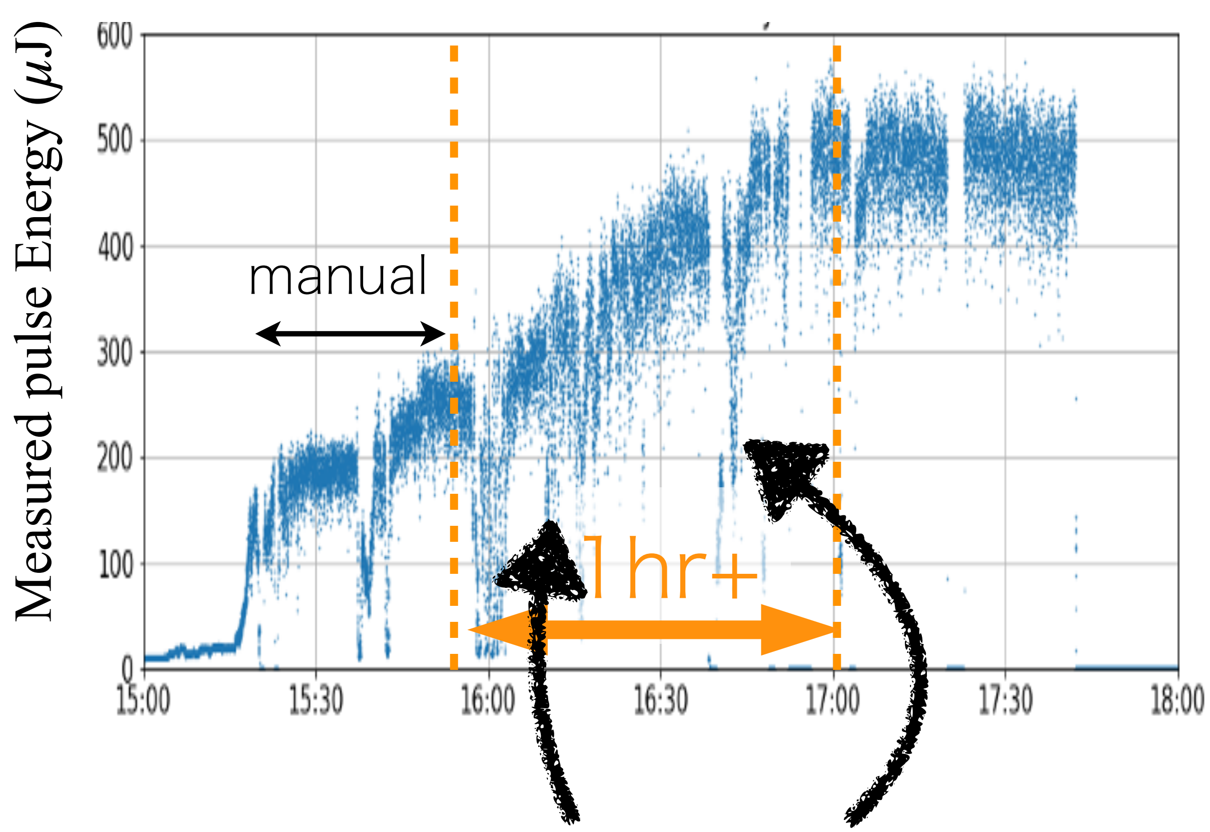
σ_0: pedestal fluctuation independent of outputs

Insensitive to total charge, iris, exposure time, and range/gain settings

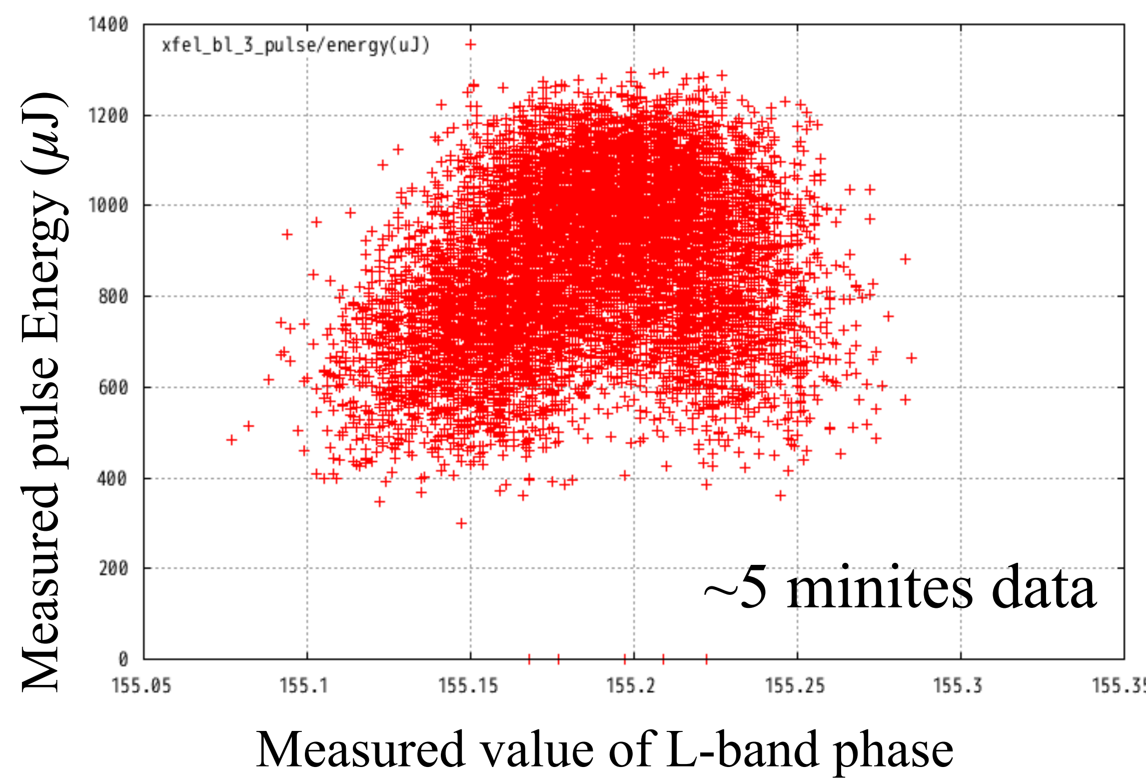


Difficulties in the GPR optimizer

- The optimizer is difficult to use during user operations because of large shot-by-shot fluctuations in XFEL pulse intensity.
  - Making the step width narrower is not efficient
- Learned knowledge of parameter correlations is temporal and not used in the next-time beam tuning.



Sudden drops during the optimization (Note: This is good in terms of the best-fit not being trapped in the local minimum.)

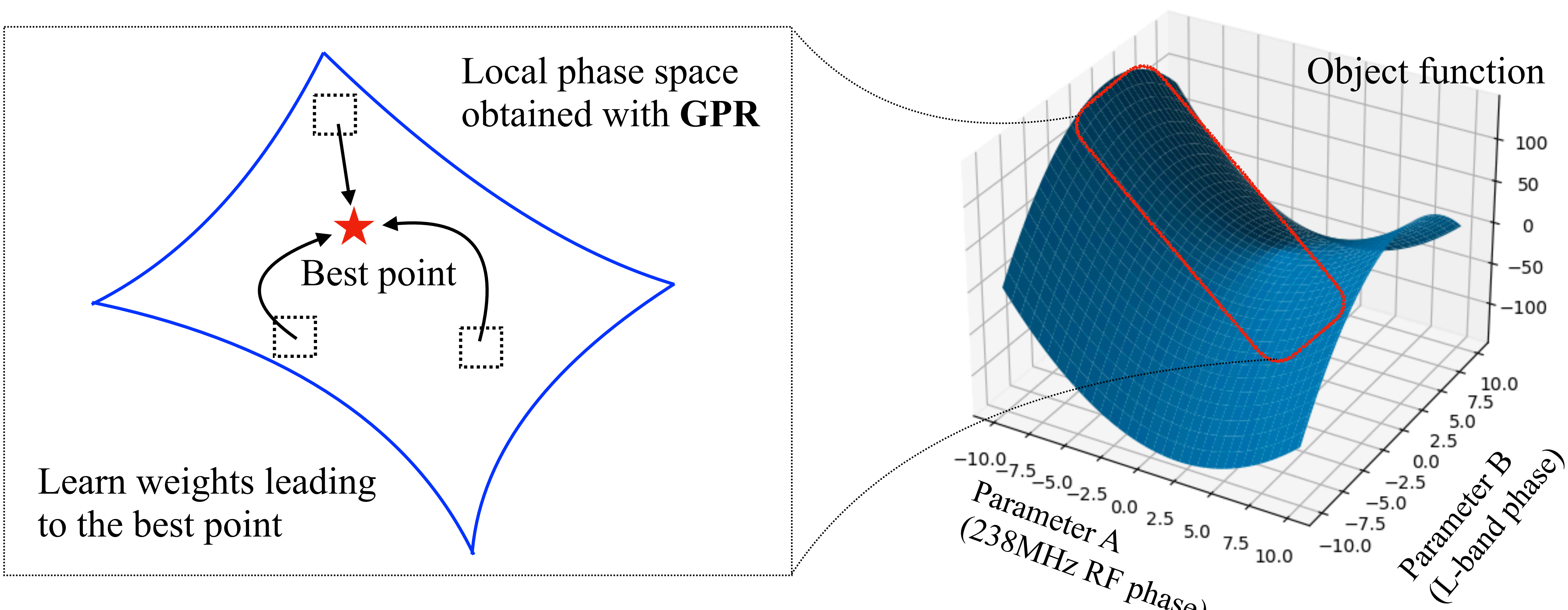


The optimizer tries to find the best-fit point without referencing the correlations found during previous beam tuning.

A deep learning method is expected to overcome these difficulties.

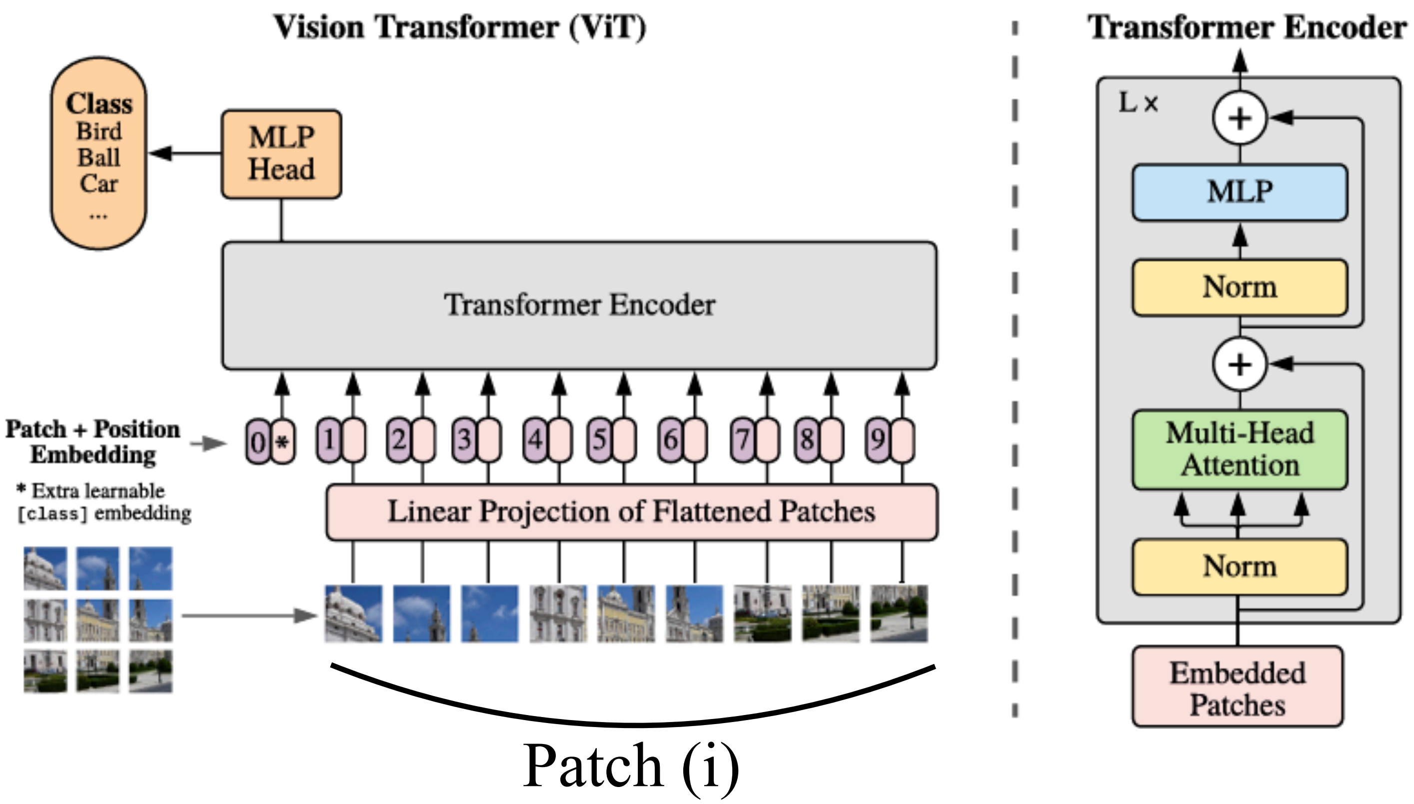
Concept of a new system

- Purpose: Faster beam tuning and automatic beam control during user time
- Learning strategy: Use the GP models obtained at each beam tuning to learn weights leading to the best parameter set depending on states
- Algorithm: At present, Vision-Transformer (VT) model
  "*An image is worth 16 * 16 words: Transformers for image recognition at scale*"
  * The idea to use VT was suggested by Prof. Yuta Nakashima (Osaka Univ.)

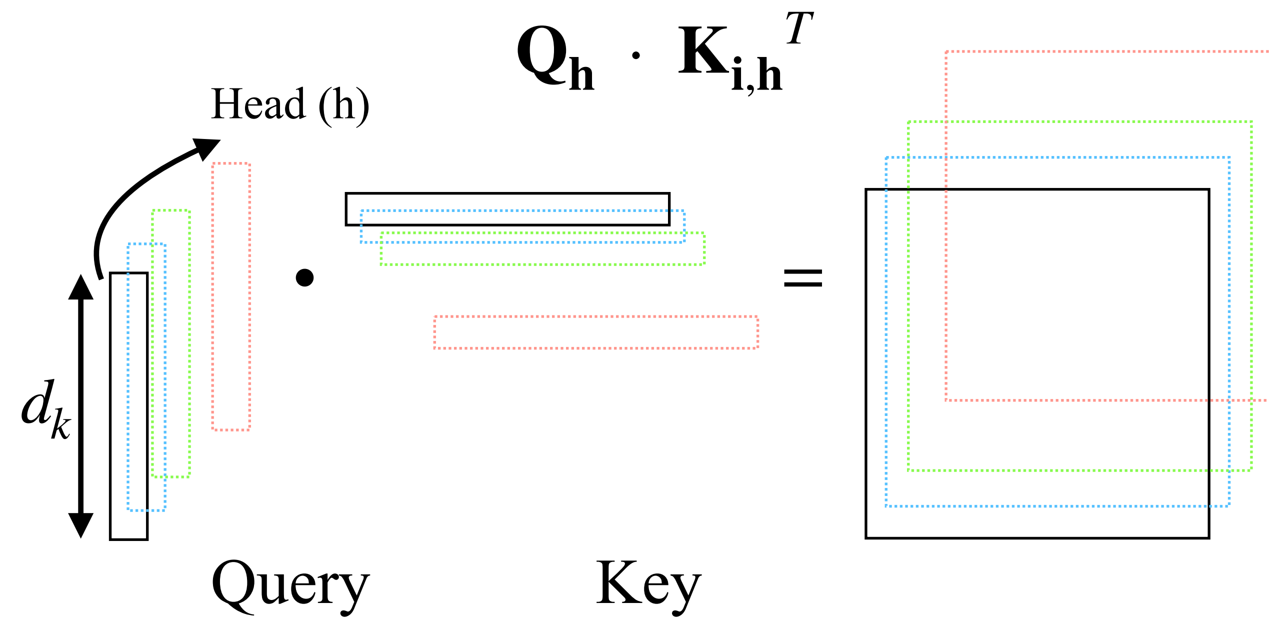


An idea to use VT ①

Diagram



Core: Attention mechanism

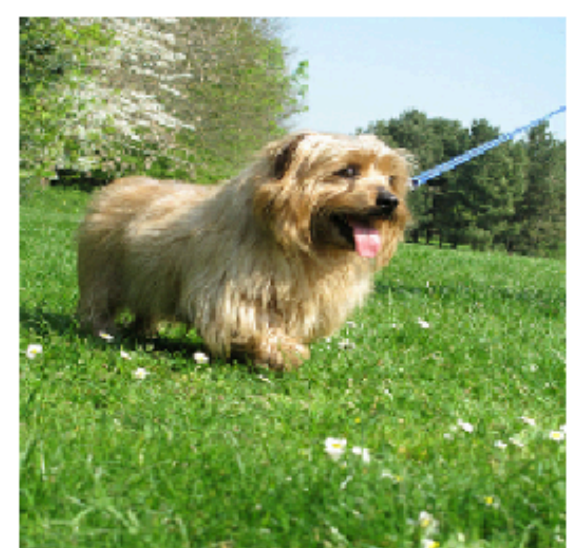


Attention ($\mathbf{q}$) = Σ_i softmax ($\mathbf{Q_h} \cdot \mathbf{K_{i,h}}^T / d_{k,i}$) $\mathbf{V}_i$

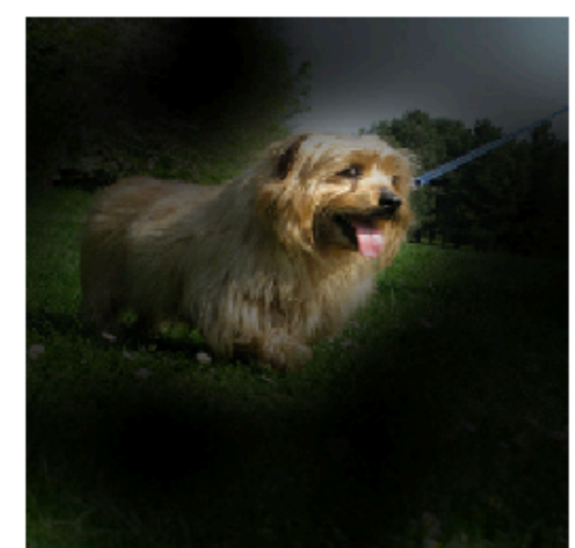
Attention is large if Q and K vectors are similar

- Why Vision Transformer?
  - Take into account the relationship between patches (parameters)
  - High-speed calculations by parallelizing on a patch-by-patch basis

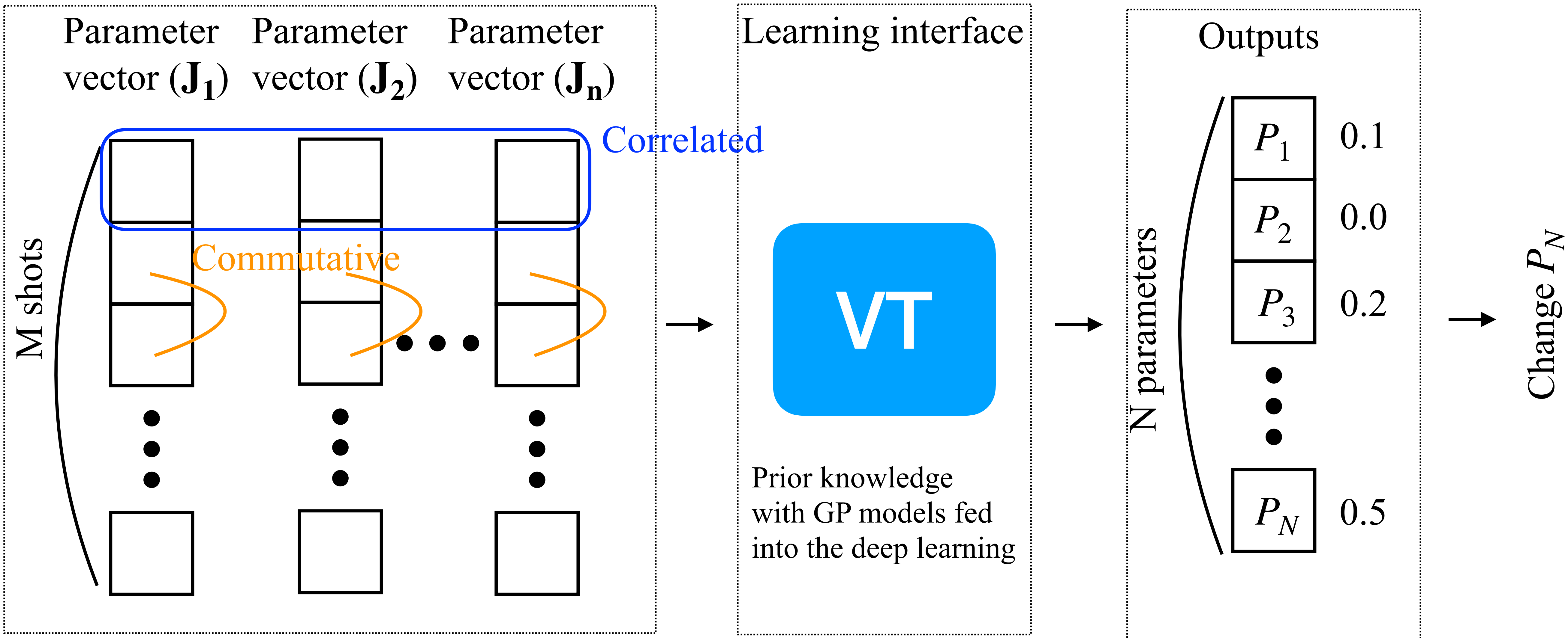
Input



Attention



An idea to use VT ②



- Inputs
  - Parameter vectors are correlated with each other on a shot-by-shot basis.
  - The vector elements are commutative.
- Outputs
  - Prediction to indicate which parameters should be changed

Tests with a simple model

Inputs

Patch 0	Patch 1		Patch 8
7×7	7×7	$\bullet\bullet\bullet$	7×7

ex.) RF phase

9 parameter vectors $(x_0, \cdots, x_8)$

$x_i \sim N(\mu_i, \sigma_i)$

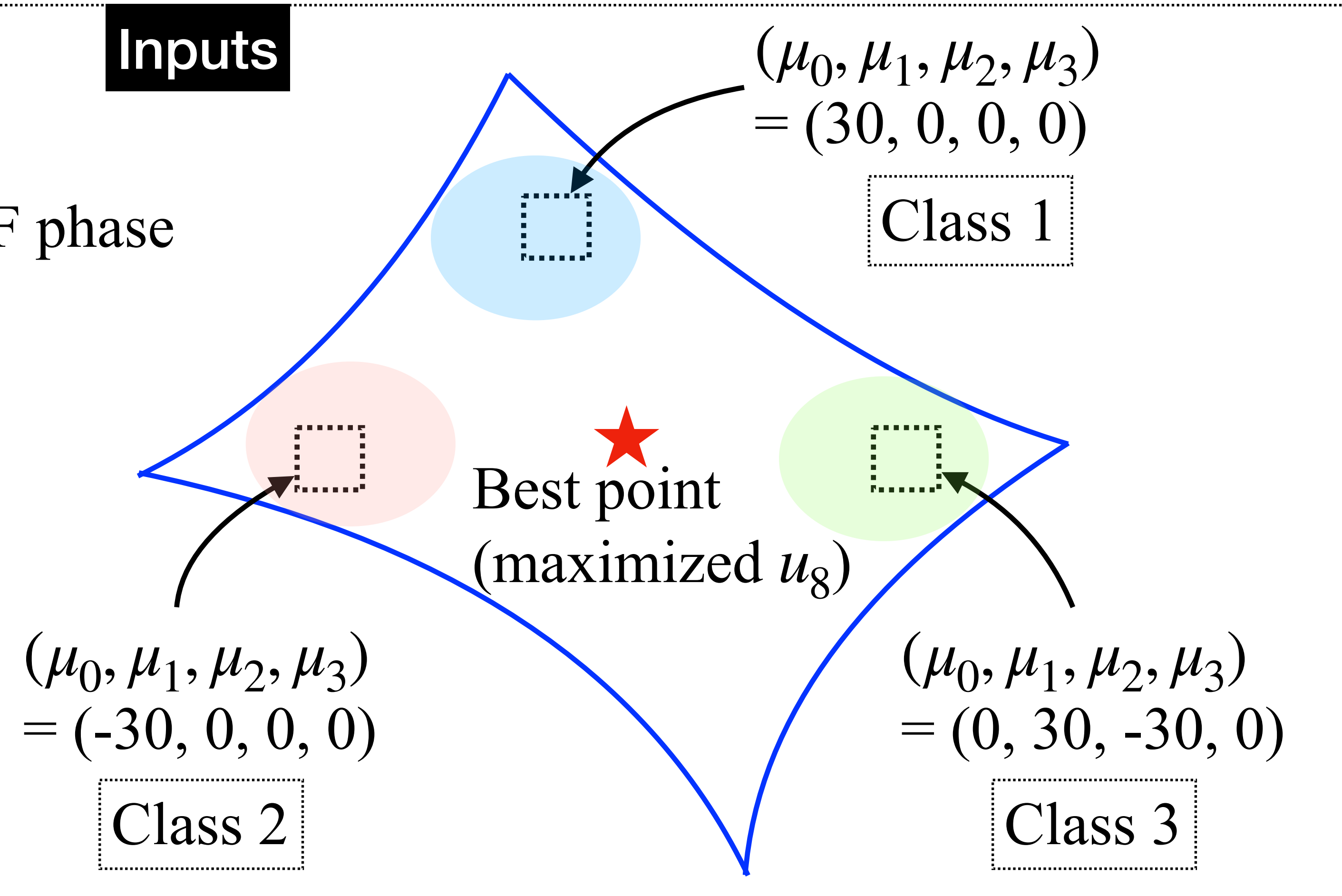
$\mu_0 \sim \mu_3$: -30 or 0 or 30, $\mu_4 \sim \mu_7$: 0

$$\mu_8 = \exp\left(-\frac{x_0^2 + x_1^2 + x_2^2 + x_3^2}{\sigma_x^2}\right)$$

Obj. Func.

$\sigma_x = 80$

$\sigma_i \ (0 \le i \le 8) : 1$



Vision Transformer

- Vision Transformer: python package, vit_pytorch (https://github.com/lucidrains/vit-pytorch)
- Test 1: Assuming a certain function over parameters $(x_0 \sim x_8)$, check if the algorithm can recognize the current phase space.
- Test 2: Check if the agent can approach the best point from an arbitrary one with the learned weights.

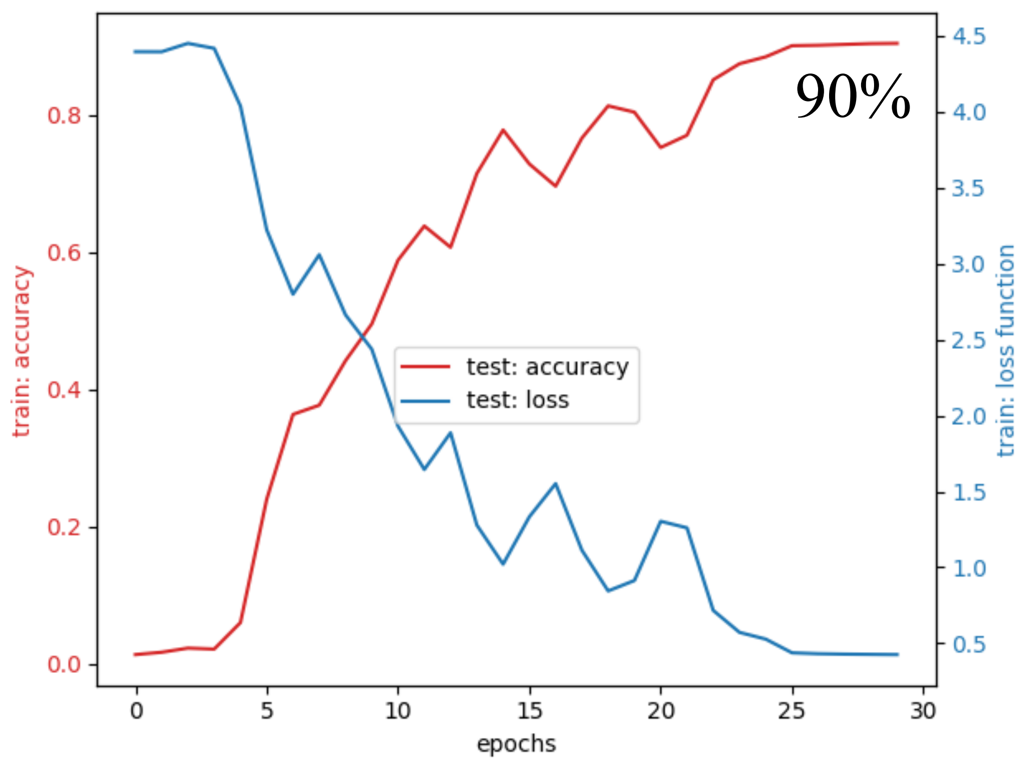
VT model variants

Layers	4
Hidden size	64
MLP size	64
Head	4
Batch	16
Num class	81

Tests with a simple model

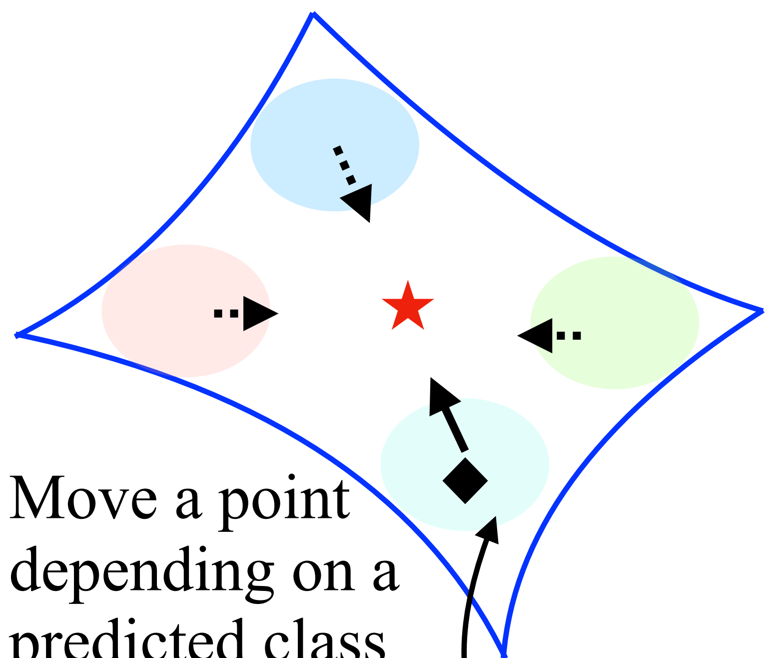
- Test 1: Assuming a certain function over parameters ($x_0 \sim x_8$), check if the algorithm can recognize the current phase space.
- Test 2: Check if the agent can approach the best point from an arbitrary one with the learned weights.

Test1



Successfully learn the weights

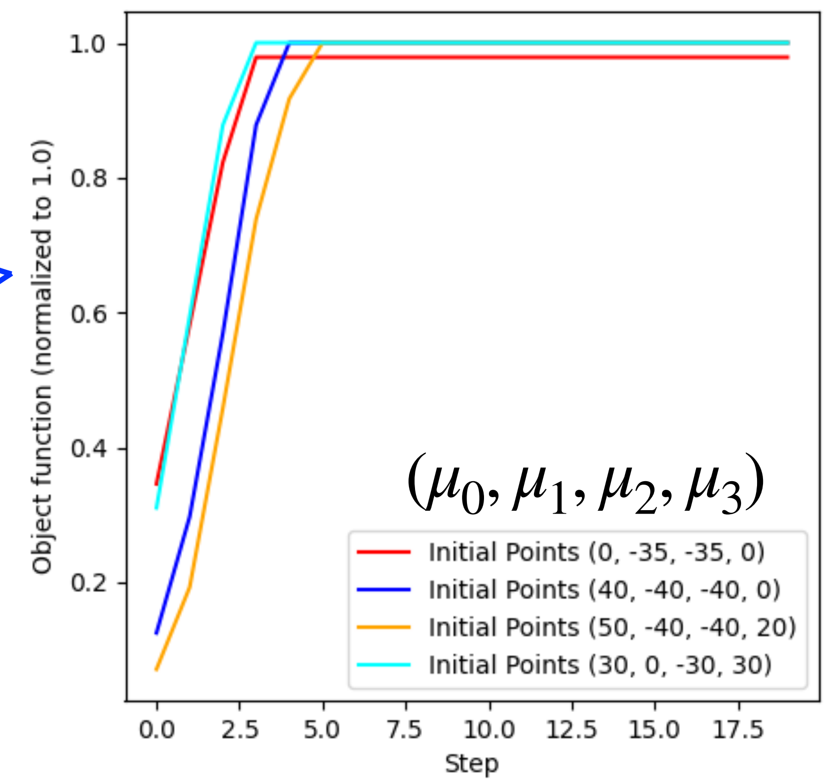
Test2



Move a point depending on a predicted class (add ±10 or 0 to $x_0 \sim x_3$)

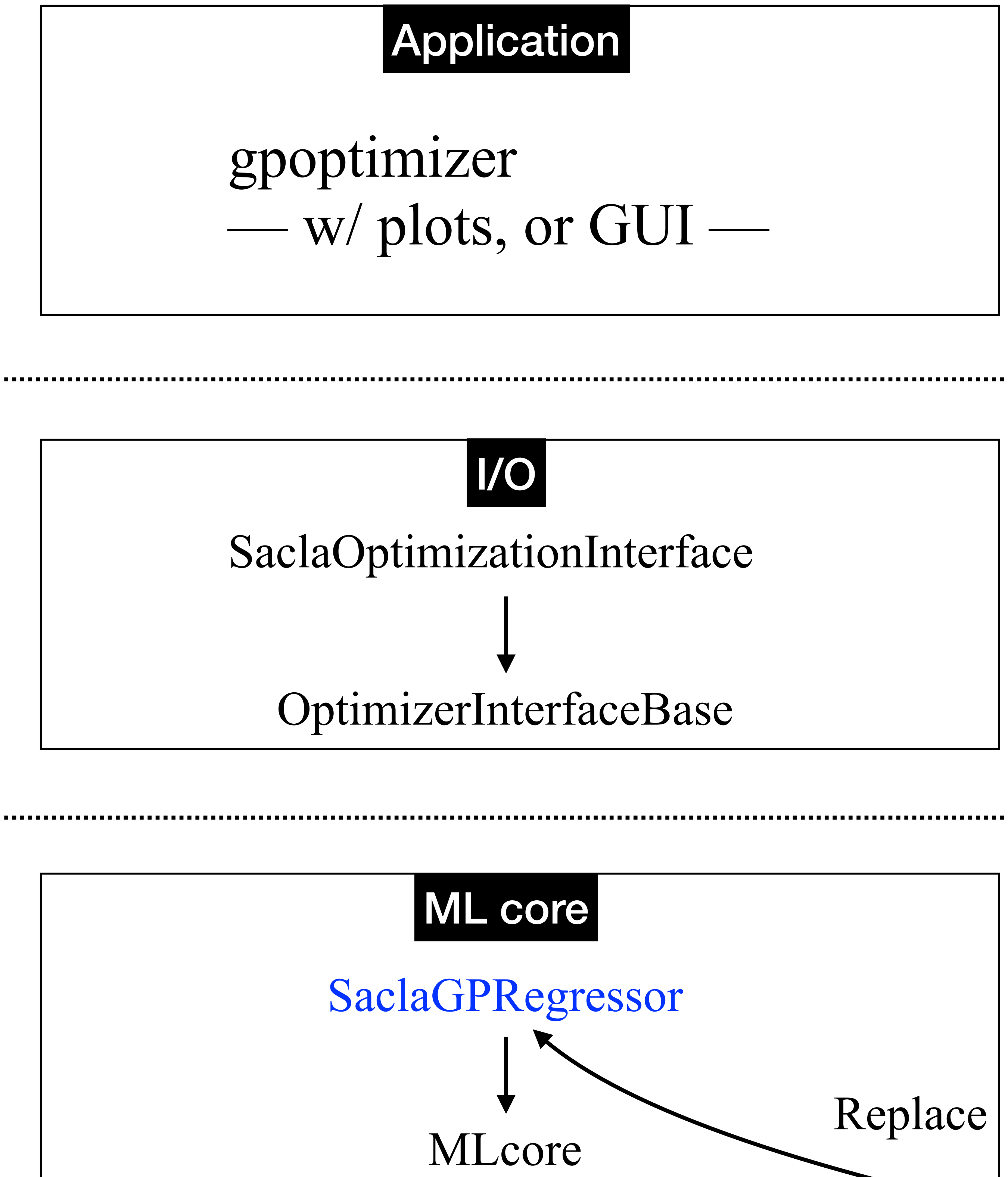
Starting point

example
Class 1 (30, 0, 0, 0): add -10 to x_0
Class 3 (0, 30, -30, 0): add -10/10 to x_1/x_2



Successfully find the points around the best point

Development of the code management



- Three layers
  - Application
    - Change tuned parameters, different beamlines
  - I/O
    - Change facility-based configurations
  - ML core
    - Change GPR/VT methods
- Facility- or purpose-specific parts are isolated from the method part.
  - Easy to replace the methods

(SaclaVT)

Summary

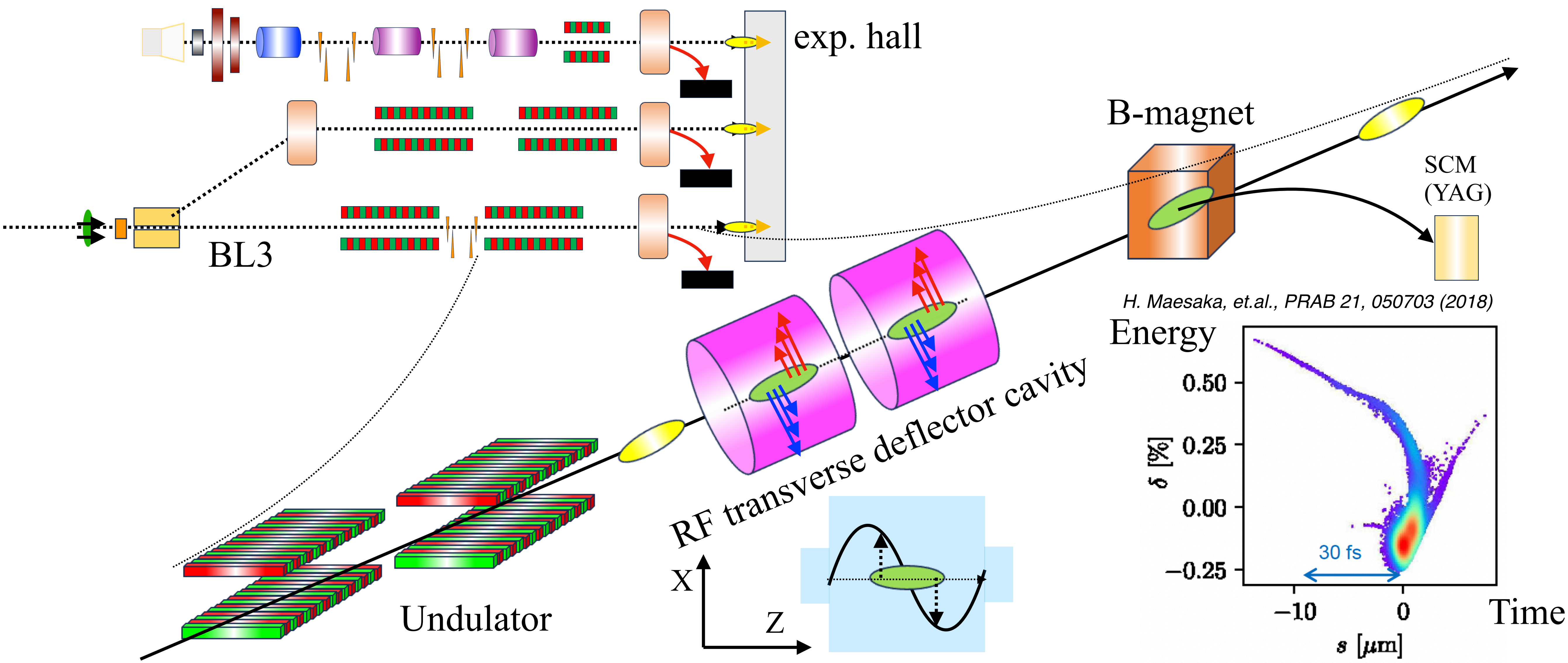
- The GPR optimizer has been successfully implemented and utilized in the XFEL facility, SACLA.
    - We reduce spectral width by half and improve spectral brightness by 1.7 times.
    - We are able to suppress side-band peak contributions at a ~3% level.
    - We are also able to tune 2D profiles with the GPR optimizer.
- We have been developing the Vision-Transformer-based deep learning method for a more efficient beam control with ML.
    - Making use of the GP models obtained in the usual beam tuning is expected to enhance the method's performance.
    - We conducted a simple test to simulate the parameter tuning with VT. The results seem to be encouraging.

Backup

Summary

- The GPR optimizer has been successfully implemented and utilized in the XFEL facility, SACLA.
    - We reduce spectral width by half and improve spectral brightness by 1.7 times.
    - We are able to suppress side-band peak contributions at a ~3% level.
- We have been developing the Vision-Transformer-based deep learning method for a more efficient beam control with ML.
    - Making use of the GP models obtained in the usual beam tuning is expected to enhance the method's performance.
- Preparation of meaningful inputs is key to achieving a higher performance of ML. To this end, we have been developing a longitudinal beam diagnostics system that will provide energy-time information on the electron beam.
    - We aim to generate a short and stable XFEL pulse at a ~1 fs scale by combining the existing and future ML methods.

Development of beam longitudinal diagnostics system

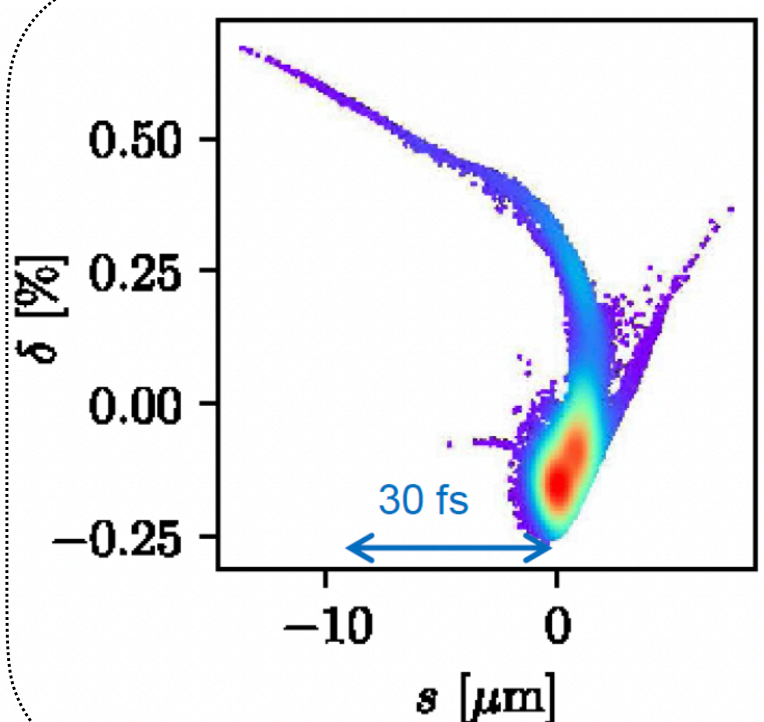


- Motivation: Diagnose the ~10 fs time structure of the electron beam
  Detect a longitudinal lasing part in electron bunches
  Realize a stable short XFEL pulse (~1 fs) to dig into “atto-physics”
- Requirements: ~1 fs time resolution with 2~3 m-long cavities in total
- Schedule: Design (2023), Construction (2024), High power test (2025), Installation and Operation (2026)

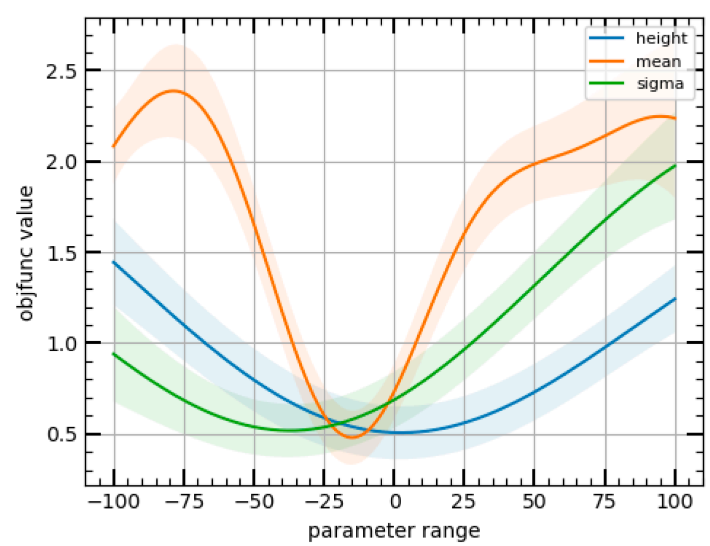
Synagy between ML and the beam diagnostics system

- Essences to enhance ML performances
  - Sophisticated algorithms
  - **Meaningful inputs** (as demonstrated in the "recent achievements")

Inputs (diagnostics system)

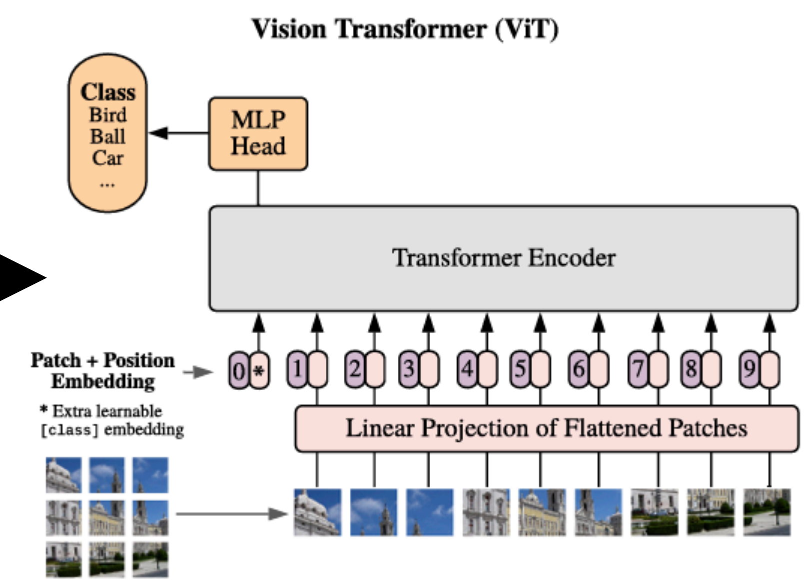


GPR



Short XFEL pulse

VT



Stable & Short pulse

ML $\underset{\text{More input options}}{\overset{\text{Higher performances}}{\rightleftarrows}}$ Development of hardwares