



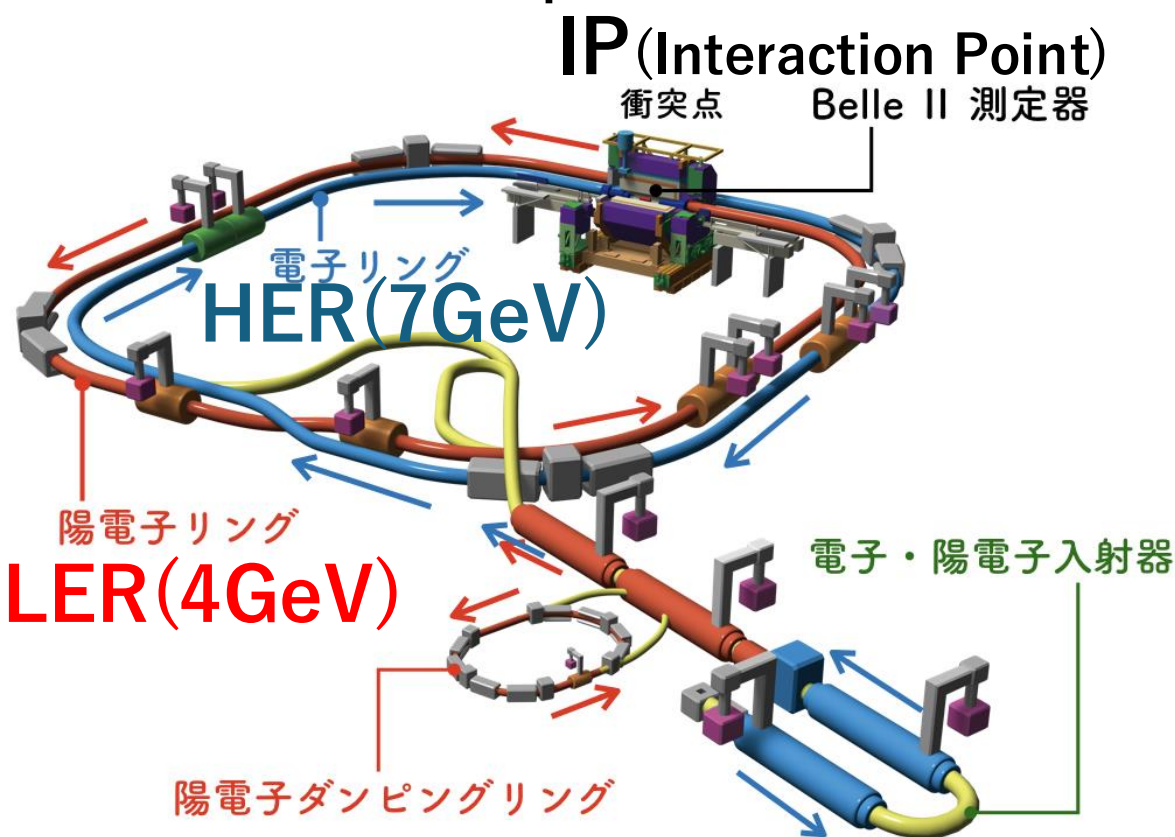
SuperKEKB加速器の 衝突パラメータ調整の高度化

瀧澤陸^A, 原康二^{B, C}, 住澤一高^{B, C, D}, 中山浩幸^{B, C}, 後田裕^{A, B, C},
古賀太一朗^{B, C}, 野尻美保子^{B, E}, Ahmed Hammad^B

東大理^A, KEK素核研^B, 総研大^C, 奈良女大^D, IPMU^E

Belle II 実験と SuperKEKB 加速器について

Belle II 実験と SuperKEKB 加速器



Belle II 実験は現在世界最高のLuminosityを誇る
SuperKEKB加速器を用いた電子陽電子衝突実験

$$N = \sigma \int L dt$$

SuperKEKB加速器において
目標Luminosity実現のため毎日行われている調整の一部

調整の種類	自動化ツールの有無
衝突パラメータ調整	無（オペレーターが手動で調整）
入射パラメータ調整	有（ベイズ最適化による自動調整）
軌道調整	有（30分に1回自動調整）

SuperKEKB加速器の衝突パラメータ (IP knob parameters)

ルミノシティ L の簡単な表式: $L \propto \frac{N_b I_{b+} I_{b-}}{\sqrt{\sigma_{x+}^2 + \sigma_{x-}^2} \sqrt{\sigma_{y+}^2 + \sigma_{y-}^2}}$

IP knob parameters (以下IP knob) を用いたビームサイズ σ の簡単な表式:

$$\sigma_y^2 = \langle y^2 \rangle = \mu^2 \epsilon_y \left(\beta_y^* + \frac{\Delta s^2}{\beta_y^*} \right) + \epsilon_x \beta_x^* (r_1 + r_3 \Delta s)^2 + \epsilon_x \frac{(r_2 + r_4 \Delta s)^2}{\beta_x^*} + ((\eta_y + \eta_{py} \Delta s) \sigma_\delta)^2$$

$r_1 \sim r_4$: X-Y(進行方向に垂直)結合を記述するIP knob

$\eta_y \sim \eta_{py}$: ビームの進行方向の運動量分散に対する応答を記述するIP knob

(μ^2 : $r_1 \sim r_4$ の関数、 Δs : 衝突点の空間的なずれ、 ϵ_y : 位相空間内のビームの広がり (エミッタンス))

β_y^* : 衝突点でのベータ関数 (twissパラメータの一つ))

・ IP knobを変えると対応する電磁石の電流値が変化する

- ・ 簡単な表式では、基本的にすべてのIP knobがゼロであればビームサイズが最小
- ・ しかしリング周回中に発生するX-Y結合や運動量分散を相殺するためにIP knobを調整する

SuperKEKB加速器の衝突パラメータ調整

衝突パラメータ調整 (IP knob tuning) :

IPでのビームの状態を決める衝突パラメータ (IP knob parameters) を調整すること

- ・本研究ではビームサイズを直接関係のある右図の12個のパラメータを扱う

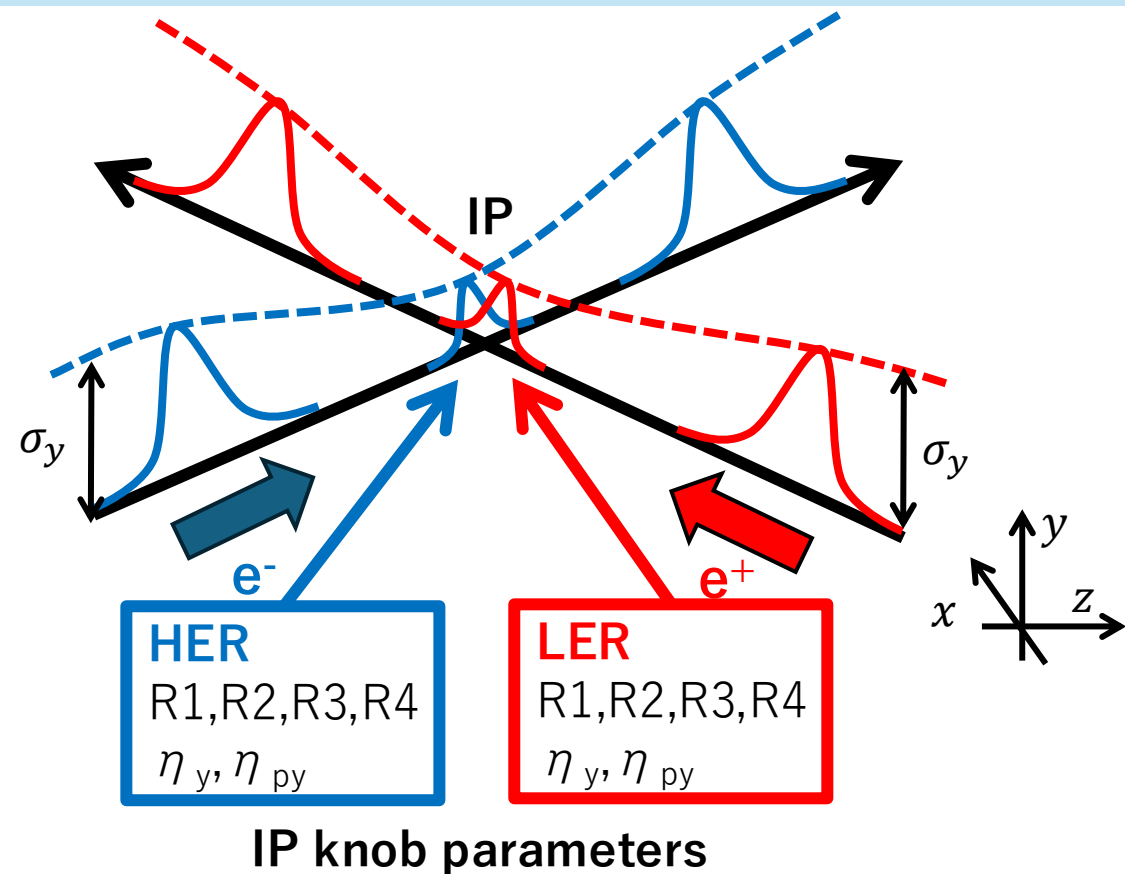
IP knob tuningの基本思想 :

Beam BackGroundを抑えつつ Luminosity増大を狙う

実際の加速器は衝突相手のビームなどからの影響で
IPでのビームサイズ σ^* 縮小→

- { Luminosity増大とは限らない
- { Beam BackGround増大の可能性

(Beam BackGround : Belle検出器付近のBeam LossがBackGroundになる)



従来のIP knob tuningの状況

一般的なtuning手法

Scan : なんらかの手段でパラメータをスキャン

Set : ある判断基準でスキャン結果から最適値を選択

従来のIP knob tuning

Scan : IP knobを一つずつ手でリニアスキャン

- ・ 元値を中心に平均的に10点ほどスキャン
- ・ スキャンの刻み幅(Step幅)はIP knob毎に定義
- ・ 各点で1分弱かかるので平均的に10分程度かかる

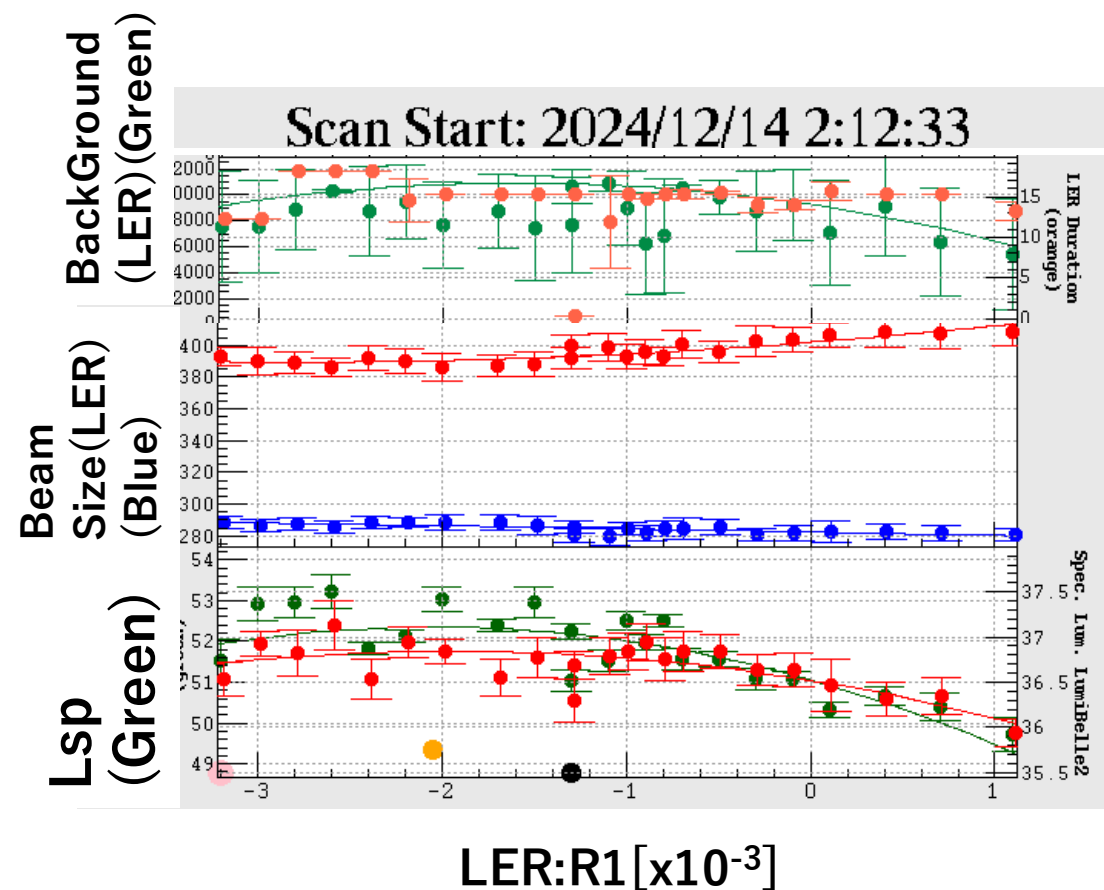
Set : スキャン結果を目で見て最適値を選択

Specific Luminosity(Lsp) :
$$Lsp := \frac{L}{I_{b+}I_{b-}N_b}$$

I_{b+} / I_{b-} : LER/HERの1bunchあたりの電流

N_b : リング一周あたりのbunch数

電流値の変動に左右されずに衝突性能を評価できる



本研究のモチベーション

従来のIP knob tuningが抱える課題

- ・オペレーターの拘束時間が大きい。
- ・調整結果がオペレーターの経験と熟練度に依存している。

→IP knob tuningに**ベイズ最適化**や**機械学習**を適用して

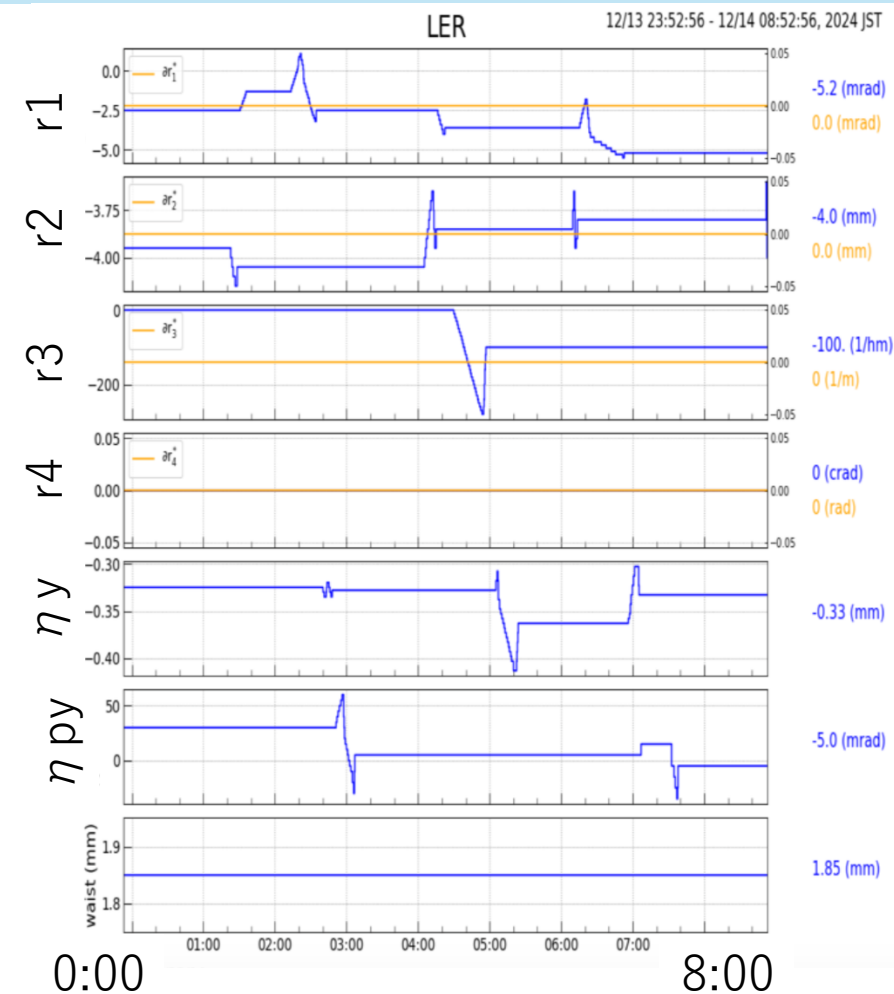
- ・現在のオペレーションよりも効率的なものにしたい
- ・オペレーターの経験差に依存しない調整を行いたい

ベイズ最適化を使用するメリット

- ・事前学習が必要ない

→状況が変化していく加速器に適した最適化手法

- ・局所的なピークにトラップされづらい
- ・先行研究による実運用例が存在する



2025年冬期2026年冬期春季運転における本研究の概要

2.(2025冬期)

機械学習に基づくIP knobの
設定値再現(提案)ツールの
実装およびテスト運用

3.(2026春期)

機械学習モデルを使用した
多次元同時ベイズ最適化ツールの
実装およびテスト運用

1.(2026冬期春期)

多次元同時ベイズ最適化ツールの
実装およびテスト運用

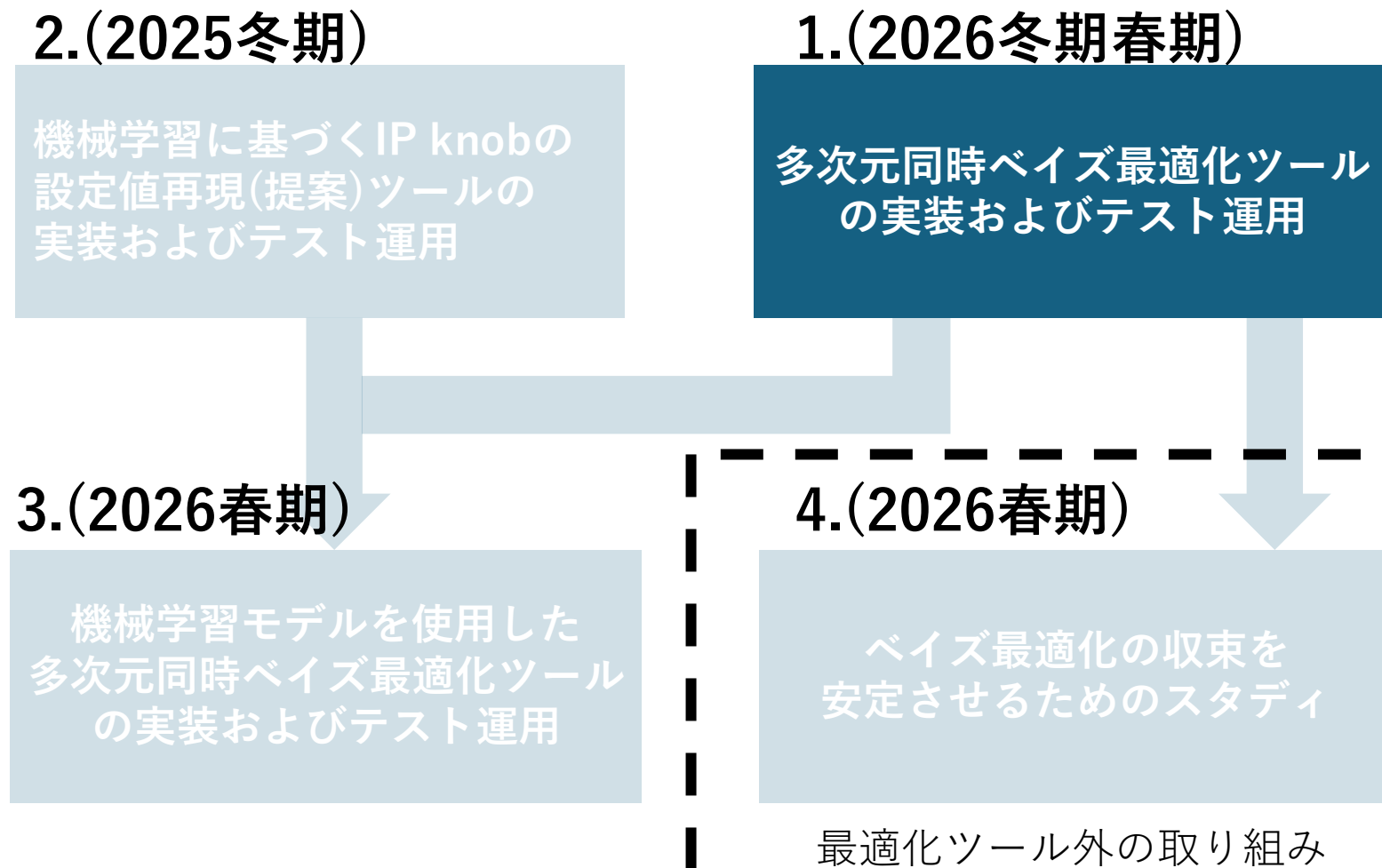
4.(2026春期)

ベイズ最適化の収束を
安定させるためのスタディ

最適化ツール外の実験

1. (Scan)多次元同時スキャンを短時間でできるようにする
(この段階ではLspを最大化)
2. (Set)その時の加速器の状況によってどの設定値が最適かを学習し、評価指標として出力
3. (Scan+Set)2.で学習した評価指標を最大化するようなスキャンを行い人の手による最適化の上位互換としてのツールを作る
4. 外的要因によるLsp等のふらつきを考慮したベイズ最適化を実装

1.多次元同時ベイズ最適化ツールの実装およびテスト運用



1. (Scan)多次元同時ベイズ最適化ツールの実装およびテスト運用

ベイズ最適化とは

ベイズ最適化…ガウス過程回帰を用いた最適化手法

ガウス過程回帰…

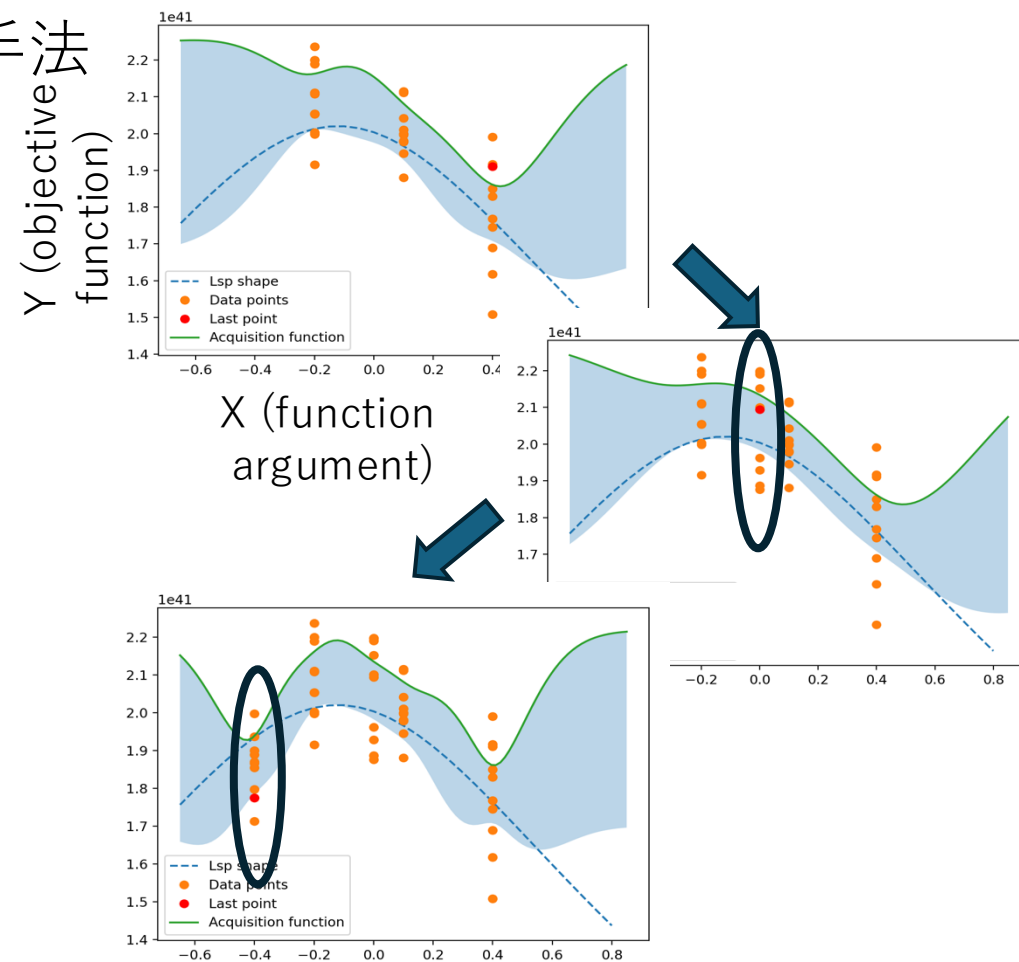
設定したカーネル関数を通して目的関数の
入力空間各点同士の相関を計算し、
関数の平均値と分散を予測

獲得関数

- ・ガウス過程回帰の出力をもとに定義される関数
- ・獲得関数が一番高いところを次にスキャンする

ベイズ最適化を使用するメリット

- ・事前学習が必要ない
- 状況が変化していく加速器に適した最適化手法



一次元ガウス過程回帰の例

1.(Scan)多次元同時ベイズ最適化ツールの実装およびテスト運用

ベイズ最適化ツールの概要

IP knobの最適値の定義

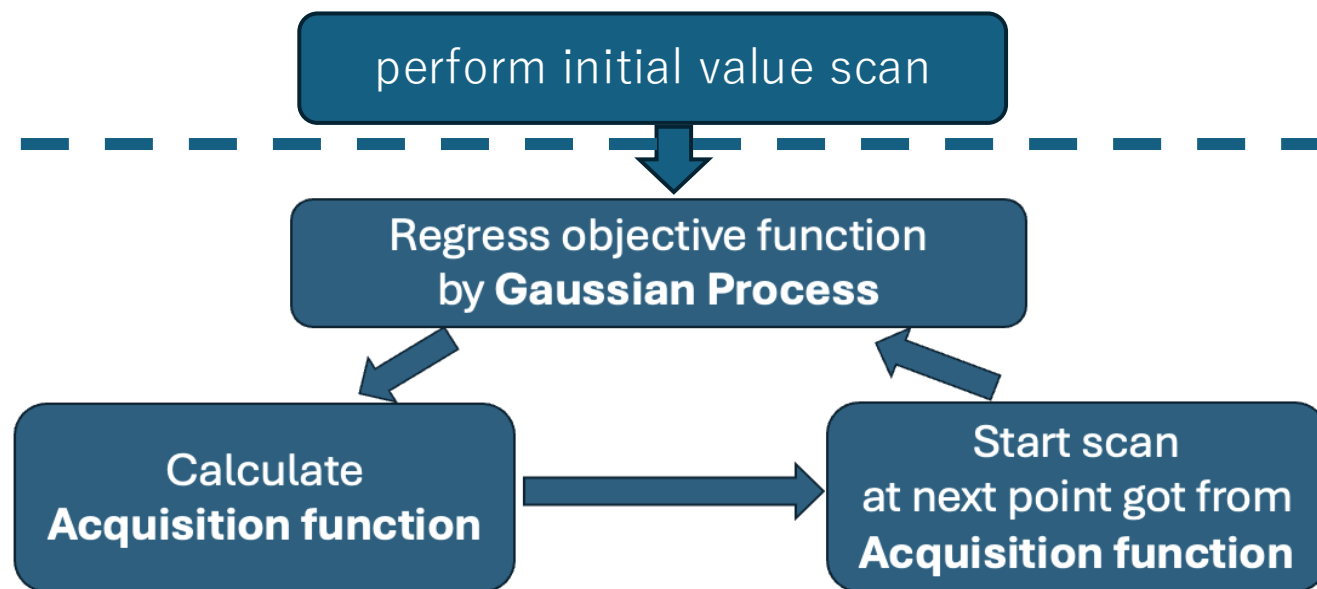
- ・ 最初是最適なknobの値はLspを最大にするものとした
- ・ 目的関数:Lsp、
獲得関数: $\mu + 0.5\sigma$
(μ と σ はガウス過程回帰の出力)

収束判定

- ・ プロセスを繰り返すうちに現在の点の近傍のみをスキャンするようになる
- ・ 次にスキャンする点が現在の点からリニアスキャン時のデータ取得幅(1step)の1/10 (0.1step) 以内であれば収束したとしてスキャンを終了する

安全面への対策

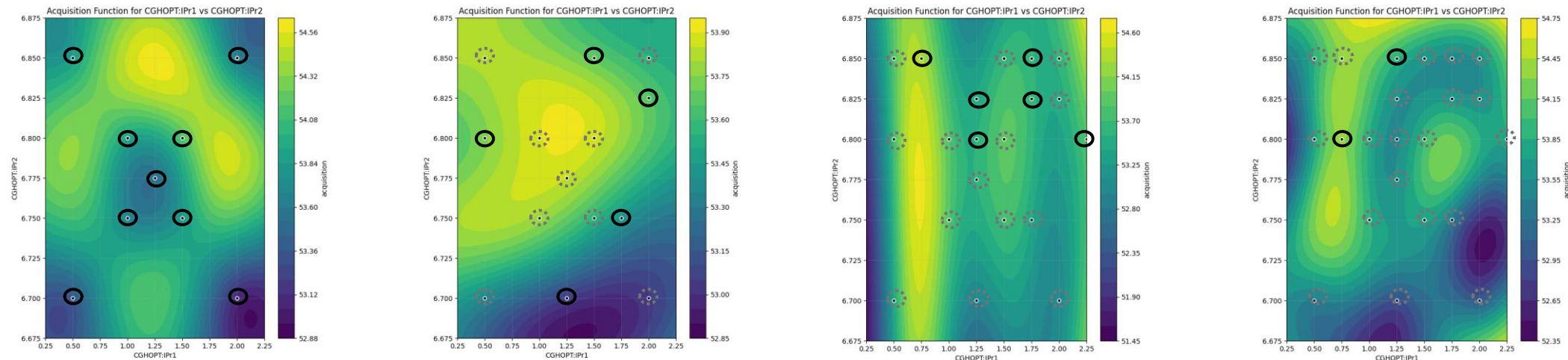
- ・ このフェーズではBackGroundを直接見ていないのでinitial scanのところでいくつかの対策を講じた



収束するまでこのループを繰り返す

1.(Scan)多次元同時ベイズ最適化ツールの実装およびテスト運用 二次元同時スキャンにおける獲得関数の変遷

物理データ取得中にツールのテストを実施した



時系列

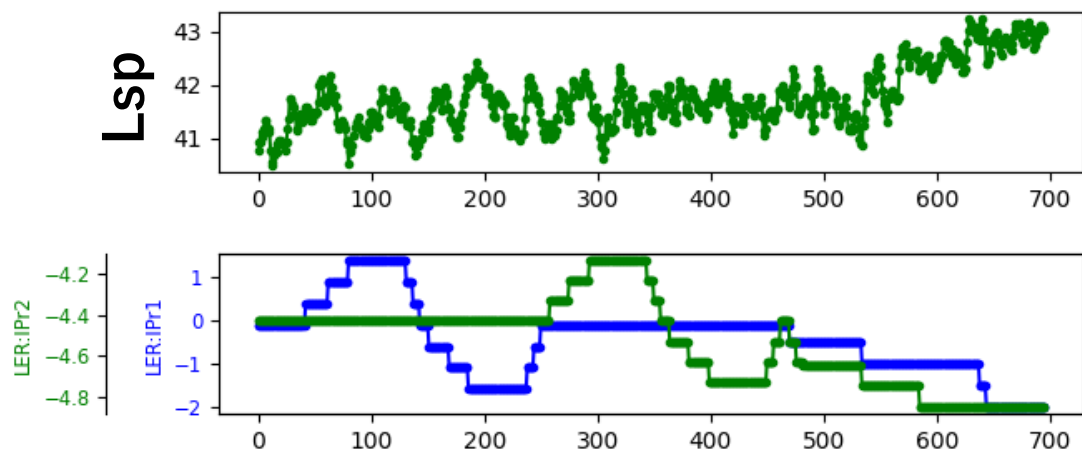
色：獲得関数、横軸：HER側R1、縦軸：HER側R2

- ・ データを取得するとそれに伴って獲得関数の形も変わっていく。
- ・ これが動きすぎるといつまで経っても収束しない。
- ・ 入力空間内の距離相関を計算する関数のパラメータ調整で二次元までなら安定して収束するようになった（ある程度の距離でも相関しているとして、関数が短い距離で急激に変化することを禁止）

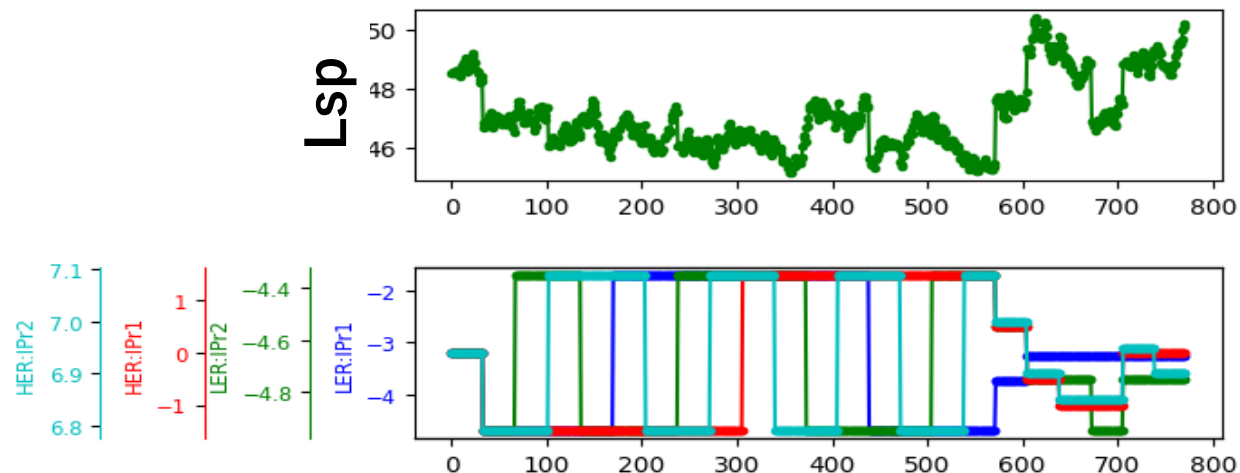
1.(Scan)多次元同時ベイズ最適化ツールの実装およびテスト運用

ベイズ最適化ツールによる多次元同時スキャンの様子

二次元スキャンへの適用結果



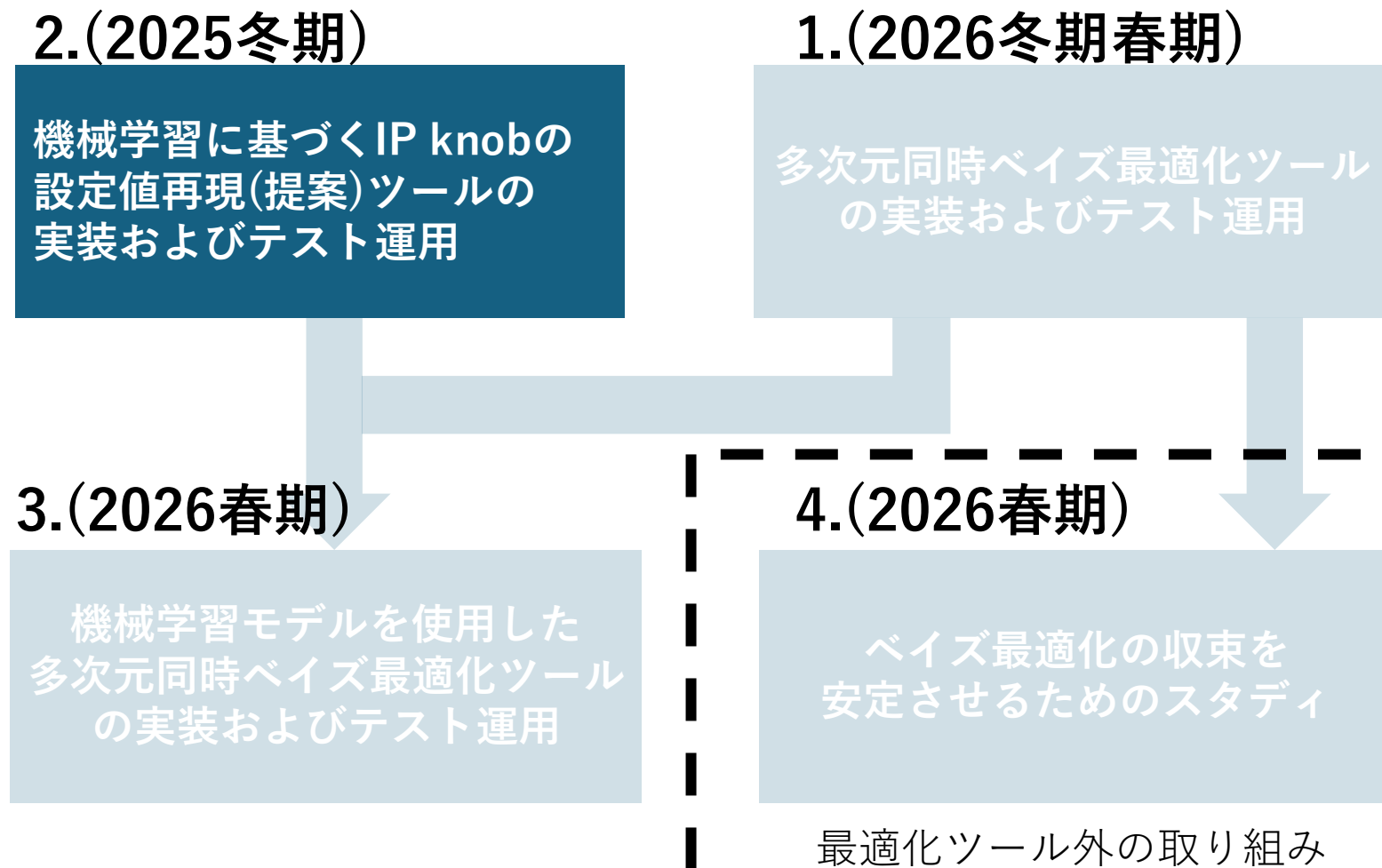
四次元スキャンへの適用結果



- ・ 物理データ取得中に二次元と四次元の同時最適化を実施
- ・ 加速器の状況に左右されるが、ビームを不安定にすることもなくスキャンが完了することも確認

本ツールによる多次元同時ベイズ最適化は四次元まで動作することを確認

2.(Set)機械学習に基づくIP knobの設定値再現(提案)ツールの実装およびテスト運用



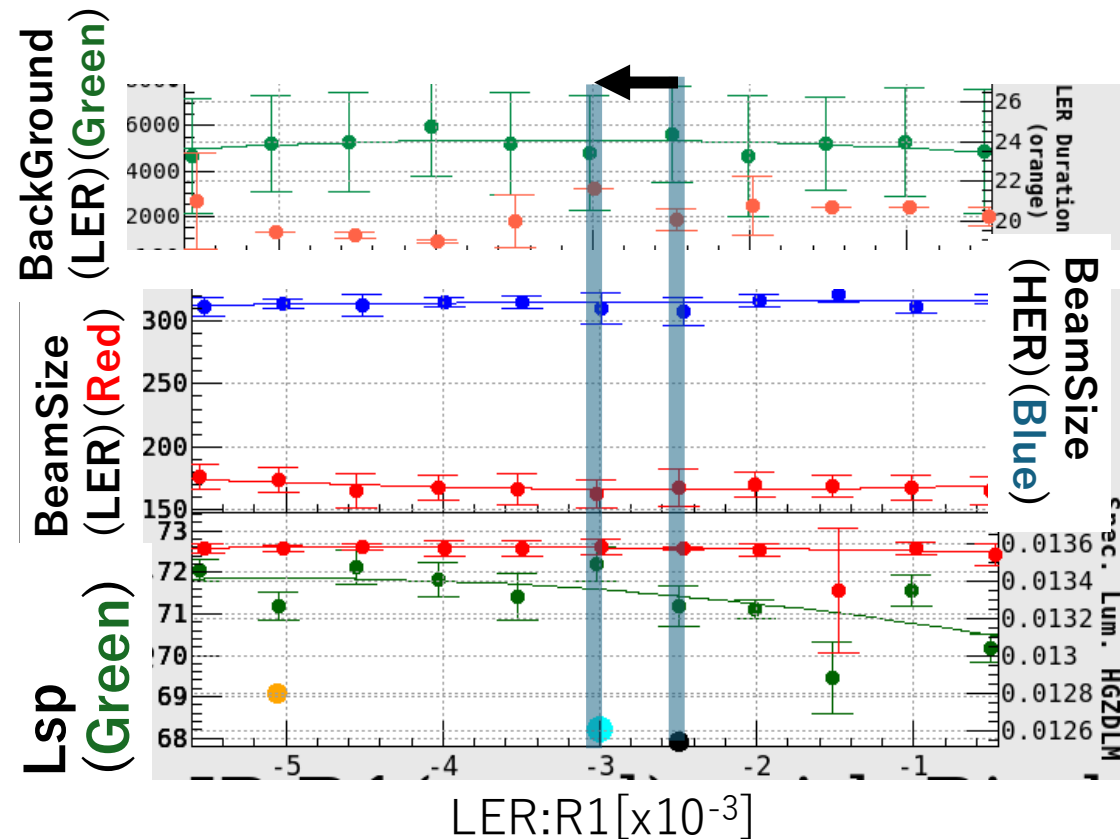
2.(Set)機械学習に基づくIP knobの設定値再現(提案)ツールの実装およびテスト運用 本スタディのモチベーション

開発目的

- ・ 機械学習の効果の確認
 - ー 人間のオペレーションを再現できることを確認する
 - ー オペレーターはLuminosityやBeam Size、Beam BackGround等を総合的にみてsetする値を判断
 - ー 機械学習を用いて過去のオペレーターの判断を再現

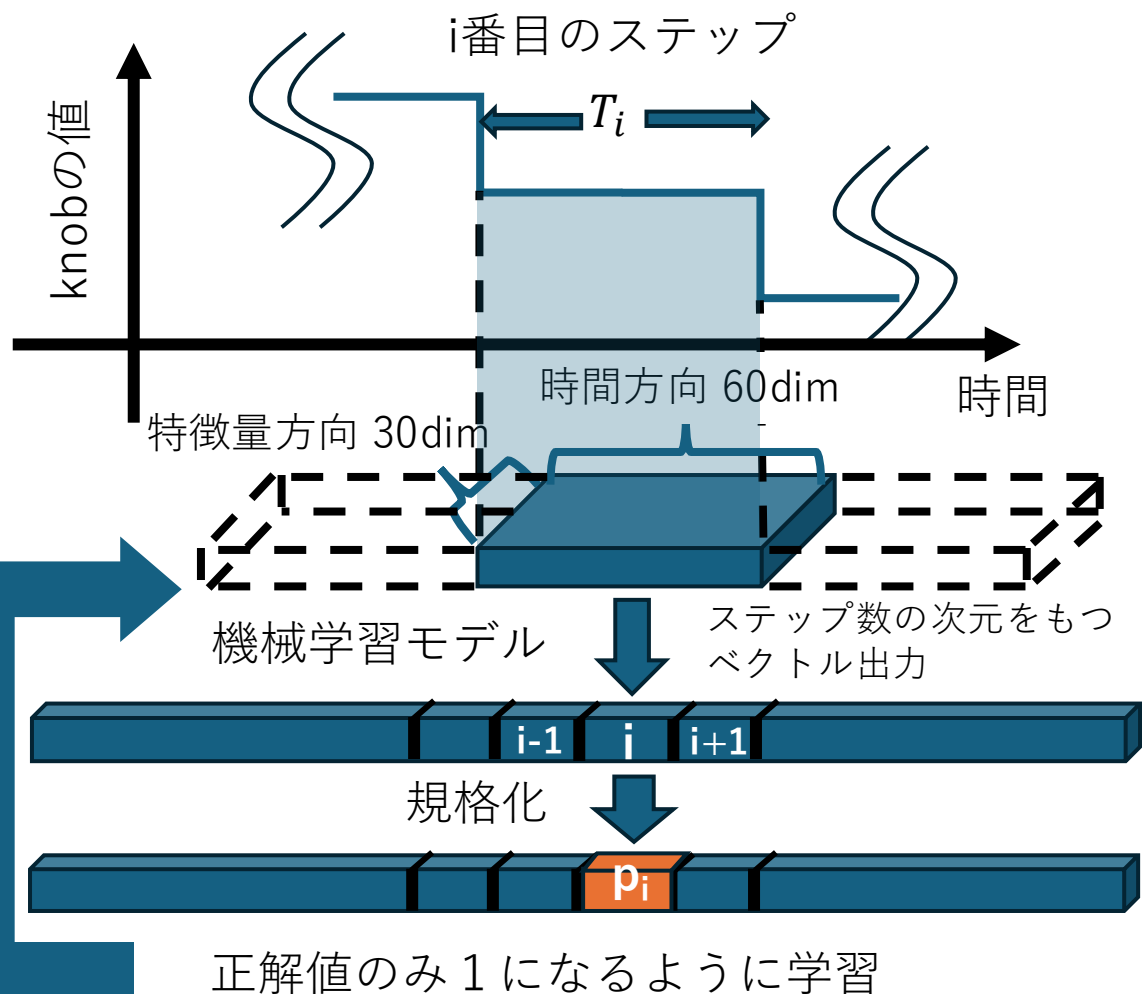
要求性能

- ・ 過去のスキャンを見てオペレーターの選択値から $\pm 2\text{step}$ 以内の設定予測値を出力すること



黒点が元値、水色点が設定値

2.(Set)機械学習に基づくIP knobの設定値再現(提案)ツールの実装およびテスト運用 本研究における機械学習のフォーマットと学習データ



- (scan+set)の1セットが1つの学習データ
- 過去のオペレーターの選択値を正解値として学習

入力特徴量：

$\{\text{IP knob; Lsp, BackGround, BeamSize, inj eff}\}_i$

添字 i は i 番目のステップでの測定値を示す

$(1 \leq i \leq (\text{そのスキャンにおけるステップ数})N_{steps})$

出力：

スキャン範囲内のある一つのknobの値

機械学習モデル $[30 \times 60 (\text{特徴量} \times \text{時間}) \rightarrow 1 (\text{評価指標})]$ ：

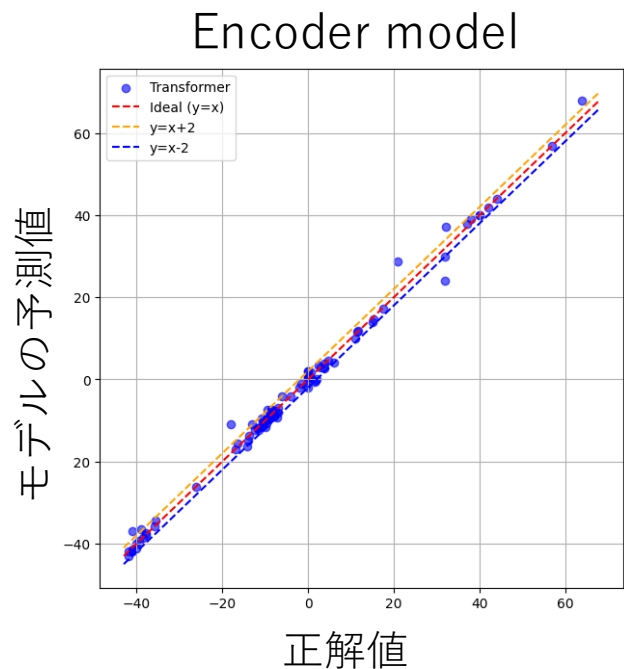
- NNモデル (全結合3層)
- CNNモデル (畳み込み2層 + 全結合3層)
- encoderモデル (encoder2層 + 全結合2層)

これらのモデルを試して比較した

学習データ：

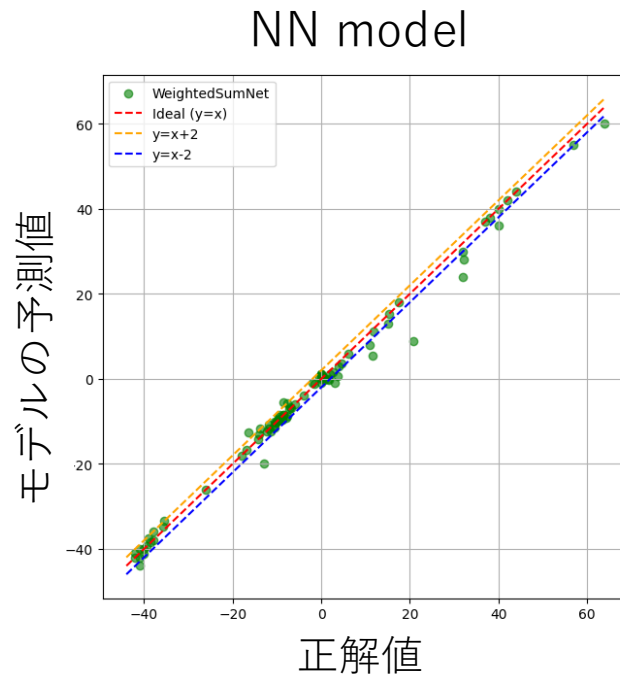
2024年、2025年の運転中のIP knob tuning全て($\sim 2k$)

2.(Set)機械学習に基づくIP knobの設定値再現(提案)ツールの実装およびテスト運用 各機械学習モデルにおけるテストデータへの適応結果



Within 1step: 72.73% , count:64

Within 2step: 89.77% , count:79



Within 1step: 72.73% , count:64

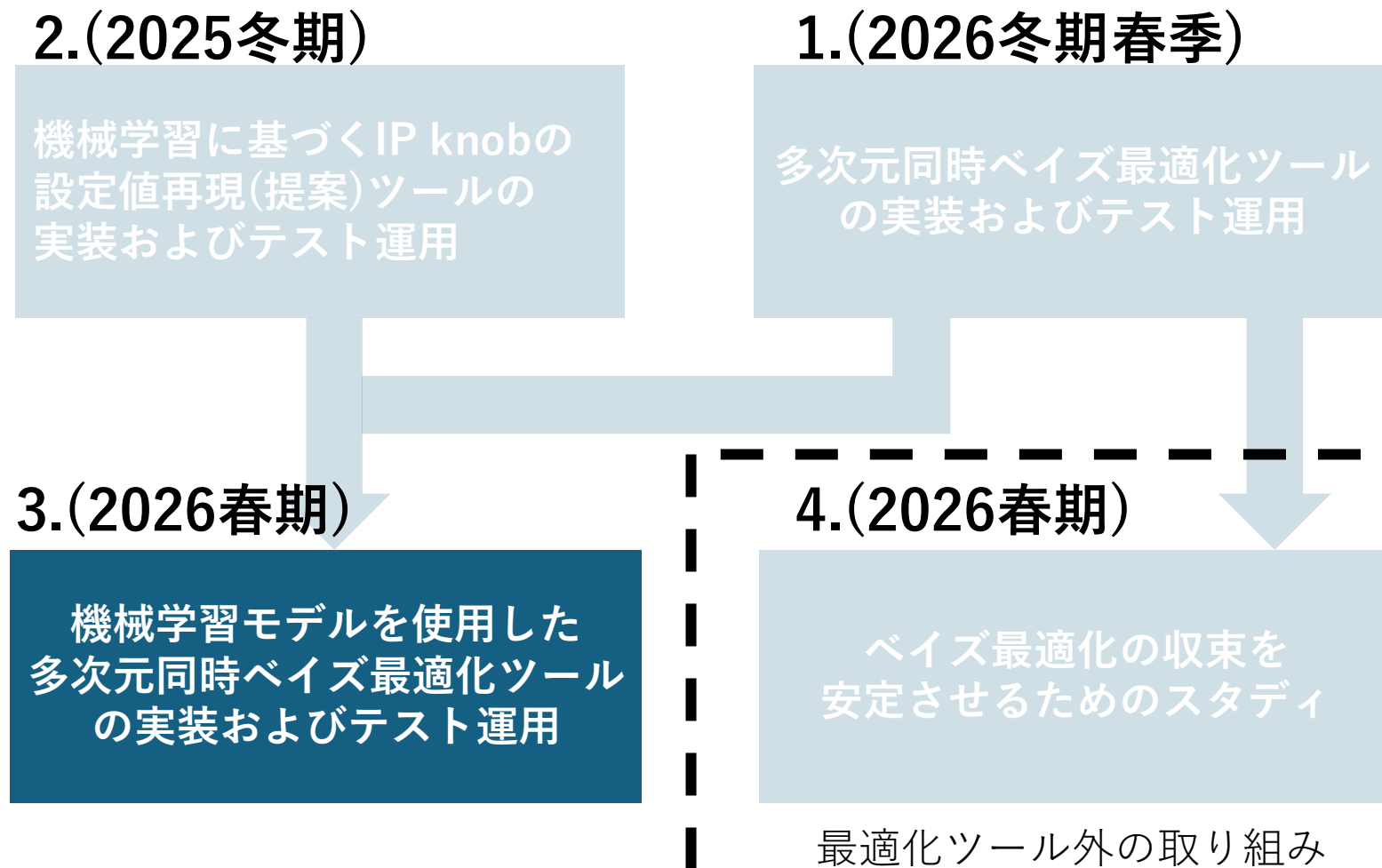
Within 2step: 84.09% , count:74

テストデータに対する
機械学習モデルの適応結果
(IP knobの値は規格化後の値)

- NNとencoderはほぼ性能が変わらない
- CNNは正常に学習が進まなかった
- NNは平均的な正解率が若干高く、外れ値が多い
- encoderは外れ値が少ない

オペレーターの選択値から ± 2 step以内
であるものが8~9割であり、
機械学習モデルは過去のオペレーターの
判断を再現可能と判断した

3.(Scan+Set)機械学習モデルを使用した多次元同時ベイズ最適化ツールの実装およびテスト運用



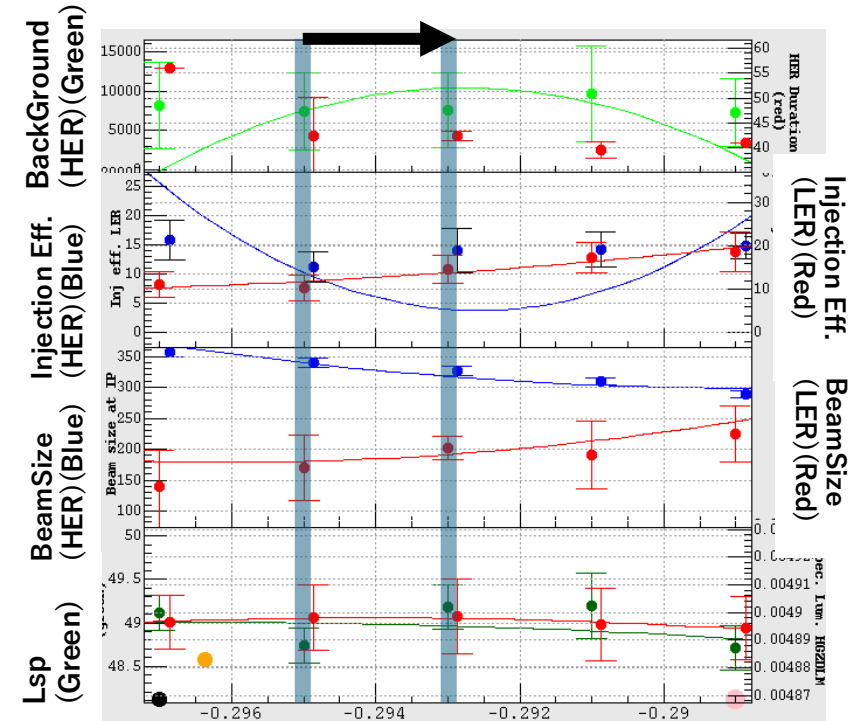
3.(Scan+Set)機械学習モデルを使用した多次元同時ベイズ最適化ツールの実装およびテスト運用 本スタディのモチベーション

IP knobの用途

{ Luminosity増大
Beam BackGround抑制
Beam Life増大
など,,,

加速器のその時の状態に合った調整を行う
(基本的にはLuminosity増大)
(右のスキャンは入射効率やBackGroundを見ている)

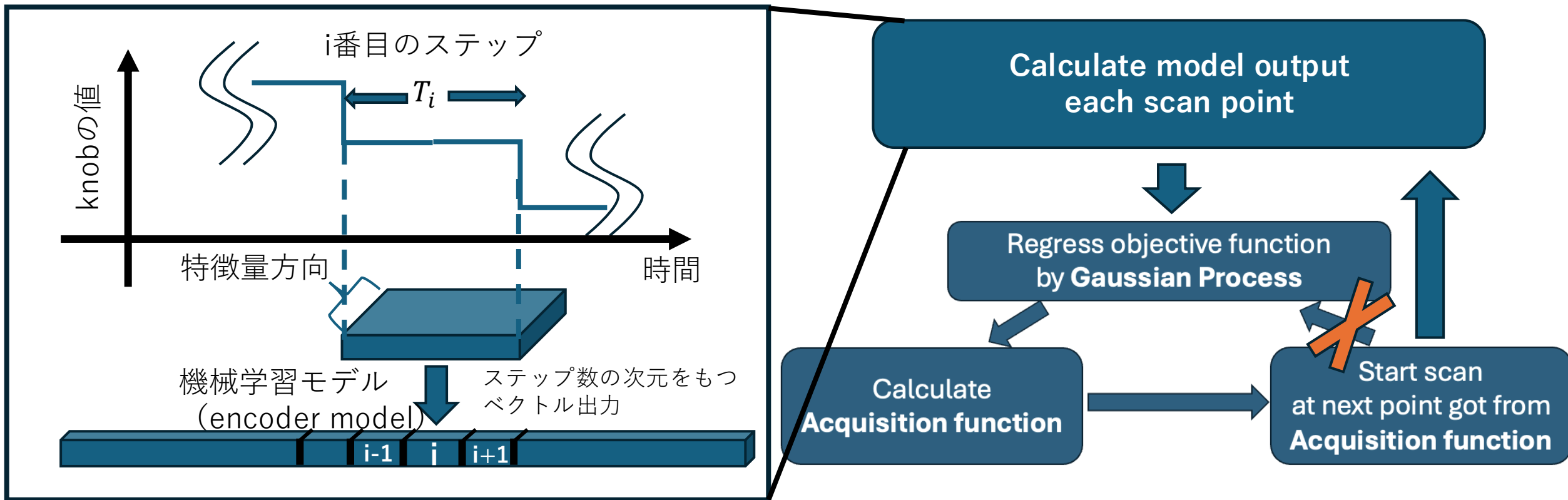
今まで説明してきた最適化ツールは
Luminosity増大の時にしか使用できない



衝突点軌道に関連するIP knob(-0.295→-0.293)

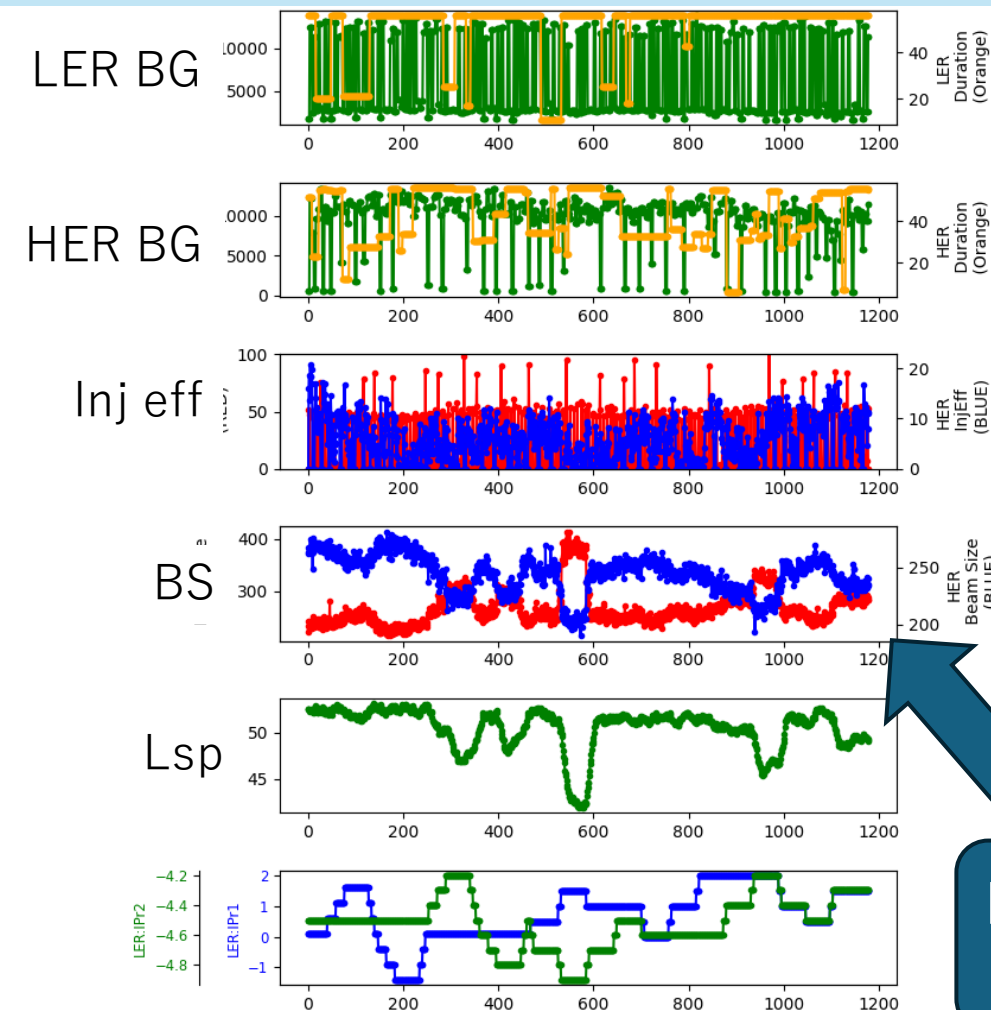
Lsp以外のパラメータも見、今の加速器にとってIP knobを
どこに設定するべきかを自動で判断できるような枠組みを作る

3.(Scan+Set)機械学習モデルを使用した多次元同時ベイズ最適化ツールの実装およびテスト運用 本モデルの構造

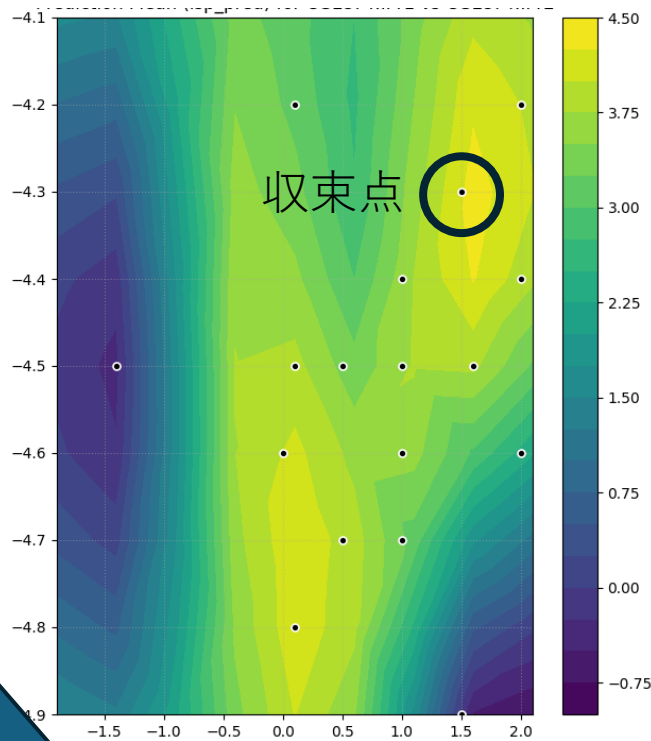


- ・使用した機械学習モデルは1.で学習させたencoder model（学習データは2024年と2025年）
ー外れ値が少ない方が安全上の理由で有利

3.(Scan+Set)機械学習モデルを使用した多次元同時ベイズ最適化ツールの実装およびテスト運用 二次元スキャンでの適用結果

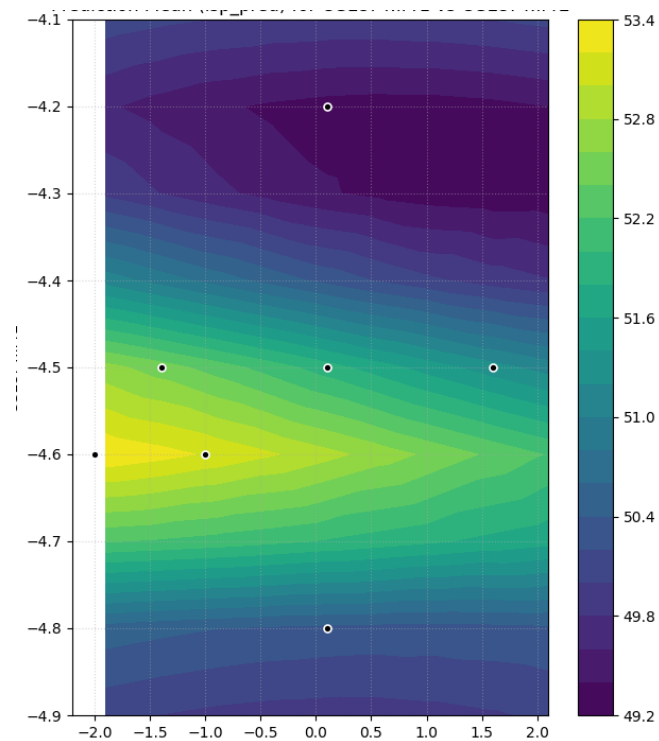


モデル出力をガウス過程回帰



Lspに関わらずBeamSizeを両方
縮小させようとする傾向がある

Lspをガウス過程回帰



色：ガウス過程回帰による μ
横軸：LER側R1、縦軸：LER側R2

ここまでのスタディでの課題

ツール自体は動くようなものが出来たが、課題が大きく分けて二つ

①. (Scan) スキャン収束に関する課題

- ・ スキャンに関わらず存在するLsp等の各種パラメータの数%のふらつき
(二時間程度にわたるふらつきなども見つかっており、
スキャン結果がこれに引っ張られてなかなか収束しない。)

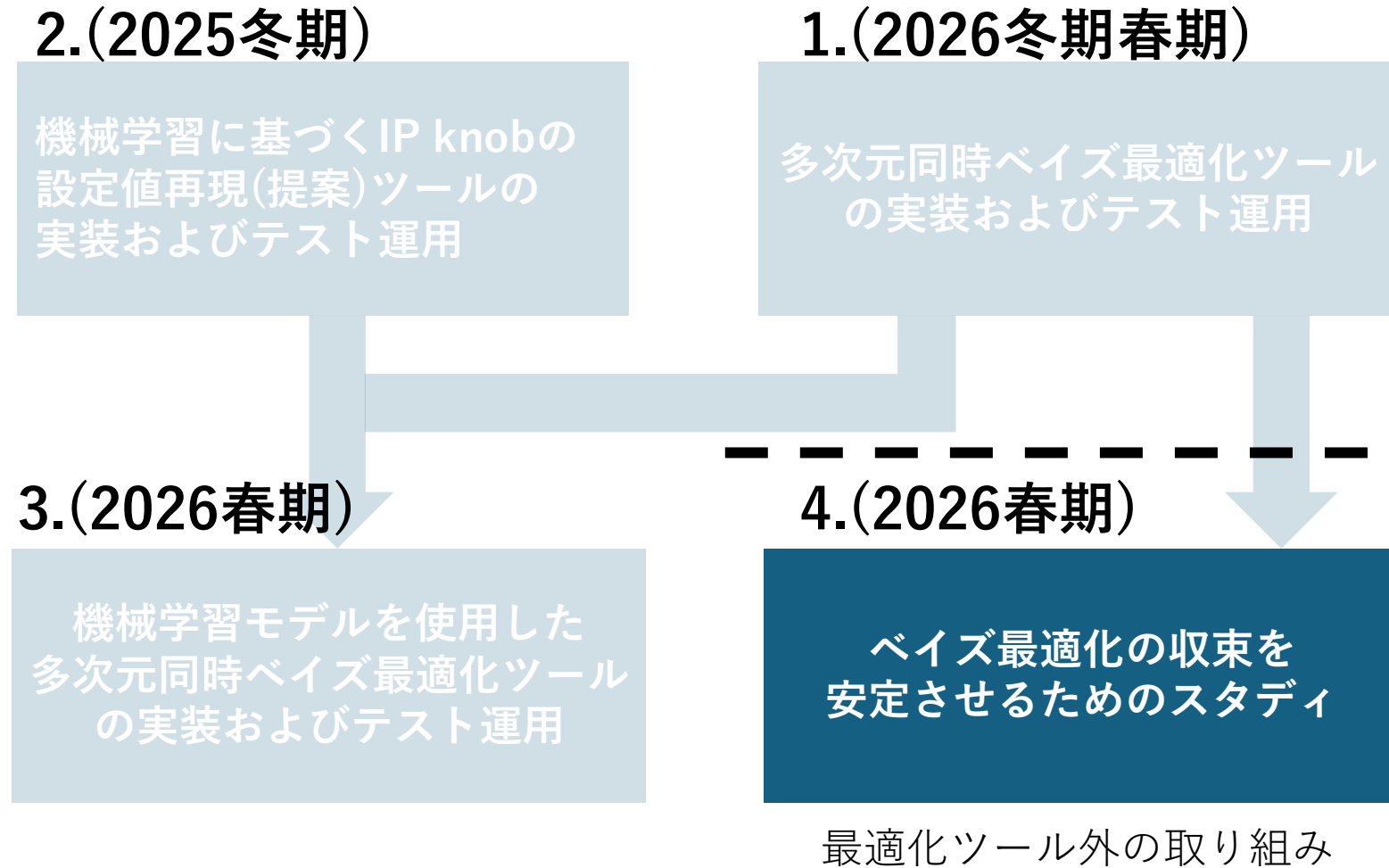
②. (Set) 最適値選択における課題

- ・ データ数（スキャン数）が少ないことによる機械学習の難しさ

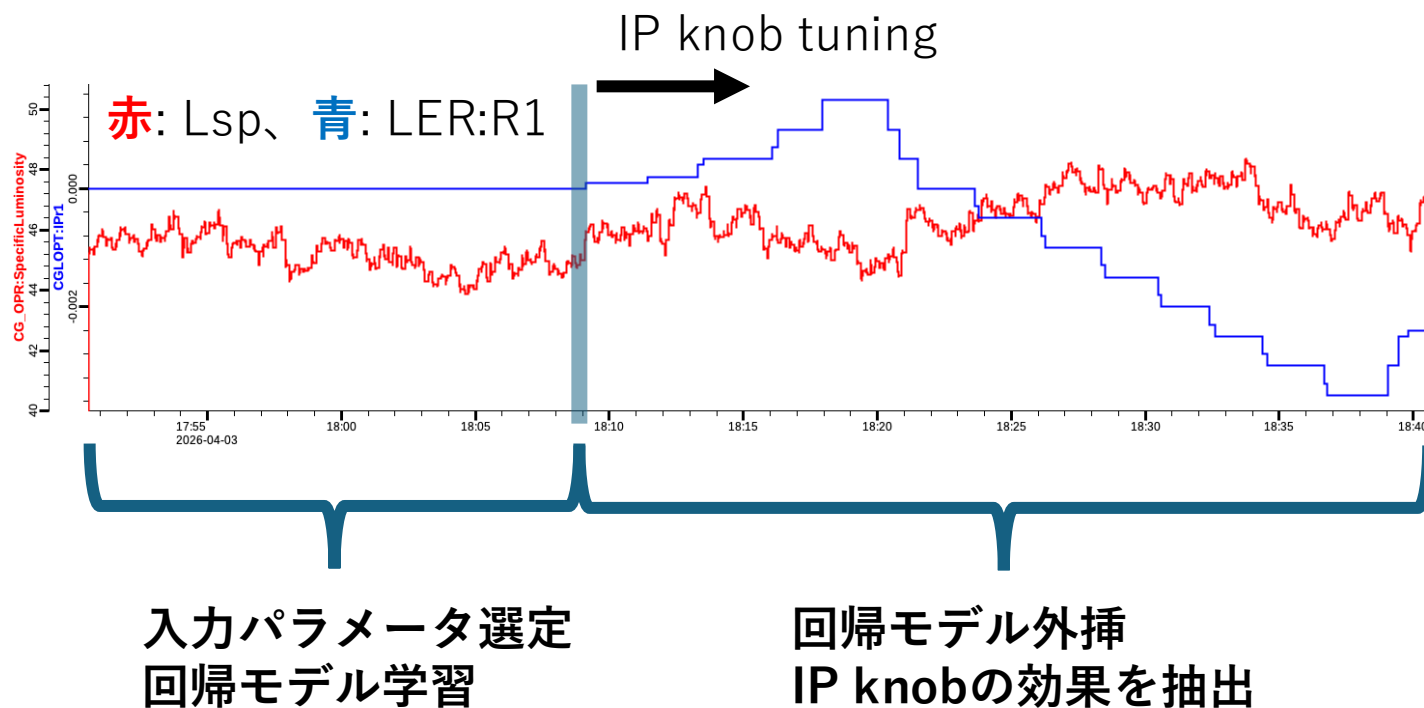
これらはツールの問題もあるが加速器側の問題でもあるので対処が難しい

①.により多次元へスキャンを拡張したくとも出来ないため、
運転期間中に①.に対してツール外で対処するような取り組みを行った

4.ベイズ最適化の収束を安定させるためのスタディ



4. ベイズ最適化の収束を安定させるためのスタディ 外的要因によるふらつきを排除するためのLsp外挿モデルの構築



入力パラメータ選定

- ・ IP knob tuning外でLspと高い相関を持つパラメータを30個ほど選定

回帰モデル学習

- ・ 選定されたパラメータを用いてLspを線形回帰(リッジ回帰)

回帰モデル外挿

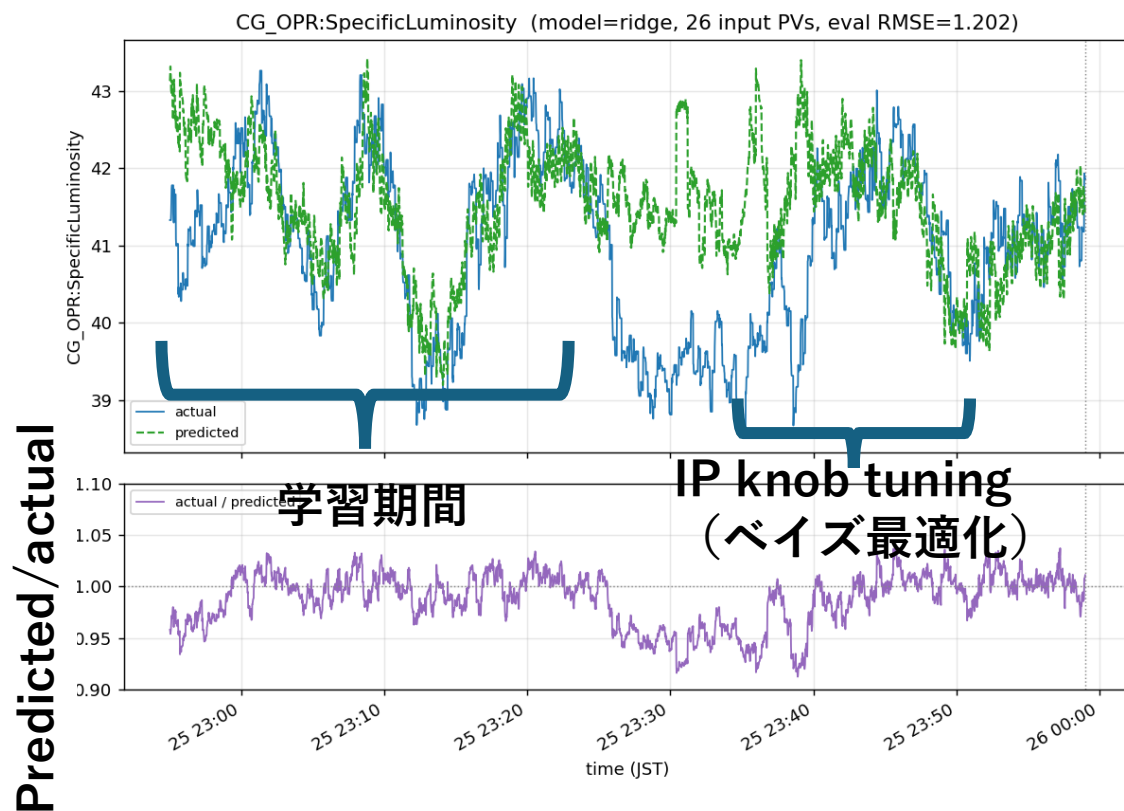
- ・ IP knob tuning中に学習したモデルを使って予測Lspを出力

IP knobの効果抽出

- ・ 予測Lspと実際のLspの比からIP knobの影響を抽出

- ・ 回帰モデルはIP knobを動かしたらどうなるかを知らないのでIP knobを動かしてもIP knobを動かしていなかった時のLspを出力してくれると期待する

4.ベイズ最適化の収束を安定させるためのスタディ モデルの学習と適用結果



- 30分間調整を止めてもらい、そのデータで学習を行なった
- IP knobを動かす前に予測Lspと実際のLspの比較を行ったが $\pm 5\%$ 程度の誤差が存在
- actual/predictedを用いてベイズ最適化を行った
- Lsp最大値では最適化が止まらず、Lspが落ちるようなところで最適化が終了

30分という限られた量の学習データでは精度のよいモデルを構築することは難しく、他の調整期間中でも学習できるようなモデルの改良や入力パラメータの精査が必要

まとめと展望

2025年冬期および2026年冬期春期運転で行ったこと

(自動化の足掛かりとして)

- ・ 多次元同時ベイズ最適化ツールの実装およびテスト運用
- ・ 機械学習に基づく IP knob の設定値再現(提案)ツールの実装およびテスト運用

(Lsp以外のパラメータや柔軟な調整を再現するために)

- ・ 機械学習モデルを使用した多次元同時ベイズ最適化ツールの実装およびテスト運用

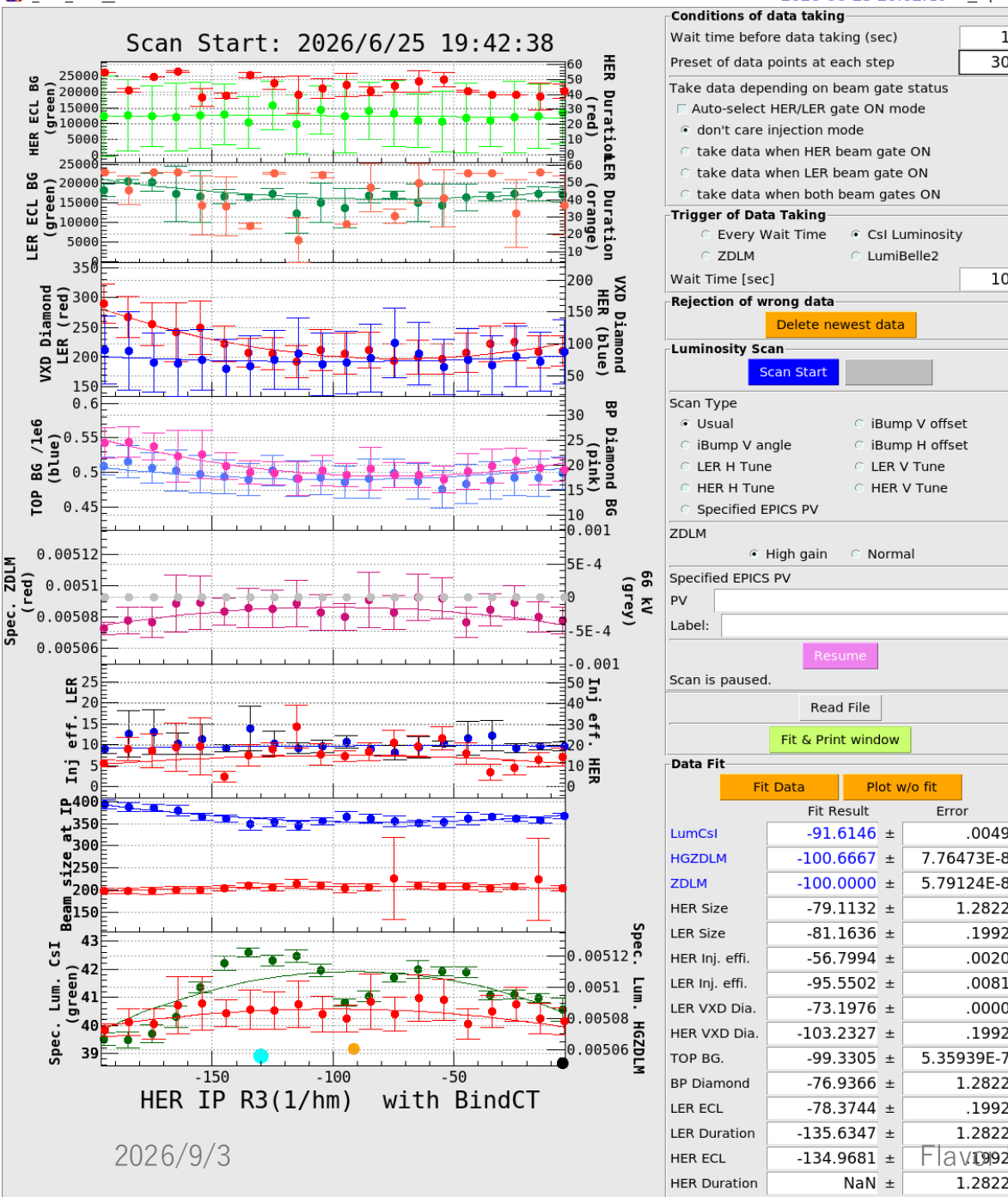
(Lspのふらつきが大きい時にベイズ最適化が収束しないという問題を解決するために)

- ・ ベイズ最適化の収束を安定させるためのスタディ

今後の展望

- ・ 各機械学習モデルの見直し (モデル再構築など)
- ・ ルミノシティ回帰に用いるパラメータの精査

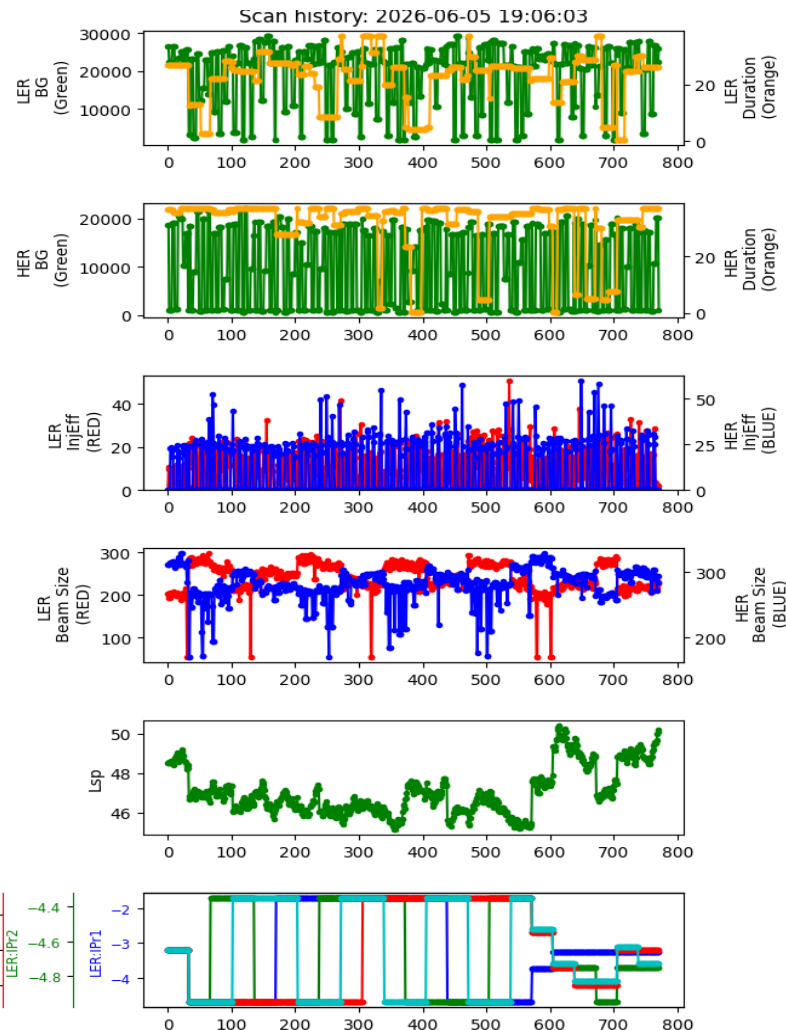
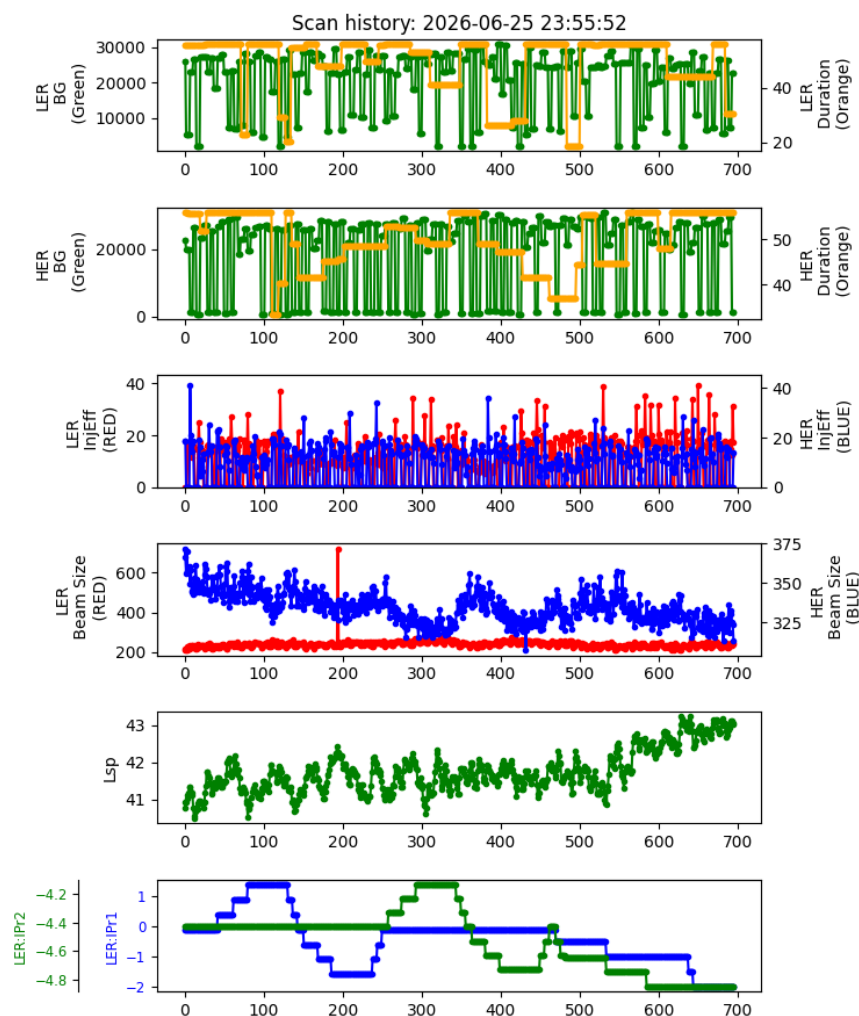
Back up



局所的なLsp最大値が顕著な例

この時はBelle II 測定器がOFFであったので、
これほど広い範囲でスキャンしているが
本来ならこの半分程度のスキャン範囲なので
二つ目のpeakに気づくことができない可能性が高い

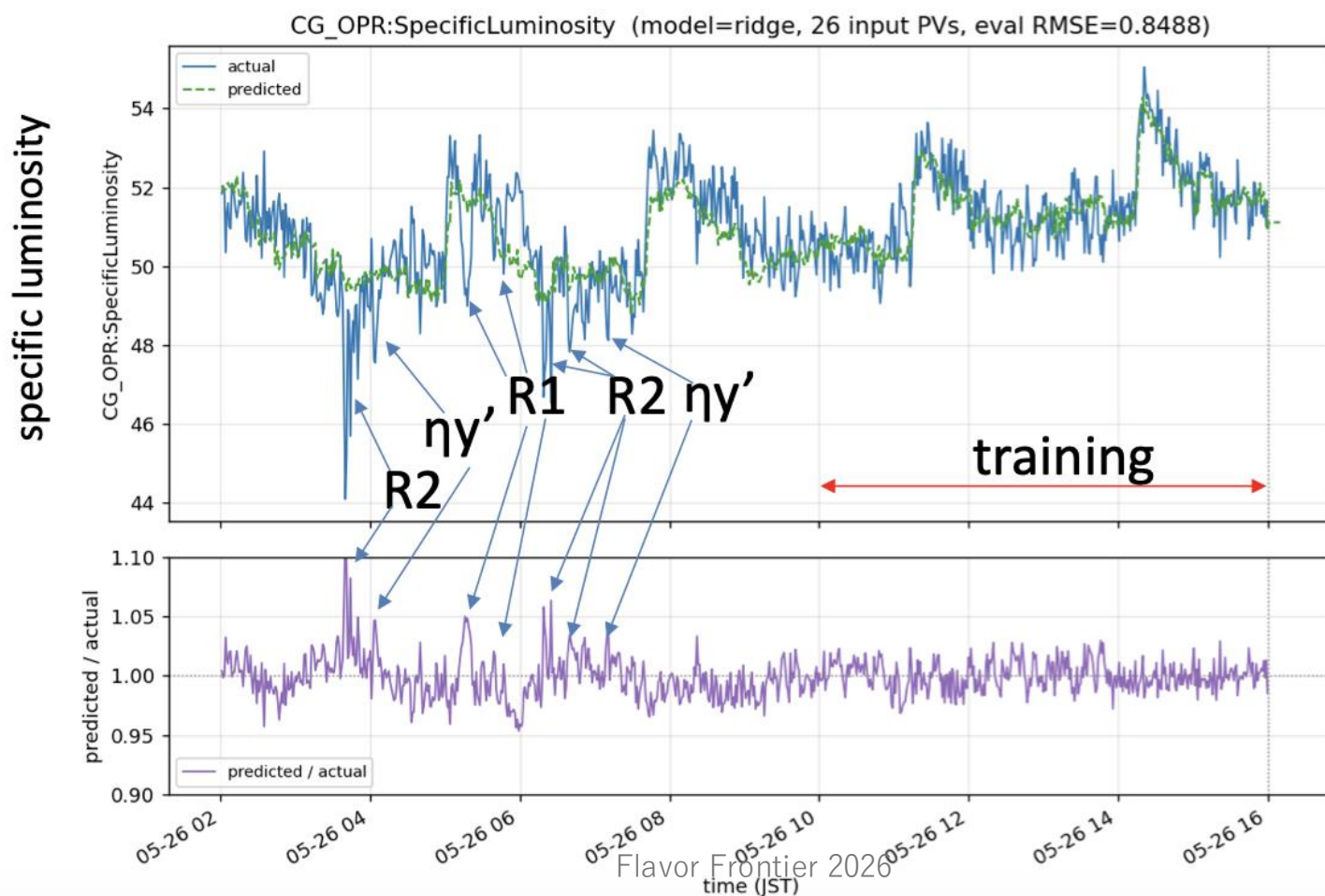
Lspベースのスキャン結果 BGやBS,inj effも含む

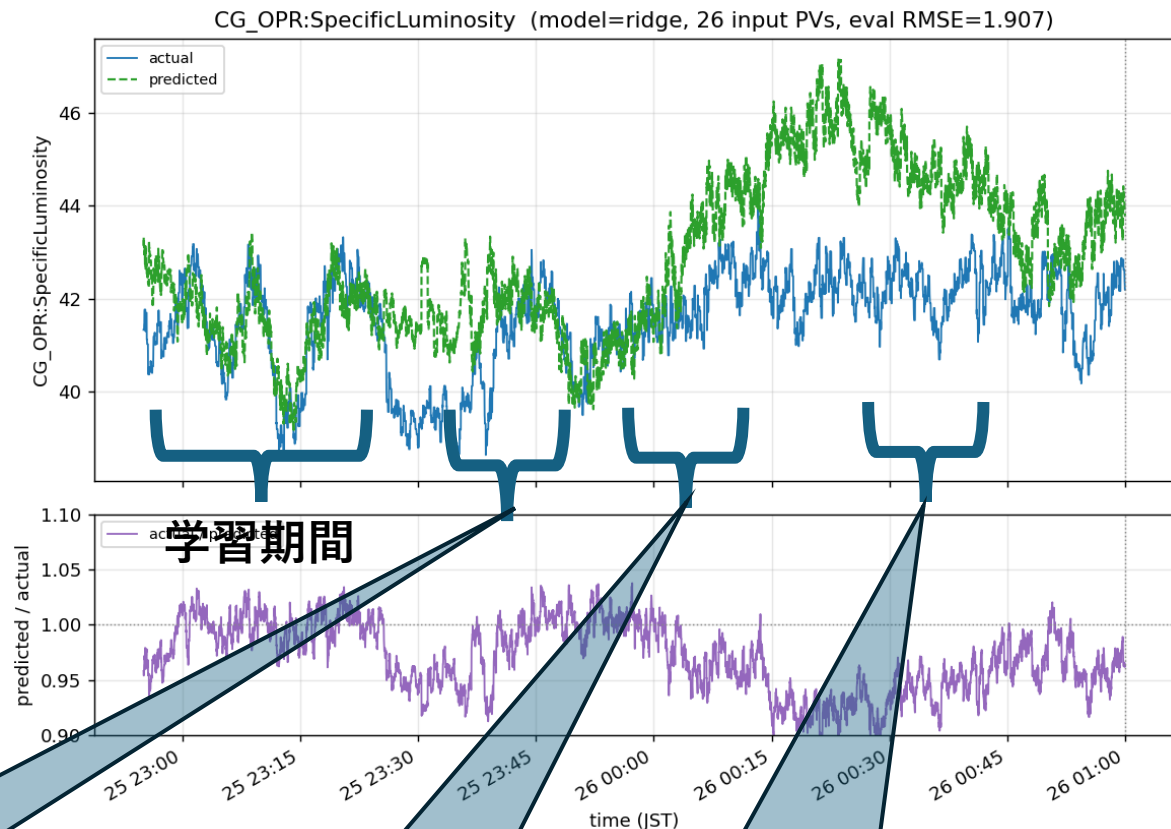
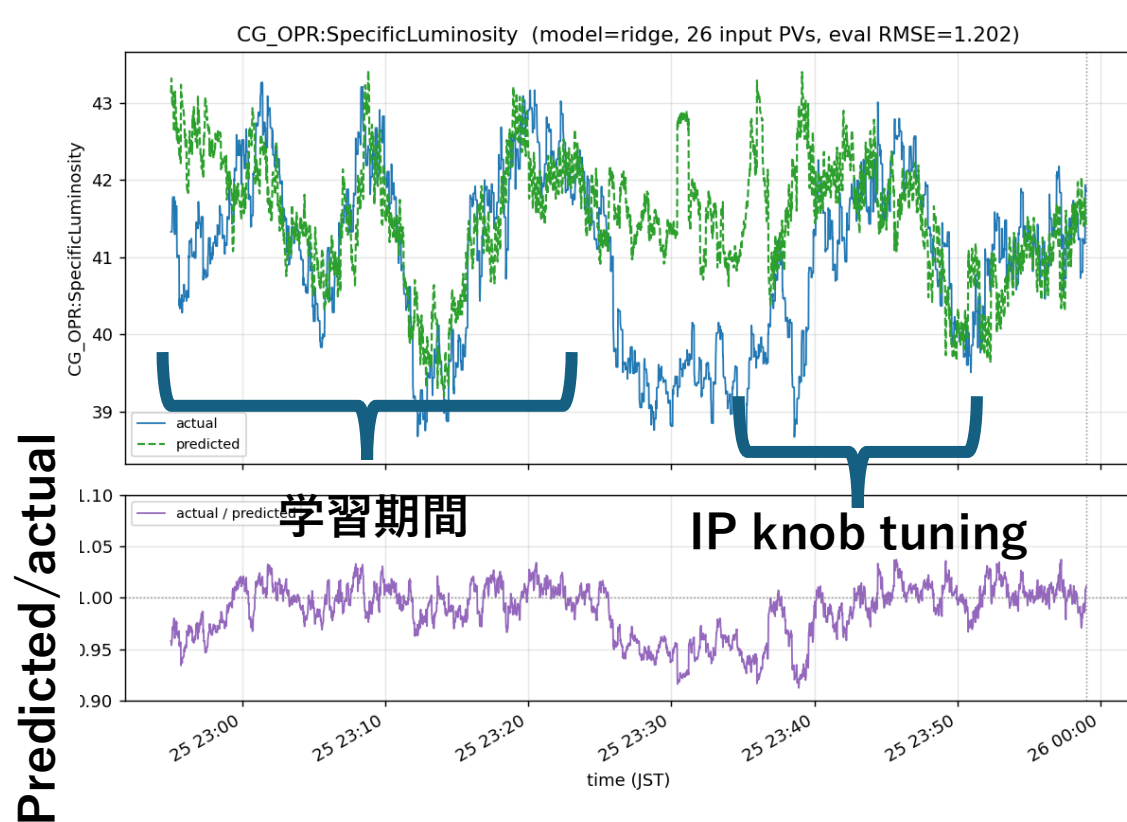


一行目：
LER BackGround(緑)
二行目：
HER BackGround(緑)
三行目：
Injection efficiency
HER(青)LER(赤)
四行目：
Beam Size
HER(青)LER(赤)
五行目：
Lsp
六行目：
IP knobの値

Training and prediction

- Different input parameters are used for comparison
- beam current + Fuji temperature + Nakazawa-san: **better prediction**
- effect from knob scan become clear**





- ・ 学習直後から(pre/actualでの)乖離が発生
- ・ 補正Lsp使用スキャンにより乖離はなくなる (Lspは落ちる) (手動補正後も一致したまま)
- ・ Lsp使用スキャンでLspは上がるが悪い方向への乖離が広がる (回帰入力パラメータがknobの影響を受けていることを示唆)
- ・ 再度補正Lsp使用スキャンで乖離はなくなる方向へ (Lspはほぼ変化していないように見える)
- ・ knobを変えてpredictのみが変化するのは意図した挙動ではない (回帰入力パラメータがknobの影響を受けているように見える)

- (多くとも) 10k程度数時間の相関を計算というのを一時間くらいずつずらしてやる
- 順位変動等でグラフが書ける？ $((\text{一個前の順位}) - (\text{次の順位})) / (\text{高い方の順位})$ とかで
- 相関のリストから消えたやつとかは順位が(考えているpv数)+1の順位になったとして計算
- 新しく相関のリストから入ったやつとかは元の順位を(考えているpv数)+1として

→ これにより短期的にたまたま相関しているやつとかがわかる

→ あまりにも激しいようならその後の回帰には使わない

