Automatic Setup of Controlled Longitudinal Emittance Blow-Up

>> Considerations on RL Feasibility <<

N. Bruchon

Outline

1. Introduction to blow-up
    - Importance for longitudinal stability
    - Blow-up settings
2. Purpose of RL application
    - Definition of State, Actions, Reward
    - Critical issues
3. Solutions with other approaches
    - Discussion
4. Conclusions

Outline

1. Introduction to blow-up
    - Importance for longitudinal stability
    - Blow-up settings
2. Purpose of RL application
    - Definition of State, Actions, Reward
    - Critical issues
3. Solutions with other approaches
    - Discussion
4. Conclusions

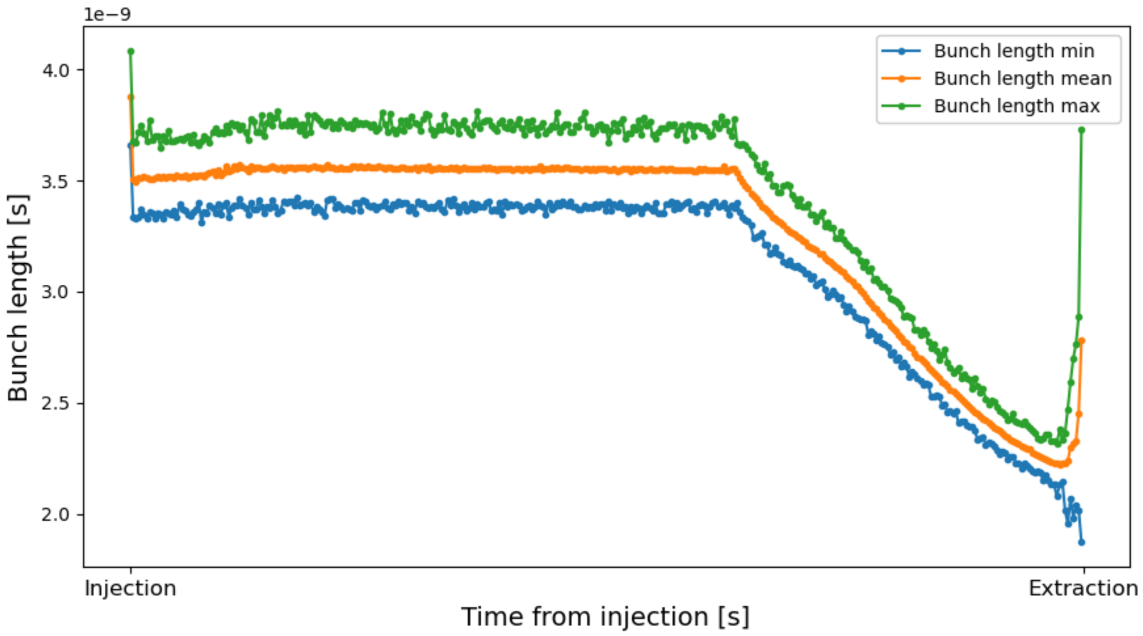
Introduction

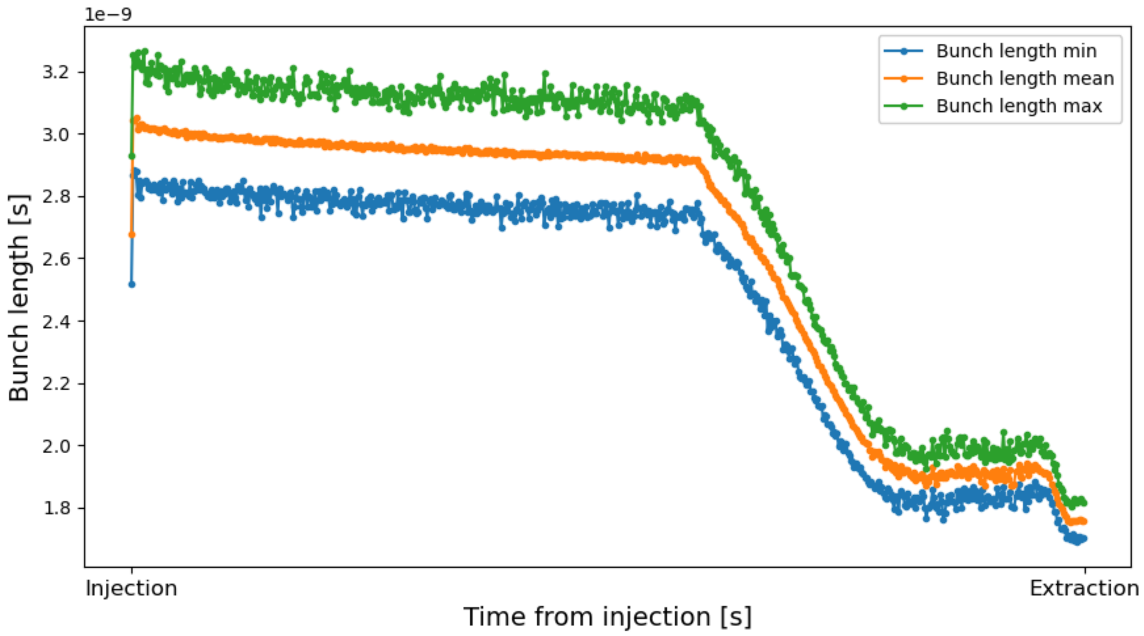
Bunch length shrinks during acceleration

- Smaller bunch length may result in **unstable beam** at extraction

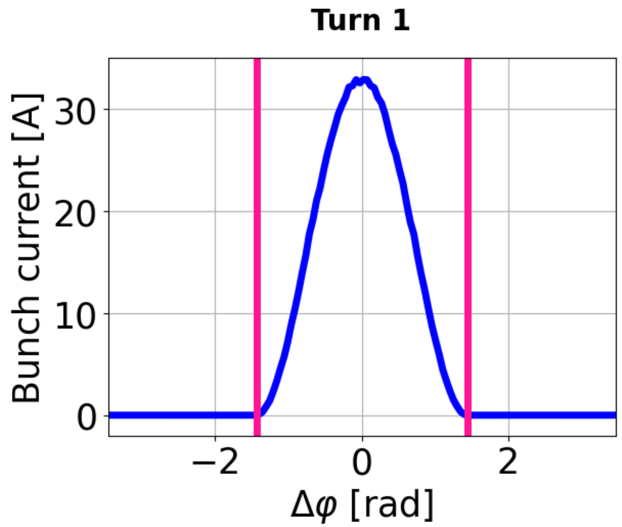
Injection of bandwidth limited noise into the phase loop

- **Purpose:** diffuse particles in the bunch core, without touching the tails
- **Effect:** improve beam stability

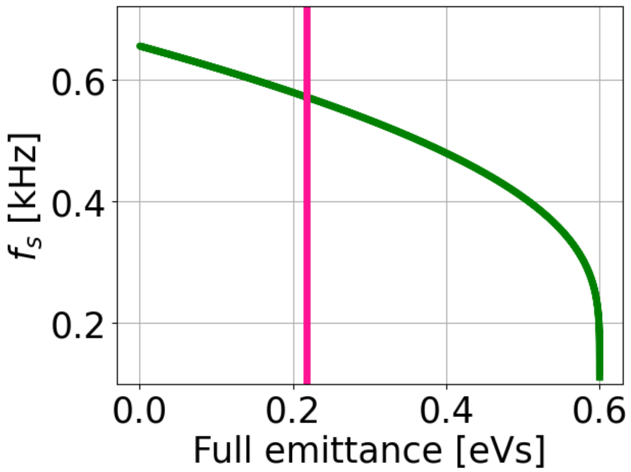




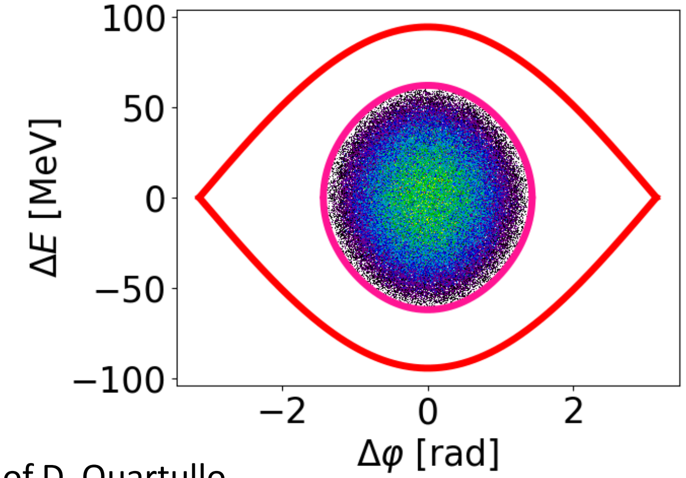
Blow-Up in a Nutshell



Profile



Synchrotron frequency
Emittance
Noise bandwidth



Longitudinal Phase Space

Particles diffuse in the bunch to increase longitudinal emittance:

- Reduced line density peak current
- Increased bunch length

Courtesy of D. Quartullo

Blow-Up Functions to Optimize

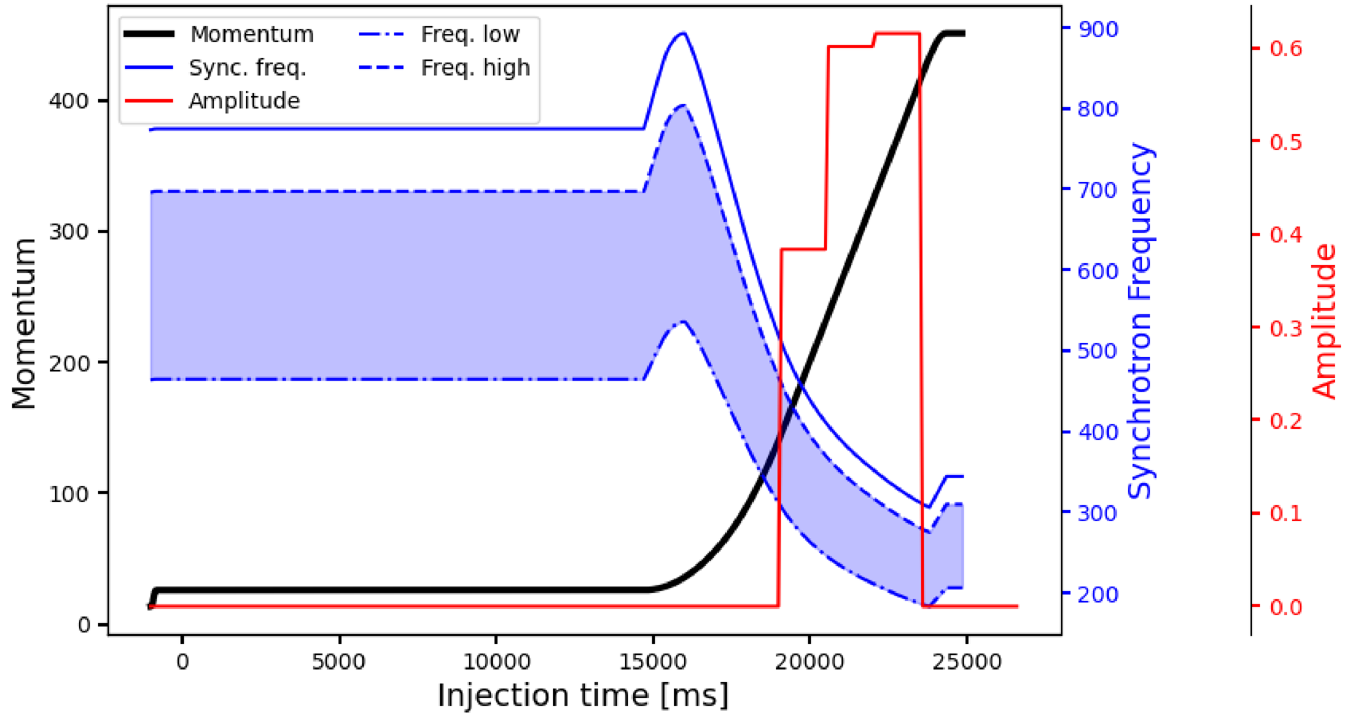
Amplitude, a:

- Defined in rms degrees of the 200 MHz RF system
- Too low-amplitude noise signal may be ineffective
- Too high amplitude can negatively affect the bunch distribution
- Zero amplitude means that no blow-up is applied

Frequency band:

- Defined by cutoff frequencies that follow variation of small-amplitude synchrotron frequency, f_{s0}
- Ratios called margins defined by normalizing those values with respect to f_{s0}
  - **Margin low**, $m_{\text{low}} = {}^{f_{\text{low}}}/_{f_{s0}}$
  - **Margin low**, $m_{\text{high}} = {}^{f_{\text{high}}}/_{f_{s0}}$
- The aim of the blow-up is to impact the bunch core exclusively, without increasing the tail population of the particle distribution, which would risk generating losses

Blow-Up Functions to Optimize



Blow-up interval:

- Time interval along the cycle during which the blow-up is applied
- Defined when RF voltages and synchronous phase are nearly constant
- Limited also by other manipulations

Outline

1. Introduction to blow-up
    - Importance for longitudinal stability
    - Blow-up settings
2. Purpose of RL application
    - Definition of State, Actions, Reward
    - Critical issues
3. Solutions with other approaches
    - Discussion
4. Conclusions

Purpose

Optimal blow-up settings are essential to ensure longitudinal stability

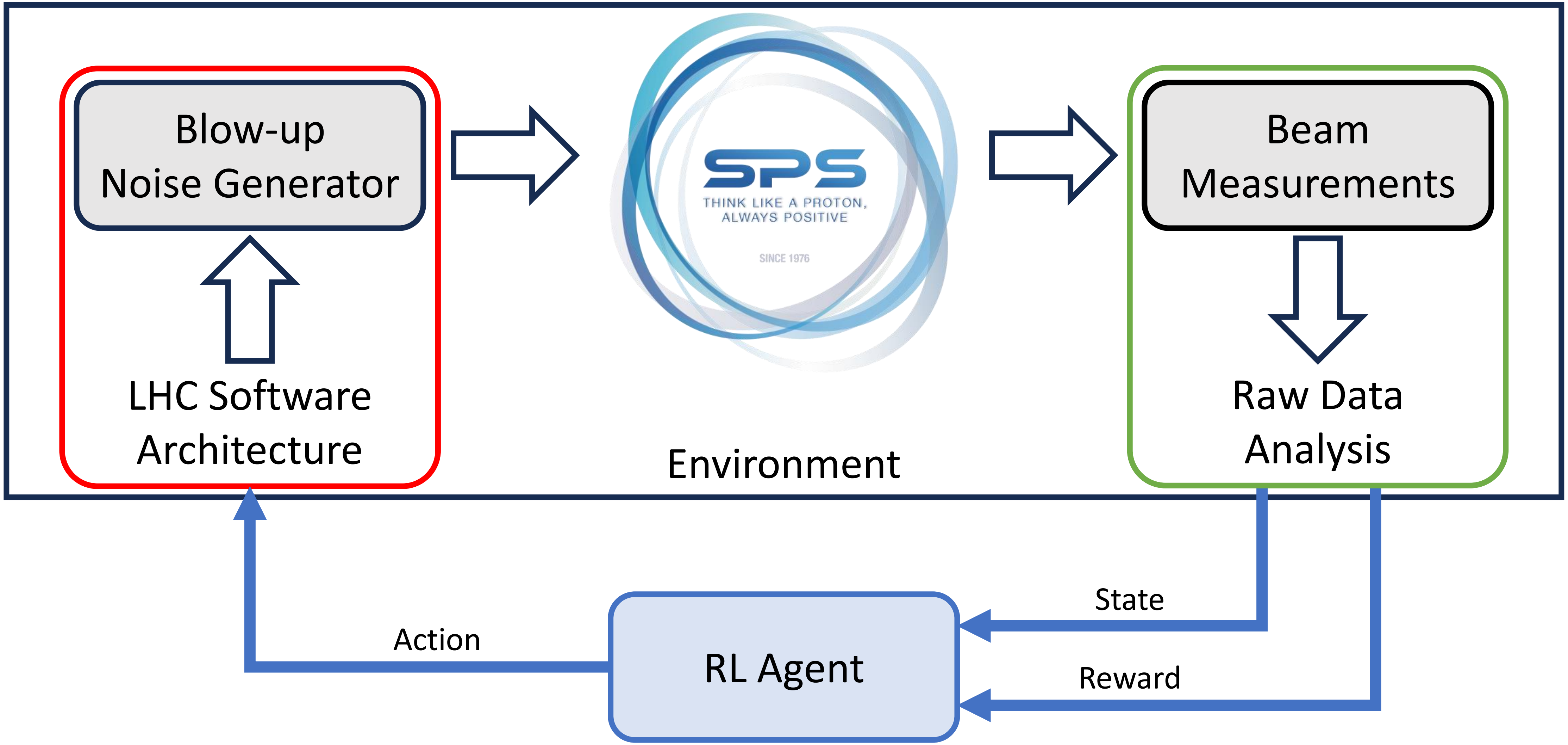
AUTOMATIZE THE SETUP!

- Replace the manual procedure
- Reduce required beam time for optimization
- Results comparability thanks to objective function

Why Reinforcement Learning?

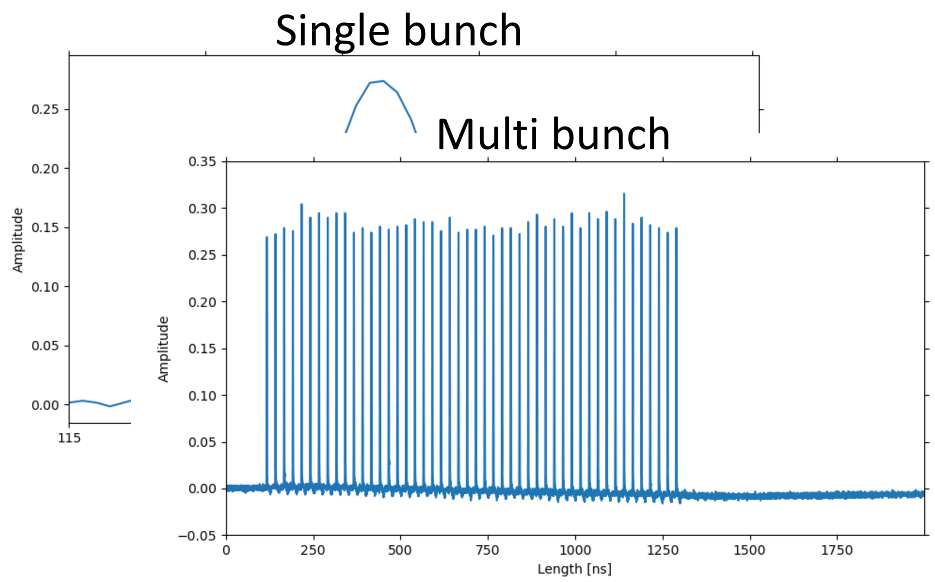
- A trained agent should efficiently solve the problem

Schematic View



Measurements/Observations

Longitudinal profiles from detectors installed in the SPS

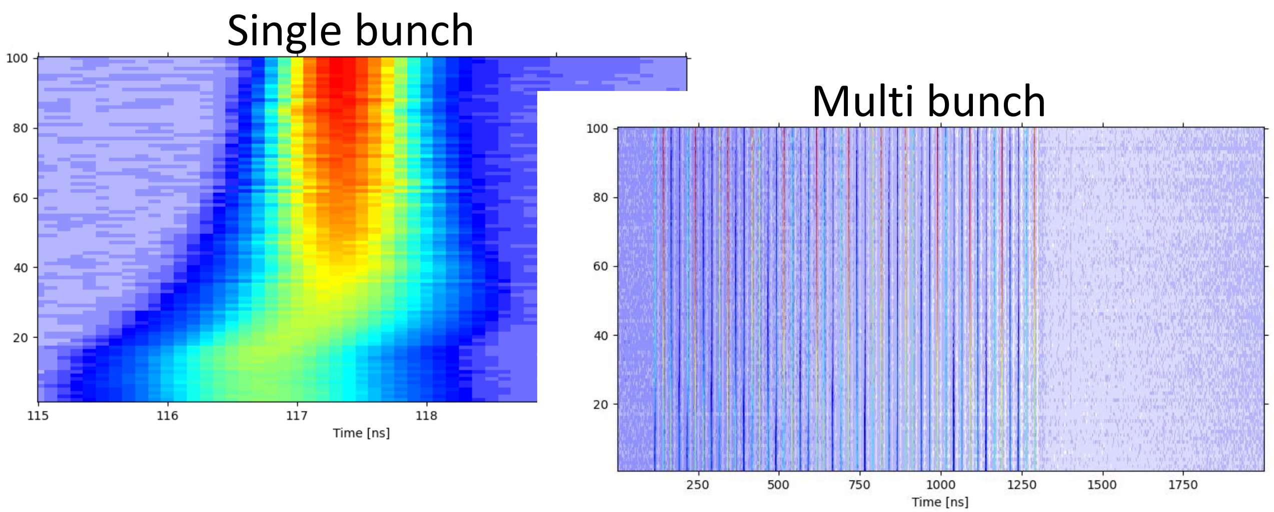


Analysis software to extract features:

- Peak, Width, FWHM, μ, σ, bunch length, …

Multiple bunches → extract features for each bunch

LHC-beam type consists of $4 \times 72 = 288$ bunches
Uniform features are desired across all bunches



Multi turn acquisitions:

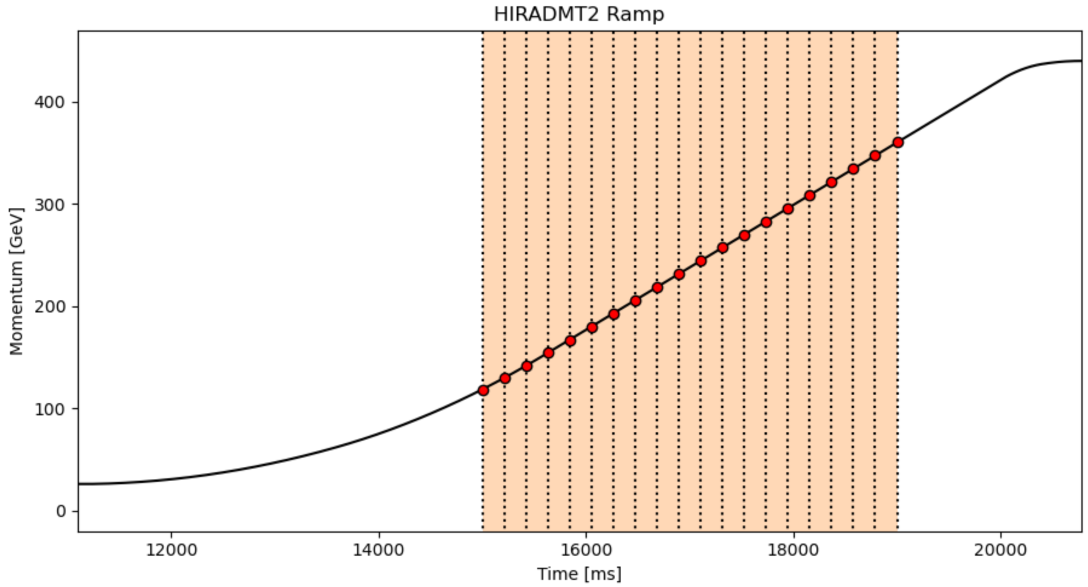
- Data refresh every 20ms

State

Represent environment: raw data and/or extracted features

- Bunch length is the most critical one
- Additional information about bunches may be useful (shape, intensity, …)

Multi turn acquisitions: **when** observation must be taken?



Few observations are sufficient but…

information during blow-up is missing

What about adding **intermediate points**?

Better knowledge and control of the beam aspects taken into account!

WARNING: do not exaggerate with the state dimensions → the problem will become unmanageable soon!

Action

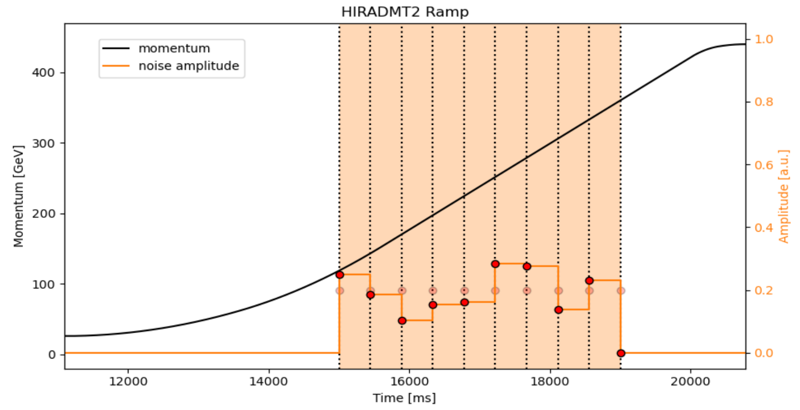
Variation of settings value

Manually: constant settings

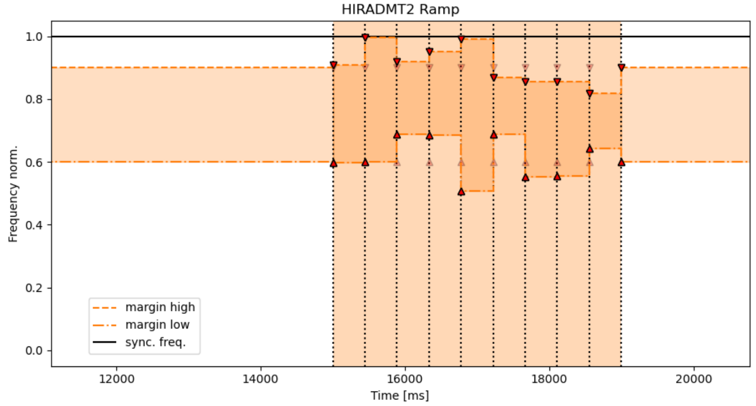
Over multiple points:

- Adapt settings to each interval independently
- More accurate time dependent setup:
  - Amplitude could go to zero in some intervals.
  - Freq. band could increase or reduce in intervals.
- Better control leads to better results
- Limit actions to avoid losses or ineffective blow-up

Variation of the AMPLITUDE



Variation of the MARGINS



Reward

The **reward** must be a **scalar**

Custom function to calculate it relying on observations

Define the most relevant **features** with which calculate the blow-up effects

General targets:

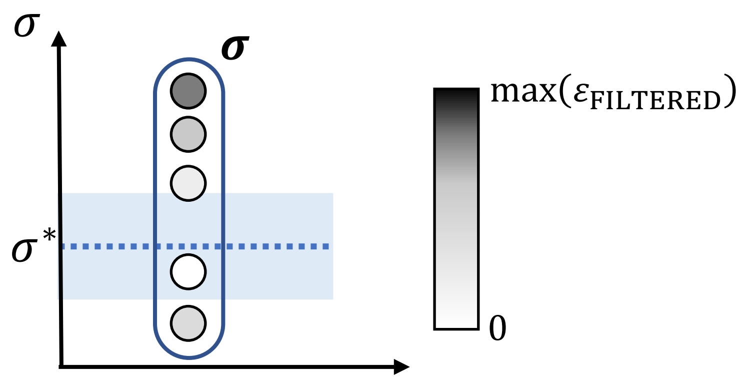
- Reach desired bunch length
- Guarantee longitudinal stability
- Ensure uniform bunches
- Control the bunch length evolution.
- Avoid losses

Bunch Length RMSE – Reward example

Multi bunch beam → n bunch lengths: $\boldsymbol{\sigma} = [\sigma_1, \sigma_2, \dots, \sigma_\mathrm{n}]^T$

- Error vector: $\varepsilon = [\sigma_\mathrm{i} - \sigma^* \quad \forall \mathrm{i} \in \{1, \dots, n\}]^T$
  - Desired bunch length: σ^*, constant for all sub-interval observations
- Filtered error vector: $\varepsilon_{FILTERED} = \begin{cases} 0 & \text{if } \varepsilon_\mathrm{i} \in tol_\mathrm{band} \\ \varepsilon_\mathrm{i} & \text{otherwise} \end{cases}, \quad \forall \mathrm{i} \in \{1, \dots, n\}$
  - With tolerance band defined by $\sigma^* \mp tol_\%$: $tol_\mathrm{band} = [\,\sigma^* - tol_\%,\ \sigma^* + tol_\%]$

$$\text{Root Mean Square Error (RMSE)} = \sqrt{\sum_{\mathrm{i}=1}^{n} \frac{(\varepsilon_\mathrm{FILTERED})^2}{n}}$$



$$\text{Reward} = -\text{RMSE}$$

Critical Issues

Training an RL agent requires data:

- from the **real system**
- from a **simulator**

1. Data efficiency is mandatory:
   - Training should rely on as little data as possible
2. Data collection should not take long time:
   - Fast creation of dataset

Real World Data

Single LHC cycle ($\sim 23\ s$) in the SPS super-cycle ($\sim 45\ s$)

State is collected at the end of the cycle

- Single sample every $\sim 45\ s$

100 samples → ~ 1.25 hours

without considering eventual issues along the acceleration chain

Relying on real data requires too much machine time → **Unfeasible!**

Simulated Data

Beam Longitudinal Dynamics (BLonD):

- CERN simulation suite for longitudinal particle tracking in synchrotrons.

BLonD

It generates the blow-up effects on the longitudinal profiles:

→"raw data" to analyse

<u>It can replace the real machine and provide data according to blow-up settings</u>

Issues on the calculation time!

- Single cycle with uniform settings >8 hours
- Grid search (16065 simulations in parallel) with uniform settings ~ 3 days

A dead end…

Too long time to collect enough data for training RL agent

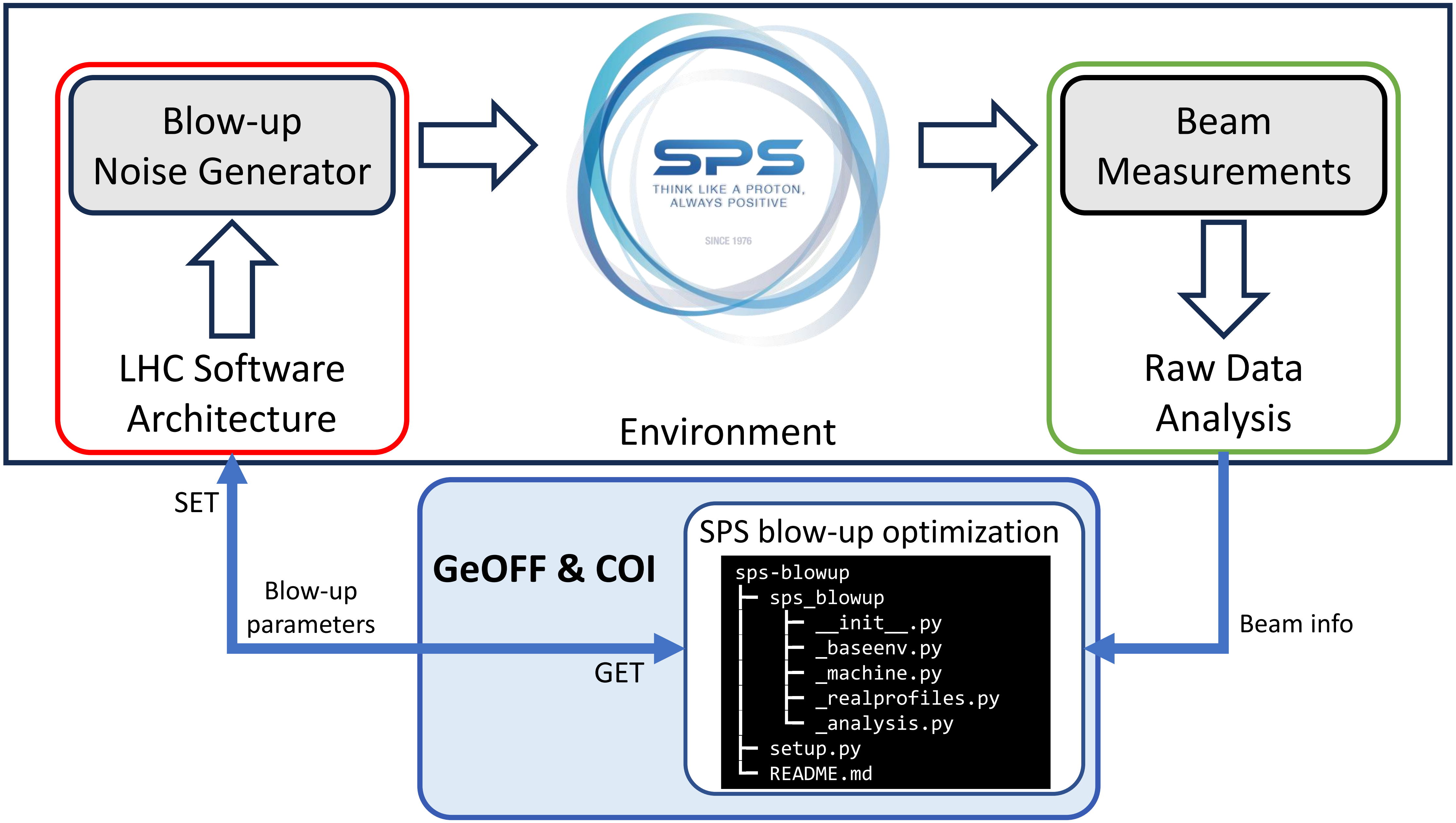
- Only uniform settings were considered, exponential growth of problem complexity with intermediate points…

Software for optimal blow-up was required **as soon as possible**

→ Generic optimizers to replace RL solutions

Outline

1. Introduction to blow-up
   - Importance for longitudinal stability
   - Blow-up settings
2. Purpose of RL application
   - Definition of State, Actions, Reward
   - Critical issues
3. Optimization loop
   - Alternative approaches
   - Discussion
4. Conclusions

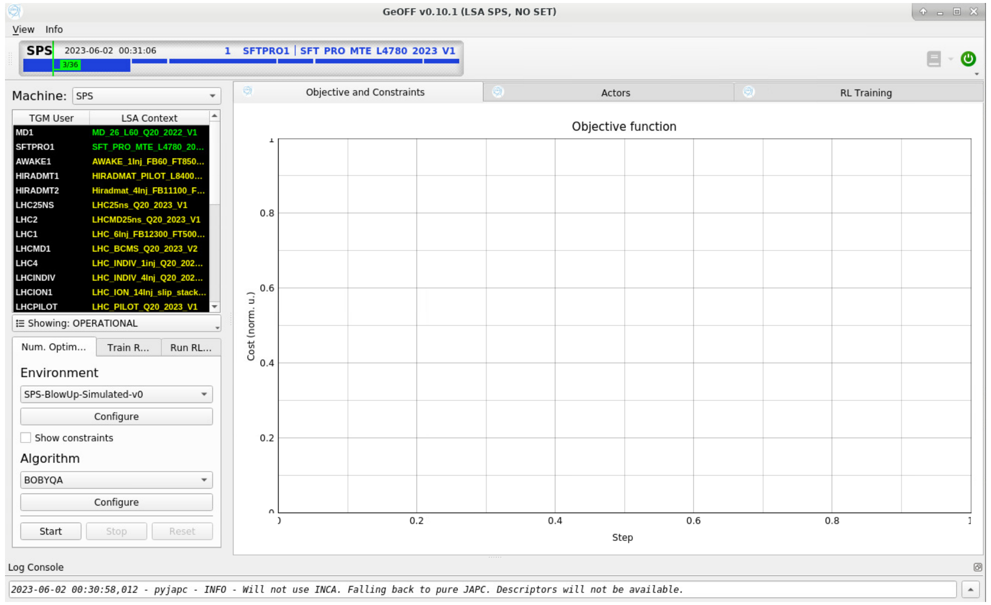

Blow-up
Noise Generator
LHC Software
Architecture
SPS
THINK LIKE A PROTON,
ALWAYS POSITIVE
SINCE 1976
Environment
Beam
Measurements
Raw Data
Analysis
SET
Blow-up
parameters
GeOFF & COI
GET
SPS blow-up optimization
sps-blowup
sps_blowup
__init__.py
_baseenv.py
_machine.py
_realprofiles.py
_analysis.py
setup.py
README.md
Beam info

Integration in the CERN Optimization Framework

Generic Optimization Frontend and Framework (GeOFF)

"… graphical application for generic numerical optimisation and reinforcement learning on CERN accelerators.
It bundles:

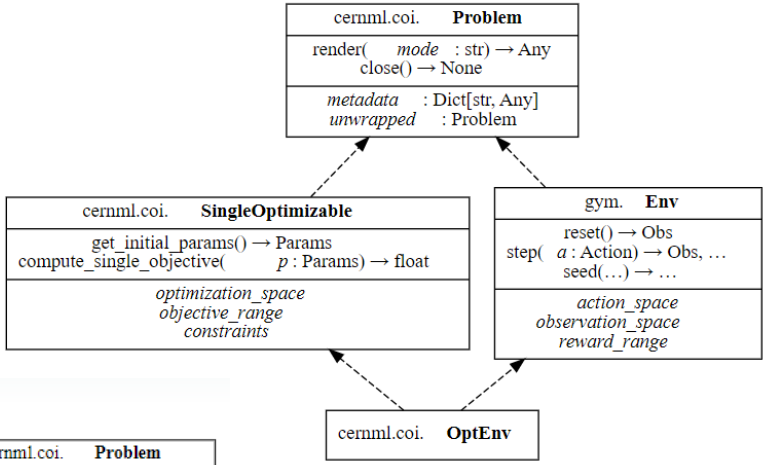
- *Interfaces to the machines and simulations thereof*
- *Numerical optimisers and reinforcement learners that can use these interfaces."*

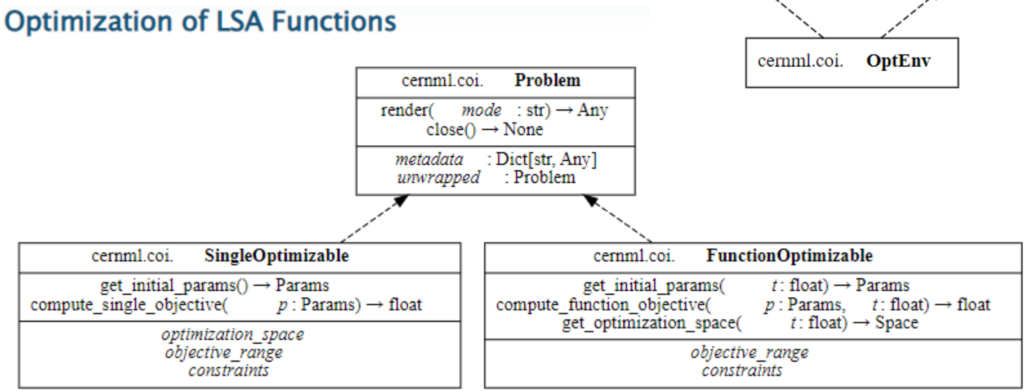


Common Optimization Interfaces (COI)

"… facilitate using numerical optimization and reinforcement learning (RL) on the same optimization problems. This makes it possible to unify both approaches into a generic optimization application in the CERN Control Center."

Inheritance diagram of the core interfaces



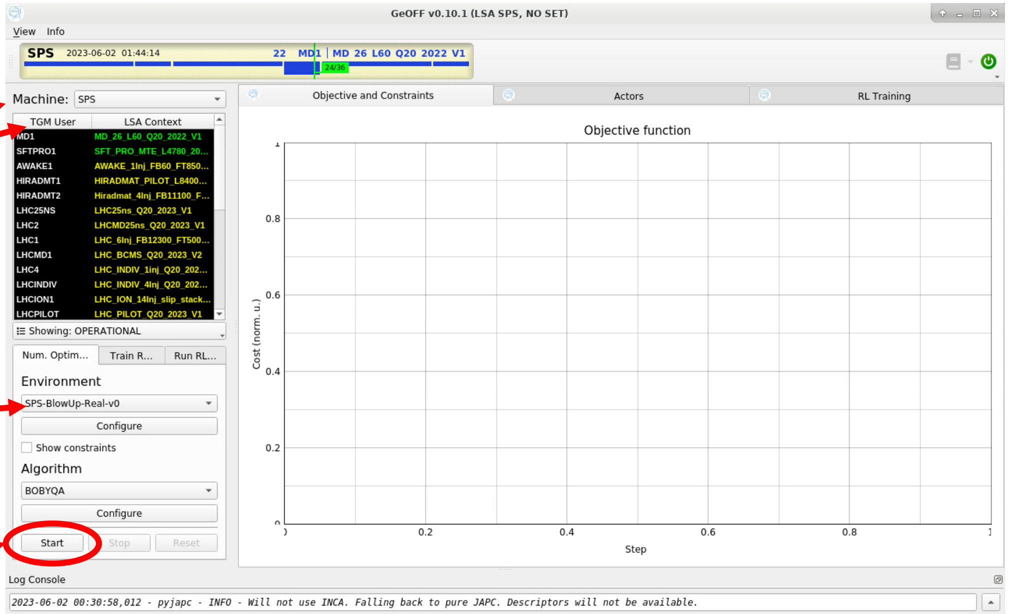


Configuring the Tool

1. Check parameters in LSA
   - Select the noise settings to optimize and set their min-max range
     - Amplitude, margin low, margin high
   - Set the number of sub-intervals, the start time and end time
   - Set the number of bunches in the circulating beam
   - Set the target bunch length

2. Run the optimization in GeOFF
   - Select the Machine: SPS
   - Select the User
   - Environment: `SPS-BlowUp-Real-v0`

3. Push Start and… sit back and relax!

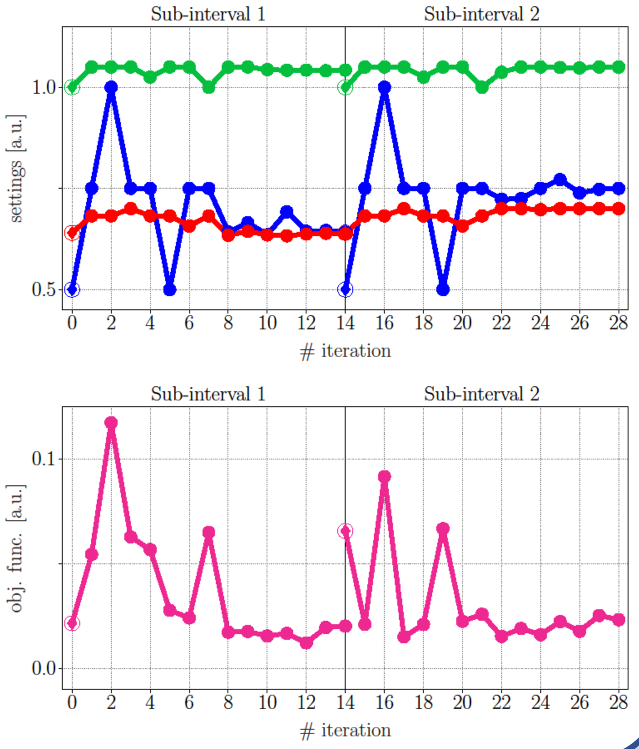


Results

GENERIC OPTIMIZATION (BOBYQA)

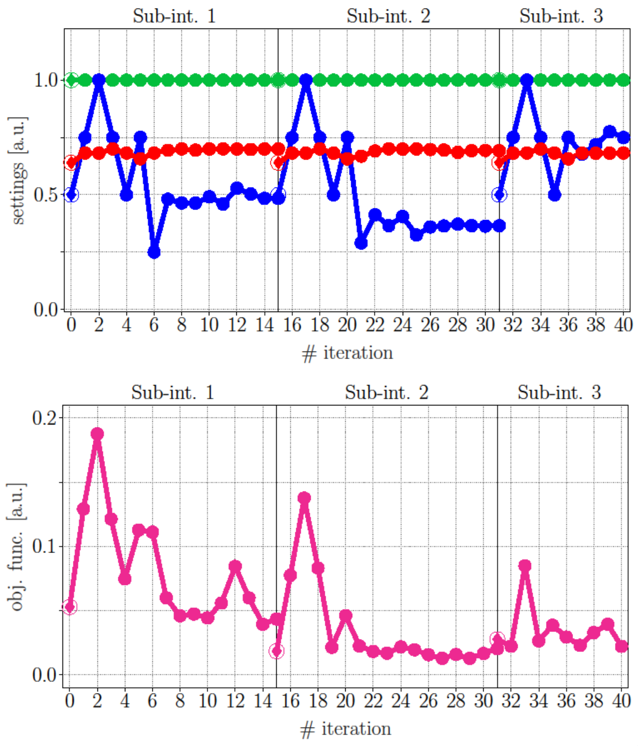
Scan of a, m_{low} and m_{high} on two sub-intervals

Target bunch length: 2.1 ns



Scan of a and m_{low} on three sub-intervals

Target bunch length: 2.0 ns

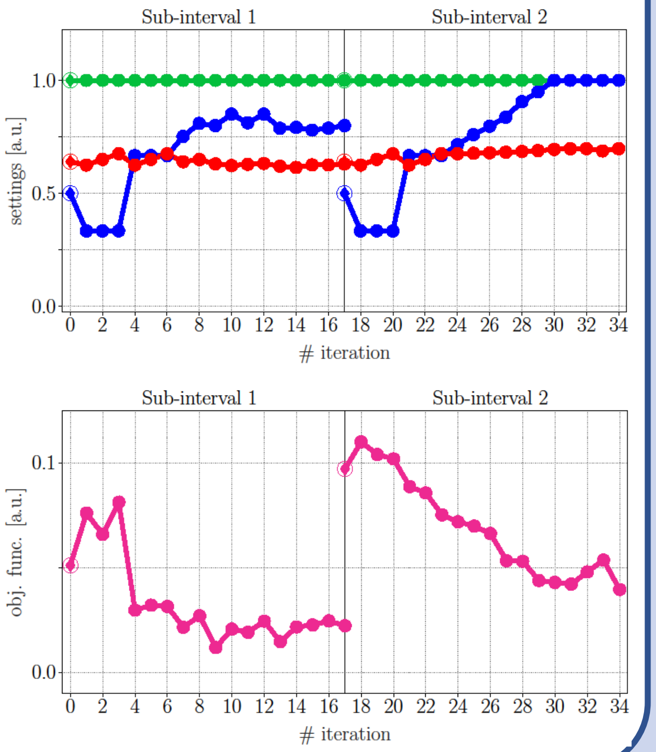


Intensity per bunch: $\mathbf{2.15 \cdot 10^{11}}$ **ppb** - # bunches: **288**

BAYESIAN OPTIMIZATION

Scan of a and m_{low} on two sub-intervals

Target bunch length: 2.0 ns



Int. per bunch: $\mathbf{1.90 \cdot 10^{11}}$ **ppb** - # bunches: **56**

Discussion

Trade-off: optimization time and bunch length constant

- More sub-intervals mean more iterations
- More sub-intervals ensure flatter bunch length

<u>Acceptable results with two and three sub-intervals</u>

Single sub-interval:
faster, suitable if a constant bunch length is not required

Four sub-intervals:
longer time required for limited improvement in bunch length

Bayesian optimization

- Dataset for exploration by grid search
- Fixed number of steps for optimization

Drawback: increased number of steps

- Need of mitigation strategies (e.g., reuse previous data, define convergence criteria, …)

Outline

1. Introduction to blow-up
   - Importance for longitudinal stability
   - Blow-up settings
2. Purpose of RL application
   - Definition of State, Actions, Reward
   - Critical issues
3. Solutions with other approaches
   - Discussion
4. Conclusions

Conclusions

An **operational tool** to automatically setup the controlled longitudinal emittance blow-up for high-intensity multi-bunch LHC beams was developed

- **Integration** in LSA and the CERN optimization framework
- Simple objective function is based on **RMSE**
- **Sequential** setup by splitting blow-up duration in sub-intervals

Blow-up setup **successful** for **high intensity** proton LHC beams

- **Promising results** with **Bayesian optimization**, improve sample-efficiency

Released software supports the extension to RL methods

- Faster way to simulate data
- Improve sample-efficiency

Automatic Setup of Controlled Longitudinal Emittance Blow-Up

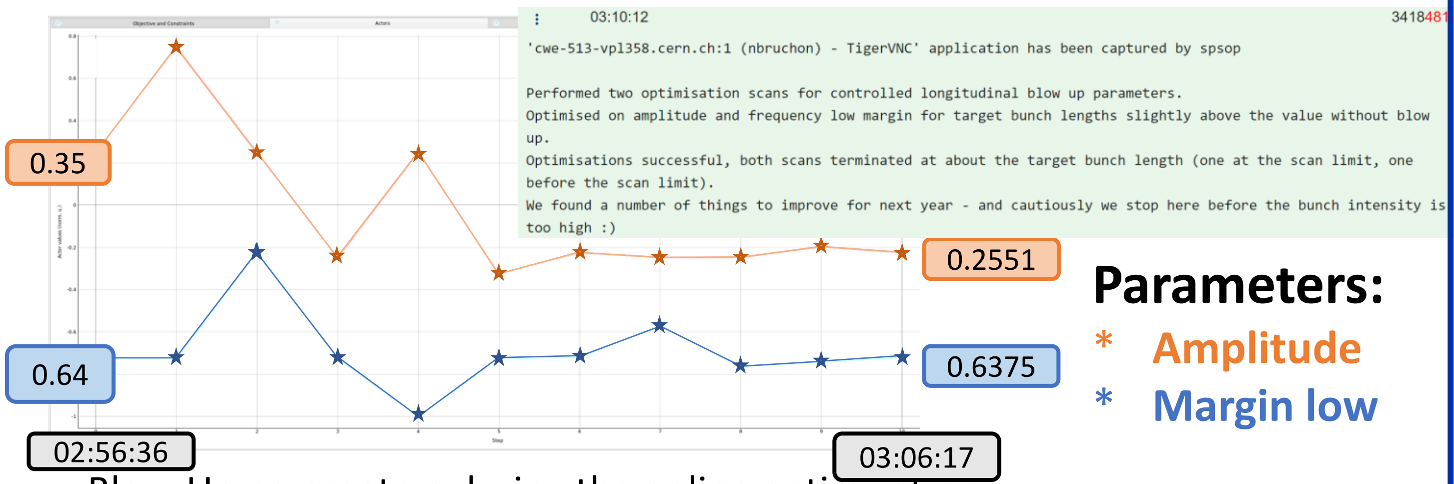
>> Considerations on RL Feasibility <<

N. Bruchon

Blow-up Past Presentations/Publications

- ML and data analysis discussions:
  - 03/03/2022
- ML Community Forum meetings:
  - 29/04/2022
  - 30/11/2022
- IPAC 22:
  - TUPOST42 (proceedings)
  - 09/06/2022 (poster)
- SPS MPC:
  - 06/06/2023
- ICALEPCS 23:
  - 07/12/2023 (flash presentation)

ML: Machine Learning

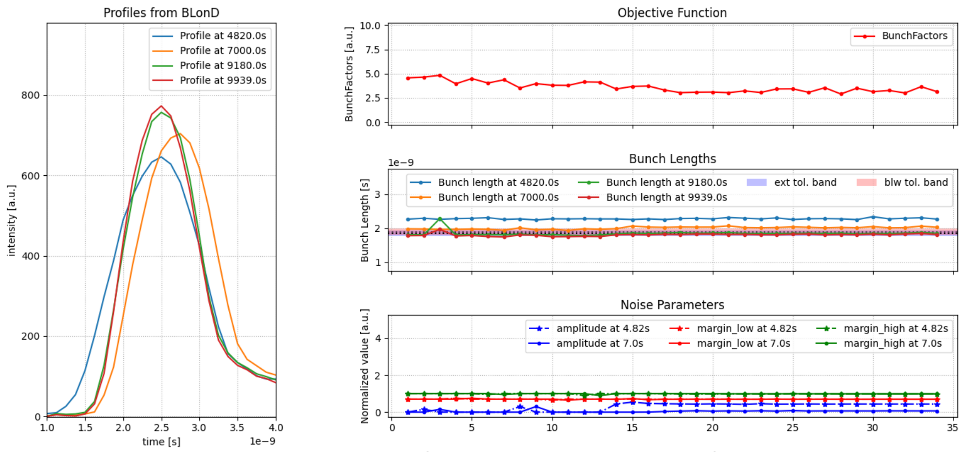


Blow-Up parameters during the online optimization

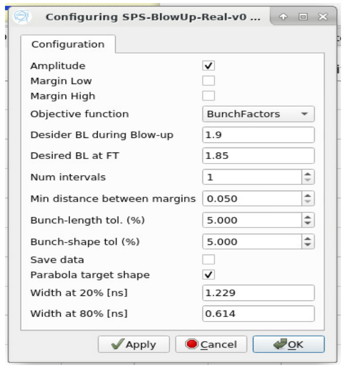
Beam process: HIRADMT2 Hiradmat_4Inj_FB11100_FT500_Q20_2021_V1

Parameters:
- * Amplitude
- * Margin low

Nov. 2021



Data on the LHCPILOT Cycle



Configurations

Summer 2022

Common Optimization Interfaces

CERN ML Project:

Numerical Optimization - Machine Learning (ML) - Reinforcement Learning (RL)
for operations in the CERN accelerator complex.

CERNML-COI:

Common Optimization Interfaces (COI) to apply CERN ML solutions on the same problems.
Unification of all approaches into a generic optimization application in the CERN Control Center.

CERNML-COI-UTILS:

Provides utility functions and classes that make it easier to work with the COI.
Improvable keeping the COI stable.

Generic Optimisation Frontend and Framework (GeOFF):

This is the graphical application for generic numerical optimisation and reinforcement learning on CERN accelerators.

ENVIRONMENT:

Statement of the problem to be managed by the generic optimizers.
It should be encapsulated in an appropriate structure and follow some rules.

Some Existing Environments

1. **Optimization of the steering for ISOLDE:**

   Problem of steering optimization for ISOLDE. The goal is to steer the ion beam via electrostatic elements such as to maximize the intensity measured by a downstream Faraday cup.

2. **Optimization of the LEIR Transfer Line:**

   Problem of alignment in the ETL transfer line towards LEIR. The goal is to adjust the current over time of the BHN10 kicker so that the first injection in a cycle has maximum injection efficiency, and all subsequent injections in a cycle land in the same orbit as the first one.

3. **Optimization of the Tune in the SPS:**

   Problem of tune optimization in the SPS. The goal is to manipulate the horizontal and vertical tune at two times within a cycle so that it maximizes the transmission. The transmission is defined as the beam intensity at the beginning of slow extraction divided by the beam intensity at the end of injection.

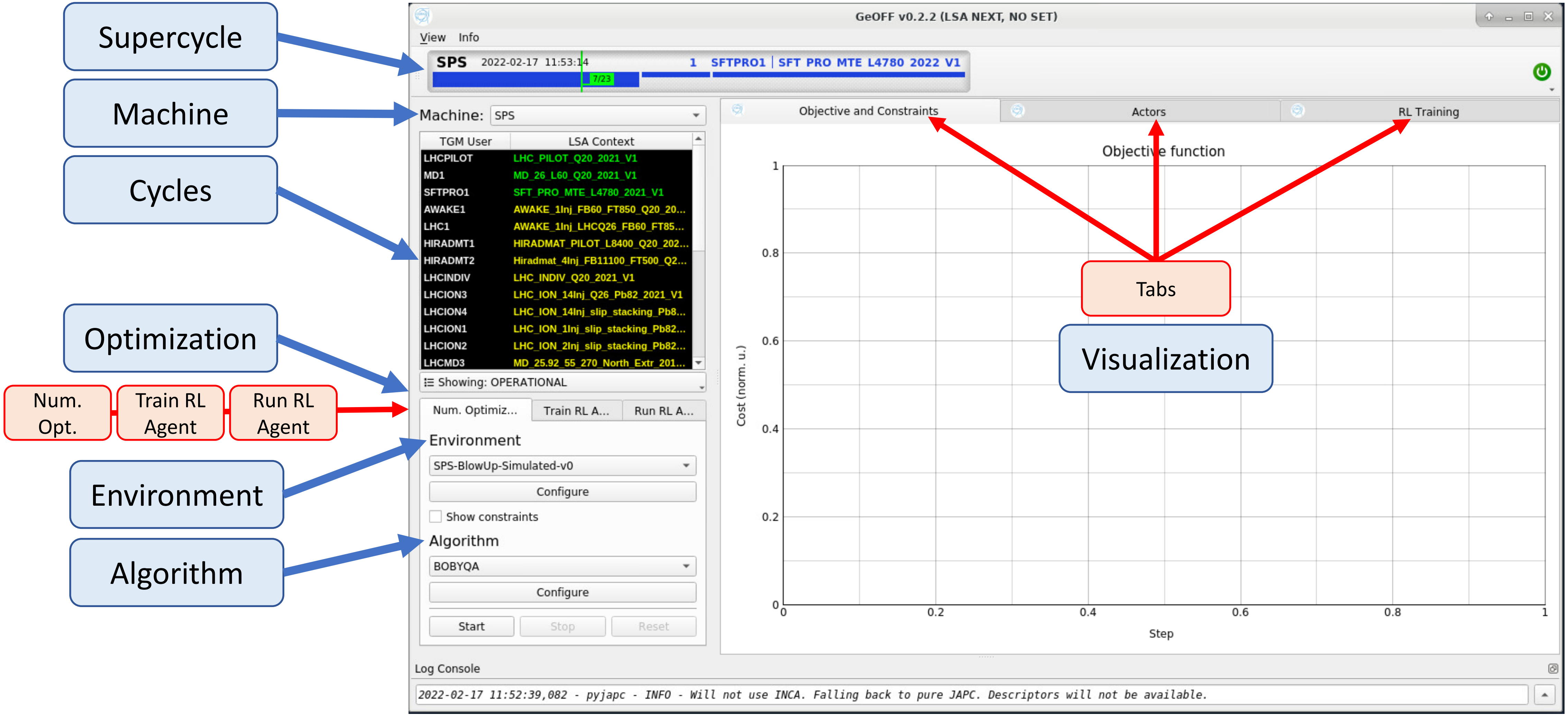
4. **Optimization of the ZS Alignment in the SPS:**

   Problem of ZS alignment in the SPS. The goal is to move 9 anodes in such a manner that during slow extraction, the beam passes the electromagnetic septum and does not hit the beamline walls.

5. **Optimization of the Blow-Up in the SPS:**

   Problem of Blow-Up setup in the SPS. The goal is to reach a desired longitudinal emittance by controlling the amplitude and the frequency margins of the noise injected in the phase.

GeOFF



Supercycle
Machine
Cycles
Optimization
Num. Opt.
Train RL Agent
Run RL Agent
Environment
Algorithm
GeOFF v0.2.2 (LSA NEXT, NO SET)
View Info
SPS 2022-02-17 11:53:14
1 SFTPRO1 | SFT PRO MTE L4780 2022 V1
Machine: SPS
TGM User
LSA Context
LHCPILOT LHC_PILOT_Q20_2021_V1
MD1 MD_26_L60_Q20_2021_V1
SFTPRO1 SFT_PRO_MTE_L4780_2021_V1
AWAKE1 AWAKE_1Inj_FB60_FT850_Q20_20...
LHC1 AWAKE_1Inj_LHCQ26_FB60_FT85...
HIRADMT1 HIRADMAT_PILOT_L8400_Q20_202...
HIRADMT2 Hiradmat_4Inj_FB11100_FT500_Q2...
LHCINDIV LHC_INDIV_Q20_2021_V1
LHCION3 LHC_ION_14Inj_Q26_Pb82_2021_V1
LHCION4 LHC_ION_14Inj_slip_stacking_Pb8...
LHCION1 LHC_ION_1Inj_slip_stacking_Pb82...
LHCION2 LHC_ION_2Inj_slip_stacking_Pb82...
LHCMD3 MD_25.92_55_270_North_Extr_201...
Showing: OPERATIONAL
Num. Optimiz...
Train RL A...
Run RL A...
Environment
SPS-BlowUp-Simulated-v0
Configure
Show constraints
Algorithm
BOBYQA
Configure
Start
Stop
Reset
Objective and Constraints
Actors
RL Training
Objective function
Tabs
Visualization
Cost (norm. u.)
Step
Log Console
2022-02-17 11:52:39,082 - pyjapc - INFO - Will not use INCA. Falling back to pure JAPC. Descriptors will not be available.

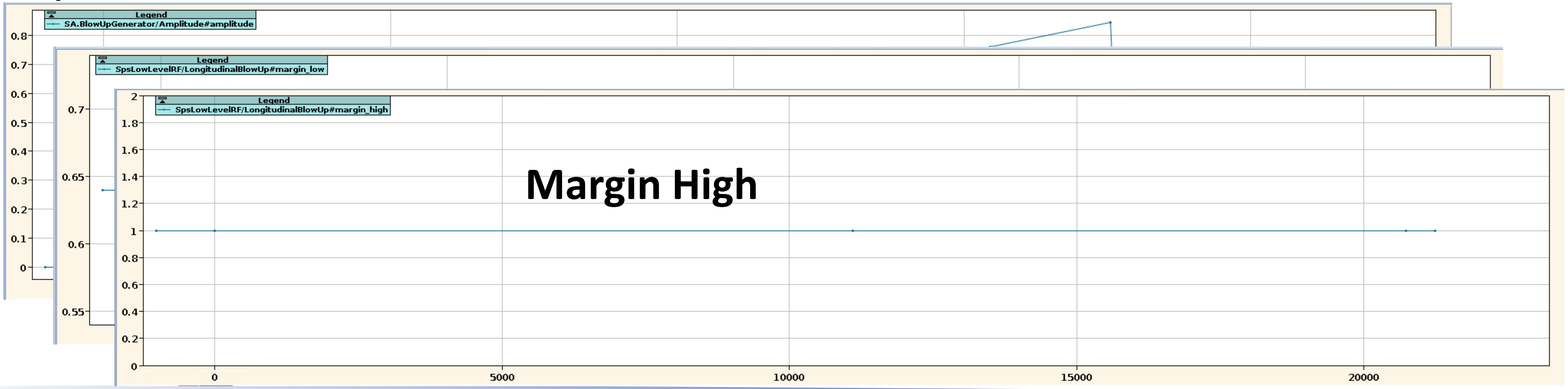
LSA Integration: Functions

Advantages of the LHC Software Architecture:

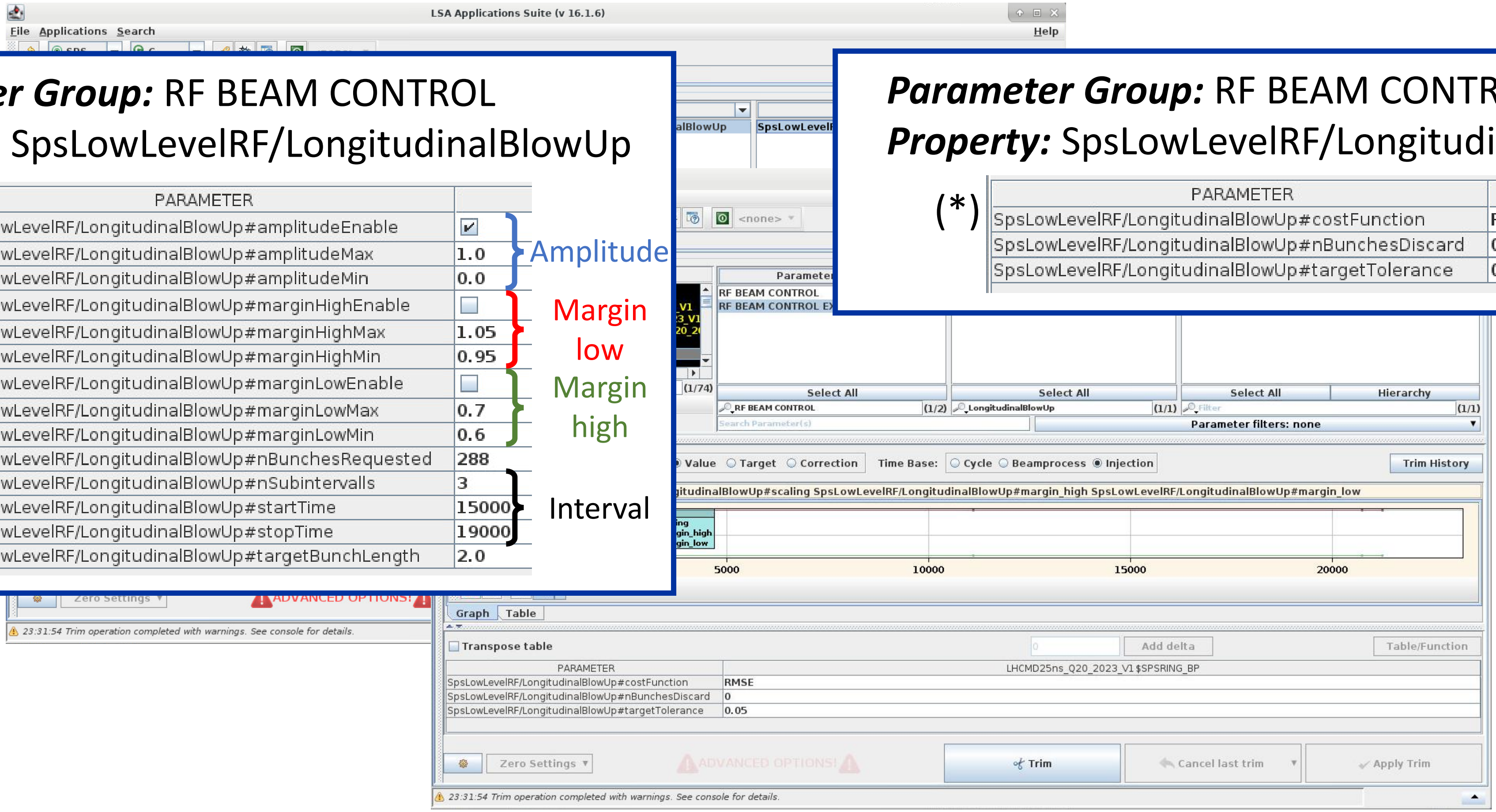
- Keep in the **database** settings of all managed device/properties
- Every change in control value (**trim**) is registered
  - Trim history and rollback

Use of LSA for all tools settings

Optimizable functions:



LSA Integration: Discrete Settings



Parameter Group: RF BEAM CONTROL

Property: SpsLowLevelRF/LongitudinalBlowUp

(*)

PARAMETER		
SpsLowLevelRF/LongitudinalBlowUp#amplitudeEnable	☑	Amplitude
SpsLowLevelRF/LongitudinalBlowUp#amplitudeMax	1.0	
SpsLowLevelRF/LongitudinalBlowUp#amplitudeMin	0.0	
SpsLowLevelRF/LongitudinalBlowUp#marginHighEnable	☐	Margin low
SpsLowLevelRF/LongitudinalBlowUp#marginHighMax	1.05	
SpsLowLevelRF/LongitudinalBlowUp#marginHighMin	0.95	
SpsLowLevelRF/LongitudinalBlowUp#marginLowEnable	☐	Margin high
SpsLowLevelRF/LongitudinalBlowUp#marginLowMax	0.7	
SpsLowLevelRF/LongitudinalBlowUp#marginLowMin	0.6	
SpsLowLevelRF/LongitudinalBlowUp#nBunchesRequested	288	
SpsLowLevelRF/LongitudinalBlowUp#nSubintervalls	3	Interval
SpsLowLevelRF/LongitudinalBlowUp#startTime	15000	
SpsLowLevelRF/LongitudinalBlowUp#stopTime	19000	
SpsLowLevelRF/LongitudinalBlowUp#targetBunchLength	2.0	

Parameter Group: RF BEAM CONTROL EXPERT

Property: SpsLowLevelRF/LongitudinalBlowUp

(*)

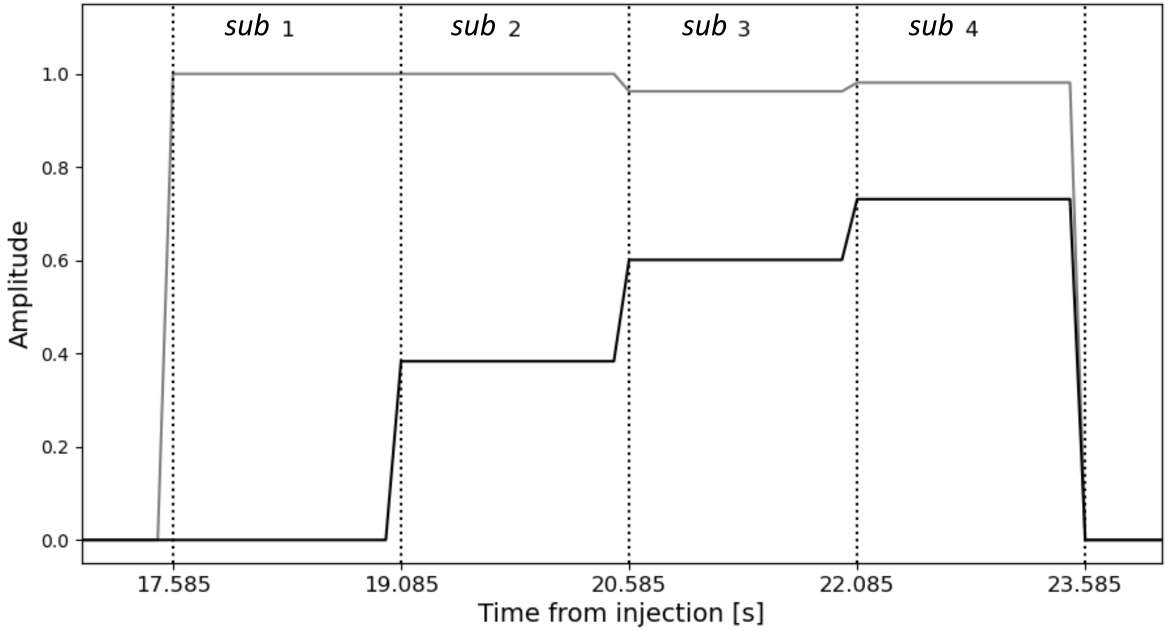
PARAMETER	
SpsLowLevelRF/LongitudinalBlowUp#costFunction	RMSE
SpsLowLevelRF/LongitudinalBlowUp#nBunchesDiscard	0
SpsLowLevelRF/LongitudinalBlowUp#targetTolerance	0.05

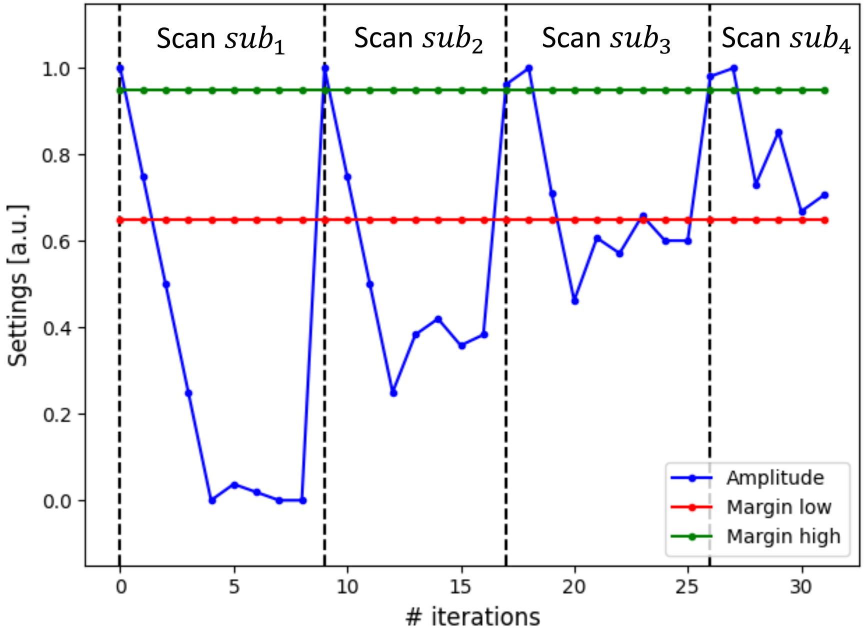
(*) Test parameters for user LHC2

AMPLITUDE SCAN IN THE SPS

SY
RF
BR

- **Blow-up interval:** 19085 – 23585 [ms]; **Num. sub-intervals:** 4
  - **Skeleton Points:** 17585, 19085, 20585, 22085 [ms]
  - **Observation Points:** 19085, 20585, 22085, 23585 [ms]
- **Parameters:** Amplitude
  - **Amplitude Range:** [0, 1]
- **Algorithm:** BOBYQA



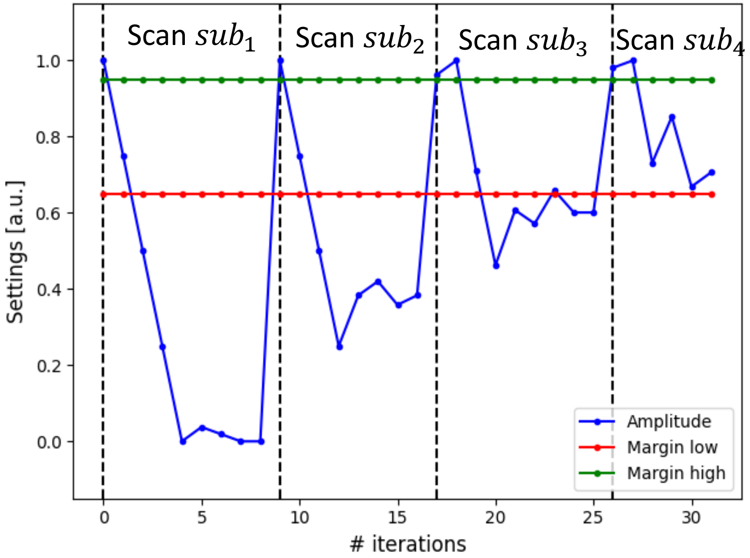


RESULTS IN THE SPS

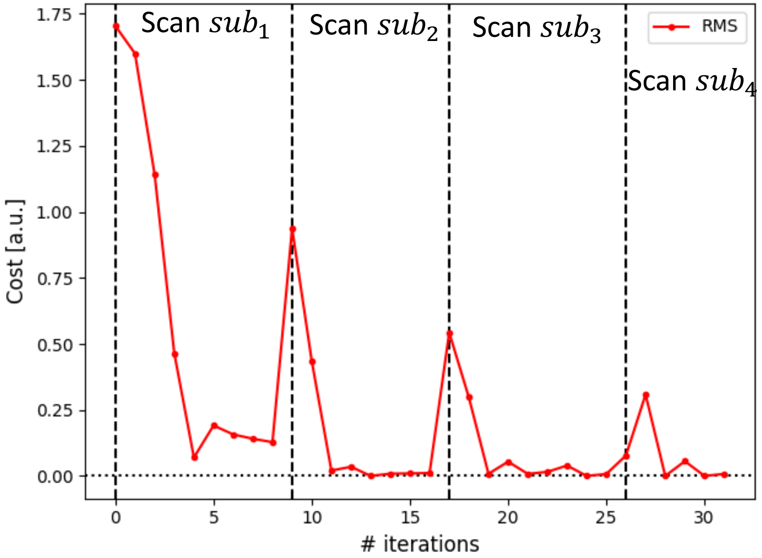
SY
RF
BR

- **Target bunch length:** $\sigma^* = 1.9\ \mathrm{ns}$
  - **Tolerance:** $tol_{\%} = \pm 5\%$ → $tol_{\mathrm{band}} = [1.805,\quad 1.995]\ \mathrm{ns}$
- Problem **solved** in 31 iterations (Num. sub-intervals: 4).

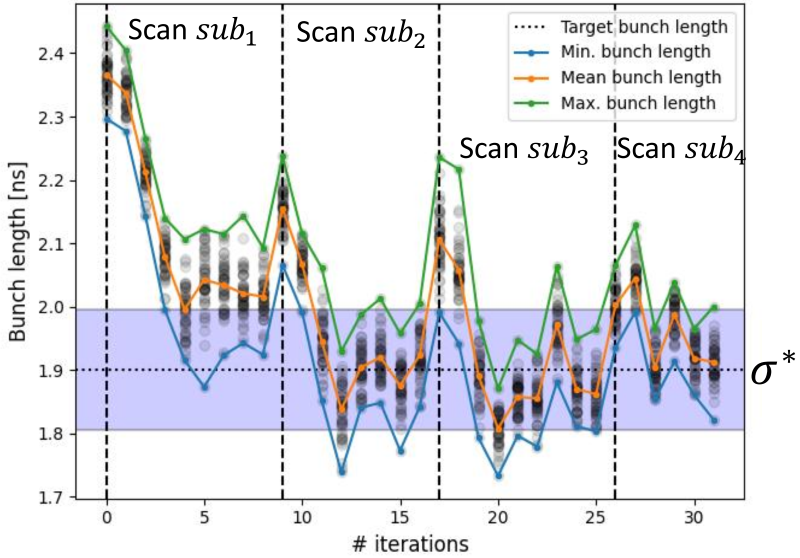
Amplitude scan



Objective function



Bunch lengths



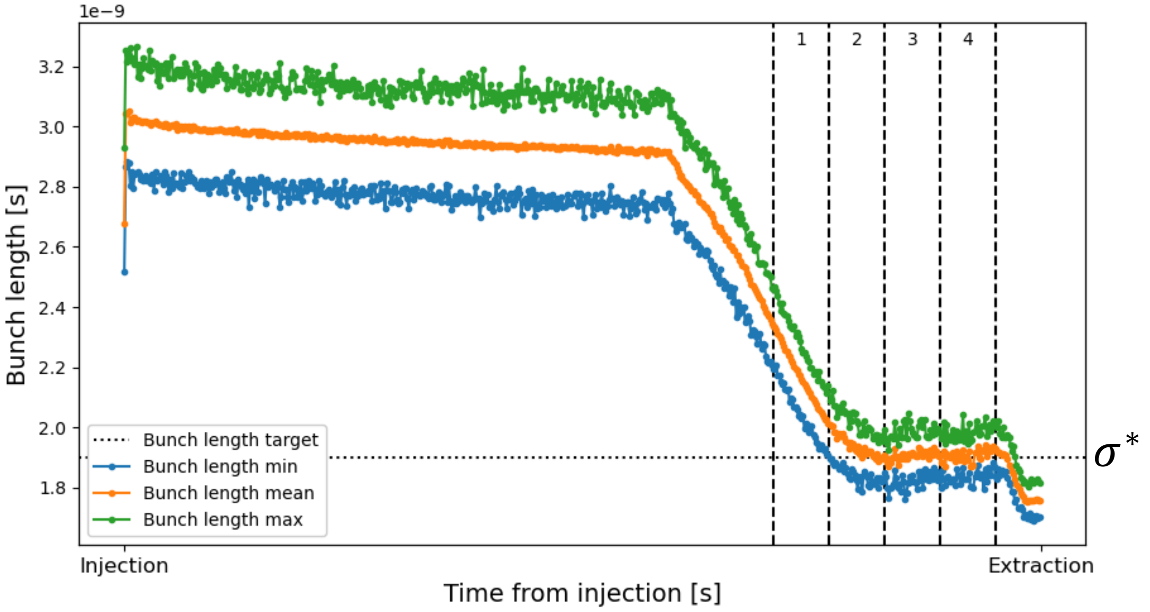
Iterations per sub-interval

Scan sub_1: 0, 8
Scan sub_2: 9, 16
Scan sub_3: 17, 26
Scan sub_4: 27, 31

RESULTS PORTABILITY

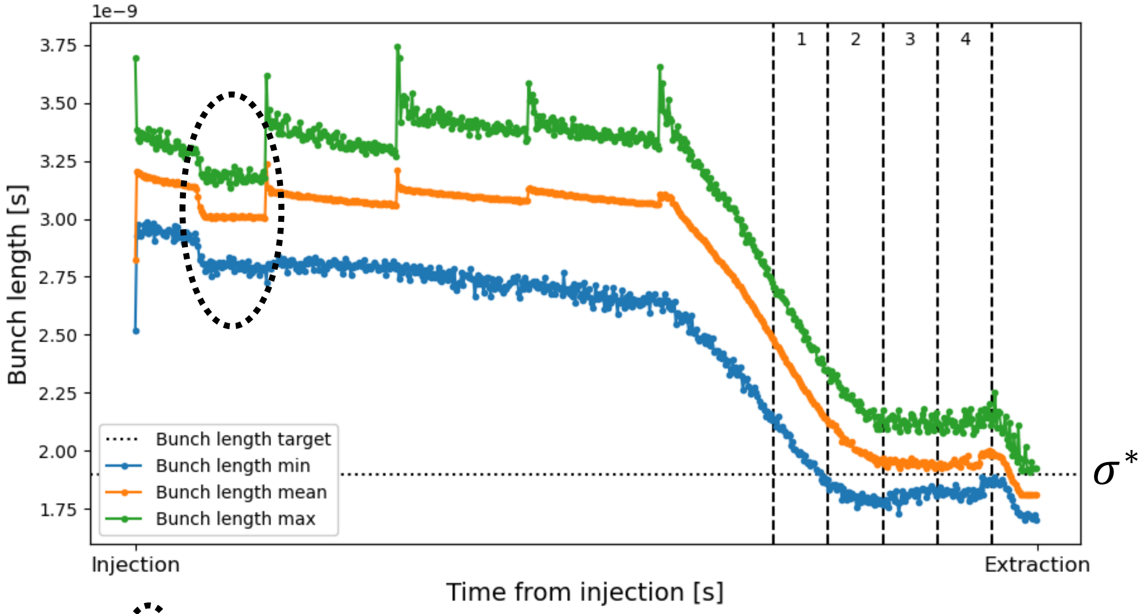
Blow-up optimized for single batch, 06-10-2022.

- **Intensity per bunch:** 1.2 e11 p/b
- **Average bunch length per extraction:** 1.76 ns
  - **Average bunch length at injection:** 2.68 ns



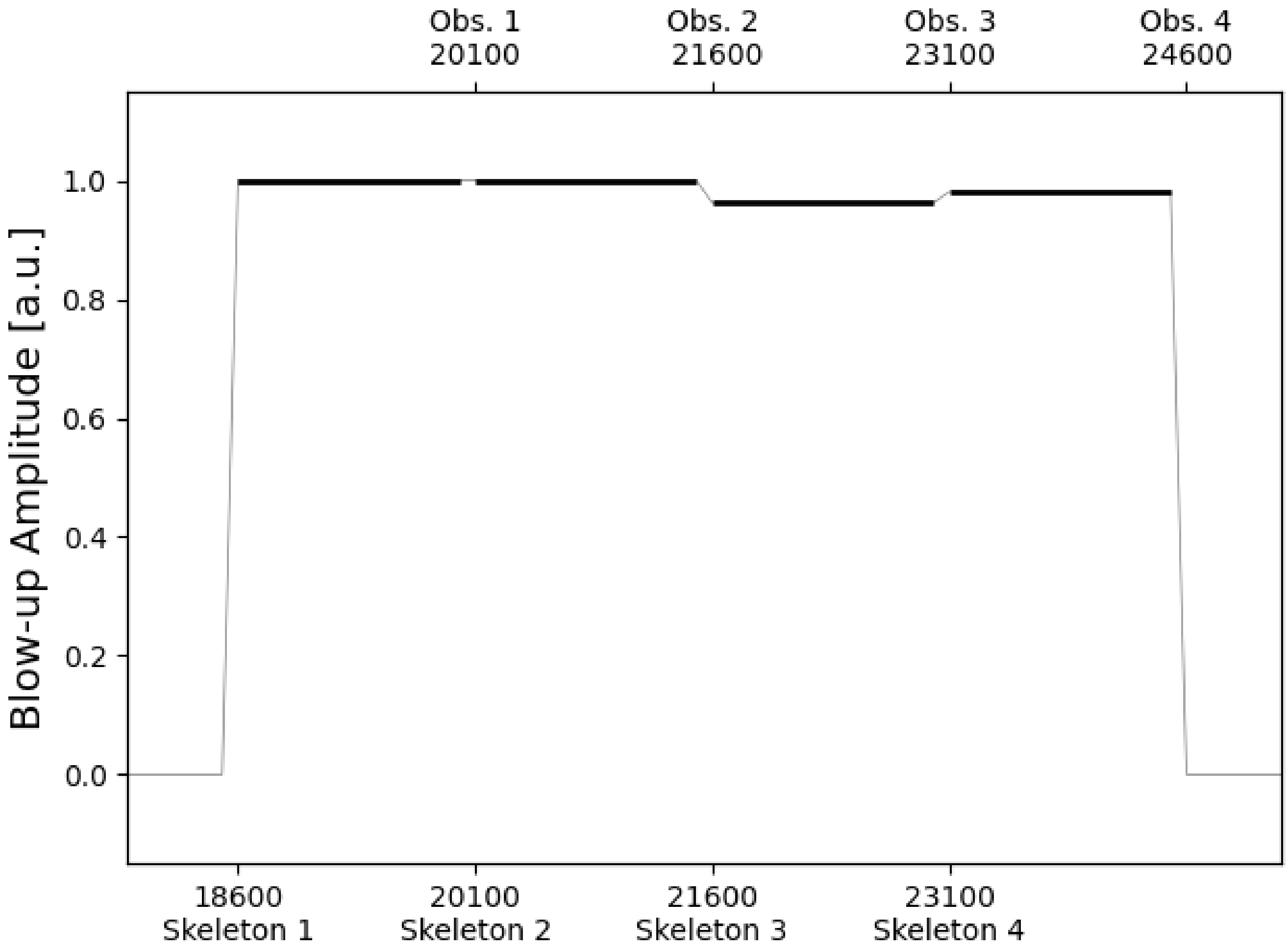
Results on 5 batches without optimizing, 13-10-2022.

- **Intensity per bunch:** 1.5 e11 p/b
- **Average bunch length per extraction:** 1.81 ns
  - **Average bunch length at injection:** 2.82 ns



SCRAPING EXERCISE

LSA Function Trim



Amplitude Optimization

Interval per interval research of the optimal amplitude to keep the bunch length constant.