Geoff: Applications and Developments in 2024

■ Nico Madysa[1] Sabrina Appel[1] Martin Bajzek[1] Erika Kazantseva[1]
Stephane Pietri[1] Helmut Weick[1] Christoph Reinwald[2] Daniel Kallendorf[2]

EURO-LABS Third Annual Meeting,
29 October 2024

[1]GSI, Darmstadt
[2]TU Darmstadt

Summary

Applications

- SIS18 multi-turn injection (MTI):
  - classical single-objective optimization
  - multi-objective Bayesian optimization (MOBO)
- closed-orbit correction via Bayesian optimization (BO)
- FRS: optimization with cross-network communication
- SIS18 slow-extraction optimization

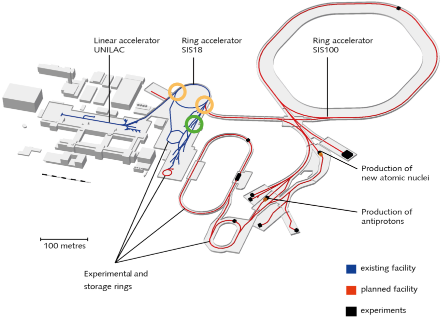
Geoff Development

- upgraded COI from Gym to Gymnasium
- extended API for Bayesian Optimization
- started process for CERN-independent GUI app

GSI/FAIR Overview

Automation & Optimization with Python

- MTI loss minimization (SIS18)
- beam steering (TK)
- closed-orbit correction for non-standard optics (SIS18)
- slow extraction loss minimization (SIS18)
- beam steering and focusing (FRS)



Methods of Interest

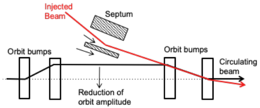
- BOBYQA + Bayesian optimization (BO)
- physics-informed Bayesian optimization
- multi-objective Bayesian optimization (MOBO)
- data-driven model predictive control (MPC)
- reinforcement learning (RL)

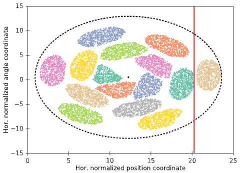
Work power

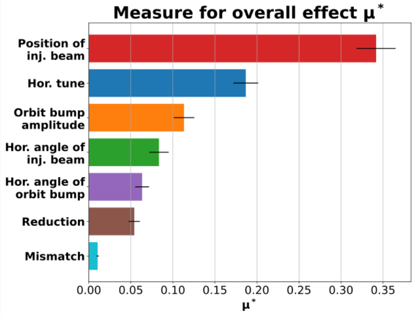
- one scientific staff financed by EURO-LABS
- one master student from PLUS (Salzburg)
- several master/PhD students from TUDa (Darmstadt)

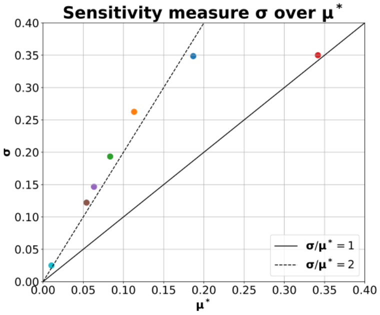
SIS18 Multi-Turn Injection

- Liouville's theorem: beam can only go into free phase space
- Competing goals:
  - maximize gain factor factor (reach space charge limit)
  - minimize losses at septum (cause vacuum degradation)
- MTI model implemented in Xsuite for fast tracking







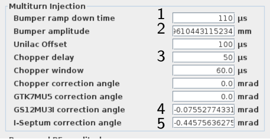


- beam position + tune most important
- all parameters have higher order dynamics (shown by σ/μ^*)

credit: C. Reinwald, master's thesis

SIS18 MTI Optimization

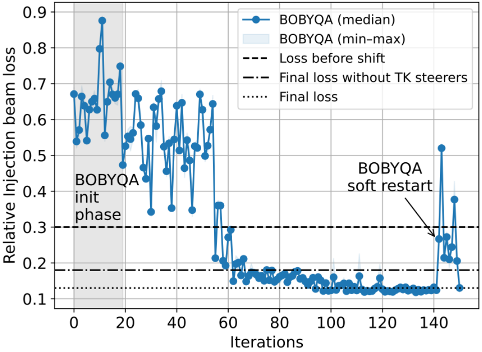
- two beam times (Nov'23, Jun'24)
- MTI is usually optimized manually
- nine optimization parameters:



Multiturn Injection			
Bumper ramp down time	1	110	µs
Bumper amplitude	2	9610443115234	mm
Unilac Offset		100	µs
Chopper delay	3	50	µs
Chopper window		60.0	µs
Chopper correction angle		0.0	mrad
GTK7MU5 correction angle		0.0	mrad
GS12MU3I correction angle	4	-0.07552774331	mrad
I-Septum correction angle	5	-0.44575636275	mrad

+ 4 TK steerers

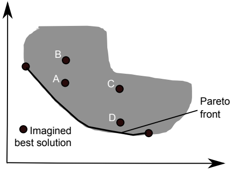
- 150 iterations ≈ 30 minutes
- 1 iteration = median of 3 evaluations
- loss reduced from 30 % to 12 %
- successful use of Geoff + BOBYQA outside of CERN



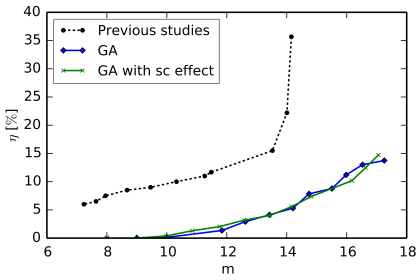
- algorithm's internal model not reusable
- gray area = initialization phase of algorithm

Mult-Objective Bayesian Optimization

- competing goals: maximize number of injections, minimize loss
- Pareto front: set of solutions strictly better than others
- prior art: simulation with genetic algorithms (S. Appel)

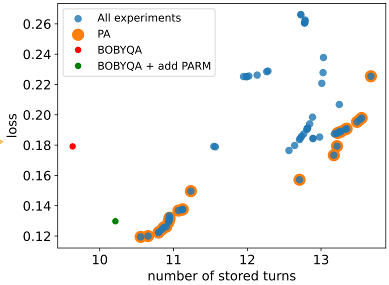


Simulation with Genetic Algorithms



Optimization of injector beam line is missing

Measurements

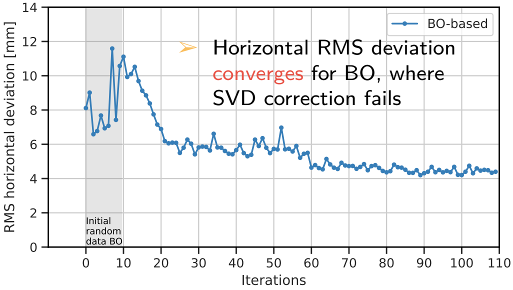


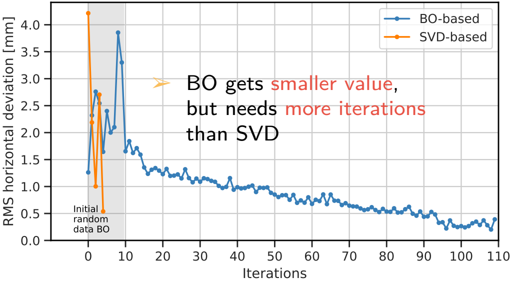
SIS18 Closed-Orbit Correction

- broken-symmetry high-transition-energy SIS18 optics: standard technique fails
- BO provides complementary approach
- additionally handles BPM noise better

Current work:

physics-informed covariance kernels for orbit model between BPMs



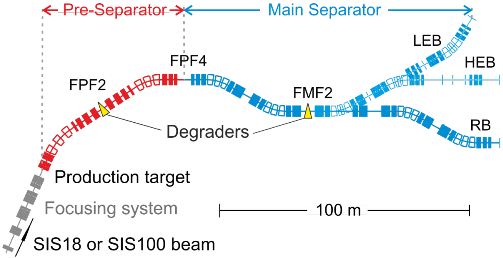


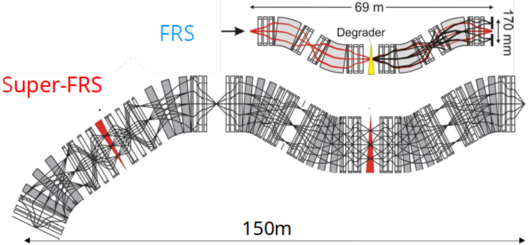
credit: V. Isensee, master's thesis

Fragment Separator (FRS) and Super-FRS

- investigates nuclear structure of exotic nuclei
- produces, separates and identifies exotic nuclei
- sends them to downstream experiments or storage ring
- Super-FRS: higher acceptance, more complex (4× more magnets), gain factors between 1000 (^{12}C) and 7500 (^{132}Sn)
- automation of operational tasks essential

	$B\rho_{max}$	$\Delta p/p$	$\Delta\Phi_x$, $\Delta\Phi_y$	resolving power	gain factor ^{19}C	gain factor ^{132}Sn
FRS	18 Tm	1.0 %	±13, ±13 mrad	1500	1	1
Super-FRS	20 Tm	2.5 %	±40, ±20 mrad	1500	5	10
				including primary rate	250	20 000





FRS and Super-FRS: Goals

Motivation

- scale: manual setup too time consuming
- complexity: different optical modes per user
- accuracy: scaling for same optics but different $B\rho$ not accurate enough

Observables

- profile grid histograms
- phase space spectra from tracking detectors

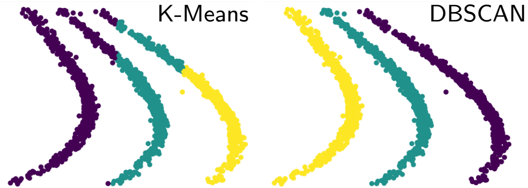
Machine Interface

- LSA database (trim steerers, magnets)
- FESA (SIS18 monitoring)
- Go4 (experiment-side TPC, current grids)

Optimization Goals

0^{th} order: center the beam (vertical steerers, main dipoles)
1^{st} order: set focus, dispersion (quadrupoles)
2^{nd} order: minimize aberrations (sextupoles)
3^{rd} order: minimize aberrations (octupoles, S-FRS only)

Figure: Track classification (simulation with disabled sextupoles)



credit: D. Kallendorf, master's thesis

FRS and Super-FRS: Observables

Central spot:

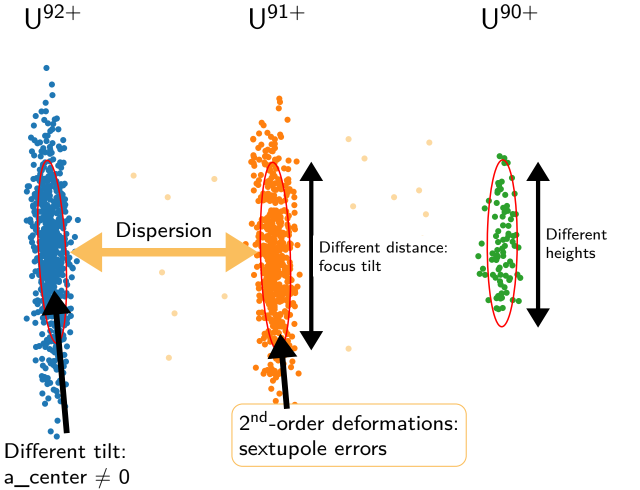
- Twiss parameters
- Non-dispersive 1st-order transfer matrix

Outer spots:

- 1st-/2nd-order dispersive transfer matrix elements

Goal: Bring the spectra as close as possible to desired 1st order parameters

⇒ optimize direct observables, not individual matrix elements!



FRS and Super-FRS: Results

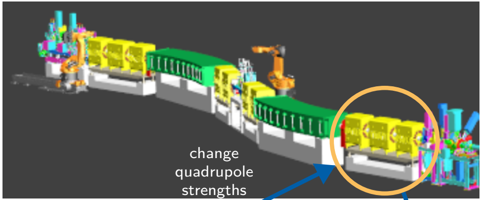
Proof of principle:

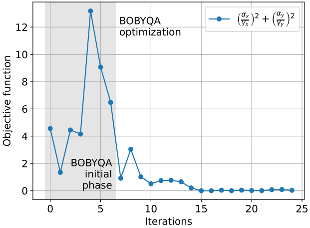
1. **steer beams** at target
2. **tune quadrupoles** to focus beam (central spot upright)
3. **tune sextupoles** to remove focal-plane tilt and “banana” deformation

Next step

- use other observables
- tune entire 1st-order optics
- include dispersion

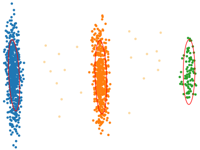
Illustration of FRS Dispersive Area





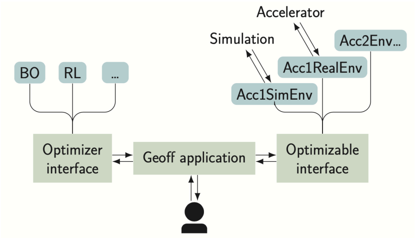
TPC Data Dnalysis

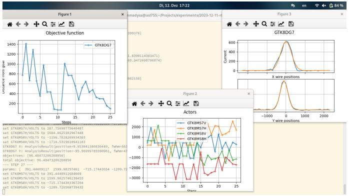
evaluation of beam spot slope



Geoff

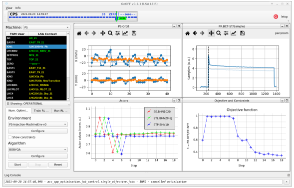
- Python-based framework for reinforcement learning and optimization
- lists, configures and runs optimization problems
- standardized interfaces and adapters
- supports plotting, data logging, and any controls system (with Python bindings)
- flexible, easy to adapt code on-the-fly





← runs in the terminal ...

... and in a GUI! →

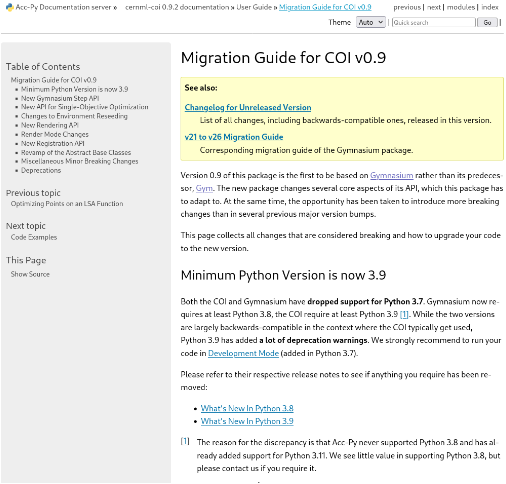


Geoff Status

- main effort: migration from outdated *Gym* library to *Gymnasium*
- required extensive rewrite
- all core packages updated
- additionally: API extended to support Bayesian optimization

Next Steps:

- update GUI app, provide backwards compatibility
- help users at CERN and GSI to update individual plugins
- Nov: start CEA proof of concept



Acc-Py Documentation server » cernml-coi 0.9.2 documentation » User Guide » Migration Guide for COI v0.9 previous | next | modules | index

Theme Auto | Quick search Go |

Table of Contents

Previous topic

Optimizing Points on an LSA Function

Next topic

Code Examples

This Page

Show Source

Migration Guide for COI v0.9

See also:

Changelog for Unreleased Version
: List of all changes, including backwards-compatible ones, released in this version.

v21 to v26 Migration Guide
: Corresponding migration guide of the Gymnasium package.

Version 0.9 of this package is the first to be based on Gymnasium rather than its predecessor, Gym. The new package changes several core aspects of its API, which this package has to adapt to. At the same time, the opportunity has been taken to introduce more breaking changes than in several previous major version bumps.

This page collects all changes that are considered breaking and how to upgrade your code to the new version.

Minimum Python Version is now 3.9

Both the COI and Gymnasium have **dropped support for Python 3.7**. Gymnasium now requires at least Python 3.8, the COI require at least Python 3.9 [1]. While the two versions are largely backwards-compatible in the context where the COI typically get used, Python 3.9 has added **a lot of deprecation warnings**. We strongly recommend to run your code in Development Mode (added in Python 3.7).

Please refer to their respective release notes to see if anything you require has been removed:

- What's New In Python 3.8
- What's New In Python 3.9

[1] The reason for the discrepancy is that Acc-Py never supported Python 3.8 and has already added support for Python 3.11. We see little value in supporting Python 3.8, but please contact us if you require it.

Summary and Outlook

Summary

- successful proof of concept at GSI, endorsement from leadership
- Gymnasium migration slower than expected, but core is done
- widespread use at CERN

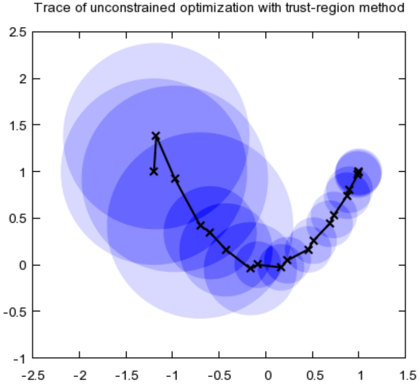
Outlook

- roll out Gymnasium upgrade at CERN and GSI
- GSI contributions to GUI app development $\Rightarrow$ beneficial to EURO-LABS!
- more experiments at GSI and CEA
- upcoming major shifts in ecosystem: Gymnasium $\rightarrow$ 1.0, NumPy $\rightarrow$ 2.0
  - chance to be ahead of the curve!

Backup

BOBYQA

- developed by Michael J. D. Powell in 2009
- sequential trust–region algorithm
  - quadratic approximations of objective
  - update model and trust radius on each iteration
- Py-BOBYQA adds global optimization and robustness against noise



Bayesian Optimization

- probabilistic model of objective
- Gaussian process prior expresses assumption about objective (mean function)
- kernel function $k(x, x')$ describes correlation in phase space
- acquisition function decides next evaluation point
- choice of kernel/acquisition function essential!
- easy extension to multi-objective optimization