ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV008

ONLINE AUTOMATIC PERFORMANCE OPTIMIZATION FOR THE ELETTRA SYNCHROTRON

G. Gaio†, S. Krecic, F. Tripaldi, Elettra-Sincrotrone Trieste S.C.p.A., Trieste, Italy

Abstract

Online automatic performance optimization is a common practice in particle accelerators. Beside the attempts based on Machine Learning, which is effective especially on non-linear systems and images but are very complex to tune and manage, one of the most simple and robust algorithms, the simplex Nelder Mead, is extensively used at Elettra to automatically optimize the synchrotron parameters. It is currently applied to optimize the efficiency of the booster injector by tuning the pre-injector energy, the trajectory and optics of the transfer lines, and the injection system of the storage ring. It has also been applied to maximize the intensity of the photon beam on a beamline by changing the electron beam position and angle inside the undulator. The optimization algorithm has been embedded in a Tango device that also implements generic and configurable multi-input multi-output feedback systems. This optimization tool is usually included in a high-level automation framework based on Behavior Trees [1] in charge of the whole process of machine preparation for the experiments.

INTRODUCTION

Similarly to most of the modern light sources, the Elettra 2-2.4GeV synchrotron relies on feedback systems to keep the beams stable and, more recently, on automatic optimization systems to help operators in maximizing the performance.

Feedback and optimization systems look in some aspects very similar. They have both sensors, actuators and have in common the objective of minimizing the distance between sensors and a reference. In feedback systems the reference is a setpoint of a machine parameter that have to kept fixed no matter the noise or external perturbations affecting the accelerator. In optimization problems the reference is still present but usually set to an arbitrarily very high /very low value if the intention is to maximize / minimize the sensor value.

By the way, excluding the algorithmic part, the software internal components (acquisition, command and control) are the same.

In light sources programmable feedback and/or optimization tools are quite common. Some of them are implemented as high-level applications [2,3] others are server applications [4].

At Elettra a programmable C++ Tango server, called MIMOFB (Multi Input Multi Output Feedback), has been developed to replace legacy applications implementing slow feedback systems and to implement automatic optimization procedures.

† email: giulio.gaio@elettra.eu

MIMOFB

When configured as a feedback, the MIMOFB implements a correction scheme based on a linear model that link sensors and actuators. This relationship is an approximation that is empirically calculated by measuring the perturbation generated on the sensors by one actuator at a time. The result of this process is a matrix, formally a Response Matrix (RM). The product between the inverse of the response matrix, usually inverted using the Singular Value Decomposition (SVD) algorithm, and the error vector returns the values to subtract from actuators to minimize the distance between the sensors and the reference.

Regarding optimization, the MIMOFB implements only model-less optimization schemes. In this case the objective function F (1) is the sum of the normalized distances between N sensor values and corresponding references multiplied by the sensor weights.

$$F = \sum_{i=0}^{N} w_i * \frac{abs(x_i - x_i^{ref})}{x_i^{\max\_thres} - x_i^{\min\_thres}} \tag{1}$$

The configuration of a MIMOFB device is stored in several device properties.

For each sensor the developer has to configure:

- the sensors Tango attributes
- the number of sensor readings per cycle
- descriptive label
- weight
- minimum threshold value
- maximum threshold value
- dead-band
- sensor update rate in ms
- type of filtering (none, mean, median) when the number of readings per cycle is higher than one
- Tango device allowed states
- list of actuators affecting the sensors

For each actuator it is required to specify:

- the actuator Tango attribute
- descriptive label
- minimum threshold value
- maximum threshold value
- scan range
- RM kick; this value is used in response matrix calculation and as the initial step in optimization algorithms
- maximum backlash (useful when working with motors)
- dead-band
- maximum difference between last read and set value
- RM kick settling time (msec.); this value changes proportionally to RM kick amplitude

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV008

- constant settling time in msec.
- tango device allowed states.

Thus, the repetition frequency of the algorithm depends only by the refresh rate and the settling time of the I/O. Acquiring sensors and setting actuators could be performed in series or parallel. When configured in series one iteration lasts the sum of all refresh and settling times. When configured in parallel, the repetition period is equal to the sum of the slowest sensor and actuator response. The repetition frequency can be tuned further by acting on a time factor that compress or expand all settling times.

Feedback Algorithm

The response matrix calculation is performed by the MIMOFB. At the end of the procedure the user can choose to load the new response matrix or just save it. In the response matrix inversion, the SVD singular values can be weighted and weights can be saved.
When the feedback is running, users can activate an adaptive procedure that improve the matching between the response matrix and the real machine. Basically, after a given feedback iteration, one actuator at a time performs a kick. The corresponding perturbation is acquired by the sensors to update the corresponding part of the response matrix that will be afterwards inverted for the next feedback step.

A procedure assists the user to estimate what is the optimal kick amplitude used for the response matrix calculation and at the start of the optimization procedures. The optimal kick amplitude is a tradeoff between a step that increases the *rms* of the sensor readings by at least 50% and the minimum step that drive at least one of the sensors out of range.

The closed loop algorithm is a proportional-integral-derivative controller (PID). The user can switch between the PID default configuration and a predefined one that the closed loop gain and guarantee a better feedback stability. Users can enable the anti-windup, the dead-band control and the slew-rate control on the actuators.

Optimization Algorithms

At the moment there are four model-less optimization algorithms available:

- 1D scan, one parameter
- 2D scan, two parameters
- Simultaneous Perturbation Stochastic Approximation (SPSA) [5], multi-parameter
- Nelder Mead [6], multi-parameter.

In the near future many model free algorithms provided by the NLOpt library [7] will be integrated in the server.

The optimization algorithm could run one shot or, by means of a programmable optimization scheduler (see Fig. 1), cycle between different combinations of optimization algorithms and actuators. For NM and SPSA the user can limit the maximum number of iterations or seconds per optimization. During execution, if no improvement occurs within one third of the remaining time, the optimization stops.

Monitoring and Recovery

The MIMOFB stops in fault state when sensor/actuator values and their statuses are out of allowed values for a number of consecutive cycles. Afterwards, the feedback/optimization can automatically restart once sensor and actuators are back in range.

When enabled, the processing automatically disables the sensors whose value remain constant for more than two samples or equal to zero.

If the sensor value is an average of more then one sample (mean/median), a threshold fixes the maximum number of outliers before marking the sensor invalid.

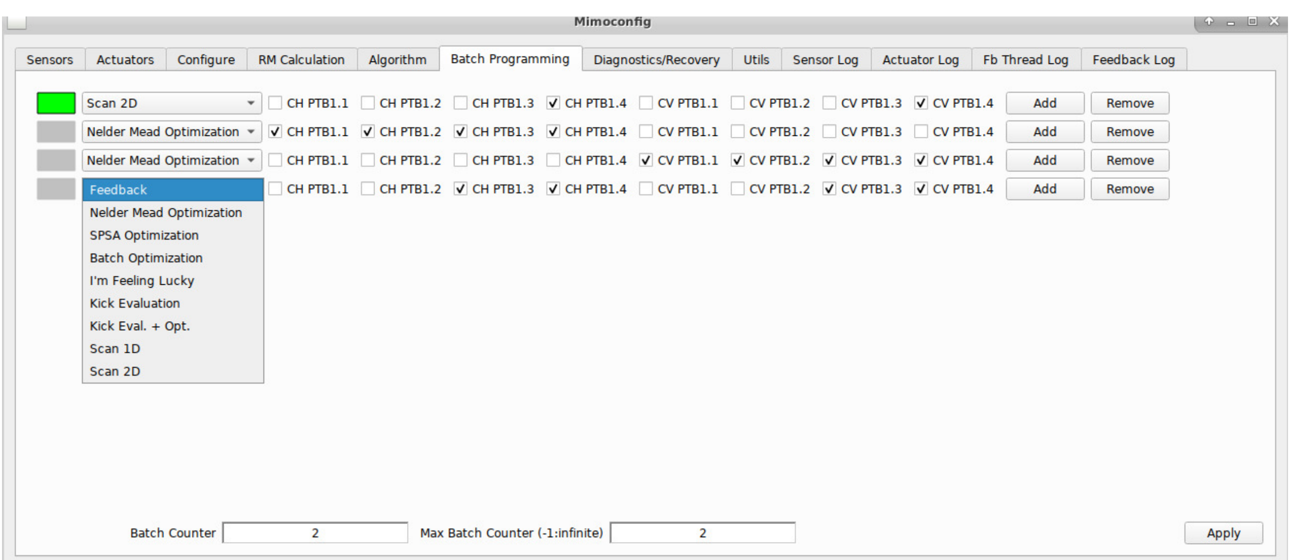

Figure 1: Configuration panel of the MIMOFB. The “Batch Programming” tab configures the optimization scheduler (example). CHV_PTBX.X are corrector magnets of a transfer-line. The optimization target is the booster accumulated current. The “I'm Feeling lucky” optimization mode couples randomly actuators and optimization algorithms.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV008

Sensor and actuator data and error messages are stored into dedicated buffers (see Fig. 2). Similarly, the optimization / feedback engine has a dedicated logging buffer for saving error and debugging messages.

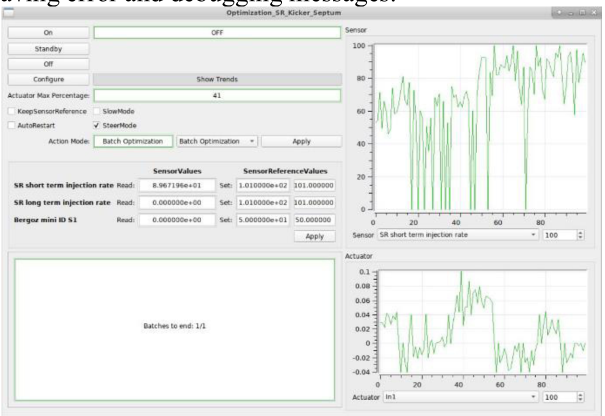

Figure 2: MIMOFB main control panel with logging buffers.

Sequencers Integration

The MIMOFB can be configured to execute a sequence of instructions, based on sequencers [1], at the start and at the end of the correction / optimization process.

Each sensor reading can be preceded and followed by the execution of a sequencer. Similarly, a sequencer can be executed before and after the setting of an actuator. In all cases the correction/optimization process waits sequencers to complete their task before going on. If the sequencer will end its task in fault state then the MIMOFB itself will fail.

The sequencers greatly expand the capabilities of the MIMOFB. A significant example is the alignment procedure, based on MIMOFB, of an electron/laser beam based on intercepting fluorescent screens. In this case the acquisition of the position of the beam on the screen has to be preceded by the insertion of the screen and followed by its extraction (to let the beam go to the next screen). This mechanical process is demanded to pre and post sensor sequencers that can be easily integrated without touching any internal logic of the MIMOFB. In another example a pre-sensor sequencer selects the most reliable sensor between two by acting on the weights of the sensors.

OPTIMIZATIONS

At Elettra the MIMOFB has slowly replaced all legacy high-level applications used for machine tuning and beam stabilization. MIMOFB implements slow local and global orbit correction in the storage ring, tune feedback, RF path length feedback and booster-to-storage ring transfer-line orbit feedback. However, the most interesting applications of the MIMOFB concern the optimization that, until recently, was performed manually by operators.

In Elettra the machine optimizations are one of the most critical steps in the automatic process that recovers the machine from a faulty state and gives back light to the users.

The optimization process has to be:

- maintainable by the operators
- aware of the machine state
- resilient to external perturbations
- able to restore the machine in case of any failure.

Basically, in all optimization schemes the Nelder-Mead (NM) algorithm together with the programmable optimization scheduler gets the best results. At Elettra the optimizer is used routinely for tuning ([description];[actuators];[algorithm];[target function]):

- pre-injector energy (see Fig. 3, num. 1); one actuator, pre-injector HV power supply; algorithm scan-1D; booster accumulated current
- pre-injector to booster RF phase matching (see Fig. 3, num. 2); one actuator, rf phase shifter; algorithm scan-1D; booster accumulated current
- booster to storage ring transfer line orbit (see Fig. 3, num. 5); four actuators, corrector magnet power supplies; algorithm Nelder-Mead; injection efficiency (see Fig. 3)
- storage ring injection system (Fig. 3, num. 6); four actuators, HV power supplies; algorithm Nelder-Mead; injection efficiency.

The optimizer is also used for tuning during machine preparation or machine physics shifts:

- pre-injector to booster transfer-line orbit (see Fig. 3, num. 3); four actuators, corrector magnet power supplies; algorithm Nelder-Mead; booster accumulated current
- pre-injector to booster transfer-line optics (see Fig. 3, num. 4); two actuators, quadrupole magnet power supplies; algorithm Nelder-Mead; booster accumulated current
- beamline photon flux (see Fig.3, num. 7); four actuators, position/angle at source point (eight correctors); algorithm Nelder Mead; beamline photodiode (see Fig. 4)

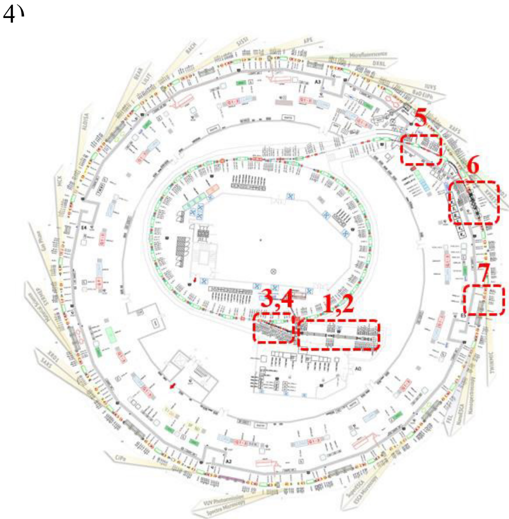

Figure 3: Elettra automatic optimized systems

CONCLUSION

The MIMOFB Tango server has become a key component in the Elettra operations. Thanks to the fact that it is a server, it can be embedded in the high level software framework based on sequencers, which carries out most of the

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV008

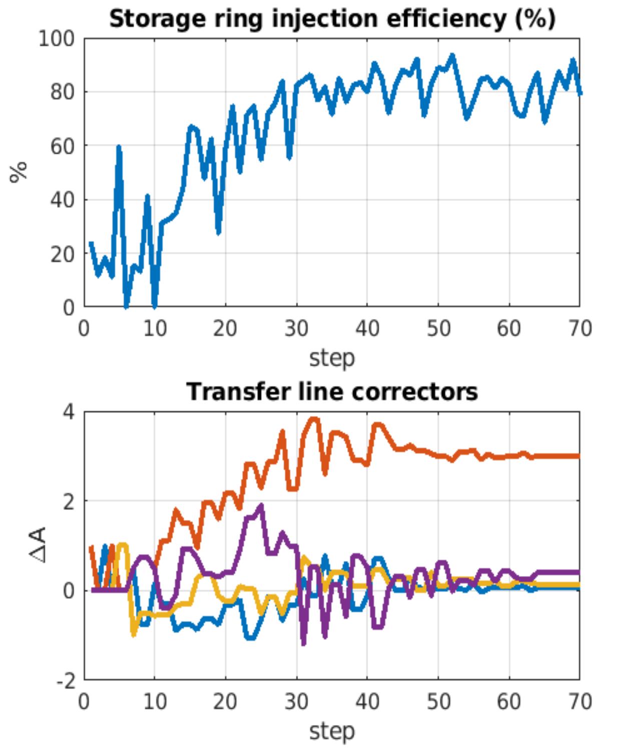

Figure 4: Optimization of the Elettra injection efficiency by changing the last correctors of the booster to storage-ring transfer line

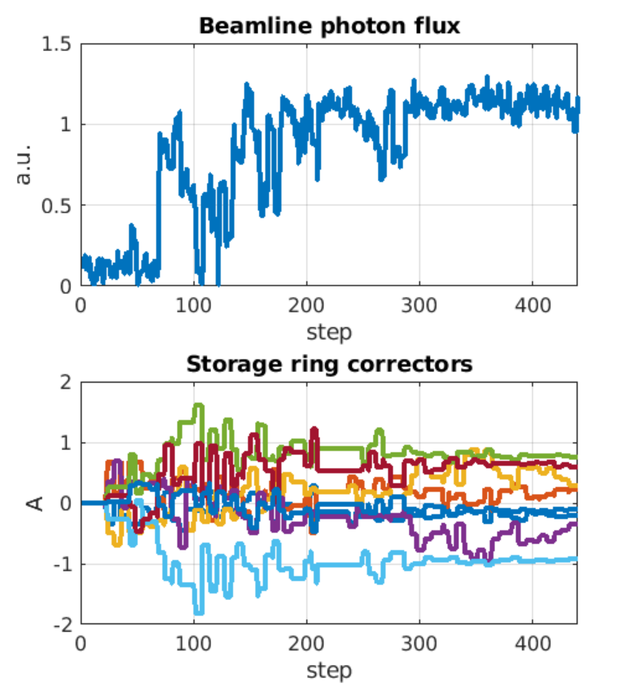

Figure 5: Optimization of the photon beam flux on a beamline by changing position and angle of the electron beam inside the undulator. Eight correctors are involved in the local orbit optimization procedure [8]. The batch programming engine runs the optimization for two cycles. In the second cycle the initial step amplitude was half the one of the first cycle.

tasks that operators should perform manually to prepare the machine and the electron beam for the beamline experiments.

Since the optimization procedures have to run without any human supervision, we have preferred robust and model-less algorithms with a minimum number of hyperparameters to tune. For the same reason we chose to limit to four the number of parameters involved contemporary in any unsupervised optimization process.

Programming batches can execute several combinations of optimization algorithms and different actuators to expand the limits of using just one setup. The Nelder-Mead restart technique [9] that can be easily implemented with batch programming speed up the process of finding the best working point. The necessity for any online algorithm to have a minimum signal for starting, can be mitigated with the possibility to performing 1D-2D wide range scans for signal recovering before running more sophisticated algorithms.

REFERENCES

[1] G. Gaio, P. Cinquegrana *et al.*, "A framework for high level machine automation based on Behavior Trees", presented at the 18th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCSC'21), Oct 2021, paper WEAL02, this conference

[2] E. Piselli and A. Akroh, "New CERN Proton Synchrotron Beam Optimization Tool", in *Proc. 16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'17)*, Barcelona, Spain, Oct. 2017, pp. 692-696. doi:10.18429/JACoW-ICALEPCS2017-TUPHA120

[3] S. Tomin, I. Agapov, *et al.*, "Online optimization of European XFEL with Ocelot", in *Proc. 16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'17)*, Barcelona, Spain, Oct. 2017, pp. 1038-1042. doi:10.18429/JACoW-ICALEPCS2017-WEAPL07

[4] P. Bell, M. Sjöström, *et al.*, "A General Multiple-Input Multiple-Output Feedback Device in Tango for the MAX IV Accelerators", in *Proc. 17th Int. Conf. on Accelerator and Large Experimental Physics Control Systems (ICALEPCS'19)*, New York, NY, USA, Oct. 2019, pp. 1084–1088. doi:10.18429/JACoW-ICALEPCS2019-WEPHA012

[5] J. Nelder and R. Mead, "A Simplex Method for Function Minimization", *The Computer Journal*, vol. 7, Jan. 1965, pp. 308-313. doi:10.1093/COMJNL/7.4.308, corrections in doi:10.1093/comjnl/8.1.27

[6] J. C. Spall, "Multivariate stochastic approximation using a simultaneous perturbation gradient approximation", *IEEE Transactions on Automatic Control*, vol. 37, pp. 332-341, March 1992. doi:10.1109/9.119632

[7] NLopt, https://nlopt.readthedocs.io/

[8] C. J. Bocchetta, D. Bulfone, *et al.*, "First operational results with the ELETTRA fast local feedback system", in *Proc. 17th IEEE Particle Accelerator Conference (PAC'97)*, Vancouver, Canada, May 1997, pp. 2356-2358. doi:10.1109/PAC.1997.751207

[9] Q. H. Zhao, D. Urosević, *et al.*, "A restarted and modified simplex search for unconstrained optimization", *Computers & Operations Research*, vol. 36, 2009, pp. 3263-3271. doi:10.1016/j.cor.2009.03.005