

OMC SOFTWARE IMPROVEMENTS IN 2014

J. Coello de Portugal, F. Carlier, A. Langner,
T. Persson, P. Skowronski, R. Tomás, CERN, Geneva, Switzerland

Abstract

We present the LHC Optics Measurement and Corrections (OMC) software developments done during 2014 on stability, performance and usability. This software is used to analyze turn-by-turn data and compute optics corrections to get the best performance of the LHC. The main developments have been an automatic local correction script to get faster and more accurate corrections in the interaction regions, a self contained test for the whole software package to avoid mistakes during the software development and the improvements in the software quality and efficiency of the Segment-by-segment technique script. We also present a study of the code quality in its current status.

DESCRIPTION OF THE SOFTWARE

The OMC software targets to measure and correct several optics parameters of the LHC. As the machine has very tight tolerances, a good control of the optics is critical for machine protection and performance.

The purpose of the software is to analyze turn-by-turn data measured during the LHC operation. Using an improved Fourier analysis [1] the spectra of the measured signal is obtained and from the spectrum the machine parameters can be computed [2–4]. This data is compared then to the nominal model of the machine and the differences are corrected so it behaves as close as possible to the designed machine.

This set of compiled programs and scripts is gathered into a GUI (*Graphical User Interface*) programmed in Java. The GUI lets the users take advantage of the software suite without having to know in depth its operating details. The chain of operations needed to analyze, compute corrections of turn-by-turn data and upload them to the machine can be done in minutes using the OMC GUI.

AUTOMATIC CORRECTIONS

During the 2012 run, the Segment-by-segment technique was used with good success to locally correct the optics at regions of importance as the interaction points [5–8]. This technique was used to identify beating in the phase advance between BPMs as phase advance is independent of BPM calibration [9]. However, these corrections were done mostly manually, fitting the phase deviation curve by varying the strength of the correctors and rerunning the scripts to watch the results. This method is slow and lacks of precision and thus, an automatic way of computing this correction has been developed.

Segment-by-segment

The Segment-by-segment technique divides the machine in a certain number of smaller segments and treats them as

standalone beam lines. In absence of errors and given the measured initial conditions, the propagation of the phase advance between BPMs using the reference model should behave as that of the measured machine. The differences found in the propagations can be more easily corrected taking only these small segments into account.

Automatic Matching

To perform the automatic correction of the phase advances in a given segment, the Segment-by-segment script has to be run for both beams, as quadrupole magnets which are shared by both beams may also be used for computing corrections. Therefore, for automatic corrections, the matching script in Segment-by-segment needs to run for both beams simultaneously. After that, the MAD-X simulation software [10] is used to match the phase deviations at each measurement point:

1. A set of variables is given to MAD-X, representing the different strengths of the corrector magnets in the chosen segment.
2. The program is provided with a set of constraints, representing the deviations in the phase advance between each BPM in the region of interest:

$$\Delta\phi_{n+1} = \phi_{n+1}^{meas} - \phi_n^{meas} - (\phi_{n+1}^{mod} - \phi_n^{mod}) \quad (1)$$

3. Then, MAD-X will try to find a configuration of the variables that satisfies the constraints as precisely as possible, using a least square minimization technique.
4. Once the phase deviation is reproduced with the desired accuracy, the resulting magnet strength corrections are flipped in sign to counteract that deviation.

If MAD-X finds a configuration of strengths that overpowers some magnet, the user can deactivate it, forcing the matching script to find another configuration. Also, if the measurement in some BPM is thought to be wrong or misleading, the algorithm can be told to ignore this matching point. Furthermore, the algorithm takes the errors in the phase measurements into account, giving less importance to those that have highest errors.

The automatic matching procedure takes around 20 seconds until it finds a solution, performing much faster and more precisely than manually. This new feature is integrated in the OMC GUI (Fig. 1).

TRACKING TEST

As 2014 has been a year of improvements not only in the performance but also in the precision and usability of the

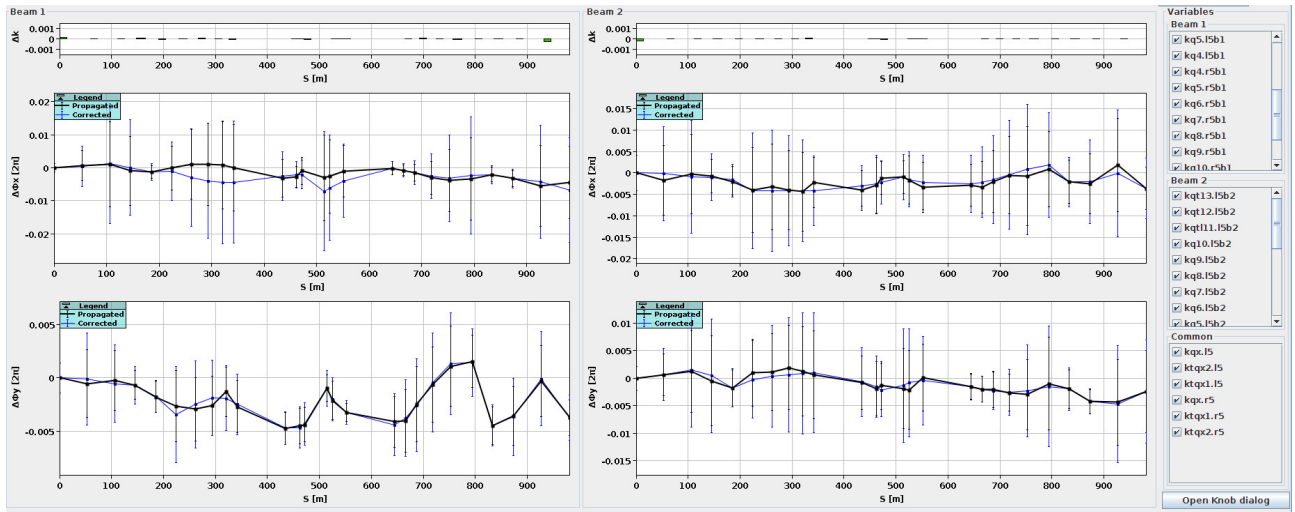


Figure 1: Screenshot of the OMC GUI showing the matching of the phase advances in IP5 performed by the automatic correction script. The top bars show the strength of the computed correction for every magnet, in black is shown the phase advance deviation computed by Segment-by-segment (Eq. 1) and in blue the computed matching result, for beam 1 (left) and beam 2 (right). On the right column are the names of the magnet strength variables that will be used to match.

different parts of the software, the old approach of using templates of previous and trusted results has become inefficient and unsafe. This is why a self-closing testing method had to be developed to make sure that the whole code behaves as designed and avoid bugs during operation.

The tracking test firstly performs a tracking simulation of the machine for some predefined optics, obtaining turn-by-turn data that can be analyzed using the software suite and the Twiss parameters of the machine that can be used to compare the results.

The turn-by-turn data is sent through the different steps of the analysis software (Fig. 2). Once the Twiss parameters are computed, they are compared with the simulation input with a certain tolerance. If manual errors are input into the tracking simulation, these checks are skipped, but the previously described automatic correction algorithm is run, checking as well if the found correction fits the manual errors (Table 1).

Table 1: Errors Input and Found by the Segment-by-segment Matching Script in IP1. There is no error on the rest of the magnets and the value found for them is always around 10^{-8} .

Magnet name	Manual error	Found error
kq8.11b1	2e-5	2.000921225e-05
kqt13.r1b1	3.5e-5	3.52556714e-05
kq5.r1b1	1.5e-5	1.500266706e-05
kq10.11b1	4.5e-5	4.498685456e-05

CORRECTIONS TEST

In the past, once global corrections were performed, there was not automatic ways of checking the real effect of the correction on the machine. This new feature has been inte-

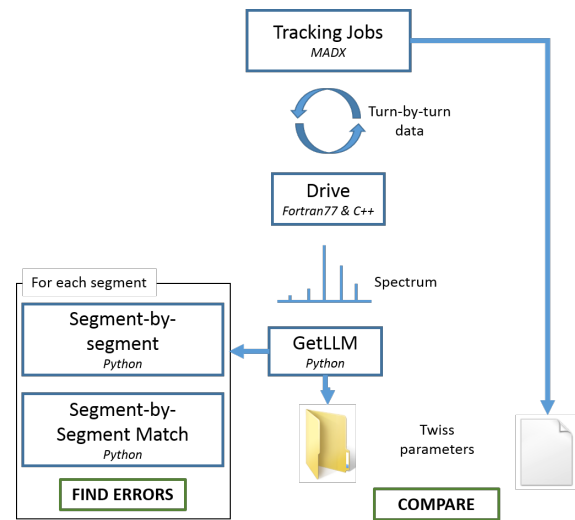


Figure 2: Flowchart of the tracking test with the analysis chain of the tracking data.

grated into the OMC GUI (Fig. 3), letting the users select a measurement, a collection of correction files, and a model of the machine. The corrections will be applied to the nominal model, showing the effect on the ideal machine and the subtracted from the measured values getting a good approximation of the expected effect of the corrections on the machine.

CODE QUALITY

General Code Status

The complexity and size of the code started to become a problem after some years of modifications by several developers.

Content from this work may be used under the terms of the CC BY 3.0 licence (© 2015). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

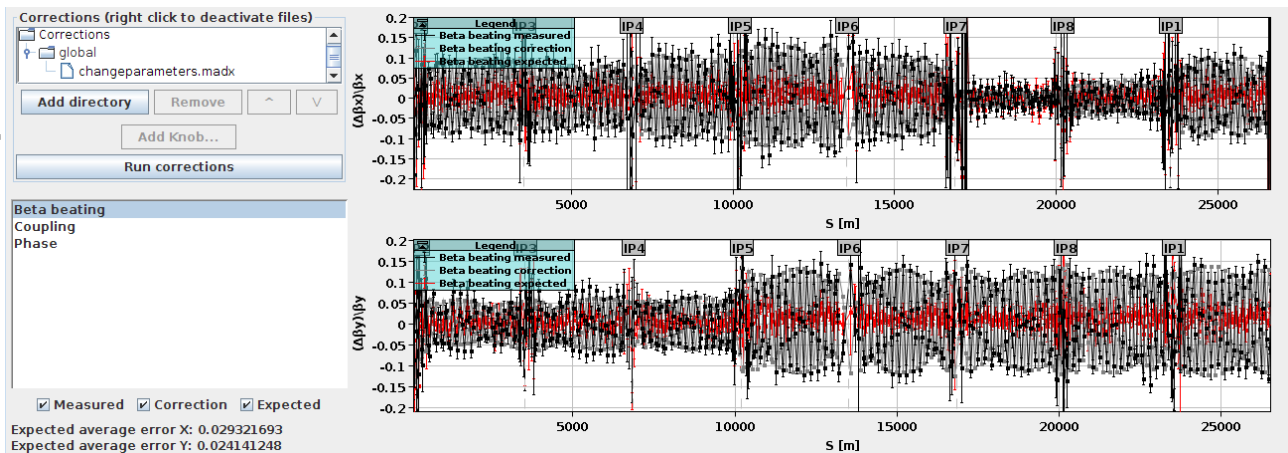


Figure 3: Screenshot of the OMC GUI showing the correction test tool using a global beta-beating knob. The black line represents the measured beta beating, the grey one indicates the effect of the correction acting on the nominal mode, and the expected effect on the machine in red, showing a good cancellation of the beta beating.

The effort started after run one on improving the quality and structure of the code [11, 12], has been continued during 2014 and the beginning of 2015. One of the results of this work is a significant decrease in the number of static analysis problems, even though the length of the code has increased, due mainly to the addition of new features.

In Table 2 is shown the evolution of the code in previous years. As it can be seen, the code length dropped to less than half the length after the first year, removing duplicated and unused code. The static analysis warnings have kept decreasing even though new features have been added.

Table 2: Code Quality Data

Metric	2013	2014	2015
Lines of code	331,312	141,195	166,370
Static analysis issues	479,680	165,531	112,227

Segment-by-segment Improvements

The Segment-by-segment code has been extensively refactored. Previously, the error bars in the propagations of the segment parameters were computed using MAD-X simulations, losing time and precision. All these MAD-X calls have been replaced with propagation formulas increasing speed and stability of the script [13].

The actual computation and error propagation of each relevant parameter, has been separated of the main script, increasing the modularity of the code. The MAD-X template for Segment-by-segment have been refactored as well, using a library with a collection of macros than can be used in the desired order, letting the automatic correction MAD-X template use the same library and thus simplifying the code and avoiding code duplication.

FUTURE TASKS

As the OMC software is result of the work of several people and it was developed in the past without any kind of version control or experts supervision, there still exists a big collection of scripts that are duplicated, lacking the recommended code styling (causing lots of static analysis issues), etc.

The number of different MAD-X templates spread over the code is getting bigger and harder to maintain, being most of them different only in some small parts. As happens with any other kind of code duplication, a slightly modification in the structure needed in the templates can take days of adaptation and testing. All these templates should be removed and replaced by a more sophisticated method of communication with MAD-X.

In the long-term further improvements in the precision and functionalities of the codes are required as LHC operation goes into more demanding regimes with lower β^* , continuing with a challenging High Luminosity upgrade HL-LHC [13].

REFERENCES

- [1] R. Bartolini and F. Schmidt, "SUSSIX: A computer code for frequency analysis of nonlinear betatron motion," 1997.
- [2] P. Castro, J. Borer, A. Burns, G. Morpurgo, and R. Schmidt, "Betatron function measurement at LEP using the BOM 1000 turns facility," *Proceedings of International Conference on Particle Accelerators*, pp. 2–4, 1993.
- [3] P. Castro, *Luminosity and beta function measurement at the electron-positron collider ring LEP*. PhD thesis, 1996.
- [4] A. Langner and R. Tomás, "Optics measurement algorithms and error analysis for the proton energy frontier," *Phys. Rev. ST Accel. Beams*, vol. 18, 2015.
- [5] M. Aiba, S. Fartoukh, A. Franchi, M. Giovannozzi, V. Kain, M. Lamont, R. Tomás, G. Vanbavinckhove, J. Wenninger, F. Zimmermann, R. Calaga, and A. Morita, "First β -beating measurement and optics analysis for the CERN Large Hadron

- Collider,” *Phys. Rev. ST Accel. Beams*, vol. 12, p. 081002, Aug. 2009.
- [6] R. Tomás, O. Brüning, M. Giovannozzi, P. Hagen, M. Lamont, F. Schmidt, G. Vanbavinckhove, M. Aiba, R. Calaga, and R. Miyamoto, “CERN Large Hadron Collider optics model, measurements, and corrections,” *Phys. Rev. ST Accel. Beams*, vol. 13, p. 121004, Dec. 2010.
- [7] R. Tomás, T. Bach, R. Calaga, A. Langner, Y. I. Levinsen, E. H. MacLean, T. H. B. Persson, P. K. Skowronski, M. Strzelczyk, G. Vanbavinckhove, and R. Miyamoto, “Record low β beating in the LHC,” *Phys. Rev. ST Accel. Beams*, vol. 15, no. 9, p. 091001, 2012.
- [8] M. Bai, J. Aronson, M. Blaskiewicz, Y. Luo, S. White, and G. Vanbavinckhove, “Optics Measurements and Corrections at RHIC,” in *Proceedings of IPAC2012*, (New Orleans, Louisiana, USA), 2012.
- [9] R. Tomás, O. Brüning, R. Calaga, S. Fartoukh, A. Franchi, M. Giovannozzi, Y. Papaphilippou, S. Peggs, and F. Zimmermann, “PROCEDURES AND ACCURACY ESTIMATES FOR BETA-BEAT CORRECTION IN THE LHC,” no. June 2006, pp. 23–26.
- [10] “MAD-X,” <http://cern.ch/mad>
- [11] V. Maier, *Software Quality Improvement of the OMC Team at CERN*. PhD thesis, 2014.
- [12] T. Bach and R. Tomas, “Improvements for Optics Measurement and Corrections software,” Aug 2013.
- [13] A. Langner, J. Coello de Portugal, P. Skowronski, and R. Tomás, “Developments of the Segment-by-Segment Technique for Optics Corrections in the LHC,” in *these proceedings, IPAC’15*, (Richmond VA, USA), 2015.