EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH
CERN – ACCELERATORS AND TECHNOLOGY SECTOR

CERN-ATS-2012-146

Evolution of MAD-X in the framework of LHC upgrade studies

L. Deniau, A. Latina, CERN, Geneva, Switzerland

Abstract

The design efforts for the High Luminosity upgrade of the Large Hadron Collider (HL-LHC) will require significant extensions of the MAD-X code widely used for designing and simulating particles accelerators. For this purpose, several new capabilities have been added to the code, namely the possibility to simulate crab cavities for crossing angle compensation, with their imperfections; and the upgrade of the interface to Sixtrack used for distributed tracking with, e.g., LHC@home. These changes are framed into a global redesign of the MAD-X architecture meant to consolidate its structure, improve its performances, and increase its robustness and flexibility. Such improvements are described in details in the present paper.

Presented at the International Particle Accelerator Conference (IPAC'12) –

May 20-25, 2012, N. Orleans, USA

Geneva, Switzerland, May 2012

Evolution of MAD-X in the framework of LHC upgrade studies

L. Deniau, A. Latina, CERN, Geneva, Switzerland

Abstract

The design efforts for the High Luminosity upgrade of the Large Hadron Collider (HL-LHC) will require significant extensions of the MAD-X code widely used for designing and simulating particles accelerators. For this purpose, several new capabilities have been added to the code, namely the possibility to simulate crab cavities for crossing angle compensation, with their imperfections; and the upgrade of the interface to Sixtrack used for distributed tracking with, e.g., LHC@home. These changes are framed into a global redesign of the MAD-X architecture meant to consolidate its structure, improve its performances, and increase its robustness and flexibility. Such improvements are described in details in the present paper.

INTRODUCTION

MAD is a project with a long history, aiming to be at the forefront of computational physics in the field of particle accelerator design and simulation. The MAD scripting language is de facto the standard to describe particle accelerators, simulate beam dynamics and optimize beam optics.

MAD-X is the successor of MAD-8 and was first released in June, 2002 [1]. It offers most of the MAD-8 functionalities, with some additions, corrections, and extensions [2]. The most important of these extensions is the Polymorphic Tracking Code (PTC) of E. Forest [3].

A decade after its first release, MAD-X is still the main tool used to design and simulate accelerators at CERN. But its original design was mainly focusing on the urgent needs for the LHC, and a large part of the code was inherited from old software written in the 80's. The framework of the LHC upgrade studies is a good opportunity to reorganize and upgrade the overall code of MAD-X to support recent hardware (64 bit, multicores) and new needs.

In parallel to long-term improvements, the project must continue to evolve toward the users' needs and be able to add new functionalities, like the two recent optical elements added for modeling thin RF-multipoles and thin nonlinear lenses with elliptical potential.

PROJECT IMPROVEMENT

The MAD-X project is undergoing a sequence of changes aiming to improve its organization before important developments can be relaunched, Fig. 1. The final objective of this sequence is the delivery of a new production release passing all the test cases and registered studies.

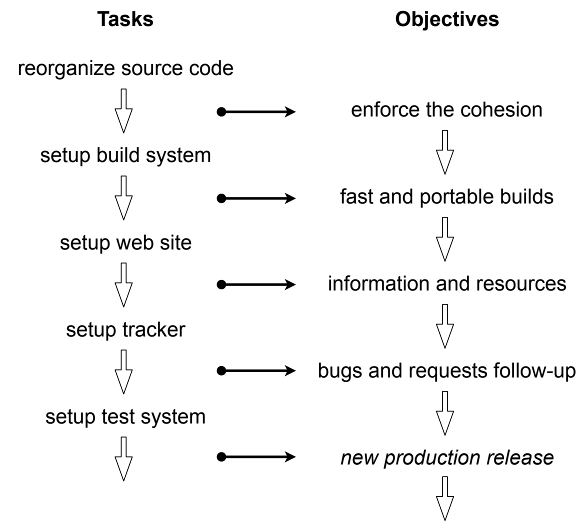

Figure 1: Sequence of tasks and corresponding objectives required before starting new MAD-X developments.

Code reorganization

The core of MAD-X was developed quickly, focusing on the design of the LHC and the optimization of its optics. Its architecture was monolithic, blurring any cohesion in the code and making it very hard to understand for human beings. This single file approach prevented proper modular architecture exposing only interfaces; all functions and most variables ended in the global namespace.

The first major task to rehabilitate the project was to split the thousands line of C code into 62 files — one module per concern — and to close the visibility of the variables and functions as much as possible; a lengthy ongoing task. The reorganization of the core helped to detect inconsistent functions signature and enforce the cohesion of the code leading to significant bugs correction. Unfortunately, this effort is not enough to get rid of structural problems, namely reentrancy and memory leaks, which will be addressed by future developments.

Portability and builds

Another important aspect of the project was the ability to deliver portably and synchronously the MAD-X releases for all the main platforms — Linux, MacOSX and Windows — almost automatically, and save the resource allocated to this task beforehand.

For this purpose, a new portable and efficient build system has been developed, which handles all the three platforms in 32 bit and 64 bit architectures, and allows to build applications and libraries mixing code written in C, C++, Fortran 77 and Fortran 95. The new build system supports 12 compilers and provides extensible, portable and easy-to-configure makefiles. As a demonstration of its flexibility, it took only a couple of hours to extend the build system to build and distribute the PTC library as part of the collaboration with the PTC-ORBIT project [4].

Website and Tracker

The MAD-X website [5] has also been completely redesigned to provide to the community an easy access to the information and materials — documentations, examples, accelerators optics and new releases for download — and the online manual will be improved during the summer.

The Trac web application [6] was setup and fed with all the project bugs and requests to trace the history and to improve the follow-up of the pending tickets. An invaluable tool that allows to evaluate the (non-)convergence of bug corrections, to classify the issues, and to draw the best strategies for long-term improvements.

Test system

Before relaunching large scale development, it is essential to equipped MAD-X with an efficient and robust test system. It will give the confidence that improvements are incremental, and it will avoid unexpected feature regressions or undesirable backward incompatibilities.

The new test system is implemented as an extension of the new build system, taking advantage of its portability and its flexibility. It heavily relies on `numdiff`, a new tool developed for running test suites that compare files with scientific content, where text and numbers are arbitrarily mixed. This tool is able to deal with global and local user defined constraints on tolerances for numerical comparisons, and understand complex specifications with both, numerical precision and physical accuracy. This avoid spurious alarms to be reported from tests run on different platforms or coming from different build configurations.

NEW FUNCTIONALITIES

RF-Multipole

Motivation The electromagnetic field of accelerating structures and crab-cavities can exhibit undesired transverse field components due to asymmetries in the azimuthal direction of the element geometry, for instance in the input and output power couplers. Tracking simulations are performed to evaluate the impact of such transverse RF-kicks on the beam dynamics.

Model The RF-field is decomposed into pulsed normal and skew components similarly to the multipolar field decomposition used to model the magnetic elements [7]:

$$C_n = B_n + i\,A_n. \tag{1}$$

The RF-multipolar coefficients oscillate at the same angular frequency ω_{RF} as the generating electromagnetic field, and vary with the relative longitudinal position $z = c\,t$ of the particle experiencing the kick to the synchronous particle. Hence, the formalism of Eq. 1 can be rewritten as:

$$\tilde{C}_n(z) = \tilde{B}_n(z) + i\,\tilde{A}_n(z), \tag{2}$$

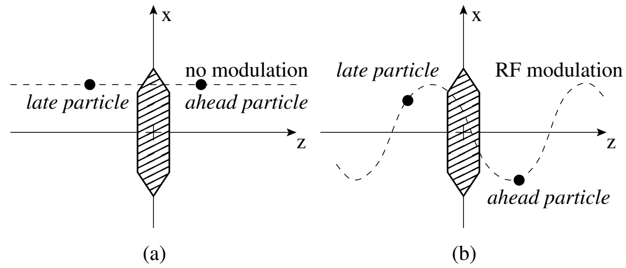

Figure 2: In an ordinary multipole (a), each particle experiences a kick that depends only on its transverse coordinates; in a RF-multipole (b), the strength of the kick is also a function of the relative longitudinal coordinate z.

with

$$\begin{aligned} \tilde{B}_n(z) &= \mathrm{Re}\left[B_n\, e^{i(\varphi_n - k_{\mathrm{RF}} z)}\right], \\ \tilde{A}_n(z) &= \mathrm{Re}\left[A_n\, e^{i(\vartheta_n - k_{\mathrm{RF}} z)}\right], \end{aligned} \tag{3}$$

where k_{RF} is the RF wave number ($k_{\mathrm{RF}} z = \omega_{\mathrm{RF}} t$), and ϑ_n and φ_n are the n-th skew and normal phases. Hence, to characterize a RF-multipole element, it is necessary to specify the RF frequency f_{RF}, plus the amplitude and the phase of each normal and skew harmonic:

$$\text{RF-Multipole:} \begin{cases} f_{\mathrm{RF}} & \textit{RF frequency}, \\ C_n = B_n + i\,A_n & 2\times n \textit{ coefficients}, \\ \phi_n = \vartheta_n + i\,\varphi_n & 2\times n \textit{ phases}. \end{cases}$$

Physics The Hamiltonian of an ultra-relativistic particle traversing a thin RF-Multipole is similar to that of an ordinary thin multipole with RF modulation due to the rotating normalized harmonics, plus a zero-order term for the longitudinal acceleration [8]:

$$\begin{aligned} H = &-\frac{V}{k_{\mathrm{RF}}}\cos(\vartheta_0 - k_{\mathrm{RF}} z) + \\ &+ \sum_{n=1}^{N} \frac{1}{n+1} \mathrm{Re}\Big[\Big(B_n \cos(\vartheta_n - k_{\mathrm{RF}} z) + \\ &+ i A_n \cos(\varphi_n - k_{\mathrm{RF}} z)\Big)(x+iy)^{n+1}\Big] \end{aligned} \tag{4}$$

From this expression one can derive the kick experienced by a particle at coordinates (x, y, z) with respect to the reference particle (Fig. 2):

$$\begin{aligned} \Delta p_x &= -\frac{\partial H}{\partial x} = -\sum_{n=1}^{N} \mathrm{Re}\Big[\Big(B_n \cos(\vartheta_n - k_{\mathrm{RF}} z) + \\ &\quad + i A_n \cos(\varphi_n - k_{\mathrm{RF}} z)\Big)(x+iy)^{n}\Big] \\ \Delta p_y &= -\frac{\partial H}{\partial y} = \sum_{n=1}^{N} \mathrm{Im}\Big[\Big(B_n \cos(\vartheta_n - k_{\mathrm{RF}} z) + \\ &\quad + i A_n \cos(\varphi_n - k_{\mathrm{RF}} z)\Big)(x+iy)^{n}\Big] \\ \Delta p_z &= -\frac{\partial H}{\partial z} = V \sin(\vartheta_0 - k_{\mathrm{RF}} z) - \\ &\quad - k_{\mathrm{RF}} \sum_{n=1}^{N} \frac{1}{n+1} \mathrm{Re}\Big[\Big(B_n \sin(\vartheta_n - k_{\mathrm{RF}} z) + \\ &\quad + i A_n \sin(\varphi_n - k_{\mathrm{RF}} z)\Big)(x+iy)^{n+1}\Big] \end{aligned} \tag{5}$$

Implementation A new thin element RFMULTIPOLE has been added to the Twiss and Track modules of MAD-X. This command accepts all the attributes of the existing command MULTIPOLE (i.e. L, LRAD, TILT, KNL, KSL), augmented with the quantities inherited from an RF element (i.e. RMF_XXX parameters), plus the normal and skew phases arrays PNL and PSL specific to this element. The syntax of the command to create a RF-multipole is [5]:

```
RFMULTIPOLE,
 L=real, LRAD=real, TILT=real,
 RFM_VOLT=real, RFM_LAG=real,
 RFM_HARMON=integer, RFM_FREQ=real,
 KNL = { knl0, knl1, ... }, ! Normal coefficients
 KSL = { ksl0, ksl1, ... }, ! Skew   coefficients
 PNL = { pnl0, pnl1, ... }, ! Normal phases [2pi]
 PSL = { psl0, psl1, ... }; ! Skew   phases [2pi]
```

RF-Multipoles as field imperfections RF-multipolar field errors can be attached to any magnetic element using the command EFCOMP. For the time being, this feature is not yet fully operational and only stores the information for future use. The syntax of the command is [5]:

```
EFCOMP, ORDER=integer, RADIUS=real,
 DKN  = { dkn0 , dkn1 , ... },
 DKS  = { dks0 , dks1 , ... },
 DKNR = { dknr0, dknr1, ... },
 DKSR = { dksr0, dksr1, ... },
 ! extra parameters available for RF-field errors
 RFM_FREQ=real, RFM_HARMON=integer, RFM_LAG=real,
 DPN = { dpn0, dpn1, ... },
 DPS = { dps0, dps1, ... };
```

Export to Sixtrack For the tracking needs of the LHC, the Sixtrack code is routinely used [9]. The data loaded for simulations are provided by the MAD-X interface, which has been extended to export the RF-Multipoles. Sixtrack currently accepts only quadrupoles, sextupoles, and octupoles. Therefore, a RF-Multipole in MAD-X is converted into and up to three equivalent thin-elements in Sixtrack and higher order terms are discarded. Simulations of RF-Multipoles associated to crab-cavities are ongoing for LHC upgrade studies, and the first results are presented in [10].

It must be noted that the RF-multipole of order 0, namely the RF-dipole component, is purposely ignored by the interface as it is equivalent to a crab-kick, and should be exported using the appropriate MAD-X element.

Intrabeam Scattering

The main improvement to the IBS implementation in MAD-X was to correct an error coming from multiplying matrices with Mathematica, which led to a loss of the terms $6\beta_x^2\phi_x^2/(H_x\epsilon_x)$ and $6\beta_x^2\beta_y\phi_x^2/(H_x\epsilon_x\epsilon_y)$ in the expressions for a_x and b_x [11]. There were also some other minor issues; the dispersion in the MAD-X twiss table is defined as $\Delta x/(\beta\Delta\delta)$ with $\beta = v/c$, which differs from the standard convention for β smaller than 1. And we also now distinguish the rms relative energy spread $\Delta E_{\rm rms}/E$ from the rms relative momentum spread, $\delta_{\rm rms}$ according to $\sigma_\delta \equiv \delta_{\rm rms} = (\Delta E_{\rm rms}/E)/\beta^2$.

Nonlinear lenses

The new NLLENS element represents a thin nonlinear lens with the potential of 'Elliptic' type as specified in [12]. The lens is used to create fully integrable 2D nonlinear accelerator lattice with very large nonlinear tune spread/shift. The NLLENS element is recognized by the thin tracking module of MAD-X and the quadrupole term of the potential is included in the transport map computation and effects the calculation of tunes and Twiss functions.

CONCLUSIONS AND OUTLOOK

MAD-X is undergoing its first renovation process since years, a process that is meant to improve the code quality and flexibility and to restore the communication within the MAD-X community. Its maintenance and portability have been simplified and strengthen, and a bug tracking system has been setup. Some efforts are also devoted to implement new elements in the legacy code, like the RF-Multipole necessary to evaluate the detrimental impact of the crab-cavities on the luminosity in the LHC upgrade scenarios. By mid-2012, MAD-X should be ready to sustain new large scale developments for the benefit of its users.

ACKNOWLEDGMENT

We would like to thank Frank Schmidt for his invaluable contribution to the project as the MAD-X custodian over the decade 2002–2011, R. De Maria for the fruitful discussions on the RF-multipole and MAD-X, and M. Giovannozzi for his useful feedback on this paper.

REFERENCES

[1] H. Grote and F. Schmidt, *"MAD-X : An Upgrade from MAD8"*, PAC'03, Portland, USA, May 2003.

[2] F. Schmidt, *"MAD-X a worthy successor for MAD8?"*, Nucl. Instrum. Methods Phys. Res., A 558, (1) 2006.

[3] E. Forest, F. Schmidt and E. Mcintosh, *"Introduction to the Polymorphic Tracking Code"*, KEK Report, 2002.

[4] E. Forest, *et* al., *"Synopsis of the PTC and ORBIT Integration"*, KEK Report, 2008.

[5] MAD-X website `http://cern.ch/madx`

[6] MAD-X Trac `https://svnweb.cern.ch/trac/madx`

[7] S. Russenschuck, *"Field Computation for Accelerator Magnets"*, Wiley-VCH Verlag, Weinheim, 2010.

[8] R. De Maria, *"Symplectic integrator for a relativistic particle in a RF cavity"*, BE/ABP–LCU meeting, April 2011.

[9] F. Schmidt, *"Sixtrack version 1.2"*, CERN-SL-94-56, 1994.

[10] A. Grudiev, *et* al., *"Study of Multipolar RF Kicks from the Main Deflecting Mode in Compact Crab Cavities for LHC"*, IPAC'12, New Orleans, May 2012 (TUPPR027).

[11] F. Antoniou and F. Zimmermann, *"Revision of IBS with Non-Ultrarelativistic Corrections and Vertical Dispersion for MAD-X"*, Report CERN-ATS-2012-066, March 2012.

[12] A. Valishev *et* al., *"Ring for Test of Nonlinear Integrable Optics"*, New York, NY, USA, 2012, WEP070.