

EFFECT OF SELF-CONSISTENCY ON PERIODIC RESONANCE CROSSING

G. Franchetti*, GSI, Darmstadt, Germany

Abstract

In high intensity bunched beams resonance crossing gives rise to emittance growth and beam loss. Both these effects build up after many synchrotron oscillations. Up to now long-term modeling have relied on frozen models neglecting the physics of self-consistency. We address here this issue and present the state of the art of simulations applied to the SIS100.

INTRODUCTION

The phenomenon of resonance crossing [1, 2, 3] can be induced by space charge in bunched beams [4]. The simulation of the beam evolution when resonance crossing cannot be avoided poses extraordinary challenges for computer modeling of long-term storage. The computation of Coulomb forces, usually performed via particle in cell (PIC) algorithms, unavoidably produces a noise on the macro-particle dynamics. Studies on the effect of this noise [5] have shown that a significant emittance growth can arise from PIC codes.

For short term simulations, where the effects of self-consistency created by space charge (coherent resonances, instability of coherent modes, etc.) are very fast, this noise does not play a role, as it has not time to build up. Different is the case for a beam dynamics that drives an emittance growth after long-term. In particular on the phenomenon of the space charge induced periodic resonance crossing, the extraction of particles from the beam happens slowly, and the small growth rate can significantly be affected by simulation code spurious effect as the noise induced by PIC algorithms.

The level of noise in PIC simulations depends on the number of macro-particles per PIC cell. Statistical fluctuations scale as $1/\sqrt{N_c}$, with N_c the number of macro-particles in a cell. Therefore the reduction of these unwanted effects is obtained by raising the number of macro-particles used to model the bunch. Hence, the prize to pay for controlling the noise is an increased CPU time required to perform the simulations. Therefore simulations on a time scale of $10^5, 10^6$ turns are not feasible with PIC algorithms.

For this reason in the studies performed till now (see for example Ref. [6]) the Coulomb force has been computed by assuming a beam distribution frozen. In this approach a frozen Coulomb force is used for tracking “test” macro-particles in the accelerator structure. This approach relies on the assumption that macro-particle loss is small (maximum of 10%). For beam loss larger than this value, simulation predictions are not reliable.

Given the substantial approximation made, benchmarking of code predictions with experiments has been performed. The benchmarking had the purpose of verifying/confirming the underlying mechanism, and to verify the accuracy of code predictions [4, 7].

At practical level the necessity of making beam loss prediction is very important for the SIS100 synchrotron in order to consolidate the effectiveness of resonance compensation schemes [8, 9]. Uncontrolled beam loss is required to be within a 5% budget in order to mitigate a progressive vacuum degradation, dangerous for beam lifetime. Therefore the study of the effect of self-consistency is relevant for the assessment of effective beam loss, crucial quantity in the discussion on the nonlinear components in magnets, residual closed orbit distortion as well as in the resonance compensation strategy.

LESSONS FROM THE MACHINE EXPERIMENT EXPERIENCE

Two benchmarking campaigns have been performed till now: the first in the CERN-PS in 2003, and later at GSI using the SIS18 in 2008-2010. In both the experiments the lattice was modeled at the best of the available informations. In Fig. 1 we report the main experimental results of the GSI campaigns, and the associated simulation results. Details and discussion of the experiment and its parameters are reported in Ref. [7]. We note that the smaller beam survival is found to be of $\sim 20\%$. The simulations instead show a minimum beam survival of $\sim 50\%$. The discrepancy of these two results is not fully understood. While on one hand it is not clear of whether the machine modeling is complete, on the other hand, the effect of the self-consistency is not included in the simulations as a frozen model is used. In Ref. [7] it was concluded that the discrepancy might be attributed to the incomplete modeling of the self-consistency in the computer code.

A GLIMPSE TO THE FUTURE

A relevant application of the frozen model is in the FAIR project [10]. The SIS100 will certainly be afflicted by a web of resonances created by superconducting magnet nonlinear components, closed orbits misalignment, and random errors [6]. In Fig. 2a is shown for a possible model of the SIS100 the resonance web, which is formed by integer, half integer, third and fourth order normal and skew resonances. These resonances are found via tune scans of the short-term dynamic aperture (1000 turns). Beam survival for several intensities after one second storage are shown in Fig. 2b. The maximum intensity corresponds to 0.625×10^{11} ions/bunch, which creates a large tune-shift represented schematically in the picture. The space

* g.franchetti@gsi.de

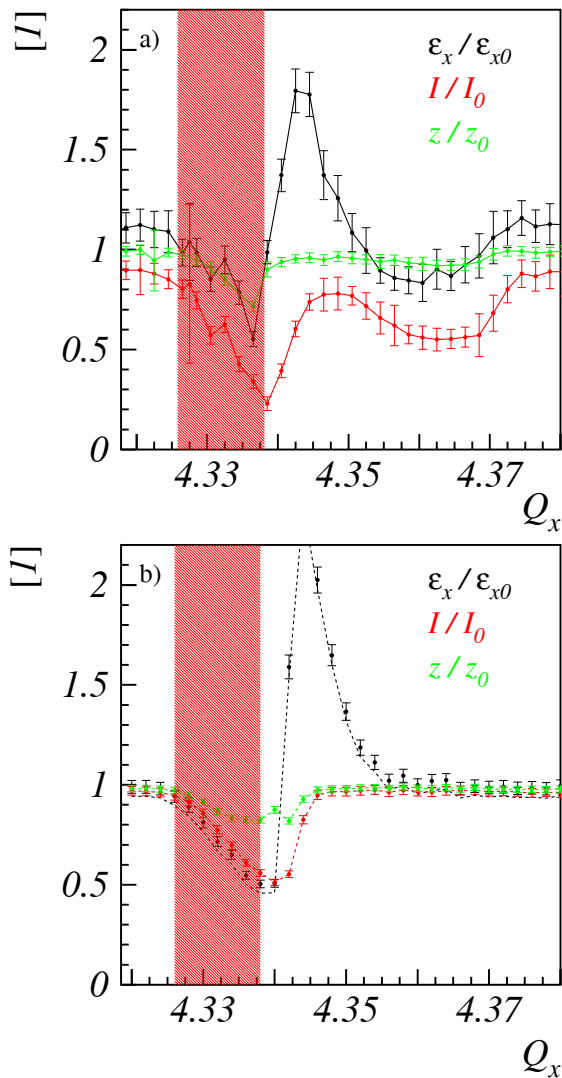


Figure 1: Emittance growth and beam survival after 1 second storage. In picture a) the experimental results are shown as function of working points around the third order resonance. In picture b) are shown the corresponding simulations.

charge tune-spread overlaps with 4 resonances hence several macro-particles will cross one or more lattice resonances, therefore periodic resonance crossing is taking over the particle dynamics. Note that the red curve is really losing almost all the beam. This effect is clearly intensity dependent as shown by the black curve that has intensity 0.125×10^{11} in which beam survival is much better with almost no beam loss.

It is clear that when frozen simulations lose 90% of the beam, this prediction severely suffers from the lack of some space charge update in the code. Note, however, that formally the problem of beam survival can be faced with regard of its source that in this case is the presence of machine resonances. Therefore a first approach is to compensate to some extent the resonances in order to see what

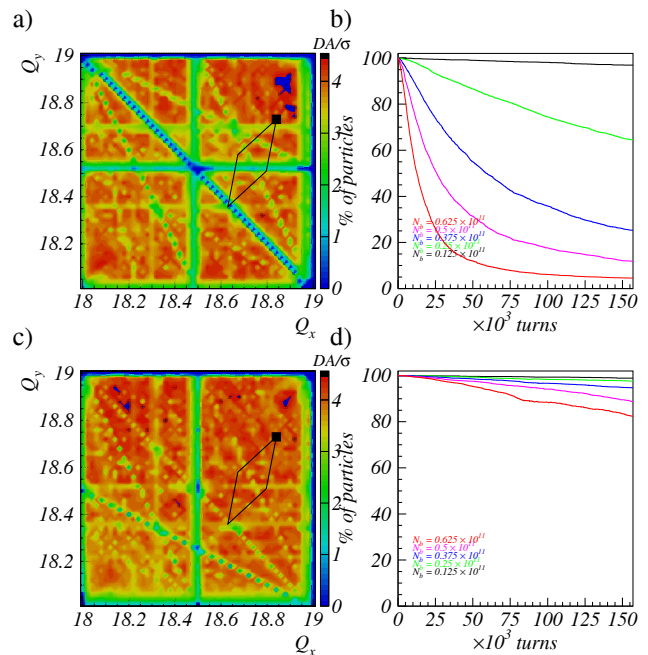


Figure 2: Simulations for SIS100: on the top pictures are simulated the working diagram (left), and beam survival for several intensities (right). In the bottom pictures: same simulations but with the resonances overlapping the tune-spread compensated.

happen to the beam survival. In Fig. 2c is shown again the working diagram with now the effect of an “ad hoc” activated compensation system, still with correcting element located in the actual position of those foreseen in SIS100. The corresponding beam survival is comforting (Fig. 2d) as the beam loss appears significantly mitigated. However, it is not clear if the effect of the self-consistency is of relevance or not to this prediction.

THE CLOSE TO THE RESONANCE COLLAPSE

Let consider now a situation in which the space charge tune-spread crosses only one resonance. For a beam with size not too small with respect to the beam pipe, the main effect of the periodic crossing of one resonance is to produce a slow particle loss. In a simulation that uses a frozen space charge model, all particles that cross that resonance will eventually be lost (after long time). If after the saturation has been reached, in the frozen model, we would update the beam intensity, particles that before did not cross the resonance would now cross it leading to more beam loss. This reasoning shows that a self-consistent process necessarily brings new particles to cross the resonance.

Beam loss stops when particles are not able to reach the beam pipe, and the diffusion of particles is limited to the phase space area spanned by the islands of the frozen system, which outer location is determined by the space charge tune-shift. Approximately, beam loss stops when the space charge tune-spread does not overlap anymore with the resonance stop-band under consideration.

A foreseeable consequence is that the most dangerous working point is the one closer and “above” the resonance (i.e. allowing the space charge tune-spread overlapping with the resonance). In fact, in this condition the distance from the resonance is very small, hence almost all particles in a bunch should be lost in “a close to the resonance collapse” in order to stop the avalanche beam loss triggered by the self-consistency. In the last part of this proceeding we will study the beam behavior in this regime.

MARKOVIAN UPDATE

We propose here as intermediate step to the treatment of the self-consistency an approximated approach. We intend to keep as feature of the tracking algorithm to be noise free, but we also want to incorporate some feature of the self-consistency. We will use the following ansatz: *at each integration time step we update in our frozen model only the intensity and leave unchanged the frozen bunch emittances and frozen particle distributions. We call this algorithm Markovian update.* This ansatz is certainly approximated as it assumes that particles are lost from everywhere inside the beam. The name Markovian is used because this type of update creates a loss of memory. In fact, after the integration step n , the beam evolves as if it started at step $n = 0$ but with the intensity found at the end of step n .

From a simulation point of view, even adopting this procedure, it arises the issue of how many particles should be used. In fact, in order to describe a continuous process of beam loss the number of macro-particles should be large enough to allow such a description. The simulations shown Fig. 2bd relied on splitting the work load among many processors in which the same simulation is run but with differently seeded macro-particles. For the simulations in Fig. 2bd, 750 processors were used and each of them tracked 4 macro-particles for a total of 3000 macro-particles tracked. Each beam intensity curves in Fig. 2bd is obtained as averages of all the 750 beam surviving curves obtained from each single processor simulation. If we apply the Markovian update to a single processor simulation that tracks only 4 macro-particles we certainly cannot expect a smooth beam loss process as when one macro-particle is lost that corresponds to an abrupt change of 25% of that single processor simulation beam intensity.

Keeping this in mind we explore the response of the Markovian update for several number of single processor macro-particle tracked. We considered the single processor tracking of 4,10,20,100 macro-particles, with a number of processors consistent with a total tracking of 3000 macro-particles. In Fig. 3 we show the results of this series of simulations for the case of the maximum intensity of SIS100 (i.e. 0.625×10^{11} ions/bunch). The picture shows surprisingly that the beam survival curves for the several cases bundle together almost regardless the number of single processor macro-particles used. This result is at the moment not fully understood, and its explanation is part of

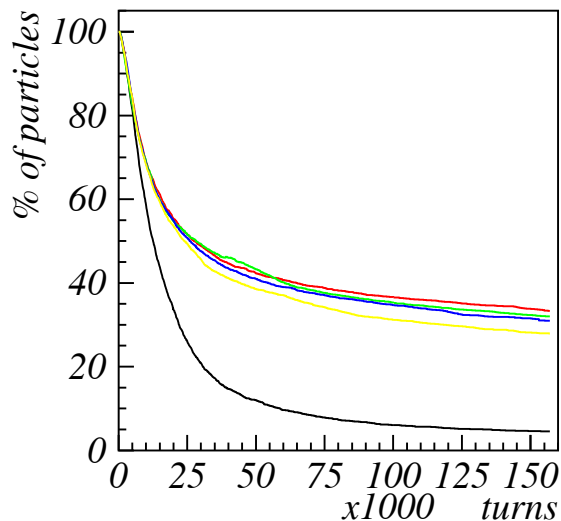


Figure 3: Beam survival for a Markovian update simulation with single processor number of macro-particles of 4 (yellow), 10 (blue), 20 (green), and 100 (red). The number of processors used in each case is to obtain a total of 3000 macro-particles tracked. The black curve shows the beam survival obtained with the frozen space charge.

an ongoing study. Nevertheless, the results of Fig. 3 show distinctly the effect of the Markovian update. The beam survival in SIS100 is found $\sim 30 \div 40\%$, which is better than the previous $\sim 5\%$ obtained with a frozen model. We omit here a discussion on the underlying physics responsible of this difference as it arises by a simultaneous crossing of 4 different resonances (see Fig. 2a), and this analysis goes beyond the purpose of this proceeding.

MARKOVIAN MAPPING

We now further develop the concept of Markovian update. Suppose we can “fit” the evolution of the beam intensity of a frozen space charge simulation with an intensity I given by the differential equation

$$\frac{dI}{dt} = -\Delta(I_0)f\left(\frac{t}{\tau(I_0)}\right)\frac{1}{\tau(I_0)}, \quad (1)$$

where $\tau(I_0)$ is a time constant of the beam loss process which depends on the initial beam current I_0 . Strictly Eq. 1 should be of a second order as the beam tracking with a space charge frozen algorithm preserve the memory. If, ideally, we knew the functions $\Delta(I_0)$, $\tau(I_0)$, and $f()$, then we could construct the Markovian update from the process in Eq. 1 and find the beam intensity I^* by substituting $I_0 \rightarrow I^*$, and by solving

$$\frac{dI^*}{dt} = -\Delta(I^*)f\left(\frac{t(I^*)}{\tau(I^*)}\right)\frac{1}{\tau(I^*)}. \quad (2)$$

Here $t(I^*)$ denotes the time at which each update process takes a frozen evolution of intensity I^* . This is the time

in which the beam loss becomes “steady”: at the beginning of the evolution of $I(t)$ we find $dI/dt < 0$, and later it changes to $dI/dt > 0$. $t(I^*)$ is found solving $d^2I[t(I^*)]/dt^2 = 0$. In the Markovian update of simulations this happens automatically because when the update of intensity is made, the beam loss flow is already “saturated”.

We apply these concepts to the simulations in Fig. 2b. After few attempts we find that the beam survival is decently fit by the function

$$I = \frac{I_0}{1 + [t/\tau(I_0)]^{1.23}}, \quad (3)$$

where the function $\tau(I_0) = 2.5 I_0^{-3.3}$ is shown in Fig. 4 right. In Fig. 4 left it is shown how the curves overlap with the frozen simulations (of Fig. 2b). Differentiating

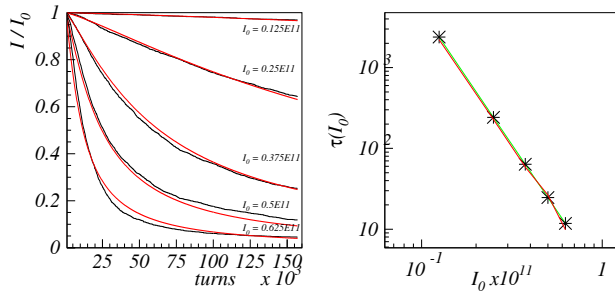


Figure 4: Application of the Markovian mapping procedure: on the left the comparison of the fit beam intensity from Eq. 3 with the simulation results of Fig. 2b. On the right, comparison of the fit function $\tau(I_0)$ with the correspondent values obtained from the simulations (red line).

Eq. 3 with respect to t we find Eq. 1, hence $\Delta(I_0), \tau(I_0)$, and $f()$. We next find $t(I^*)$ solving $d^2I[t(I^*)]/dt^2 = 0$ and finally we solve the Markovian mapping equation Eq. 2 and find

$$I^* = I_0 [1 + 0.87 I_0^{3.3} t]^{-1/3.3}.$$

The comparison of I^* with the result of Markovian update simulations is shown in Fig. 5. In the picture the upper three curves refer to the intensity of 0.25×10^{11} ions/bunch. The green curve is the Markovian mapping, and the red the Markovian update. The Markovian mapping closely follows the Markovian update. The other three curves with higher beam loss are relative to the case of full intensity (0.625×10^{11} ions/bunch). We note that green and red curves do not perfectly overlap, but still are close. The black curve at the bottom is the beam survival for the frozen simulation. This discrepancy is not surprising as it is the result of a somewhat arbitrary fitting curve Eq. 1. It is interesting, however, that although the heuristic approach, the Markovian mapping is not that “wrong” with respect to the Markovian update. This is interesting because this procedure allows retrieving Markovian update results from space charge frozen simulation without actually running

self-consistent simulations. The advantage of this approach is to avoid the use of increased number of macro-particles for describing smooth processes of beam loss and using a fitting procedure associated with an analytic treatment that allows to retrieve Markovian update results. The procedure needs a series of frozen space charge simulation that allows the estimate of $\Delta(I_0), \tau(I_0)$, and $f()$ in Eq. 1.

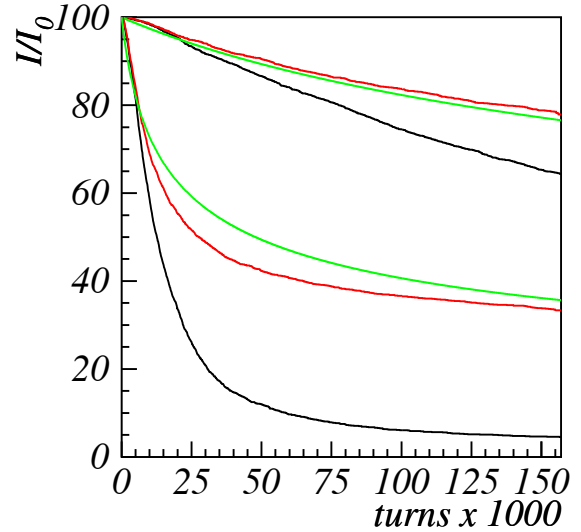


Figure 5: Comparison of the Markovian update (red curves) with a Markovian mapping (green curves). In black are shown the frozen space charge simulation results. The 3 lower curves are for 0.624×10^{11} ions/bunch; The 3 upper curves are for 0.250×10^{11} ions/bunch.

ASYMPTOTIC LIMIT

We now use the Markovian mapping to explore the asymptotic limit of beam survival. In order to obtain clearer results we do not use here the SIS100 lattice, but we confine ourself to a constant focusing lattice having one sextupole. The bunched beam is taken Gaussian in all 3 dimensions, and we apply the Markovian update algorithm. In Fig. 6 we show the beam survival after a storage of 5×10^6 turns. The black curve shows the result for a frozen space charge algorithm. The red curve is of the Markovian update. The resonance excited is centered in $Q_{xr} = 4.3333$ and the space charge incoherent tune-shift is $\Delta Q_x = 0.15$. The synchrotron tune requires 144 turns for one oscillation. The beam pipe is located at $4.5\sigma_r$, and the tracking has used 1000 macro-particles. We observe that the beam loss is divided into 2 regions:

1 Self-consistency dominated region

A region in which the self-consistency created by the Markovian update makes the beam survival to differ from that of the frozen space charge simulation. This region extends in $4.35 \lesssim Q_x \lesssim 4.39$. The

curve of beam survival becomes a straight line because the beam loss will stop when the space charge tune-spread becomes equal to the distance from the resonance which is set by Q_x in abscissa. This feature allows distinguishing inside this region other 2 sub-region: one is where the beam with a Markovian update survives better than those in the frozen tracking, and another with the opposite pattern.

2 Frozen dominated region

In Fig. 6 it is also remarkable that for $4.32 \lesssim Q_x \lesssim 4.35$ the beam survival of the Markovian update simulations is equal to those of the frozen space charge simulation. This feature contradicts the expected “close to the resonance collapse”, which seems not to occur. The overlapping of the red with the black curve in this region suggests that the avalanche process stops spontaneously. This feature depends directly from the longitudinal distribution, and this result is valid for a Gaussian longitudinal distribution.

The straight line on which the frozen simulations converge is obtained by the number of particles populating the shell in longitudinal phase space, which crosses the resonance. The amount of these particles was estimated in Ref. [11] as $\Delta N/N = (Q_x - Q_{x,r})/\Delta Q_x$. This relation shows that the functional dependency of the beam survival has to be linear (in Fig. 6 the green line).

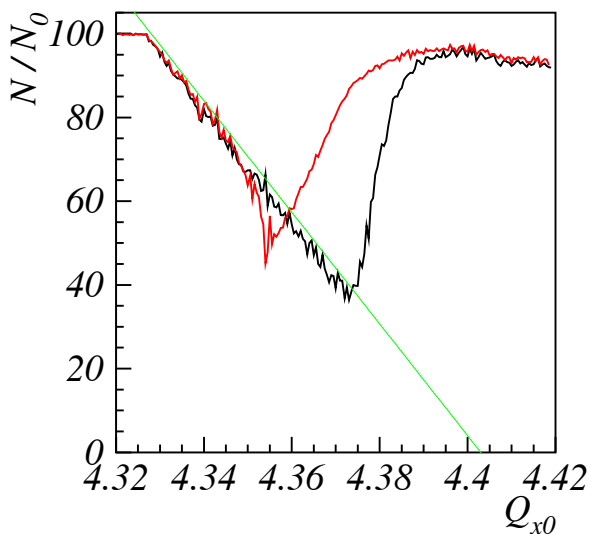


Figure 6: Beam survival in the limit of large number of turns (5×10^6 turns). In red the Markovian update curve, and in black the frozen simulation results.

SUMMARY/OUTLOOK

In this work we introduced the self-consistency in a Markovian approach (Markovian update). The application

of it to SIS100 suggests that self-consistency might mitigate long-term beam loss in SIS100 (here studied for the case without resonance compensation). Tests on the applicability of the method for several single processor numbers of macro-particles revealed a remarkable robustness, which is still subject of studies. We also discussed the Markovian mapping concept as a fast method to estimate a full Markovian update simulation. The results obtained for the SIS100 are encouraging and indicate that this technique might be an interesting tool for making predictions. By using a Markovian update algorithm we also investigated the very long-term beam survival for a simple case of a system subject to a single 3rd order resonance. Surprisingly we find that “the close to the resonance collapse” does not happen and that the beam survival converges to that of the frozen space charge simulations.

The explanation of this behavior requires further space not available in this proceeding, in addition further simulations have to be performed in order to understand the effect of the change of beam size due to beam loss.

The experimental campaign performed in CERN-PS in June 2012 in the framework of the LIU program at CERN [12] and its continuation will provide extremely valuable data for verifying the study here presented.

REFERENCES

- [1] A. Schoch, CERN 57-21 (Proton Synchrotron Division), Section 14 (1958).
- [2] A.W. Chao and M. Month, Nucl. Instrum. Methods **121**, 129 (1974).
- [3] A.I. Neishtadt and A.A. Vasiliev, Nucl. Instr. and Meth. A **561**, (2006), p 158-165.
- [4] G. Franchetti, *et al.*, Phys. Rev. ST Accel. Beams **6**, 124201 (2003); E. Metral, *et al.*, Nucl. Instr. and Meth. A **561**, (2006), 257-265.
- [5] J. Struckmeier, Phys. Rev. ST Accel. Beams, **3** 034202, 2000.
- [6] G. Franchetti *et al.*, IPAC 2011, MOPSO02. p. 589.
- [7] G. Franchetti, O. Chorniy, I. Hofmann, W. Bayer, F. Becker, P. Forck, T. Giacomini, M. Kirk, T. Mohite, C. Omet, A. Parfenova, P. Schuett Phys. Rev. ST Accel. Beams **13**, 114203 (2010).
- [8] P. Spiller *et al.*, Proc. of EPAC 2008, p. 298, MPOC100, (2008).
- [9] G. Franchetti and I. Hofmann, FAIR Technical Design Report SIS100 2008, pp. 39-49.
- [10] O. Kester, “High Intensity Issues at FAIR”, HB 2012, MO11B01, (2012).
- [11] G. Franchetti and I. Hofmann, Nucl. Instr. and Meth. A **561**, (2006), 195-202.
- [12] S. Gilardoni, *et al.*, “The High Intensity/High Brightness Upgrade Program at CERN: Status and Challenges”, HB 2012, TU01A01, (2012).