

Modeling Transverse Space Charge effects in IOTA with pyORBIT

Runze Li², Tanaji Sen ^{*1}, and Jean-Francois Ostiguy¹

¹Fermi National Accelerator Laboratory, Batavia, IL 60510

² University of Wisconsin, Madison, WI 53706

Abstract

The role and mitigation of space charge effects are important aspects of the beam physics research to be performed at the Integrable Optics Test Accelerator (IOTA) at Fermilab. The impact of nonlinear integrability (partial and complete) on space charge driven incoherent and coherent resonances will be a new feature of this accelerator and has the potential to influence the design of future low energy proton synchrotrons. In this report we model transverse space charge effects using the PIC code pyORBIT. First we benchmark the single particle tracking results against MADX with checks on symplecticity, tune footprints, and dynamic aperture in a partially integrable lattice realized with a special octupole string insert. Our space charge calculations begin with an examination of the 4D symplecticity. Short term tracking is done first with the initial bare lattice and then with a simple matching of the initial rms values with space charge. Next, we explore slow initialization of charge as a technique to establish steady state and reduce emittance growth and beam loss following injection into a mismatched lattice. We establish values of space charge simulation parameters so as to ensure numerical convergence. Finally, we compare the simulated space charge induced tune shifts and footprints against theory.

Contents

1 Introduction

2

*tсен@fnal.gov

This manuscript has been authored by Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the U.S. Department of Energy, Office of Science, Office of High Energy Physics.

2	Single-Particle Tracking	3
2.1	Symplecticity Tests	4
2.2	Tune Footprint	7
2.3	Dynamic Aperture Tests	8
2.4	Hamiltonian Test	10
3	pyORBIT Space Charge Model	11
3.1	Symplectic tests with space charge	12
4	Emittance Growth and Particle Loss	12
4.1	Initial rms matching	14
4.2	Slow Initialization	16
4.3	Convergence Tests on simulation parameters	18
5	Small amplitude tune shifts and tune footprints	22
5.1	Small Amplitude Tune Shift	24
5.2	Tune Footprints	25
6	Conclusions	27
A	Appendix A: Additions and changes to pyORBIT	29
A.1	Adding Dipole Edge Element in pyORBIT	29
A.2	Nonlinear Effects of Magnet Edge	29
A.3	Other Changes	30

1 Introduction

The Fermilab Integrable Optics Test Accelerator (IOTA) is a storage ring for beam physics research. An important aspect of the research program is to explore the potential of integrable optics to mitigate deleterious effects of space charge in high intensity proton synchrotrons. In theory, integrable single particle dynamics eliminate resonances, providing stable motion over a wide tune range. In a beam, large betatron tune spread is known to be effective at suppressing instabilities. To the extent that desirable properties of integrable single particle motion persist in the presence of strong space charge, integrability may help to preserve stability in intense beams, but that remains to be verified.

The IOTA ring is currently operating with electrons in the energy range 100 - 150 MeV. At a later stage, experiments will be performed with protons at a kinetic energy of 2.5

IOTA proton parameters	
Circumference	39.97 [m]
Kinetic Energy	2.5 [MeV]
Maximum bunch intensity /current	$9 \times 10^{10} / 8$ [mA]
Transverse normalized rms emittance	(0.3, 0.3) [mm-mrad]
Betatron tunes	(5.3, 5.3)
Average transverse beam sizes (rms)	(2.22, 2.22) [mm]
Kinematic γ / Transition γ_T	1.003 / 3.75
Rf voltage	400 [V]
Rf frequency / harmonic number	2.2 [MHz] / 4
Bucket wavelength	~ 10 [m]
Bucket half height in $\Delta p/p$	3.72×10^{-3}
rms bunch length	1.7 [m]
rms energy /momentum spread	$1.05 \times 10^{-5} / 1.99 \times 10^{-3}$
Beam pipe radius	25 [mm]
Bunch density	$\sim 4 \times 10^{14}$ [m $^{-3}$]
Debye length λ_D	~ 60 [μ m]

Table 1: Machine and beam parameters of the IOTA proton ring

MeV. By turning on/off special nonlinear inserts, the ring can be operated with either conventional or integrable optics. Space charge in the proton beam will be a determinant factor for beam stability. As in any other low energy proton synchrotron, both incoherent and coherent space charge effects will play a role. When the ring is configured with integrable optics, the impact of the perturbation due to space charge on properties associated with integrability needs to be understood. In this context, it is important to assess the capability and suitability of existing simulation codes. We report here on a variety of relevant validation tests that were performed using pyORBIT [1], a PIC code developed and maintained at ORNL and motivated by the need to simulate certain aspects of SNS.

Table 1 shows the relevant parameters of the IOTA proton ring.

2 Single-Particle Tracking

In this section, we aim to validate single-particle tracking in pyORBIT against MADX, a well-established and extensively tested code. To minimize discrepancies possibly introduced by subtle differences in the way distributed nonlinearities are modeled by the two codes, all sources of nonlinearity other than those confined to a special insert region are turned off. The insert region is initially populated with octupole magnets whose strengths vary in inverse proportion to $\beta^3(s)$; theoretically, the dynamics of this system is characterized by a single invariant. Later, the octupoles are replaced by special nonlinear magnets, making the optics (again, theoretically) fully integrable (two invariants). Good agreement between the codes provides confirmation that the lattice element sequence is both read in

and interpreted in a consistent manner and that basic single particle motion is modeled correctly by pyORBIT.

2.1 Symplecticity Tests

In Hamiltonian mechanics, a transformation of the coordinates is said to be canonical if it preserves the form of Hamilton's equations. A related result is that dynamical evolution is itself a succession of infinitesimal canonical transformations and therefore, the dynamical evolution from initial to final phase space coordinates is also a canonical transformation. It can be shown that a canonical transformation must satisfy a local analytic condition, the so-called symplectic condition. Conversely, a transformation that satisfies the symplectic condition is canonical. The symplectic condition can be stated as follows:

$$\mathbf{J}^T \mathbf{S} \mathbf{J} = \mathbf{S} \quad (2.1)$$

where J is the Jacobian matrix of the transformation. Assuming the transformation is defined by the dynamical evolution map \mathcal{M} connecting the initial and final phase space coordinates $\mathbf{X}(s_0)$ and $\mathbf{X}(s_f)$ one has

$$\mathbf{X}(s_f) = \mathcal{M} \mathbf{X}(s_0) \quad (2.2)$$

$$J_{k\ell} = \frac{\partial X_k(s_f)}{\partial X_\ell(s_0)} \quad (2.3)$$

where $\mathbf{X}(s_i)$ represents the initial phase space vector of canonical coordinates while $X_k(s_i)$ denotes individual components of this vector. The symplectic matrix \mathbf{S} is defined as

$$\mathbf{S} = \begin{pmatrix} \mathbf{s} & 0 & \cdots & 0 \\ 0 & \mathbf{s} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{s} \end{pmatrix} \quad \mathbf{s} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad (2.4)$$

Geometrically, the determinant of the Jacobian matrix represents the ratio between final and initial differential (oriented) phase space volumes enclosing corresponding points in phase space. Taking the determinant on both sides of 2.1, and using $\det(\mathbf{S}) = 1$, one finds $\det \mathbf{J} = \pm 1$. However, in the limit of a vanishingly small step along s the dynamical transformation must converge to the identity I i.e. $\lim_{s \rightarrow 0} J = I$, and one concludes that $\det \mathbf{J} = 1$ i.e. the phase space volume is a dynamical invariant. This is the well-known Liouville's theorem.

A matrix is called symplectic if it satisfies Eq. (2.1), known as the symplectic condition. While a unimodular Jacobian determinant is a necessary condition for symplecticity, it is not sufficient test in phase space of dimensions higher than two. We apply both the determinant test and the full symplectic condition test Eq. (2.1) as measures of the deviations from symplecticity.

A residual deviation from exact symplecticity of the one-turn map computed by the code due to numerical round off errors or approximations will translate into a long term

violation of Liouville’s theorem. Therefore, for accurate long term simulation of the dynamics, the Jacobian of the one-turn maps generated by a tracking code should be as close as possible to 1. The finite size representation of floating point numbers sets a minimum for the achievable deviation. Assuming 64-bit floating point representation (53-bit significance) the minimal deviation due to round-off is

$$|\det(\mathbf{J}) - 1| \simeq 10^{-14}$$

By tracking two or more test particles originating from closely neighboring points in phase space, one can compute partial derivatives using finite differences. Since the latter are approximations, deviations from the exact values of the partial derivatives will translate into larger apparent deviation from symplecticity. A rough error bound for the accuracy of the derivative may be obtained from Taylor’s theorem. For simple one-sided (either forward or backward) differences

$$\frac{\Delta X}{h} = \frac{f(X+h) - f(h)}{h} = f' + \mathcal{O}(h) \quad (2.5)$$

while for centered differences,

$$\frac{\Delta f}{h} = \frac{f(X+h) - f(X-h)}{2h} = f' + \mathcal{O}(h^2) \quad (2.6)$$

Assuming that the Jacobian matrix is diagonally dominant with diagonal entries near unity and off diagonal entries of magnitude $\ll 1$ it can be shown that

$$1 - h < \det(J+h) < (1 + 2h + nh^2)^{n/2} \quad (2.7)$$

where n is the matrix dimension and it is assumed that the magnitude of the error affecting all the Jacobian entries is about the same. On the basis of this bound, using centered differences and $h \sim 10^{-3}$, deviations from unity of the determinant on the order of 10^{-6} are to be expected; larger deviations would be indicative either of code issues (i.g. incorrectly computed map) or of the presence of chaotic motion.

The previous discussion has implicitly assumed well-behaved, smooth differentiable dynamical maps. It should be obvious that to the extent that maps associated with chaotic motion are not (practically) differentiable, the Jacobian determinant test is expected to fail in chaotic regions. In a typical accelerator the dynamics is smooth in the vicinity of the reference orbit. Chaotic motion is observed as one moves away from reference trajectory, where nonlinearity become increasingly significant. An abrupt deviation the Jacobian determinant away from unity may be used to delineate the dynamic aperture boundary.

We now discuss the results of the symplectic tests of single particle tracking with py-orbit. Since this report is limited to 4D (transverse) space charge studies, we consider 4D symplectic tests both here and in a later Section where space charge is turned on. To perform numerical tests of symplecticity, it is best to work in normalized Floquet canonical coordinates. The latter are defined as follows

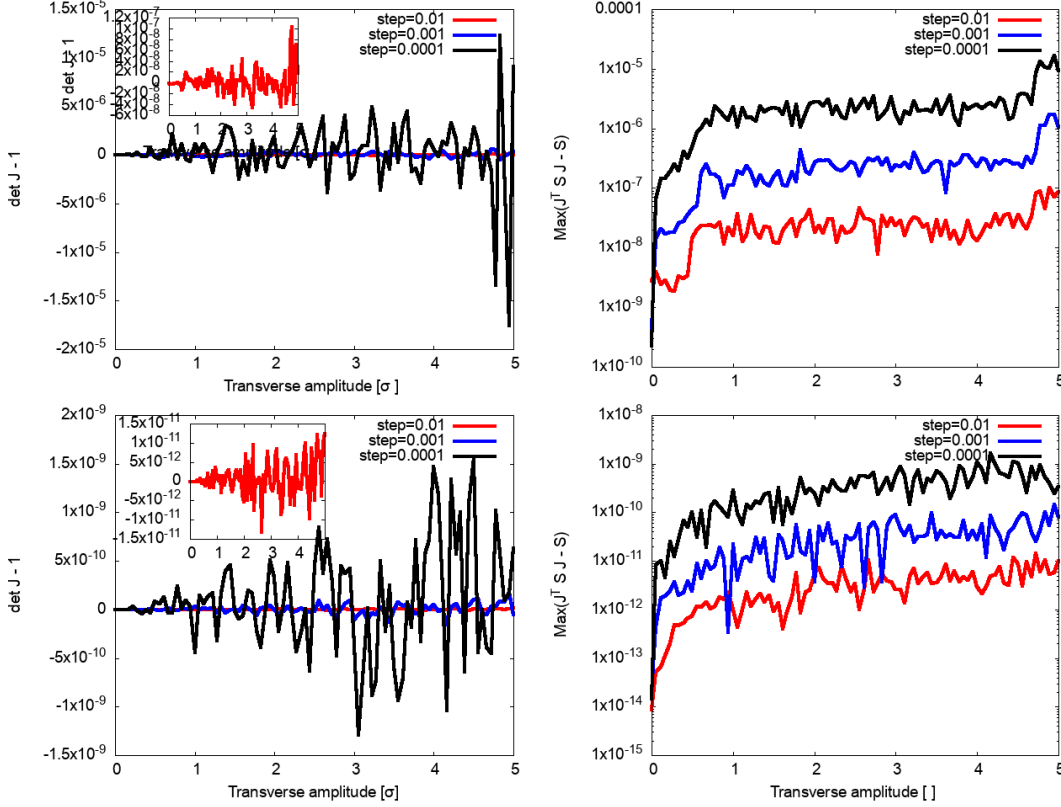


Figure 1: Linear lattice. Difference $\text{Det}[\mathbf{J}] - 1$ (left) and maximum element of the matrix $\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}$ (right) as functions of the transverse amplitude in units of the rms beam size. Top: MADX, Bottom: pyORBIT. The left plot in both rows has insets showing the smallest values of the difference which result for the step size 0.01.

$$\mathbf{X} = \left[\frac{x}{\sqrt{\beta_x}}, \frac{(\beta_x x' + \alpha_x x)}{\sqrt{\beta_x}}, \frac{y}{\sqrt{\beta_y}}, \frac{(\beta_y y' + \alpha_y y)}{\sqrt{\beta_y}} \right] \quad (2.8)$$

In these coordinates, all components of \mathbf{X} have dimension $[\sqrt{L}]$ and the elements of the Jacobian matrix \mathbf{J} are dimensionless.

We first present results of the symplecticity tests for the linear lattice. Figure 1 shows the deviation from unity of $\text{det}[\mathbf{J}]$ and the maximum norm $|\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}|$ as functions of the transverse amplitude for this linear lattice.

In both figures, the deviations are shown for three small increments (0.01, 0.001, 0.0001) of the initial amplitude. We find that the deviation from unity $\text{det}[\mathbf{J}] - 1$ is nearly 10^{-7} with MADX. For pyORBIT the deviation is at most 10^{-11} for the at the largest step size of 0.01 and increases for smaller step sizes. Similarly, the maximum norm $|\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}|$ is $O(10^{-8})$ with MADX and $O(10^{-12})$ with pyORBIT for at step size 0.01 and increases by an order of magnitude with each successive decrease in step size. The larger deviation from unity of the Jacobian determinant observed with MADX appears to be related to its handling of dipole edge focusing, which is turned off in pyorbit.

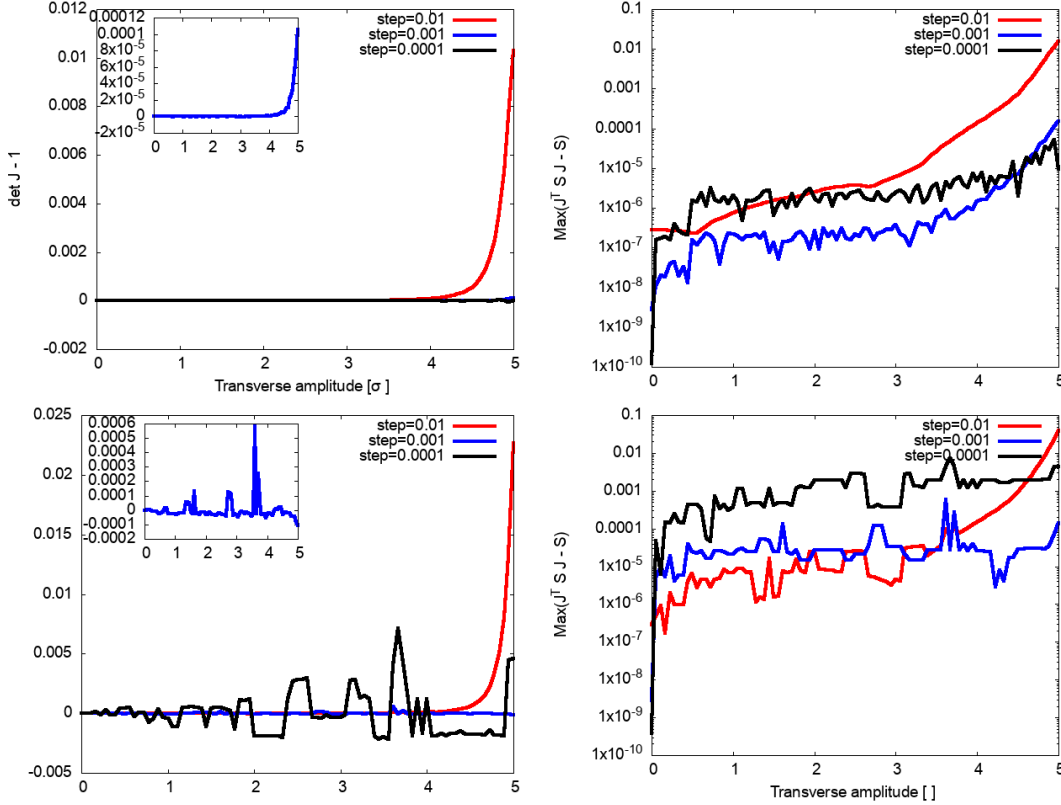


Figure 2: Lattice with octupoles. Difference $\text{Det}[\mathbf{J}] - 1$ (left) and maximum element of the matrix $\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}$ (right) as functions of the transverse amplitude in units of the rms beam size. Top: MADX, Bottom: pyORBIT

Results for the lattice with the octupole string in IOTA are shown in Figure 2. Up to amplitudes of 4σ , $\text{Det}[\mathbf{J}] - 1$ at step size 0.01 is $\sim 10^{-8}$ with MADX and $\sim 10^{-4}$ with pyORBIT. The deviation increases steeply and is the largest at higher betatron amplitudes with both codes. For the smaller step of 0.001, the largest deviation over 0-5 σ is $\sim \times 10^{-4}$ and can be considered the optimum step size; the deviations are larger again at the smaller step size= 0.0001. The same pattern is observed for the maximum norm $|\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}|$.

2.2 Tune Footprint

A meaningful way to validate pyORBIT against MADX is to compare tune footprints. To do so, particles are initialized uniformly from $(0, 0, 0, 0)$ to $(5\sigma_x, 0, 5\sigma_y, 0)$ in the x and y planes. They are then tracked for 5000 turns and the transverse positions (x, y) of every particle are recorded for each turn at a fixed location. For each particle, the fractional part of the tunes Q_x and Q_y are obtained by performing a Fast Fourier Transform (FFT) of $x(n)$ and $y(n)$. Q_x and Q_y are presented as a scatter plot in tune space. The left plot in Figure 3 shows the tune footprint with octupoles added to the IOTA lattice. Both MADX and pyORBIT show a similar tune spread, which serves as a cross-check of the pyORBIT single particle tracking model. To make a more meaningful quantitative comparison between the codes,

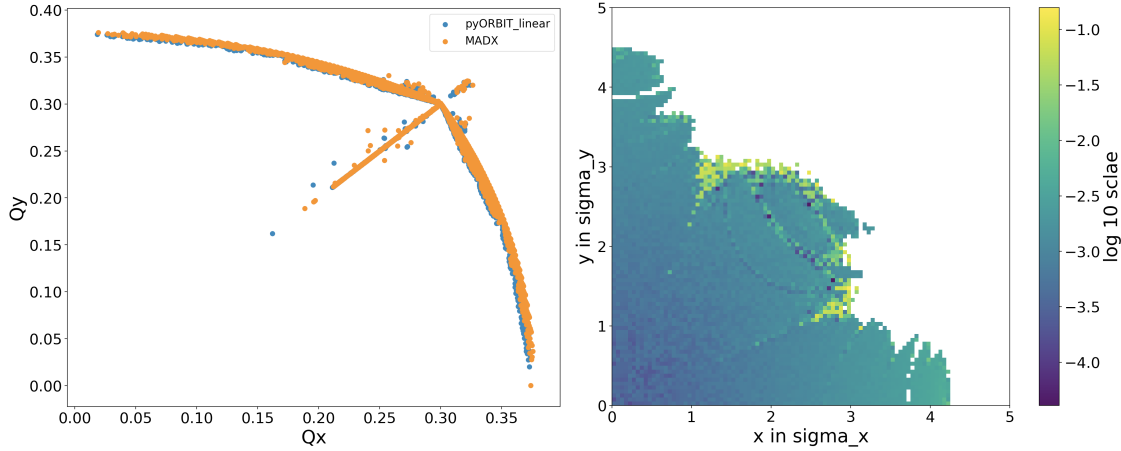


Figure 3: Left: Tune footprint of the IOTA lattice with octupoles, obtained with pyORBIT and MADX. Right: Difference in tune shifts (on a log scale) between pyORBIT and MADX as functions of the initial positions of the test particles.

we present the difference in the tune shift computed by the two codes, and it is computed as

$$dQ = |Q_{x,P} - Q_{x,M}| + |Q_{y,P} - Q_{y,M}| \quad (2.9)$$

where e.g. $Q_{x,M}, Q_{x,P}$ are the horizontal tunes calculated with MADX and pyORBIT respectively. The right plot in Figure 3 shows the tune difference dQ for all test particles as functions of their initial positions. Figure 3b shows the tune difference dQ for all test particles as function of their initial positions. For most test particles the difference between the tunes predicted by MADX and pyORBIT falls within the range $10^{-4} : 10^{-3}$, which is comparable to the resolution of the FFT sampling. The result provides another confirmation of the correctness of single particle tracking in pyORBIT. Particles exhibiting the largest discrepancies are located close to the dynamic aperture boundary, as shown in Fig. 3. Beyond this boundary, the particles are lost and $\log dQ$ is undefined. All particles in the tune the footprint are initialized at coordinates $(x, 0, y, 0, 0, 0)$. The boundary defined by large deviations seen in 3 corresponds to the 4D dynamic aperture in Figure 4a. Small differences between the tune calculated by MADX and pyORBIT are likely due to increasingly unstable motion in the vicinity of the dynamic aperture boundary. We remark in passing that the fractal nature of this boundary is made evident by the tune difference diagram and that a similar procedure is used in a frequency map analysis except that in that case the same code is used to calculate tune differences between neighboring particles.

2.3 Dynamic Aperture Tests

The 4D, 5D and 6D dynamic apertures are calculated for the IOTA lattice with and without octupoles using both MADX and pyORBIT. Circular physical apertures with radius 25 mm are assigned to all elements. 5000 particles are initialized using one of the following procedures: The 4D dynamic aperture is calculated with the rf cavity turned off and particles are initialized at $(x_i, 0, y_i, 0, 0, 0)$ so that the motion is restricted to transverse

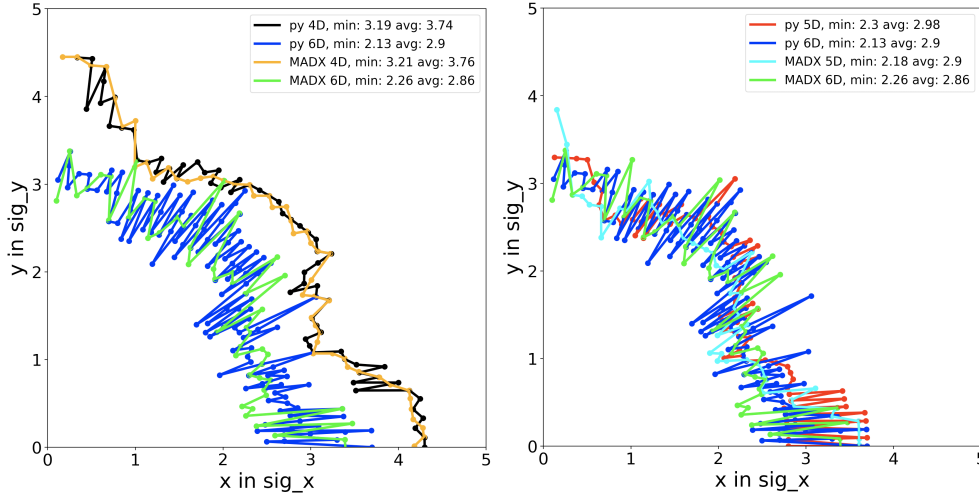


Figure 4: 4D, 5D, and 6D dynamic apertures (expressed in units of the rms beam sizes) for IOTA with octupoles from pyORBIT and MADX. Left: Comparison of 4D and 6D apertures, Right: Comparison of 5D and 6D apertures, they are nearly the same.

Aperture type	Minimum DA		Average DA	
	PYORBIT	MADX	PYORBIT	MADX
4D	3.19	3.21	3.74	3.76
5D	2.3	2.18	2.98	2.9
6D	2.13	2.26	2.9	2.86

Table 2: Minimum and average dynamic apertures with octupoles calculated using pyORBIT and MADX.

4D phase space. The 5D dynamic aperture is calculated with the rf cavity turned off, and particles are initialized similarly but with a constant momentum offset of $1 \sigma_p$ so that chromaticity can affect the dynamic aperture. The rf cavity is turned on for the 6D calculations with the same initial coordinates so that the effects of synchrotron motion are included.

A perfect lattice is used; no errors, e.g. field or alignment errors, are included. Particles are tracked for 10000 turns, and the initial coordinates in the x - y plane of all surviving particles are recorded. The largest excursions of the surviving particles yield an upper bound of the dynamic aperture. In the purely linear lattice, the horizontal dynamic aperture is approximately $5\sigma_x$ and $6\sigma_y$ vertically. As expected, the dynamic aperture is about the same as the physical aperture at the peak beta locations $\hat{\beta}_{x,y}$. Results from pyORBIT and MADX are in good agreement.

Results of dynamic aperture calculations with octupoles added to the linear lattice are shown in Figure 4 and summarized in Table 2 of minimum and average aperture sizes expressed in terms of rms beam sizes.

Referring to Table 2 and to Figure 4, there is good agreement between MADX and pyORBIT, providing additional confirmation of the soundness of the single particle track-

ing model in pyORBIT. The dynamic apertures for 6D and 5D tracking are much smaller than for 4D. For an emittance of $0.3\mu m$ the latter is about 3σ . This indicates that with a transverse Gaussian distribution, truncation at 3σ should prevent particle loss. However, introduction of a realistic lattice with errors and space charge will likely significantly increase particle losses.

2.4 Hamiltonian Test

In the original paper on the integrable lattices [2] that provided the motivation for IOTA, the derivation of the main result assumed axisymmetric focusing. This can be provided by solenoids in low energy machines. In IOTA, focusing in the linear portions of the ring is provided by quadrupoles and edge focusing in dipoles. With equal focusing in both planes, the complete Hamiltonian in terms of the independent variable s is

$$H = \frac{1}{2}(p_x^2 + p_y^2) + \frac{1}{2}K(s)[x^2 + y^2] + V(s) \quad (2.10)$$

Axial symmetry results in $\beta_x(s) = \beta_y(s) = \beta(s) \forall s$. This is not quite the case in IOTA, but we'll assume that it is approximately satisfied.

The octupole potential can be written as

$$V(s) = \frac{K_3(s)}{4!}[x^4 + y^4 - 6x^2y^2], \quad K_3 = \frac{1}{(B\rho)} \frac{\partial^3 B_y}{\partial x^3} \quad (2.11)$$

where the strength $K_3(s)$ (which has the definition as in MADX) varies with s .

If the strength of each octupole in the string is chosen to vary as the inverse cube of the beta function in a section of the ring where $\beta_x(s) = \beta_y(s) = \beta(s)$ [2], then

$$K_3(s) = \kappa/\beta(s)^3 \quad (2.12)$$

where κ is a constant independent of s . Expressed in terms of Floquet coordinates $x_N == x/\sqrt{\beta_x}, y_N == y/\sqrt{\beta_y}$, and transforming to the phase advance $\psi = \int ds/\beta(s)$ as the independent variable, the rescaled potential is

$$U = \beta(s)V(s) = \frac{1}{24}\kappa[x_N^4 + y_N^4 - 6x_N^2y_N^2] \quad (2.13)$$

which is independent of the time variable s .

The required longitudinal variation of the potential can be well-approximated by a string of octupole magnets. Summing the contributions of individual magnets we obtain $\kappa = \sum_j N_{oct} \kappa_j$ where N_{oct} is the number of octupoles ($N_{oct} = 17$ in the IOTA string).

The Hamiltonian invariant is

$$H = \frac{1}{2}(p_{x_N}^2 + p_{y_N}^2 + x_N^2 + y_N^2) + \frac{1}{24}\kappa[x_N^4 + y_N^4 - 6x_N^2y_N^2] \quad (2.14)$$

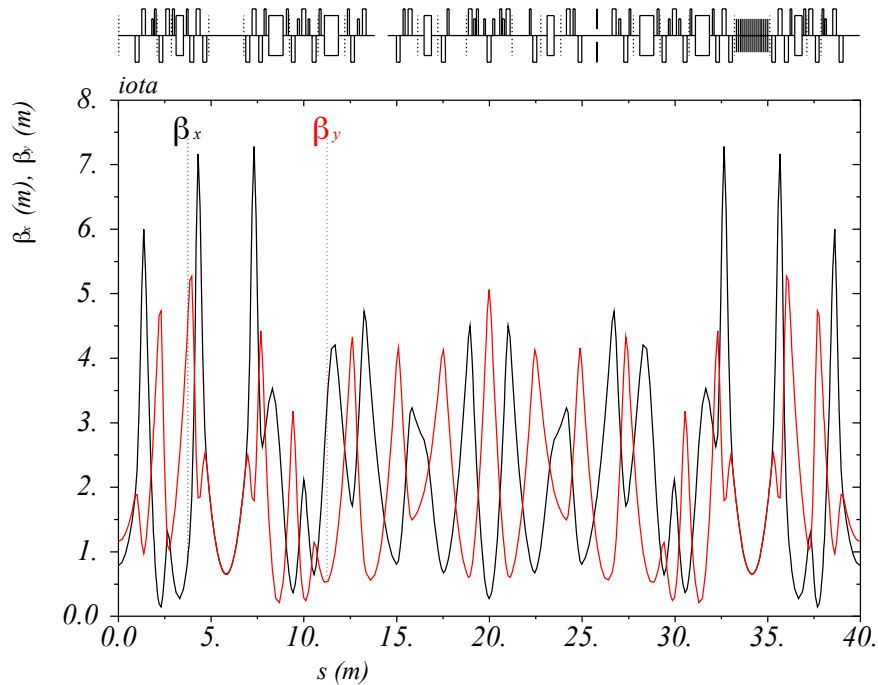


Figure 5: Beta functions in the IOTA ring; the octupoles are in the region from 33.3 to 35m.

Figure 5 shows the beta functions in the IOTA ring; the octupoles occupy the 1.7 m section from 33 m to 35 m. Within the octupoles, the beta functions are axially symmetric but not so in the rest of the ring. This lack of complete symmetry and the replacement of a continuous octupole potential by a sum of discrete potentials will have an impact on the invariance of the Hamiltonian.

Figure 6 shows that the relative variation in the Hamiltonian with both codes is about $\pm 2\%$ for a small amplitude particle with both MADX and pyORBIT. The variation increases for larger amplitude particles but the variation in both codes is always nearly the same.

3 pyORBIT Space Charge Model

PYORBIT is a particle-in-cell code in which macro-particles represent charged particles in a bunch. The macroparticles are deposited on a grid and a smoothed density is extracted by interpolation between the grid points. The electric potential is found by numerically solving Poisson's equation in the beam rest frame which thereby allows evaluation of the space charge forces on the macro-particles. The details of the physical (as opposed to numerical) space charge force depend on the bunch intensity, the particle distribution and boundary conditions. In this report we consider only the transverse space charge

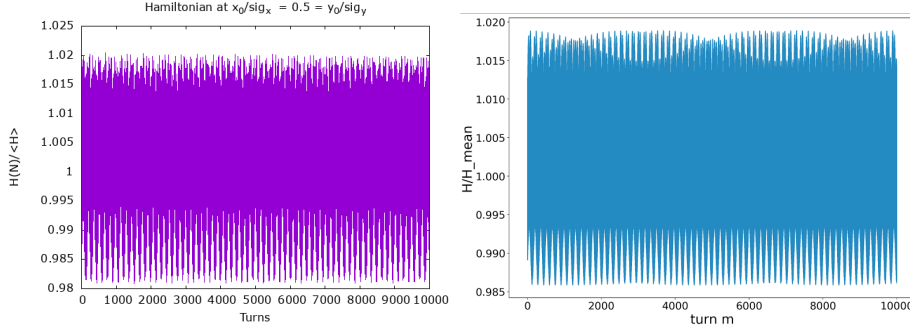


Figure 6: Variation in the single particle Hamiltonian relative to its initial value in MADX (left) and pyORBIT (right). Initial coordinates are $x/\sigma_x = 0.5 = y/\sigma_y$, $x' = 0 = y'$.

forces and we assume open boundary conditions. The transverse distributions are assumed to be Gaussian since that is experimentally well-founded. The transverse forces vary along the bunch length due to the change in local density, so the longitudinal distribution also matters.

In our simulations, a waterbag distribution is used in the longitudinal plane. To prevent particles leaking out of the bucket when longitudinal space charge forces (to be reported in a separate study) are considered, particles are uniformly distributed within an inner Hamiltonian contour inside the separatrix defined by the RF cavity. An example using 10^6 macro-particles is plotted in Figure 8.

3.1 Symplectic tests with space charge

Figure 9 shows the results of symplecticity tests with space charge. The largest deviation $\det \mathbf{J} - 1$ is 0.4 using a step size of 0.01, it falls to 0.15 at step size 0.001 and is an order of magnitude smaller (0.02) at step size 0.0001. The maximum norm $|\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}|$ exhibits a similar behavior and reaches its smallest value – 0.01 – at step size 0.01. Compared to the cases with octupoles but without space charge in Figure 2, the deviations from symplecticity are at least two orders of magnitude larger. Another difference is that the largest deviations occur at a lower betatron amplitude (about 2σ).

The fact that the deviations from symplecticity in pyORBIT in the presence of space charge are significant is not surprising as it is observed in most PIC codes to date. This state of affair makes these codes not suitable for long term tracking.

4 Emittance Growth and Particle Loss

Predictions of emittance growth and particle loss are two of the many reasons for using a space charge code. We first discuss the results of emittance growth without space charge to estimate the background growth and then at low and full bunch intensity to observe the impact of increasing space charge effects. The top row in Fig. 10 shows the emittance in

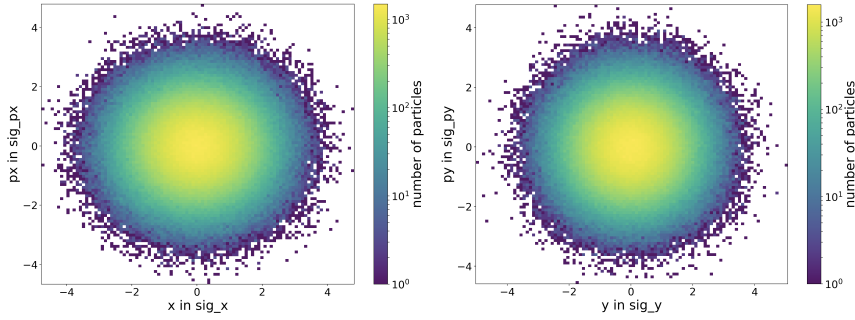


Figure 7: Particle distribution in x-pz (left) and y-py plane (right)

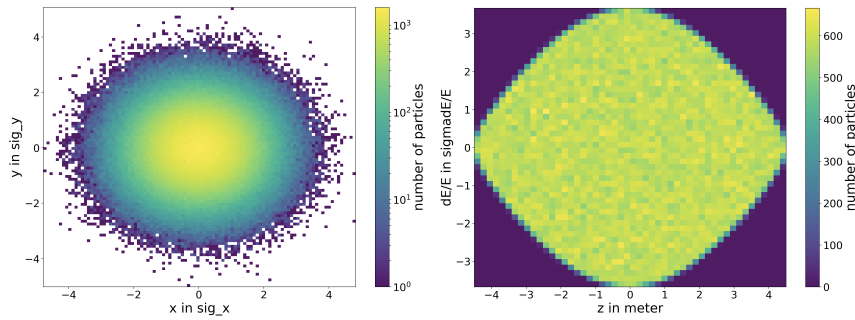


Figure 8: Particle distribution in x-y (left) and z-dE plane (right)

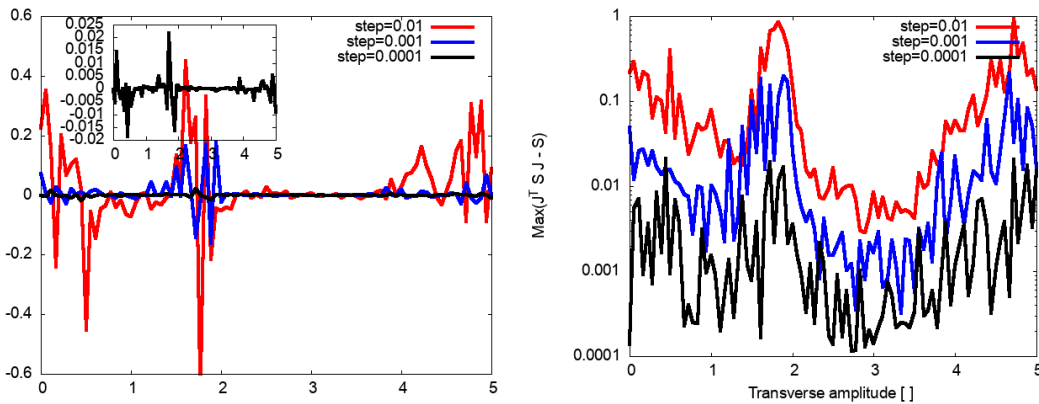


Figure 9: Symplecticity tests with space charge. Difference $\text{Det}[\mathbf{J}] - 1$ (left) and maximum element of the matrix $\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}$ (right) as functions of the transverse amplitude in units of the rms beam size.

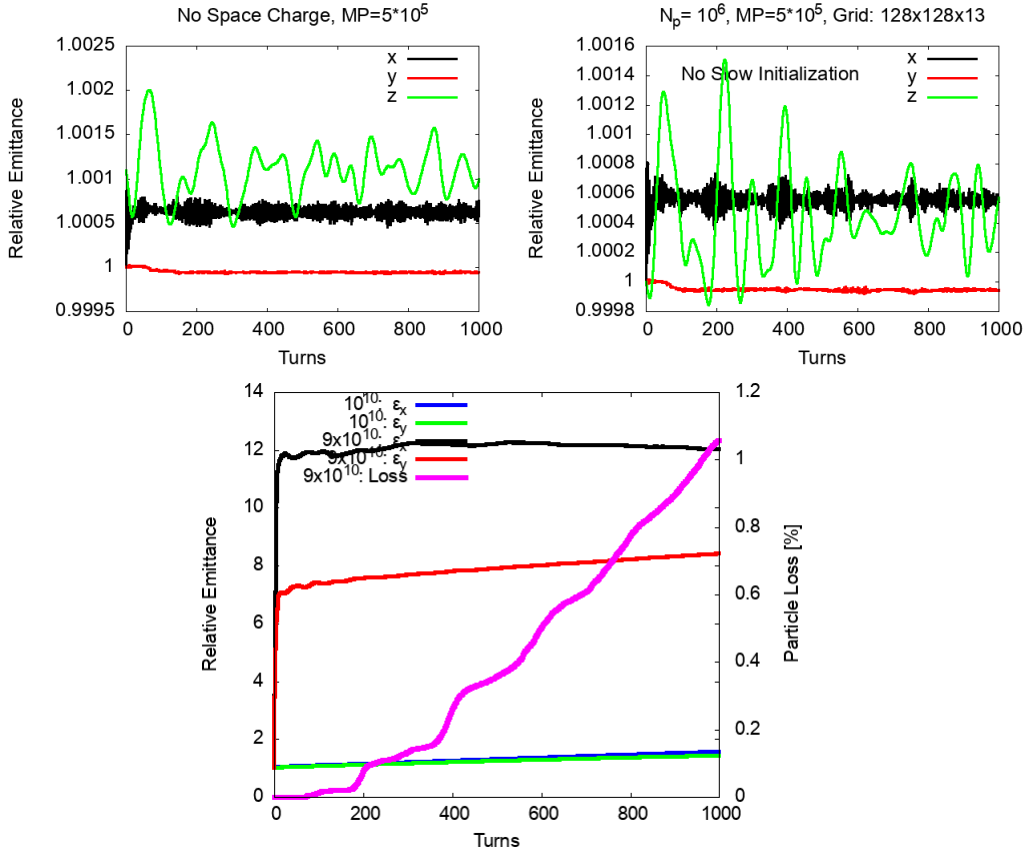


Figure 10: Top: Emittance growth in each plane without space charge (left) and at a low bunch intensity of 10^6 (right). Bottom: Emittance growth (left vertical scale) at intensities 10^{10} and 9×10^{10} and the particle loss (right vertical scale) at the higher intensity. There is no loss at 10^{10} over this time scale, hence is not shown.

all three planes without space charge and at a low bunch charge of 10^6 protons, five orders of magnitude lower than the full intensity. There is no particle loss in either case. The influence of dispersion and momentum spread is visible in the transverse emittance. The oscillations in both the horizontal and longitudinal emittance have the same period. The background noise in the transverse emittances is less than 0.1 %. The bottom plot in this figure shows the emittance growth at bunch intensities 10^{10} and 9×10^{10} . At the lower intensity, the emittance grows by 20% and there is no loss. At the higher intensity, the emittance growth is dramatic, up to 12 times (ϵ_x) and 7 times (ϵ_y). The growth occurs over the first few turns and quickly reaches a plateau. The predicted total loss over 1000 turns is $\sim 1\%$, which is unacceptably high for such a short time scale.

4.1 Initial rms matching

As seen in Fig. 10, the mismatch between the bare lattice functions and those with space charge leads to a rapid initial emittance growth and particle loss at full intensity. Properly

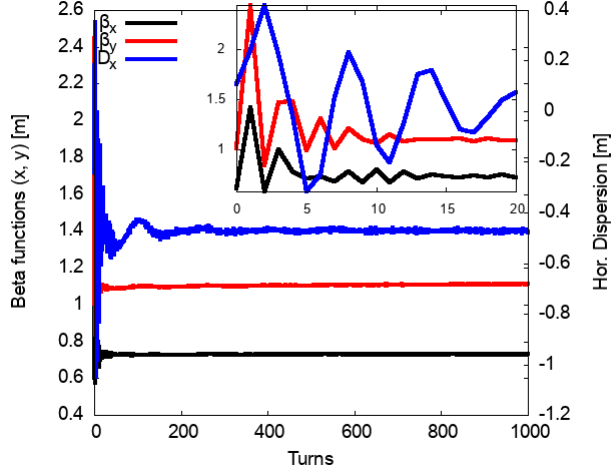


Figure 11: The statistical beta functions (on the left vertical scale) and dispersion (right vertical scale) over time. The inset shows that the beta functions relax to their equilibrium values within 20 turns, the dispersion relaxes over about 200 turns.

matching a lattice to space charge induced changes requires matching the Twiss functions including phase advances so that the beam envelopes have the space charge equilibrium values all around the ring. This matching can be done with the linear part of the space charge forces. Here we simply match the initial Twiss functions at the injection point and test the benefits, if any. We track the unmatched macroparticle distribution for 1000 turns. At the end of this period, we extract the statistical Twiss functions e.g in the horizontal plane as (here $\delta = \Delta p/p$)

$$\langle \beta_x \rangle = \frac{\langle x^2 \rangle}{\sqrt{\langle x^2 \rangle \langle (x')^2 \rangle - \langle (xx')^2 \rangle}}, \quad \langle \alpha_x \rangle = -\frac{\langle xx' \rangle}{\sqrt{\langle x^2 \rangle \langle (x')^2 \rangle - \langle (xx')^2 \rangle}} \quad (4.1)$$

$$\langle D_x \rangle = \frac{\langle x \delta \rangle}{\langle \delta^2 \rangle}, \quad \langle D'_x \rangle = \frac{\langle x' \delta \rangle}{\langle \delta^2 \rangle} \quad (4.2)$$

Figure 11 shows the evolution of the beta functions. There are sharp transients as the beam adjusts to the space charge forces but the beta functions reach stable values close to their initial values within 20 turns, the same time scale as for the emittance as seen in Fig. 10. The initial beta functions are $(\beta_x, \beta_y) = (0.79, 1.16)\text{m}$ and the final values after 1000 turns are $(0.73, 1.11)\text{m}$. The dispersion function relaxes over a longer time scale and changes more significantly from -0.24m to -0.43m . Fig. 12 shows the relative emittance and particle loss without matching and matching just the (β, α) functions and also including the matching the horizontal dispersion D_x . The emittance growth is relatively insensitive to this simple matching, the loss reduces from $\sim 1\%$ to about 0.8% over 1000 turns.

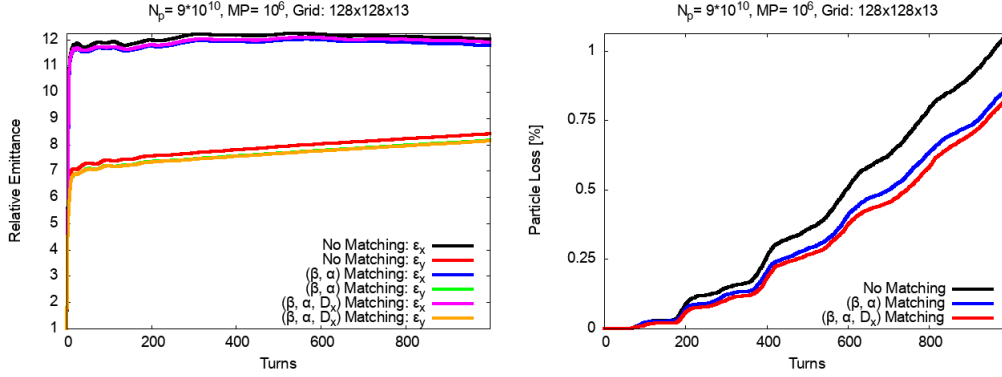


Figure 12: Emittance growth with and without matching (left), particle loss for the same cases (right).

4.2 Slow Initialization

In this section we describe an alternative procedure to numerically reach a steady state. Rather than injecting with the full charge, the charge per macroparticle is increased linearly from zero to its full value at turn T_{init} , the initialization time. Provided this process is sufficiently adiabatic, one expects the beam to remain in near equilibrium at every step as the beam has time to adjust to a slowly changing space charge force.

We note that in this procedure, the number of macro-particles stays constant while the charge per macro-particle increases with time. This is easy to implement in a simulation since all the macro-particles are drawn from an initial distribution; however, the procedure does not mimic the way bunch intensity increases in a real accelerator while the machine is being filled. There are a number of methods available that would allow to more realistically populate each bunch; the details of the subsequent dynamics will be influenced by the chosen method. Such an investigation is left for a future study.

The IOTA beam pipe radius is 25 mm; this value is used as the limiting aperture for simulation purposes. In the actual machine, the available aperture in the nonlinear elements such as the octupoles or the nonlinear lens have smaller apertures. These nonlinear elements are not included here. To distinguish halo growth from core emittance growth, we will also discuss the dynamics with a larger aperture e.g. 100 mm. Figures 13 shows emittance growth and beam loss for two choices of T_{init} with the physical aperture set to 25 mm. Losses start when $3 \sigma_x$ reaches the limiting aperture radius.

Figure 14 shows emittance growth and particle loss after 500 turns as a function of the slow initialization time. We observe that the emittance growth decreases monotonically as T_{init} increases and falls by about 15% in the range studied. The losses increase by about 40 fold from the minimum to maximum over this range and are significantly more sensitive to the initialization time. We also observe that there is no reduction in the rms emittances over this range of T_{init} , so clearly the core does not reach the aperture. These observations suggest that the halo is strongly affected by the charge on the macroparticles while the growth of the core is less affected. It may be useful to determine why the loss is minimum at 40 turns. We note in passing here that the relative emittance change in the transverse planes

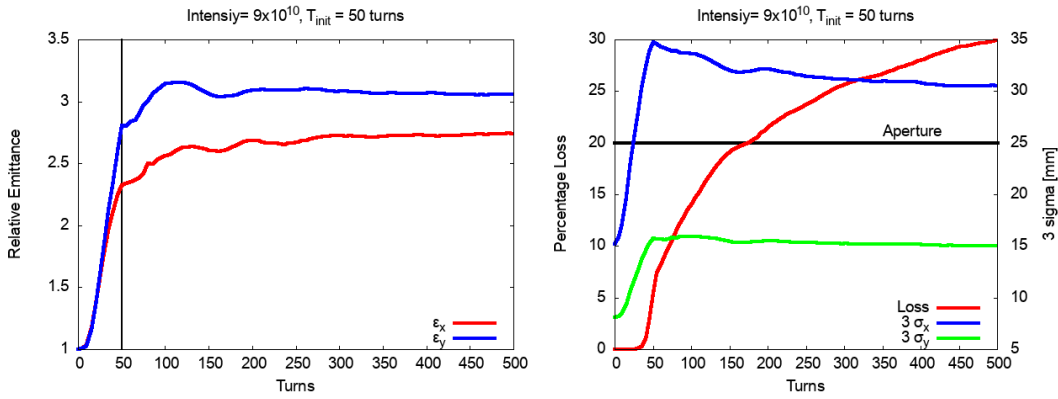


Figure 13: Emittance growth and losses at a slow initialization time of 50 turns and with an aperture of 25 mm.

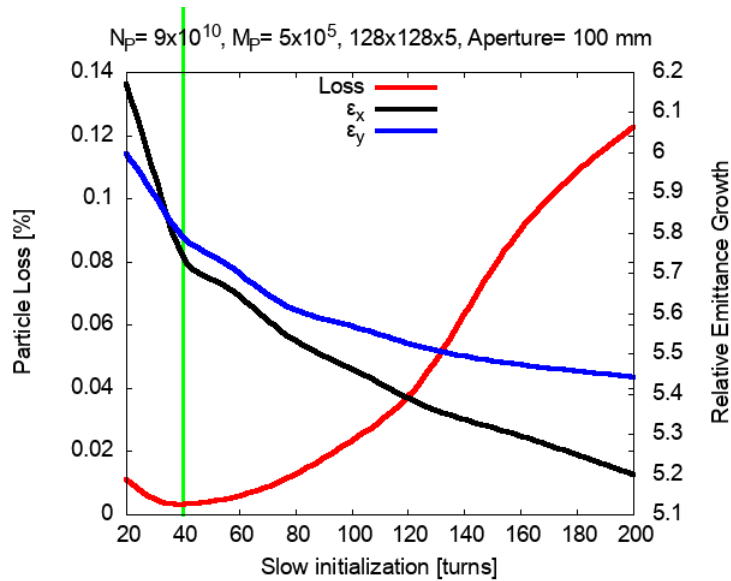


Figure 14: Loss and emittance growth as a function of the slow initialization time.

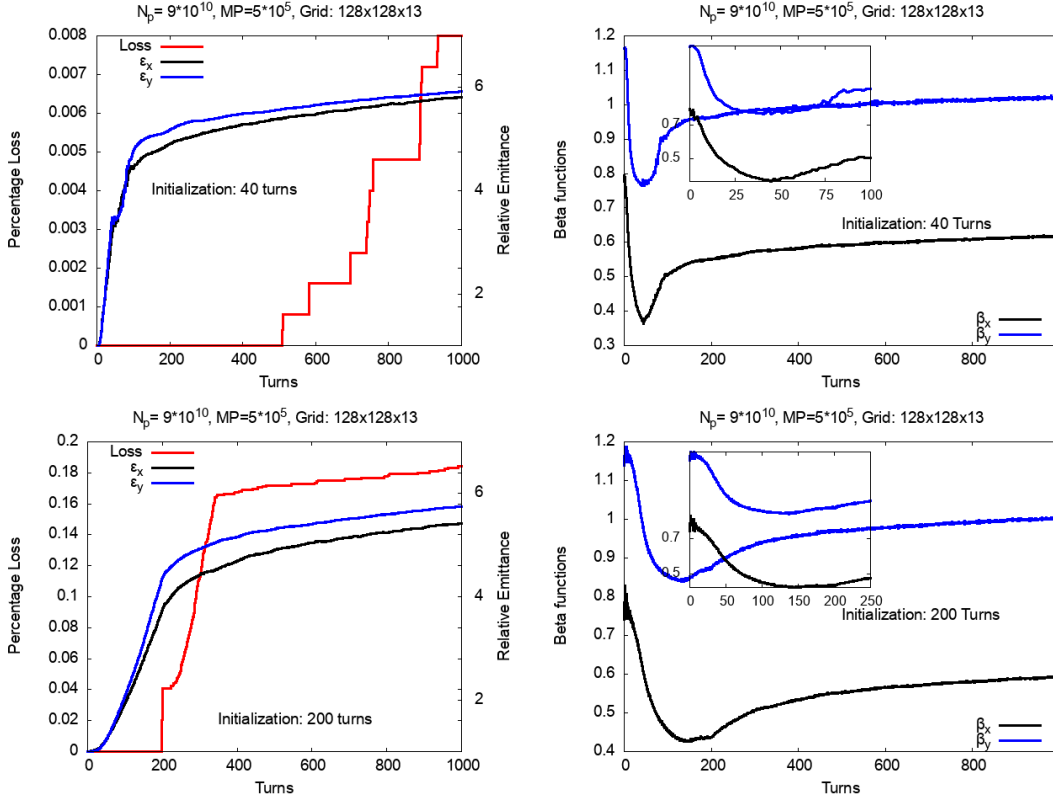


Figure 15: Left: Particle losses (left vertical scale) and relative emittance (right vertical scale), Right: Statistical beta functions for two initialization times : 40 turns (Top) and 200 turns (Bottom).

are nearly equal at this value of T_{init} , so that equipartitioning of emittances is apparently correlated with a minimum of losses. The two left plots in Fig. 15 show the evolution of the emittance growth and particle loss for two initialization times $T_{init} = 40, 200$ turns. In both cases, the vertical emittance grows slightly more than the horizontal emittance but the change is about the same for both 40 and 200 initializations. The horizontal emittance growth is slightly less at $T_{init} = 200$ turns. The losses are significantly smaller at 40 turns. The right plots in this figure shows the statistical beta functions for the same two initializations. The inset plots show that both $\langle \beta_x \rangle, \langle \beta_y \rangle$ reach different stable values depend on T_{init} and that the time to stabilize increases with T_{init} . The top plots in Fig. 16 shows the initial and final horizontal and vertical profiles while the bottom plot shows the difference profiles between the two initializations in both planes. These plots show that after stabilization, the profiles do not change much with the initialization time.

4.3 Convergence Tests on simulation parameters

There are three parameters in the pyORBIT simulations that need to be tested for convergence : the number of macroparticles (M_p), the number of space charge kicks per betatron wavelength (N_{sc}) and the number of spatial grid points ($N_x \times N_y \times N_z$) used while

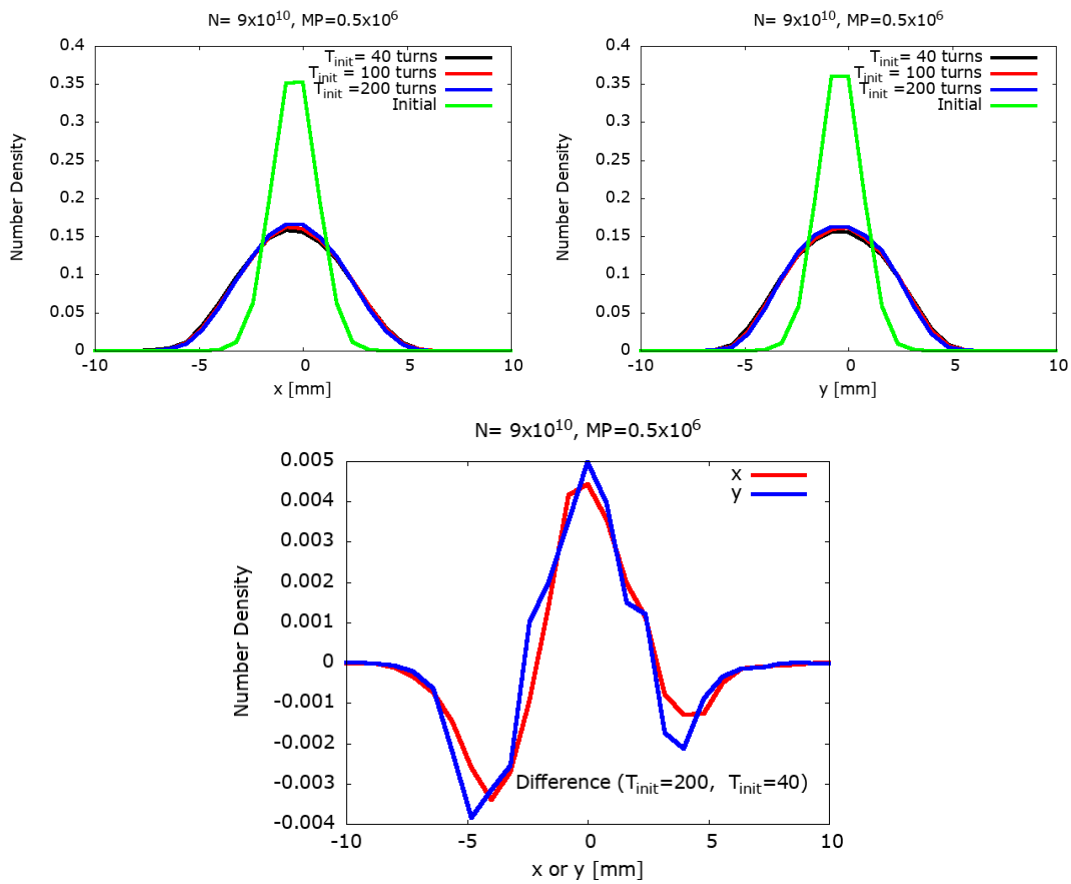


Figure 16: Top: Density profiles for the initialization times that correspond to the smallest and largest losses; horizontal (left) and vertical (right). Bottom: difference in the profiles.

solving Poisson’s equation. Increasing M_P reduces the statistical noise and a larger N_{sc} ensures better sampling of the spatially dependent space charge kick as the transverse beam sizes vary along the ring. These two parameters have practical upper limits set by the computing time required. Increasing the number of grid points improves the resolution of the sampling of the space charge force. An insufficient number of grid cause a increase the numerical noise with a PIC code and can lead to artificial emittance growth. Similarly, due to round-off error accumulation, an excessive grid size can lead to the same problem. There is therefore an optimum sampling that provides low enough noise [5] for reasonable computational cost. The number of grid points in each plane is distributed from $-3\sigma_u$ to $3\sigma_u$, $u = x, y, z$ for Gaussian distributions in these planes, although about 1% of the macro-particles may be at larger amplitudes. We can do a rough estimate of the grid spacing to avoid grid heating by requiring grid spacing to be comparable or less than the Debye length $\Delta x < \lambda_D \sim 60\mu\text{m}$, cf. Table 1. Setting the grid spacing $\Delta x = 6\sigma_x/N_x$ requires

$$N_x > 6 \frac{\sigma_x}{\lambda_D} \simeq 222$$

This sets the scale of N_x ; the value we use is determined by the convergence tests described below.

We describe tests performed to determine parameters that ensure convergence by examining the behavior of beam losses and emittance growth. The convergence tests are performed at full intensity (9×10^{10}) and we use the slow initialization procedure described above with $T_{init} = 40$ turns.

pyORBIT assigns at least one space charge kick to each element so there are at least a hundred space charge kicks around the IOTA ring. The exact number of space charge kicks per element is determined by the ratio $L/\Delta s$ where L is the element length and Δs is a step length input parameter. Summing over the elements and dividing by the integer part of the tune, ~ 5 in IOTA yields the number of space charge kicks per betatron period, N_{SC} . For example, $\Delta s = 0.1\text{m}$ yields $N_{SC} = 119$ while at a lower resolution choosing $\Delta s = 0.4\text{m}$ yields $N_{SC} = 56$.

Figure 17 shows the particle loss and emittance growth for different numbers of macro-particles, grid sizes and space charge kicks. Other parameters that are held constant are shown at the top of each plot. We observe that losses are generally more sensitive to changes in parameters than the emittance. The losses plot shows that 10^5 macro-particles and a grid size of $64 \times 64 \times 5$ insufficient. The emittance plot shows that 10^5 macro-particles suffices while a $64 \times 64 \times 5$ grid yields a lower emittance than finer grids.

Fig. 18 shows the dependence of particle loss and emittance growth on the macro-particle number M_P with the number of grid points fixed at $128 \times 128 \times 5$ and number of space charge kicks N_{sc} per betatron wavelength fixed at 63. Over the range $10^3 \leq M_P < 5 \times 10^5$, the loss is reduced by about a factor of four while at larger values of $M_P \geq 5 \times 10^5$ the loss changes by less than 0.05%. The emittance stays nearly constant over this latter range from which we conclude that $M_P = 5 \times 10^5$ is the minimum number of macro-particles required. Fig. 19 shows the variation of the loss and emittance growth with the number of grid points $N_x = N_y$ with the number of macro-particles fixed at 10^6 and N_{sc}

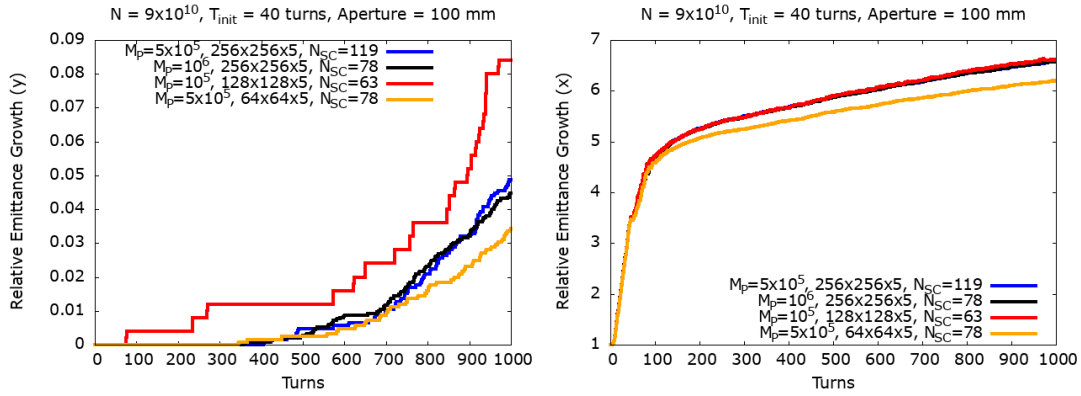


Figure 17: Particle losses (left) and horizontal emittance growth (right) for different combinations of macro-particle numbers, grid sizes and number of space charge kicks. The vertical emittance has the same dependence as the horizontal emittance on these parameters.

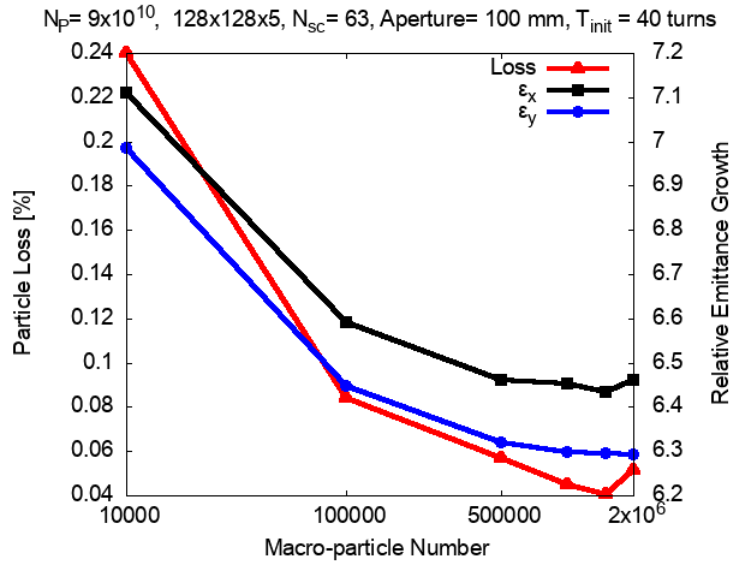


Figure 18: Particle loss (red; left vertical scale) , relative emittance growth (black, blue; right vertical scale) after 1000 turns as a function of the macro-particle number varying over three orders of magnitude, shown on a log scale. The constant parameters are shown at the top.

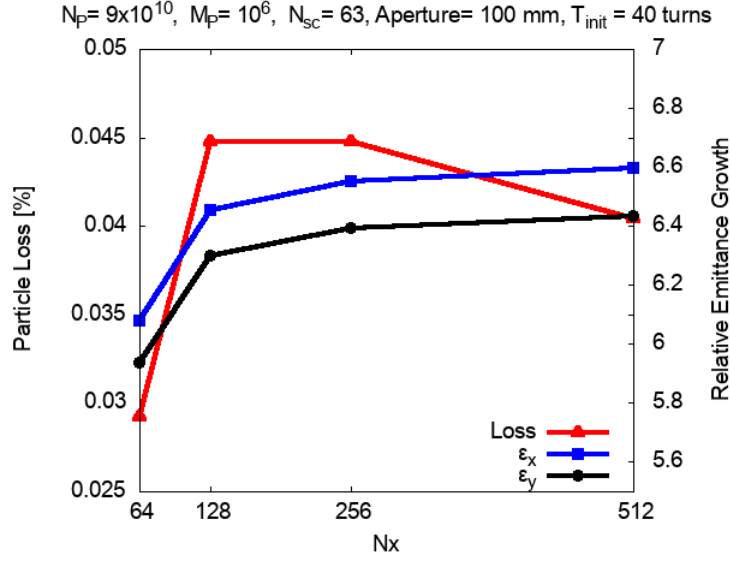


Figure 19: Particle loss (red) , relative emittance growth (black, blue) after 1000 turns as a function of the grid parameter $N_x = N_y$, $N_z = 5$. The constant parameters are shown at the top.

fixed at 78. The emittance stays nearly constant for $128 \leq N_x \leq 512$ and the loss fluctuates by less than 0.005% over this range. This shows that 128 is the minimum number of grid points required. Fig. 20 shows the convergence with respect to N_{sc} with the macro-particle number $M_p = 10^6$ and grid size $128 \times 128 \times 5$ held constant. We observe that the loss fluctuates by $\sim 0.015\%$ while the emittances are nearly constant with $N_{sc} \geq 63$.

5 Small amplitude tune shifts and tune footprints

The goal of this exercise is to validate the accuracy of pyORBIT's space charge model by comparing the tune shift obtained by tracking to analytical predictions. As mentioned earlier, the only nonlinearity in the lattice model arises from space charge.

It has been pointed out [7] that chaotic motion is observed for small amplitude particles in PIC codes due to numerical noise. One can expect that the effects of this chaos to mostly disappear when ensemble averaging over many particles to calculate statistical variables such as emittances, beam sizes etc. There are likely few analytical estimates of emittance growth with nonlinear space charge.

Keeping in mind the above caveat for tunes at small amplitudes, tune shifts as a function of amplitude due to space charge can nevertheless be compared to analytical results for a Gaussian distribution.

The analytical small amplitude tune shift for a transverse Gaussian distribution is:

$$\Delta Q_{0,sc} = \frac{r_p}{\beta_k \gamma^2 \epsilon_N} \bar{\lambda}_L N_p R \quad (5.1)$$

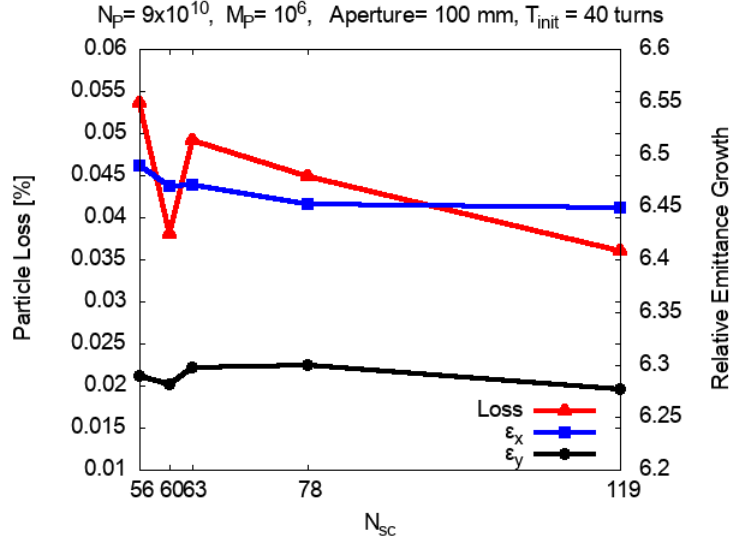


Figure 20: Particle loss (red), relative emittance growth (black, blue) after 1000 turns as a function of the number of space charge kicks N_{SC} per betatron period. The constant parameters are shown at the top. The values of N_{SC} correspond to the step length parameter $\Delta s = 0.1, 0.2, 0.3, 0.4$ m discussed in the text.

where r_p is the classical proton radius, β and γ are Lorentz factors, R the effective machine radius, ϵ_N the normalized transverse emittance, N_p the number of protons per bunch, and $\bar{\lambda}_L$, the effective line density. In all tests, the reference particle at $(z = 0, dE = 0)$ does not move.

The exact position dependent line density $\lambda(z)$ for a longitudinal water-bag distribution and the average effective line density $\bar{\lambda}_L$ are given by [6]

$$\lambda(z) = \frac{1}{C_{wb}} \sqrt{\cos(k_{rf}z) - \cos(k_{rf}z_{max})} \quad (5.2)$$

$$\bar{\lambda}_L = \frac{1}{2C_{wb}} \sqrt{1 - \cos(k_{rf}z_{max})} \quad (5.3)$$

$$C_{wb} = \frac{4}{k_{rf}} \sqrt{1 - \cos(k_{rf}z_{max})} E \left(\frac{1}{2} k_{rf} z_{max} \middle| \csc^2 \left(\frac{1}{2} k_{rf} z_{max} \right) \right) \quad (5.4)$$

Here k_{rf} is the rf wave-number z_{max} is half the total bunch length and E is the complete elliptic integral of the second kind. In the limit of short bunches where $k_{rf}z_{max} \ll 1$ the water-bag distribution reduces to the elliptic distribution with a normalized density

$$\lambda(z) = \frac{2}{\pi z_{max}} \sqrt{1 - \left(\frac{z}{z_{max}} \right)^2} \quad (5.5)$$

The IOTA bunch is long and fills the bucket, so the elliptic distribution is not a good approximation.

For the water-bag distribution with $z_{max} = 4.5$ meters, the effective line density $\bar{\lambda}_L = 0.083$. Plugging in all known parameters, and accounting for the fact that ϵ_N changes

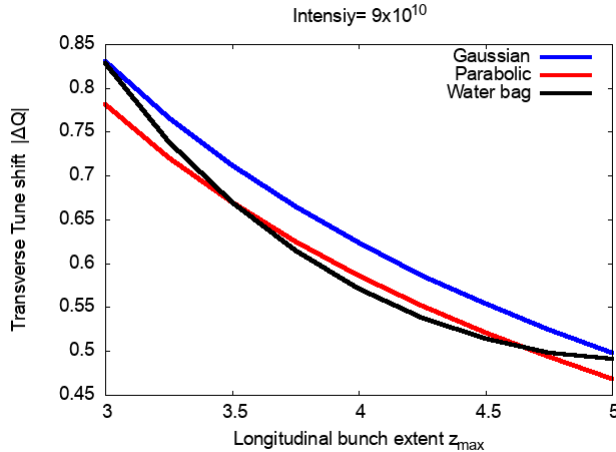


Figure 21: Absolute transverse tune shift as a function of the longitudinal extent z_{max} for different distributions.

during tracking we calculate the small amplitude tune shift:

$$\Delta Q_{0,sc} = \frac{-1.09 * 10^{-17} * N_p}{\langle \epsilon_N \rangle} \quad (5.6)$$

where $\langle \epsilon_N \rangle$ denotes a suitably time-averaged emittance. We find that the transverse tune shift does not depend very sensitively on the longitudinal distribution. Figure 21 shows the absolute value of the space charge tune shift as a function of the maximum longitudinal extent z_{max} of a bunch for different distributions. At $N_b = 9 \times 10^9$ and $z_{max} = 4.5\text{m}$, the tune shifts are $\Delta Q_{sc,0} = (-0.554, -0.521, -0.514)$ for the Gaussian, parabolic and water-bag distributions respectively.

5.1 Small Amplitude Tune Shift

We calculate the transverse tunes using two different methods. First we use the definition of the tune as the number of transverse oscillations in one revolution to calculate the small amplitude tune. The bunch is initialized as shown in Figure 8. It is tracked for 500 turns and the first 200 turns are used for slow initialization. At this point, a test particle is added at a nearly zero amplitude $(0.01\sigma_x, 0, 0.01\sigma_y, 0, 0, 0)$. The bunch is tracked for an additional 1000 turns, during which all monitors in the IOTA lattice record the transverse position of the test particle at every turn. Counting the total number of zero crossings and dividing by twice the number of turns monitored yields a lower bound for the small amplitude tune. Fig. 22 shows the betatron oscillations over two turns for two bunch intensities. The number of zero crossings allows us to extract the tune with a precision of 0.25 over the following ranges: (a) Zero and 10^9 intensities: $5.25 < Q_x < 5.5$, (b) 10^{10} intensity: $5.0 < Q_x < 5.25$ and (c) 10^{10} intensity: $4.75 < Q_x < 5.0$. Table 3 shows the tunes calculated from betatron oscillations with higher precision using 1000 turns and the fractional tunes calculated with an FFT using the same number of turns. The FFT tune is averaged

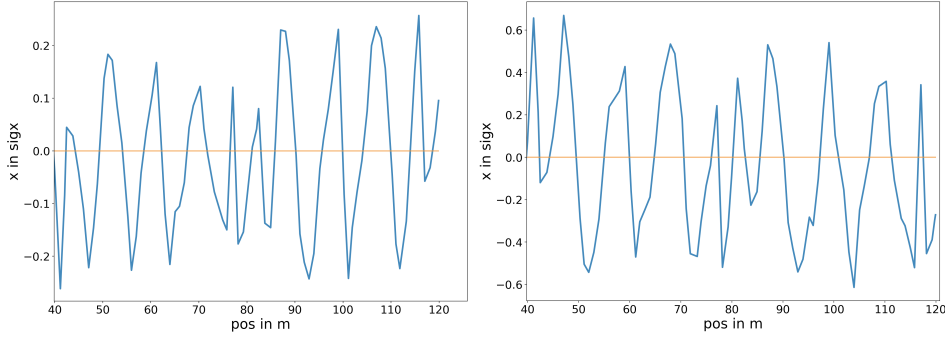


Figure 22: Betatron oscillations over two turns : (a) Zero intensity (left), 21 zero crossings (b) Bunch charge 9×10^{10} (right), 19 zero crossings.

Intensity	Small amplitude tunes Q_x, Q_y	
	Betatron Oscillations	FFT
0	(5.300, 5.301)	(0.2988, 0.3)
10^9	(5.2681, 5.267)	(0.2663, 0.265)
10^{10}	(5.037, 5.023)	(0.039, 0.035)
9×10^{10}	(4.873, 4.973)	(0.951, 0.975)
Intensity	Small amplitude tune shifts	
	Theory	Simulation
0	(0.0, 0.0)	(0.0, 0.001)
10^9	(-0.037, -0.037)	(-0.03, -0.034)
10^{10}	(-0.29, -0.29)	(-0.262, -0.266)
9×10^{10}	(-0.514, -0.514)	(-0.50, -0.536)

Table 3: Small amplitude tunes from betatron oscillations and the fractional tunes from an FFT, in both cases using 1000 turns of data. Also shown are the small amplitude transverse tune shifts calculated from theory and simulations at different intensities.

over 100 particles placed at the same small amplitude in an effort to reduce the effects of chaotic motion reported in [7].

5.2 Tune Footprints

The amplitude-dependent tunes for a transverse Gaussian distribution can be calculated analogously to those from a head-on beam-beam interaction between two Gaussian beams.

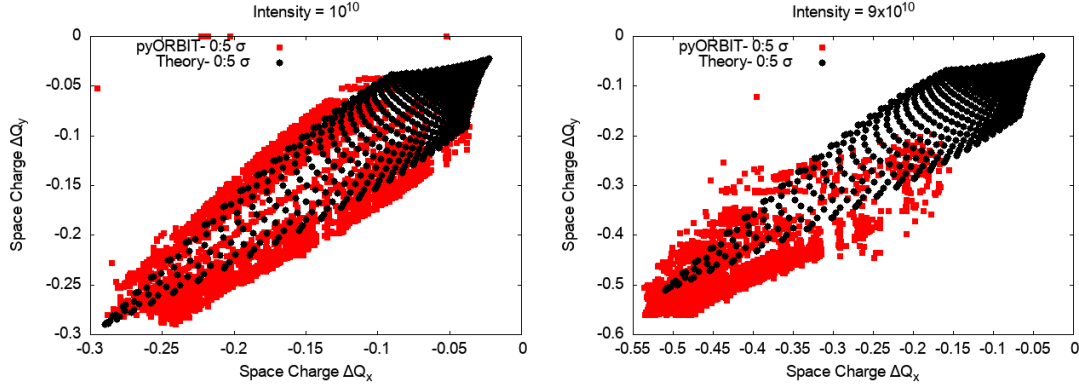


Figure 23: Tune shifts with amplitude due to space charge from pyORBIT and theory at intensity 10^{10} (left) intensity 9×10^{10} (right). In both cases, the pyORBIT simulation ranges from $0-5\sigma$ (due to aperture limitations) and the theoretical footprint using Eqs.(5.7) and (5.8) extends to 5σ .

For a round beam, the tunes are [6]

$$\Delta Q_{x,sc} = \Delta Q_{0,sc} \int_0^\infty \exp[-(\alpha_x + \alpha_y)u] I_0(\alpha_y u) [I_0(\alpha_x u) - I_1(\alpha_x u)] du \quad (5.7)$$

$$\Delta Q_{y,sc} = \Delta Q_{0,sc} \int_0^\infty \exp[-(\alpha_x + \alpha_y)u] I_0(\alpha_x u) [I_0(\alpha_y u) - I_1(\alpha_y u)] du \quad (5.8)$$

$$\alpha_x = \frac{a_x^2}{4}, \quad \alpha_y = \frac{a_y^2}{4}, \quad a_x = \frac{x}{\sigma_x}, \quad a_y = \frac{y}{\sigma_y} \quad (5.9)$$

where a_x, a_y are the dimensionless amplitudes. Slightly more complex expressions exist for non-round beams, but those will not be used here.

The tune footprints are calculated with a bunch initially tracked for 1000 turns (including 40 turns of slow initialization, as described in Section 4.2) to allow emittance to stabilize and get closer to an equilibrium value. After this initial period, 5000 test particles are injected with initial conditions distributed along semi-circular arcs on radii from $(0-5)\sigma$ in $x-y$ space; particles initially at larger amplitudes would be lost at the physical aperture. The bunch and the test particles are tracked for another 1000 turns to obtain the fractional tunes using an FFT. At intensity $N_b = 9 \times 10^{10}$, the lattice tune is shifted to $Q_x = 5.46$ and $Q_y = 5.44$ to avoid the presence of resonances (possibly “stochastic resonances”) within the footprint. The tune footprints in Figure 23 show both theoretical and pyORBIT simulation results. The theoretical footprint also extends to 5σ . At intensities at below 10^{10} and below, the simulated footprint matches the theoretical footprint quite well but is wider. At 9×10^{10} , the asymmetry between $\Delta Q_{x,sc}, \Delta Q_{y,sc}$ in the simulated tunes increases with $|\Delta Q_{x,sc}| < |\Delta Q_{y,sc}|$, possibly due to the dispersion contribution to the horizontal beam size. Simulations of the emittance growth (discussed in Section 4) show that the final emittance is about 6.4 times larger than the initial emittance; this was the value assumed for ϵ_N in the theoretical footprint. The small amplitude tuneshift in Table 3 shows that the difference between the average of the simulated tune shifts $(\Delta Q_{x,sc} + \Delta Q_{y,sc})/2$

and the theoretical value is 0.004, a few times 0.001, the precision of the FFT calculation; however, the simulated footprint is significantly wider (by about 0.05 in tune units) than the theoretical footprint. At large amplitudes close to 5σ , the simulated tune shifts are larger than expected from theory. Some differences are to be expected since the theoretical model assumes a zero length bunch, so does not include the longitudinal dependence of the transverse space charge force. Nor does the theory include the effect of dispersion but this can be done easily, if required. Finally there is numerical noise in the PIC code which will impact the tune footprint which is calculated without averaging.

6 Conclusions

- The symplectic nature of single particle tracking with pyORBIT was checked for both the linear lattice and the lattice with octupoles using two measures: $\det[\mathbf{J}] - 1$ and $\text{Max}[\mathbf{J}^T \mathbf{S} \mathbf{J} - \mathbf{S}]$. \mathbf{J} is the Jacobian matrix of the transfer map and \mathbf{S} is the symplectic matrix. On both measures, the pyORBIT model's deviation from symplecticity is $O(10^{-11})$ for the linear lattice. This deviation increases to $O(10^{-4})$ with the octupoles. By comparison, the same tests using MADX show the same measures to be $O(10^{-8})$ for the linear lattice and $O(10^{-4})$ with octupoles.

The variance of the Hamiltonian is similar with both codes.

The transverse tunes calculated with octupoles in the lattice agree to within 10^{-3} over the stable region of phase space. The boundary where the differences are about an order of magnitude larger is close to the dynamic aperture; the tune difference map resembles an FMA map produced by a single code. The dynamic apertures found by the two codes agree remarkably to within 1% with or without synchrotron oscillations (see Table 2).

- The main purpose of using pyORBIT is for its space charge tracking which is based on a PIC style method. Such codes are not symplectic and indeed find that the deviations from symplecticity are about two orders of magnitude larger than in the single particle version with octupoles. This shows that the space charge tracking results cannot be used with confidence for long-term tracking. However we expect that short term results ($\sim 10^3 - 10^4$ turns) should be reliable for parametric studies.
- The IOTA lattice we have used for the space charge studies is the original linear lattice. This lattice is not matched to the space charge lattice functions and the mismatch increases with the bunch intensity. At full intensity, this mismatch leads to almost ten-fold increase in transverse emittance and the beam loss is about 1% after 1000 turns. Most of the emittance increase occurs over the first few turns due to a large mismatch. In order to reduce the emittance growth and loss in this mismatched lattice, we introduced a slow initialization time T_{init} during which the charge on each macro-particle is increased linearly over time. We find that this reduces both the emittance growth to about five-fold and especially the losses to less than 0.1% over the same time. The optimum initialization time T_{init} is found to be 40 turns for which the loss is about 0.01%.

- With the optimum slow initialization time fixed at 40 turns, we tested for convergence by examining the behavior of the emittance growth and the particle loss as one of three important parameters was varied while keeping the other two parameters fixed. In general, the loss shows more fluctuations than the emittance as the parameters are varied. However with the aperture set at 100 mm (to avoid an emittance reduction due to significant losses), the losses are typically around 0.05% and fluctuations at the 10% level are not significant. Figure 18 shows the minimum number of macroparticles needed is 5×10^5 keeping the grid size fixed at 128x128x5 and the number of space charge kicks per betatron wavelength $N_{sc} = 63$. Figure 19 shows that the emittance growth stabilizes at the grid size 128x128x5 and does not change much upto grid sizes 512x512x5. Finally 20 shows that convergence is achieved with $N_{sc} = 78$ but $N_{sc} = 63$ can be used without significant differences. This value of N_{sc} is quite a bit larger than typically expected.
- The space charge small amplitude tune shifts and tune footprints agree quite well with theoretical predictions up to a bunch intensity of 10^{10} . At the maximum intensity of 9×10^{10} , the small amplitude tune shifts still agree reasonably but the simulated footprint is somewhat wider than the theoretical footprint (see Fig 23b). In making the comparison, we have to take into account the emittance growth at full intensity.

To sum up, we have established that pyORBIT matches MADX very closely for single particle tracking. The space charge model of pyORBIT is not designed to be symplectic, so naturally the deviations from symplecticity are significant. Nevertheless, we find that the space charge tuneshifts calculated over 1000 turns with pyORBIT agree reasonably well with theory. We believe therefore that over the time scale of 10^3 - 10^4 turns, pyORBIT can be used with confidence. This report dealt only with the transverse space charge effects, so similar issues will be need to be examined when the longitudinal space charge effects are included.

Acknowledgments

We thank the Lee Teng internship program at Fermilab which enabled the participation of Runze Li in this study. Fermilab is operated by Fermi Research Alliance LLC under DOE contract No. DE-AC02CH11359.

A Appendix A: Additions and changes to pyORBIT

This appendix describes additions and modifications to the original pyORBIT program[1], including a new dipole edge element and the optional elimination of nonlinearities in the dipole and quadrupole elements. A number of bugs found in pyORBIT have been corrected. All additions and modifications are included in our repository on github [1].

A.1 Adding Dipole Edge Element in pyORBIT

In the original version of pyORBIT, there is no distinct dipole edge element. In order to add dipole edge corrections to pyORBIT, changes were made in three sections of the code: the parser, the tracking method and the python element wrapper.

In the parser, found in the source file `py-orbit/py/orbit/teapot/teapot.py` a new dipole edge element class named `DipedgeTEAPOT` was introduced. When an element of type "dipedge" is encountered by the pyORBIT MADX sequence parser, a new `DipedgeTEAPOT` element is instantiated. Its parameters, `e1`, `h`, `hgap`, and `fint` correspond to the dipole edge parameters in MADX. `e1` is the rotation angle for the pole face, `h` is the curvature $\frac{1}{\rho}$, `hgap` is the half gap of the (upstream/downstream) bending magnet and `fint` is the edge field integral. The edge is implemented as a thin element i.e. it has length 0. The `DipedgeTEAPOT` tracking method invoked by the tracker is defined in the source file `py-orbit/src/teapot/teapotbase.cc`. The implementation considers only first order terms. The corresponding transfer matrix is[8]:

$$\begin{aligned} \begin{bmatrix} x \\ x' \end{bmatrix} &= M = \begin{bmatrix} 1 & 0 \\ h \tan(e) & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ x'_0 \end{bmatrix} \\ \begin{bmatrix} y \\ y' \end{bmatrix} &= M = \begin{bmatrix} 1 & 0 \\ -h \tan(e - \psi) & 1 \end{bmatrix} \begin{bmatrix} y_0 \\ y'_0 \end{bmatrix} \\ \psi &= \text{fint} \cdot 2 \cdot \text{hgap} \cdot h \cdot \frac{1 + \sin^2(e)}{\cos(e)} \end{aligned}$$

where e is either the entrance or the exit edge angle.

A new wrapper method was added to make the edge element and its methods available from python. The implementation can be found in the file `py-orbit/src/teapot/wrap_teapotbase.cc`.

pyORBIT now handles the dipole edges in the IOTA lattice MADX sequence. When these edges are accounted for, the bare lattice tune Q_y calculated by pyORBIT is 5.3, in agreement with MADX; without them pyORBIT predicts 5.42.

A.2 Nonlinear Effects of Magnet Edge

MADX and pyORBIT use different approaches for tracking. MADX operates converts every element to thin kicks. In contrast pyORBIT treats some elements as thick lenses.

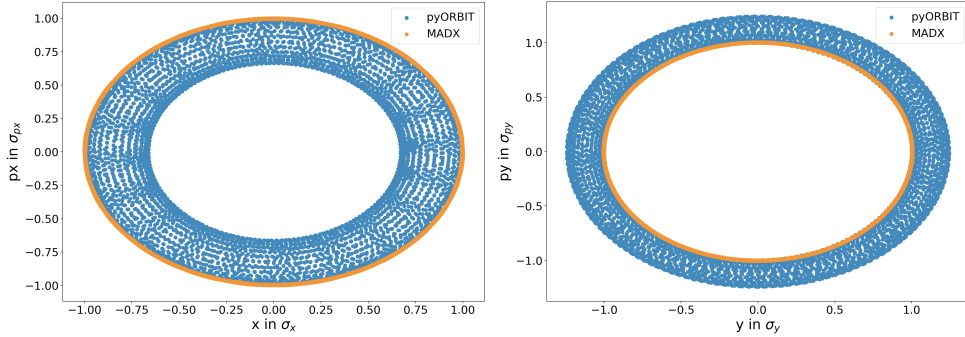


Figure 24: Single particle phase space diagram by MADX and pyORBIT

These elements are further subdivided; a variable called `nparts` controls the subdivision. For example, a dipole with `nparts = n`, is treated as follows by pyORBIT:

```
Fringe In, Entrance, Dipole_1, . . . . ., Dipole_n-2, Exit,
Fringe Out
```

`Fringe In` and `Fringe Out` account for edge effects the other `n` parts (including entrance and exit) are main body of this dipole element. The length (and bending angle for dipole) is distributed among each of these `n` parts according to $l_{entrance} = l_{exit} = \frac{l_{others}}{2} = \frac{L}{2*(n-1)}$. The entrance contains only linear transformation and other elements have both linear and nonlinear part. Since every element needs an entrance and an exit, `nparts` ≥ 2 .

When tracking a test particle with initial conditions $(\sigma_x, 0, \sigma_y, 0, 0, 0)$ for 5000 turns in both MADX and pyORBIT, the two codes produce very different results.

As shown in figure 24, the trajectories in the $x-p_x$ and $y-p_y$ planes show differences on the order of 30%. The shape of phase space trajectory produced by pyORBIT suggests the excitation of a resonance by presence of nonlinearities in pyORBIT. After removing all sources of nonlinearity in pyORBIT, we identified the `Fringe In` and `tt Fringe Out` sections of the dipole and quadrupole element as the culprit. Their contributions are defined in the file `py-orbit/src/teapot/teapotbase.cc`. Removing these methods completely produced the figure 25.

MADX does not consider nonlinear fringe effects; however, for small a ring like IOTA they may be significant. Additional tests would be required to assess the importance of nonlinear edge effects in IOTA.

A.3 Other Changes

A number of bugs in the pyORBIT code were found and corrected. They are listed here.

- In the IOTA sequence generated by MADX, the strength of solenoids is set to the default value 0; this triggered a divide by 0 error in pyORBIT. As a workaround the strength of such solenoids can be manually set to a very small value like 10^{-10} .

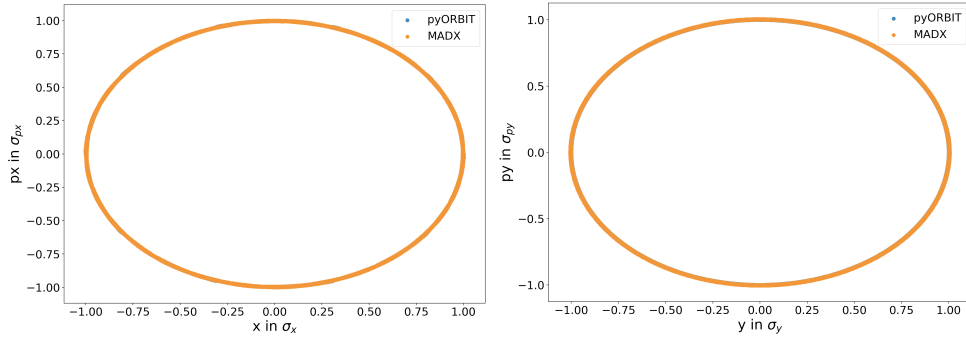


Figure 25: Single particle phase space diagram by MADX and pyORBIT after removing the magnet edge effect

- In the file `py-orbit/py/orbit/teapot/teapot.py`, the method `initialize()` of class `ApertureTEAPOT` initializes the `Aperture` object with `Aperture(shape, dim[0], dim[1], 0.0, 0.0)`, however the `Aperture` object is created by `Aperture(int shape, double a, double b, double c, double d, double pos)` so a total of 6 parameters are needed. A dummy value 0.0 was added to fix this.
- the treatment of apertures in pyORBIT still has issues with sequence files generated by MADX. The pyORBIT parser looks for elements of type "Aperture" in the sequence; however, in the IOTA lattice an aperture can be defined using a marker with attribute "aperture". The current workaround is to remove all apertures defined in the sequence and add them back later using `python code code`.
- Other bugs fixed in pyORBIT include 3 methods in `py-orbit/py/orbit/aperture/ApertureLatticeRangeModifications.py`, which add apertures upstream and downstream of all elements, and one method in the file `py-orbit/src/orbit/ Aperture/Aperture.cc`, which checks for particles intercepted at the aperture and moves them to the `lostbunch`.

References

- [1] <https://github.com/PyORBIT-Collaboration/py-orbit>
- [2] V. Danilov and S. Nagaitsev, *Nonlinear Accelerator Lattices with One and Two Analytic Invariants*, Phys. Rev. ST-AB, **13**, 084002 (2010)
- [3] S. Antipov et al, *IOTA (Integrable Optics Test Accelerator): Facility and Experimental Beam Physics Program*, JINST **12**, T03002 (2017)
- [4] I. Hoffmann and O. Boine-Frankenheim, *Grid dependent noise and entropy growth in anisotropic 3d particle-in-cell simulation of high intensity beams*, Phys. Rev. ST-AB, **17**, 124201 (2014)

- [5] F. Kesting and G. Franchetti, *Propagation of numerical noise in particle-in-cell tracking*, Phys. Rev. ST-AB, **18**, 114201 (2015)
- [6] T. Sen, unpublished notes
- [7] F. Schmidt et al, *Code bench-marking for long-term tracking and adaptive algorithms*, Proceedings of HB2016., p 357 (2016)
- [8] Karl L. Brown. A First- and Second-Order Matrix Theory for the Design of Beam Transport Systems and Charged Particle Spectrometers. SLAC Report-75