inside OPA version 4.2.4

Andreas Streun, PSI, February 4, 2026

Contents

1 Introduction

This note documents the beam dynamics algorithms used in and partially developed for OPA during the past decades. It may help OPA users to understand the meaning of the results and the limitations of the models used. For understanding of the code implementation a documentation is available [1]. In this report `[unit>procedure]` tells where to find the relevant documentation or piece of code. The code itself is available at `https://github.com/opa-code/opa-4`.

The remark $\Longrightarrow$ *to do...* gives suggestions were things could be completed or improved in future.

2 Beam optics

OPA uses the most simple accelerator model: the lattice is a concatenation of elements, which are either long ($L \geq 0$) and linear and described by a symplectic transfer matrix, or thin ($L = 0$) and non-linear and described by a single kick map. Long non-linear elements are modeled as a series of drifts and kicks (which corresponds to a basic second order symplectic integrator).

OPA is a code for transverse beam dynamics and does not include longitudinal dynamics (i.e. acceleration). An offset from design momentum $u = \Delta p / p_o$ is considered as constant (adiabatic approximation). The transfer matrices are of size 5×5 where the fifth row is just $(0, 0, 0, 0, 1)$. Thus they propagate the particle vector $(x, x', y, y', u)^T$ without changing u.

OPA uses the highly relativistic, paraxial and large-ring approximations as detailed in appendix E.6. It's main area of application are high energy electron storage rings.

Up to version 3 OPA was restricted to flat lattices (only vertical bends, no skew elements), where horizontal and vertical motion are decoupled and all transfer matrices are block-diagonal. In version 4 the Sagan-Rubin representation [9] of the Edward-Teng formalism [8] was implemented as a major upgrade in order to include transverse coupling: it introduces the "normal-mode transformation", where the two transverse planes are decoupled again, but they do not correspond anymore to the physical planes x and y, and therefore are denoted a and b.

Linear beam optics at any location in the accelerator in general is fully described by the orbit, which is a 4-vector, accompanied by the scaler momentum offset u, the Twiss parameters of the two transverse planes (beta and alpha-functions), and the dispersion 4-vector $\vec{D} = (D_x, D'_x, Dy, D'_y)^T$. In coupled lattices, in addition the 2×2 coupling matrix C is required; then the Twiss parameters correspond to the "normal modes".

2.1 The Periodic solution

Beam parameters along the lattice are calculated by propagating initial parameters. In a circular lattice (storage ring), the periodic solutions have to be calculated beforehand to generate the initial parameters.

The closed orbit is the fixpoint of the one-turn map including also kicks from non-linear elements. It is iteratively found by application of the standard Newton-Raphson root finding method using the *local* transfer matrix (i.e. the one turn matrix including gradient down feeds from non-linear multipoles) as the Jacobian, i.e. as the gradient for the root finder [linoplib> ClosedOrbit].

When the orbit is found, the local transfer matrix is used to find the periodic dispersion and the (normal mode) Twiss parameters as described in [9]: The 5×5 one-turn matrix is composed from a the 4×4 transverse focusing matrix R and the 4-vector of dispersion production $\vec{Z}$, with 2×2 sub-matrices non-coupling M, N and coupling m, n and horizontal and vertical 2-vectors of dispersion production $\vec{V}, \vec{w}$:

$$R = \begin{pmatrix} N & m \\ n & N \end{pmatrix} \qquad \vec{Z} = \begin{pmatrix} \vec{V} \\ \vec{w} \end{pmatrix} \tag{1}$$

A normal mode decomposition of R may be found

$$R = T^{-1} \mathcal{G}^{-1} \mathcal{O} \mathcal{G} T \tag{2}$$

where $\mathcal{O}$ is the matrix containing the eigen-tunes

$$\mathcal{O} = \begin{pmatrix} O_a & 0 \\ 0 & O_b \end{pmatrix} \quad \text{and} \quad O_{a,b} = \begin{pmatrix} \cos \mu_{a,b} & \sin \mu_{a,b} \\ -\sin \mu_{a,b} & \cos \mu_{a,b} \end{pmatrix} \tag{3}$$

and $\mathcal{G}$ is the matrix that contains the local normal-mode beta functions. It is given by

$$\mathcal{G} = \begin{pmatrix} G_a & 0 \\ 0 & G_b \end{pmatrix} \quad \text{and} \quad G_{a,b} = \begin{pmatrix} \frac{1}{\sqrt{\beta_{a,b}}} & 0 \\ \frac{\alpha_{a,b}}{\sqrt{\beta_{a,b}}} & \sqrt{\beta_{a,b}} \end{pmatrix} \tag{4}$$

where the indices a and b label the two eigen modes. It is related to the normalized and periodic sigma matrix

$$S_{a,b} = G_{a,b}^{-1} G_{a,b}^{-T} = \begin{pmatrix} \beta_{a,b} & -\alpha_{a,b} \\ -\alpha_{a,b} & \gamma_{a,b} \end{pmatrix} \quad \text{with} \quad \gamma_{a,b} = \frac{1 + \alpha_{a,b}^2}{\beta_{a,b}}. \tag{5}$$

The coupling matrix T and its inverse T^{-1} are given by [9]

$$T = \begin{pmatrix} gI & -C \\ C^+ & gI \end{pmatrix} \quad \text{and} \quad T^{-1} = \begin{pmatrix} gI & C \\ -C^+ & gI \end{pmatrix}, \tag{6}$$

with the 2×2 identity matrix I and the 2×2 coupling matrix C. Its symplectic conjugate C^+ and the scalar g are given by

$$C^+ = C^{-1} \det C, \quad \text{and} \quad g^2 = 1 - \det C. \tag{7}$$

Explicit calculations are described in [9]. For the coupling terms they derived

$$g = \sqrt{\frac{1}{2} \pm \frac{1}{2} \frac{|\mathrm{Tr}(M - N)|}{h}} \qquad C = \mp \frac{\mathrm{sign}(\mathrm{Tr}(M - N))}{g\,h}(m + n^+) \tag{8}$$

$$\text{with} \quad h = \sqrt{(\mathrm{Tr}(M - N))^2 + 4\det(m + n^+)}. \tag{9}$$

Existence of a periodic solution corresponds to a real value of h. The normal mode transformation is not always unique, and a second solution exists for $\det(m + n^+) > 0$ as indicated by the lower sign in the equations. Usually the upper sign is valid in weakly coupled lattices. Knowing the coupling terms, the normal-mode beta functions are calculated as[1]:

$$A = G_a^{-1} O_a G_a = \begin{pmatrix} \cos\mu_a + \alpha_a \sin\mu_a & \beta_a \sin\mu_a \\ -\gamma_a \sin\mu_a & \cos\mu_a - \alpha_a \sin\mu_a \end{pmatrix} \tag{10}$$

$$= g^2 M - g(Cn + mC^+) + CNC^+ \tag{11}$$

$$B = g^2 N + g(nC + C^+m) + C^+MC \tag{12}$$

It is then straightforward to calculate the fractional tunes from the trace of A, the sign of tune and beta function from the A_{12} element ($\beta > 0$ always), and the alpha-function from $A_{11} - A_{22}$, or in a similar way [10]. In an uncoupled lattice C is a null-matrix and $g = 1$, simply giving $A = A_x = M$ and $B = A_y = N$.

The periodic physical dispersion is obtained from

$$\vec{D} = R\vec{D} + \vec{Z} \qquad \longrightarrow \qquad \vec{D} = (I - R)^{-1}\vec{Z}. \tag{13}$$

These procedures are implemented in `[linoplib> Periodic]`, switching to `[>NormalMode]` or `[>FlatPeriodic]` with and without coupling. When building the lattice, a switch is set to activate coupled calculation if any elements have non-zero rotation angles, or if explicit rotations or solenoids are found `[globlib> MakeLattice]`. However, later it may happen that the method is changed from uncoupled to coupled due to skew quadrupolar downfeeds in non-linear multipoles from vertical orbit excursions.

Alternatively to the periodic solution, a symmetric solution can be calculated as detailed in [10], where the alpha-functions at both sides of the lattice are zero but the beta-functions may be unequal `[linoplib> Symmetric]`.

$\Longrightarrow$ *Symmetric solution is only implemented for uncoupled lattice and will give wrong results with coupling: to be extended (or to be dropped, since rarely used)*

---

[1]note, that the first version of [9] had an erratum here

2.2 Propagation

As a code for lattice design from scratch it was OPA's aim from the beginning to calculate all lattice parameters, even equilibrium parameters like emittance, by forward propagation irrespective of the existence of a periodic solution. Physically this makes no sense, but it is useful for design tasks, and therefore emphasis was placed on forward (or backward) calculations. Closing the ring to enable the periodic solution may be done as a later step in design from scratch.

In a periodic lattice, initial parameters at one locations are obtained by the procedure as described above. One can calculate them at any location, by just starting and ending the concatenation of the one-turn matrix there. This option is included for cross-checking [linoplib> PropPer], however it is slow and it is not the "OPA spirit". Fortunately, Sagan & Rubin gave a recipe [9] how to propagate the periodic solution from one location to another. In a single pass system, e.g. a transfer line, we anyway have to do forward propagation (of course, in this case equilibrium parameters really make no sense). We quote here the relevant equations from [9]. The transfer matrix from some location "1" to another location "2" is known (sub-matrices M, N, m, n), because it was used for concatenating the one-turn matrix. If the optics parameters at "1" are known, they are calculated at "2" as follows:

$$g_2^2 = \det(n_{12}C_1 + g_1 N_{12}) \tag{14}$$

If $g_2^2 < 0$ we encounter the so-called "mode flip" (see below), for the moment we assume $g_2^2 > 0$. Then a block-diagonal transfer matrix is constructed with 2×2 sub-matrices given by

$$E_{12} = \frac{g_1 M_{12} - m_{12}C_1^+}{g_2} \qquad F_{12} = \frac{g_1 N_{12} + n_{12}C_1}{g_2} \tag{15}$$

The one-turn matrices from Eq.10 propagate as

$$A_2 = E_{12}A_1 E_{12}^{-1} \qquad B_2 = F_{12}B_1 F_{12}^{-1} \tag{16}$$

Transformation to another location does not change the tune, because at any location we have $A_i = G_{ai}^{-1}O_a G_{ai}$ The transformation of the sigma matrix S_a (eq.5) follows from

$$A_2 = G_{a2}^{-1}O_a G_{a2} = E_{12}G_{a1}^{-1}O_a G_{a1}E_{12}^{-1} \tag{17}$$

$$S_{a2} = G_{a2}^{-1}G_{a2}^{-T} = E_{12}G_{a1}^{-1}(E_{12}G_{a1}^{-1})^T = E_{12}G_{a1}^{-1}G_{a1}^{-T}E_{12}^T = E_{12}S_{a1}E_{12}^T \tag{18}$$

and the same for S_b with matrices F and B. The coupling matrix propagates as

$$C_2 = (g_1 m_{12} + M_{12}C_1) F_{12}^{-1} . \tag{19}$$

The phase advance from 1 to 2 is calculated by extracting the rotation part from matrix E_{12}:

$$E_{12} = G_{a2}^{-1}O_{a12}G_{a1} \quad \longrightarrow \quad O_{a12} = G_{a2}E_{12}G_{a1}^{-1} \quad \longrightarrow \quad \cos\Delta\mu_{a12} = \mathrm{Tr}(O_{a12})/2. \tag{20}$$

Phase advances accumulate, i.e. $\mu_{a2} = \mu_{a1} + \Delta\mu_{a12}$.

If the transfer matrix from "1" to "2" is not coupling, then we simply get

$$g_2 = g_1, \quad E_{12} = M_{12}, \quad F_{12} = N_{12}, \quad C_2 = M_{12}C_1 N_{12}^{-1}. \tag{21}$$

A "mode flip" occurs for $g_2^2 < 0$. This may happen in strongly coupled lattices: depending on the sign of $\det(m + n^+)$ one or two solutions exist at different locations, and the initially chosen solution may terminate at some location, and we have to continue with the other one. At least one has to exist everywhere in a periodic lattice, otherwise there would be no periodic solution. The transfer matrix from "1" to "2" becomes off-block diagonal, which means that the modes a and b change places. We also quote the corresponding equations for propagation with a mode flip:

$$g_2^2 = \det(M_{12}C_1 + g_1 m_{12}) \tag{22}$$

Nothing can go wrong now, because this is our "fall-back" which has to exist if Eq.14 didn't! We then get the off-diagonal block matrices

$$E_{12} = \frac{g_1 n_{12} - N_{12}C_1^+}{g_2} \qquad F_{12} = \frac{g_1 m_{12} + M_{12}C_1}{g_2} \tag{23}$$

The one-turn matrices propagate "cross wise" as

$$A_2 = F_{12}B_1 F_{12}^{-1} \qquad B_2 = E_{12}A_1 E_{12}^{-1}. \tag{24}$$

The tunes are interchanged now, so the transformation of the sigma matrices follows from

$$A_2 = G_{a2}^{-1}O_b G_{a2} = F_{12}G_{b1}^{-1}O_b G_{b1} F_{12}^{-1} \tag{25}$$

$$S_{a2} = G_{a2}^{-1}G_{a2}^{-T} = F_{12}S_{b1}F_{12}^T \qquad S_{b2} = G_{b2}^{-1}G_{b2}^{-T} = E_{12}S_{a1}E_{12}^T. \tag{26}$$

Thee coupling matrix transforms as

$$C_2 = (g_1 M_{12} - m_{12}C_1^+) E_{12}^{-1}. \tag{27}$$

The 4×4 transfer matrix for propagation of the normal modes becomes block off-diagonal

$$\begin{pmatrix} 0 & F_{12} \\ E_{12} & 0 \end{pmatrix} = \begin{pmatrix} G_{a2}^{-1} & 0 \\ 0 & G_{b2}^{-1} \end{pmatrix} \cdot \begin{pmatrix} 0 & O_{a12} \\ O_{b12} & 0 \end{pmatrix} \cdot \begin{pmatrix} G_{a1} & 0 \\ 0 & G_{b1} \end{pmatrix} \tag{28}$$

Extracting the rotation matrices gives

$$O_{a12} = G_{a2}F_{12}G_{b1}^{-1} \qquad O_{b12} = G_{b2}E_{12}G_{a1}^{-1} \tag{29}$$

After this operation the propagation proceeds in flipped mode, i.e. the phase advances are accumulated cross-wise as

$$\mu_{a2} = \mu_{a1} + \Delta\mu_{b12} \qquad \mu_{b2} = \mu_{b1} + \Delta\mu_{a12} \tag{30}$$

6

until the next mode flip happens– the "flip status" changes at each mode flip. Actually in case of mode flips one cannot tell which mode is horizontal or vertical, there are just two orthogonal modes, and depending if the calculation starts flipped or not, the total accumulate tunes are interchanged.

The tune increments from Eq. 30 may show weird numbers with little physical meaning, but they will sum up to give the eigen-tunes from Eq. 3 after one turn. As emphasized in [9] there should be an even number of mode-flips in the lattice, otherwise the accumulated tunes make no sense. Therefore, when propagating in flip mode, an intentional mode flip back should be done at the next occasion (i.e. at a coupling element) or at the end of the lattice.

$\implies$ *The integer part of the tune becomes wrong in some cases with mode flip. It's not yet understood why this happens and how it could be improved.*

These procedures are implemented in [elemlib> Propagate], branching to [>MCC_prop] with and [>MBD_prop] without coupling, and using [>PhaseAdvance] for the phase advances.

2.3 Projections

The projections of the beta functions to the physical coordinates x, y are derived from the expressions in [9] [elemlib> GetBeta12]

$$\beta_{xa} = g^2 \beta_a \qquad \beta_{xb} = \beta_b \left(\left[C_{11}^2 - C_{12} \frac{\alpha_b}{\beta_b} \right]^2 + \left[\frac{C_{12}}{\beta_b} \right]^2 \right) \tag{31}$$

$$\beta_{yb} = g^2 \beta_b \qquad \beta_{ya} = \beta_a \left(\left[C_{22}^2 - C_{12} \frac{\alpha_a}{\beta_a} \right]^2 + \left[\frac{C_{12}}{\beta_a} \right]^2 \right) \tag{32}$$

$$\tag{33}$$

Here β_{xa}, β_{yb} are the principal modes, which become β_x, β_y without coupling, and β_{xb} and β_{ya} are the cross-talks from the other transverse plane which disappear without coupling.

A normalized normal-mode sigma-matrix $\bar{\sigma}$ has just the normal mode emittances as diagonal elements. The real sigma matrix σ in physical quantities is obtained by transformation to local normal mode betas and from normal mode coordinates to real space coordinates [linoplib> N2SfromOp]:

$$\sigma = \begin{pmatrix} \langle x^2 \rangle & \langle xx' \rangle & \langle xy \rangle & \langle xy' \rangle \\ \langle xx' \rangle & \langle x'^2 \rangle & \langle x'y \rangle & \langle x'y' \rangle \\ \langle xy \rangle & \langle x'y \rangle & \langle y^2 \rangle & \langle yy' \rangle \\ \langle xy' \rangle & \langle x'y' \rangle & \langle yy' \rangle & \langle y'^2 \rangle \end{pmatrix} = T^{-1} \mathcal{G}^{-1} \begin{pmatrix} \epsilon_a & 0 & 0 & 0 \\ 0 & \epsilon_a & 0 & 0 \\ 0 & 0 & \epsilon_b & 0 \\ 0 & 0 & 0 & \epsilon_b \end{pmatrix} \mathcal{G}^{-T} T^{-T} \tag{34}$$

The projected emittances in the physical dimensions are given by the determinants of the 2×2 block diagonal matrices:

$$\epsilon_x = \sqrt{\langle x^2 \rangle \langle x' \rangle^2 - \langle xy \rangle^2} \tag{35}$$

7

In order to include contributions from energy spread $\sigma_u = \sqrt{\langle u^2 \rangle}$ and dispersion the 4×4 matrices can be extended to 5×5 including dispersions in the fifth column of $T^{-1}\mathcal{G}^{-1}$ and σ_u^2 as 5,5-element of the normalized sigma matrix, or the contributions are added explicitly. For visualization of the different contributions to the envelope the equivalent formulation with projected beta functions is preferred [linoplib> PlotEnv]:

$$\sigma_x = \sqrt{\epsilon_a \beta_{xa} + \epsilon_b \beta_{xb} + (\sigma_u D_x)^2} \qquad \sigma_y = \sqrt{\epsilon_b \beta_{yb} + \epsilon_a \beta_{ya} + (\sigma_u D_y)^2} \tag{36}$$

Finally, the tilt angle θ of the beam in the $(x, y)-$plane is given by [linoplib> FillBetaTab]

$$\tan 2\theta = \frac{2\langle xy \rangle}{\langle x^2 \rangle - \langle y^2 \rangle} \tag{37}$$

3 Elements

The lattice transfer matrix is obtained by concatenation (multiplication) of the single elements matrices. On-axis only linear magnets contribute, these are the general bending magnet with drift, dipole, quadrupole as special cases, the edge focusing kicks, the rotation and within some approximation the solenoid.

Off-axis the orbit receives kicks from (thin) non-linear multipoles, and the *local* transfer matrix at the orbit has to include focusing terms from the local regular and skew gradients encountered off-axis in non-linear multipoles.

Off-axis also quadrupoles and higher multipoles become "bending magnets", because they bend the orbit, and thus will produce dispersion. Also dispersion production in a real bending magnet will be modified off-axis.

Orbit correctors are small dipoles to create or correct orbit excursions. Uundulators are a series of rectangular dipoles alternating with drifts. Kickerr magnets are time dependent dipoles or multipoles.

The 5×5 transfer matrix of a linear magnet in general is given by

$$R = \left(\begin{array}{cc|c} M & m & \vec{V} \\ n & N & \vec{w} \\ 0 & 0 & 1 \end{array} \right) \tag{38}$$

with horizontal and vertical 2×2 transfer matrices M and N and the dispersion production 2-vectors $\vec{V}$ and $\vec{w}$ 2-vectors. For a regular magnet which does not introduce coupling, the transfer matrix is block-diagonal, i.e. $m = n = 0$ and $\vec{w} = 0$. A sixth row and column for pathlength are omitted sine OPA does not contain longitudinal dynamics.

3.1 On-axis

The general bending magnet, deflecting in the horizontal plane, has a block-diagonal matrix and does not produce vertical dispersion. Other orientations of a bending magnet in the lattice are realized by sandwiching it between coordinate rotations.

A general bending magnet has a curvature $h = 1/\rho = B_y/(B\rho)$ and a gradient $k = \partial B_y/\partial x/(B\rho)$, with $(B\rho) = p/e$ the magnetic rigidity of the particle. $h > 0$ bends the beam towards the ring center, i.e. in $-x$ direction and thus creates positive dispersion. $k > 0$ focuses horizontally, i.e. kicks a particle at $x > 0$ to $-x$ direction.

Introducing the abbreviations

$$K = h^2 + k, \qquad p = \sqrt{|K|}, \qquad q = \sqrt{|k|} \tag{39}$$

with the longitudinal coordinate s, the sub matrices are given by [22]

$$M = \begin{cases} \begin{pmatrix} \cos ps & \frac{\sin ps}{p} \\ -p\sin ps & \cos ps \end{pmatrix} \\ \begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix} \\ \begin{pmatrix} \cosh ps & \frac{\sinh ps}{p} \\ p\sinh ps & \cosh ps \end{pmatrix} \end{cases} \qquad \vec{V} = \begin{cases} \begin{pmatrix} \frac{h}{p^2}(1 - \cos ps) \\ \frac{h}{p}\sin ps \end{pmatrix} & \text{for } k > -h^2 \\ \begin{pmatrix} \frac{hs^2}{2} \\ hs \end{pmatrix} & \text{for } k = -h^2 \\ \begin{pmatrix} \frac{h}{p^2}(\cosh ps - 1) \\ \frac{h}{p}\sinh ps \end{pmatrix} & \text{for } k < -h^2 \end{cases} \tag{40}$$

and N is given by

$$N = \begin{cases} \begin{pmatrix} \cos qs & \frac{\sin qs}{q} \\ -q\sin qs & \cos qs \end{pmatrix} & \text{for } k < 0 \\ \begin{pmatrix} 1 & s \\ 0 & 1 \end{pmatrix} & \text{for } k = 0 \qquad \vec{w} = 0 \\ \begin{pmatrix} \cosh qs & \frac{\sinh qs}{q} \\ q\sinh qs & \cosh qs \end{pmatrix} & \text{for } k > 0 \end{cases} \tag{41}$$

Here $k = 0$ describes a pure dipole, $h = 0$ a quadrupole, and for $h = 0$ and $k = 0$ we recover a drift space. Defining $C_p = \cos ps$ or $\cosh ps$ and $\bar{S}_p = (\sin ps)/p$ or $(\sinh ps)/p$ depending on the sign of K, and similar for q, we can write in a compact way:

$$M = \begin{pmatrix} C_p & \bar{S}_p \\ -K\bar{S}_p & C_p \end{pmatrix} \qquad \vec{V} = \begin{pmatrix} (1 - C_p)h/K \\ h\bar{S}_p \end{pmatrix} \qquad N = \begin{pmatrix} C_q & \bar{S}_q \\ k\bar{S}_q & C_q \end{pmatrix} \tag{42}$$

[elemlib> Drift_Matrix, Quad_Matrix, Sector_Matrix, Bend_Matrix]

For a bending magnet where the entry and exit edges are not perpendicular to the beam direction (e.g. rectangular bend), we need to take the pole face rotations into account by means of thin kick matrices before and after the sector [elemlib> EdgeKick_Matrix]: :

$$M = \begin{pmatrix} 1 & 0 \\ h\tan\theta & 0 \end{pmatrix} \qquad N = \begin{pmatrix} 1 & 0 \\ -h\tan(\theta - \psi) & 0 \end{pmatrix} \tag{43}$$

Here θ is the angle between the entrance or exit edge of the magnet and the plane perpendicular to the beam. $\theta > 0$ turns the sector bend towards a rectangular bends, where $\theta_1 = \theta_2 = \Phi/2$ with $\Phi = hL$ the bending angle. The parameter ψ takes into account a finite extension of the fringe field, characterized by the parameter K_1 and the gap g [21]:

$$\psi = K_1 hg \frac{1 + \sin^2 \theta}{\cos \theta} \qquad K_1 = \int \frac{B_y(s)\,(B_0 - B_y(s))}{gB_0^2}\,ds \tag{44}$$

The rotation is a change of coordinate system. For example, a vertical bending magnets would be introduced by a 90°-rotation. The rotation matrix is given by

$$M = N = \begin{pmatrix} \cos\phi & 0 \\ 0 & \cos\phi \end{pmatrix} \qquad m = -n = \begin{pmatrix} \sin\phi & 0 \\ 0 & \sin\phi \end{pmatrix} \tag{45}$$

with ϕ the angle of rotation, [elemlib> Rotation_Matrix].

The solenoid has a longitudinal field and thus is only described correctly by the Hamiltonian of Eq.138. Using the approximation $B_s = 0$ from Sec.E.6 as implemented in OPA, the treatment is not completely correct. In our simplified framework the solenoid can be modeled as a product of a rotation matrix with a quadrupole-like matrix focusing in both planes. With the strength $k = B_s/(2B\rho)$ and $\phi = kL$ the sub-matrices are [elemlib> Sol_Matrix]

$$M = N = \begin{pmatrix} \cos^2\phi & \frac{\cos\phi\sin\phi}{k} \\ -k\cos\phi\sin\phi & \cos^2\phi \end{pmatrix} \qquad m = -n = \begin{pmatrix} -\sin\phi\cos\phi & -\frac{\sin^2\phi}{k} \\ k\sin^2\phi & -\sin\phi\cos\phi \end{pmatrix} \tag{46}$$

$\implies$ *The implementation of the solenoid is incomplete with regard to focusing and dispersion down-feeds, chromaticity and radiation integrals.*

3.2 Off-axis

The orbit $\vec{x} = (X, Y', Y, Y', U)^T$ (here $U = \Delta p/p$ is the momentum deviation of the orbit) is propagated by application of the transfer matrix in linear elements and by a series of drifts and thin kicks in non-linear elements.

The linear matrices were collected in the previous section. They are independent of the orbit, because the elements are linear. As derived in appendix E (see Eq.157) the parameters h, k and K all scale with momentum as

$$h = \frac{h_0}{1+U}, \quad k = \frac{k_0}{1+U}, \quad K = \frac{h_0^2 + k_0}{1+U} \quad \longrightarrow \quad \left.\frac{dh}{dU}\right|_{U=0} = -h_0 \qquad \text{etc.} \tag{47}$$

because h entered in a two-fold way, as curvature of the local coordinate system in a bending magnet of cylindrical symmetry and as geometric effect leading to a pseudo-focusing effect.

Dispersion is the local derivative of the orbit with respect to U. Consider the horizontal dispersion:

$$D_x = \dot{X} = \left.\frac{dX}{dU}\right|_0 = R_{11}\left.\frac{dX_0}{dU}\right|_0 + \left.\frac{dR_{11}}{dU}\right|_0 X_o + R_{12}\left.\frac{dX_0'}{dU}\right|_0 + \left.\frac{dR_{12}}{dU}\right|_0 X_0' + R_{15}$$

A last term $dR_{15}/dU \cdot U$ disappeared, because we take the limit $U \to 0$. Derivatives of X_0 and X_0' are just the dispersion and its derivative D_{xo} and D_{xo}' at the magnet entry, so first, third and fifth term recover the on-axis transformation. For off-axis we need the derivatives of the matrix elements R_{11} and R_{12} to get the additional dispersion production on the orbit entering the magnet at position X_o and angle X_o'.

We find for both trigonometric and hyperbolic quantities contained in the matrix elements

$$\left.\frac{dp}{dU}\right|_{U=0} = \mp\frac{K}{2p} = -\frac{p}{2} \quad\longrightarrow\quad \left.\frac{dC_p}{dU}\right|_{U=0} = \bar{S}_p\frac{Ks}{2}, \qquad \left.\frac{d\bar{S}_p}{dU}\right|_{U=0} = (\bar{S}_p - sC_p)\frac{1}{2},$$

and finally for the *additional* dispersion production $\Delta\vec{D}(s) = dX(s)/dU|_{U=0}$

$$\begin{pmatrix} \Delta D_x \\ \Delta D_x' \end{pmatrix} = \frac{1}{2}\begin{pmatrix} \bar{S}_p Ks & (\bar{S}_p - sC_p) \\ (\bar{S}_p + sC_p)K & \bar{S}_p Ks \end{pmatrix} \cdot \begin{pmatrix} X_0 \\ X_0' \end{pmatrix} \tag{48}$$

$$\begin{pmatrix} \Delta D_y \\ \Delta D_y' \end{pmatrix} = \frac{1}{2}\begin{pmatrix} -\bar{S}_q ks & (\bar{S}_q - sC_q) \\ -(\bar{S}_q + sC_q)k & -\bar{S}_q ks \end{pmatrix} \cdot \begin{pmatrix} Y_0 \\ Y_0' \end{pmatrix} \tag{49}$$

These contributions have to be added to the dispersion production vectors $\vec{V}$ and $\vec{w}$.

In the special case that $K = 0$ or $k = 0$ we get null-matrices for the dispersion productions.

The kicks from a thin $2n$-pole of integrated strengths $b_n l$ are given by

$$\Delta X' - i\Delta Y' = -b_n l(X + iY)^{n-1}, \tag{50}$$

The multipole does not change the coordinates X, Y, because it is thin, and the kick is not a function of the angles X', Y'.

Local gradients and dispersion productions are found from expansion:

$$\Delta X' = \Delta X_o' + \left.\frac{d\Delta X'}{dX}\right|_0 + \left.\frac{d\Delta X'}{dY}\right|_0 + \left.\frac{d\Delta X'}{dU}\right|_0,$$

and the same for $\Delta Y'$. The second and third terms correspond to local integrated regular and skew quadrupole gradients $k_r l$ and $k_s l$, defined by $k_r l = -d\Delta X'/dX = +d\Delta Y'/dY$ and $k_s l = +d\Delta X'/dY = +d\Delta Y'/dY$:

$$k_r l + ik_s l = (n-1)b_n l(X + iY)^{n-2} \qquad n \geq 2 \tag{51}$$

11

The last term is the dispersion production, following from $db_n/dU|_0 = -b_n$:

$$\Delta D'_x = -X'_o \qquad \Delta D'_y = -Y'_o \tag{52}$$

Explicitly for the first multipoles:

	n=1	n=2	n=3	n=4
$\Delta X'_o = -\Delta D'_x$	$-b_1 l$	$-b_2 lX$	$-b_3 l(X^2 - Y^2)$	$-3b_4 l(X^3 - 3XY^2)$
$\Delta Y'_o = -\Delta D'_y$	0	$b_2 lY$	$2b_3 lXY$	$3b_4 l(Y^3 - 3X^2 Y)$
$k_r l$	0	$b_2 l$	$2b_3 lX$	$3b_4 l(X^2 - Y^2)$
$k_s l$	0	0	$2b_3 lY$	$6b_4 XY$

$$(53)$$

The off-axis multipole transfer matrices is [elemlib> ThinSextupole, Multipole]:

$$R = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -k_r l & 1 & k_s l & 0 & -X'_o \\ 0 & 0 & 1 & 0 & 0 \\ k_s l & 0 & k_r l & 1 & -\Delta Y'_0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{54}$$

3.3   Other elements

Combined: The element type "Combined" is just a bending magnet. It has deflection angle and focusing strength, but in addition it may have a sextupole moment. Internally it is subdivided in a number of slices with thin sextupoles between [elemlib> Combined].

$\Longrightarrow$ *For historical reasons the elements "Bending" and "Combined" exist in parallel, basically Combined includes Bending, except a second fringe field parameter K_2 which actually was never ever used. So these elements may be merged into one.*

Corrector and monitor: a corrector is just a short multipole of integrated field $b_1 l$ ("H-corrector") or $a_1 l$ ("V-corrector"). Unlike a multipole the V-corrector is not obtained by rotation of a H-corrector, but it is an element of its own. Correctors kick the orbit but the impact on beam optics is neglected, except that they also create local dispersion.

If correctors are given the names CH or CV all appearances in the lattice become independent elements (automatically renamed CH001 etc.) to be addressed individually in orbit correction (see below). The same applies to BPMs, element type "Monitor", if the name is MON.

Undulator: the alternating field of an N-pole wiggler (or undulator) can be well approximated by a series of $2N$ rectangular dipole magnets, where the end poles on both sides are attenuated in order to center the wiggling orbit on axis. The field of half pole k is $B_k = p_k \hat{B}$ with $\hat{B}$ the maximum field occurring in the central poles only, and

$$p_k = \left\{ \frac{1}{4}, -\frac{3}{4}, 1, -1 \ldots 1, -1, \frac{3}{4}, -\frac{1}{4} \right\} \quad k = 1, 2 \ldots 2N$$

for optimum centering of the electron beam.

A general filling factor, i.e. the ratio of rectangular bend length to half pole length $(= \lambda/2)$ is defined by

$$f_n = \frac{1}{\lambda} \int_o^\lambda \left| \frac{B_y(s)}{\hat{B}} \right|^n ds \tag{55}$$

where f_1 affects the orbit, and f_2, f_3 the radiation integrals, see Sec.4.1. An ideal sinusoidal wiggler has

$$f_1 = \frac{2}{\pi} = 0.637 \qquad f_2 = \frac{1}{2} = 0.5000 \qquad f_3 = \frac{4}{3\pi} = 0.424. \tag{56}$$

Kicker: a kicker is a time-dependent $2n$-pole, in the simplest case $(n = 1)$ a dipole. It has a time constant, where we assume a half-sine shaped pulse of duration τ, and a delay δ, so its time dependence is

$$f(t) = \cos \pi \frac{t-\delta}{\tau} \quad \text{for} \quad -\frac{\tau}{2} < t - \delta < \frac{\tau}{2} \quad \text{else} \quad 0$$

OPA has two implementations, a regular multipole and a non-linear kicker of a type as described e.g. in [26]. In the latter case a value $\hat{x}$ is given, where the field reaches maximum deflection $\Delta x'$. Then the integrated field (i.e. the actual kick) and the local gradient vary as

$$b_1 l(x,t) = \Delta x'\, f(t)\, \sin^{n-1} Ax \qquad b_2 l(x,t) = \Delta x'\, A\, f(t)\, \sin^{n-2} Ax \cos Ax \tag{57}$$

with $A = \pi/2/\hat{x}$, for $|x| < 2\hat{x}$ and $n \geq 2$. So near the origin this kicker looks like a regular $2n$-pole.

A regular multipole-kicker is identified by $\hat{x} = 0$ and instead the multipole strength is given. Then Eq. 53 applies and is just multiplied with the time dependence $f(t)$ [elemlib> Kicker], [tracklib> TMatKick].

4 Radiation integrals

Radiation integrals were derived in detail in [2]. The results are summarized in Table 1. Here, the normal-mode dispersion $\vec{\mathcal{D}}$ and the normalized normal-mode dispersion $\vec{\tilde{D}}$ are used:

$$\vec{\mathcal{D}} = T\vec{D} \qquad \vec{\tilde{D}} = \mathcal{G}T\vec{D} \tag{58}$$

Without coupling $(T = 1, \ a = x, \ b = y)$ and on-axis $(X(s) = Y(s) = 0)$ the well known elementary formulae are recovered:

$$I_{4[x]} = \int D_x(s)(2k(s)h(s) + h(s)^3)\, ds - \sum_{i=1,2} D_{xi} h_i^2 \tan \theta_i \qquad I_{4y} = 0$$

$$I_{5[x]} = \int \mathcal{H}_x(s)|h(s)|^3\, ds \qquad I_{5y} = 0$$

13

Table 1: Radiation integrals for a general bending magnet with curvature h_0 and focusing strength k including coupling and orbit distortions.

$$
\begin{aligned}
I_1 &= \int (h_X D_x + h_Y D_y)\, ds \qquad I_2 = \int |h(s)|^2\, ds \qquad I_3 = \int |h(s)|^3\, ds \\
I_{4a} &= \int [C_x(s) g\mathcal{D}_a(s) + C_y(s)(D_y(s) - g\mathcal{D}_b(s))]\, ds - \sum_{i=1,2} \tan\theta_i |h_i|^2 g\mathcal{D}_{ai} \\
I_{4b} &= \int [C_y(s) g\mathcal{D}_b(s) + C_x(s)(D_x(s) - g\mathcal{D}_a(s))]\, ds - \sum_{i=1,2} \tan\theta_i \bar{C}_{xi}(D_{xi} - g\mathcal{D}_{ai}) \\
I_{5a} &= \int \vec{\tilde{D}}_a^T(s) \cdot \vec{\tilde{D}}_a(s)\, |h(s)|^3\, ds \qquad I_{5b} = \int \vec{\tilde{D}}_b^T(s) \cdot \vec{\tilde{D}}_b(s)\, |h(s)|^3\, ds
\end{aligned}
$$

$$
\text{with} \qquad h_X(s) = h_0 + kX(s) \qquad h_Y(s) = -kY(s) \qquad |h(s)| = \sqrt{(h_X(s))^2 + (h_Y(s))^2}
$$

$$
C_x(s) = h_X(s)(|h(s)|^2 + 2k(s)) \qquad C_y(s) = h_Y(s)(|h(s)|^2 - 2k(s))
$$

$$
\text{with} \qquad \mathcal{H}_x = (G_x \vec{D}_x) \cdot (G_x \vec{D}_x)^T = \gamma_x D_x^2 + 2\alpha_x D_x D' + \beta_x D_x'^2
$$

On-axis only the bending magnets contribute to the integrals. Off-axis basically all magnets contribute, but, as discussed in [2], the non-linear multipoles are negligible. Analytic solutions for the integrals in a general bending magnet with constant parameters h,k were implemented in earlier versions of OPA, however these equations (generated by an algebraic code), were rather lengthy and not intuitive, further it was not possible to include off-axis contributions. Therefore a semi-numeric implementation was chosen by using a Simpson-integration over the bending magnets. The number of points is estimated from the phase advance over the bending magnet in order to catch variations of the optical functions [elemlib> Bending, Quadrupole].

The radiation integrals are sums over the contributions from all magnets. If mode-flips occur in the lattice, each flip sets or releases a flag to add the integral contributions crosswise.

Finally the radiation equilibrium parameters are calculated from the integrals [linoplib> LinOp]

$$
\text{Momentum compaction factor} \qquad \alpha = I_1/C \tag{59}
$$

$$
\text{Energy loss per turn [eV]} \qquad U_o = C_\gamma \gamma^4 I_2 \tag{60}
$$

$$
\text{Damping partition numbers} \qquad J_{a,b} = 1 - \frac{I_{4a,b}}{I_2} \qquad J_u = 4 - J_a - J_b \tag{61}
$$

$$
\text{Relative energy spread} \qquad \sigma_u^2 = C_q \gamma^2 \frac{I_3}{I_2 J_u} \tag{62}
$$

$$
\text{Normal mode emittances [m]} \qquad \epsilon_{a,b} = C_q \gamma^2 \frac{I_{5a,b}}{I_2 J_{a,b}} \tag{63}
$$

with the constants (μ_o is vacuum permeability) `[globlib> OpaInit]`

$$C_\gamma = \frac{ec^2\mu_o}{6\pi} = 9.60 \cdot 10^{-10}\text{Vm} \qquad C_q = \frac{55\hbar}{32\sqrt{3}m_oc} = 3.83 \cdot 10^{-13}\text{m} \tag{64}$$

4.1 Simplified undulator integrals

Since optical parameters usually do not change much over one period of an undulator, and since an undulator may have many poles, we introduce some approximations in order to simplify and speed up calculations.

We define the peak curvature $\hat{h} = \hat{B}/(B\rho)$ and the peak gradient $\hat{k} = \hat{B}'/(B\rho)$. For the first integral, the path length, we need only the dispersion in the direction of deflection, i.e. D_x:

$$\Delta I_{1k} = \int_0^{\lambda/2} D_x h(s)\, ds \approx f_1 \frac{\lambda}{2} \langle D_x \rangle p_k \hat{h} \tag{65}$$

For the average of the dispersion and other optical functions we simple use the mean of values before and after the pole, $\langle D \rangle \approx (D_0 + D_1)/2$. This neglects the wiggling of the dispersion inside the undulator, which is justified for undulators with small periods as used in low emittance rings. Either the undulator is located in a dispersion free region, than its inner dispersion is too small to have a signficant impact on the first, fourth and fifth integral, or it is located in a dispersive region, but then its inner dispersion oscillation is small compared to the offset from outer dispersion.

The second and third integral do not depend on optical functions and thus are simply given by

$$\Delta I_{2k} = f_2 \frac{\lambda}{2} (p_k \hat{h})^2 \qquad \Delta I_{3k} = f_3 \frac{\lambda}{2} |p_k \hat{h}|^3 \tag{66}$$

where we used the factor from Eq.55. The fourth integrals for a short, rectangular magnet of length L and gradient k were derived in [2]:

$$I_{4a} \approx 2g\mathcal{D}_a hkL \qquad I_{4b} = I_{40} - I_{4a} \qquad I_{40} = 2D_x hkL \tag{67}$$

The h^3 term usually present in the fourth integral is canceled by the rectangular edge effects in a short magnet. Since both h and k follow the sinusoidal variation, we have to use f_2 and get

$$\Delta I_{4ak} = f_2 \frac{\lambda}{2} g \langle \mathcal{D}_a \rangle p_k^2 \hat{h}\hat{k} \qquad \Delta I_{4bk} = f_2 \frac{\lambda}{2} \langle D_x \rangle p_k^2 \hat{h}\hat{k} - I_{4ak} \tag{68}$$

The fifth integrals contain the generalized dispersion invariants $\mathcal{H}_{a,b} = \vec{\tilde{D}}_{a,b} \cdot \vec{\tilde{D}}_{a,b}^T$,

$$\Delta I_{5ak} = f_3 \frac{\lambda}{2} \langle \mathcal{H}_a \rangle |p_k \hat{h}|^3 \qquad \Delta I_{5bk} = f_3 \frac{\lambda}{2} \langle \mathcal{H}_b \rangle |p_k \hat{h}|^3 \tag{69}$$

15

5   Optics calculations

Once a periodic solution is found, the 17 optics parameters (orbit, normal mode beta functions, dispersions, coupling matrix, and momentum offset) are defined at the begin/end of the lattice and are then propagated along the lattice to each element. Alternatively, the parameters can be given explicitly at the begin (or at any other location of the lattice using an "optics-marker" element) to be propagated forward (and/or backwards). Phase advances, chromaticities and radiation integrals are accumulated in propagation [`linoplib> LinOp`].

As an interactive design tool OPA emphasizes visualization by resolving the optics *inside* the elements. Zoom functions allow the optical functions and the length scale to be magnified, for example to visualize the wiggling dispersion inside an undulator. Elements are subdivided in slices of a maximum length corresponding to one pixel on the screen [`opalinop> TOptic.Zoom`], and matrices are calculated for the slices and also for the skipped part of the element outside the plot window [`elemlib> SliceSet` etc.]. If an element is rotated, e.g. a skew quadrupole, the rotations are applied to each slice to show the optical functions in the original coordinate systems, otherwise each rotation would cause a discontinuity in the plot (of course, explicit rotations can be entered as well to do this). Furthermore, a 45°-rotation corresponds to a massive coupling and may cause a mode-flip, whereas a (weak) skew quadrupole represented by a quadrupole sandwiched between ±45°rotations may cause only moderate coupling.

Matching of optical functions from one point to another is an important feature when building a lattice: functions at a final location, and if needed, also at an intermediate location, may be selected, target values are entered, and a couple of "knobs" is selected, which can be focusing strengths of elements, lengths of drift spaces or variables in order to build "super-knobs". Of course, the number of knobs has to be as large or larger than the number of target functions. The minimizer then establishes a sensitivity matrix and uses the most efficient knobs to set up a linear system, which is solved by LU-decomposition. The procedure thus corresponds to a Newton-Raphson minimization using the sensitivity matrix as Jacobian (like the closed orbit finder) and converges quadratically. However, due to the non-linearity of the problem it may not converge if it starts too far away from the target values, then a manual pre-matching should be done (which is also useful to understand the problem...) or the step size has to be reduced to make convergence more robust on the expense of speed [`omatching> TMatch.Step`].

$\Longrightarrow$ *The matching module is one of the eldest part of OPA. Coupling is not included, i.e. it works only for flat lattices. A more powerful and flexible optimizer should be implemented which also may include range limits for the knobs.*

Momentum dependence of the solutions is studied by a series of calculations for different momentum offsets and several parameters can be plotted and fitted by polynomials [`momentumlib> Calculate`]. If path length was selected the polynomial coefficients corresponding to the

orders of the momentum compaction factor are saved internally [`momentumlib> MakePlot> fit_and_plot`] and can be used to calculate and draw the bucket [`opabucket> CalcBucket`]. The calculations to get fixpoints and separatrix are explained in appendix D. However, OPA does not contain longitudinal dynamics. The bucket is only used for visualization and estimation of momentum acceptance and bunch length.

If a periodic solution is found, a tune-diagram shows the working point and the resonances in its vicinity [`opatune`]. After a momentum scan, the resulting chromatic tune footprint is shown in the diagram. Other modules also will plot their results in the tune diagram (see below).

In order to shift the working smoothly by small amounts, a 2×2 sensitivity matrix can be set up, which connects the tune shifts to a scaling factor for the two groups of horizontally and vertically focusing quadrupoles [`otunematrix`].

6   Non-linear optimization

The Hamiltonian modes for sextupoles have been calculated in first and second order by J. Bengtsson [16, 17] and confirmed by C.-x. Wang [18], also including chromatic quadrupole contributions, and octupoles and decapoles were supplemented in [19], so we don't need to repeat these (rather lengthy) results here.

The terms are of the form (taking a first order sextupole term as an example)

$$h_{jklmp} = \sum b_3 l \beta_x^{(j+k)/2} \beta_y^{(l+m)/2} D_x^p \exp i[(j-k)\mu_x + (l-m)\mu_y] \quad [+\text{quadrupoles}\ldots], \quad (70)$$

where the sum is over all (thin) sextupole kicks. Second orders are more complicated since they are generated by interference of the first order terms.

OPA includes the 10 first order sextupole terms (2 linear chromaticities and 8 resonances) and 13 second order sextupole, first order octupole terms (2 quadratic chromaticities, 3 amplitude dependant tune shifts (ADTS) and 8 octupolar resonances). Decapoles are only considered with regard to cubic chromaticity. For the resonances, the modes and their complex conjugates are considered as one term. Linear, quadratic and cubic chromaticities are calculated by numeric differentiation [`chromlib> ChromDiff`] whereas all other quantities are calculated analytically [`chromlib> DriveTerms`].

The *effective* Hamiltonian is obtained by multiplication of the modes with betatron amplitudes. The n-th order is

$$H_n = \sum_{j+k+l+m+p=n} h_{jklmp} (2J_x)^{(j+k)/2} (2J_y)^{(l+m)/2} \delta^p .$$

In order to make these terms comparable with regard to their impact on non-linear dynamics, the relevant betatron amplitudes as given by machine apertures and the relevant momentum range have to be considered.

The actual impact is given by the equation of motion in action angle variables, i.e. $\dot{\phi} = dH/dJ$, how the angle variable is affected. In order to provide a practical scaling for relative weighting of the different terms, the derivative dH/dJ is thus used. With regard to coupling terms containing both J_x and J_y, an average

$$\bar{J}^{(j+k+l+m)/2} = J_x^{(j+k)/2} J_y^{(l+m)/2}$$

is defined, assuming that J_x and J_y are of comparable magnitude. Note, that this empirical approach is not intended to provide an exact solution but only to guide the minimizer. For optimization of the sextupole pattern, a penalty function is constructed, which is a function of all sextupole strengths:

$$P = \sum_{jklmp} \left(\omega_{jklmp} A_{jklm}, \delta^p \left| h_{jklmp} - t_{jklmp} \right| \right)^2 + P_o \tag{71}$$

where the amplitude factor is given as [chromlib> HamScaling].

$$A_{jklm} = \frac{j+k+l+m}{2} \cdot \frac{(2J)_x^{(j+k)/2} (2J_y)^{(l+m)/2}}{\bar{J}}$$

The factors ω_{jklmp} are weights manually attached to the corresponding modes. For the non-resonant modes, where $\Im(h) = 0$, a target value t_{jklmp} can be given, which is always zero for the resonant modes [chromlib> Penalty].

Finally, a term just summing up all sextupole strenghts is added

$$P_o = \omega_o \sum (b_3 l)^2$$

with an adjustable weight ω_o, because this turned out to be useful for an overall reduction of higher orders and to avoid excessive sextupole strength.

Calculations are usually done for one superperiod of the lattice. Scaling the results to many periods of a storage ring requires multiplication of the resonant terms with a complex factor, see Appendix A. The phase independant terms are just multiplied with the number of periods, of course.

A non-linear Powell-optimizer [20] including limits by asymptotic scaling of knobs is used to suppress the penalty function by varying the sextupole strength. Chromaticity correction can be done separately or on top by a 2×2 matrix connecting two selected chromatic sextupole families and the linear chromacities.

The first order octupole terms form a linear system, which is solved by SVD [20] with optional filtering of weight factors to select hard or soft solutions (complete solutions at high octupole strength or incomplete at moderate strengths). It can run in a "slave mode" to the Powell minimizer to make the octupoles follow and, for example, keep the ADTS constant.

$\implies$ *This two-fold scheme has a historical background. It would be possible and interesting, and probably much faster, to run SVD on the first order sextupole system and even on the tangential second order system, using the local Jacobian.*

$\implies$ *The non-linear calculations and optimizations are only valid for uncoupled lattices without vertical dispersion at the non-linear multipoles.*

7 Tracking

Three particle tracking routines are implemented, `[opatrackps]` for tracking single particles in phase space, fourier transform and identification of resonances, `[opatrackda]` for evaluating the dynamic apertures, and `[opatracktt]` for calculation of Touschek lifetime based on local momentum acceptance obtained from tracking. These programs mainly control the GUI, most of the calculations is done in unit `[tracklib]` used by all three.

In order to speed up calculations, series of linear elements (between non-linear thin kicks) are concatenated to transfer matrices `[tracklib> TrackingMatrix]`. In tracking the matrices are multiplied to the particle coordinates, and then a non-linear (or time dependent) thin kick is applied `[tracklib> TMatKick]`.

$\Longrightarrow$ *Particle loss is only checked at the non-linear kick locations, not at peak beta locations, which are usually at quadrupole centers.*

For phase space and dynamic aperture tracking, first the physical acceptance is calculated, i.e. the silhouette from the linear projection of all apertures to the trackpoint as described in Appendix B, `[tracklib> AmpKappa]`, `[opatrackda> TtrackDA.Silhouette]`. For tracking one can select if this acceptance is used as limitation (this would give dynamic aperture including physical limits) or if a manually set aperture is used (this would give the pure dynamic aperture if the set value is sufficiently large). The range for probing aperture or phase space is then set based on this choice `[tracklib> Acceptances]`.

When the reference momentum (momentum of particle to be tracked) is changed, matrix concatenation and physical acceptance has to be redone `[tracklib> Initdpp]`. Dynamic aperture tracking has four modes, x vs. y with optional, fixed $\delta = \Delta p/p$ offset, or x vs. δ, or y vs. δ and a 3-d scan x vs. y for a set of momenta. The momentum dependent modes require calculation of momentum dependent physical aperture for a specified momentum range `[tracklib> Acc_dpp]`.

Tracking may start at any location, even inside an element. This is in particular important for proper sampling of local momentum aperture in Touschek tracking `[tracklib> TrackingMatrix, Trackpoint]`. In phase space tracking it can be useful to observe the non-linear eigenfigure inside an element.

$\Longrightarrow$ *This feature is only implemented for phase space and Touschek tracking but not for dynamic aperture tracking (not really needed)* .

7.1 Phase space tracking

Starting coordinates of a particle are given either manually or stepped up when using the ADTS loop `[opatrackps> TtrackP.butTushClick]`. After tracking a number of turns (which is a power of 2), a fast Fourier transform (FFT) of the particle coordinates at the trackpoint is calculated [20]. Before a sine window is applied, which allows the frequencies to be interpolated,

since the peak is known to be of $\sin x/x$ shape, see [16] and [`tracklib> FindPeaks`] (and procedures called therein).

The resonances corresponding to the found frequencies are then "guessed" using the fact, that the fundamental horizontal tune is subtracted from the frequency in the horizontal spectrum (and v.v. for the vertical spectrum), i.e. a resonance $aQ_x + bQ_y$ is visible at $(a-1)Q_x + bQ_y$ in the horizontal spectrum, and at $aQ_x + (b-1)Q_y$ in the vertical spectrum [16], [`tracklib> ResoGuess`] (and procedures called therein).

Phase space tracking is also used for injection simulation: a beam ellipse can be set which is populated with a number of particles and tracked. Time dependence of kickers is included.

7.2 Dynamic aperture tracking

There are three methods of probing dynamic aperture: binary serach if fast, but may result in fake large aperture due to trapping the serach in stable islands. Grid probing is thorough on the expense of computing time, therefore, in order to early see results, the grid is not probed sequentially but refined successively [`opatrackda> TtrackDA.dagridsetup`]. Flood fill performs fast by testing the unstable area only until reaching the stable separatrix [23].

7.3 Touschek tracking

Touschek lifetime is calculated as described in appendix C.1 by evaluating the integral over bunch dimensions and local momentum aperture.

The Touschek module [`opatracktt`] at first serves as a worksheet calculating several parameters based on manual inputs or previous calculations (using bucket and coupling parameters). Touschek lifetime is integrated based on linear momentum acceptance as obtained from RF-acceptance and physical aperture limitations, see Eq.81. However, linear calculations only consider horizontal apertures assuming that there is pure horizontal dispersion. Coupling effects are only included as vertical emittance affects the scattering rate. Coulomb lifetime is calculated too by the simple average of Eq.103. It assumes an elliptical beam pipe and takes into account both apertures. Bremsstrahlung lifetime is closely related to Touschek lifetime since it also depends on momentum acceptance, see Eq.104.

In a second step local momentum acceptance is calculated from tracking. Synchrotron oscillations are not yet included, i.e. the momentum is assumed to be fixed. There are two methods:

In the classical mode Touschek scattered particles start on-axis with a sudden momentum offset from sufficiently densely sampled lattice locations and are tracked for a given number of turns or until they are lost. Binary search on momentum offset gives the local momentum acceptance.Coupling is included, since it will affect the momentum acceptance found in tracking.

In the "Fast Touschek Tracking" mode (FTT), the 3-d (x, y, δ) volume has to be calculated previously in dynamic aperture. Then Touschek scattered particles are only tracked to the

trackpoint, usually begin/end of lattice, and tested to be inside the volume. For details and limitations of this method see [23].

Touschek and bremsstrahlung lifetime are then recalculated using the tracked momentum acceptance.

$\implies$ *As a third mode, there is an improvised 5-D tracking option, varying the momentum deviation in turn k simply as $\delta_k = \delta_o \cos(2\pi\nu_s k)$, with ν_s the synchrotron tune. However, implementation of true synchrotron oscillations would be straightforward using cavity parameters set for the bucket. But setting up the tracking matrix for changed momentum in every turn as implemented now is inefficient and very slow.*

$\implies$ *Coulomb scattering only includes physical apertures and should be extended by re-using results from previous dynamic aperture calculations or by including a binary search for maximum deflection angle before starting. Furthermore, pressure profiles may be imported instead of assuming a constant pressure.*

$\implies$ *Including misalignments in tracking has been started defining a concatenated misalignment vector to be applied with the concatenated tracking matrix, but this does not work well yet. Work is in progress.*

8   Orbit correction

The module [opaorbit] implements a standard orbit correction method as used in many other lattice codes too: misalignments are applied to the elements, the closed orbit is calculated, the response matrix is established, how correctors affect the orbit as recorded at the BPMs, and "inverted" using SVD [20] in order to set the correctors and minimize or zero the orbit at the BPMs.

Assuming that elements are mounted to girder, correlated misalignments are set in a hierarchical manner distinguishing between girder absolute errors, girder-to-girder errors and element-to-girder errors. The girder structure is established by assigning each element to a "parent" girder and calculating its location on the girder. In the process compound elements, for example a magnet containing thin non-linear multipoles, are identified by checking for zero spaces between adjacent elements, and treated as rigid bodies to assign correlated misalignments, [globlib> GirderSetup]. The response matrix is calculated for the ideal lattice [opaorbit> GetResponseMatrix] and decomposed. After application of misalignments [opaorbit> Torbit.setMisalignments] the periodic closed orbit is searched [linoplib> ClosedOrbit] and corrected [opaorbit> TOrbit.orbitCorrection].

There are more features to run a loop over many error seeds [opaorbit> TOrbit.LoopAction], to get statistical data [opaorbit> TOrbit.CodStat], and to plot orbit, corrector values and misalignments.

9   Injection

Calculation of an injected beam trajectory is closely related to orbit correction, since a kicker is a time-dependent corrector, which however may not be just a dipole in case of a multipole kicker. Kickers are synchronized by adjusting their delays to the distance from the point of injection, which usually is the start of the lattice structure, [opaorbit> TOrbit.ButSyncClick]. Calculation then is done in single pass mode not periodic, of course, starting from manually set injected beam offsets and angles. Tracking of an injected beam ellipse is done in the phase space tracking module, [opatrackps> TtrackP.StartEnsemble, ButBeamRunClick].

10   Geometry Plot

Module [opageometry] shows the geometrical layout of the lattice. Presently the elements are only plotted as flat polygons. Geometric matching is performed by an SVD procedure using length and angle variables in order to adjust the lattice structure to a target position and angle [opageometry> TGeometry.ButMatchClick]. Files of geometric data can be exported and also imported to overplot different lattices.

$\implies$ *Up to now the elements are represented at flat faces. Extension to 3D-bodies and 3D-view has been prepared, but is not yet implemented. A hidden surface algorithm would be needed to get a nice 3D-plot, but perhaps this is beyond the scope of OPA.*

11   LGB optimizer

Module [opalgbedit] is used to optimize the field profile $B_y(s)$ of a longitudinal gradient bend of given angle, length and maximum field in order to obtain lowest emittance, resp. minimum I_5 radiation integral. The magnet is divided in slices, and a non-linear minimizer of Powell type [20] sets the field of the slices while maintaining the total field integral, i.e. deflection angle.

A The sextupole Hamiltonian for many periods

For the first order sextupole modes the complex factor a_N to transform the Hamiltonian mode for one period into N periods, i.e. $h^N = a_N \cdot h$ was derived by J. Bengtsson in [16]:

$$a_{\vec{m}}^N = \sum_{n=0}^{N-1} e^{in\vec{m}\vec{\mu}} = \frac{1 - e^{iN\vec{m}\vec{\mu}}}{1 - e^{i\vec{m}\vec{\mu}}}, \tag{72}$$

with $\vec{m} = (j-k, l-m)$ the mode of h_{ijkl} and $\vec{\mu} = 2\pi(Q_x, Q_y)$ the ring tune. Defining $\psi = \vec{m}\cdot\vec{\mu}/2$ this can be expressed as

$$a_{\vec{m}}^N = \frac{\sin\psi + \sin(2N-1)\psi}{2\sin\psi} + i\frac{\cos\psi - \cos(2N-1)\psi}{2\sin\psi} \tag{73}$$

In the limit $N \to \infty$ we may neglect the fast varying term and get

$$a_{\vec{m}}^\infty = \frac{1}{2} + i\frac{1}{2}\cot\psi \qquad \longrightarrow \qquad |a_{\vec{m}}^\infty| = \frac{1}{2\sin\psi} \tag{74}$$

revealing the resonance denominator.

For the second order modes we define

$$\sum_{\ominus} = \sum_{s=1}^{S}\sum_{s'=1}^{s-1} \quad ; \quad \sum_{\oplus} = \sum_{s=1}^{S}\sum_{s'=s+1}^{S} \quad ; \quad \sum_{\odot} = \sum_{s=1}^{S}\sum_{s'=1}^{S} \quad r_s r_{s'} e^{-i(\vec{m}\vec{\phi}_s + \vec{m}'\vec{\phi}_{s'})} \tag{75}$$

with $r_s = (b_3 L)_s \beta_{xs}^{\frac{(j+k)}{2}} \beta_{ys}^{\frac{(l+m)}{2}}$ and $\vec{\phi}_s = (\phi_{xs}, \phi_{ys})$ containing beta functions and phases of the sextupole s.

The second order sextupole cross talk terms contributing to the octupolar resonance $\vec{m} + \vec{m}'$ are given for one period by

$$h_{\vec{m}\vec{m}'} = iA\left(\sum_{\oplus} - \sum_{\ominus}\right) \tag{76}$$

with A a scalar factor[17]. For N periods this becomes

$$h_{\vec{m}\vec{m}'}^N = iA\left(a_{-(\vec{m}+\vec{m}')}^N\left(\sum_{\oplus} - \sum_{\ominus}\right) + b_{\vec{m}\vec{m}'}^N \sum_{\odot}\right) \tag{77}$$

with a^N as defined above (inserting $-(\vec{m} + \vec{m}')$ for $\vec{m}$) and

$$b_{\vec{m}\vec{m}'}^N = \frac{-\mathcal{S}(N, \vec{m}, \vec{m}') + \mathcal{S}(N, \vec{m}', \vec{m}) - i\cdot(-\mathcal{C}(N, \vec{m}, \vec{m}') + \mathcal{C}(N, \vec{m}', \vec{m}))}{8\sin\psi\,\sin\psi'\,\sin(\psi+\psi')}, \tag{78}$$

$$\mathcal{S}(N, \vec{p}, \vec{q}) := \sin\vec{p}\vec{\mu} - \sin(-N\vec{p})\vec{\mu} + \sin((1-N)\vec{p} + \vec{q})\vec{\mu} - \sin((1-N)\vec{p} - N\vec{q})\vec{\mu} \tag{79}$$

and $\mathcal{C}$ the same with cos instead of sin.

For $N \to \infty$ we again neglect the fast varying terms and get

$$b_{\vec{m}\vec{m}'}^\infty = \frac{-\sin 2\psi + \sin 2\psi' - i\cdot(-\cos 2\psi + \cos 2\psi')}{8\sin\psi\,\sin\psi'\,\sin(\psi+\psi')} \to |b_{\vec{m}\vec{m}'}^\infty| = \frac{1}{4\sin\psi\,\sin\psi'}\frac{\sin(\psi-\psi')}{\sin(\psi+\psi')} \tag{80}$$

23

B   Calculation of geometric acceptance

A particle touching an elliptical beampipe of half axis a_x and a_y fulfills the condition

$$\left(\frac{|x_o| + \sqrt{A_x \beta_x}}{a_x}\right)^2 + \left(\frac{|y_o| + \sqrt{A_y \beta_y}}{a_y}\right)^2 = 1$$

x_o, y_o are the closed orbit coordinates, in the error free lattice given by dispersion. Betatron amplitudes are given by

$$A_x = \gamma_x (x - x_o)^2 + 2\alpha_x (x - x_o)(x' - x'_o) + \beta_x (x' - x'_o)^2$$

with the corresponding formula for y. Betatron amplitudes thus are given by the starting conditions of the tracked particle. For coupled motion at a given and constant ratio of betatron amplitudes κ, with

$$A = A_x + A_y \qquad A_y = \kappa A \qquad A_x = (1 - \kappa)A,$$

i.e. $\kappa = 0$ pure horizontal, $\kappa = 1$ pure vertical oscillation, the maximum total amplitude $\tilde{A}(\kappa)$ accepted by the lattice is the minimum of the limitations from all the elliptical apertures of the machine:

$$\sqrt{\tilde{A}(\kappa)} = \min_k \left(-p_k + \sqrt{p_k^2 - q_k}\right) \qquad \text{with}$$

$$p_k = \frac{a_{yk}^2 |x_{ok}| \sqrt{1 - \kappa} \sqrt{\beta_{xk}} + a_{xk}^2 |y_{ok}| \sqrt{\kappa} \sqrt{\beta_{yk}}}{n_k}$$

$$q_k = \frac{(a_{yk} x_{ok})^2 + (a_{xk} y_{ok})^2 - (a_{xk} a_{yk})^2}{n_k}$$

$$n_k = a_{yk}^2 (1 - \kappa) \beta_{xk} + a_{xk}^2 \kappa \beta_{yk}$$

The contour of the geometric acceptance in the (x, y) plane at the location of the track point (t) is thus given by

$$\tilde{x}(\kappa) = x_{ot} \pm \sqrt{(1 - \kappa)\tilde{A}(\kappa)\beta_{xt}} \qquad \tilde{y}(\kappa) = y_{ot} \pm \sqrt{\kappa\tilde{A}(\kappa)\beta_{yt}} \qquad \kappa = 0 \ldots 1$$

C   Beam lifetime calculation

In a low emittance electron storage ring, the beam lifetime is determined by three processes: Touschek scattering of two electrons turning transverse into longitudinal momentum exceeding the momentum acceptance, elastic (Coulomb) scattering on residual gas nuclei causing transverse momenta leading to losses at beam pipe apertures, and inelastic (bremsstrahlung) scattering on residual gas nuclei leading to loss of momentum exceeding the momentum acceptance.

Lifetimes are added inversely no matter if they are two-particle processes like Touschek scattering following a hyperbolic decay or one-particle processes like gas scattering following an exponential decay, because lifetime values are extrapolations of the loss rate to zero current, and the loss rates just add up [15].

C.1   Touschek lifetime

The non-relativistic Touschek lifetime formulae is given by [11] [12] [13]

$$
\frac{1}{T} = \frac{r_e^2 cq}{8\pi e \gamma^3 \sigma_s} \cdot \frac{1}{C} \oint_C \frac{F(\zeta(s))}{\sigma_x(s)\sigma_y(s)\sigma_{x'o}(s)(\delta_{\mathrm{acc}}(s))^2} \, ds \quad \text{with} \ \ \zeta(s) := \left(\frac{\delta_{\mathrm{acc}}(s)}{\gamma \sigma_{x'o}(s)} \right)^2 .
$$

r_e is the classical electron radius, q the bunch charge, σ_s the bunch length (assumed to be constant along the lattice), C the machine circumference, σ_x and σ_y the transverse rms beam envelopes and $\sigma_{x'o}$ the divergence for $x \approx 0$, given by

$$
\sigma_{x'o}(s) = \sqrt{\frac{\epsilon_x + \mathcal{H}(s)\sigma_\delta^2}{\beta_x}},
$$

with σ_δ the rms energy spread and $\mathcal{H}$ the lattice invariant.

$\delta_{\mathrm{acc}}(s)$ is the local lattice momentum acceptance, which is the minor of the lattice and the RF acceptance. In the linear case, the lattice momentum acceptance at $s = s_o$ is given by

$$
\delta_{\mathrm{acc}}^L(s_o) = \min \left(\frac{a_x(s)}{\sqrt{\mathcal{H}_o \beta_x(s)} + \eta(s)} \right) \tag{81}
$$

where a_x is the horizontal aperture. In the non-linear case it is obtained from tracking.

The "Touschek function" $F(\zeta)$ is defined as

$$
F(\zeta) = \int_o^1 \left(\frac{1}{u} + \frac{1}{2} \ln \frac{1}{u} - 1 \right) e^{-\zeta/u} \, du
$$

For small arguments $\zeta < \zeta_{\mathrm{small}} = 0.0013$ the asymptotic expression $F(\zeta) = \ln(E/\zeta) - 3/2$ is used, with $E = 0.5772$ Euler's number.

For large arguments, F becomes very small and is dominated by rounding errors, so for $\zeta > \zeta_{\mathrm{big}} = 22.8$ it is set to $F(\zeta_{\mathrm{big}}) = 10^{-16}$ (usually lattice locations where this happens are irrelevant anyway for the final lifetime result).

For $\zeta_{\mathrm{small}} < \zeta < 10$ a fair approximation is given by a 7th-order polynomial

$$
F(\zeta) \approx \exp \sum_{k=0}^7 A_k (\ln \zeta)^k, \quad \text{with} \qquad
\begin{aligned}
A = \{ \ & -3.10811, -2.19156, -0.615641, \\
& -0.160444, -0.0460054, -0.0105172, \\
& -1.31192 \cdot 10^{-3}, -6.3898 \cdot 10^{-5} \ \}
\end{aligned}
$$

The agreement within $\pm 2.5\%$ compared to the integration is acceptable considering that the integral itself deviates up to 10% from detailed Monte Carlo simulations of Touschek scattering [14].

Note, that the classical non-relativistic formula used here is appropriate for low emittance storage rings, where Touschek scattering has a dominant effect, because due to small emittance, the transverse momenta $\sigma_{px} = \beta\gamma m_o c \sigma_{x'o}$ are not relativistic (here β, γ are the relativistic parameters).

It is interesting to compare the numbers for the low emittance light source MAX IV at 3.0 GeV and for the compact light source AURORA-1 at 575 MeV:

In the MAX-IV storage ring σ_{px} varies around the lattice. High values are found in the center of the bending magnets, where $\mathcal{H} > 0$ and β_x is small. In MAX IV we have $\beta_x = 0.98$ m, $\mathcal{H} \approx 0.0015$ m, $\epsilon_x = 0.33$ nm, $\sigma_\delta = 7.7 \cdot 10^{-4}$ and $\gamma = 5870$. This results in a value of $\sigma_{px} = 0.21 m_o c$ and a transverse velocity of $v_x = 0.2c$.

In AURORA-1 σ_{px} is constant. With $\beta_x = 0.63$ m, $\mathcal{H} = 0.99$ m (!), $\epsilon_x = 1670$ nm, $\sigma_\delta = 6.3 \cdot 10^{-4}$ and $\gamma = 1125$ we get $\sigma_p = 2.04 m_o c$ and $v_x = 0.9c$.

Large emittance and large $\mathcal{H}$ have more effect than higher beam energy. So the non-relativistic approximation is applicable to MAX IV at 3 GeV but not to AURORA at 575 MeV.

C.2 Coulomb scattering

Coulomb or elastic scattering is deflection of electrons by residual gas nuclei. Electrons are lost if the deflection angle exceeds the machine acceptance. The differential cross section of Rutherford scattering of electrons (one unit charge) is given by

$$\frac{d\sigma}{d\Omega} = \left(\frac{Ze^2}{8\pi\epsilon_o m_o \gamma (\beta c)^2}\right)^2 \frac{1}{\sin^4(\theta/2)} \tag{82}$$

with Z the charge multiplicity of scattering nuclei, Ω solid angle and θ scattered angle. For relativistic electrons, and using the "classical electron radius"

$$r_e = \frac{e^2}{4\pi\epsilon_o m_o c^2} = 2.82 \cdot 10^{-15}\,\mathrm{m} \qquad \longrightarrow \qquad \frac{d\sigma}{d\Omega} = \frac{Z^2 r_e^2}{4\gamma^2} \frac{1}{\sin^4(\theta/2)} \tag{83}$$

The cross section σ for losses is obtained from integration over all angles θ which are not accepted by the machine aperture. We distinguish three cases, and to save writing we define $\bar{\sigma}$ without the leading constants.

1. The one-dimenional acceptance: we take the limiting polar angle $\hat{\theta}$ as given following Wiedemann [24] (Vol.I, Ch.11):

$$\bar{\sigma} = \int_{\mathrm{lost}} \frac{d\bar{\sigma}}{d\Omega} d\Omega = \int_0^{2\pi} \int_{\hat{\theta}}^\pi \frac{\sin\theta}{\sin^4(\theta/2)}\, d\theta\, d\phi = 2\pi \frac{2}{\tan^2(\hat{\theta}/2)} \tag{84}$$

26

Since the limiting angle in a storage ring is rather small, we expand the tangens and get

$$\bar{\sigma} \approx \frac{16\pi}{\hat{\theta}^2} \tag{85}$$

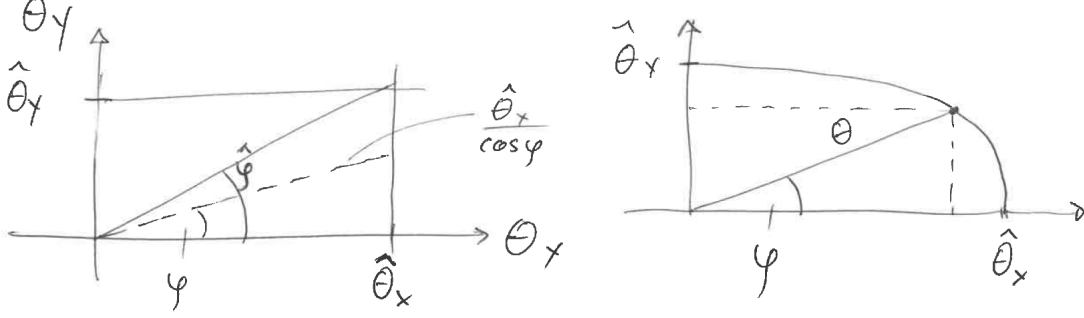The smaller of horizontal and vertical acceptance angles is used as a worst case estimate.

Figure 1: Rectangular and elliptic acceptances

2. The rectangular acceptance. We reproduce Wiedemann's calculation with angles $\hat{\theta}_x$, $\hat{\theta}_y$: Depending on the azimuth angle, losses occur horizontally and vertically, where the angle

$$\hat{\phi} = \arctan\left(\frac{\hat{\theta}_y}{\hat{\theta}_x}\right) \tag{86}$$

splits the two regimes, see fig. 1, left. Thus the integral is split in two and multiplied by four, since the situation is the same in all four quadrants:

$$\bar{\sigma} = 4 \left(\int_0^{\hat{\phi}} \int_{\frac{\hat{\theta}_x}{\cos\phi}}^{\pi} \frac{\sin\theta}{\sin^4(\theta/2)} \, d\theta \, d\phi + \int_{\hat{\phi}}^{\pi/2} \int_{\frac{\hat{\theta}_y}{\sin\phi}}^{\pi} \frac{\sin\theta}{\sin^4(\theta/2)} \, d\theta \, d\phi \right) \tag{87}$$

Calculating the integral over θ as before and expanding the tangens for small angles gives

$$\bar{\sigma} = 32 \left(\int_0^{\hat{\phi}} \frac{\cos^2\phi}{\hat{\theta}_x^2} \, d\phi + \int_{\hat{\phi}}^{\pi/2} \frac{\sin^2\phi}{\hat{\theta}_y^2} \, d\phi \right) \tag{88}$$

$$= 8 \left(\frac{\pi}{\hat{\theta}_y^2} + \sin(2\hat{\phi}) \left[\frac{1}{\hat{\theta}_x^2} + \frac{1}{\hat{\theta}_y^2} \right] + 2\hat{\phi} \left[\frac{1}{\hat{\theta}_x^2} - \frac{1}{\hat{\theta}_y^2} \right] \right) \tag{89}$$

Using Wiedemann's definition $R = \hat{\theta}_y / \hat{\theta}_x$ this can be written as

$$\bar{\sigma} = \frac{16\pi}{\hat{\theta}_y^2} \frac{F}{2\pi} \qquad \text{with} \qquad F = \pi + \sin(2\hat{\phi})\,(R^2 + 1) + 2\hat{\phi}(R^2 - 1) \tag{90}$$

3. The elliptic acceptance. From the basic ellipse geometry, see fig. 1, right,

$$\left(\frac{\theta_x}{\hat{\theta}_x}\right)^2 + \left(\frac{\theta_y}{\hat{\theta}_y}\right)^2 = 1 \qquad \theta_x = \theta \cos\phi, \quad \theta_y = \theta \sin\phi, \tag{91}$$

27

follows for the length of the radius vector

$$\theta(\phi) = \frac{1}{\sqrt{\frac{\cos^2\phi}{\hat{\theta}_x^2} + \frac{\sin^2\phi}{\hat{\theta}_y^2}}}. \tag{92}$$

As before the cross section is obtained from the four quadrants

$$\bar{\sigma} = 4\int_0^{\pi/2}\int_{\theta(\phi)}^{\pi} \frac{\sin\theta}{\sin^4(\theta/2)}\, d\theta\, d\phi = 4\int_0^{\pi/2} \frac{2}{\left(\tan\left[\frac{1}{2}\frac{1}{\sqrt{\cos^2\phi/\hat{\theta}_x^2+\sin^2\phi/\hat{\theta}_y^2}}\right]\right)^2}\, d\phi \tag{93}$$

After expanding the tangens again, the integral is easy to solve:

$$\bar{\sigma} = 32\int_0^{\pi/2}\left(\frac{\cos^2\phi}{\hat{\theta}_x^2} + \frac{\sin^2\phi}{\hat{\theta}_y^2}\right)d\phi = \frac{16\pi}{2}\left(\frac{1}{\hat{\theta}_x^2} + \frac{1}{\hat{\theta}_y^2}\right) \tag{94}$$

The three approximations give following values in terms of

$$f = 16\pi/\bar{\sigma}, \tag{95}$$

which will be proportional to the lifetime:

1-D formula and elliptic for $\hat{\theta}_x = \hat{\theta}_y := \hat{\theta}$ (round acceptance): $f = 1.00\,\hat{\theta}^2$

Rectangular for $\hat{\theta}_x = \hat{\theta}_y := \hat{\theta}$ (square acceptance): $f = 1.22\,\hat{\theta}^2$

Rectangular and elliptic for extreme cases $\hat{\theta}_x \gg \hat{\theta}_y := \hat{\theta}$ or $\hat{\theta}_y \gg \hat{\theta}_x := \hat{\theta}$ (slits): $f = 2.00\,\hat{\theta}^2$

The loss rate is given by product of cross section for losses and "luminosity", i.e. incoming particles (electrons) per time ΔT multiplied with target particles (nuclei) present in an area A. The interaction length is given by $\Delta T\beta c$, and, multiplied with the area gives the volume V:

$$\dot{N} = \sigma\frac{N}{\Delta T}\frac{N_{\text{nuclei}}}{A} = \sigma N_e(\beta c)\frac{N_{\text{nuclei}}}{V} \tag{96}$$

The last fraction is just the density of nuclei in space. The number of nuclei is the number of gas molecules N_{gas} multiplied with the number $\mathcal{N}$ of atoms per molecule. The molecule number is the product of Avogadro's number $N_A = 6.02\cdot 10^{23}\,\text{mol}^{-1}$ with the amount of gas measured in moles n. It is given by the ideal gas equation as function of volume V, temperature T and pressure P:

$$N_{\text{nuclei}} = \mathcal{N}N_{\text{gas}} \qquad N_{\text{gas}} = N_A n \qquad n = \frac{PV}{RT} \tag{97}$$

Here $R = 8.31$ J/K/mol is the universal gas constant. When inserting the volume is eliminated and we get the exponential decay with the Coulomb scattering beam lifetime τ:

$$\frac{\dot{N}}{N} = \sigma\beta\frac{cN_A}{R}\frac{\mathcal{N}P}{T} = \frac{1}{\tau} \qquad \longrightarrow \qquad N(t) = N_o e^{-t/\tau} \tag{98}$$

We now use the calculated cross section from eq.95 and insert the leading constants from eq.83:

$$\frac{1}{\tau} = \frac{4\pi c r_e^2 N_A}{R} \cdot \frac{1}{\gamma^2} \cdot \frac{\mathcal{N} Z^2 P}{T} \cdot \frac{1}{f} \tag{99}$$

The four fractions contain constants, beam properties (energy), gas properties and storage ring properties (acceptance). The constants may be combined in one number

$$K = 2168 \, \mathrm{m\,s\,K/kg} \tag{100}$$

in SI-units[2]. We confirm the formula from Wiedemann, who uses GGS units and gives beam lifetime ([24] Eq.11.12) in hours for the special case of nitrogen ($Z = 7$, $\mathcal{N} = 2$) at 0°C ($T = 273 \, \mathrm{K}$), with the pressure measured in nTorr (1 Torr = 133.3 Pa) , beam energy in GeV and limiting angle in mrad (i.e. acceptance in mm.mrad), for the 1-D case:

$$\tau[\mathrm{h}] = 10.25 \frac{(E\,[\mathrm{GeV}])^2}{P\,[\mathrm{nTorr}]} (\hat{\theta}\,[\mathrm{mrad}])^2 \tag{101}$$

The acceptance of the storage ring in linear approximation is obtained as silhouette of all local apertures scaled by the local beta functions, see sec.B. The dynamic acceptance is obtained from tracking including local apertures. Physical acceptance has a polygonal, convex shape which is something between ellipse and rectangle, dynamic acceptance can be anything, but should be reasonably smooth and convex if the nonlinear dynamics has been optimized well.

We prefer the elliptic approximation for the following reasons: it fits quite well, it is not too optimistic like the rectangular one, and it is simple (see eq.94), which allows a straightforward averaging to be implemented:

Eq.99 gives a loss rate for given acceptance angles contained in f. This angle is given by

$$\hat{\theta}_z(s) = \sqrt{\frac{A_z}{\beta_z(s)}} \qquad z = x, y \tag{102}$$

with A_z the invariant betatron amplitude of the acceptance (denoted ϵ_A by Wiedemann) and $\beta_z(s)$ the local beta function. Total loss rate is then obtained by averaging over all local loss rates. In case of the 1-dimensional or elliptic case this is very simple since

$$\frac{1}{\tau} \propto \frac{1}{\hat{\theta}_z^2} = \frac{\beta_z}{A_z}$$

Due to the linear dependence just averaging over the beta-functions, as usually done in the 1-dimensional case, is completely sufficient in the elliptic case too, whereas it is not sufficient in the rectangular case. OPA 4 only uses the elliptic case:

$$\frac{1}{\tau} = \frac{K}{\gamma^2} \cdot \frac{\mathcal{N} Z^2 P}{T} \cdot \frac{1}{2} \left(\frac{\langle \beta_x \rangle}{A_x} + \frac{\langle \beta_y \rangle}{A_y} \right) \tag{103}$$

---

[2]The implementation of the rectangular aperture in our paper [15] is only correct for $\rho \leq 1$, but else it is valid; however K was defined differently, using only half the value.

The acceptances are just given by the points of the physical or dynamic acceptance, where the polygon cuts the y or x axis and the beta function at track point, i.e. $A_x = [x(y = 0)]^2/\beta_x$, and vice versa for A_y. In case of an asymmetric shape we take the smaller value.

$\Longrightarrow$ *If dynamic acceptance is significantly smaller than the physical one, the lifetime calculation should be preceded by a binary search for maximum stable amplitudes in tracking.*

C.3 Bremsstrahlung

Bremsstrahlung or inelasic scattering leads to an energy loss of the particle and thus is a process similar to Touschek scattering. Different formulae are found in literature for cross section or probability of energy loss, which do not agree, although discrepancies cause only a few percent variation for the relevant range of parameters. We trust Wiedemann, who introduces the appropriate approximations (relativistic, thin gas, energy acceptance $\ll$ energy etc.) and gives a simple equation for the loss rate:

$$\frac{1}{\tau} = \frac{16\alpha c r_e^2 N_A}{3R} \cdot \frac{\mathcal{N}Z^2 P}{T} \left[\ln\left(\frac{183}{Z^{1/3}}\right) + \frac{1}{18}\right] \cdot \ln\left(\frac{1}{\delta_{\mathrm{ac}}}\right) \tag{104}$$

The three terms combine natural constants ($\alpha \approx 1/137$ the fine structure constant), gas properties and storage ring properties. The latter contains only the relative energy acceptance δ_{ac}. The loss rate does not depend on electron energy, except the fact that it is highly relativistic.

A bremsstrahlung event corresponds to a sudden loss of energy. The electron thus behaves like a Touschek scattered particle, but while Touschek scattering is a two-particle event, where one electron loses, the other one gains energy, energy is only lost in a bremsstrahlung event.

In Touschek calculations (sec. C.1) local positive and negative momentum acceptances, $\delta_{\mathrm{ac}}^{\pm}(s)$ are obtained from tracking. Thus it is straightforward to re-use these data for a more precise than usual bremsstrahlung lifetime calculation. However, unlike Touschek, the dependence on momentum acceptance is weak. We just use in the above equation the average

$$\langle\ln\left(\frac{1}{\delta_{\mathrm{ac}}^{-}}\right)\rangle \tag{105}$$

We use only the negative acceptance. (In 6-D Touschek tracking it's value would imply limitations on the positive side due to synchrotron oscillations.)

D RF bucket plot

In a storage ring, the longitudinal equations of motion are approximately decoupled,

$$\dot{\delta} = f(\phi) = -\frac{\partial H}{\partial \phi} \qquad \frac{\partial f}{\partial \delta} = 0 \tag{106}$$

$$\dot{\phi} = g(\delta) = \frac{\partial H}{\partial \delta} \qquad \frac{\partial g}{\partial \phi} = 0 \tag{107}$$

with $\delta = \Delta p / p \approx \Delta E / E$ and $\phi = 2\pi c \Delta t / \lambda$, so the Hamiltonian, from which they originated, is immediately obtained from integration:

$$H = -\int f \, d\phi + \int g \, d\delta \tag{108}$$

Since synchrotron motion is slow, differentiation is approximated by

$$\dot{\phi} = \frac{\Delta_t \phi}{T_o} \qquad \dot{\delta} = \frac{\Delta_t \delta}{T_o}, \tag{109}$$

with $T_o = C/c$ the recirculation time and C the ring circumference. Note that Δ_t is the change in one recirculation time, whereas Δ indicates the deviation from a reference particle propagating on the closed orbit.

Path length or time of flight is given by the momentum compaction, including higher orders:

$$\Delta_t s / C = \alpha_1 \delta + \alpha_2 \delta^2 + \ldots \tag{110}$$

$$\implies \quad g(\delta) = \frac{\Delta_t \phi}{T_o} = \frac{2\pi c}{\lambda} \sum_{k=1}^{n} \alpha_k \delta^k \tag{111}$$

Energy change in the RF cavity is given by

$$\Delta_t E = eV_1 \sin \phi \tag{112}$$

with V_1 the RF voltage. Radiation loss per turn U_o gives the synchronous phase,

$$U_o = eV \sin \phi_s, \tag{113}$$

which is the center of motion for phase oscillations. Thus

$$\Delta_t \Delta E = eV \left(\sin(\phi + \phi_s) - \sin \phi_s \right) \tag{114}$$

If there are harmonic cavities, we assume for simplicity, that they are at zero phase and all acceleration is done by the fundamental only:

$$\Delta_t \Delta E = -\sum_{k=2}^{n} eV_k \sin(k\phi) \tag{115}$$

$$\implies \quad f(\phi) = \frac{\Delta_t \delta}{T_o} \approx \frac{e}{E_o T_o} \left[V_1 \left(\sin(\phi + \phi_s) - \sin \phi_s \right) + s_h \sum_{k=2}^{n} V_k \sin(k\phi) \right] \tag{116}$$

Here we have to distinguish two cases:

$\alpha_1 < 0 \longrightarrow 0 < \phi_s < \pi/2, \qquad s_h = -1$

$\alpha_1 > 0 \longrightarrow \pi/2 < \phi_s < \pi, \qquad s_h = +1$

Integration gives the Hamiltonian:

$$H = \frac{2\pi c}{\lambda} \sum_{k=1}^{n} \alpha_k \frac{\delta^{k+1}}{k+1} + \frac{e}{T_o E_o} \left[V_1 \left(\cos(\phi_s + \phi) + \phi \sin \phi_s \right) + s_h \sum_{k=2}^{n} \frac{V_k}{k} \cos(k\phi) \right]. \qquad (117)$$

Since the equations of motion are decoupled, fix points are simply given by the roots

$$f(\phi) \overset{!}{=} 0 \qquad \textbf{and} \qquad g(\delta) \overset{!}{=} 0. \qquad (118)$$

A fixpoint is a combination of two roots, i.e. $F_{ij} = (\phi_{oi}, \delta_{oj})$.

The Jacobian tells if it is elliptic or hyperbolic: The mapping

$$\phi \leftarrow \phi + g(\delta) \qquad \delta \leftarrow \delta + f(\phi) \qquad (119)$$

has the Jacobian

$$J = \begin{pmatrix} 1 & \partial g/\partial \delta \\ \partial f/\partial \phi & 1 \end{pmatrix} \qquad (120)$$

and eigenvalues

$$|J - \mu| = 0 \quad \longrightarrow \quad \mu = 1 \pm \sqrt{\frac{\partial f}{\partial \phi} \cdot \frac{\partial g}{\partial \delta}}, \qquad (121)$$

which are elliptic/hyperbolic if the product of the derivatives is negative/positive. The hyperbolic fixpoints are the origins of the separatrices between stable and unstable areas.

Practically, the longitudinal phase space is plotted by application of a contour plot program [25] to the Hamiltonian. Fixpoints are obtained from a root finder, the characteristics from the derivatives at the roots. Finally separatrices are plotted as contours starting at hyperbolic fixpoints. [opabucket> CalcBucket]

E The Hamiltonian for transverse beam dynamics

E.1 The curvilinear system

A beam consists of particles travelling closely along a curve in space, therefore it is convenient to introduce a local curvilinear coordinate system traveling along the design orbit of the accelerator, which is a curve in space:

$$\vec{r}(s) = \vec{r}_o(s) + \hat{\vec{x}}(s) \cdot x(s) + \hat{\vec{y}}(s) \cdot y(s) \qquad (122)$$

The vectors $\{\hat{\vec{x}}; \hat{\vec{s}}; \hat{\vec{y}}\}$ form a right hand side system of unit vectors [7], with $\hat{\vec{s}}$ tangential to the curve, pointing in forward direction, $\hat{\vec{x}}$ radial to the curvature of the curve, pointing away from the center of curvature, and $\hat{\vec{y}} = \hat{\vec{s}} \times \hat{\vec{x}}$. These unit vectors are defined a priori, i.e. they define the curve of our choice, whereas $x(s)$ and $y(s)$ will be solutions of the equations of motion in

the curvilinear system. The independent coordinate s is the length of the curve, measured from some point. We neglect here a possible torsion of the curve.

Derivatives of the vectors with respect to s [6, 3]:

$$\frac{d\vec{r}_o(s)}{ds} = \hat{\vec{s}} \text{ because } \vec{r}_o(s+ds) = \vec{r}_o + \hat{\vec{s}} \cdot ds \tag{123}$$

Following the curvature by an angle $d\phi = h\,ds$ with $h = 1/\rho$ the local curvature (inverse radius of curvature) applies a rotation around the y-axis:

$$\vec{a} + d\vec{a} = \begin{pmatrix} \cos d\phi & -\sin d\phi & 0 \\ \sin d\phi & \cos d\phi & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \vec{a} \xrightarrow{d\phi \ll 1} d\vec{a} = \begin{pmatrix} 0 & -h\,ds & 0 \\ h\,ds & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{124}$$

The unit vectors are given by

$$\hat{\vec{x}} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \qquad \hat{\vec{s}} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \qquad \hat{\vec{y}} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

So the derivatives follow as

$$\frac{d\hat{\vec{x}}}{ds} = h\hat{\vec{s}} \qquad \frac{d\hat{\vec{s}}}{ds} = -h\hat{\vec{x}} \qquad \frac{d\hat{\vec{y}}}{ds} = 0 \tag{125}$$

E.2 The relativistic Hamiltonian

In the most general form, the Hamiltonian with electromagnetic fields is given by

$$H_o = c\sqrt{(\underline{\vec{p}} - q\underline{\vec{A}})^2 + m_o^2 c^2} + qV \tag{126}$$

with m, q particle mass and charge, V the electrostatic potential and $\underline{\vec{A}}$ the magnetic vector potential. The underline indicates that the vector is expressed in a global, cartesian coordinate system. $\underline{\vec{p}}$ is the *canonical momentum*, not the kinetic momentum, which would be given by

$$m\underline{\vec{v}} = \underline{\vec{p}} - q\underline{\vec{A}}. \tag{127}$$

The Hamilton-Jacobi equations of motion,

$$\dot{\underline{x}}_k = \frac{\partial H_o}{\partial \underline{p}_k} \qquad \dot{\underline{p}}_k = -\frac{\partial H_o}{\partial \underline{x}_k}, \tag{128}$$

apply only to the canonic coordinates $\underline{\vec{x}}$ and $\underline{\vec{p}}$, but not to the kinetic momentum.

E.3 Canonical transformations in general

This is well explained in [6]: Canonical transformations perform a coordinate transformation of the Hamiltonian from an old set of canonical variables $\{p; q\}$ to a new set of variables $\{P; Q\}$ which are canonical too. So, a canonical transformation is more than just a coordinate transformation.

The transformation is performed through a *generating function F*. Its explicit time dependency is added to the Hamiltonian:

$$H_{\text{new}}(P, Q, t) = H_{\text{old}}(p, q, t) + \frac{\partial F}{\partial t} \tag{129}$$

Four types of generating functions are distinguished, depending on which set of old and new coordinates they contain:

$$F_1(q, Q, t), \qquad F_2(q, P, t), \qquad F_3(p, Q, t), \qquad F_4(p, P, t)$$

One uses the one which is most convenient for the problem given.

As an example, let's use F_2. We want the equation of motion in the new coordinates:

$$\dot{Q}_k = \frac{\partial H_{\text{new}}(P, Q, t)}{\partial P_k} = \frac{\partial}{\partial P_k} \left(\partial H_{\text{old}}(p, q, t) + \frac{\partial F_2(q, P, t)}{\partial t} \right) = \frac{\partial}{\partial t} \frac{\partial F_2(q, P, t)}{\partial P_k}$$

$$\rightarrow Q_k = \frac{\partial F_2(q, P, t)}{\partial P_k}$$

$$\dot{p}_k = -\frac{\partial H_{\text{old}}(p, q, t)}{\partial q_k} = -\frac{\partial}{\partial q_k} \left(\partial H_{\text{new}}(P, Q, t) - \frac{\partial F_2(q, P, t)}{\partial t} \right) = \frac{\partial}{\partial t} \frac{\partial F_2(q, P, t)}{\partial q_k}$$

$$\rightarrow p_k = \frac{\partial F_2(q, P, t)}{\partial q_k}$$

For other cases, see [6] or calculate analogously.
The signs for the four cases are as $+-$, $++$, $--$, $-+$.

E.4 The contact transformation

Now we want to apply a canonical transformation to transform the general Hamiltonian from eq.126 into the curvilinear system. We have the *new* coordinates $q = \{x; s; y\}$ and the *old* momenta $P = \{p_x; p_y; p_z\}$, so we need a generating function of third type [6]. A possible solution is given by

$$F_3(\vec{p}, \vec{r}, t) = -\vec{p} \bullet \vec{r}(s) \tag{130}$$

with $\vec{r}(s)$ from eq.122 ($\bullet$ = scalar product). The old coordinates (we are not interested in) are obtained from

$$\underline{x} = -\frac{\partial F_3}{\partial \underline{p_x}} = \vec{r}_x \quad \text{etc,}$$

i.e. $\underline{\vec{r}}(s) = \vec{r}(s)$, which is trivial and was just the definition of our curve in space. What we need are the new momenta in the curvilinear system:

$$p_s = -\frac{\partial F_3}{\partial s} = \underline{\vec{p}} \bullet \left(\frac{\partial \vec{r}_o(s)}{\partial s} + \frac{\partial \hat{\vec{x}}(s)}{\partial s} \cdot x + \frac{\partial \hat{\vec{y}}(s)}{\partial s} \cdot y \right) \tag{131}$$

From eqs.123 and 125 follows:
$$p_s = \underline{\vec{p}} \bullet \hat{\vec{s}} \cdot (1 + hx) \tag{132}$$
For the other components we get directly with eq.122
$$p_x = \underline{\vec{p}} \bullet \hat{\vec{x}} \qquad p_y = \underline{\vec{p}} \bullet \hat{\vec{y}} \tag{133}$$
Since F_3 does not contain an explicit time dependence, the new Hamiltonian is the old:
$$H_1 = H_o \tag{134}$$

It is important to note, that p_s is *not* the tangential component of the momentum (which would be just given by $\vec{p} \bullet \hat{\vec{s}}$), but the new $\vec{p}$ is the *canonical* momentum in the curvilinear system.

The next step is handled differently by the different authors: In refs.[7, 3, 4] the vector potential is considered a *canonical vector potential* and transformed in the same way as the momentum, but refs.[17, 6] transform only the canonical momentum, since transforming both would imply a transformation of the kinetic momentum (see eq.127), which is not canonic. We follow the second line, since it will lead to the correct dipole focusing term in the simplified Hamiltonian using an obvious definition of the vector potential. So, we insert the canonical transformation in the components of the momentum, i.e. its projection along the new coordinate axis:

$$\underline{\vec{p}} \bullet \hat{\vec{s}} = \frac{p_s}{1 + hx} \qquad \underline{\vec{p}} \bullet \hat{\vec{x}} = p_x \qquad \underline{\vec{p}} \bullet \hat{\vec{y}} = p_y$$

and insert this into the Hamiltonian:

$$H_1 = c\sqrt{\left(\frac{p_s}{1 + hx} - qA_s \right)^2 + (p_x - qA_x)^2 + (p_y - qA_y)^2 + m_o^2 c^2} + qV \tag{135}$$

E.5   Change of variable $t \to s$

For beam transformation we would like to use s, the length along the curve describing the accelerator as the independent variable rather than the time t [4]. The equations of motion are given by (y not shown):

$$\dot{x} = \frac{\partial x}{\partial t} = \frac{\partial H_1}{\partial p_x}, \quad \dot{p}_x = -\frac{\partial H_1}{\partial x} \qquad \dot{s} = \frac{\partial H_1}{\partial p_s}, \quad \dot{p}_s = -\frac{\partial H_1}{\partial s}$$

Just inverting the last two equations we get

$$t' = \frac{\partial t}{\partial s} = \frac{1}{\dot{s}} = \frac{\partial p_s}{\partial H_1} \qquad H_1' = \frac{\partial H_1}{\partial s} = -\frac{\partial p_s}{\partial t}$$

and for x (and for y the same way)

$$x' = \frac{\partial x}{\partial s} = \frac{\partial x/\partial t}{\partial s/\partial t} = \frac{\partial H_1/\partial p_x}{\partial H_1/\partial p_s}$$

H_1 must not change if we vary the canonical momenta, i.e.

$$0 = dH_1 = \left.\frac{\partial H_1}{\partial p_x}\right|_{p_s} dp_x + \left.\frac{\partial H_1}{\partial p_s}\right|_{p_x} dp_s \rightarrow \left.\frac{\partial p_s}{\partial p_x}\right|_{H_1} = -\frac{\partial H_1/\partial p_x}{\partial H_1/\partial p_s}$$

$$\rightarrow x' = -\frac{\partial p_s}{\partial p_x} \qquad p_x' = \frac{\partial p_s}{\partial x} \tag{136}$$

Now $-p_s$ plays the role of a new Hamiltonian for a new set of coordinates, given by $\{x, p_x, y, p_y, H_1, t\}$, so we define $H_2 = -p_s$, with the equations of motion

$$x' = \frac{\partial H_2}{\partial p_x}, \quad p_x' = -\frac{\partial H_2}{\partial x}, \qquad H_1' = \frac{\partial H_2}{\partial t}, \quad t' = -\frac{\partial H_2}{\partial H_1}, \tag{137}$$

The new Hamiltonian is obtained by solving eq.135 for $-p_s$:

$$H_2 = -p_s = -(1 + hx) \cdot \left(\sqrt{\left(\frac{H_1 - qV}{c}\right)^2 - m_o^2 c^2 - (p_x - qA_x)^2 - (p_y - qA_y)^2} + qA_s \right) \tag{138}$$

E.6   Approximations

Several approximations will be introduced, which are justified at least for large high energy machines [4], and make it much simpler for us to proceed:

- High relativistic: $E = pc$

- Static: $\partial \vec{A}/\partial t = 0$

- No electric fields, i.e. no energy gain: $V = 0$. The H_1 from eq.135 is just the total energy of the particle: $H_1 = E$.

- No fringe fields, i.e. piece-wise constant, hard-edge fields: $B_s = 0$. In cartesian and cylindrical coordinates, which are the two cases of a tangential coordinate system to a torsion-free curve in space we consider, the curl operator gives

$$B_s = \frac{\partial A_x}{\partial y} - \frac{\partial A_y}{\partial x}$$

This allows $A_x = A_y = 0 \rightarrow B_x = \frac{\partial A_s}{\partial y} \qquad B_y = \frac{\partial A_s}{\partial x}$

- Paraxial approximation: particles stay close to the curve, and their transverse momenta are small compared to the total momentum, i.e. $p_{x,y} \ll p$.

The first four items simplify the Hamiltonian from eq.138 as

$$H_2 = -(1 + hx) \left(\underbrace{\sqrt{E^2/c^2 - m_o^2 c^2}}_{p^2} - p_x^2 - p_y^2 + qA_s \right) \tag{139}$$

The fifth item allows to expand the square root, since $p_{x,y}/p \ll 1$:

$$H_2 = -(1 + hx) \left(p \cdot \left(1 - \frac{p_x^2 + p_y^2}{2p^2} \right) + qA_s \right) \tag{140}$$

Finally we refer to a reference momentum p_o the machine is designed for, and consider particles with relative momentum deviations

$$\delta = \frac{p - p_o}{p_o} \tag{141}$$

Introducing p_o and δ in the Hamiltonian gives

$$\tilde{H}_2 = -(1 + hx) \left(1 + \delta - \frac{\tilde{p}_x^2 + \tilde{p}_y^2}{2(1 + \delta)} + \frac{q}{p_o} A_s \right) \tag{142}$$

Here we introduced dimensionless momenta normalized to the reference momentum $\tilde{p}_{x,y} = p_{x,y}/p_o$ and also normalized the Hamiltonian, such that $\tilde{H}_2 = H_2/p_o$ is dimensionless.

E.7 Modeling the vector potential

The curve in space could be any curve, but we restrict it to a curve which is composed from pieces which are either straight (cartesian geometry) or curved with constant curvature (cylindrical geometry) - like tracks of a toy train. In both geometries the magnetic field, given by $\vec{B} = \nabla \times \vec{A}$, simplifies due to $A_x = A_y = 0$:

In **cartesian** geometry this gives

$$B_x = -\frac{\partial A_s}{\partial y} \qquad B_y = +\frac{\partial A_s}{\partial x} \tag{143}$$

In **cylindrical** geometry we have to use the actual radius, given by $\rho = \rho_o + x = \frac{1+hx}{h}$, with h the local *design* curvature of our curve. Using the curl in cylindrical coordinates from any mathematics book, we have to identify $\hat{\vec{\rho}} = \hat{\vec{x}}$, $\hat{\vec{\phi}} = \hat{\vec{s}}$, $\hat{\vec{z}} = \hat{\vec{y}}$. This yields

$$B_x = -\frac{\partial A_s}{\partial y} \qquad B_y = \frac{h}{1 + hx} A_s + \frac{\partial A_s}{\partial x} \tag{144}$$

37

Since there are no currents, we have $\nabla \times \vec{B} = 0$.

In **cartesian** geometry a possible expression for $\vec{A}$ is given by a *multipole expansion*:

$$\frac{q}{p_o} As = -\Re \sum_{n=1}^{\infty} \frac{ia_n + b_n}{n}(x + iy)^n \tag{145}$$

We prove that $\nabla \times \vec{B} = 0$:

$$\nabla \times \vec{B} = \nabla \times \nabla \times \vec{A} = \frac{\partial^2 A_s}{\partial x^2} + \frac{\partial^2 A_s}{\partial y^2}$$

$$\frac{\partial^2 A_s}{\partial x^2} = -\frac{p_o}{q}\Re\sum(n-1)(ia_n + b_n)(x+iy)^{n-2} = -\frac{\partial^2 A_s}{\partial y^2} \rightarrow \text{ok.}$$

Here b_n refer to regular multipoles ($n = 1$ dipole, $n = 2$ quadrupole etc.) and a_n to the corresponding *skew* multipoles, which we will not consider further for now. For the field components we obtain immediately

$$\begin{aligned} B_y &= -\frac{p_o}{q}(b_1 &+ b_2 x + b_3(x^2 - y^2)\ldots) \\ B_x &= -\frac{p_o}{q}(&+ b_2 y + b_3 xy \ldots) \end{aligned} \tag{146}$$

The **cylindrical** geometry is much more complicated. Calculating $\nabla \times \vec{B}$ in cylindrical coordinates is simplified by the assumption of a piece-wise constant field, i.e. $B_s = 0$ and $\partial B_{(x,y)}/\partial s = 0$:

$$\nabla \times \vec{B} = \left(\frac{\partial B_x}{\partial y} - \frac{\partial B_y}{\partial x}\right) \cdot \hat{\vec{s}} \overset{!}{=} 0$$

Inserting eq.144 we obtain

$$\nabla \times \vec{B} = -\hat{\vec{s}} \cdot \left(\frac{\partial^2 A_s}{\partial x^2} + \frac{\partial^2 A_s}{\partial y^2} + \frac{h}{1 + hx}\frac{\partial A_s}{\partial x} - \left(\frac{h}{1 + hx}\right)^2 A_s\right) \overset{!}{=} 0 \tag{147}$$

A solution of this differential equation for A_s is given by [17]

$$\frac{q}{p_o}A_s = -\frac{b_1}{2}\frac{1 + hx}{h} \tag{148}$$

Calculating the magnet field from eq.144 gives

$$B_y = -\frac{p_o}{q}b_1 \tag{149}$$

which is just dipole field as it exists in a sector magnet.

Considering combined function magnets (i.e. bending magnets with gradients as commonly used in accelerators), a similar multipole representation of the vector potential can be calculated, but is rather complicated (see p.362 in [5]).

Assuming a separate function lattice, we restrict ourselves to pure dipole fields so far. The simplified Hamiltonian we will find thus is not valid for combined function magnets. However in large rings, with strong gradients and low curvatures, the common but inconsistent procedure to treat the dipole component in cylindrical geometry and the higher multipoles in cartesian geometry, is good enough.

E.8 The simplified Hamiltonian

Assuming a separate function machine with pure sector magnets (cylindrical geometry) and cartesian quadrupoles and higher multipoles, we express the vector potential A_s by the sum of eqs.148 and 145 and introduce it into eq.142. Further, we consider only regular multipoles and show only the first terms of the multipole sum:

$$\tilde{H}_2 = -(1 + hx)\left(1 + \delta - \frac{\tilde{p}_x^2 + \tilde{p}_y^2}{2(1 + \delta)}\right) + \frac{b_1}{2}\frac{(1 + hx)^2}{h} + \frac{b_2}{2}(x^2 - y^2) + \dots \tag{150}$$

Here the multiplication of all terms with $(1 + hx)$ was not executed for the multipole sum since we assumed all these multipoles as cartesian, i.e. $h = 0$.

Now the dipole field is *adjusted* to provide the curvature h, i.e. we set $h = b_1$, but note that these two quantities are not necessarily identical. Subtracting constant terms, which are irrelevant for the equations of motion from the Hamiltonian, we thus arrive at

$$\mathcal{H} = \tilde{H}_2 + \frac{1}{2} + \delta = (1 + hx)\frac{\tilde{p}_x^2 + \tilde{p}_y^2}{2(1 + \delta)} - b_1 x\delta + \frac{b_1^2}{2}x^2 + \frac{b_2}{2}(x^2 - y^2) + \dots \tag{151}$$

In a last step, the curvature at the kinematic term is dropped, which is justified for large rings. Also including again the sextupole term, we finally arrive at the simplified Hamiltonian useful for transverse dynamics in large rings:

$$\mathcal{H} = \frac{\tilde{p}_x^2 + \tilde{p}_y^2}{2(1 + \delta)} - b_1 x\delta + \frac{b_1^2}{2}x^2 + \frac{b_2}{2}(x^2 - y^2) + \frac{b_3}{3}(x^3 - xy^2) + \dots \tag{152}$$

E.9 Equations of motion

Since we changed the independent variable from t to s, the equations of motions are

$$x' = \frac{\partial \mathcal{H}}{\partial \tilde{p}_x} = \frac{\tilde{p}_x}{1 + \delta} = \frac{p_x}{p} \tag{153}$$

which is just the divergence angle (for $x' \ll 1$) (same for y'), and

$$\tilde{p}_x' = -\frac{\partial \mathcal{H}}{\partial x} = b_1\delta - (b_1^2 + b_2)x - b_3(x^2 - y^2) \tag{154}$$

39

$$\tilde{p}'_y = -\frac{\partial \mathcal{H}}{\partial y} = b_2 y + 2b_3 xy \tag{155}$$

Differentiating again and inserting gives the differential equation

$$x'' = \frac{\tilde{p}'_x}{1+\delta} = \underbrace{\frac{b_1}{1+\delta}}_{h} \cdot \delta - \underbrace{\frac{b_1^2 + b_2}{1+\delta}}_{K} \cdot x - \ldots \tag{156}$$

Note the different scaling of b_1 with momentum for the δ and x-terms! The solution is

$$
\begin{aligned}
x(s) &= C(s)\, x_o + \frac{S}{p}\, x'_o + (1 - C(s))\frac{h}{K}\, \delta \qquad \text{with} \quad p = \sqrt{|K|} \\
x'(s) &= -\frac{K}{p} S(s)\, x_o + C(s)\, x'_o + S(s)\frac{h}{p}\, \delta
\end{aligned}
\tag{157}
$$

with $p = \sqrt{|K|}$, and $C(s) = \cos ps$ or $\cosh ps$, $S(s) = \sin ps$ or $\sinh ps$, for positive or negative K.

References

[1] OPA-4 code documentation, January 2026,
`https://github.com/opa-code/opa4-documentation/blob/main/opadocu.pdf`.

[2] V. Ziemann and A. Streun, Equilibrium parameters in coupled storage ring lattices and practical applications, Phys. Rev. Accel. Beams, 25, 050703, 2022.
`https://link.aps.org/doi/10.1103/PhysRevAccelBeams.25.050703`.

[3] E.D.Courant & H.S.Snyder, Theory of the alternating gradient synchrotron, Ann. Phys. 3,1(1958).
`http://ab-abp-rlc.web.cern.ch/ab-abp-rlc/AP-literature/Courant-Snyder-1958.pdf`

[4] D.A.Edwards & M.J.Syphers, An Introduction to the physics of high energy accelerators, Wiley-VCH, 1992.

[5] E.Forrest, Beam Dynamics, Harwood Academic Publishers, Amsterdam 1998.

[6] J.R.Rees, Symplecticity in beam dynamics, SLAC-PUB 9939, 2003.
`http://www.slac.stanford.edu/cgi-wrap/getdoc/slac-pub-9939.pdf`

[7] R.D.Ruth, Single particle dynamics in circular accelerators, SLAC-PUB 4103, 1986.
`http://www.slac.stanford.edu/cgi-wrap/getdoc/slac-pub-4103.pdf`

[8] D. Edward and L.Teng, *Parametrization of linear coupled motion in periodic systems,* IEEE Trans.Nucl.Sci. 20, 885 (1973).

[9] D. Sagan, D. Rubin, *Linear Analysis of coupled lattices,* Physical Review Special Topics–Accelerators and Beams 2 (1999) 074001.

[10] K. Wille, Physik der Teilchenbeschleuniger und Synchrotronstrahlungsquellen, Teubner Studienbücher, Stuttgart 1992.

[11] H. Bruck, Circular Particle Accelerators, Los Alamos 1966, LA-TR-72-10.

[12] M. Zisman et al., ZAP user's manual, Berkeley 1986, LBL-21270.

[13] A. Streun, Momentum acceptance and Touschek lifetime, SLS-Note 18/97.

[14] S. Khan, Simulation of the Touschek effect for Bessy II – a Monte Carlo approach, BESSY-TB 177/93.

[15] Å. Andersson and A. Streun, Lifetime and acceptance of the SLS storage ring, EPAC-2006, Edinburgh, THPLS060, p.3421.

[16] J. Bengtsson, Non-linear transverse dynamics, CERN-88-05.

[17] J. Bengtsson, The sextupole scheme for the SLS, SLS-Note 9/97, `http://slsbd.psi.ch/pub/slsnotes/sls0997.pdf`

[18] Chun-xi Wang, Explicit formulas for 2nd-order driving terms due to sextupoles and chromatic effects of quadrupoles, ANL/APS/LS-330, 2012

[19] S. C. Leemann and A. Streun, Perspectives for future light source lattices incorporating yet uncommon magnets, Phys. ST Rev. Accel. Beams, 14, 030701, 2011.

[20] W. H. Press et al., Numerical Recipes in Pascal, Cambridge 1989.

[21] K. L. Brown, A first- and second-order matrix theory, Stanford 1967, SLAC-75.

[22] A.W.Chao, M.Tigner, Handbook of accelerator physics and engineering, Singapore 1998.

[23] B. Riemann, M. Aiba, J. Kallestrup, A. Streun, Efficient algorithms for dynamic aperture and momentum acceptance calculation in synchrotron light sources, Phys. Rev. Accel. Beams, 27, 094002,2024, `https://doi.org/10.1103/PhysRevAccelBeams.27.094002`.

[24] Wiedemann

[25] `http://paulbourke.net/papers/conrec/`

[26] T. Atkinson et al., Development of a non-liner kicker system..., IPAC-2011, San Sebastian, THPO024, p.3394

[27] `http://rosettacode.org/wiki/Arithmetic_Evaluator/Pascal`