

Correction key The quasi-local scheme (Neuffer).

SSC-N-366
E. FOREST
7/87

1. Differential equation for a map

Consider the following differential equation:

$$\frac{dM}{ds} = M H(s) \quad (1.1)$$

Here, M and H are part of some non-commuting algebra.

We will assume that H is made of two parts:

$$H = \epsilon H_0 + \eta V(s) \quad (1.2)$$

Here H_0 is independent of s and V depends implicitly on s .
 ϵ and η are "smallness" parameters.

Obviously we can solve for the case $\eta=0$ (at least in a formal sense). Calling N the solution of (1.1) with $\eta=0$:

$$\frac{dN}{ds} = N \epsilon H_0 ; N(0) = I$$

$$\Rightarrow N(s) = \exp(s \epsilon H_0) \quad (1.3)$$

Now, we rewrite M as a product ordered in η :

$$M(s) = P(s) N(s) \quad (1.4a)$$

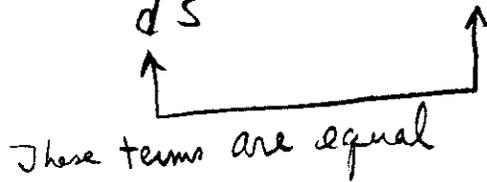
$$P(s) = \exp(: \eta f_1 :) \exp(: \eta^2 f_2 :) \dots \quad (1.4b)$$

6. Conclusions

7. References

We get a differential equation for ρ

$$\frac{dM}{ds} = \frac{d\rho}{ds} n + \rho \frac{dn}{ds} = \rho n \epsilon H_0 + \rho n \eta V$$



 These terms are equal

⇓

$$\frac{d\rho}{ds} = \eta \rho n V n^{-1} \quad (1.5)$$

Using the commutator operator \hat{H}_0 defined as

$$\hat{H}_0 g = [H_0, g] \leftarrow \text{commutator,}$$

we rewrite (1.5) as

$$\frac{d\rho}{ds} = \eta \exp(s\epsilon \hat{H}_0) V \quad (1.6a)$$

$$\text{since, } \exp(s\epsilon \hat{H}_0) V \exp(-s\epsilon \hat{H}_0) = \exp(s\epsilon \hat{H}_0) V \quad (1.6b)$$

Using (1.4b) and (1.6b), we equate powers of

η :

$$f_1 = \int_0^L \exp(s\epsilon \hat{H}_0) V(s) ds \quad (1.7)$$

PARA

line 2 data DFIX DTUNE DCHRO

comment lines: DDEL = Momentum step of $\Delta p/p_0$ for chromaticity determination
DSBE = Quadrupole step size for tune fitting in units of
 $K = B' / B\rho$ "thick" quadrupole
 $K = L B' / B\rho$ "thin" quadrupole
DSEX = Sextupole stepsize for chromaticity fitting in units of
 $S = L B'' / 2B\rho$

PARA

line 3 data DDEL DSBE DSEX

comment lines: Amplitudes in Floquet - Variables :
 $AMPLITUDE_{k,j} = SCL_k * (AMIN + j * (AMAX - AMIN) / NITRA)$
with $k = 1, 2 ; j = 1, \dots, NITRA$
SCL1 = Scale factor , horizontal plane
SCL2 = Scale factor , vertical plane
AMIN = " Minimum " Amplitude
AMAX = " Maximum " Amplitude
NITRA = Number of particles tracked
NUMLF = Number of turns tracked
NIR = Number of amplitude reduction steps in tracking
MARY = Order of map ; MARY = 0 : no map

PARA

line 4 data SCL1 SCL2 AMIN AMAX NITRA NUMLF NIR MARY

comment lines: NH $\neq 0$: Closed orbit correction
XMON = x-rms value for monitors
YMON = y-rms value for monitors
NREAD = Read from file 'HIRO' (FOR013.DAT) values of :
quads, sextupoles, ichmax horizontal correctors and
icvmax vertical correctors.
NWRITE = Write to file 'HIRON' (FOR012.DAT) values for :
quads, sextupoles, ichmax horizontal correctors and
icvmax vertical correctors.

PARA

line 5 data NH XMON YMON NREAD NWRITE

The difference between Lumped (Achromat) and (quasi-)local correction is found in the following Requirements on f_i .

a) Lumped (Achromat).

Find $V(s)$ and ε such that

$$f_i = 0 \quad \text{for a given } L.$$

$V(s)$ need not to vanish.

b) Local

Find $V(s)$ such that

$$f_i = 0 \quad \forall \varepsilon \quad \text{at a given } L$$

$$\implies V(s) = 0 \implies f_j = 0 \quad \forall j.$$

Obviously, if we correct f_i for all ε , we are just asking for a local correction system.

c) Quasi-Local, n^{th} order.

Find $V(s)$ such that

$$0 \leq k \leq n \quad \frac{d^k f_i}{d\varepsilon^k} = 0 \quad \text{at a given } L.$$

do-able with k knobs in $V(s)$!

followed by up to 10 lines containing the systematic and random multipole coefficients :

$$\begin{array}{cccccc} -b_n & b_n\text{-rms} & a_n & a_n\text{-rms} & & (n = 1, \dots, 10) \\ \text{(format 5X, F15.7, F15.7, F15.7, F15.7)}, & & & & & \end{array}$$

where in accordance with eq. (7) the sign of b_n has been inverted. The multipole coefficient table has to be closed by a line

$$\begin{array}{c} \text{NEXT} \\ \text{(format A4)}. \end{array}$$

In the single element list and in the structure input list the multipole blocks are represented by a name ('NAME') consisting of 4 characters (blank is excluded here), the corresponding multipole coefficients, however, are distinguished only by the first three characters ('TNM') of 'NAME'. This allows to use one set of multipole coefficients for magnets of identical design but of different integrated strength as is explained next.

The connection between these numbers and the magnet data on one hand and the resulting kicks on the other hand is best explained by the following expression for the field expansion :

$$B_y + iB_x = \text{SCF} \cdot B_0 \sum_{k=1}^N (b_n + i a_n) \left(\frac{x + iy}{\text{RSC} \cdot R_0} \right)^k \quad (10)$$

For a magnet of length L and dipole field strength B_0 the resulting multipole kicks are then

$$\frac{\Delta p_x}{p_0} = - \text{STR} \cdot B_y, \quad \frac{\Delta p_y}{p_0} = \text{STR} \cdot B_x \quad (11)$$

with

$$\text{STR} = \frac{L}{B\rho} \quad (12)$$

Note : There are a couple of redundancies in terms of scaling factors which if not realized could lead to errors. The quantities SCF and R_0 reflect the fact that magnet designers like to express the multipole expansion coefficients a_n and b_n in units of 10^{-4} at a given radius r_0 , e.g. $r_0 = 0.01$ m. In this case one can use the magnet builders a_n and b_n values together with $\text{SCF} = 0.0001$ and $R_0 = 0.01$.

As one sees, the quasi-local scheme is less sensitive to ε (i.e. H_0) than a lumped correctors. In the next section, we derive such a scheme for the half dipole cell with errors. (3 correctors)

2. Application. The Neuffer Scheme.

Let us write $V(s)$ as :

$$V(s) = (\alpha(s) + \beta(s)) v. \quad (2.1)$$

Here v is an s -independent operator. The R -number α and β represent the "errors" and their corrections respectively.

Let us solve the case of a Half-cell with 6-dipoles and 3 correctors. For this we must have :

$$\alpha(s) = \sum_{i=1}^6 \alpha_i \theta_i \quad (2.2a)$$

$$\theta_i = 1 \quad \frac{(i-1)L}{6} \leq s \leq \frac{iL}{6} \quad i=1, 6 \quad (2.2b)$$

$$\beta(s) = \sum_{i=1}^3 L b_i \delta\left(s - \frac{(i-1)L}{2}\right) \quad i=1, 3. \quad (2.2c)$$

5. Input format

5a.) Lattice file input

After removal of the control commands from the lattice input files for FASTRAC or THINTRAC the lattice files consist of up to nine sections :

- Single element list,
- Block definitions,
- Structure input,
- Multipole coefficients,
- Cut of fluctuation random number distribution,
- Fluctuation random number seed,
- Tune variation,
- Chromaticity correction .
- Monitor position error random number seed and distribution cut

As mentioned in the first chapter the RACETRACK input format was kept, thus there are only a few changes and new additions which are discussed next. The expansion from one to many different multipole blocks required the following changes :

In the section 'Single element list' the input format for a multipole block element remains unchanged :

```
NAME * KZ STR RSC
( format A4, A1, I3, F12.7 F12.5 ),
```

where NAME consists of 4 characters, while KZ = 11 . The meanings of STR and RSC are the same as in RACETRACK and will be discussed below. The presence of such an element in the 'Single element list' requires a corresponding input of the multipole expansion coefficients in the section 'Multipole coefficients'. After the section title

MULTIPOLE COEFFICIENTS

comes a line containing the three input variables

```
TNM R0 SCF
( format A3, 7X, F10.5, F10.5 ),
```

Breaking f_1 in two parts:

$$f_1 = f_\alpha + f_\beta = \left\{ \int_0^L \exp(s \varepsilon \hat{H}_0) \alpha(s) ds \right\} v + \left\{ \int_0^L \exp(s \varepsilon \hat{H}_0) \beta(s) ds \right\} v \quad (2.3)$$

Using equation (2.2):

$$f_\alpha = \frac{\left\{ \exp\left(\frac{L}{6} \varepsilon \hat{H}_0\right) - 1 \right\}}{\varepsilon \hat{H}_0} \left\{ \sum_j \exp\left(\frac{(j-1)L}{6} \varepsilon \hat{H}_0\right) \alpha_j \right\} v$$

$$f_\beta = (L b_1 + \exp\left(\frac{L}{2} \varepsilon \hat{H}_0\right) L b_2 + \exp(L \varepsilon \hat{H}_0) L b_3) v \quad (2.4)$$

Now, we expand (2.4) in powers of ε :

$$f_\alpha = \left(\frac{L}{6} \sum \alpha_j \right) v + \left(\frac{L}{6} \sum \frac{(j-1)L}{6} \alpha_j + \frac{1}{2} \left(\frac{L}{6}\right)^2 \sum \alpha_j \right) \varepsilon \hat{H}_0 v + \left(\frac{1}{2} \left(\frac{L}{6}\right)^2 \sum \frac{(j-1)L}{6} \alpha_j + \frac{L}{6} \sum \left(\frac{(j-1)L}{6}\right)^2 \alpha_j + \frac{1}{6} \left(\frac{L}{6}\right)^2 \sum \alpha_j \right) \varepsilon^2 \hat{H}_0^2 v$$

$$f_\beta = (L \sum b_i) v + \left\{ \frac{L^2}{2} b_2 + L^2 b_3 \right\} \varepsilon \hat{H}_0 v + \left\{ L \left(\frac{L}{2}\right)^2 \frac{b_2}{2} + \frac{L^3}{2} b_3 \right\} \varepsilon^2 \hat{H}_0^2 v \quad (2.5)$$

a 4-d closed orbit

4. Code description

4.a) Description of flow chart of FASTRAC / THINTRAC

The original program RACETRACK had a predetermined order of execution. We kept this questionable philosophy. It is most simple to understand FASTRAC / THINTRAC by looking at the unambiguous tree (Fig. 1) which the execution must follow.

The first three boxes prepare the lattice before any operations (fitting) can be performed. The parameter 'NREAD' allows for the insertion of pre-calculated corrector strengths (more precisely : 2 quadrupoles, 2 sextupoles and the horizontal and vertical dipole correctors) . At this point we have added another new feature : The ability to generate a map for an arbitrary lattice (non - periodic, non - closing, even unstable) , by taking the branch $NUML1 = 0$. If instead $NUML1 = \pm 1$ or ± 2 , the user is assuming that a stable, 4 - dimensional optical solution exists for his lattice. In case of negative values for $NUML1$ the program will execute fitting procedures and produce a modified lattice. After completion of these operations the lattice is now (hopefully !) set according to the 4 - dimensional specifications of unit 5 and unit 16 (lattice input and command input file) .

Finally , we reach the last node. In THINTRAC the user decides between continuing all the map producing and tracking algorithm in 4 - d or going into 6 - d with the cavities set. Clearly, in FASTRAC , only the 4 - d branch is available. Furthermore it is important to note that a stable 6 - d optics solution must exist, should the user decide to use the 6 - d branch in THINTRAC. (The 6-d branch is interactive, the user sees the 6-d fixed point iterations and decides by himself to stop the procedure).

4.b) The map generating routine

The program has the ability to produce a map (from linear to 5th order) around any trajectory and between any two points of the lattice. This feature is activated by selecting an integer between 2 and 6 for the parameter 'MARY' of unit 16, the zero - value deactivates this option.

The program will generate the closed orbit (4 - d or 6 - d) map of order $N = MARY - 1$ for the full lattice if the parameter $NUML1$ has the values ± 1 or ± 2 (stable solution expected) .

For an arbitrary ray one selects $NUML1 = 0$ and enters the ray $EF(j)$, $j = 1, \dots, 4$ or 6 on the first line of the "GES" - file of unit 16.

Finally, if the maps for portions of the lattice are required, the user can use the new print command of the lattice input file, unit 5, by means of the element code number ('Kennzahl') $KZ = 13$, in order to print the map at selected locations. FASTRAC produces a 4 - d map and THINTRAC generates a 6 - d map (i.e. with p_{τ} -, and possible τ - dependence) .

Equating by orders $f_\alpha = -f_\beta$, we get three equations

$$\frac{1}{6} \{ \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 + \alpha_5 + \alpha_6 \} = -b_1 - b_2 - b_3$$

$$\frac{1}{72} \{ \alpha_1 + 3\alpha_2 + 5\alpha_3 + 7\alpha_4 + 9\alpha_5 + 11\alpha_6 \} = -\frac{b_2}{2} - b_3$$

$$\frac{1}{1296} \{ \alpha_1 + 7\alpha_2 + 19\alpha_3 + 37\alpha_4 + 61\alpha_5 + 91\alpha_6 \} = -\frac{b_2}{8} - \frac{b_3}{2} \quad (2.6)$$

Solving (2.6) gives the answers:

$$b_1 = \frac{-83\alpha_1 - 41\alpha_2 - 11\alpha_3 + 7\alpha_4 + 13\alpha_5 + 7\alpha_6}{648}$$

$$b_2 = - \left(\frac{4\alpha_1 + 10\alpha_2 + 13\alpha_3 + 13\alpha_4 + 10\alpha_5 + 4\alpha_6}{81} \right)$$

$$b_3 = \frac{7\alpha_1 + 13\alpha_2 + 7\alpha_3 - 11\alpha_4 - 41\alpha_5 - 83\alpha_6}{648} \quad (2.7)$$

Notice that for the case $\alpha_j = \alpha \forall j$, one gets:

$$\begin{cases} b_2 = -\frac{2}{3} \alpha \\ b_1 = b_3 = -\frac{\alpha}{6} \end{cases} \quad (2.8)$$

3c.) Multipole blocks

RACETRACK recognizes two kinds of non-linear elements :

A 'single multipole moment' of a given order $2n$ recognized by :

- a) length $EL = 0.0$ (non-linear element)
- b) single element type code $KZ = +n$ (normal multipole)
 $KZ = -n$ (skew multipole), $n = 1, \dots, 10$

or the 'multipole block', representing skew and normal multipoles up to 20 - pole identified by :

- a) length $EL = 0.0$ (non-linear element)
- b) single element type code $KZ = 11$ (multipole block).

However, the multipole block feature is limited to one magnet type (i.e. one set of multipole coefficients) . In FASTRAC and THINTRAC this shortcoming has been corrected and these codes accept up to 80 different blocks. Since most of this modification is done by input format changes the details of this feature are given in chapter 5 "Input format".

3d.) Input files

RACETRACK uses a single input file containing the lattice information as well as all the control commands. In FASTRAC and THINTRAC these two functions have been separated : The lattice information is contained in the 'INPUT' - file, while all the control commands are read from the 'GES' - file. The details are given in chapter 5 "Input format" .

3e.) Map printing command

This feature allows the user to print out the map at any location in the lattice . These locations are recognized by a special marker which is introduced as a dummy element with $KZ = 13$ in the 'single element list'. If the map routine is invoked the program initializes the map at the beginning of the structure. It concatenates the operations of the code until it encounters a type code 13 element (or the end of the structure, if no elements with $KZ = 13$ are used). At this point the map is printed in unit 18 and is re-initialized and the local orbit is written to unit 6 . Hence, one obtains the full map as the product of the maps between the print command markers. The map producing cycle is completed after the prompt '*MARYLIE 5.0*'.
written

- Conclusion -

The local and quasi-local correction emphasize the independence of M on " ϵ " (or H_0). Hence, one never has to rely on tunes or other H_0 -lattice parameters.

As a result, the quasi-local scheme effectiveness is determined by the validity of the " ϵ " expansion. This sets the limit for the size of the errors $\alpha(s)$. Misalignments and imperfections of quadrupoles and dipoles will not greatly change the validity of the " ϵ "-expansion and therefore one can expect (and

E.F. and J.P. have verified it numerically) the correction to be independent of H_0 -errors.

To change the limit on the errors $\alpha(s)$, the H_0 -errors (coupling, etc..) would have to increase the half-cell phase advance well beyond 45° . This is why the scheme can be properly called quasi-local.

The rotation proved to be a very useful feature in tracking realistic SSC-lattices that included elements common to both beams such as the IR-quad triplets. A combination of phase space rotation -- kick -- phase space rotation provides then the off-axis displacement of the beam typical for those common elements. The element type code permits the user to produce phase space rotations of angles Θ_x and Θ_y . The angles (Θ_x , Θ_y) are read in degrees and stored in the arrays ED and EK, respectively. This element must be given a non-zero length in order for the code to recognize it as a linear element. However, for all purposes where the actual element length matters, e.g. the accumulated length in the linear optics, the length of any element with KZ = 9 is given a zero value. The effect of a rotation is described by the following matrix :

$$\begin{bmatrix} x \\ p_x \\ y \\ p_y \end{bmatrix} = \begin{bmatrix} \cos\Theta_x & \sin\Theta_x & 0 & 0 \\ -\sin\Theta_x & \cos\Theta_x & 0 & 0 \\ 0 & 0 & \cos\Theta_y & \sin\Theta_y \\ 0 & 0 & -\sin\Theta_y & \cos\Theta_y \end{bmatrix} * \begin{bmatrix} x_o \\ p_{x,o} \\ y_o \\ p_{y,o} \end{bmatrix} \quad (8)$$

THINTRAC is equipped to handle synchrotron oscillations correctly. Instead of using the element name 'CAVI' to designate a RF - cavity, as is done in RACETRACK, the element type code KZ = 12 defines a RF - cavity. The kick produced by a type code 12 element is given by :

$$\begin{cases} \tau^* = \tau - \tau_o \\ p_\tau^f = p_\tau^i + A \cdot \sin\left(\frac{\omega}{c} \cdot \tau^*\right) \end{cases} \quad (9)$$

The quantity A is entered in the 'single element list' of the lattice input file (unit 5) in place of the variable EK, ω/c in place of ED and τ_o in place of EL. It is worth noting that both ED and EL are set to zero just after reading and storing the two quantities in the arrays XPL and ZPL, respectively. This allows the code to treat the cavities in the "non-linear" loop. The units of the cavity parameters are given in chapter 5a.) "Lattice file input".

Note : As a precaution one should not use the name 'CAVI' for any single element, in order to avoid any possibility of invoking some feature of the original code still hidden in FASTRAC or THINTRAC.