Using Eigenvectors as Constraints in MAD-X Matching Module

Valery Kapin, *Moscow Engineering Physics Institute*

Yuri Alexahin, *Fermi National Accelerator Laboratory*

Frank Schmidt, *European Organization for Nuclear Research*

1. Introduction

Computer simulation of beam optics is one of the tools for understanding and upgrading performances of existing machines. A correct simulation model of the machine lattice requires a precise knowledge of physical parameters for every lattice element. Effective parameters of the physical elements can be reconstructed from beam measurements, while they can be related with measurement data either directly with an orbit response matrix [1] or by means of intermediate parameters, e.g. lattice optical functions [2]. Presently, both algorithms are implemented for the correction of linear coupling in TEVATRON [2, 3].

In this note, the second approach for two-dimensional (2D) linearly coupled betatron motion is implied, while the turn-by-turn (TBT) data are considered to be related with normal modes of betatron oscillations [2]. Correct values of physical parameters can be found by a matching procedure of a simulation code when physical parameters are varied in order to match to constraints given by values of lattice optical functions.

The 2D linear coupled motion can be decoupled into normal modes either using the Edward-Teng method [4] or using the approach developed by Ripken [5]. Simple relations between the Edwards-Teng and the Ripken lattice functions are known [6]. The Ripken parameterization explicitly deals with normal modes and is most suitable for the TBT data analysis [7]. In this note, the lattice eigenvectors given by the Ripken parameters are used as constraints in the matching procedure.

The MAD code is well established tool for beam dynamics simulations [8]. We are going to adjust its successor MAD-X [9] to simulate the beam optics in TEVATRON rings. The MAD-X (version 3.03.13) is able to calculate both Edwards-Teng and Ripken parameters using the commands "TWISS" and "PTC_TWISS", respectively. In former case, the coupled matrix and the TWISS parameters are derived from the 4-by-4 one-turn linear matrix [10]. In the latter case, the 6-by-6 eigenvector matrix is extracted from the normal forms derived from the non-linear map by PTC-library [11]. The "TWISS" command deals with simple 4-by-4 matrix calculations and run faster than the "PTC_TWISS" which dealt with non-linear maps and normal forms and intended for more general 3D calculations.

A matching procedure with a simulation code for large rings like TEVATRON may use huge numbers of both the varying parameters and the targeting constrains. Therefore, it requires fast and simple calculations of a penalty function, which is in fact an algebraic combination of all constraints. In this note, we have shown that the Ripken parameters for 2D case can be calculated using only the "TWISS" command utilizing simple expressions in the MAD-X input file. The results of the "TWISS" command have been tested with a simple ring lattice and verified using results provided by the "PTC_TWISS" command. The regular MAD-X matching procedure with macros existing in "MAD-X-3.03.13" [12] is used in our linear case for both commands. Note, that this matching is primarily intended for the nonlinear parameters. In our tests, matching with the "TWISS" command has run up to ten times faster than one with the "PTC_TWISS" command.

We may conclude that similar simple formulae can be implemented in the TWISS-module of MAD-X for 2D calculations of the Ripken parameters. Such update of MAD-X may be desirable in order to reduce computation time of 2D matching procedures for large rings. An alternative way is to implement the new PTC matching [13] speeding up the computations by one-two orders.

Calculations and Matching for the Eigenvectors with PTC_TWISS

The "PTC_TWISS" command of MAD-X [12] calculates the Ripken parameters and the eigenvectors for the 3D case using the PTC_TWISS module based on the PTC code. This module uses the "Normal-Form" technique of PTC which is a universal tool for treatments of non-linear maps. The module performs a normal form on a given map. Normal form contains all sorts of information, including the lattice functions. The 6-by-6 eigenvector matrix is also extracted from the normal forms.

PTC_TWISS command can list both the Ripken parameters and components of eigenvectors. According to the Ripken notation [5], the point in the 4-D phase space $\vec{z}(s) = (x, x', y, y')^T$ at position s is expressed as:

$$\vec{z}(s) = \sqrt{\varepsilon_{\mathrm{I}}}\left[\vec{z}_1(s)\cos\phi_{\mathrm{I}} - \vec{z}_2(s)\sin\phi_{\mathrm{I}}\right] + \sqrt{\varepsilon_{\mathrm{II}}}\left[\vec{z}_3(s)\cos\phi_{\mathrm{II}} - \vec{z}_4(s)\sin\phi_{\mathrm{II}}\right], \quad (1)$$

where $\varepsilon_{\mathrm{I,II}}$ and $\phi_{\mathrm{I,II}}$ are constants for the I-st and II-nd eigen-modes, the column-vector $\vec{z}_j$ ($j = 1,\ldots,4$) has four components $\vec{z}_j = \left(V_{1,j} \quad V_{2,j} \quad V_{3,j} \quad V_{4,j}\right)^T$, which are related to the Ripken parameters with the following matrix equation:

$$\left|V_{i,j}\right| = \begin{pmatrix} V_{11} & V_{12} & V_{13} & V_{14} \\ V_{12} & V_{22} & V_{23} & V_{24} \\ V_{31} & V_{32} & V_{33} & V_{34} \\ V_{41} & V_{42} & V_{43} & V_{44} \end{pmatrix} = \begin{pmatrix} \sqrt{\beta_{x\mathrm{I}}}\cos\Phi_{x\mathrm{I}} & \sqrt{\beta_{x\mathrm{I}}}\sin\Phi_{x\mathrm{I}} & \sqrt{\beta_{x\mathrm{II}}}\cos\Phi_{x\mathrm{II}} & \sqrt{\beta_{x\mathrm{II}}}\sin\Phi_{x\mathrm{II}} \\ \sqrt{\gamma_{x\mathrm{I}}}\cos\breve{\Phi}_{x\mathrm{I}} & \sqrt{\gamma_{x\mathrm{I}}}\sin\breve{\Phi}_{x\mathrm{I}} & \sqrt{\gamma_{x\mathrm{II}}}\cos\breve{\Phi}_{x\mathrm{II}} & \sqrt{\gamma_{x\mathrm{II}}}\sin\breve{\Phi}_{x\mathrm{II}} \\ \sqrt{\beta_{y\mathrm{I}}}\cos\Phi_{y\mathrm{I}} & \sqrt{\beta_{y\mathrm{I}}}\sin\Phi_{y\mathrm{I}} & \sqrt{\beta_{y\mathrm{II}}}\cos\Phi_{y\mathrm{II}} & \sqrt{\beta_{y\mathrm{II}}}\sin\Phi_{y\mathrm{II}} \\ \sqrt{\gamma_{y\mathrm{I}}}\cos\breve{\Phi}_{y\mathrm{I}} & \sqrt{\gamma_{y\mathrm{I}}}\sin\breve{\Phi}_{y\mathrm{I}} & \sqrt{\gamma_{y\mathrm{II}}}\cos\breve{\Phi}_{y\mathrm{II}} & \sqrt{\gamma_{y\mathrm{II}}}\sin\breve{\Phi}_{y\mathrm{II}} \end{pmatrix} \quad (2)$$

After substitution eqn. (2) into eqn. (1), one gets the particle coordinates expressed with the Ripken parameters

$$\begin{pmatrix} x \\ x' \\ y \\ y' \end{pmatrix} = \begin{pmatrix} \sqrt{\varepsilon_{\mathrm{I}}\beta_{x\mathrm{I}}}\left(\cos\phi_{\mathrm{I}}\cos\Phi_{x\mathrm{I}} - \sin\phi_{\mathrm{I}}\sin\Phi_{x\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}\beta_{x\mathrm{II}}}\left(\cos\phi_{\mathrm{II}}\cos\Phi_{x\mathrm{II}} - \sin\phi_{\mathrm{II}}\sin\Phi_{x\mathrm{II}}\right) \\ \sqrt{\varepsilon_{\mathrm{I}}\gamma_{x\mathrm{I}}}\left(\cos\phi_{\mathrm{I}}\cos\breve{\Phi}_{x\mathrm{I}} - \sin\phi_{\mathrm{I}}\sin\breve{\Phi}_{x\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}\gamma_{x\mathrm{II}}}\left(\cos\phi_{\mathrm{II}}\cos\breve{\Phi}_{x\mathrm{II}} + \sin\phi_{\mathrm{II}}\sin\breve{\Phi}_{x\mathrm{II}}\right) \\ \sqrt{\varepsilon_{\mathrm{I}}\beta_{y\mathrm{I}}}\left(\cos\phi_{\mathrm{I}}\cos\Phi_{y\mathrm{I}} - \sin\phi_{\mathrm{I}}\sin\Phi_{y\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}\beta_{y\mathrm{II}}}\left(\cos\phi_{\mathrm{II}}\cos\Phi_{y\mathrm{II}} - \sin\phi_{\mathrm{II}}\sin\Phi_{y\mathrm{II}}\right) \\ \sqrt{\varepsilon_{\mathrm{I}}\gamma_{y\mathrm{I}}}\left(\cos\phi_{\mathrm{I}}\cos\breve{\Phi}_{y\mathrm{I}} - \sin\phi_{\mathrm{I}}\sin\breve{\Phi}_{y\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}\gamma_{y\mathrm{II}}}\left(\cos\phi_{\mathrm{II}}\cos\breve{\Phi}_{y\mathrm{II}} + \sin\phi_{\mathrm{II}}\sin\breve{\Phi}_{y\mathrm{II}}\right) \end{pmatrix} \quad (3)$$

The "PTC_TWISS" commands tracks the eigenvectors and prints components of the above matrix $\mathbf{V}$ to the "Twiss" table, if they are selected with the "SELECT" command (flag=ptc_twiss). Particular components $V_{i,j}$ can be selected with the names `EIGNij`. The Ripken parameters can be printed either directly by selecting corresponding names (e.g., `beta11`, `beta12`,…) or calculated from the eigen-vector components within input script according to following formulae:

$$\begin{array}{cccc} \beta_{x\mathrm{I}} = V_{11}^2 + V_{12}^2; & \beta_{x\mathrm{II}} = V_{13}^2 + V_{14}^2; & \beta_{y\mathrm{I}} = V_{31}^2 + V_{32}^2; & \beta_{y\mathrm{II}} = V_{33}^2 + V_{34}^2; \\ \cos\Phi_{x\mathrm{I}} = V_{11}\big/\sqrt{\beta_{x\mathrm{I}}}; & \cos\Phi_{x\mathrm{II}} = V_{13}\big/\sqrt{\beta_{x\mathrm{II}}}; & \cos\Phi_{y\mathrm{I}} = V_{31}\big/\sqrt{\beta_{y\mathrm{I}}}; & \cos\Phi_{y\mathrm{II}} = V_{33}\big/\sqrt{\beta_{y\mathrm{II}}}; \\ \sin\Phi_{x\mathrm{I}} = V_{12}\big/\sqrt{\beta_{x\mathrm{I}}}; & \sin\Phi_{x\mathrm{II}} = V_{14}\big/\sqrt{\beta_{x\mathrm{II}}}; & \sin\Phi_{y\mathrm{I}} = V_{32}\big/\sqrt{\beta_{y\mathrm{I}}}; & \sin\Phi_{y\mathrm{II}} = V_{34}\big/\sqrt{\beta_{y\mathrm{II}}}. \end{array} \quad (4)$$

Note, that in the PTC_TWISS table, the beta-functions $\beta_{x\mathrm{I}}$, $\beta_{x\mathrm{II}}$, $\beta_{y\mathrm{I}}$, $\beta_{y\mathrm{II}}$ are denoted as `beta11`, `beta21`, `beta22`, `beta12`, respectively.

The matching procedure using the eigenvectors as the constraints can be done with the regular MAD-X matching with USE_MACRO keyword [12], which allows multiple the user-defined expressions within a macro script and primarily intended for the nonlinear parameters. Here, it is mainly used to verify our calculations with the "PTC_TWISS". The example below shows the general input structure for the matching procedure with the "PTC_TWISS" command.

1) The target values of constraints are assigned by following expressions:

```
V11_TBT_M01 = 2.393862974; V12_TBT_M02 = 0.4119203017;...
```

2) the calculated eigen-vector components $V_{1,1}$, $V_{1,2}$ at the elements labeled with `M01`, `M02` are extracted from "PTC_TWISS"-table and assigned to the variables `V11_MR_M01`, `V12_MR_M02`.

```
V11_MR_M01:=table(ptc_twiss,M01,eign11);
V12_MR_M02:=table(ptc_twiss,M02,eign12); ...
```

3) The following macro named as `ptc_eigen` contains a setup of PTC-environment, the "SELECT"-ion of the used eigen-vector components $V_{1,1}$, $V_{1,2}$ and the call of the "PTC_TWISS" command:

```
ptc_eigen: macro={
ptc_create_universe; ptc_create_layout,model=2,method=6,nst=10,exact;

select,flag=ptc_twiss,clear;
select,flag=ptc_twiss,column=name,eign11,eign12,...;
ptc_twiss,closed_orbit,icase=5;
ptc_end;};
```

4) The matching as defined above macro is initiated with the command `match, use_macro;` and is terminated with the command `endmatch;`. The matching block defines variable parameters (the strengths `K1` of two quadrupoles `QF1` and `QD1`), the name of used macro (`ptc_eigen`), the constrains imposed for the variables `V11_MR_M01`, `V12_MR_M02`, and the matching method (`jacobian`):

```
match, use_macro;
 VARY, NAME=QF1->K1;  VARY, NAME=QD1->K1;
 use_macro, name=ptc_eigen;
 constraint, expr=V11_MR_M01=V11_TBT_M01;
 constraint, expr=V12_MR_M02=V12_TBT_M02;...
 jacobian, CALLS=100, TOLERANCE=1.0E-10;
endmatch;
```

Calculations and Matching for Eigenvectors with TWISS

The "TWISS" command of MAD-X [12] calculates the Edwards-Teng parameters using the TWISS module, which is based on the corresponding subroutines copied from the MAD-8 code. The "Physical Guide" of MAD-8 [10] outlines the algorithm for the computations of the Edwards-Teng parameters. The method is similar to reference [10]. However, the FORTRAN code of the TWISS module in MAD-X uses slightly modified formulae presented below.

In the 4-D phase space, the linear 4×4 transfer matrix $\mathbf{M}$ partitioned into four 2×2 blocks ($\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$) can be transformed to a new block diagonal matrix $\mathbf{U}$ with two non-zero 2×2 matrices $\mathbf{E}^x$ and $\mathbf{E}^y$, i.e.:

$$\mathbf{M}=\begin{pmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{pmatrix} \text{ and } \mathbf{U}=\begin{pmatrix}\mathbf{E}^x & \mathbf{0}\\ \mathbf{0} & \mathbf{E}^y\end{pmatrix}. \tag{5}$$

The "symplectic rotation" is performed using the 2×2 coupling matrix $\boldsymbol{R}$. The TWISS module calculates the matrices $\boldsymbol{R}$, $\mathbf{E}^x$, $\mathbf{E}^y$ according to the following formulae:

$$\boldsymbol{R}=-\left\{\frac{1}{2}\left[\mathrm{Tr}(\mathbf{A})-\mathrm{Tr}(\mathbf{D})\right]+\mathrm{sign}\left[\mathrm{Tr}(\mathbf{A})-\mathrm{Tr}(\mathbf{D})\right]\sqrt{\det(\mathbf{C}+\overline{\mathbf{B}})+\frac{1}{4}\left[\mathrm{Tr}(\mathbf{A})-\mathrm{Tr}(\mathbf{D})\right]^2}\right\}^{-1}\left[\mathbf{C}+\overline{\mathbf{B}}\right], \tag{6}$$

$$\mathbf{E}^x=\mathbf{A}-\mathbf{B}\boldsymbol{R},\ \mathbf{E}^y=\mathbf{D}+\boldsymbol{R}\mathbf{B} \tag{7}.$$

The "TWISS" command calculates the components of the coupling matrix (`R11`, `R12`, `R21`, `R22` in the MAD-X notation) and the decoupled lattice functions for two-planes (`BETX`, `ALFX`, `MUX`, `BETY`, `ALFY`, `MUY`). At the initial point of the periodical lattice, the coupling matrix is calculated according to the eq. (5). The lattice functions for every plane are calculated similar to the Courant-Snyder parameters for uncoupled linear optics. For optimal numeric precision MAD uses the following formulas [10] for tunes $\mu^0_{x,y}$:

$$\begin{cases} \cos\left(2\pi\mu_{x,y}^{0}\right) = \left(E_{1,1}^{x,y} + E_{2,2}^{x,y}\right)/2 \\ \sin\left(2\pi\mu_{x,y}^{0}\right) = \mathrm{sign}\left(E_{1,2}^{x,y}\right)\sqrt{-E_{1,2}^{x,y}E_{2,1}^{x,y} - \left(E_{1,1}^{x,y} - E_{2,2}^{x,y}\right)^2/4} \\ \mu_{x,y}^{0} = (2\pi)^{-1}\arctan\left[\sin\left(2\pi\mu_{x,y}^{0}\right)/\cos\left(2\pi\mu_{x,y}^{0}\right)\right] \end{cases}. \tag{8}$$

The initial lattice functions are given as

$$\beta_{x,y}^{0} = E_{1,2}^{x,y}/\sin\left(2\pi\mu_{x,y}^{0}\right) \text{ and } \alpha_{x,y}^{0} = \left(E_{1,1}^{x,y} - E_{2,2}^{x,y}\right)/\left[2\sin\left(2\pi\mu_{x,y}^{0}\right)\right]. \tag{9}$$

Then, the initial decoupled lattice functions are tracked throughout the ring elements.

The eigenvectors of the coupled system are expressed by

$$\mathbf{V}_1 = \left(1 + \det(\boldsymbol{R})\right)^{-1/2}\begin{pmatrix} \boldsymbol{V}_1 \\ -\boldsymbol{R}\boldsymbol{V}_1 \end{pmatrix} \text{ and } \mathbf{V}_2 = \left(1 + \det(\boldsymbol{R})\right)^{-1/2}\begin{pmatrix} \overline{\boldsymbol{R}}\boldsymbol{V}_2 \\ \boldsymbol{V}_2 \end{pmatrix}, \tag{10}$$

where $\boldsymbol{V}_{1,2} = \begin{pmatrix} \beta_{x,y}^{1/2} & 0 \\ -\alpha_{x,y}\beta_{x,y}^{-1/2} & -\beta_{x,y}^{-1/2} \end{pmatrix}$.

The particle coordinates can be expressed by the following equation:

$$\begin{pmatrix} x \\ x' \\ y \\ y' \end{pmatrix} = \mathbf{V}_1\begin{pmatrix} a_x'\cos\varphi_x - a_x''\sin\varphi_x \\ a_x'\sin\varphi_x + a_x''\cos\varphi_x \end{pmatrix} + \mathbf{V}_2\begin{pmatrix} a_y'\cos\varphi_y - a_y''\sin\varphi_y \\ a_y'\sin\varphi_y + a_y''\cos\varphi_y \end{pmatrix}, \tag{11}$$

where $a_x = a_x' + ia_x''$ and $a_y = a_y' + ia_y''$ are the complex constants, and $\varphi_{x,y} = 2\pi\mu_{x,y}$.

After the substitution of eqn. (6)-(10) into eqn.(11), one gets the particle coordinates expressed with the Edward-Teng parameters:

$$\begin{pmatrix} x \\ x' \\ y \\ y' \end{pmatrix} = \begin{pmatrix} \left[1 + \det(\boldsymbol{R})\right]^{-1/2}\beta_x^{1/2}\left(a_x'\cos\varphi_x - a_x''\sin\varphi_x\right) \\ -\left[1 + \det(\boldsymbol{R})\right]^{-1/2}\beta_x^{-1/2}\left[\alpha_x\left(a_x'\cos\varphi_x - a_x''\sin\varphi_x\right) + \left(a_x'\sin\varphi_x + a_x''\cos\varphi_x\right)\right] \\ \left[1 + \det(\boldsymbol{R})\right]^{-1/2}\left[\left(-R_{11}\beta_x^{1/2} + R_{12}\alpha_x\beta_x^{-1/2}\right)\left(a_x'\cos\varphi_x - a_x''\sin\varphi_x\right) + R_{12}\beta_x^{-1/2}\left(a_x'\sin\varphi_x + a_x''\cos\varphi_x\right)\right] \\ \left[1 + \det(\boldsymbol{R})\right]^{-1/2}\left[\left(-R_{21}\beta_x^{1/2} + R_{22}\alpha_x\beta_x^{-1/2}\right)\left(a_x'\cos\varphi_x - a_x''\sin\varphi_x\right) + R_{22}\beta_x^{-1/2}\left(a_x'\sin\varphi_x + a_x''\cos\varphi_x\right)\right] \end{pmatrix} +$$

$$+\begin{pmatrix}
\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(+R_{22}\beta_y^{1/2}+R_{12}\alpha_y\beta_y^{-1/2}\right)\left(a_y'\cos\varphi_y-a_y''\sin\varphi_y\right)+R_{12}\beta_y^{-1/2}\left(a_y'\sin\varphi_y+a_y''\cos\varphi_y\right)\right] \\
\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(-R_{21}\beta_y^{1/2}-R_{11}\alpha_y\beta_y^{-1/2}\right)\left(a_y'\cos\varphi_y-a_y''\sin\varphi_y\right)-R_{11}\beta_y^{-1/2}\left(a_y'\sin\varphi_y+a_y''\cos\varphi_y\right)\right] \\
\left[1+\det(\boldsymbol{R})\right]^{-1/2}\beta_y^{1/2}\left(a_y'\cos\varphi_y-a_y''\sin\varphi_y\right) \\
-\left[1+\det(\boldsymbol{R})\right]^{-1/2}\beta_y^{-1/2}\left[\alpha_y\left(a_y'\cos\varphi_y-a_y''\sin\varphi_y\right)+\left(a_y'\sin\varphi_y+a_y''\cos\varphi_y\right)\right]
\end{pmatrix} \tag{12}$$

There is a simple relation between The Ripken and Edward-Teng expressions:

$$\boxed{\begin{aligned}
&\beta_{x\mathrm{I}}=\left[1+\det(\boldsymbol{R})\right]^{-1}\beta_x, \quad \varphi_x=\Phi_{x\mathrm{I}}, \quad a_x'=\varepsilon_{\mathrm{I}}^{1/2}\cos\phi_{\mathrm{I}}, \quad a_x''=\varepsilon_{\mathrm{I}}^{1/2}\sin\phi_{\mathrm{I}}, \\
&\beta_{y\mathrm{II}}=\left[1+\det(\boldsymbol{R})\right]^{-1}\beta_y, \quad \varphi_y=\Phi_{y\mathrm{II}}, \quad a_y'=\varepsilon_{\mathrm{II}}^{1/2}\cos\phi_{\mathrm{II}}, \quad a_y''=\varepsilon_{\mathrm{II}}^{1/2}\sin\phi_{\mathrm{II}},
\end{aligned}} \tag{13}$$

$$\boxed{\begin{aligned}
\beta_{x\mathrm{II}}&=\left[1+\det(\boldsymbol{R})\right]^{-1}\left[R_{22}\left(R_{22}\beta_y+2R_{12}\alpha_y\right)+R_{12}^2\gamma_y\right], \\
\beta_{y\mathrm{I}}&=\left[1+\det(\boldsymbol{R})\right]^{-1}\left[R_{11}\left(R_{11}\beta_x-2R_{12}\alpha_x\right)+R_{12}^2\gamma_x\right],
\end{aligned}} \tag{14}$$

$$\boxed{\begin{aligned}
\sin\Phi_{x\mathrm{II}}&=\beta_{x\mathrm{II}}^{-1/2}\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(R_{22}\beta_y^{1/2}+R_{12}\alpha_y\beta_y^{-1/2}\right)\sin\varphi_y-R_{12}\beta_y^{-1/2}\cos\varphi_y\right], \\
\cos\Phi_{x\mathrm{II}}&=\beta_{x\mathrm{II}}^{-1/2}\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(R_{22}\beta_y^{1/2}+R_{12}\alpha_y\beta_y^{-1/2}\right)\cos\varphi_y+R_{12}\beta_y^{-1/2}\sin\varphi_y\right],
\end{aligned}} \tag{15}$$

$$\boxed{\begin{aligned}
\sin\Phi_{y\mathrm{I}}&=\beta_{y\mathrm{I}}^{-1/2}\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(-R_{11}\beta_x^{1/2}+R_{12}\alpha_x\beta_x^{-1/2}\right)\sin\varphi_x-R_{12}\beta_x^{-1/2}\cos\varphi_x\right] \\
\cos\Phi_{y\mathrm{I}}&=\beta_{y\mathrm{I}}^{-1/2}\left[1+\det(\boldsymbol{R})\right]^{-1/2}\left[\left(-R_{11}\beta_x^{1/2}+R_{12}\alpha_x\beta_x^{-1/2}\right)\cos\varphi_x+R_{12}\beta_x^{-1/2}\sin\varphi_x\right]
\end{aligned}} \tag{16}$$

where $\gamma_{x,y}\equiv\beta_{x,y}^{-1}\left(\alpha_{x,y}^2+1\right)$. The above equations can be derived by comparison of the equations (3) and (12) for every coordinate. Using eqns. (15, 16), one may obtain relations between phases

$$\Phi_{y\mathrm{I}}=\Phi_{x\mathrm{I}}-\Theta_{y\mathrm{I}} \text{ and } \Phi_{x\mathrm{II}}=\Phi_{y\mathrm{II}}-\Theta_{x\mathrm{II}}, \tag{17}$$

where $\Theta_{y\mathrm{I}}=\operatorname{arccot}\left[\alpha_x-\beta_x\left(R_{11}/R_{12}\right)\right]$ and $\Theta_{x\mathrm{II}}\equiv\operatorname{arccot}\left[\beta_y\left(R_{22}/R_{12}\right)+\alpha_y\right]$.
Using the above equations, we have tested the matching procedure using the Ripken parameters as the constraints derived from Edward-Teng parameters calculated by the "TWISS" command.

The example below shows the general input structure for the matching procedure with the "TWISS" command. The input structure includes all items used for the previously discussed "PTC_TWISS" command and adds the item (numbered by "2") which defines the equations (13)-(16).

1) The target values of constraints are assigned by following expressions:

```
V11_TBT_M01 = 2.393862974; V12_TBT_M02 = 0.4119203017;...
```

2) The equations (13)-(16) are coded in the following lines:

```
nonzero := 1.0; ! <==== to avoid C-warning messages ("divisions by zero")
ETAX := nonzero; BETAY := nonzero; detR := nonzero; RM12 := nonzero;

GAMAX := (ALFAX^2+1)/BETAX; GAMAY := (ALFAY^2+1)/BETAY;
detR := RM11*RM22-RM12*RM21; den_detR := 1.0/(1+detR);

betaX1_ET := den_detR*BETAX;  ! <<<< ===== X1 =====>>>>>
PhiX1_ET := 2*Pi*MU_X; cos_PhiX1_ET := cos(PhiX1_ET); sin_PhiX1_ET := sin(PhiX1_ET);
V11_ET := sqrt(betaX1_ET)*cos_PhiX1_ET; V12_ET := sqrt(betaX1_ET)*sin_PhiX1_ET;

betaY2_ET := den_detR*BETAY; ! <<<<<======== Y2 =======>>>>>>
PhiY2_ET := 2*Pi*MU_Y; cos_PhiY2_ET := cos(PhiY2_ET); sin_PhiY2_ET := sin(PhiY2_ET);
V33_ET := sqrt(betaY2_ET)*cos_PhiY2_ET; V34_ET := sqrt(betaY2_ET)*sin_PhiY2_ET;

betaX2_ET := den_detR*(RM22*(RM22*BETAY+2.0*RM12*ALFAY)+GAMAY*RM12^2); ! <= X2 =>
py1_ET := RM22*sqrt(BETAY)+RM12*ALFAY/sqrt(BETAY); py2_ET := RM12/sqrt(BETAY);
sin_PhiX2_ET :=
(py1_ET*sin_PhiY2_ET-py2_ET*cos_PhiY2_ET)/sqrt(betaX2_ET)*sqrt(den_detR);
cos_PhiX2_ET :=
(py1_ET*cos_PhiY2_ET+py2_ET*sin_PhiY2_ET)/sqrt(betaX2_ET)*sqrt(den_detR);
V13_ET := sqrt(betaX2_ET)*cos_PhiX2_ET; V14_ET := sqrt(betaX2_ET)*sin_PhiX2_ET;

betaY1_ET := den_detR*(RM11*(RM11*BETAX-2.0*RM12*ALFAX)+GAMAX*RM12^2); ! <= Y1 =>
px1_ET := -RM11*sqrt(BETAX)+RM12*ALFAX/sqrt(BETAX); px2_ET := RM12/sqrt(BETAX),
sin_PhiY1_ET :=
(px1_ET*sin_PhiX1_ET-px2_ET*cos_PhiX1_ET)/sqrt(betaY1_ET)*sqrt(den_detR);
cos_PhiY1_ET :=
(px1_ET*cos_PhiX1_ET+px2_ET*sin_PhiX1_ET)/sqrt(betaY1_ET)*sqrt(den_detR);
V31_ET := sqrt(betaY1_ET)*cos_PhiY1_ET; V32_ET := sqrt(betaY1_ET)*sin_PhiY1_ET;

! <<<< ==== Get E-T parameters from the TWISS table ====>>>>
ALFAX := table(twiss,ALFX);  BETAX := table(twiss,BETX); MU_X := table(twiss,MUX);
ALFAY := table(twiss,ALFY);  BETAY := table(twiss,BETY); MU_Y := table(twiss,MUY);
RM11 := table(twiss, R11); RM12 := table(twiss, R12);
RM21 := table(twiss, R21); RM22 := table(twiss, R22);
```

2) the calculated eigen-vector components $V_{1,1}$, $V_{1,2}$ at the elements labeled with `M01`, `M02` are extracted from "TWISS"-table and assigned to the variables `V11_MR_M01`, `V12_MR_M02`.

```
V11_ET_M01:=table(twiss,M01,V11_ET);
V12_ET_M02:=table(twiss,M02,V12_ET); ...
```

3) The following macro named as `TWISS_eigen` does not contain a setup of PTC-environment as the `ptc_eigen` -macro, and contains only the SELECTion of the Edwards-Teng parameters and used eigen-vector components $V_{1,1}$, $V_{1,2}$, and the call of "TWISS" command:

```
TWISS_eigen: macro={
select,flag=twiss,clear;
select, flag=twiss,column=name,keyword, s, L , ANGLE,
BETX, ALFX, MUX, BETY, ALFY, MUY, R11, R12, R21, R22,
V11_ET, V12_ET, V13_ET, V14_ET, V31_ET, V32_ET, V33_ET, V34_ET,
betaX1_ET, betaX2_ET, betaY2_ET, betaY1_ET,
cos_PhiX1_ET, sin_PhiX1_ET, cos_PhiX2_ET , sin_PhiX2_ET,
cos_PhiY1_ET, sin_PhiY1_ET, cos_PhiY2_ET , sin_PhiY2_ET;
twiss;};
```

4) The matching block is essentially the same as in the case of `ptc_eigen` -macro. It defines variable parameters (the strengths `K1` of two quadrupoles `QF1` and `QD1`), the name of used macro (`TWISS_eigen`), the constrains imposed for the variables `V11_MR_M01`, `V12_MR_M02`, and the matching method (`jacobian`):

```
match, use_macro;
 VARY, NAME=QF1->K1; VARY, NAME=QD1->K1;
use_macro, name=TWISS_eigen;
```

```
constraint, expr = V11_ET_M01 = V11_TBT_M01;
constraint, expr = V12_ET_M02 = V12_TBT_M02;...
JACOBIAN, CALLS = 100, TOLERANCE = 1.0E-10;
endmatch;
```

The Testing Results

The testing computations have been performed on PC with Pentium-IV (2 GHz) processor.

A simple four-bend ring lattice consisting of four FODO-cells has been explored. The calculations of eigenvector components as well as the Ripken parameters with "TWISS" command provide numerical results identical to one with the "PTC_TWISS". This confirms that the formulae (13)-(16) used for our calculations with the "TWISS" command are correct.

The matching has been tested with two variables and eight constraints given by the eigenvectors at the different ring locations. Three methods of MAD-X matching module have been explored for both "TWISS" and "PTC_TWISS" commands. The results are combined in the table below.

	LMDIF	SIMPLEX	JACOBIAN
Number of calls	16	100	6
Minimum of the penalty function	$6.54 \cdot 10^{-13}$	$3.83 \cdot 10^{-3}$	$5.97 \cdot 10^{-16}$
Time for "TWISS", sec	8	12	8
Time for "PTC_TWISS",sec	27	140	31

The number of the penalty function calls and its minimal values are the same for both "TWISS" and "PTC_TWISS" commands. We can see that the computation times with the "TWISS" command is noticeably less then time with the "PTC_TWISS" command ranging from 3 to 10 times for different methods. Note, that the above computation times are not net times spent by matching, they also include the time spent by MAD-X for other commands of the input script. Therefore, the times for the fast methods ("LMDIFF" and "JACOBIAN") include a considerable part spent for other irrelevant commands. Thus, the actual ratio between the computational times for the fast methods

should be higher than a factor 3. However, this result is easily predictable since our 2D linear task is not objective of the 3D "PTC_TWISS" and the MAD-X (version 3.03.13) nonlinear matching with macros. Recent improvements for the PTC matching [13] have been implemented in the MAD-X version 3.03.18. They can potentially improve the computation times by a factor 10 and practically equalized times for the matching with "TWISS" and "PTC_TWISS".

The described method can also take into account both calibration factors and tilts of BPMs. The formulation of this approach is given in Appendix.

References

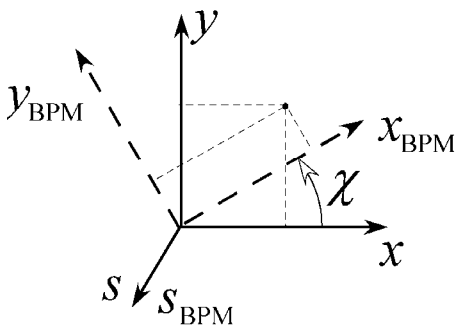
1. J. Safranek, "Experimental determination of storage ring optics using orbit response measurements", Nucl. Instr. And Meth. In Phys. Res., A-388 (1997), pp.27-36.
2. Y. Alexahin, E. Gianfelice-Wendt, "A New Algorithm for the Correction of the Linear Coupling at TEVATRON", Proc. 2006 European Part. Acc. Conf., pp. 2047-2049.
3. V. Sajaev, V. Lebedev, V. Nagaslaev, V. Valishev, "Fully Coupled Analysis of Orbit Response Matrices at the FNAL TEVATRON", Proc. 2005 Part. Acc. Conf., pp. 3662-3665.
4. D.A. Edwards and L.C. Teng, "Parametrization of Linear Coupled Motion in Periodic Systems", IEEE Trans. Nucl. Sci., 20 (1973), pp.885-888.
5. G. Ripken and F. Willeke, "Methods of Beam Optics", DESY 88–114, 1988.
6. V.A. Lebedev and S.A. Bogacz, "Betatron motion with coupling of horizontal and vertical degrees of freedom", JLAB-ACC-99-19, 2001 (the updated version at http://www-bdnew.fnal.gov/pbar/organizationalchart/lebedev/Articles/index.htm).
7. Y. Alexahin, E. Gianfelice-Wendt, "Determination of linear optics functions from turn-by-turn data", FERMILAB-PUB-06-093-AD, FNAL, 2006.
8. H. Grote and F. Iselin, "The MAD Program - User's Reference Manual", CERN/SL/90-13 (AP), Rev.5
9. F. Schmidt, "MAD-X PTC Integration", Proc. 2005 Part. Acc. Conf., pp. 1272.
10. F. Iselin, "The MAD Program – Physical Methods Manual", http://mad.home.cern.ch/mad/mad8web/mad8.html.

11. E. Forest, F. Schmidt and E. McIntosh, "Introduction to the Polymorphic Tracking Code", CERN-SL-2002-044-AP, KEK report 2002-3, July 2002.
12. "MAD-X User Guide", http://mad.home.cern.ch/mad/uguide.html
13. P. Skowronski, "PTC: new matching method and other improvements", http://frs.web.cern.ch/frs/MAD-X_minutes/Meeting-21/minutes.html.

Appendix

Connection between the measured Ripken parameters (derived from the TBT data) and eigenvectors in MADX code is described. Constraints for MADX-matching are built.

Tilted BPM monitors



The beam-position monitors (BPM) are supposed to be tilted by some angle. The tilt angle of monitor is the angle χ between the monitor abscissa and X-axis of the accelerator. For small $\chi << 1$, a beam position given by x, y in the "global" accelerator coordinate system is expressed by x_{BPM}, y_{BPM} in the (local) BPM's coordinate system by the following formulae.

$$\begin{cases} x_{\mathrm{BPM}} = x\cos\chi + y\sin\chi \\ y_{\mathrm{BPM}} = -x\sin\chi + y\cos\chi \end{cases} \xrightarrow{\chi<<1} \begin{cases} x_{\mathrm{BPM}} = x + \chi y \\ y_{\mathrm{BPM}} = -\chi x + y \end{cases}. \qquad \text{(A-1)}$$

Let's r_{HBPM} and r_{VBPM} are the calibration factor taking into consideration electronics errors for the horizontal (H) and vertical (V) BPMs, respectively. Then, the coordinates reported by horizontal and vertical BPMs are $x^{\mathrm{rep}}_{\mathrm{HBPM}} = x_{\mathrm{BPM}}/r_{\mathrm{HBPM}}$ and $y^{\mathrm{rep}}_{\mathrm{VBPM}} = y_{\mathrm{BPM}}/r_{\mathrm{VBPM}}$, respectively.

The measurements give the relative value of calibration factor $\delta r_{\mathrm{HBPM}} = \left(x^{\mathrm{rep}}_{\mathrm{HBPM}} - x_{\mathrm{HBPM}}\right)\big/ x_{\mathrm{HBPM}}$. The relation between absolute and relative calibration factors is $\delta r_{\mathrm{HBPM}} = 1/r_{\mathrm{HBPM}} - 1$. Therefore, the absolute calibration factor is expressed by

$$r_{\mathrm{HBPM}} = 1/\left(1 + \delta r_{\mathrm{HBPM}}\right). \tag{A-2}$$

The beam coordinates in the Ripken notation are

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \sqrt{\varepsilon_{\mathrm{I}}}\left(\sqrt{\beta_{x\mathrm{I}}}\cos\Phi_{x\mathrm{I}}\cos\phi_{\mathrm{I}} - \sqrt{\beta_{x\mathrm{I}}}\sin\Phi_{x\mathrm{I}}\sin\phi_{\mathrm{I}}\right) + \\ + \sqrt{\varepsilon_{\mathrm{II}}}\left(\sqrt{\beta_{x\mathrm{II}}}\cos\Phi_{x\mathrm{II}}\cos\phi_{\mathrm{II}} - \sqrt{\beta_{x\mathrm{II}}}\sin\Phi_{x\mathrm{II}}\sin\phi_{\mathrm{II}}\right) \\ \sqrt{\varepsilon_{\mathrm{I}}}\left(\sqrt{\beta_{y\mathrm{I}}}\cos\Phi_{y\mathrm{I}}\cos\phi_{\mathrm{I}} - \sqrt{\beta_{y\mathrm{I}}}\sin\Phi_{y\mathrm{I}}\sin\phi_{\mathrm{I}}\right) + \\ + \sqrt{\varepsilon_{\mathrm{II}}}\left(\sqrt{\beta_{y\mathrm{II}}}\cos\Phi_{y\mathrm{II}}\cos\phi_{\mathrm{II}} - \sqrt{\beta_{y\mathrm{II}}}\sin\Phi_{y\mathrm{II}}\sin\phi_{\mathrm{II}}\right) \end{pmatrix} \tag{A-3}$$

Let's use the notations for the 4-by-4 eigenvector-matrix

$$\begin{array}{llll} V_{11} = \sqrt{\beta_{x\mathrm{I}}}\cos\Phi_{x\mathrm{I}} & V_{12} = \sqrt{\beta_{x\mathrm{I}}}\sin\Phi_{x\mathrm{I}} & V_{13} = \sqrt{\beta_{x\mathrm{II}}}\cos\Phi_{x\mathrm{II}} & V_{14} = \sqrt{\beta_{x\mathrm{II}}}\sin\Phi_{x\mathrm{II}} \\ V_{31} = \sqrt{\beta_{y\mathrm{I}}}\cos\Phi_{y\mathrm{I}} & V_{32} = \sqrt{\beta_{y\mathrm{I}}}\sin\Phi_{y\mathrm{I}} & V_{33} = \sqrt{\beta_{y\mathrm{II}}}\cos\Phi_{y\mathrm{II}} & V_{34} = \sqrt{\beta_{y\mathrm{II}}}\sin\Phi_{y\mathrm{II}} \end{array}. \tag{A-4}$$

Then, one may rewrite (3)

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \sqrt{\varepsilon_{\mathrm{I}}}\left(V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}}\left(V_{13}\cos\phi_{\mathrm{II}} - V_{14}\sin\phi_{\mathrm{II}}\right) \\ \sqrt{\varepsilon_{\mathrm{I}}}\left(V_{31}\cos\phi_{\mathrm{I}} - V_{32}\sin\phi_{\mathrm{I}}\right) + \sqrt{\varepsilon_{\mathrm{II}}}\left(V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right) \end{pmatrix}. \tag{A-5}$$

The constants are $a'_{\mathrm{I}} = \varepsilon_{\mathrm{I}}^{1/2}\cos\phi_{\mathrm{I}}$, $a''_{x} = \varepsilon_{\mathrm{I}}^{1/2}\sin\phi_{\mathrm{I}}$, $a'_{y} = \varepsilon_{\mathrm{II}}^{1/2}\cos\phi_{\mathrm{II}}$, $a''_{y} = \varepsilon_{\mathrm{II}}^{1/2}\sin\phi_{\mathrm{II}}$, while

$\varepsilon_{\mathrm{I}}^{1/2} = \sqrt{a'^{\,2}_{x} + a''^{2}_{x}} \equiv a_x$ and $\varepsilon_{\mathrm{II}}^{1/2} = \sqrt{a'^{\,2}_{y} + a''^{2}_{y}} \equiv a_y$, and the coordinates are expressed as

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a_x\left(V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right) + a_y\left(V_{13}\cos\phi_{\mathrm{II}} - V_{14}\sin\phi_{\mathrm{II}}\right) \\ a_x\left(V_{31}\cos\phi_{\mathrm{I}} - V_{32}\sin\phi_{\mathrm{I}}\right) + a_y\left(V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right) \end{pmatrix} \tag{A-6}$$

Let's excite beam coherent oscillations by a single kick. In the presence coupling, the excitation of one of the two modes will excite an oscillation in the other mode too. The horizontal kick excites oscillations with $a_x \neq 0$ and $a_y = 0$, and the vertical kick excites oscillations with $a_x = 0$ and $a_y \neq 0$:

$$\left.\begin{pmatrix} x \\ y \end{pmatrix}\right|_{\mathrm{H-kick}} = \left.\begin{pmatrix} a_x\left(V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right) \\ a_x\left(V_{31}\cos\phi_{\mathrm{I}} - V_{32}\sin\phi_{\mathrm{I}}\right) \end{pmatrix}\right|_{\mathrm{H-kick}} \text{ and } \left.\begin{pmatrix} x \\ y \end{pmatrix}\right|_{\mathrm{V-kick}} = \left.\begin{pmatrix} a_y\left(V_{13}\cos\phi_{\mathrm{II}} - V_{14}\sin\phi_{\mathrm{II}}\right) \\ a_y\left(V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right) \end{pmatrix}\right|_{\mathrm{V-kick}}. \tag{A-7}$$

Horizontal BPMs

Two sets of data obtained by the horizontal BPM from the horizontal and vertical kicks:

$$x^{\mathrm{rep}}_{\mathrm{HBPM}} = \xrightarrow{x^{\mathrm{rep}}_{\mathrm{HBPM}} = \frac{x_{\mathrm{BPM}}}{r_{\mathrm{HBPM}}}} \left\{ \left.\frac{x_{\mathrm{BPM}}}{r_{\mathrm{HBPM}}}\right|_{\mathrm{H-kick}} ; \left.\frac{x_{\mathrm{BPM}}}{r_{\mathrm{HBPM}}}\right|_{\mathrm{V-kick}} \right\} = \xrightarrow{x_{\mathrm{BPM}} = x + \chi y} = \left\{ \left.\frac{x + \chi y}{r_{\mathrm{HBPM}}}\right|_{\mathrm{H-kick}} ; \left.\frac{x + \chi y}{r_{\mathrm{HBPM}}}\right|_{\mathrm{V-kick}} \right\} =$$

$$x^{\mathrm{rep}}_{\mathrm{HBPM}} = \left\{ \begin{array}{l} \left.\dfrac{a_x\left(V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right) + \chi a_x\left(V_{31}\cos\phi_{\mathrm{I}} - V_{32}\sin\phi_{\mathrm{I}}\right)}{r_{\mathrm{HBPM}}}\right|_{\mathrm{H-kick}} ; \\ \left.\dfrac{a_y\left(V_{13}\cos\phi_{\mathrm{II}} - V_{14}\sin\phi_{\mathrm{II}}\right) + \chi a_y\left(V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right)}{r_{\mathrm{HBPM}}}\right|_{\mathrm{V-kick}} \end{array} \right. = .$$

$$x^{\mathrm{rep}}_{\mathrm{HBPM}} = \left\{ \left.\frac{a_x\left[\left(V_{11} + \chi V_{31}\right)\cos\phi_{\mathrm{I}} - \left(V_{12} + \chi V_{32}\right)\sin\phi_{\mathrm{I}}\right]}{r_{\mathrm{HBPM}}}\right|_{\mathrm{H-kick}} ; \left.\frac{a_y\left[\left(V_{13} + \chi V_{33}\right)\cos\phi_{\mathrm{II}} - \left(V_{14} + \chi V_{34}\right)\sin\phi_{\mathrm{II}}\right]}{r_{\mathrm{HBPM}}}\right|_{\mathrm{V-kick}} \right\}$$

The coupling is assumed to be weak, that is the eigenvectors $V_{31}/V_{11} << 1$ and $V_{32}/V_{12} << 1$ are small values of the 1st order. With $\chi << 1$, one may neglect the small terms of the second order at H-kick:

$$x_{\mathrm{HBPM}}^{\mathrm{rep}} = \left\{ \left. \frac{a_x\left[V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right]}{r_{\mathrm{HBPM}}} \right|_{\mathrm{H-kick}} ; \quad \left. \frac{a_y\left[\left(V_{13}+\chi V_{33}\right)\cos\phi_{\mathrm{II}} - \left(V_{14}+\chi V_{34}\right)\sin\phi_{\mathrm{II}}\right]}{r_{\mathrm{HBPM}}} \right|_{\mathrm{V-kick}} \right\}.$$

Using different initial phases and defining the initial values during processing of TBT data [2,7], one may extract components of eigenvectors, and obtain the set of 4 eigenvectors:

$$\left\{ \underbrace{V_{11}^{\mathrm{HBPM}} = \frac{V_{11}}{r_{\mathrm{HBPM}}}; \quad V_{12}^{\mathrm{HBPM}} = \frac{V_{12}}{r_{\mathrm{HBPM}}}}_{\mathrm{H-kick}}; \quad \underbrace{V_{13}^{\mathrm{HBPM}} = \frac{V_{13}+\chi_{\mathrm{HBPM}}V_{33}}{r_{\mathrm{HBPM}}}; \quad V_{14}^{\mathrm{HBPM}} = \frac{V_{14}+\chi_{\mathrm{HBPM}}V_{34}}{r_{\mathrm{HBPM}}}}_{\mathrm{V-kick}} \right\}. \qquad \text{(A-8)}$$

Vertical BPMs

Two sets of data obtained by the vertical BPM from the horizontal and vertical kicks:

$$y_{\mathrm{VBPM}}^{\mathrm{rep}} = \xrightarrow{\substack{x_{\mathrm{VBPM}}^{\mathrm{rep}} = \frac{y_{\mathrm{BPM}}}{r_{\mathrm{VBPM}}}; \\ y_{\mathrm{BPM}} = -\chi x + y}} = \left\{ \left. \frac{-\chi x + y}{r_{\mathrm{VBPM}}} \right|_{\mathrm{H-kick}} ; \quad \left. \frac{-\chi x + y}{r_{\mathrm{VBPM}}} \right|_{\mathrm{V-kick}} \right\} =$$

$$y_{\mathrm{VBPM}}^{\mathrm{rep}} = \left\{ \begin{array}{l} \left. \dfrac{-\chi a_x\left(V_{11}\cos\phi_{\mathrm{I}} - V_{12}\sin\phi_{\mathrm{I}}\right) + a_x\left(V_{31}\cos\phi_{\mathrm{I}} - V_{32}\sin\phi_{\mathrm{I}}\right)}{r_{\mathrm{VBPM}}} \right|_{\mathrm{H-kick}} ; \\ \left. \dfrac{-\chi a_y\left(V_{13}\cos\phi_{\mathrm{II}} - V_{14}\sin\phi_{\mathrm{II}}\right) + a_y\left(V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right)}{r_{\mathrm{VBPM}}} \right|_{\mathrm{V-kick}} \end{array} \right. =$$

$$y_{\mathrm{VBPM}}^{\mathrm{rep}} = \left\{ \left. \frac{a_x\left[\left(V_{31}-\chi V_{11}\right)\cos\phi_{\mathrm{I}} - \left(V_{32}-\chi V_{12}\right)\sin\phi_{\mathrm{I}}\right]}{r_{\mathrm{VBPM}}} \right|_{\mathrm{H-kick}} ; \quad \left. \frac{a_y\left[\left(V_{33}-\chi V_{13}\right)\cos\phi_{\mathrm{II}} - \left(V_{34}-\chi V_{14}\right)\sin\phi_{\mathrm{II}}\right]}{r_{\mathrm{VBPM}}} \right|_{\mathrm{V-kick}} \right\}$$

The coupling is assumed to be weak, that is the eigenvectors $V_{13}/V_{33} << 1$ and $V_{14}/V_{34} << 1$ are small values of the 1st order. With $\chi << 1$, one may neglect the small terms of the second order at V-kick:

$$y_{\mathrm{VBPM}}^{\mathrm{rep}} = \left\{ \left. \frac{a_x\left[\left(V_{31}-\chi V_{11}\right)\cos\phi_{\mathrm{I}} - \left(V_{32}-\chi V_{12}\right)\sin\phi_{\mathrm{I}}\right]}{r_{\mathrm{VBPM}}} \right|_{\mathrm{H-kick}} ; \quad \left. \frac{a_y\left[V_{33}\cos\phi_{\mathrm{II}} - V_{34}\sin\phi_{\mathrm{II}}\right]}{r_{\mathrm{VBPM}}} \right|_{\mathrm{V-kick}} \right\}.$$

Using different initial phases and defining the initial values during processing of TBT data [A, G], one may extract components of eigenvectors, and obtain the set of 4 eigenvectors:

$$\left\{ \underbrace{V_{31}^{\mathrm{VBPM}} = \frac{V_{31} - \chi_{\mathrm{VBPM}} V_{11}}{r_{\mathrm{VBPM}}};\quad V_{32}^{\mathrm{VBPM}} = \frac{V_{32} - \chi_{\mathrm{VBPM}} V_{12}}{r_{\mathrm{VBPM}}}}_{\mathrm{H-kick}};\quad \underbrace{V_{33}^{\mathrm{VBPM}} = \frac{V_{33}}{r_{\mathrm{VBPM}}};\quad V_{34}^{\mathrm{VBPM}} = \frac{V_{34}}{r_{\mathrm{VBPM}}}}_{\mathrm{V-kick}} \right\}. \qquad \text{(A-9)}$$

Building of Constraints

One may use two pairs of VARY parameters r_{HBPM} and χ_{HBPM}, and r_{VBPM} and χ_{VBPM} with the following 8 constraints:

$$\overbrace{\begin{cases} r_{\mathrm{HBPM}} V_{11}^{\mathrm{HBPM}} = V_{11} \\ r_{\mathrm{HBPM}} V_{12}^{\mathrm{HBPM}} = V_{12} \end{cases}}^{\mathrm{H-kick\ \&\ HBPM}}, \quad \overbrace{\begin{cases} V_{13}^{\mathrm{HBPM}} r_{\mathrm{HBPM}} = V_{13} + \chi_{\mathrm{HBPM}} V_{33} \\ V_{14}^{\mathrm{HBPM}} r_{\mathrm{HBPM}} = V_{14} + \chi_{\mathrm{HBPM}} V_{34} \end{cases}}^{\mathrm{V\text{-}kick\ \&\ HBPM}},$$

$$\overbrace{\begin{cases} V_{31}^{\mathrm{VBPM}} r_{\mathrm{VBPM}} = V_{31} - \chi_{\mathrm{VBPM}} V_{11} \\ V_{32}^{\mathrm{VBPM}} r_{\mathrm{VBPM}} = V_{32} - \chi_{\mathrm{VBPM}} V_{12} \end{cases}}^{\mathrm{H\text{-}kick\ \&\ VBPM}} \quad \overbrace{\begin{cases} r_{\mathrm{VBPM}} V_{33}^{\mathrm{VBPM}} = V_{33} \\ r_{\mathrm{VBPM}} V_{34}^{\mathrm{VBPM}} = V_{34} \end{cases}}^{\mathrm{V\text{-}kick\ \&\ VBPM}} \qquad .(10)$$

A manual writing of MAD-X input-script realizing the above method is very tedious for a large machine. For example, a plain input-script for TEVATRON contains 3700 lines (the file-size is 220kb). Additional input scripts realizing the described matching have a total size up to 1 Mb. The code for creating MAD-X input-scripts has been written in FORTRAN and applied for TEVATRON lattice at injection. It allows different combinations of constraints and variable parameters. The table with the code switches managing types of constraints and example of input-file are given below.

Table. ON-status of constraints and parameters vs the code switches.

Constraint or variable	ON	HBPM_OFF =.FALSE.	H_Kick_OFF =.FALSE.	VBPM_OFF= .FALSE.	V_Kick_OFF =.FALSE.
$\overbrace{\begin{cases} r_{\mathrm{HBPM}} V_{11}^{\mathrm{HBPM}} = V_{11} \\ r_{\mathrm{HBPM}} V_{12}^{\mathrm{HBPM}} = V_{12} \end{cases}}^{\text{H-kick \& HBPM}}$	+	+	+		
$\overbrace{\begin{cases} V_{13}^{\mathrm{HBPM}} r_{\mathrm{HBPM}} = V_{13} + \chi_{\mathrm{HBPM}} V_{33} \\ V_{14}^{\mathrm{HBPM}} r_{\mathrm{HBPM}} = V_{14} + \chi_{\mathrm{HBPM}} V_{34} \end{cases}}^{\text{V-kick \& HBPM}}$	+	+			+
$\overbrace{\begin{cases} V_{31}^{\mathrm{VBPM}} r_{\mathrm{VBPM}} = V_{31} - \chi_{\mathrm{VBPM}} V_{11} \\ V_{32}^{\mathrm{VBPM}} r_{\mathrm{VBPM}} = V_{32} - \chi_{\mathrm{VBPM}} V_{12} \end{cases}}^{\text{H-kick \& VBPM}}$	+		+	+	
$\overbrace{\begin{cases} r_{\mathrm{VBPM}} V_{33}^{\mathrm{VBPM}} = V_{33} \\ r_{\mathrm{VBPM}} V_{34}^{\mathrm{VBPM}} = V_{34} \end{cases}}^{\text{V-kick \& VBPM}}$	+			+	+
r_{HBPM}	+	+			
r_{VBPM}	+			+	
χ_{HBPM}	+	+			
χ_{VBPM}	+			+	

Example of input file for TEVATRON at injection is given below:

```
&NML_UNIT_for_output ! Default value .TRUE. (to terminal)
    Terminal_output =.False.
&END

&NML_MADX_exe_filename
  MADX_exe_filename =  'madxp_cwg95_20060913.exe'
&END

&NML_data_directory
  data_dir_prefix='../MADX_TBT_data/' ! default directory for data
&END
```

```
&NML_madx_TBT_debug
  madx_TBT_debug = .TRUE.
$END

&NML_FILENAME_MADX_LINE
! File with only elements and beam lines without action conmmands
  Filename_madX_Line = 'tev_inj_090806_vvk.madx'
&END

&NML_FILENAME_MADX_LINE_corrections ! File with corrections to the beam-line,
e.g.
                                    ! new shifted LINE started from BPMf18 or
                                    ! new values for physical elements
   Filename_madX_Line_corrections = 'line_shifted_tev_inj_090806_vvk.madx'
&END

&NML_MADX_BEAM_command !beam, particle=proton, energy:= 150; ! INJECTION
   Particle_MADX='proton'
   Energy_MADX_GEV=150
&END

&NML_MADX_USE_command ! USE, PERIOD=tevb0_shifted;
   LineName_USE_MADX='tevb0_shifted' ! Name of beamline to USEd
&END

&NML_FILENAMES_TBT_data
  Filename_HBPM_TBT_DATA = "tbt_data_hbpm_tev_inj_090806.dat"
  Filename_VBPM_TBT_DATA = "tbt_data_vbpm_tev_inj_090806.dat"
&END

&NML_Number_of_BPMs
  Number_HBPM_TBT = 118, Number_VBPM_TBT= 118
&END

&NML_HBPM_2_signatures
  HBPM_signature(1)='HBPM', HBPM_signature(2)='HMONITOR'
&END

&NML_VBPM_2_signatures
  VBPM_signature(1)='VBPM', VBPM_signature(2)='VMONITOR'
&END

&NML_Multipole_Norm_K1_2_signatures
  Multipole_Norm_K1_signature(1)='mqsf',
Multipole_Norm_K1_signature(2)='MULTIPOLE'
&END

&NML_Multipole_Skew_K1_2_signatures
  Multipole_Skew_K1_signature(1)='mskw',
Multipole_Skew_K1_signature(2)='MULTIPOLE'
&END

&NML_Constraints_type
   HBPM_OFF=.FALSE., VBPM_OFF=.TRUE.      ! defaults  HBPM_OFF=.FALSE.,
VBPM_OFF=.FALSE.
   H_kick_OFF=.TRUE., V_kick_OFF=.FALSE.  ! H_kick_OFF=.FALSE.,
V_kick_OFF=.TRUE.
```

```
!                          HBPM_OFF   VBPM_OFF   H_kick_OFF   V_kick_OFF
! 1. H-Kick
! 1.a  H-optics :            .FALSE.    .TRUE.      .FALSE.      .TRUE.
! 1.b  H-coupling:           .TRUE.     .FALSE.     .FALSE.      .TRUE.
! 1.c  H-optics & coupling   .FALSE.    .FALSE.     .FALSE.      .TRUE.
!
! 2. V-Kick
! 2.a  V-optics :            .TRUE.     .FALSE.     .TRUE.      .FALSE.
! 2.b  V-coupling:           .FALSE.    .TRUE.      .TRUE.      .FALSE.
! 2.c  V-optics & coupling   .FALSE.    .FALSE.     .TRUE.      .FALSE.
!
! 3 V- & H- kiks             .FALSE.    .FALSE.     .FALSE.     .FALSE.
&END

&NML_Tune_correctors
  TYPE_for_delta_K1_FOC='FOC', delta_K1_FOC=0.0, & ! delta_K1_FOC=0.0001
  TYPE_for_delta_K1_DEF='DEF',  delta_K1_DEF=0.0  ! delta_K1_DEF=-0.0001
&END

&NML_VARY_LIMITS_r
  !  VARY, r, LOWER=r_ini*(1-r_rel)-r_abs, UPPER=r_ini*(1+r_rel)+r_abs
    Use_limits_r=.TRUE.,   r_ini=1,   r_abs=0.01, r_rel=0.01 ! Calibration
factor
&END
&NML_VARY_STEP_r
  USE_VARY_STEP_r=.TRUE., VARY_STEP_r=1.0E-04
&END

&NML_VARY_LIMITS_chi
  Use_limits_chi=.TRUE., chi_ini=0, chi_abs=0.01, chi_rel=0.01 ! Tilt angle
(rad)
&END
&NML_VARY_STEP_chi
  USE_VARY_STEP_chi=.TRUE., VARY_STEP_chi=1.0E-04
&END

&NML_VARY_LIMITS_K1
  Use_limits_k1=.FALSE.,  k1_ini=0, k1_abs=0.01, k1_rel=0.01 ! Multipole
strength K1 (quadrupole)
&END
&NML_VARY_STEP_K1
  USE_K1_VARY_STEP=.TRUE., K1_VARY_STEP=1.0E-05
&END

&NML_VARY_OFF
   VARY_K1_OFF=.FALSE.,  VARY_r_OFF=.TRUE., VARY_chi_OFF=.TRUE.
! defaults    VARY_K1_OFF=.FALSE.,  VARY_r_OFF=.TRUE., VARY_chi_OFF=.TRUE.
&END

&NML_TYPE_Multipole_K1_OFF !
   ! Max Number of TYPES=1000, the string Length <= 20
    TYPE_Multipole_K1_OFF(1)='FOC'
    TYPE_Multipole_K1_OFF(2)='DEF'
    !TYPE_Multipole_K1_OFF(3)='SKW'
&END
```