SLAC-PUB-9446
August 2002

ACCELERATOR MODELING WITH MATLAB ACCELERATOR TOOLBOX

A. Terebilo, SLAC, Menlo Park, CA 94025, USA

Abstract

This paper introduces Accelerator Toolbox (AT) - a collection of tools to model storage rings and beam transport lines in the MATLAB environment. The objective is to illustrate the flexibility and efficiency of the AT–MATLAB framework. The paper discusses three examples of problems that are analyzed frequently in connection with ring-based synchrotron light sources.

1 BACKGROUND

AT is being developed to support the ongoing design and future operations of the SPERA3 light source [1]. Unlike most existing accelerator codes, AT is not a standalone program or a class library, but a MATLAB toolbox. The user accesses it from within MATLAB.

AT is a collection of functions and scripts that:

- Create and manipulate accelerator data structures in the MATLAB workspace (lattice tools)
- Simulate particle motion through elements and sequences of elements (low-level physics tools)
- Compute accelerator parameters and beam properties (high-level physics tools)

This approach has a number of benefits for the end user and developer.

- AT tools take advantage of a large library of math functions that are part of MATLAB or its toolboxes such as matrix algebra, FFT, optimization and control.
- The users can add new physics tools to AT with *minimum* programming in MATLAB scripting language.
- Results of all calculations are immediately available for further analysis and visualization using MATLAB graphics.

A detailed introduction and tutorial on AT can be found in [2]. Current version 1.1 for Windows and Linux is available for download from [3].

2 ACCELERATOR MODELING

Examples in this section illustrate the use of several high-level physics functions recently added to AT. Scripts, that demonstrate the use of these functions and produce the plots in this paper, are included with AT distribution in the `ATROOT/atdemos` directory.

2.1 Linear Analysis of Coupled Lattices

The formalism developed in [4] leads to a full-turn transfer matrix in the form:

$$T = \begin{bmatrix} M & m \\ n & N \end{bmatrix} = VUV^{-1} \qquad (1)$$

$$U = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix} \qquad V = \begin{bmatrix} \gamma I & C \\ -C^{+} & \gamma I \end{bmatrix} \qquad (2)$$

where $^{+}$ denotes the symplectic conjugate of a matrix.

Symplectic matrix V defines the transformation to the normal mode basis. Matrixes A and B should be interpreted as the Twiss matrixes of the normal modes. All matrices in (1) and (2) are functions of the ring azimuth s. They characterize *local* coupling.

The final expressions for A, B, and C in terms of M, N, m, and n at one location in the ring, require about 60 scalar and 2-by-2 matrix operations such as sum, product, transpose, symplectic conjugate, and determinant calculation. About 20 operations are required for each additional point around the ring.

In AT, one fuction `findm44` generates transfer matrixes at specified locations around the ring. Another function `linopt` uses `findm44` and calculates all matrix elements in (2). In addition it can calculate other linear optics parameters, such as normal mode tunes.

`linoptdemo` is a script in `ATROOT/atdemos` directory which shows the use of `linopt` for the SPEAR storage ring. It perturbs the uncoupled model by introducing random rotations around the s-axis in some of the quadrupole magnets. Figures 1,2 show the elements of matrix C and mixing parameter γ.

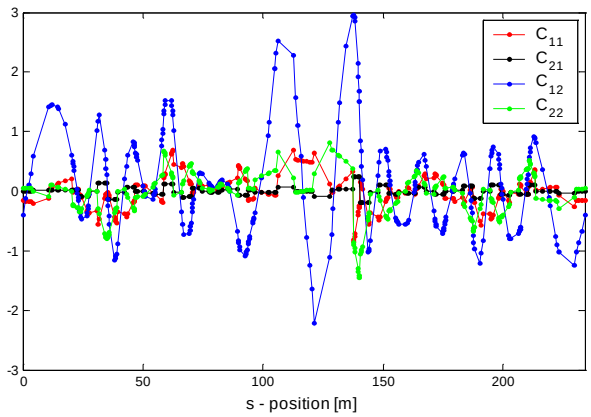

Figure 1: Elements of coupling matrix C_{ij}.

Work supported by the Department of Energy contract DE-AC03-76SF00515.

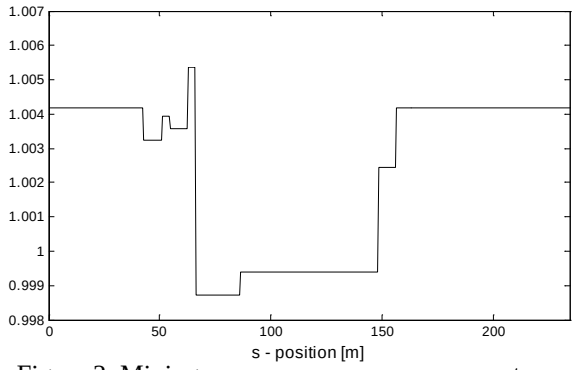

Figure 2: Mixing parameter γ.

2.2 Beam Envelope with Linear Coupling

Calculation of the equilibrium beam envelope in the presence of linear, but not necessarily weak, coupling is a typical problem in storage ring physics. The implementation of the formalism [5] is another instructive example of how AT utilizes the matrix capabilities of MATLAB. The formalism assumes a Gaussian beam distribution near the closed orbit:

$$\psi(x)=\frac{1}{(2\pi)^3\sqrt{\det(R)}}\exp\left\{-\frac{1}{2}R^{-1}{}_{i,j}x_i x_j\right\} \quad (3)$$

$$R_{i,j}=\left\langle x_i x_j\right\rangle$$

Matrix R propagates between s_0 and s according to (4).

$$R(s)=M(s,s_0)R(s_0)M^T(s,s_0)+\overline{B}(s,s_0), \quad (4)$$

where $M(s, s_0)$ is the 6-by-6 transfer matrix between s and s_0, near the closed orbit. $\overline{B}(s, s_0)$ is the cumulative diffusion matrix accumulated between s and s_0.

$$\overline{B}(s,s_0)=\int_{s_0}^{s} M(s,s')B(s')M^T(s,s')ds' \quad (5)$$

The local diffusion matrix $B(s)$ depends on the local properties of magnetic field and the closed orbit. In the case of a storage ring, given the transfer matrix M_i and the cumulative diffusion B_i between the entrance and exit of the i-th element, the equilibrium condition for R_1 at the first element can be obtained as:

$$\begin{aligned}
R_2 &= M_1R_1M_1{}^T+\overline{B}_1\\
R_3 &= M_2R_2M_2{}^T+\overline{B}_2=\\
&= M_2M_1R_1(M_2M_1)^T+M_2\overline{B}_1M_2{}^T+\overline{B}_2\\
&= M_{1,2}R_1M_{1,2}{}^T+\overline{B}_{1,2}\\
&\ldots\\
R_{N+1} &= R_1=M_{1\ldots N}R_1M_{1\ldots N}{}^T+\overline{B}_{1..N}
\end{aligned} \quad (6)$$

AT implements this algorithm in the high-level physics function `ohmienvelope`, which calls three other AT functions to find all Ms and Bs in (6):

`findorbit6` computes the closed orbit with classical treatment of radiation at each element.

`findmpoleraddiffmatrix` computes the radiation diffusion matrix for an element, whose transverse magnetic field can be written as a multipole expansion.

`findelemm66` computes the 6-by-6 transfer matrix through the element near the closed orbit.

In MATLAB language, it takes only a few lines of code to assemble the last equations in (6).

$$R=M_{Ring}RM_{Ring}+\overline{B}_{Ring} \quad (7)$$

Notice that we can write it as Lyapunov matrix equation commonly used in linear control theory.

$$AR+RB=-C \quad (8)$$

Our `ohmienvelope` uses MATLAB control toolbox function `lyap` to solve it. `ohmienvelope` also calculates the beam ellipse size and the orientation in the x-y plane. Figure 3 shows the result of such a calculation for the SPEAR ring with a few randomly tilted quadrupoles. This plot is produced by `ohmienvelopedemo` in `AROOT/atdemos` subdirectory.

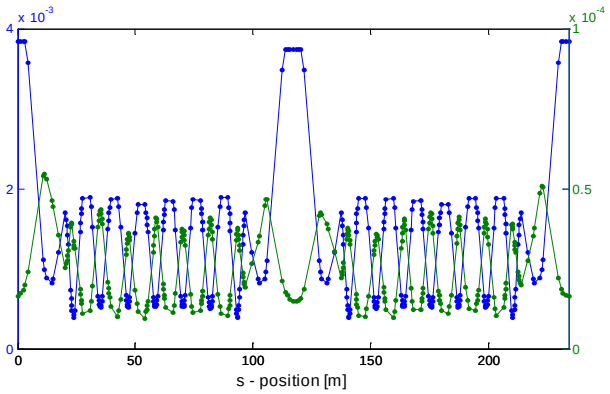

Figure 3: Beam sizes σ [m]

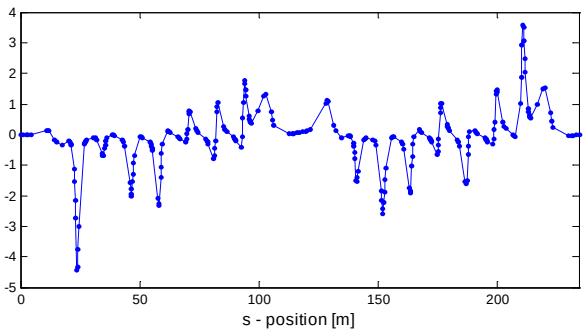

Figure 4: Tilt angle of the beam ellipsoid [degrees].

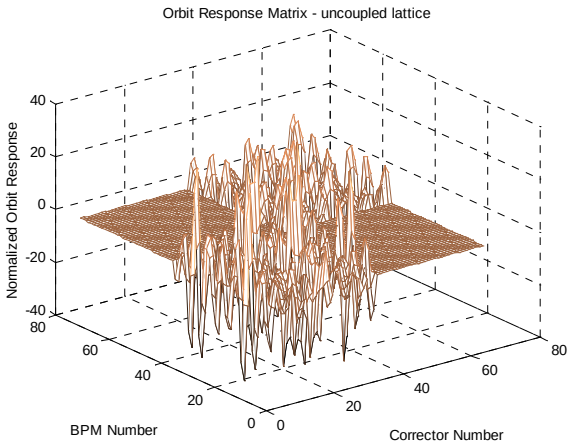

Figure 5: Orbit response matrix (uncoupled lattice) generated with `findrespm`

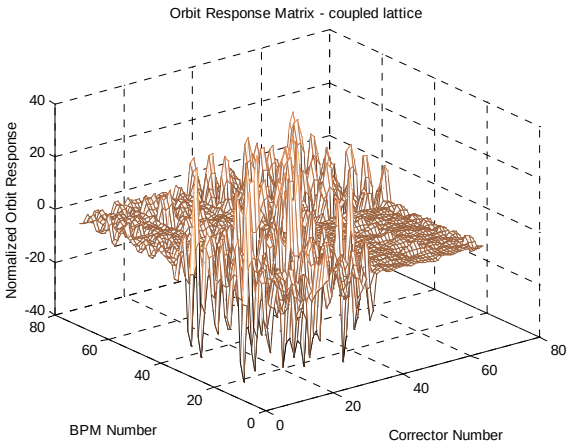

Figure 6: Orbit response matrix (coupled lattice) generated with `findrespm`

2.3 Orbit Response Matrix

Orbit response matrix R_{ij} measures the change in the transverse orbit position at some location s_i, caused by a transverse kick (typically, with a corrector magnet) at some other location s_j. The response matrix can be measured in a real accelerator or computed with an accelerator code.

A popular technique [6] for linear optics determination fits the parameters in the model, such as the K-values of quadrupoles, corrector gains, and BPM errors, to minimize the difference between the measured and the model response matrix. Easy to use matrix tools, namely `svd`, make MATLAB a good candidate for the fitting part. The numerical fitting procedure needs the *model* response matrix and its derivatives with respect to the fitted parameters. For this purpose it is convenient to use the AT function `findrespm`.

Demo script `findrespmdemo` generates the model response matrixes for the SPEAR lattice without coupling (Figure 5) and with coupling (Figure 6) caused by random tilts in one of the quadrupole families. Response matrixes are immediately available in MATLAB for visualization and for the model parameter fitting.

3 ACNOWLEDGEMENTS

The author would like to thank J. Corbett, SLAC, for the helpful feedback that results from his work on AT and MATLAB applications for SPEAR 3 light source [7].

4 REFERENCES

[1] J. Corbett, et al., “SPEAR 3 Upgrade Project: A Status Report”, these proceedings.

[2] A. Terebilo, “Accelerator Toolbox for MATLAB”, SLAC-PUB-8732, May 2001.

[3] http://www-ssrl.slac.stanford.edu/at/

[4] D. Sagan, D. Rubin “Linear Analysis of Coupled Lattices”, Phys.Rev.Spec.Top. - Accelerators and Beams, vol.2, 1999.

[5] K. Ohmi, et al., “From the Beam-Envelope Matrix to Synchrotron-Radiation Integrals”, Phys.Rev.E.,vol. 49, January 1994.

[6] J. Safranek, Nucl. Instr. and Methods, A388, p.27, 1997.

[7] J. Corbett, A. Terebilo, “Intereactive Orbit Control in MATLAB”, these proceedings.