RELEASE OF MARYLIE 3.0*

A. Dragt, F. Neri, J. van Zeijts, and G. Rangarajan
Department of Physics, University of Maryland, College Park, MD 20742

Introduction

The code MARYLIE 3.0, A FORTRAN Program for Charged Particle Beam Transport Based on Lie Algebraic Methods, is expected to be available for general use in the near future.[1] MARYLIE 3.0 is a program for the computation, manipulation, analysis, and application (ray tracing) of transfer maps. It may be used both for beam transport systems and for circular machines. The full 6-dimensional phase space is treated exactly (including fringe field and small machine effects) through third (octupole) order. When used for ray tracing, it can be orders of magnitude faster than kick codes. When used for analysis, it can compute all third order effects including aberrations, first and second order chromaticities, anharmonicities, distortion functions, nonlinear invariants, nonlinear resonance widths, normal forms, and KAM tori. It has extensive fitting and optimization capabilities. There are many other features such as the ability to transform particle distributions, transport moments, compute emittance invariants, and find dynamic apertures. Over 100 built in beamline elements and commands are available plus the ability to add user specified elements and commands.[2]

MARYLIE 3.1, a code having the same capabilities as MARYLIE 3.0 plus the ability to treat alignment, placement, and powering errors, is under development. Work is also being done on MARYLIE 5.1, a code that includes all effects through fifth (12 pole) order.[3]

Program Organization

MARYLIE is structured to facilitate the input and analysis of both small and very large lattices. It is controlled by a master input file organized into seven components. These components and their purposes are summarized below:

Component	Purpose
#comments	Allows user to write comments about lattice under study, calculations to be performed, etc.
#beam	Specifies magnetic rigidity and relativistic γ factor of the beam. Also specifies particle charge and the scale length to be used.
#menu	Contains a list of user specified beamline elements and commands.
#lines	Contains a list of user specified names for collections of elements and/or commands drawn from the menu. Permits the construction of macros.
#lumps	Contains a user specified list of collections of items from the menu and/or lines that are to be combined together to form individual transfer maps.
#loops	Contains user specified list of collections of elements, lines, and lumps to be used for extensive tracking calculations.
#labor	Specifies a lattice or beamline, and the actual operations and calculations to be performed. MARYLIE may be viewed as a kind of programing language; the #labor component contains the program to be executed. This is done by invoking user specified names drawn from the #menu, #lines, #lumps, and #loops components of the master input file.

Experience has shown that the above organization is both flexible and efficient. For example, even lattices as large as the proposed Superconducting Super Collider can be completely specified in relatively few lines. Also, very elaborate calculations and fitting operations can be performed with a minimum of effort.

Elements Treated

As indicated above, the #menu component of the master MARYLIE input file contains a list of user specified beamline elements and commands. Each element or command is given a user specified name, and is identified by a type code mnemonic. These user names may then be invoked in other components of the master input file. Shown below are the currently available beamline elements and their type code mnemonics:

Type Code	Element
drft	Drift Space.
	Dipole Bend Magnets:
nbnd	a) Normal Entry Bending Magnet, with or without Fringe Fields.
pbnd	b) Parallel Faced Bending Magnet, with Fringe Fields and equal entry and exit angles.
gbnd	c) General Bending Magnet.
prot	d) Used for Leading and Trailing Pole Face Rotations.
gbdy	e) Used for the Body of a General Bending Magnet.
frng	f) Used for Hard-Edge Dipole Fringe Fields.
cfbd	g) Combined Function Bend.
sol	Solenoid.
quad	Magnetic Quadrupole.
cfqd	Combined Function Magnetic Quadrupole.
recn	Rare Earth Cobalt Quadrupole Multiplet.
sext	Magnetic Sextupole.
octm	Magnetic Octupole.
octe	Electric Octupole.
srfc	Short RF Cavity.
arot	Axial Rotation.
thℓm	"Thin lens" approximation to low order multipoles.
cpℓm	"Compressed" approximation to low order multipoles.
twsm	Linear matrix transformation specified in terms of twiss parameters.

* Supported in part by the Department of Energy Contract No. DE-AS05-80ER10666.

Procedures and Fitting and Optimization Commands

Finally, the #menu component of the master input file may contain procedures and fitting and optimization commands. This feature makes it possible to set up fitting, optimization, and other computational loops within the #labor component of the master input file. For example, it is possible to vary any selected set of the parameters defining a beam line or ring in such a way as to fit or optimize any set of the quantities (both linear and nonlinear) that can be computed by MARYLIE. Thus it is possible to fit tunes, chromaticities, anharmonicities, linear and nonlinear lattice functions, linear and nonlinear parts of transfer maps, aberrations, moments, etc. Optimization can be performed employing least square and user written merit functions. Computational loops can be set up to find a wide array of quantities ranging from the dynamic aperture of a circular machine to the pinhole pattern of a charged particle optical system. Shown below are the currently available procedures and fitting and optimization commands:

Type Code	Procedure/Command
bip	Begin inner procedure.
bop	Begin outer procedure.
tip	Terminate inner procedure.
top	Terminate outer procedure.
aim	Specifiy quantities to be fit or optimized and set target values.
vary	Specify quantities to be varied.
fit	Carry out fitting operation.
mrt0	Merit Function (Least Squares).
mrtj	Merit Functions (User Written).
opt	Carry out optimization.
conj	Constraints.
cpsj	Capture parameter set j.
fps	Free parameter set.
dapt	Compute dynamic aperture.

Program Performance

MARYLIE may be used for particle tracking around or through a lattice and for analysis of linear and nonlinear lattice properties. When used for tracking, it is both versatile and extremely fast. Tracking can be performed element to element, lump to lump, or any mixture of the two. The speed for element to element tracking is equivalent to that of other tracking codes. When collections of elements can be lumped together to form single transfer maps, tracking speeds can be orders of magnitude faster. For example, experience indicates that the proposed Superconducting Super Collider (SSC) lattices can be treated to high accuracy by using 18 lumps. When so treated, MARYLIE 3.0 can track an SSC lattice for 50,000 turns of full 6-dimensional phase-space motion (including synchrotron oscillations) using less than 6 minutes of CRAY X-MP computer time. As a second example, MARYLIE can track up to 10,000 particles simultaneously. It can be used in this way to compute quantities ranging from simulated beam position monitor readings for multiple turns in a ring to spot patterns for a charged particle optical system.

In applications involving fitting, optimization, and design, MARYLIE can fit or optimize up to 20 quantities simultaneously. These quantities can be selected from essentially any of the quantities computable by MARYLIE. Applications range from setting linear and nonlinear ring parameters, to correcting telescope aberrations, to designing third-order achromats.[4]

Auxiliary Programs

MARYLIE has been written to make use of several external files both for input and output. This feature makes it possible to design and use other computer programs both to prepare input for MARYLIE and to postprocess MARYLIE output. Currently several such programs exist or are in preparation. A brief list of some of these programs is given below:

PREP. This program runs interactively, and may be used to prepare the Master Input File for MARYLIE.

GENMAP. In normal operation, MARYLIE uses an internal library of idealized beamline elements. However, MARYLIE also has the capability to read in and use externally specified transfer maps. GENMAP refers to a series of programs that take as input real (calculated or measured) magnetic and/or electric field data for some beam-line element, and then generate the transfer map describing motion through that element. This transfer map can subsequently be used by MARYLIE. In this way, it is possible to use MARYLIE to treat real beamline elements including the effect of fringe fields, magnetic imperfections, etc.

DIST. MARYLIE traces rays by reading phase-space initial conditions from an external file, applying a transfer map or series of transfer maps to these initial conditions, and writing the final results in an external final condition file. DIST is a program for generating various initial phase-space distributions.

PLOT. When performing ray traces, MARYLIE reads phase-space initial conditions from an external file and writes the final phase-space coordinates on another external file. PLOT is a plotting program that can be used to make phase-space plots of both initial and final condition files. It can also be used to plot various other kinds of MARYLIE output.

References

1. For a review of Lie algebraic methods, see Dragt, A. et al, Ann. Rev. Nucl. Part. Sci. 38, p. 455 (1988). See also Dragt, A., Lectures on Nonlinear Orbit Dynamics, Am. Inst. Phys. Conf. Proc. 87, ed. R. Carrigan et al, New York AIP (1982).

2. For a more complete description of MARYLIE, see Dragt, A. et al, draft MARYLIE 3.0 User's Manual, University of Maryland Physics Department Report (1988).

3. For a description of the Lie algebraic treatment of errors, see Healy, L. and A. Dragt, Concatenation of Lie Algebraic Maps, to appear in Proceedings of Second Workshop on Lie Optics, Cocyoc Mexico, Springer-Verlag (1989). See also Healy, L., University of Maryland Physics Department Ph.D. thesis (1986).

4. For the theory of third-order achromats, see Dragt, A., Nucl. Instrum. Methods A258, p. 339 (1987).

iftm	Linear matrix transformation specified in terms of initial-final twiss parameters.
jmap	Map with matrix part J.
mark	Marker.
dp	Data Point.
usrj	User Specified Subroutines that act on phase-space data, or produce or act on maps.
r****	Random counterpart of the element with type-code mnemonic ****.

Simple Commands

The #menu component of the master input file may also contain simple commands. These commands can be invoked elsewhere to manipulate maps and perform various ray trace and tracking operations. Commands are also given user specified names, and are also identified by type code mnemonics. Shown below are the currently available simple commands and their type codes:

Type Code	Command
end	Halt execution. Must be last element of a labor list.
of	Open files.
cf	Close files.
rt	Perform a ray trace.
num	Number lines in a file.
circ	Set parameters and circulate.
wcℓ	Write contents of a loop.
rapt	Aperture the beam with a rectangular aperture.
eapt	Aperture the beam with an elliptic aperture.
swnd	Static window a beam.
dwnd	Dynamic window a beam.
whst	Write history of beam loss.
pmif	Print contents of Master Input File.
ptm	Print transfer map.
tmi	Input matrix elements and polynomial coefficients for a map from an external file.
tmo	Output matrix elements and polynomial coefficients for a map to an external file.
psj	Parameter set j specification.
rpsj	Random parameter set j specification.
wps	Write out the parameters in a parameter set.
stm	Store the existing transfer map.
gtm	Get transfer map from storage.
mask	Mask off specified portions of existing transfer map.
ftm	Filter the existing transfer map.
sqr	Square the existing transfer map.
symp	Symplectify matrix in transfer map.
iden	Replace existing transfer map by the identity map.
inv	Replace existing transfer map by its inverse.
rev	Replace existing transfer map by its reversed map.
revf	Replace existing transfer map by reverse factorized form.
tran	Replace existing transfer map by its "transpose".
time	Write out execution time.
cdf	Change drop file.
bell	Ring bell at terminal.
wmrt	Write out value of merit function.
paws	Pause.

Advanced Commands

The #menu component of the master input file may also contain advanced commands. These commands can also be invoked to analyze maps and carry out various complicated operations. Shown below are the currently available advanced commands and their type codes:

Type Code	Command
cod	Compute off-momentum closed orbit data.
tasm	Twiss analyze static map.
tadm	Twiss analyze dynamic map.
ctr	Change tune range.
snor	Static normal form analysis.
dnor	Dynamic normal form analysis.
asni	Apply script N inverse.
rasm	Resonance analyze static map.
radm	Resonance analyze dynamic map.
sia	Static invariant analysis.
dia	Dynamic invariant analysis.
psnf	Compute power of static normal form.
pdnf	Compute power of dynamic normal form.
pnℓp	Compute power of nonlinear part.
trsa	Transport Static script A.
trda	Transport Dynamic script A.
fasm	Fourier analyze static map.
fadm	Fourier analyze dynamic map.
pold	Polar decomposition of a map.
tbas	Translate basis.
exp	Compute exponential.
gbuf	Get buffer contents.
amap	Apply map to a function or moments.
smul	Multiply a polynomial by a scalar.
padd	Add two polynomials.
pmul	Multiply two polynomials.
pb	Poisson bracket two polynomials.
pval	Evaluate a polynomial.
sq	Select Quantities.
wsq	Write Selected Quantities.