Update on Accelerator Toolbox Code and Computations for the ESRF EBS

2 December 2016,
LER design workshop
LUND
B. Nash
ESRF
OUTLINE

• History of AT, atcollab

• Structure of AT

• Integrators

• Application to ESRF EBS

• Conclusions
WIDELY USED CODES FOR ELECTRON RINGS

Elegant
APS-Borland et. al.

MAD-X/PTC
CERN- module structure with organizers

BMAD
Cornell- D. Sagan
http://www.lepp.cornell.edu/~dcs/bmad/

(OPA, Tracy, …?)
HISTORY OF AT

AT is a Matlab based tool box for accelerator simulation originally created by Andrei Terebilo during the late 1990’s at SLAC. He left acc. phys. 2010

Some interaction with Tracy code (Nishimura et al) at early stage
http://inspirehep.net/record/264780?ln=en

AT is used within MML and LOCO (Portmann, Safranek), but may also be used as a stand alone modeling code.

No one was coordinating the AT development, after Andrei stopped,

After I joined ESRF in 2009, I decided to try to coordinate AT development in an open source collaboration.

atcollab
How to organize AT development across multiple labs, when it is already in use by so many?

Can we support an open source collaborative development for the AT code?

Needed parts of open source project:

*) Repository in version control system: SourceForge chosen with svn
*) communication amongst users and developers: (mailing list, wiki, website…)
*) documentation, bug reporting, etc
*) well defined releases

http://sourceforge.net/projects/atcollab/

considerations:
> copyright issues? (I asked A. Terebillo and he suggested we go ahead)
> branching and forking of code?
> How to find common ground and bring different development together?
mailing list: atcollab-general@lists.sourceforge.net

27 members from 15 institutes

ESRF (5)
NSRC, Taiwan (3)
LBL (3)
IHEP, China (3)
ALBA (3)
SLAC
Australian Synchrotron
Mayo Clinic
INFIN
Diamond
KIT, ANKA
SOLEIL
ELETTRA
Canadian Light Source
LAL

on Twitter:
@atcollab_tweets

Archives and subscription info at
https://lists.sourceforge.net/lists/listinfo/atcollab-general
http://sourceforge.net/projects/atcollab/

ATCOLLAB SVN REPOSITORY

Having repository online facilitates email discussion. Can refer to code
with URL. e.g. ringpass function is here:
http://sourceforge.net/p/atcollab/code-0/HEAD/tree/trunk/atmat/attract/ringpass.m

get copy of repository with:
svn checkout https://svn.code.sf.net/p/atcollab/code-0/trunk atcollab

AT is also distributed with MML by Greg Portmann.
MiddleLayer/Release/at

If possible, we should bring these versions together!
Backwards compatibility. Previous lattice construction methods (using global variables) and integrators written with AT1.3 should still work.

Some tests with MML integration show no issues. More tests are needed, and feedback greatly appreciated!
(Any problems with LOCO?)

Compilation should be simply done by calling `atmexall`. We test this on as many different versions of Matlab, and as many platforms (Linux, Windows, Mac) as we can.

Tools that are working but not fully integrated or adequately tested go into the `pubtools` directory. After further testing, we migrate these tools into the other main AT directories.
Issue: how to write functions that identify particular elements that will work for lattices at different labs?
Solution: add a “Class” field to all elements

Example: indsext=findcells('ring','Class','Sextupole')
gives indices of all sextupoles
• AT has the following integrators in the atintegrators directory:

AperturePass.c
BendLinearPass.c
BndMPoleSymplectic4E2Pass.c
BndMPoleSymplectic4E2RadPass.c
BndMPoleSymplectic4Pass.c
BndMPoleSymplectic4RadPass.c
CavityPass.c
CorrectorPass.c
DriftPass.c
QuantDiffPass.c

EAperturePass.c
IdTablePass.c
IdentityPass.c
Matrix66Pass.c
QuadLinearPass.c
SolenoidLinearPass.c
StrMPoleSymplectic4Pass.c
WiggLinearPass.c
GWigSymplecticPass.c

Others have developed different integrators. We should find the most accurate and fastest integrators.
MOVING AWAY FROM GLOBAL VARIABLES
ELEMENT AND RING CREATION

global FAMLIST, THERING

element creation

ring structure

AT 1.3 uses

Issue: We want to be able to create many lattices in one Matlab session

For the element creation, output elements directly. For example

DR1=atdrift('D1',0.7)
QF=atquadrupole('QF1',0.94,0.39,'QuadMPoleFringePass')
SH=atsextupole('SH',0,0,'StrMPoleSymplectic4Pass');
B1=atsbend('B1',0,0,0,'BndMPoleSymplectic4Pass');

QF =

FamName: 'QF1'
PassMethod: 'QuadMPoleFringePass'
Length: 0.9400
Class: 'Quadrupole'
K: 0.3900
PolynomB: [0 0.3900]
PolynomA: [0 0]
MaxOrder: 1
NumIntSteps: 10
Many models out there.
We have a magnetic field.
Usually, fourth order symplectic integrator
(Ruth, Forest, Yoshida...)
Depends on Hamiltonian, should be separable into two solvable parts

\[ H = \frac{\gamma}{B \rho} - (1 + \delta) h x + (1 + \delta x) \frac{p_x^2 + p_y^2}{2(1 + \delta)} \]

X. Huang (improves drift in dipole)
RADIA model script provided by G. Le Bec. Performs tracking of electrons in the complete field generated by the magnet (full line). Squares represent the particles tracking in AT for the equivalent hard edge, iron length, straight multipole model.
**NEW TOOLS**

- **atmatch (S. Liuzzo)**
- **atfastring (L. Farvacque)**
- **linear matrix, chromatic and amplitude tune shift quantum diffusion element used for collective effects, spin dynamics studies**
- **atplot (L. Farvacque)**
- **allows arbitrary function together with synoptic**
- **quantum diffusion element QuantDiffPass (N. Carmignani, B. Nash)**
- **collective effects- multiparticle tracking with impedance (S. White)**
- **Apertures in elements for improved loss mapping (L. Farvacque, R. Versteegen)**
- **recently added by S. Liuzzo:**
- **error setting and correction tools**
- **see Nash et. al. IPAC 14 for some details**
  
  http://inspirehep.net/record/1417208
Available functions, documentation in progress, draft available in atmat/pubtools

2 Corrections

2.1 Response matrices

2.2 Tune

2.3 Chromaticity

2.4 RF cavity

2.5 Orbit

2.5.1 Closed orbit bumps

2.6 Dispersion

2.7 Dispersion free steering

2.8 Correction of RDT

2.9 Coupling free steering

2.10 Open trajectory correction

2.11 Commissioning like correction sequence

Finally, a single function does the whole commissioning like procedure

(S. Liuzzo)
This function can be used on a fitted lattice or on the lattice with errors.

alpha: $0.80 \text{ (h disp)} \times 0.10 \times \text{tune} \times 0.80 \text{ (v disp)}$

RDT Disp. Tune Steering iter 1, n-eig: 150, 50
RDT Disp. Tune Steering iter 2, n-eig: 250, 100

before $\rightarrow$ after

rq: 9.174 $\rightarrow$ 5.711
rs: 15.024 $\rightarrow$ 7.079
dX: 5.839 $\rightarrow$ 1.810mm
dY: 1.466 $\rightarrow$ 0.715mm
tX: 0.008 $\rightarrow$ 0.007
tY: -0.000 $\rightarrow$ -0.004

min mean max
qs:-0.03 -0.00 0.03 1/m2
ss:-0.01 -0.00 0.01 1/m2
dpp: 0

The normal and skew quadrupole components are computed from gradient errors and from alignment errors.

(S. Liuzzo)
Possible to correct to reference orbit
Usable for bumps
Defaulted RM computation from model
Correctors limits
Average of correctors to zero and DPP correction
Easy iteration varying the number of eigenvectors.

[S. Liuzzo)
OPEN TRAJECTORY CORRECTION

```
[rcor,inCOD]=atfirstturntrajectory(...
   rerr,...
inCOD,...
indBPM,...
indHCor,...
indVCor,...
2.0e-3,...
60,...
true true,...
ModelRM,...
zeros(2,length(indBPM)),....
[5e-3 5e-3]);
```

Possible to correct to reference trajectory
Defaulted RM computation from model
Correctors limits
Average of correctors to zero and DPP correction

(S. Liuzzo)
Improve documentation
continue standardization of lattice format
merge physics tools and integrators into common repository
more tests of compatibility with MML

Python AT- pyAT is in development in collaboration with Will Rogers from Diamond
https://github.com/willrogers/at
<table>
<thead>
<tr>
<th></th>
<th>current ESRF</th>
<th>EBS</th>
</tr>
</thead>
<tbody>
<tr>
<td>circumference (m)</td>
<td>844.39</td>
<td>843.979</td>
</tr>
<tr>
<td>emittance (pm)</td>
<td>4000</td>
<td>135</td>
</tr>
<tr>
<td>beta_x,y</td>
<td>37/3</td>
<td>6.9/2.6</td>
</tr>
<tr>
<td>damping times (msec)</td>
<td>7/7/3.5</td>
<td>8.5/13/8.8</td>
</tr>
<tr>
<td>momentum compaction</td>
<td>1.78e-4</td>
<td>0.87e-4</td>
</tr>
<tr>
<td>tunes</td>
<td>36.44/13.39</td>
<td>75.6/27.6</td>
</tr>
<tr>
<td>Energy loss/turn (MeV)</td>
<td>4.88</td>
<td>2.60</td>
</tr>
<tr>
<td>Natural chrom</td>
<td>-130/-58</td>
<td>-100/-84</td>
</tr>
</tbody>
</table>

(atplot)
function atx() gives linear optics plus Ohmi Envelope results

\[
\begin{align*}
\text{[linopt, p]} &= \text{atx(esrf)} \\
\text{[linoptS28D, pS28D]} &= \text{atx(S28D)}
\end{align*}
\]

\[
\begin{align*}
p &= \\
\text{ll: } 844.3904 \\
\text{alpha: } 1.7758e-04 \\
\text{fractunes: } [0.4396, 0.3901] \\
\text{fulltunes: } [36.4396, 13.3901] \\
\text{nuh: } 36.4396 \\
\text{nuv: } 13.3901 \\
\text{chromaticity: } [5.7983, 6.1301] \\
\text{dampingtime: } [0.0070, 0.0070, 0.0035] \\
\text{espread: } 0.0011 \\
\text{blength: } 0.0043 \\
\text{modemittance: } [4.0131e-09, 1.3081e-36, 4.5432e-06] \\
\text{energy: } 6.0400e+09 \\
\text{fs: } 2.1040e+03 \\
\text{eloss: } 4.8806e+06 \\
\text{synchrophase: } 0.5732 \\
\text{momcompact: } 1.7758e-04
\end{align*}
\]

\[
\begin{align*}
pS28D &= \\
\text{ll: } 843.9773 \\
\text{alpha: } 8.5140e-05 \\
\text{fractunes: } [0.2100, 0.3400] \\
\text{fulltunes: } [76.2100, 27.3400] \\
\text{nuh: } 76.2100 \\
\text{nuv: } 27.3400 \\
\text{chromaticity: } [5.9996, 3.9997] \\
\text{dampingtime: } [0.0088, 0.0134, 0.0091] \\
\text{espread: } 9.3359e-04 \\
\text{blength: } 0.0029 \\
\text{modemittance: } [1.3211e-10, -5.0066e-38, 2.7210e-06] \\
\text{energy: } 6.0000e+09 \\
\text{fs: } 1.3017e+03 \\
\text{eloss: } 2.5224e+06 \\
\text{synchrophase: } 0.3985 \\
\text{momcompact: } 8.5140e-05
\end{align*}
\]
Zn = .67 ohms for bunch lengthening

booster beam params for injection efficiency calcs:
$\epsilon_x, y = (120, 5), (60, 5), (30, 30)$
tau = 16.3 +/- 2.8 hrs

computed on cluster (parallelized tracking) by N. Carmignani
### Mode Parameters

<table>
<thead>
<tr>
<th>Mode</th>
<th>Intensity</th>
<th>$I_b$ (mA)</th>
<th>Lifetime</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>200</td>
<td>.202</td>
<td>8.5 hrs</td>
</tr>
<tr>
<td>16 bunch</td>
<td>90</td>
<td>5.62</td>
<td>2.42 hrs</td>
</tr>
<tr>
<td>4 bunch</td>
<td>40</td>
<td>10</td>
<td>1.78 hrs</td>
</tr>
</tbody>
</table>
CONCLUSIONS

AT is still widely used in light source community

An attempt has been made to coordinate development, called atcollab

AT is used in two ways, as an online modeling tool, usually integrated within MML, and secondly as a design tool for low emittance light sources.

We (at ESRF) have added tools to make AT powerful for design purposes, and others can benefit from our work.

We have attempted to maintain backwards compatibility, so we hope that our version of AT can also be used widely with MML and a standard version can be maintained.

AT has been the main tool used in the modeling for the ESRF EBS.

Let’s work together for a release of AT2.0!!
To fit at best the tracking in RADIA for QF8, with a trapezoid, (field integral unchanged)

\[ F_1 = 0.079 \, \text{m} \]
\[ (I_{1+} = I_{1-} = 0.00013) \]

Also similar or better then a sliced model
**Issue:** if we want to define many different ring structures at the same time, we need a way to define global parameters for each one.

**Idea:** create an element in each ring to store global parameters for that ring.

\[ \text{ATRINGPARAM(FAMNAME,E0,NBPERIODS)} \]

- reference energy
- periodicity of lattice

Not many functions yet developed to take advantage of this feature.

*IdentityPass needed as PassMethod for RingPass*