PTC Integration into MAD-X

- What is PTC? (Etienne's words)
- Some Facts about PTC
- What are the advantages for MAD-X?
- Magnet Treatment in PTC
- How will we use PTC in MAD-X?
- New MAD-X Module with Attributes PTC_TWISS and NORMAL

What is PTC?

In Etienne's own words:

The code PTC is composed of two distinct parts.

A) The independent polymorphic library FPP* which handles the operations on polymorphs. Polymorphs are capable of transforming real numbers into Taylor at execution. FPP also contains all the critical operation on Taylor maps: Normal Forms, various factorizations, etc…

B) The code PTC proper is at heart a symplectic integrator with classical radiation added on top. It also moves around quadratic stochastic beam envelopes.

*(based in an idea by Bengtsson and relying on Berz's package)

Some Facts about PTC

- Medium sized Package:

  PTC 68'000 Lines

  MAD-X 53'000 Lines

- Part I of PTC: The Truncated Power Series Analysis (TPSA) called FPP, comprises 20 Years of Development in Power Series and Normal Form Analysis. Takes full advantage of object oriented programming and polymorphism.
- Part II of PTC proper: (5 last years) makes use of FPP to track through magnets in a truly symplectic way.

What are the advantages for MAD-X?

- Despite its success and usefulness, MAD has always suffered from fundamental limitations: ➔
- PTC addresses these Issues:
  - Physics Model ➔➔
  - Element – Tunnel Relation ➔➔
  - The Principal of ONE Code ➔
- However: More Fiddling and Time penalty maybe hefty!

➔ As a consequence:

1. Keep and Use MAD-X for most Applications
2. For small Machines, Normal Form and Testing ➔ PTC

Magnet Treatment in PTC

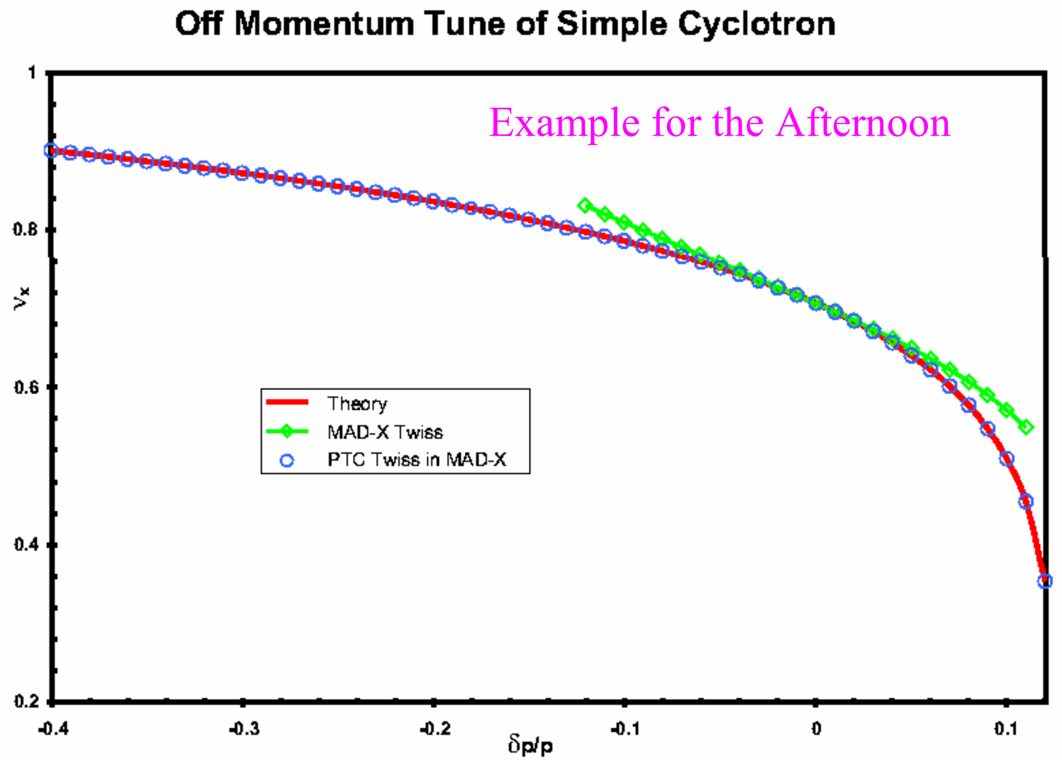
- Same Elements as in MAD-X, including all special features ε_1, ε_2, H_1, H_2 etc.
- Rectangular and Sector bends in all generality
- Magnets are preferably treated as thick elements including multipole errors, which will be split for symplectification
- Elaborate Symplectification Techniques
  - Various Split Combinations: Drift_Kick, Matrix_Kick, Delta_Matrix_Kick etc
  - Different Split Method to several Orders
  - Number of Splits
  - Exact Option: on/off
- Normal Form: Order & Phase Space variables

How will we use PTC in MAD-X?

- Linear coupled Lattice Functions TWISS3 (presently available in rudimentary form)
- Replace Modules that involve Normal Form Analysis DYNAMIC & STATIC (presently available in rudimentary form)
- Parameter Dependence of Variables of Interest on other Variables ➔ Nonlinear Matching
- Extend the Physics Models in particular for small Machines
- Proper treatment of Geometry of Magnet in the Tunnel
- Survey with CAD like presentation including particle tracks

PTC-TWISS MAD-X Module

E. Courant et al, "A comparison of several Lattice Tools for Computation of Orbit Functions of an Accelerator", published in PAC2003 Portland, shown is ν_x versus $\delta p/p$ for a simple cyclotron. Standard MAD-X gives the green curve which deviates since the MAD-X (like MAD8) uses the expanded Hamiltonian. In PTC the exact attribute allows to the treat the true Hamiltonian. Note, that PTC has read-in the structure from MAD-X input. There is now PTC_TWISS as attribute of the PTC MAD-X module (still rudimentary!) that allows to produce the Ripken/ Willeke lattice functions called TWISS3 in MAD8.



Normal Form MAD-X Module

There is now also a NORMAL attribute of the PTC MAD-X module (still rudimentary!) to calculate dispersion, tune and anharmonicities to high orders and as function of delta. This module will be eventually become the replacement of the DYNAMIC/STATIC of MAD8.

What have we achieved ?

- Better treatment of Cavity (standing and traveling wave) including fringe fields →This concludes the implementation of all standard Elements
- Misalignment in PTC to reproduce the MAD-X results
- **Central Point**: Structures of both Codes are too different to actually integrate fully →
- A) Some Passing of Parameters (Matching) via tables
- B) Feeding PTC with Machine Structures
- C) Control Handling of PTC via MAD-X Commands
- First acceptable Set-up of this Handling

```
use, period=ring;
show beam;
SELECT, FLAG=ERROR, CLEAR = true;
SELECT, FLAG=ERROR, CLASS=sbend, RANGE=bd;
!EALIGN, DX=1e-5,DPHI=0e-7,DTHETA=0e-8,DPSI=0e-9;
!EALIGN,  DX=1e-5,Dphi=1e-4,DTHETA=3e-4,DPSI=2e-4;
EALIGN,  DX=1e-5,Dy=1e-5;
mydeltap=0e-4;
select,flag=twiss,clear; select,flag=twiss,column=name,s,mux,muy,betx,bety,muy,dx,x,px,t,pt;
twiss,tolerance=1e-10,file="madx-twiss",deltap=mydeltap;
ptc_create_universe;
ptc_create_layout,model=3,method=6,nst=100,exact;
ptc_create_layout,model=3,method=6,nst=100,exact;
ptc_move_to_layout, index=1;
ptc_normal,icase=6,no=1,deltap=mydeltap;
ptc_move_to_layout, index=2;
ptc_align;
ptc_normal,icase=6,no=1,deltap=mydeltap;
ptc_end;
```