Wiggler Simulations

Steps of the Simulations

- Calculate the magnetic field map
  - Wiggler design
- Determine adapted field map
  - To simplify tracking, ensure field consistency
- Track particles through wiggler field
  - Full tracking using precision methods
- Determine map to represent wiggler in tracking code
  - Simplified treatment for fast multi-turn simulations

Magnetic Field Map Calculations

- Specialised codes for magnet design and field calculation
- Measurements of the field
- A simplified representation often useful
  - Reduce noise
  - Ensure consistency of magnetic field
- Magnetic potential

$$\Psi = \sum_{l,n} c_{l,n} \cos(l k_x x) \sin(n k_z z) \cosh(k_y y)$$

$$k_y^2 = (l k_x)^2 + (n k_z)^2$$

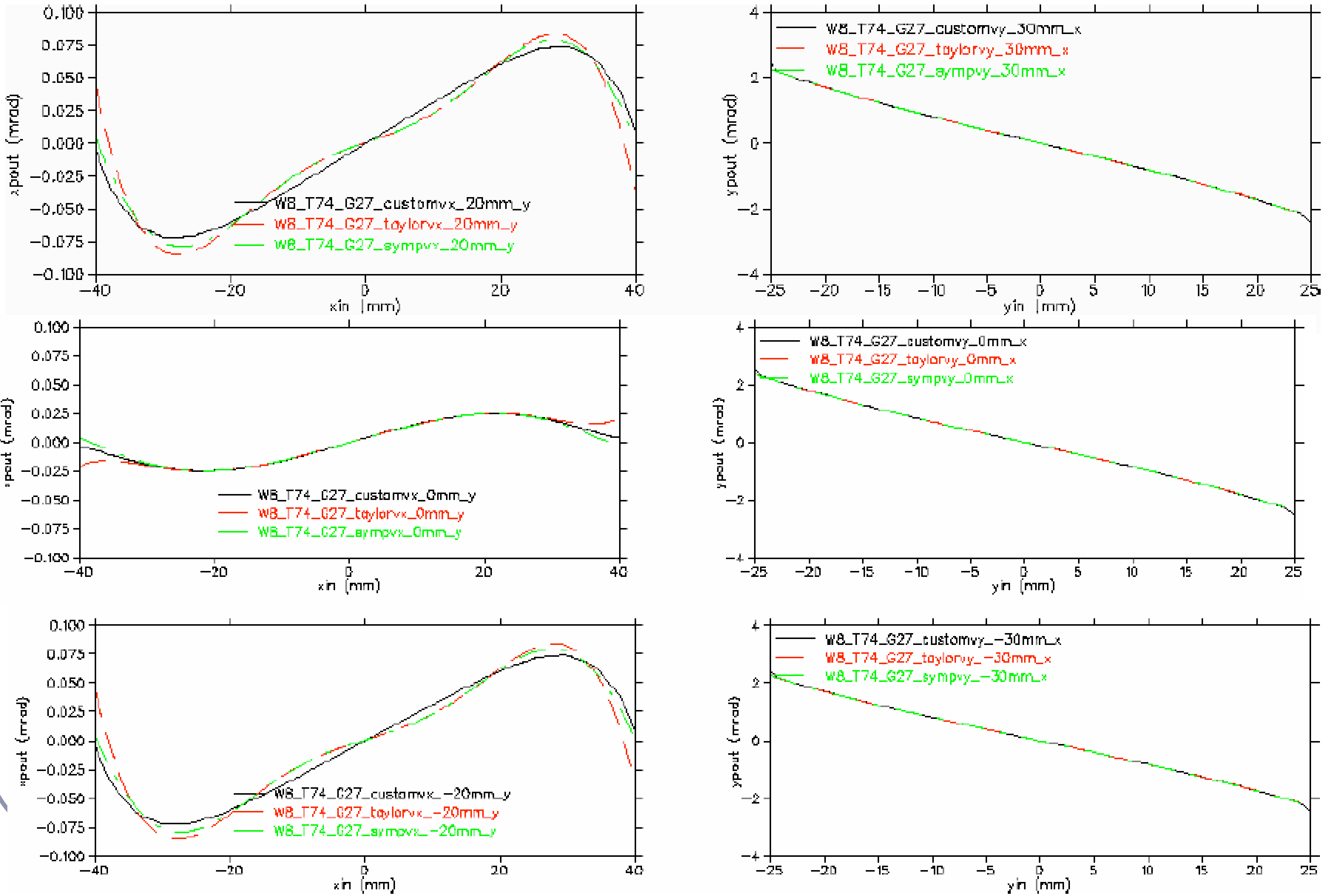
Particle Tracking

- Two main options
  - High precision ordinary tracking routine
    - Generic but normally quite time consuming
    - Can create small but not negligeable errors in Hamiltonian (example: energy in harmonic oszillator)
  - Symplectic tracking
    - Needs a bit more thought
    - Avoids errors in Hamiltonian (e.g. ensure energy is preserved in harmonic oszillator)
    - Error is only in less important variable (e.g. phase in oszillator)

Transfer Maps

- For fast multi-turn tracking replace (part of) the wiggler by a map to transform initial position and momentum q_i, p_i to final values Q_i, P_i
- Can be done by tracking particles with different initial conditions and creating look-up table with interpolation
- Normally need to be symplectic (amplitude of motion is important, actual position not as much)
- So use canonical transformation, i.e. use generating function, e.g. $F(q_i, P_i)$, $p_i = \partial F/\partial q_i$, $Q_i = \partial F/\partial P_i$

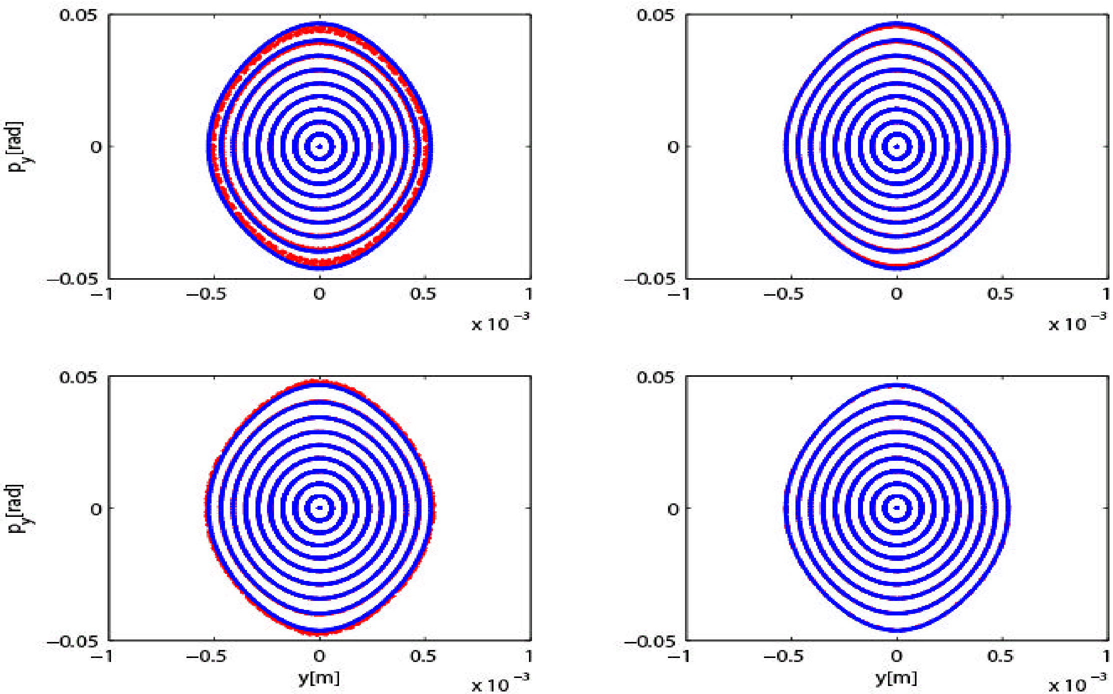
CESR-c Transfer Functions



Symplectic Conditions in Hamiltonian System

6th order

8th order



Increment: $5\sigma_y$

Taylor map (Zlib)

Mix-variable generating function (Zlib)

element-by-element tracking (LEGO)

Numerical generating function (GF)

explicit orbit integration through the ID yields, rearranged: $(q_{xi}, q_{yi}, P_{xf}, P_{yf}) \Rightarrow (Q_{xf}, Q_{yf}, p_{xi}, p_{yi})$ $i, f =$ initial, final

construct a polynomial GF of type F_2

$$F_2(q_{xi}, q_{yi}, P_{xf}, P_{yf}) = \sum_{k+l+m+n=1}^{M} a_{klmn} q_{xi}^k q_{yi}^l P_{xf}^m P_{yf}^n$$

$M = 4 \dots 6$

with the properties

$$Q_{xf} = \partial F_2 / \partial P_{xf}, \quad Q_{yf} = \partial F_2 / \partial P_{yf}$$
$$p_{xi} = \partial F_2 / \partial q_{xi}, \quad p_{yi} = \partial F_2 / \partial q_{yi}$$

and fit numerically the a_{klmn}

implicit equations of motion solved by Newton fit routine

Codes

- Field map generation -> task of designer
- Field map fits -> ask Marco, Pavel, Winni, Jeremy
- Tracking: generic routines exist, symplectic ones should be not difficult
- Fast tracking by maps -> can implement generating function (not too difficult), routines exist in at least in LEGO and SIXTRACK