Status of the MADX 'thintrack' module

Yi-Peng Sun, Frank Schmidt, Frank Zimmermann

AB/ABP Group, CERN

Thanks to: Alex Koschik, Massimo Giovannozzi, Bernard Jeanneret

Supported by the European Community-Research Infrastructure Activity under the FP6 "Structuring the European Research Area" programme (CARE, contract number RII3-CT-2003-506395)

Thin-lens Tracking module

- File trrun.F
- '*Makethin*'→ tracks symplecticly through drifts and kicks
- **canonical** coordinates or **action-angle** variables
- *Onepass:* reference orbit
- Usual tracking: **closed orbit**
- Aperture: *apertype* and *aperture*
- **canonical** coordinates: (x,px,y,py,t,pt)

Things to do

1. 6-D closed orbit (subroutine)
2. Aperture (Racetrack)
3. Initial small values

6-D closed orbit (1)

$$\vec{x} = \{x, px, y, py, t, pt\}$$

The goal is: $\vec{x_0} = \mathbf{M} \cdot \vec{x_0}$

$\vec{x_2} = \mathbf{M_1} \cdot \vec{x_1}$ ($\vec{x_1}$ from Twiss module)

$$\vec{x_{11}} = \{x + \Delta_1, px, y, py, t, pt\} \ ...$$
$$\vec{x_{16}} = \{x, px, y, py, t, pt + \Delta_6\}$$

6-D closed orbit (2)

$$\tilde{M} = \begin{pmatrix} \frac{X_{21}-X_2}{\Delta_1} & \frac{X_{22}-X_2}{\Delta_1} & \frac{X_{23}-X_2}{\Delta_1} & \frac{X_{24}-X_2}{\Delta_1} & \frac{X_{25}-X_2}{\Delta_1} & \frac{X_{26}-X_2}{\Delta_1} \\ \frac{X_{21}-X_2}{\Delta_2} & \frac{X_{22}-X_2}{\Delta_2} & \frac{X_{23}-X_2}{\Delta_2} & \frac{X_{24}-X_2}{\Delta_2} & \frac{X_{25}-X_2}{\Delta_2} & \frac{X_{26}-X_2}{\Delta_2} \\ \frac{X_{21}-X_2}{\Delta_3} & \frac{X_{22}-X_2}{\Delta_3} & \frac{X_{23}-X_2}{\Delta_3} & \frac{X_{24}-X_2}{\Delta_3} & \frac{X_{25}-X_2}{\Delta_3} & \frac{X_{26}-X_2}{\Delta_3} \\ \frac{X_{21}-X_2}{\Delta_4} & \frac{X_{22}-X_2}{\Delta_4} & \frac{X_{23}-X_2}{\Delta_4} & \frac{X_{24}-X_2}{\Delta_4} & \frac{X_{25}-X_2}{\Delta_4} & \frac{X_{26}-X_2}{\Delta_4} \\ \frac{X_{21}-X_2}{\Delta_5} & \frac{X_{22}-X_2}{\Delta_5} & \frac{X_{23}-X_2}{\Delta_5} & \frac{X_{24}-X_2}{\Delta_5} & \frac{X_{25}-X_2}{\Delta_5} & \frac{X_{26}-X_2}{\Delta_5} \\ \frac{X_{21}-X_2}{\Delta_6} & \frac{X_{22}-X_2}{\Delta_6} & \frac{X_{23}-X_2}{\Delta_6} & \frac{X_{24}-X_2}{\Delta_6} & \frac{X_{25}-X_2}{\Delta_6} & \frac{X_{26}-X_2}{\Delta_6} \end{pmatrix}.$$

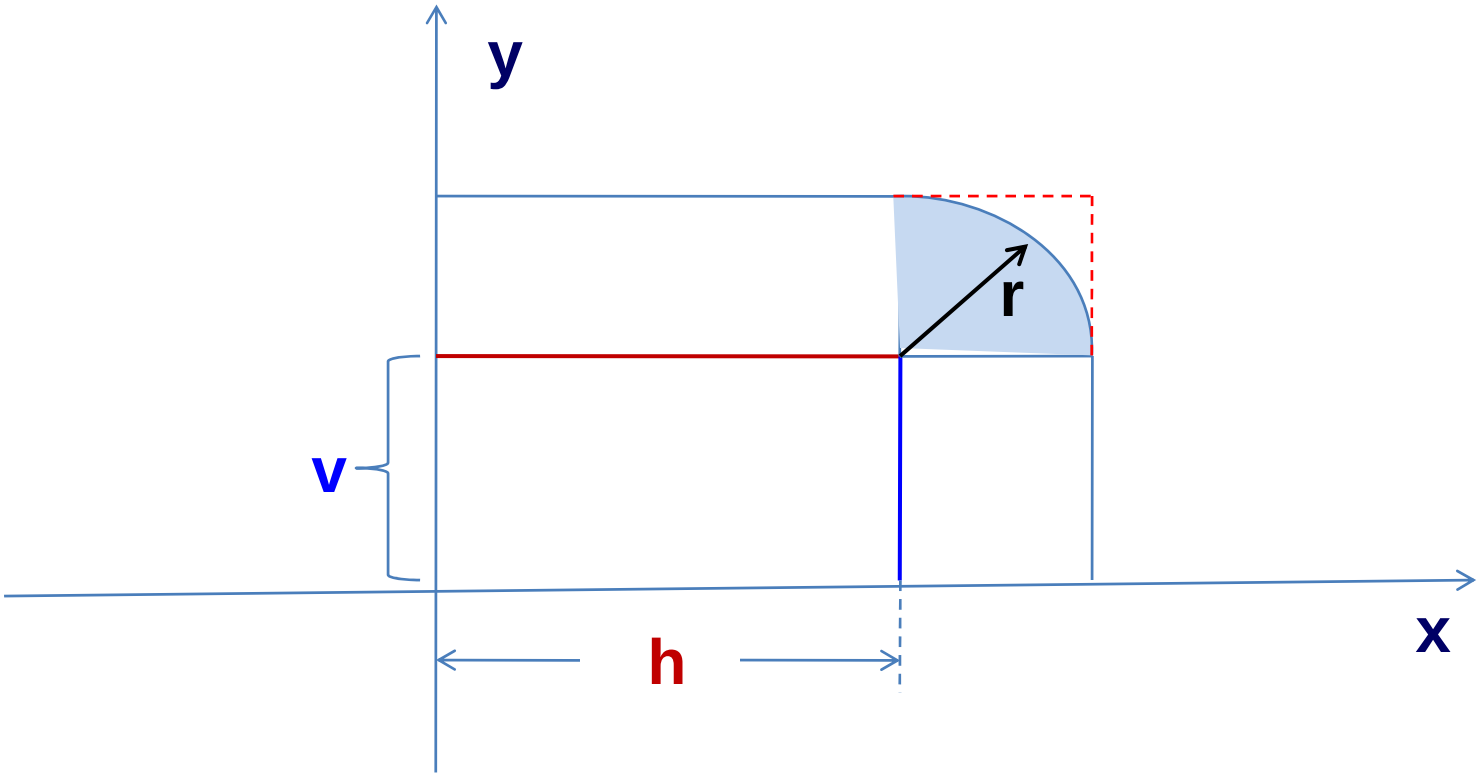
From $\vec{x1} - \vec{x2} = 0$ and $\tilde{M}\Delta x_1 = \Delta x_2$
we get $\tilde{M}^{-1}\Delta x_2 = \delta$ and $\vec{x_3} = \vec{x_1} + \delta$

Repeat this process to get closed orbit.

List of Aperture types (MADX)

APERTYPE	# of parameters	meaning of parameters
CIRCLE	1	radius
ELLIPSE	2	horizontal half axis, vertical half axis
RECTANGLE	2	half width and half height
LHCSCREEN	3	half width, half height (of rect.) and radius (of circ.)
MARGUERITE	3	half width, half height (of rect.) and radius (of circ.)
RECTELLIPSE	4	half width, half height (of rectangle), horizontal half axis, vertical half axis (of ellipse)
RACETRACK	**3**	**radius, horizontal shift, vertical shift**
FILENAME	0	where the file contains a list of x and y coordinates outlining the shape. This option is only supported by the aperture module, see below.

Racetrack type



Racetrack (4, h, v, r)

Aperture: $x<(h+r)$ & $y<(v+r)$ & $(x-h)^2+(y-v)^2 <= r^2$

Code in trrun.F

```
!------------  racetrack case ---------------------------------
        if(aptype.eq.'racetrack') then
          apx = aperture(1)
          apy = aperture(2)
          apr = aperture(3)
          call trcoll(4, apx, apy, turn, sum, part_id, last_turn,      &
                      &last_pos, last_orbit, track, ktrack,al_errors)


 subroutine trcoll(flag, apx, apy, turn, sum, part_id, last_turn,  &
     &last_pos, last_orbit, z, ntrk,al_errors)
……
else if(flag .eq. 4 .and. (abs(z(1,i)-al_errors(11)) .gt. (apx+apr)     &
     &.or. abs(z(3,i)-al_errors(12)) .gt. (apy+apr) .or. &
     & (((z(1,i)-apx)-al_errors(11))/apr)**2+(((z(3,i)-apy)-al_errors(12))/apr)**2 .gt. one))   )
then
          go to 99
……
99   Call trkill()
98     continue
```