Combined function magnet modeling

Malte's Publication 1 of 4 PRSTAB 19, 054002 (2016) Potential G

Included fields up to sextupoles "h" and used an expansion of up to total order 4. In the implementation we have treated K_x and K_y separated and ignored sextupoles/

$$G = \frac{K_x^2 g}{24} y^4 - \frac{K_x^2 x^2}{2} + \frac{K_x K_y}{6} g x y^3 - K_x K_y x y - \frac{K_x g}{3} x^3$$
$$+ \frac{K_x g}{2} x y^2 - \frac{K_x h}{8} x^4 + \frac{K_x h}{2} x^2 y^2 - \frac{K_x h}{24} y^4$$
$$- K_x x - \frac{K_y^2 g}{24} y^4 - \frac{K_y^2 y^2}{2} + \frac{K_y g}{6} y^3 + \frac{K_y h}{6} x y^3$$
$$- K_y y - \frac{g x^2}{2} + \frac{g y^2}{2} - \frac{h x^3}{6} + \frac{h x}{2} y^2,$$

Malte's Publication 2 of 4
PRSTAB 19, 054002 (2016)
Symplectic Map EQ. 38

$$x^f = x^i,$$

$$p_x^f = p_x^i + \Delta s(K_x(1 + \hat{\eta}) + \partial_x G)$$

$$y^f = y^i,$$

$$p_y^f = p_y^i + \Delta s(K_y(1 + \hat{\eta}) + \partial_y G)$$

$$\sigma^f = \sigma^i - \Delta s\left(K_x x + K_y y\right)\left(1 + \beta_0^2 p_\sigma^i\right)/\left(1 + \hat{\eta}\right)$$

Malte's Publication 3 of 4
PRSTAB 19, 054002 (2016)
Drift-Kick Map EQ. 39

$$x^f = x^i + \Delta s(K_x x^i + K_y y^i) p_x^i / h_0,$$

$$p_x^f = p_x^i + \Delta s(K_x h_0 + \partial_x G),$$

$$y^f = y^i + \Delta s(K_x x^i + K_y y^i) p_y^i / h_0,$$

$$p_y^f = p_y^i + \Delta s(K_y h_0 + \partial_y G),$$

$$\sigma^f = \sigma^i - \Delta s(K_x x^i + K_y y^i)(1 + \beta_0^2 p_\sigma^i)/h_0$$

Malte's Publication 4 of 4
PRSTAB 19, 054002 (2016)
Slice Map EQ. 34

$$x^f = x^i + \Delta s(1 + K_x x^i + K_y y^i)(u_x^i/h + \Delta s K_x),$$

$$p_x^f = u_x^i + \Delta s K_x h,$$

$$y^f = y^i + \Delta s(1 + K_x x^i + K_y y^i)(u_y^i/h + \Delta s K_y),$$

$$p_y^f = u_y^i + \Delta s K_y h,$$

$$\sigma^f = \sigma^i + \Delta s - \Delta s(1 + K_x x^i + K_y y^i)(1 + \beta_0^2 p_\sigma^i)/h$$

Interlude: RE/TE versus DA Maps

• Despite decade-long developments (early ‘80s) and since ~2001 even at CERN, with the explicit effort to construct PTC (Forest et al.) DA Maps and NormalForm techniques have not truly entered our culture.

• In desperate need to properly implement Malte’s formalism into SixTrack, I have made a first attempt to turn RE & TE, i.e. first and second terms, into the equivalent DA Map. Mind you the substantial advantage to get at the MAD-X RE & TE terms automatically, by just properly defining the equations of motion!

• This allows to either compare each term in my case at the location of a single CFM or to compare directly either one-turn DA Mapd by getting at the chromaticity using NormalForm.

• This was very useful because MAD-X and SixTrack differ in their conventions and since there are: signs, factors and choices of canonical coordinates that this is far from trivial to get it right! I could get it clarified in the present state.

• I strongly suggest that we create a small MAD-X module that produces DA Maps as a benchmarking tool. Mine has only been a first start.

Simulation Results for the PS

rel. beta:	0.94764
PTC Set-Up:	model=2,method=4,nst=3

Disclaimer: ' is with respect to MAD-X pt!

THICK	THIN	Mad-X Twiss		PTC time=true,exact=true		PTC: time=true,exact=false	
		Q'_h	Q'v	Q'_h	Q'v	Q'_h	Q'v
x		-5.53489	-7.38032	-5.53478	-7.38020	-6.81463	-7.74048
	x	-5.53511	-7.37520	-6.81461	-7.74046	-6.81461	-7.74046
Deviation [%]		0.00397	-0.06939	23.12341	4.88144	23.12097	4.87973

Space Charge	x

Mad-X Twiss		PTC time=true,exact=true		Sixtrack Eq. 38- symplectic		Sixtrack Eq. 39- symplectic no co		Sixtrack Eq. 34- symplectic		
				Q'_h*beta	Q'v*beta	Q'_h*beta	Q'v*beta	Q'_h*beta	Q'v*beta	
				-6.04548	-7.27541	-5.57020	-7.05577	-5.56977	-7.05553	
Q'_h	Q'v	Q'_h	Q'v	Q'_h	Q'v	Q'_h	Q'v	Q'_h	Q'v	
-5.87808	-7.44631	-7.20223	-7.81966	-6.37948	-7.67736	-5.87794	-7.44559	-5.87749	-7.44534	
Deviation		22.52689	5.01395	8.53004	3.10291	-0.00231	-0.00968	-0.01003	-0.01308	%

Implementation into SixTrack 1 of 3
Eq. 38

```
! M. Titze, PRSTAB 19, 054002 (2016), EQ. 38
!FOX  CRKVE=XL ;
!FOX  CIKVE=ZL ;
!FOX  YV1J=BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(HALF*ZL*ZL-XL*XL)*C1M3-
!FOX  DKI(IX,1)*DPDA*C1E3DKI(IX,1)*DKI(IX,1)/DKI(IX,3)*XL ;
!FOX  YV2J=-BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(C1M6*DKI(IX,1)/DKI(IX,3)*ZL*ZL*ZL/SIX-XL*ZL*C1M3) ;
! M. Titze, PRSTAB 19, 054002 (2016), EQ. 38
!FOX SIGMDA=SIGMDA+DKI(IX,1)*XL*(ONE+(EJ1-E0)/E0)/(ONE+DPDA) ;
```

Implementation into SixTrack 2 of 3
Eq. 39

```
! M. Titze, PRSTAB 19, 054002 (2016), EQ. 39
!FOX  CRKVE=XL ;
!FOX  CIKVE=ZL ;
!FOX  TEMPI(2) = Y(1)*(ONE+DPDA) ;
!FOX  TEMPI(4) = Y(2)*(ONE+DPDA) ;
!FOX  H0=SQRT((ONE+DPDA)*(ONE+DPDA)-TEMPI(2)*TEMPI(2)*C1M6-TEMPI(4)*TEMPI(4)*C1M6) ;
!FOX  CRKVE=CRKVE-DKI(IX,1)*XL*TEMPI(2)*C1M3/H0 ;
!FOX  CIKVE=CIKVE-DKI(IX,1)*XL*TEMPI(4)*C1M3/H0 ;
!FOX  YV1J=BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(HALF*ZL*ZL-XL*XL)*C1M3+DKI(IX,1)*(ONE-H0)*C1E3-
!FOX  DKI(IX,1)*DKI(IX,1)/DKI(IX,3)*XL ;
!FOX  YV2J=-BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(C1M6*DKI(IX,1)/DKI(IX,3)*ZL*ZL*ZL/SIX-XL*ZL*C1M3) ;
!FOX  XL=CRKVE ;
!FOX  ZL=CIKVE ;
! M. Titze, PRSTAB 19, 054002 (2016), EQ. 39
!FOX  SIGMDA=SIGMDA+DKI(IX,1)*XL*(ONE+(EJ1-E0)/E0)/H0 ;
```

Implementation into SixTrack 3a of 3 Eq. 34

```
! M. Titze, PRSTAB 19, 054002 (2016), EQ. 34
!FOX  CRKVE=XL ;
!FOX  CIKVE=ZL ;
!FOX  TEMPI(2) = Y(1)*(ONE+DPDA) ;
!FOX  TEMPI(4) = Y(2)*(ONE+DPDA) ;
!FOX  PZ=SQRT((ONE+DPDA)*(ONE+DPDA)*C1E6-(TEMPI(2)*TEMPI(2)+TEMPI(4)*TEMPI(4)))*C1M3 ;
!FOX  CRKVE=CRKVE-HALF*DKI(IX,3)*TEMPI(2)/PZ ;
!FOX  CIKVE=CIKVE-HALF*DKI(IX,3)*TEMPI(4)/PZ ;
if(idp.eq.1.and.iabs(ition).eq.1) then
!FOX  SIGMDA=SIGMDA-HALF*DKI(IX,3)*(ONE-RV/PZ*(ONE+DPDA))*C1E3 ;
  endif
!FOX  XL=CRKVE ;
!FOX  ZL=CIKVE ;
!FOX  UX=TEMPI(2)+BBIV(2,I)*XL+BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(HALF*ZL*ZL-
XL*XL)*C1M3+DKI(IX,1)*C1E3-
!FOX  DKI(IX,1)*DKI(IX,1)/DKI(IX,3)*XL ;
!FOX  UY=TEMPI(4)-BBIV(2,I)*ZL-
BBIV(2,I)*DKI(IX,1)/DKI(IX,3)*(C1M6*DKI(IX,1)/DKI(IX,3)*ZL*ZL*ZL/SIX-XL*ZL*C1M3) ;
!FOX  XI= -UX*DKI(IX,1)/(ONE+DKI(IX,1)*DKI(IX,1)) ;
!FOX  ZETA=(UX*UX+UY*UY-C1E6*(ONE+DPDA)*(ONE+DPDA))/(ONE+DKI(IX,1)*DKI(IX,1)) ;
!FOX  HH=-XI+SQRT(XI*XI-ZETA) ;
!FOX  CRKVE=CRKVE+(DKI(IX,3)*C1E3-DKI(IX,1)*XL)*(UX/HH-DKI(IX,1)) ;
!FOX  CIKVE=CIKVE+(DKI(IX,3)*C1E3-DKI(IX,1)*XL)*UY/HH ;
!FOX  YV1J=UX-TEMPI(2)-DKI(IX,1)*HH ;
!FOX  YV2J=UY-TEMPI(4) ;
    if(idp.eq.1.and.iabs(ition).eq.1) then
!FOX  SIGMDA=SIGMDA+DKI(IX,3)*C1E3-DKI(IX,3)*(C1E3-DKI(IX,1)/DKI(IX,3)*XL)*(ONE+(EJ1-
E0)/E0)/HH*C1E3 ;
    endif
```

Implementation into SixTrack 3b of 3 Eq. 34

```
!FOX  TEMPI(2) = Y(1)*(ONE+DPDA)+YV1J ;
!FOX  TEMPI(4) = Y(2)*(ONE+DPDA)+YV2J ;
!FOX  PZ=SQRT((ONE+DPDA)*(ONE+DPDA)*C1E6-(TEMPI(2)*TEMPI(2)+TEMPI(4)*TEMPI(4)))*C1M3 ;
!FOX  CRKVE=CRKVE-HALF*DKI(IX,3)*TEMPI(2)/PZ ;
!FOX  CIKVE=CIKVE-HALF*DKI(IX,3)*TEMPI(4)/PZ ;
if(idp.eq.1.and.iabs(ition).eq.1) then
!FOX  SIGMDA=SIGMDA-HALF*DKI(IX,3)*(ONE-RV/PZ*(ONE+DPDA))*C1E3 ;
endif
!FOX  XL=CRKVE ;
!FOX  ZL=CIKVE ;
```

Todo List

- I have also done the purely vertical dipole "Kx=0 & Ky != 0" should be benchmarked at some point

- One could also add the CFM including a sextupole component

- Creating DA Maps from co, RE & TE, i.e. 0^{th}, 1^{st} and 2^{nd} terms as a benchmarking tool

- More implementations are sorely needed!!!

    - Tracking in SixTrack

    - MAD-X needs to be moved to Eq. 34 both in Twiss and Tracking!!! Presently, it is not generally symplectic!

    - Unfortunately, also true for PTC, where thin CFM are simply ignored!