MADX and Optics Examples

- Introduction to MADX
  - Input of elements and beam line
  - beta functions, tune
  - matching
- Examples
  - Collider lattice
  - SLC final focus
  - Synchrotron radiation sources

MADX Introduction

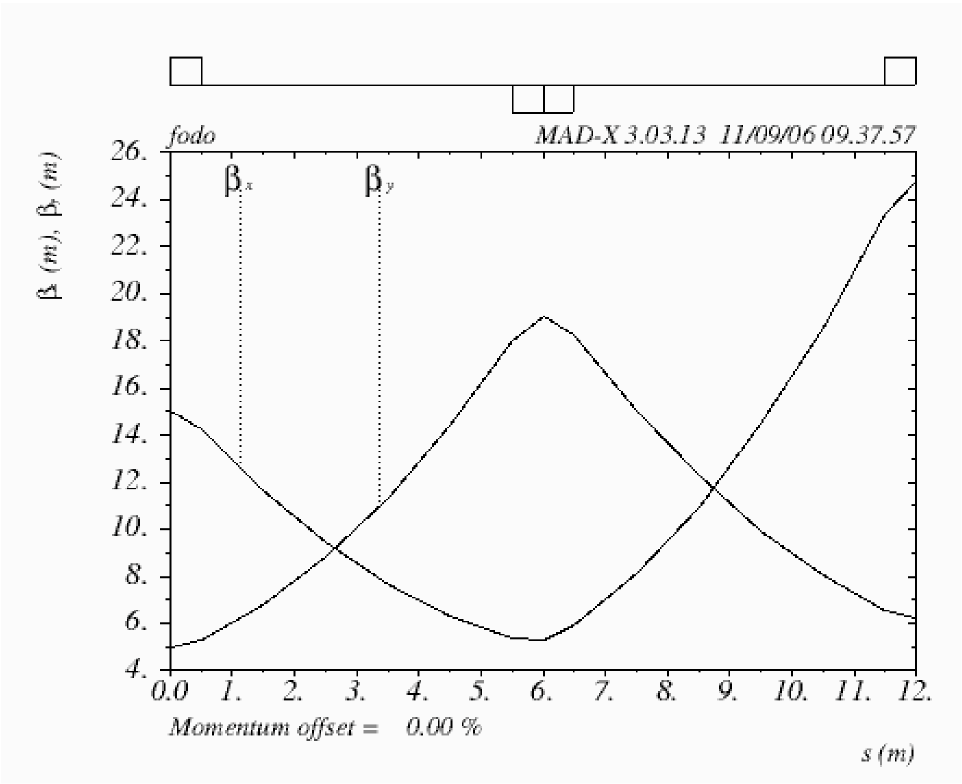
- Program to calculate beam optics
- Input magnets and their locations
- Derive beta functions, tune, dispersion, chromaticity, momentum compaction in numerical form
- Run in command window: madx < FODO.MADX
- Generates tables and plots (madx.ps)
- Might need to install ghostscript to view .ps file
- madx_manual.pdf is on the CD

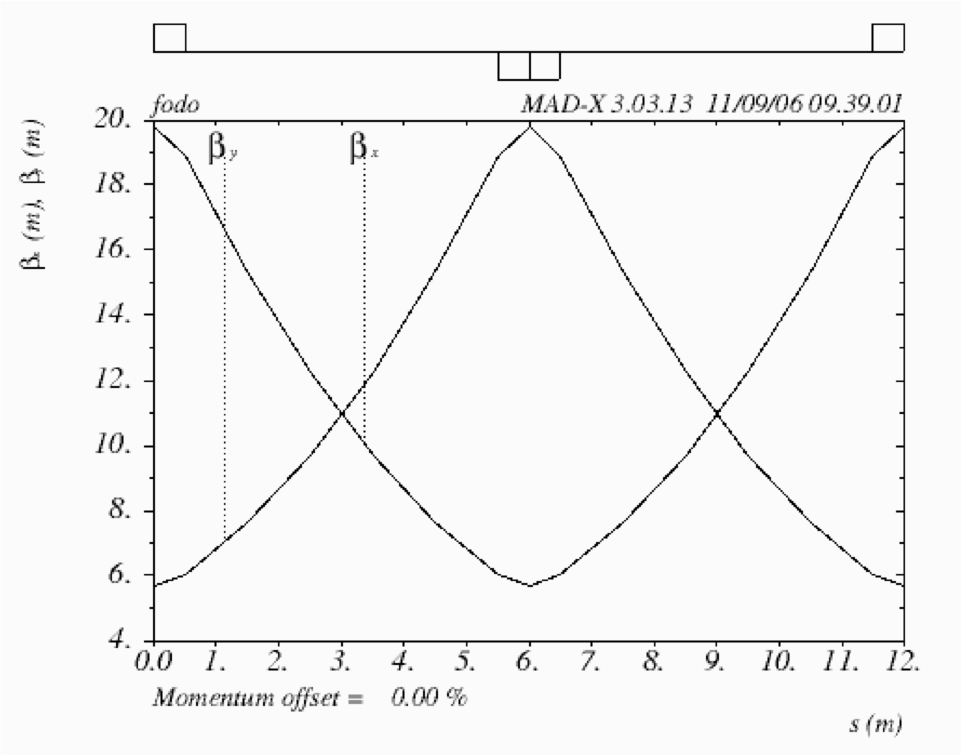
Simple example input file

```
//
// MADX Example 1: FODO cell
// Author: V. Ziemann, Uppsala University
// Date: 060910

TITLE,'Example 1: FODO.MADX';

BEAM, PARTICLE=ELECTRON,PC=3.0;

D: DRIFT,L=1.0;
QF: QUADRUPOLE,L=0.5,K1=0.2;
QD: QUADRUPOLE,L=0.5,K1=-0.2;

FODO: LINE=(QF,5*(D),QD,QD,5*(D),QF);
USE, PERIOD=FODO;

TWISS,SAVE,BETX=15.0,BETY=5.0;
PLOT,HAXIS=S, VAXIS=BETX, BETY;

MATCH, SEQUENCE=FODO;
PLOT,HAXIS=S, VAXIS=BETX, BETY;

Value, TABLE(SUMM,Q1);
Value, TABLE(SUMM,Q2);
WRITE,TABLE=SUMM,FILE=print.dat;
```

- Header and Title
- Define particle type and momentum
- Elements
- Beamline and USE it
- Calculate betafunctions from starting beta values and propagate and plot
- MATCH the periodic solution and plot it
- Print the tunes and a badly formatted table of all derived quantities

Result of the MADX run





- With betas at start set to 15 and 5 m
- Note lattice on top

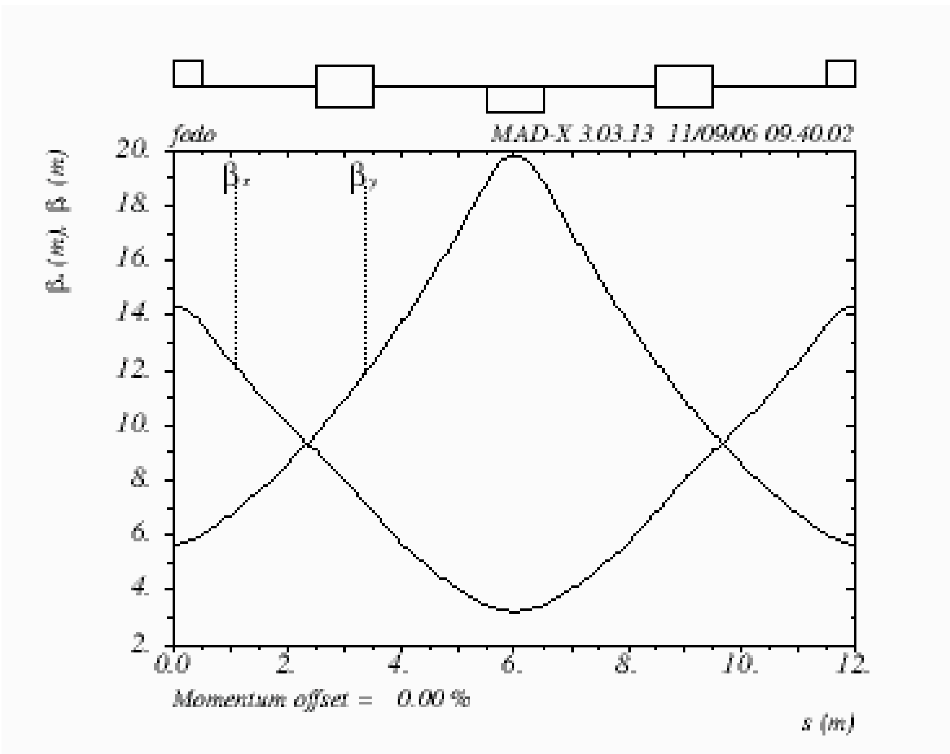
- Periodic solution
- beta_start=beta_end

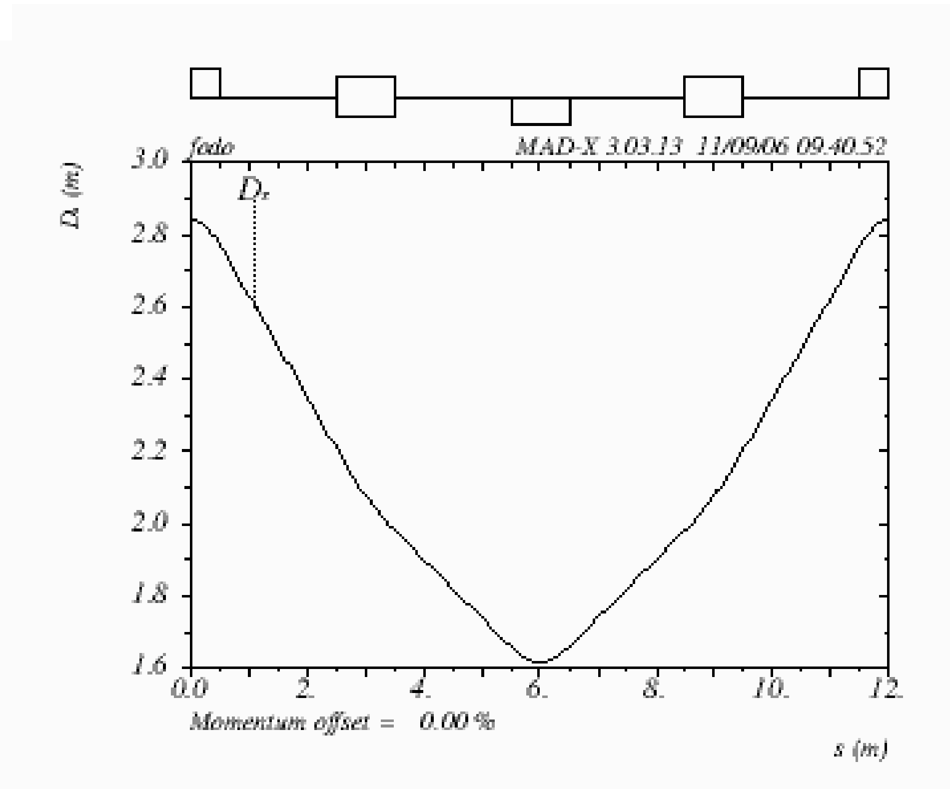
Now with bending magnets

```
// file: FODO2.MADX
// MADX Example 2: FODO cell with dipoles
// Author: V. Ziemann, Uppsala University
// Date: 060911

BEAM, PARTICLE=ELECTRON,PC=3.0;

DEGREE:=PI/180.0;                // for readability

QF: QUADRUPOLE,L=0.5,K1=0.2;     // still half-length
QD: QUADRUPOLE,L=1.0,K1=-0.2;    // changed to full length
B: SBEND,L=1.0,ANGLE=15.0*DEGREE; // added dipole

FODO: SEQUENCE,REFER=ENTRY,L=12.0;
  QF1:   QF,      AT=0.0;
  B1:    B,       AT=2.5;
  QD1:   QD,      AT=5.5;
  B2:    B,       AT=8.5;
  QF2:   QF,      AT=11.5;
ENDSEQUENCE;

USE, PERIOD=FODO;

MATCH, SEQUENCE=FODO;
PLOT,HAXIS=S, VAXIS=BETX, BETY, INTERPOLATE=TRUE;
PLOT,HAXIS=S, VAXIS=DX, INTERPOLATE=TRUE;
```

- Introduce parameters, but watch the :=
- Add dipoles
- Sequence format, no drifts needed, they are generated automatically
- REFER=ENTRY or CENTRE
- periodic solution
- plot beta functions
- plot horizontal dispersion
- interpolate

Results





- Periodic betafunctions
- SBEND adds horizontal focussing

- Periodic Dispersion
- Could calculate momentum compaction or chromaticity by hand from tables

Access to tabular values

- Add the following stanza to the MADX file to get tabular betafunctions and the transfer matrices

```
SELECT,FLAG=SECTORMAP,clear;
SELECT,FLAG=TWISS,column=name,s,betx,bety;
TWISS, file=optics.dat,sectormap;
```

- will write files optics.dat with betafunctions
- and a file called sectormap with the transfer matrices

```
@ SIGE       %le              0
@ SIGT       %le              0
@ NPART      %le              0
@ EX         %le              1
@ EY         %le              1
@ ET         %le              1
@ LENGTH     %le             12
@ ALFA       %le   0.0910331259
@ ORBIT5     %le             -0
@ GAMMATR    %le    3.314364527
@ Q1         %le   0.2909501025
@ Q2         %le   0.1913459932
@ DQ1        %le  -0.2152485772
@ DQ2        %le  -0.2276020063
@ DXMAX      %le     2.84272104
@ DYMAX      %le              0
@ XCOMAX     %le              0
@ YCOMAX     %le              0
@ BETXMAX    %le    14.36014477
@ BETYMAX    %le    18.85637615
```

```
* NAME                     S          BETX          BETY
$ %s                     %le           %le           %le
 "FODO$START"              0   14.36014477   5.674619595
 "QF1"                   0.5   13.67114589   6.007906055
 "DRIFT_0"               2.5   9.081076899   9.690010744
 "B1"                    3.5   6.880079877   12.25970476
 "DRIFT_0"               5.5   3.471640314   18.85637615
 "QD1"                   6.5   3.471640314   18.85637615
 "DRIFT_0"               8.5   6.880079877   12.25970476
 "B2"                    9.5   9.081076899   9.690010744
 "DRIFT_0"              11.5   13.67114589   6.007906055
 "QF2"                    12   14.36014477   5.674619595
 "FODO$END"               12   14.36014477   5.674619595
```

Matching

- Define constraints and magnets to vary to achieve an objective

```
MATCH, SEQUENCE=FODO;
CONSTRAINT,SEQUENCE=FODO,RANGE=#E,MUX=0.16666666,MUY=0.25;
VARY,NAME=QF->K1,STEP=1E-6;
VARY,NAME=QD->K1,STEP=1E-6;
LMDIF,CALLS=500,TOLERANCE=1E-20;
ENDMATCH;
```

- #E means at end of beamline
- mux is the phaseadvance in units of 2π
- QF->K1 means the K1 value of QF

The full matching file example

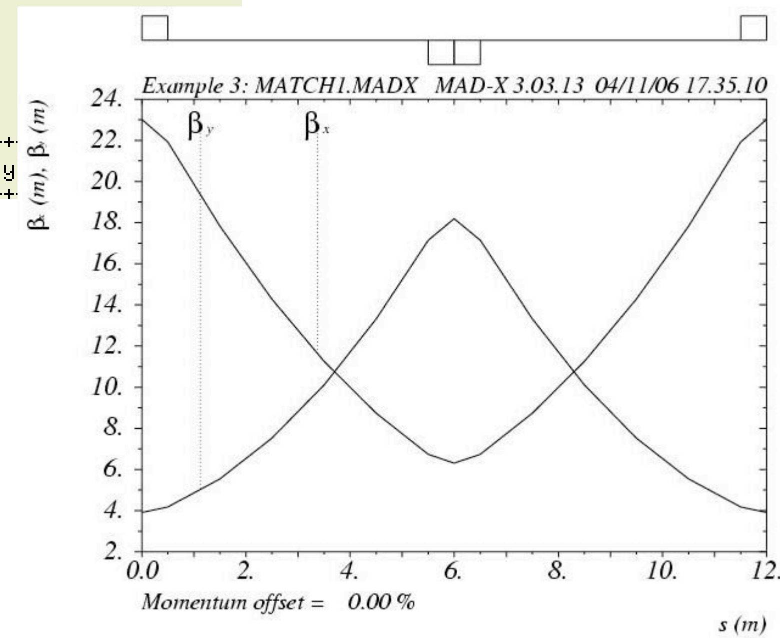
```
TITLE,'Example 3: MATCH1.MADX';

BEAM, PARTICLE=ELECTRON,PC=3.0;

D: DRIFT,L=1.0;
QF: QUADRUPOLE,L=0.5,K1:=0.2;
QD: QUADRUPOLE,L=0.5,K1:=-0.2;

FODO: LINE=(QF,5*(D),QD,QD,5*(D),QF);
USE, PERIOD=FODO;

//....match phase advance at end of cell to 60 and 90 degrees
MATCH, SEQUENCE=FODO;
CONSTRAINT,SEQUENCE=FODO,RANGE=#E,MUX=2*0.16666666,MUY=0.25;
VARY,NAME=QF->K1,STEP=1E-6;
VARY,NAME=QD->K1,STEP=1E-6;
LMDIF,CALLS=500,TOLERANCE=1E-20;
ENDMATCH;

SELECT,FLAG=SECTORMAP,clear;
SELECT,FLAG=TWISS,column=name,s,betx,alfx,bety,alfy;
TWISS, file=optics.dat,sectormap;


PLOT,HAXIS=S, VAXIS=BETX, BETY;

Value, TABLE(SUMM,Q1);  // verify result
Value, TABLE(SUMM,Q2);
```

- Define elements and beam line
- Match
- Make tabular output
- plot results

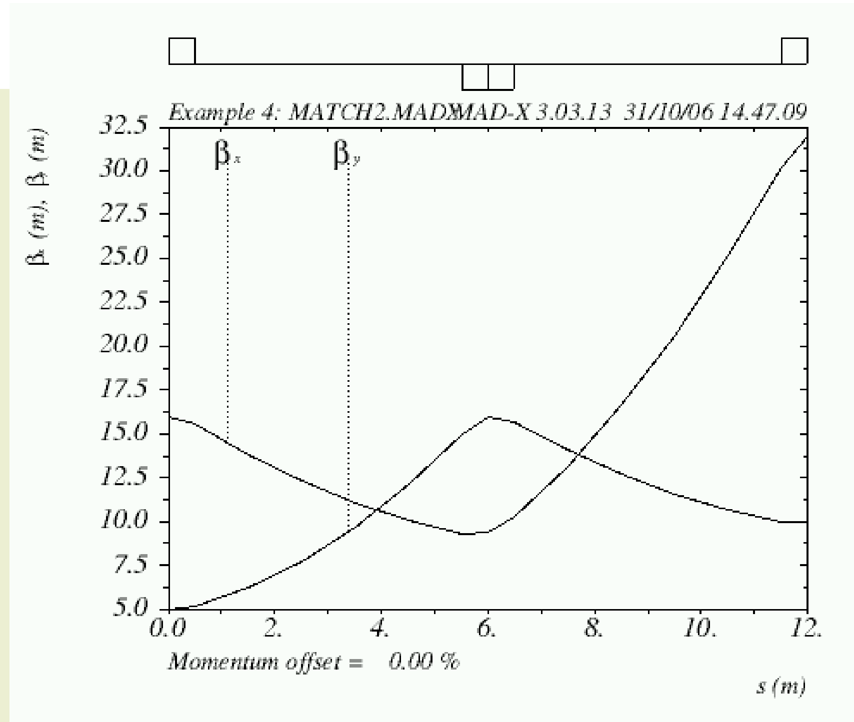
madx < MATCH1.MADX

```
START MATCHING

number of sequences: 1
sequence name: fodo
CONSTRAINT,SEQUENCE=FODO,RANGE=#E,MUX=0.16666666,MUY=0.25;

VARY,NAME=QF->K1,STEP=1E-6;

VARY,NAME=QD->K1,STEP=1E-6;

LMDIF,CALLS=500,TOLERANCE=1E-20;

number of variables:    2
user given constraints: 2
total constraints:      2

START LMDIF:

Initial Penalty Function =   0.40493620E+00


call:       4   Penalty function =   0.14656358E-02
call:       7   Penalty function =   0.22665315E-07
call:      10   Penalty function =   0.74450477E-16
call:      13   Penalty function =   0.29241225E-24
 ++++++++++ LMDIF ended: converged successfully
call:      13   Penalty function =   0.29241225E-24
ENDMATCH;


MATCH SUMMARY

Node_Name                  Constraint   Type  Target Value       Final Value        Penalty
--------------------------------------------------------------------------------------------------
fodo$end:1                 mux          4     1.66666660E-01     1.66666660E-01     2.92333363E-25
fodo$end:1                 muy          4     2.50000000E-01     2.50000000E-01     7.88860905E-29


Final Penalty Function =   2.92412249e-25



Variable                   Final Value       Lower Limit        Upper Limit
--------------------------------------------------------------------------------
qf->k1                     1.96577819E-01    -1.00000000E+20    1.00000000E+20
qd->k1                    -2.39111916E-01    -1.00000000E+20    1.00000000E+20

END MATCH SUMMARY
```

```
PLOT,HAXIS=S, VAXIS=BETX, BETY;

++++++ info: Zero value of SIGT replaced by 1.
++++++ info: Zero value of SIGE replaced by 1/1000.

 GXPLOT-X11  1.50 initialized

 plot number =  1


Value, TABLE(SUMM,Q1);  // verify result

table( summ q1 ) =         0.16666666 ;
Value, TABLE(SUMM,Q2);

table( summ q2 ) =               0.25 ;




  Number of warnings: 0

  ++++++++++++++++++++++++++++++++++
  + MAD-X 3.03.13 finished normally
  ++++++++++++++++++++++++++++++++++
```



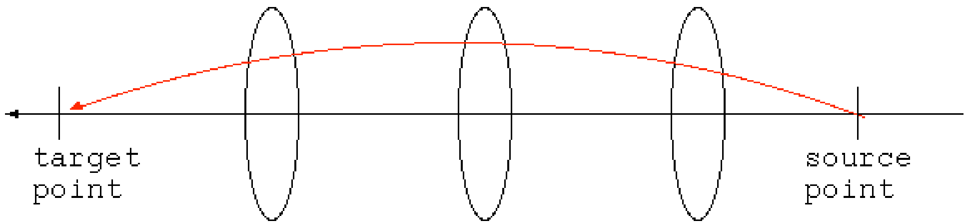
Fitting final beta functions




```
//
// MADX Example 1: FODO matching final beta function
// Author: V. Ziemann, Uppsala University
// Date: 061031

TITLE,'Example 4: MATCH2.MADX';

BEAM, PARTICLE=ELECTRON,PC=3.0;

D: DRIFT,L=1.0;
QF: QUADRUPOLE,L=0.5,K1:=0.2;
QD: QUADRUPOLE,L=0.5,K1:=-0.2;

FODO: LINE=(QF,5*(D),QD,QD,5*(D),QF);
USE, PERIOD=FODO;

MATCH, SEQUENCE=FODO,BETX=16,BETY=5;                // initial betas
CONSTRAINT,SEQUENCE=FODO,RANGE=#E,BETX=32,BETY=10; // final betas
VARY,NAME=QF->K1;
VARY,NAME=QD->K1;
LMDIF,CALLS=500,TOLERANCE=1E-20;
ENDMATCH;

PLOT,HAXIS=S, VAXIS=BETX, BETY;
```

Transfer-Matrix Matching

- Sometimes it is desirable to constrain transfer matrix elements to some value.
- For example R_{16}=0 and R_{26}=0 will make the horizontal position and angle independent of the momentum after a beamline.
- This is called an 'Achromat'.
- Other versions are imaginable point-to-point imaging → R_{12} = 0. This means sinμ = 0 or a phase advance of n x 180 degree

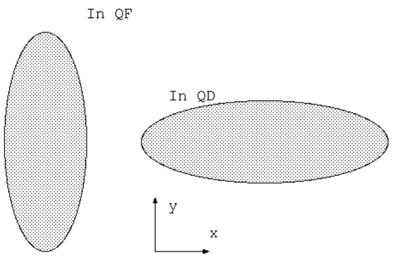


Beam Optics Examples

- FODO arcs
- Dispersion suppressor
- Telescope and mini beta
- SLC final focus
- Synchrotron radiation lattices
  - Emittance generation in electron machines
  - double bend achromat, triplet achromat, Triple bend achromat
- Bunch compressor chicane

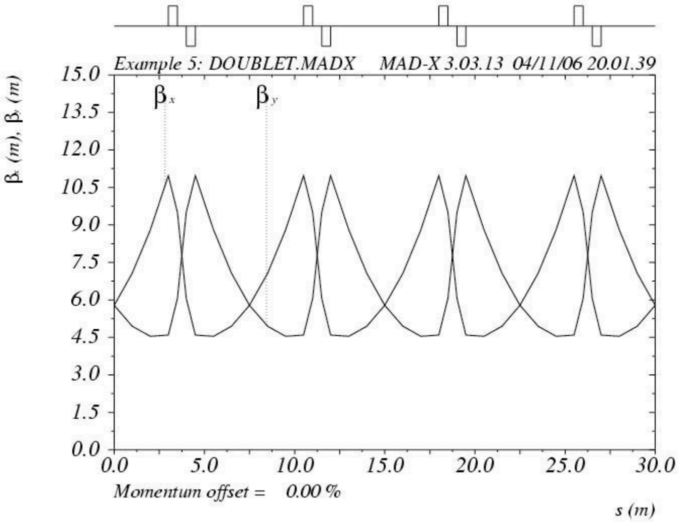
FODO arcs in colliders

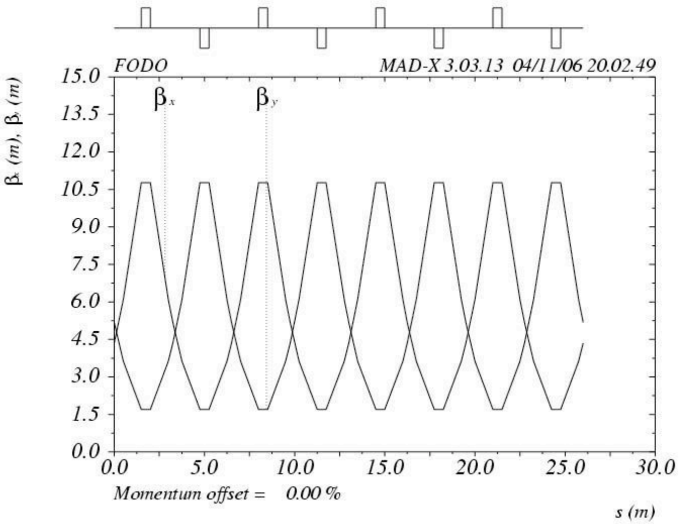
- Transport beam from interaction region to the next
- Simple
- moderate distance between quads
- Easily tunable because β_x large in QF and β_y large in QD
- Moderate excitation of the quadrupoles (power consumption)
- But beam is not round
- In arcs the dipoles will generate dispersion



Doublet lattice

- More space between quads
- but stronger quads
- round beams
- used in CTF3 linac

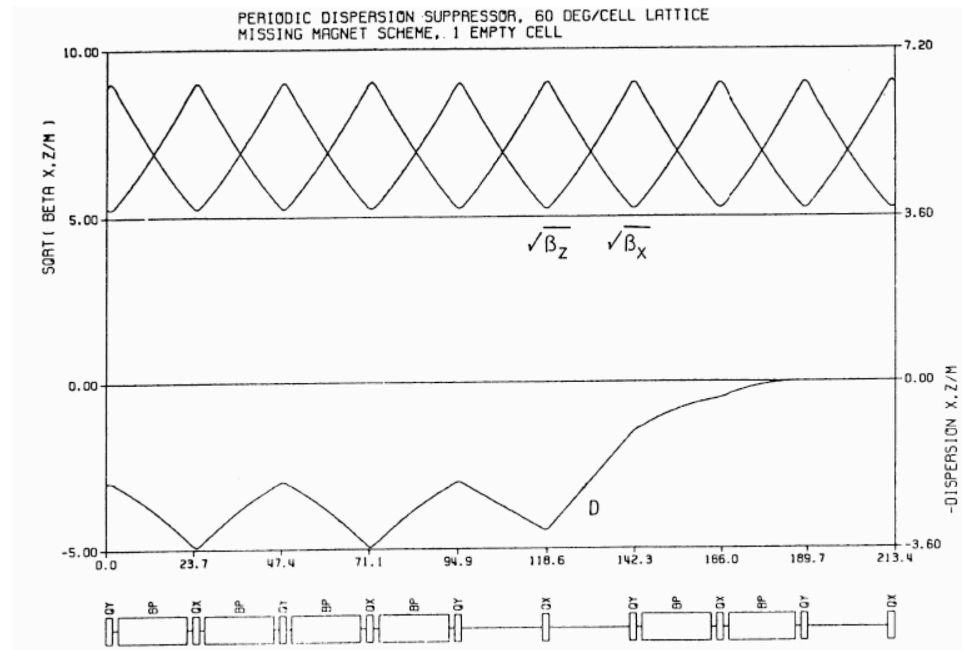




Dispersion suppessor

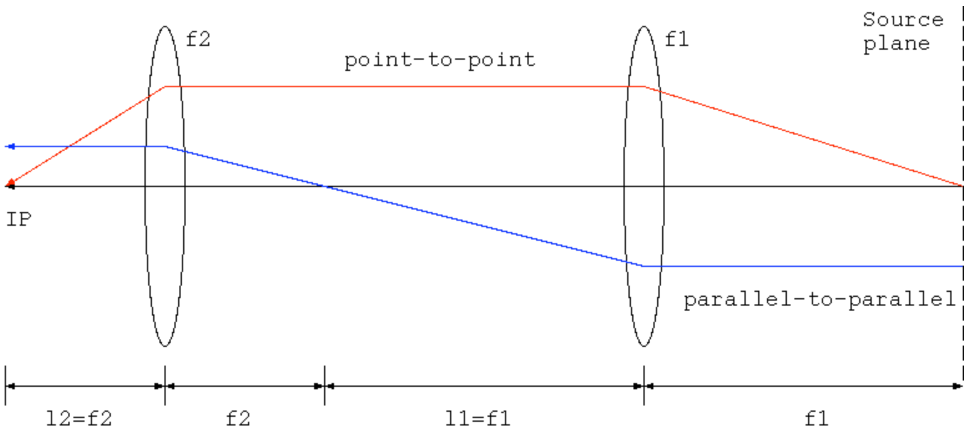
- Want small spot size at the interaction point and therefore minimize spot size

$$\sqrt{\varepsilon\beta + (D\Delta p/p)^2}$$



- Missing Magnet dispersion suppressor
- works with proper phase advance between elements

Telescope and low-beta



$$\begin{pmatrix} 1 & l_1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -1/f_1 & 1 \end{pmatrix} \begin{pmatrix} 1 & l_1 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 - l_1/f_1 & 2l_1 - l_1^2/f_1 \\ -1/f_1 & 1 - l_1/f_1 \end{pmatrix}$$

TM for one module with $l_1=f_1$:

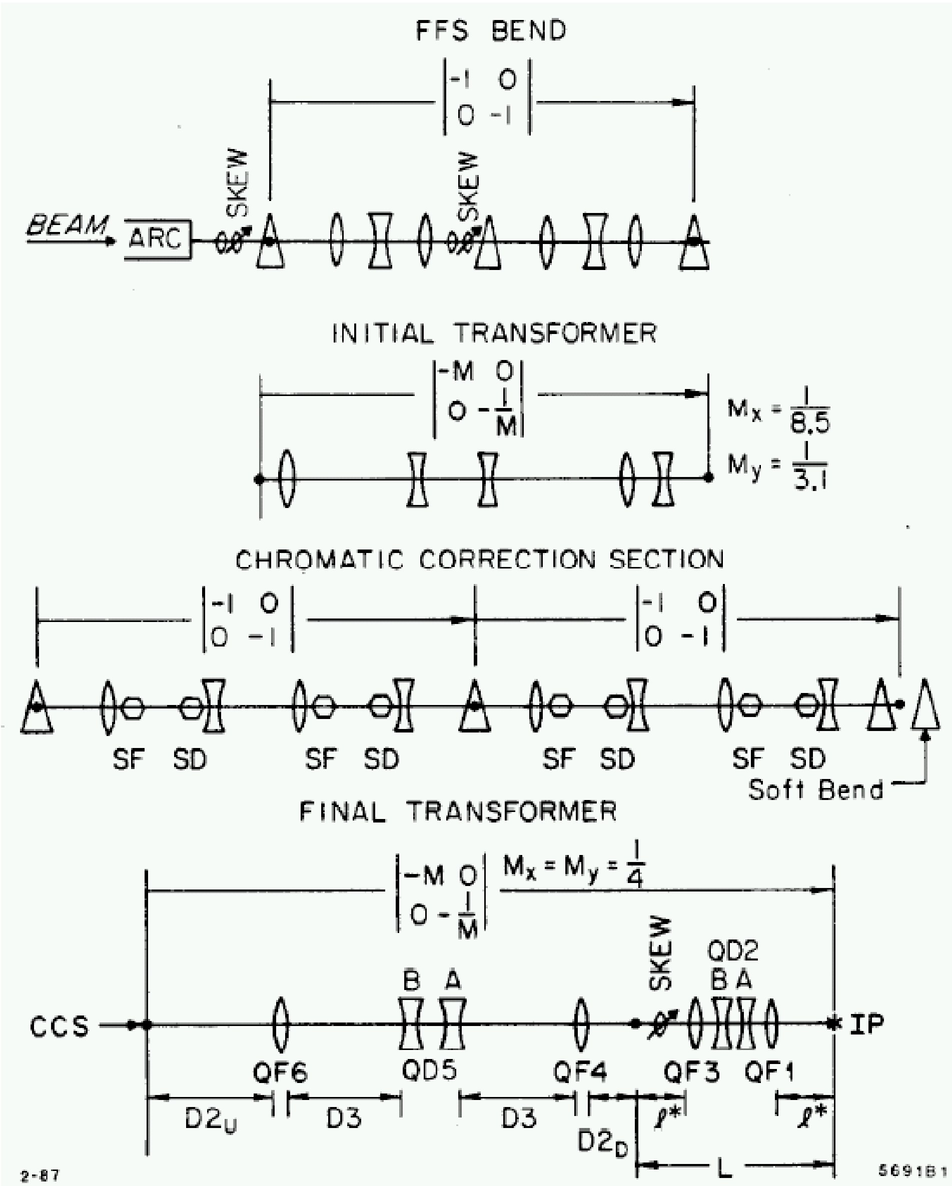
$$\begin{pmatrix} 0 & f_1 \\ -1/f_1 & 0 \end{pmatrix}$$

$$R = \begin{pmatrix} -f_2/f_1 & 0 \\ 0 & -f_1/f_2 \end{pmatrix}$$

TM for both modules

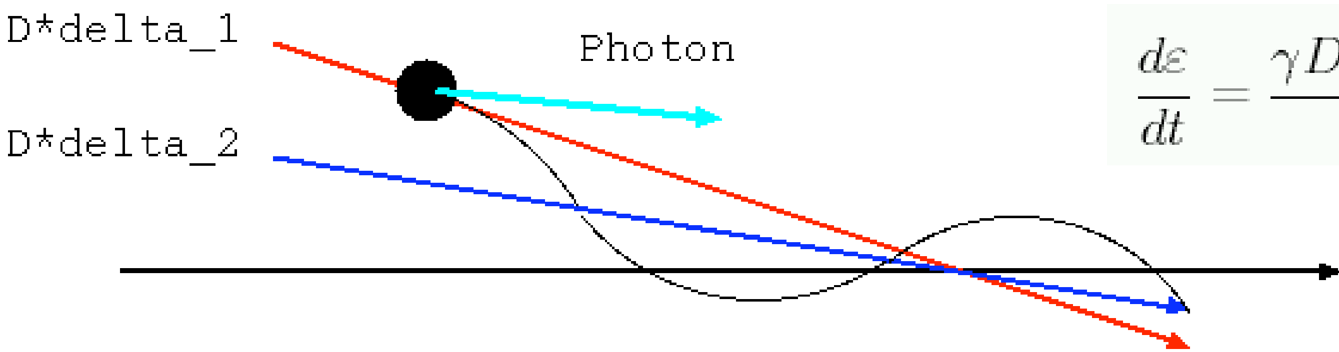
- After disperson suppressor and some matching quadrupoles we need to demagnify the beam
- Point-to-point $R_{12}=0$
- Parallel-to parallel: $R_{21}=0$
- R_{11} = demagnification
- ratio of focal lengths
- realize this to work in both transverse planes with douplets or triplets.

SLC Final Focus



- Dispersion suppressor, matching and skew correction
- Initial telescope
- Final telescope with strong triplets and large beta functions and skew quad
- Chromatic correction section with dipoles to generate dispersion and sextupoles

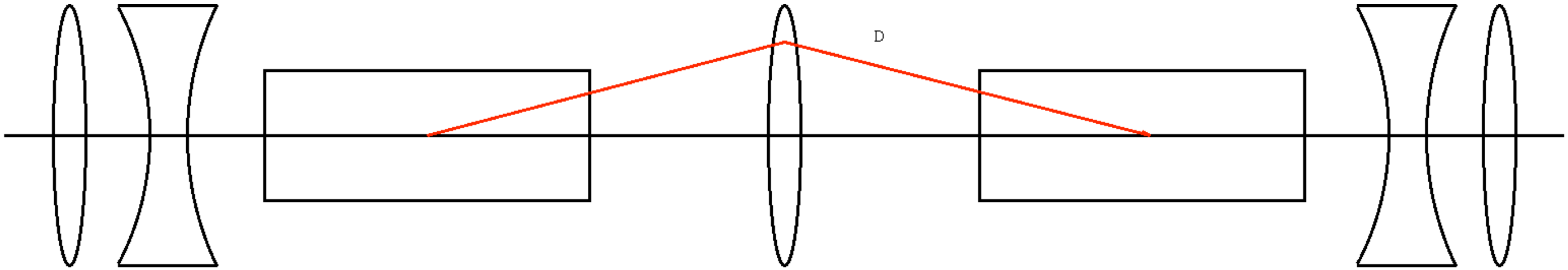
Emittance Generation



$$\frac{d\varepsilon}{dt} = \frac{\gamma D^2 + 2\alpha D D' + \beta D'^2}{T}\delta_{rms}^2$$

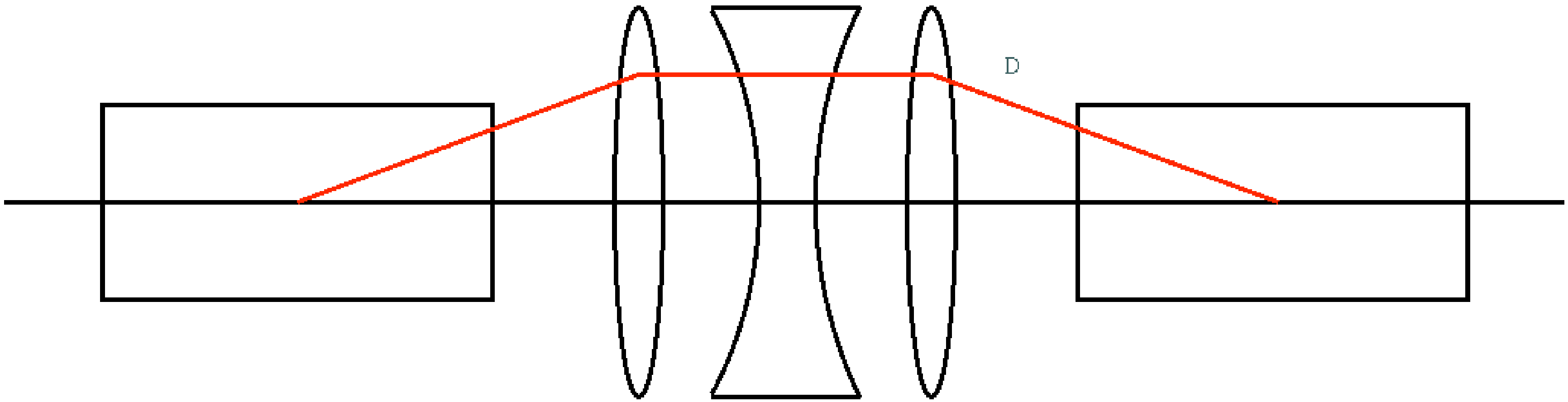
- Emission of s.r is a quantum mechanical statistical process that happens in dipoles
- Particle with momentum δ_1 revolves on Dispersion orbit $D\delta_1$
- If it emits a photon it gets different momentum δ_2 and its equilibrium orbit becomes $D\delta_2$
- But particle sits at 'old' position and starts betatron oscillations around the new quilibrium orbit.
- Effective kick is $\Delta\delta$ (D,D') which can be (incoherently = statistically independent kicks) summed up over many turns and yields the emittance growth rate
- Minimize Dispersion and betafunction where random energy loss occurs.

Double bend achromat



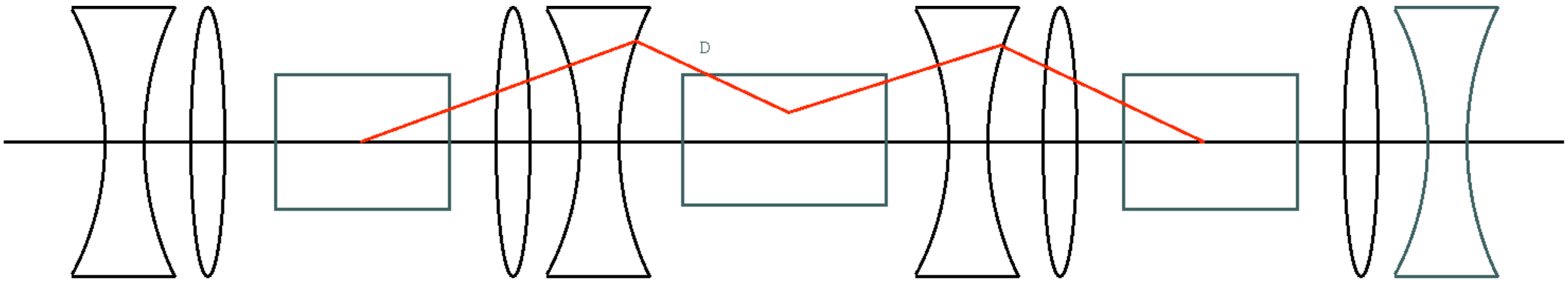
- One dipole generates dispersion and the next, which is 180 degrees apart will take it out again
- Remember: the dispersion is the orbit of a particle with slightly too high momentum w.r.t the reference particle
- Quadrupoles are used to make β_x in dipoles small

Triplet Achromat



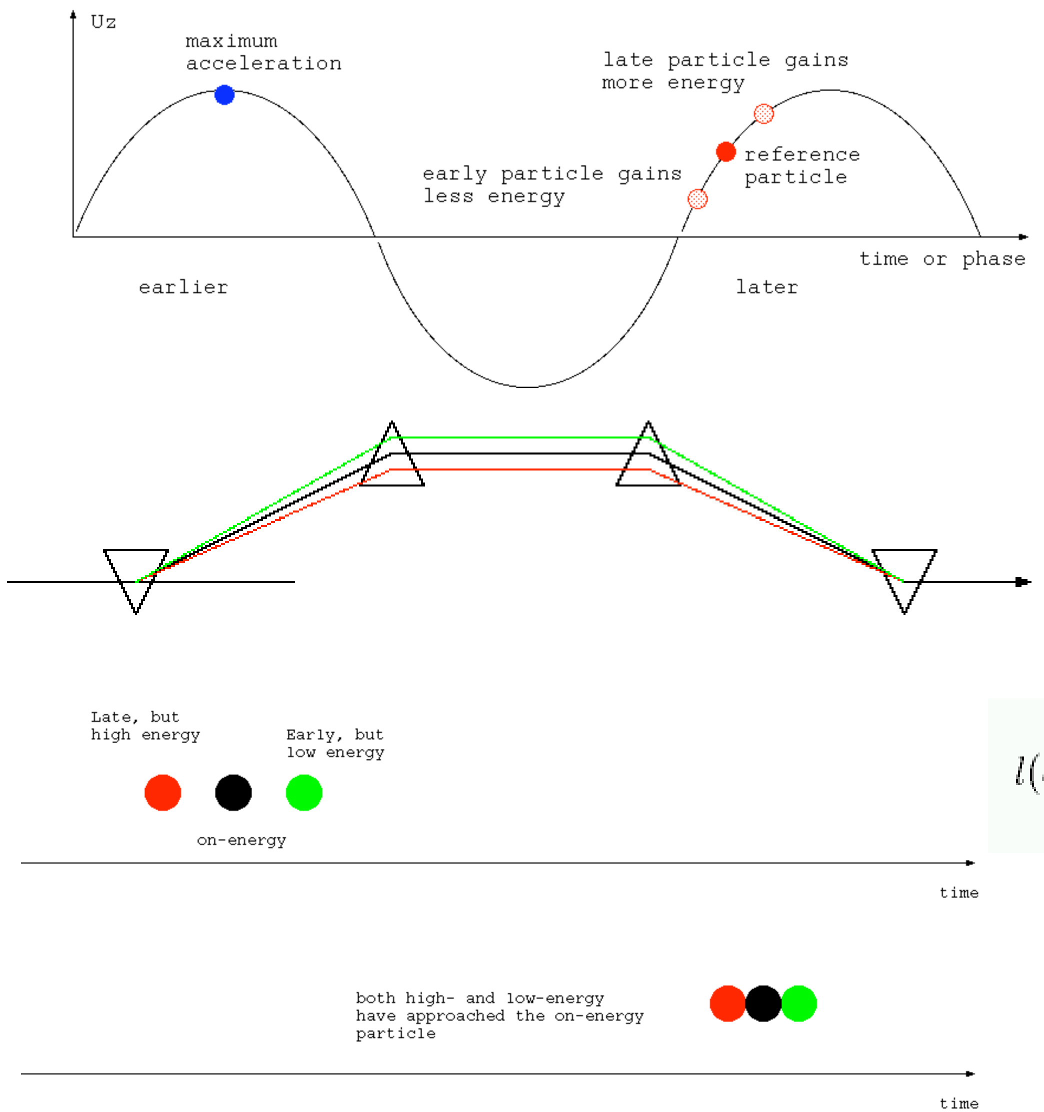
- Do the 180 degrees in the horizontal plane and the beta matching by quads between dipoles
- very compact, few magnets, but not flexible

Triple bend Achromat



- Small emittance.
- very flexible due to large number of quadrupoles.
- Adjacent drift space can be made long to accomodate undulators/wigglers.

Bunch compressor chicane



- Create correlated momentum spread by accelerating off-crest
- Pass beam through chicane with momentum dependent path length

$$l = \frac{2L}{\cos\phi} \approx \frac{2L}{1-\phi^2/2} \approx 2L\left(1+\frac{\phi^2}{2}\right)$$

$$l(\delta) = 2L\left(1+\frac{\phi^2}{2(1+\delta)^2}\right) \approx 2L\left(1+\frac{\phi^2}{2}\right) - 2L\phi^2\delta$$

$$R_{56} = \frac{l(\delta)-l(0)}{\delta} \approx -2L\phi^2$$