Summer School Student's Report

Pion beam optimization & Pion transport line design for parallel running of experiments

Student: Ho Quoc Trung
Institute for Interdisciplinary Research in Science and Education, ICISE, Quy Nhon, Vietnam

Project Supervisor: Dr. Jaroslaw Pasternak
Imperial College, London, United Kingdom

Abstract

In this study we aim for optimization of pion beam acceptance of out-going particles from a magnetic horn, and later build a pion transport line with optimized beam acceptance that allows for the separation of pions and kaons, which are then directed into nuSTORM and ENUBET experiments for parallel running both of these facilities. The current design of pion transport line is built using only three element types: quadrupole magnets for a FODO lattice to focus the beam, dipole magnets for generating an x-axis dispersion to extract 5 GeV/c pions from the beam of kaons and pions with designed momentum 6.25 GeV/c and the momentum spread of each type of particle beam of 10%. We find out that it is sufficient to generate a right amount of dispersion with a lattice consisting of five alternating dipole-quadrupole magnets.

1 Introduction

The nuSTORM (neutrinos from Stored Muons) ([1]) is a short-baseline neutrino project based on a low energy muon decay ring. In this experiment, a beam of ~ 100 GeV proton from the Super Proton Synchrotron (SPS) are used to produce pions off a conventional solid target, which later are collected and focused by magnetic horn and quadrupole magnets before being transported to a storage ring. This storage ring is optimized for utilizing a 3.8 GeV/c muon momentum with 15% momentum acceptance. This unique experiment is designed to serve as a mean for (see e.g [2]):

- Detailed studies of neutrino-nucleus scattering over the energy of both long- and short-baseline neutrino oscillations programs.
- searching for sterile neutrinos in the disappearance $\nu_e \to \nu_X$ and $\nu_\mu \to \nu_X$ channels.
- Providing a technology test-bed for the implementation of the next step in a muon-accelerator based particle-physics program.

Meanwhile, the ENUBET (Enhanced NeUtrino BEams from kaon Tagging) project that identifies positrons in Ke3 decays while operating in the harsh environment of a conventional neutrino beam decay tunnel (see e.g [3]). The neutrino beam produced by this experiment is a conventional narrow band beam with a short (~ 20 m) transfer line followed by a 40 m long decay tunnel. It is realized that kaon decays are particularly well suited for single-particle monitoring, hence the decay tunnel is optimized to have only one source of electron neutrinos in the process $K^+ \to \pi^0 e^+ \nu_e$.

With the similar structure of the two experiments, in the current study we aim to achieve two main goals:

- The transport line lattice could be constructed for parallel running of both nuSTORM and ENUBET, since both projects are meant to exploit the SPS proton beam of order $100 GeV$ to produce beam of neutrinos.
- The horn and the transport line should be designed to capture as most out-going particles as possible. We want to optimize the number of pions that go through the transport line, with the suitable matching of optical functions at the end point.

1.1 Optimization of pion beam from magnetic horn

Knowing the horn geometry and parameters, and the kinematics of the incoming beam of particles produced after the collision of the SPS proton beam with target, one can obtain the kinematics of the outgoing beam. In our current investigation, we use a magnetic horn based on the design of Ao Liu ([4]) and generate a beam of pions with specified position and momentum on 3D space. The task here is to design a lattice that minimize the number of lost pions, thus we need to vary the value of optical functions at the starting point of our transport line to see which range of values optimize the pion beam acceptance.

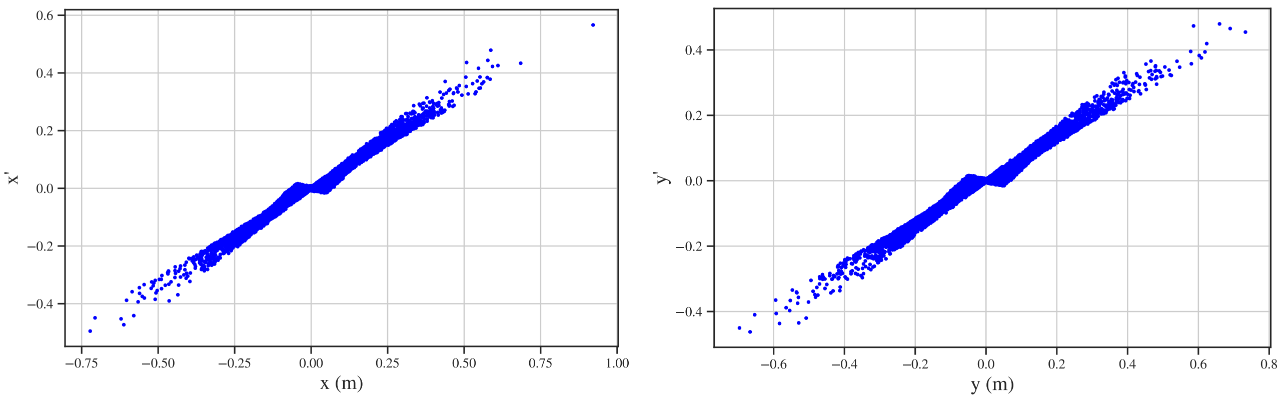

Figure 1: The beam distribution in phase-space (x, x') and (y, y') of an example output beam from the magnetic horn.

Given a beam distribution in phase-space of N particles, a constant emittance ε, and a maximum value of lost particles N_{lost} in both axes, we can point out a domain in the (α, β) plane where each point inside this domain gives rise to a phase space ellipse that enclose more than $N - N_{lost}$ particles. In the next study where we attempt to design a lattice for pion transport, we select the optical functions α and β within the optimized domain above as input to the initial optical functions of the twiss table. More details about the set-up for optimization of pion beam acceptance and design of the pion transport line will be presented in the next sections.

1.2 A new design for novel pion transport line front-end

The second goal of our study is to design the pion transport line that can separate the out-going beam from the horn into a beam of pions and another beam of kaons, which allows the capability of running in parallel both nuSTORM and ENUBET facilities.

Since the products after collision of SPS proton beams contain both pions and kaons, we came up with the idea of running both experiments at the same time by design a transport line that can separate the incoming particles into two beams depending on the designed momentum of each type of particles.

If such a lattice for transport line is designed, we gain a lot of benefits from economical point of view as well as saving space for building the experiments, with the fact that both experiments using the same target, magnetic horn and part of a transportline for producing desired beams and focusing them before redirect these beams into the decay tunnel. The schematic plot illustrating the idea is shown below, where we can redirect the pion beam into the nuSTORM storage ring, and bend the kaon beam toward the ENUBET transport line in order to produce beam of neutrinos for furthur studies.

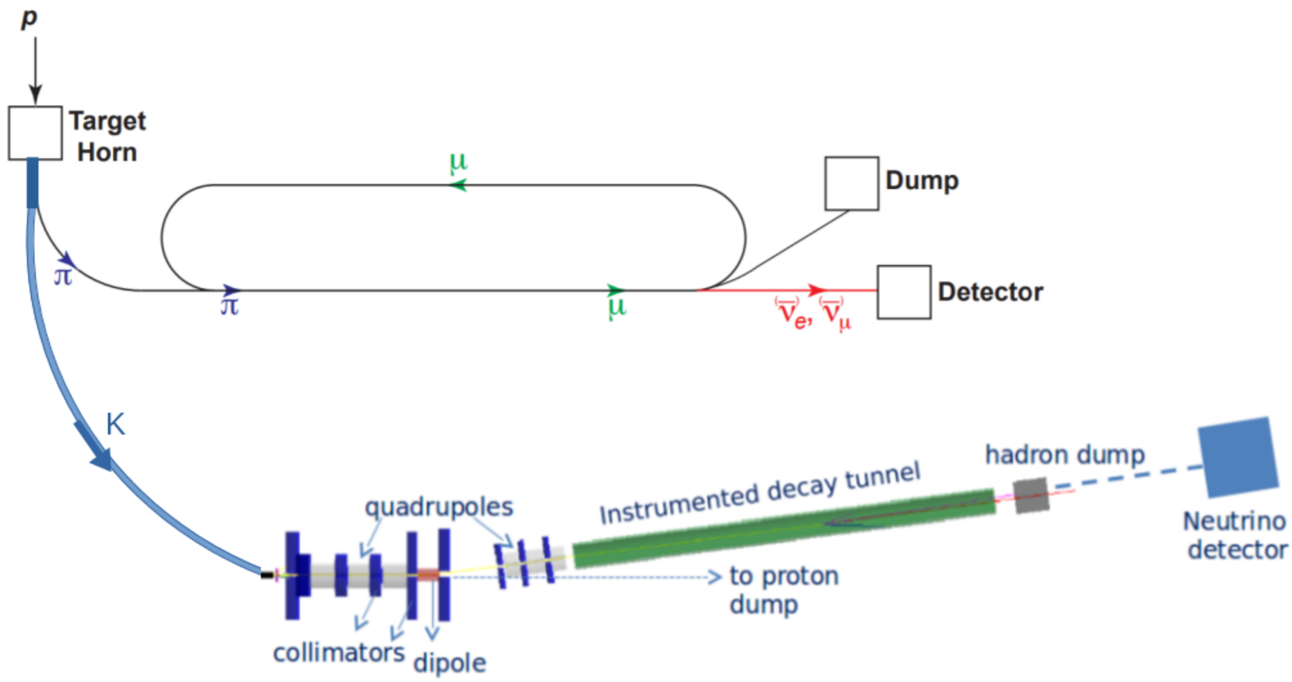

Figure 2: The desired lattice allows parallel running of both nuSTORM and ENUBET experiment. This is rought sketch to describe the idea, and the scale is not realistic. (figure taken from [2, 5]

2 Theoretical and Calculation Framework

2.1 Preliminaries: Coordinates system and conventions

For consistency with numerical calculations, we use the MAD-X ([6]) local curvilinear right-handed coordinates system. Each point in 3D space is now parameterized by the tripod (x, y, s) where the x-axis is in the bend plane and perpendicular to the reference orbit, y-axis is perpendicular to the bend plane and s-axis is the tangent of the reference orbit.

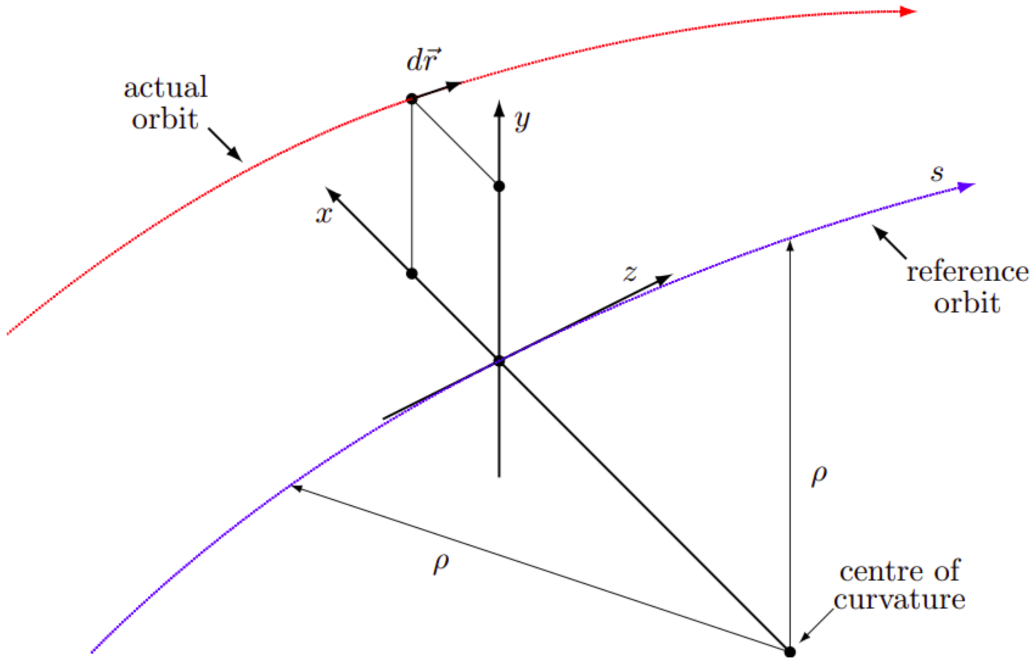

Figure 3: MAD-X local reference frame (x, y, s). Source: Figure 1.1 of [6].

2.2 phase space ellipse

In principle, if the initial condition (x_0, x'_0, y_0, y'_0) of a particle is given then one can solve for the trajectory along a beam line. This is impossible if we consider a large collection of a particles, and here comes the powerful usage of statistical mechanics to describe a particle beam as a whole. Using Liouville's theorem, one can prove that the phase-space density of a beam remains constant along the beam trajectory once the applied forces are originated from macroscopic electromagnetic fields. We can further prove that each trajectory of an arbitrary particle inside the beam traces its individual ellipse in phase-space with the same shape and orientation, but possibly with a different size compared to other particles. Hence, it is customary in beam dynamics to capture a collection of particles inside a beam in a phase space ellipse described by the following equation:

$$\gamma_u u^2 + 2\alpha_u u u' + \beta_u u'^2 = \epsilon_u, \quad (u = x, y) \tag{2.1}$$

with the so-called Twiss parameters $\alpha_u(s)$, $\beta_u(s)$ and $\gamma_u(s)$ [1] can be solved if the specific structure of the lattice and the beam emittance are given.

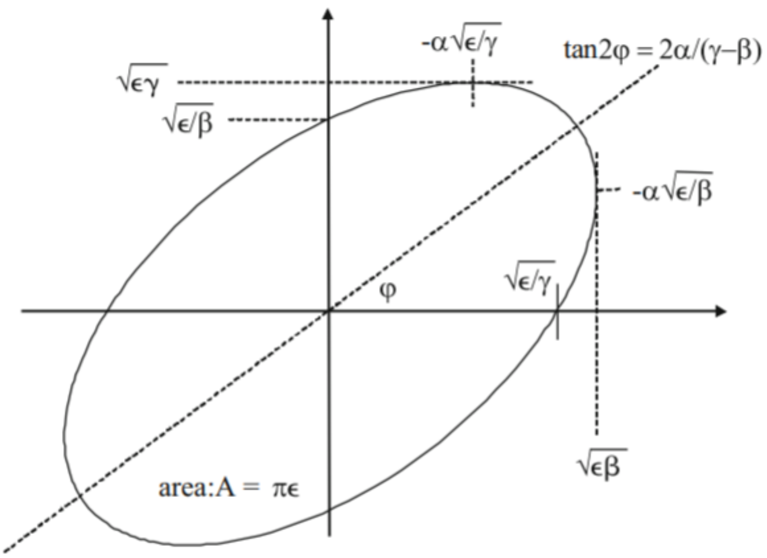

Figure 4: Phase-space ellipse parameterized by four parameters α, β, γ and ϵ. Source: Fig 8.2 in [7].

[1] These three parameters are not independent since $\gamma\beta - \alpha^2 = 1$

Since all particles enclosed by a phase space ellipse remain in the interior of the ellipse along the trajectory. Given a distribution of particles in a beam, we can thus design a lattice whose optical functions at the initial point corresponding to a ellipse that enclose a large portion of the particles in phase-space, hence optimize the beam acceptance of the lattice.

3 Numerical Results and Visualization

3.1 A calculation framework in designing the pion transport line

In our numerical calculations, the beam emittance on x-axis and y-axis are fixed, the beam distributions in phase-spaces are derived from the output beam profile from the magnetic horn. One can then set-up a transport line from basic elements (e.g drift spaces, quadrupole magnets and bending dipoles) and calculate the Twiss variables along this lattice using a public package, namely MAD-X. A brief summary of the MAD-X's usage are captured in the following block diagram.

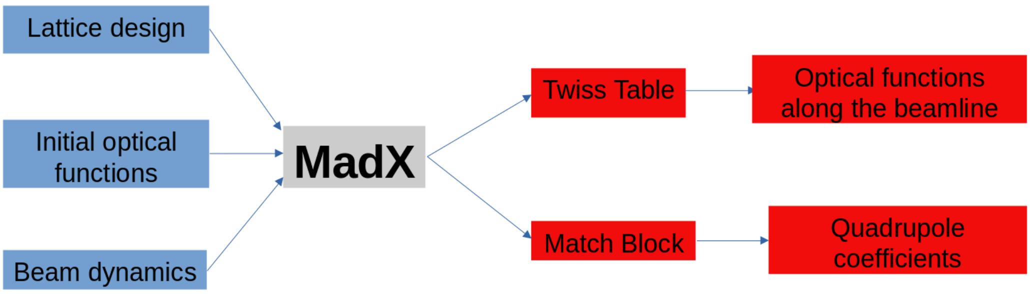

Figure 5: A diagram of our set-up in designing the transport line.

Main input components of MAD-X including:

- **Lattice design**: using the LINE command to order the elements in the lattice. In our current design, only the MARKER, DRIFT and QUADRUPOLE are used, with the quadrupole coefficients are not fixed and can be find using VARY command with certain constraints on the optical properties.
- **Initial optical functions** including $\alpha_x(0)$, $\alpha_y(0)$, $\beta_x(0)$ and $\beta_y(0)$ are selected to capture as much pions from the magnetic horn as possible.
- **Beam dynamics** specified by the BEAM command, where we fix the variables ENERGY=4.86238 (GeV) and energy spread SIGE=4.5×10^{-4}.

The output of MAD-X are values of linear lattice functions (calculated using TWISS command) and the quadrupole coefficients (using VARY command within a MATCH block, with all of the constraints and optical matches are specified inside this block). For a flexibility to manipulate the parameters as well as handling data, we mainly use a private Python code for all calculations and visualizations, with a helping hand of the package `cpymad` as an interface between Python and MAD-X. A more complete description of the MAD-X utilities and basic usage of `cpymad` is given in Appendix A.

3.2 Results from optimization of pion beam acceptance

In the following results, we take the input beam of the form `(# x, y, z, Px, Py, Pz, t, PDGid, EventID, TrackID, ParentID, Weight)`, which contains information about the positions and momentums of pions. This beam profile is produced by tracking in the magnetic field of the Ao Liu's design of the horn. The distribution of this beam is shown in the figure below.

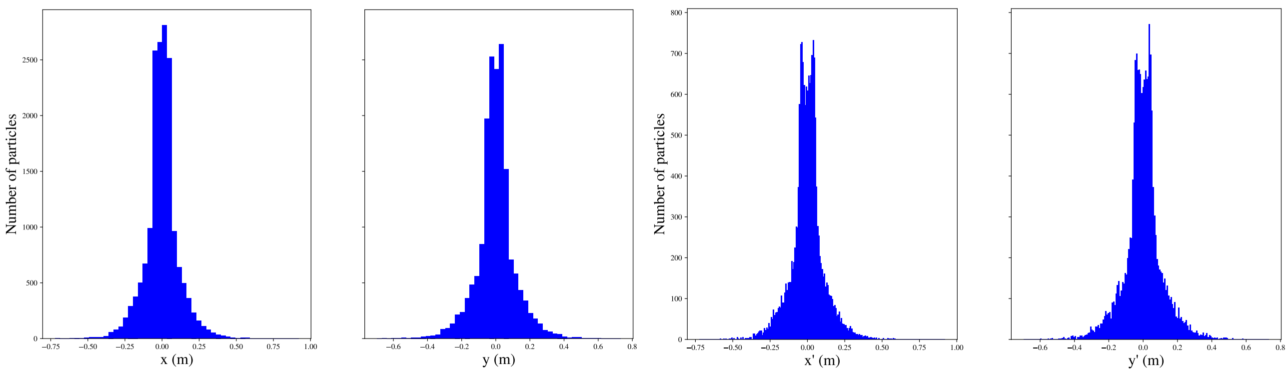

Figure 6: The Gaussian-like frequency plot of pions after being focused by the magnetic horn.

Since magnetic horn treats the particle beam similarly in both x and y direction, the distribution is seemingly the same for x and y axis, as well as x' and y' axis. Hence, we thus only need to optimize the beam acceptance in one direction and the other one will be optimized automatically.

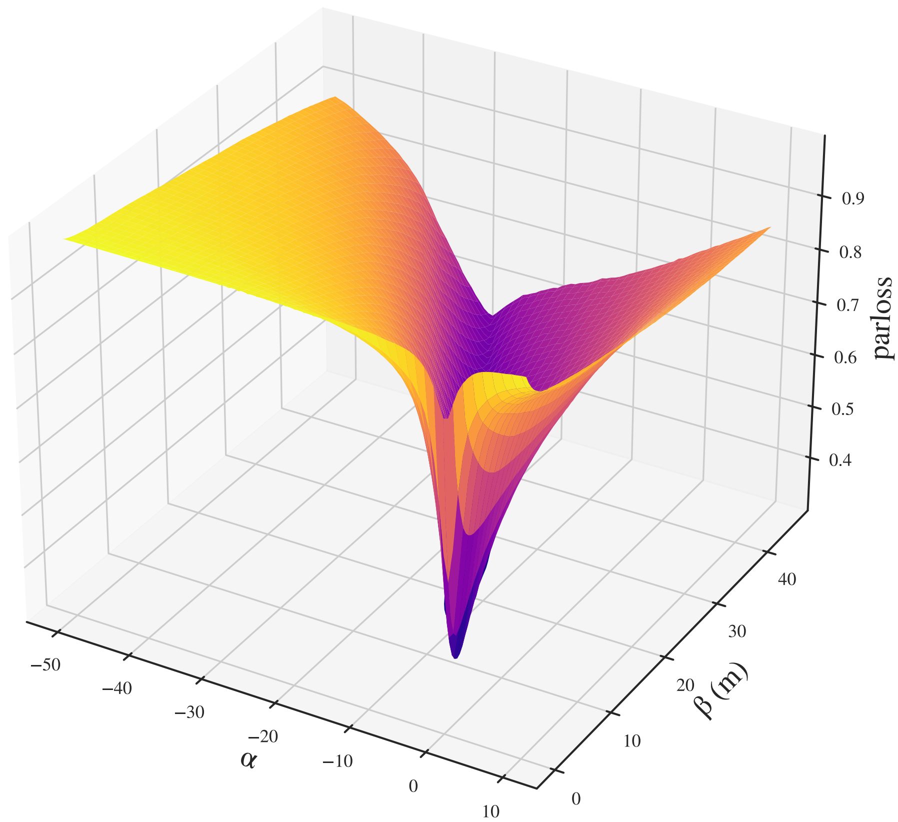

Figure 7: Number of lost particles (those that are not captured by the phase-space ellipse) divided by total particles (named as parloss on the z-axis), with respect to the initial optical functions α and β. This result is obtained after analyzing the magnetic horn output particles using private Python code.

In Fig. 7, we show the visualization of the lost particles with respect to the parameters α ranges from -50 to 10, and β ranges from 0 m up to 45 m. The lost function attains a minimum at around $(\alpha, \beta) = (-1.75879397, 4.38693467)$ and remains small around a line in the (α, β) plane. This can be seen clearly in Fig. 8.

Since the particle distribution after being focused by the horn has a z pattern that is squeezed in a small portion of phase-space (see Fig. 1), the main axis of the phase-ellipse should be oriented alongside this distri-

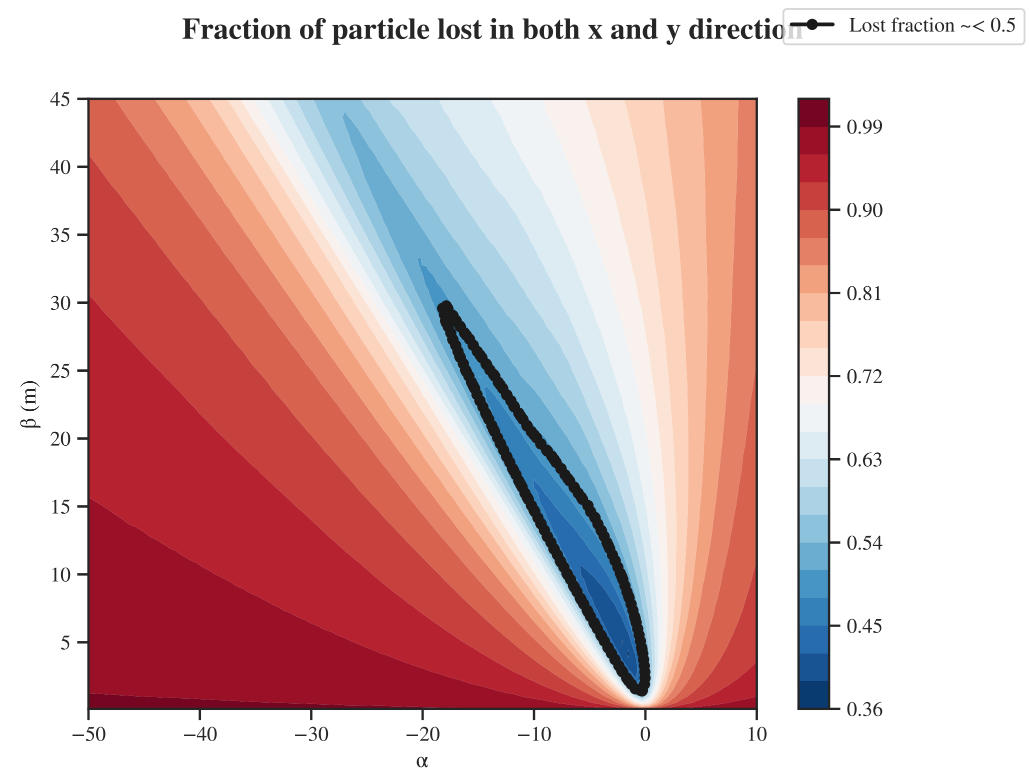

Figure 8: Heat map of the normalized lost particles with respect to the initial optical functions α and β. The black closed contour enclose a region where less than half of the pions are lost.

bution with a small variation when twiss parameters are varied. The collection of points (α, β) that minimize the loss functions when β is fixed traces a curve (that looks like somewhat linear) in the (α, β)-plane, and those points should corresponds to nearly the same orientation of the main-axis of phase space ellipse. This means that the darker region in Fig. 8 should distribute around a curve parameterized as (see Fig. 4)

$$\tan 2\varphi = \text{para} \rightarrow \frac{2\alpha\beta}{1+\alpha^2-\beta^2} = \text{para}, \tag{3.1}$$

where para depends on the phase-space distribution of particle beam, and varying closely around tangent of twice of the angle between the least square regression line of the particle distribution and the x-axis.

As an example, we select a point which minimizes the particle lost function and draws the corresponding ellipse in the phase space. It is obvious that the major axis of the ellipse is nearly parallel with the least squares regression line of the particle distribution; this explains why the darker region in Fig. 8 (which optimize the beam acceptance) lie in the negative part of α.

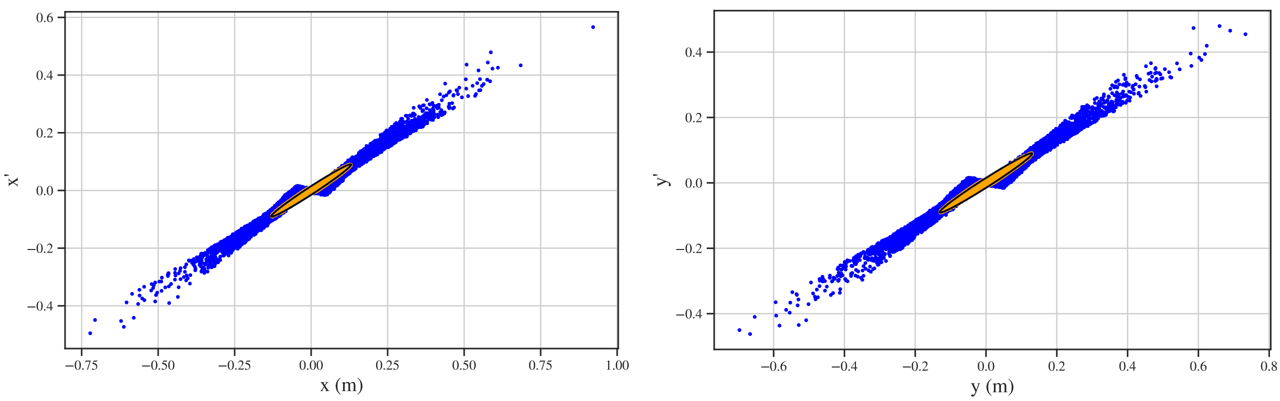

Figure 9: The phase space ellipse that has been selected to captured maximum number of particles, where we varied the initial α and β parameters, and keep the emittance unchanged in our design. This plot is produced by a private Python code.

In the next goal where we need to build a pion beam transport line, we can select points within the region closed by the black contour in Fig. 8 as the initial optical functions to calculate the twiss table.

3.3 Results on the design of pion transport line

Hereafter in this subsection, we attempt to design the pion beam transport line that has well-behaved optical properties that allowed the separation of pions and kaons, which can then be redirected to two other facilities for parallel running. As a starting point, let us consider the following constraints on the beam optics:

- **Constraint on beam envelope size**: Since the size of realistic quadrupoles are limited, in order to prevent the particles to collide with the elements and being lost, we impose a constraint on the maximum value of β. With the emittance $\varepsilon = 2$ mm and the diameter of a quadrupole $d = 30$ cm, the corresponding upper boundary is: $\beta < \beta_{max} = 45$ m.
- **Matching constraint at the end of the lattice**: We want to keep the rest part of the nuSTORM lattice as in the current design, hence we need to match the optical functions at the end of the pion transport line, specifically:

$$
\begin{aligned}
\beta_x &= 5.903272532 \ m, \\
\beta_y &= 8.378419334 \ m, \\
\alpha_x &= 1.490004715, \\
\alpha_y &= -1.241069731.
\end{aligned}
$$

Let us assign the variable `nustormbl` as the designed lattice, m_i and m_f are the marker at the starting point and end point of the lattice respectively. We rewrite the above two constraints in the language of MadX as:

```
  constraint, sequence=nustormbl, range=mi/mf, betx<45., bety<45.;
  constraint, sequence=nustormbl, range=mf, betx=5.903272532, alfx=1.490004715,
                                         bety=8.378419334, alfy=-1.241069731;
```

Listing 1: Constraints for current lattice

We calculate the twiss table for the transport line with the beam dynamics defined as

```
Beam, particle = pion, sequence = nustormbl, energy = 4.86238, NPART=1.05E11, sige=4.5e-4 ;
```

Listing 2: Beam dynamics

The first transport line's design considered took advantages of the solenoids to capture the beam of charged particles from the target. Since the solenoid lens is quite sensitive to the value of the beam momentum, [1] such a design for a 5 GeV beam of pions requires a $\sim 10 - 15$ (T). This requirement is impractical, and we proceed our design with a horn followed by a typical FODO lattice including four quadrupole magnets and drift spaces in between as follows:

```
  mi: marker;
  o1: Drift, L=1.8;
  Q1: Quadrupole, L=1., K1:=k11;
  o2: Drift, L=2.0;
  Q2: Quadrupole, L=1., K1:=k21;
  o3: Drift, L=2.;
  Q3: Quadrupole, L=1., K1:=k31;
  o4: Drift, L=2.;
  Q4: Quadrupole, L=1.0, K1:=k41;
  o5: Drift, L=2.;
  mf: marker;
  nustormbl:line=(mi, o1, Q1, o2, Q2, o3, Q3, o4, Q4, o5, mf);
```

Listing 3: Four quadrupoles transport line

[1] The focal length of a solenoid depends on the squared of momentum as can be seen in the formula $\frac{1}{f_{sol}} = \int \left(\frac{eB}{2p}\right)^2 ds$.

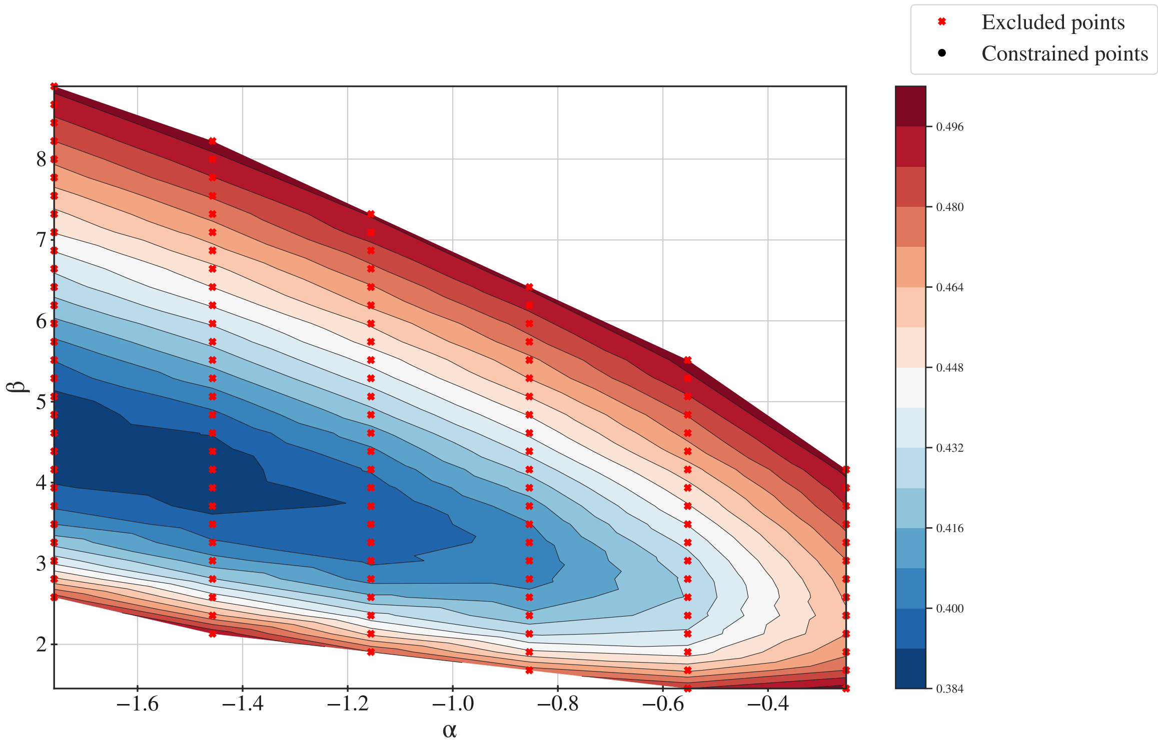

Figure 10: Illustration of initial optical functions (α, β) with the corresponding twiss tables satisfying the given constraints.

In Fig. 10 the red cross mark are values of α and β at the beginning of the transport line that do not satisfy the given constraints, and the black dots are those that satisfy. We see that a system of four quadrupoles do not have enough degrees of freedom to vary in the MATCH block, and that we need to add more elements into this lattice.

Next, let us improve the origin design minimally by adding one quadrupole into the starting lattice. The total length of this design is 13.9 m, which is close to the length of the old lattice.

```
mi: marker;
o1: Drift, L=1;
Q1: Quadrupole, L=1.5, K1:=k11;
o2: Drift, L=1.2;
Q2: Quadrupole, L=1.5, K1:=k21;
o3: Drift, L=1.2;
Q3: Quadrupole, L=1.5, K1:=k31;
o4: Drift, L=1.2;
Q4: Quadrupole, L=1.5, K1:=k41;
o5: Drift, L=1.2;
Q5: Quadrupole, L=1.5, K1:=k51;
o6: Drift, L=0.6;
mf: marker;
nustormbl:line=(mi,o1,Q1,o2,Q2,o3,Q3,o4,Q4,o5,Q5,o6,mf);
```

Listing 4: Five quadrupoles transport line

This gives more flexibility in varying the quadrupole coefficients and possibly more initial optical functions are allowed, which is represented in Fig. 11. The quadrupole coefficients calculated using the constraints on beam envelope size and matching at the end are given below:

$$k11 = -0.282425, \quad k12 = 0.254574, \quad k32 = -0.463095, \quad k42 = 0.341654, \quad k52 = -0.199937.$$

We note that most of the allowed initial optics are concentrated on the region with small $|\alpha|$, hence we show here part of the scanning region where $\alpha > -2$.

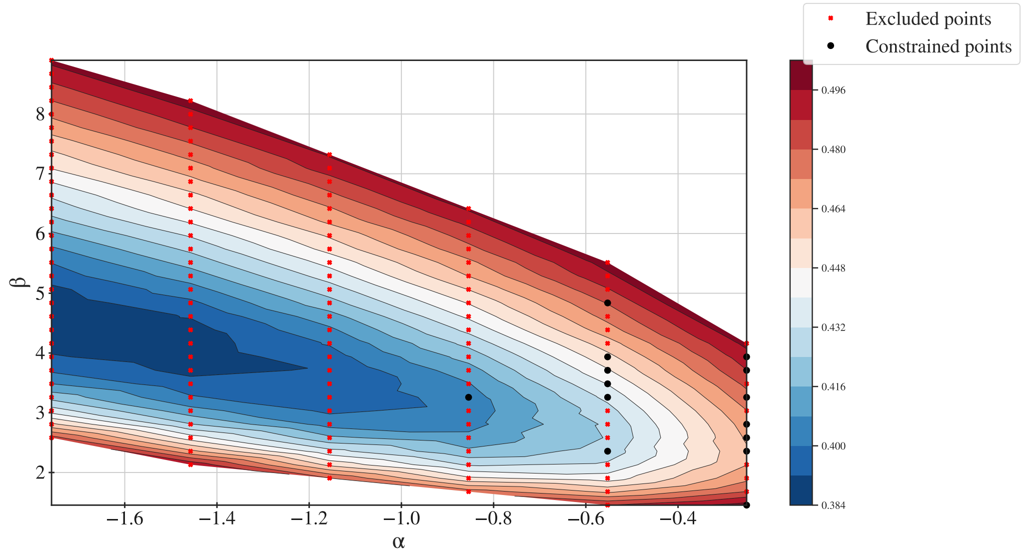

Figure 11: Illustraion of initial optical functions (α, β) for the lattice of 5 quadrupoles.

With the original idea of a lattice that allows the separation of pion beam and kaon beam, the x-axis dispersion should be large enough. Given the designed momentum $p = 6.25$ GeV, the pion beam before the injection into storage ring is optimized with momentum 5 GeV/c, the designed momentum of the kaon beam is 7.5 GeV/c and the momentum spread of both beam is 10%, in order to separate these two beams we need to generate a dispersion $D_x \geq 1.76$ m. As an ansatz, we considered the insertion of one dipole into the system and realized that it is harder to constraint the optical functions this way. We thus come up with the idea of implementing an alternating system of five dipole-quadrupole into the lattice, which allows a broader range of initial conditions for TWISS command, and the dispersion can be generated more effectively compared to the case of only one long dipole.

```
mi: marker;
o11: Drift, L=1;
Q1: Quadrupole, L=1., K1:=k11;
o12: Drift, L=0.2;
ds1: Sbend, L=1.5, angle=0.108
o21: Drift, L=0.2;
Q2: Quadrupole, L=1., K1:=k21;
o22: Drift, L=0.2;
ds2: Sbend, L=1.5, angle=0.108
o31: Drift, L=0.2;
Q3: Quadrupole, L=1., K1:=k31;
o32: Drift, L=0.2;
ds3: Sbend, L=1.5, angle=0.108
o41: Drift, L=0.2;
Q4: Quadrupole, L=1., K1:=k41;
o42: Drift, L=0.2;
ds4: Sbend, L=1.5, angle=0.108
o43: Drift, L=0.2;
ds5: Sbend, L=1.5, angle=0.108
o51: Drift, L=0.2;
Q5: Quadrupole, L=1., K1:=k51;
o52: Drift, L=0.2;
mf: marker;
nustormbl: line = (mi, o11, Q1, o12, ds1, o21, Q2, o22, ds2, o31, Q3, o32, ds3, o41, Q4, o42,
                   ds4, o43, ds5, o51, Q5, o52, mf);
```

Listing 5: Five quadrupoles and 6 bending dipoles transport line

Here we loosen the matching constraint on the optical functions at end of our lattice since the bending dipoles for generating x-axis dispersion are implemented along the transport line. For simplicity, we set the derivative of D_x and D_y to zero, and obtain the results shown in Fig. 12.

Of all 136 points of (α, β) as initial optical functions, only 14 points give results that satisfy the given constraints. We have lengthen the dipoles and see that to some extent, more black points show up in the plot, but the dispersion is getting larger. Adding a constraint on the maximum value of dispersion will reject these black points when the dipole length is getting larger, hence we choose to present the design in Listing. 5. With the allowed point corresponding to minimum lost particles, the optical functions are visualized in Fig. 13.

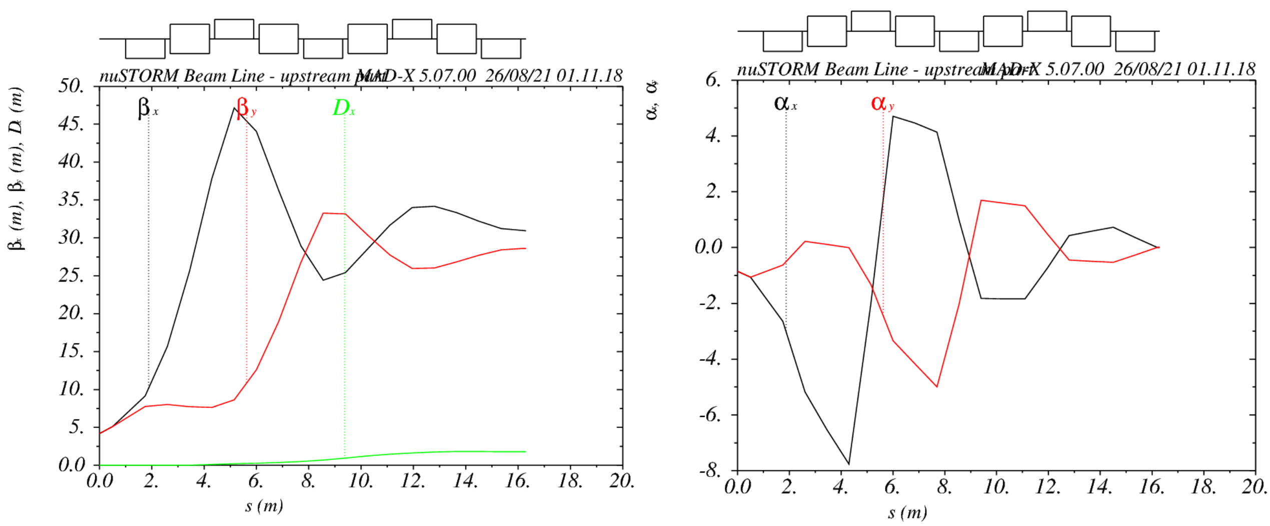

Figure 13: The optical functions for the design in Listing. 5.

The quadrupoles coefficients are calculated using the MATCH block reads as follows

$$
\begin{aligned}
k11 &= -0.165668, \\
k21 &= 0.337351, \\
k31 &= -0.342928, \\
k41 &= 0.222492, \\
k51 &= 0.559195,
\end{aligned}
$$

with the x-axis dispersion is $D_x = 1.798685039$ m, this lattice allow a seperation of the pion and kaon beams with the momentum specified above. However, one could point out that this design requires the quadrupoles to be operated with superconducting magnets, with the field at the poles as large as 3.5 T.

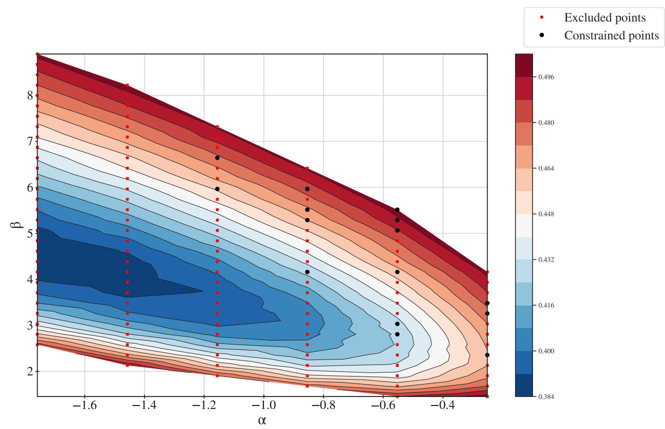

Figure 12: Constraints for initial optical functions of new design 5 to allow the separation between pion and kaon beam.

4 Conclusion

Given the input file of pion beam produced from the horn, we can select initial optical functions for designing the transport line lattice that optimize the pion beam. The initial optical functions above are used as input for designing the pion transport line with the constraints on maximum beta functions and matching at the end of the lattice. With the design shown in Listing. 5, the dispersion is large enough to efficiently seperate the ~ 5 GeV/c pions from the beam with reference momentum 6.25 GeV/c.

There are some points to proceed our next stage of studying: The Python codes for optimization the pion beam and scanning over initial conditions are still slow and require some optimization. Furthermore, we notice that the current design including 5 quadrupoles and 5 bending dipoles requires superconducting magnets to work properly. We believe that this design can be adjust to allow the lattice work with only normal magnets, and there are plenty of space for the optimization. We should also focus on the mechanism to seperate these two beams, and design two seperate lattices for transporting the appropriate particle beam into two facilities, namely nuSTORM and ENUBET.

APPENDICES

A Usage of MadX in designing a lattice

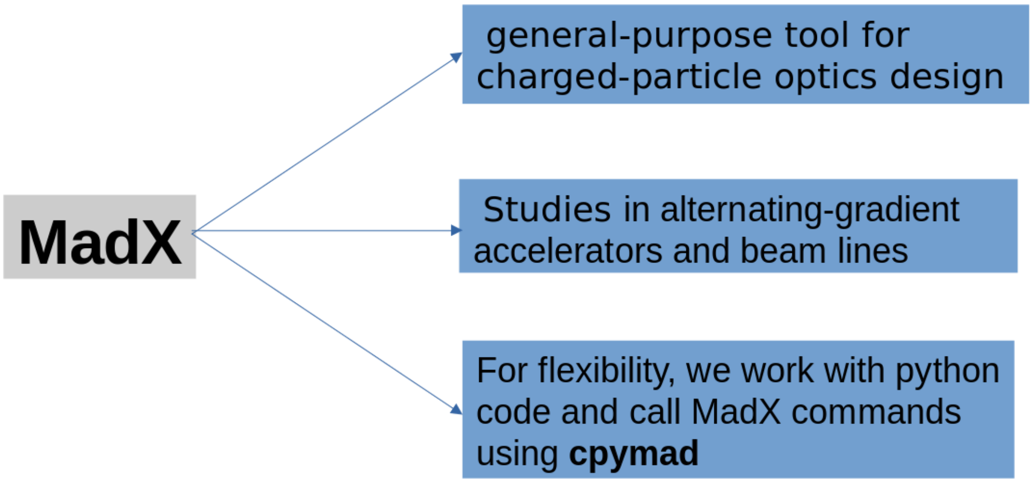

Figure 14: Basic usage of MAD-X. In our code, we call the MAD-X commands via wrapper functions that are imported from the package `cpymad`.

Below we summarize basic MAD-X commands and conventions that are used in our code. A more complete description can be found in e.g [6, 8].

A.1 Linear lattice functions and TWISS command

The list of linear lattice functions (a.k.a optical functions) used in MAD-X are given below, with the quantity inside each bracket indicates the measure of the corresponding function: [1]

- BETX Amplitude function β_x , [m]
- ALFX Correlation function $\alpha_x = -\frac{1}{2}\,(\partial\beta_x/\partial s)$, [1]
- MUX Phase function $\mu_x = \int \frac{ds}{\beta_x}$, [2π]
- R DX Dispersion of x: $D_x = \partial x/\partial\delta p$, [m]
- DPX Dispersion of p_x : $D_{px} = \partial^2 p_x/(\partial\delta p\partial s)$, [1]
- BETY Amplitude function β_y , [m]
- ALFY Correlation function $\alpha_y = -\frac{1}{2}\,(\partial\beta_y/\partial s)$, [1]
- MUY Phase function $\mu_y = \int \frac{ds}{\beta_y}$, [2π]
- R DY Dispersion of y: $D_y = \partial y/\partial\delta p$, [m]
- DPY Dispersion of py : $D_{py} = \partial^2 p_y/(\partial\delta p\partial s)/\partial s$, [1]
- R11, R12, R21, R22 : Coupling Matrix

Given a lattice line, once we know the beam dynamics and the initial conditions, we can solve the values of these optical functions at any point on the lattice. In MAD-X, the beam dynamics can be specified by the BEAM command, with the syntax as follows

[1] Here we define $\delta p \equiv \Delta p/p$.

```
BEAM, PARTICLE=string, MASS=real, CHARGE=real,
  ENERGY=real, PC=real, GAMMA=real, BETA=real, BRHO=real,
  EX=real, EXN=real, EY=real, EYN=real,
  ET=real, SIGT=real, SIGE=real,
  KBUNCH=integer, NPART=real, BCURRENT=real,
  BUNCHED=logical, RADIATE=logical,
  BV=integer, SEQUENCE=string;
```

Listing 6: BEAM command syntax

In current study, we need to specify the energy and energy dispersion of a pion beam, hence we have to specify the PARTICLE, ENERGY and SIGE attributes. The initial conditions are set inside the TWISS command, and we can then calculate the linear lattice functions (and possible the chromatic functions).

```
TWISS, SEQUENCE=seqname, LINE=linename, RANGE=range,
  DELTAP=real{,real}||initial:final:step,
  CHROM=logical,
  CENTRE=logical, TOLERANCE=real,
  FILE=filename,
  TABLE=tabname, NOTABLE=logical,
  RMATRIX=logical, SECTORMAP=logical,
  SECTORTABLE=tabname, SECTORFILE=filename,
  SECTORPURE=logical,
  EIGENVECTOR=tabname, EIGENFILE=filename,
  KEEPORBIT=name, USEORBIT=name,
  COUPLE=logical,
  RIPKEN=logical, TAPERING=logical ;
```

Listing 7: TWISS command syntax

A.2 MATCH block, VARY and CONSTRAINT

In order to deal with matching optical properties along the beamline, we use the MATCH block of MAD-X and basically go through the following procedure:

- Define the sequence(s) the matching module will work on
- Set initial conditions for transfer line matching
- Define constraints
- Define the parameters to be varied
- Match by different methods.

Take a part from our code for instance, we want to use the MATCH block to find the quadrupole coefficients of the four alternating quadrupole magnets-drift space transport line. The corresponding MAD-X code is

```
match, sequence=nustormbl, betx=7.5, alfx=0.79, bety=7.5, alfy=0.79, Dx=0.0, DPx=0.0;
  vary, name=k11, step=0.0001;
  vary, name=k21, step=0.0001;
  vary, name=k31, step=0.0001;
  vary, name=k41, step=0.0001;
  constraint, sequence=nustormbl, range=mf, betx=5.903272532, alfx=1.490004715,
 bety=8.378419334, alfy=-1.241069731;
  constraint, sequence=nustormbl, range=mi/mf, betx<45., bety<45.;
  simplex, calls=600000, tolerance=1.0e-16;
endmatch;
```

In this example, we only consider one sequence namely the transport line, with the initial values of optical functions fixed. Four quantities (which are the quadrupole coefficients) are varied with the Simplex Minimisation algorithm to match the optical functions at the end point, as well as the constraint on upper boundaries of beta functions along the lattice.

A.3 `cpymad`: a Python implementation of MadX

Python language is commonly used for scientific computing tasks due to its performance nature and simple syntax, especially a diversed utilities from a wide range of external libraries. In achieving the optimization of pion beam acceptance, we use Python to handle the input beam profile, proceed the data using `numpy` and finally exploit `matplotlib` for visualization purposes. Hence it would be suitable to call MAD-X commands directly from Python, and store the results (e.g the TWISS table or quadrupole coefficients) for further investigation. Thanks to the package **cpymad**, [1] we have a full control and access the state of MAD-X processes from Python via a Madx instance:

```
from cpymad.madx import Madx
madx = Madx()
```

There are a few functions in **cpymad** that give a flexibility in calling MAD-X commands, but the most straightforward method is to use input() method where we simply pass a string of MAD-X code into the madx object:

```
madx.input('CALL, FILE="fodo.madx";')
```

The collection of elements and their properties are easily accessed:

```
# list of element names:
print(list(madx.elements))

# check whether an element is defined:
print('qp1' in madx.elements)

# get element properties:
elem = madx.elements.qp1
print(elem.k1)
print(elem.l)
```

Finally, we can easily view and extract information from MAD-X tables as follows:

```
# list of existing table names
print(list(madx.table)):

# get table as dict-like object:
twiss = madx.table.twiss

# get columns as numpy arrays:
alfx = twiss.alfx
betx = twiss.alfy

# get all twiss variables for 10th element:
row = twiss[10]
```

With a little effort, one can analyze and visualize the results obtained from the twiss table with the help of **cpymad** package. Without manually writing a .madx file and run from the command lines, in Python code we can easily modify a madx object and looping over a parameter space, as we have done in 3.3.

[1] The source code is available for download at https://github.com/hibtc/cpymad, which is shipped with a short tutorial and some basic examples.

References

[1] D. Adey et al. "nuSTORM - Neutrinos from STORed Muons: Proposal to the Fermilab PAC". In: (July 2013). arXiv: 1308.6822 [physics.acc-ph] (cit. on p. 1).

[2] C. C. Ahdida et al. "nuSTORM at CERN: Feasibility Study". In: (Oct. 2020). DOI: 10.17181/CERN.FQTB.08QN (cit. on pp. 1, 2).

[3] F Acerbi et al. *The ENUBET project*. Tech. rep. Geneva: CERN, Oct. 2018. URL: https://cds.cern.ch/record/2645532 (cit. on p. 1).

[4] Ao Liu. "Design and simulation of the nuSTORM facility". PhD thesis. Indiana U., 2015. DOI: 10.2172/1209538 (cit. on p. 1).

[5] F. Acerbi et al. "Decay tunnel instrumentation for the ENUBET neutrino beam". In: *JINST* 15.05 (2020). Ed. by Marzio Nessi, p. C05059. DOI: 10.1088/1748-0221/15/05/C05059. arXiv: 2004.02532 [physics.ins-det] (cit. on p. 2).

[6] Hans Grote, Ghislain Roy, and Frank Schmidt. "The MAD-X Program". In: *European laboratory for particle physics* () (cit. on pp. 2, 3, 11).

[7] Helmut Wiedemann. *Particle Accelerator Physics*. Graduate Texts in Physics. Berlin, Germany: Springer, 2015. ISBN: 978-3-319-18316-9, 978-3-319-18317-6. DOI: 10.1007/978-3-319-18317-6 (cit. on p. 3).

[8] W. Herr and F. Schmidt. "A MAD-X Primer". In: (2004). DOI: 10.5170/CERN-2006-002.505 (cit. on p. 11).