

Bucknell University

Bucknell Digital Commons

Honors Theses

Student Theses

Spring 2023

Particle Swarm Optimization for High Rigidity Spectrometer

Yicheng Wang

Bucknell University, yw018@bucknell.edu

Follow this and additional works at: https://digitalcommons.bucknell.edu/honors_theses



Part of the [Nuclear Commons](#), and the [Optics Commons](#)

Recommended Citation

Wang, Yicheng, "Particle Swarm Optimization for High Rigidity Spectrometer" (2023). *Honors Theses*. 639.

https://digitalcommons.bucknell.edu/honors_theses/639

This Honors Thesis is brought to you for free and open access by the Student Theses at Bucknell Digital Commons. It has been accepted for inclusion in Honors Theses by an authorized administrator of Bucknell Digital Commons. For more information, please contact dcadmin@bucknell.edu.

**PARTICLE SWARM OPTIMIZATION FOR HIGH
RIGIDITY SPECTROMETER**

by

Yicheng Wang

A Thesis

Presented to the Faculty of
Bucknell University

in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science with Honors in Physics and Astronomy

April 28, 2023

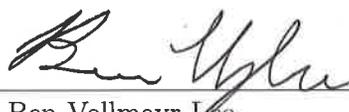
Approved:



Matt Amthor
Thesis Advisor



Sally Koutsoliotas
Second reader



Ben Vollmayr-Lee
Chair, Department of Physics and Astronomy

Acknowledgments

Thanks to Prof. Matt Amthor for detailed instructions on the project. Thanks to Prof. Sally Koutsoliotas for thorough discussion on thesis writing. Thanks to the collaborators from Facility for Rare Isotope beams at Michigan State University for providing computer simulation codes. Special thanks to all the faculties and students I have worked with at Bucknell University.

Contents

Abstract	vi
1 Nuclear physics	1
2 Charged particles in magnetic fields	4
2.1 Particle beams and rare-isotopes	4
2.2 Magnetic field and Lorentz force	7
2.3 Bending magnets	8
2.4 Focusing magnets	10
2.5 Ion optical system	12
2.5.1 Definition of ion optical system	12
2.5.2 Magnetic rigidity	13
2.5.3 Higher-order aberration	14
3 High Rigidity Spectrometer	18
3.1 Facility for Rare Isotope Beams	18

<i>CONTENTS</i>	iv
3.2 High Rigidity Spectrometer	19
3.2.1 The Spectrometer Section	19
3.2.2 High-Transmission Beamline (HTBL)	21
4 Monte Carlo simulation	26
4.1 Monte Carlo simulation	26
4.2 Input particle distribution	27
4.3 COSY Infinity: transfer matrices and particle tracking	28
4.4 Optimization objectives	30
5 Particle Swarm Optimization (PSO)	34
5.1 Basic function of PSO	34
5.2 Internal parameters: inertia and acceleration	39
5.3 Defining objective functions	42
6 Data acquisition and analysis	45
6.1 Method of group simulation and analysis	45
6.2 Results	47
6.2.1 Optimization for the Spectrometer Section	47
6.2.2 Optimization for HTBL	50
7 Conclusion and future	62

A	COSY Infinity code for spectrometer in section 2.5	64
B	ROOT code for HTBL optimization	66

Abstract

The goal of this project is to find reliable parameter settings for a multi-dimensional global optimizer to optimize the performance of a large acceptance ion optical system for the requirements of nuclear physics experiments. We develop and test the Particle Swarm Optimization (PSO), a global optimization algorithm designed for continuous multi-dimensional problems, on a large acceptance particle beam separator, the High Rigidity Spectrometer (HRS) at the Facility for Rare Isotope Beams (FRIB), which is a laboratory specializing in the production and experimental study of short-lived nuclear matter. We split the HRS into two sections, the High-Transmission Beamline (HTBL) and the Spectrometer Section. The objective of the PSO is to improve the transmission rate for the Spectrometer Section and the beam spot size as well as dispersive foci for the HTBL, in both cases by adjusting the settings of the higher-order magnetic field elements—sextupoles and octupoles. We successfully improved the transmission rate of fission fragments from 57.5% to 63% in the Spectrometer Section and shrank the beam spot size in horizontal direction of ^{40}Mg from 1.8 mm to 1.0 mm in the HTBL. Finally, we compare the performance of PSO with multiple, different internal parameter settings (*inertia*, *acceleration*). We find a significant difference in

performance across the PSO parameter settings considered, with the best internal parameter setting for the PSO being (0.9, 0.45). Future work will explore the PSO parameter space for additional input particle distributions (e.g. fission fragments in the HTBL) and differently constructed objective functions to better understand the operation of the PSO in these ion optical systems.

Chapter 1

Nuclear physics

Nuclear physics is the subfield of physics that studies nuclear structures and reactions between nuclei, which are at the core of atoms. Back in the 5th century BCE, ancient Greek philosopher Leucippus of Miletus originated the idea of atomic philosophy, in which matter is considered to consist of an infinite number of indivisible building blocks “atomos” and the different properties of matter are determined by the types of “atomos” [4]. In the early 19th century, John Dalton converted the atomic philosophy into scientific theories through physical observations and experiments [17]. In 1897, William Thomson developed the first atomic structure including the discovery of electrons by J.J. Thomson. Thomson’s atomic model, or the “plum-pudding” model, describes the atom as a uniform sphere of positively charged matter with embedded electrons (negative charges) like raisins in a plum pudding [15]. A few years later, Ernest Rutherford conducted the famous gold foil alpha scattering experiment, in which he demonstrated that most of the volume occupied by atoms was actually

empty space with only a tiny, solid part inside, called the nucleus. The discovery of the nucleus led scientists to the study of nuclear structures, which became the sub-field of nuclear physics. Although the concept of nuclear physics already exists for decades, the study of nuclear properties and dynamics are largely inaccessible experimentally except for limited cases of naturally radioactive isotopes, since the energy scales involved are much higher than those of chemical reactions [10].

In nuclear physics, scientists not only explore nuclear theories to describe microscopic particles—such as protons, neutrons, and the nuclei constructed from them that comprise the majority of the mass of physical matter of our everyday experience—but also conduct experiments to verify and expand their theoretical understanding. Those experiments often require large accelerators to collide particles at speeds of a significant fraction of the speed of light, to enable them to probe inside the nucleus in the context of the extremely strong Coulomb repulsion between nuclei. By studying nuclear physics, we are also able to better understand some of the largest scale physical processes in the universe: stellar burning, evolution, and explosions (novae, supernovae, and x-ray bursts), as well as the concurrent nucleosynthesis processes that generate much of the nuclear matter in the universe [10].

Modern nuclear physics uses beams of high-speed particles for the experiments, to examine the properties of specific nuclear structures in a systematic way. This systematic study requires the ability to produce, separate, and identify the particles within the beams. Scientists build particle accelerators to generate high-speed beams and spectrometers to separate and identify the particles.

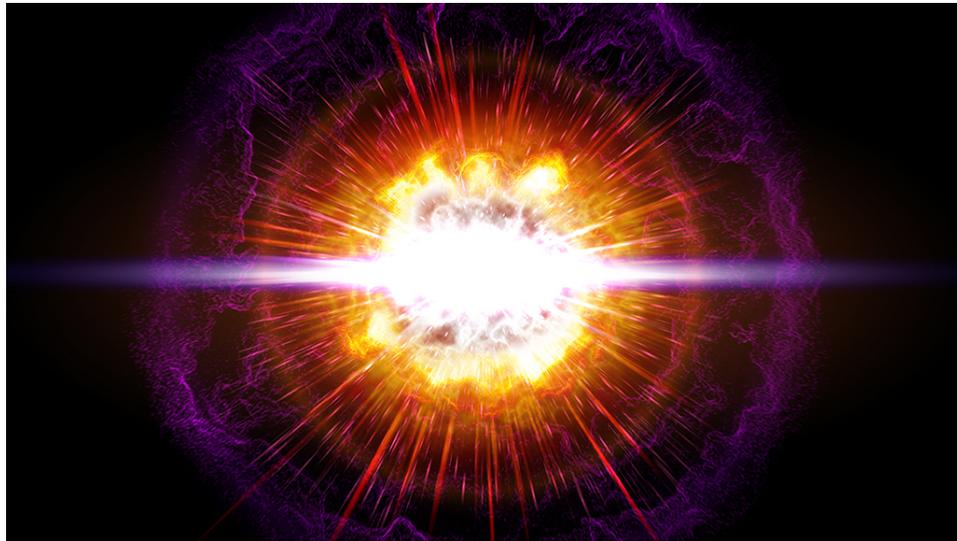


Figure 1.1: An artist's interpretation of supernova, a stellar explosion, spotted in a galaxy 4.6 billion light-years away in 2016 [20].

Chapter 2

Charged particles in magnetic fields

2.1 Particle beams and rare-isotopes

Most of the particles in daily life are neutral particles, but in nuclear physics experiments we ionize them to become charged particles (particles carrying net charges). Before discussing the motion of charged particles in magnetic fields, it is beneficial first to explain the definition of a charged particle beam. One may think of a particle beam by its relation to a laser beam or light beam, which is a thin straight stream of photons travelling at the speed of light. A charged particle beam, by its name, is a stream of charged particles like electrons or ions—atoms with more or fewer electrons than protons in their nucleus. Unlike a typical laser beam, particle beams are not

always thin and straight. For example, a beam consisting of the products of a nuclear fission reaction originates as a cluster of particles at the reaction target (where the target-beam interaction induced the fission) but then spreads out isotropically due to the random orientation of the fission axis in the center of mass frame. Since the particles are moving forward at a significant fraction of speed of light, however, they end up continuing along their initial direction of motion with only a slight angular deviation (Figure 2.1).

A typical opening angle for fission fragments is around five degrees. In beam physics, we typically use milli-radians for the spreading angle, and five degrees is equal to 100 mrad. Such a small angle will still result in the ultimate divergence of the beam particles if they are not captured and redirected properly, in our case with magnetic fields. It is also worth noting that a particle beam does not necessarily contain only a single type of particle. Nuclear fission will produce a mix of different daughter isotopes, even from a single parent isotope, as is the case in well-known Uranium-235 fission (e.g. producing fragments like cesium, xenon, and strontium, among many others) [3].



Figure 2.1: Illustration of a travelling particle beam. The beam spot spreads out as it travels due to the angular divergence of the trajectories. The red lines show the trajectories of several example particles.

In this project, we are focusing on a specific category of particles, called rare-

isotopes. Isotopes are elements with the same number of protons and different number of neutrons in the nucleus [13]. For instance, the element carbon has six protons in the nucleus, but the number of neutrons varies resulting in different isotopes, like ^{12}C and ^{14}C , which have six and eight neutrons in the nucleus, respectively. The superscript gives the total number of nucleons (protons and neutrons) in the nucleus. Rare-isotopes are radioactive isotopes that will ultimately decay, some in an extremely short period of time (e.g. less than a millisecond) [16]. Some examples of rare-isotopes are Sodium-35 (^{35}Na) and Magnesium-38 (^{38}Mg), both with half-lives (time taken for half of the particles to decay) to be approximately 10 ms [5]. The rare-isotopes exist in nature where nuclear reactions are taking place, like core of stars, supernovae, and neutron stars. Experimental study of rare-isotopes thus require that they be produced through nuclear reactions like projectile fission—where a single, large parent nucleus is separated into two, smaller daughter nuclei—or projectile fragmentation where some number of neutrons and protons are removed by abrasion with the reaction target nuclei. Nuclear physicists are interested in measuring the properties of rare-isotopes—such as mass, momentum, charge, and excitation structure—and in conducting experiments involving further nuclear reactions induced by the rare-isotopes generated. Details of the rare-isotopes we use in our work will be discussed in section 4.2.

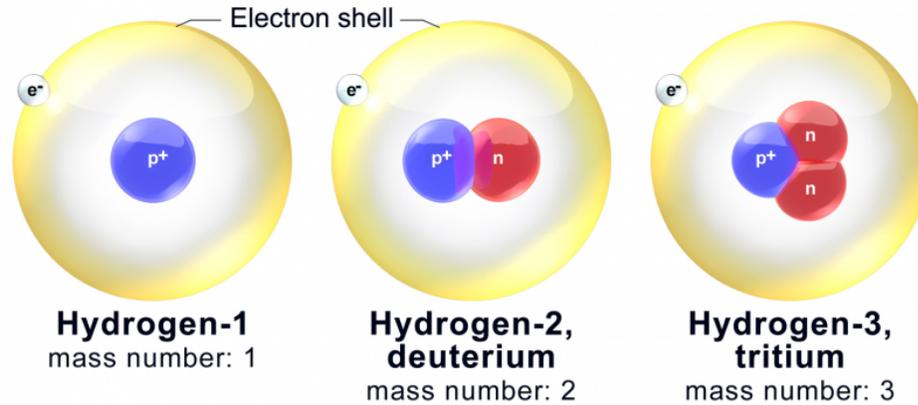


Figure 2.2: Isotopes of the element Hydrogen, ${}^1\text{H}$, ${}^2\text{H}$, and ${}^3\text{H}$, all have one proton in their nucleus but have 0, 1, and 2 neutrons, respectively. The size of the nucleus relative to the electron cloud is not scaled. The actual size of nucleus is on the order of 10^5 times less than the electron cloud [13].

2.2 Magnetic field and Lorentz force

A common bar magnet consists of a north pole (N) and a south pole (S). The magnetic field is a vector field defined as pointing out of the north pole and pointing into the south pole. By connecting those vectors together we can obtain the magnetic field lines, which always form a closed loop going out from N and into S outside the magnet, and continuing to close the loop by going from S to N inside the magnet. Note that the magnetic field lines never cross each other, since at any given point, the magnetic field has a unique magnitude and direction.

Magnetic fields influence the motion of moving charged particles including electrons and ions, due to the fact that the magnetic field exerts a force, called the Lorentz

force, on any moving charged particles.

$$\vec{F}_{\text{mag}} = Q (\vec{v} \times \vec{B}) \quad (2.1)$$

Here Q is the charge of the particle, \vec{v} is its velocity vector, and \vec{B} is the magnetic field. Due to the nature of the vector cross product, the direction of the Lorentz force is always orthogonal to both the direction of the particle's velocity and the direction of the magnetic field. This results in a change in the direction of motion of the charged particle, while leaving the magnitude of the particle's velocity unchanged [6].

2.3 Bending magnets

The simplest magnetic field one can imagine is a constant, uniform field, which has both a constant direction and strength. Such a field may be approximated by positioning the North pole directly opposite the South pole, as shown in Figure 2.3. In the ideal case, the magnetic field lines inside the region between the poles are all pointing straight downward. A charged particle beam travelling through the dipole magnetic field will be bent in the direction orthogonal to both the direction of the B-field and direction of motion of the particle with the curvature of bending linearly proportional to the dipole field strength, according to the Lorentz force (Eq. 2.1). If we have a large enough dipole field, and the particle's motion is perpendicular to the field, the particle will move in a circle—the Lorentz force and the corresponding acceleration being always perpendicular to the direction of motion. This phenomenon

is called the cyclotron motion, which can be used to experimentally determine the particle's charge-to-mass ratio. In our ion optical systems dipoles are used to bend the particle beam either left or right, consequently producing an angular and spatial divergence between particles of different momenta [23].

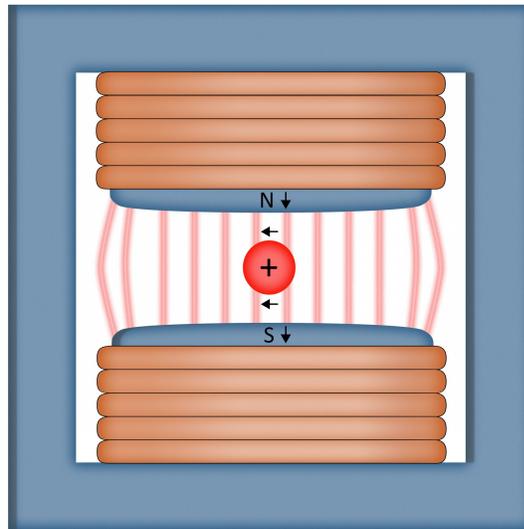


Figure 2.3: Schematic diagram of a magnetic dipole. It includes an iron yoke (dark blue), including two pole tips, and electric wires (orange) surrounding the pole tips. In this case, the upper pole is N and the lower is S. The black arrows besides “N” and “S” pointing downward represent the direction of the magnetic field vectors. Light orange lines between the pole tips represents the magnetic field lines. The red sphere at the center represents a positively-charged particle travelling into the page, with the force exerted by the field therefore pointing to left [14].

Modern dipoles for ion optical systems in nuclear physics use current carrying coils to induce the magnetic fields, as shown in Figure 2.3 by the orange wires wrapping the blue poles. The magnetic field strength—and thus the degree of particle bending—can then be controlled by changing the current flowing in the coils.

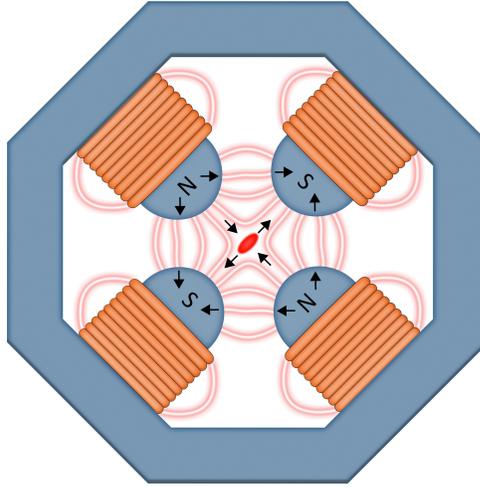


Figure 2.4: Schematic diagram of a magnetic quadrupole consisting of an iron yoke (dark blue), four shaped pole tips labeled in “N” and “S”, and current-carrying coils (orange) wrapping the poles. The black arrows beside “N” and “S” represent the direction of the magnetic field. Light orange lines between the pole tips represent the magnetic field lines, and the red ellipse at the center represents a positively-charged particle [14].

2.4 Focusing magnets

Just as a magnetic dipole has two poles (N and S), a magnetic quadrupole has four poles (Figure 2.4), with N and S arranged alternatively around the central, symmetry axis. The magnetic quadrupole will focus the moving charged particles in one plane (horizontal or vertical) and defocus them in the other plane (vertical or horizontal). As before, the beam deflection must be perpendicular to both the direction of the particle motion and the direction of the magnetic field, according to the Lorentz force law. Conceptually one may think of the function of a magnetic quadrupole as being similar to a thin lens that is convex in one plane and is concave in the perpendicular plane [23].

Consider sending a positively-charged particle beam into the page through the quadrupole in Figure 2.4. In the x -direction (horizontal plane), particles to the left will be influenced mostly by the left N-S pair of poles and will be deflected to the left (outward). Similarly, particles directly right of the central axis, between the right N-S pair will be deflected to the right (outward again). Thus in the x -direction, positively-charged particles will be deflected outward from the center—defocused—by this quadrupole. The opposite situation happens in the y -direction, where positively charged particles will get deflected inward—focused—by this quadrupole. Like a spherical lens, the bending strength of the quadrupole (both in the horizontal and vertical directions) increases linearly with the distance from the center. Modern quadrupoles also use electromagnets to allow for variable focusing and defocusing strength [23].

Similar to the design of the quadrupole, the magnetic sextupole has six magnetic poles and the magnetic octupole has eight magnetic poles. In both cases, N and S poles are arranged alternately. These higher-order focusing elements act in some ways similar to the quadrupole, with deflection strengths depending on the distance from the central axis, but instead of focusing and defocusing along the x - y directions, they do this along the multiple directions of their own planes of symmetry. Also, the deflection strength of the sextupole and octupole increases with the square and the cube of the distance from the center, respectively, rather than increasing linearly as is the case for the quadrupole [23].

In this project, instead of using only quadrupoles, we combine the quadrupole together with superimposed sextupole and octupole fields as a multipole. The purpose

of the sextupole and octupole elements is the elimination of higher-order optical aberrations, as will be discussed in section 2.5.3. The sextupoles and octupoles thus serve as adjustable higher-order correction elements inside the multipoles.

2.5 Ion optical system

2.5.1 Definition of ion optical system

Telescopes and microscopes are two examples of typical light optical systems, consisting of a combination of lenses, mirrors, films, screens, and detectors to expand our vision beyond our purely biological capability. Similar to light optical systems, an ion optical system consists of a combination of different electric or magnetic fields that can bend (magnetic dipole), focus (magnetic quadrupole), or otherwise modify the trajectories of the moving ions (magnetic sextupole and octupole) as well as detectors that can measure various properties of ions like mass, momentum, kinetic energy, and charge [10].

For secondary beam experiments involving rare-isotopes produced by nuclear reactions induced by primary beam particles impacting a production target, the beam does not consist of purely one type of particle. Rather the beam is a mixture of various isotopes. To make measurements involving only one specific type of particle, we have to first separate the beam constituents according to different criteria like momentum and charge. An ion optical system that is able to separate particles

according to momentum and charge is called a spectrometer. Figure 2.5 shows a ray plot (diagram including magnetic elements and selected particle trajectories) of a simple spectrometer.

The example spectrometer consists of a triplet of magnetic multipoles and a magnetic dipole. The purpose of the multipole triplet is to refocus the divergent ion beam, whereas the dipole is used to separate the ions by momentum. The resolving power (ability to distinguish or separate particles) of the spectrometer depends on the dispersion of the beam produced by the dipoles and depends inversely on the final beam spot width. With the separated beams at the end of the spectrometer, we are able to measure and identify each type of particle.

2.5.2 Magnetic rigidity

The magnetic rigidity $B\rho$ refers to the resistance of ions to the bending when travelling through the magnetic dipole. It is the product of the magnetic field strength B and the bending radius ρ . This magnetic rigidity is equal to the ratio of momentum to charge of the ions, and has the unit of Tesla · meter (Tm).

$$B\rho = \frac{p}{q} \tag{2.2}$$

Here p is the momentum of the ion, and q is the charge carried. While the magnetic rigidity is a property of a given ion, the beam magnetic rigidity is also used to determine the field settings for the spectrometer. This is because the curvature of the

beam trajectory through the dipoles is fixed by the design of the spectrometer, due to the fixed location of the upstream and downstream beamlines. Thus the dipole fields must always be tuned to bend the ions to be transmitted with the predefined bend radius. It is worth noting that the product of the maximum design dipole magnetic field, B_{\max} , and the bend radius defined by the beamlines, ρ , therefore defines the maximum magnetic rigidity beams transmissible by a given spectrometer [22].

2.5.3 Higher-order aberration

There are two main reasons why modern spectrometers and separators have grown in size and complexity relative to the simple spectrometer shown above. The first one is that we want to make measurements like momentum of the particles at different positions inside the spectrometer to obtain more information about the particles' properties and how their properties may change through the system, since properties like angle, momentum, and charge may change after the nuclear reaction.

The second reason is related to the optical aberrations. Optical aberrations are produced by deviations of the beam trajectories relative to the trajectories predicted by the thin lens approximation (assuming all bending occur at same place) and the paraxial approximation (assuming particles' angles are close to zero, where $\sin(\theta)$ is approximately equal to θ). In our systems, the magnetic focusing elements have significant lengths, which do not follow the thin lens approximation. Thus large acceptance systems inherently have significant optical aberrations. Also, defects of magnetic fields far from the central axis of the system (for these large acceptance

systems) can produce additional optical aberrations. For a light optical example, consider a pair of glasses. When looking through the edge of the glasses, especially for a strong prescription, the image will be blurred compared to looking through the center of the glass lenses. The same defects occur in ion optical systems, where trajectories far from the center of the magnetic lenses will deviate from the trajectories predicted by ideal approximations, resulting in a blurred focus of the particle beams. As a first approximation, we simulate an ion optical system (i.e. the spectrometer) in first-order, which is the ideal case with no aberrations. To better predict the real situation, however, we must simulate the system in higher-order (fifth-order in this project) which will include the dominant aberrations. Figure 2.6 shows a fifth-order spectrometer ray plot.

To eliminate the higher-order aberrations, we engage the higher-order correcting elements, namely the sextupoles and octupoles in the multipoles, as well as adding extra magnetic elements to the spectrometer. The ion optical system will have more degrees of freedom of tunable magnetic elements which makes the tuning process a multi-dimensional problem.

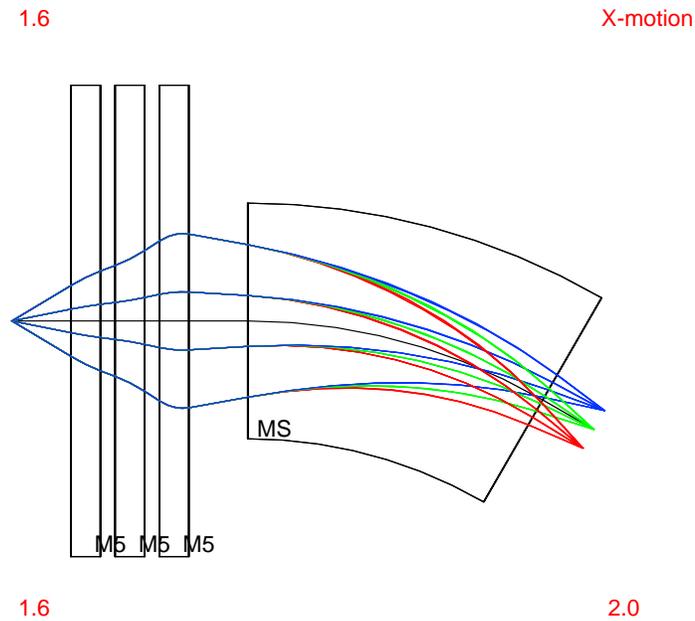


Figure 2.5: Ray plot of a simple spectrometer with three focusing elements and one bending element. “M5” represents one magnetic multipole focusing element, and “MS” represents a magnetic dipole. The input particle beam that enters the system with diverging rays from the left is separated according to different momentum-to-charge ratios (later defined as the magnetic rigidity in section 2.5.2), illustrated in blue, green, and red color. Ions (blue) with higher momentum-to-charge ratio will be bent less, thus appearing at the top of the final focal plane. The red numbers on the top and bottom left indicate the diameter of multipoles in meters, whereas the red number on the bottom right indicates the total length of the system in meters. These system parameters also appear on all the following ray plots.

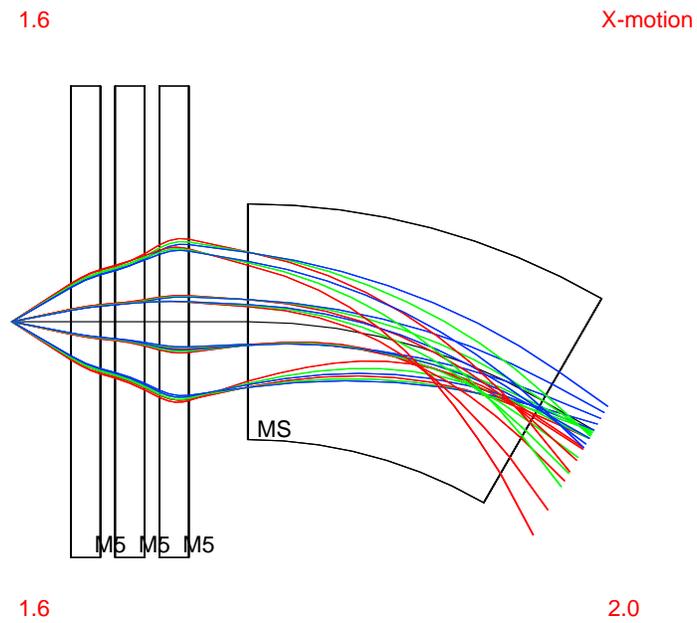


Figure 2.6: Fifth-order ray plot of the spectrometer. The major change from first-order to fifth-order spectrometer is the final focus points for each representative particle. The low-momentum particles (red) focus much earlier than the high-momentum particles (blue), which have not yet been focused to one point in the ray plot. This is caused by the higher-order aberrations.

Chapter 3

High Rigidity Spectrometer

3.1 Facility for Rare Isotope Beams

The Facility for Rare Isotope Beams (FRIB) is a user facility for nuclear physics experiments operated by Michigan State University and funded by the U.S. Department of Energy Office of Science. FRIB is designed to produce a variety of rare-isotopes predicted to exist and to conduct experiments with these isotopes. There are three main sections in FRIB: a $200\text{MeV}/u$ superconducting heavy-ion driver linear accelerator used to accelerate stable isotopes and produce rare-isotopes through nuclear reactions, the Advanced Rare-Isotope Separator (ARIS) [8] which is capable of collecting and purifying the rare isotope beams from the initial reaction at maximum magnetic rigidity of 8 Tm , and the High Rigidity Spectrometer (HRS) which is in the design phase [12].

3.2 High Rigidity Spectrometer

The HRS will consist of two sections: an analysis beamline called the High-Transmission Beamline (simplified as HTBL) and the Spectrometer Section. For the rare-isotope experiments, the high magnetic rigidity allows us to study all the isotopes of interest at the energies at which they are most efficiently produced. The purpose of HTBL is to allow the large beam envelope and rigidity from ARIS to reach the Spectrometer Section and make in-flight measurements of momentum of the particles before going to the Spectrometer Section. The purpose of the Spectrometer Section is to separate, identify, and refocus the products of nuclear reactions at the target (labelled “FS0” in Figure 3.1) involving the incoming rare isotope particles of interest. Details follow in the remainder of this section [12].

3.2.1 The Spectrometer Section

The Spectrometer Section of HRS starts from focal plane FS0 (end of HTBL) to FS2 (end of the whole HRS) in Figure 3.1, which includes six multipoles listed as QS1 through QS6 (the final letters “A” and “B” in the names refer to the specific multipole designs), and two dipoles DS1 and DS2. Figure 3.2 and 3.3 show the first-order and fifth-order ray plots for the Spectrometer Section in both horizontal (x) and vertical (y) directions. The Spectrometer Section is designed to separate and identify the products of nuclear reactions between incoming particles from the upstream HTBL and the reaction target at FS0. The particle distribution starts as a cluster at FS0, and particles with different momenta focus separately at FS2. Thus we call FS2 a

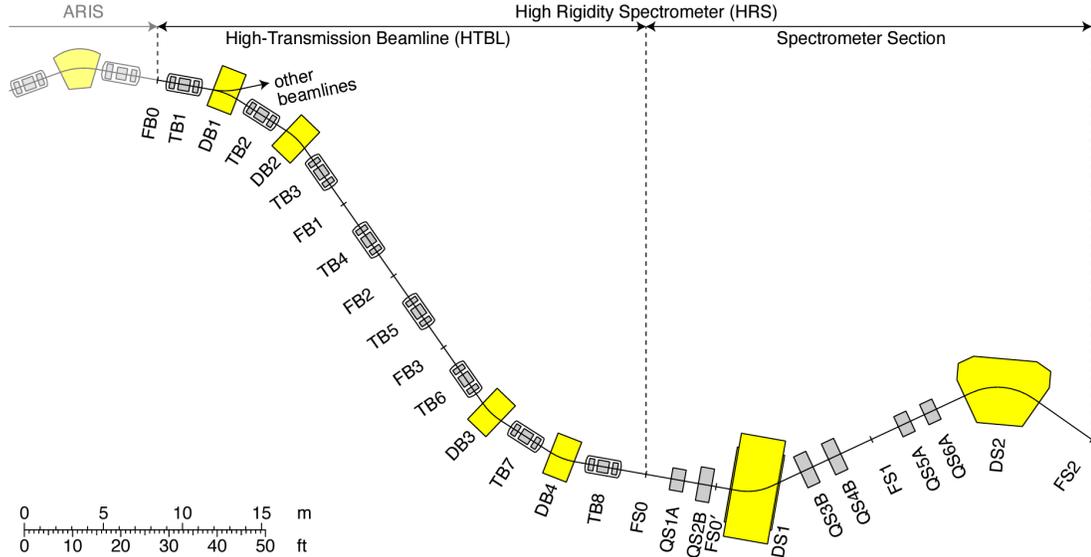


Figure 3.1: Layout of HRS consisting of HTBL and the Spectrometer Section. The gray and yellow boxes represent the quadrupole and dipole magnets respectively. The first letters of the magnets’ names denote the element types: “F” for a focal plane, “D” for a dipole magnet, “Q” for a multipole magnet, and “T” for a multipole triplet (Used with permission from [12]). FS0 is the location of the reaction target.

dispersive focal plane.

The magnetic field strengths of the quadrupoles are determined by the basic properties desired of the optical tune, to give a clear dispersive focal plane at FS2 while containing the ray envelope within the system apertures (see the first-order ray plot in Figure 3.2 top) [12]. But we have to correct for the effect of the higher-order aberrations by tuning the higher-order correcting elements, namely the sextupoles and octupoles (including in total 12 tunable elements: 6 multipoles times 2 per multipole). Because the products of rare-isotope reaction are measured at FS2, we need to minimize beam losses in the system before FS2. Therefore the goal of higher-order tuning for the Spectrometer Section is to maximize the transmission rate. The uncor-

rected Spectrometer Section has the transmission rate of 57.5% for fission fragments that are challenging to be transmitted.

3.2.2 High-Transmission Beamline (HTBL)

The High-Transmission Beamline (HTBL) starts from FB0, the end of ARIS, and ends at FS0, the reaction target of the Spectrometer Section. It consists of eight multipole triplets (a series of three multipoles) listed as TB1 through TB8, and four dipoles DB1 to DB4. Figure 3.4 and 3.5 show the first-order and fifth-order ray plots for HTBL in both x and y -directions. HTBL is designed to transport rare-isotopes from ARIS to the Spectrometer Section and measure particles' momenta at focal planes FB1 and FB3. Thus the particles enter HTBL at a single achromatic beam spot (all momenta focused at the same location), with dispersive foci at FB1 and FB3 (horizontal position depends linearly on momentum), and eventually is refocused again to an achromatic image (all particle beams back to the same position) at FS0.

Similar to the tuning of the Spectrometer Section, we need to reduce the higher-order aberrations in HTBL by tuning the sextupoles and octupoles inside each multipole (totally 48 tunable elements: 8 triplets times 3 multipoles per triplet times 2 per multipole). The goal of higher-order tuning for HTBL is much more complicated than that for the Spectrometer Section, because it needs to provide a high transmission rate, a small beam spot size at FS0 (which determines the resolving power of the Spectrometer Section), and clear dispersive foci at FB1 and FB3. Details of how to achieve this goal will be discussed in section 4.4.

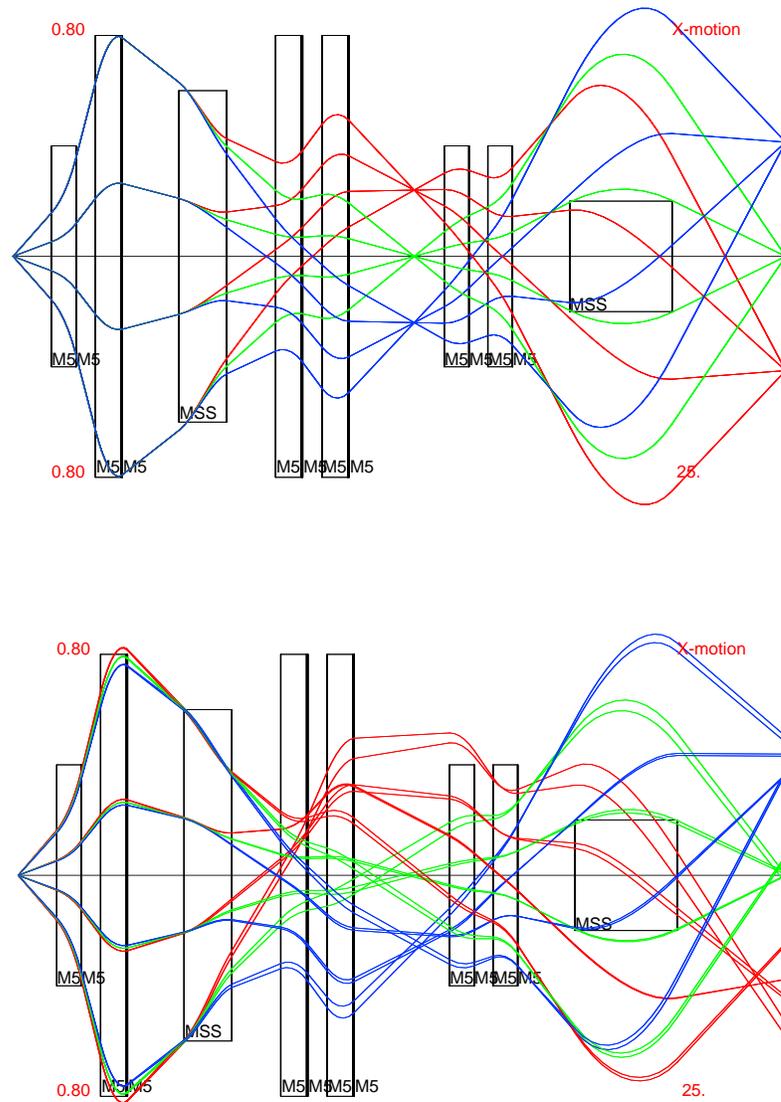


Figure 3.2: Top: first-order ray plot of the Spectrometer Section in horizontal (x) direction with clear dispersive focus at FS2 (end of the system), and all the particles trajectories contained inside the system apertures. Bottom: fifth-order ray plot. Blurred dispersive focus at FS2 with some particles in the distribution outside the acceptance of the system (top red lines at 5th “M5” multipole). Note that unlike the simple spectrometer in section 2.5, the central axis of the ray plot here has been straightened for clarity of the plotting. This technique has been used for all the rest of the ray plots.

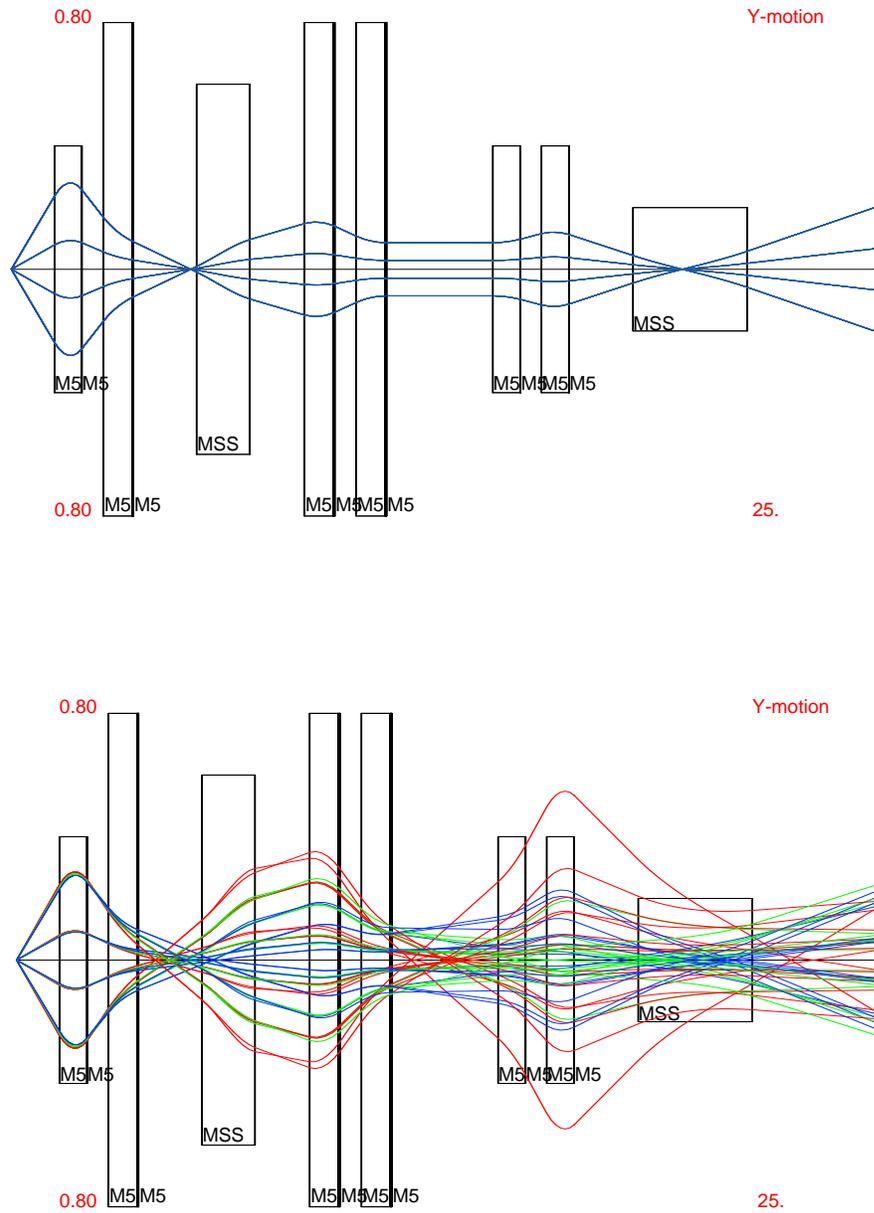


Figure 3.3: Top: first-order ray plot of the Spectrometer Section in vertical (y) direction. Bottom: fifth-order ray plot in y -direction. Since the Spectrometer Section separates particles by momentum only in x -direction, ray plots in y -direction do not carry much important information.

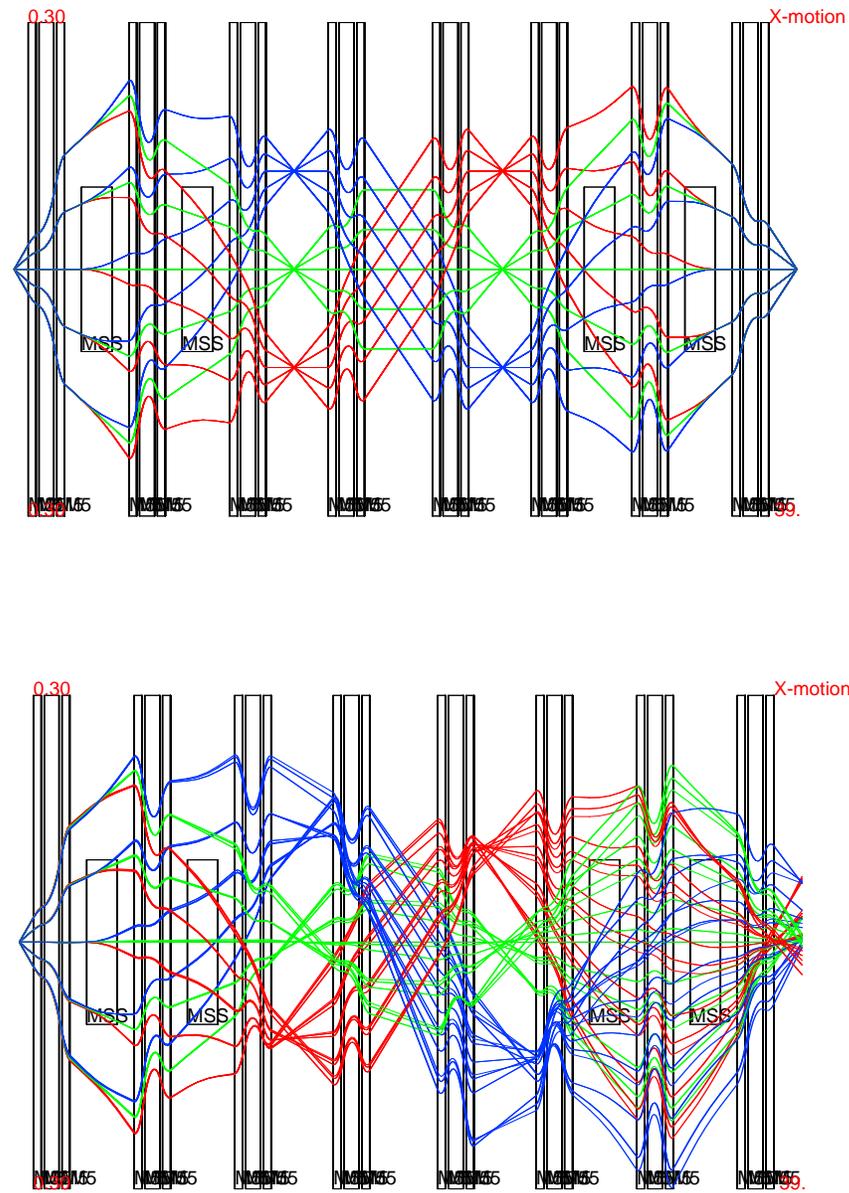


Figure 3.4: Top: first-order ray plot of HTBL in x -direction. Clear dispersive foci at FB1 (between 3rd and 4th triplet from left side) and FB3 (between 5th and 6th triplet), and perfect achromatic focus at FS0 (end of HTBL). Bottom: fifth-order ray plot of HTBL in x -direction. Blurred dispersive foci at FB1 and FB3, and a large beam spot (unfocused) at FS0.

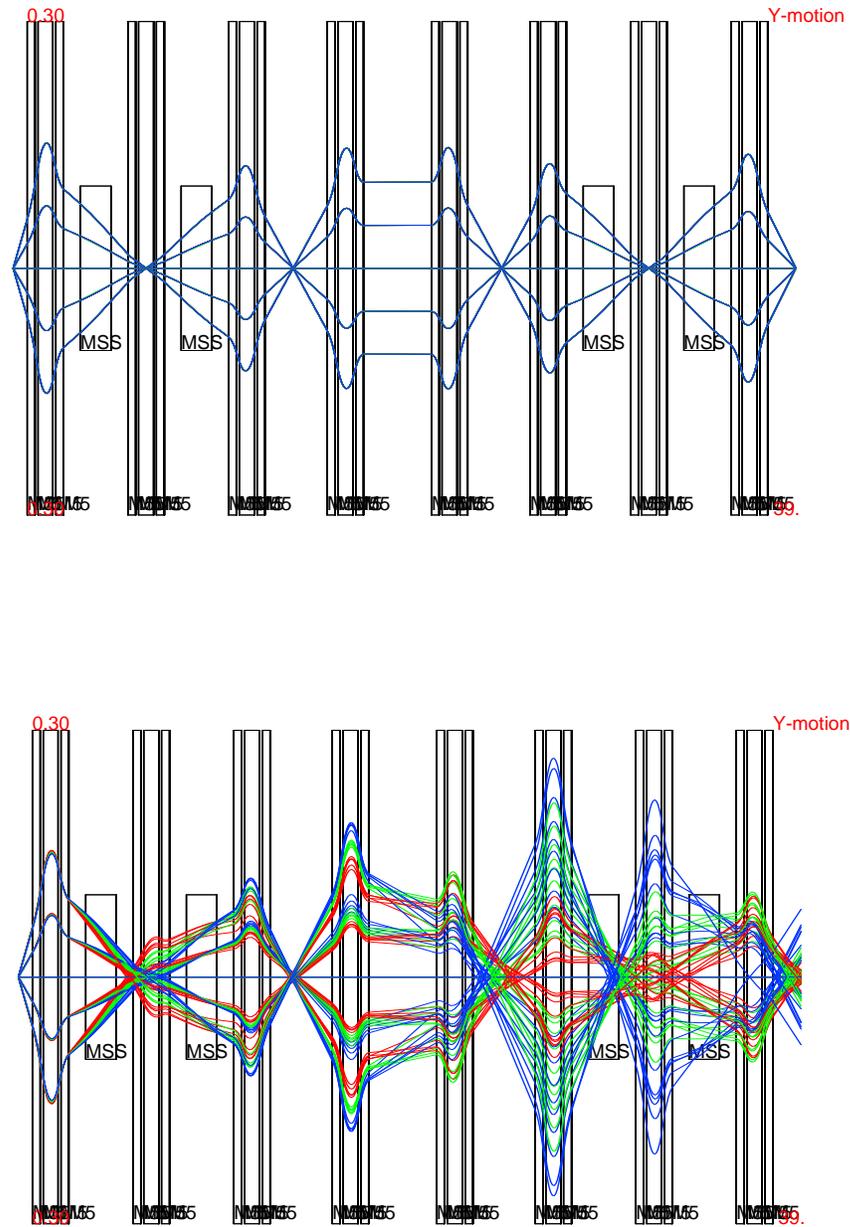


Figure 3.5: Top: first-order ray plot of HTBL in y -direction. Clear achromatic foci at FB1, FB3, and FS0. Bottom: fifth-order ray plot of HTBL in y -direction. Blurred achromatic foci at FB3 and FS0.

Chapter 4

Monte Carlo simulation

4.1 Monte Carlo simulation

Since HRS has not yet been built, we use computer programs to simulate the trajectories of rare isotope beams travelling through the system. Monte Carlo simulation is a method of random sampling and statistical modeling to estimate mathematical functions and mimic the operations of complex systems. A well-known example of using Monte Carlo simulation is the “Buffon’s needle”, which is a method designed by Georges-Louis LeClerc, Comte de Buffon in the 18th century to estimate the value of π using random tosses of needles onto a equally-spaced lined sheet [7]. In this project, we use Monte Carlo simulation to mimic the operation of particle generation from the upstream of HRS to calculate the transmission rate, beam spot size, and clarity of dispersion at dispersive foci.

4.2 Input particle distribution

Since the Spectrometer Section and HTBL have different operation goals, we use separate input particle distributions for the simulation. For the Spectrometer Section, we use data of real fragments from nuclear fission reactions from the Los Alamos Laboratory [24]. The reason of using this specific particle distribution for the transmission rate optimization is that the fission fragments are extremely challenging to be transmitted with a huge divergent distribution. The particle distribution has five main coordinates: horizontal position x and horizontal angle a , vertical position y and vertical angle b , and kinetic energy deviation dk . Figure 4.1 shows the shape of particle distribution being used.

We use Monte Carlo simulation to generate separate input particle distributions for HTBL, which in this project is Magnesium-40 (a rare-isotope near the very edge of nuclear stability) whose observation was first reported in 2007 [18]. We choose this particle distribution because we want to tune the HTBL system so that it can both transmit rare-isotopes and make in-flight measurements of their momentum and the broad distribution of ^{40}Mg —due to the energy straggling during its production in a very thick reaction target—makes it a challenging case. The particle distribution of ^{40}Mg is generated with the distribution properties provided by LISE⁺⁺, a program designed to simulate the operation of radioactive beam production [19]. We use Gaussian distributions to approximate the real distribution of particles with respect to each of five coordinates, which approximates the production distribution of fragmentation products well.

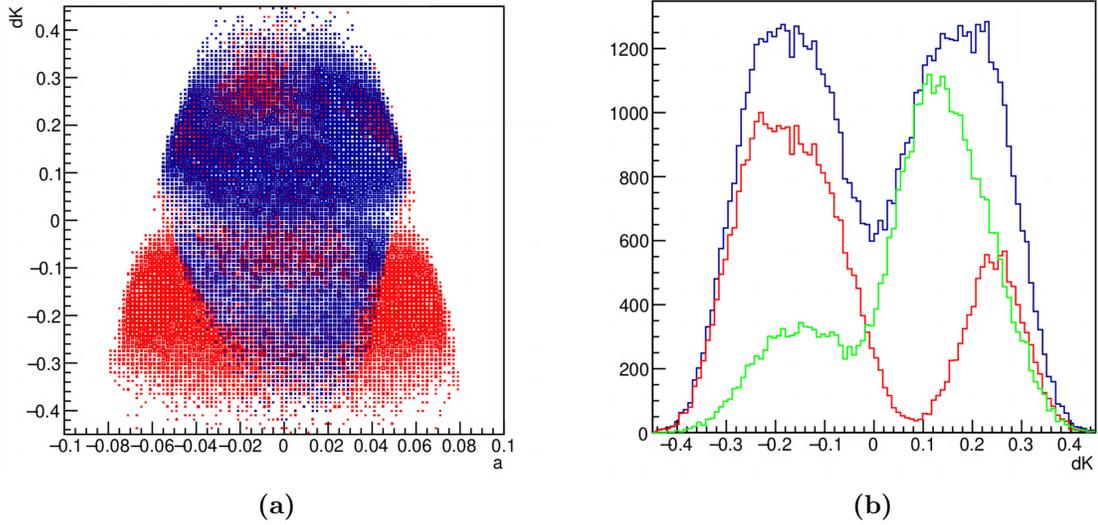


Figure 4.1: The input particle distribution of fission fragments generated through Monte Carlo simulation and used for the Spectrometer Section of HRS. (a) Shape of particle distribution with respect to horizontal angle a and kinetic energy deviation dk . The horizontal axis has units of mrad and the vertical axis is dimensionless. Blue particles are particles that can be transmitted through the Spectrometer Section, whereas red ones are rejected. We chose to center particles with $dk = 0.0825$, to maximize the transmission rate in the uncorrected system, thus most transmitted particles have relatively high kinetic energies. (b) Histogram showing the particle distribution with respect to kinetic energy deviation dk . Blue line represents the total distribution, red for rejected particles, and green for transmitted ones.

4.3 COSY Infinity: transfer matrices and particle tracking

The computer program we use for the simulation of moving particles in the Spectrometer Section and HTBL is called COSY Infinity (simply referred as COSY). COSY is able to simulate the effect of the different magnetic field shapes on the trajectories of particles by integrating the field derivatives along the system to produce

transfer matrices. These matrices allow the tracking of changes in positions x and y , angle a and b , and kinetic energy deviation dk as particles travel through the magnetic elements (i.e. multipole or dipole) in the ion optical system [11].

The basic principle of how we track the changes of positions, angles and energy deviation is through matrix multiplication shown in Eq. 4.1. COSY is able to produce transfer matrices, a matrix with each element representing the relation between old coordinates to new coordinates after going through the magnetic fields (e.g. $(x_f|a_i)$ represents the relation between final horizontal position x_f and initial horizontal angle a_i). By multiplying the transfer matrix with each original particle trajectory, a vector with elements x_i, a_i, y_i, b_i, dk_i , it will return the new particle trajectory as a vector with x_f, a_f, y_f, b_f, dk_f . The dots in the particle trajectory vectors and transfer matrix refer to the other particle coordinates that are less important than positions, angle and energy deviation.

$$\begin{bmatrix} x_f \\ a_f \\ y_f \\ b_f \\ dk_f \\ \vdots \end{bmatrix} = \begin{bmatrix} (x_f|x_i) & (x_f|a_i) & (x_f|y_i) & (x_f|b_i) & (x_f|dk_i) & \dots \\ (a_f|x_i) & (a_f|a_i) & (a_f|y_i) & (a_f|b_i) & (a_f|dk_i) & \dots \\ (y_f|x_i) & (y_f|a_i) & (y_f|y_i) & (y_f|b_i) & (y_f|dk_i) & \dots \\ (b_f|x_i) & (b_f|a_i) & (b_f|y_i) & (b_f|b_i) & (b_f|dk_i) & \dots \\ (dk_f|x_i) & (dk_f|a_i) & (dk_f|y_i) & (dk_f|b_i) & (dk_f|dk_i) & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \cdot \begin{bmatrix} x_i \\ a_i \\ y_i \\ b_i \\ dk_i \\ \vdots \end{bmatrix} \quad (4.1)$$

The matrix multiplication shown in Eq. 4.1 is only in first-order, where we assume

linear relations between each pair of coordinates. However, in our fifth-order simulation, coordinates have more complicated relations, up to fifth-order polynomials. For example, the relation between x_f and a_i can be a Taylor expansion up to fifth-order in Eq. 4.2.

$$x_f = (x_f|a_i)a_i + (x_f|a_i^2)a_i^2 + (x_f|a_i^3)a_i^3 + (x_f|a_i^4)a_i^4 + (x_f|a_i^5)a_i^5 \quad (4.2)$$

where $(x_f|a_i^k)$ refers to k^{th} -order relation between x_f and a_i for $k = 1, 2, 3, 4, 5$. In addition, there are other relations among three or more coordinates. For instance, $(x_f|a_i^2y_ib_idk_i)$ and $(x_f|a_iy_ib_i^2dk_i)$ refer to fifth-order relations among all five coordinates. This results in a much larger and more complex transfer matrix which is cumbersome to show here.

COSY computes one transfer matrix from the beginning up to a given position in the ion optical system (e.g. from the beginning of HTBL to FB1). By doing the matrix multiplication at each element, we can monitor the trajectories of particles travelling through the whole system. The ray plots like Figure 3.2 and 3.4 are produced by connecting the particles after each element with smooth curves.

4.4 Optimization objectives

The tuning process of both the Spectrometer Section and HTBL is the optimization to be considered, in which we want to find the best magnetic field strengths for all of the higher-order magnetic elements in the systems. To quantitatively optimize

the systems, we need to define the optimization objectives by mathematical expressions involving physical observables (variables that can be physically measured). For example, the optimization objective for the Spectrometer Section is simply the transmission rate, which can be measured as the fraction of the number of transmitted particles to the total number of incoming particles.

As mentioned in section 3.2.2, the optimization of the HTBL needs to consider the transmission rate, the beam spot size at FS0, and the clarity of the dispersive foci at FB1 and FB3. The transmission rate can be calculated using the same method as for the Spectrometer Section. The beam spot at FS0 (at the end of the HTBL) is a two-dimensional distribution in the xy -plane. The uncorrected beam spot produced by the simulated ^{40}Mg beam distribution is shown in the upper pane of Figure 4.2. To describe the spot size, we assume the distribution in both x - and y -directions follows Gaussian distribution and use the standard deviations of fitted Gaussians as the objectives, labeled as $x\text{RMS}$ and $y\text{RMS}$ (RMS stands for root mean square, the way to calculate standard deviation).

For the dispersive foci at FB1 and FB3, the HTBL should separates particles horizontally purely according to momentum. Thus, at the dispersive foci we expect that for a certain value of momentum, which is related to the coordinate of kinetic energy deviation dk (the fractional energy deviation from the average kinetic energy), the particles should focus at a single, precise horizontal position. This results in a linear relationship also between the position x and dk at these dispersive focal planes. Figure 4.2 (middle and lower panes) show the linear relationship between x and dk at FB1 and FB3 respectively.

We divide the continuous particle distribution into 100 slices in dk , with each slice expected to be a thin Gaussian distribution in the case of a pure, momentum-dispersive focus. We then take the average standard deviation of x over all the slices as an objective to be minimized, since this describes the quality of the dispersive foci. These average slice standard deviations at each focus are denoted x_{FB1} and x_{FB3} in the objective function (Eq. 5.4). The smaller the values for x_{FB1} and x_{FB3} are, the better the quality of the dispersive foci present. Therefore, the overall optimization objective can be described using transmission rate, x_{RMS} , y_{RMS} , x_{FB1} , and x_{FB3} .

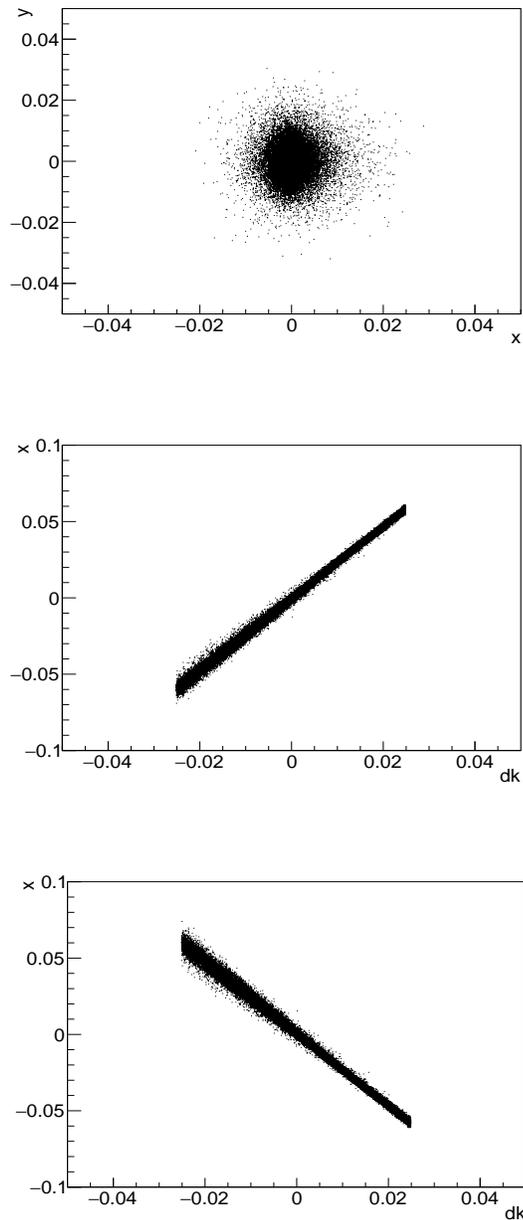


Figure 4.2: Top: uncorrected beam spot at FS0 at the end of the HTBL. We assume a Gaussian distribution in x and y . Middle and bottom: Graphs show the linear relationship between x and dk at the dispersive focal planes FB1 and FB3 respectively. Note that the linear relation is not perfectly uniform, which indicates blurred dispersive foci corresponding to Figure 3.4 bottom. The units for x and y are meters, and dk is dimensionless.

Chapter 5

Particle Swarm Optimization (PSO)

5.1 Basic function of PSO

In section 3.2, we explain why we need to optimize the field strengths of the higher-order correction elements (i.e. the sextupoles and octupoles inside multipoles) for the Spectrometer Section and HTBL. Optimization here means finding the magnetic field settings to improve the spectrometer performance as described quantitatively relative to certain tuning goals. For both systems, the optimization involves multiple variables (12 for the Spectrometer Section and 48 for HTBL). To systematically solve the optimization problem for both the Spectrometer Section and HTBL, we use a multi-dimensional global optimization algorithm, Particle Swarm Optimization (PSO), to

numerically determine the best magnetic field settings for each magnetic element through a series of steps. The word “global” indicates that PSO is designed to avoid finding and being trapped by local best solutions, and “multi-dimensional” means PSO is able to optimize multiple variables simultaneously.

PSO is a parallel evolutionary optimization technique first developed in 1995 by Kennedy and Eberhart [9]. “Parallel” implies that multiple trial solutions can be evaluated simultaneously—allowing simulation-based optimizations to run orders of magnitude faster on multi-core compute machines—and “evolutionary” implies that the algorithm optimizes over a series of steps [21]. The algorithm is based on a social behavior metaphor—how a swarm of bees or a flock of birds find the location of a food source in a three-dimensional space. At first, the bees move in random directions gathering information about the position of food sources in each of their local neighborhoods. As the swarm explores the space, one of them will reach the closest position yet found to food and communicate that “best” position to the other bees in the swarm, telling them to focus further exploration in that region. All the other randomized swarm members will then accelerate toward the current best position (either changing the direction or magnitude of their velocity vector in the multi-dimensional solution space). During the process, another bee may find an even closer (improved) best position, and once this new best position is communicated to the swarm, all other bees will then experience accelerations toward the updated best position. After many optimization steps, bees eventually will gather together (i.e. converge) at the global best position, metaphorically, the location of the food.

With this similar technique, we develop the optimization algorithm replacing

“bees” with generalized swarm particles. A swarm particle represents a multi-dimensional vector, with each dimension representing a complete set of higher-order magnetic field settings for the system (i.e. a 12-dimensional vector for the Spectrometer Section and a 48-dimensional vector for the HTBL). The reader is advised to carefully avoid confusing the swarm “particles” moving through solution space with the physical particles (real or simulated ions) being transmitted through the spectrometer in the experiment. These two are not the same thing.

The algorithm works as follows. Before the optimization starts, we define the optimization objective using a mathematical function, which includes all the values we want to reduce in order to improve the system performance (lower objective function value is always better). At the beginning (step 0), twenty swarm particles are given randomized values from a Gaussian distribution for each dimension of the position and velocity vectors, within the limits of the magnetic field strength. The higher-order magnet tunes defined by each swarm particle’s position are taken as input to the COSY simulation of the system and—using Monte Carlo calculations—the performance of each tune is rated according to the objective function value, with one of them having the lowest objective function value. The particle position that achieved the best objective function value so far (**gBestObj**) is then called the global best position (\vec{x}_{gbest}). The rest of particles in the swarm will then accelerate with randomized accelerations toward the current global best position (\vec{x}_{gbest}) and its personal best position (\vec{x}_{pbest} —the best value achieved by that particular swarm particle so far) while continuing to explore the solution space trying to find an even better one. Equations

5.1 and 5.2 show how a swarm particle evolves at each optimization step.

$$\vec{x}' = \vec{x} + \vec{v} \quad (5.1)$$

where \vec{x} and \vec{x}' are the old and new positions of the swarm particle, whereas \vec{v} is the current velocity. PSO then updates the velocity of the swarm particle by Eq. 5.2:

$$\begin{aligned} \vec{v}' = & \text{inertia} \cdot \vec{v} \\ & + \text{acceleration} \cdot \text{randUniform}(0, 1) \cdot (\vec{x}_{\text{gbest}} - \vec{x}) \\ & + \text{acceleration} \cdot \text{randUniform}(0, 1) \cdot (\vec{x}_{\text{pbest}} - \vec{x}) \end{aligned} \quad (5.2)$$

where \vec{v} and \vec{v}' are the old and new velocities of one swarm particle, \vec{x}_{gbest} is the current global best position (shared by all the swarm particles), and \vec{x}_{pbest} is the personal best position for this swarm particle. The random number from the uniform distribution provides flexibility for the swarm particles to explore over the whole solution space [9]. Parameters `inertia` and `acceleration` are the internal parameters of PSO, which will be discussed in section 5.2.

In the ideal case, where the algorithm finds the real global best value at step 0, the rest of swarm particles will eventually converge to the close neighborhood of that gBest value over a series of steps. It is far more likely that during the process of converging, another swarm particle finds a better value at some later optimization step. This new gBest will then replace the previous gBest value and all other swarm members will be attracted toward it, as the new gBest value. The output is the vector with each dimension (magnetic element) optimized to provide the best overall gBest value

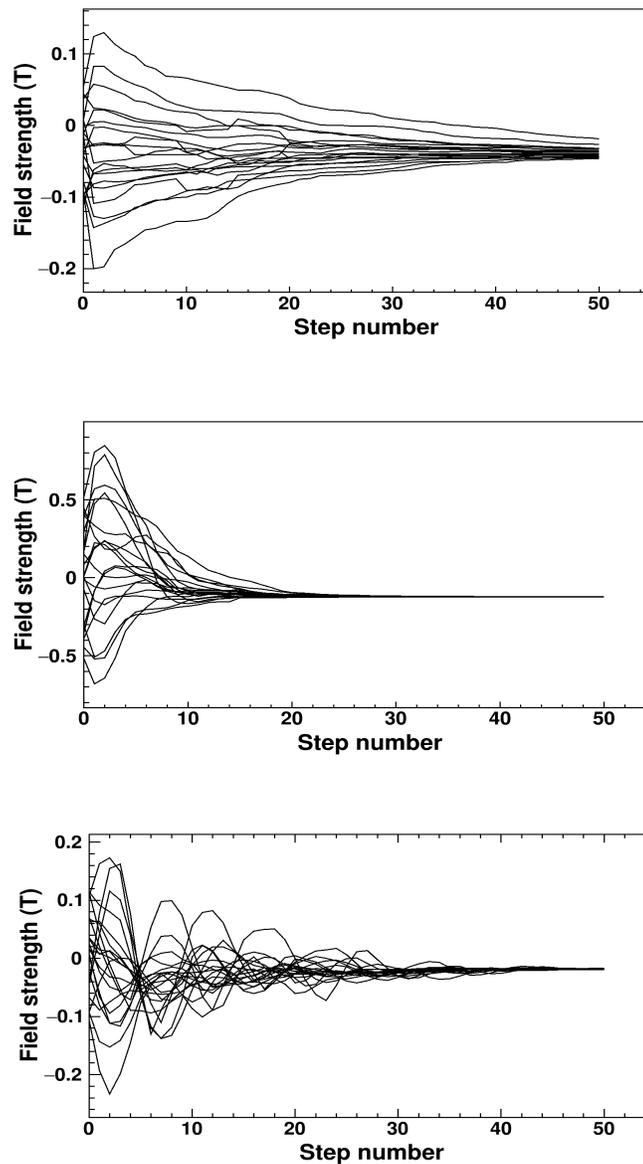


Figure 5.1: Plots showing the evolution of one dimension of a particle swarm during an optimization for different internal PSO parameter settings. Here the different swarms evolve analogously to over-damped [upper plot], critically-damped [center plot], and under-damped [lower plot] mechanical oscillations. The randomized initial values all converge or tend to converge toward one position (i.e. field setting) over the course of 50 optimization steps. The internal parameter settings for the swarm plots are: top - (0.1, 0.1); middle - (0.475, 0.25); bottom - (0.85, 0.4), which will be discussed in section 5.2.

when we stop the optimization at 50 steps. Due to the significant computation time required, we choose to optimize the system for 50 steps, which takes between 6 and 12 hours (depending on the size of the system and the number of particles simulated in Monte Carlo) to complete a single simulated optimization. The exploration of one dimension of a swarm (representing the setting of a single magnetic field strength) can be illustrated using swarm plots in Figure 5.1. The difference among three swarm plots will be explained later in this chapter.

For the Spectrometer Section and HTBL, the complete behavior of the multi-dimensional swarms is shown in Figures 5.2 and 5.3 respectively, with each pane representing one dimension of the evolving swarm.

5.2 Internal parameters: inertia and acceleration

There are two essential internal parameters in the PSO algorithm that determine how the swarms explore the multi-dimensional solution space in search of the global best location:

- **inertia** - the tendency of swarm particles to maintain their initial velocities. Mathematically it is the fraction of the swarm particles' velocities that will be preserved in the next step.
- **acceleration** - how quickly swarm particles accelerate toward gBest values.

These internal parameters are unitless because they apply as scaling factors in cal-

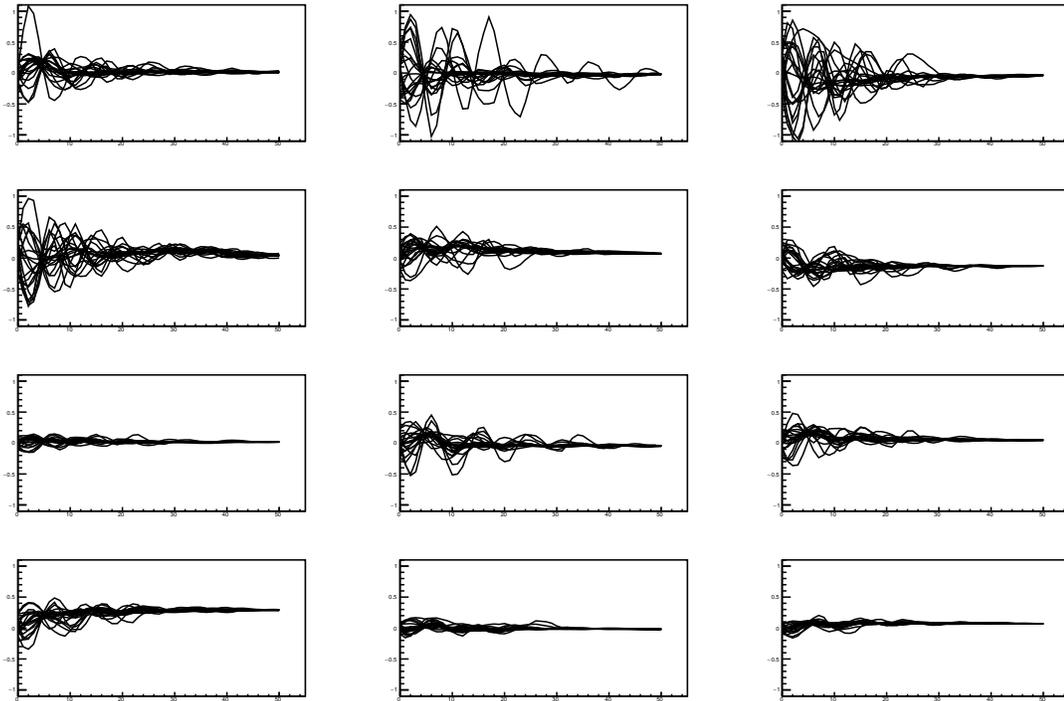


Figure 5.2: Complete swarm plots for the Spectrometer Section. Each swarm plot represents one dimension of the swarms (one magnetic field strength value). The horizontal axes are steps from 0 to 50, and vertical axes are magnetic field strength in Tesla with the same upper and lower limits set to be ± 1 T.

ulation. The parameter pair (*inertia*, *acceleration*) defines one internal parameter setting of the PSO. The ranges of *inertia* and *acceleration* values that provide reasonable dynamic behavior of the swarm particles are $[0,4]$ for *inertia* and $[-1,1]$ for *acceleration* [21]. Note that the *acceleration* parameter does not give the acceleration of the swarm particles directly. Rather, the *acceleration* parameter determines the range of randomized accelerations possible. The effects of changing the internal parameter settings can be observed in the previous Figure 5.1, with the top one from setting $(0.1, 0.1)$, middle one from setting $(0.475, 0.25)$, and bottom one from setting $(0.85, 0.4)$. In general, we prefer the setting whose swarm plot has a pattern of

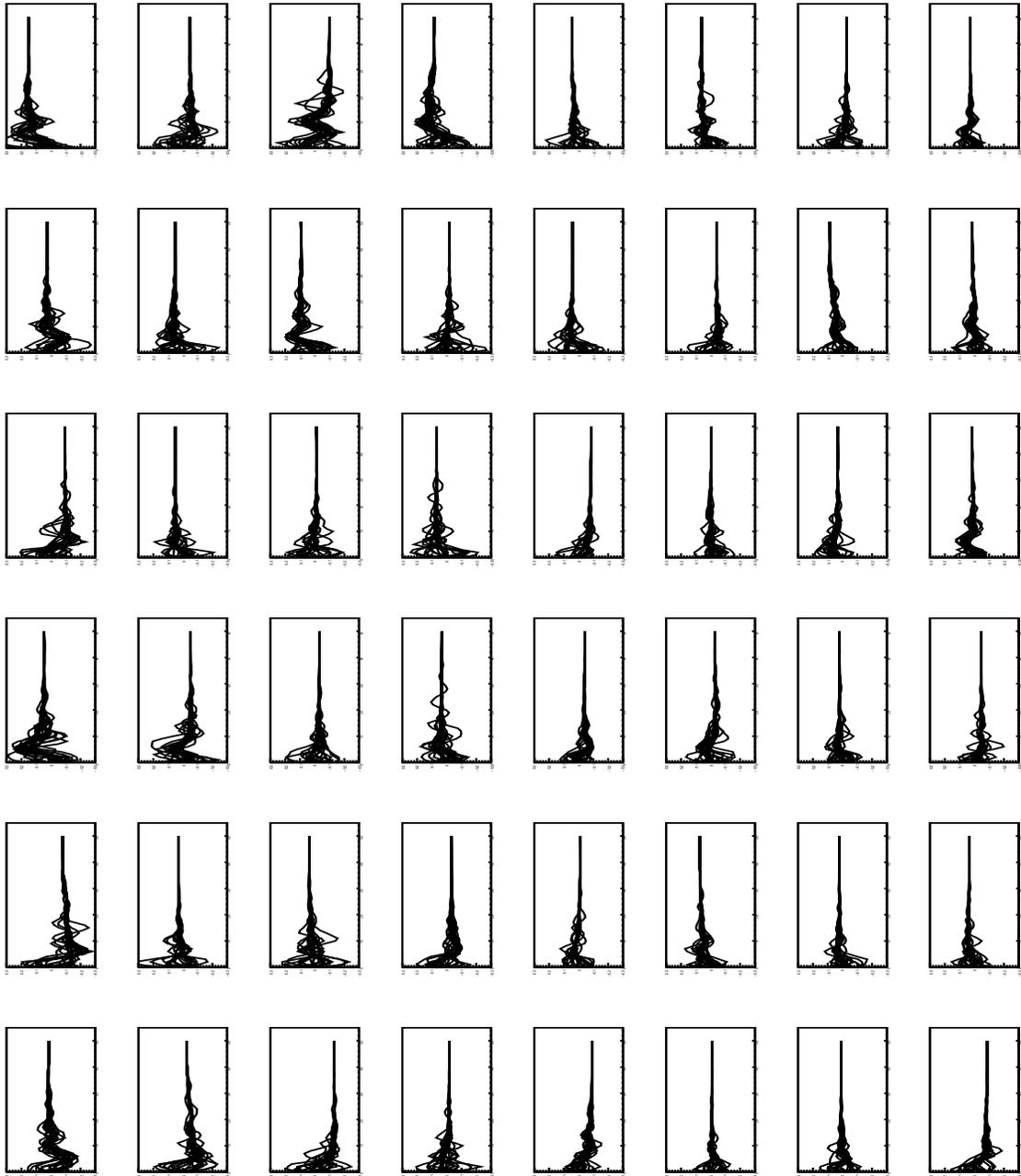


Figure 5.3: Complete swarm plots for a higher-order optimization for the HTBL. The left twenty four plots (top if viewed in landscape) are the sextupoles' optimizations and the right twenty four plots (bottom if viewed in landscape) are the octupoles' optimizations. The horizontal axes (in this case increasing up the page) are steps from 0 to 50, and vertical axes (in this case increasing to the left) are magnetic field strength in Tesla with the same upper and lower limits set to be ± 0.3 T.

under-damped oscillation because we want the swarm to be able to explore the spaces repeatedly while still converging relatively quickly (within 50 steps). Therefore the preferred setting among these three is (0.85, 0.4).

However, we cannot simply conclude that the internal parameter setting (0.85, 0.4) is the best setting of the PSO for the optimization of the HRS without systematic verification. We aim to find the internal parameter setting of the PSO that gives the best performance for both the Spectrometer Section and the HTBL by exploring a range of internal parameters centered at (0.85, 0.4) changing up and down in steps of 0.05 for both parameters (i.e. testing *inertia* values of 0.8, 0.85, and 0.9 together with *acceleration* values of 0.35, 0.4, and 0.45). By analyzing the performance of the PSO under each internal parameter setting, we can identify the best setting for HRS optimization. To make statistically significant conclusion on the robustness of the optimization based on each parameter setting, we complete twenty optimizations with same parameter setting.

5.3 Defining objective functions

As mentioned in section 5.1, the PSO algorithm optimizes the higher-order magnetic field strength settings for HRS by reducing the objective function value. For each of the unique optimization goals for the Spectrometer Section and the HTBL, we design a specific objective function. For the single-objective optimization of the

Spectrometer Section, we define the objective function as

$$\text{gBestObj} = 1 - \text{fracTrans} \quad (5.3)$$

where gBestObj represents the global best objective function value and fracTrans represents the transmission rate defined as the fraction of beam particles entering the system that are successfully transmitted through.

For the multi-objective optimization of the HTBL, we need to take the importance of each objective into account. Due to the design purpose of HTBL [12], the most significant objective is the beam spot size in x -direction, referred as xRMS as it determines the spectrometer's resolving power. The second most significant objective is the quality of the dispersive foci at FB1 and FB3, referred as xFB1 and xFB3 . The transmission rate fracTrans and beam spot size in the y -direction— yRMS —are the least significant objectives. Thus we need a weighted objective function to appropriately prioritize the reduction of the various objective function components. We first apply weights so that the final desired values of each objective function term will be of the same numerical order. For instance, to make xRMS (typically around 0.002 m) in the same number order as $1 - \text{fracTrans}$ (typically around 0.01 for 40Mg in the HTBL), we add a multiplicative factor of five to xRMS . To properly emphasize each of the remaining terms, we add more factors from high to low in order, finally arriving at the weighted objective function for the HTBL as:

$$\text{gBestObj} = 1 - \text{fracTrans} + 25 \cdot \text{xRMS} + 3 \cdot \text{yRMS} + 10 \cdot \text{xFB1} + 10 \cdot \text{xFB3} \quad (5.4)$$

The computational framework we use for PSO is called ROOT, which is a data analysis framework written in C++ capable of statistical calculation and visualization [2].

Chapter 6

Data acquisition and analysis

6.1 Method of group simulation and analysis

As different internal parameter setting (**inertia, acceleration**) of the PSO affects the performance of the optimization, the purpose of this project is to find the best internal parameter setting for the Spectrometer Section and HTBL. To define a “good” set of internal parameters, we consider the quality of the final optimization results (i.e. the global best objective function value, lower being better) and the reliability, since we need to ensure when we apply PSO to a real system, it can optimize the system reliably (i.e. without the need for many repeated optimizations to find a well-performing tune).

The way we quantify the goodness of the final result and reliability is to use the

average result of multiple optimizations with the same set of internal PSO parameters. Specifically, we take the average of the best 60% of the optimization runs, because we only want the PSO to work well most of the time instead of always, and the top 60% also represent the best results for one setting. Due to the time limit, we decide to operate the PSO 20 times for each one of the internal parameter settings. To produce different optimizations for each of the 20 trial runs, we use unique random number generator seeds inside the PSO algorithm, doing our best to provide randomization processes of initial swarm positions and velocities covering the whole exploration spaces of swarms.

We designate the internal parameter space (collection of all internal parameter settings) that we want to explore as being centered at setting (0.85, 0.4), which has been tested to be a good setting in earlier experimental online optimization [1]. The exploration of internal parameter space for optimizing the transmission of the Spectrometer Section for fission fragments was completed in the summer of 2021, testing all possible combinations of the parameter settings: **inertia** [0.8, 0.85, 0.9] and **acceleration** [0.35, 0.4, 0.45] (a total of nine settings). For the more challenging multi-objective HTBL optimization, we explored a wider internal parameter space, hoping to extract more information about the overall shape of the internal parameter space because earlier exploration in the Spectrometer Section did not identify a minimum within the range. The PSO parameter space for the optimization of the HTBL included **inertia** values [0.75, 0.8, 0.85, 0.9, 0.95] and **acceleration** values [0.3, 0.35, 0.4, 0.45, 0.5] (25 settings in total).

6.2 Results

6.2.1 Optimization for the Spectrometer Section

Before evaluating the performance of the PSO with different internal parameter settings, we first compare the optimization objectives and the overall objective function values (defined as Eq. 5.3) before and after the optimization to confirm that the system has been successfully optimized. Table 6.1 shows how the objectives were improved by the optimization.

The transmission rate of the fission fragments through the Spectrometer Section has been successfully improved by 10% for one of the PSO internal parameter settings. We are also interested in seeing how the trajectories of particles change in the ray plot. Figure 6.1 shows the fifth-order ray plot after optimization. The difference is not obvious because the ray plot only shows the trajectories of selected characteristic rays through the system, rather than the full, realistic particle distribution. However, we do see fewer particles escape from the system at the entrance of the 5th “M5” multipole and clearer dispersive foci (not our priority and thus not included in the objective function) at the end of the Spectrometer Section.

Table 6.1: Improvement of optimization objectives for the Spectrometer Section with one PSO internal parameter setting.

	Uncorrected	Corrected
gBest Obj.	0.425	0.370
fracTrans	0.575	0.630

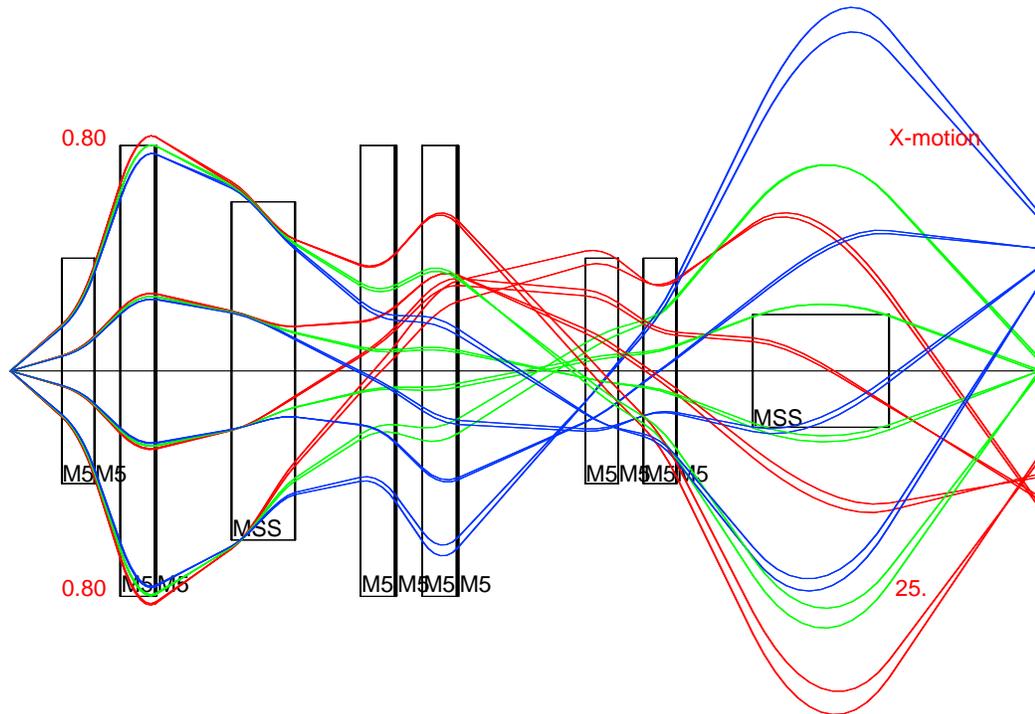


Figure 6.1: Corrected fifth-order plot of rays through the Spectrometer Section on the horizontal plane (x direction) with optimized higher-order magnetic field strengths. A clearer dispersive focus at FS2 than uncorrected fifth-order ray plot. Note that the goal of optimization was only the transmission rate.

After we confirm the effect of the optimization, we compare the performance of the different internal parameter settings by comparing the average of the global best objective function value obtained for the best 60% of the optimization runs from each setting. We then produced a contour plot with the internal parameter space on the x and y axes and the corresponding best 60% averaged global best objective function values on the z axis. Figure 6.2 shows the performance of the optimizer over the explored internal parameter space for the Spectrometer Section. Note that lower gBest objective function values imply better optimization results.

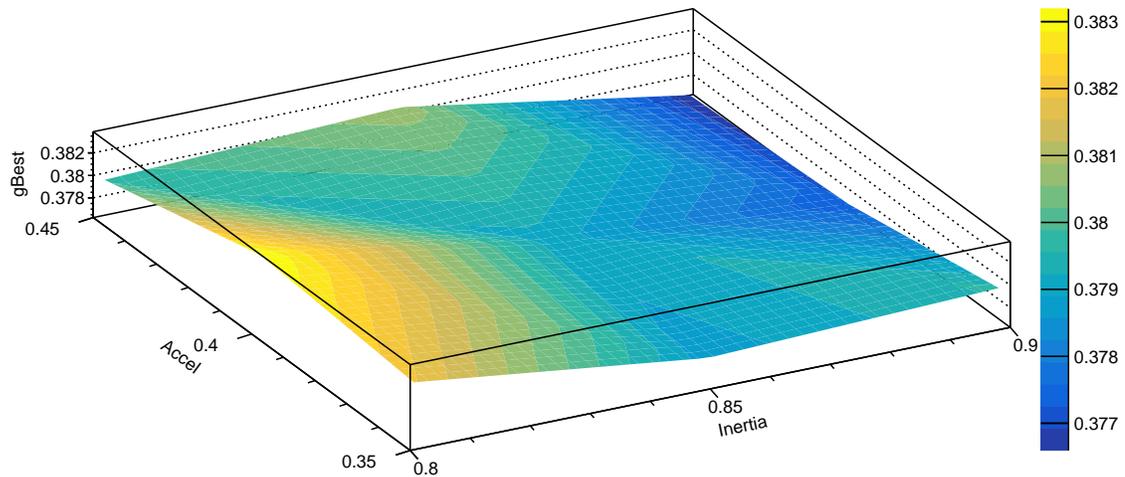


Figure 6.2: Plot of internal parameter space of PSO for the Spectrometer Section. The internal parameter space is on the x and y axes and the corresponding best 60% averaged global best objective function value is on the z axis. Setting $(0.9, 0.45)$ gives the best performance (top right in blue).

By looking at the overall shape, we clearly see a lowest spot at the top right corner of the parameter space, which corresponds to the setting $(0.9, 0.45)$. The parameter space tends to get systematically lower when exploring toward the direction with higher inertia and acceleration. Due to the limited size of explored parameter space, we are unable to demonstrate that $(0.95, 0.4)$ is the best parameter setting for the Spectrometer Section. The analysis of the internal parameter space for the Spectrometer Section is simple because it uses a single-objective optimization.

6.2.2 Optimization for HTBL

The optimization of the HTBL is much more complex than that of the Spectrometer Section, mainly because it has more correcting elements and includes multiple objectives that must be simultaneously improved by the optimization algorithm. As PSO optimizes HTBL by lowering the objective function value defined in Eq. 5.4, we are able to compare the optimization objectives before and after the optimization, as shown in Table 6.2. Together with the uncorrected and corrected fifth-order ray plot of HTBL (Figure 3.4 and 6.3), we are able to see clear and gratifying improvement of the system performance.

In Table 6.2, all of the individual optimization objectives have been improved in addition to a reduced global best objective function value after the optimization. Individually, xRMS and xFB1 have been improved the most, with a fractional decrease of 45% and 27% respectively (even though xRMS only decreases by 0.8 mm, it tremendously improves the momentum resolving power of the system). The behavior can be explained by our design of the objective function. As xRMS has the greatest weight in the objective function, it is encouraged for the optimizer to improve xRMS

Table 6.2: Improvement of optimization objectives for HTBL with one PSO internal parameter setting.

	Uncorrected	Corrected
gBestObj	0.1187	0.0764
fracTrans	0.9813	0.9825
xRMS (at FS0)	0.0018 (m)	0.0010 (m)
yRMS (at FS0)	0.0044 (m)	0.0040 (m)
xFB1 (average dk slice xRMS at FB1)	0.0015 (m)	0.0011 (m)
xFB3 (average dk slice xRMS at FB3)	0.0013 (m)	0.0012 (m)

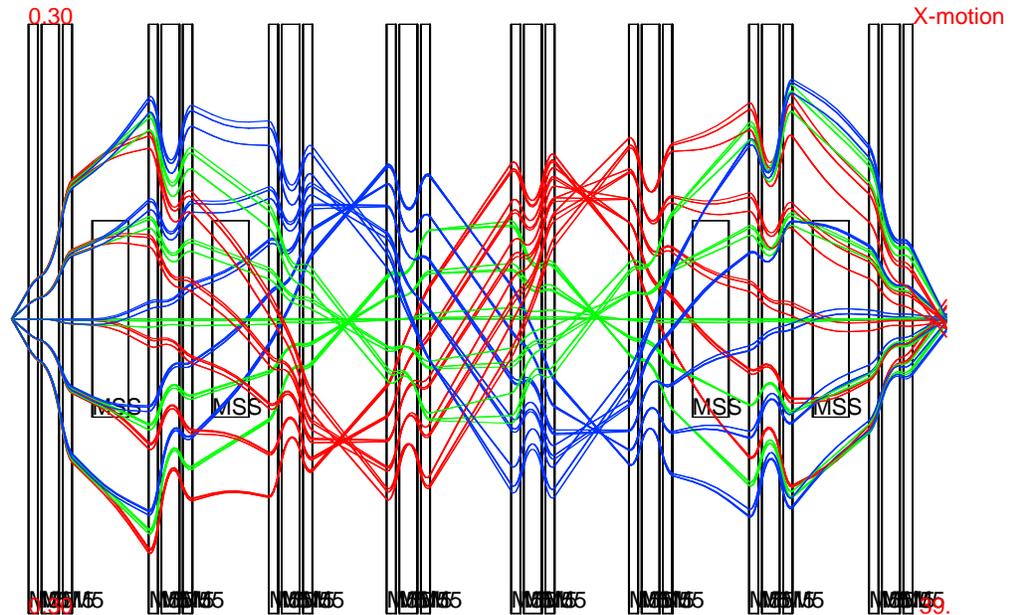


Figure 6.3: Corrected fifth-order plot of selected characteristic rays through the HTBL in the horizontal plane with optimized higher-order magnetic field strengths. Clearer dispersive foci at FB1 and FB3 than in the uncorrected fifth-order plot are evident, as well as a much smaller beam spot in x -direction at FS0. One thing that is unexpected is that the trajectories become more symmetric along the central axis, which is significantly different from the asymmetric shape before optimization.

more than other objectives. Similarly, x FB1 and x FB3 are expected to be improved moderately since they have the second significant weights. The reason why x FB1 is improved more than x FB3 could be that a clear dispersive image at FB1 is sufficient to provide the desired performance even without improving the dispersive image at FB3. The other two objectives, fracTrans and y RMS have not changed much due to their low weights in the objective function. These changes of optimization objectives are also reflected in Figure 6.4, where we plot the beam spot at FS0 and linear relation

between x and dk at FB1 and FB3.

Table 6.2 shows the best optimization results we get among all of the PSO trial runs, where all the objectives are improved by some amount. Nevertheless, as we analyze the optimization results using different internal parameter settings, we found that in most cases, not all of the objectives are improved, especially the ones that have relatively low weights in the objective function (i.e. `yRMS` and `fracTrans`). The low weights of these terms can result in their actually getting worse over the course of the optimization, but it is still necessary to include them in the objective function to avoid getting out of control. Therefore to see how each objective evolves during the optimization, we plot the objectives along with the optimization steps for one internal parameter setting, as shown in Figure 6.5 to 6.8. In short conclusion, the objectives that have relatively high weights are improved continuously over optimization steps, and the ones with less significant weights keep fluctuating between being sacrificed and being improved.

Finally, we compare the performance of PSO optimizations with different internal parameter settings with contour plots. Similar to the analysis of how each optimization objective evolves in the trending graphs, we make a separate contour plot for each component of the objective function. Looking at the shape of the contour plot for the overall objective function value (Figure 6.9), we see a clear minimum close to the center and gentle increment over the rest of the PSO parameter space. Since lower objective function values represent better overall performance, we conclude that the best pair of internal parameters (illustrated by the darkest blue spot toward the top right corner) are the pair with `inertia` of 0.9 and `acceleration` of 0.45. The 60% mean

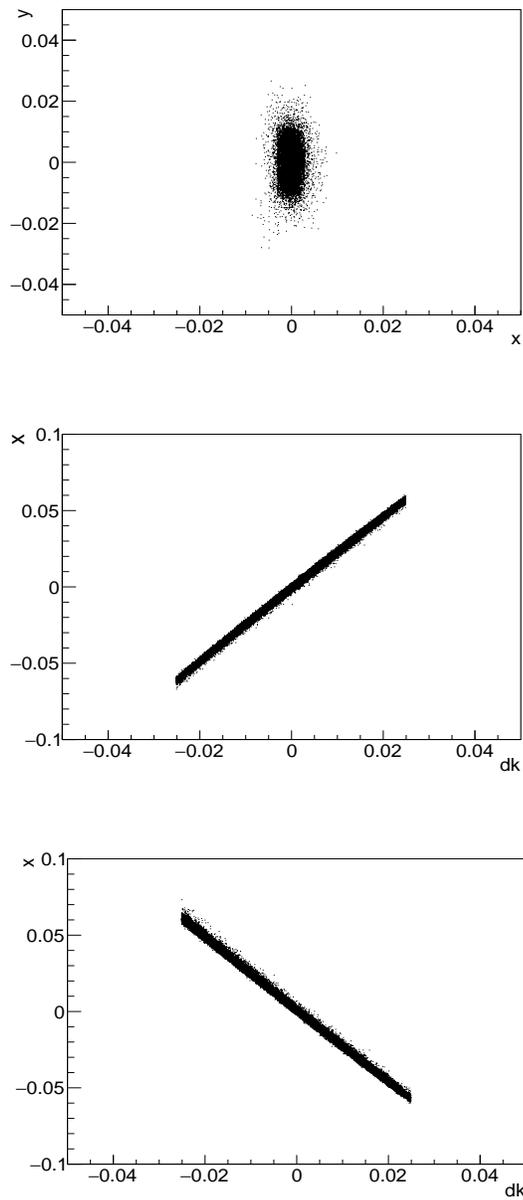


Figure 6.4: Upper: corrected beam spot at FS0 from HTBL with both axes in meters. Compared to the uncorrected beam spot in the upper pane of Figure 4.2, the beam spot size has been significantly reduced in the x -direction. Middle and lower: graphs showing the relationship between x and dk at dispersive foci FB1 and FB3, respectively, after optimization. x is in meters, and dk is dimensionless. The left half of the linear relation has become thinner, indicating better dispersive foci at FB1 and FB3 as shown in Figure 6.3.

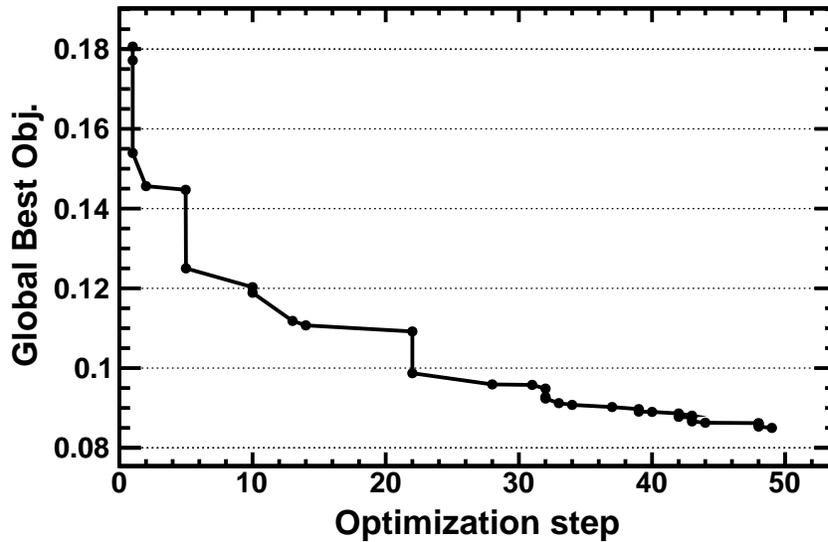


Figure 6.5: Graph of the evolution of the global best objective function value during optimization of the HTBL in one specific trial run of PSO. The global best objective function value decreases monotonically, which means the system is being optimized according to our definition of the optimization objective. Note that the vertical axis does not start from 0 (same for all the following evolution plots).

objective value for this parameter setting is 0.0798, which is approximately 10% lower than the 60% mean of the worst setting (0.0895). On the other hand, compared to the contour plots of the Spectrometer Section, the range of the parameter space has been increased in both dimensions, which gives more information about the shape of the whole parameter space.

In the ideal case, we expect to find one internal parameter setting of the PSO, which gives the best performance for the Spectrometer Section as well as the HTBL system. The comparison of parameter spaces of the Spectrometer Section and the one for the HTBL shows similar trending across the parameter space inertia [0.8, 0.9]

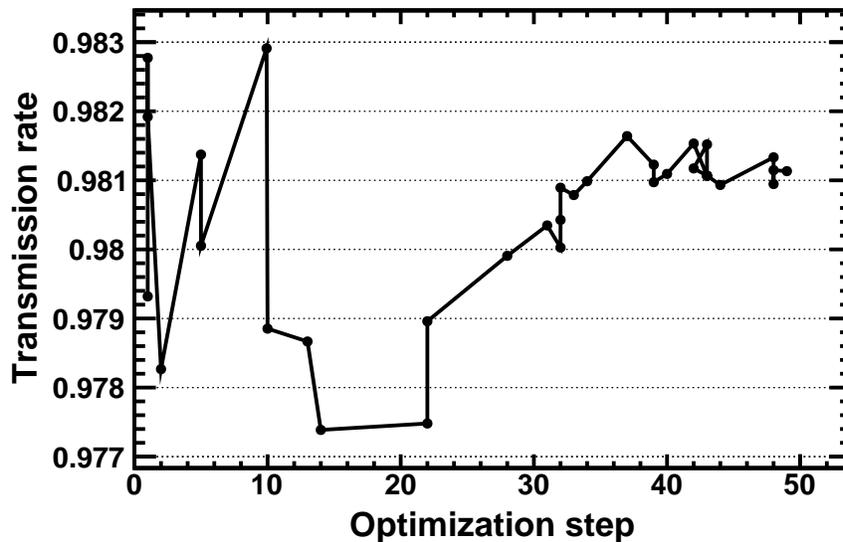


Figure 6.6: Graph of the evolution of the transmission rate during optimization of the HTBL in one specific trial run of the PSO. Since the transmission rate has a relatively low weight in the objective function, it is not improved significantly (only from 0.983 to 0.981 in this case).

and acceleration $[0.35, 0.45]$, which in both cases the setting at the top right corner $(0.9, 0.45)$ gives the lowest 60% mean objective value and the performance gets worse gradually across the diagonal.

Meanwhile, we have already obtained the best internal parameter setting for the HTBL system by simply finding the setting with the lowest 60% mean objective value. However, since the optimization involves a complicated objective function consisting of five objectives with different weights, we may learn more about the internal parameter space by considering the objectives one by one and making separate contour plots.

As we expected, some of the separated parameter spaces have similar shapes as the

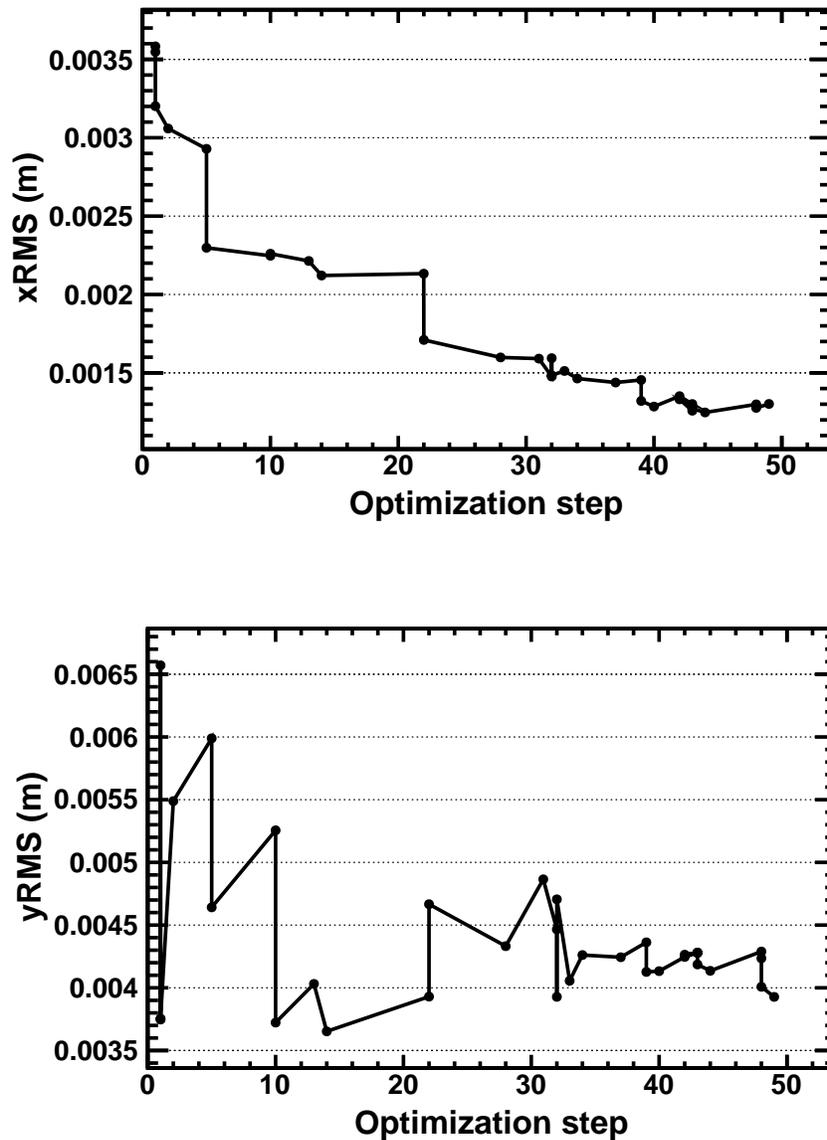


Figure 6.7: Top: Graph of the evolution of $xRMS$ during the optimization of the HTBL in one specific trial run of the PSO. Note that the shape is extremely similar to the one for the global best objective function value, because it has the highest weight. $xRMS$ decreases most of the time during the optimization process. Bottom: Graph of the evolution of $yRMS$ during the optimization of the HTBL in one specific trial run of the PSO. Since $yRMS$ has a relatively low weight in the objective function, it keeps fluctuating during the optimization process (not prioritized).

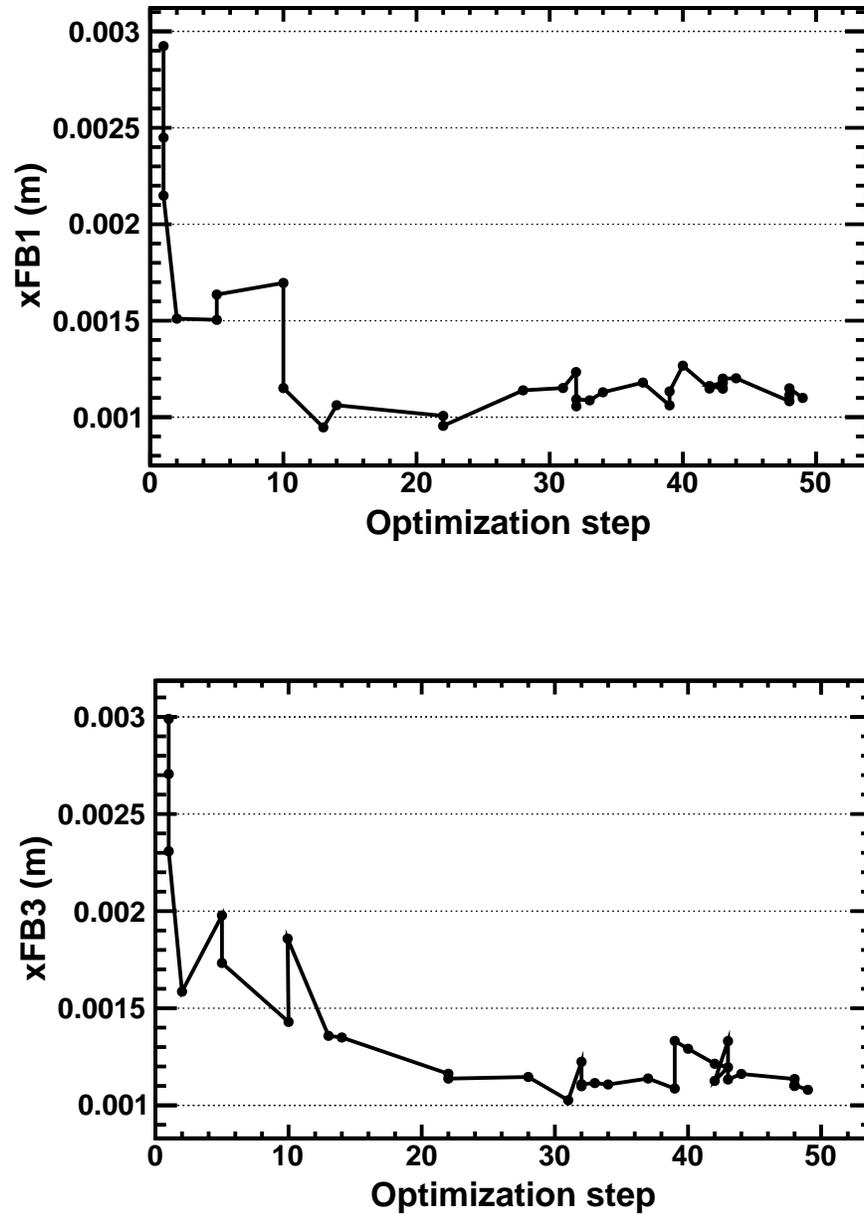


Figure 6.8: Top: Graph of the evolution of xFB1 during the optimization of the HTBL in one specific trial run of the PSO. Bottom: Graph of the evolution of xFB3 during the optimization of the HTBL in one specific trial run of the PSO. Both objectives decrease fast in early stage (within 20 steps) and fluctuate for the rest of steps since they have moderate weights in the objective function.

one for the overall objective function. For example, the parameter space for the $xRMS$ objective has almost exactly the same shape, with a blue valley sitting in the middle and gradually getting worse as it expands away from the lowest point. This result matches our prediction because the objective of $xRMS$ with the highest weighting factor contributes the most to the objective function. When the optimization reduces the overall objective function value over the steps, it must prioritize reducing the value for the $xRMS$, as a small change would result in a large difference for the objective value. We are able to draw the same conclusion quantitatively.

For the best result shown in Table 6.2, the $xRMS$ term occupies 30% of the overall objective value, with the second one, transmission rate, occupying 22%. By comparing the parameter space of the $xRMS$ objective to the one for overall objective function both quantitatively and qualitatively, we conclude that the internal parameter setting with better $xRMS$ result performs better in general.

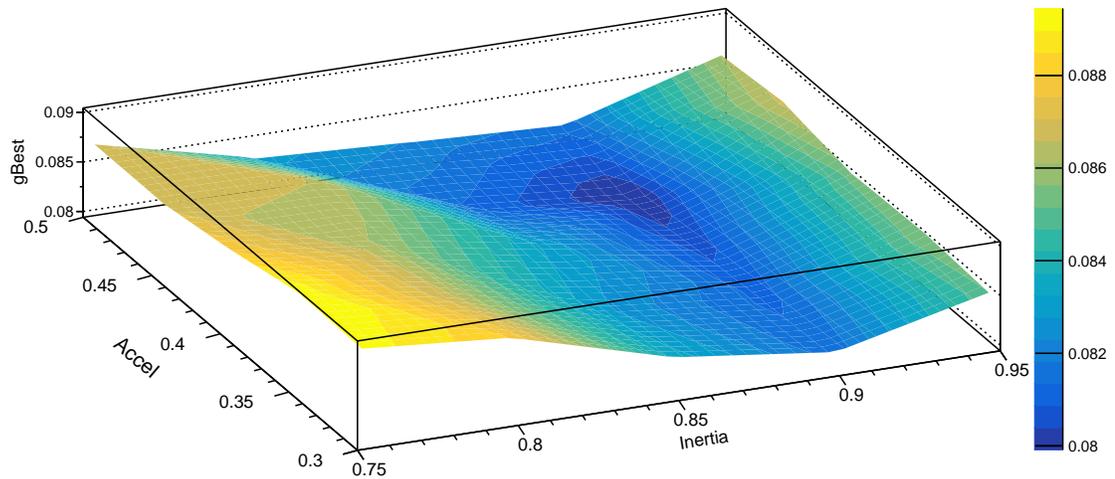


Figure 6.9: Contour plot of the mean of the best 60% achieved overall objective function values vs. the PSO internal parameters used for the HTBL optimization. Setting (0.9, 0.45) gives the best performance (near the top right corner in dark blue).

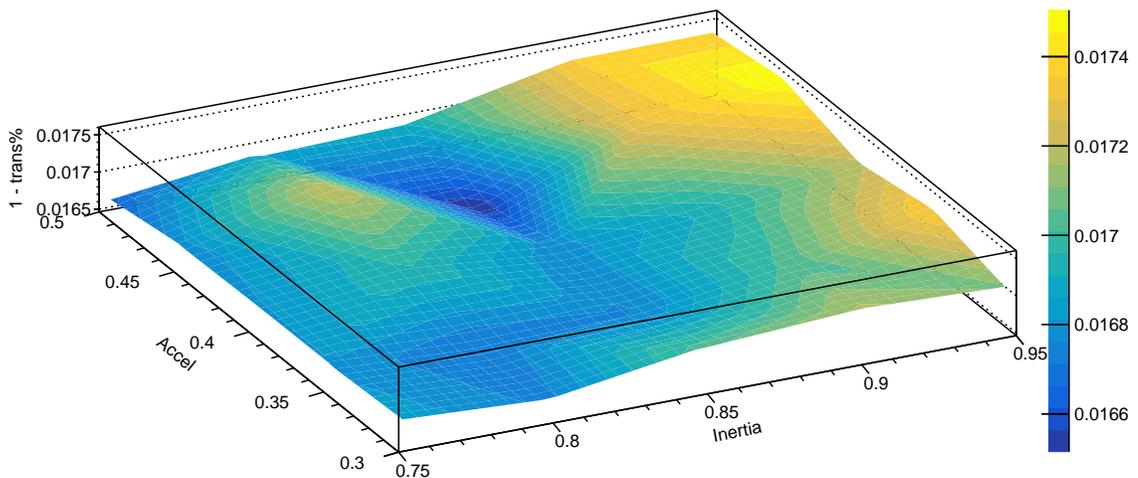


Figure 6.10: Contour plot of 1-fracTrans in the internal parameter space of the PSO for HTBL. Completely different shape than the one of global best objective function value, with overall flat shape and a step drop at the center.

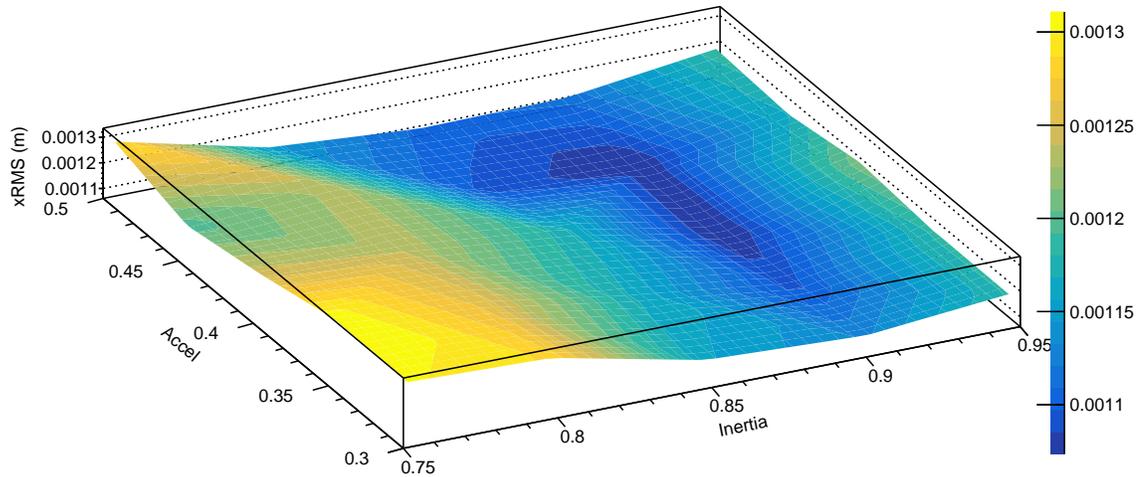


Figure 6.11: Contour plot of xRMS in the internal parameter space of the PSO for HTBL. It has the most similar shape as the one of global best objective function value, due to its highest weight in the objective function. Setting (0.9, 0.45) gives the best performance.

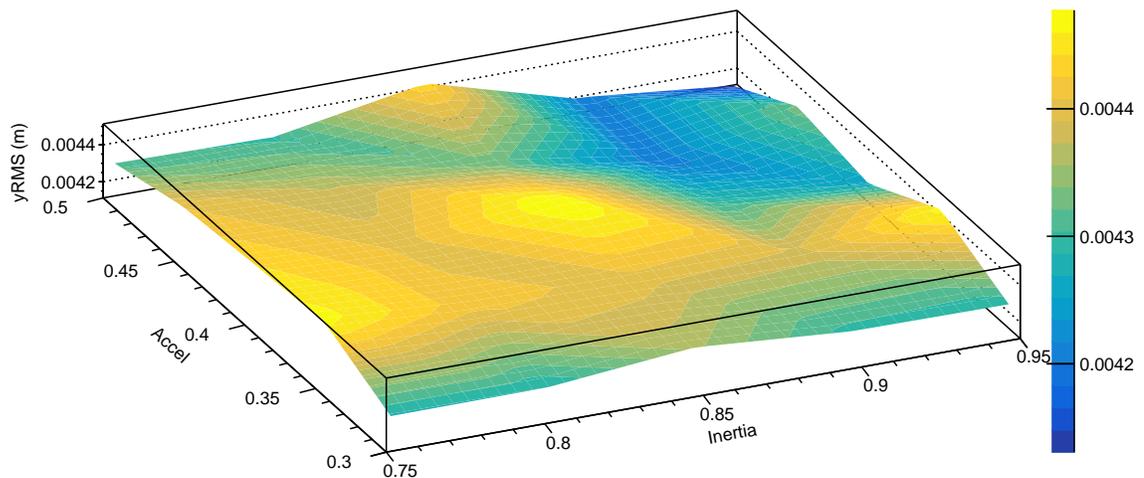


Figure 6.12: Contour plot of yRMS in the internal parameter space of the PSO for HTBL. Completely different shape with “hills” across most part of the space and the lowest point at (0.9, 0.5).

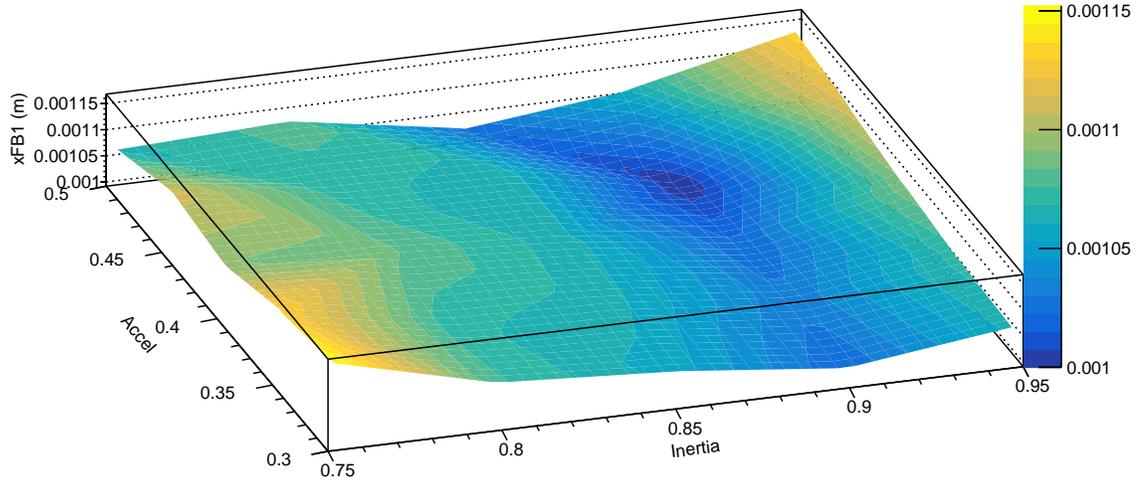


Figure 6.13: Contour plot of xFB1 in the internal parameter space of the PSO for the HTBL. It has a similar shape as the one of global best objective function value.

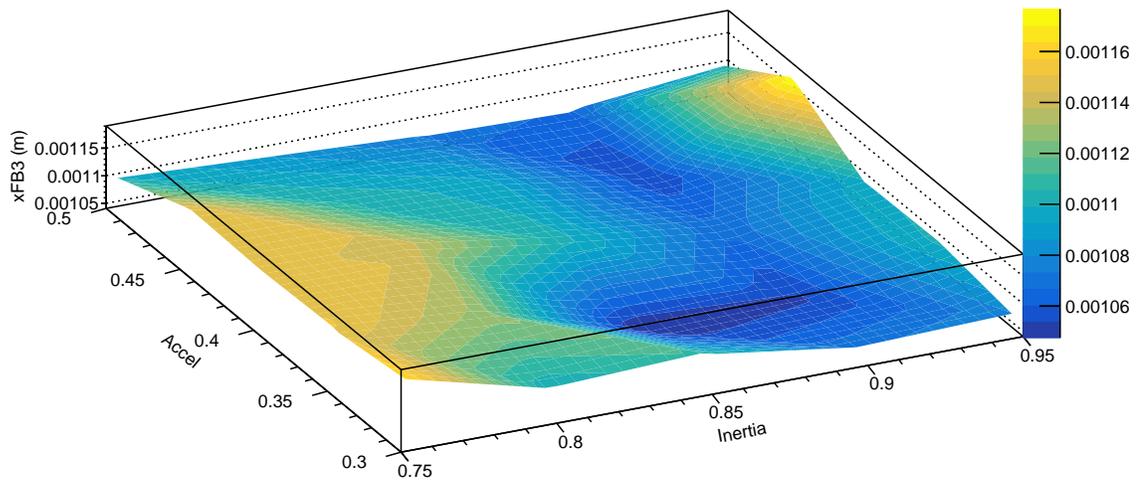


Figure 6.14: Contour plot of xFB3 in the internal parameter space of the PSO for HTBL. Completely different shape with two “valleys” and the lowest point at (0.85, 0.35).

Chapter 7

Conclusion and future

This project formally started in summer 2021 and ends in spring 2023. During this period, we successfully developed and tested the Particle Swarm Optimization for the Spectrometer Section and the High-Transmission Beamline in the High Rigidity Spectrometer. We explored the internal parameter spaces of PSO. From the results discussed in section 6.2, we conclude that for both ion optical systems, the best internal parameter setting (inertia, acceleration) of the Particle Swarm Optimization is (0.9, 0.45). This result may indicate that this internal parameter setting is well performing for a wide variety of ion optical systems and very different input particle distributions. For the Spectrometer Section, the optimization successfully improved the transmission rate from 57.5% to 63%. For the HTBL, the optimization dramatically shrank the beam spot size at FS0 in the x -direction by 44%, while also improving the dispersive foci at FB1 and FB3, which helps increase the resolving power of the downstream Spectrometer Section.

To complete our theory regarding how the internal parameter setting affects the performance of the PSO for optimizations of large-acceptance rare-isotope spectrometers, we should further explore the parameter space to describe the quality of the PSOs performance over a wider range. Instead of centering at the setting that we did in this work ($\text{inertia}=0.85$, $\text{acceleration}=0.4$), we should recenter the parameter space at $(0.9, 0.45)$, which has been shown to be the best setting for the cases we considered.

We should also try different input particle distributions such as Nickel-84, which is another benchmark rare-isotope production case for the HRS. Further, since the optimization algorithm includes multiple objectives in the objective function, we should consider the benefit of assigning the weights differently in the objective function for the HTBL to see how different objective functions affect the performance of the PSO. Our assigned weights successfully optimized the HTBL with an emphasis on shrinking the beam spot size at FS0, while preserving similar beam transmission and maintaining a similar final vertical spot size. We may consider balancing weights for the beam spot size at FS0 and for the quality of dispersive images at FB1 and FB3 for better overall performance of the HTBL. This would require several additional CPU-years of calculations to draw reliable conclusions comparable to those in the present work.

Appendix A

COSY Infinity code for spectrometer in section 2.5

```

INCLUDE '/home/ama018/optics/MA/hrs/cosyModel/cosy91-master/
  bin/COSY91' ;
PROCEDURE RUN ;

VARIABLE Q1 1 ; VARIABLE Q2 1 ;VARIABLE Q3 1 ;
VARIABLE Q4 1 ; VARIABLE Q5 1 ;VARIABLE Q6 1 ;
VARIABLE Q7 1 ; VARIABLE Q8 1 ;VARIABLE Q9 1 ;
VARIABLE OBJ 1 ;

PROCEDURE TRIPLET A1 A2 A3 B1 B2 B3 C1 C2 C3;
  M5 .1 A1 A2 A3 0 0 .2 ;
  DL .05 ;
  M5 .1 -B1 -B2 -B3 0 0 .2 ;
  DL .05 ;
  M5 .1 C1 C2 C3 0 0 .2 ;
ENDPROCEDURE ;

OV 5 3 0 ;
RP 1 1 1 ;
SB .015 .15 0 .015 .15 0 0 .1 0 0 0 ;
PTY 4.0 ;
Q1 := .5 ; Q2 := .5 ; Q3 := .5 ;

```

APPENDIX A. COSY INFINITY CODE FOR SPECTROMETER IN SECTION 2.565

```
Q4 := .5 ; Q5 := .5 ; Q6 := .5 ;  
Q7 := .5 ; Q8 := .5 ; Q9 := .5 ;
```

```
FIT Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9;  
  UM ; CR ; ER 1 4 1 4 1 3 1 1 ;  
  BP ; DL .2 ; TRIPLET Q1 Q2 Q3 Q4 Q5 Q6 Q7 Q8 Q9; DL .2 ;  
  MS 2 30 .1 0 0 0 0 0 ; DL .2 ;  
  EP ; {PP -101 0 0 ; PP -102 0 90 ;}  
  OBJ := ABS(ME(1,2))+ABS(ME(1,22))+ABS(ME(1,222)) ;  
  WRITE 6 'STRENGTHS_Q1, Q2, Q3, OBJECTIVE_FUNCTION: ' Q1 Q2  
    Q3 Q4 Q5 Q6 Q7 Q8 Q9 OBJ ;  
  ENDFIT 1E-5 1000 1 OBJ ;{ PP -7 0 0 ; PP -7 0 90 ;}  
  PG -12 -12 ;  
  PM 6 ;  
  
ENDPROCEDURE ; RUN ; END ;
```

Appendix B

ROOT code for HTBL optimization

```
#include <iostream>
#include <fstream>
#include <cmath>
#include "TMath.h"
#include "TH2F.h"
#include "TF1.h"
#include "TGraph.h"
#include "TColor.h"
#include "TStyle.h"
#include "TCanvas.h"
#include "TFile.h"
#include "TRandom.h"
using namespace std;
#include "../rootOpticsMonteCarlo/s3mapap.C"

const Int_t TOTRAYS = 75000;
//const Int_t TOTMAPS = 1;
const Int_t TOTMAPS = 36;
const Int_t SWARMDIM = 48;
const Int_t KSLICETOT = 100;
const Double_t MAXDK = 0.025;
const Double_t MINDK = -0.025;
```

```

void PSOCosyWorkerMCHTBL_Clean() {
  Cs3mapap *s3MapAp;
  Char_t buffer[100];
  Char_t bufferPSO[100];
  Char_t mapfilename[100];
  Char_t histname[100];
  Double_t gBestObj;
  Double_t gBestPos[SWARMDIM];
  Double_t pBestObj;
  Double_t pBestPos[SWARMDIM];
  Double_t nowObj;
  Double_t nowPos[SWARMDIM];
  Double_t nowVel[SWARMDIM];
  Double_t allRaysIn[TOTRAYS][8], rayIn[8], outVec[8], xRMS,
    yRMS;
  Bool_t bCalc[5], trans[TOTRAYS];
  Double_t INERTIA = 0.9;
  Double_t ACCEL = 0.5;
  Double_t centA = 120;
  Double_t centQ = 49;
  Double_t xMeanFB1;
  Double_t xMeanFB3;
  Double_t xRMSFS0;
  Double_t yRMSFS0;
  Double_t rn;
  TH2F *fpYvX;
  TH1F *fpX;
  TF1* fc = new TF1("fc", "[2]*TMath::CauchyDist(x,_[0],_[1])",
    -5, 5);
  Double_t Fieldlimit[SWARMDIM]= {0.225, 0.225, 0.225, 0.225,
    0.225, 0.225, 0.225, 0.225, 0.225, 0.225, 0.225, 0.225,
    0.225, 0.225, 0.225, 0.225, 0.225, 0.225, 0.225, 0.225,
    0.225, 0.225, 0.225, 0.225,
    0.16875, 0.16875, 0.16875,
    0.16875, 0.16875,
    0.16875, 0.16875,
    0.16875, 0.16875,
    0.16875, 0.16875,

```

```

0.16875, 0.16875,
0.16875, 0.16875,
0.16875, 0.16875,
0.16875, 0.16875,
0.16875, 0.16875,
0.16875, 0.16875,
0.16875};

Double_t kSliceW = (MAXDK-MINDK)/KSLICETOT;
Int_t kNum;
Int_t transNum;
TH1F *FB1x_KSlice [KSLICETOT];
TH1F *FB3x_KSlice [KSLICETOT];

//variables for fission fragment inputs
Int_t fragA, fragZ;
Double_t fragM, fragPx, fragPy, fragPz, fragBrho, fragTKE;
Double_t sumTKE, centTKE;

Double_t fragBrhoCent = 8.0;

Bool_t actDim[SWARMDIM];
Double_t distA, distB, distK, waterBagR, distX, distY;

// Monte Carlo beam distribution size settings of Mg-40
distA = 0.00625; //half width in radians
distB = 0.0156; //half width in radians
distK = 0.0154; //half width in fraction of K
distX = 0.0006;
distY = 0.0017;

// Monte Carlo beam distribution size settings of Ni-84
/* distA = 0.01025; //half width in radians
distB = 0.03165; //half width in radians
distK = 0.016; //half width in fraction of K
distX = 0.0005;
distY = 0.00085; */

```



```

/*Int_t mapNum[TOTMAPS] = {60};
Int_t apType[TOTMAPS] = {1};
Double_t apxhw[TOTMAPS] = {0.1};
Double_t apyhw[TOTMAPS] = {0.1};*/

/*
//version with TOTMAPS = 8
Int_t mapNum[TOTMAPS] = {16,11,5,6,2,12,13,14};
Int_t apType[TOTMAPS] = {2,1,2,2,1,1,1,1};
Double_t apxhw[TOTMAPS] =
    {0.75,0.2,0.45,0.45,0.2,0.2,0.2,0.2};
Double_t apyhw[TOTMAPS] =
    {0.1,0.2,0.30,0.30,0.2,0.2,0.2,0.2};

//version with TOTMAPS = 3
Int_t mapNum[TOTMAPS] = {16,11,5};
Int_t apType[TOTMAPS] = {2,1,2};
Double_t apxhw[TOTMAPS] = {0.75,0.2,0.45};
Double_t apyhw[TOTMAPS] = {0.1,0.2,0.30};
*/

//Initialize the random number generator based on user
input
Int_t randSeed, PSOnum;
cout << "Enter the number of PSO" << endl;
cin >> PSOnum;
cout << "Enter the pseudo-random number generator seed" <<
endl;
cin >> randSeed;
TRandom3 *rand1 = new TRandom3(randSeed*SWARMDIM*TOTRAYS*
PSOnum);

//Setup the log file for this swarm particle
sprintf(buffer, "pso%d/p%3.3d.log", PSOnum, randSeed);

ofstream logFile(buffer, ios::out);
//Read in the current status of the global optimization run

```

```

sprintf(bufferPSO,"pso%d/gBest.dat",PSOnum);
ifstream gBestFile(bufferPSO,ios::in);
gBestFile >> gBestObj;
cout << "gBestObj_=_ " << gBestObj << endl;
/*
nowPos[6] = -0.0149303 ;
nowPos[5] = 0.00826769 ;
nowPos[4] = 0.0189009 ;
nowPos[3] = -1.45959e-05;
nowPos[2] = -0.0443817 ;
nowPos[1] = -0.029516 ;
nowPos[0] = 0.00814963 ;
*/

//Initialize the position of the current swarm particle AND
//read the global best position from the file

for(Int_t dim = 0; dim<SWARMDIM; dim++) {
  nowPos[dim] = rand1->Gaus(0,Fieldlimit [dim]/3);
  while (abs(nowPos [dim])>Fieldlimit [dim]) {
    nowPos [dim] = rand1->Gaus(0,Fieldlimit [dim]/3);
  }

  logfile << nowPos [dim] << " " ;
  nowVel [dim] = rand1->Gaus(0,Fieldlimit [dim]/4);
  //nowVel [dim] += rn/1.5;
  //nowVel [dim] = rand1->Gaus(0,0.05);
  pBestPos [dim] = nowPos [dim];
  pBestObj = 1e9;
  gBestFile >> gBestPos [dim];
}
logfile << endl;
gBestFile.close();
//BEGIN LOOP OVER SWARM STEPS FOR THIS PARTICLE
step = 0;

//Read input data from Remco, Fission product pairs from
LANL group
// ifstream rayFileIn("/home/ama018/optics/MA/hrs/

```

```

    InputDistributions/output.dat", ios::in);
//File Format:
//Mass number fragment 1
//Atomic Number 1
//Mass (from mass table in MeV/c^2) 1
//Momentum in X plane 1
//Momentum in Y Plane 1
//Momentum in Z plane 1
//Brho 1
//Total Kinetic energy 1
//Followed by the same for fragment 2.
//
//The central Brho is 5.25
//Int_t fragA, fragZ;
//Double_t fragM, fragPx, fragPy, fragPz, fragBrho, fragTKE
;
//Doublt_t fragBrhoCent = 5.25;

/*sumTKE = 0.;
for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {

    rayFileIn >> fragA
                >> fragZ
                >> fragM
                >> fragPx
                >> fragPy
                >> fragPz
                >> fragBrho
                >> fragTKE;
//cout << fragA << " " << fragZ << " " << fragM << endl;
allRaysIn[rayNum][1]=fragPx/fragPz; //a
allRaysIn[rayNum][3]=fragPy/fragPz; //b
//this should be the deltaK, but we will update these TKE
to delta_TKE later in a loop
allRaysIn[rayNum][4]=fragTKE; //delta k
sumTKE+=fragTKE;
allRaysIn[rayNum][6] = (fragA-centA)/centA; //fractional
delta -M
allRaysIn[rayNum][7] = (fragZ-centQ)/centQ; //fractional

```

```

        delta-Q
    }

    centTKE=sumTKE/TOTRAYS;
    cout << "Average TKE is " << centTKE << endl;

    //change the TKE values in the allRaysIn array to be
    delta_TKE from the central TKE
    for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {
        allRaysIn[rayNum][4] = (allRaysIn[rayNum][4]-centTKE)/
            centTKE;
    }*/

    //produce a set of random particle trajectories that can be
    used at all apertures
    for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {
        // Gaussian Randomization of x and y distribution
        allRaysIn[rayNum][0]=gRandom->Gaus(0,distX);
        allRaysIn[rayNum][2]=gRandom->Gaus(0,distY);
        //allRaysIn[rayNum][0]=0.;//safe assumption here that the
        size of the input beam doesn't matter
        //allRaysIn[rayNum][2]=0.;
        // and angle distributions
        allRaysIn[rayNum][1]=gRandom->Gaus(0,distA);
        allRaysIn[rayNum][3]=gRandom->Gaus(0,distB);
        // and energy distribution
        //allRaysIn[rayNum][4]=gRandom->Uniform(-distK,distK);
        allRaysIn[rayNum][4]=gRandom->Gaus(0,distK);
        //allRaysIn[rayNum][4]=0.0;
    }

    while(gBestObj>0.0 && step < 1) {

        for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {
            trans[rayNum]=true; //all rays start without having
            been stopped yet
        }
    }

```

```

step++;

//      INERTIA-=0.0001;
//      ACCEL+=0.0001;
//
//
//
//Write the new setting to be used by COSY to calculate
  the maps
sprintf(buffer,"pso%d/HTBLTune%3.3d.in",PSOnum,randSeed);
ofstream fieldSettingsOut(buffer,ios::out);
for(Int_t dim = 0; dim<SWARMDIM; dim++) {
  fieldSettingsOut << nowPos[dim] << endl;
}
fieldSettingsOut.close();

//
//
//Call COSY to calculate the transfer maps for the given
  system setting
// the COSY script will need to be edited to take input
  magnet field
// settings and to write maps with a swarm particle ID to
  keep maps
// from different swarm particles separate
//
//
//Call COSY to produce the new maps
sprintf(buffer,"!./runCosy91Multi.sh_cosy_simplified/
  HTBLHA.fox_%d_%d",PSOnum,randSeed);
//sprintf(buffer,"!./runCosy91Multi.sh_HRSdev.fox_7.40
  %d_5_y",randSeed);
gROOT->ProcessLine(buffer);
cout << "Cosy_finished" << endl;
for(int ii=0; ii<KSLICETOT; ii++) {
  sprintf(histname,"FB1x_kSl_%d",ii);

```

```

    //cout << histname << endl;
    FB1x_KSlice[ ii ] = new TH1F(histname , histname , 100,
        -0.05, 0.05);
}

for(int iii=0; iii<KSLICETOT; iii++) {
    sprintf(histname,"FB3x_kSl_%d",iii);
    //cout << histname << endl;
    FB3x_KSlice[ iii ] = new TH1F(histname , histname , 100,
        -0.05, 0.05);
}

fpYvX = new TH2F("fpYvX" , "fpYvX" , 100, -0.05, 0.05, 100,
    -0.05, 0.05);
fpX = new TH1F("fpX" , "fpX" , 100, -0.05, 0.05);

cout << "Created histograms" << endl;

//Start going through the maps one-by-one, eliminating
//stopped rays
//Int_t mapIdx = 0;
for(Int_t mapIdx=0; mapIdx<TOTMAPS; mapIdx++) {

    //
    //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
    //Read a map file to use for the Monte Carlo simulation
    //
    //////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

    sprintf(mapfilename , "pso%d/HTBLP%3.3d_A%3.3d.MAP" ,
        PSONum, randSeed , mapNum[ mapIdx ] );
    cout << "Reading map_" << mapfilename << endl;
    // s3MapAp = new Cs3mapap(mapfilename , sysOrder , apType ,
        apxhw , apyhw);
    s3MapAp = new Cs3mapap(mapfilename , calcOrder , apType [
        mapIdx ] , apxhw [ mapIdx ] , apyhw [ mapIdx ] );
}

```

```

//
//
//Calculate an objective function value for this
  particular solution
//based on a Monte Carlo simulation
//
//
//Sets which final state beam coordinates are
  calculated for each particle
//with "bCalc" array. Calculating fewer dimensions will
  save time.
// 0->x, 1->a, 2->y, 3->b, 4->t
for (Int_t iii=0; iii < 5; iii++) bCalc[iii]=true; //set
  all true to start
bCalc[1]=false; //do not calculate final horizontal
  angles
//bCalc[2]=false; //do not calculate final vertical
  position
bCalc[3]=false; //do not calculate final vertical
  angles
bCalc[4]=false; //do not calculate ToF deviations

xRMS = 0.;
yRMS = 0.;
fracTrans = 0;
//xMeanFB1 = 0.;
//xMeanFB3 = 0.;
transNum = 0;

for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {
  //only calculate this ray if it has not yet been
  stopped
  if(trans[rayNum]) {
    rayIn[0]=allRaysIn[rayNum][0];
    rayIn[2]=allRaysIn[rayNum][2];
  }
}

```

```

// and angle distributions
rayIn [1]= allRaysIn [rayNum] [ 1 ] ;
rayIn [3]= allRaysIn [rayNum] [ 3 ] ;
// and energy distribution
//rayIn [4]=gRandom->Uniform(-distK , distK ) ;
rayIn [4]= allRaysIn [rayNum] [ 4 ] ;
rayIn [6]= allRaysIn [rayNum] [ 6 ] ;
rayIn [7]= allRaysIn [rayNum] [ 7 ] ;

trans [rayNum] = s3MapAp->Trans ( rayIn [0] , rayIn [1] ,
    rayIn [2] , rayIn [3] , rayIn [4] , rayIn [6] , rayIn [7] ,
    bCalc , outVec ) ;

if ( rayIn [4] >= MINDK && rayIn [4] <= MAXDK ) {
    transNum += 1 ;

    if ( mapNum [mapIdx] == 23 ) {
        kNum = ( rayIn [4] - MINDK ) / kSliceW ;
        //cout << kNum << endl ;
        FB1x_KSlice [kNum] -> Fill ( outVec [0] ) ;
        //cout << outVec [0] << endl ;
    }
    if ( mapNum [mapIdx] == 37 ) {
        kNum = ( rayIn [4] - MINDK ) / kSliceW ;
        FB3x_KSlice [kNum] -> Fill ( outVec [0] ) ;
    }
}

if ( mapNum [mapIdx] == 60 ) {
    fpYvX -> Fill ( outVec [0] , outVec [2] ) ;
    fpX -> Fill ( outVec [0] ) ;
}
//cout << outVec [0] << endl ;
if ( trans [rayNum] ) { //repeated if statment?
    fracTrans += 1.0 ;
}

```

```

        xRMS += pow(outVec[0],2);

        //cout << rayNum << endl;
    }

}
//FB1x_KSlice[10]->Draw();

if(mapNum[mapIdx]==23) {
    xMeanFB1 = 0.;
    for(Int_t ii=0;ii<KSLICETOT;ii++) {
        //cout << "RMS=" << FB1x_KSlice[ii]->GetRMS(1) <<
            endl;
        xMeanFB1 += (FB1x_KSlice[ii]->GetRMS(1));
        //cout << "sum=" << xMeanFB1 << endl;
    }
}

}

if(mapNum[mapIdx]==37) {
    xMeanFB3 = 0.;
    for(Int_t ii=0;ii<KSLICETOT;ii++) {
        xMeanFB3 += (FB3x_KSlice[ii]->GetRMS(1));
    }
}

}

if(mapNum[mapIdx]==60) {
    xRMSFS0 = fpYvX->GetRMS(1);
    yRMSFS0 = fpYvX->GetRMS(2);
    /*fc->SetParameters(0, 1, 10000);
    fpX->Draw();
    fpX->Fit("fc","","",-0.005,0.005);*/
}

cout << mapIdx << "┘" << fracTrans << endl;

```



```

    pBestObj = nowObj;
    cout << "This is a new pBest!" << endl;
    for(Int_t dim = 0; dim<SWARMDIM; dim++) {
        pBestPos[dim] = nowPos[dim];
    }
}

//
//
//If this is a new global best position then update the
//global best file.
//
//
//Read in the current global best position and objective
//function value
sprintf(bufferPSO,"pso%d/gBest.dat",PSOnum);
ifstream gBestFile(bufferPSO,ios::in);
gBestFile >> gBestObj;
for(Int_t dim = 0; dim<SWARMDIM; dim++) {
    gBestFile >> gBestPos[dim];
}
gBestFile.close();

//Test
if(nowObj<gBestObj) {
    gBestObj = nowObj;

    cout << "New_gbest_written_by_particle_"
        << randSeed
        << " :_gBestObj_="
        << gBestObj
        << endl;
    sprintf(bufferPSO,"pso%d/gBest.dat.tmp",PSOnum);
    ofstream gBestFile(bufferPSO,ios::out);
    gBestFile << gBestObj << endl;
    for(Int_t dim = 0; dim<SWARMDIM; dim++) {

```

```

    gBestFile << nowPos[dim] << endl;
}
gBestFile.close();

sprintf (buffer, ".! _cp_pso%d/gBest.dat.tmp_pso%d/gBest.
dat",PSOnum,PSOnum);
gROOT->ProcessLine(buffer);
//sprintf(buffer, ".! cp_pso%d/HRS_P%3.3d_A%3.3d.MAP_pso
%d/gBest.MAP",PSOnum,randSeed,mapNum[mapIdx]);
sprintf(buffer, ".! _echo_%d->>_pso%d/gBest.log",step,
PSOnum);
gROOT->ProcessLine(buffer);
sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
gBestObj,PSOnum);
gROOT->ProcessLine(buffer);
//gROOT->ProcessLine(".! cp_pic001.ps_pic001.best.ps");

sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
fracTrans,PSOnum);
gROOT->ProcessLine(buffer);
sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
xRMSFS0,PSOnum);
gROOT->ProcessLine(buffer);
sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
yRMSFS0,PSOnum);
gROOT->ProcessLine(buffer);
sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
xMeanFB1,PSOnum);
gROOT->ProcessLine(buffer);
sprintf(buffer, ".! _echo_%12g->>_pso%d/gBest.log",
xMeanFB3,PSOnum);
gROOT->ProcessLine(buffer);

//Produce data files for transmission plots
sprintf(buffer, "pso%d/bestTransP%3.3d_S%3.3d.dat",
PSOnum,randSeed,step);

```

```

ofstream gBestTrans(buffer, ios::out);
gBestTrans << TOTRAYS << endl;
for (Int_t rayNum=0; rayNum<TOTRAYS; rayNum++) {
    //print out the incoming beam particles and whether
    //each was transmitted
    gBestTrans << trans[rayNum] << " "
        << allRaysIn[rayNum][0] << " "
        << allRaysIn[rayNum][1] << " "
        << allRaysIn[rayNum][2] << " "
        << allRaysIn[rayNum][3] << " "
        << allRaysIn[rayNum][4] << endl;
}
gBestTrans.close();
}

//
////////////////////////////////////////////////////////////////////

//Update the current position and velocity of this swarm
//particle
//
////////////////////////////////////////////////////////////////////

if (rand1->Uniform(0,1.0) < 1.1) {
    for(Int_t dim = 0; dim<SWARMDIM; dim++) {
        nowPos[dim] += nowVel[dim];
        //Add mutation every 5 iterations 2-3 of the 12
        //dimensions will jump to see the other options
        /*if(step>0 && step < 21 && remainder(step, 5.) == 0.0
        && rand1->Uniform(0,1.0) > 0.8){
            nowPos[dim] += rand1->Uniform(-0.02,0.02);
        }*/
        logFile << nowPos[dim] << " " ;
        nowVel[dim] = (INERTIA * nowVel[dim]
            + ACCEL * rand1->Uniform(0,1.0) * (
                pBestPos[dim] - nowPos[dim])
            + ACCEL * rand1->Uniform(0,1.0) * (
                gBestPos[dim] - nowPos[dim]));
    }
}

```

```
    }

} else { //what does this do?
    for(Int_t dim = 0; dim < SWARMDIM; dim++) {
        nowPos[dim] = rand1->Gaus(0,0.005);
        logFile << nowPos[dim] << " " ;
        nowVel[dim] = rand1->Gaus(0,0.005);
        pBestPos[dim] = nowPos[dim];
        pBestObj = 1e9;
    }
}
logFile << endl;

} // while (gBestObj > 0.0)

}
```

References

- [1] A.M. Amthor and Z.M. Schillaci et al. Experimental test of an online ion-optics optimizer. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 895:90–99, 2018.
- [2] Rene Brun and Fons Rademakers. Root - an object oriented data analysis framework. *Nuclear Instruments and Methods in Physics Research A*, 389:81–86, 1997.
- [3] FRIB Science Community. Frib400 the scientific case for the 400 mev/u energy upgrade of frib. Technical report, Michigan State University, Facility for Rare Isotope Beams, June 2019.
- [4] Britannica Encyclopædia. Development of atomic theory. <https://www.britannica.com/science/atom/Development-of-atomic-theory>, 2023.
- [5] H. L. Crawford et al. Crossing $n = 28$ toward the neutron drip line: First measurement of half-lives at frib. *Phys. Rev. Lett.*, 129:212501, Nov 2022.
- [6] David J. Griffiths. *Introduction to Electrodynamics*. Cambridge University Press, Reed College, Oregon, 2017.
- [7] Robert L. Harrison. Introduction to monte carlo simulation. *AIP Conf Proc*, 1204:17–21, 2010.
- [8] M. Hausmann, A.M. Aaron, and A.M. Amthor et al. Design of the advanced rare isotope separator aris at frib. *Nuclear Instruments and Methods in Physics Research Section B: Beam Interactions with Materials and Atoms*, 317:349–353, 2013.
- [9] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 - International Conference on Neural Networks*, volume 4, pages 1942–1948 vol.4, 1995.

- [10] Kenneth S. Krane. *Introductory Nuclear Physics*. John Wiley Sons, Oregon State University, 1955.
- [11] Kyoko Makino and Martin Berz. Cosy infinity version 9. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 558(1):346–350, 2006.
- [12] S. Noji, R.G.T. Zegers, G.P.A. Berg, A.M. Amthor, T. Baumann, D. Bazin, E.E. Burkhardt, M. Cortesi, J.C. DeKamp, M. Hausmann, M. Portillo, D.H. Potterveld, B.M. Sherrill, A. Stolz, O.B. Tarasov, and R.C. York. Design of the high rigidity spectrometer at frib. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 1045:167548, 2023.
- [13] Department of Energy Office of Science. DOE Explains...Isotopes. <https://www.energy.gov/science/doe-explainsisotopes>, 2022.
- [14] Jerald Pinson. The power of attraction: magnets in particle accelerators. *Fermi National Accelerator Laboratory*, 2020.
- [15] David Prutchi and Shanni R. Prutchi. *Exploring quantum physics through hands-on projects*. A John Wiley and Sons, Hoboken, N.J, 2012.
- [16] Matteo Rini. Rare isotopes for the choosing. *American Physical Society*, May 2022. Accessed: 2022-09-15.
- [17] Dh Rouvray. Dalton, john - the worlds first stereochemist. *ENDEAVOUR*, 19(2):52–57, 1995.
- [18] D. Bazin et al. T. Baumann, A. Amthor. Discovery of 40mg and 42al suggests neutron drip-line slant towards heavier isotopes. *Nature*, 449:1022–1024, 2007.
- [19] O.B. Tarasov and D. Bazin. Lise++: Radioactive beam production with in-flight separators. *Nuclear Instruments and Methods in Physics Research B*, 266:4657–4664, 2008.
- [20] Maria Temming. A weird stellar explosion may have caused the brightest supernova yet seen, 2020.
- [21] Ioan Trelea. Trelea, i.c.: The particle swarm optimization algorithm: convergence analysis and parameter selection. *information processing letters* 85, 317-325. *Information Processing Letters*, 85:317–325, 03 2003.

- [22] E. Wilson and B. J. Holzer. *Beam Dynamics*, pages 15–50. Springer International Publishing, Cham, 2020.
- [23] Hermann Wollnik. *Optics of Charged Particles*. Academic Press, Orlando, 1987.
- [24] R. G. T. Zegers. Personal communication, 2021.