ABSTRACT

DIFFERENTIAL ALGEBRAIC METHODS FOR SPACE CHARGE MODELING AND APPLICATIONS TO THE UNIVERSITY OF MARYLAND ELECTRON RING

Edward W. Nissen, Ph.D.
Department of Physics
Northern Illinois University, 2011
Dr. Bela Erdelyi, Director

The future of particle accelerators is moving towards the intensity frontier; the need to place more particles into a smaller space is a common requirement of nearly all applications of particle accelerators. Putting large numbers of particles in a small space means that the mutual repulsion of these charged particles becomes a significant factor; this effect is called space charge. In this work we develop a series of differential algebra-based methods to simulate the effects of space charge in particle accelerators. These methods were used to model the University of Maryland Electron Ring, a small 3.8 meter diameter 10 KeV electron storage ring designed to observe the effects of space charge in a safe, cost-effective manner. The methods developed here are designed to not only simulate the effects of space charge on the motions of the test particles in the system but to add their effects to the transfer map of the system. Once they have been added, useful information about the beam, such as tune shifts and chromaticities, can be extracted directly from the map. In order to make the simulation self-consistent, the statistical moments of the distribution are used to create a self-consistent Taylor series representing the distribution function, which is

combined with pre-stored integrals solved using a Duffy transformation to find the potential. This method can not only find the map of the system, but also advance the particles, under most conditions. For conditions where it cannot be used to accurately advance the particles a differential algebra-based fast multipole method is implemented. By using differential algebras to create local expansions, noticeable time savings are found.

NORTHERN ILLINOIS UNIVERSITY
DE KALB, ILLINOIS

AUGUST 2011

DIFFERENTIAL ALGEBRAIC METHODS FOR SPACE CHARGE MODELING AND APPLICATIONS TO THE UNIVERSITY OF MARYLAND ELECTRON RING

BY

EDWARD W. NISSEN

A DISSERTATION SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE
DOCTOR OF PHILOSOPHY

DEPARTMENT OF PHYSICS

Dissertation Director:
Dr. Bela Erdelyi

ACKNOWLEDGEMENTS

There are a number of people and organizations who have helped me with my work throughout the years whom I would like to thank. I would like to thank Professor Court Bohn who brought me into this group and this field. Though he is no longer with us, his influence will carry on indefinitely. I would also like to thank Dr. Balša Terzić, who stepped in and helped me finish both the research and my master's thesis. I would also like to thank my current advisor Dr. Bela Erdelyi for the support and guidance he has shown me throughout this process.

I would not have been able to complete this research without the help of the dedicated people at the University of Maryland Electron Ring. I would especially like to thank Santiago Bernal, Dave Sutter, Tim Koeth, Rami Kishek, and Brian Beaudoin, as well as Karen Fiuza and Chao Wu. Many of whom took significant time to help me take my experimental data.

I would also like to thank both the Department of Energy as well as the Directed Energy Professional Society for their financial support throughout my studies, as well as Northern Illinois University for giving me a Dissertation Completion Award to finish up my degree. I would also like to thank the organizing committees of PAC '07, IPAC '10, and PAC '11 as well as Dr. Albright and the organizing committee for HB2010 for their generous travel assistance which allowed me to showcase my research.

DEDICATION

For my family

TABLE OF CONTENTS

Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

Ever since J. J. Thompson proved that so-called "Cathode Rays" were in fact charged particles [1], particle accelerators of various sizes and energies have allowed scientists to study a variety of phenomena, and engineers to create a number of useful processes and technologies [2]. While large-scale experiments such as the Large Hadron Collider and Fermilab's Tevatron are what most people think of when they think of particle accelerators, there are also technologies which use the particle beams in accelerators to create intense X-rays and laser beams [3], cancer treatments [4], accurate views of molecular structures [5], and commercial semiconductors with well-defined and placed impurities [6].

When dealing with large-scale experimental accelerators two of the main quantities of interest in the beam are energy and luminosity, essentially how fast the particles are moving and how close together they are. The increase in energy increases the amount of energy available to create the exotic particles that these machines are trying to detect, while an increase in luminosity increases the likelihood that one of the exotic particles an experimenter is trying to detect will be created [7]. In particle colliders, the luminosity $\mathcal{L}$ is given as

$$\mathcal{L} = f\frac{n_1 n_2}{4\pi\sigma_x\sigma_y}, \tag{1.1}$$

where f is the collision frequency, n_1 and n_2 are the number of particles in each colliding bunch, and σ_x and σ_y are the transverse standard deviations of the size of the beam.

One of the other major uses of large-scale particle accelerators is the creation of X-rays. These range from a basic low-energy source caused by bremsstrahlung that is used in a doctor's office, to high-intensity light sources such as the Advanced Photon Source at Argonne National Lab. These light sources use synchrotron radiation to create high-power X-rays with a narrow frequency spectrum. They work by sending the particles, usually electrons, through a series of alternating dipoles that cause the electrons to accelerate, and thus emit synchrotron radiation. The actual frequency spectrum is difficult to determine, and is machine specific, but it is sufficient to know that the frequency is inversely proportional to the bending radius of the magnets, while the power scales between n and n^2 in particle number. Since the frequency is inversely proportional to the bending radius of the beam, which depends on both the field strength of the magnet and the energy of the particle, keeping the spread in the energy of the electrons at a minimum will narrow the frequency spectrum, ensuring that more of the photons are at the frequency desired.

In both of the cases mentioned above the ultimate goal is to have the largest possible number of particles taking up the smallest possible volume. For single particles going through the machine one at a time, magnetic and electrostatic elements can be used like lenses to focus the particles down into a very small area. The mathematics of this can be performed in a number of ways, but are well understood and can be easily modeled both analytically and computationally [8, 9]. Once we want to determine how the many particles moving through the beam interact with each other, there are significant bottlenecks. Since these particles are all the same charge, there is a mutual repulsion between them. This leads to two effects, space charge and intra-beam scattering. Intra-beam scattering concerns the individual particles interacting with each other, while space charge deals with the overall effects of this mutual repulsion. Space charge is so named because it assumes that the individual charged particles are so spread out and numerous that they can be treated as a smooth distribution.

There are currently several ways to model the motion of particles as they interact with each other while moving through a system. The simplest method is to use point-to-point Coulomb interactions to model the system as it evolves. This is as accurate as a model can get, and it can theoretically model both space charge and intra-beam scattering. Unfortunately this method's strengths are also its weaknesses; since it has to determine the effect on each particle caused by every other particle, the number of operations is of the order $n(n-1)$. Since beam bunches frequently have n between 10^9 and 10^{13} particles it is frequently referred to as $\mathcal{O}(n^2)$. Furthermore, in order to model the effects of intra-beam scattering, the step size over which each evaluation must be made becomes very small, since the particles must be able to move close to one another so that they can bounce off of each other, and too large a stepsize will allow the particles to pass through each other without scattering.

The point-to-point method's main advantage is that it is as accurate as is theoretically possible. The main purpose of these models is to simulate many possible machine configurations on a computer, in order to determine the best design for a desired purpose. Since a computer is being used there is now a hard limit on the accuracy of any quantity being calculated, as computers only store numbers up to 16 decimal places. Since there is a limit to the accuracy of a simulation, it is no longer necessary to expend vast resources to simulate a beam to infinite accuracy, and time saving measures can be employed. Another time saving measure that is common to all methods is to replace the particles in question with macro particles. In essence we are defining each of the test particles in the simulation as some number of the particles in question. A simulation of a proton ring might have each macroparticle represent 10^5 protons. This will speed processing times, but will make any intra-beam scattering measurements suspect since if one macroparticle is lost it is difficult to say that 10^5 have been lost.

Space charge modeling methods all make a great use of the fact that the user is willing to accept some degree of inaccuracy in exchange for a fast answer [10]. The methods that currently exist for space charge modeling fall into three categories. The first is the direct point-to-point method mentioned earlier. The second is a particle mesh method which distributes the charge in the system onto set grid points and then uses some sort of expansion to model the field. This is usually a fast Fourier transformation, but other methods exist, with the number of operations $\mathcal{O}(N+M\log(M))$, where M is the number of mesh points. The third is a hierarchical tree method where the set of particles is divided into smaller and smaller subregions, such that distant particles can be approximated by simpler expansions and close particles using more direct methods. This scales as $n \log n$. Furthermore there are a number of ways that these methods can be combined with each other [11].

Modeling these effects is important for the design and optimization of an accelerator. An accurate simulation regarding quantities like the dynamic aperture or the overall beam size can give a designer a good idea of just how large the beam pipe or the magnet poles need to be. An accurate answer will show whether a given design can be affordably built and operated without spending too much time or money on the design. No matter how accurate the Poisson solver is, there is one final method that has not been mentioned, and that is to use what can best be described as an analogue computer [12]. An analogue beam physics computer is simply an experimental setup; this can be either used to benchmark code, or to excite phenomena that the user wishes to study.

In order to experimentally measure the effects of space charge on a particle beam in a cost-effective manner, the University of Maryland Electron Ring (UMER) was built by the Institute for Research in Applied Physics at the University of Maryland. This is a small electron storage ring that operates at low energy, which allows the electrons to experience the effects of space charge that would normally only be felt in a heavy ion accelerator [13, 14, 15, 16]. While this design allows for a compact and inexpensive experimental setup, it presents issues of its own. The earth's magnetic field must be taken into account when tracking particles through the machine, while its novel geometry poses modeling issues [17]. A number of studies have been performed by others at UMER on the ring, including waves propagating longitudinally along the beam, and phase space tomography on the transverse parts of the beam [18, 19].

This research required the creation of a number of new tools to model the unique characteristics of UMER. Due to the unique injection geometry and external fields, an initial investigation was made of the single particle dynamics of UMER [20]. Once that had been completed, a study of the multiple particle effects was performed. The ring was modeled using the code COSY Infinity 9.0 which uses differential algebras to determine exact numerical derivatives to arbitrary order. The method developed for this study uses the statistical moments of the particles in question to determine a Taylor series which represents that distribution. This Taylor series is then composed with a series of pre-stored integrals to find the potential of the distribution, as well as its electric fields. This method works in a manner opposite to most self-consistent space charge codes. In other codes, for each particle, we calculate the effects of space charge on it caused by every other particle. In this method we calculate for each particle its effect on space charge as a whole, producing a space charge algorithm that can be massively parallelized, which scales linearly with particle number.

The method used to add space charge to the map, from now on referred to as the moment method, will run into problems with distributions that are not easily modeled using a Taylor series [21, 22]. Because of this, for distributions which are difficult to model, we have implemented a differential algebra-based version of the fast multipole method. This method divides the region of interest into a series of boxes. For boxes far away from the point in question, the multipole expansions are combined to form a local Taylor series of the far field. This series is combined with the point-to-point Coulomb interactions of the nearby particles to determine the potential and electric fields of the point in question. In this work the expansion of the multipole expansions to local ones is performed using differential algebraic techniques which allow for an increase in speed.

First Chapter 2 examines the mathematical background of these methods. Then Chapter 3 develops the mathematical tools that we use, while Chapter 4 examines how these codes match up to other methods. Chapter 5 looks at some applications of the methods, and Chapter 6 examines some of the experiments performed on UMER.

CHAPTER 2

BACKGROUND INFORMATION

The tools and experiments used in this research require a number of mathematical principles that are not widely known outside or even inside the accelerator physics community. The information presented in this chapter is the general background information that is used in all of the subsequent chapters. The research performed involves the use of objects known as transfer maps which are explained in Section 2.1. These maps are created using the code COSY Infinity 9.0. COSY Infinity creates these maps using mathematical objects known as differential algebras which are explained in Section 2.2. COSY Infinity also analyzes these maps using methods known as normal form methods, which are shown in Section 2.3. COSY Infinity itself is examined in Section 2.4. Finally the method used to add external effects to these maps, known as Strang splitting, is discussed in Section 2.5. This will be followed in Section 2.6 by an overview of the University of Maryland Electron Ring which was used to perform a number of experiments as part of this research.

2.1 Transfer Maps

A transfer map is a method of moving a particle from its initial position to its final position in a system without having to follow the particle all the way through [23]. In the linear case, the map takes the form of a simple matrix equation, and for a one-dimensional linear map the equation is

$$\begin{pmatrix} x_f \\ p_{xf} \end{pmatrix} = \begin{pmatrix} (x|x) & (x|a) \\ (a|x) & (a|a) \end{pmatrix} \begin{pmatrix} x_i \\ p_{xi} \end{pmatrix}. \tag{2.1}$$

A good first-order approximation is to think of the elements in a beam as a collection of thin lenses with a drift, a focusing lens, and a defocusing lens, given as

$$\mathcal{M}_D = \begin{pmatrix} 1 & \ell \\ 0 & 1 \end{pmatrix}, \quad \mathcal{M}_f = \begin{pmatrix} 1 & 0 \\ -K & 1 \end{pmatrix}, \quad \mathcal{M}_{df} = \begin{pmatrix} 1 & 0 \\ K & 1 \end{pmatrix}. \tag{2.2}$$

If we wanted to determine how a series of lenses and drifts work together we would combine their maps together. If we wanted to start halfway between a lens, pass through a drift, pass through another lens, and then a drift, and finally another half lens, we would have

$$\mathcal{M} = \mathcal{M}_{f_3}\mathcal{M}_{D_2}\mathcal{M}_{f_2}\mathcal{M}_{D_1}\mathcal{M}_{f_1} \tag{2.3}$$

$$\mathcal{M} = \begin{pmatrix} 1 & 0 \\ -K_3 & 1 \end{pmatrix} \begin{pmatrix} 1 & \ell_2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -K_2 & 1 \end{pmatrix} \begin{pmatrix} 1 & \ell_1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ -K_1 & 1 \end{pmatrix} \tag{2.4}$$

By assuming that K_1=K_3, which each model half of a lens, we get the map for an object known as a FODO lens, which stands for FOcusing De-fOcusing. Most accelerators operate as a series of FODO sections added together to form what is often referred to as a lattice.

In a linear machine these focusing lenses would be made of quadrupole magnets. These do allow the user to have a focusing section in one plane, with the drawback that it will be defocusing in the perpendicular plane; if it is focusing for the horizontal it will be defocusing in the vertical. This would mean that in the example above, K_2 would have an opposite sign to K_1 and K_3.

In order to keep the beam from expanding out to infinity, it is necessary to adjust the focusing strengths with respect to the drift lengths so that the beam stays within an understood boundary. These boundaries are determined by a quantity known as the Twiss parameters α, β, and γ [24]. These three quantities define an ellipse in phase space that represents the trajectory of a particle as it moves through the lattice. β determines the outer boundary of the beam in physical space, γ is the largest angle or momentum, and α determines the amount of coupling between the angle and the position. With the physical size of the beam determined by the emittance, this relation is shown in Figure 2.1.

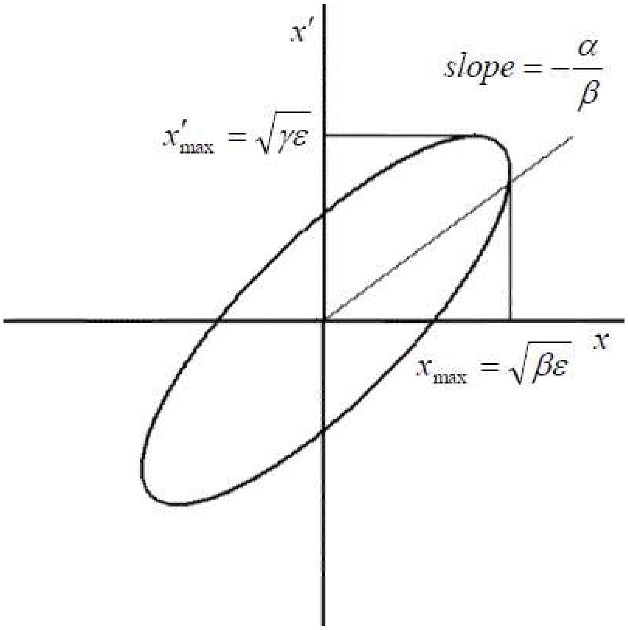


Figure 2.1: This is an example of the way the Twiss parameters describe an orbit in phase space.

In COSY Infinity, nonlinear maps can also be calculated. Instead of linear matrix elements, each final state is represented by a Taylor model with the initial state as inputs. This expands the equation from matrix multiplication to a composition of the Taylor models with the initial state

$$\begin{pmatrix} x_f \\ p_{xf} \\ y_f \\ p_{yf} \end{pmatrix} = \begin{pmatrix} \sum\limits_i \sum\limits_j \sum\limits_k \sum\limits_\ell ({}_xC_{ijk\ell}) x^i p_x^j y^k p_y^\ell \\ \sum\limits_i \sum\limits_j \sum\limits_k \sum\limits_\ell ({}_{p_x}C_{ijk\ell}) x^i p_x^j y^k p_y^\ell \\ \sum\limits_i \sum\limits_j \sum\limits_k \sum\limits_\ell ({}_yC_{ijk\ell}) x^i p_x^j y^k p_y^\ell \\ \sum\limits_i \sum\limits_j \sum\limits_k \sum\limits_\ell ({}_{p_y}C_{ijk\ell}) x^i p_x^j y^k p_y^\ell \end{pmatrix} \circ \begin{pmatrix} x_i \\ p_{xi} \\ y_i \\ p_{yi} \end{pmatrix}. \tag{2.5}$$

These nonlinear maps allow us to model many different nonlinear beam elements as well as the nonlinear aspects of space charge [25].

2.2 Differential Algebras

Differential algebraic (DA) methods are ways to determine exact numerical derivatives on a computer without truncation errors. This is a large field of study, so we will focus on using them to create Taylor models of flows of differential equations. Some examples are shown to allow the reader to better understand how useful these can be. The method works by creating objects known as DA vectors which have their basic mathematical relations redefined [25].

In the simplest case, a first-order, one-variable structure ${}_1D_1$, with its elements being an ordered set of two real numbers (q_0, q_1), is defined by the following relations:

$$(q_0, q_1) + (r_0, r_1) = (q_0 + r_0, q_1 + r_1), \tag{2.6}$$

$$t(q_0, q_1) = (tq_0, tq_1), \tag{2.7}$$

$$(q_0, q_1) \cdot (r_0, r_1) = (q_0 r_0, q_0 r_1 + q_1 r_0), \tag{2.8}$$

$$(q_0, q_1)^n = (q_0^n, n q_0^{n-1} q_1), \tag{2.9}$$

where t and n are scalars.

2.2.1 Computation of Derivatives

Assuming we are using the ${}_1D_1$ structure; the DA vector x is shown as (x_0, x_1). In this example we have the function f and its derivative,

$$f(x) = x^3 + 3x - \frac{2}{x+1}, \tag{2.10}$$

$$f'(x) = 3x^2 + 3 + \frac{2}{(x+1)^2}, \tag{2.11}$$

which at $x = 3$ gives

$$f(3) = 35\frac{1}{2}, \tag{2.12}$$

$$f'(3) = 30\frac{1}{8}. \tag{2.13}$$

If instead we evaluate the function replacing x with $(3, 1)$, we will get

$$f((3,1)) = ((3,1))^3 + 3(3,1) - \frac{2}{(3,1)+1}, \tag{2.14}$$

$$f((3,1)) = (27, 3(3)^2 1) + (9,3) - \frac{2}{(4,1)}, \tag{2.15}$$

$$f((3,1)) = (27,27) + (9,3) - 2(\frac{1}{4}, \frac{-1}{16}), \tag{2.16}$$

$$f((3,1)) = (35\frac{1}{2}, 30\frac{1}{8}). \tag{2.17}$$

The first entry (or the constant part) in the vector is the value the function takes at that particular point, and the second entry is the derivative at the same point. For higher orders and larger variable numbers the vectors become longer and more complicated, but conceptually the same procedure is followed. The vector eventually takes the form of essentially a high-order multivariate Taylor series of the function f.

2.2.2 Integration

An antiderivative can be quickly determined through simple polynomial integration of the DA vector. Assuming a vector q,

$$q = (q_0, q_1, q_2, q_3, ...), \tag{2.18}$$

its integral Q can be determined simply through a coefficient change,

$$Q = (c, q_0, \frac{q_1}{2}, \frac{q_2}{3}, \frac{q_3}{4}, ...), \tag{2.19}$$

where c is an arbitrary constant. Furthermore, since DA vectors work by redefining mathematical relations, other basic programs like a Runge-Kutta integrator will carry the DA vectors through; this would allow in our case for a Taylor series of the potential for an entire region to be calculated using only an integration of a reference particle.

Another way that this can be used is in the case of a function such as $\sin(x)$. It can be calculated as $\sin(x + dx)$ in DA. On the other hand, $\sin(x + dx)$ can also be rewritten as $\sin(x) + \sin'(x)dx$, or $\sin(x) + \cos(x)dx$.

2.2.3 DA-ODE Integration

One method that has been a tremendous speed boost in this work is the use of DA methods to solve ordinary differential equations (ODEs). This method uses DA integration to solve a system of first-order ODEs. The method works in cases with no explicit s dependence, since any first-order ODE of this type can be written as

$$\frac{d\vec{z}}{ds} = \vec{F}(\vec{z}), \tag{2.20}$$

$$\vec{z}\,(s_i) = \vec{z}_i. \tag{2.21}$$

This equation is then recast as an integral equation,

$$\vec{z}\,(s_f) = \vec{z}_i + \int_{s_i}^{s_f} \vec{F}(\vec{z})ds, \tag{2.22}$$

and solved using DA-integration as shown above in 2.2.2.

2.3 Normal Form Methods

Normal forms are just a coordinate transformation from the particle optical coordinates $\vec{z} = (\vec{q}, \vec{p})$ to action angle coordinates $\vec{j}$. A representation of this process is shown in Figure 2.2.

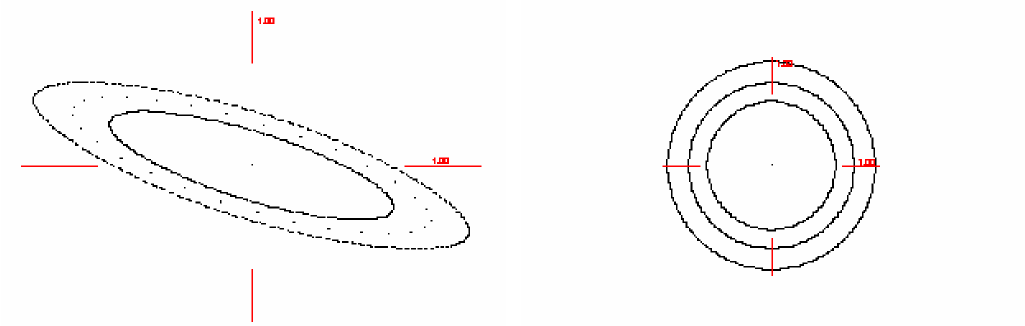


Figure 2.2: On the left is a picture of a set of phase space orbits; on the right is those same orbits in action angle coordinates.

This is done with a transformation $\mathcal{A}$, such that $\vec{j} = \mathcal{A}(\vec{z})$. When these action angle coordinates undergo a transformation $\mathcal{N}$ they move in circles with radius $|\vec{j}|$. The normal form $\mathcal{N}$ is related to the original map $\mathcal{M}$ by the relation $\mathcal{N} = \mathcal{A} \circ \mathcal{M} \circ \mathcal{A}^{-1}$. This transformation generates a map of the form:

$$\mathcal{N} = \begin{pmatrix} \mathcal{N}_1 & & 0 \\ & \mathcal{N}_2 & \\ 0 & & \mathcal{N}_3 \end{pmatrix}, \tag{2.23}$$

where

$$\mathcal{N}_m = \begin{pmatrix} \cos(\theta_m(\vec{j})) & \sin(\theta_m(\vec{j})) \\ -\sin(\theta_m(\vec{j})) & \cos(\theta_m(\vec{j})) \end{pmatrix}. \tag{2.24}$$

This makes θ of the form $\theta(\vec{j}) = \theta_0 + aj_1 + bj_2 + cj_1j_2 + dj_1^2 + \ldots$ where θ_0 is the tune and $a, b, c, d, \ldots$ are the amplitude-dependent tune shifts. If energy is declared a parameter δ, i.e., $\theta_0 \rightarrow \theta_0(\delta)$, then there will be an expansion $\theta_1\delta + \theta_2\delta^2 + \ldots$ where θ_1 is the chromaticity. The normalizing map $\mathcal{A}$ gives the resonance driving terms, and can be used to obtain the matching conditions [25].

In the linear case, this would mean that the previous map would be transformed by

$$\mathcal{M} = \begin{pmatrix} \sqrt{\beta} & 0 \\ -\frac{\alpha}{\sqrt{\beta}} & \frac{1}{\sqrt{\beta}} \end{pmatrix} \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{\beta}} & 0 \\ \frac{\alpha}{\sqrt{\beta}} & \sqrt{\beta} \end{pmatrix}, \tag{2.25}$$

where α and β are two of the Twiss parameters.

2.4 COSY Inifinity 9.0

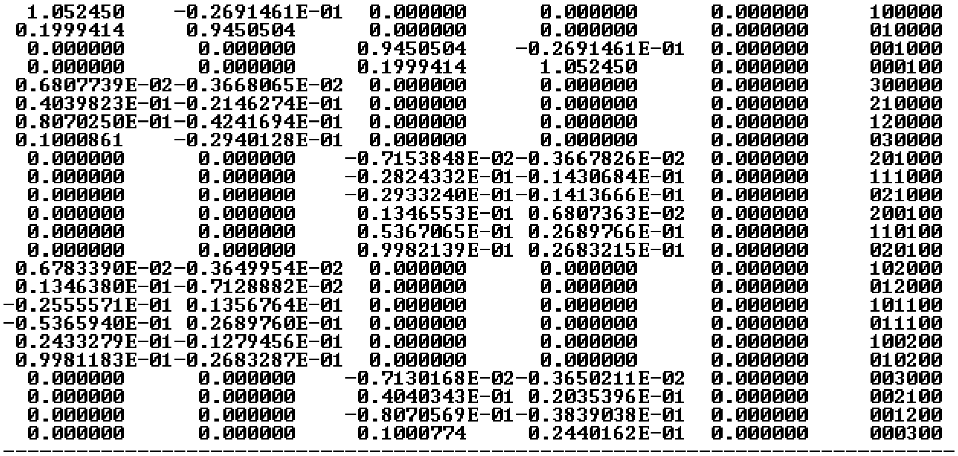
This research used the accelerator physics code COSY Infinity 9.0. This code uses both differential algebras and normal form methods to model and analyze charged particle beams. The software uses differential algebras to create high-order Taylor maps of the given elements and to compose them with one another to determine a beam line [26]. For example, if we wanted to produce the map of a simple drift-quadrupole-drift system, we would write;

```
INCLUDE 'COSY';{This includes the COSY beam libraries.}
Procedure run; {This sets up the list of tasks we wish to perform in
 our simulation, it is the equivalent of main in a c++ program.}
OV 3 2 0; {This sets the order and number of variables, in this case
third order with two space dimensions (four phase space dimensions)
and no extra variables.}
RPE .1; UM;{ This sets the particle type in this case electrons and
 the particle energy, in this case 100 KeV.}
DL .04;{ This creates a drift with a length of 4cm.}
MQ .02 .05*.02 .02;{ This creates a Quadrupole with a length of
2cm, a pole tip flux of .0006 tesla meters and an aperture of 2cm.}
DL .08;
MQ .02 -.05*.02 .02;
DL .04;
PM 6;{ This will print the map to the screen, other numbers will
 send it to different types of file.}
ENDPROCEDURE;
```

```
RUN; End;{This executes the tasks and ends the program.}
```

The output to the screen would be of the kind shown in Figure 2.3.

If we change the third and fourth lines to read:



```
  1.052450    -0.2691461E-01  0.000000      0.000000      0.000000        100000
 0.1999414      0.9450504     0.000000      0.000000      0.000000        010000
  0.000000      0.000000      0.9450504   -0.2691461E-01  0.000000        001000
  0.000000      0.000000      0.1999414     1.052450      0.000000        000100
 0.6807739E-02-0.3668065E-02  0.000000      0.000000      0.000000        300000
 0.4039823E-01-0.2146274E-01  0.000000      0.000000      0.000000        210000
 0.8070250E-01-0.4241694E-01  0.000000      0.000000      0.000000        120000
 0.1000861    -0.2940128E-01  0.000000      0.000000      0.000000        030000
  0.000000      0.000000    -0.7153848E-02-0.3667826E-02  0.000000        201000
  0.000000      0.000000    -0.2824332E-01-0.1430684E-01  0.000000        111000
  0.000000      0.000000    -0.2933240E-01-0.1413666E-01  0.000000        021000
  0.000000      0.000000     0.1346553E-01 0.6807363E-02  0.000000        200100
  0.000000      0.000000     0.5367065E-01 0.2689766E-01  0.000000        110100
  0.000000      0.000000     0.9982139E-01 0.2683215E-01  0.000000        020100
 0.6783390E-02-0.3649954E-02  0.000000      0.000000      0.000000        102000
 0.1346380E-01-0.7128882E-02  0.000000      0.000000      0.000000        012000
-0.2555571E-01 0.1356764E-01  0.000000      0.000000      0.000000        101100
-0.5365940E-01 0.2689760E-01  0.000000      0.000000      0.000000        011100
 0.2433279E-01-0.1279456E-01  0.000000      0.000000      0.000000        100200
 0.9981183E-01-0.2683287E-01  0.000000      0.000000      0.000000        010200
  0.000000      0.000000    -0.7130168E-02-0.3650211E-02  0.000000        003000
  0.000000      0.000000     0.4040343E-01 0.2035396E-01  0.000000        002100
  0.000000      0.000000    -0.8070569E-01-0.3839038E-01  0.000000        001200
  0.000000      0.000000     0.1000774     0.2440162E-01  0.000000        000300
--------------------------------------------------------------------------------
```

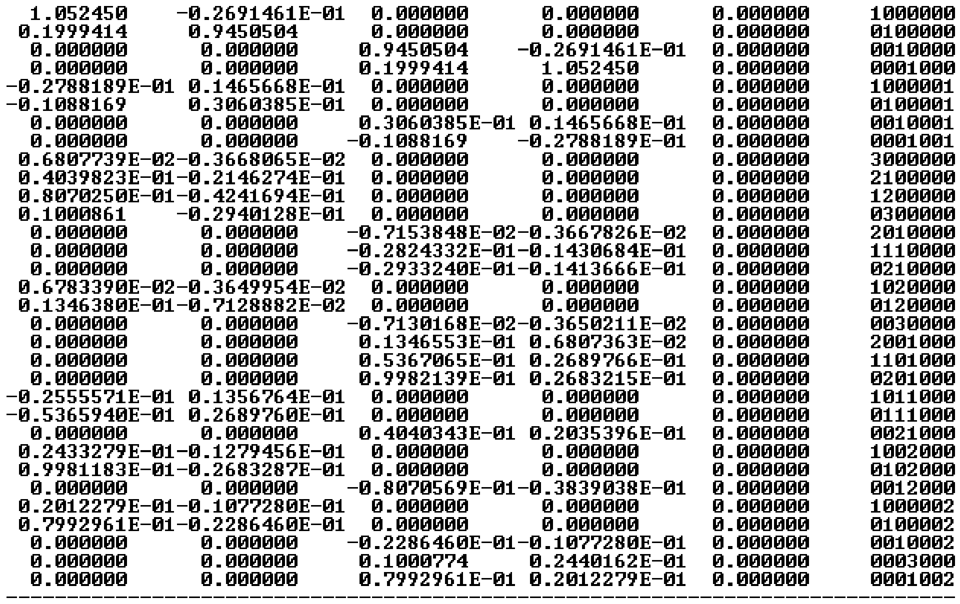
Figure 2.3: This is an example of a map generated using COSY Infinity. The map represents the system in the example code to third order.

```
OV 3 2 1;

RPE .1*para(1); UM;
```

then we will have declared the energy of the particles to be a variable. This gives a map like the one seen in Figure 2.4. This map shows how the various dimensions change with respect to the energy, which has an additional column added to it.

If we now have the more complete code:

```
INCLUDE 'COSY';

Procedure run;

VARIABLE MU 100 3;{ This declares an array named MU with three

entries with 100 double precision memory blocks allocated to each.}

OV 3 2 1;
```

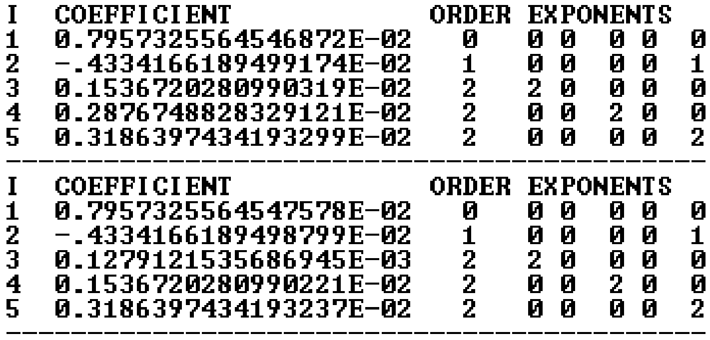


```
  1.052450    -0.2691461E-01  0.000000      0.000000      0.000000     1000000
 0.1999414     0.9450504      0.000000      0.000000      0.000000     0100000
  0.000000      0.000000     0.9450504    -0.2691461E-01  0.000000     0010000
  0.000000      0.000000     0.1999414      1.052450      0.000000     0001000
-0.2788189E-01 0.1465668E-01  0.000000      0.000000      0.000000     1000001
-0.1088169     0.3060385E-01  0.000000      0.000000      0.000000     0100001
  0.000000      0.000000     0.3060385E-01 0.1465668E-01  0.000000     0010001
  0.000000      0.000000    -0.1088169    -0.2788189E-01  0.000000     0001001
 0.6807739E-02-0.3668065E-02  0.000000      0.000000      0.000000     3000000
 0.4039823E-01-0.2146274E-01  0.000000      0.000000      0.000000     2100000
 0.8070250E-01-0.4241694E-01  0.000000      0.000000      0.000000     1200000
 0.1000861    -0.2940128E-01  0.000000      0.000000      0.000000     0300000
  0.000000      0.000000    -0.7153848E-02-0.3667826E-02  0.000000     2010000
  0.000000      0.000000    -0.2824332E-01-0.1430684E-01  0.000000     1110000
  0.000000      0.000000    -0.2933240E-01-0.1413666E-01  0.000000     0210000
 0.6783390E-02-0.3649954E-02  0.000000      0.000000      0.000000     1020000
 0.1346380E-01-0.7128882E-02  0.000000      0.000000      0.000000     0120000
  0.000000      0.000000    -0.7130168E-02-0.3650211E-02  0.000000     0030000
  0.000000      0.000000     0.1346553E-01 0.6807363E-02  0.000000     2001000
  0.000000      0.000000     0.5367065E-01 0.2689766E-01  0.000000     1101000
  0.000000      0.000000     0.9982139E-01 0.2683215E-01  0.000000     0201000
-0.2555571E-01 0.1356764E-01  0.000000      0.000000      0.000000     1011000
-0.5365940E-01 0.2689760E-01  0.000000      0.000000      0.000000     0111000
  0.000000      0.000000     0.4040343E-01 0.2035396E-01  0.000000     0021000
 0.2433279E-01-0.1279456E-01  0.000000      0.000000      0.000000     1002000
 0.9981183E-01-0.2683287E-01  0.000000      0.000000      0.000000     0102000
  0.000000      0.000000    -0.8070569E-01-0.3839038E-01  0.000000     0012000
 0.2012279E-01-0.1077280E-01  0.000000      0.000000      0.000000     1000002
 0.7992961E-01-0.2286460E-01  0.000000      0.000000      0.000000     0100002
  0.000000      0.000000    -0.2286460E-01-0.1077280E-01  0.000000     0010002
  0.000000      0.000000     0.1000774     0.2440162E-01  0.000000     0003000
  0.000000      0.000000     0.7992961E-01 0.2012279E-01  0.000000     0001002
------------------------------------------------------------------------------
```

Figure 2.4: This is an example of a map generated using COSY Infinity. The map represents the system in the example code to third order with the addition of energy as a parameter; its exponent is in the farthest right column.

```
RPE .1*para(1); UM;

DL .04;

MQ .02 .0006 .02;

DL .08;

MQ .02 -.0006 .02;

DL .04;

TS MU;{ This determines the Betatron Tunes.}

WRITE 6 MU(1) MU(2);{This prints the betatron tunes to the screen.}

PM 6;

ENDPROCEDURE; RUN; End;
```

we will receive as output the betatron tunes, which are seen in Figure 2.5. The first entries are simply the x and y tunes, while the subsequent entries are the higher order derivatives of the tune with respect to the energy; the first-order derivatives of the betatron tune are referred to as the chromaticities.



```
I  COEFFICIENT            ORDER EXPONENTS
1  0.7957325564546872E-02   0   0 0  0 0  0
2  -.4334166189499174E-02   1   0 0  0 0  1
3  0.1536720280990319E-02   2   2 0  0 0  0
4  0.2876748828329121E-02   2   0 0  2 0  0
5  0.3186397434193299E-02   2   0 0  0 0  2
-------------------------------------------
I  COEFFICIENT            ORDER EXPONENTS
1  0.7957325564547578E-02   0   0 0  0 0  0
2  -.4334166189498799E-02   1   0 0  0 0  1
3  0.1279121535686945E-03   2   2 0  0 0  0
4  0.1536720280990221E-02   2   0 0  2 0  0
5  0.3186397434193237E-02   2   0 0  0 0  2
-------------------------------------------
```

Figure 2.5: These are the tunes calculated with energy as a variable.

2.5 Strang Splitting

Strang splitting is a way of taking a system where there are multiple physical effects that can each be separately described by a differential equation in the coordinates $\vec{z} = (\vec{q}, \vec{p})$:

$$\begin{aligned} \frac{d\vec{z}}{ds} &= \vec{g_1}(\vec{z}, s) \Longrightarrow \vec{f_1}(s), && (2.26) \\ \frac{d\vec{z}}{ds} &= \vec{g_2}(\vec{z}, s) \Longrightarrow \vec{f_2}(s), && (2.27) \end{aligned}$$

where s is the independent variable, $\vec{g_1}(\vec{z}, s)$ and $\vec{g_2}(\vec{z}, s)$ are arbitrary smooth vector functions, and the initial conditions are known. This method works for any set of differential equations, and does not necessarily require them to be Hamiltonian. Since the dominant forces in this effect are the horizontal and vertical magnetic fields, an A_z can always be derived to account for these fields, and symplecticity is not an issue. While there is a small longitudinal field, COSY has a routine that will symplectify the map [27]. While the method can be adapted for non-autonomous systems [28, 29, 30], for numerical purposes the system was broken into sections small enough to be considered autonomous. Assuming known solutions $(\vec{f}_1(s),\ \vec{f}_2(s))$ to the separate equations, Strang splitting shows that a good approximation of the solution is [31, 32]

$$\frac{d\vec{z}}{ds} \;=\; \vec{g}_1(\vec{z}) + \vec{g}_2(\vec{z}) \Longrightarrow \vec{f}_1(\frac{s}{2}) \circ \vec{f}_2(s) \circ \vec{f}_1(\frac{s}{2}) + \mathcal{O}(s^3), \tag{2.28}$$

which gets increasingly accurate with decreasing s.

In order to change as little as possible the existing COSY elements, we apply Strang splitting with the following identifications:

$$\vec{f}_1(s) \;\longmapsto\; \mathcal{M}(s), \tag{2.29}$$

$$\vec{f}_2(s) \;\longmapsto\; \mathcal{K}(s), \tag{2.30}$$

where $\mathcal{K}$ is a kick (due to the earth's field or space charge), $\mathcal{M}$ is the COSY-generated map for the element, and s is (a fraction of) the length of the element.

2.6 UMER Design and Overview

The University of Maryland Electron Ring is a small-radius, low-energy electron storage ring located at the University of Maryland College Park campus. In most large-scale electron beams the electron's low mass means that it can be accelerated very quickly. Since the force caused by space charge scales as $\frac{1}{\gamma^2}$, space charge is generally not a major issue with normal intensities. However, for ion beams their much larger masses mean that it takes a much longer time for the particles to get to a speed where space charge no longer matters. The purpose of UMER is to use electrons moving at a low enough energy that they will feel significant space charge effects as a proxy for a higher energy ion beam [33, 34, 35]. A COSY Infinity-generated diagram of the ring is shown in Figure 2.6, while the important elements of the ring are shown in Table 2.1 [36].

Table 2.1: Beam values for various apertures in UMER

Aperture	Current (mA)	Emittance (mm-mr)	r_0 (mm) at aperture	r'_0 (mr) at aperture	Average r (mm)
pencil	0.6	7.6	0.25	−1.3	1.56
6 mA	6.0	25.5	0.875	−4.3	3.33
21 mA	21.0	30.0	1.5	−6.7	5.11
80 mA	78.0	58.9	2.85	−12.7	9.5
100 mA	104.0	64.0	3.2	−14.3	10.91

The beam uses a 10 KeV electron gun to inject electrons into the ring. The peak current of the beam is controlled using a series of metal apertures that are rotated in front of the electron gun. These electrons then pass through a matching section before being injected into the ring. The injection re-circulation section, referred to as the Y-section, has a unique geometry where the injected and recirculated beams both enter the section with an offset. This section is shown in Figure 2.7.

There are three main diagnostics that can be used on UMER: current monitors, phosphor screens and beam position monitors. There are two kinds of current monitors in use on UMER: a Bergoz coil in the matching section, and induction gap current monitors in the main ring. These are connected to oscilloscopes; an example of the readout is shown in Figure 2.8.

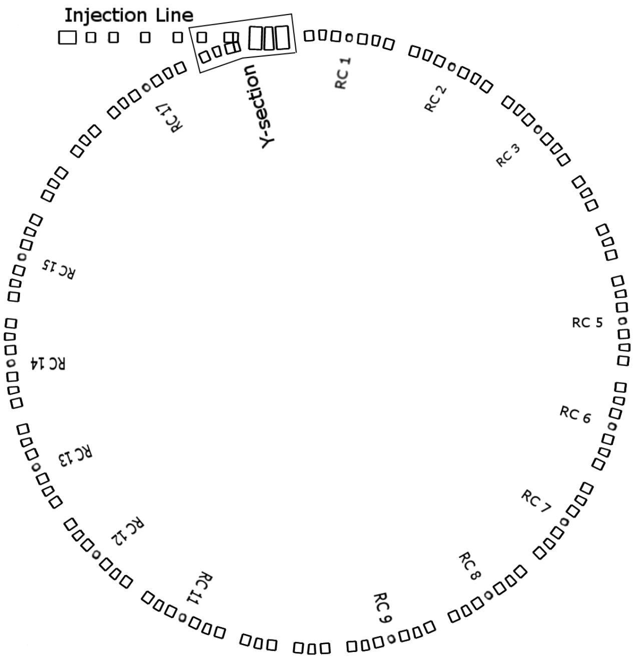


Figure 2.6: A simplified diagram of the ring made using COSY Infinity's graphics package. The sections labeled RC 1, 2, etc., have diagnostic chambers containing phosphor screens and beam position monitors. RC 4, 10 and 16 do not have beam position monitors to make room for other diagnostics.

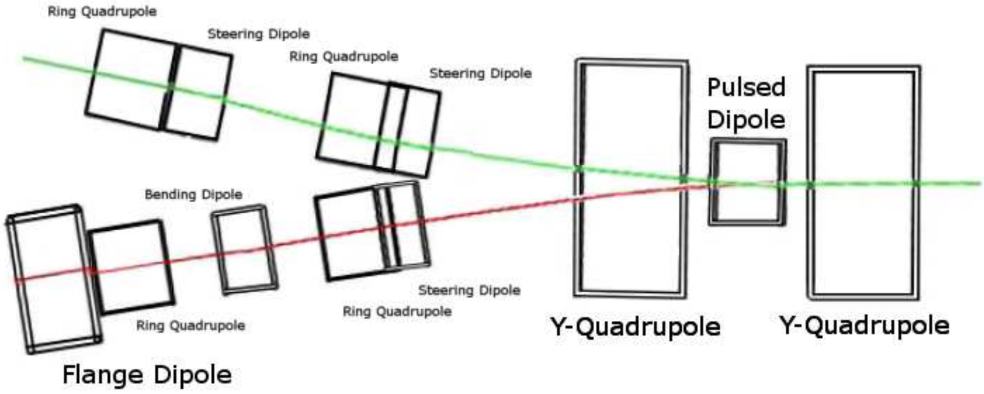
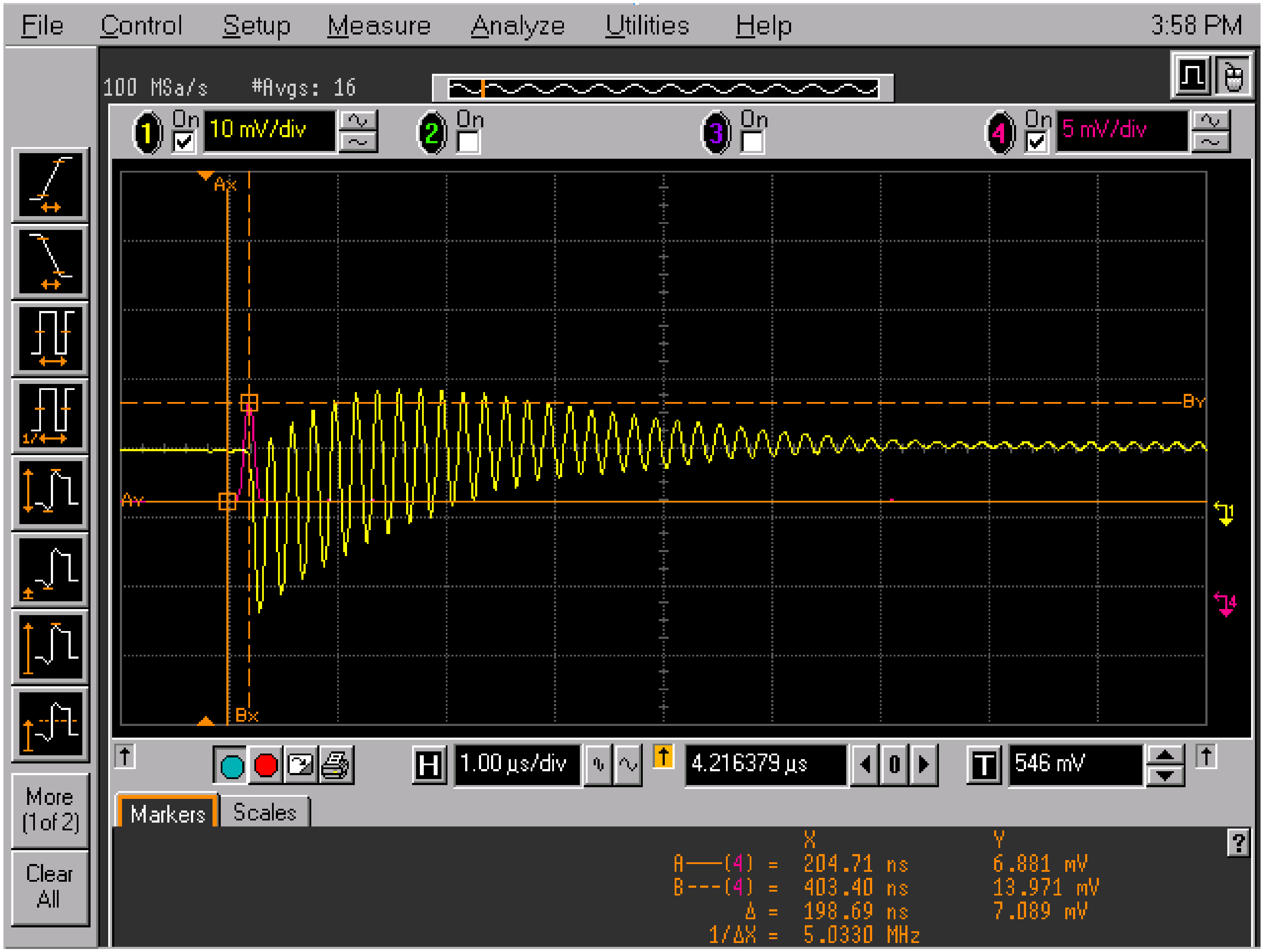


Figure 2.7: A diagram of one particular operating point steering the beam through both injection (green) and recirculation (red) sections.



100 MSa/s #Avgs: 16

1 On ☑ 10 mV/div

2 On ☐

3 On ☐

4 On ☑ 5 mV/div

H 1.00 µs/div

4.216379 µs

T 546 mV

```
                 X              Y
A——(4) =    204.71 ns      6.881 mV
B---(4) =   403.40 ns      13.971 mV
      Δ =   198.69 ns      7.089 mV
   1/ΔX =   5.0330 MHz
```

Figure 2.8: This is a screenshot of the oscilloscope used to monitor the current monitors. The yellow track is the induction gap current monitor in the ring, while the purple track is the Bergoz coil in the injection line.

The phosphor screens are mounted below the beam position monitors and are intercepting. The data is collected using a camera apparatus mounted to the outside of the ring. An example of the apparatus and the type of image that appears is shown in Figure 2.9.

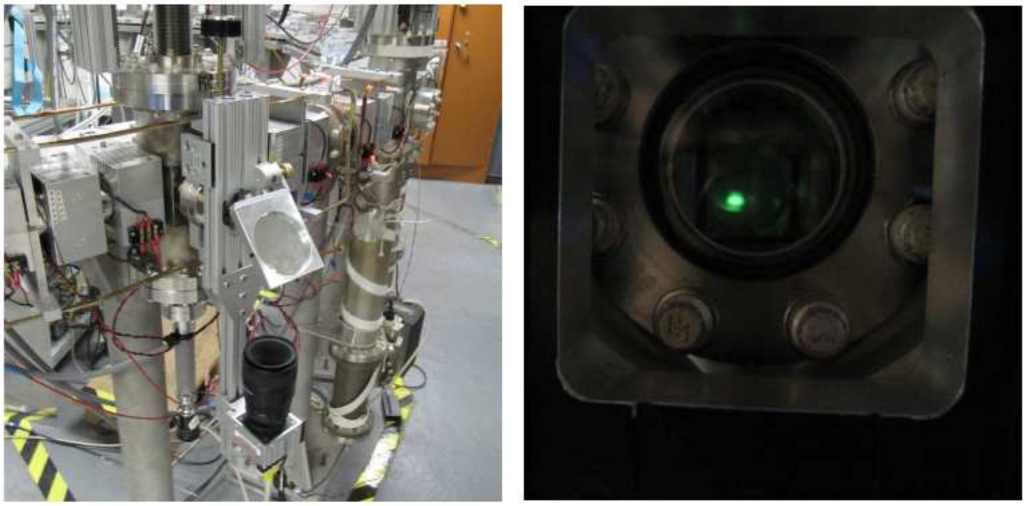


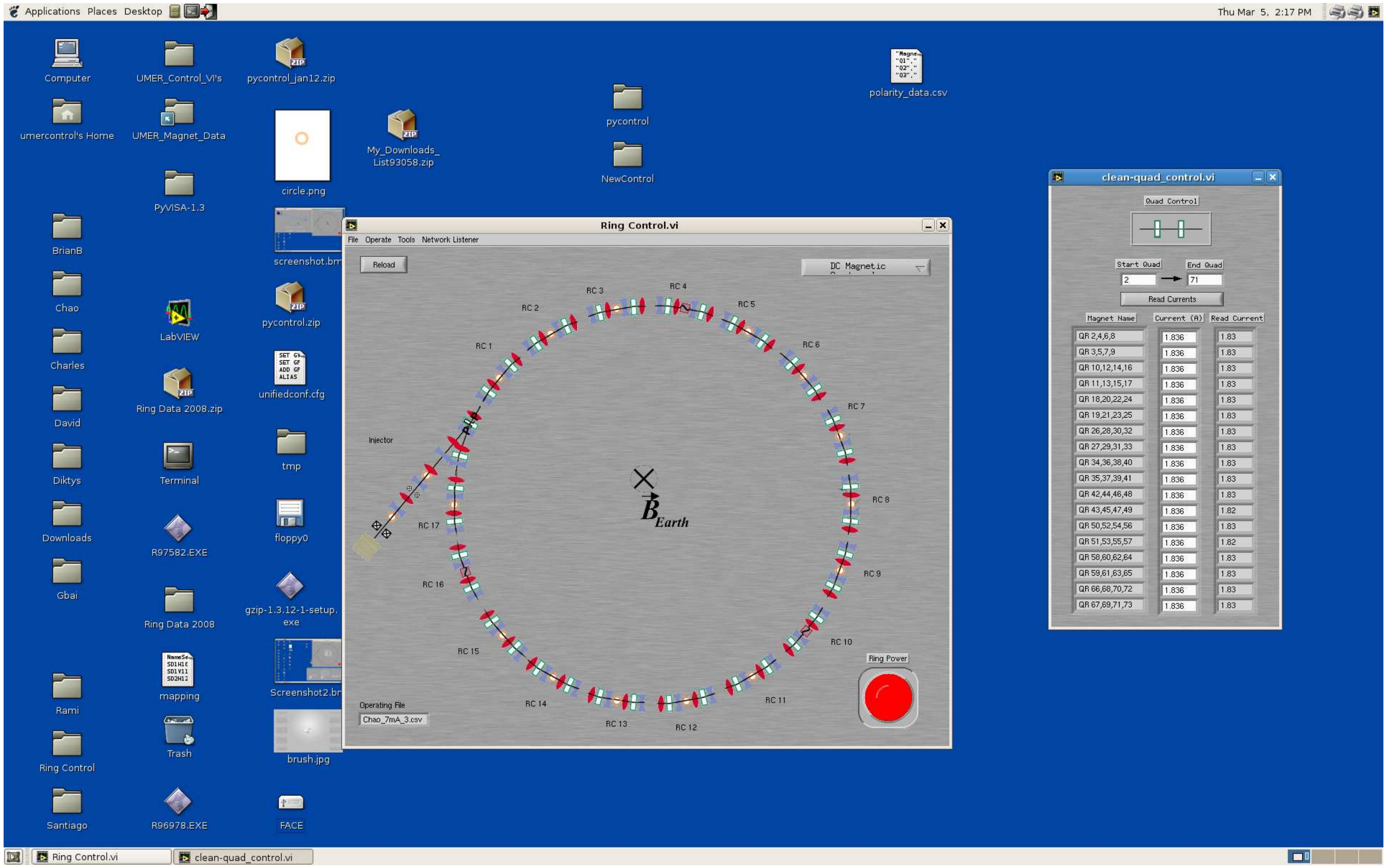
Figure 2.9: On the left is a picture of the camera setup that was used for a number of distribution pictures. The picture on the right shows what a beam looks like on the phosphor screen.

The beam position monitors are of the parallel plate type and can be read using an oscilloscope, and stored using an automated Matlab script. The automated method is used to collect large amounts of data for analysis. An example of one experiment being performed on the ring using the automated method is shown in Figure 2.10.

The entire setup is controlled through a central GUI which allows the user to enter magnet strength settings using either individual numbers or pre-defined files. An example of the GUI is shown in Figure 2.11.

UMER's cost-effective and user-friendly structure allows it to be used for a wide variety of beam dynamics experiments. While the low-energy structure leads to

Figure 2.10: This is a screenshot of the GUI for the automated beam position monitor recording program. The blue line shows the horizontal and the red line the vertical excursion from the centerline of the BPM.



Operating File: Chao_7mA_3.csv

Start Quad: 2 → End Quad: 71

Magnet Name	Current (A)	Read Current
QR 2,4,6,8	1.836	1.83
QR 3,5,7,9	1.836	1.83
QR 10,12,14,16	1.836	1.83
QR 11,13,15,17	1.836	1.83
QR 18,20,22,24	1.836	1.83
QR 19,21,23,25	1.836	1.83
QR 26,28,30,32	1.836	1.83
QR 27,29,31,33	1.836	1.83
QR 34,36,38,40	1.836	1.83
QR 35,37,39,41	1.836	1.83
QR 42,44,46,48	1.836	1.83
QR 43,45,47,49	1.836	1.82
QR 50,52,54,56	1.836	1.83
QR 51,53,55,57	1.836	1.82
QR 58,60,62,64	1.836	1.83
QR 59,61,63,65	1.836	1.83
QR 66,68,70,72	1.836	1.83
QR 67,69,71,73	1.836	1.83

Figure 2.11: This is a screenshot of the GUI developed to control the magnets in the ring. This directly controls the power supplies for each magnet.

some confounding factors not present in other machines, it also allows for a degree of experimentation with machine settings that are not available with higher energy machines where any particles hitting the beam pipe can cause radioactive activation.

CHAPTER 3

TOOLS DEVELOPED

In order to fully model the University of Maryland Electron Ring, a number of tools had to be developed and implemented in COSY Infinity. In some cases it was meant to fill holes in the original COSY framework that UMER's unique structure brought forth; in other instances it was meant to dramatically increase the abilities of the code.

First a set of tools are developed to analyze the single-particle physics in the University of Maryland Electron Ring. This is shown in Section 3.1. In order to simulate the single-particle physics in the ring, new elements such as a UMER short solenoid and the steering dipoles are implemented in subsections 3.1.1 and 3.1.3 respectively. This section will also explore how to include the effects of Earth's magnetic field in the simulations of the ring. This is described in subsection 3.1.2.

Next the tools needed to add the effects of space charge to the system are described. The purpose of this research is to add the effects of space charge to the transfer map of the system. The way this is done is shown in Section 3.2. In order for this method to remain self-consistent, the effects of space charge on the particles needs to be included in their trajectories. While this normally can be done using the map calculated, it was not immediately apparent that it could be used this way back when the research first began. Therefore another method of advancing the particles with space charge was developed as an implementation of the fast multipole method (fmm), which is outlined in Section 3.3.

The moment and fast multipole methods have been implemented as their own stand-alone add-on to COSY Infinity in order to make sure that only one set of code needs to be changed for each version update. To make the code as user friendly as possible we have created a new binary file which is used in place of the COSY file which is called COMFY. Therefore in order to use these methods the user need only have the first line of code read:

```
INCLUDE ’COMFY’;
```

This plugin is explained in detail in Appendix A.

3.1 Single Particle Tools

COSY infinity is a single-particle design code. While this should make the act of modeling the single-particle dynamics of UMER a trivial matter, there were several complications that required a fair amount of tailoring of the original COSY routines.

3.1.1 Short Solenoid

Immediately after the aperture of the electron gun in UMER is a short solenoid. This solenoid is almost entirely fringe fields, and its field profile does not conform to any of the pre-stored fields used in the various solenoids that COSY possesses. The field profile of the UMER solenoid is shown in Figure 3.1.

This field was calculated by the UMER staff as following the equation [37]:

$$B_z(z) \quad = \quad B_0 e^{-\frac{z^2}{d^2}} \left(sech(\frac{z}{b}) + c_0 sinh^2(\frac{z}{b}) \right), \tag{3.1}$$

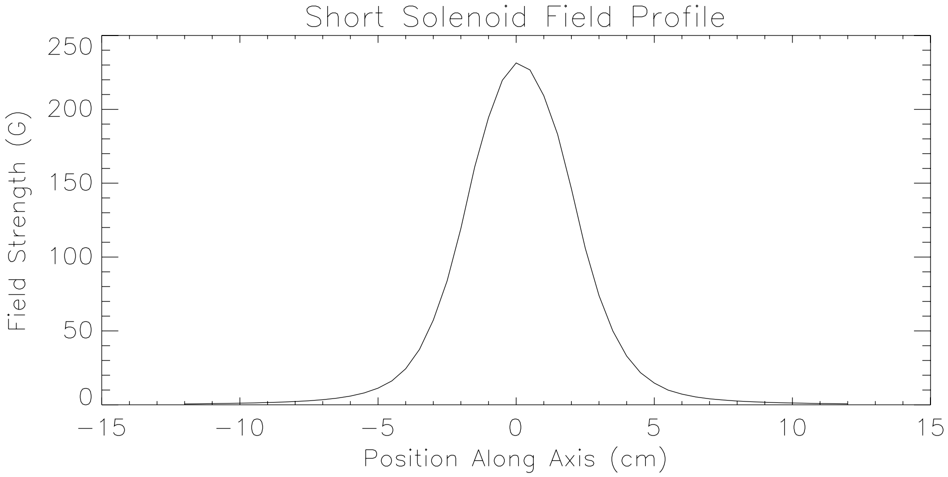


Figure 3.1: This is a plot of the field profile of the UMER short solenoid.

where B_0 is the field at the center of the solenoid, and the parameters d, b, and c_0 are fitted as 4.8197cm, 3.4274cm, and 0.016976 respectively. The profile was then added into the COSY beam package integrator. COSY determines its maps by integrating the path of a reference particle through a potential defined in one specific routine. An example of this addition is shown:

```
PROCEDURE CMU B ld lb lc D L; {UMER Magnetic Solenoid}
 NPOL:=1; PPOL(1):=B PPOL(2):=ld;
 PPOL(3):=lb; PPOL(4):=lc; PPOL(5):=L;
 MPOL(1):=20; NSDP:=15; LOFF:=1; CE:=’CMU’;
 DSDED -7.5*D -7.5*D -7.5*D 7.5*D+L 1e-4*D 3*D D; ENDPROCEDURE;
```

The NDSP number is called in the actual element; this tells the integrator which field profile to use.

3.1.2 Earth's Magnetic Field

Due to UMER's low energy, the earth's magnetic field becomes a nontrivial confounding factor. The vertical field is so strong that it actually accounts for 15% of the bending in the system. The field lines vary around the ring due to placement of metal equipment as well as effects from other experiments in the same building. The field is measured at each dipole using a Gauss meter that fits into a special enclosure, ensuring that the field is measured at the same point at each dipole. The periods of data taking at UMER were approximately 21 months apart. Because of this the field was measured at both occasions. The two different field profiles are shown in Figure 3.2.

In order to properly model this field it was necessary to use the data given for each dipole and interpolate the values of the field for the elements in between. Linear interpolation was used since the values measured were not in a straight line. A series of tests was performed to determine how many magnetic field kicks were required to accurately model each element. For quadrupoles, 15 kicks were required, while for every other element six were required. A similar method was used to examine the effect of image charge forces on the beam during injection. This method is outlined in Appendix B. In Figure 3.3 we see an example of a field-free UMER section, and one in which the earth's field is included.

In order to keep the correct magnetic fields aligned with the correct elements, the field values were stored in files. The elements of the ring were arranged in sections each containing four quadrupoles and two dipoles. For each of these sections an array is created which has as its elements strings containing the names of the files that contain the proper magnetic field values.

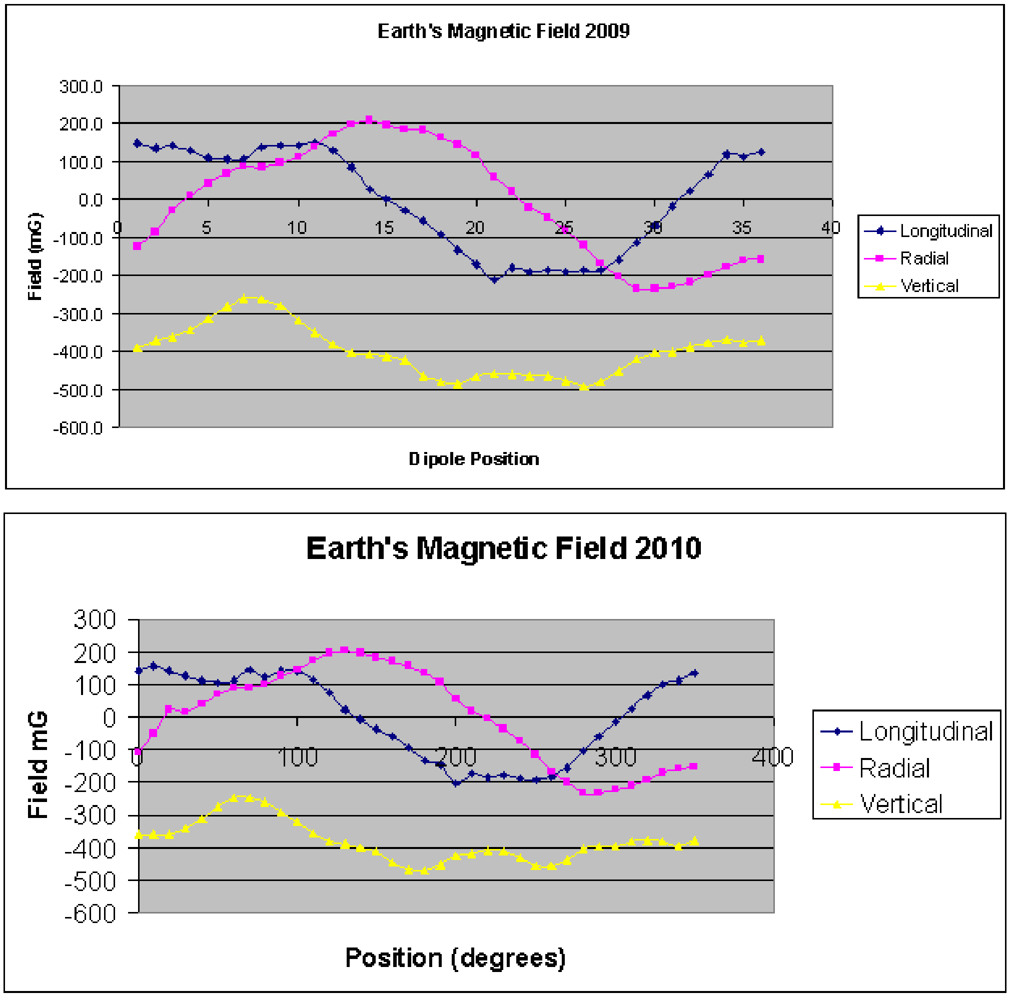


Figure 3.2: This is a plot of the measured Earth's magnetic field taken at two different times.

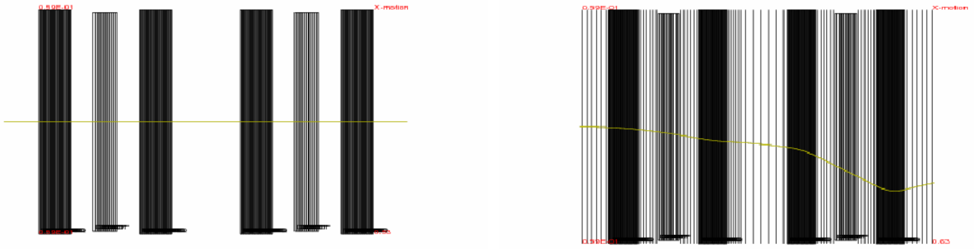


Figure 3.3: On the left is an example of the field-free double FODO cell used in UMER. On the right the effects of the earth's magnetic field are added.

3.1.3 Steering Dipole

In order to counteract the effects of the earth's magnetic field, the angle values for the bending dipoles must be changed. However COSY uses a method of creating bending dipoles that does not directly allow this [38]. The code asks for the angle and the bending radius and then calculates the magnetic fields required for the desired angle by itself. This means that if a beam enters the dipole straight through the center it will exit straight through the center no matter what the actual bending field is. This was remedied by creating a wrap-around routine that makes sure that the bending radius scales based on the angle and the length of the element, and there is also a geometric correction to the exiting beam so that it will have a proper angle with respect to the exiting plane. A diagram of this change is shown in Figure 3.4.

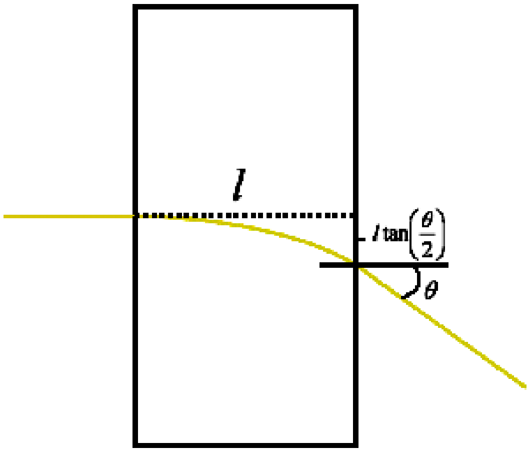


Figure 3.4: This is a diagram of the changes made to the dipoles in COSY to model UMER.

3.1.4 Parallelization

A rudimentary form of parallelization was implemented for the single-particle studies. As will be seen in later sections, most of the single-particle simulations involved scanning across a number of operating points and simulating the entire ring for each of them. This is an example of an embarrassingly parallel system since no communication between computers is necessary. Since a heterogeneous collection of computers were available for this task, it became necessary to create a method for the machines to perform these simulations as they became available.

This was accomplished by creating a master file which contains a number or series of numbers that tell each computer which operating point to simulate. When the computer reads this number it then replaces it with the next in the series, until the final machine replaces it with a number pre-determined that will end the simulation.

This has allowed for a diverse group of computers to be able to quickly perform all of their required tasks in an efficient manner.

3.2 Multiple Particle Tools: Moment Method

Since the effects of space charge are nontrivial, we need to examine how they can be modeled. Currently methods such as particle-in-cell or frozen distributions are used for space charge calculations. Frozen distribution methods work by assuming the beam maintains a certain specific distribution which scales in size but doesn't change its overall shape [39]. Unfortunately while this is a very fast method the accuracy goes down since the distribution may deviate from the assumed one. Particle-in-cell methods work by dividing a distribution into a series of grids, and then averaging the charge over these cells to provide a far-field distribution for each particle.

Any reasonably accurate space charge method will have to act in a self-consistent manner, meaning that it will need to take into account the changing nature of the charge distribution like a particle-in-cell method, but will also need to be valid throughout the region of the beam to be useful for a map like a frozen distribution method.

For this reason we assume that the distribution is a smooth function which can be expanded as a series of smooth functions of the type $\varphi(x)$:

$$f(x,y) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} D_{ij}\varphi_i(x)\varphi_j(y), \tag{3.2}$$

with coefficients D_{ij}. The task is to choose some specific basis functions and to determine the corresponding coefficients based on some limited information available about f.

3.2.1 Determining the Expansion Coefficients

One example of information that can be gathered from a sample of test particles which obey a smooth distribution function is the various statistical moments of f. The moments may be known exactly to all orders, or only approximately to some finite order. The challenge is to extract the coefficients D_{ij} as accurately as possible. There are several ways that this can be done. Here we examine two methods. One method involves calculating the moments of the distribution to find a sum of monomials; the other involves casting the distribution as a series of projections onto orthogonal polynomials.

There are other options for smooth particle approximation of the density function, which are an ongoing field of study.

3.2.1.1 Use of High-Order Moments to Find Taylor Series

This method uses the fact that if two numerical distributions have the same statistical moments then they are mathematically indistinguishable [40]. This will allow us to transform a distribution made up of a collection of delta functions into a Taylor series. If the moments up to a given order are known, the series coefficients up to the same order can be determined. We assume that the higher the maximum order, the more accurate the approximation will be. Nevertheless, the answers will still not be unique, though it can be shown that among all single-variable probability density functions ρ with the same first $2p$ moments as f satisfy the relation [41],

$$|f\left(x\right)-\rho\left(x\right)|\leq\frac{1}{\mathbf{V}_p^T\left(x\right)\mathbf{M}_p^{-1}\mathbf{V}_p\left(x\right)},\tag{3.3}$$

where $\mathbf{V}_p\left(x\right)=\left(1,x,x^2,\ldots,x^p\right)$ and

$$\mathbf{M}_p=\begin{pmatrix} m_1 & m_2 & \cdots & m_p \\ m_2 & m_3 & \cdots & m_{p+1} \\ m_3 & m_4 & \cdots & m_{p+2} \\ \vdots & \vdots & \vdots & \vdots \\ m_{p+1} & m_{p+2} & \cdots & m_{2p} \end{pmatrix},\tag{3.4}$$

m_i being the moments that the functions have in common. Thus, the tail probabilities will become indistinguishable first, followed by core probabilities. Convergence speed scales with x^{-2p}. This is useful in providing practical guidance about the truncation orders to be employed in practice.

The statistical moments are defined as

$$M_{nm}=\int_{x_\ell}^{x_u}\int_{y_\ell}^{y_u}x^ny^mf(x,y)dxdy.\tag{3.5}$$

In our case the test particle distribution $f(x,y)$ is merely a sum of delta functions $\sum\limits_i\delta(x-x_i)\delta(y-y_i)$. The moments can be directly calculated from the distribution,

$$M_{nm}=\sum_i x_i^ny_i^m.\tag{3.6}$$

In the code this is accomplished with algorithm 1. We can also determine, assuming the basis functions are monomials, that the moments for a generic Taylor series are calculated as

```
for k=1,Number of Particles do
  $\ell = 0$
  for i=0,max order do
    for j=0,max order-i do
      Moment($\ell + j$) = Moment($\ell + j$)+$x_k^j y_k^i$
    end for
    $\ell = \ell + j$
  end for
end for
```

$$M_{nm} = \int_{-x_r}^{x_r} \int_{-y_r}^{y_r} x^n y^m \sum_i \sum_j C_{ij} x^i y^j dxdy. \tag{3.7}$$

we combine the two,

$$\sum_i x_i^n y_i^m = \int_{-x_r}^{x_r} \int_{-y_r}^{y_r} x^n y^m \sum_i \sum_j C_{ij} x^i y^j dxdy, \tag{3.8}$$

the integrals on the right are trivially solved,

$$\sum_i x_i^n y_i^m = \sum_i \sum_j N_{(n+i)(m+j)}(x_r, y_r) C_{ij}, \tag{3.9}$$

where

$$N_{kl}(x,y) = \frac{x^{k+1} y^{l+1}}{(k+1)(l+1)} ((1-(-1)^{k+1})(1-(-1)^{l+1})). \tag{3.10}$$

In the code we assume that $x_r = y_r = 1$. This is because, since we are using very high-order powers of these bounds of integration, we want to minimize numerical problems from the system; if we used small-scale bounds of integration, the high-order terms would go to zero, whereas if we used large bounds of integration, the high-order terms would drown out the lower order ones. This method, using $x_r = y_r = 1$, is represented using algorithm 2. This leads to a matrix equation equating a vector containing the moments to a matrix containing the integrals times a vector containing the Taylor coefficients. In order to find the Taylor coefficients it is necessary to invert the matrix; however, the matrix that results is of the form seen in (3.11), which is an ill-conditioned matrix. Because of this, a singular value decomposition must be performed to invert the matrix [42]. When the matrix is inverted and multiplied by the calculated moments, the coefficients of the Taylor series are extracted:

```
m=0
for k=1,max order do
  for l=1,max order-(k-1) do
    n=0
    for i=1,max order do
      for j=1,max order-(i-1) do
        $T_{(j+n)(l+m)} = \frac{(1-(-1)^{k+i-1})(1-(-1)^{l+j-1})}{(k+i-1)(l+j-1)}$
      end for
      n=n+j
    end for
  end for
  m=m+l
end for
```

$$T = \begin{pmatrix} 1 & 0 & \frac{1}{3} & 0 & \frac{1}{5} & 0 & \cdots & \frac{1}{p+1} \\ 0 & \frac{1}{3} & 0 & \frac{1}{5} & 0 & \cdots & \frac{1}{p+1} & 0 \\ \frac{1}{3} & 0 & \frac{1}{5} & 0 & \cdots & \frac{1}{p+1} & 0 & \vdots \\ 0 & \frac{1}{5} & 0 & \cdots & \frac{1}{p+1} & 0 & & \vdots \\ \frac{1}{5} & 0 & \cdots & \frac{1}{p+1} & 0 & & & \vdots \\ 0 & \cdots & \frac{1}{p+1} & 0 & & \ddots & & \vdots \\ \vdots & \frac{1}{p+1} & 0 & & & & \ddots & \vdots \\ \frac{1}{p+1} & 0 & \cdots & \cdots & \cdots & \cdots & \cdots & 0 \end{pmatrix}. \tag{3.11}$$

3.2.1.2 Use of Legendre Polynomials

Another way to approximate the distribution is using a series of Legendre polynomials as the φ in (3.2). This is accomplished using the orthogonality relations,

$$\int_{-1}^{1} P_\ell(x) P_m(x) dx = \frac{2}{2\ell + 1} \delta_{\ell m}, \tag{3.12}$$

and assuming that the distribution function can be characterized as

$$\rho(x, y) = \sum_{i+j \leq n} \tilde{C}_{ij} P_i(x) P_j(y). \tag{3.13}$$

The coefficients $\tilde{C}_{ij}$ can be calculated as

$$\int_{-1}^{1}\int_{-1}^{1} P_i(x)P_j(y)\rho(x,y)dxdy = \tilde{C}_{ij}\frac{2}{2i+1}\frac{2}{2j+1}. \tag{3.14}$$

We are essentially calculating the Legendre polynomial moments of the distribution instead of the monomial moments used in the moment method. On the other hand, if we assume we already know the monomial moments, it is easy to calculate the Legendre coefficients directly, as follows. The same moment matching conditions are applied; that is, we require that the moments of ρ are the same as the actual moments of f up to a fixed order n. Since every product of Legendre polynomials can be re-expanded into a power series,

$$P_i(x)P_j(y) = \sum_{k+l\leq i+j} a_{kl}x^k y^l, \tag{3.15}$$

the left-hand side of (3.14) can be written as a linear combination of these moments, and we obtain

$$\tilde{C}_{ij} = \frac{(2i+1)\,(2j+1)}{4}\vec{A}\cdot\vec{M}, \tag{3.16}$$

where $\vec{A}$ is a coefficient vector built from a_{kl}. Hence, we obtain an approximation of the original distribution as a series of Legendre polynomials without the need to invert any matrices.

3.2.2 Using the Series to Find the Potential

The methods used previously allow us to determine an approximation of the distribution function of the system using smooth functions; now we intend to use these functions to determine the potential of the system also described as a set of smooth functions. Using the differential algebraic methods that were outlined in Section 2.2 we will create a differential algebraic vector which represents the potential of the distribution out to its edges. The gradient can then be determined using differential algebras, thus creating two DA vectors that will describe the electric fields throughout the region in question.

3.2.2.1 Determining the Integrals

Once the distribution function has been determined it should be a simple matter to integrate the distribution times the proper Green's function to determine the potential. Unfortunately this integral can be problematic since the expansion point is inside the distribution. While it should be technically possible to integrate with the stepsize changed such that it skips over the singular point, it is time consuming, it reduces accuracy, and it must be re-computed for each integral. If we are interested in a 25^{th}-order expansion then we would need to solve 325 integrals, and have the singular points skipped in each. Since each of these integrals go to zero at different rates, the time and accuracy savings of having an adaptable stepsize become significant. For this reason a Duffy transformation has been implemented [43].

If an analytic charge distribution function is denoted by ρ and the potential is evaluated at the point (x_0, y_0), the value of the integral,

$$I = \int_c^d \int_a^b \rho(x,y) \ln(\sqrt{(x-x_0)^2 + (y-y_0)^2}) dx dy, \tag{3.17}$$

is the potential at that point. We assume that the support of ρ is inside the rectangle $[a,b] \times [c,d]$ and we are interested in the potential inside this region.

First, the rectangle is subdivided into four rectangles as shown in Figure 3.5, which moves the singularity from the interior of the domain to the corners of four regions.

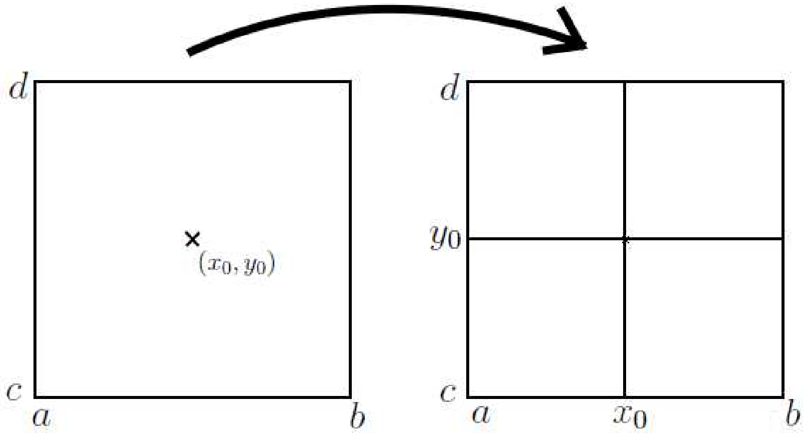


Figure 3.5: The full integration region is subdivided into four smaller regions with one of their corners on the expansion point.

The total integral is now transformed into four integrals which can have their limits of integration rearranged such that the equation becomes

$$\begin{aligned} I = &\int_{y_0}^{c}\int_{x_0}^{a} \rho(x,y)\ln(\sqrt{(x-x_0)^2+(y-y_0)^2})dxdy \\ &- \int_{y_0}^{c}\int_{x_0}^{b} \rho(x,y)\ln(\sqrt{(x-x_0)^2+(y-y_0)^2})dxdy \\ &- \int_{y_0}^{d}\int_{x_0}^{a} \rho(x,y)\ln(\sqrt{(x-x_0)^2+(y-y_0)^2})dxdy \\ &+ \int_{y_0}^{d}\int_{x_0}^{a} \rho(x,y)\ln(\sqrt{(x-x_0)^2+(y-y_0)^2})dxdy. \end{aligned} \tag{3.18}$$

These are now all the same kind of integrals, merely with different bounds of integration. Henceforth we will look at the first integral in the series, remembering that all four will be solved in the same way. We continue by rescaling the rectangles to unit squares by the coordinate transformation,

$$u_1 = \frac{x - x_0}{a - x_0}, \tag{3.19}$$

$$u_2 = \frac{y - y_0}{c - y_0}, \tag{3.20}$$

$$dx = (a - x_0)du_1, \tag{3.21}$$

$$dy = (c - y_0)du_2. \tag{3.22}$$

For expedience we will use $\lambda_1 = (a - x_0)$ and $\lambda_2 = (c - y_0)$. The first integral in (3.18) now becomes

$$I_{ac} = \int_0^1 \int_0^1 \lambda_1 \lambda_2 \rho(\lambda_1 u_1 + x_0, \lambda_2 u_2 + y_0) \ln(\sqrt{\lambda_1^2 u_1^2 + \lambda_2^2 u_2^2}) du_1 du_2. \tag{3.23}$$

As shown in Figure 3.6, we cut the square into two triangles that are integrated separately:

$$\begin{aligned} I_{ac} = & \int_0^1 \int_0^{u_1} \lambda_1 \lambda_2 \rho(\lambda_1 u_1 + x_0, \lambda_2 u_2 + y_0) \ln(\sqrt{\lambda_1^2 u_1^2 + \lambda_2^2 u_2^2}) du_1 du_2 \\ & + \int_0^{u_2} \int_0^1 \lambda_1 \lambda_2 \rho(\lambda_1 u_1 + x_0, \lambda_2 u_2 + y_0) \ln(\sqrt{\lambda_1^2 u_1^2 + \lambda_2^2 u_2^2}) du_1 du_2. \end{aligned} \tag{3.24}$$

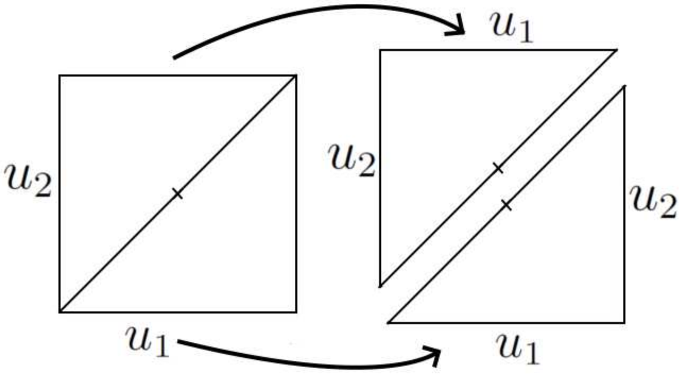


Figure 3.6: This shows how the integration region is further subdivided into triangles.

In order to finish the Duffy transformation the triangles are now converted to squares yet again. This is accomplished with the transformation

$$u_1 = w_1, \tag{3.25}$$

$$u_2 = w_1 w_2, \tag{3.26}$$

in the first integral in (3.24), which makes the integral take the following form:

$$\int_0^1 \int_0^1 \lambda_1 \lambda_2 w_1 \rho(\lambda_1 w_1 + x_0, \lambda_2 w_1 w_2 + y_0) \ln(\sqrt{\lambda_1^2 w_1^2 + \lambda_2^2 w_1^2 w_2^2}) dw_1 dw_2. \tag{3.27}$$

This is simplified to

$$\lambda_1\lambda_2\int_0^1\int_0^1 \rho(\lambda_1 w_1 + x_0, \lambda_2 w_1 w_2 + y_0)(w_1\ln(w_1) + w_1\ln(\sqrt{\lambda_1^2+\lambda_2^2 w_2^2}))dw_1dw_2. \tag{3.28}$$

Since $\lim_{w_1\to 0} w_1\ln(w_1) = 0$, and the argument of the other logarithm is never zero, the singularity is removed. A similar transform is made to the second triangle (and the second term in [3.24]),

$$u_1 = w_1 w_2, \tag{3.29}$$

$$u_2 = w_2, \tag{3.30}$$

converting it into the following:

$$\lambda_1\lambda_2\int_0^1\int_0^1 \rho(\lambda_1 w_1 w_2 + x_0, \lambda_2 w_2 + y_0)(w_2\ln(w_2) + w_2\ln(\sqrt{\lambda_2^2+\lambda_1^2 w_1^2}))dw_1dw_2.$$

Assembling the partial results, we finally obtain a singularity-free formula for the computation of the potential,

$$\begin{aligned}\phi(x_0,y_0) = &\sum_{n=1}^{4}\lambda_{1,n}\lambda_{2,n}\int_0^1\int_0^1 dw_1dw_2\\ &\times\left[\begin{array}{c}\rho(\lambda_{1,n}w_1 + x_0, \lambda_{2,n}w_1w_2 + y_0)(w_1\ln(w_1) + w_1\ln(\sqrt{\lambda_{1,n}^2+\lambda_{2,n}^2w_2^2}))\\ +\rho(\lambda_{1,n}w_1w_2 + x_0, \lambda_{2,n}w_2 + y_0)(w_2\ln(w_2) + w_2\ln(\sqrt{\lambda_{2,n}^2+\lambda_{1,n}^2w_1^2})))\end{array}\right].\end{aligned} \tag{3.31}$$

3.2.2.2 Scaling the Integrals

It was previously shown that for high-order distributions a large number of integrals would be required. Having to recompute each integral for each section would be prohibitive in a system that requires a large number of potential evaluations. It would be useful to be able to perform the integrals once and then simply combine them with the Taylor coefficients that have been calculated. If we assume the region that is being investigated is a square, then we can pre-store the integrals and then scale them to the size needed.

The scaling of the integrals is given using dimensional analysis. If the distributions are given as

$$\rho_{a.u.}(x_{a.u.}, y_{a.u.}) = C_{00} + C_{10}x_{a.u.} + C_{01}y_{a.u.} + ..., \tag{3.32}$$

$$\rho_{S.I.}(x_{S.I.}, y_{S.I.}) = D_{00} + D_{10}x_{S.I.} + C_{01}y_{S.I.} + ..., \tag{3.33}$$

the coefficients are going to have a form $C_{nm} = [[\rho]]/\ell_{a.u.}^{n+m+2}$ and $D_{nm} = [[\rho]]/\ell_{S.I.}^{n+m+2}$. Then, assuming that $x_{S.I.} = ax_{a.u.}$ and $y_{S.I.} = ay_{a.u.}$ with the same scaling factor a, the potentials will be given as

$$\phi_{a.u.}(x_{0,a.u.}, y_{0,a.u.}) = \sum_{n,m} C_{nm} \int_{-1}^{1}\int_{-1}^{1} x_{a.u.}^{n} y_{a.u.}^{m}$$

$$\times \ln(\sqrt{(x_{a.u.} - x_{0,a.u.})^2 + (y_{a.u.} - y_{0,a.u.})^2}) dx_{a.u.} dy_{a.u.}, \tag{3.34}$$

$$\phi_{S.I.}(x_{0,S.I.}, y_{0,S.I.}) = \sum_{n,m} D_{nm} \int_{-a}^{a}\int_{-a}^{a} x_{S.I.}^{n} y_{S.I.}^{m}$$

$$\times \ln(\sqrt{(x_{S.I.} - x_{0,S.I.})^2 + (y_{S.I.} - y_{0,S.I.})^2}) dx_{S.I.} dy_{S.I.}. \tag{3.35}$$

The integral can be scaled in the following way:

$$I_{nm,S.I.}(x_0, y_0) = \int_{-1}^{1}\int_{-1}^{1} (a x_{a.u.})^n (a y_{a.u.})^m$$

$$\times \ln(\sqrt{(a x_{a.u.} - a x_{0,a.u.})^2 + (a y_{a.u.} - a y_{0,a.u.})^2}) a dx_{a.u.} a dy_{a.u.}, \tag{3.36}$$

$$I_{nm,S.I.}(x_0, y_0) = a^{n+m+2} \int_{-1}^{1}\int_{-1}^{1} x_{a.u.}^{n} y_{a.u.}^{m} \ln(a)$$

$$+ x_{a.u.}^{n} y_{a.u.}^{m} \ln(\sqrt{(x_{a.u.} - x_{0,a.u.})^2 + (y_{a.u.} - y_{0,a.u.})^2}) dx_{a.u.} dy_{a.u.}, \tag{3.37}$$

$$I_{nm,S.I.}(x_0, y_0) = a^{n+m+2} (\frac{(1-(-1)^{n+1})(1-(-1)^{m+1})}{(n+1)(m+1)} \ln(a)$$

$$+ \int_{-1}^{1}\int_{-1}^{1} x_{a.u.}^{n} y_{a.u.}^{m} \ln(\sqrt{(x_{a.u.} - x_{0,a.u.})^2 + (y_{a.u.} - y_{0,a.u.})^2}) dx_{a.u.} dy_{a.u.}), \tag{3.38}$$

$$I_{nm,S.I.}(x_0, y_0) =$$

$$a^{n+m+2} \left[\frac{(1-(-1)^{n+1})(1-(-1)^{m+1})}{(n+1)(m+1)} \ln(a) + I_{nm,a.u.}\left(\frac{x_{0.}}{a}, \frac{y_0}{a}\right) \right]. \tag{3.39}$$

Combined with the scaling of the coefficients, the potential becomes

$$\phi_{S.I.}(x_0, y_0) = \sum_{nm} C_{nm} \left(\frac{(1-(-1)^{n+1})(1-(-1)^{m+1})}{(n+1)(m+1)} \ln(a) + I_{nm,a.u.}\left(\frac{x_0}{a}, \frac{y_0}{a}\right) \right), \qquad (3.40)$$

where $I_{nm,a.u.}(x, y)$ have been pre-computed and stored.

3.2.3 Using the Potential to Find the Map

Once the potential has been calculated it is necessary to add it to the actual map. Since the potential has been calculated as a DA object taking its gradient is a trivial procedure. Now that the electric fields have been calculated they can be used to solve the equations of motion to determine the kick that will be applied in (2.28). Since we are assuming that this is a zero-length kick, the equations of motion become [25]

$$x' = 0, \qquad (3.41)$$

$$a' = \left[\frac{1+\eta}{1+\eta_0} \frac{p_0}{p_s} \frac{E_x}{\chi_{e0}} + b \frac{B_z}{\chi_{m0}} \frac{p_0}{p_s} - \frac{B_y}{\chi_{m0}} \right] (1+hx) + h \frac{p_s}{p_0}, \qquad (3.42)$$

$$y' = 0, \qquad (3.43)$$

$$b' = \left[\frac{1+\eta}{1+\eta_0} \frac{p_0}{p_s} \frac{E_y}{\chi_{e0}} - a \frac{B_z}{\chi_{m0}} \frac{p_0}{p_s} + \frac{B_x}{\chi_{m0}} \right] (1+hx), \qquad (3.44)$$

where $\eta = \frac{E}{mc^2}$, χ_m and χ_e are the magnetic and electric rigidities respectively, and $\frac{P_0}{P_s}$ is the total momentum over the longitudinat momentum. Furthermore, h is the inverse radius of curvature, and $E_{x,y}$ and $B_{x,y}$ are the electric and magnetic fields respectively. These equations are solved using the COSY routine DAFLO which uses a flow integration mechanism, as outlined in 2.2.3 [44].

3.2.4 Parallelization

The structure of this method readily lends itself to parallelization. Since no gridding or particle passing is required in this method, the test particles can be evenly distributed across all of the machines that are available. Parallelization is achieved using the ploop command in COSY Infinity, which is still undergoing beta testing [45]. Since all of the information about the distribution of the particles is contained in the statistical moments, this is the only information that needs to be passed between nodes. The relationship can be shown in Figure 3.7, where we see that for each evaluation of the potential only one communication between nodes is required.

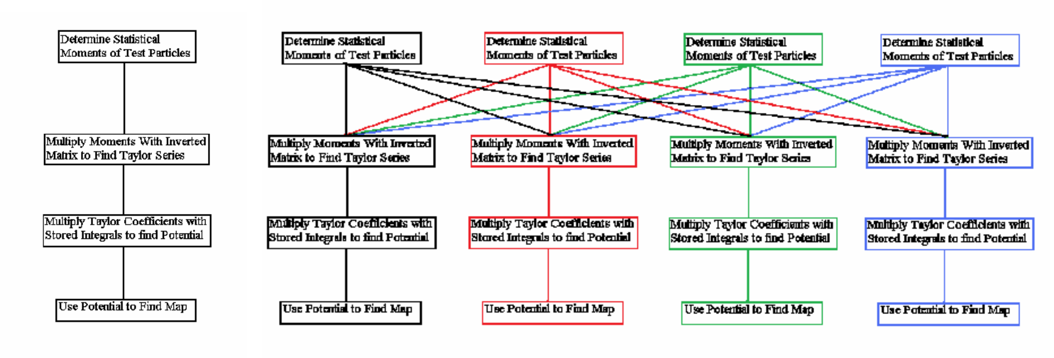


Figure 3.7: This shows how the parallel version of the moment method is produced from the serial version of this method.

Since the method does not involve binning, and any particle can be placed on any node, this algorithm can be massively parallelized. Since the particles can be divided up onto any node, the scaling of execution time with respect to number of CPUs is roughly n^{-1} while the scaling of the execution time with respect to particle number is on the order n. These effects can be seen in Figures 3.8 and 3.9. The script used to implement the parallel version is shown in Appendix C.

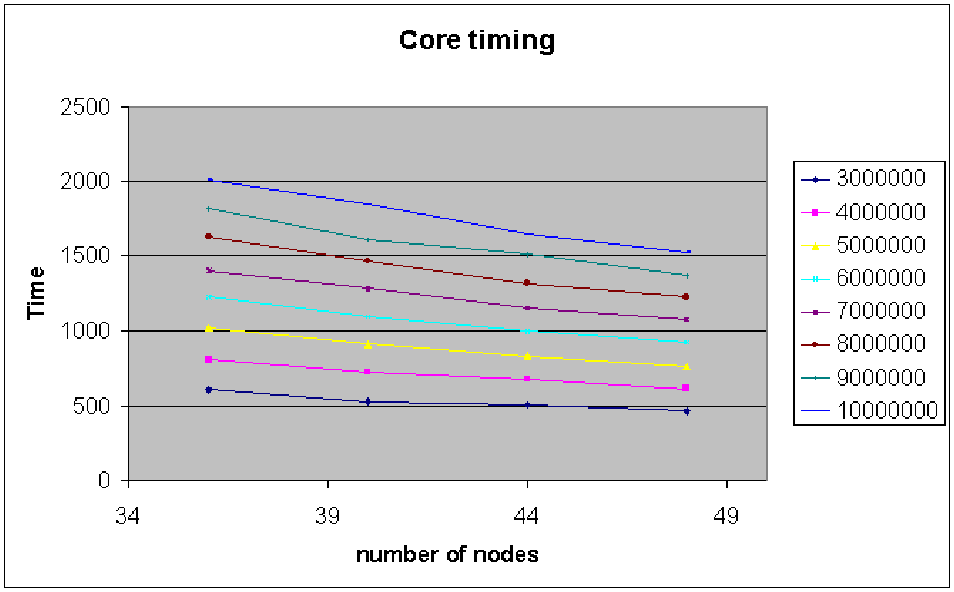


Figure 3.8: This shows how the parallel version of the moment method scales with the number of CPUs.

3.3 Multiple Particle Tools: Fast Multipole Method

While under the proper circumstances the map generated in Section 3.2 can be used to advance the particles through the element, some types of distributions can be problematic. For those types of distributions it is necessary to use another type

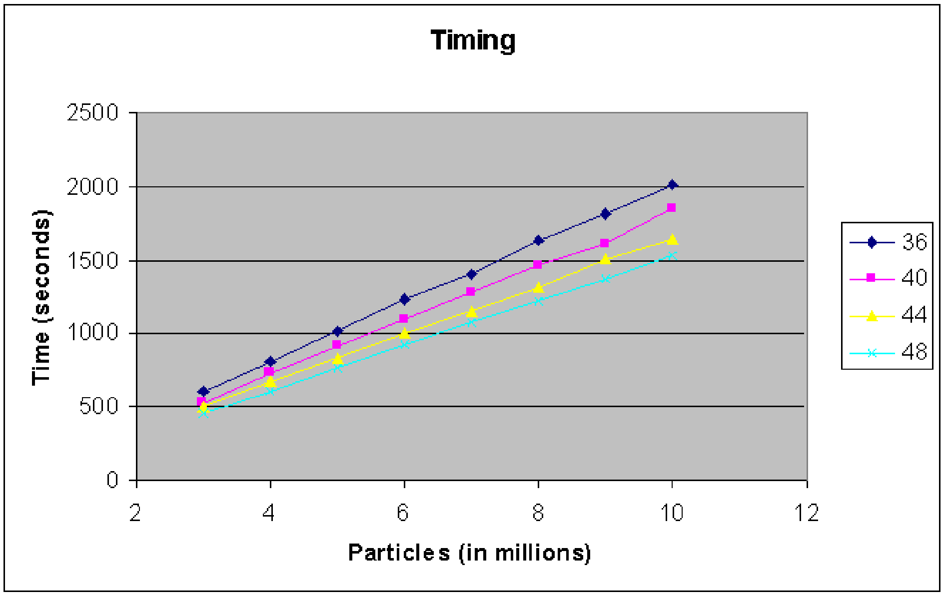


Figure 3.9: This shows how the parallelized version of the moment method scales with particle number.

of Poisson solver. In order to advance particles under space charge in a fast and accurate manner, a version of the fast multipole method has been implemented.

The fast multipole method was first developed in 1985 by Greengard and Rokhlin as a way to solve the n-body problem; it works by using multipole expansions to determine the potential from distant particles and direct Coulomb interactions for near particles [46]. This has produced a fast method that has been heralded by SIAM as one of the top ten algorithms of the 20^{th} century [47].

3.3.1 Overview of FMM

The fast multipole method (FMM) will determine the potential and electric fields of a distribution at a specific point using multipole expansions for distant particles, and Coulomb interactions for close particles. The method works by dividing up the reion of interest into a grid as can be seen in Figure 3.10.

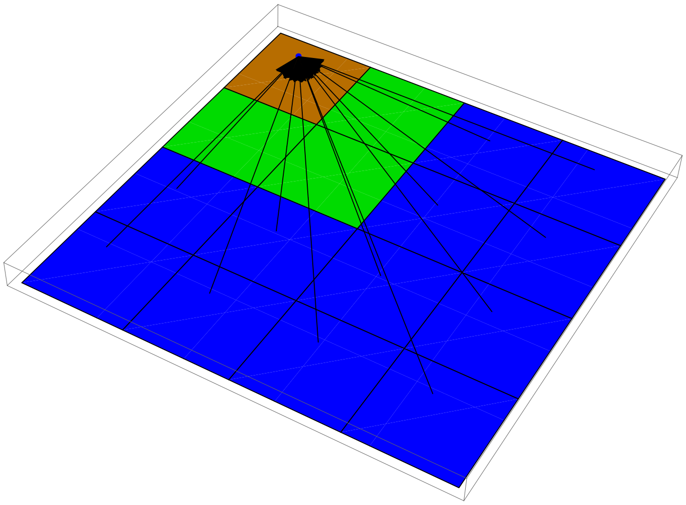


Figure 3.10: The region containing the particles is subdivided into boxes. In this instance it is a two-dimensional region divided into sixteen boxes. The orange box denotes the region containing the particles in question, the green boxes are the nearest neighbors, and the blue boxes are the distant boxes. The distant boxes all contribute their multipole expansions to the local expansion in the orange box.

For distant regions, shown in blue, a multipole transformation is calculated,

$$\phi(z) \quad = \quad Q \log(z) + \sum_{k=1}^{\infty} \frac{a_k}{z^k}, \tag{3.45}$$

where

$$Q = \sum_{i=1}^{m} q_i \quad and \quad a_k = \sum_{i=1}^{m} \frac{-q_i z_i^k}{k}, \tag{3.46}$$

with $z = x + iy$, and q_i,z_i is the charge and complex position of each test particle. The expansion is truncated at order m. These multipole transformations are then converted to a local Taylor expansion for the box in question,

$$\phi(z) \quad = \quad \sum_{\ell=0}^{\infty} b_\ell z^\ell, \tag{3.47}$$

using

$$b_0 \quad = \quad a_0 log(-z_0) + \sum_{k=1}^{\infty} \frac{a_k}{z_0^k}(-1)^k, \tag{3.48}$$

whereas for $\ell \geq 1$,

$$b_\ell \quad = \quad -\frac{a_0}{\ell z_0^\ell} + \frac{1}{z_0^\ell}\sum_{k=1}^{\infty} \frac{a_k}{z_0^k}\binom{\ell+k-1}{k-1}(-1)^k. \tag{3.49}$$

For close regions simple Coulomb point-to-point interactions are given:

$$\phi(z) \quad = \quad \sum_{i=1}^{n} q_i \log(z - z_i), \tag{3.50}$$

where n is the number of particles in the nearby region. Then for each particle the local expansion is combined with the sum of the point particles from the near regions to determine the potential, as shown in figure 3.11.

3.3.2 Overview of DA expansion

The process for creating the local expansion in (3.47) is a time-consuming one; therefore, it was decided to use differential algebras to speed up the process of

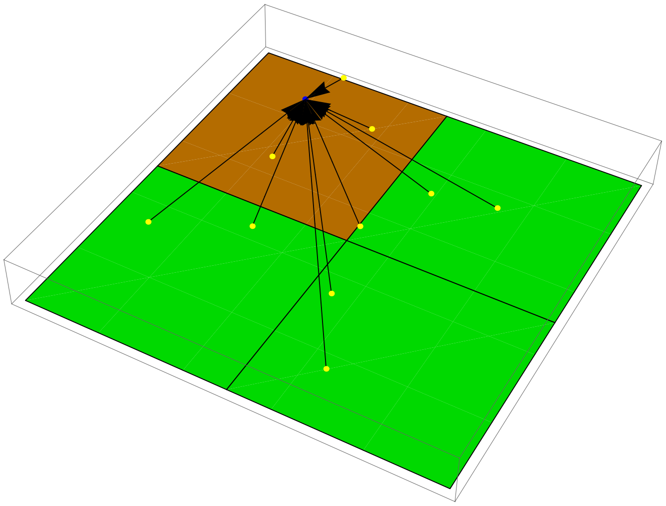


Figure 3.11: This is a closeup of the box in question and its nearest neighbors. The potential is determined by the local expansion and, as is shown here, the point-to-point Coulomb interactions of the particles in the box and their nearest neighbors.

creating the local expansion. This was done by evaluating the multipole expansions of each distant box at the center of the box in question. The differential algebraic vectors then create an expansion that is valid throughout the region in question. The expansion is then calculated using

$$\phi(z+dz) \quad = \quad Q\log(z+dz)+\sum_{k=1}^{\infty}\frac{a_k}{(z+dz)^k}, \tag{3.51}$$

where z is the distance to the local expansion point from the center of the distant box.

3.3.3 Overview of FMM with Multiple Species

One advantage that the FMM has is that since it calculates the dynamics of each particle it can use particles with different masses and charges together in the same beam. The particulars of this are outlined in Appendix A. The method works by taking an input file composed of four numbers for each particle:

```
<charge per macroparticle (elementary charges)> <Energy (MeV)>
 <Mass (amu)> <charge per particle (elementary charge)>
```

Internally it creates a different file which has as its elements in S.I. units:

```
<charge per macroparticle> <Magnetic Rigidity>
<Relativistic eta> <charge per nucleon>
```

This is used in conjunction with COSY's DA methods; it is possible to set the magnetic rigidity as a variable, then the maps of all of the elements can be calculated with a variable magnetic rigidity allowing the effects of the different particles to be carried through the system.

CHAPTER 4

BENCHMARKING

Now that we know what we want to model, and have developed the framework with which to model it, we need to determine the best ways to use these tools to model the system. For example, in Section 2.5 we mentioned that the kicks would have to be performed in such a manner that the magnetic field in each section can be considered autonomous; we therefore need to find the best ways to use each of our tools, both for accuracy and for speed.

First, in Section 4.1, we will examine how to get the greatest accuracy out of the method we use to model the earth's magnetic field. In Section 4.2 we benchmark the moment method, and in Section 4.3 we benchmark the fast multipole method.

4.1 Earth's Magnetic Field Benchmarking

The effects of the earth's magnetic field need to be included in any accurate model of the University of Maryland Electron Ring. The model used the measured values for the earth's magnetic field; as shown in Figure 3.2, the vertical component is the largest one and matters most to the dynamics. These values were measured at the locations of each of the dipoles using a magnetometer. Since each element will include the effects of the earth's magnetic field, it was necessary to find the value of the fields at each point where we would like to have a kick. We used linear interpolation to find the values in between the measured points. The number of kicks required was measured by sending a particle through the map and determining how many kicks it took such that the location of the particle at the end was constant out to the number of significant figures that were available for the earth's field. It was found that dipoles and drifts require six kicks and that quadrupoles require 15 kicks. An example of the effects of the earth's field on a test trajectory is shown in Figure 3.3.

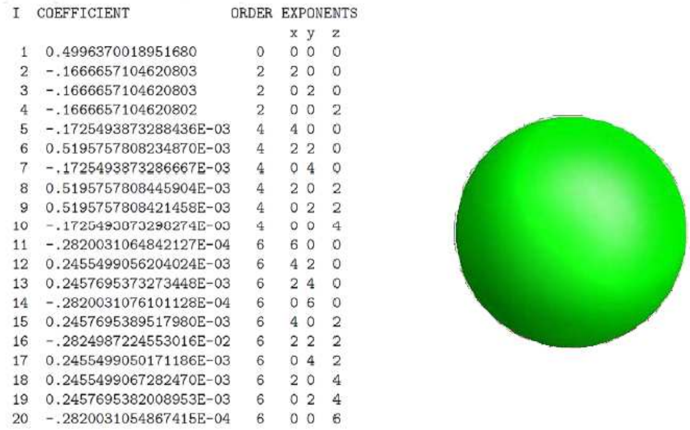
4.2 Moment Method Benchmarking

Now that the moment method has been implemented it is necessary to compare it with outside calculations to make sure that it is operating properly, and that the results it produces correspond to those predicted by other theories. It is also necessary to determine how big the moments and integration order have to be.

4.2.1 Comparison with Theory

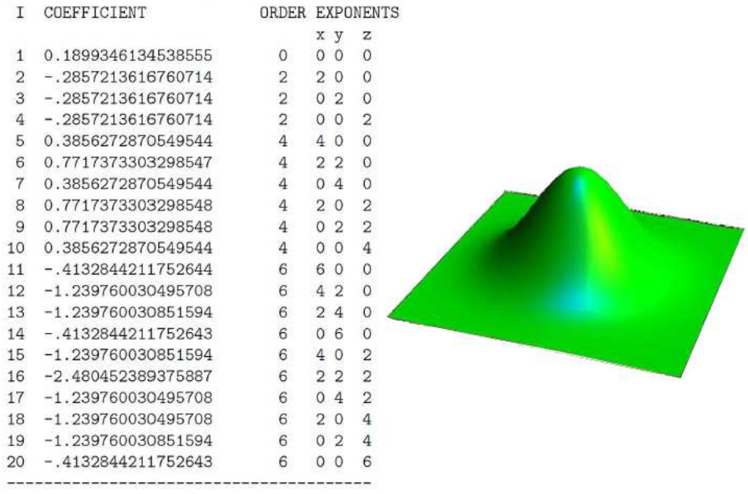
One method of benchmarking involves comparing the results from this method to an outside example with an analytic solution. The moment method is comprised of a number of parts which we need to test one at a time. First we need to look at how the Duffy transformation works with respect to distributions with known quantities; these are shown in Figures 4.1 and 4.2.

Next we will compare how the potentials of an easily integrated distribution compare using the various methods we have available; this can be seen in Figure 4.3.



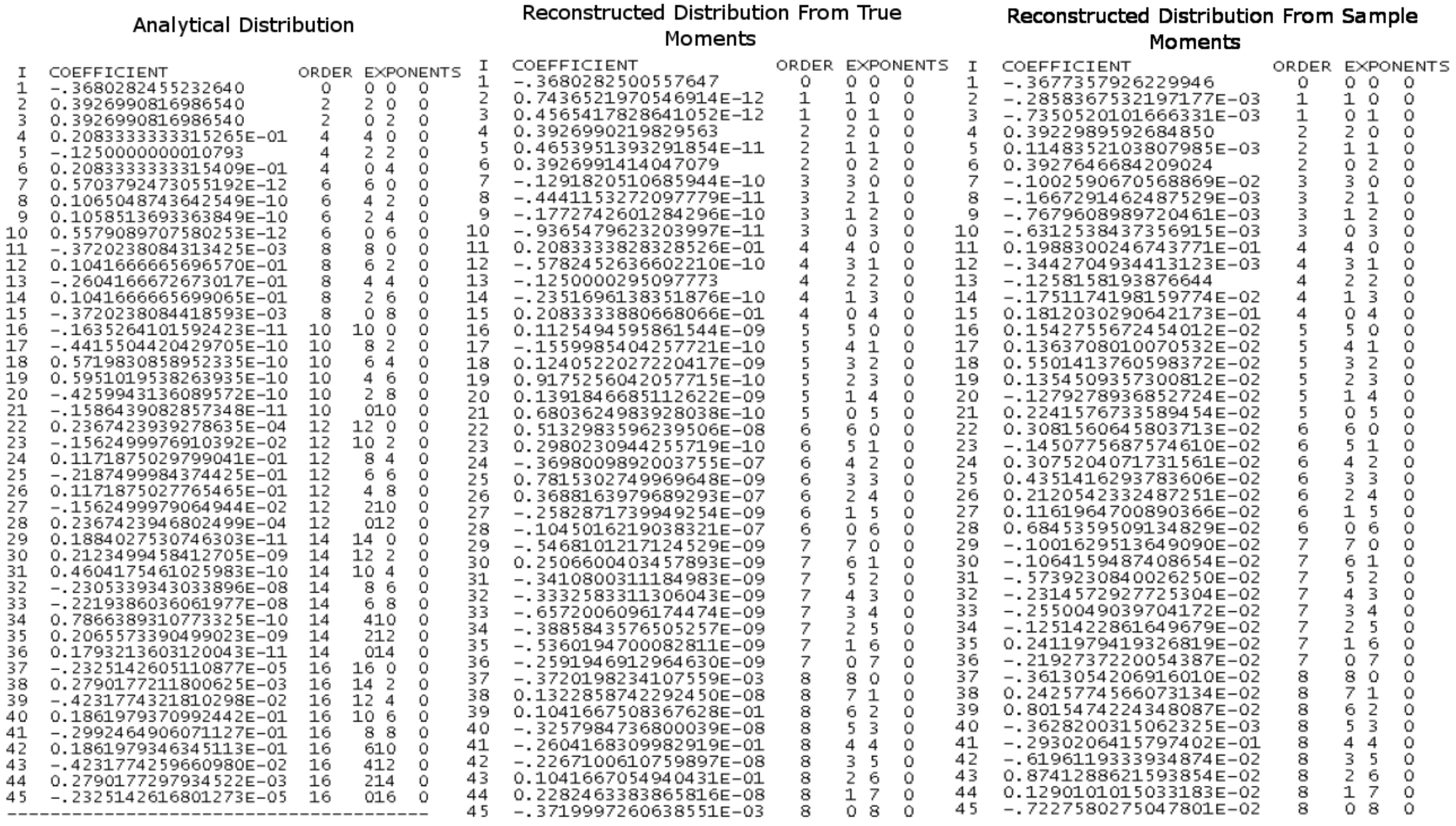
```
 I  COEFFICIENT             ORDER EXPONENTS
                                   x y  z
 1  0.4996370018951680        0    0 0  0
 2  -.1666657104620803        2    2 0  0
 3  -.1666657104620803        2    0 2  0
 4  -.1666657104620802        2    0 0  2
 5  -.1725493873288436E-03    4    4 0  0
 6  0.5195757808234870E-03    4    2 2  0
 7  -.1725493873286667E-03    4    0 4  0
 8  0.5195757808445904E-03    4    2 0  2
 9  0.5195757808421458E-03    4    0 2  2
10  -.1725493873290274E-03    4    0 0  4
11  -.2820031064842127E-04    6    6 0  0
12  0.2455499056204024E-03    6    4 2  0
13  0.2457695373273448E-03    6    2 4  0
14  -.2820031076101128E-04    6    0 6  0
15  0.2457695389517980E-03    6    4 0  2
16  -.2824987224553016E-02    6    2 2  2
17  0.2455499050171186E-03    6    0 4  2
18  0.2455499067282470E-03    6    2 0  4
19  0.2457695382008953E-03    6    0 2  4
20  -.2820031054867415E-04    6    0 0  6
```

Figure 4.1: This shows the results of a direct DA integration of a uniform spherical distribution using the Duffy transformation. We use an analytic distribution function given by $\rho(x, y, z) = (1 - \tanh(20(x^2 + y^2 + z^2 - 1)))/2$.



```
 I  COEFFICIENT             ORDER EXPONENTS
                                   x y  z
 1  0.1899346134538555        0    0 0  0
 2  -.2857213616760714        2    2 0  0
 3  -.2857213616760714        2    0 2  0
 4  -.2857213616760714        2    0 0  2
 5  0.3856272870549544        4    4 0  0
 6  0.7717373303298547        4    2 2  0
 7  0.3856272870549544        4    0 4  0
 8  0.7717373303298548        4    2 0  2
 9  0.7717373303298548        4    0 2  2
10  0.3856272870549544        4    0 0  4
11  -.4132844211752644        6    6 0  0
12  -1.239760030495708        6    4 2  0
13  -1.239760030851594        6    2 4  0
14  -.4132844211752643        6    0 6  0
15  -1.239760030851594        6    4 0  2
16  -2.480452389375887        6    2 2  2
17  -1.239760030495708        6    0 4  2
18  -1.239760030495708        6    2 0  4
19  -1.239760030851594        6    0 2  4
20  -.4132844211752643        6    0 0  6
---------------------------------------
```

Figure 4.2: This the potential of a directly integrated Gaussian distribution in 3 dimensions using the Duffy transformation. This used the analytic distribution function $\rho(x, y, z) = (27/(2\pi)^{\frac{3}{2}}) \exp(-(9/2)(x^2 + y^2 + z^2))$.



Analytical Distribution

```
  I   COEFFICIENT              ORDER  EXPONENTS
  1   -.3680282455232640         0    0 0   0
  2   0.3926990816986540         2    2 0   0
  3   0.3926990816986540         2    0 2   0
  4   0.2083333333315265E-01     4    4 0   0
  5   -.1250000000010793         4    2 2   0
  6   0.2083333333315409E-01     4    0 4   0
  7   0.5703792473055192E-12     6    6 0   0
  8   0.1065048743642549E-10     6    4 2   0
  9   0.1058513693363849E-10     6    2 4   0
 10   0.5579089707580253E-12     6    0 6   0
 11   -.3720238084313425E-03     8    8 0   0
 12   0.1041666665696570E-01     8    6 2   0
 13   -.2604166672673017E-01     8    4 4   0
 14   0.1041666665699065E-01     8    2 6   0
 15   -.3720238084418593E-03     8    0 8   0
 16   -.1635264101592423E-11    10   10 0   0
 17   -.4415504420429705E-10    10    8 2   0
 18   0.5719830858952335E-10    10    6 4   0
 19   0.5951019538263935E-10    10    4 6   0
 20   -.4259943136089572E-10    10    2 8   0
 21   -.1586439082857348E-11    10    010   0
 22   0.2367423939278635E-04    12   12 0   0
 23   -.1562499976910392E-02    12   10 2   0
 24   0.1171875029799041E-01    12    8 4   0
 25   -.2187499984374425E-01    12    6 6   0
 26   0.1171875027765465E-01    12    4 8   0
 27   -.1562499979064944E-02    12    210   0
 28   0.2367423946802499E-04    12    012   0
 29   0.1884027530746303E-11    14   14 0   0
 30   0.2123499458412705E-09    14   12 2   0
 31   0.4604175461025983E-10    14   10 4   0
 32   -.2305339343033896E-08    14    8 6   0
 33   -.2219386036061977E-08    14    6 8   0
 34   0.7866389310773325E-10    14    410   0
 35   0.2065573390499023E-09    14    212   0
 36   0.1793213603120043E-11    14    014   0
 37   -.2325142605110877E-05    16   16 0   0
 38   0.2790177211800625E-03    16   14 2   0
 39   -.4231774321810298E-02    16   12 4   0
 40   0.1861979370992442E-01    16   10 6   0
 41   -.2992464906071127E-01    16    8 8   0
 42   0.1861979346345113E-01    16    610   0
 43   -.4231774259660980E-02    16    412   0
 44   0.2790177297934522E-03    16    214   0
 45   -.2325142616801273E-05    16    016   0
-----------------------------------------------
```

Reconstructed Distribution From True Moments

```
  I   COEFFICIENT              ORDER  EXPONENTS
  1   -.3680282500557647         0    0 0   0
  2   0.7436521970546914E-12     1    1 0   0
  3   0.4565417828641052E-12     1    0 1   0
  4   0.3926990219829563         2    2 0   0
  5   0.4653951393291854E-11     2    1 1   0
  6   0.3926991414047079         2    0 2   0
  7   -.1291820510685944E-10     3    3 0   0
  8   -.4441153272097779E-11     3    2 1   0
  9   -.1772742601284296E-10     3    1 2   0
 10   -.9365479623203997E-11     3    0 3   0
 11   0.2083333828328526E-01     4    4 0   0
 12   -.5782452636602210E-10     4    3 1   0
 13   -.1250000295097773         4    2 2   0
 14   -.2351696138351876E-10     4    1 3   0
 15   0.2083333880668066E-01     4    0 4   0
 16   0.1125494595861544E-09     5    5 0   0
 17   -.1559985404257721E-10     5    4 1   0
 18   0.1240522027220417E-09     5    3 2   0
 19   0.9175256042057715E-10     5    2 3   0
 20   0.1391846685112622E-09     5    1 4   0
 21   0.6803624983928038E-10     5    0 5   0
 22   0.5132983596239506E-08     6    6 0   0
 23   0.2980230944255719E-10     6    5 1   0
 24   -.3698009892003755E-07     6    4 2   0
 25   0.7815302749969648E-09     6    3 3   0
 26   0.3688163979689293E-07     6    2 4   0
 27   -.2582871739949254E-09     6    1 5   0
 28   -.1045016219038321E-07     6    0 6   0
 29   -.5468101217124529E-09     7    7 0   0
 30   0.2506600403457893E-09     7    6 1   0
 31   -.3410800311184983E-09     7    5 2   0
 32   -.3332583311306043E-09     7    4 3   0
 33   -.6572006096174474E-09     7    3 4   0
 34   -.3885843576505257E-09     7    2 5   0
 35   -.5360194700082811E-09     7    1 6   0
 36   -.2591946912964630E-09     7    0 7   0
 37   -.3720198234107559E-03     8    8 0   0
 38   0.1322858742292450E-08     8    7 1   0
 39   0.1041667508367628E-01     8    6 2   0
 40   -.3257984736800039E-08     8    5 3   0
 41   -.2604168309982919E-01     8    4 4   0
 42   -.2267100610759897E-08     8    3 5   0
 43   0.1041667054940431E-01     8    2 6   0
 44   0.2282463383865816E-08     8    1 7   0
 45   -.3719997260638551E-03     8    0 8   0
```

Reconstructed Distribution From Sample Moments

```
  I   COEFFICIENT              ORDER  EXPONENTS
  1   -.3677357926229946         0    0 0   0
  2   -.2858367532197177E-03     1    1 0   0
  3   -.7350520101666331E-03     1    0 1   0
  4   0.3922989592684850         2    2 0   0
  5   0.1148352103807985E-03     2    1 1   0
  6   0.3927646684209024         2    0 2   0
  7   -.1002590670568869E-02     3    3 0   0
  8   -.1667291462487529E-03     3    2 1   0
  9   -.7679608989720461E-03     3    1 2   0
 10   -.6312538437356915E-03     3    0 3   0
 11   0.1988300246743771E-01     4    4 0   0
 12   -.3442704934413123E-03     4    3 1   0
 13   -.1258158193876644         4    2 2   0
 14   -.1751174198159774E-02     4    1 3   0
 15   0.1812030290642173E-01     4    0 4   0
 16   0.1542755672454012E-02     5    5 0   0
 17   0.1363708010070532E-02     5    4 1   0
 18   0.5501413760598372E-02     5    3 2   0
 19   0.1354509357300812E-02     5    2 3   0
 20   -.1279278936852724E-02     5    1 4   0
 21   0.2241576733589454E-02     5    0 5   0
 22   0.3081560645803713E-02     6    6 0   0
 23   -.1450775687574610E-02     6    5 1   0
 24   0.3075204071731561E-02     6    4 2   0
 25   0.4351416293783606E-02     6    3 3   0
 26   0.2120542332487251E-02     6    2 4   0
 27   0.1161964700890366E-02     6    1 5   0
 28   0.6845359509134829E-02     6    0 6   0
 29   -.1001629513649090E-02     7    7 0   0
 30   -.1064159487408654E-02     7    6 1   0
 31   -.5739230840026250E-02     7    5 2   0
 32   -.2314572927725304E-02     7    4 3   0
 33   -.2550049039704172E-02     7    3 4   0
 34   -.1251422861649679E-02     7    2 5   0
 35   0.2411979419326819E-02     7    1 6   0
 36   -.2192737220054387E-02     7    0 7   0
 37   -.3613054206916010E-02     8    8 0   0
 38   0.2425774566073134E-02     8    7 1   0
 39   0.8015474224348087E-02     8    6 2   0
 40   -.3628200315062325E-03     8    5 3   0
 41   -.2930206415797402E-01     8    4 4   0
 42   -.6196119333934874E-02     8    3 5   0
 43   0.8741288621593854E-02     8    2 6   0
 44   0.1290101015033183E-02     8    1 7   0
 45   -.7227580275047801E-02     8    0 8   0
```

Figure 4.3: Comparison of various expansions of the potential of a uniform square of charge distribution. The left column shows the Duffy integration of the exact distribution. The middle column uses the moment method with the moments calculated exactly from the analytical distribution, truncated at order 16. The column on the right shows the potential expansion obtained from sample moments using one million test particles sampled from the exact distribution function, truncated at the same order.

4.2.1.1 Expansion Method

It was mentioned in Sections 3.2.1.1 and 3.2.1.2 that there were two types of smooth functions that could be used to expand the potential: a sum of monomials, and a sum of orthogonal polynomials. For this purpose we used Legendre polynomials, but there are other kinds which are being examined for possible use at this time. The two objects of interest are execution time and accuracy, with accuracy being more important. The first comparison is between the potentials calculated using analytical distribution functions with a numerical integrator to determine the potential expansion; the comparison between the monomial and Legendre polynomial expansions is shown in Figure 4.4. As can be seen in this comparison the Legendre polynomial version is more accurate at higher order.

The purpose of this method is to determine the density function and hence the potential for a system of test particles, not analytic distributions. We therefore compared the two using a set of test particles, the results of which can be seen in Figure 4.5. As can be seen there is a distinct difference, and a drop off in quality at high order when comparing the monomials to the Legendre polynomials.

The use of particles presents an inherent graininess to the distribution, which can be thought of as an error in the right-hand side of two systems of linear equations. The error in the system is proportional to the condition number, κ, of the matrix [48],

$$\vec{C} = N\vec{M}, \tag{4.1}$$

$$\frac{\parallel \delta\vec{C} \parallel}{\parallel \vec{C} \parallel} \leq \kappa(N)\frac{\parallel \delta\vec{M} \parallel}{\parallel \vec{M} \parallel}. \tag{4.2}$$

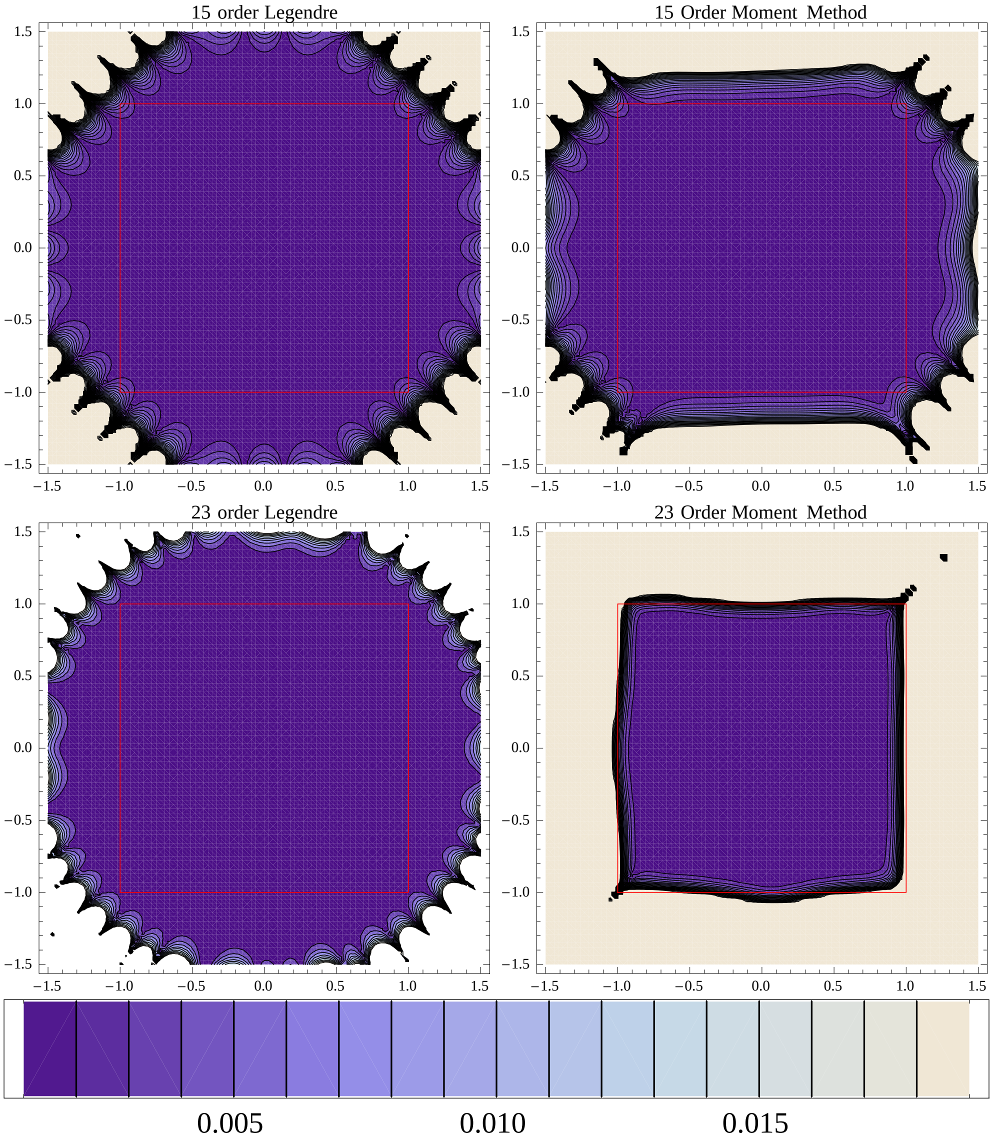


Figure 4.4: Comparison of the exact moment method on the right with the Legendre method on the left at 15^{th} and 23^{rd} order. These simulations used a uniform distribution with the red square denoting the region of integration. The potential was found both analytically using direct expansion around each point, and then it was determined using either the moment or Legendre methods to the indicated order, with the absolute difference indicated on the contour plot.

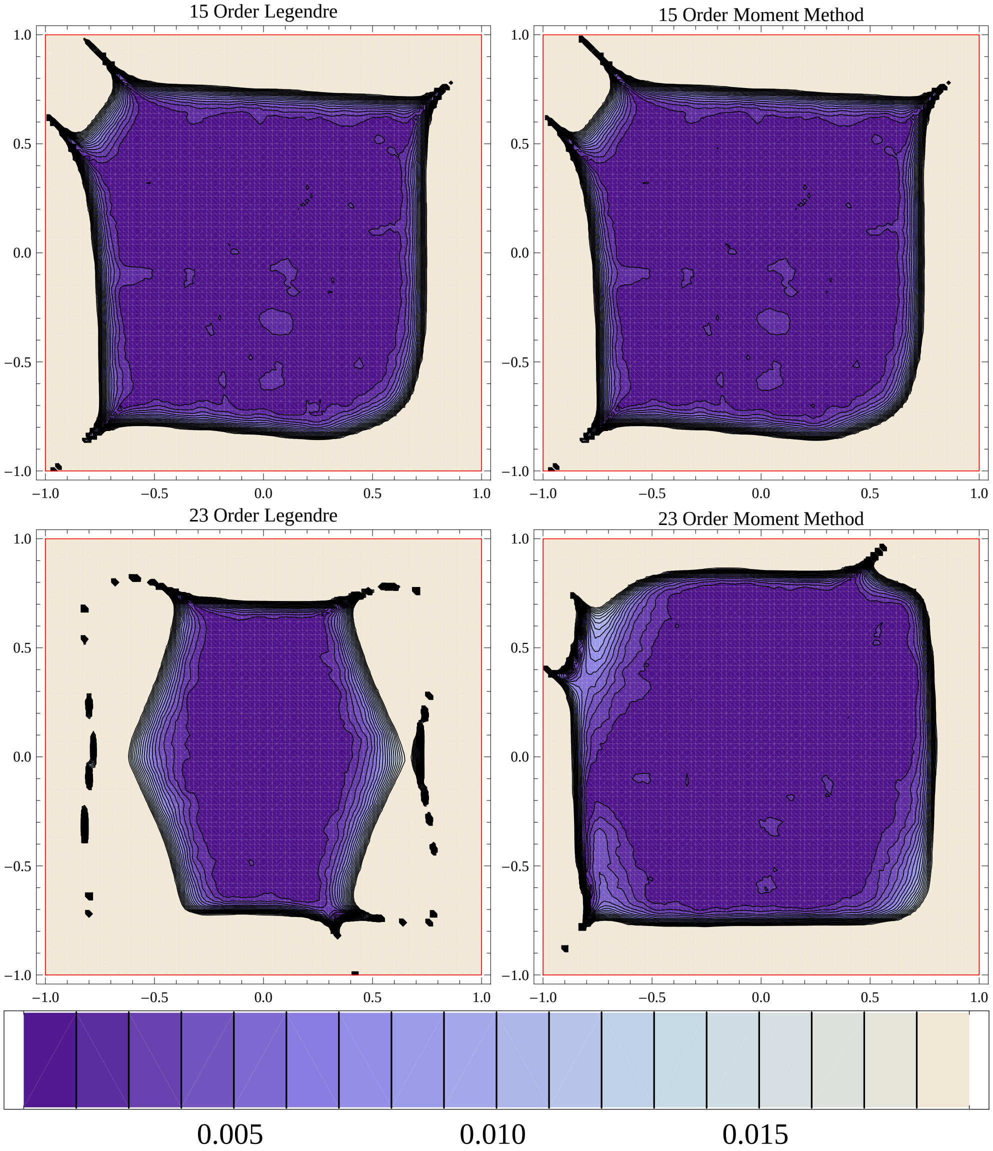


Figure 4.5: Comparison of the sample moment method on the right with the Legendre method on the left at 15^{th} and 23^{rd} order. These simulations used a uniform distribution of particles with one million test particles. The potential was found both analytically using direct expansion around each point, and then it was determined using either the moment or Legendre methods to the indicated order, with the absolute difference indicated on the contour plot.

Since the moment method uses truncated SVD methods to invert its matrix, it reduces the condition number, thus making the moment method more accurate for particle systems at high moment orders. Combined with its higher speed, the moment method is the method of choice for the extensive simulations.

4.2.1.2 Particle Advancement

Now we wish to put the potential, and by extension the electric fields, to the test by advancing a set of test particles through a system. The example shown involves tracking a 100 KeV laminar electron beam with an initial radius of 5mm and a current of 1 amp. This beam is propagated over a distance of 20cm and approximated using 1×10^6 test particles. This has an analytic solution for the final beamsize as a multiple of the initial beamsize given by [49],

$$\frac{r_m}{r_0} = 1 + 5.87 \times 10^{-5} \frac{I}{(\gamma^2 - 1)^{\frac{3}{2}}} (\frac{z}{r_0})^2, \tag{4.3}$$

which predicts a beam size growth of 1.33, or 33%. When these conditions are applied using the moment method, a method for determining what the initial and final beam size is needs to be ascertained. The two methods employed involved placing test particles on the edge, and comparing their initial and final positions as well as looking at the horizontal and vertical linear map elements. A comparison of the initial to final distribution can be seen in Figure 4.6 and a comparison of the final results can be seen in Table 4.1.

As can be seen, the two methods neatly bracket the analytic solution. A slightly higher credence should be given to the point comparison. First, by looking at the

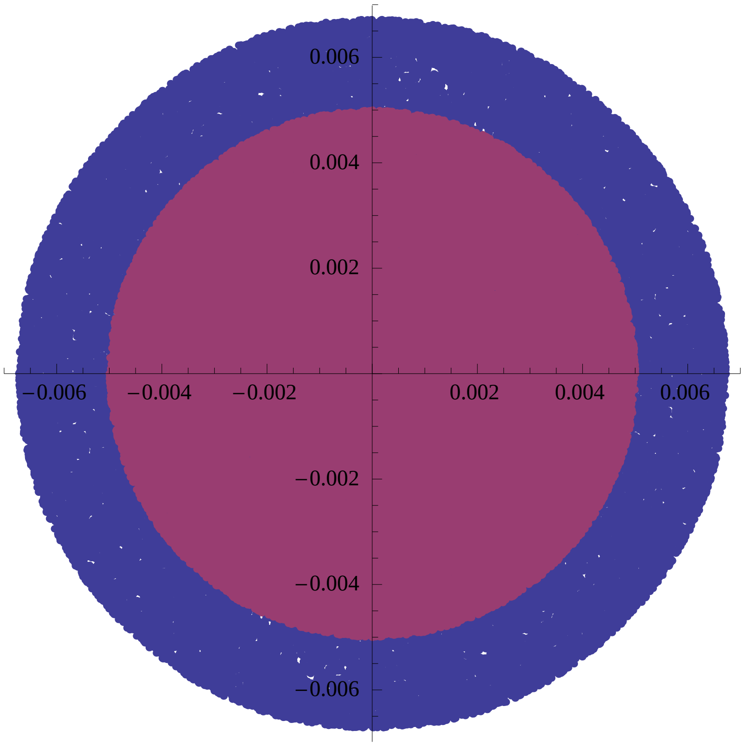


Figure 4.6: A comparison of the initial and final horizontal and vertical test particle positions in the 20cm drift. Purple indicates the initial points, blue the final.

Table 4.1: Table of different methods for finding the increase in size from beginning to end in the described problem.

Method	Growth
Edge Point x	35.27 %
Edge Point y	35.30 %
Map Element x	31.21 %
Map Element y	31.34 %

figure it appears to be accurate, and second, since the map itself is nonlinear, the linear part will not be a complete representation of the size.

4.2.2 Accuracy

What order we calculate the moments at is an important consideration. Since the moment calculation is one of the slower parts of the method, the order at which a given accuracy is desired should be minimized. Also since the number of particles needed also causes issues, how quickly a series converges will be an issue that needs to be investigated.

Accuracy is our foremost concern. This was tested by first creating a set of initial conditions of uniform density with a circular cross section. We then set about determining a radial baseline for comparison by calculating the potential using strict Coulomb interactions. This was done by calculating the potential at a set radius at 10 equally spaced points around the center of expansion. The radius is then increased in 100 equal increments out to 1.5 times the size of the integration region. This process is repeated for each potential calculated using the different integration and moment orders. The average differences between the Coulomb potential and the moment calculated potential are summed up to a desired radius. These are then averaged and used as the error magnitude in Figure 4.7.

As can be seen in Figure 4.7, the accuracy only converges when the integration order is greater than or equal to the moment order. This follows from the fact that the values for each of the Taylor coefficients in the potential are dependent on what order the moments are, and if the moments are larger than the integration order, the higher order terms will not be included. We would assume that there

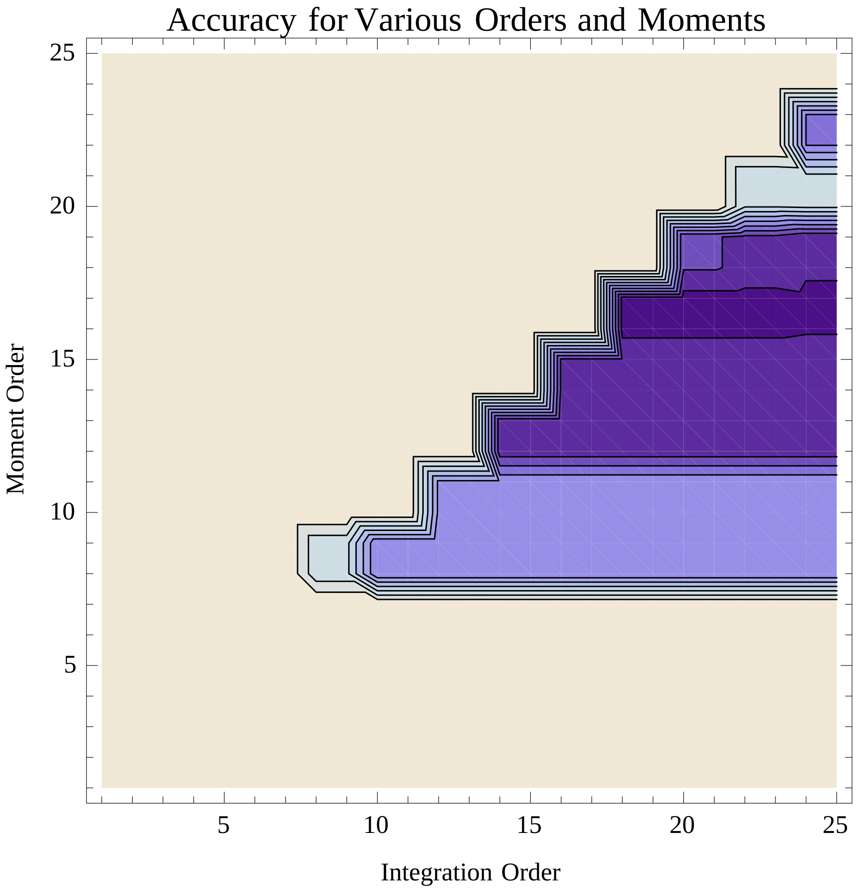


Figure 4.7: This calculated the potential of a uniform circular distribution for different integration and moment orders. This was done by creating a grid of the orders: the darker the area, the higher the accuracy. The contours show the region between zero and .002.

would be increasing accuracy with increasing order, but when we examine the data there appears to be a drop-off in accuracy in the high teens. This is caused by the ill-conditioned nature of the matrix that gets inverted; in the high teens and low twenties the SVD algorithm can no longer fully cast the matrix into something that is well conditioned. This gives us a small band of moments in which we get optimum performance.

4.2.3 Kick Number

The method updates the shape of the beam each time the potential is calculated, so the more kicks per element the more accurate the method is. However, this is time consuming. An examination was made by comparing the analytical results from Section 4.2.1 for different numbers of kicks, with saturation for distances per kick equal to 3 times the initial radius. An example of this effect is shown in Figure 4.8.

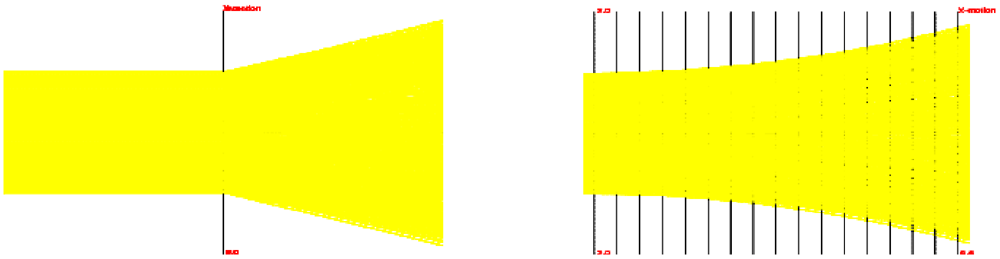
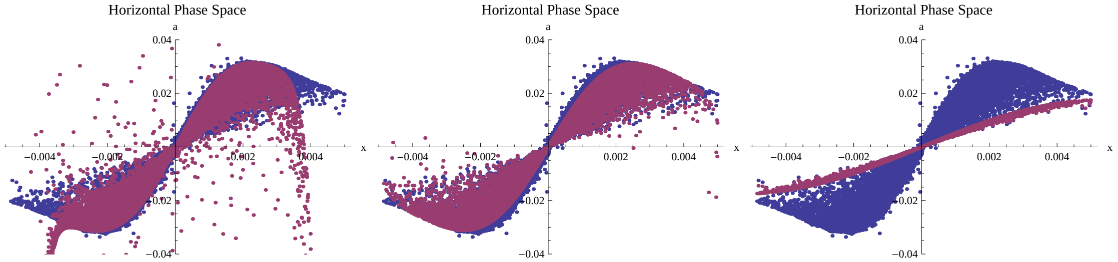


Figure 4.8: This is a demonstration of a drift with the effects of space charge added to the map with one kick shown on the left, and 17 kicks shown on the right.

4.2.4 Integration Region

The region over which the transformation matrix is calculated has a strong influence on what the final answer will be. Since we are using a Taylor series to approximate the distribution, it is necessary to choose the region being studied carefully. The series must converge within that region fairly well, so bounds must contain the points needed and not too much empty space. In distributions such as KV where there is a definite edge the difference is not as strong, but in continuous distributions such as Gaussian this can cause complications. Examples of this sort of behavior are shown in Figure 4.9.


Horizontal Phase Space

Horizontal Phase Space

Horizontal Phase Space

Figure 4.9: In all figures the blue represents the expected kick for each particle while the purple shows the moment method calculated one. The left picture shows integration bounds that are too small, the center shows good bounds, the rightmost shows bounds that are too large.

In the example where the region of integration is too small, many particles enter the region where the Taylor series diverges, and they are thrown away or pulled in in a non-physical manner. If the region chosen is too large, however, the higher order terms are washed away and a very simple model emerges. Unfortunately the size which gives good values changes based on the distribution type, with a side-by-side comparison between multiple distributions shown in Figure 4.10. For the purposes of the implementation we use 5.25 standard deviations.

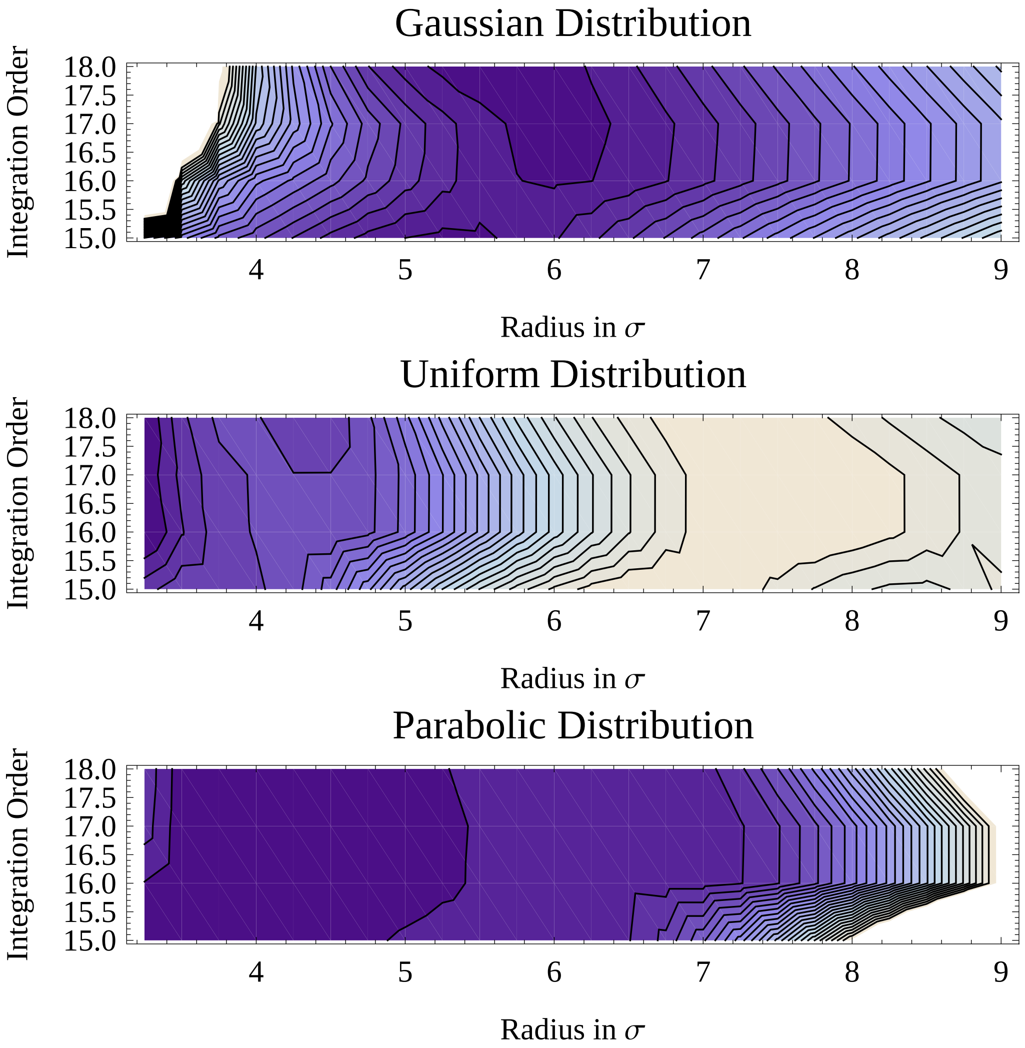


Figure 4.10: These plots show the accuracy of the potential for integration radii as measured in standard deviations for three different distributions.

4.2.5 Timing

The timing of this method is influenced by a number of factors. Due to COSY Infinity's original design as a single-particle code there are some limitations to the manner in which particles can be tracked. At the Fortran level there is a hard ceiling to the number of particles that can be stored, which is currently 50,000. However, we would like to be able to store larger numbers of particles than that, so a file-based tracking system was developed that reads in the particles in batches of 1,000, applies any transformations that are needed to them, and then writes them to a different file. As can be seen in the single CPU comparisons in Figure 4.11, there is a large difference between the way the times scale when using the file-based system and the vector-based system; one scales linearly, the other scales quadratically.

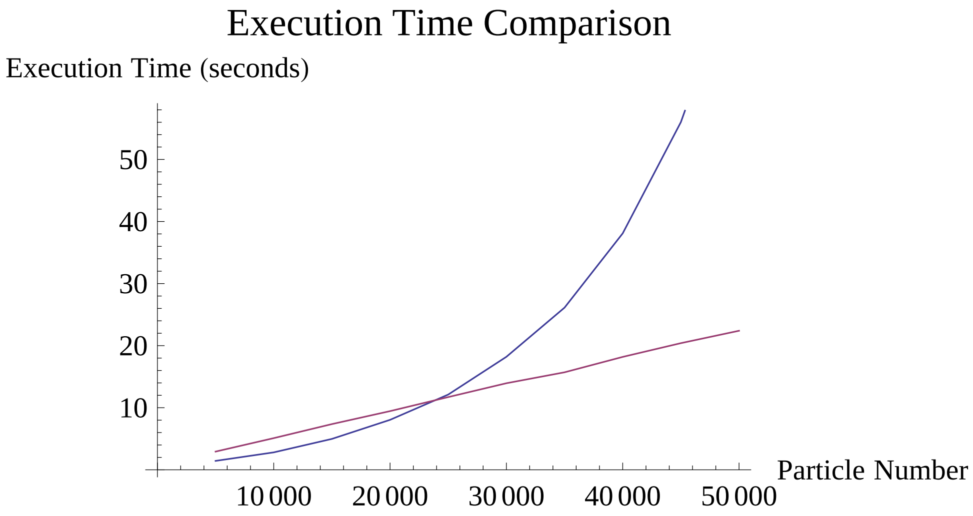


Figure 4.11: The blue line shows the execution time using the vector system, while the purple line uses the file system.

4.3 Fast Multipole Method Benchmarking

Another set of studies was done to examine how accurately the fast multipole method agrees with theory. Some of the tests shown for the moment method will not work in exactly the same manner since there is no analytic counterpart the way there is with the moment method, so all benchmarking has to be performed using particles.

4.3.1 Comparison with Theory

Since we have to use only test particles for our comparisons, the simplest check will be to compare the potential calculated using the fast multipole method to the potential calculated using point-to-point Coulomb potentials. The results for three different potentials are shown in Figure 4.12.

A series of tests similar to 4.6 was performed for the fast multipole method, and it too had a respectable degree of accuracy, as is shown in Figure 4.13.

4.3.2 Accuracy

We wanted to determine what expansion order and what number of boxes would give us the optimum accuracy. For this purpose we created a system of particles and performed a space charge kick on them using the fast multipole method at a variety of expansion orders and box numbers while creating a point-by-point comparison of the angle change for the same set of particles. The average difference between the

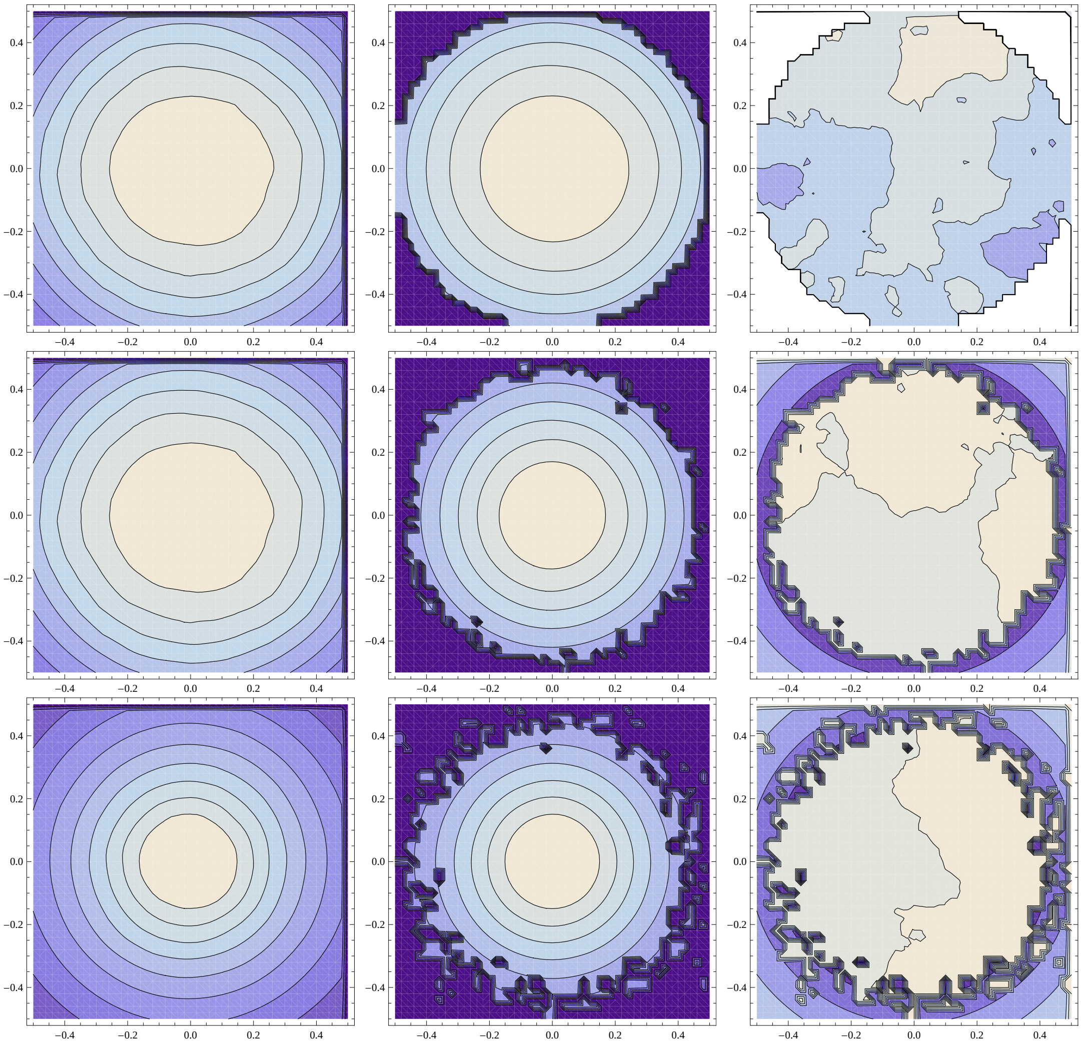


Figure 4.12: These graphs show a series of potentials determined for three different distributions, in descending order, uniform parabolic and Gaussian. The left column contains the point-to-point Coulomb potentials, the second column contains the FMM-determined potential, while the third contains the difference. The edge effects for the FMM-determined potential were caused by a time saving measure which doesn't create a local expansion for boxes that don't contain particles.

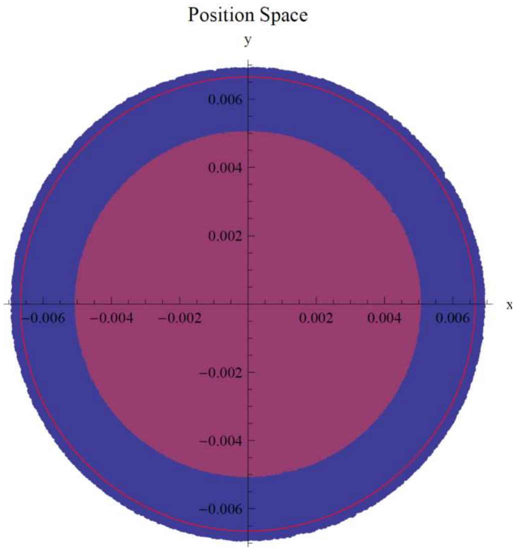


Figure 4.13: This plot shows the initial conditions (purple) and the final conditions (blue) of a set of particles under the same conditions used in Section 4.2.1. The red circle is the analytically determined final radius.

angular kicks for between the fast multipole method and the point-to-point method were used to judge the relative accuracy of the system. In Figure 4.14 we see that for the distribution examined, the accuracy becomes saturated at and above fifth order.

4.3.3 Timing

As can be seen in Figure 4.15, for the regions we are interested in there are noticeable time savings with the differential algebraic methods we have developed. The timing of the code has been measured using quantities such as number of particles, multipole order and number of boxes. As can be seen in Figure 4.16, the time required to advance the particles scales between N, N^2, and $N \log N$. The experiment shown in Figure 4.16 uses a fixed order and box number with a varying number of particles. The results show that for a small number of particles the timing

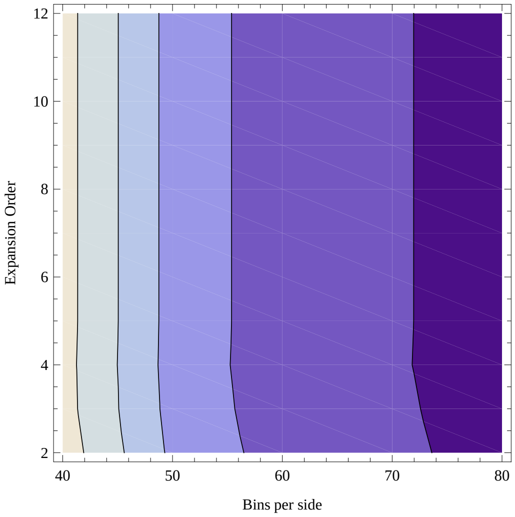


Figure 4.14: This is a plot of the accuracy of the fast multipole method with different numbers of boxes and different expansion orders. The order value becomes saturated at fifth order.

scales with N^2; this occurs when the number of particles in a given box are small, and it becomes more time consuming to make a local expansion than to simply expand each particle. At a certain point, in this case 20,000 particles, the expansion is no longer more trouble than it's worth and the rate of growth slows down, until as the number rises it becomes $N \log N$, asymptotically.

The DA methods involved are capable, within the larger COSY framework, of applying the fields of the given elements to particles with different masses or energies. This also carries through to the effects of space charge using the FMM.

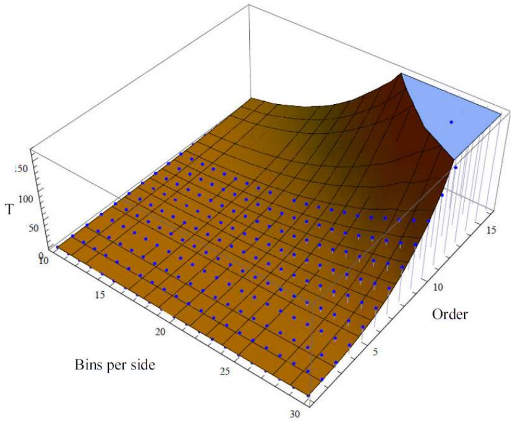


Figure 4.15: This graph shows a comparison of the amount of time needed to determine the potential of a distribution of 20000 particles. The surface shows the time taken using the differential algebraic method, while the points represent the time taken by the method in the section entitled FMM Basics. As can be seen, the DA method is faster through tenth order.

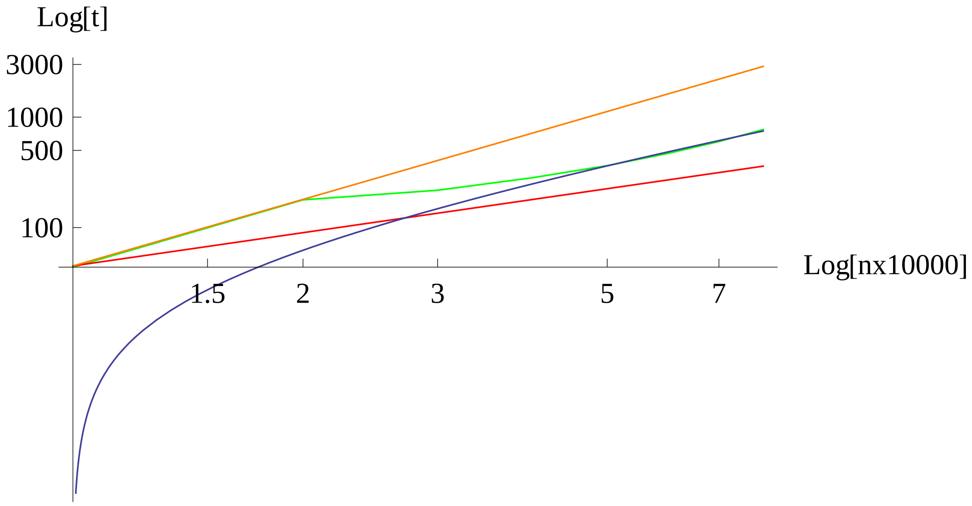


Figure 4.16: This log-log plot shows in green the measured time of a fixed order, fixed box number distribution as the number of particles are increased. The orange line represents N^2, while the red line represents N, with the blue line representing $N \log N$.

CHAPTER 5

APPLICATIONS

Chapter 3 developed a number of mathematical tools for simulating a beam with space charge, while Chapter 4 determined the operating points at which to use them. Now it is time to examine how these tools can be used to analyze a beam, and at some point design one. Some of these tools are the same kind that most beam physics codes have available to them, while others are unique to COSY, and still more require the tools that have been specifically developed here.

5.1 UMER Single-Particle Studies

Since UMER has a unique operating point and geometry, an in-depth study of the single-particle dynamics was necessary to fully understand any space charge effects that might become an issue. A simulation of the entire UMER ring from cathode through as many turns as desired was created for these studies; an overview of this model is examined in Appendix D [50].

The matching settings for the injection line were determined using a set of 5000 particles arranged for a given aperture that were sent through the solenoid, then matched to the ring's Twiss parameters as generated by COSY, and the steering dipoles were then fitted to move the beam through the injection line to the center of the beam pipe with zero angle. The two quadrupoles in the injection section keep the same values that were determined for the ring matching, but the pulsed dipole can be varied to steer the beam towards its closed orbit. The matching algorithm uses the linear elements of the map, and takes linear coupling into account. This was performed with COSY's internal optimization algorithms.

5.1.1 Betatron Tunes

The large quadrupoles in the Y-section were used to control the betatron tune. The tune that would result if instead of an injection section with 17 ring sections the ring was made of 18 identical sections (i.e., the optimal value for the fully symmetric layout) is called the ideal tune. We use the magnetic field, quadrupole and dipole settings of the other 17 sections combined with a similar set for the 18^{th} section. The ideal tunes were calculated to be $(\mu_1, \mu_2) = (6.76457, 6.63717)$. The differences in the quadrupole strengths greatly affect the steering of the beam since the beam enters them off-center. Therefore the tune matters not only as the operating point, but also as part of the steering. We see that the x-tune can be varied more according to this study than the y-tune.

5.1.2 Chromaticities

Using the DA normal form methods that are an integral part of COSY, it is very simple to calculate the chromaticities. The chromaticities are shown in Table 5.1. These compare the calculated values for four operating points that were the default settings used for four currents during March of 2009. The labels are historical in nature, as the beam current in the simulation only affects the image charge force, which has a very small effect. The reason these settings are different, is that they must be matched in a different manner which affects the steering through injection and ultimately the betatron tune. The addition of the higher order terms can help further distinguish operating points; if we compare the chromaticities for the pencil beam and the 7 mA beam, the pencil is only larger by 57%, whereas the next term is not only four times larger in magnitude but also opposite in sign. The beams were distinguished by peak current, with the pencil beam being the lowest at 0.6 mA. In the simulations, the chromaticities depend heavily on the quadrupoles in the Y-injection section, as that was the major change for the repetitive portions of the ring.

Table 5.1: Betatron tune predictions for four operating points. The first column contains the order of the term: 0^{th} order is the tune itself, 1^{st} is $\frac{d\mu}{d\delta}$, 2^{nd} is $\frac{d^2\mu}{d\delta^2}$, etc. The second column is the value predicted by COSY. The four operating points are labeled above the terms.

Pencil Beam		7 mA Beam	
Order	$\mu(\delta)$	Order	$\mu(\delta)$
0	.747	0	.63722
1	-7.1058	1	-4.4947
2	111.479	2	-25.83
3	-4787.3359	3	-969.6789
23 mA Beam		80 mA Beam	
Order	$\mu(\delta)$	Order	$\mu(\delta)$
0	.6796	0	.6900
1	-6.172	1	-5.0174
2	-11.2667	2	-31.2928
3	2034.3401	3	-1221.4241

5.1.3 Dispersion

Another quantity that needs to be understood is the maximum dispersion of the various operating points. Since the dipole settings within the ring remain the same for all of the operating points, it is the steering settings through the Y-section that contribute to the differences. It is also informative to investigate the manner in which the dispersion changes as the map is generated element by element; this behavior is shown in Figure 5.1. Furthermore, if the beam is measured at various places around the beam, the levels of dispersion will differ. The behavior of the one-turn dispersion element is plotted as a function of distance around the ring in Figure 5.2. There is a distinct repetitive pattern to the dispersions of the various maps as we move around the ring, with the range of values going through periods being small, and then quite large. A comparison of the manner in which this range of values is distributed with respect to tune space is shown in Figure 5.3. These four plots show that the morphology of the distribution in tune space is not a function of the range of variables, since the two on the left side share similar morphologies while having very different ranges of values.

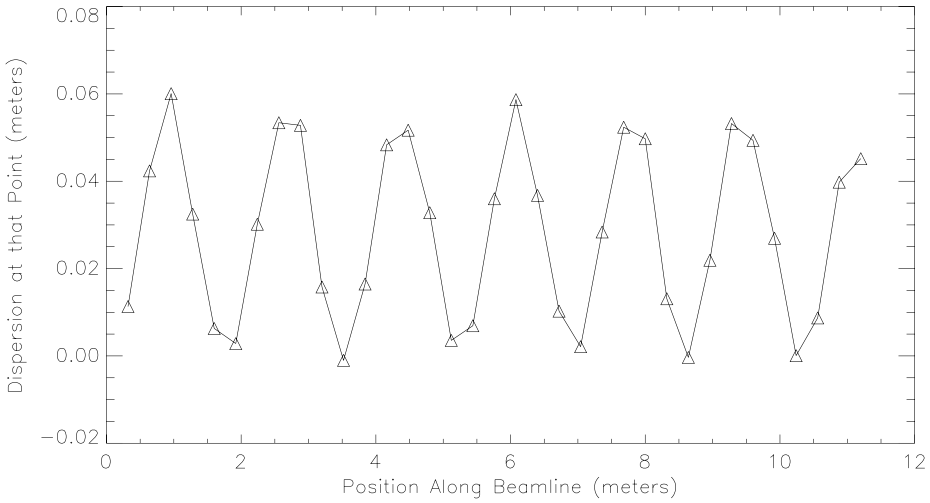


Figure 5.1: A plot of the progression of the dispersion as the one turn map is generated FODO section by FODO section.

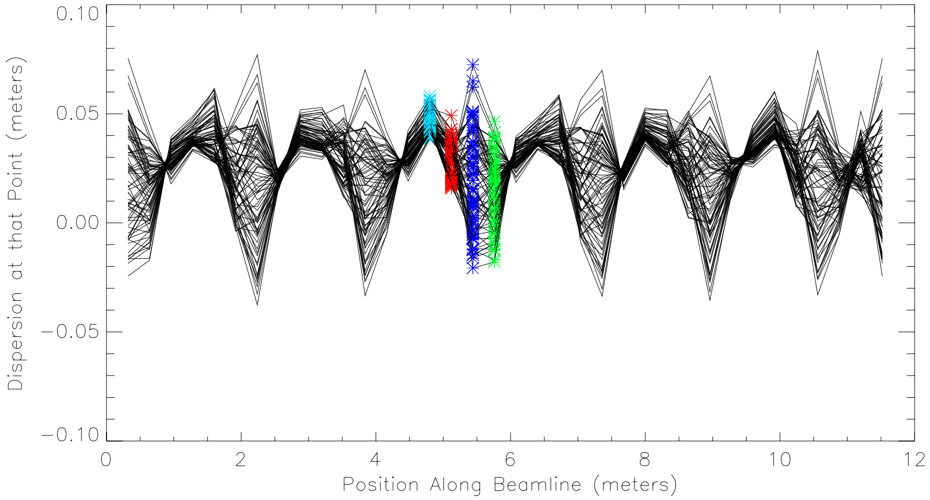


Figure 5.2: A plot showing the dispersion of the 81 operating points used in this study. All are superimposed to show the variability in the dispersion of the one-turn map as calculated around different points of the ring. The colors denote the positions of the contour plots shown in Figure 5.3 in order.

5.1.4 Amplitude-Dependent Tune Shifts

Using the normal form methods available in COSY it is possible to calculate amplitude-dependent tune shifts; these lead to footprints of the kind seen in Figure 5.4. Other operating points have different patterns, and some do not display this behavior altogether. Within the tune scan that was performed as part of this analysis it was the operating points with higher y-tunes that showed the fewest examples of amplitude-dependent tune shifts, and when they were calculated for default UMER settings, the ones detected were very small.

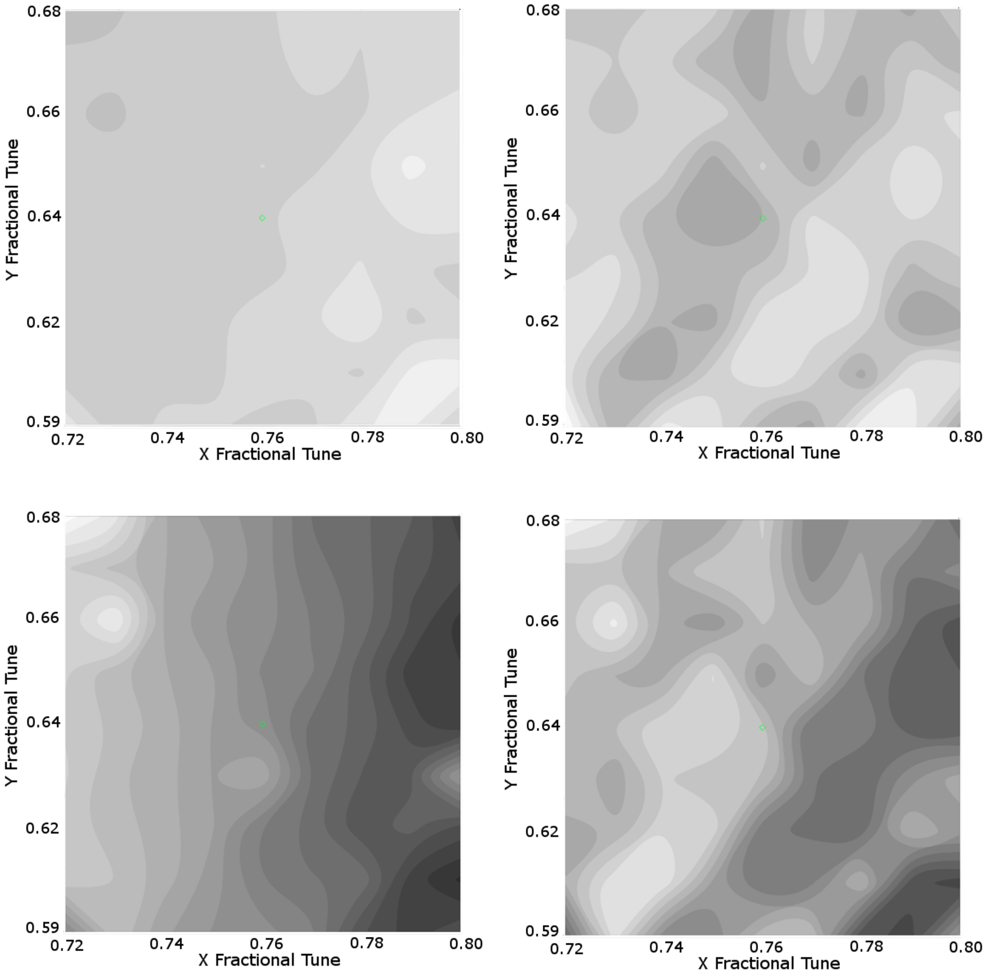


Figure 5.3: Four contour plots of the dispersion across the scanned tune space. Darker areas denote lower levels of dispersion. The values are contoured between −.02m and .08m. The locations for the upper left, upper right, lower left, and lower right are marked with the light blue, red, dark blue, and green points in Figure 5.2 respectively. The green ⋄ symbol represents the location of the ideal tunes.

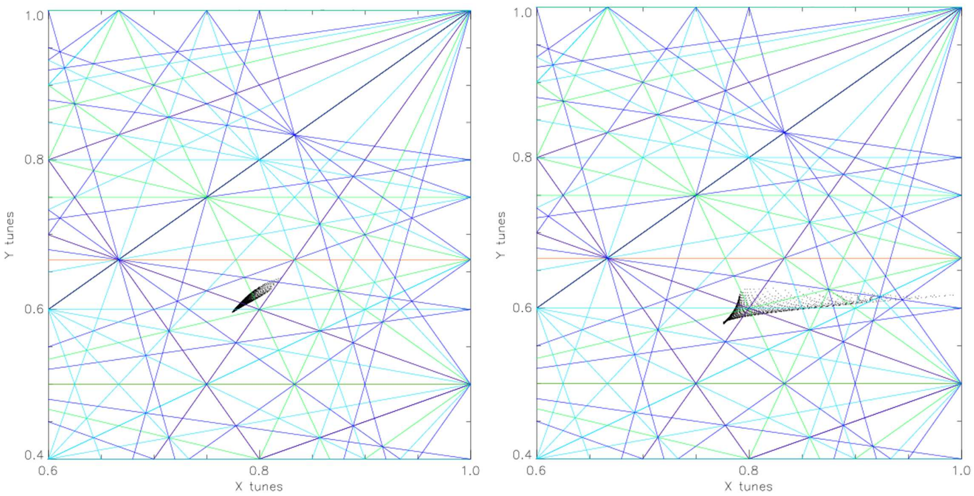


Figure 5.4: The left side shows a well behaved tune footprint, while the right side shows a more diffuse tune footprint. The differences are caused by different quadrupole settings. Superimposed are the resonance lines up to sixth order: red indicates 2nd order, orange indicates third order, green indicates fourth order, light blue indicates fifth order, and dark blue indicates sixth order.

5.1.5 Momentum Compaction

It is also possible to measure the momentum compaction of the operating points; the comparison is shown in Figure 5.5. The momentum compaction values range from $-.0006$ to $.0005$ and can be seen to be connected to both the x and y tunes.

5.1.6 Tracking

In order for the beam to be used for long-term studies, accurate steering both at injection and during recirculation is necessary. Steering through the injection line is accomplished using steering dipoles, whereas steering throughout the ring is accomplished by varying the current through the bending dipoles. Determining

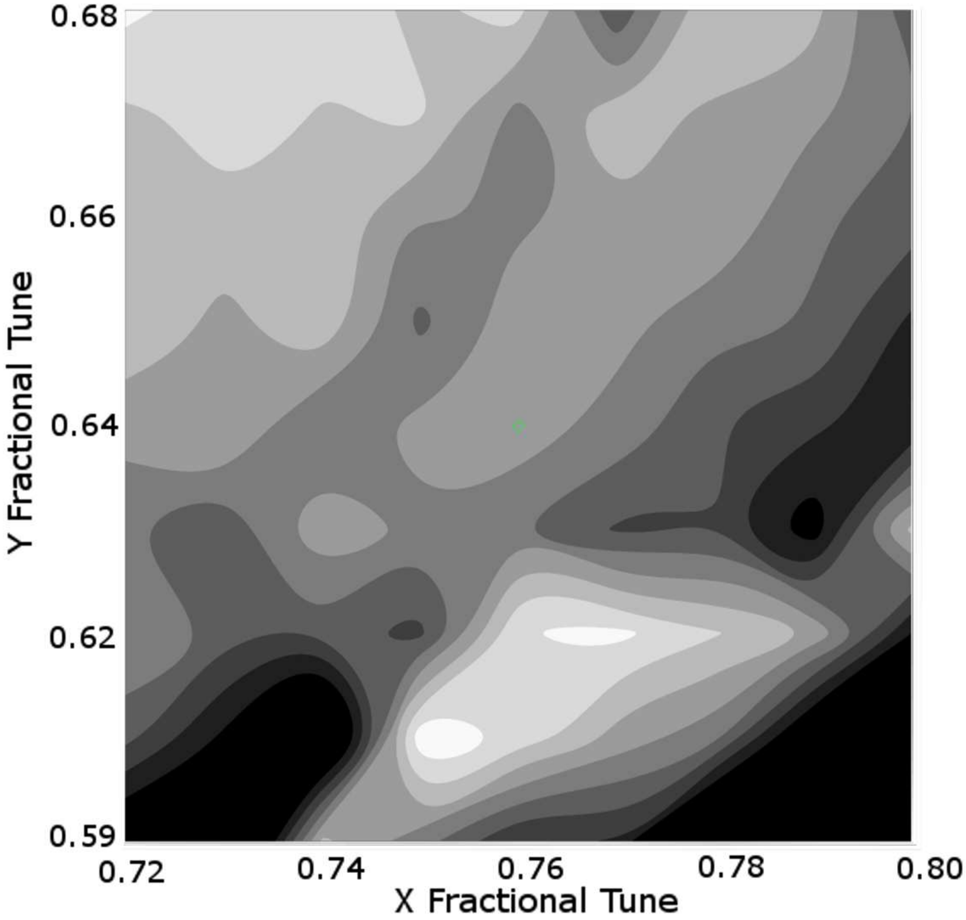


Figure 5.5: A contour plot of the momentum compaction across the scanned tune space. Darker areas denote a smaller compaction factor. The range of values is between −.0006 to .0005. The green ⋄ symbol represents the location of the ideal tunes.

a closed orbit that stays close to the centerline of the machine is important, and the earth's magnetic field makes this task much more difficult. The initial round of simulations involved steering through each two-dipole section by assuming that the beam would enter the section at the center of the beam pipe with zero lateral velocity, and the dipoles were changed such that the beam would exit the section at the center of the beam pipe also with zero lateral velocity. Also, the beam would enter and leave each section moving straight through the section without any angle with respect to the centerline of the beam pipe.

Previous steering solutions [51] for the ring had used the earth's magnetic field data to try to reduce the dipole currents by an amount equal to the equivalent of the earth's integrated field, which gave similar predictions as can be seen in Figure 5.6. From an operations standpoint, the relative smoothness or jaggedness of the current settings are inconsequential. Attempts that were made to smooth the setting (global, polynomial, running averages, etc.) deteriorated the steering of the beam. The ring steering as calculated with COSY assumed that the beam centroid would enter the first turn at the center of the pipe with zero angle, and a fitting algorithm was used to have it exit the turn at the center of the pipe with zero angle. Therefore the steering algorithm was designed such that the beam centroid would leave the electron gun aperture and enter the ring under those conditions. This involves both the steering dipoles and the matching quadrupoles working together to bring the beam into the ring. For recirculation the simulations used a combination of the bending dipole on the recirculation side of the ring, the recirculation steering dipole, and the pulsed dipole. All elements are at fixed locations. The purpose of this optimization was to make all perturbations as local as possible. An example of the closed orbit for one particular operating point is shown in Figure 5.7. In this figure it can be seen that with a good set of dipole corrections the deviation of the closed orbit from the center can be kept in the horizontal plane to the sub-millimeter scale. The number of vertical steering degrees of freedom made optimization of the vertical trajectory more challenging. One method that increased the speed at which a solution was found, was to add the position and angle terms to the objective function in quadrature. In the vertical direction the displacement is a few millimeters.

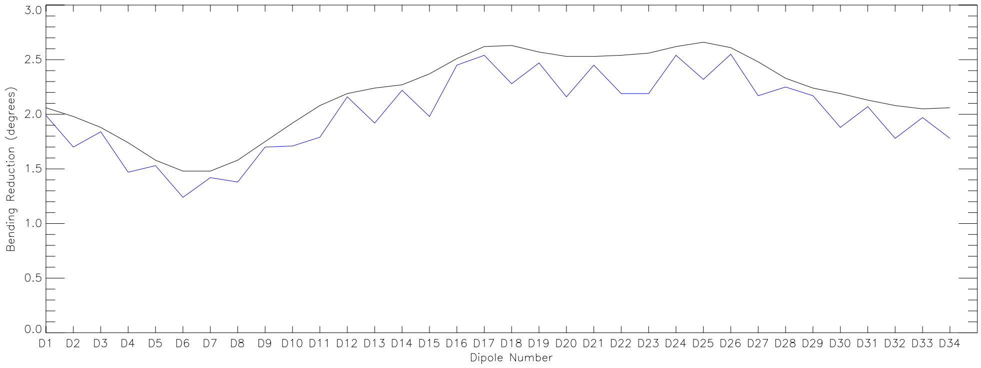


Figure 5.6: A comparison of values for the ring steering. In order to account for the earth's magnetic field, the ring-bending dipoles must be altered from their physical bending setting of 10 degrees. This is a plot of that change. The black line is the previous method of reducing the dipole currents by the amount equal to the equivalent of the integrated magnetic field over the section. The blue line is the result of having COSY Infinity steer the beam to the center of the pipe with zero angle at the end of each FODO section. We attempted to smooth out the settings but every attempt made the steering significantly worse.

5.1.7 Dynamic Aperture

A total of 81 values for the betatron tune, centered on the ideal tune, were investigated to determine how much beam remained within the beam pipe, a measure of the dynamic aperture, after a set number of turns. These values cover a square of tunespace with a magnitude of .08 per side. This method of measuring the dynamic aperture involved creating a set of initial conditions at the location of the

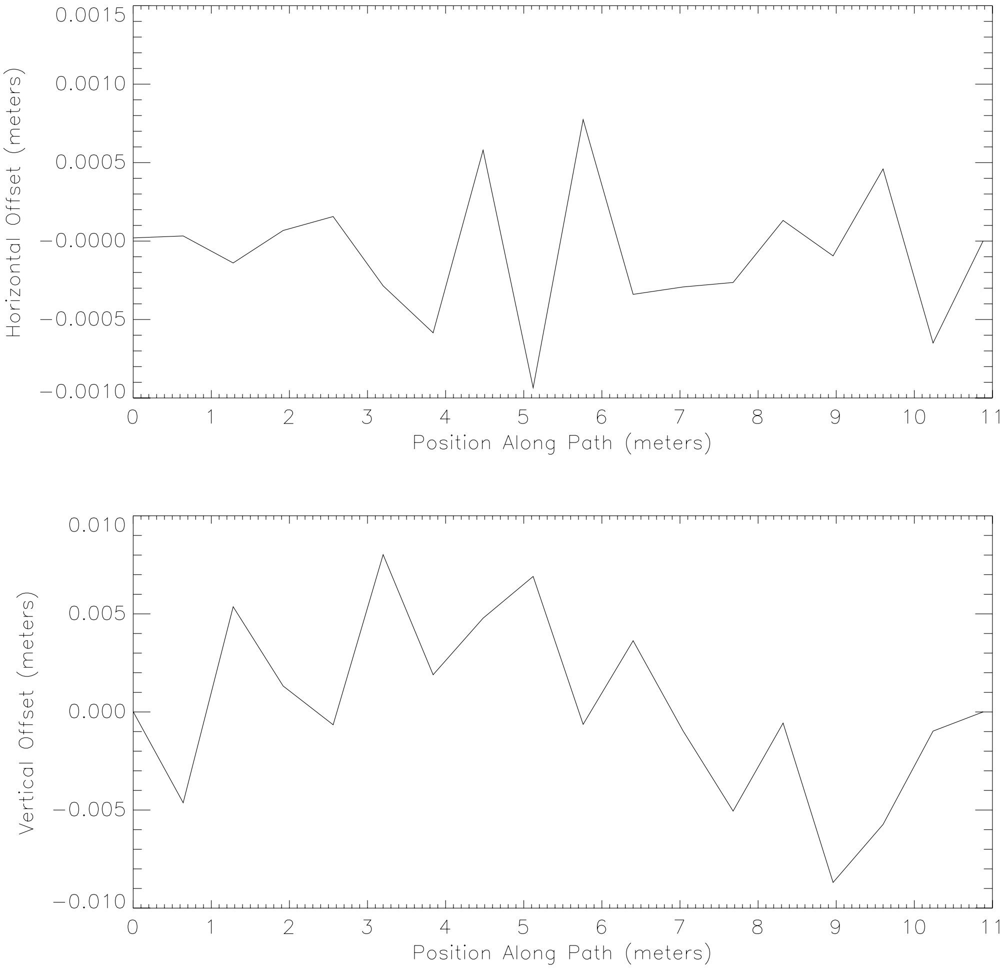


Figure 5.7: A plot of the closed orbits for one operating point both horizontally and vertically.

electron gun's aperture, and following the particles through the beamline as they go through injection and into the ring for a set number of turns, the number of initial conditions that remain in the beam pipe (2.95 cm radius) is used to determine the dynamic aperture. The initial conditions had a uniform distribution in space, using the cathode radius of 4 mm, and were Gaussian in angle distribution, using a temperature of $1100°C$. In the study shown in Figure 5.8 that number could be as much as 100% of the particles, or as low as 0%. Large differences are seen in the survival rates of the different tunes. The best operating point seems to be in the vicinity of the ideal tune. Figure 5.8 also has the low-order resonance lines shown in it; it should be noted that this contour plot is created by interpolating a grid of 81 operating points. This means that even if a low-order resonance seems to show a small effect, that does not mean that it is weak, only that there was no operating point calculated on that resonance.

5.1.8 Resonance Strengths

The strengths of the various resonances can also be calculated. In Figure 5.9 the resonance strengths for four operating points are shown; the 7 mA operating point has smaller resonance strengths than the others. The four operating points in Figure 5.9 have different steering and magnet settings which lead to different resonances being dominant. However, the $(1, 2)$, $(1, 0)$ and $(2, 1)$ resonances are prominent in each operating point, which is most likely the resonances excited by the 17 ring sections since these are not changed as significantly as the injection line and the Y-section between the operating points. Note the absence of large

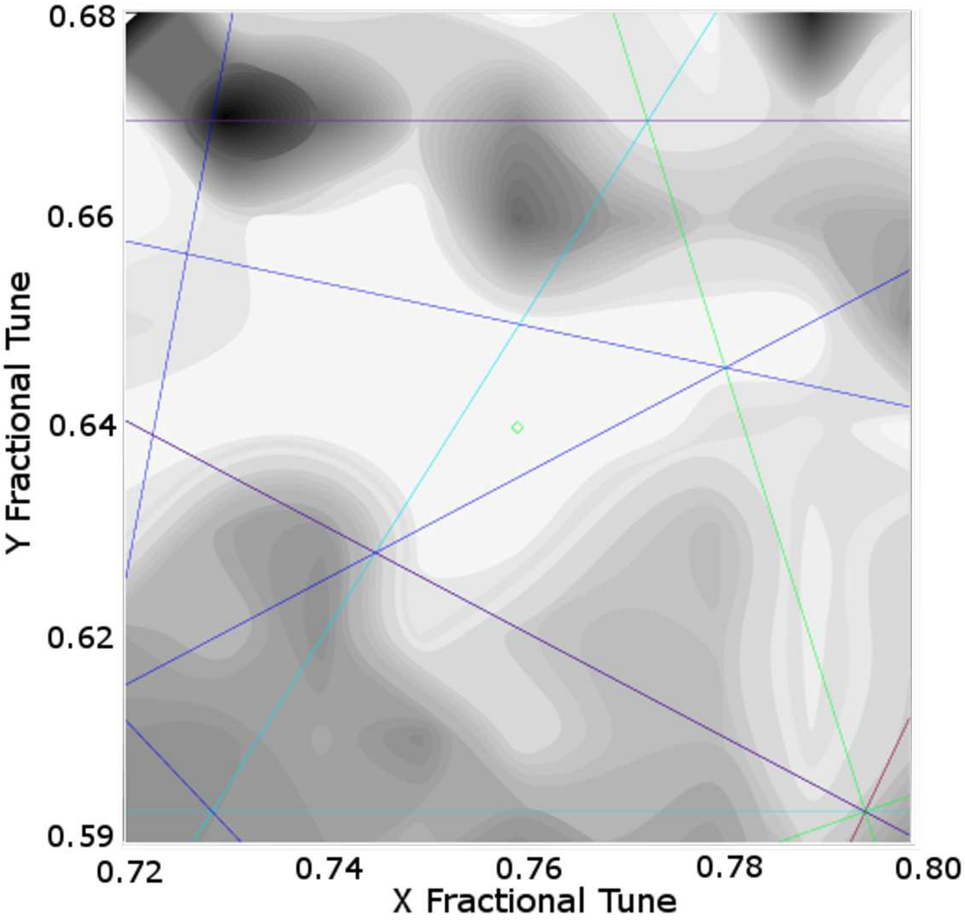


Figure 5.8: A contour plot of the scanned tune space. Darker areas denote less of the beam surviving after 100 turns. The green ◇ symbol represents the location of the ideal tunes. The lines follow the same color scheme as Figure 5.4

high-order resonances in any of the operating points. It is interesting to note that operationally the 7 mA beam seems to be the most well-behaved beam.

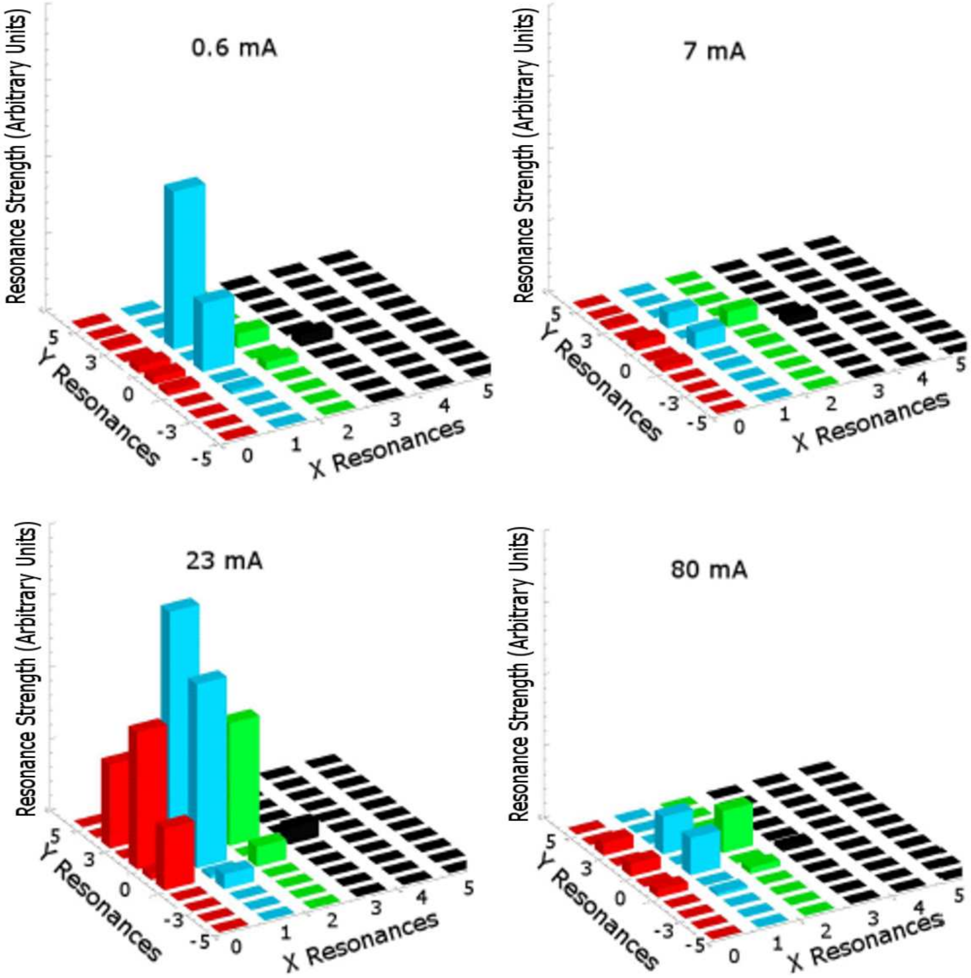


Figure 5.9: A plot of up to sixth-order resonance strengths for the four current values examined. Strengths are based on a 50 μm emittance and all vertical axes are to the same scale. Colors are for visual differentiation only.

5.2 Moment Method Applications

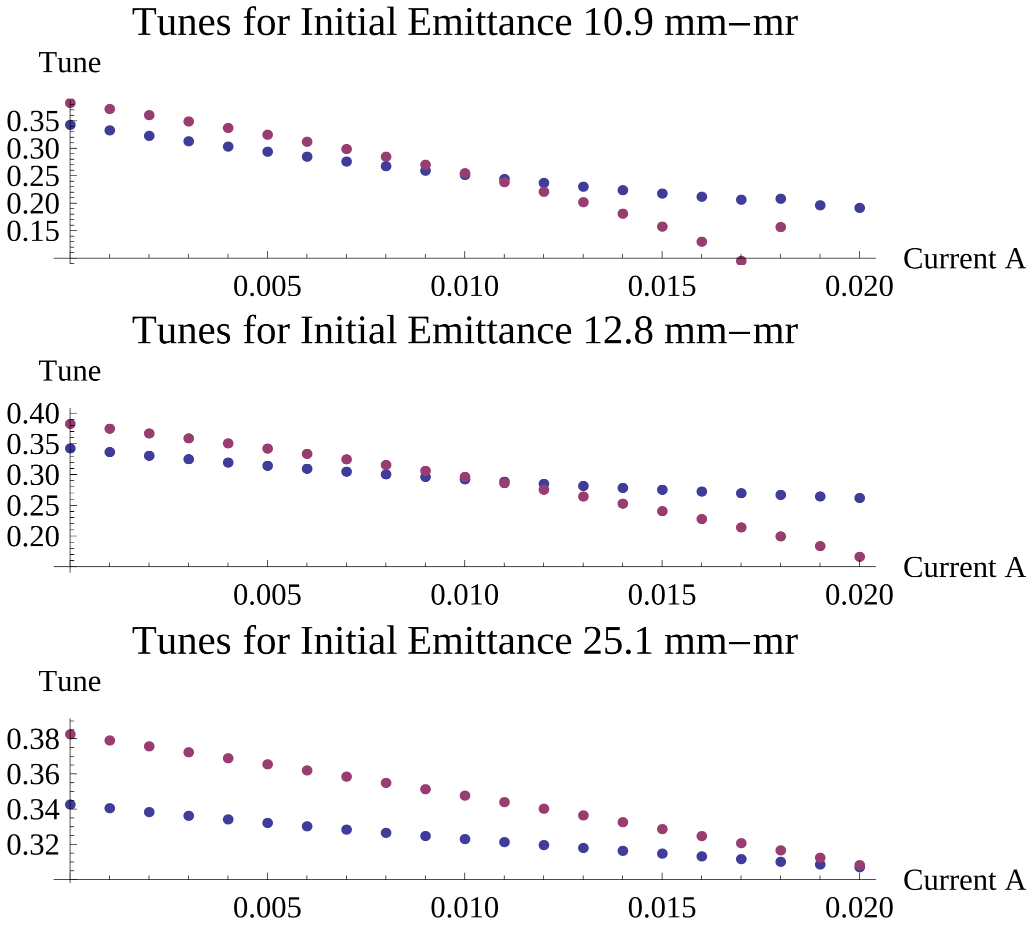
Now that the single particle dynamics have been analyzed we would like to look at how space charge affects the system. This is done using the moment method both to calculate the map and to advance the particles through the system; for simple distributions this works well, for more complex distributions an alternate method of advancing the particles is needed. This is shown in Section 5.3 when we look at the fast multipole method.

5.2.1 Betatron Tunes

The addition of the effects of space charge to the transfer map of a system allows the user to directly calculate the betatron tune in a manner which includes space charge. The tune shift can then be calculated by subtracting the bare tune from the shifted tune. If we look at the change in the tune over a single cell of UMER we can see the effects of space charge directly on the tune in Figure 5.10. As we can see, the effect scales both with the current of the beam and with the initial emittance of the beam.

Furthermore we can use the closed-form versions of the Laslett tune shift to determine what we should expect. For a circular machine such as UMER the equation for the tune shift is [49]

$$\Delta\nu = \frac{IR}{I_0 \epsilon_n \beta^2 \gamma^2}, \tag{5.1}$$



Tunes for Initial Emittance 10.9 mm−mr

Tunes for Initial Emittance 12.8 mm−mr

Tunes for Initial Emittance 25.1 mm−mr

Figure 5.10: This shows the tune shift caused by space charge in an example of the UMER lattice.

Table 5.2: Table of calculated tune shifts

Name	Laslett Tune shift	COSY tune shift x	COSY tune shift y
pencil	$-.5134$	$-.3285$	$-.45807$

Table 5.3: Table of calculated tune shifts using a Gaussian instead of a uniform distribution.

Name	Bare Tune	COSY tune	Tune shift
pencil x	0.16644	0.28843	-0.87801
pencil y	0.88433	0.34253	-0.5418

where I is the current, R is the radius of the ring, I_0 is the Alfven current, ϵ_n is the normalized emittance, and β and γ are the relativistic factors. For both of these cases a uniform cylindrical distribution has been used that has been matched to the lattice being observed. This gives tune shifts for the smallest space charge value shown in Table 5.2. The tune shift that is calculated for the same matching settings but using a Gaussian distribution is shown in Table 5.3.

5.2.2 Chromaticities

Another quantity that can be directly extracted from the map of the beam is the chromaticity, which is the derivative of the tune with respect to energy. If a particle does not have the same energy as the rest of the beam then it will see a different betatron tune. As can be seen in Figure 5.11, the change is not a large one, but the effects of space charge on the chromaticity will become saturated. This is to be expected since as the current increases space charge will become a greater influence on the beam than the lattice.

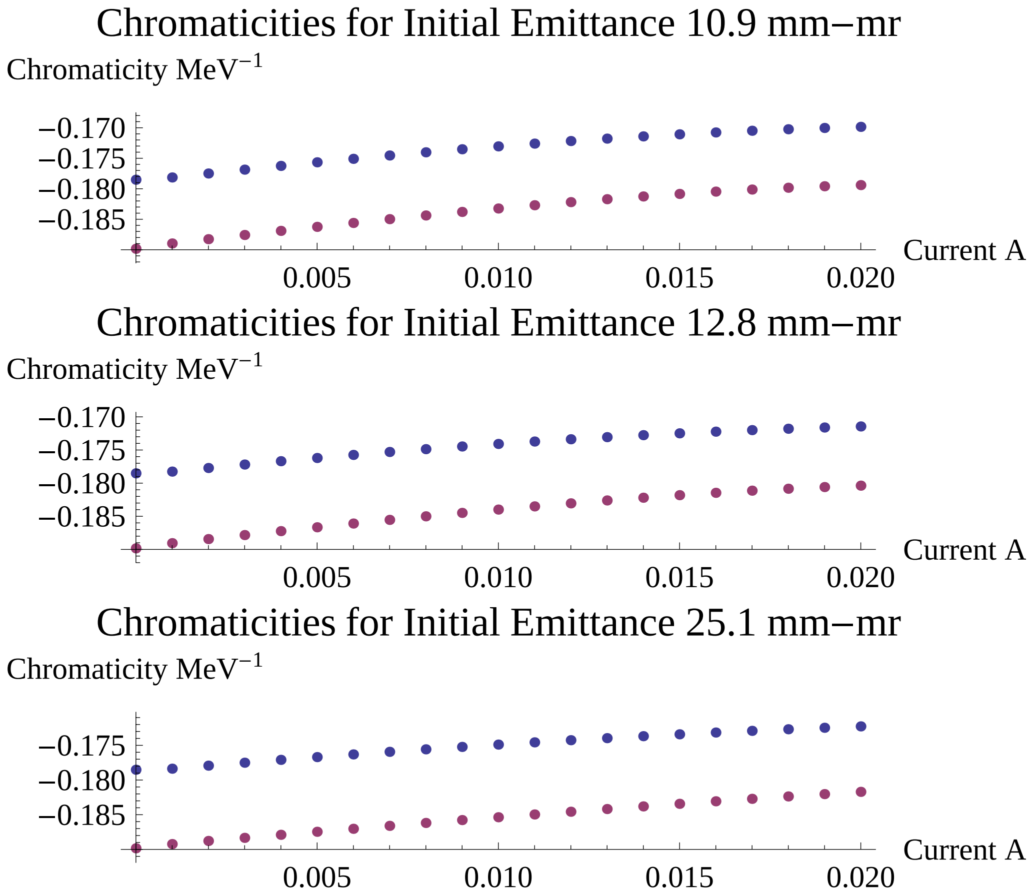


Figure 5.11: This shows the change in chromaticity for a series of initial conditions over one section of the UMER lattice.

5.2.3 Resonance Strengths

The effects of space charge can strengthen resonances within the beam to the point that they can cause instabilities. An example of this is shown in Figure 5.12.

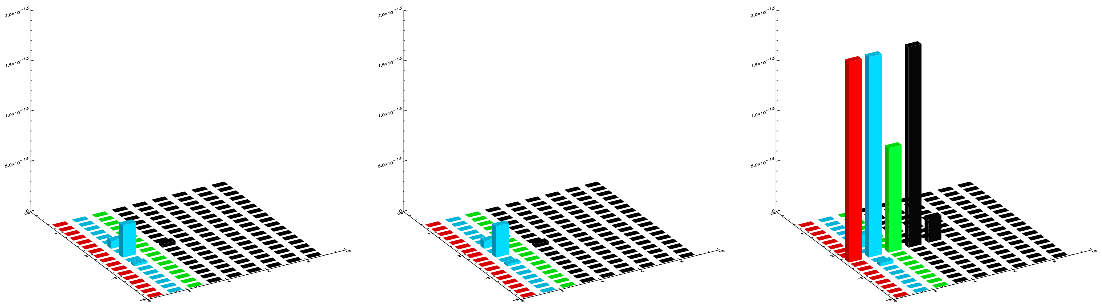


Figure 5.12: This shows the resonance strengths for 0, 1, and 2 mA beam currents.

5.2.4 Long-Term Tracking

The abilities of the moment method to quickly track large numbers of particles through a ring in an accurate manner with the effects of space charge included can be seen in Figure 5.13. With lower levels of space charge, the effects largely balance out within the first few turns. Furthermore the emittances stay very stable over the large number of turns.

5.3 Fast Multipole Method Applications

The moment method provides a large number of opportunities to measure phenomena that it was difficult to simulate before. The fact that it can not only

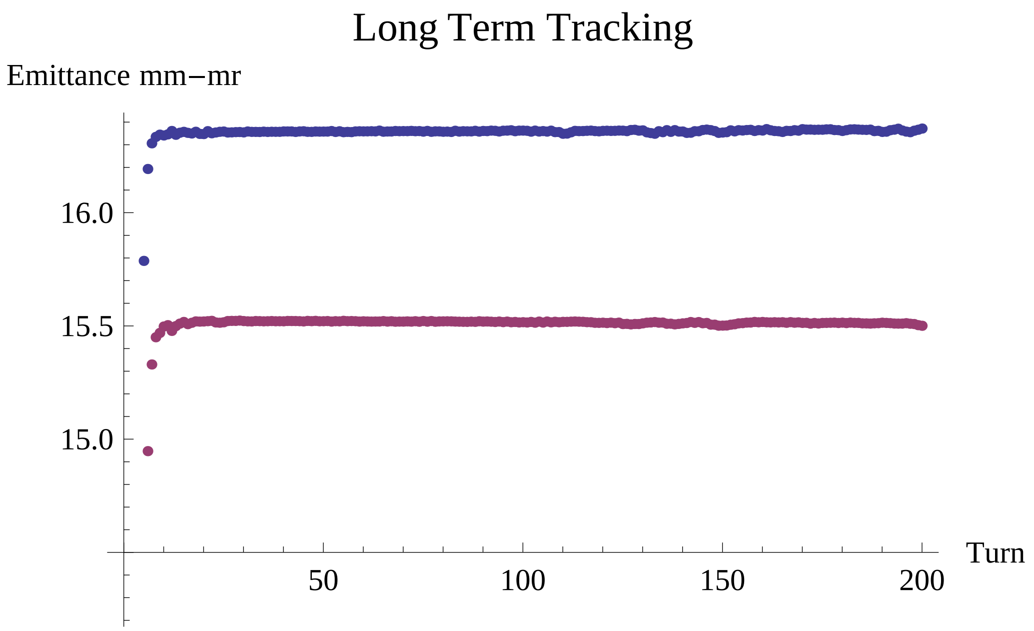


Figure 5.13: This shows the horizontal and vertical emittances of the 6mA beam in an idealized UMER lattice.

determine the map of a system, but also advance particles through it would lead us to believe that that is all we need to fully simulate a beam. However there are some cases where the moment method is not a complete enough way to model the system. In the following subsections several uses of the fast multipole method are shown for systems we may wish to simulate, but would be unable to using only the moment method.

5.3.1 Multimodal Distributions

Here we can use some of the new features of the FMM to analyze systems that it would not be possible to understand with the moment method. The example we show here allows an examination of both a multimodal distribution, and different species of particles evolving through a system. In this instance we start with two parallel beams: one of electrons, one of positrons. As they move through a drift, their evolution is shown in Figure 5.14. This ability can be used to model many different phenomena, from heavy ion beams with multiple isotopes, to electron cooling systems.

5.3.2 Diffuse Halo

The moment method works well when the distribution function is easily approximated with a 16^{th}-order Taylor series. However, in some cases, such as a diffuse halo, there will be complications. An example is shown in Figure 5.15. It is clear that the Taylor series that is being used to find the potential has its edge effects intercepting the beam, causing them to undergo non-physical motion. The fast multipole method does not have this sort of constraint; therefore, it can be used for distributions which are not easily modeled using a Taylor series.

5.3.3 Long-Term Tracking

As with the moment method, it is desireable to track statistical information regarding a beam as it makes a series of trips around the ring. This was performed using the fast multipole method for 10,000 particles over a simple model of the UMER lattice. The results of this tracking are shown in Figure 5.16. The distribution was a smaller subset of the one used for the moment method long-term tracking simulation, and is matched to the space charge free lattice. As can be seen from the figure the emittances quickly reach their equilibrium values and oscillate there.

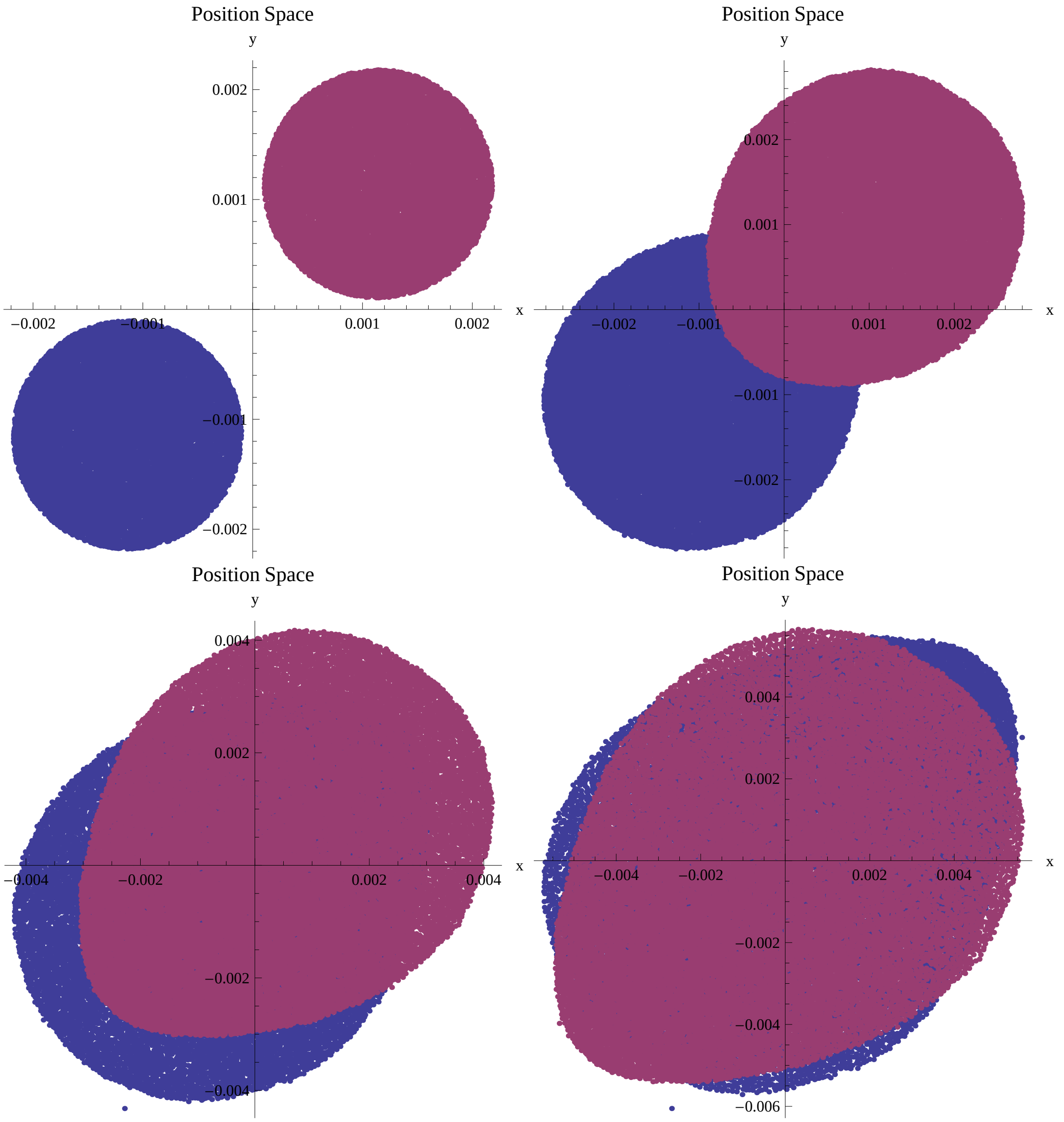


Figure 5.14: This shows the evolution of two parallel beams: one of electrons and one of positrons; annihilations are not modeled.

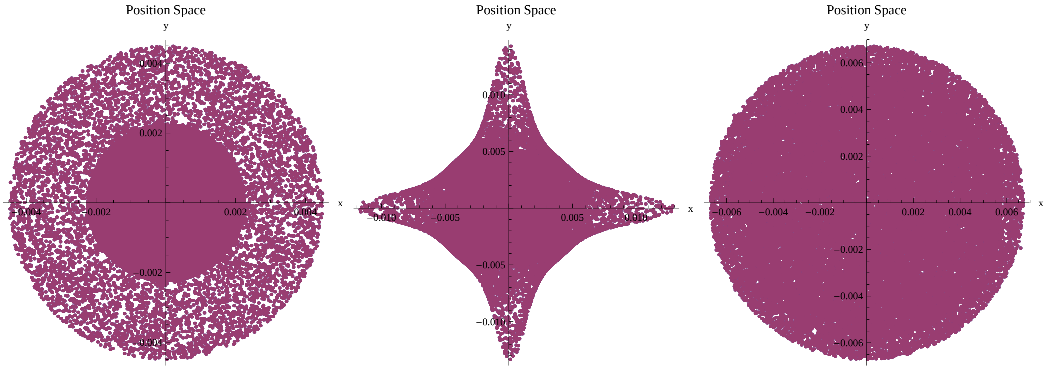


Figure 5.15: On the left are the initial conditions of the system, in the center is the moment method attempt, and on the right is the fast multipole method.

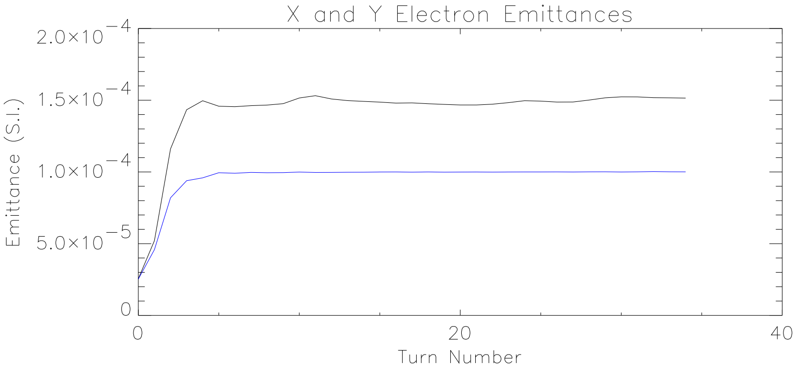


Figure 5.16: This graph shows the emittances of a beam over a number of turns, the blue line shows the emittance of the vertical dimension, the black line the horizontal.

CHAPTER 6

UMER EXPERIMENTS

In order to make sure that the tools that have been developed to simulate intense particle beams in general, and the University of Maryland Electron Ring in particular are accurate, a series of experiments were run over two time periods. The first set of experiments measured single-particle effects and were performed during the first half of March in 2009 [50]. These are shown in Sections 6.1, 6.2, and 6.3. The second set of experiments attempted a series of multiple-particle experiments, and were performed during early December of 2010. These are shown in Sections 6.4 and 6.5.

6.1 Particle Tracking

The first experiment was to compare the results of the steering solution provided by COSY and the previous magnetic field compensation values, as described in Section 5.6. The results of simulating both steering settings, as well as their measured values, are shown in Figure 6.1. We see fairly good agreement between the predicted and measured values as they progress around the ring. The relative movement between the previous values and the COSY derived values in both the measured and predicted cases are similar. Technical difficulties prevented accurate tracking through the 11^{th} beam position monitor. Therefore, most tracking studies were only performed up through RC 9. These technical issues prevented a full examination of the tracking predictions, but the data that was taken can be used as a tool to diagnose misalignments [52]. Since the steering solution is off-center in most of the ring elements, all non-linear effects are present due to feed-down. Hence, a good agreement between measured and predicted steering solutions is indicative of an accurate modeling of the full non-linear dynamics.

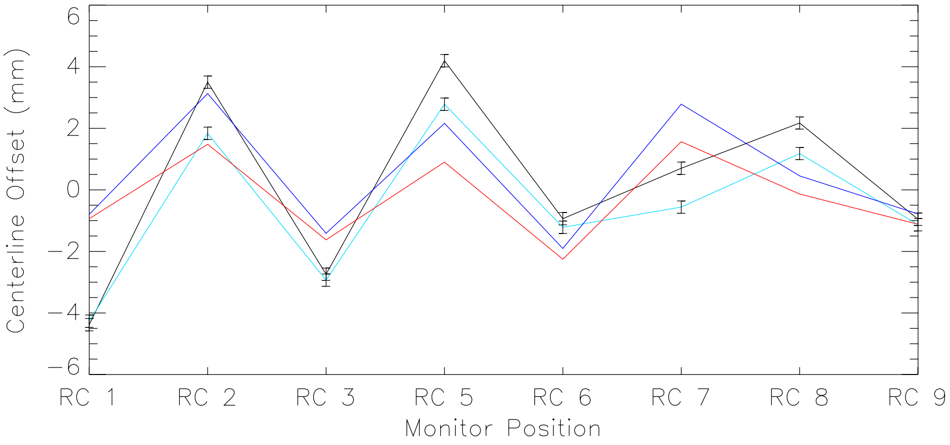


Figure 6.1: A comparison of the predicted and measured ring steering trajectories. The black line is the previous magnetic field compensation value, while the light blue line is the one calculated in this study. The dark blue line is the predicted trajectory for the previous compensation value, while the red line is the predicted trajectory for the settings determined in this study.

The COSY predicted settings compared with the UMER calculated settings, as well as another set of UMER settings based on a LOCO-type response matrix steering algorithm [53], were used to provide a common basis for comparison of the steering solutions; see Figure 6.2. The original steering solution in COSY assumed that the beam would be injected straight along the centerline, which is not necessarily the case, so a new steering solution was devised based on the UMER stock matching settings. As can be seen in Figure 6.3, there was significant improvement.

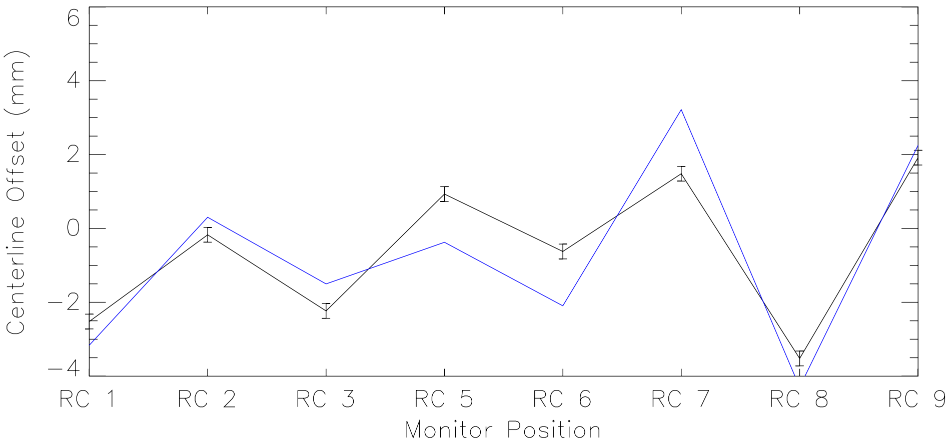


Figure 6.2: Steering solution using the default values for the UMER ring; a comparison of measured (black) to predicted values (blue).

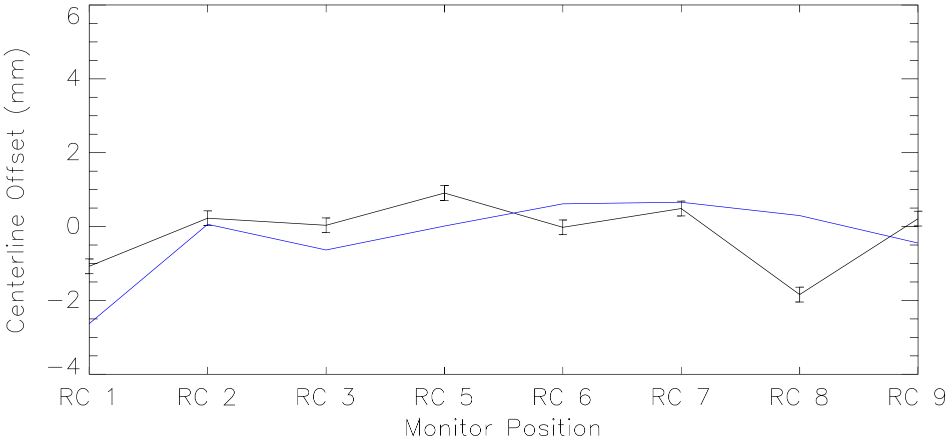


Figure 6.3: The blue line is the predicted value for the new steering settings, while the black line is the measured trajectory.

6.2 Tune Measurements

The tunes were measured directly in the ring at each BPM using a four-point method outlined in [54]. It works assuming that there is BPM data for four subsequent turns, with the tune calculated with

$$\cos(2\pi\mu) = \frac{x_n - x_{n+1} + x_{n+2} - x_{n+3}}{2(x_{n+1} - x_{n+2})}. \tag{6.1}$$

This method is useful since it requires only four measurements per BPM and is immune to calibration offsets since every offset is subtracted out. Unfortunately, sometimes if the measurements come out a certain way, they can give non-physical results. A series of tune measurements were performed on different sets of operating points and currents on the UMER ring; these results are shown in Table 6.1.

The large standard deviations shown in Table 6.1 do show that there is still some uncertainty in the measurements taken. Theoretically the beam position monitors are accurate to 0.1 mm; however, the measured values correspond more closely to 1 to 2 mm. A major factor in this has to do with signal to noise ratios in the beam position monitoring equipment which can add a great deal of error to the measured voltage across the BPM plates.

6.3 Chromaticity Measurements

The chromaticities were measured for the ring by changing the voltage across the electron gun and measuring the tune using (6.1). The chromaticity is defined

Table 6.1: Measured horizontal tunes measured in the UMER ring using the four-point method. Some measurements cannot give a good tune, so those have been left blank. The averages and standard deviations do not include these missing values.

BPM	Pencil Beam	7mA	20mA	80 mA
RC1		.429265	.439089	.251808
RC2		.469785	.455473	.322365
RC3		.249922		.285294
RC5		.429028	.443181	.324413
RC6	.240492	.43788	.348265	.373094
RC7	.257451	.362942	.434079	.298863
RC8	.09181	.421028	.358952	.303061
RC9	.265539	.410606	.415081	.120793
RC11	.294148	.415142		.30992
RC12	.292334	.40613	.406399	
RC13	.327488	.421758	.474573	.332532
RC14	.292729	.404955	.401591	.348996
RC15	.245612	.417701		
RC17	.195635	.422057	.409515	.397896
AVG	.250324	.407014	.416927	.305753
σ_μ	.06663	.050635	.038423	.069966

as the change in tune with respect to the energy of the system, so a simple fitting method will work to find the chromaticity from measured tunes. The results of these measurements are shown in Table 6.2.

The 80mA settings had a great deal of beam scraping in the off-energy particles leading to some issues with gaining accurate measurements; for this reason those are not shown.

6.4 Distribution Tracking

An initial test that can be performed is to simply simulate the ring as it moves through each section and observe the distribution. This was performed for RC1, RC2, and RC3 using the 6mA beam. These can be contrasted with the COSY-

Table 6.2: Measured horizontal tunes measured in the UMER ring using different accelerating voltages in the electron gun to help find the chromaticity.

Energy (kev)	Pencil Beam	7mA	20mA
9.7	.279371	.317212	.406874
9.8	.33028	.359018	.434898
9.9	.271305	.366088	.449118
10.0	.334731	.407014	.435164
10.1	.368859	.435358	.411967
10.2	.351122	.441598	.385869
10.3	.364947	.422368	.358888
$\frac{d\mu}{\delta}$	6.3299	2.8719	-2.0682

generated distributions that come out of the system. It is important to note that currently the space charge method in COSY is a two-dimensional method, so effects of dispersion are not included. The distribution arrived at experimentally, and computationally as well as the vertical profiles of the beam both experimentally and computationally are shown in Figures 6.4 and 6.5 for the phosphor screens at RC1 and RC2 respectively.

The biggest problem that has been encountered is the lack of modeling of dispersion in the ring, which significantly narrows the horizontal spread of the beam. As can be seen in Figures 6.4 and 6.5, when the horizontal dimensions are scaled up with respect to the vertical, we get a much closer agreement between simulation and observation. The somewhat unique beam profile seen in the simulations and the data are caused by the hollow velocity distribution that UMER possesses [55].

The next experiment was an attempt to examine how a series of distributions would evolve through the machine over time. In order to have a set of readymade distributions and watch how they evolve over a period of repeating cells, the first quadrupoles after the beginning of the injection line, labeled QR2 and QR3, were scanned across a series of values giving a series of different distributions at RC1. Images of the beam at RC2 and RC3 were taken, allowing for a series of measure-

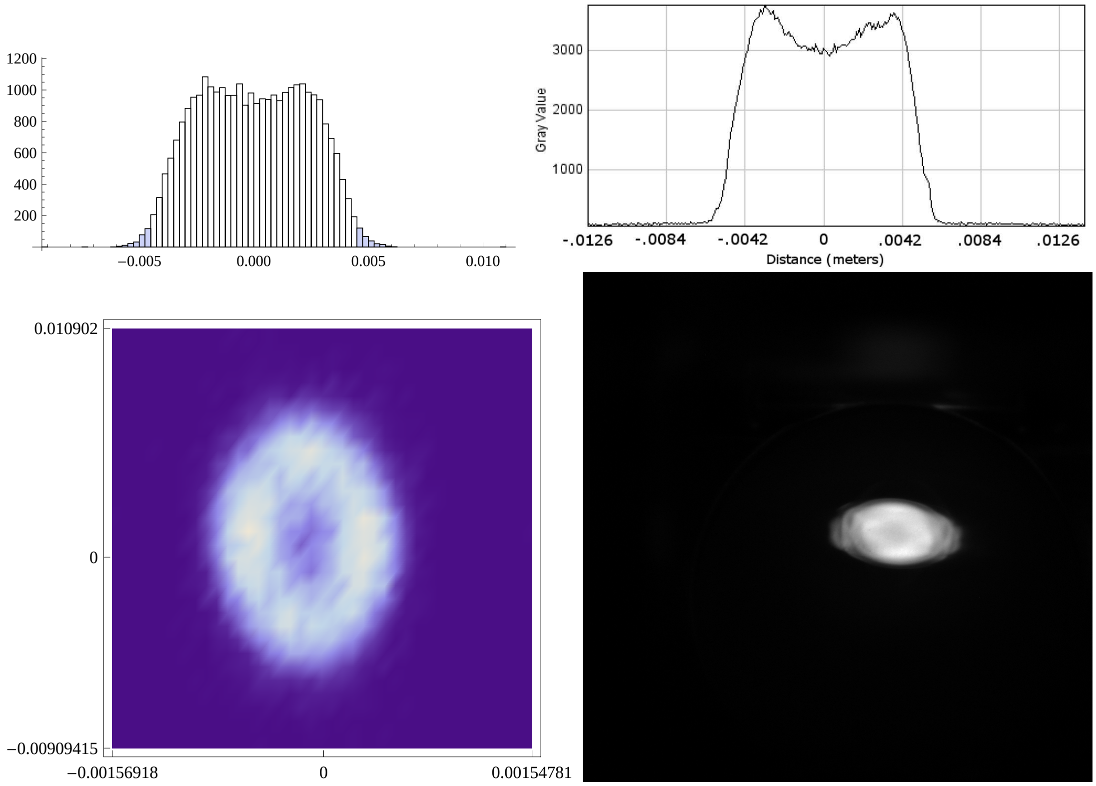


Figure 6.4: The first image is a vertical beam profile using the simulated UMER ring, with the positions normalized to the center of the beam bunch. The second image is the vertical profile measured from the data taken on the ring. The third image is a density plot in position space using the simulated beam, and the fourth is the image taken at RC1.

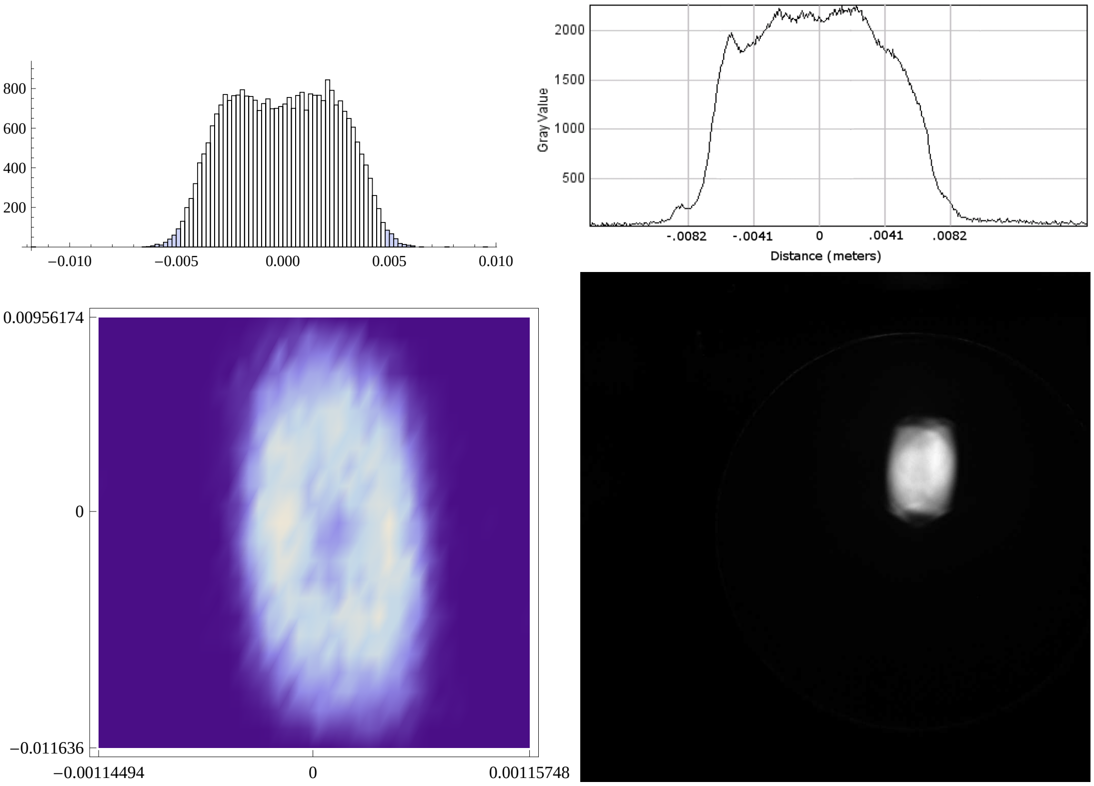


Figure 6.5: The first image is a vertical beam profile using the simulated UMER ring, with the positions normalized to the center of the beam bunch. The second image is the vertical profile measured from the data taken on the ring. The third image is a density plot in position space using the simulated beam, and the fourth is the image taken at RC2.

ments to be taken of the way a beam evolves as it moves through various FODO sections. This setup is shown in Figure 6.6.

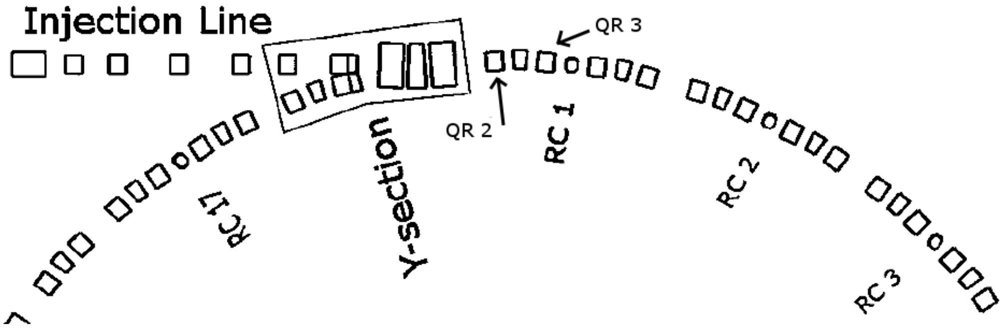


Figure 6.6: This closeup view of the ring shows the area used for the distribution experiments.

This was done so that a comparison of different distributions moving through the ring could be made and compared to the simulations. The results for RC1 are shown measured in Figure 6.7 and simulated in Figure 6.8. The results for RC2 are shown in Figure 6.9 with the simulated results shown in Figure 6.10. Finally, the measured results for RC3 are shown in Figure 6.11 and the measured results are shown in Figure 6.12. Once again the issue of the horizontal spread being smaller than expected is seen. However the vertical sizes once again match closely.

6.5 Distribution Scraping

In order to measure how the effects of space charge depend on the shape of the beam, an experiment was carried out to scrape off part of the beam as it passed through RC9. The intent of the experiment was to alter the shape of the beam and then observe how this new version of the beam propagated through the system. Data was then taken at subsequent phosphor screens. The results are shown in Figures

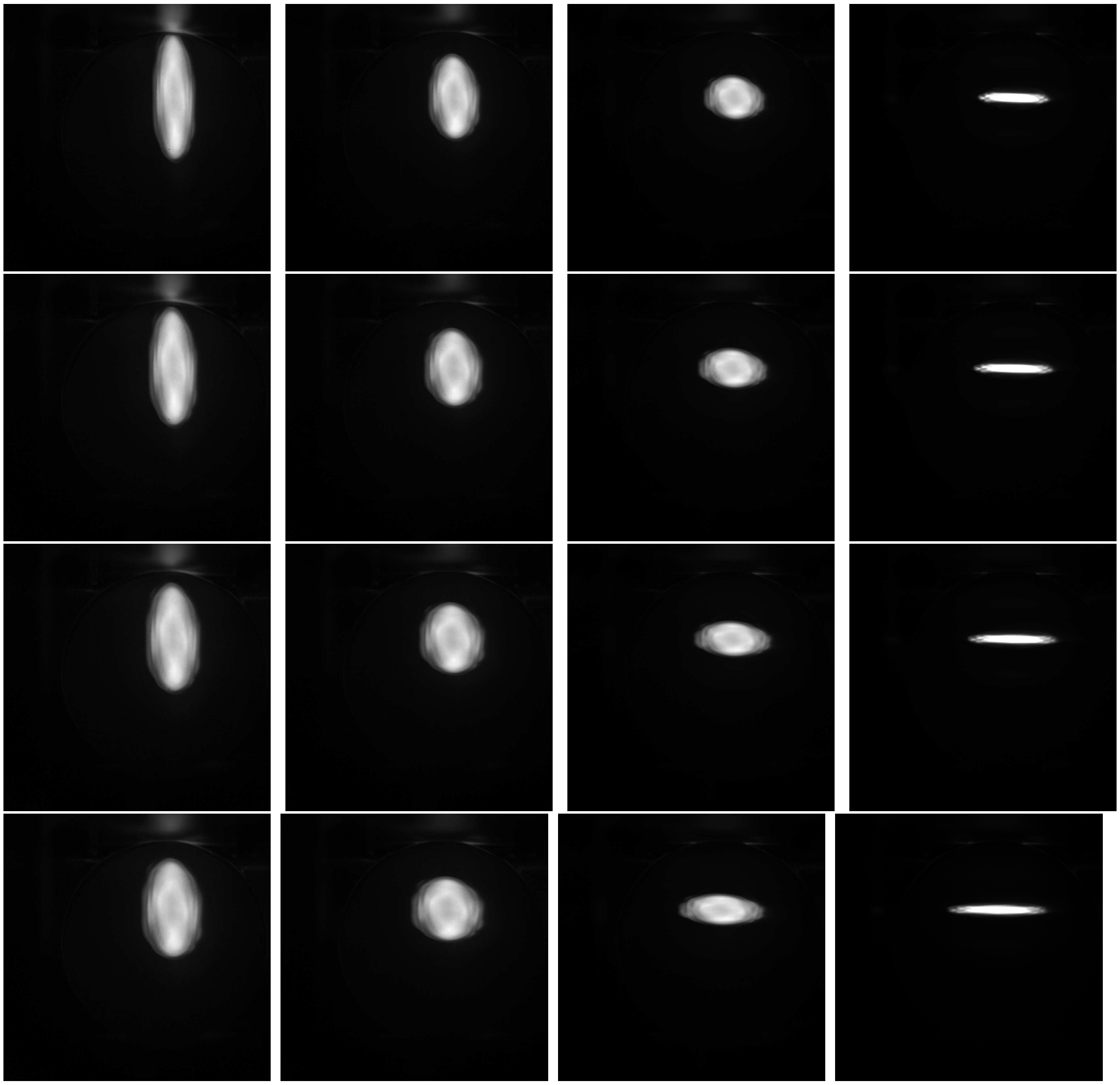


Figure 6.7: This shows data taken from the ring at RC1. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, while QR2 is scanned horizontally.

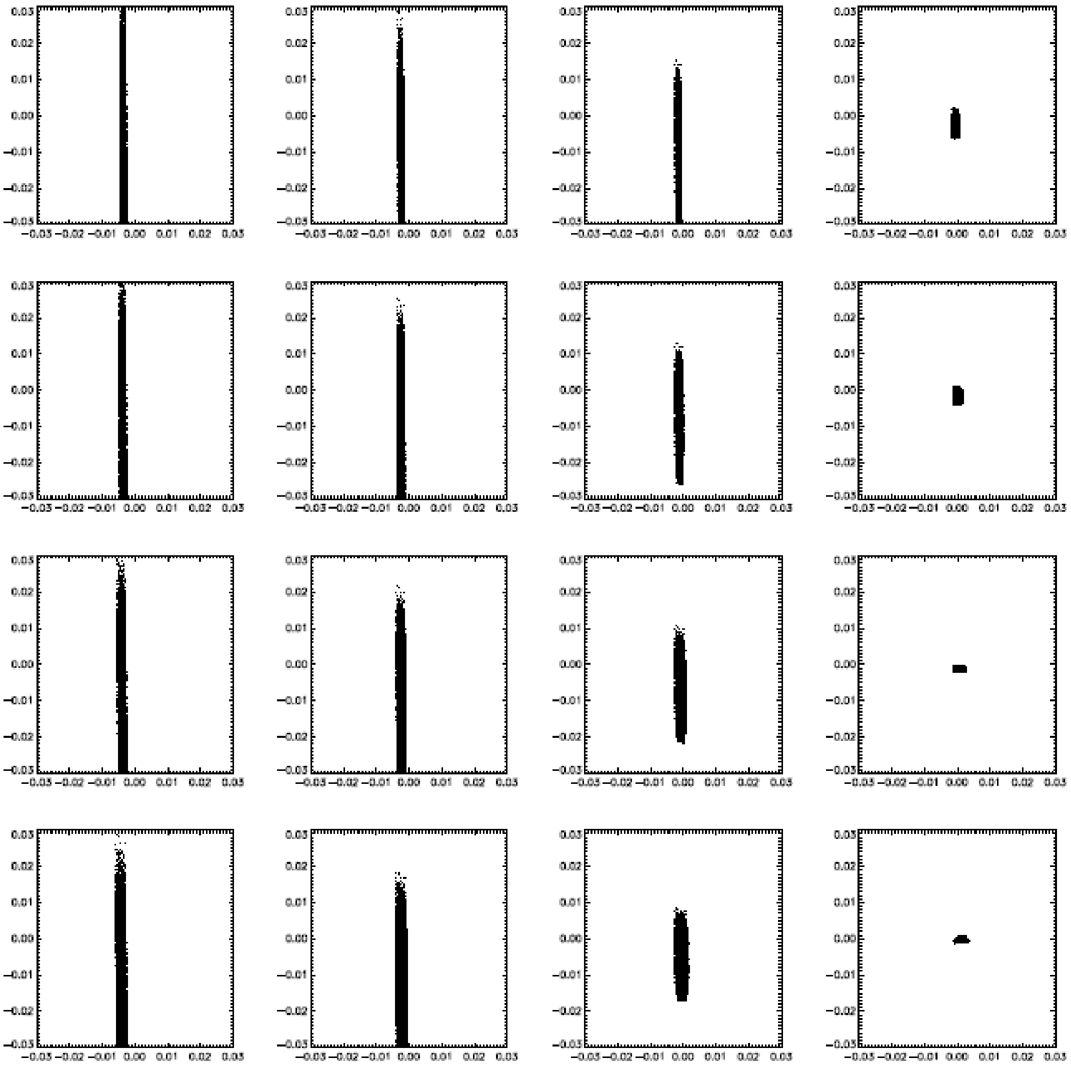


Figure 6.8: This shows data simulated in the ring at RC1. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, QR2 is scanned horizontally.

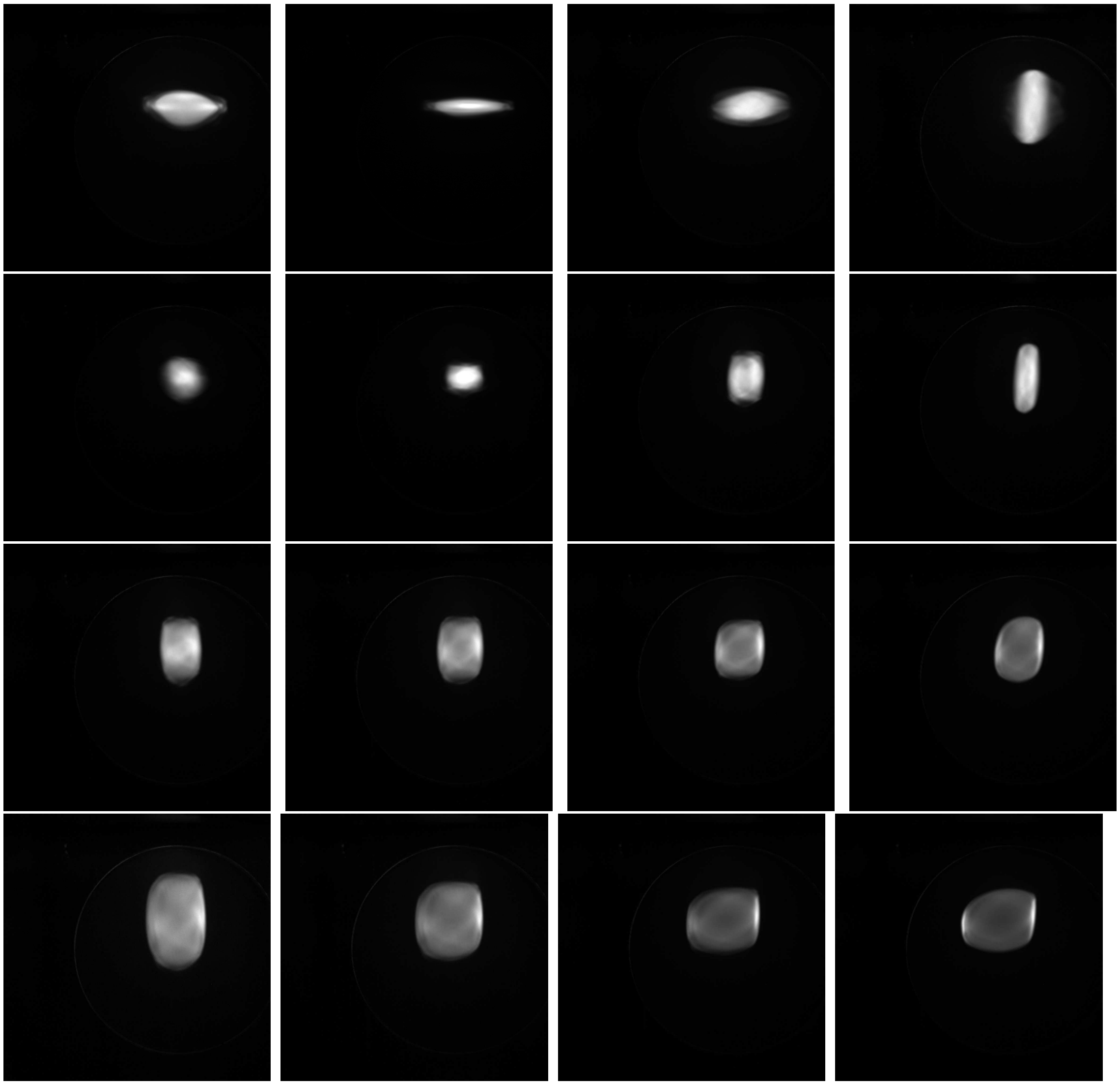


Figure 6.9: This shows data taken from the ring at RC2. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, while QR2 is scanned horizontally.

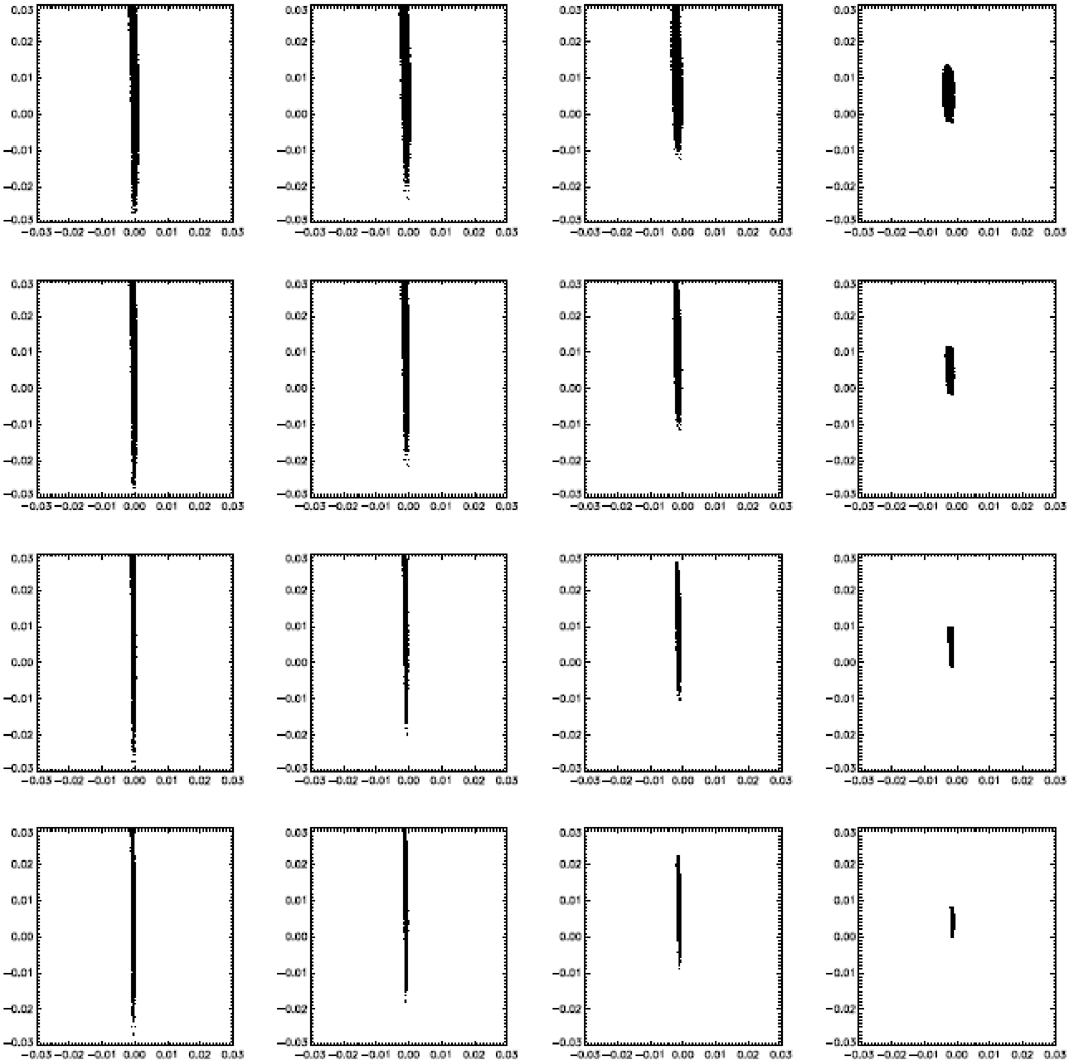


Figure 6.10: This shows data simulated in the ring at RC2. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, while QR2 is scanned horizontally.

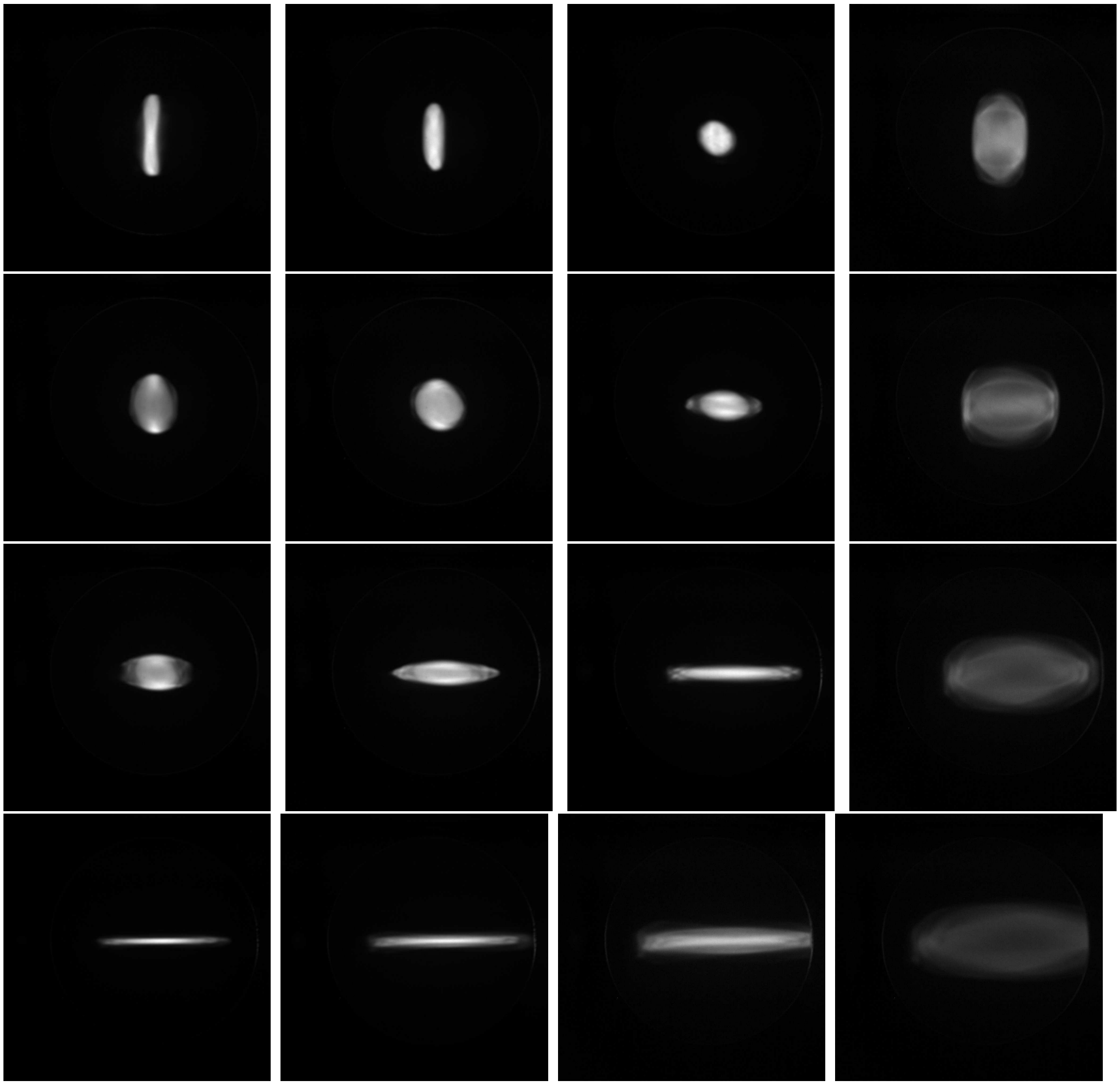


Figure 6.11: This shows data taken from the ring at RC3. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, while QR2 is scanned horizontally.

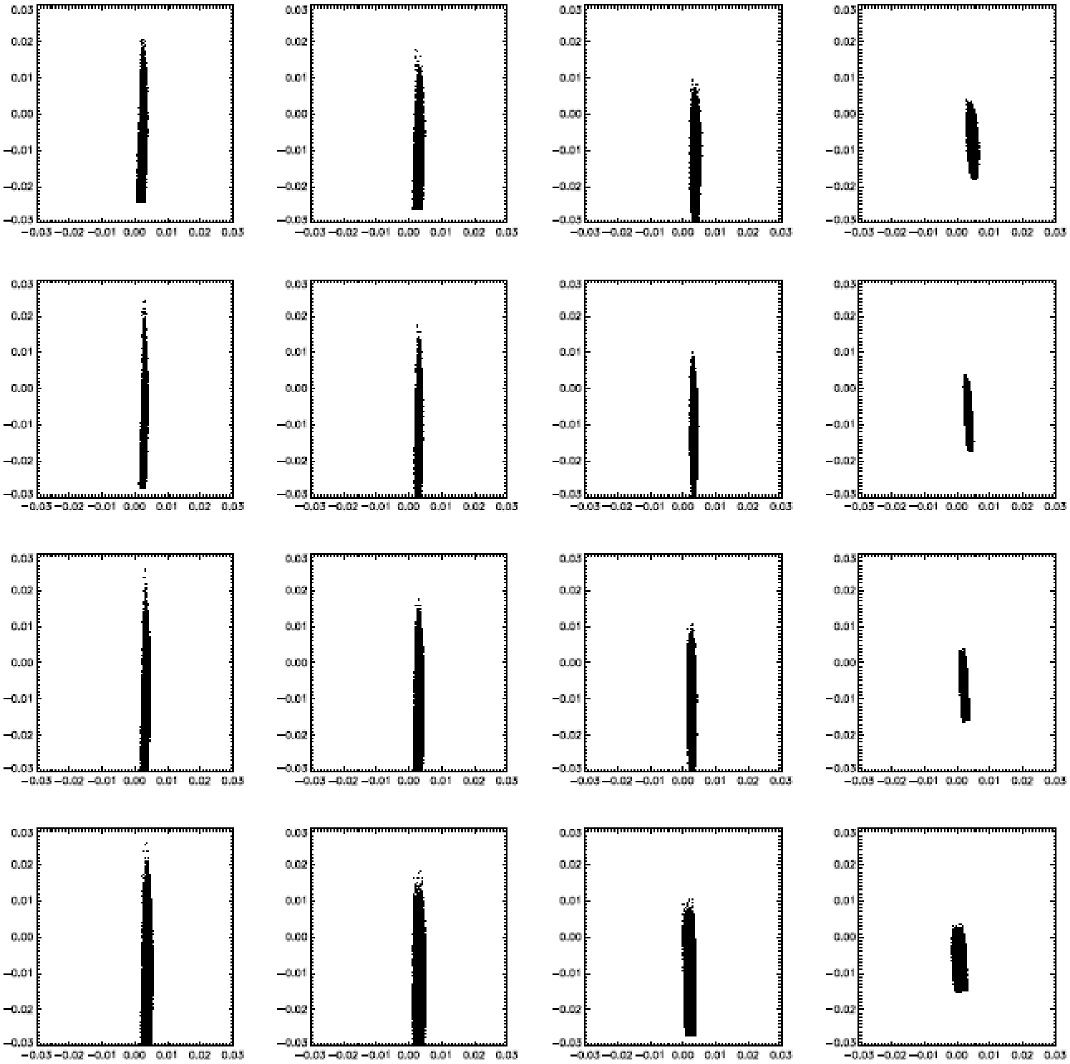


Figure 6.12: This shows data simulated in the ring at RC3. The quadrupoles QR2 and QR3 are scanned between .75 and 2.25 in equal increments. QR3 is scanned vertically, while QR2 is scanned horizontally.

6.13 and 6.14. The beam was scraped using a hand-controlled fast phosphorus screen that was put in RC9. Unfortunately this fast phosphorus screen is difficult to take data from, so a best guess had to be made regarding the initial distribution that would be scraped; furthermore, since the beam size needed for multiple scraping distances necessitated the 21 mA beam, steering issues prevented an accurate set of initial conditions for the beam. For the purposes of the simulation, the size of the scraped region was calculated using pixel values from fast phosphor screen images.

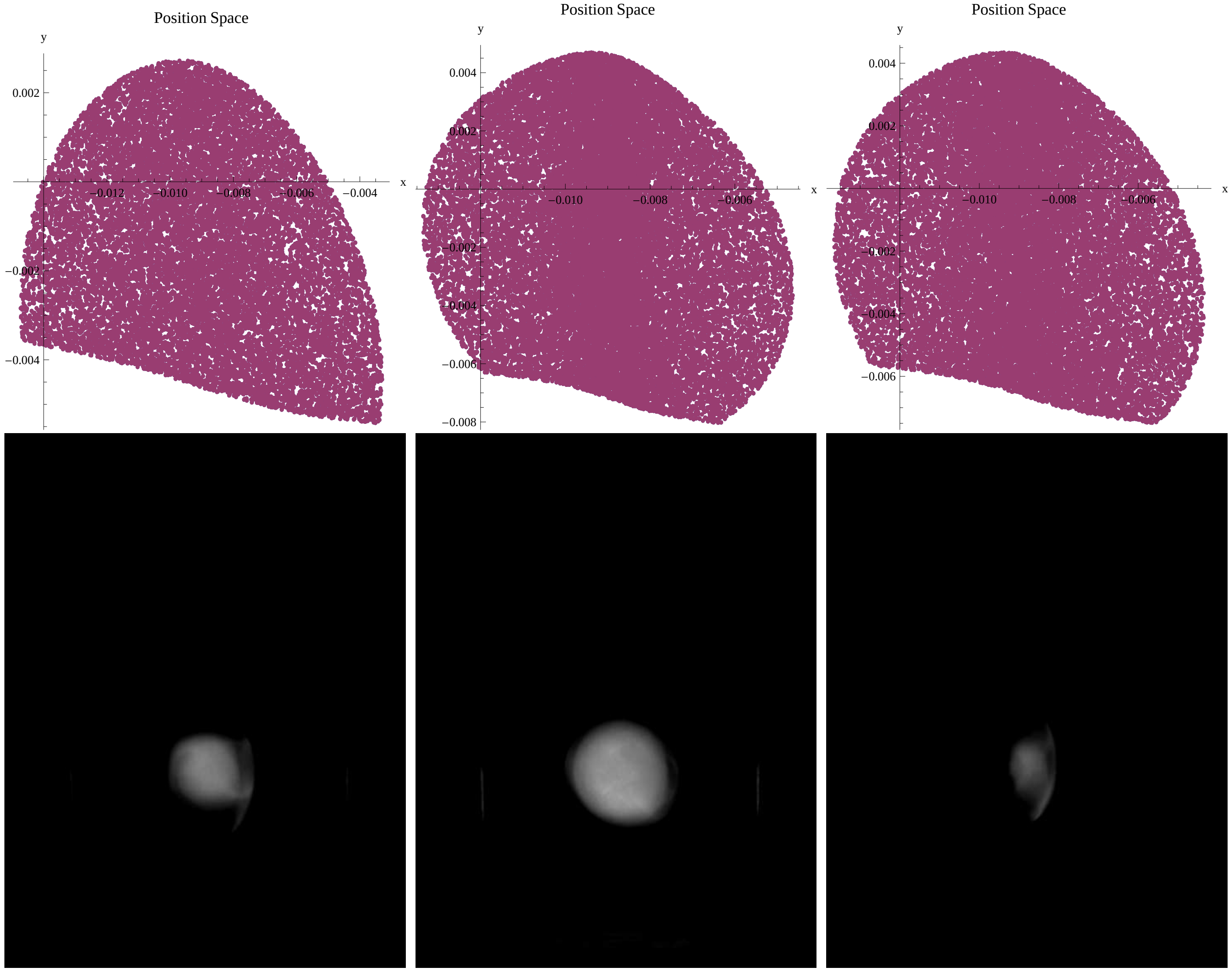


Figure 6.13: On the top are the simulated distributions and on the bottom are the recorded distributions for RC11.

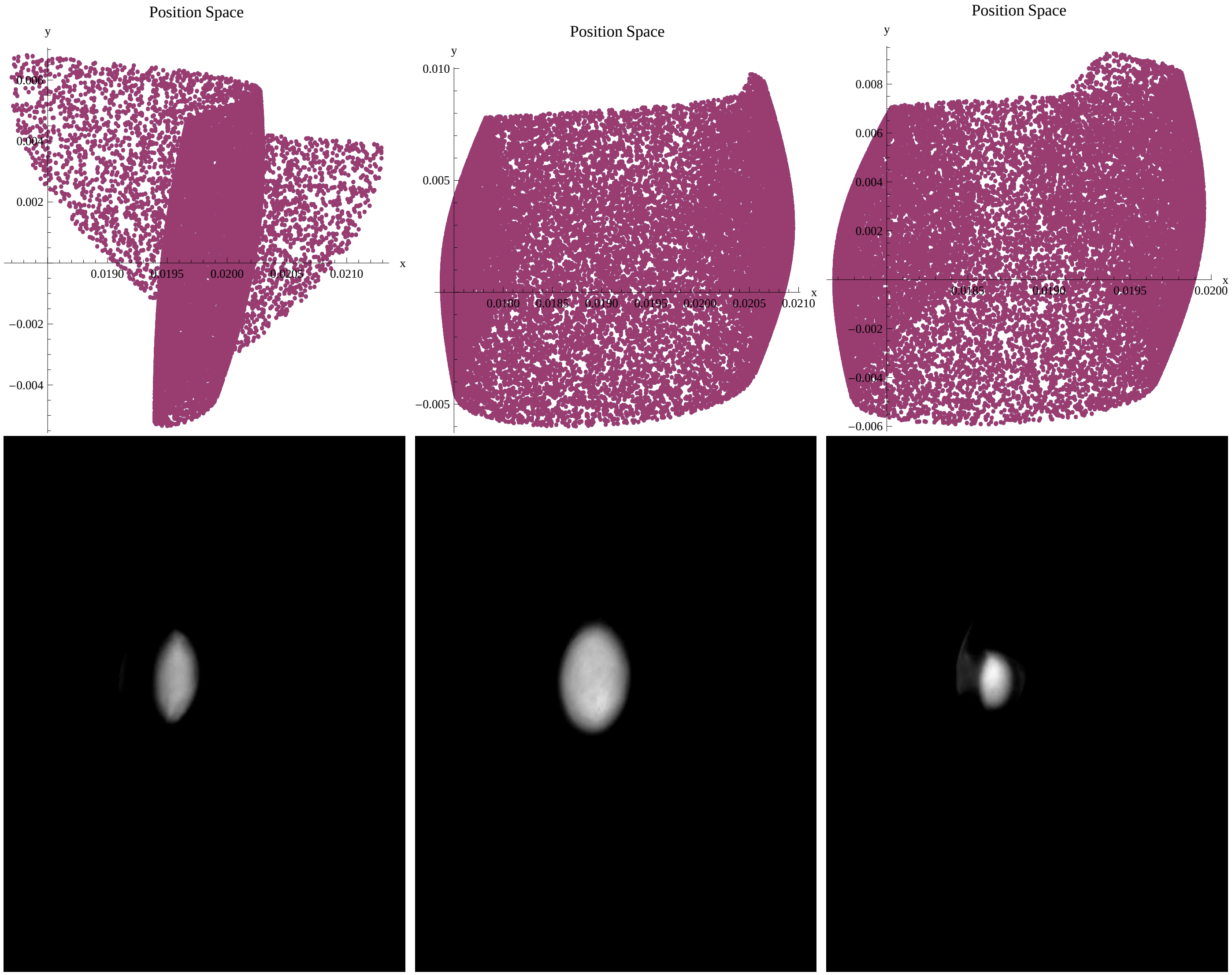


Figure 6.14: On the top are the simulated distributions and on the bottom are the recorded distributions for RC12.

CHAPTER 7

CONCLUSIONS

In this work we have developed a number of new tools that allow the user to add the effects of space charge to the map of a system. By adding the map to the system we are able to extract information from the system that had only been found analytically for very specific sets of conditions, for any system that we choose. Furthermore, this method can also be used to advance particles in a self-consistent manner that will allow a very fast simulation that includes space charge. This method does have some circumstances where it does not work perfectly; under those conditions a fast multipole method has been implemented to advance the particles.

These tools and methods were used to successfully model the University of Maryland Electron Ring. By using low-energy electrons as proxies for high-energy heavy particles, UMER allows the user to have access to a space-charge-dominant machine on a very flexible schedule. Furthermore, due to its small size and power requirements it has a distinct availability and cost advantage over other accelerators. Finally, since the electrons are low energy, if a potential steering solution is incorrect, the prospects of machine damage or radioactive activation are non-existent. It is also capable of operating beams with space charge regimes significantly beyond current large accelerators; for instance, the Fermilab booster has a Laslett tune shift of 0.3 while the lowest current beam in UMER has a tune shift of 0.5.

In order to fully understand the unique design and dynamics of beams in UMER we had to develop a number of new tools with which to analyze the single- and multiple-particle effects. The low-energy nature of the machine means that the earth's magnetic field is no longer a triviality, and must be treated as a source of 15% of the bending in the system. Based on measurements of the earth's field, a kick-based method was developed to model the effects of the earth's field throughout the ring. Using this model, a series of experiments was run that could not only examine many of the long-term high-order effects of the beam, but could also produce trajectories which match recorded data on the beam.

Once the single-particle dynamics had been examined, it was necessary to look into multiple-particle effects, more specifically space charge. Since the particles in the beam all want to repel each other, the dynamics of the beam will be altered in a number of ways. In order to provide new ways to analyze a beam-based system, we have developed a method of adding the effects of space charge to a beam map. This allows the system to be analyzed using normal form methods which can extract useful information directly from the transfer map. There are a number of other methods to simulate the movements of particles through a system by themselves. The methods developed in this work have allowed us to not only track the movements of test particles, but also to determine the transfer map of the system with space charge included.

Using the new tools developed it was possible to simulate tunes, chromaticities, and long-term emittances, and how they are directly related to the strength of space charge in the system. These quantities are normally measured using analytical approximations that assume that the distribution of the beam always follows the same distribution function. The method developed here allows these quantities to be extracted from the map in a self-consistent manner, adapting to changes in the distribution of the beam. The method also has a degree of parallelizeability that allows for the use of very large numbers of computers to work together in an efficient manner to simulate beams with particle numbers far larger than was previously thought to be feasible.

The method is currently implemented in two dimensions, and has been found to have its maximum accuracy when using a sixteenth-order polynomial series to simulate space charge. Above this the accuracy decreases; therefore, if the distribution function cannot be accurately modeled using a sixteenth-order polynomial, there will be a loss of accuracy. In order to be able to advance particles under these conditions, a differential algebraic version of the fast multipole method has been developed. This method uses differential algebras to simplify the creation of local expansions to help speed up the process.

This work will allow for an increased understanding of the ways that space charge not only evolves the distribution of test particles in a beam, but also how this evolution affects other physical quantities in the beam. This will allow for a greater understanding of multiple-particle dynamics in charged particle beams, as well as self-consistent methods to advance very large numbers of particles.

REFERENCES

[1] J.J. Thompson. Cathode rays. *Philosophical Magazine*, 44(243), 1897.

[2] Walter Henning and Charles Shank, editors. *Accelerators for America's Future.* Department of Energy, 2010.

[3] H. Hsieh, S. Krinsky, A. Luccio, C. Pellegrini, and A. Van Steenbergen. Wiggler, undulator and free electron laser radiation sources development at the national synchrotron light source. *Nuclear Instruments and Methods in Physics Research*, 208(1-3):79 – 90, 1983.

[4] Daniela Schulz-Ertner and Hirohiko Tsujii. Particle radiation therapy using proton and heavier ion beams. *Journal of Clinical Oncology*, 25(8):953–964, 2007.

[5] David B. Williams, C. Barry Carter, David B. Williams, and C. Barry Carter. The transmission electron microscope. In *Transmission Electron Microscopy*, pages 3–22. Springer US, 2009.

[6] A. W. Chao and W. Chou, editors. *Reviews of Accelerator Science and Technology.* World Scientific Publishing Co. Pte. Ltd., 5 Toh Tuck Link, Singapore, 2008.

[7] Hans Frauenfelder and Ernest Henley. *Subatomic Physics.* Prentice-Hall, 1991.

[8] M. Berz. Differential algebraic description of beam dynamics to very high orders. *Particle Accelerators*, 24:109–124, 1989.

[9] John M. Dawson. Particle simulation of plasmas. *Rev. Mod. Phys.*, 55(2):403–447, Apr 1983.

[10] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles.* McGraw-Hill, New York, 1981.

[11] A. Adelmann. *3D Simulations of Space Charge Effects in Particle Beams.* PhD thesis, Paul Scherrer Institut, 2002.

[12] R. A. Kishek, G. Bai, S. Bernal, D. Feldman, T.F. Godlove, I. Haber, P. G. O'Shea, B. Quinn, C. Papadopoulos, M. Reiser, D. Stratakis, K. Tian, C.J. Tobin, and M. Walter. Umer: An analogue computer for dynamics of swarms interacting via long-range forces. *Nuclear Instruments and Methods in Physics Research Section A*, 561(2):266–271, 2006.

[13] R. A. Kishek, G. Bai, S. Bernal, D. Feldman, T.F. Godlove, I. Haber, P. G. O'Shea, B. Quinn, C. Papadopoulos, M. Reiser, D. Stratakis, K. Tian, C.J. Tobin, and M. Walter. Scaled models: Space-charge dominated electron storage rings. *International Journal of Modern Physics A*, 22(22):3838–3851, 2007.

[14] S. Bernal, B. Beaudoin, T. Koeth, M. Cornacchia, D. Sutter, K. Fiuza, I. Haber, R.A. Kishek, C. Wu, C. Papadopoulos, M. Reiser, and P. G. O'Shea. Resonance phenomena over a broad range of beam intensities in an electron storage ring. In *Proceedings of the 2009 IEEE Particle Accelerator Conference*, 2009.

[15] C. F. Papadopoulos, R. Fiorito, P.G. O'Shea, A. Shkvarunets, M. Conde W. Gai, and J. G. Power. Determination of true rms emittance from otr measurements. In *Proceedings of the 2009 IEEE Particle Accelerator Conference*, 2009.

[16] D. Sutter, S. Bernal, C. Wu, M. Cornacchia, B. Beaudoin, K. Fiuza, I. Haber, R.A. Kishek, M. Reiser, and P.G. O'shea. Coherent phenomena over a range of beam intensities in the storage ring umer. In *Proceedings of the 2009 IEEE Particle Accelerator Conference*, 2009.

[17] L. G. Vorobiev, X. Wu, and R.C. York. Single-particle beam dynamics studies for the university of maryland electron ring. In *Proceedings of the 1999 Particle Accelerator Conference*, pages 3116–3118, 1999.

[18] R. A. Kishek, S. Bernal, C. L. Bohn, D. Grote, I. Haber, H. Li, P. G. O'Shea, M. Reiser, and M. Walter. Simulations and experiments with space-charge-dominated beams. *Physics of Plasmas*, 10(5):2016–2021, 2003.

[19] I. Haber, S. Bernal, B. Beaudoin, M. Cornacchia, D. Feldman, R. B. Feldman, R. Fiorito, K. Fiuza, T.F. Godlove, R. A. Kishek, P. G. O'Shea, B. Quinn, C. Papadopoulos, M. Reiser, D. Stratakis, D. Sutter, J. C. T. Tangaraj, K. Tian, M. Walter, and C. Wu. Scaled electron studies at the university of maryland. *Nuclear Instruments and Methods in Physics Research Section A*, 606:64–68, 2009.

[20] E. Nissen and B. Erdelyi. Modeling single particle dynamics in low energy and small radius accelerators. In *Proceedings of the 10th International Computational Particle Accelerator Conference*, 2009.

[21] E. Nissen and B. Erdelyi. Method to extract transfer maps in the presence of space charge in charged particle beams. In *Proceedings of the 2010 International Particle Accelerator Conference*, 2010.

[22] E. Nissen and B. Erdelyi. A new paradigm for modeling, simulation, and analysis of intense beams. In *Proceedings of the 46th ICFA Advanced Beam*

Dynamics Workshop on High Intensity and High Brightness Hadron Beams, 2010.

[23] A. J. Dragt. Lie methods for nonlinear dynamics with applications to accelerator physics, 2010. Unpublished.

[24] E. D. Courant and H. S. Snyder. Theory of the alternating-gradient synchrotron,. *Annals of Physics*, 3(1):1 – 48, 1958.

[25] M. Berz. *Advances in Imaging and Electron Physics vol 108.* Academic Press, London, 1999.

[26] K. Makino and M. Berz. Cosy infinity version 9. *Nuclear Instruments and Methods A*, 558:346–350, 2005.

[27] B. Erdelyi. *Symplectic Approximation of Hamiltonian Flows and Accurate Simulation of Fringe Field Effects.* PhD thesis, Michigan State University Center for Dynamical Systems and Mathematical Physics, (2001).

[28] S. Blanes, F. Casas, J.A. Oteo, and J. Ros. The magnus expansion and some of its applications. *Physics Reports*, 470(5-6):151 – 238, 2009.

[29] S. Blanes, F. Casas, and Ander Murua. Splitting and composition methods in the numerical integration of differential equations. *Boletin de la Sociedad Española de Mathemática Aplicada*, (45):89 – 145, 2008.

[30] A. Bátkai, P. Csomós, B. Farkas, and Gregor Nickel. Operator splitting for non-autonomous evolution equations. arXiv:1002.0164v2, 2010.

[31] H. F. Baker. On the integration of linear differential equations. In *Proceedings of the London Mathematical Society*, pages 333–378, 1902.

[32] J. E. Campbell. On the theory of simultaneous partial differential equations. In *Proceedings of the London Mathematical Society*, pages 235–263, 1898.

[33] S. Bernal, D. Sutter, M. Cornacchia, B. Beaudoin, I. Haber, R.A. Kishek, M. Reiser, C. Wu, and P.G. O'Shea. Operational studies of the 10 kev electron storage ring umer. In *Proceedings of the 13th workshop on Advanced Accelerator Concepts (AAC)*, pages 738–743. New York: AIP Press, 2009.

[34] M. Venturini et al. Single-particle dynamics study for a university of maryland electron ring. *Fusion Engineering and Design*, 32-33:283–291, 1996.

[35] M. Venturini. *Lie Methods, Exact Map Computation, and the Problem of Dispersion in Space Charge Dominated Beams.* PhD thesis, University of Maryland Physics Department, 1998.

[36] R.A. Kishek. Umer beam parameters. Technical Report UMER-09-0731-RAK-v2, UMER, 2010.

[37] S. Bernal. New hard-edge model for umer solenoid. Technical Report 060306-SB, UMER, 2006. Unpublished.

[38] M. Berz and K. Makino. Cosy infinity beam physics manual. Technical Report MSUHEP-60803, Department of Physics and Astronomy, Michigan State University, 2006.

[39] X. Huang, S.Y. Lee, K.Y. Ng, and Y Su. Emittance measurement and modeling for the fermilab booster. *Phys. Rev. ST Accel. Beams*, 9:014202, 2006.

[40] M. G. Bulmer. *Principles of Statistics.* MIT Press, Cambridge Massachusettes, 1967.

[41] Bruce G. Lindsay and Prasanta Basak. Moments determine the tail of a distribution (but not much else). *The American Statistician*, 54(4):pp. 248–251, 2000.

[42] W.H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C++*. Cambridge University Press, New York, 2002.

[43] M. G. Duffy. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM Journal on Numerical Analysis*, 19(6):1260–1262, 1982.

[44] M. Berz and K. Makino. Cosy infinity programmer's manual. Technical Report MSUHEP-60803, Department of Physics and Astronomy, Michigan State University, 2006.

[45] Y.K. Kim and M. Berz. Parallel constructs in cosy infinity. Technical Report MSUHEP-060805, Michigan State University, 2006. Unpublished.

[46] J. Carrier et. al. A fast adaptive multipole algorithm for particle simulations. *Siam J SCI. STAT. COMPU.*, 9(4), July 1988.

[47] Barry Cipra. The best of the 20th century: Editors name top 10 algorithms. *SIAM News*, 33(4), 2000.

[48] L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, Philadelphia, 1997.

[49] M. Reiser. *Theory and Design of Charged Particle Beams*. WILEY-VCH Verlag GmbH & Co., Weinheim, 2008.

[50] E. Nissen and B. Erdelyi. Differential algebraic methods for single particle dynamics studies of the university of maryland electron ring. *Phys. Rev. ST Accel. Beams*, 13(7):074001, Jul 2010.

[51] D. Sutter. Private communication, 2008.

[52] E. Nissen, B. Erdelyi, S. Bernal, and D. Sutter. Single particle dynamics in the university of maryland electron ring. In *Proceedings of the 2009 IEEE Particle Accelerator Conference*, 2009.

[53] C. Wu, E. H. Abed, B. Beaudoin, S. Bernal, K. Fiuza, I. Haber, R. A. Kishek, P. G. O'Shea, M. Reiser, and D. Sutter. A novel beam steering algorithm using orbit response matrix. In *Proceedings of the 2009 IEEE Particle Accelerator Conference*, 2009.

[54] P. Bryant and K. Johnson. *The Principles of Circular Accelerators and Storage Rings*. Cambridge University Press, 1993.

[55] D. Stratakis. *Tomographic Measurement of the Phase Space Distribution of a Space-Charge-Dominated Beam*. PhD thesis, University of Maryland, 2008.

APPENDIX A

SOFTWARE USE

A.1 Introduction

This is a guide to the use of the COMFY expansion to COSY Infinity 9.0. This expansion allows the user to calculate the transfer map of a beam system which includes the effects of space charge. This operates by sending a group of test particles through the system, converting their distribution into a charge density function, and then calculating the electric fields for the beam. These electric fields are added to the map. Furthermore this expansion also includes a fixed-level fast multipole method to advance particles under circumstances that might be difficult for the map itself.

A.2 Setup and First Use

In order to use COMFY for the first time, simply take the folder you wish to use your code in, and make sure it contains COSY.bin. Then install the file COMFY.fox as well as the folder COMFYFILES. First compile COMFY.fox. Now when you use your program, write INCLUDE 'COMFY'; as the first line, and you are ready to begin.

A.3 Using COMFY

We will compare the use of a basic COSY code to one that uses COMFY and then discuss the difference. A generic COSY simulation takes the form,

```
INCLUDE 'COSY';
```

```
procedure run;

OV 6 2 0;

RPE .1; um;

DL .2;

pm 6;

endprocedure; run; end;
```

whereas the space charge version takes the form,

```
INCLUDE 'COMFY';

procedure run;

variable radii 1 2;

variable scratch 1 2;

COV 6 2 0 16;

RPE .1; um;

radii(1):=.01; radii(2):=.01;

scratch(1):=.005; cratch(2):=.005;

comfyinit 1 30000 .005 radii scratch;

CDL .2;

pm 6;

endprocedure; run; end;
```

The first difference is the use of COV instead of OV to set the order and variables.

`OV NO NV NP;`

uses NO number of orders,NV number of dimensions, and NP number of parameters.

`COV NO NV NP OM;`

uses the same NO, NV and NP, but OM is the order that the space charge distribution will be calculated to. If you want the order of the transfer map and the order of the space charge model to be the same, then COV is not necessary; merely use OV as you normally would. However, the space charge model is most accurate at 16^{th} order.

The next difference is the use of the line:

```
COMFYINIT <distribution type> <particle number>
<distribution radius> <aperture array> <moment order>
<current> <scratchspace>
```

The entries and their meanings are given in Table A.1.

Table A.1: The entries used in the COMFYINIT command, as well as descriptions of what they do.

Entry	Description
`<distribution type>`	Determines the distribution used in the simulation, options are discussed in A.4.
`<particle number>`	This is the number of particles, above 1,000,000 particles there will be no increase of accuracy, above 50,000 particles the hard disk must be used causing an increase in the amount of time required. For fitting or large machines smaller numbers should be used.
`<distribution radius>`	This is the range within which the distribution is populated, see A.4.
`<aperture array>`	This is a two dimensional array giving the size of the beam pipe for particle removal. Positive values give a rectangle, while negative values give an ellipse.
`<moment order>`	This is the order of moments that are calculated for the space charge model.
`<current>`	this is the beam current, COMFY currently models an infinitely long beam, so peak current is used.
`<scratchspace>`	This is used for some of the distributions (see A.4).

If you are restarting a section over and over again (i.e., as part of an optimization routine) you can use the procedure:

```
COMFYREINIT  <distribution type> <particle number>
 <distribution radius> <aperture array>  <current> <scratchspace>
```

Finally we replace the element DL with the element CDL. CDL uses the space charge algorithms for a list of supported elements; see Section A.5. There are a number of operating modes which are discussed in Section A.6.

A.4 Distribution Options

There are multiple actions that can create the initial conditions of the beam. If you have a set of initial conditions that you wish to use put them in a file with x,a,y,b on each line separated by a single space (COSY does not recognize tabs) then in the `<distribution type>` section of the COMFYINIT line put the string literal of the filename. If you do this distribution radius and scratchspace are not called.

A.4.1 Integer Distributions

The basic distributions are given in Table A.2. There is also an outlier, that is distribution 0. In this case the `<initrad>` would take the value $R_x\&R_a\&R_y\&R_b$, while scratch would be a DA vector that contains the desired distribution function. If you wish to have a zero value for one of the elements, simply zero out its radius.

Table A.2: The different possible integer distributions and their descriptions.

number	name	scratch
1	Uniform ellipse	Array with two entries containing the x and y semi-axes.
2	Gaussian	Array with two entries containing σ_x and σ_y.
3	Parabolic	Array with two entries containing the limits of x and y.

A.4.2 Two-Integer Distributions

If you wish to have a different distribution for the position distribution from the angle distribution, then combine the two. If we wanted to have a distribution that was uniform in xy and Gaussian in ab we would use the following line:

`COMFYINIT 1&2 nparticles` $R_{xy}\&R_{ab}$ `aperturearray moment`

`order current scratch`

where scratch is a four-part array:

scratch(1)=xradius

scratch(2)=σ_a

scratch(3)=yradius

scratch(4)=σ_b

A.4.3 Four-Integer Distribution

This allows for different distributions for all four distributions. We would use it in this manner (allocate memory accordingly):

`COMFYINIT 0&1&2&3 nparticles` $R_x\&R_a\&R_y\&R_b$ `aperturearray moment`

`order current scratch`

scratch(1)=DA vector

scratch(2)=a radius

scratch(3)=σ_y

scratch(4)=b outer radius

A.5 Supported Elements

In Table A.5 we see the COSY element and the COMFY element; they both take the same arguments. If you want to use an element that is not available, a stand-alone kick has been implemented with the routine,

```
UPSKICKER length h positionsin positionsout
```

where length is the length over which you wish to kick, h is the inverse of the radius of curvature, and positionsin and positionsout are deprecated.

Table A.3: Table of supported COMFY elements and their COSY equivalents.

COMFY element	COSY element
CDL	DL
CMQ	MQ
CMH	MH
CMO	MO
CMD	MD
CMZ	MZ
CM5	M5
CMM	MM
CMS5	MS
CDP	DP
CDI	DI

A.6 Operating Modes

There are currently five operating modes that are supported using COMFY. The first is the default mode, where COMFY will advance the particles through the system and calculate the space charge kick. If the procedure,

```
USEFMM nbins ncoef chargemass
```

is called, the effects of space charge will be applied to the system using the fast multipole method, while the map will still be calculated using the moment method. For more information, see Section A.8. If the procedure,

```
COMFYOFF
```

is called, the simulation will revert to simple COSY elements, no particles will be advanced, and space charge will not play a role in the system. This could be used if you wanted to compare the maps of a given system with space charge and without. To begin advancing particles again, call the procedure,

```
COMFYON
```

If you would like to advance the particles without using the space charge map calculation, you can call the procedure,

```
COMFYPAUSE
```

To start calculating the space charge map again, use the procedure,

```
COMFYUNPAUSE
```

Finally, if you are interested in tracking the particles using the FMM you can call the procedure,

```
COMFYFMMTRACK
```

after the `COMFYINIT` line. This will advance particles using the space charge determined by the FMM but will not calculate the space charge map.

A.7 Parallel COMFY

In order to use COMFY in parallel, the bin file `COMFY_MPI` has been created. You will see significant time savings when dealing with the map method. Please note that it currently does not support the FMM method in parallel.

The only difference has to do with input and output files. If you are using an input file you should make one for each processor core you are using, with its number appended to the beginning, and with the correct number of particles for each node. For example, if I want to use 40,000 particles on four nodes I would make four files with 10,000 particles each, with the names, '1initial.dat' '2initial.dat' '3initial.dat' '4initial.dat' then I would initialize COMFY with the line,

```
COMFYINIT 'initial.dat' nparticles 0 aperturearray
 moment order current 0
```

with the other entries what they would be normally. Second, the output files are a little different, but the command is the same:

```
OUTPUTFILE Filename suffix;
```

As in the previous case you will get four files, that are named, 1Filename.suffix 2Filename.suffix 3Filename.suffix 4Filename.suffix At this point I just use a short program to stitch them all together into one.

A.8 FMM Use

The fast multipole method used here is a method of calculating the potential and electric fields that will work for any distribution. In order to use this method, before the COMFYINIT line you must use,

```
USEFMM nbins ncoef chargemass
```

where nbins is the number of bins on a side, and ncoef is the number of coefficients to calculate the DA vector (note, you must declare the order for COSY to be greater than or equal to this number). Chargemass is a method for involving multiple species in the simulation; inputting the number 0 will use the species defined as the reference particle. If you wish to use a set of predetermined particles, first you must add a parameter in the OV procedure. COMFY automatically uses para(1) for this method. Then you must use the RPR procedure to define the particle by its magnetic rigidity. The line should look like this:

`RPR` χ_m`*para(1)` M_0 Z_0`;`

Next put the string literal for a file which contains in space separated columns:

(Charge per Macroparticle [in elementary charge units])

(Particle energy in MeV)

(particle mass [in amu])

(Particle charge)

As an example, if we wanted a distribution of carbon 12 at 10MeV where each macroparticle represents 1000 atoms, the line would read,

```
6000 10 12 6
```

If we wanted some carbon 14, it would read,

```
6000 10 14 6
```

It should be noted that since this method uses a Taylor expansion the difference should not be large if magnetic elements are desired.

These are read separately from the initial conditions so, if you want a random spread of particles to go along with the separately assigned particles, this can be done. Simply make sure there are enough entries for the particles you want. Please note that this significantly increases the amount of time required for each kick.

APPENDIX B

IMAGE CHARGE KICK

In order to properly account for the effects of image forces on the beam it becomes important to determine the magnitude of the force in order to apply a kick. We assume a perfectly conducting straight cylindrical beam pipe, and a beam which has cylindrical symmetry, with a constant longitudinal charge density that is long enough for end effects to be neglected; see Figure B.1. This is a good approximation for UMER. Using a cylindrical coordinate system, the differential equation is of the form:

$$\ddot{\xi} - \frac{q\lambda}{2\pi\epsilon_0 m}\frac{\xi}{R^2 - \xi^2} = 0, \tag{B.1}$$

where ξ is the offset from the center of the beam pipe, R is the radius of said beam pipe, q is the charge of an electron, $\lambda < 0$ is the charge density along the beam pulse, m is the mass of the electron and ϵ_0 is the electric permittivity. This models the forces on the centroid of the beam.

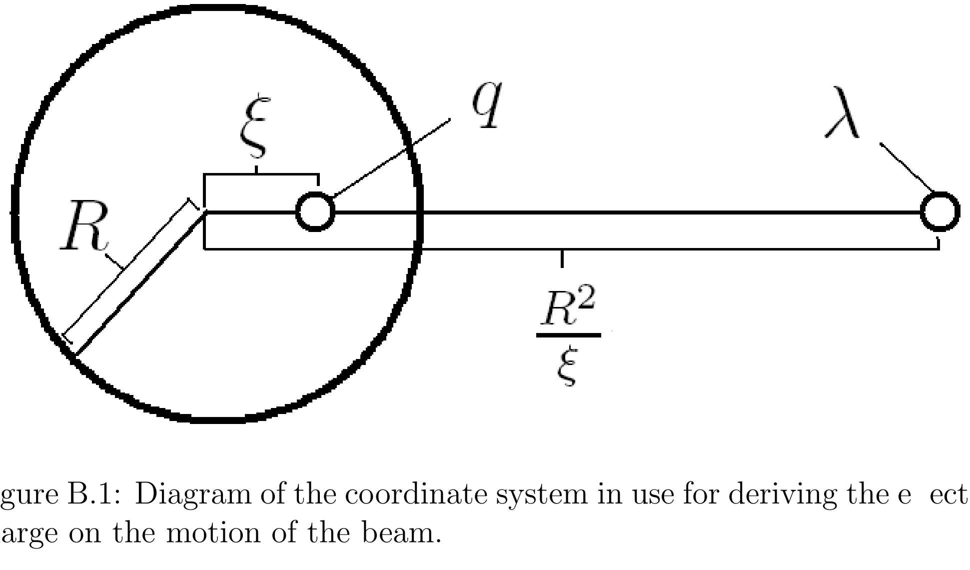


Figure B.1: Diagram of the coordinate system in use for deriving the effects of image charge on the motion of the beam.

Since the differential equation is of the form $\xi''+G'(\xi)=0$ it has the conservation law,

$$\frac{1}{2}\xi'^2+G(\xi)=E. \tag{B.2}$$

We assume $\xi'=0$ at $\xi=0$, since the image charge in a cylinder only affects off-center beams. This leads to

$$\xi' \;=\; \pm\sqrt{-\frac{q\lambda}{2\pi\epsilon_0 m}\ln(\frac{R^2-\xi^2}{R^2})}, \tag{B.3}$$

where ξ is the magnitude of the displacement and equal to $\sqrt{x^2+y^2}$. Using a, b to represent the angles while assuming that the particles are non-relativistic, we get their kick strength equal to

$$\Delta a \;=\; \frac{x}{v_0\sqrt{x^2+y^2}}\sqrt{-\frac{q\lambda}{2\pi\epsilon_0 m}\ln(\frac{R^2-(x^2+y^2)}{R^2})}, \tag{B.4}$$

$$\Delta b \;=\; \frac{y}{v_0\sqrt{x^2+y^2}}\sqrt{-\frac{q\lambda}{2\pi\epsilon_0 m}\ln(\frac{R^2-(x^2+y^2)}{R^2})}. \tag{B.5}$$

Now we derive the equation of motion for an individual (test) electron in the beam so that the image force can be applied to an arbitrary set of initial conditions. So we define the variable Υ as the line between the particle in question and the image charge; see Figure B.2.

Once again we get that

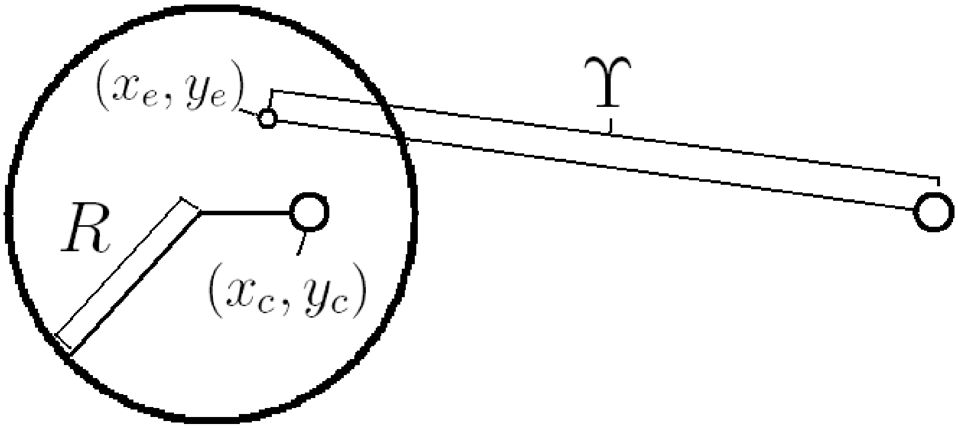


Figure B.2: Diagram of the coordinate system in use for deriving the effects of image charge on a test electron in the beam.

$$\ddot{\Upsilon} - \frac{q\lambda}{2\pi\epsilon_0 m}\frac{1}{\Upsilon} = 0. \tag{B.6}$$

This is solved in a similar manner to the centroid problem:

$$\dot{\Upsilon} = \sqrt{2C + \frac{q\lambda}{\pi\epsilon_0 m}\ln(\Upsilon)}. \tag{B.7}$$

Now we define the coordinates of the image charge and the particle. The position of the image charge is based on the location of the beam centroid with coordinates (x_c, y_c), and the location of the test electron will be (x_e, y_e). This leads to

$$\Upsilon_x = \frac{R^2 x_c}{\xi^2} - x_e, \tag{B.8}$$

$$\Upsilon_y = \frac{R^2 y_c}{\xi^2} - y_e, \tag{B.9}$$

where in the notation just introduced $\xi = \sqrt{x_c^2 + y_c^2}$. This entails

$$\sqrt{\Upsilon_x^2 + \Upsilon_y^2} \;=\; \sqrt{(\frac{R^2}{\xi})^2 - 2\frac{R^2}{\xi}(\frac{x_e x_c + y_e y_c}{\xi}) + x_e^2 + y_e^2}. \tag{B.10}$$

Inserting this into (B.7) we obtain

$$\dot{\Upsilon} \;=\; \sqrt{C + \frac{q\lambda}{\pi\epsilon_0 m}\ln(\sqrt{(\frac{R^2}{\xi})^2 - 2\frac{R^2}{\xi}(\frac{x_e x_c + y_e y_c}{\xi}) + x_e^2 + y_e^2})}. \tag{B.11}$$

To determine the value of C, we require that if the test electron coincides with the centroid of the beam, $x_e = x_c, y_e = y_c$, then $\dot{\Upsilon} = \dot{\xi}$. The solution for C is re-inserted in the equation, where we apply the unit vectors to get the direction of the kick, and we scale it to the particle optical coordinates a and b. Finally, we obtain

$$x(s) \;=\; x_i, \tag{B.12}$$

$$a(s) \;=\; a_i + \frac{\frac{R^2}{\xi}x_c - \xi x_i}{\sqrt{R^4 - 2R^2(x_i x_c + y_i y_c) + (x_i^2 + y_i^2)\xi^2}} \times \times \sqrt{\frac{q\lambda}{2\pi\epsilon_0 m}\ln(\frac{R^6 - 2R^4(x_i x_c + y_i y_c) + (x_i^2 + y_i^2)R^2\xi^2}{(R^2 - \xi^2)^3})}\frac{1}{v_0}, \tag{B.13}$$

$$y(s) \;=\; y_i, \tag{B.14}$$

$$b(s) \;=\; b_i + \frac{\frac{R^2}{\xi}y_c - \xi y_i}{\sqrt{R^4 - 2R^2(x_i x_c + y_i y_c) + (x_i^2 + y_i^2)\xi^2}} \times \times \sqrt{\frac{q\lambda}{2\pi\epsilon_0 m}\ln(\frac{R^6 - 2R^4(x_i x_c + y_i y_c) + (x_i^2 + y_i^2)R^2\xi^2}{(R^2 - \xi^2)^3})}\frac{1}{v_0}. \tag{B.15}$$

The kick $\mathcal{K}$ is given by evaluating at $s = \ell$.

From a programming perspective, it is possible, when fitting, for ξ to become greater than R. For this reason the software was written so that if ξ was greater than R the procedure would return a zero kick, and the objective function for the kick would be increased by a set amount.

APPENDIX C

PARALLEL SCRIPT

In order to properly execute a simulation in parallel it is necessary to use a proper script to set up the environment. An example of the script most often used is given.

```
#!/bin/bash
if [ "0$1" == "0" ]; then
 echo "Usage: $0 nproc"
 exit 0;
else nnproc=$1
nproc= ‘expr $1 / 2‘
echo $nnproc
mpdboot -v -f machines.Linux -n $nproc
mpdtrace -l
time mpiexec -n $nnproc mpicosy >> "Timing.dat"
mpdallexit
fi
```

As can be seen in the example, the mpicosy routine is called using twice as many CPUs have been set aside as have nodes that have been activated. This is because there are two CPUs per node, and the MPI exec function launches threads; that way two executions will be put on each node, thus using both CPUs. This type of script can then be placed into a larger script, as is seen in this example:

```
#!/bin/bash
echo 3 > "Timing.dat"
for j in 1000000 #2000000 3000000 4000000
do
for i in 50 #48 50 52 54 56 #2 4 8 10 18
```

```
do
nproc=$i
echo $j >"particlenumber.dat"
echo $j >>"Timing.dat"
echo $i $j
./vstartII.bat $nproc
done
done
```

These scripts are designed to be used with the NICADD Beowulf cluster; it contains 56 dual core processor nodes connected through a 100Mbit network. It uses Fedora Linux, and its nodes use AMD Athlon 2400+ with 1GB of RAM per node.

APPENDIX D

SIMULATING THE UNIVERSITY OF MARYLAND ELECTRON RING

A series of modifications had to be made in order to properly simulate UMER. In order to ease programming, the ring was divided up into repeating sections. In order to properly start and stop at the regions which contain the beam position monitors and phosphor screens, these sections were split into two. An example of this is given.

```
procedure MNSEC1 I1 I2 D1 D2 theta VD1 VD2 ind value maggy label;
variable dummy 1;
MFD .05183 theta .04726 maggy(1) 6;
MDL .00235 maggy(2) 6;
if ind=1; dummy:=value;
elseif ind#1; dummy:=I1;
endif;
MMQ .05164 dummy .0295 maggy(3) 15;
MDL .03508 maggy(4) 6;
MBD .0382 .02872 D1 VD1 maggy(5) 6;
MDL .03508 maggy(6) 6;
if ind=2; dummy:=value;
elseif ind#2; dummy:=I2;
endif;
MMQ .05164 dummy .0295 maggy(7) 15;
MDL .02718 maggy(8) 6;
endprocedure;
```

The elements MFD, and MBD are flange dipoles and bending dipoles with the earth's magnetic field included; MDL and MMQ are drifts and quadrupoles with the earth's magnetic field included. The term “maggy” is an array which contains the string that names the file which contains the magnetic field data for the given element. As an example of how this is implemented, an example of the bending dipole is shown.

```
procedure MBD L D dtheta atheta file order;
variable i 1; variable j 1; variable k 1;
variable data 1 3; variable input 1;
variable R 1;
variable ell 1;
ell:=LSYS;
R:=((180*L)/((abs(dtheta-atheta)+.0000000000001)*PI));
openf 20 file 'OLD';{ Here file is the string contained
 in the array maggy from the previous set of code.}
if dtheta<0;
CB;
endif;
i:=order;
loop j 1 i;{ This loops over the number of kicks.}
loop k 1 3;{ This loops over the number of field
 elements contained.}
read 20 input;{This reads in each magnetic field number}
input:=input*.0000001;{This scales it to S.I. units}
data(k):=input;
endloop;
DP R abs(dtheta-atheta)/(2.0*i) D;
if j=1;
LSYS:=0;
```

```
endif;
MKICK L/i data(1) data(2) data(3) D i/L;
if j=order;
LSYS:=ell;
endif;
DP R abs(dtheta-atheta)/(2.0*i) D;
endloop;
if dtheta<0;
CB;
endif;
closef 20;
TA -atheta 0.0;
SA -L*(tan(((atheta)*PI)/360.0)) 0.0;
endprocedure;
```

There are other sections for the injection and re-circulation lines which also include the effects of image charges that are outlined in B. If we wanted to simulate an entire turn of the ring from cathode to the end of the recirculation section the code would look like this:

```
STRAIGHT1 SOL SD1 IQ1 SD2 IQ2 SD3 IQ3 IQ4;
STRAIGHT2 SOL SD1 SD1V IQ1 SD2 SD2V IQ2 SD3 SD3V IQ3 IQ4;
STRAIGHT3 SOL SD1 IQ1 SD2 IQ2 SD3 IQ3 IQ4;


MINJY SD4V IQ5 SD5 SD5V IQ6 SD6 YQD -pdvq QR1 YSEC1 obj;


MNSEC1 curr(1) curr(2) 10.0 10.0 vert(1) steer(1) steer(2)
```

```
 switch(1) supp(1) NSEC1 'RC1';
MNSEC2 curr(3) curr(4) 10.0 10.0 vert(1) steer(1) steer(2)
 switch(1) supp(1) NSEC1 'RC1';

MNSEC1 curr(5) curr(6) 10.0 10.0 vert(2) steer(3) steer(4)
 switch(2) supp(2) NSEC2 'RC2';
MNSEC2 curr(7) curr(8) 10.0 10.0 vert(2) steer(3) steer(4)
 switch(2) supp(2) NSEC2 'RC2';

MNSEC1 curr(9) curr(10) 10.0 10.0 vert(3) steer(5) steer(6)
 switch(3) supp(3) NSEC3 'RC3';
MNSEC2 curr(11) curr(12) 10.0 10.0 vert(3) steer(5) steer(6)
 switch(3) supp(3) NSEC3 'RC3';

MNSEC1 curr(13) curr(14) 10.0 10.0 vert(4) steer(7) steer(8)
 switch(4) supp(4) ISEC1 'RC4';
MNSEC2 curr(15) curr(16) 10.0 10.0 vert(4) steer(7) steer(8)
 switch(4) supp(4) ISEC1 'RC4';

MNSEC1 curr(17) curr(18) 10.0 10.0 vert(5) steer(9) steer(10)
 switch(5) supp(5) NSEC4 'RC5';
MNSEC2 curr(19) curr(20) 10.0 10.0 vert(5) steer(9) steer(10)
 switch(5) supp(5) NSEC4 'RC5';

MNSEC1 curr(21) curr(22) 10.0 10.0 vert(6) steer(11) steer(12)
 switch(6) supp(6) NSEC5 'RC6';
```

```
MNSEC2 curr(23) curr(24) 10.0 10.0 vert(6) steer(11) steer(12)
 switch(6) supp(6) NSEC5 'RC6';

MNSEC1 curr(25) curr(26) 10.0 10.0 vert(7) steer(13) steer(14)
 switch(7) supp(7) NSEC6 'RC7';
MNSEC2 curr(27) curr(28) 10.0 10.0 vert(7) steer(13) steer(14)
 switch(7) supp(7) NSEC6 'RC7';

MNSEC1 curr(29) curr(30) 10.0 10.0 vert(8) steer(15) steer(16)
 switch(8) supp(8) NSEC7 'RC8';
MNSEC2 curr(31) curr(32) 10.0 10.0 vert(8) steer(15) steer(16)
 switch(8) supp(8) NSEC7 &'RC8';

MNSEC1 curr(33) curr(34) 10.0 10.0 vert(9) steer(17) steer(18)
 switch(9) supp(9) NSEC8 'RC9';
MNSEC2 curr(35) curr(36) 10.0 10.0 vert(9) steer(17) steer(18)
 switch(9) supp(9) NSEC8 'RC9';

MNSEC1 curr(37) curr(38) 10.0 10.0 vert(10) steer(19) steer(20)
 switch(10) supp(10) ISEC2 'RC10';
MNSEC2 curr(39) curr(40) 10.0 10.0 vert(10) steer(19) steer(20)
 switch(10) supp(10) ISEC2 'RC10';

MNSEC1 curr(41) curr(42) 10.0 10.0 vert(11) steer(21) steer(22)
 switch(11) supp(11) NSEC9 'RC11';
MNSEC2 curr(43) curr(44) 10.0 10.0 vert(11) steer(21) steer(22)
```

```
 switch(11) supp(11) NSEC9 'RC11';

MNSEC1 curr(45) curr(46) 10.0 10.0 vert(12) steer(23) steer(24)
 switch(12) supp(12) NSEC10 'RC12';
MNSEC2 curr(47) curr(48) 10.0 10.0 vert(12) steer(23) steer(24)
 switch(12) supp(12) NSEC10 'RC12';

MNSEC1 curr(49) curr(50) 10.0 10.0 vert(13) steer(25) steer(26)
 switch(13) supp(13) NSEC11 'RC13';
MNSEC2 curr(51) curr(52) 10.0 10.0 vert(13) steer(25) steer(26)
 switch(13) supp(13) NSEC11 'RC13';

MNSEC1 curr(53) curr(54) 10.0 10.0 vert(14) steer(27) steer(28)
 switch(14) supp(14) NSEC12 'RC14';
MNSEC2 curr(55) curr(56) 10.0 10.0 vert(14) steer(27) steer(28)
 switch(14) supp(14) NSEC12 'RC14';

MNSEC1 curr(57) curr(58) 10.0 10.0 vert(15) steer(29) steer(30)
 switch(15) supp(15) NSEC13 'RC15';
MNSEC2 curr(59) curr(60) 10.0 10.0 vert(15) steer(29) steer(30)
 switch(15) supp(15) NSEC13 'RC15';

MNSEC1 curr(61) curr(62) 10.0 10.0 vert(16) steer(31) steer(32)
 switch(16) supp(16) ISEC3 'RC16';
MNSEC2 curr(63) curr(64) 10.0 10.0 vert(16) steer(31) steer(32)
 switch(16) supp(16) ISEC3 'RC16';
```

```
MNSEC1 curr(65) curr(66) 10.0 10.0 vert(17) steer(33) steer(34)
 switch(17) supp(17) NSEC14 'RC17';
MNSEC2 curr(67) curr(68) 10.0 10.0 vert(17) steer(33) steer(34)
 switch(17) supp(17) NSEC14 'RC17';

MIYSEC curr(69) curr(70) QR1 YQD 10.0 steer(35) pdv 0 SD6 0.0 YSEC1
 ofs tlt obj;
```

The first three sections give the injection and matching line, and `MINJY` is the injection side of the Y-section. The seventeen ring sections are then shown, followed by the element `MIYSEC` which is the recirculation side of the Y-section.