RIGOROUS ANALYSIS OF NONLINEAR MOTION IN PARTICLE ACCELERATORS

By

Kyoko Makino

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Physics and Astronomy

1998

ABSTRACT

RIGOROUS ANALYSIS OF NONLINEAR MOTION
IN PARTICLE ACCELERATORS

By

Kyoko Makino

Methods to obtain rigorous descriptions for the motion of ensembles of particles relative to a reference curve are developed. To account for the most general reference curves, a Lagrangian and Hamiltonian formulation of the relativistic motion of charged particles in electromagnetic fields in three dimensional curvilinear coordinates including torsion and with arclength as independent variable is derived. To allow for the use of the most general fields of particle optical devices, a wavelet-based method to include measured field data in the equations of motion in an appropriate manner is discussed.

The method of transfer maps is a powerful tool for the study of weakly nonlinear dynamical systems, especially for the case of large phase space acceptances as in modern particle spectrographs, and the intricate dynamical behaviour of repetitive systems as in circular accelerators and storage rings. The Differential Algebraic (DA) techniques have proven fruitful for various computational problems in beam physics, including the determination of high order Taylor transfer maps.

A new approach, the Remainder-enhanced Differential Algebraic (RDA) method, is presented, which extends the method to allow the determination of remainder bounds for functional dependencies and solutions of ODEs. First, the basic theory of the method is developed and applied to problems of verified optimization and quadrature. Next, schemes are derived that allow the construction of numerical integrators of arbitrary order with rigorous verification of the error for both the integration of individual initial conditions as well as Taylor transfer maps. The methods are based on a differential algebraic fixed point problem which is studied using Schauder's theorem and other functional analysis tools. Employing various compactness arguments on suitable function spaces in combination with the RDA tools, in each integration step a proof of existence of a solution within a tight inclusion is performed.

The resulting computational tools are implemented in the arbitrary order beam physics code COSY INFINITY, and their behavior and performance is studied. Using integration orders around ten and suitable step sizes, rigorous remainder bounds in the range of 10^{-10} for transfer maps of orders around ten are obtained.

To the ultimate truth and those who love it

ACKNOWLEDGMENTS

Seeing my Ph.D. thesis work summarized in front of me, my thoughts float around in the memories of these last years. Probably merely a short period of one's life, it has been a dramatical one for me, who had not imagined to work in the academic society again.

The chance was opened by my academic advisor Prof. Martin Berz, who believes in the nobility of science, and who makes great efforts to support and encourage his students. While I had been exposed to scientific research earlier as a graduate student in high energy physics and afterward as a staff scientist working on computer codes, it was quite shocking to me when I saw the way Prof. Berz attacks a scientific question. In 1995 we were working on the problem of the relativistic equation of motion of spin. Prof. Berz questioned the meaning of the first component of the spin four vector, challenging us by asking whether he even could "put his grandmother" there. It then turned out that the axiomatic foundation of this time-like component is indeed rather shaky, and I had to realize that such a pure critical attitude is an important catalyst for real progress in science. The riddle of the spin four-vector continues to have several open subtleties, and unfortunately my dissertation does not discuss this interesting topic.

I would like to use the occasion of the completion of my thesis work to thank, first of all, Prof. Berz who has guided me with pure scientific rigor through various interesting topics, as well as various practical opportunities to gain exposure to the wider scientific community at conferences and other occasions. My respect is not only tied to his scientific rigor, but also to his continuous support and warm encouragement of his students.

In the period of the spin, we were having greatly exciting discussions in the group every day, and it was one of the wonderful moments in my life. The enjoyment of exciting scientific discussions is only possible with the right kind of people. I would like to express my appreciation to those people who share their time in the group of Prof. Berz with me: Dr. Ralf Degenhardt, Dr. Georg Hoffstätter, Dr. Weishi Wan, Silke Rolles, Dr. Vladimir Balandin, Dr. Nina Golubeva and Meng Zhao, in the past; and at the moment, Khodr Shamseddine, Jens Hoefkens, Bela Erdelyi, Lars Diening, Jens von Bergmann and Michael Lindemann. My recent thanks go to a long-term warm friendship with Khodr Shamseddine, and mighty help from Jens Hoefkens.

My life at Michigan State University is supported and encouraged also by those professors who kindly have served on my advisory committee; Prof. Jerzy Borysowicz, Prof. Julius S. Kovacs, Prof. Jerry Nolen at Argonne, Prof. Jon Pumplin and Prof. Brad Sherrill. My thesis work has received substantial help from various people at the National Superconducting Cyclotron Laboratory at MSU. In particular, the group working on the S800 spectrograph has supplied valuable data and discussions, and I would like to thank Prof. Sherrill, Jac Caggiano, and Dr. Daniel Bazin, also for his last minute help supplying valuable pictures of the S800. Several pictures in this dissertation were made using programs by David Johnson. Since my main work has been focused on computer simulations, I have received a lot of help from the computer department.

For financial support of this research I owe thanks to the US Department of

Energy, the Alfred P. Sloan Foundation, and the National Science Foundation, and I am grateful for their continuous support for basic science and their contribution to the search for the ultimate truth.

Outside MSU, I have also encountered strong encouragement from various people. Especially I would like to thank Dr. Carol Johnstone at Fermilab, Dr. Christian Bischof at Argonne, the US Particle Accelerator School, Japanese visiting scientists I had the pleasure to meet in the USA, and several scientists at KEK who had shared their time with me back in Japan. My special thanks go to Prof. Seigi Iwata at KEK and Prof. Ryouichi Kajikawa, who were my academic advisors back at Nagoya in Japan, and have strongly supported and encouraged my continued work in the academic society.

Before closing, I would like to thank my family who has always loved me. Unfortunately I had to confront the loss of my father, who had believed in the power in science, in the last December. My mother, who has shared the grief of the loss of my father, has been a big mental support for me through her unconditional love since my childhood. I appreciate my relatives, my sister and my friends for their encouragement and support. Finally my wish is that my beloved son Masayuki and daughter Kazuko may enjoy reading this thesis some time in the future.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Following the general scientific approach, the study of the motion of particles is performed in two steps. The first step is to establish the equations of motion, the second step is to solve the equations and analyze the results. The work in this thesis covers both aspects for the motion of an ensemble of particles in beam physics. The standard way to treat such a system is to pick a typical particle, the reference particle, and to describe the motion of the other particles relative to the reference motion, after which the problem is cast into the framework of weakly nonlinear dynamical systems. This thesis work focuses on the rigorous approaches to analyze such systems.

The first elaboration in this thesis is to derive the relativistic equation of motion of a charged particle in electromagnetic fields relative to a general three dimensional reference orbit. In chapter 2, Newton's equation as well as a Lagrangian and a Hamiltonian describing the motion in nonplanar curvilinear coordinates are derived. The various forms were checked for consistency, and an interchange of the independent variables from time t to arclength s is performed.

Besides providing the complete set of equations, the importance of this work lies in the fact that the relative motion with nonplanar reference orbit can still be recognized as Hamiltonian, which has key consequences in the study of this weakly nonlinear dynamical system. One of the by-products of the elaboration is the representation of the Laplacian operator $\triangle$ in nonplanar curvilinear coordinates, which enables the usage of the Differential Algebraic (DA) fixed point theorem [20] to solve the three dimensional potential problem. Another by-product is a set of equations of motion in the more conventional curvilinear coordinates in which the reference orbit is restricted to a plane.

At this point, the electromagnetic fields are described by $\vec{E}$ and $\vec{B}$, or the scalar potential Φ and the vector potential $\vec{A}$ in the equations of motions. When we study and design a beam optical system at a laboratory, the detailed knowledge of the fields sometimes becomes a crucial point to determine the quality of the device. Modern large acceptance spectrographs represent one such example. In chapter 3, a new method to take measured data of the fields into the equations of motion in an appropriate manner is discussed. The method has to fit to the principal work horse of our study in beam physics, the Differential Algebraic (DA) technique [3] [4] [5] [8] [20], so the interpolation method of the field distribution function has to supply differentiability to any order. For this purpose, we used the Gaussian wavelet approach. The method is studied to check the numerical performance, then is applied to the magnetic dipoles of the S800 spectrograph at the National Superconducting Cyclotron Laboratory at Michigan State University [60], where the accuracy of the fields is required to about 10^{-4} in order to achieve its high energy resolution of $1/10,000$ [16].

The method of transfer maps has various advantages to study beam optical systems compared to the conventional ray tracing method. It offers direct access to aberrations for single pass systems, and to chromaticities, amplitude tune shifts, and resonance strengths for repetitive systems such as circular accelerators and storage rings. Another topic where the map method is especially powerful is the analysis of the long-term behaviour of particles in repetitive systems. The DA techniques have offered a very elegant and accurate way to obtain high order Taylor transfer maps of the action on phase space. Typically derivatives of up to order ten in six variables are needed to study the long-term behaviour of particles in repetitive systems, so other methods are far from providing a robust way to study the weakly nonlinear behavior of beams.

The problem to estimate the long-term stability of weakly nonlinear systems finds its origin in the detailed study of the solar system. Many perturbative methods for repetitive motion have been developed from this question. Recently, ideas of Lyapunov, Nekhoroshev and others triggered an analysis of stability in particle accelerators based on approximate invariants[66] [14] [39], and the question of long-term stability can be re-cast into a highly complicated optimization problem [14] [15] [39].

The upper left picture in Figure 1.1 shows a tracking picture of particles launched from five different locations in a repetitive system described by a six dimensional Taylor transfer map in the sixth order in actual coordinates x and $a = p_x/p_0$. The upper right picture shows a tracking picture of the same system in normal form coordinates after a nonlinear normal form transformation, where the motion is seen approximately on invariant circles. The deviations from invariant circles in the normal form coordinates directly relate to the number of stable turns, and thus to the time for particles to be lost. The sharpness to which these deviations can be bounded is the key to guaranteeing a large number of stable turns.

The deviation functions are multidimensional polynomials up to roughly 500th order, and consist of about 10^5 floating-point operations. The functions have a very large number of local extrema, many large terms cancel each other, and very small fluctuations have to be estimated. To be useful, the maxima have to be sharp to about 10^{-6}, and for some applications to 10^{-12}. The lower picture in Figure 1.1 shows the

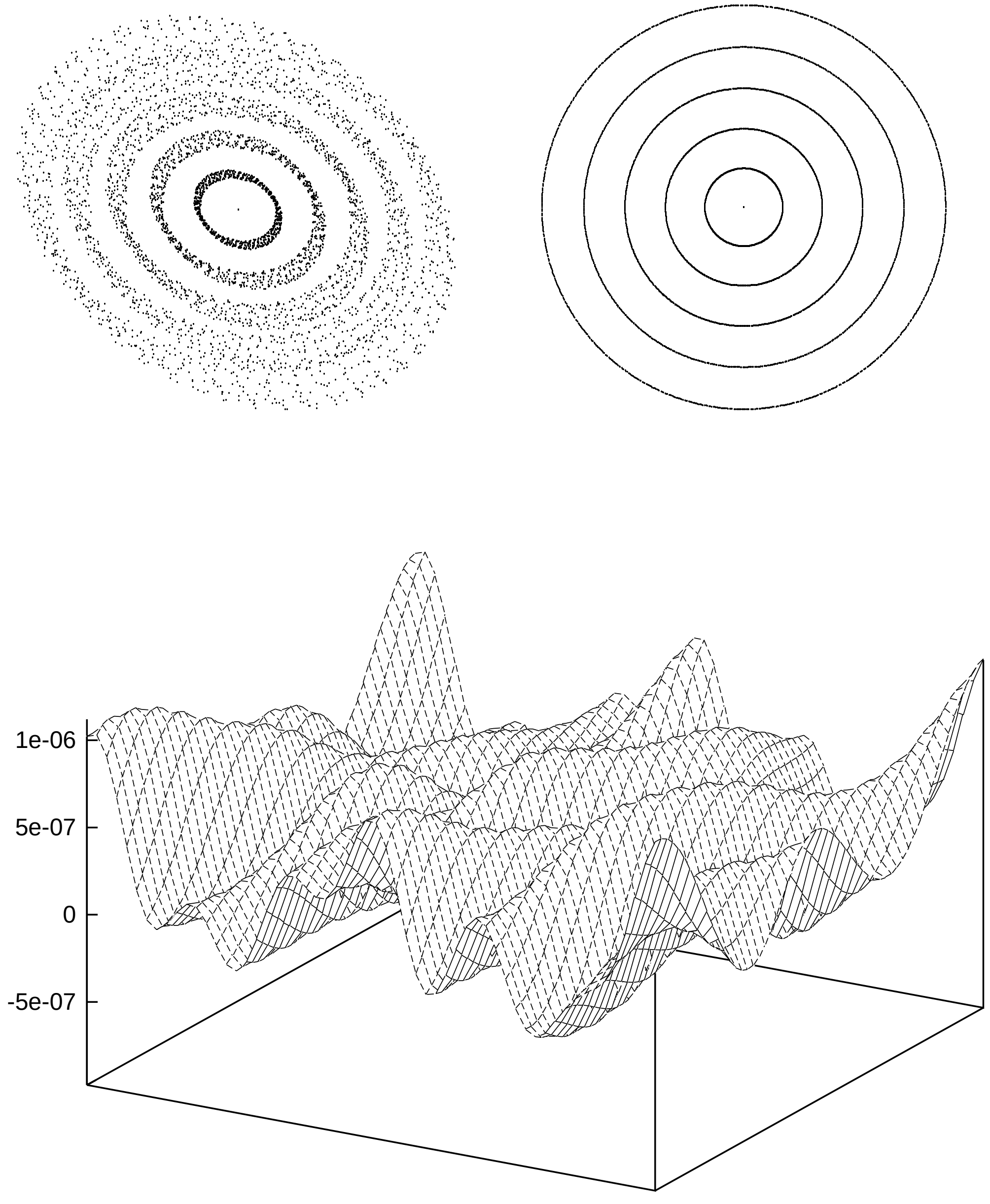

Figure 1.1: Upper: Tracking pictures of a repetitive system. The left is in particle optical coordinates x and $a = p_x/p_0$, and the right is in normal form coordinates. Lower: Deviation from a normal form invariant circle.

deviation from a normal form invariant circle as a function of two phase space angles.

The problem of finding rigorous bounds on the extrema of functions has contributed to the development of many methods in numerical analysis. In our case, the irregularity of the deviation functions as well as the high dimensionality makes the question very troublesome for conventional optimization methods. Interval methods give a mathematically rigorous estimate of the bounds, but complicated functions like ours cannot avoid the severe blow-up problem, the control of which is the key to get a practical estimate. Usually the division of the domain of interest into many small subintervals helps to suppress the severe blow-up problem. The deviation functions require about 10^4 subintervals per dimension to be sharply bounded in a small phase space volume 0.02^6 as discussed in subsection 5.5.3, and the total number of subinterval boxes goes up to 10^{24} to cover the whole six dimensional small domain, which limits the practicality.

A new technique presented in the following chapters, the method of Remainder-enhanced Differential Algebras (RDA) [52] [9] [12] [54], combines the DA technique to express the model function by a Taylor polynomial, and interval computations to evaluate the bound of the Taylor remainder. The resulting error bounds are usually rather sharp, in particular at higher orders. Figure 1.2 shows bound enclosures of a one dimensional function using a seventh order (upper left) and eighth order (upper right) RDA method, in comparison with the interval method applied to the divided domain, using 25 (lower left) and 200 (lower right) subintervals. The RDA method can be used for rigorous global optimization of highly complex multi-dimensional objective functions. Now a practical answer to the bounds of the deviation functions can be given.

Chapter 4 discusses the development of the method as well as some example cal-

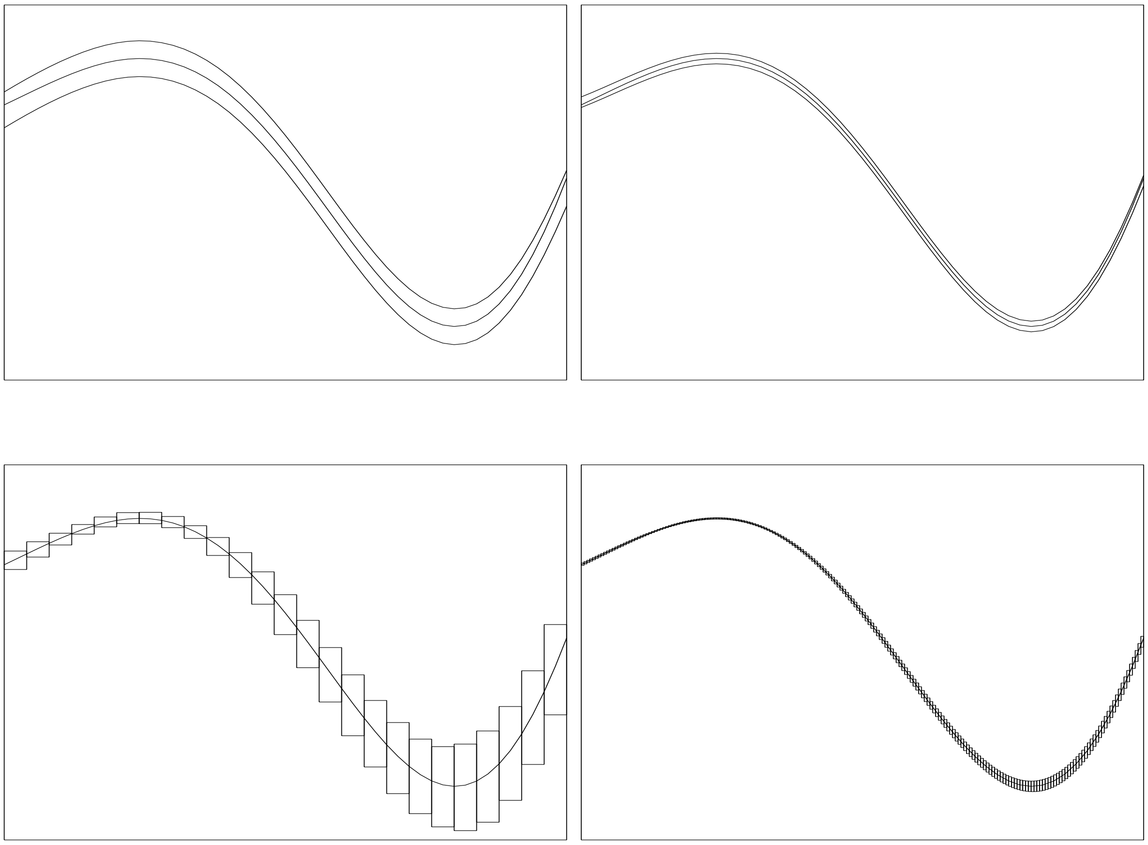

Figure 1.2: Bound enclosures of a function. Upper: RDA method by 7th (left) and 8th(right) order. Lower: Interval method with 25 (left) and 200 (right) subintervals.

culations. The implementation of the method in the arbitrary order general purpose beam optics code COSY INFINITY [11] [15] [51] [10] is discussed in chapter 5. Example computations in the chapter include a new scheme to compute multi-dimensional integrals, as well as the normal form deviation function.

To complete the question, the other important aspect is to have rigorous representations of transfer maps to describe weakly nonlinear systems. Chapter 6 develops the theory for a rigorous integration scheme for flows of ordinary differential equations within the framework of RDA, which enables to obtain rigorous remainders of Taylor transfer maps[18]. The task requires to start from the foundations in func-

tional analysis and Schauder's fixed point theorem was utilized. The method has been implemented in COSY INFINITY, and some example calculations are shown.

Chapter 2

The Particle Optical Equations of Motion

In this chapter, we will derive the relativistic equations of motion of a particle in an electromagnetic field in the so-called curvilinear coordinates. These coordinates are measured in a moving right-handed coordinate system that has one of its axes attached and parallel to a given reference curve in space; furthermore, usually the time is replaced as the independent variable by the arclength along the given reference curve.

While this approach seems to complicate the description of the motion, it has several advantages. Firstly, if the chosen reference curve in space is itself a valid orbit, then the resulting transfer map will be origin preserving, because the origin just corresponds to the reference curve itself. This fact then opens the door to the use of perturbative techniques for the analysis of the motion in order to study how small deviations from the reference curve propagate. In particular, if the system of interest is repetitive and the reference curve is closed, then the origin will be a fixed point of the motion. If the arclength is used as the independent variable, then after one turn around the reference orbit, the system is repetitive, and perturbative techniques around fixed points can be employed to study the one-turn transfer map, which here corresponds to the Poincare map of the motion.

Secondly, the method is also very practical in the sense that beams themselves are usually rather small in size, while they often cover large territory. So it is more convenient to describe them in a local coordinate system following a reference particle instead of a Cartesian coordinate system attached to the laboratory. Expressing the motion in terms of value of the transfer map at a given arclength very directly corresponds to the measurements by detectors, which usually determine particle coordinates at a fixed plane around the system instead of at a given time. And expressing the motion in terms of the arclength as independent variable directly provides a natural scale, since it is more natural to measure in meters along the system instead of nano- or microseconds.

The following sections describe in detail the derivation of the motion in curvilinear coordinates. We will study the transformations of Maxwell's equations and the resulting fields and their potentials to the new coordinates, and then derive the explicit forms of Newton's equations as well as the Lagrangian and Hamiltonian with time as the independent variable. Finally, a special transformation on Hamiltonians is applied that replaces time as the independent variable by the arclength, while maintaining the Hamiltonian structure of the motion.

2.1 Nonplanar Curvilinear Coordinates

Let $\{\vec{e}_1, \vec{e}_2, \vec{e}_3\}$ denote a Dreibein, a right-handed set of fixed orthonormal basis vectors, which defines the so-called Cartesian coordinate systems. For any point in space, let (x_1, x_2, x_3) denote its Cartesian coordinates. In order to introduce the curvilinear coordinates, let $\vec{R}(s)$ be an infinitely often differentiable curve parameterized in terms of its arc length s, the so-called reference curve. For each value of s, let the vector $\vec{e}_s$

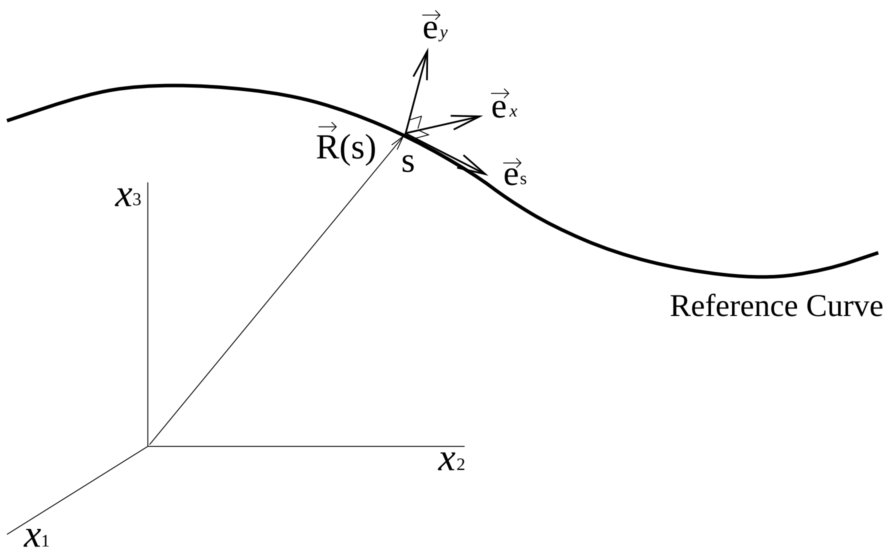

Figure 2.1: Reference curve and the locally attached Dreibeins.

be parallel to the reference curve, i.e.

$$\vec{e}_s(s) = \frac{d\vec{R}}{ds}. \tag{2.1}$$

We now choose the infinitely often differentiable vectors $\vec{e}_x(s)$ and $\vec{e}_y(s)$ such that for any value of s, the three vectors $\{\vec{e}_s, \vec{e}_x, \vec{e}_y\}$ form a Dreibein, a right-handed orthonormal system. For notational simplicity, in the following we also sometimes denote the curvilinear basis vectors $\{\vec{e}_s, \vec{e}_x, \vec{e}_y\}$ by $\{\vec{e}_1^{\,C}, \vec{e}_2^{\,C}, \vec{e}_3^{\,C}\}$.

Apparently, for a given curve $\vec{R}(s)$ there are a variety of choices for $\vec{e}_x(s)$ and $\vec{e}_y(s)$ that result in valid Dreibeins since $\vec{e}_x(s)$ and $\vec{e}_y(s)$ can be rotated around $\vec{e}_s$. A specific choice is often made such that additional requirements are satisfied; for example, if the curve $\vec{R}(s)$ is never parallel to the vertical Cartesian coordinate $\vec{e}_3$, one may demand that $\vec{e}_x(s)$ always lie in the horizontal plane spanned by $\vec{e}_1$ and $\vec{e}_2$.

The functions $\vec{R}(s)$, $\vec{e}_x(s)$, and $\vec{e}_y(s)$ describe the so-called curvilinear coordinate

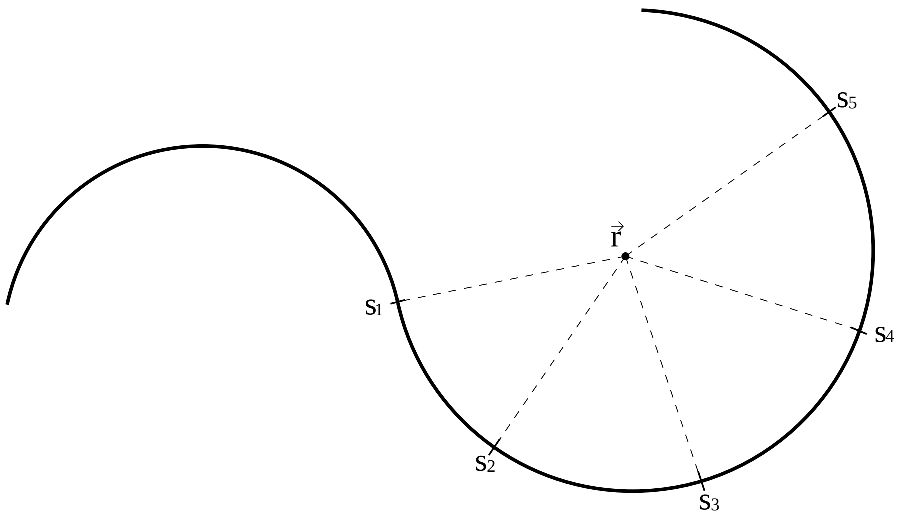

Figure 2.2: Non-uniqueness of curvilinear coordinates.

system, in which a position is described in terms of s, x and y via

$$\vec{r} = \vec{R}(s) + x\,\vec{e}_x + y\,\vec{e}_y.$$

Apparently the position $\vec{r}$ in Cartesian coordinates is uniquely determined for any choice of (s, x, y). The converse, however, is not generally true: a point with given Cartesian coordinates $\vec{r}$ may lie in several different planes that are perpendicular to the curve $\vec{R}(s)$, as shown in Figure 2.2.

The situation can be remedied if the curvature $\kappa(s)$ of the reference curve $\vec{R}(s)$ never grows beyond a threshold, i.e. if

$$r_1 = 1/\max_s |\kappa(s)| \tag{2.2}$$

is finite. As Figure 2.3 illustrates, if in this case we restrict ourselves to the inside of a tube of radius r_1 around $\vec{R}(s)$, for any vector within the tube, there is always one and only one set of coordinates (s, x, y) describing the point $\vec{r}$.

Let us now study the transformation matrix from the Cartesian basis $\{\vec{e}_1, \vec{e}_2, \vec{e}_3\}$ to the local basis of the curvilinear system $\{\vec{e}_s, \vec{e}_x, \vec{e}_y\} = \{\vec{e}_1^{\,C}, \vec{e}_2^{\,C}, \vec{e}_3^{\,C}\}$. The transfor-

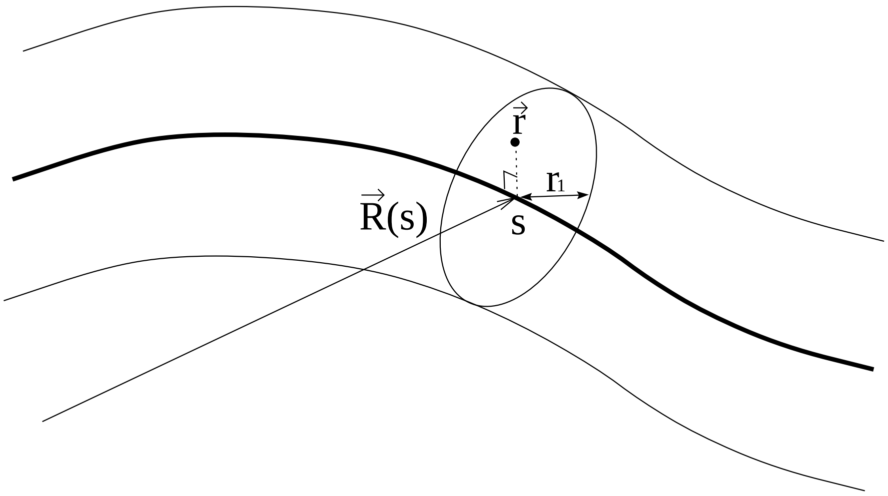

Figure 2.3: Uniqueness of curvilinear coordinates within a tube.

mation between these basis vectors and the old ones is described by the matrix $\hat{O}(s)$ which has the form

$$\hat{O}(s) = \left(\begin{array}{c|c|c} \vec{e}_s(s) & \vec{e}_x(s) & \vec{e}_y(s) \end{array} \right) = \begin{pmatrix} (\vec{e}_s \cdot \vec{e}_1) & (\vec{e}_x \cdot \vec{e}_1) & (\vec{e}_y \cdot \vec{e}_1) \\ (\vec{e}_s \cdot \vec{e}_2) & (\vec{e}_x \cdot \vec{e}_2) & (\vec{e}_y \cdot \vec{e}_2) \\ (\vec{e}_s \cdot \vec{e}_3) & (\vec{e}_x \cdot \vec{e}_3) & (\vec{e}_y \cdot \vec{e}_3) \end{pmatrix}. \tag{2.3}$$

Because the system $\{\vec{e}_s, \vec{e}_x, \vec{e}_y\}$ is orthonormal, so is $\hat{O}(s)$, and hence it satisfies

$$\hat{O}(s) \cdot \hat{O}(s)^t = \hat{I} \quad \text{and} \quad \hat{O}(s)^t \cdot \hat{O}(s) = \hat{I}. \tag{2.4}$$

Since both the old and the new bases have the same handedness, we also have

$$\det(\hat{O}(s)) = 1, \tag{2.5}$$

and hence altogether, $\hat{O}(s)$ belongs to the group SO(3). We remind ourselves that elements of SO(3) preserve cross products, i.e. for $\hat{O} \in \mathrm{SO}(3)$ and any vectors $\vec{a}$, $\vec{b}$, we have

$$(\hat{O}\vec{a}) \times (\hat{O}\vec{b}) = \hat{O}(\vec{a} \times \vec{b}). \tag{2.6}$$

One way to see this is to study the requirement of orthonormality on the matrix elements of $\hat{O}$. The elements of the matrix $\hat{O}$ describe the coordinates of the new parameter dependent basis vectors in terms of the original Cartesian basis; explicitly, we have

$$[\vec{e}_s]_k = O_{k1}, \quad [\vec{e}_x]_k = O_{k2}, \quad [\vec{e}_y]_k = O_{k3}. \tag{2.7}$$

The demand of the right-handedness then reads

$$\vec{e}_l^{\,C} \times \vec{e}_m^{\,C} = \sum_{n=1}^{3} \epsilon_{lmn} \vec{e}_n^{\,C},$$

where ϵ_{ijk} is the common totally antisymmetric tensor of rank three defined as

$$\epsilon_{ijk} = \begin{cases} 1 & \text{for } (i,j,k) = (1,2,3) \text{ and any cyclic permutation thereof} \\ -1 & \text{for other permutations of } (1,2,3) \\ 0 & \text{for two or more equal indices} \end{cases},$$

and boils down to a condition on the elements of the matrix $\hat{O}$

$$\sum_{i,j=1}^{3} \epsilon_{ijk} O_{il} O_{jm} = \sum_{n=1}^{3} \epsilon_{lmn} O_{kn}. \tag{2.8}$$

We remind ourselves that the symbol ϵ_{ijk} is very useful for the calculation of vector cross products; for vectors $\vec{a}, \vec{b}$, we have

$$[\vec{a} \times \vec{b}]_k = \sum_{i,j=1}^{3} \epsilon_{ijk} a_i b_j.$$

Using condition (2.8), we readily obtain (2.6).

For the following discussion, it is useful to study how the transformation matrix $\hat{O}$ changes with s. Differentiating (2.4) with respect to the parameter s, we have

$$0 = \frac{d}{ds}(\hat{O}^t \cdot \hat{O}) = \frac{d\hat{O}^t}{ds}\hat{O} + \hat{O}^t \frac{d\hat{O}}{ds} = \left(\hat{O}^t \frac{d\hat{O}}{ds}\right)^t + \hat{O}^t \frac{d\hat{O}}{ds}.$$

So, the matrix $\hat{T} = \hat{O}^t \cdot d\hat{O}/ds$ is antisymmetric; we describe it in terms of its three free elements via

$$\hat{O}^t \cdot \frac{d\hat{O}}{ds} = \hat{T} = \begin{pmatrix} 0 & -\tau_3 & \tau_2 \\ \tau_3 & 0 & -\tau_1 \\ -\tau_2 & \tau_1 & 0 \end{pmatrix}. \tag{2.9}$$

The three elements we group into the vector $\vec{\tau}$, which has the form

$$\vec{\tau} = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix}.$$

We observe that for any vector $\vec{a}$, we then have the relation

$$\hat{T} \cdot \vec{a} = \vec{\tau} \times \vec{a}.$$

The components of the vector $\vec{\tau}$, and hence the elements of the matrix $\hat{T}$, can be computed as

$$\begin{aligned} \tau_1 &= \vec{e}_y \cdot \frac{d\vec{e}_x}{ds} = -\vec{e}_x \cdot \frac{d\vec{e}_y}{ds} \\ \tau_2 &= \vec{e}_s \cdot \frac{d\vec{e}_y}{ds} = -\vec{e}_y \cdot \frac{d\vec{e}_s}{ds} \\ \tau_3 &= \vec{e}_x \cdot \frac{d\vec{e}_s}{ds} = -\vec{e}_s \cdot \frac{d\vec{e}_x}{ds}. \end{aligned} \tag{2.10}$$

These relationships give some practical meaning to the components of the vector $\vec{\tau}$: Apparently, τ_1 describes the current rate of rotation of the Dreibein around the reference curve $\vec{R}(s)$; τ_2 describes the current amount curvature of $\vec{R}(s)$ in the plane spanned by $\vec{e}_y$ and $\vec{e}_s$; and τ_3 similarly describes the curvature of $\vec{R}(s)$ in the plane spanned by $\vec{e}_x$ and $\vec{e}_s$. In more mathematical terms, because of

$$\vec{e}_s \cdot \frac{d\vec{e}_s}{ds} = 0, \quad \vec{e}_x \cdot \frac{d\vec{e}_x}{ds} = 0, \quad \vec{e}_y \cdot \frac{d\vec{e}_y}{ds} = 0, \tag{2.11}$$

we have

$$\begin{aligned} \frac{d\vec{e}_s}{ds} &= \tau_3 \vec{e}_x - \tau_2 \vec{e}_y \\ \frac{d\vec{e}_x}{ds} &= -\tau_3 \vec{e}_s + \tau_1 \vec{e}_y \\ \frac{d\vec{e}_y}{ds} &= \tau_2 \vec{e}_s - \tau_1 \vec{e}_x, \end{aligned} \tag{2.12}$$

as successive multiplication with $\vec{e}_s$, $\vec{e}_x$ and $\vec{e}_y$ and comparison with (2.10) reveals.

2.2 Differential Operators in Nonplanar Curvilinear Coordinates

As the first step in the transformation of the Maxwell's equations as well the equations of motion to the curvilinear coordinates, it is necessary to study the form of common differential operators in the new coordinates. From (2.7), which has the form

$$\vec{r} = \sum_{k=1}^{3} x_k \vec{e}_k = \sum_{k=1}^{3} \left\{ \vec{R} \cdot \vec{e}_k + x O_{k2} + y O_{k3} \right\} \vec{e}_k,$$

we see that the Cartesian components of $\vec{r}$ are

$$x_k = \vec{R} \cdot \vec{e}_k + x O_{k2} + y O_{k3}, \text{ for } k = 1, 2, 3. \tag{2.13}$$

Hence the partial derivatives of x_k with respect to s, x and y are

$$\begin{aligned} \frac{\partial x_k}{\partial s} &= \frac{d\vec{R}(s)}{ds} \cdot \vec{e}_k + x\frac{dO_{k2}}{ds} + y\frac{dO_{k3}}{ds} = O_{k1} + x\frac{dO_{k2}}{ds} + y\frac{dO_{k3}}{ds} \\ \frac{\partial x_k}{\partial x} &= O_{k2}, \quad \text{and} \quad \frac{\partial x_k}{\partial y} = O_{k3}, \end{aligned}$$

where (2.1) and (2.7) have been used. Thus, the Jacobian matrix $\hat{C}$ is

$$\begin{aligned} \hat{C} &= \left(\frac{\partial(x_1, x_2, x_3)}{\partial(s, x, y)} \right) = \begin{pmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_2}{\partial s} & \frac{\partial x_3}{\partial s} \\ \frac{\partial x_1}{\partial x} & \frac{\partial x_2}{\partial x} & \frac{\partial x_3}{\partial x} \\ \frac{\partial x_1}{\partial y} & \frac{\partial x_2}{\partial y} & \frac{\partial x_3}{\partial y} \end{pmatrix} = \begin{pmatrix} O_{11} & O_{21} & O_{31} \\ O_{12} & O_{22} & O_{32} \\ O_{13} & O_{23} & O_{33} \end{pmatrix} \\ &+ \begin{pmatrix} x\frac{dO_{12}}{ds} + y\frac{dO_{13}}{ds} & x\frac{dO_{22}}{ds} + y\frac{dO_{23}}{ds} & x\frac{dO_{32}}{ds} + y\frac{dO_{33}}{ds} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ &= \hat{O}^t + \begin{pmatrix} 0 & x & y \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \frac{d\hat{O}^t}{ds} = \hat{O}^t + \begin{pmatrix} 0 & x & y \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \frac{d\hat{O}^t}{ds} \cdot (\hat{O} \cdot \hat{O}^t) \\ &= \left\{ \hat{I} + \begin{pmatrix} 0 & x & y \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \cdot \hat{T}^t \right\} \cdot \hat{O}^t = \begin{pmatrix} 1 - \tau_3 x + \tau_2 y & -\tau_1 y & \tau_1 x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \hat{O}^t. \end{aligned}$$

It is convenient to denote the first part of the Jacobian matrix $\hat{C}$ by $\hat{A}$, i.e.

$$\hat{A} = \begin{pmatrix} 1 - \tau_3 x + \tau_2 y & -\tau_1 y & \tau_1 x \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix};$$

then the Jacobian matrix can be written as

$$\hat{C} = \hat{A} \cdot \hat{O}^t.$$

The inverse matrix of $\hat{A}$ is found easily; we obtain

$$\hat{A}^{-1} = \begin{pmatrix} \dfrac{1}{1 - \tau_3 x + \tau_2 y} & \dfrac{\tau_1 y}{1 - \tau_3 x + \tau_2 y} & \dfrac{-\tau_1 x}{1 - \tau_3 x + \tau_2 y} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

For later convenience, it is advantageous to introduce the abbreviation

$$\alpha = 1 - \tau_3 x + \tau_2 y. \tag{2.14}$$

We note that for x and y sufficiently close to zero, α does not vanish and is positive. Hence besides the restriction for the motion to be inside a tube of radius r_1 imposed by the need for uniqueness of the transformation to curvilinear coordinates in (2.2), there is another condition; defining

$$r_2 = \frac{1}{2} \min_s \left(\left|\frac{1}{\tau_3}\right|, \left|\frac{1}{\tau_2}\right|\right), \tag{2.15}$$

then if we restrict x, y to satisfy $|x|, |y| < r_2$, the quantity α never vanishes.

In this case, for the inverse matrix of the Jacobian matrix $\hat{C}$ we have

$$\hat{C}^{-1} = \hat{O} \cdot \hat{A}^{-1} = \hat{O} \cdot \begin{pmatrix} \dfrac{1}{\alpha} & \dfrac{\tau_1 y}{\alpha} & \dfrac{-\tau_1 x}{\alpha} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{2.16}$$

Now all the necessary preparations are made for the calculation of partial differential operators such as gradient, divergence, curl and Laplacian in the curvilinear coordinate system.

2.2.1 Gradient

Let f be a scalar function, expressed either in the Cartesian coordinates (x_1, x_2, x_3), or the curvilinear coordinates (s, x, y). From the chain rule, we have

$$\begin{pmatrix} \frac{\partial}{\partial s} \\ \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} f = \begin{pmatrix} \frac{\partial x_1}{\partial s} & \frac{\partial x_2}{\partial s} & \frac{\partial x_3}{\partial s} \\ \frac{\partial x_1}{\partial x} & \frac{\partial x_2}{\partial x} & \frac{\partial x_3}{\partial x} \\ \frac{\partial x_1}{\partial y} & \frac{\partial x_2}{\partial y} & \frac{\partial x_3}{\partial y} \end{pmatrix} \cdot \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{pmatrix} f \;=\; \hat{C} \cdot \vec{\nabla}^{ct} f,$$

where $\vec{\nabla}^{ct}$ is the Cartesian differential operator vector. Multiplying with $\hat{C}^{-1}$, we obtain the expression of this vector in terms of partial derivatives with respect to the particle optical coordinates as

$$\vec{\nabla}^{ct} f = \begin{pmatrix} \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} \end{pmatrix} f = \hat{C}^{-1} \cdot \begin{pmatrix} \frac{\partial}{\partial s} \\ \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} f = \hat{O} \cdot \begin{pmatrix} \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) \\ \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} f,$$

where (2.16) was used. We now define the vector differential operator $\vec{\nabla}^C$ as

$$\vec{\nabla}^C f = \begin{pmatrix} \nabla_1^C \\ \nabla_2^C \\ \nabla_3^C \end{pmatrix} f \;=\; \begin{pmatrix} \nabla_s \\ \nabla_x \\ \nabla_y \end{pmatrix} f \;=\; \begin{pmatrix} \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) \\ \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} f. \tag{2.17}$$

Then we have

$$\vec{\nabla}^{ct} f = \hat{O} \cdot \vec{\nabla}^C f \text{ and } \vec{\nabla}^C f = \hat{O}^t \cdot \vec{\nabla}^{ct} f, \tag{2.18}$$

or, for later use, in components,

$$\nabla_k^{ct} f = \sum_{l=1}^{3} O_{kl} \nabla_l^C f \text{ and } \nabla_k^C f = \sum_{l=1}^{3} O_{lk} \nabla_l^{ct} f. \tag{2.19}$$

Let us consider a vector function $\vec{A}$; we express it in both Cartesian and curvilinear coordinates:

$$\vec{A} = A_1\vec{e}_1 + A_2\vec{e}_2 + A_3\vec{e}_3 = A_s\vec{e}_s + A_x\vec{e}_x + A_y\vec{e}_y.$$

We denote the component vectors in Cartesian and curvilinear coordinates with $\vec{A}^{ct}$ and $\vec{A}^{C}$, respectively, and have

$$\vec{A}^{ct} = \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix}, \quad \vec{A}^{C} = \begin{pmatrix} A_1^C \\ A_2^C \\ A_3^C \end{pmatrix} = \begin{pmatrix} A_s \\ A_x \\ A_y \end{pmatrix}.$$

Then, because of (2.3)

$$\vec{A} = A_s\vec{e}_s + A_x\vec{e}_x + A_y\vec{e}_y = A_s(\hat{O}\vec{e}_1) + A_x(\hat{O}\vec{e}_2) + A_y(\hat{O}\vec{e}_3) = \hat{O}\cdot(A_s\vec{e}_1 + A_x\vec{e}_2 + A_y\vec{e}_3),$$

and so we have

$$\vec{A}^{ct} = \hat{O}\cdot\vec{A}^{C} \text{ as well as } \vec{A}^{C} = \hat{O}^t\cdot\vec{A}^{ct}\ . \tag{2.20}$$

As a first step, we now want to determine the form of the gradient operator in curvilinear coordinates. In the Cartesian system, the gradient operation is

$$\mathrm{grad}^{ct} f = \vec{\nabla}^{ct} f \ = \begin{pmatrix} \dfrac{\partial f}{\partial x_1} \\ \dfrac{\partial f}{\partial x_2} \\ \dfrac{\partial f}{\partial x_3} \end{pmatrix} = \frac{\partial f}{\partial x_1}\vec{e}_1 + \frac{\partial f}{\partial x_2}\vec{e}_2 + \frac{\partial f}{\partial x_3}\vec{e}_3.$$

As in the situation with the more common coordinate systems, the gradient operator in the curvilinear system should determine the Cartesian gradient of a function, and then express it in terms of curvilinear coordinates; so we must have

$$\mathrm{grad}^{C} f = \hat{O}^t\cdot\mathrm{grad}^{ct} f.$$

We find that $\vec{\nabla}^C f$ defined in (2.18) satisfies this demand; so the gradient operation in the curvilinear system is

$$\mathrm{grad}^C f = \vec{\nabla}^C f = \begin{pmatrix} \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) f \\ \frac{\partial}{\partial x} f \\ \frac{\partial}{\partial y} f \end{pmatrix},$$

that is

$$\mathrm{grad}^C f = \left[\frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) f\right] \vec{e}_s + \frac{\partial f}{\partial x}\vec{e}_x + \frac{\partial f}{\partial y}\vec{e}_y. \tag{2.21}$$

2.2.2 Divergence

The divergence in the curvilinear system is calculated as follows. In the Cartesian system,

$$\mathrm{div}\vec{A} = \frac{\partial A_1}{\partial x_1} + \frac{\partial A_2}{\partial x_2} + \frac{\partial A_3}{\partial x_3}.$$

Our goal is to express it in terms of the curvilinear system. For this purpose, we apply (2.19) to the three components $A_k^{ct} = \sum_m O_{km} A_m^C$

$$\begin{aligned} \mathrm{div}\vec{A} &= (\vec{\nabla}^{ct} \cdot \vec{A}^{ct}) = \sum_k \nabla_k^{ct} \cdot A_k^{ct} = \sum_k \sum_l O_{kl} \nabla_l^C \left(\sum_m O_{km} A_m^C\right) \\ &= \sum_k \sum_{l,m} O_{kl} O_{km} \nabla_l^C A_m^C + \sum_k \sum_{l,m} O_{kl} (\nabla_l^C O_{km}) A_m^C. \end{aligned}$$

Since $\hat{O} = \hat{O}(s)$, we have

$$\nabla_l^C O_{km} = \delta_{ls} \nabla_s O_{km} = \delta_{ls} \frac{1}{\alpha} \frac{dO_{km}}{ds}. \tag{2.22}$$

Using this relationship, we obtain

$$\mathrm{div}\vec{A} = \sum_{l,m} \sum_k O_{kl} O_{km} \nabla_l^C A_m^C + \sum_k \sum_{l,m} O_{kl} \delta_{ls} \frac{1}{\alpha} \frac{dO_{km}}{ds} A_m^C$$

$$
\begin{aligned}
&= \sum_{l,m} [\hat{O}^t\hat{O}]_{lm} \nabla_l^C A_m^C + \sum_{l,m} \delta_{ls} \frac{1}{\alpha} [\hat{O}^t \frac{d\hat{O}}{ds}]_{lm} A_m^C \\
&= \sum_{l,m} \delta_{lm} \nabla_l^C A_m^C + \sum_m \frac{1}{\alpha} T_{sm} A_m^C = \sum_m \nabla_m^C A_m^C + \frac{1}{\alpha}(-\tau_3 A_x + \tau_2 A_y) \\
&= \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) A_s + \frac{\partial A_x}{\partial x} + \frac{\partial A_y}{\partial y} + \frac{1}{\alpha}(-\tau_3 A_x + \tau_2 A_y).
\end{aligned}
$$

Thus, the divergence expressed in the curvilinear system is obtained as

$$
\mathrm{div}\vec{A} = \frac{1}{\alpha}\left\{\left(\frac{\partial}{\partial s} + \tau_1 y \frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) A_s + \frac{\partial}{\partial x}(\alpha A_x) + \frac{\partial}{\partial y}(\alpha A_y)\right\}. \tag{2.23}
$$

2.2.3 Curl

The derivation of the curl in the curvilinear coordinates is a little more involved. In the Cartesian system,

$$
\begin{aligned}
\mathrm{curl}^{ct}\vec{A} &= \vec{\nabla}^{ct} \times \vec{A}^{ct} = [\vec{\nabla}^{ct} \times \vec{A}^{ct}]_1 \vec{e}_1 + [\vec{\nabla}^{ct} \times \vec{A}^{ct}]_2 \vec{e}_2 + [\vec{\nabla}^{ct} \times \vec{A}^{ct}]_3 \vec{e}_3 \\
&= \begin{pmatrix} \dfrac{\partial A_3}{\partial x_2} - \dfrac{\partial A_2}{\partial x_3} \\ \dfrac{\partial A_1}{\partial x_3} - \dfrac{\partial A_3}{\partial x_1} \\ \dfrac{\partial A_2}{\partial x_1} - \dfrac{\partial A_1}{\partial x_2} \end{pmatrix}.
\end{aligned}
$$

The curl in the curvilinear coordinates, which we denote by $\mathrm{curl}^C\vec{A}$ and which has the components

$$
\mathrm{curl}^C\vec{A} = \begin{pmatrix} \left[\mathrm{curl}^C\vec{A}\right]_s \\ \left[\mathrm{curl}^C\vec{A}\right]_x \\ \left[\mathrm{curl}^C\vec{A}\right]_y \end{pmatrix},
$$

has to satisfy the condition

$$
\mathrm{curl}^{ct}\vec{A} = \hat{O} \cdot \mathrm{curl}^C\vec{A}. \tag{2.24}
$$

First, let us express each component of $\mathrm{curl}^{ct}\vec{A}$ in terms of the curvilinear system, again using the transformation rules for derivative components (2.19). We obtain

$$
\begin{aligned}
[\mathrm{curl}^{ct}\vec{A}]_k &= [\vec{\nabla}^{ct} \times \vec{A}]_k = \sum_{i,j} \epsilon_{ijk} \nabla_i^{ct} A_j^{ct} = \sum_{i,j}\sum_{l,m} \epsilon_{ijk} O_{il} \nabla_l^C (O_{jm} A_m^C) \\
&= \sum_{l,m}\sum_{i,j} \epsilon_{ijk} O_{il} O_{jm} \nabla_l^C A_m^C + \sum_{l,m}\sum_{i,j} \epsilon_{ijk} O_{il} (\nabla_l^C O_{jm}) A_m^C \\
&= \sum_{l,m} \left(\sum_n \epsilon_{lmn} O_{kn} \right) \nabla_l^C A_m^C + \sum_{l,m}\sum_{i,j} \epsilon_{ijk} O_{il} \delta_{ls} \frac{1}{\alpha} \frac{dO_{jm}}{ds} A_m^C \\
&= \sum_n O_{kn} \left(\sum_{l,m} \epsilon_{lmn} \nabla_l^C A_m^C \right) + \frac{1}{\alpha} \sum_m \left(\sum_{i,j} \epsilon_{ijk} O_{is} \frac{dO_{jm}}{ds} \right) A_m^C,
\end{aligned}
$$

where (2.8) and (2.22) are used to obtain the third line. Making use of the fact that

$$
\sum_{i,j} \epsilon_{ijk} O_{is} \frac{dO_{jm}}{ds} = [\vec{e}_s \times \frac{d\vec{e}_m^C}{ds}]_k
$$

as well as the relationships in (2.12) which entail

$$
\vec{e}_s \times \frac{d\vec{e}_s}{ds} = \tau_3 \vec{e}_y + \tau_2 \vec{e}_x,\ \vec{e}_s \times \frac{d\vec{e}_x}{ds} = -\tau_1 \vec{e}_x,\ \text{and } \vec{e}_s \times \frac{d\vec{e}_y}{ds} = -\tau_1 \vec{e}_y,
$$

we obtain

$$
\begin{aligned}
[\mathrm{curl}^{ct}\vec{A}]_k &= \sum_n O_{kn} [\vec{\nabla}^C \times \vec{A}^C]_n + \frac{1}{\alpha} \{ (\tau_3 O_{k3} + \tau_2 O_{k2}) A_s - \tau_1 O_{k2} A_x - \tau_1 O_{k3} A_y \} \\
&= \sum_n O_{kn} [\vec{\nabla}^C \times \vec{A}^C]_n + \frac{1}{\alpha} \{ O_{k2} (\tau_2 A_s - \tau_1 A_x) + O_{k3} (\tau_3 A_s - \tau_1 A_y) \}.
\end{aligned}
$$

So

$$
\mathrm{curl}^{ct}\vec{A} = \hat{O} \cdot \left\{ \begin{pmatrix} \left[\vec{\nabla}^C \times \vec{A}^C\right]_1 \\ \left[\vec{\nabla}^C \times \vec{A}^C\right]_2 \\ \left[\vec{\nabla}^C \times \vec{A}^C\right]_3 \end{pmatrix} + \frac{1}{\alpha} \begin{pmatrix} 0 \\ \tau_2 A_s - \tau_1 A_x \\ \tau_3 A_s - \tau_1 A_y \end{pmatrix} \right\}.
$$

Now transforming the curl vector to curvilinear coordinates according to (2.24), we have

$$\begin{aligned}
\mathrm{curl}^C \vec{A} &= \hat{O}^t \cdot \mathrm{curl}^{ct} \vec{A} = \begin{pmatrix} \left[\vec{\nabla}^C \times \vec{A}^C\right]_1 \\ \left[\vec{\nabla}^C \times \vec{A}^C\right]_2 \\ \left[\vec{\nabla}^C \times \vec{A}^C\right]_3 \end{pmatrix} + \frac{1}{\alpha} \begin{pmatrix} 0 \\ \tau_2 A_s - \tau_1 A_x \\ \tau_3 A_s - \tau_1 A_y \end{pmatrix} \\
&= \begin{pmatrix} \nabla_x A_y - \nabla_y A_x \\ \nabla_y A_s - \nabla_s A_y + \dfrac{1}{\alpha}(\tau_2 A_s - \tau_1 A_x) \\ \nabla_s A_x - \nabla_x A_s + \dfrac{1}{\alpha}(\tau_3 A_s - \tau_1 A_y) \end{pmatrix}.
\end{aligned}$$

So altogether, expressed in terms of partial derivatives with respect to the curvilinear coordinates, we have

$$\begin{aligned}
\mathrm{curl}^C \vec{A} &= \begin{pmatrix} \dfrac{\partial A_y}{\partial x} - \dfrac{\partial A_x}{\partial y} \\ \dfrac{\partial A_s}{\partial y} - \dfrac{1}{\alpha}\left(\dfrac{\partial}{\partial s} + \tau_1 y \dfrac{\partial}{\partial x} - \tau_1 x \dfrac{\partial}{\partial y}\right) A_y + \dfrac{1}{\alpha}(\tau_2 A_s - \tau_1 A_x) \\ \dfrac{1}{\alpha}\left(\dfrac{\partial}{\partial s} + \tau_1 y \dfrac{\partial}{\partial x} - \tau_1 x \dfrac{\partial}{\partial y}\right) A_x - \dfrac{\partial A_s}{\partial x} + \dfrac{1}{\alpha}(\tau_3 A_s - \tau_1 A_y) \end{pmatrix} \\
&= \frac{1}{\alpha} \begin{pmatrix} \alpha \dfrac{\partial A_y}{\partial x} - \alpha \dfrac{\partial A_x}{\partial y} \\ \dfrac{\partial}{\partial y}(\alpha A_s) - \tau_1 A_x - \left(\dfrac{\partial}{\partial s} + \tau_1 y \dfrac{\partial}{\partial x} - \tau_1 x \dfrac{\partial}{\partial y}\right) A_y \\ -\dfrac{\partial}{\partial x}(\alpha A_s) + \left(\dfrac{\partial}{\partial s} + \tau_1 y \dfrac{\partial}{\partial x} - \tau_1 x \dfrac{\partial}{\partial y}\right) A_x - \tau_1 A_y \end{pmatrix}. \qquad (2.25)
\end{aligned}$$

2.2.4 Laplacian

The Laplacian operator in the Cartesian system is

$$\triangle^{ct} f = (\vec{\nabla}^{ct} \cdot \vec{\nabla}^{ct}) f = (\vec{\nabla}^{ct})^t \cdot \vec{\nabla}^{ct} f = \frac{\partial^2 f}{\partial x_1^2} + \frac{\partial^2 f}{\partial x_2^2} + \frac{\partial^2 f}{\partial x_3^2}.$$

In terms of the curvilinear system, utilizing (2.19), we have

$$\triangle^C f = (\vec{\nabla}^{ct})^t \cdot \vec{\nabla}^{ct} f = \sum_k \sum_{l,m} O_{kl} \nabla_l^C (O_{km} \nabla_m^C) f$$

$$
\begin{aligned}
&= \sum_{l,m}\sum_{k} O_{kl}O_{km}\nabla_l^C(\nabla_m^C f) + \sum_{k}\sum_{l,m} O_{kl}(\nabla_l^C O_{km})(\nabla_m^C f) \\
&= \sum_{l,m} \delta_{lm}\nabla_l^C(\nabla_m^C f) + \sum_{k}\sum_{m}\frac{1}{\alpha}O_{ks}\frac{dO_{km}}{ds}(\nabla_m^C f) = \sum_{m}\left(\nabla_m^C + \frac{1}{\alpha}T_{sm}\right)(\nabla_m^C f) \\
&= \nabla_s(\nabla_s f) + \left(\nabla_x - \frac{\tau_3}{\alpha}\right)(\nabla_x f) + \left(\nabla_y + \frac{\tau_2}{\alpha}\right)(\nabla_y f),
\end{aligned}
$$

where (2.4) and (2.22) are used from the second to the third line. Thus, the Laplacian operator in the curvilinear system, expressed in partials of curvilinear coordinates, has the form

$$
\begin{aligned}
\triangle^C f &= \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y\frac{\partial}{\partial x} - \tau_1 x\frac{\partial}{\partial y}\right)\left\{\frac{1}{\alpha}\left(\frac{\partial f}{\partial s} + \tau_1 y\frac{\partial f}{\partial x} - \tau_1 x\frac{\partial f}{\partial y}\right)\right\} \\
&\quad + \frac{1}{\alpha}\frac{\partial}{\partial x}\left(\alpha\frac{\partial f}{\partial x}\right) + \frac{1}{\alpha}\frac{\partial}{\partial y}\left(\alpha\frac{\partial f}{\partial y}\right). \qquad (2.26)
\end{aligned}
$$

2.2.5 Velocity Vector in Nonplanar Curvilinear Coordinates

The final differential quantity we want to express in terms of curvilinear coordinates is the velocity vector $\vec{v}$. It is expressed as

$$\vec{v} = v_1\vec{e}_1 + v_2\vec{e}_2 + v_3\vec{e}_3 = v_s\vec{e}_s + v_x\vec{e}_x + v_y\vec{e}_y,$$

and similar to before, we define

$$\vec{v}^{ct} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \end{pmatrix}, \quad \vec{v}^C = \begin{pmatrix} v_s \\ v_x \\ v_y \end{pmatrix},$$

and we have $\vec{v}^{ct} = \hat{O}\cdot\vec{v}^C$. To determine the velocity expressed in curvilinear coordinates, we differentiate the position vector $\vec{r}$ with respect to time t; from (2.13), we have

$$\vec{v}^{ct} = \frac{d\vec{r}}{dt} = \sum_{k=1}^{3}\frac{d}{dt}\{\vec{R}\cdot\vec{e}_k + xO_{k2} + yO_{k3}\}\vec{e}_k$$

$$= \sum_{k=1}^{3} \left\{ O_{k1}\dot{s} + O_{k2}\dot{x} + O_{k3}\dot{y} + \dot{s}\frac{dO_{k2}}{ds}x + \dot{s}\frac{dO_{k3}}{ds}y \right\} \vec{e}_k$$

$$= \hat{O} \cdot \begin{pmatrix} \dot{s} \\ \dot{x} \\ \dot{y} \end{pmatrix} + \dot{s}\frac{d\hat{O}}{ds} \cdot \begin{pmatrix} 0 \\ x \\ y \end{pmatrix} = \hat{O} \cdot \left\{ \begin{pmatrix} \dot{s} \\ \dot{x} \\ \dot{y} \end{pmatrix} + \dot{s}\hat{O}^t \cdot \frac{d\hat{O}}{ds} \cdot \begin{pmatrix} 0 \\ x \\ y \end{pmatrix} \right\}$$

$$= \hat{O} \cdot \left\{ \begin{pmatrix} \dot{s} \\ \dot{x} \\ \dot{y} \end{pmatrix} + \dot{s}\hat{T} \cdot \begin{pmatrix} 0 \\ x \\ y \end{pmatrix} \right\} = \hat{O} \cdot \begin{pmatrix} \dot{s}\,(1 - \tau_3 x + \tau_2 y) \\ \dot{x} - \dot{s}\tau_1 y \\ \dot{y} + \dot{s}\tau_1 x \end{pmatrix},$$

where (2.1) is used from the first line to the second line. Comparing with the previous equation, we see that the velocity expressed in terms of curvilinear coordinates is given by

$$\vec{v}^C = \begin{pmatrix} v_s \\ v_x \\ v_y \end{pmatrix} = \begin{pmatrix} \dot{s} \cdot (1 - \tau_3 x + \tau_2 y) \\ \dot{x} - \dot{s}\tau_1 y \\ \dot{y} + \dot{s}\tau_1 x \end{pmatrix} = \begin{pmatrix} \dot{s}\alpha \\ \dot{x} - \dot{s}\tau_1 y \\ \dot{y} + \dot{s}\tau_1 x \end{pmatrix}, \tag{2.27}$$

where $\alpha = 1 - \tau_3 x + \tau_2 y$ as (2.14). For future reference, we note that because of the orthonormality of $\hat{O}$, we also have the relationships

$$v^2 = \vec{v}^{ct} \cdot \vec{v}^{ct} = \vec{v}^C \cdot \vec{v}^C \tag{2.28}$$

$$\vec{v}^{ct} \cdot \vec{A}^{ct} = \vec{v}^C \cdot \vec{A}^C. \tag{2.29}$$

2.3 Dynamics in Nonplanar Curvilinear Coordinates

2.3.1 Electromagnetic Fields and Lorentz Force

In the Cartesian system, the electric field $\vec{E}$ and the magnetic field $\vec{B}$ are expressed in terms of the scalar potential Φ and the vector potential $\vec{A}$ as

$$\vec{E}^{ct} = -\mathrm{grad}^{ct}\Phi - \frac{\partial \vec{A}^{ct}}{\partial t} = -\vec{\nabla}^{ct}\Phi - \frac{\partial \vec{A}^{ct}}{\partial t}$$

$$\vec{B}^{ct} = \mathrm{curl}^{ct}\vec{A} = \vec{\nabla}^{ct} \times \vec{A}^{ct}.$$

Our goal here is to express these fields and the Lorentz force in terms of the curvilinear coordinates. We need to find $\vec{E}^C$ and $\vec{B}^C$ and their relationships to the potentials such that

$$\vec{E}^{ct} = \hat{O} \cdot \vec{E}^C, \text{ and } \vec{B}^{ct} = \hat{O} \cdot \vec{B}^C,$$

where

$$\vec{E}^{ct} = \begin{pmatrix} E_1 \\ E_2 \\ E_3 \end{pmatrix}, \ \vec{E}^C = \begin{pmatrix} E_s \\ E_x \\ E_y \end{pmatrix}, \text{ and } \vec{B}^{ct} = \begin{pmatrix} B_1 \\ B_2 \\ B_3 \end{pmatrix}, \ \vec{B}^C = \begin{pmatrix} B_s \\ B_x \\ B_y \end{pmatrix}.$$

Using the differential operators from the last section, we have

$$\begin{aligned} \vec{E}^C &= \hat{O}^t \vec{E}^{ct} = \hat{O}^t \left(-\vec{\nabla}^{ct}\Phi - \frac{\partial \vec{A}^{ct}}{\partial t} \right) = -\hat{O}^t \vec{\nabla}^{ct}\Phi - \hat{O}^t \frac{\partial \vec{A}^{ct}}{\partial t} \\ &= -\vec{\nabla}^C \Phi - \frac{\partial}{\partial t}(\hat{O}^t \vec{A}^{ct}) = -\vec{\nabla}^C \Phi - \frac{\partial \vec{A}^C}{\partial t} = -\begin{pmatrix} \nabla_s \\ \nabla_x \\ \nabla_y \end{pmatrix} \Phi - \frac{\partial}{\partial t} \begin{pmatrix} A_s \\ A_x \\ A_y \end{pmatrix}, \end{aligned}$$

or explicitly expressed in terms of partials of curvilinear coordinates:

$$\vec{E}^C = \begin{pmatrix} -\dfrac{1}{\alpha}\left(\dfrac{\partial}{\partial s} + \tau_1 y \dfrac{\partial}{\partial x} - \tau_1 x \dfrac{\partial}{\partial y}\right)\Phi - \dfrac{\partial A_s}{\partial t} \\ -\dfrac{\partial \Phi}{\partial x} - \dfrac{\partial A_x}{\partial t} \\ -\dfrac{\partial \Phi}{\partial y} - \dfrac{\partial A_y}{\partial t} \end{pmatrix}. \tag{2.30}$$

The magnetic field $\vec{B}^C$can be determined in a straightforward way from the transformation rule for the curl (2.25), and we have

$$\vec{B}^C = \mathrm{curl}^C \vec{A} = \begin{pmatrix} \nabla_x A_y - \nabla_y A_x \\ \nabla_y A_s - \nabla_s A_y + \dfrac{1}{\alpha}(\tau_2 A_s - \tau_1 A_x) \\ \nabla_s A_x - \nabla_x A_s + \dfrac{1}{\alpha}(\tau_3 A_s - \tau_1 A_y) \end{pmatrix}$$

$$= \begin{pmatrix} \dfrac{\partial A_y}{\partial x} - \dfrac{\partial A_x}{\partial y} \\ \dfrac{\partial A_s}{\partial y} - \dfrac{1}{\alpha}\left(\dfrac{\partial}{\partial s} + \tau_1 y\dfrac{\partial}{\partial x} - \tau_1 x\dfrac{\partial}{\partial y}\right) A_y + \dfrac{1}{\alpha}(\tau_2 A_s - \tau_1 A_x) \\ \dfrac{1}{\alpha}\left(\dfrac{\partial}{\partial s} + \tau_1 y\dfrac{\partial}{\partial x} - \tau_1 x\dfrac{\partial}{\partial y}\right) A_x - \dfrac{\partial A_s}{\partial x} + \dfrac{1}{\alpha}(\tau_3 A_s - \tau_1 A_y) \end{pmatrix}. \quad (2.31)$$

In the Cartesian system, the Lorentz force per unit charge, denoted as $\vec{f}$, is given as

$$\vec{f}^{ct} = \vec{E}^{ct} + \vec{v}^{ct} \times \vec{B}^{ct},$$

and we want to find $\vec{f}^C$ such that $\vec{f}^{ct} = \hat{O} \cdot \vec{f}^C$, where we write the components as

$$\vec{f}^{ct} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}, \quad \vec{f}^C = \begin{pmatrix} f_s \\ f_x \\ f_y \end{pmatrix}.$$

Because of the orthonormality of $\hat{O}$, we first observe

$$\begin{aligned} \vec{f}^C &= \hat{O}^t \vec{f}^{ct} = \hat{O}^t(\vec{E}^{ct} + \vec{v}^{ct} \times \vec{B}^{ct}) = \hat{O}^t(\hat{O}\vec{E}^C + (\hat{O}\vec{v}^C) \times (\hat{O}\vec{B}^C)) \\ &= \vec{E}^C + \vec{v}^C \times \vec{B}^C = \begin{pmatrix} E_s + v_x B_y - v_y B_x \\ E_x + v_y B_s - v_s B_y \\ E_y + v_s B_x - v_x B_s \end{pmatrix}, \end{aligned}$$

where the invariance of the cross product under SO(3) transformations (2.6) was used. Expressed in terms of curvilinear coordinates, we have explicitly for the components of $\vec{f}^C$.

$$\begin{aligned} f_s &= E_s + v_x B_y - v_y B_x \\ &= -\nabla_s \Phi - \frac{\partial A_s}{\partial t} + v_x\left\{\nabla_s A_x - \nabla_x A_s + \frac{1}{\alpha}(\tau_3 A_s - \tau_1 A_y)\right\} \\ &\quad -v_y\left\{\nabla_y A_s - \nabla_s A_y + \frac{1}{\alpha}(\tau_2 A_s - \tau_1 A_x)\right\} \\ &= -\nabla_s \Phi - \frac{dA_s}{dt} + \dot{s}\frac{\partial A_s}{\partial s} + \dot{x}\frac{\partial A_s}{\partial x} + \dot{y}\frac{\partial A_s}{\partial y} + v_x \nabla_s A_x - v_x \frac{\partial A_s}{\partial x} \end{aligned}$$

$$
\begin{aligned}
& +\frac{v_x}{\alpha}(\tau_3 A_s - \tau_1 A_y) - v_y\frac{\partial A_s}{\partial y} + v_y\nabla_s A_y - \frac{v_y}{\alpha}(\tau_2 A_s - \tau_1 A_x) \\
= \; & -\frac{dA_s}{dt} - \nabla_s(\Phi - v_s A_s - v_x A_x - v_y A_y) - A_s\nabla_s v_s - A_x\nabla_s v_x \\
& -A_y\nabla_s v_y - v_s\nabla_s A_s + \frac{v_s}{\alpha}\frac{\partial A_s}{\partial s} + (v_x + \dot{s}\tau_1 y)\frac{\partial A_s}{\partial x} + (v_y - \dot{s}\tau_1 x)\frac{\partial A_s}{\partial y} \\
& -v_x\frac{\partial A_s}{\partial x} + \frac{v_x}{\alpha}(\tau_3 A_s - \tau_1 A_y) - v_y\frac{\partial A_s}{\partial y} - \frac{v_y}{\alpha}(\tau_2 A_s - \tau_1 A_x),
\end{aligned}
$$

where (2.27) is used in the last step. Utilizing the relationships

$$
\begin{aligned}
\nabla_s v_s &= \frac{1}{\alpha}\left(-\dot{s}\frac{d\tau_3}{ds}x + \dot{s}\frac{d\tau_2}{ds}y - \dot{s}\tau_1\tau_3 y - \dot{s}\tau_1\tau_2 x\right) \\
\nabla_s v_x &= \frac{1}{\alpha}\left(-\dot{s}\frac{d\tau_1}{ds}y + \dot{s}\tau_1^2 x\right) \\
\nabla_s v_y &= \frac{1}{\alpha}\left(\dot{s}\frac{d\tau_1}{ds}x + \dot{s}\tau_1^2 y\right) \\
\nabla_s A_s &= \frac{1}{\alpha}\left(\frac{\partial A_s}{\partial s} + \tau_1 y\frac{\partial A_s}{\partial x} - \tau_1 x\frac{\partial A_s}{\partial y}\right),
\end{aligned}
$$

we have

$$
\begin{aligned}
f_s &= -\frac{dA_s}{dt} - \nabla_s(\Phi - \vec{v}^C\cdot\vec{A}^C) \\
& +\frac{A_s}{\alpha}\left(\dot{s}\frac{d\tau_3}{ds}x - \dot{s}\frac{d\tau_2}{ds}y + \dot{s}\tau_1\tau_3 y + \dot{s}\tau_1\tau_2 x + v_x\tau_3 - v_y\tau_2\right) \\
& +\frac{A_x}{\alpha}\left(\dot{s}\frac{d\tau_1}{ds}y - \dot{s}\tau_1^2 x + v_y\tau_1\right) + \frac{A_y}{\alpha}\left(-\dot{s}\frac{d\tau_1}{ds}x - \dot{s}\tau_1^2 y - v_x\tau_1\right) \\
&= -\frac{dA_s}{dt} - \nabla_s(\Phi - \vec{v}^C\cdot\vec{A}^C) \\
& +\frac{1}{\alpha}\left\{A_s\frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x)\right\}.
\end{aligned}
$$

f_x and f_y are computed in the similar but more straightforward way to yield

$$
f_x = -\frac{dA_x}{dt} - \nabla_x(\Phi - \vec{v}^C\cdot\vec{A}^C)
$$

$$f_y = -\frac{dA_y}{dt} - \nabla_y(\Phi - \vec{v}^C \cdot \vec{A}^C).$$

Thus, the Lorentz force expressed in curvilinear coordinates is

$$\begin{aligned}\vec{f}^C &= \vec{E}^C + \vec{v}^C \times \vec{B}^C = -\frac{d\vec{A}^C}{dt} - \vec{\nabla}^C(\Phi - \vec{v}^C \cdot \vec{A}^C) \\ &+\frac{1}{\alpha}\left\{A_s\frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x)\right\} \cdot \vec{e}_s, \qquad (2.32)\end{aligned}$$

knowing the Lorentz force is the first step towards determining Newton's equations of a charged particle in the curvilinear system.

The momentum of the particle $\vec{p}$ is expressed as

$$\vec{p} = p_1\vec{e}_1 + p_2\vec{e}_2 + p_3\vec{e}_3 = p_s\vec{e}_s + p_x\vec{e}_x + p_y\vec{e}_y,$$

and similar to before, we define

$$\vec{p}^{ct} = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \end{pmatrix}, \quad \vec{p}^C = \begin{pmatrix} p_s \\ p_x \\ p_y \end{pmatrix},$$

and have as before that $\vec{p}^{ct} = \hat{O} \cdot \vec{p}^C$. In the Cartesian system, a particle with the charge e obeys Newton's equation

$$\frac{d\vec{p}^{ct}}{dt} = e\vec{f}^{ct}. \qquad (2.33)$$

Now we merely have to express the momentum derivatives in terms of curvilinear coordinates:

$$\begin{aligned}\frac{d\vec{p}^{ct}}{dt} &= \frac{d}{dt}(\hat{O}\vec{p}^C) = \hat{O} \cdot \frac{d\vec{p}^C}{dt} + \frac{d\hat{O}}{dt} \cdot \vec{p}^C = \hat{O} \cdot \left(\frac{d\vec{p}^C}{dt} + \dot{s}\hat{O}^t\frac{d\hat{O}}{ds}\vec{p}^C\right) \\ &= \hat{O} \cdot \left(\frac{d\vec{p}^C}{dt} + \dot{s}\hat{T}\vec{p}^C\right) = \hat{O} \cdot \left(\frac{d\vec{p}^C}{dt} + \dot{s}\vec{\tau} \times \vec{p}^C\right),\end{aligned}$$

and then (2.33) can be written as $\hat{O} \cdot (d\vec{p}^C/dt + \dot{s}\vec{\tau} \times \vec{p}^C) = e\,\hat{O} \cdot \vec{f}^C$, or directly

$$\frac{d\vec{p}^C}{dt} + \dot{s}\vec{\tau} \times \vec{p}^C = e\,\vec{f}^C. \qquad (2.34)$$

Explicitly we then have

$$\frac{d}{dt}\begin{pmatrix} p_s \\ p_x \\ p_y \end{pmatrix} + \dot{s}\cdot\begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} \times \begin{pmatrix} p_s \\ p_x \\ p_y \end{pmatrix} = e\cdot\left[\begin{pmatrix} E_s \\ E_x \\ E_y \end{pmatrix} + \begin{pmatrix} v_s \\ v_x \\ v_y \end{pmatrix} \times \begin{pmatrix} B_s \\ B_x \\ B_y \end{pmatrix}\right]$$

$$= e\cdot\left[-\frac{d}{dt}\begin{pmatrix} A_s \\ A_x \\ A_y \end{pmatrix} - \begin{pmatrix} \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y\frac{\partial}{\partial x} - \tau_1 x \frac{\partial}{\partial y}\right) \\ \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} \cdot (\Phi - \vec{v}^C \cdot \vec{A}^C) \right.$$

$$\left. +\frac{1}{\alpha}\left\{ A_s \frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x \frac{d}{dt}(\tau_1 y) + A_y \frac{d}{dt}(-\tau_1 x)\right\} \cdot \vec{e}_s \right]. \tag{2.35}$$

2.3.2 The Lagrangian and Lagrange's Equations

Now we are ready to develop Lagrangian and Hamiltonian methods in curvilinear coordinates. Following the transformation properties of Lagrangians, it is conceptually directly possible, albeit practically somewhat involved, to obtain the Lagrangian in curvilinear coordinates. To this end, we merely have to take the Lagrangian of a charged particle in an electromagnetic field in the Cartesian system

$$L(x_1, x_2, x_3; \dot{x}_1, \dot{x}_2, \dot{x}_3; t) = -mc^2\sqrt{1 - \frac{v^2}{c^2}} - e\Phi + e\vec{v}^{ct}\cdot\vec{A}^{ct}$$

and express all Cartesian quantities in terms of the curvilinear quantities. In this respect, it is very convenient that the scalar product of the velocity with itself and with $\vec{A}$ is just the same in the Cartesian and curvilinear systems, according to (2.28) and (2.29). So the Lagrangian in the curvilinear system is obtained completely straightforwardly as

$$L(s, x, y; \dot{s}, \dot{x}, \dot{y}; t) = -mc^2\sqrt{1 - \frac{\vec{v}^{C2}}{c^2}} - e\Phi + e\vec{v}^C\cdot\vec{A}^C, \tag{2.36}$$

where

$$\vec{v}^{C2} = v_s^2 + v_x^2 + v_y^2, \quad \text{and} \quad \vec{v}^C\cdot\vec{A}^C = v_sA_s + v_xA_x + v_yA_y.$$

Here Φ and $\vec{A}^C$ are dependent on the position, i.e. $\{s, x, y\}$ and the time t. The quantities $\hat{O}$, $\hat{T}$, and hence τ_1, τ_2, τ_3 used below are dependent on s.

The derivatives of v_s, v_x, v_y with respect to $s, x, y, \dot{s}, \dot{x}, \dot{y}$ are useful in order to determine the explicit form of Lagrange's equations.

$$
\begin{array}{lll}
\dfrac{\partial v_s}{\partial \dot{s}} = \alpha, & \dfrac{\partial v_s}{\partial \dot{x}} = 0, & \dfrac{\partial v_s}{\partial \dot{y}} = 0, \\
\dfrac{\partial v_x}{\partial \dot{s}} = -\tau_1 y, & \dfrac{\partial v_x}{\partial \dot{x}} = 1, & \dfrac{\partial v_x}{\partial \dot{y}} = 0, \\
\dfrac{\partial v_y}{\partial \dot{s}} = \tau_1 x, & \dfrac{\partial v_y}{\partial \dot{x}} = 0, & \dfrac{\partial v_y}{\partial \dot{y}} = 1, \\
 & & \\
\dfrac{\partial v_s}{\partial s} = \dot{s}\,(-\dfrac{d\tau_3}{ds}x + \dfrac{d\tau_2}{ds}y), & \dfrac{\partial v_s}{\partial x} = -\dot{s}\,\tau_3, & \dfrac{\partial v_s}{\partial y} = \dot{s}\,\tau_2, \\
\dfrac{\partial v_x}{\partial s} = -\dot{s}\,\dfrac{d\tau_1}{ds}y, & \dfrac{\partial v_x}{\partial x} = 0, & \dfrac{\partial v_x}{\partial y} = -\dot{s}\,\tau_1, \\
\dfrac{\partial v_y}{\partial s} = \dot{s}\,\dfrac{d\tau_1}{ds}x, & \dfrac{\partial v_y}{\partial x} = \dot{s}\,\tau_1, & \dfrac{\partial v_y}{\partial y} = 0.
\end{array}
\tag{2.37}
$$

The Lagrange equation for x is derived as follows. Using the derivatives of v_s, v_x, v_y in (2.37), we have

$$
\begin{aligned}
\frac{\partial v^2}{\partial \dot{x}} &= 2v_s\frac{\partial v_s}{\partial \dot{x}} + 2v_x\frac{\partial v_x}{\partial \dot{x}} + 2v_y\frac{\partial v_y}{\partial \dot{x}} = 2v_x \\
\frac{\partial(\vec{v}^C \cdot \vec{A}^C)}{\partial \dot{x}} &= A_s\frac{\partial v_s}{\partial \dot{x}} + A_x\frac{\partial v_x}{\partial \dot{x}} + A_y\frac{\partial v_y}{\partial \dot{x}} = A_x \\
\frac{\partial v^2}{\partial x} &= 2v_s\frac{\partial v_s}{\partial x} + 2v_x\frac{\partial v_x}{\partial x} + 2v_y\frac{\partial v_y}{\partial x} = -2v_s\dot{s}\tau_3 + 2v_y\dot{s}\tau_1 \\
&= -2\dot{s}\,(\tau_3 v_s - \tau_1 v_y) = -2\dot{s}\,[\vec{\tau} \times \vec{v}^C]_2.
\end{aligned}
$$

So altogether we have

$$
\frac{\partial L}{\partial \dot{x}} = \frac{m}{\sqrt{1 - v^2/c^2}}v_x + e\,A_x = p_x + e\,A_x, \tag{2.38}
$$

where $\vec{p}^{ct} = m\vec{v}^{ct}/\sqrt{1 - v^2/c^2}$ and correspondingly $\vec{p}^C = m\vec{v}^C/\sqrt{1 - v^2/c^2}$ was used.

We also have

$$\frac{\partial L}{\partial x} = -\frac{m}{\sqrt{1-v^2/c^2}}\,\dot{s}\,[\vec{\tau}\times\vec{v}^C]_2 - e\,\frac{\partial}{\partial x}(\Phi - \vec{v}^C\cdot\vec{A}^C) = -\dot{s}\,[\vec{\tau}\times\vec{p}^C]_2 - e\,\frac{\partial}{\partial x}(\Phi - \vec{v}^C\cdot\vec{A}^C).$$

Thus, the Lagrange equation for x is

$$\frac{dp_x}{dt} + \dot{s}\,[\vec{\tau}\times\vec{p}^C]_2 = e\,\left[-\frac{dA_x}{dt} - \frac{\partial}{\partial x}(\Phi - \vec{v}^C\cdot\vec{A}^C)\right]. \tag{2.39}$$

The Lagrange equation for y is derived in the same way, and it is

$$\frac{dp_y}{dt} + \dot{s}\,[\vec{\tau}\times\vec{p}^C]_3 = e\,\left[-\frac{dA_y}{dt} - \frac{\partial}{\partial y}(\Phi - \vec{v}^C\cdot\vec{A}^C)\right]. \tag{2.40}$$

It is a little more complicated to derive the Lagrange equation for s. Using the derivatives of v_s, v_x, v_y in (2.37), we obtain

$$\begin{aligned}
\frac{\partial v^2}{\partial \dot{s}} &= 2v_s\alpha - 2v_x\tau_1 y + 2v_y\tau_1 x \\
\frac{\partial(\vec{v}^C\cdot\vec{A}^C)}{\partial \dot{s}} &= A_s\alpha - A_x\tau_1 y + A_y\tau_1 x \\
\frac{\partial v^2}{\partial s} &= 2v_s\dot{s}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right) - 2v_x\dot{s}\,\frac{d\tau_1}{ds}y + 2v_y\dot{s}\,\frac{d\tau_1}{ds}x \\
&= -2\dot{s}x\left[\frac{d\vec{\tau}}{ds}\times\vec{v}^C\right]_2 - 2\dot{s}y\left[\frac{d\vec{\tau}}{ds}\times\vec{v}^C\right]_3,
\end{aligned}$$

and so

$$\begin{aligned}
\frac{\partial L}{\partial \dot{s}} &= \frac{m}{\sqrt{1-v^2/c^2}}\cdot(v_s\alpha - v_x\tau_1 y + v_y\tau_1 x) + e\,(A_s\alpha - A_x\tau_1 y + A_y\tau_1 x) \\
&= (p_s + eA_s)\,\alpha - (p_x + eA_x)\,\tau_1 y + (p_y + eA_y)\,\tau_1 x
\end{aligned} \tag{2.41}$$

as well as

$$\begin{aligned}
\frac{\partial L}{\partial s} &= \frac{m}{\sqrt{1-v^2/c^2}}\cdot\left\{-\dot{s}x\left[\frac{d\vec{\tau}}{ds}\times\vec{v}^C\right]_2 - \dot{s}y\left[\frac{d\vec{\tau}}{ds}\times\vec{v}^C\right]_3\right\} - e\,\frac{\partial}{\partial s}(\Phi - \vec{v}^C\cdot\vec{A}^C) \\
&= -\dot{s}x\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_2 - \dot{s}y\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_3 - e\,\frac{\partial}{\partial s}(\Phi - \vec{v}^C\cdot\vec{A}^C).
\end{aligned}$$

Thus, the Lagrange equation for s is

$$
\begin{aligned}
&\frac{d}{dt}(p_s\alpha - p_x\tau_1 y + p_y\tau_1 x) + \dot{s}\, x\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_2 + \dot{s}\, y\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_3 \\
= \;& e\left[-\frac{d}{dt}(A_s\alpha - A_x\tau_1 y + A_y\tau_1 x) - \frac{\partial}{\partial s}(\Phi - \vec{v}^C\cdot\vec{A}^C)\right]. \qquad (2.42)
\end{aligned}
$$

The left hand side is modified as follows

$$
\begin{aligned}
&\alpha\frac{dp_s}{dt} - \tau_1 y\frac{dp_x}{dt} + \tau_1 x\frac{dp_y}{dt} + p_s(-\tau_3\dot{x} + \tau_2\dot{y}) - p_x\tau_1\dot{y} + p_y\tau_1\dot{x} \\
&+p_s(-\dot{s}\frac{d\tau_3}{ds}x + \dot{s}\frac{d\tau_2}{ds}y) - p_x\dot{s}\frac{d\tau_1}{ds}y + p_y\dot{s}\frac{d\tau_1}{ds}x + \dot{s}x\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_2 + \dot{s}y\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_3 \\
=\;& \alpha\frac{dp_s}{dt} - \tau_1 y\frac{dp_x}{dt} + \tau_1 x\frac{dp_y}{dt} - \dot{x}[\vec{\tau}\times\vec{p}^C]_2 - \dot{y}[\vec{\tau}\times\vec{p}^C]_3 \\
=\;& \alpha\left(\frac{dp_s}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_1\right) - \tau_1 y\left(\frac{dp_x}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_2\right) + \tau_1 x\left(\frac{dp_y}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_3\right) \\
&-v_s[\vec{\tau}\times\vec{p}^C]_1 - v_x[\vec{\tau}\times\vec{p}^C]_2 - v_y[\vec{\tau}\times\vec{p}^C]_3 \\
=\;& \alpha\left(\frac{dp_s}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_1\right) - \tau_1 y\left(\frac{dp_x}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_2\right) + \tau_1 x\left(\frac{dp_y}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_3\right),
\end{aligned}
$$

where (2.27) is used from the second step to the third step, and

$$
v_s[\vec{\tau}\times\vec{p}^C]_1 + v_x[\vec{\tau}\times\vec{p}^C]_2 + v_y[\vec{\tau}\times\vec{p}^C]_3 = \vec{v}^C\cdot(\vec{\tau}\times\vec{p}^C) = 0
$$

is used in the last step. So, the Lagrange equation for s simplifies to

$$
\begin{aligned}
&\alpha\left(\frac{dp_s}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_1\right) - \tau_1 y\left(\frac{dp_x}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_2\right) + \tau_1 x\left(\frac{dp_y}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_3\right) \\
=\;& e\left[-\alpha\frac{dA_s}{dt} + \tau_1 y\frac{dA_x}{dt} - \tau_1 x\frac{dA_y}{dt} + A_s\frac{d}{dt}(\tau_3 x - \tau_2 y)\right. \\
&\left.+A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x) - \frac{\partial}{\partial s}(\Phi - \vec{v}^C\cdot\vec{A}^C)\right].
\end{aligned}
$$

The equations for x and y, (2.39) and (2.40), can be used to simplify the above equation. Doing this, we obtain

$$
\alpha\left(\frac{dp_s}{dt} + \dot{s}[\vec{\tau}\times\vec{p}^C]_1\right) = e\left[-\alpha\frac{dA_s}{dt} - \left(\frac{\partial}{\partial s} + \tau_1 y\frac{\partial}{\partial x} - \tau_1 x\frac{\partial}{\partial y}\right)(\Phi - \vec{v}^C\cdot\vec{A}^C)\right.
$$

$$+A_s\frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x)\Big],$$

and with the above requirement that x and y are small enough such that $\alpha = 1 - \tau_3 x + \tau_2 y > 0$, the equation can be written as

$$\begin{aligned}\frac{dp_s}{dt} + \dot{s}\,[\vec{\tau}\times\vec{p}^C]_1 &= e\left[-\frac{dA_s}{dt} - \frac{1}{\alpha}\left(\frac{\partial}{\partial s} + \tau_1 y\frac{\partial}{\partial x} - \tau_1 x\frac{\partial}{\partial y}\right)(\Phi - \vec{v}^C\cdot\vec{A}^C)\right.\\ &\left.+\frac{1}{\alpha}\left\{A_s\frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x)\right\}\right].\end{aligned}$$

Thus the set of three Lagrange equations can be summarized as below; it apparently agrees with Newton's equations in curvilinear coordinates (2.35).

$$\begin{aligned}&\frac{d}{dt}\begin{pmatrix}p_s\\p_x\\p_y\end{pmatrix} + \dot{s}\cdot\begin{pmatrix}\tau_1\\\tau_2\\\tau_3\end{pmatrix}\times\begin{pmatrix}p_s\\p_x\\p_y\end{pmatrix}\\ &= -\frac{d}{dt}\begin{pmatrix}e\,A_s\\e\,A_x\\e\,A_y\end{pmatrix} - \frac{e}{\alpha}\cdot\begin{pmatrix}\frac{\partial}{\partial s} + \tau_1 y\frac{\partial}{\partial x} - \tau_1 x\frac{\partial}{\partial y}\\ \alpha\,\frac{\partial}{\partial x}\\ \alpha\,\frac{\partial}{\partial y}\end{pmatrix}(\Phi - \vec{v}^C\cdot\vec{A}^C)\\ &+\frac{e}{\alpha}\left\{A_s\frac{d}{dt}(\tau_3 x - \tau_2 y) + A_x\frac{d}{dt}(\tau_1 y) + A_y\frac{d}{dt}(-\tau_1 x)\right\}\,\vec{e}_s. \qquad (2.43)\end{aligned}$$

2.3.3 The Hamiltonian and Hamilton's Equations

To obtain the Hamiltonian now is also conceptually standard fare, although practically it gets somewhat involved. We adopt the curvilinear coordinates $\{s, x, y\}$ as generalized coordinates, and we denote the corresponding generalized momentum by $\vec{P}^G = \left(P_s^G, P_x^G, P_y^G\right)$. The generalized momentum is obtained via the partials of L with respect to the generalized velocities; using (2.38) and (2.41), we obtain

$$P_s^G = \frac{\partial L}{\partial \dot{s}} = (p_s + eA_s)\alpha - (p_x + eA_x)\tau_1 y + (p_y + eA_y)\tau_1 x$$

$$\begin{aligned}
P_x^G &= \frac{\partial L}{\partial \dot{x}} = p_x + eA_x \\
P_y^G &= \frac{\partial L}{\partial \dot{y}} = p_y + eA_y.
\end{aligned} \tag{2.44}$$

It is worthwhile to express the mechanical momentum $\vec{p}^{\,C}_{Mech}$, namely $\vec{p}^{\,C}$, in terms of the generalized momentum $\vec{P}^G = \left(P_s^G, P_x^G, P_y^G\right)$. By combining the above expression (2.44), we have

$$P_s^G = (p_s + eA_s)\,\alpha - P_x^G \tau_1 y + P_y^G \tau_1 x,$$

and so

$$p_s + eA_s = \frac{1}{\alpha}\left(P_s^G + P_x^G \tau_1 y - P_y^G \tau_1 x\right),$$

and altogether

$$\vec{p}^{\,C}_{Mech} = \vec{p}^{\,C} = \begin{pmatrix} \frac{1}{\alpha}\left(P_s^G + P_x^G \tau_1 y - P_y^G \tau_1 x\right) - eA_s \\ P_x^G - eA_x \\ P_y^G - eA_y \end{pmatrix}. \tag{2.45}$$

Squaring $\vec{p}^{\,C} = \gamma m \vec{v}^{\,C} = m\vec{v}^{\,C}/\sqrt{1-(\vec{v}^{\,C})^2/c^2}$ and re-organizing yields

$$(\vec{v}^{\,C})^2 = \frac{c^2(\vec{p}^{\,C})^2}{(\vec{p}^{\,C})^2 + m^2c^2},$$

and because $\vec{v}^{\,C}$ and $\vec{p}^{\,C}$ are parallel we even have

$$\vec{v}^{\,C} = \frac{c\vec{p}^{\,C}}{\sqrt{(\vec{p}^{\,C})^2 + m^2c^2}}. \tag{2.46}$$

We also observe that

$$\frac{1}{\gamma} = \sqrt{1-(\vec{v}^{\,C})^2/c^2} = \sqrt{1 - \frac{(\vec{p}^{\,C})^2}{(\vec{p}^{\,C})^2 + m^2c^2}} = \frac{mc}{\sqrt{(\vec{p}^{\,C})^2 + m^2c^2}}. \tag{2.47}$$

The Hamiltonian in the curvilinear system H is defined from the Lagrangian L (2.36) and the generalized momentum $\vec{P}^G$ (2.44) via the Legendre transformation

$$\begin{aligned}
H &= \dot{s}P_s^G + \dot{x}P_x^G + \dot{y}P_y^G - L \\
&= \dot{s}P_s^G + \dot{x}P_x^G + \dot{y}P_y^G + mc^2\sqrt{1 - \frac{\vec{v}^{\,C2}}{c^2}} + e\Phi - e\vec{v}^{\,C}\cdot\vec{A}^C,
\end{aligned}$$

and the subsequent expression in terms of only s, x, y, P_s^G, P_x^G, P_y^G and t, if this is possible. Using (2.45), (2.46) and (2.47), we have from (2.27) that

$$\begin{aligned}
\dot{s} &= \frac{v_s}{\alpha} = \frac{1}{m\gamma}\frac{1}{\alpha}\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}, \\
\dot{x} &= v_x + \dot{s}\tau_1 y = \frac{1}{m\gamma}\left[P_x^G - eA_x + \frac{\tau_1 y}{\alpha}\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}\right] \\
&= \frac{1}{m\gamma}\frac{1}{\alpha^2}\left\{\tau_1 y\, P_s^G + \left(\alpha^2 + \tau_1^2 y^2\right)\, P_x^G - \tau_1^2 xy\, P_y^G - e\tau_1 y\alpha\, A_s - e\alpha^2\, A_x\right\} \\
\dot{y} &= v_y - \dot{s}\tau_1 x \\
&= \frac{1}{m\gamma}\frac{1}{\alpha^2}\left\{-\tau_1 x\, P_s^G - \tau_1^2 xy\, P_x^G + \left(\alpha^2 + \tau_1^2 x^2\right) P_y^G + e\tau_1 x\alpha\, A_s - e\alpha^2\, A_y\right\},
\end{aligned}$$

where we used the abbreviation γ from (2.47), which is in terms of the generalized coordinates and the generalized momenta

$$\frac{1}{m\gamma} = \frac{c}{\sqrt{\dfrac{(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x - \alpha e A_s)^2}{\alpha^2} + (P_x^G - eA_x)^2 + (P_y^G - eA_y)^2 + m^2c^2}}. \tag{2.48}$$

We also have

$$\begin{aligned}
\vec{v}^C \cdot \vec{A}^C &= \frac{1}{m\gamma}\vec{p}^C \cdot \vec{A}^C \\
&= \frac{1}{m\gamma}\left[\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\} A_s \right. \\
&\qquad \left. + (P_x^G - eA_x)A_x + (P_y^G - eA_y)A_y\right],
\end{aligned}$$

and in particular it proved possible to invert the relationships between generalized velocities and generalized momenta. Hence the Hamiltonian H can be expressed in curvilinear coordinates, and it is given by

$$\begin{aligned}
H &= \frac{1}{m\gamma}\left[\frac{1}{\alpha^2}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) P_s^G + \frac{\tau_1 y}{\alpha^2}P_s^G P_x^G \right. \\
&\quad + \left\{1 + \frac{(\tau_1 y)^2}{\alpha^2}\right\}\left(P_x^G\right)^2 - \frac{\tau_1 x}{\alpha^2}P_s^G P_y^G + \left\{1 + \frac{(\tau_1 x)^2}{\alpha^2}\right\}\left(P_y^G\right)^2
\end{aligned}$$

$$
\begin{aligned}
&-\frac{2\tau_1^2 xy}{\alpha^2}P_x^G P_y^G - \frac{1}{\alpha}P_s^G eA_s - \frac{\tau_1 y}{\alpha}P_x^G eA_s - P_x^G eA_x + \frac{\tau_1 x}{\alpha}P_y^G eA_s \\
&-P_y^G eA_y - \frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right)eA_s + e^2A_s^2 \\
&-(P_x^G - eA_x)eA_x - (P_y^G - eA_y)eA_y + m^2c^2\Big] + e\Phi \\
= {}& \frac{1}{m\gamma}\left[\frac{1}{\alpha^2}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right)^2 - 2\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right)eA_s + e^2A_s^2\right. \\
&\left. +(P_x^G - eA_x)^2 + (P_y^G - eA_y)^2 + m^2c^2\right] + e\Phi \\
= {}& \frac{1}{m\gamma}\left[\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}^2\right. \\
&\left. +(P_x^G - eA_x)^2 + (P_y^G - eA_y)^2 + m^2c^2\right] + e\Phi \\
= {}& \frac{1}{m\gamma}(mc\gamma)^2 + e\Phi = mc^2\gamma + e\Phi.
\end{aligned}
$$

Explicitly, the Hamiltonian in curvilinear coordinates is

$$
\begin{aligned}
H &= c\sqrt{\frac{(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x - \alpha eA_s)^2}{\alpha^2} + (P_x^G - eA_x)^2 + (P_y^G - eA_y)^2 + m^2c^2} \\
&\quad + e\Phi, \qquad (2.49)
\end{aligned}
$$

where again $\alpha = 1 - \tau_3 x + \tau_2 y$. Thus we derive Hamilton's equations as follows.

$$
\begin{aligned}
\dot{s} &= \frac{\partial H}{\partial P_s^G} = \frac{1}{m\gamma}\frac{1}{\alpha}\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}, \\
\dot{x} &= \frac{\partial H}{\partial P_x^G} = \frac{1}{m\gamma}\left[\frac{\tau_1 y}{\alpha}\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\} + P_x^G - eA_x\right], \\
\dot{y} &= \frac{\partial H}{\partial P_y^G} = \frac{1}{m\gamma}\left[-\frac{\tau_1 x}{\alpha}\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\} + P_y^G - eA_y\right], \\
\dot{P}_s^G &= -\frac{\partial H}{\partial s} = \frac{1}{m\gamma}\left[-\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}\right. \\
&\quad \cdot\left\{\frac{1}{\alpha^2}\left(\frac{d\tau_3}{ds}x - \frac{d\tau_2}{ds}y\right)\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right)\right. \\
&\quad \left. +\frac{1}{\alpha}\left(P_x^G\frac{d\tau_1}{ds}y - P_y^G\frac{d\tau_1}{ds}x\right) - e\frac{\partial A_s}{\partial s}\right\} \\
&\quad \left. +e(P_x^G - eA_x)\frac{\partial A_x}{\partial s} + e(P_y^G - eA_y)\frac{\partial A_y}{\partial s}\right] - e\frac{\partial \Phi}{\partial s}, \qquad (2.50)
\end{aligned}
$$

$$
\begin{aligned}
\dot{P}_x^G &= -\frac{\partial H}{\partial x} = \frac{1}{m\gamma}\left[-\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}\right. \\
&\quad \cdot\left\{\frac{\tau_3}{\alpha^2}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - \frac{\tau_1}{\alpha}P_y^G - e\frac{\partial A_s}{\partial x}\right\} \\
&\quad \left. + e(P_x^G - eA_x)\frac{\partial A_x}{\partial x} + e(P_y^G - eA_y)\frac{\partial A_y}{\partial x}\right] - e\frac{\partial \Phi}{\partial x}, \qquad (2.51)
\end{aligned}
$$

$$
\begin{aligned}
\dot{P}_y^G &= -\frac{\partial H}{\partial y} = \frac{1}{m\gamma}\left[-\left\{\frac{1}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - eA_s\right\}\right. \\
&\quad \cdot\left\{-\frac{\tau_2}{\alpha^2}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) + \frac{\tau_1}{\alpha}P_x^G - e\frac{\partial A_s}{\partial y}\right\} \\
&\quad \left. + e(P_x^G - eA_x)\frac{\partial A_x}{\partial y} + e(P_y^G - eA_y)\frac{\partial A_y}{\partial y}\right] - e\frac{\partial \Phi}{\partial y}, \qquad (2.52)
\end{aligned}
$$

where the abbreviation (2.48) is used.

To verify the derivations, we check Hamilton's equations to agree with previous results. It is shown easily that the first three equations agree with (2.27). The last three equations are shown to agree with Lagrange's equations (2.39), (2.40) and (2.42). We have from (2.51)

$$
\begin{aligned}
\dot{P}_x^G &= -\frac{1}{\alpha}v_s\left\{\frac{\tau_3}{\alpha}\left(P_s^G + P_x^G\tau_1 y - P_y^G\tau_1 x\right) - \tau_1 P_y^G\right\} \\
&\quad + ev_s\frac{\partial A_s}{\partial x} + ev_x\frac{\partial A_x}{\partial x} + ev_y\frac{\partial A_y}{\partial x} - e\frac{\partial \Phi}{\partial x} \\
&= -\frac{\dot{s}\,\tau_3}{\alpha}P_s^G - \frac{\dot{s}\,\tau_1\tau_3 y}{\alpha}P_x^G + \dot{s}\,\tau_1\left(\frac{\tau_3 x}{\alpha} + 1\right)P_y^G \\
&\quad - e\left(\frac{\partial \Phi}{\partial x} - v_s\frac{\partial A_s}{\partial x} - v_x\frac{\partial A_x}{\partial x} - v_y\frac{\partial A_y}{\partial x}\right).
\end{aligned}
$$

Expressing the equation in terms of the mechanical momentum $\vec{p}^C$ rather than the generalized momentum $\vec{P}^G$ according to (2.44) and using (2.37), we have

$$
\begin{aligned}
\dot{p}_x + e\frac{dA_x}{dt} &= -\frac{\dot{s}\tau_3}{\alpha}\left\{(p_s + eA_s)\alpha - (p_x + eA_x)\tau_1 y + (p_y + eA_y)\tau_1 x\right\} \\
&\quad - \frac{\dot{s}\,\tau_1\tau_3 y}{\alpha}(p_x + eA_x) + \dot{s}\,\tau_1\left(\frac{\tau_3 x}{\alpha} + 1\right)(p_y + eA_y)
\end{aligned}
$$

$$
\begin{aligned}
&-e\frac{\partial}{\partial x}(\Phi - v_s A_s - v_x A_x - v_y A_y) - e\left(A_s\frac{\partial v_s}{\partial x} + A_x\frac{\partial v_x}{\partial x} + A_y\frac{\partial v_y}{\partial x}\right) \\
= \quad & -\dot{s}\left(\tau_3 p_s - \tau_1 p_y\right) - e\frac{\partial}{\partial x}(\Phi - \vec{v}^C \cdot \vec{A}^C),
\end{aligned}
$$

which is in agreement with the first Lagrange equation (2.39). The Hamilton equation for y is similarly modified from (2.52)

$$
\begin{aligned}
\dot{P}_y^G \quad = \quad & \frac{\dot{s}\tau_2}{\alpha}P_s^G + \dot{s}\tau_1\left(\frac{\tau_2 y}{\alpha} - 1\right)P_x^G - \frac{\dot{s}\tau_1\tau_2 x}{\alpha}P_y^G \\
& -e\left(\frac{\partial\Phi}{\partial y} - v_s\frac{\partial A_s}{\partial y} - v_x\frac{\partial A_x}{\partial y} - v_y\frac{\partial A_y}{\partial y}\right).
\end{aligned}
$$

In terms of the mechanical momentum $\vec{p}^C$, we have

$$
\dot{p}_y + e\frac{dA_y}{dt} = -\dot{s}(\tau_1 p_x - \tau_2 p_s) - e\frac{\partial}{\partial y}(\Phi - \vec{v}^C \cdot \vec{A}^C),
$$

and it again agrees with the second Lagrange equation (2.40). Similarly, the Hamilton equation for s is modified from (2.50)

$$
\begin{aligned}
\dot{P}_s^G \quad = \quad & \frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right)P_s^G + \left\{\frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right)\tau_1 y - \dot{s}\frac{d\tau_1}{ds}y\right\}P_x^G \\
& + \left\{-\frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right)\tau_1 x + \dot{s}\frac{d\tau_1}{ds}x\right\}P_y^G \\
& -e\left(\frac{\partial\Phi}{\partial s} - v_s\frac{\partial A_s}{\partial s} - v_x\frac{\partial A_x}{\partial s} - v_y\frac{\partial A_y}{\partial s}\right).
\end{aligned}
$$

In terms of the mechanical momentum $\vec{p}^C$, it takes the form

$$
\begin{aligned}
& \frac{d}{dt}\left\{(p_s + eA_s)\alpha - (p_x + eA_x)\tau_1 y + (p_y + eA_y)\tau_1 x\right\} \\
= \quad & \frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right)\left\{(p_s + eA_s)\alpha - (p_x + eA_x)\tau_1 y + (p_y + eA_y)\tau_1 x\right\} \\
& + \left\{\frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x + \frac{d\tau_2}{ds}y\right)\tau_1 y - \dot{s}\frac{d\tau_1}{ds}y\right\}(p_x + eA_x)
\end{aligned}
$$

$$+\left\{-\frac{\dot{s}}{\alpha}\left(-\frac{d\tau_3}{ds}x+\frac{d\tau_2}{ds}y\right)\tau_1 x+\dot{s}\frac{d\tau_1}{ds}x\right\}(p_y+eA_y)$$

$$-e\left(\frac{\partial\Phi}{\partial s}-v_s\frac{\partial A_s}{\partial s}-v_x\frac{\partial A_x}{\partial s}-v_y\frac{\partial A_y}{\partial s}\right),$$

and a little reorganization leads to

$$\frac{d}{dt}\left(p_s\alpha-p_x\tau_1 y+p_y\tau_1 x\right)+\dot{s}x\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_2+\dot{s}y\left[\frac{d\vec{\tau}}{ds}\times\vec{p}^C\right]_3$$

$$= e\left[-\frac{d}{dt}\left(A_s\alpha-A_x\tau_1 y+A_y\tau_1 x\right)-\frac{\partial}{\partial s}(\Phi-\vec{v}^C\cdot\vec{A}^C)\right],$$

which agrees with the third Lagrange equation (2.42), as it should.

2.3.4 Arclength as Independent Variable for the Hamiltonian

As the last step, we perform a change of the independent variable from the time t to the space coordinate s. For such an interchange, there is a surprisingly simple procedure which merely requires viewing t as a new position variable, $-H$ as the associated momentum, and $-P_s^G$ as the new Hamiltonian, and expressing it in terms of the new variables, if this is possible. Then the equations are

$$\frac{dx}{ds}=\frac{\partial(-P_s^G)}{\partial P_x^G},\quad \frac{dy}{ds}=\frac{\partial(-P_s^G)}{\partial P_y^G},\quad \frac{dt}{ds}=\frac{\partial(-P_s^G)}{\partial(-H)},$$

$$\frac{dP_x^G}{ds}=-\frac{\partial(-P_s^G)}{\partial x},\quad \frac{dP_y^G}{ds}=-\frac{\partial(-P_s^G)}{\partial y},\quad \frac{d(-H)}{ds}=-\frac{\partial(-P_s^G)}{\partial t}.$$

To begin, let us try to express $-P_s^G$ in terms of $t, x, y, -H, P_x^G, P_y^G$. From (2.49) we obtain that

$$\left\{\frac{1}{\alpha}\left(P_s^G+P_x^G\tau_1 y-P_y^G\tau_1 x\right)-eA_s\right\}^2+(P_x^G-eA_x)^2+(P_y^G-eA_y)^2+m^2c^2$$

$$=\frac{1}{c^2}(H-e\Phi)^2,$$

so

$$
\begin{aligned}
& \left(P_s^G + P_x^G \tau_1 y - P_y^G \tau_1 x - \alpha e A_s\right)^2 \\
= \;& \alpha^2 \left\{ \frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2 \right\}.
\end{aligned}
$$

Considering the case that $\vec{A} = 0$ and x and y are small, we demand p_s should be positive (and stay that way throughout); we also remind ourselves that $\alpha > 0$, and hence the choice of sign is done such that

$$
\begin{aligned}
P_s^G \;=\;& -P_x^G \tau_1 y + P_y^G \tau_1 x + \alpha e A_s \\
& + \alpha \sqrt{\frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2}.
\end{aligned}
$$

Thus, $-P_s^G$ and hence the new Hamiltonian H^s is obtained as

$$
\begin{aligned}
H^s \;=\;& -P_s^G = P_x^G \tau_1 y - P_y^G \tau_1 x - \alpha e A_s \\
& - \alpha \sqrt{\frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2}.
\end{aligned}
$$

Here, for later convenience, note

$$
\begin{aligned}
& \sqrt{\frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2} \\
= \;& \frac{1}{\alpha}\left(P_s^G + P_x^G \tau_1 y - P_y^G \tau_1 x\right) - eA_s = p_s. \qquad (2.53)
\end{aligned}
$$

Then, the equations of motion are

$$
\frac{dx}{ds} = \frac{\partial(-P_s^G)}{\partial P_x^G} = \tau_1 y + \frac{\alpha(P_x^G - eA_x)}{\sqrt{\frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2}}, \qquad (2.54)
$$

$$
\frac{dy}{ds} = \frac{\partial(-P_s^G)}{\partial P_y^G} = -\tau_1 x + \frac{\alpha(P_y^G - eA_y)}{\sqrt{\frac{1}{c^2}(H - e\Phi)^2 - (P_x^G - eA_x)^2 - (P_y^G - eA_y)^2 - m^2c^2}}, \qquad (2.55)
$$

$$\frac{dt}{ds} = \frac{\partial(-P_s^G)}{\partial(-H)} = \frac{\alpha\frac{1}{c^2}(H-e\Phi)}{\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2}}, \tag{2.56}$$

$$\begin{aligned}\frac{dP_x^G}{ds} &= -\frac{\partial(-P_s^G)}{\partial x} = P_y^G\tau_1 - e\tau_3 A_s + \alpha e\frac{\partial A_s}{\partial x} \\ &-\tau_3\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2} \\ &-\alpha e\frac{\frac{1}{c^2}(H-e\Phi)\frac{\partial\Phi}{\partial x}-(P_x^G-eA_x)\frac{\partial A_x}{\partial x}-(P_y^G-eA_y)\frac{\partial A_y}{\partial x}}{\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2}},\end{aligned} \tag{2.57}$$

$$\begin{aligned}\frac{dP_y^G}{ds} &= -\frac{\partial(-P_s^G)}{\partial y} = -P_x^G\tau_1 + e\tau_2 A_s + \alpha e\frac{\partial A_s}{\partial y} \\ &+\tau_2\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2} \\ &-\alpha e\frac{\frac{1}{c^2}(H-e\Phi)\frac{\partial\Phi}{\partial y}-(P_x^G-eA_x)\frac{\partial A_x}{\partial y}-(P_y^G-eA_y)\frac{\partial A_y}{\partial y}}{\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2}},\end{aligned} \tag{2.58}$$

$$\begin{aligned}\frac{d(-H)}{ds} &= -\frac{\partial(-P_s^G)}{\partial t} \\ &= e\alpha\left[\frac{\partial A_s}{\partial t} - \frac{\frac{1}{c^2}(H-e\Phi)\frac{\partial\Phi}{\partial t}-(P_x^G-eA_x)\frac{\partial A_x}{\partial t}-(P_y^G-eA_y)\frac{\partial A_y}{\partial t}}{\sqrt{\frac{1}{c^2}(H-e\Phi)^2-(P_x^G-eA_x)^2-(P_y^G-eA_y)^2-m^2c^2}}\right].\end{aligned} \tag{2.59}$$

For the sake of convenience and checking purposes, we replace P_x^G, P_y^G and H by $\vec{p}^C$ using (2.44) and (2.49) and with the help of (2.46) and (2.53). Then we have from

(2.54), (2.55) and (2.56)

$$\frac{dx}{ds} = \tau_1 y + \alpha \frac{p_x}{p_s} \tag{2.60}$$

$$\frac{dy}{ds} = -\tau_1 x + \alpha \frac{p_y}{p_s} \tag{2.61}$$

$$\frac{dt}{ds} = \alpha \frac{1}{p_s} \frac{\sqrt{(\vec{p}^C)^2 + m^2 c^2}}{c}.$$

And we have from (2.57)

$$\begin{aligned} \frac{dp_x}{ds} + e\frac{dA_x}{ds} &= (p_y + eA_y)\tau_1 - e\tau_3 A_s + \alpha e \frac{\partial A_s}{\partial x} - \tau_3 p_s \\ &\quad - \alpha \frac{e}{p_s} \left\{ \frac{\sqrt{(\vec{p}^C)^2 + m^2 c^2}}{c} \frac{\partial \Phi}{\partial x} - p_x \frac{\partial A_x}{\partial x} - p_y \frac{\partial A_y}{\partial x} \right\}, \end{aligned}$$

and organizing the expression using (2.37), (2.27) and (2.46) we find

$$\frac{dp_x}{ds} + \left[\vec{\tau} \times \vec{p}^C\right]_x = e\left[-\frac{dA_x}{ds} - \frac{1}{\dot{s}} \frac{\partial}{\partial x}(\Phi - \vec{v}^C \cdot \vec{A}^C)\right]. \tag{2.62}$$

In a similar way, we obtain from (2.58)

$$\frac{dp_y}{ds} + \left[\vec{\tau} \times \vec{p}^C\right]_y = e\left[-\frac{dA_y}{ds} - \frac{1}{\dot{s}} \frac{\partial}{\partial y}(\Phi - \vec{v}^C \cdot \vec{A}^C)\right], \tag{2.63}$$

and from (2.59)

$$\frac{dH}{ds} = \frac{1}{\dot{s}} \frac{\partial}{\partial t} \left[e(\Phi - \vec{v}^C \cdot \vec{A}^C)\right].$$

This concludes the derivations of dynamics in curvilinear coordinates. In particular, we have succeeded to derive the equations of motion of a particle moving in an electromagnetic field in curvilinear coordinates, with the arclength s as the independent variable. Moreover, we know that these equations of motion are Hamiltonian in nature, which has important consequences for theoretical studies.

2.4 Planar Motion

As an application of the concepts just derived, let us consider a particularly important special case, namely the situation in which the reference curve stays in the x_1-x_2 plane. This so-called planar curvilinear system occurs frequently in practice, in particular if the reference curve is an actual orbit and the fields governing the motion have a symmetry around the horizontal plane. The basis vectors in this 2D curvilinear system can be expressed by the Cartesian basis vectors via

$$\begin{aligned}
\vec{e}_y &= \vec{e}_3 \\
\vec{e}_s &= \cos\theta \vec{e}_1 - \sin\theta \vec{e}_2 \\
\vec{e}_x &= \sin\theta \vec{e}_1 + \cos\theta \vec{e}_2,
\end{aligned}$$

where θ depends on the arclength s; denoting its derivative by h, i.e.

$$h = h(s) = \frac{d\theta(s)}{ds}.$$

From (2.7), all the elements of the matrix $\hat{O}$ are determined as

$$\hat{O} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

So, the antisymmetric matrix $\hat{T}$ of (2.9) has the form

$$\begin{aligned}
\hat{T} &= \hat{O}^t \cdot \frac{d\hat{O}}{ds} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} -\sin\theta\cdot h & \cos\theta\cdot h & 0 \\ -\cos\theta\cdot h & -\sin\theta\cdot h & 0 \\ 0 & 0 & 0 \end{pmatrix} \\
&= \begin{pmatrix} 0 & h & 0 \\ -h & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix},
\end{aligned}$$

thus the elements of $\hat{T}$ and hence $\vec{\tau}$ are given as

$$\vec{\tau} = \begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -h \end{pmatrix};$$

finally, we have

$$\alpha = 1 - \tau_3 x + \tau_2 y = 1 + hx.$$

The partial differential operators in this 2D curvilinear system are, from (2.17), (2.21), (2.23), (2.25) and (2.26)

$$\begin{aligned}
\vec{\nabla}^C f &= \begin{pmatrix} \nabla_s \\ \nabla_x \\ \nabla_y \end{pmatrix} f = \begin{pmatrix} \dfrac{1}{1+hx}\dfrac{\partial}{\partial s} \\ \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{pmatrix} f \\
\mathrm{grad}^C f &= \frac{1}{1+hx}\frac{\partial f}{\partial s}\vec{e}_s + \frac{\partial f}{\partial x}\vec{e}_x + \frac{\partial f}{\partial y}\vec{e}_y \\
\mathrm{div}\vec{A} &= \frac{1}{1+hx}\frac{\partial A_s}{\partial s} + \frac{1}{1+hx}\frac{\partial}{\partial x}\{(1+hx)A_x\} + \frac{\partial A_y}{\partial y} \\
\mathrm{curl}^C \vec{A} &= \begin{pmatrix} \dfrac{\partial A_y}{\partial x} - \dfrac{\partial A_x}{\partial y} \\ \dfrac{\partial A_s}{\partial y} - \dfrac{1}{1+hx}\dfrac{\partial A_y}{\partial s} \\ \dfrac{1}{1+hx}\dfrac{\partial A_x}{\partial s} - \dfrac{1}{1+hx}\dfrac{\partial}{\partial x}\{(1+hx)A_s\} \end{pmatrix} \\
\triangle^C f &= \frac{1}{1+hx}\frac{\partial}{\partial s}\left(\frac{1}{1+hx}\frac{\partial f}{\partial s}\right) + \frac{1}{1+hx}\frac{\partial}{\partial x}\left\{(1+hx)\frac{\partial f}{\partial x}\right\} + \frac{\partial^2 f}{\partial y^2}.
\end{aligned}$$

The velocity expressed in this system is, from (2.27)

$$\vec{v}^C = \begin{pmatrix} v_s \\ v_x \\ v_y \end{pmatrix} = \begin{pmatrix} \dot{s}\,(1+hx) \\ \dot{x} \\ \dot{y} \end{pmatrix}.$$

The electromagnetic fields and the Lorentz force expressed in this system are, from (2.30), (2.31) and (2.32)

$$\vec{E}^C = \begin{pmatrix} E_s \\ E_x \\ E_y \end{pmatrix} = \begin{pmatrix} -\dfrac{1}{1+hx}\dfrac{\partial \Phi}{\partial s} - \dfrac{\partial A_s}{\partial t} \\ -\dfrac{\partial \Phi}{\partial x} - \dfrac{\partial A_x}{\partial t} \\ -\dfrac{\partial \Phi}{\partial y} - \dfrac{\partial A_y}{\partial t} \end{pmatrix}$$

$$\vec{B}^C = \begin{pmatrix} B_s \\ B_x \\ B_y \end{pmatrix} = \begin{pmatrix} \dfrac{\partial A_y}{\partial x} - \dfrac{\partial A_x}{\partial y} \\ \dfrac{\partial A_s}{\partial y} - \dfrac{1}{1+hx}\dfrac{\partial A_y}{\partial s} \\ \dfrac{1}{1+hx}\dfrac{\partial A_x}{\partial s} - \dfrac{1}{1+hx}\dfrac{\partial}{\partial x}\{(1+hx)A_s\} \end{pmatrix}$$

$$\begin{aligned} \vec{f}^C &= \begin{pmatrix} f_s \\ f_x \\ f_y \end{pmatrix} = \begin{pmatrix} E_s + v_x B_y - v_y B_x \\ E_x + v_y B_s - v_s B_y \\ E_y + v_s B_x - v_x B_s \end{pmatrix} = \begin{pmatrix} E_s + \dot{x} B_y - \dot{y} B_x \\ E_x + \dot{y} B_s - \dot{s}(1+hx) B_y \\ E_y + \dot{s}(1+hx) B_x - \dot{x} B_s \end{pmatrix} \\ &= -\frac{d}{dt}\begin{pmatrix} A_s \\ A_x \\ A_y \end{pmatrix} - \begin{pmatrix} \dfrac{1}{1+hx}\dfrac{\partial}{\partial s} \\ \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{pmatrix} (\Phi - \vec{v}^C \cdot \vec{A}^C) + \begin{pmatrix} \dfrac{A_s d/dt(-hx)}{1+hx} \\ 0 \\ 0 \end{pmatrix}. \end{aligned}$$

Thus, the equations of motion expressed in this system are, using the Newton's equation derived at (2.35)

$$\begin{aligned} \frac{d}{dt}\begin{pmatrix} p_s \\ p_x \\ p_y \end{pmatrix} + \dot{s}\begin{pmatrix} 0 \\ 0 \\ -h \end{pmatrix} \times \begin{pmatrix} p_s \\ p_x \\ p_y \end{pmatrix} &= \begin{pmatrix} \dfrac{dp_s}{dt} + \dot{s}hp_x \\ \dfrac{dp_x}{dt} - \dot{s}hp_s \\ \dfrac{dp_y}{dt} \end{pmatrix} \\ &= e\left[-\frac{d}{dt}\begin{pmatrix} A_s \\ A_x \\ A_y \end{pmatrix} - \begin{pmatrix} \dfrac{1}{1+hx}\dfrac{\partial}{\partial s} \\ \dfrac{\partial}{\partial x} \\ \dfrac{\partial}{\partial y} \end{pmatrix} (\Phi - \vec{v}^C \cdot \vec{A}^C) + \begin{pmatrix} \dfrac{A_s d/dt(-hx)}{1+hx} \\ 0 \\ 0 \end{pmatrix}\right]. \end{aligned}$$

Furthermore, the equations of motion after space-time interchange in this system are, from (2.60), (2.61), (2.62) and (2.63)

$$\frac{dx}{ds} = (1+hx)\frac{p_x}{p_s}, \tag{2.64}$$

$$\frac{dy}{ds} = (1+hx)\frac{p_y}{p_s}, \tag{2.65}$$

$$\begin{aligned} \frac{dp_x}{ds} - hp_s &= e\left[-\frac{dA_x}{ds} - \frac{1}{\dot{s}}\frac{\partial}{\partial x}(\Phi - \vec{v}^C \cdot \vec{A}^C)\right] \\ &= \frac{e}{\dot{s}}f_x = e\left[\frac{1}{\dot{s}}E_x + \frac{dy}{ds}B_s - (1+hx)B_y\right], \end{aligned} \tag{2.66}$$

$$\begin{aligned} \frac{dp_y}{ds} &= e\left[-\frac{dA_y}{ds} - \frac{1}{\dot{s}}\frac{\partial}{\partial y}(\Phi - \vec{v}^C \cdot \vec{A}^C)\right] \\ &= \frac{e}{\dot{s}}f_y = e\left[\frac{1}{\dot{s}}E_y + (1+hx)B_x - \frac{dx}{ds}B_s\right]. \end{aligned} \tag{2.67}$$

It is customary to have the equations of motion regarding the momentum slopes $a = p_x/p_0$ and $b = p_b/p_0$, instead of p_x and p_y, where p_0 is the initial momentum of the reference particle. Then the equations (2.64) and (2.65) can be expressed as

$$\frac{dx}{ds} = (1+hx)\frac{a}{p_s/p_0}, \tag{2.68}$$

$$\frac{dy}{ds} = (1+hx)\frac{b}{p_s/p_0}, \tag{2.69}$$

where $p_s/p_0 = \sqrt{(p^2 - p_x^2 - p_y^2)/p_0^2} = \sqrt{(p/p_0)^2 - a^2 - b^2}$ can be utilized. Because p_0 is s-independent, the equations (2.66) and (2.67) can be expressed as

$$\frac{da}{ds} = h\frac{p_s}{p_0} + \frac{e}{p_0}\left[\frac{1}{\dot{s}}E_x + \frac{dy}{ds}B_s - (1+hx)B_y\right], \tag{2.70}$$

$$\frac{db}{ds} = \frac{e}{p_0}\left[\frac{1}{\dot{s}}E_y + (1+hx)B_x - \frac{dx}{ds}B_s\right]. \tag{2.71}$$

Chapter 3

Transfer Maps for Elements Characterized by Measured Fields

The use of transfer maps is widespread in the design and study of particle optical systems such as accelerators, spectrometers, beamlines, electron microscopes as well as glass optical system. Combined with the Differential Algebraic (DA) techniques [3] [4] [5] [8], computation of Taylor transfer maps to high order are now performed extensively. The higher order aberrations obtained in this approach offer a more efficient analysis of the particle optical systems, and hence often simplify the design of a system.

One of the crucial particle optical systems which require the knowledge of high order aberrations are modern high resolution spectrographs for nuclear physics. Their large phase space acceptance sometimes turns out to demand a careful study of higher order aberrations up to seventh order [16]. To this end, the most critical question is that the simulation treats the field as precisely as possible. Since such spectrographs use large aperture magnets, a careful consideration of the fringe fields is indispensable. Some efficient methods to take account of the fringe field were discussed in the references [36] [37] [39] [38] in detail.

In this chapter, we will discuss a technique to compute Taylor transfer maps for the fields from measured data in the framework of the DA methods. We limit ourselves to the common situation in which measurements only exist in the midplane of the device, based on the understanding that the out-of-plane fields can then be calculated. The analysis in this chapter is based on this paradigm, and is applicable whenever the out-of-plane expansion does indeed represent the true fields. The method has been implemented in the the DA based code COSY INFINITY [11] [51], and has been used for the simulations of various spectrographs, including the S800 spectrograph [60] at the National Superconducting Cyclotron Laboratory at Michigan State University and the spectrographs at Jefferson Laboratory. Figure 3.1 shows the measured field data of a bending magnet of the S800 spectrograph[24] [23] [61] [62] [63], and these field data are used for the computation of the transfer map of the system.

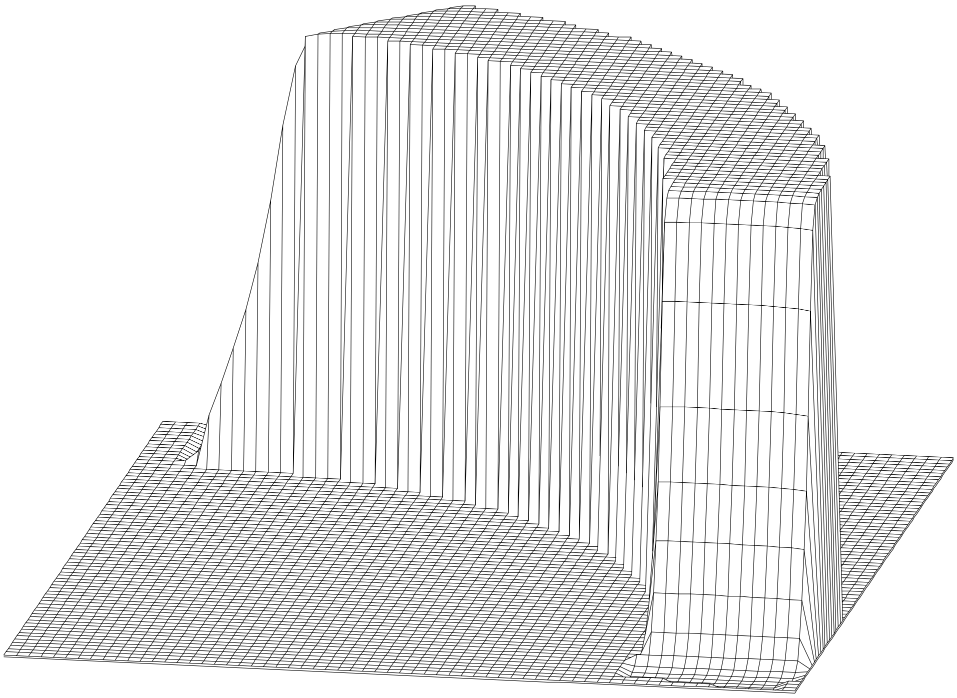

Figure 3.1: Measured field of a bending magnet of the S800 spectrograph. The picture shows the field at 65×74 points.

If there is reliable knowledge of the error bounds of the system, the method can be combined with the Remainder-enhanced Differential Algebraic (RDA) approach to obtain rigorous bounds for the induced errors in the transfer map. The topics related to the RDA are covered in the next chapters.

3.1 Wavelet Representations

For the purpose of including measured midplane field data as a part of a particle optical system, a good interpolation method is required to obtain a field value from the limited information contained in the measured field data. The interpolation method has to fit to the DA technique, and in particular the result of the interpolation has to be differentiable as often as needed. Furthermore, the interpolation should be localized in the sense that for the interpolated fields at a certain point, only the fields at neighboring points contribute.

Both of these requirements are met by the approach of wavelets [27]. Since our target data is more or less smooth as typically shown in Figure 3.1, we chose the method of Gaussian wavelets, which assures the required differentiability and locality.

Assume a set of data Y_i is given at N equidistant points x_i for $i = 1, ..., N$. Then, the interpolated value at a point x is expressed as

$$V(x) = \sum_{i=1}^{N} \mathrm{Y}_i \frac{1}{\sqrt{\pi}\mathrm{S}} \exp\left[-\frac{(x-x_i)^2}{\triangle x^2 \mathrm{S}^2}\right],$$

where $\triangle x$ is the distance of two neighbouring points x_i and x_{i+1}, and S is the factor to control the width of Gaussian wavelets.

Figure 3.2 shows how the Gaussian interpolation as a sum of Gaussian wavelets works for several one dimensional functions, including linear functions, a trigonometric function and a Gaussian function. The data values Y_i are supplied by taking their function values at each point x_i to simulate the function.

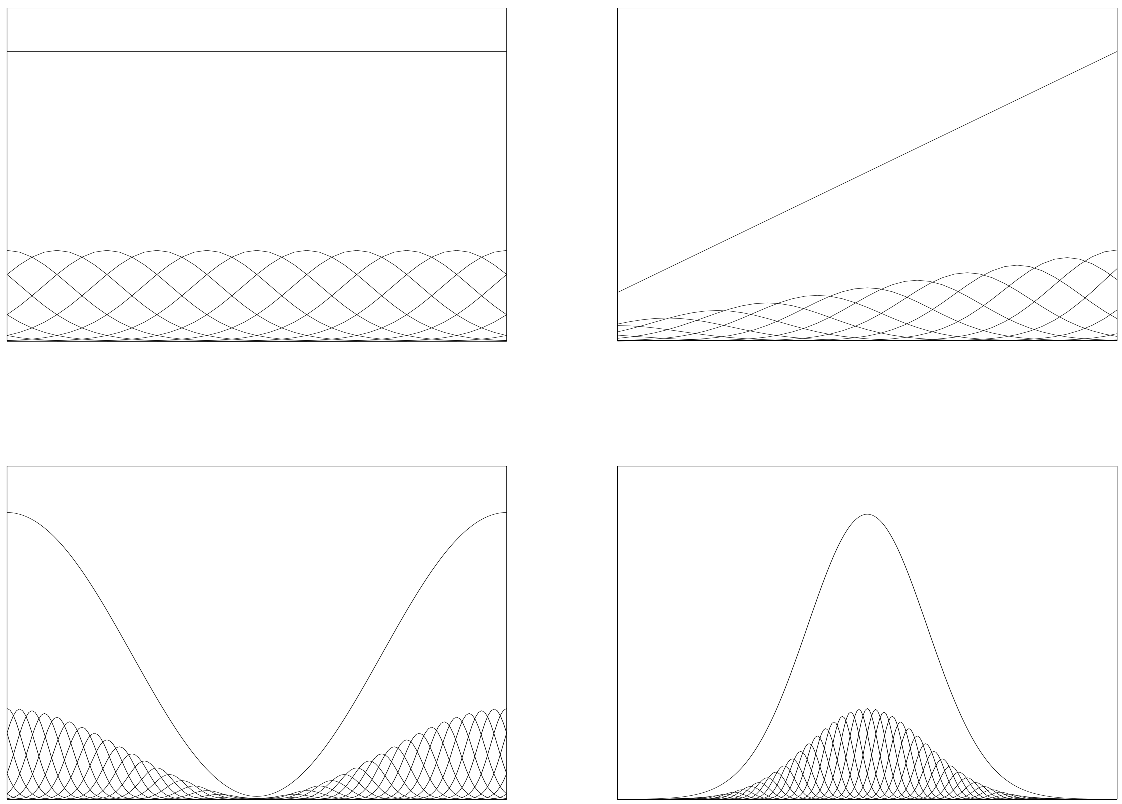

Figure 3.2: Gaussian wavelets representation for $f(x) = 1$ (upper left), $f(x) = x + 1$ (upper right), $f(x) = \cos x + 1$ (lower left) and $f(x) = \exp(-x^2)$ (lower right).

Table 3.1: Accuracy of the Gaussian wavelets representation for one dimensional functions.

Test Function	Number of Gaussian Wavelets N	Average Error	Maximum Error
1	10	2.6×10^{-14}	2.6×10^{-14}
$x+1$	10	1.5×10^{-14}	2.6×10^{-14}
$\cos(x)+1$	400	6.3×10^{-5}	1.0×10^{-4}
$\exp(-x^2)$	600	4.6×10^{-5}	1.6×10^{-4}

There are several strategies for choosing the height of each Gaussian. One approach is to set up a local least squares problem to determine the heights that fit the data optimally. While this approach usually yields the best approximation, it does not preserve the integral of the function, which for our cases is very desirable. Also, the method is not particularly well suited for the treatment of noisy data, which benefits from the smoothing effect achieved by wider Gaussians.

Both of these criteria are important for our situation, and so we decided to choose the approach where the height of the data Y_i directly determines the height of a Gaussian wavelet $\exp[-(x-x_i)^2/\triangle x^2 \mathrm{S}^2]$ that is placed at x_i.

The resulting interpolated function is shown in each picture, which represents the original function very well. S is chosen to be 1.8 for all the cases. Table 3.1 summarizes the accuracy of the method for those functions, although local accuracy for our purposes is of a lesser concern compared to smoothing and preservation of area. Figure 3.3 shows the behavior of derivatives of interpolated functions up to third order. As an example, the Gaussian function $f(x) = \exp(-x^2)$ is chosen to be the original function shape.

The advantage of the Gaussian function and many other wavelets is that it falls off quickly. Thus the potentially time consuming summation over all wavelets can be replaced by the summation of only the neighboring Gaussian wavelets in the range of ± 8S, which is in the vein of other wavelet transforms and greatly improves efficiency.

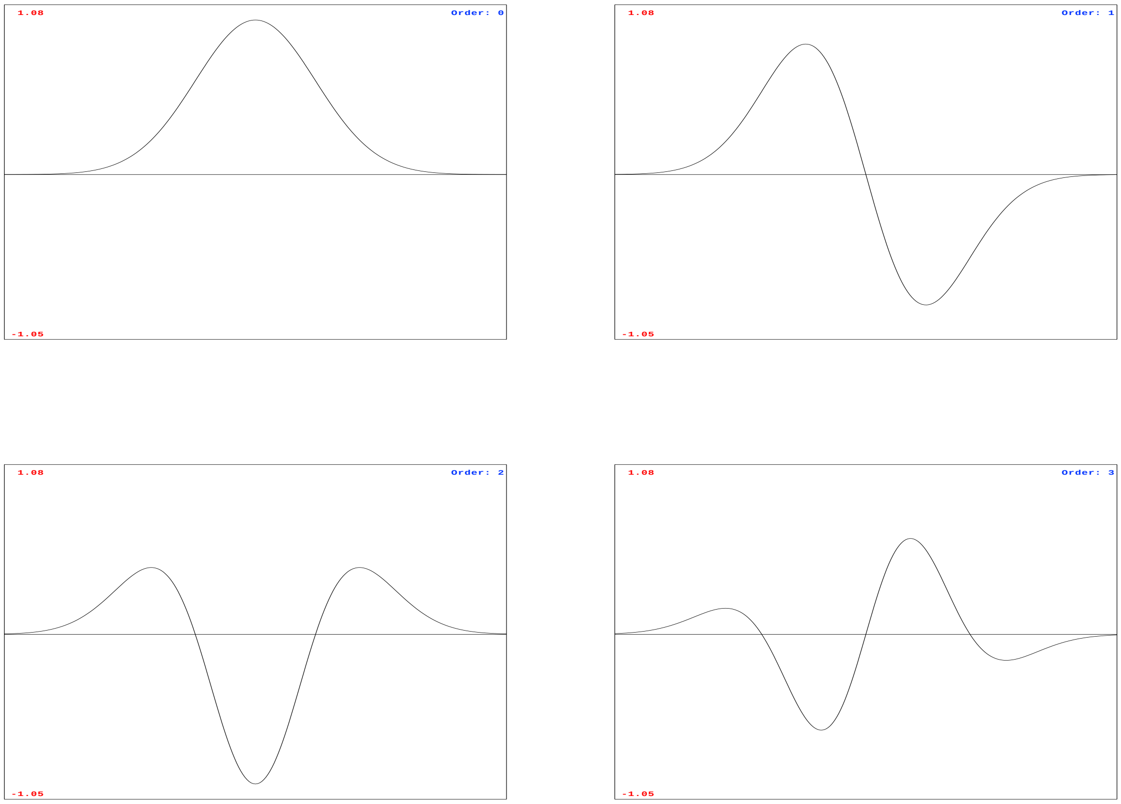

Figure 3.3: Derivatives of the function $f(x) = \exp(-x^2)$ when represented by an ensemble of Gaussian wavelets. The interpolated function (upper left), and the first (upper right), the second (lower left) and the third (lower right) order derivatives of the interpolated function.

3.2 Elements Characterized by Measured Fields

The Gaussian wavelets representation discussed above is utilized to represent fields which are specified by measured data. Because of the favorable features of the Gaussian function, the method allows the computation of the transfer map of such a magnetic field element as long as the out-of-plane expansion of the midplane data is in fact accurate enough to describe the field in the whole space.

Similar to the procedure discussed in the previous section, the measured field data is given at equidistant grid points in two dimensional Cartesian coordinates. Figure 3.4 shows how the data grid is specified and the corresponding Cartesian coordinates to the data grid. Assume a set of data $\mathrm{B}(i_x, i_z)$ is given at equidistant $N_x \times N_z$ points (x_{i_x}, z_{i_z}) for $i_x = 1, ..., N_x$ and $i_z = 1, ..., N_z$. Then, the interpolated value at a point (x, z) is expressed as

$$B_y(x,z) = \sum_{i_x=1}^{N_x} \sum_{i_z=1}^{N_z} \mathrm{B}(i_x, i_z) \frac{1}{\pi \mathrm{S}^2} \exp\left[-\frac{(x - x_{i_x})^2}{\triangle x^2 \mathrm{S}^2} - \frac{(z - z_{i_z})^2}{\triangle z^2 \mathrm{S}^2} \right], \tag{3.1}$$

where $\triangle x$ and $\triangle z$ are the grid spacing in x and z directions respectively, and S is the control factor of the width of Gaussian wavelets.

A suitable choice for the control factor S depends on the behaviour of the original supplied data. If S is too small, the mountain structure of individual Gaussian wavelets is observed. On the other hand, if S is too large, the original value supplied by the data is washed out, which can also be used effectively for purposes of smoothing noisy data. For constant fields, the suitable S is about 1.8. For quickly varying fields, it should be about 1.0. Larger values of S usually provide more accurate evaluation of the derivatives.

We implemented the Gaussian wavelet method in COSY INFINITY [11] [51] in the form of a general particle optical element to compute the transfer map from measured

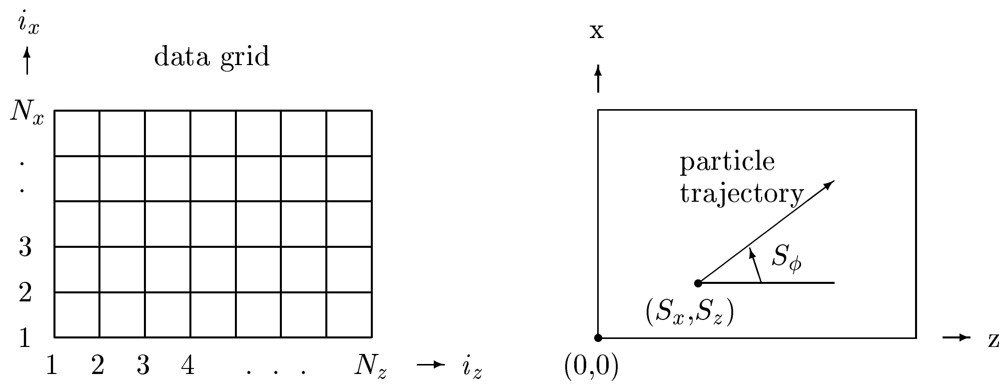

Figure 3.4: Specification of measured field data for a particle optical element in COSY INFINITY.

field data. For the purpose to specify the position of the particle in the element, the starting point (S_x, S_z) and the direction S_ϕ of the trajectory of reference particle have to be given as shown in Figure 3.4. The rest of this chapter discusses the transfer maps computed using the new method with Gaussian wavelet representation to treat measured field data in comparison with those transfer maps obtained by default in COSY INFINITY.

3.3 Transfer Maps using Linear Field Data

The first test is to check the performance of the new method using artificially created data. We begin with the two perhaps most common cases, namely the main fields of a homogeneous dipole, and an inhomogeneous dipole field with a linear inhomogeneity. We computed the high order transfer maps in four dimensions using the new method, and compared the result with the geometrically computed transfer maps which are obtained by the default procedures in COSY INFINITY.

3.3.1 The Homogeneous Dipole Field

The first example is a set of data with a constant value, which corresponds to a homogeneous magnetic dipole field. A beam of 200MeV protons was chosen to be the reference, and a total arclength of 1m was demanded in the constant magnetic field of 1Tesla. The reference trajectory bends 26.65° with a radius of 2.15m. The Taylor transfer map computed geometrically by default in COSY INFINITY is listed in Table 3.2 in the standard output notation in COSY INFINITY. Because of the limitation of the space, we show the list of the transfer map only up to third order. The artificial data were prepared with the constant value of 1Tesla positioned at 200×200 equidistant points spaced by $\triangle x$=$\triangle z$=1cm, covering the area 2m×2m. The Taylor transfer map was computed using the Gaussian wavelet representation scheme, where the control factor S in (3.1) is set to be 1.8. The difference between the two transfer maps is listed in Table 3.3, where any errors smaller than 10^{-10} are listed as 0. Very good agreement was found.

Table 3.2: Taylor transfer map of a homogeneous dipole of the radius 2.15m with a bending angle of 26.65° computed geometrically by the default procedure in COSY INFINITY.

Transfer Map of a Homogeneous Dipole by default in COSY (Radius: 2.15m, Angle: 26.65°)				
Expansion coefficients of x,a,y,b depending on the exponents of xayb				
(x,	(a,	(y,	(b,	xayb
0.8937341	-0.2086851	0	0	1000
0.9643205	0.8937341	0	0	0100
0	0	1.000000	0	0010
0	0	1.000000	1.000000	0001
-0.4680778E-01	0	0	0	2000
0.4009265	0	0	0	1100
0.1020792	-0.2242986	0	0	0200
0	0	0.4485971	0	1001
0	0	0.2284330	0	0101
-0.1142165	-0.2242986	0	0	0002
-0.1006197	0	0	0	1200
0.4309231	0	0	0	0300
0	0	0.4821603	0	0201
-0.1006197	0	0	0	1002
0.4309231	0	0	0	0102
0	0	0.4821603	0	0003

Table 3.3: Difference between the Taylor transfer maps of a homogeneous dipole computed by the default procedure in COSY INFINITY and computed using the new element based on measured field data.

Difference of Homogeneous Dipole Maps (Error > 10^{-10})				
Expansion coefficients of x,a,y,b depending on the exponents of xayb				
(x,	(a,	(y,	(b,	xayb
0	0.1636561E-09	0	0	2000
0.1045300E-09	0.3457759E-09	0	0	1100
0	0.1274906E-09	0	0	0200
0	0	0	-0.3409426E-09	1010
0	0	-0.1101961E-09	-0.3730441E-09	0110
0	0	-0.1065634E-09	-0.3581052E-09	1001
0	0	0	-0.2630541E-09	0101
-0.3376048E-09	-0.5700778E-09	0	0	0020
-0.4243799E-09	-0.4655945E-09	0	0	0011
-0.4319352E-07	-0.4920250E-07	0	0	3000
-0.1845052E-07	-0.2518897E-07	0	0	2100
0.2875618E-08	-0.9350099E-08	0	0	1200
0.2609582E-08	-0.4923973E-08	0	0	0300
0	0	0.1326292E-06	0.1607858E-06	2010
0	0	0.3670618E-07	0.5794910E-07	1110
0	0	-0.3425433E-08	0.1218639E-07	0210
0	0	0.1880360E-07	0.2800360E-07	2001
0	0	-0.5815043E-08	0.2099834E-07	1101
0	0	-0.7957903E-08	0.1552648E-07	0201
0.1161989E-06	0.1614849E-06	0	0	1020
0.1088566E-07	0.5549496E-07	0	0	0120
0.2288686E-07	0.1029208E-06	0	0	1011
-0.1375535E-07	0.9141796E-07	0	0	0111
-0.6363773E-08	0.4116490E-07	0	0	1002
-0.9467137E-08	0.5294305E-07	0	0	0102
0	0	-0.3965713E-07	-0.5944019E-07	0030
0	0	-0.1135347E-07	-0.5934921E-07	0021
0	0	0.6816748E-08	-0.4799466E-07	0012
0	0	0.3175650E-08	-0.1827340E-07	0003

3.3.2 Inhomogeneous Dipole Field

The second example is a set of data which depend linearly on the radius r of the motion of the particles. The field strength is described as

$$B_y(r) = B_0 \cdot \left(1 - n \cdot \frac{r - R_0}{R_0}\right),$$

where R_0 is the radius of the reference trajectory, B_0 is the field strength on the reference trajectory and n is the linear inhomogeneity constant. This is the field of an inhomogeneous dipole magnet with a linear inhomogeneity. The same setting was chosen, namely 200 MeV proton beam with the total arclength of 1m. By the choice of $B_0 = 1$Tesla, the same bend is defined for the reference trajectory, namely an angle of $26.65°$ and a radius of 2.15m. The linear inhomogeneity constant n was set to be 0.5.

The computations of the Taylor transfer maps were performed in a similar way as in the case of the homogeneous dipole. Table 3.4 shows the third order Taylor transfer map in six dimension computed geometrically by the default procedure in COSY INFINITY, and Table 3.5 shows the difference between the two transfer maps.

Table 3.4: Taylor transfer map of an inhomogeneous dipole of radius 2.15m and bending angle 26.65° with inhomogeneity 0.5 computed geometrically by the default procedure in COSY INFINITY.

Transfer Map of a Inhomogeneous Dipole by default in COSY (Radius: 2.15m, Angle: 26.65°, Inhomogeneity: 0.5)				
Expansion coefficients of x,a,y,b depending on the exponents of xayb				
(x,	(a,	(y,	(b,	xayb
0.9463845	-0.1062624	0	0	1000
0.9820634	0.9463845	0	0	0100
0	0	0.9463845	-0.1062624	0010
0	0	0.9820634	0.9463845	0001
0.2228769E-03	0.4766584E-01	0	0	2000
0.4486863	0.4810034E-01	0	0	1100
0.1131941	-0.2120960	0	0	0200
0	0	-0.4854609E-01	-0.4678238E-01	1010
0	0	-0.2449431E-01	-0.2360442E-01	0110
0	0	0.4323567	-0.4854609E-01	1001
0	0	0.2181490	-0.2449431E-01	0101
-0.1247084E-01	-0.2471637E-01	0	0	0020
-0.1152539	-0.2284255	0	0	0002
0.1103990E-01	0.4053979E-03	0	0	3000
0.1668532E-01	0.1144351E-01	0	0	2100
-0.1673401E-01	0.8570764E-02	0	0	1200
0.4663577	0.2705151E-01	0	0	0300
0	0	-0.1084069E-01	0.1201080E-02	2010
0	0	-0.1840065E-01	-0.9773169E-02	1110
0	0	-0.2976516E-01	-0.5771860E-03	0210
0	0	-0.9231508E-02	-0.1073923E-01	2001
0	0	0.3906643E-01	-0.2020779E-01	1101
0	0	0.4790956	-0.3070280E-01	0201
-0.5797469E-02	-0.6097624E-03	0	0	1020
-0.1991462E-02	0.5388325E-02	0	0	0120
0.3675865E-02	-0.1014555E-03	0	0	1011
-0.5031829E-01	0.1868718E-02	0	0	0111
-0.7733214E-01	-0.4731772E-02	0	0	1002
0.4297353	0.2417023E-01	0	0	0102
0	0	-0.1828454E-02	-0.3428371E-02	0030
0	0	0.1808343E-02	-0.5390550E-02	0021
0	0	-0.7450909E-01	0.9639483E-03	0012
0	0	0.4479615	-0.2512844E-01	0003

Table 3.5: Difference between the Taylor transfer maps of an inhomogeneous dipole computed by the default procedure in COSY INFINITY and computed using the measured field element.

Difference of Inhomogeneous Dipole Maps (Error > 10^{-10})				
Expansion coefficients of x,a,y,b depending on the exponents of xayb				
(x,	(a,	(y,	(b,	xayb
-0.9479292E-06	-0.1894282E-05	0	0	1000
-0.3228052E-06	-0.9815221E-06	0	0	0100
0	0	-0.9313577E-06	-0.1828878E-05	0010
0	0	-0.3127151E-06	-0.9313569E-06	0001
-0.4311228E-06	0.3774559E-06	0	0	2000
0.3413067E-05	0.3804320E-06	0	0	1100
0.9055087E-06	-0.1825359E-05	0	0	0200
0	0	-0.3898681E-06	0.5115693E-06	1010
0	0	-0.4219984E-06	0.4020970E-07	0110
0	0	0.3580534E-05	-0.3969166E-06	1001
0	0	0.1822555E-05	-0.2028945E-06	0101
0.2050580E-06	0.3808266E-06	0	0	0020
0.4247148E-06	0.1243242E-05	0	0	0011
-0.8749727E-06	-0.1466316E-05	0	0	0002
0.1363762E-06	-0.6688967E-07	0	0	3000
0.1537549E-06	0.1227274E-06	0	0	2100
-0.3419442E-06	0.5660030E-07	0	0	1200
-0.6032425E-06	0.4360596E-06	0	0	0300
0	0	-0.4121860E-07	-0.1766775E-06	2010
0	0	-0.2255669E-06	-0.2357229E-06	1110
0	0	-0.4998128E-06	-0.1999980E-07	0210
0	0	-0.1613373E-06	-0.1114431E-06	2001
0	0	0.6451915E-06	-0.2972702E-06	1101
0	0	-0.3780246E-06	-0.5160206E-06	0201
-0.8747955E-07	-0.3234010E-06	0	0	1020
0.2332158E-07	-0.2365737E-06	0	0	0120
0.2060690E-06	-0.3131300E-06	0	0	1011
-0.7502058E-06	-0.3136700E-06	0	0	0111
-0.1221911E-05	0.3597991E-07	0	0	1002
-0.1183294E-05	0.3798253E-06	0	0	0102
0	0	0.5015356E-07	0.1325949E-06	0030
0	0	0.2808341E-06	0.2199493E-06	0021
0	0	-0.1067846E-05	0.2708509E-06	0012
0	0	-0.8565441E-06	-0.3690355E-06	0003

3.4 Transfer Maps using Analytical Field Data

In the previous section, we have discussed the performance of the method to compute Taylor transfer maps for typical main fields using the Gaussian wavelet representation. In the next example we apply the method to analytically created data of a typical fringe field profile. We prepared the data such that the computation can be checked with some existing procedures in COSY INFINITY.

For the purpose to obtain the analytically described field values for the fringe field effects, we used an internal mechanism in COSY INFINITY for the treatment of the dipole field. It is based on the standardized description of the s-dependence of multipole strengths by an Enge function as in the program RAYTRACE [47]. The Enge function has the form

$$F(z) = \frac{1}{1+\exp(a_1 + a_2 \cdot (z/D) + ... + a_6 \cdot (z/D)^5)},$$

where z is the distance perpendicular to the effective field boundary, and D is the full aperture of the dipole. a_1 through a_6 are the Enge coefficients, which depend on the details of the geometry of the element, including shimming and saturation effects.

We used the same dipole magnet setup with the example in the previous section, namely a bend angle of 26.65° and a radius of 2.15m. For the purpose to have enough fringing region, we have drifts with the length of 0.4m before and after the dipole, which results in a total arclength of 1.8m. The aperture of the dipole magnet was set to be 10cm, and the default Enge coefficients for a dipole in COSY INFINITY were used. The analytical field distribution, which has a maximum field strength of 1 Tesla, is shown in Figure 3.5. In the upper picture, a beam enters from the center at the left edge rightward, then goes clockwise to the right down.

We computed the first order transfer maps in four dimension using the default

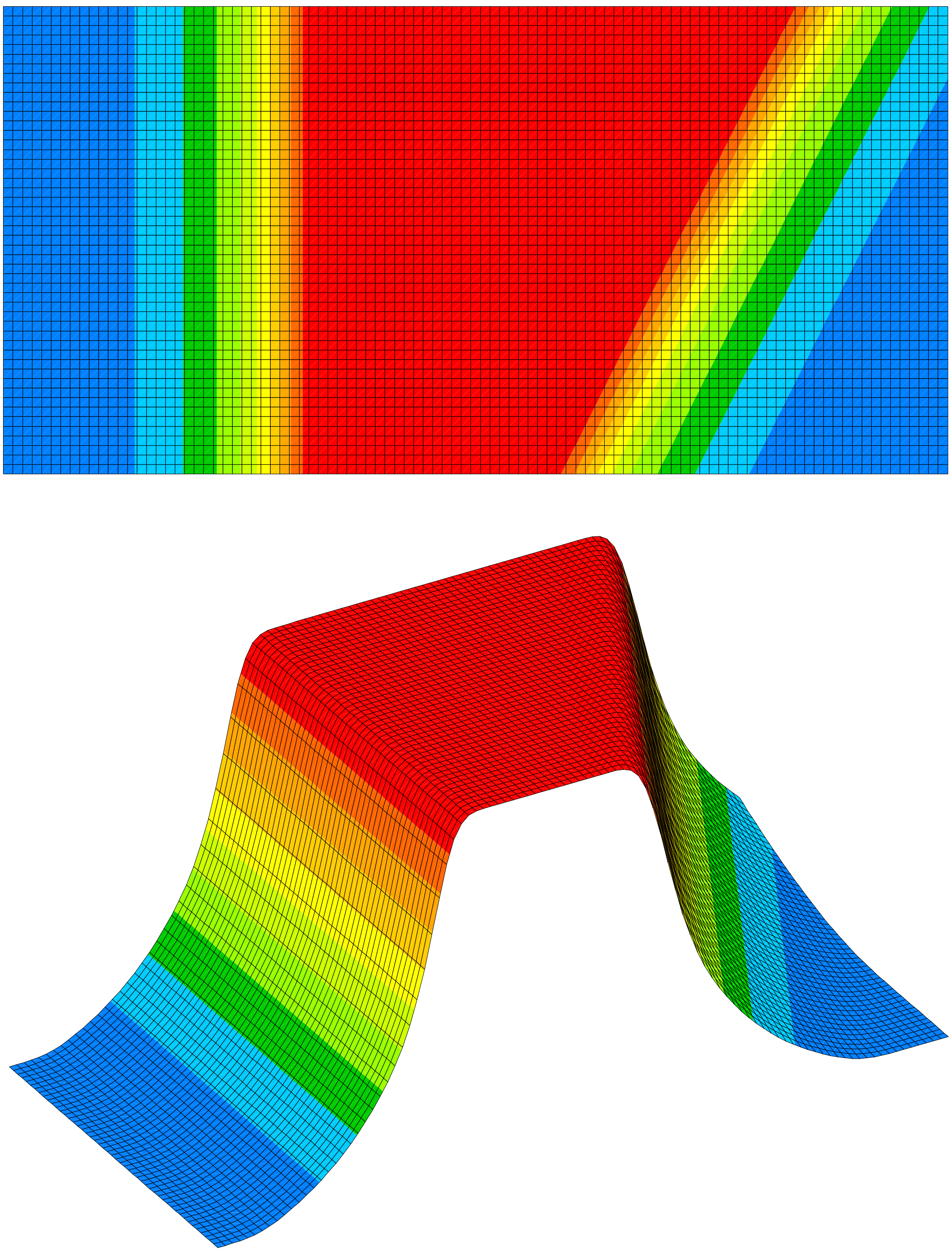

Figure 3.5: Analytical dipole field distribution with fringe field effects. Bend: 26.65°. Radius: 2.15m. Aperture: 10cm. In the upper picture, a beam enters from the center at the left rightward then goes clockwise. The lower picture is viewed from a lower angle.

procedure in COSY INFINITY and using the analytically created field data with the Gaussian wavelet representation method with various values of the control factor S ranging from 1.8 to 5.0. The result in Table 3.6 shows the transfer map obtained by the default procedure in COSY INFINITY, and the difference between the default computation and the field data computation. The analytical data was prepared at $N_x \times N_z = 100 \times 200$ equidistant points spaced by $\triangle x$=$\triangle z$=1cm, which covers the area 1m$\times$2m.

Because of the somewhat coarse spacing, the resulting transfer map agrees with the true map to only about 10^{-3}. In the next computation, a finer spacing of the data grid points is being used, resulting in a higher overall accuracy of about 5×10^{-5} up to order five.

Table 3.6: Transfer maps of a dipole with fringe field effects computed by default in COSY INFINITY, and using analytically generated field data.

Transfer Map of a Dipole with Fringe Field Effects by default in COSY (Radius: 2.15m, Angle: 26.65°, Aperture: 10cm, two drifts of 0.4m)				
Expansion coefficients of x,a,y,b depending on the exponents of xayb				
(x,	(a,	(y,	(b,	xayb
0.8102451	-0.2086852	0	0	1000
1.646121	0.8102227	0	0	0100
0	0	1.029143	0.3268649E-01	0010
0	0	1.820497	1.029503	0001
Errors of Transfer Map using Analytical Field Data ($> 10^{-10}$)				
S=1.8				
-0.2102107E-04	-0.1463520E-05	0	0	1000
-0.4156249E-05	0.1911708E-04	0	0	0100
0	0	-0.1495750E-03	-0.1895449E-03	0010
0	0	-0.1400689E-03	-0.1901141E-03	0001
S=2.6				
-0.2377301E-04	-0.1707500E-05	0	0	1000
-0.8942274E-05	0.2260579E-04	0	0	0100
0	0	-0.3316729E-03	-0.3975747E-03	0010
0	0	-0.2919843E-03	-0.3807626E-03	0001
S=3.4				
-0.2754365E-04	-0.1973192E-05	0	0	1000
-0.1546368E-04	0.2751592E-04	0	0	0100
0	0	-0.5748396E-03	-0.6755873E-03	0010
0	0	-0.4952997E-03	-0.6357372E-03	0001
S=4.2				
-0.3237764E-04	-0.2361021E-05	0	0	1000
-0.2370848E-04	0.3368500E-04	0	0	0100
0	0	-0.8752840E-03	-0.1019264E-02	0010
0	0	-0.7472205E-03	-0.9510957E-03	0001
S=5.0				
-0.3862635E-04	-0.4325084E-05	0	0	1000
-0.3407721E-04	0.3861323E-04	0	0	0100
0	0	-0.1228760E-02	-0.1422527E-02	0010
0	0	-0.1044417E-02	-0.1320222E-02	0001

The last computation using artificially created field data assesses the prospects of computing a Taylor transfer map from the measured field data of the S800 spectrograph. The system setup consists of a dipole magnet, which has the same design parameters with those dipoles of the S800 spectrograph, and two drifts before and after the dipole. The parameters of the S800 system is listed in Table 3.8. The bend angle of the reference trajectory is 75° with a bending radius of 2.68m, and the dipole has an edge angle of 30°. The aperture is 7.62cm, and the length of the two drifts is 0.7m each, so the total arclength adds up to 4.91m. The default Enge coefficients for a dipole in COSY INFINITY were used to take the fringe field effects into account, and a beam of tritium ^{3}H with the magnetic rigidity 2.704Tesla·m was used. The resulting analytical field distribution is shown in Figure 3.6, where the maximum field strength is 1.01Tesla. Despite of the bending angle of 75°, the angle of 45° seen in Figure 3.6 is due to the exit edge angle of 30°. In the upper picture, the beam enters from the upper left rightward, then goes clockwise and ends at the lower right.

We computed the fifth order transfer maps in six dimensions using the default procedure in COSY INFINITY, and using the analytically created field data with the Gaussian wavelet representation method with the control factor S 1.8, 2.2 or 2.6. As a further check, it was calculated to what extent the transfer map breaks the symplectic symmetry, which is a general overall test for the accuracy of calculations. For reasons of space, we only list the relevant parts of the resulting maps in Table 3.7. The analytical data was prepared at $N_x \times N_z = 661 \times 801$ equidistant points spaced by $\triangle x$=$\triangle z$=5mm, which covers the area 3.3m×4.0m.

The result of the transfer maps using the data is very accurate, and shows the practical usefulness of the new method.

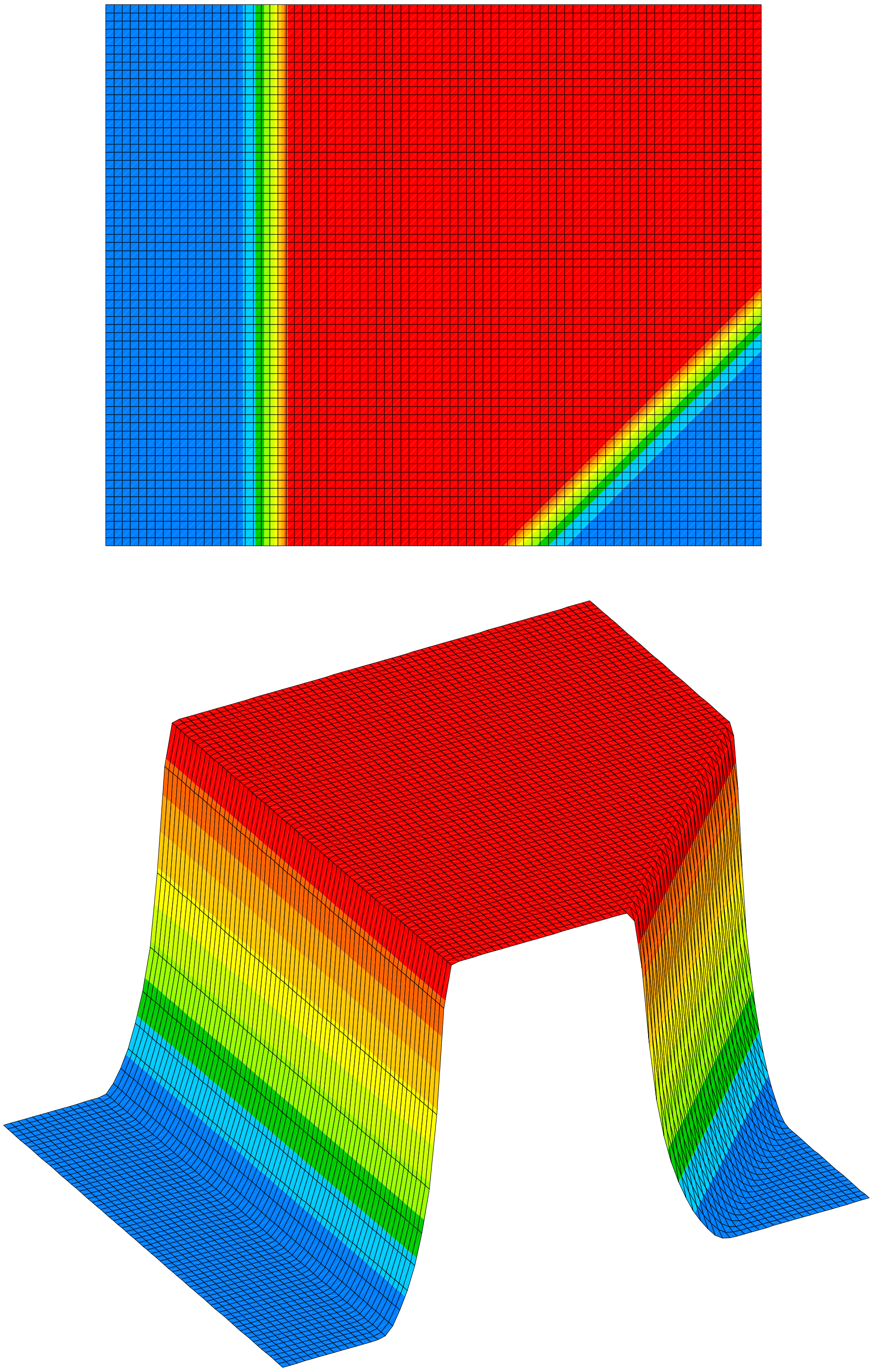

Figure 3.6: Analytical S800-like dipole field distribution with fringe field effects. In the upper picture, a beam enters from the upper left rightward, then goes clockwise. The lower picture is viewed from a lower angle.

Table 3.7: A part of the fifth order Taylor transfer map of a S800-like dipole with the fringe field effects computed by the default procedure in COSY INFINITY, and the error of transfer map using analytical field data.

Transfer Map of a S800-like Dipole with Fringe Field Effects and Two 0.7m Drifts by default in COSY (Radius: 2.682m, Angle: 75°, Aperture: 7.62cm, Exit edge angle: 30°)					
Symplectic Error: 2.07406×10^{-7}					
Expansion coefficients of x,a,y,b,t depending on the exponents of xaybδ					
(x,	(a,	(y,	(b,	(t,	xaybd
0.46777E-1	-0.30432	0	0	-0.49223	10000
3.19321	0.60316	0	0	-1.35783	01000
0	0	0.88841	-0.19817	0	00100
0	0	4.34429	0.15655	0	00010
1.50829	0.71012	0	0	0.93807	00001
0.46761	-0.51206	0	0	-0.90161	02000
-0.59510E-1	-0.13825	0	0	0.66269	01001
-0.96599	-0.31226	0	0	-0.82018	00002
0.17337	0.71513E-1	0	0	-0.38834	03000
-0.11038	-0.21087	0	0	-0.24251	04000
-0.32441	0.46850E-1	0	0	-0.22583	05000
Errors of Transfer Map using Analytical Field Data ($> 10^{-10}$)					
S=1.8, Symplectic Error: 2.46947×10^{-6}					
-0.37943E-5	-0.40253E-6	0	0	-0.17484E-5	10000
0.31532E-5	0.95363E-6	0	0	-0.40188E-5	01000
0	0	-0.10045E-3	-0.42308E-4	0	00100
0	0	-0.18879E-3	-0.14706E-3	0	00010
0.12098E-4	0.32933E-5	0	0	0.38790E-5	00001
-0.84006E-5	-0.49003E-5	0	0	-0.94809E-5	02000
-0.17281E-4	-0.29639E-5	0	0	-0.16631E-5	01001
-0.25206E-4	-0.34907E-5	0	0	-0.10033E-4	00002
0.16000E-4	0.31675E-5	0	0	-0.45384E-6	03000
-0.31202E-4	-0.80058E-5	0	0	-0.13688E-4	04000
0.70371E-3	0.99258E-4	0	0	-0.27143E-3	05000
S=2.2, Symplectic Error: 1.38298×10^{-6}					
S=2.6, Symplectic Error: 8.55948×10^{-7}					

3.5 Transfer Maps of the S800 Spectrograph

The S800 spectrograph [60] at the National Superconducting Cyclotron Laboratory at Michigan State University is a modern high resolution spectrograph for nuclear physics, which offers large phase space acceptances with $\triangle\theta = \pm 60$mr, $\triangle\phi = \pm 90$mr, and $\triangle E = \pm 5\%$. Because of the large phase space acceptances, higher order aberrations severely affect the resolution. A method to reconstruct the particle trajectories in spectrographs was presented earlier [16], where an energy resolution of 1/9,100 was recovered with fifth order reconstructive correction from an uncorrected nonlinear resolution of 1/100, almost reaching the linear resolution of 1/9,400. The limitation of the reconstructive correction method comes from various inaccuracies including the transfer map of the system. The two huge dipole magnets, which are the primary parts of the S800 spectrograph as a beam optical device, play a very important role for the transfer map, and a rough estimate shows that a knowledge of the overall fields with a relative accuracy of 10^{-4} is necessary to achieve the design energy resolution of 1/10,000.

In the following we utilize the method to include measured field data to transfer maps developed in this chapter, and assess to what extent the midplane field data are indeed sufficient to describe the field in the entire space. The laboratory layout and the system parameters of the S800 spectrograph are shown in Figure 3.7 and Table 3.8.

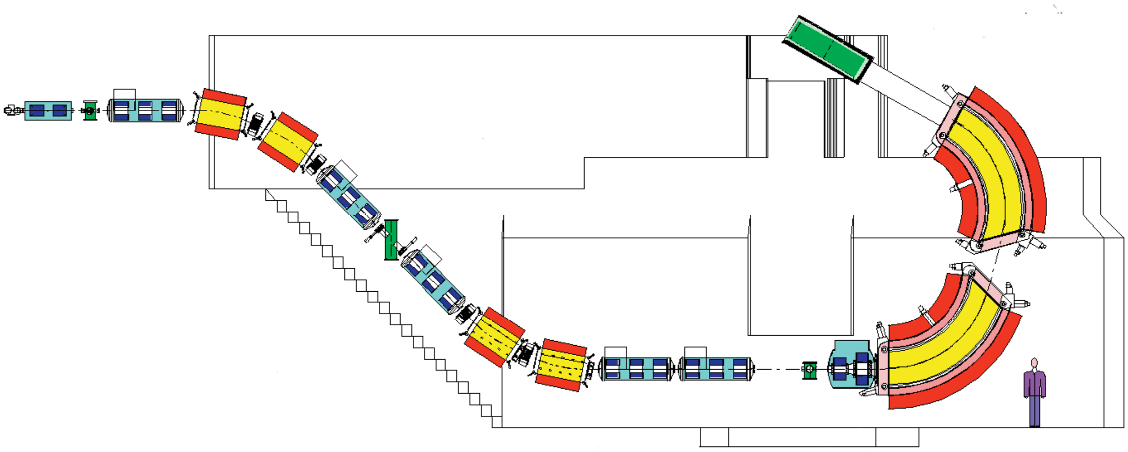

Figure 3.7: Layout of the S800 spectrograph of the National Superconducting Cyclotron Laboratory at Michigan State University. Courtesy Daniel Bazin.

Table 3.8: Design parameters of the S800 spectrograph.

Drift	$l = 60$ cm				
Quad	$l = 40$ cm	$G_{max} = 21$ T/m	$r = 0.1$ m		
Drift	$l = 20$ cm				
Quad	$l = 40$ cm	$G_{max} = 6.8$ T/m	$r = 0.2$ m		
Drift	$l = 50$ cm				
Dipole	$\rho = 2.675$ m	$B_{max} = 1.5$ T	$\phi = 75°$	$\epsilon_1 = 0°$	$\epsilon_2 = 30°$
Drift	$l = 140$ cm				
Dipole	$\rho = 2.675$ m	$B_{max} = 1.5$ T	$\phi = 75°$	$\epsilon_1 = 30°$	$\epsilon_2 = 0°$
Drift	$l = 257.5$ cm				

3.5.1 Measured Field Data of the S800 Dipole

The fields of the magnets of the S800 have been mapped [24] [23] [61] [62] [63]; for convenience, the data were taken in cylindrical coordinates whose z axis is perpendicular to the plane of Figure 3.7. The raw data was converted to the equidistant Cartesian grid data after a smoothing process to attempt the control of noise. The measured field data are prepared in a rectangular area. Figure 3.8 shows the two rectangular areas to cover the two dipole magnets. Figure 3.9 shows the field distribution of the first S800 dipole magnet, which has a maximum field strength 0.965Tesla. In the upper picture, a beam enters from the lower left rightward and travels counterclockwise. The central area of the magnet, which is seen to have a constant field in Figure 3.9, has a structure as shown in Figure 3.10 by cutting the field values to only those above 0.962Tesla.

The measured field data have been supplied for use with the Gaussian wavelet representation method in the following settings.

1. $N_x \times N_z = 328 \times 370$ equidistant point data spaced by $\triangle x$=$\triangle z$=1cm, covering the area 3.27m×3.69m.

2. $N_x \times N_z = 158 \times 184$ equidistant point data spaced by $\triangle x$=$\triangle z$=2cm, covering the area 3.14m×3.66m.

3. $N_x \times N_z = 631 \times 736$ equidistant point data spaced by $\triangle x$=$\triangle z$=5mm, covering the area 3.15m×3.675m.

The second and the third are based on the same data with the different spacing. The directions of x- and z axes follow the notation in Figure 3.4.

For the third data set, which is the finest, further details of the data are studied. Figure 3.11 covers a small rectangular area 34.5cm×34.5cm with grid points ranging i_x=82,..,151, which is around 1/6 from the bottom vertically, and i_z=331,..,400, which is around the center horizontally in the upper picture of Figure 3.9. In this region, the field values are ranging from 0.964Tesla to 0.965Tesla. The step structure seen in the top picture in Figure 3.11 is due to the fact that the data are available only to an accuracy of 6 digits. While any obvious noise from the raw data was removed, this step structure in the data will be a source of a numerical problem in the detailed high order Taylor transfer map computation. To study this aspect, the data is smoothed to double precision accuracy on computers using the Gaussian interpolation method, which is shown in the middle picture of Figure 3.11. Then the original data (upper) and the smoothed data (middle) were compared, and the result is shown in the bottom picture. The difference ranges between -6.79×10^{-6} and 7.44×10^{-6}, as expected.

This subtle difference is sufficient to induce enough noise in the out-of-plane expansion to become noticeable in the transfer map. The difference in the transfer maps range from about 1% in the linear part, and about 10% in the second order part, to as much as 100% for certain third order terms. This seems to be an indication that the common practice of midplane-only measurements has to be used with great care. Fortunately for the S800, out of plane data are available, and it is planned that they be used in the future for a more accurate representation of the overall field.

On the other hand, the S800 dipoles are rather extreme regarding the total amount of deflection of the orbits, which is one of the main reasons why the higher order part of the transfer map is so sensitive. For other systems with smaller deflection angles, the situation is much less dramatic; as an example, for certain dipoles in the Fermilab recycler ring, which have a main field strength 0.13Tesla and a bend of 2°, several digits of accuracy in the matrix elements were found [65]. In any case, for the system like the S800 spectrograph, it turned out that it is doubtful whether in-plane data are sufficient at the current level of noise to describe the out-of-plane fields sufficiently accurately without further processing.

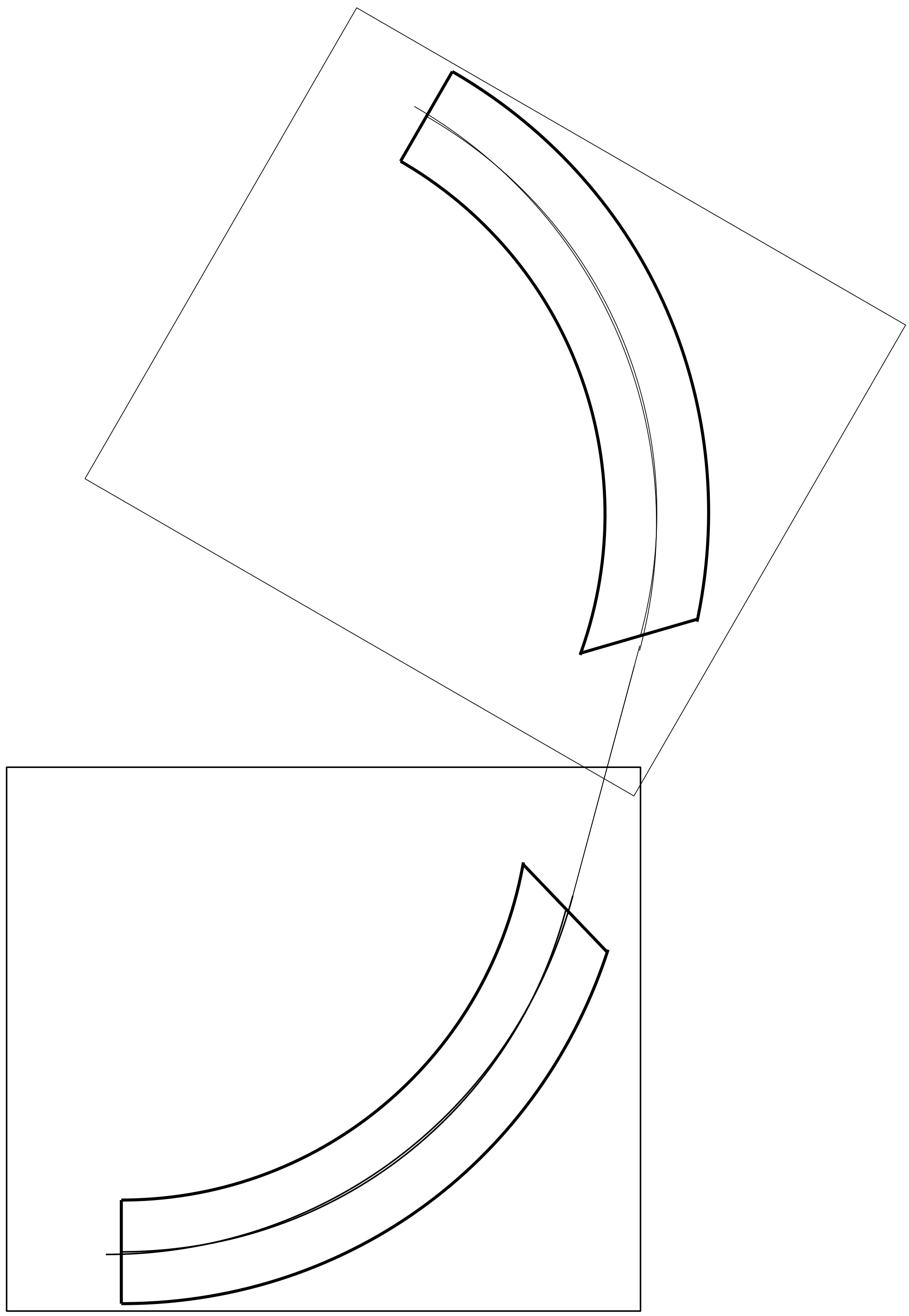

Figure 3.8: Two rectangular areas of the measured field data of the two S800 dipoles. Courtesy Daniel Bazin.

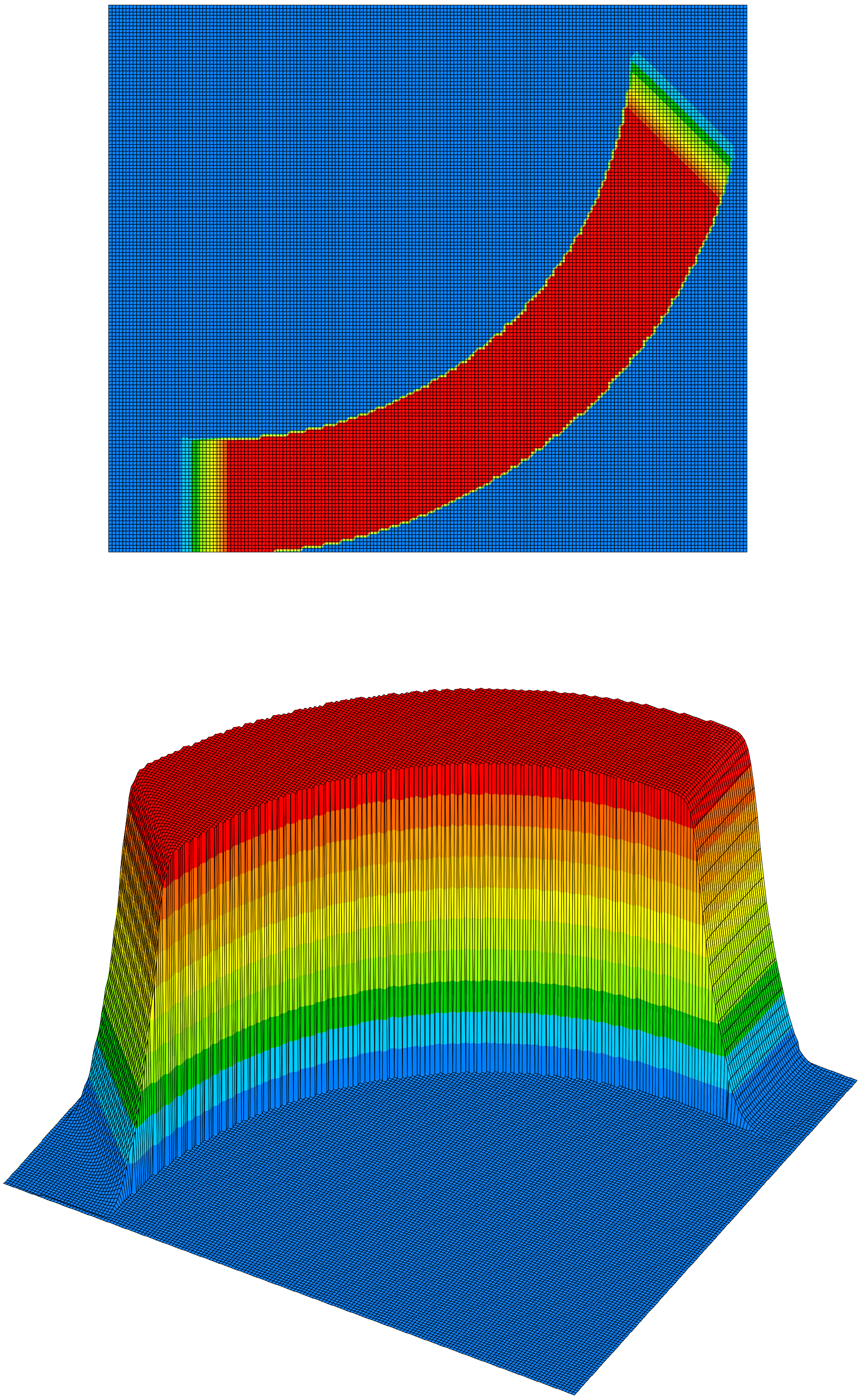

Figure 3.9: Field distribution of the S800 dipole measured data. Maximum: 0.965 Tesla. In the upper picture, a beam enters from the lower left rightward and goes counterclockwise. The lower picture is viewed from a lower angle.

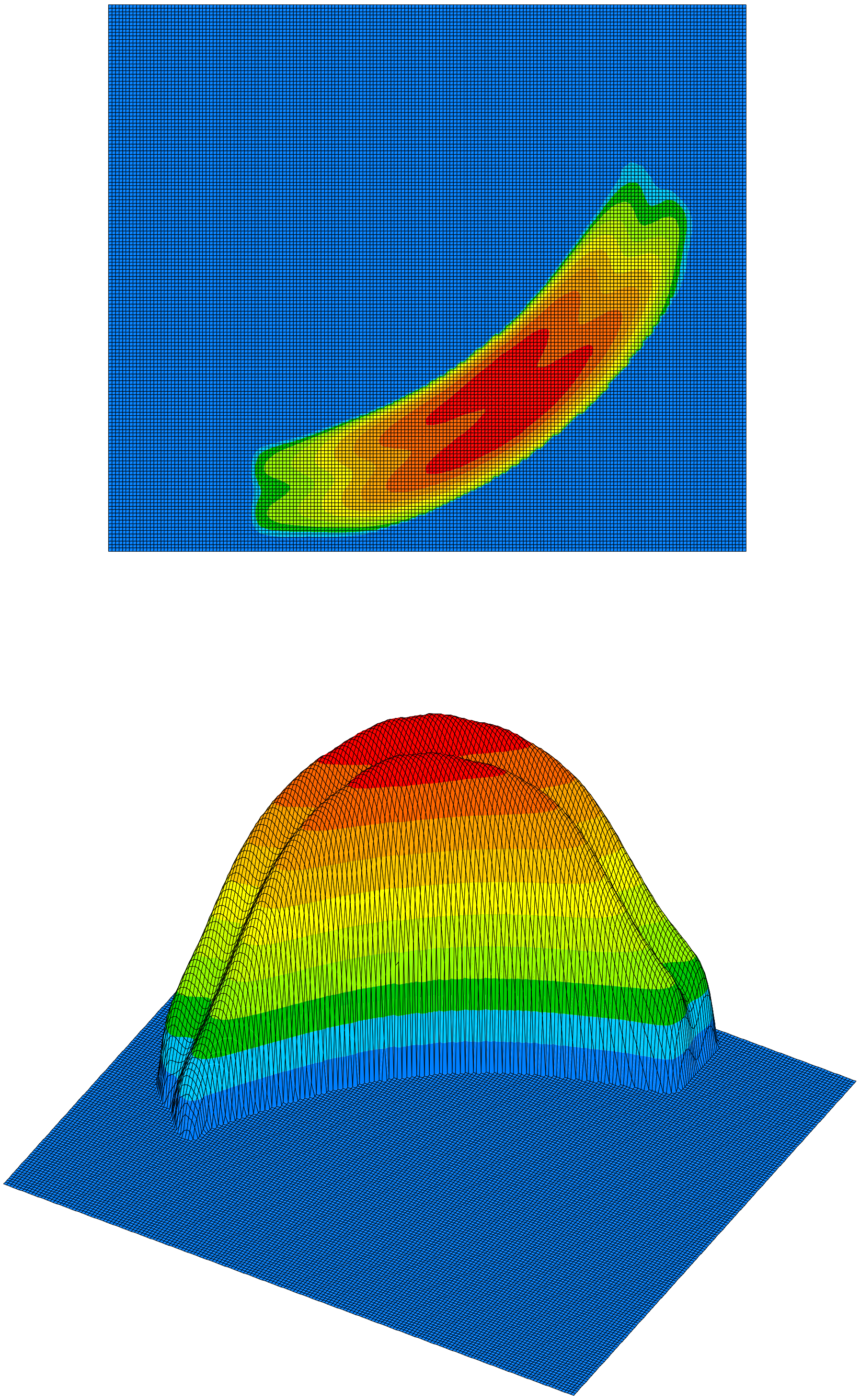

Figure 3.10: Top part of the field distribution of the S800 dipole measured data. The picture shows the field values ranging from 0.962Tesla to the maximum 0.965Tesla.

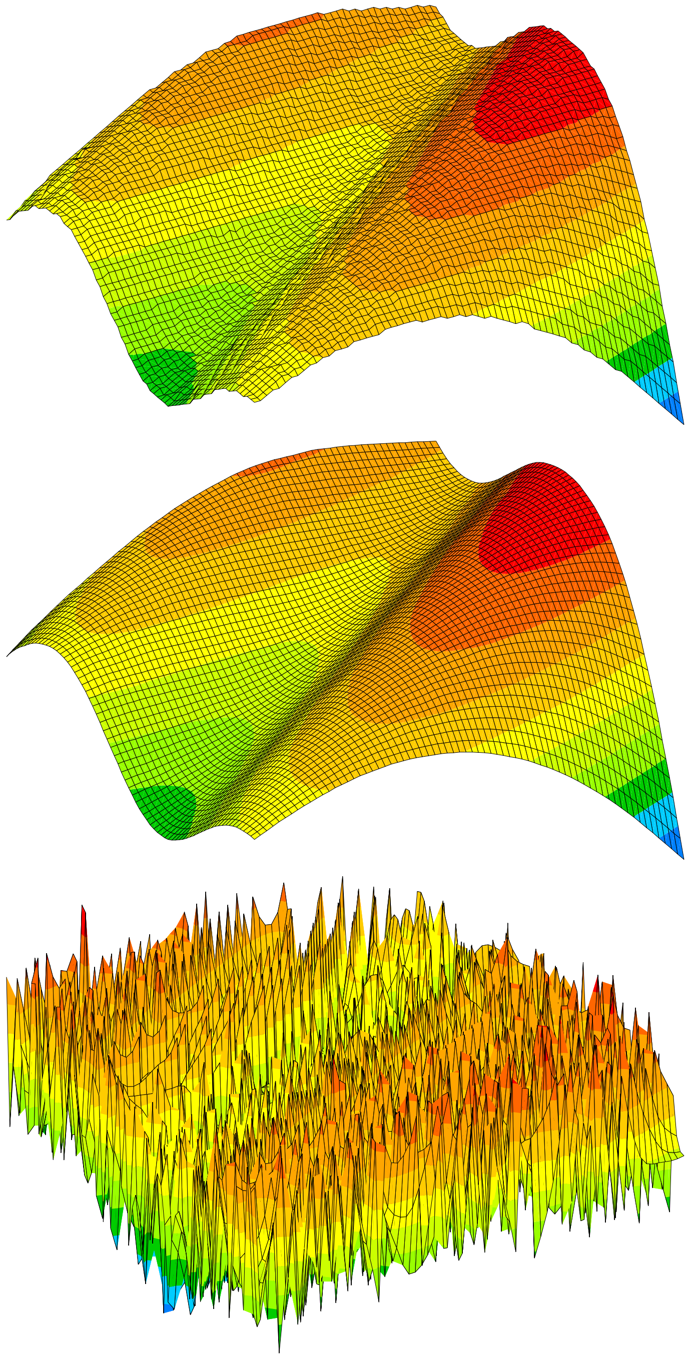

Figure 3.11: Further detail of the S800 dipole measured data in a small area 34.5cm $\times$ 34.5cm on the top region shows the step structure due to the limited data digits (top). The data is smoothed (middle), and compared (bottom).

Chapter 4

Remainder-enhanced Differential Algebra (RDA) Methods

In this chapter it is shown how, in parallel to the accumulation of derivatives, Taylor remainder bounds for all functional dependencies can be carried along the computation. Chapter 6 addresses the necessary Differential Algebraic (DA) formalism that allows the solutions of ODEs and the determination of flows. The additional computational effort is limited, and the resulting bounds are usually rather sharp, in particular at higher orders. Compared to other rigorous bounding methods, significant advantages arise for modest to complicated functional dependencies, especially in higher dimensions. The next chapter discusses an implementation of the method in the code COSY INFINITY [6] [7] [11] [15].

4.1 Introduction

The significant advances in computer hardware that we have experienced particularly in the past decade allow the study of ever more complex problems. In many practical problems, one must locally model nonlinear functional dependencies, for example to study parameter sensitivity or to perform optimization. This is typically done through the computation of derivatives, and the methods of computational differentiation [13] [31], and for the solutions of ODEs and PDEs the Differential Algebraic (DA) enhancements, have excelled in providing such derivatives accurately and inexpensively.

While the derivatives themselves are accurate except for computational errors that are typically very small, rigor is lost when the derivatives are used to model a functional dependence, because of the lack of information about the size of remainder terms. The method of interval arithmetic (see, for example, [48]) often provides a means for keeping the mathematical rigor in the computation of model functions. However, the naive use of interval methods in large problems is prone to blow-up, which at times limits the practical usefulness of such methods.

The new technique, the method of Remainder-enhanced Differential Algebras (RDA), combines the DA approach to obtain Taylor polynomials with interval techniques to determine bounds for the Taylor remainder. The new method uses the advantages and diminishes the disadvantages of either of the two methods.

In beam physics and weakly nonlinear dynamics in general, the DA technique [3] [4] [5] [8] has offered a robust way to study the nonlinear behavior of beams and has evolved into one of the essential tools. However, many difficult yet important questions remained, including the long-term stability of beams in circular accelerators or other dynamical systems, such as the planets in the solar system. In light of modern stability theories, this problem can be cast in the form of an optimization problem [14] [15], which from the computational point of view is very complex.

4.2 Differential Algebra and Interval Arithmetic

While DA methods [6] [7] [11] [15] can provide the derivatives of functional dependencies and solutions of ODEs to high orders, in a rigorous sense they fail to provide information about the range of the function. A simple example that dramatically illustrates this phenomenon is the function shown in Figure 4.1

$$f(x) = \begin{cases} 0 & \text{if } x = 0 \\ \exp(-1/x^2) & \text{else.} \end{cases} \tag{4.1}$$

The value of the function and all the derivatives at $x = 0$ are 0. Thus the Taylor polynomial at the reference point $x = 0$ is just the constant 0. In particular, this also implies that the Taylor expansion of f converges everywhere, but it fails to agree with $f(x)$ everywhere but at $x = 0$.

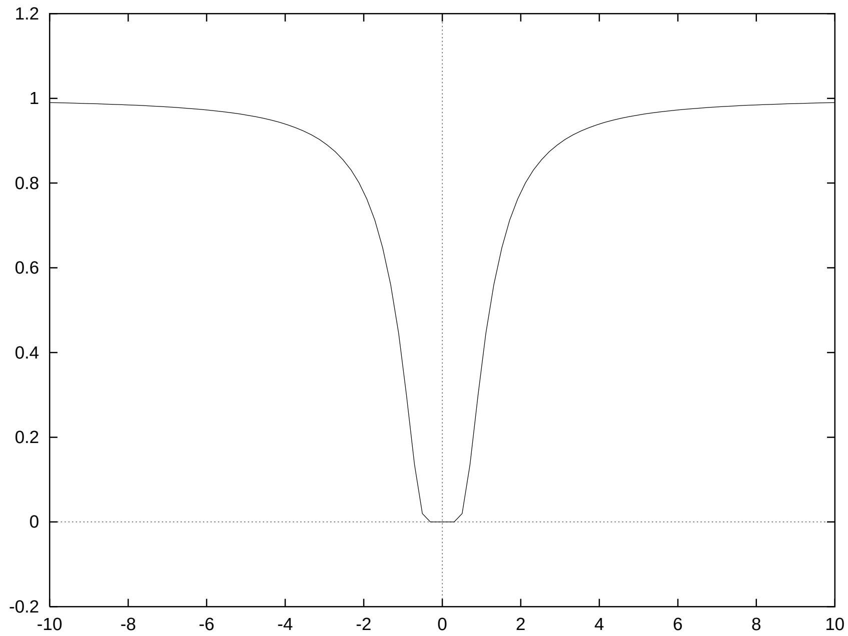

Figure 4.1: Function $f(x) = \exp(-1/x^2)$ if $x \neq 0$; 0 else, and its Taylor polynomial, which vanishes identically.

For the purpose of bounding functional dependencies, the methods of interval arithmetic provides a conceptual contrast. Both extended domains of numbers as well as individual real numbers are represented via rigorous inclusions of floating point intervals. Arithmetic operations are introduced on intervals such that for any numbers in the intervals, a real arithmetic operation on the two numbers always

Table 4.1: Elementary properties of interval arithmetic; $I_1 = [a_1, b_1], I_2 = [a_2, b_2]$.

$I_1 + I_2 = [a_1 + a_2, b_1 + b_2]$
$-I_1 = [-b_1, -a_1]$
$I_1 \cdot I_2 = [\min(a_1a_2, a_1b_2, b_1a_2, b_1b_2), \max(a_1a_2, a_1b_2, b_1a_2, b_1b_2)]$
If $0 \notin I_1$, $1/I_1 = [1/b_1, 1/a_1]$

leads to a result that is contained in the interval obtained from the corresponding arithmetic operation on the intervals. Table 4.1 lists some elementary properties of interval arithmetic.

By evaluating a function in interval arithmetic, it is thus possible to carry rigorous bounds information through the operations, and in the end obtain rigorous bounds of the function. However, while reasonably fast in practice, interval methods have some severe disadvantages, which limits their applicability for complicated functions. First, the width of resulting intervals scales with the width of the original intervals; and second, artificial blow-up usually occurs in extended calculations. Another practical limitation arises if scanning with small intervals is needed in the case of multiple dimensions because of the fast increase of the computational expense.

To illustrate the blow-up phenomenon with a trivial example, we consider the interval $I = [a, b]$, which has the width $b - a$. We compute the addition of I to itself and its subtraction from itself:

$$\begin{aligned} I + I &= [a, b] + [a, b] = [a + a, b + b] = [2a, 2b] \\ I - I &= [a, b] - [a, b] = [a, b] + [-b, -a] = [a - b, b - a]. \end{aligned}$$

In both cases the resulting width is $2(b-a)$, which is twice the original width, although we know that regardless of what unknown quantity x is characterized by I, certainly $x - x$ should equal zero.

Polynomials under their conventional operations form a commutative algebra with unity. However, interval arithmetic does not have even a group structure for either addition or multiplication, since intervals with nonzero width have no inverses. Furthermore, instead of distributivity, we have the sub-distributivity,

$$I_1 \cdot (I_2 + I_3) \subseteq I_1 \cdot I_2 + I_1 \cdot I_3. \tag{4.2}$$

The concepts of interval methods are discussed at length and in depth in several sources, including [48].

4.3 Remainder-enhanced Differential Algebraic Operations

In this section, we develop a new method that combines the advantage of rigor of the interval approach, while largely avoiding the blow-up problem through the use of DA techniques. The key idea is to describe the bulk of the functional dependence through a Taylor polynomial, and bound the deviation of the original function from the Taylor polynomial by an interval. In this endeavor, the Taylor theorem plays an important role.

Theorem (Taylor): *Suppose that a function* $f : [\vec{a}, \vec{b}] \subset R^v \to R$ *is* $(n+1)$ *times continuously partially differentiable on* $[\vec{a}, \vec{b}]$. *Assume* $\vec{x}_0 \in [\vec{a}, \vec{b}]$. *Then for each* $\vec{x} \in [\vec{a}, \vec{b}]$, *there is* $\theta \in R$ *with* $0 < \theta < 1$ *such that*

$$f(\vec{x}) = \sum_{\nu=0}^{n} \frac{1}{\nu!} \left((\vec{x} - \vec{x}_0) \cdot \vec{\nabla} \right)^{\nu} f(\vec{x}_0) + \frac{1}{(n+1)!} \left((\vec{x} - \vec{x}_0) \cdot \vec{\nabla} \right)^{n+1} f\left(\vec{x}_0 + (\vec{x} - \vec{x}_0)\theta\right),$$

where the partial differential operator $\left(\vec{h}\cdot\vec{\nabla}\right)^k$ *operates as*

$$\left(\vec{h}\cdot\vec{\nabla}\right)^k = \sum_{\substack{0\le i_1,\cdots,i_v\le k\\ i_1+\cdots+i_v=k}} \frac{k!}{i_1!\cdots i_v!} h_1^{i_1}\cdots h_v^{i_v} \frac{\partial^k}{\partial x_1^{i_1}\cdots\partial x_v^{i_v}}.$$

Depending on the situation at hand, the remainder term also can be cast into a variety of well-known other forms.

Taylor's theorem allows a quantitative estimate of the error that is to be expected when approximating a function by its Taylor polynomial. Furthermore, it even offers a way to obtain bounds for the error in practice, based on bounding the $(n+1)$st derivative, a method that has sometimes been employed in interval calculations.

Roughly speaking, Taylor's theorem suggests that in many cases the error decreases with the order as the width of the interval raised to the order being considered, and its practical use is often connected to this observation. However, certain examples illustrate that this behavior does not have to occur; one such example is (4.1) in the previous section.

For notational convenience, we introduce a parameter α to describe the details of a given Taylor expansion, namely, the order of the Taylor polynomial n, and the reference point of expansion $\vec{x}_0$. For the purpose to derive bounds for the remainder, it is also necessary to include the domain interval $[\vec{a},\vec{b}]$ on which the function is to be considered; altogether, we have

$$\alpha = (n, \vec{x}_0, [\vec{a},\vec{b}]). \tag{4.3}$$

We now write an $(n+1)$ times continuously partially differentiable function $f : [\vec{a},\vec{b}] \subset R^v \to R$ as a sum of its Taylor polynomial $P_{\alpha,f}$ of nth order and a remainder $\varepsilon_{\alpha,f}$ as

$$f(\vec{x}) = P_{\alpha,f}(\vec{x}-\vec{x}_0) + \varepsilon_{\alpha,f}(\vec{x}-\vec{x}_0),$$

where $\varepsilon_{\alpha,f}(\vec{x}-\vec{x}_0)$ is continuous (even continuously differentiable) on the domain interval and thus bounded. Let the interval $I_{\alpha,f}$ be such that

$$\forall \vec{x} \in [\vec{a}, \vec{b}], \quad \varepsilon_{\alpha,f}(\vec{x}-\vec{x}_0) \in I_{\alpha,f}.$$

Then

$$\forall \vec{x} \in [\vec{a}, \vec{b}], \quad f(\vec{x}) \in P_{\alpha,f}(\vec{x}-\vec{x}_0) + I_{\alpha,f}. \tag{4.4}$$

Because of the special form of the Taylor remainder term $\varepsilon_{\alpha,f}$, in practice the remainder usually decreases as $|\vec{x}-\vec{x}_0|^{n+1}$. Hence, if $|\vec{x}-\vec{x}_0|$ is chosen to be small, the interval $I_{\alpha,f}$, which from now on we refer to as the interval remainder bound, can become so small that even the effect of considerable blow-up is not detrimental. The set $P_{\alpha,f}(\vec{x}-\vec{x}_0)+I_{\alpha,f}$ containing f consists of the Taylor polynomial $P_{\alpha,f}(\vec{x}-\vec{x}_0)$ and the interval remainder bounds $I_{\alpha,f}$. We say a pair $(P_{\alpha,f}, I_{\alpha,f})$ of a Taylor polynomial $P_{\alpha,f}(\vec{x}-\vec{x}_0)$ and an interval remainder bounds $I_{\alpha,f}$ is a Taylor model of f if and only if (4.4) is satisfied. In this case, we denote the Taylor model by

$$T_{\alpha,f} = (P_{\alpha,f}, I_{\alpha,f}).$$

We call n the order of the Taylor model, $\vec{x}_0$ the reference point of the Taylor model, $[\vec{a}, \vec{b}]$ the domain interval of the Taylor model, and α the parameter of the Taylor model.

In the following, we develop tools that allow us to efficiently calculate Taylor models for all functions representable on a computer. The key is to begin with the Taylor model for the identity function, which is trivial, and then successively build up Taylor models for the total function from its pieces. This requires methods to determine Taylor models for sums and products from those of the summands or factors, as well as from intrinsics applied to functions with known Taylor model.

4.3.1 Addition and Multiplication

In this subsection, we discuss how a Taylor model of a sum or product of two functions can be obtained from the Taylor models of the two individual functions. This represents the first step toward the computation of Taylor models for any function that can be represented on a computer.

Let the functions $f, g : [\vec{a}, \vec{b}] \subset R^v \to R$ have Taylor models

$$T_{\alpha,f} = (P_{\alpha,f}, I_{\alpha,f}) \quad \text{and} \quad T_{\alpha,g} = (P_{\alpha,g}, I_{\alpha,g}),$$

which entails that

$$\begin{aligned} \forall \vec{x} \in [\vec{a}, \vec{b}], \quad f(\vec{x}) &\in P_{\alpha,f}(\vec{x} - \vec{x}_0) + I_{\alpha,f} \qquad \text{and} \\ g(\vec{x}) &\in P_{\alpha,g}(\vec{x} - \vec{x}_0) + I_{\alpha,g}. \end{aligned}$$

Then it is straightforward to obtain a Taylor model for $f+g$; in fact, for any $\vec{x} \in [\vec{a}, \vec{b}]$,

$$\begin{aligned} f(\vec{x}) + g(\vec{x}) &\in (P_{\alpha,f}(\vec{x} - \vec{x}_0) + I_{\alpha,f}) + (P_{\alpha,g}(\vec{x} - \vec{x}_0) + I_{\alpha,g}) \\ &= (P_{\alpha,f}(\vec{x} - \vec{x}_0) + P_{\alpha,g}(\vec{x} - \vec{x}_0)) + (I_{\alpha,f} + I_{\alpha,g}), \end{aligned}$$

so that a Taylor model $T_{\alpha,f+g}$ for $f+g$ can be obtained via

$$P_{\alpha,f+g} = P_{\alpha,f} + P_{\alpha,g} \quad \text{and} \quad I_{\alpha,f+g} = I_{\alpha,f} + I_{\alpha,g}. \tag{4.5}$$

Thus we define

$$T_{\alpha,f} + T_{\alpha,g} = (P_{\alpha,f} + P_{\alpha,g}, I_{\alpha,f} + I_{\alpha,g}),$$

and we obtain that $T_{\alpha,f} + T_{\alpha,g} = (P_{\alpha,f+g}, I_{\alpha,f+g})$ is a Taylor model for $f+g$. Note that the above addition of Taylor models is both commutative and associative.

The goal in defining a multiplication of Taylor models is to determine a Taylor model for $f \cdot g$ from the knowledge of the Taylor models $T_{\alpha,f}$ and $T_{\alpha,g}$ for f and g.

Observe that for any $\vec{x} \in [\vec{a}, \vec{b}]$,

$$\begin{aligned} f(\vec{x}) \cdot g(\vec{x}) &\in (P_{\alpha,f}(\vec{x}-\vec{x}_0) + I_{\alpha,f}) \cdot (P_{\alpha,g}(\vec{x}-\vec{x}_0) + I_{\alpha,g}) \\ &\subseteq P_{\alpha,f}(\vec{x}-\vec{x}_0) \cdot P_{\alpha,g}(\vec{x}-\vec{x}_0) \\ &\quad + P_{\alpha,f}(\vec{x}-\vec{x}_0) \cdot I_{\alpha,g} + P_{\alpha,g}(\vec{x}-\vec{x}_0) \cdot I_{\alpha,f} + I_{\alpha,f} \cdot I_{\alpha,g}. \end{aligned}$$

Note that $P_{\alpha,f} \cdot P_{\alpha,g}$ is a polynomial of $(2n)$th order. We split it into the part of up to nth order, which agrees with the Taylor polynomial $P_{\alpha,f\cdot g}$ of order n of $f \cdot g$, and the extra polynomial P_e, so that we have

$$P_{\alpha,f}(\vec{x}-\vec{x}_0) \cdot P_{\alpha,g}(\vec{x}-\vec{x}_0) = P_{\alpha,f\cdot g}(\vec{x}-\vec{x}_0) + P_e(\vec{x}-\vec{x}_0). \tag{4.6}$$

A Taylor model for $f \cdot g$ can now be obtained by finding a bound interval for all the terms except $P_{\alpha,f\cdot g}$. For this purpose, let $B(P)$ be bounds of the polynomial $P : [\vec{a}, \vec{b}] \subset R^v \to R$, namely,

$$\forall \vec{x} \in [\vec{a}, \vec{b}], \quad P(\vec{x}) \in B(P).$$

Apparently the efficient practical determination of $B(P)$ is not completely trivial; depending on the order and number of variables, different strategies may be employed, ranging from analytical estimates to interval evaluations. However, thanks to the specific circumstances, the occurring contributions are very small, and even moderate overestimation is not critical. Various methods for determining $B(P)$ will be discussed below.

Altogether, interval remainder bounds for $f \cdot g$ can be found via

$$I_{\alpha,f\cdot g} = B(P_e) + B(P_{\alpha,f}) \cdot I_{\alpha,g} + B(P_{\alpha,g}) \cdot I_{\alpha,f} + I_{\alpha,f} \cdot I_{\alpha,g}. \tag{4.7}$$

Thus we define $T_{\alpha,f} \cdot T_{\alpha,g} = (P_{\alpha,f\cdot g}, I_{\alpha,f\cdot g})$, and obtain that $T_{\alpha,f} \cdot T_{\alpha,g}$ is a Taylor model for $f \cdot g$. Note that commutativity of multiplication holds, $T_{\alpha,f} \cdot T_{\alpha,g} = T_{\alpha,g} \cdot T_{\alpha,f}$, while multiplication is not generally associative, and also distributivity does not generally hold.

While the idea of Taylor models of constant functions is almost trivial, we mention it for the sake of completeness. For a constant function $f(\vec{x}) \equiv t$, the Taylor model of f is

$$T_{\alpha,f} \equiv T_{\alpha,t} = (P_{\alpha,t}, I_{\alpha,t}) = (t, [0,0]).$$

Having introduced addition and multiplication as well as scalar multiplication, we can compute any polynomial of a Taylor model. Let $Q(f)$ be a polynomial of a function f, that is, $Q(f) = t_0 + t_1 f + t_2 f^2 + \cdots + t_k f^k$. In practice it is useful to evaluate $Q(f)$ via Horner's scheme,

$$Q(f) = t_0 + f \cdot \Big(t_1 + f \cdot \Big(t_2 + f \cdot (\cdots(t_{k-1} + f \cdot t_k)\cdots)\Big)\Big), \tag{4.8}$$

in order to minimize operations. Furthermore, Horner's scheme is often of advantage for interval related arithmetic because of the sub-distributivity (4.2) of interval arithmetic. Assume that we have already found the Taylor model of the function f to be $T_{\alpha,f} = (P_{\alpha,f}, I_{\alpha,f})$. Then, using additions and multiplications of Taylor models described above, we can compute a Taylor model for the function $Q(f)$ via

$$T_{\alpha,Q(f)} = \left(P_{\alpha,Q(f)}, I_{\alpha,Q(f)}\right).$$

4.3.2 Intrinsic Functions

In the preceding subsection, we showed how Taylor models for sums and products of functions can be obtained from those of the individual functions. The computation led to the definition of addition and multiplication of Taylor models. Here we study the computation of Taylor models for intrinsic functions, including the reciprocal applied to a given function f from the Taylor model of f.

The key idea is to employ Taylor's theorem of the function under consideration: However, in order to ensure that the resulting remainder term yields a small remainder interval and does not contribute anything to the Taylor polynomial, several additional steps are necessary.

Let us begin the study with the **exponential** function. Assume that we have already found the Taylor model of the function f to be $T_{\alpha,f} = (P_{\alpha,f}, I_{\alpha,f})$. Write the constant part of the function f around $\vec{x}_0$ as $c_{\alpha,f}$, which agrees with the constant part of the Taylor polynomial $P_{\alpha,f}$, and write the remaining part as $\bar{f}$; that is,

$$f(\vec{x}) = c_{\alpha,f} + \bar{f}(\vec{x}).$$

A Taylor model of $\bar{f}$ is then $T_{\alpha,\bar{f}} = (P_{\alpha,\bar{f}}, I_{\alpha,\bar{f}})$, where

$$P_{\alpha,\bar{f}}(\vec{x} - \vec{x}_0) = P_{\alpha,f}(\vec{x} - \vec{x}_0) - c_{\alpha,f} \qquad \text{and} \qquad I_{\alpha,\bar{f}} = I_{\alpha,f}.$$

Now we can write

$$\begin{aligned} \exp(f(\vec{x})) &= \exp\left(c_{\alpha,f} + \bar{f}(\vec{x})\right) = \exp(c_{\alpha,f}) \cdot \exp\left(\bar{f}(\vec{x})\right) \\ &= \exp(c_{\alpha,f}) \cdot \left\{1 + \bar{f}(\vec{x}) + \frac{1}{2!}(\bar{f}(\vec{x}))^2 + \cdots + \frac{1}{k!}(\bar{f}(\vec{x}))^k \right. \\ &\qquad \left. + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1} \exp\left(\theta \cdot \bar{f}(\vec{x})\right)\right\}, \end{aligned}$$

where $0 < \theta < 1$. Taking $k \geq n$, where n is the order of Taylor model, the part

$$\exp(c_{\alpha,f}) \cdot \left\{1 + \bar{f}(\vec{x}) + \frac{1}{2!}(\bar{f}(\vec{x}))^2 + \cdots + \frac{1}{n!}(\bar{f}(\vec{x}))^n\right\}$$

is a polynomial of $\bar{f}$, of which we can obtain the Taylor model as outlined in the preceding subsection. The remainder part of $\exp(f(\vec{x}))$,

$$\exp(c_{\alpha,f}) \cdot \left\{\frac{1}{(n+1)!}(\bar{f}(\vec{x}))^{n+1} + \cdots + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1} \exp\left(\theta \cdot \bar{f}(\vec{x})\right)\right\}, \tag{4.9}$$

will be bounded by an interval. Since $P_{\alpha,\bar{f}}(\vec{x} - \vec{x}_0)$ does not have a constant part, $(P_{\alpha,\bar{f}}(\vec{x} - \vec{x}_0))^m$ starts from mth order. Thus, in the Taylor model computation, the remainder part (4.9) has vanishing polynomial part. The remainder bound interval for the Lagrange remainder term

$$\exp(c_{\alpha,f})\frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1}\exp\left(\theta\cdot\bar{f}(\vec{x})\right)$$

can be estimated because, for any $\vec{x}\in[\vec{a},\vec{b}]$, $P_{\alpha,\bar{f}}(\vec{x}-\vec{x}_0)\in B(P_{\alpha,\bar{f}})$, and $0<\theta<1$, and so

$$(\bar{f}(\vec{x}))^{k+1}\exp\left(\theta\cdot\bar{f}(\vec{x})\right)\in\left(B(P_{\alpha,\bar{f}})+I_{\alpha,\bar{f}}\right)^{k+1}\exp\left([0,1]\cdot(B(P_{\alpha,\bar{f}})+I_{\alpha,\bar{f}})\right). \quad (4.10)$$

Since the exponential function is monotonically increasing, the estimation of the interval bounds of the part $\exp\left([0,1]\cdot(B(P_{\alpha,\bar{f}})+I_{\alpha,\bar{f}})\right)$ is achieved by inserting the upper and lower bounds of the argument in the exponential.

A Taylor model for the **logarithm** of a function f can be computed in a similar manner from the Taylor model of the function. In this case, there is the limitation that it has to be ensured that the range of the function f lies entirely within the range of definition of the logarithm, which will be the case if

$$\forall\vec{x}\in[\vec{a},\vec{b}],\ P_{\alpha,f}(\vec{x}-\vec{x}_0)+I_{\alpha,f}\subset(0,\infty).$$

For the actual computation, we again split the constant part of the function f around $\vec{x}_0$ from the rest $f(\vec{x})=c_{\alpha,f}+\bar{f}(\vec{x})$. Then we obtain

$$\begin{aligned}\log(f(\vec{x})) &= \log\left(c_{\alpha,f}+\bar{f}(\vec{x})\right)=\log\left\{c_{\alpha,f}\cdot\left(1+\frac{\bar{f}(\vec{x})}{c_{\alpha,f}}\right)\right\}\\ &= \log c_{\alpha,f}+\log\left(1+\frac{\bar{f}(\vec{x})}{c_{\alpha,f}}\right)\\ &= \log c_{\alpha,f}+\frac{\bar{f}(\vec{x})}{c_{\alpha,f}}-\frac{1}{2}\frac{(\bar{f}(\vec{x}))^2}{c_{\alpha,f}^2}+\cdots+(-1)^{k+1}\frac{1}{k}\frac{(\bar{f}(\vec{x}))^k}{c_{\alpha,f}^k}\\ &\quad +(-1)^{k+2}\frac{1}{k+1}\frac{(\bar{f}(\vec{x}))^{k+1}}{c_{\alpha,f}^{k+1}}\frac{1}{\left(1+\theta\cdot\bar{f}(\vec{x})/c_{\alpha,f}\right)^{k+1}},\end{aligned}$$

where $0 < \theta < 1$. Taking $k \geq n$, the part

$$\log c_{\alpha,f} + \frac{\bar{f}(\vec{x})}{c_{\alpha,f}} - \frac{1}{2}\frac{(\bar{f}(\vec{x}))^2}{c_{\alpha,f}^2} + \cdots + (-1)^{n+1}\frac{1}{n}\frac{(\bar{f}(\vec{x}))^n}{c_{\alpha,f}^n}$$

is again treated as a polynomial of $\bar{f}$ in the Taylor model computation. The Lagrange remainder part of $\log(f(\vec{x}))$ becomes part of the remainder bound interval of the Taylor model of $\log(f)$. The remainder term can be estimated as

$$(-1)^{k+2}\frac{1}{k+1}\frac{(B(P_{\alpha,\bar{f}}) + I_{\alpha,\bar{f}})^{k+1}}{c_{\alpha,f}^{k+1}}\frac{1}{\left(1 + [0,1] \cdot (B(P_{\alpha,\bar{f}}) + I_{\alpha,\bar{f}})/c_{\alpha,f}\right)^{k+1}}.$$

In a rather similar fashion, it is possible to determine Taylor models of reciprocals, square roots, trigonometric functions and so on as soon as a Taylor model for the argument is known. In the following, we give short recipes to obtain Taylor models for other intrinsic functions.

Multiplicative inverse: Under the condition

$$\forall \vec{x} \in [\vec{a}, \vec{b}],\ 0 \notin P_{\alpha,f}(\vec{x} - \vec{x}_0) + I_{\alpha,f},$$

$$\begin{aligned}\frac{1}{f(\vec{x})} &= \frac{1}{c_{\alpha,f}} \cdot \left\{1 - \frac{\bar{f}(\vec{x})}{c_{\alpha,f}} + \frac{(\bar{f}(\vec{x}))^2}{c_{\alpha,f}^2} - \cdots + (-1)^k \frac{(\bar{f}(\vec{x}))^k}{c_{\alpha,f}^k}\right\} \\ &\quad + (-1)^{k+1}\frac{(\bar{f}(\vec{x}))^{k+1}}{c_{\alpha,f}^{k+2}}\frac{1}{\left(1 + \theta \cdot \bar{f}(\vec{x})/c_{\alpha,f}\right)^{k+2}}. \qquad (4.11)\end{aligned}$$

Square root: Under the condition

$$\forall \vec{x} \in [\vec{a}, \vec{b}],\ P_{\alpha,f}(\vec{x} - \vec{x}_0) + I_{\alpha,f} \subset (0, \infty),$$

$$\begin{aligned}\sqrt{f(\vec{x})} &= \sqrt{c_{\alpha,f}} \cdot \left\{1 + \frac{1}{2}\frac{\bar{f}(\vec{x})}{c_{\alpha,f}} - \frac{1}{2!2^2}\frac{(\bar{f}(\vec{x}))^2}{c_{\alpha,f}^2} + \cdots + (-1)^{k-1}\frac{(2k-3)!!}{k!2^k}\frac{(\bar{f}(\vec{x}))^k}{c_{\alpha,f}^k}\right\} \\ &\quad + (-1)^k \sqrt{c_{\alpha,f}} \cdot \frac{(2k-1)!!}{(k+1)!2^{k+1}}\frac{(\bar{f}(\vec{x}))^{k+1}}{c_{\alpha,f}^{k+1}}\frac{1}{\left(1 + \theta \cdot \bar{f}(\vec{x})/c_{\alpha,f}\right)^{k+1/2}}.\end{aligned}$$

Multiplicative inverse of square root: Under the condition

$$\forall \vec{x} \in [\vec{a}, \vec{b}],\ P_{\alpha,f}(\vec{x}-\vec{x}_0) + I_{\alpha,f} \subset (0,\infty),$$

$$\begin{aligned}
\frac{1}{\sqrt{f(\vec{x})}} &= \frac{1}{\sqrt{c_{\alpha,f}}} \cdot \left\{ 1 - \frac{1}{2}\frac{\bar{f}(\vec{x})}{c_{\alpha,f}} + \frac{3!!}{2!2^2}\frac{(\bar{f}(\vec{x}))^2}{c_{\alpha,f}^2} + \cdots + (-1)^k \frac{(2k-1)!!}{k!2^k}\frac{(\bar{f}(\vec{x}))^k}{c_{\alpha,f}^k} \right\} \\
&\quad + (-1)^{k+1}\frac{1}{\sqrt{c_{\alpha,f}}} \cdot \frac{(2k+1)!!}{(k+1)!2^{k+1}}\frac{(\bar{f}(\vec{x}))^{k+1}}{c_{\alpha,f}^{k+1}}\frac{1}{\left(1+\theta\cdot\bar{f}(\vec{x})/c_{\alpha,f}\right)^{k+3/2}}.
\end{aligned}$$

Sine:

$$\begin{aligned}
\sin(f(\vec{x})) &= \sin(c_{\alpha,f}) + \cos(c_{\alpha,f})\cdot\bar{f}(\vec{x}) - \frac{1}{2!}\sin(c_{\alpha,f})\cdot(\bar{f}(\vec{x}))^2 \\
&\quad - \frac{1}{3!}\cos(c_{\alpha,f})\cdot(\bar{f}(\vec{x}))^3 + \cdots + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1}\cdot J,
\end{aligned}$$

where

$$J = \begin{cases} -J_0 & \text{if } \bmod(k,4) = 1,2 \\ J_0 & \text{else} \end{cases}$$

$$J_0 = \begin{cases} \cos(c_{\alpha,f}+\theta\cdot\bar{f}(\vec{x})) & \text{if } k \text{ is even} \\ \sin(c_{\alpha,f}+\theta\cdot\bar{f}(\vec{x})) & \text{else.} \end{cases}$$

Cosine:

$$\begin{aligned}
\cos(f(\vec{x})) &= \cos(c_{\alpha,f}) - \sin(c_{\alpha,f})\cdot\bar{f}(\vec{x}) - \frac{1}{2!}\cos(c_{\alpha,f})\cdot(\bar{f}(\vec{x}))^2 \\
&\quad + \frac{1}{3!}\sin(c_{\alpha,f})\cdot(\bar{f}(\vec{x}))^3 + \cdots + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1}\cdot J,
\end{aligned}$$

where

$$J = \begin{cases} -J_0 & \text{if } \bmod(k,4) = 0,1 \\ J_0 & \text{else} \end{cases}$$

$$J_0 = \begin{cases} \sin(c_{\alpha,f}+\theta\cdot\bar{f}(\vec{x})) & \text{if } k \text{ is even} \\ \cos(c_{\alpha,f}+\theta\cdot\bar{f}(\vec{x})) & \text{else.} \end{cases}$$

Hyperbolic sine:

$$\begin{aligned}\sinh(f(\vec{x})) &= \sinh(c_{\alpha,f}) + \cosh(c_{\alpha,f}) \cdot \bar{f}(\vec{x}) + \frac{1}{2!}\sinh(c_{\alpha,f}) \cdot (\bar{f}(\vec{x}))^2 \\ &\quad + \frac{1}{3!}\cosh(c_{\alpha,f}) \cdot (\bar{f}(\vec{x}))^3 + \cdots + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1} \cdot J,\end{aligned}$$

where

$$J = \begin{cases} \cosh(c_{\alpha,f} + \theta \cdot \bar{f}(\vec{x})) & \text{if } k \text{ is even} \\ \sinh(c_{\alpha,f} + \theta \cdot \bar{f}(\vec{x})) & \text{else.} \end{cases}$$

Hyperbolic cosine:

$$\begin{aligned}\cosh(f(\vec{x})) &= \cosh(c_{\alpha,f}) + \sinh(c_{\alpha,f}) \cdot \bar{f}(\vec{x}) + \frac{1}{2!}\cosh(c_{\alpha,f}) \cdot (\bar{f}(\vec{x}))^2 \\ &\quad + \frac{1}{3!}\sinh(c_{\alpha,f}) \cdot (\bar{f}(\vec{x}))^3 + \cdots + \frac{1}{(k+1)!}(\bar{f}(\vec{x}))^{k+1} \cdot J,\end{aligned}$$

where

$$J = \begin{cases} \sinh(c_{\alpha,f} + \theta \cdot \bar{f}(\vec{x})) & \text{if } k \text{ is even} \\ \cosh(c_{\alpha,f} + \theta \cdot \bar{f}(\vec{x})) & \text{else.} \end{cases}$$

Arcsine: Under the condition

$$\forall \vec{x} \in [\vec{a}, \vec{b}],\ P_{\alpha,f}(\vec{x} - \vec{x}_0) + I_{\alpha,f} \subset (-1, 1),$$

using an addition formula for the arcsine, we have

$$\arcsin(f(\vec{x})) = \arcsin(c_{\alpha,f}) + \arcsin\left(f(\vec{x}) \cdot \sqrt{1 - c_{\alpha,f}^2} - c_{\alpha,f} \cdot \sqrt{1 - (f(\vec{x}))^2}\right).$$

Utilizing that

$$g(\vec{x}) \equiv f(\vec{x}) \cdot \sqrt{1 - c_{\alpha,f}^2} - c_{\alpha,f} \cdot \sqrt{1 - (f(\vec{x}))^2}$$

does not have a constant part, we have

$$\begin{aligned}\arcsin(g(\vec{x})) &= g(\vec{x}) + \frac{1}{3!}(g(\vec{x}))^3 + \frac{3^2}{5!}(g(\vec{x}))^5 + \frac{3^2 \cdot 5^2}{7!}(g(\vec{x}))^7 + \cdots \\ &\quad + \frac{1}{(k+1)!}(g(\vec{x}))^{k+1} \cdot \arcsin^{(k+1)}(\theta \cdot g(\vec{x})),\end{aligned}$$

where $\arcsin'(a) = 1/\sqrt{1-a^2}$, $\arcsin''(a) = a/(1-a^2)^{3/2}$, $\arcsin^{(3)}(a) = (1+2a^2)/(1-a^2)^{5/2}$, ... A recursive formula for the higher order derivatives of arcsin

$$\arcsin^{(k+2)}(a) = \frac{1}{1-a^2}\{(2k+1)a\arcsin^{(k+1)}(a) + k^2\arcsin^{(k)}(a)\}$$

is useful [58].

Arccosine: Use $\arccos(f(\vec{x})) = \pi/2 - \arcsin(f(\vec{x}))$.

Arctangent: Using an addition formula for the arctangent, we have

$$\arctan(f(\vec{x})) = \arctan(c_{\alpha,f}) + \arctan\left(\frac{f(\vec{x}) - c_{\alpha,f}}{1 + c_{\alpha,f}\cdot f(\vec{x})}\right).$$

Utilizing that

$$g(\vec{x}) \equiv \frac{f(\vec{x}) - c_{\alpha,f}}{1 + c_{\alpha,f}\cdot f(\vec{x})} = \frac{\bar{f}(\vec{x})}{1 + c_{\alpha,f}\cdot f(\vec{x})}$$

does not have a constant part, we obtain

$$\begin{aligned}\arctan(g(\vec{x})) &= g(\vec{x}) - \frac{1}{3}(g(\vec{x}))^3 + \frac{1}{5}(g(\vec{x}))^5 - \frac{1}{7}(g(\vec{x}))^7 + \cdots + \frac{1}{k+1}(g(\vec{x}))^{k+1} \\ &\quad \cdot\cos^{k+1}\left(\arctan(\theta\cdot g(\vec{x})\right)\cdot\sin\left((k+1)\cdot\left(\arctan(\theta\cdot g(\vec{x})) + \frac{\pi}{2}\right)\right).\end{aligned}$$

Altogether, it is now possible to compute Taylor models along for any function that can be represented in a computer environment along with the mere evaluation of the function by simple operator overloading, in much the same way as the mere computation of derivatives, Taylor polynomials, or interval bounds, along with the mere evaluation of the function.

4.3.3 Derivations and Antiderivations

In the spirit of the idea of embedding the elementary operations of addition, multiplication, and differentiation and their inverses that are defined on the class of C^∞ functions onto the structure of Taylor Models, we now come to the mapping of the derivation operation ∂ as well as its inverse ∂^{-1}. Similar to the case of the DA, and following one of the main thrusts of the history of differential algebra, we will use these for the solution of the initial value problem

$$\frac{d}{dt}\vec{r}(t) = \vec{F}(\vec{r}(t), t),$$

where $\vec{F}$ is continuous and bounded. We are interested in both the case of a specific initial condition $\vec{r}_0$, as well as the case in which the initial condition $\vec{r}_0$ is a variable, in which case our interest is in the flow of the differential equation

$$\vec{r}(t) = \mathcal{M}(\vec{r}_0, t).$$

As in the case of the conventional DA method, in order to prevent loss of order in the differentiation process, the derivation ∂ can be evaluated only in the context of a Lie derivative $L_g = g \cdot \partial$, where $g(\vec{x}_0) = 0$. However, in the case of Taylor models, an additional complication is connected to the fact that from the Taylor model alone, it is impossible to determine bounds for the derivative, since nothing is known about the rate of change of the function $(f - P_{\alpha,f})$ within the remainder bounds I_f. The situation can be remedied by a further extension of the Taylor model concept to contain not only bounds for the remainder, but also a low-parameter bounding sequence for all the higher derivatives that can occur. In contrast to the derivation ∂, Taylor models of its inverse ∂^{-1} are readily available.

Given an n-th order Taylor model $(P_{n,f}, I_{n,f})$ of a function $f : [\vec{a}, \vec{b}] \subset R^v \to R$ around the reference point $\vec{x}_{0,}$ we can determine a Taylor model for the indefinite integral $\partial_i^{-1} f = \int f \; dx_i$ with respect to the variable x_i. The Taylor polynomial part is obviously just given by $\int_0^{x_i} P_{n-1,f}(\vec{x}) dx_i$. Since the part of the Taylor polynomial $P_{n,f}$ that is of precise order n is $P_{n,f} - P_{n-1,f}$, remainder bounds can be obtained as $(B(P_{n,f} - P_{n-1,f}) + I_{n,f}) \cdot B(x_i)$, where $B(x_i)$ is obtained from the range of definition of x_i as $b_i - a_i$. We thus define the operator ∂_i^{-1}on the space of Taylor models as

$$\partial_i^{-1}(P_{n,f}, I_{n,f}) = (P_{n,\partial^{-1}f}, I_{n,\partial^{-1}f})$$

$$= \left(\int_0^{x_i} P_{n-1,f}(\vec{x}) dx_i \,,\ (B(P_{n,f} - P_{n-1,f}) + I_{n,f}) \cdot B(x_i) \right). \tag{4.12}$$

With this definition, bounds for a definite integral over variable x_i from x_{il} to x_{iu} both in $[a_i, b_i]$, the domain of validity of the Taylor model of a function, can be obtained as

$$\int_{x_{il}}^{x_{ij}} f\,(\vec{x}) dx_i \in (P_{n,\partial^{-1}f}(\vec{x}|_{x_i=x_{iu}-x_{i0}}) - P_{n,\partial^{-1}f}(\vec{x}|_{x_i=x_{il}-x_{i0}}), I_{n,\partial^{-1}f}). \tag{4.13}$$

4.4 Examples

RDA have many applications, including global optimization, quadrature, and solution of differential equations. We begin our discussion with the determination of sharp bounds for a simple example function using RDA. The sharpness of the resulting bounds will be compared with the results that can be obtained in other ways. Secondly, we show schematically how the method compares with the interval method to obtain bound enclosures of functions in one and two dimensional cases.

4.4.1 A Simple Function

The function under consideration is

$$f(x) = \frac{1}{x} + x. \tag{4.14}$$

For an actual computation, we set the parameter α of (4.3) to $\alpha = (n, x_0, [a, b]) = (3, 2, [1.9, 2.1])$.

As in the case of DA, the evaluation begins with the representation of the identity function, expressed in terms of a Taylor polynomial expanded at the reference point. This identity function i has the form

$$i(x) = x = x_0 + (x - x_0) = 2 + (x - 2).$$

Since this representation is exact, the remainder bound interval is $[0,0]$. Hence, a Taylor model of the identity function i is

$$T_{\alpha,i} = (x_0 + (x - x_0), [0,0]) = (2 + (x-2), [0,0]).$$

The constant part of i around $x_0 = 2$ is $c_{\alpha,i} = x_0 = 2$, and the nonconstant part of i is $\bar{i}(x) = x - x_0 = x - 2$. The Taylor model of $\bar{i}$ is

$$T_{\alpha,\bar{i}} = ((x - x_0), [0,0]) = ((x-2), [0,0])\,.$$

The computation of the inverse requires the knowledge of bounds of $P_{\alpha,\bar{i}}$, which here is readily obtained: $B(P_{\alpha,\bar{i}}) = B(x - x_0) = [a - x_0, b - x_0] = [-0.1, 0.1]$. We have furthermore $B(P_{\alpha,\bar{i}}) + I_{\alpha,\bar{i}} = [-0.1, 0.1] + [0,0] = [-0.1, 0.1]$. Using (4.6) and (4.7), we have for the Taylor model of $(\bar{i})^2$

$$T_{\alpha,(\bar{i})^2} = \left((x-2)^2, [0,0]\right).$$

The Taylor model of $(\bar{i})^3$ is computed similarly: $T_{\alpha,(\bar{i})^3} = ((x-2)^3, [0,0])$. As can be seen, so far all remainder intervals are of zero size. The first nonzero remainder interval comes from the evaluation of the Taylor remainder term, which is

$$\begin{aligned}\frac{(\bar{i}(x))^4}{c_{\alpha,i}^5}\frac{1}{\left(1 + \theta \cdot \bar{i}(x)/c_{\alpha,i}\right)^5} \quad &\in \quad \frac{(B(P_{\alpha,\bar{i}}) + I_{\alpha,\bar{i}})^4}{x_0^5 \cdot \left(1 + [0,1] \cdot (B(P_{\alpha,\bar{i}}) + I_{\alpha,\bar{i}})/x_0\right)^5} \qquad (4.15)\\ &\subseteq \quad \frac{[0, 0.0001]}{2^5 \cdot ([0.95, 1.05])^5} \subseteq [0, 4.038 \times 10^{-6}].\end{aligned}$$

As expected, this remainder term is "small of order four". According to (4.11), the Taylor model of $1/i$ is then

$$T_{\alpha,\frac{1}{i}} = \left(\frac{1}{2} - \frac{1}{2^2}(x-2) + \frac{1}{2^3}(x-2)^2 - \frac{1}{2^4}(x-2)^3, [0, 4.038 \times 10^{-6}]\right),$$

and the remainder interval is indeed still very sharp. Using (4.5), we obtain as the final Taylor model of $1/i + i$

$$T_{\alpha,\frac{1}{i}+i} = T_{\alpha,\frac{1}{i}} + T_{\alpha,i} = \left(P_{\alpha,\frac{1}{i}+i}, I_{\alpha,\frac{1}{i}+i}\right) \qquad (4.16)$$

Table 4.2: The remainder bound interval $I_{1/x+x}$ for various orders; $x_0 = 2$, $[a, b] = [1.9, 2.1]$.

Order	The remainder bound interval
1	$[\;0\;, 1.4579384 \times 10^{-3}\;]$
2	$[-7.6733603 \times 10^{-5}, 7.6733603 \times 10^{-5}]$
3	$[\;0\;, 4.0386107 \times 10^{-6}\;]$
4	$[-2.1255845 \times 10^{-7}, 2.1255845 \times 10^{-7}]$
5	$[\;0\;, 1.1187287 \times 10^{-8}\;]$
6	$[-5.8880459 \times 10^{-10}, 5.8880459 \times 10^{-10}]$
7	$[\;0\;, 3.0989715 \times 10^{-11}\;]$
8	$[-1.6310376 \times 10^{-12}, 1.6310376 \times 10^{-12}]$
9	$[\;0\;, 8.5844087 \times 10^{-14}\;]$
10	$[-4.5181098 \times 10^{-15}, 4.5181098 \times 10^{-15}]$
11	$[\;0\;, 2.3779525 \times 10^{-16}\;]$
12	$[-1.2515539 \times 10^{-17}, 1.2515539 \times 10^{-17}]$
13	$[\;0\;, 6.5871262 \times 10^{-19}\;]$
14	$[-3.4669085 \times 10^{-20}, 3.4669085 \times 10^{-20}]$
15	$[\;0\;, 1.8246887 \times 10^{-21}\;]$

$$\begin{aligned} &= \left(\left(2+\frac{1}{2}\right)+\left(1-\frac{1}{2^2}\right)(x-2)+\frac{1}{2^3}(x-2)^2-\frac{1}{2^4}(x-2)^3, [0, 4.038 \times 10^{-6}]\right) \\ &= \left(2.5+0.75(x-2)+0.125(x-2)^2-0.0625(x-2)^3, [0, 4.038 \times 10^{-6}]\right). \end{aligned}$$

Since the polynomial $P_{\alpha,\frac{1}{i}+i}$ is monotonically increasing in the domain $[a, b] = [1.9, 2.1]$, the bound interval of the polynomial is

$$B\left(P_{\alpha,\frac{1}{i}+i}\right) = \left[P_{\alpha,\frac{1}{i}+i}(-0.1), P_{\alpha,\frac{1}{i}+i}(0.1)\right] = [2.42631, 2.57618].$$

The width of the bound interval of the Taylor polynomial is 0.14987, and the width of the remainder bound interval is 4.038×10^{-6} in the third-order Taylor model evaluation; thus the remainder part is just a minor addition. The size of the remainder bounds depends strongly on the order and decreases quickly with order. Table 4.2 shows the remainder bound interval for various orders in the Taylor model computation.

Table 4.3: The width of the bound interval of $f(x) = 1/x + x$ by various methods; $x_0 = 2$, $[a, b] = [1.9, 2.1]$.

Method		Width of Bound Interval
Intervals	$n_d = 1$	0.25012531
	$n_d = 10$	0.15993589
	$n_d = 10^2$	0.15088206
	$n_d = 10^3$	0.14997543
	$n_d = 10^4$	0.14988476
	$n_d = 10^5$	0.14987569
	$n_d = 10^6$	0.14987478
	$n_d = 10^7$	0.14987469
	$n_d = 10^8$	0.14987468
Taylor models	1st order	0.15145793
	2nd order	0.15015346
	3rd order	0.14987903
	4th order	0.14987542
	5th order	0.14987469
	6th order	0.14987468
Exact		0.14987468

The Taylor model computation is assessed by noting the bound interval B of the original function (4.14), which is

$$B\left(\frac{1}{x} + x\right) = \left[\frac{1}{a} + a, \frac{1}{b} + b\right] = [2.42631, 2.57619].$$

It is illuminating to compare the sharpness of the bounding of the function with the sharpness that can be obtained from conventional interval methods. Evaluating the function with just one interval yields

$$\frac{1}{[a, b]} + [a, b] = \frac{1}{[1.9, 2.1]} + [1.9, 2.1] \subseteq [2.37619, 2.62631].$$

The width of the bound interval obtained by interval arithmetic is 0.25012, and so this simple example already shows a noticeable blow-up. By dividing the domain interval into many subintervals, the blow-up can be suppressed substantially. However, to achieve the sharpness of the third-order Taylor model, the domain has to be split into about $24,000$ subintervals. Table 4.3 shows a comparison of the widths of bound interval for the exact value, the method of Taylor models, and the divided interval method, where n_d indicates the number of division of the domain interval. Of course, sophisticated interval optimization methods [33] [34] [41] [43] can find sharp bounds for the function using substantially fewer interval evaluations.

Practically more important are optimization problems in several variables, and in this case, the situation becomes more dramatic. We wish first to illustrate the computational effort necessary for an accurate calculation of the result by estimating the required number of floating-point operations. We use a simple example function of six variables such as $f(\vec{x}) = \sum_j (1/x_j + x_j)$ to get a rough idea of the computational expense in the case of functions of many variables. In the one-dimensional case, one interval calculation $1/[a,b] + [a,b]$ requires two additions and two divisions. To compare with the third-order Taylor model computation, we divide the domain into 10^4 subintervals, on which additions and divisions total $\sim 10^5$ floating-point operations. Thus, in the multidimensional case with six independent variables, the number of floating-point operations explodes to $(10^4)^6 \times (\sim 10) = \sim 10^{25}$. Again, sophisticated interval optimization methods will be more favorable than these numbers suggest, but typically there is still a very noticeable growth of complexity.

To estimate the performance of the Taylor model approach, we note that the one-dimensional Taylor model in the third-order computation involves a total of about 35 additions, multiplications, and divisions, as counted in (4.15) and (4.16). As we use more variables, however, the total number of terms in the polynomial grows only modestly. For example, order three in six variables requires only a total of 84 terms. Thus in total, the number of floating-point operations of the third-order Taylor model is $\sim 10^4$. A summary of the number of floating-point operations is given in Table 4.4.

Table 4.4: The total number of FP operations required to bound a simple function like $f(\vec{x}) = \sum_j (1/x_j + x_j)$.

	One Dimensional	Six Dimensional
Interval	~ 10	~ 10
10^4 divided intervals	$\sim 10^5$	$\sim 10^{25}$
3rd order Taylor model	~ 10	$\sim 10^4$

4.4.2 Bound Enclosures of Functions

In this subsection, we use some simple functions in one dimension and two dimensions to show schematically how the RDA method bounds functions in comparison with the interval method. The first function is a one dimensional function

$$f(x) = x(x-1.1)(x+2)(x+2.2)(x+2.5)(x+3) \cdot \sin(1.7x+0.5).$$

Figure 4.2 shows the function and its bound enclosures in the domain $[-0.5, 1.0]$. The interval method is applied to bound the function using smaller domain intervals divided into 25 subintervals and 50 subintervals. The method of Taylor models computes the polynomial part of the function and the remainder bound interval. The pictures show the bands of the enclosures of the function around its polynomial parts by the remainder bounds using 7th order and 8th order Taylor models.

A similar schematical comparison between the interval method and the method of Taylor models is made for a two dimensional function. We worked on a function

$$f(x,y) = \sin(1.7x+0.5) \cdot (y+2) \cdot \sin(1.5y)$$

in the domain $[-1,1]\times[-1,1]$. Figure 4.3 shows the function enclosures by the interval method with the smaller domain intervals divided into 10×10, 20×20, 40×40 and 80×80. Figure 4.4 shows the function enclosures around its polynomial parts by the remainder bounds using 7th, 8th, 9th and 10th order Taylor models. Even in this modest case of only two dimensions, the Taylor model approach requires much less effort to provide a similar level of sharpness; the 1600 subintervals used to include the function are in contrast to only 66 expansion coefficients, plus one remainder bound interval. As dimension is increased, the number of subintervals necessary to provide an accurate modeling increases dramatically, while the number of Taylor coefficients grows much more slowly.

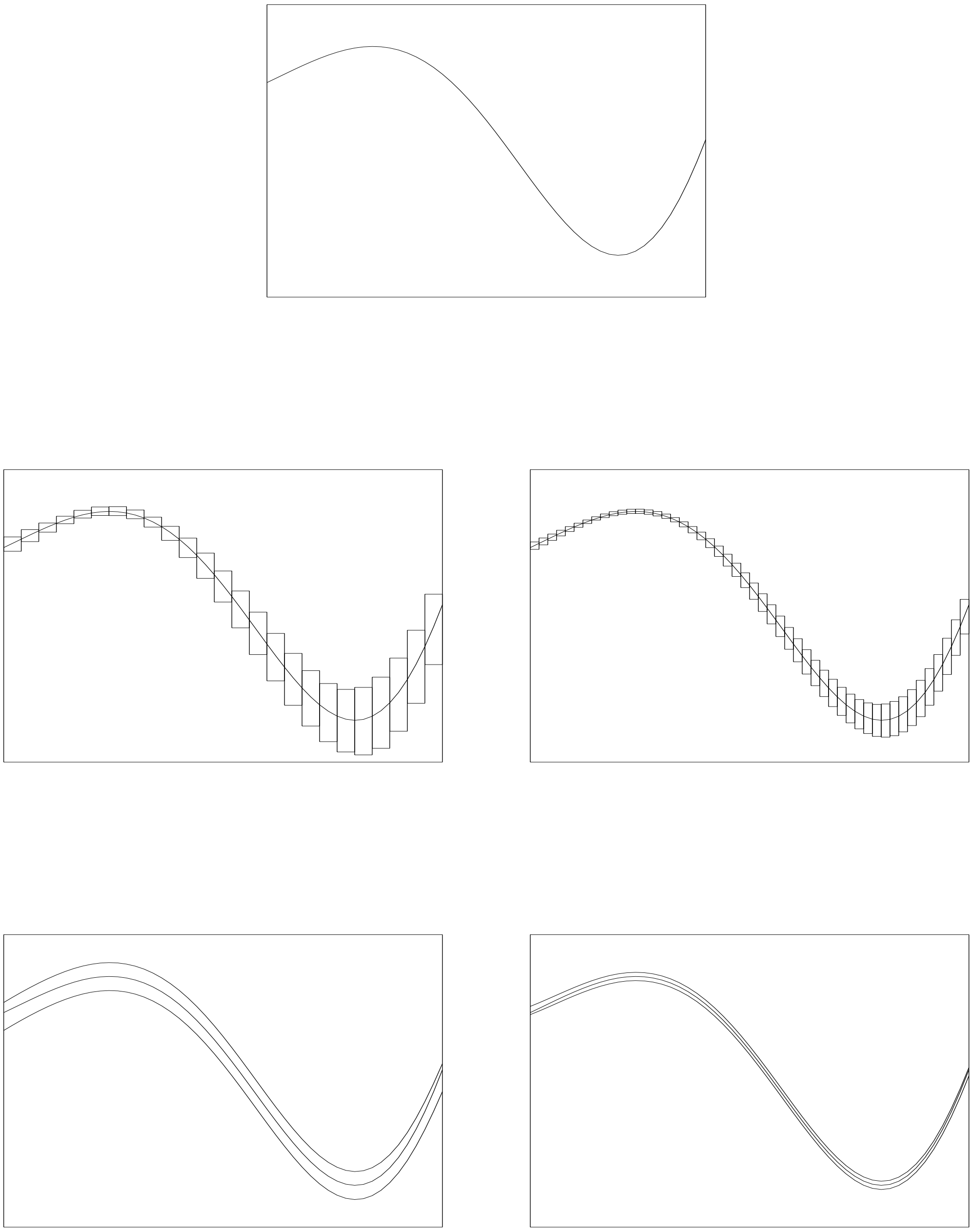

Figure 4.2: One dimensional function and its bound enclosures. From top to bottom right: the function, the bounds by the interval method with the 25 and 50 divided domain intervals, the bounds by the 7th and 8th order Taylor models.

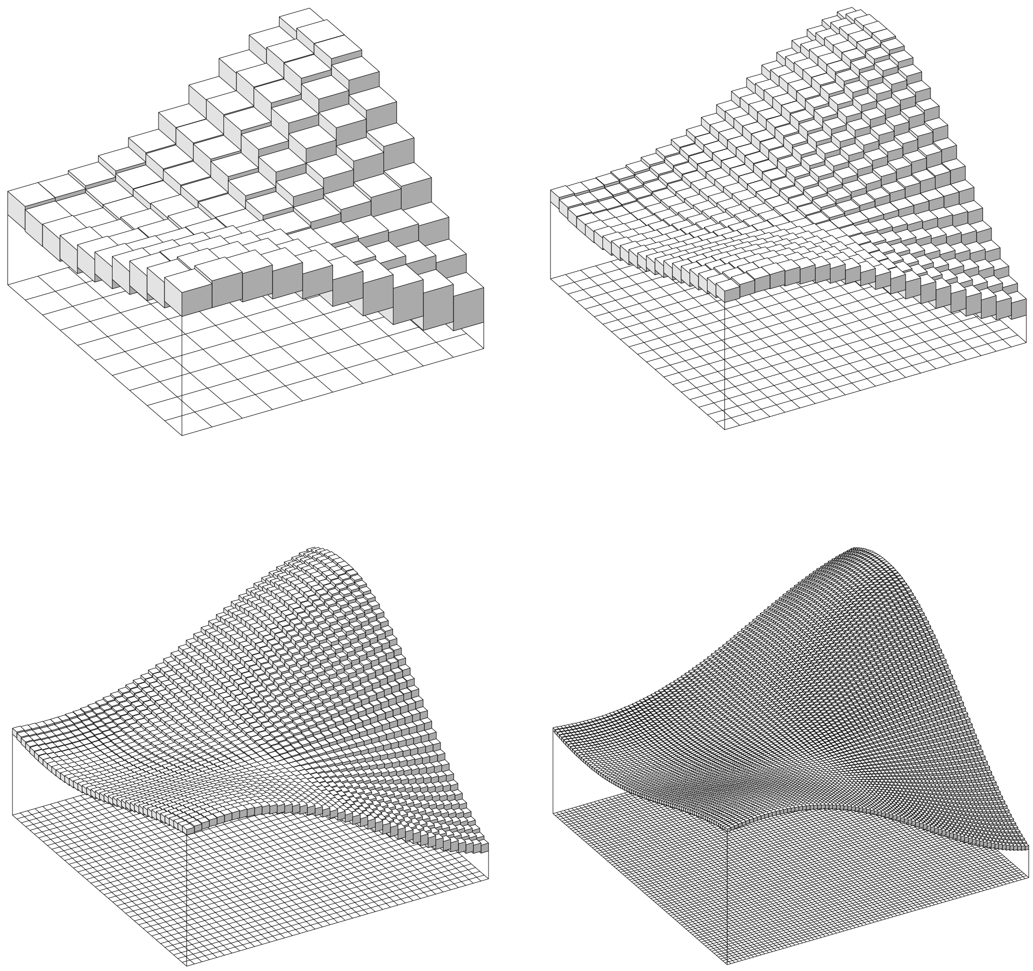

Figure 4.3: Bound enclosures of a two dimensional function by the interval method. From top to bottom right: The domain is divided to 10×10, 20×20, 40×40 and 80×80 subintervals.

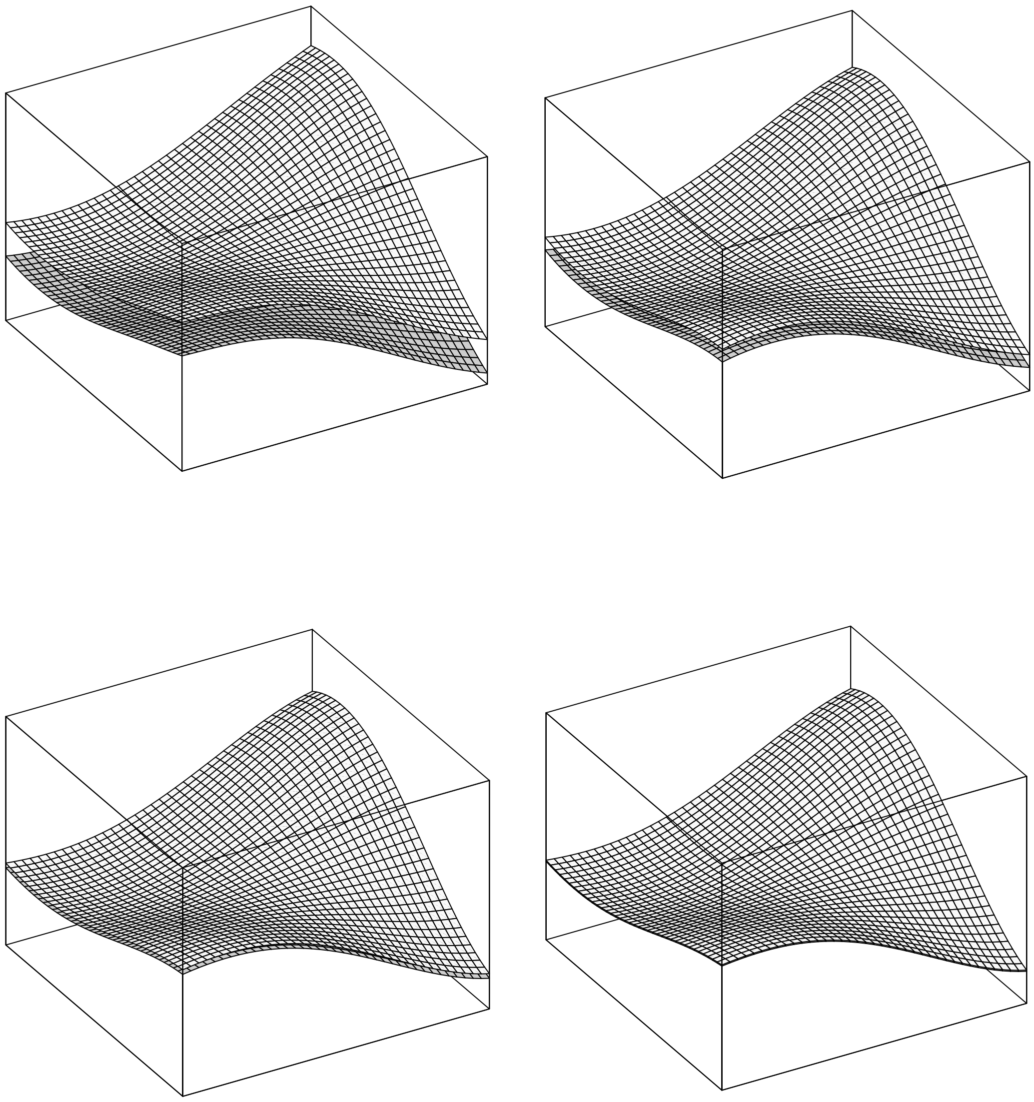

Figure 4.4: Bound enclosures of a two dimensional function by the RDA method. From top to bottom right: Computations in 7th, 8th, 9th and 10th order Taylor models.

Chapter 5

Implementation of Remainder-enhanced Differential Algebraic Operations in COSY INFINITY

Any language environment with object-oriented features or one that supports operator overloading can be utilized to implement the Remainder-enhanced Differential Algebraic (RDA) Method in a conceptually similar way as the conventional Differential Algebraic (DA) method. We pursued an implementation in the code COSY INFINITY [6] [7] [11] [15], which is a major vehicle for studies in beam physics, and even for other numerical studies has the advantage of supporting the dynamic change of data types.

In this chapter, we discuss the data structure of Taylor models and the algorithms on the currently available elements and features in COSY INFINITY. An important aspect to optimize the sharpness of the remaining bounds is the development of efficient polynomial bounding tools, because in many of the algorithms the width of polynomial bounds directly reflects to the width of the remainder bounds of the Taylor model. Several implemented polynomial bounders as well as more ideas in the future are discussed. The last section of the chapter covers some examples and applications of the RDA technique, which show the power of the method as a computational tool.

5.1 Supported Elements and Features

Among various data types available in the current version of COSY INFINITY, the following list in Table 5.1 shows the data types related to a supported set of RDA Operations. In the rest of the chapter, these abbreviations with two characters are used to describe the data types.

Table 5.1: Data types related to RDA in COSY INFINITY.

RD	Remainder-enhanced Differential Algebra Object
RE	8 Byte Real Number
DA	Differential Algebra Vector
IN	8 Byte Interval Number
VE	Real Number Vector
IV	Interval Vector

Binary operations involving an RD data type make mathematical sense only when operations are performed between two objects of RD data type, or between RD data type and RE data type. Because of this, the RDAs in COSY INFINITY allow the following binary operations listed in Table 5.2. As commonly understood, multiplication (*) and division (/) have higher priority than addition (+) and subtraction (-), if there are no parentheses. The resulting data type is always RD data type.

Table 5.2: Binary operations related to RDA in COSY INFINITY.

Operation	Left Type	Right Type	Type of Result
+	RD	RD	RD
+	RD	RE	RD
+	RE	RD	RD
-	RD	RD	RD
-	RD	RE	RD
-	RE	RD	RD
*	RD	RD	RD
*	RD	RE	RD
*	RE	RD	RD
/	RD	RD	RD
/	RD	RE	RD
/	RE	RD	RD

Now follows a list of intrinsic functions available for RD data type as an argument.

Table 5.3: Intrinsic functions available for RDA in COSY INFINITY.

Function	Description	Type of Result
EXP	exponential	RD
LOG	logarithm	RD
SQR	square	RD
SQRT	square root	RD
ISRT	reciprocal of square root	RD
SIN	sine	RD
COS	cosine	RD
TAN	tangent	RD
ASIN	arc sine	RD
ACOS	arc cosine	RD
ATAN	arc tangent	RD
SINH	hyperbolic sine	RD
COSH	hyperbolic cosine	RD
TANH	hyperbolic tangent	RD
CONS	constant part	RE
IN	total bound interval	IN

In practice we need some additional tools to perform computation in RDA, which include procedures to create a starting Taylor model, procedures to extract the polynomial part and the remainder bound interval from a Taylor model, a procedure to determine a Taylor model for the integral of a function described by a Taylor model, and some other service tools. The following is the description of these intrinsic procedures prepared for computation in RDA.

RDAVAR (7 arguments) creates a Taylor model. Arguments are (DA) the polynomial part $P_{n,f}(\vec{x})$, (RE) the order n, (RE) the number of independent variables v, (RE or VE) the reference point $\vec{x}_0$, (IN or IV) the domain $[\vec{a},\vec{b}]$, (IN or IV) the bound intervals for remainder $I^R_{n,f}$ and each order I^1_f, I^2_f,... , and (RD) the resulting Taylor model.

RDVAR (4 arguments) creates the i-th identity Taylor model. Arguments are (RE) the index number i of the variable x_i, (RE or VE) the reference point $\vec{x}_0$, (IN or IV) the domain $[\vec{a}, \vec{b}]$, and (RD) the resulting i-th identity Taylor model.

RDANOT (3 arguments) truncates a Taylor model to a lower order Taylor model. Arguments are (RD) the Taylor model $T_{n,f}$ to be truncated, (RE) the truncation order k, and (RD) the resulting truncated Taylor model $T_{k,f}$. A discussion is found in subsection 5.3.1.

RDREA (2 arguments) reads a Taylor model from a file written in the specified format for Taylor models. Arguments are (RE) the file unit number to read, and (RD) a variable name of the Taylor model.

DAEXT (2 arguments) extracts the polynomial part $P_{n,f}(\vec{x})$ from a Taylor model $T_{n,f}$. Arguments are (RD) the Taylor model $T_{n,f}$, and (DA) the resulting extracted polynomial part $P_{n,f}(\vec{x})$.

RDRBND (2 arguments) extracts the remainder bound interval $I^R_{n,f}$ from the Taylor model $T_{n,f}$. Arguments are (RD) the Taylor model $T_{n,f}$, and (IN) the resulting remainder bound interval $I^R_{n,f}$.

RDITIG (1 argument) sets the algorithm number for the RDA polynomial bounder. Argument is (RE) the algorithm number. A discussion is found in subsection 5.4.1.

RDNPNT (1 argument) sets the total number of points N_{total} for scanning for RDA polynomial bounders with real scanning algorithms. Argument is (RE) the total number of points N_{total}. A discussion is found in subsection 5.4.1.

RDINT (3 arguments) performs an integral of a Taylor model. Arguments are (RE) the index number i of the variable x_i with respect to which to integrate, (RD) the Taylor model $T_{n,f}$ to be integrated, and (RD) the resulting Taylor model $T_{n,\partial^{-1}f}$. A discussion is found in subsection 5.3.4.

5.2 Data Structure of Taylor Models

The data structure of Taylor models consists of two series of array elements. One is a series of double precision real array elements, and it stores all the coefficients of the polynomial $P_{n,f}$, the coordinate values of the reference point $\vec{x}_0$, and the lower and upper bounds of the domain intervals $[\vec{a}, \vec{b}]$, the remainder bound interval $I^R_{n,f}$ and the order bound intervals I^0_f, I^1_f,..., I^n_f. The other one is a series of integer array elements, and it stores all the necessary indices to specify the data and the information on the order n and the number of independent variables v. The addresses of elements in the two arrays correspond to each other. Table 5.4 shows the data structure of a Taylor model in COSY INFINITY schematically.

An order bound interval I^k_f is defined as a bound interval of the part of the polynomial $P_{n,f}$ of exact order k. The 0th order bound interval I^0_f is basically the constant part of the Taylor model, $I^0_f = [c_{n,f}, c_{n,f}]$. The first order bound interval I^1_f also can be obtained straightforward; suppose the linear part of the polynomial $P_{n,f}$ of the Taylor model is $\sum_i c_i \cdot x_i$, then $I^1_f = \sum_i c_i \cdot ([a_i, b_i] - x_{i0})$. However any higher order bound intervals I^2_f,..., I^n_f are not trivial to obtain. This topic is discussed in section 5.4 in detail.

The data which represents a Taylor model carries the following information:

1. the polynomial part $P_{n,f}$
2. the reference point $\vec{x}_0$ and the domain intervals $[\vec{a}, \vec{b}]$
3. the remainder bound interval $I^R_{n,f}$ and the order bound intervals I^0_f, I^1_f,..., I^n_f
4. other information, including the order n and the number of independent variables v, etc.

Table 5.4: Data structure of a Taylor model (RD data type) in COSY INFINITY.

Double precision array	Integer array	
c_{M_1}	I_{M_1}	Polynomial coefficients
c_{M_2}	I_{M_2}	in DA notation
c_{M_3}	I_{M_3}	
$\vdots$	$\vdots$	
$c_{M_{n_{DA}}}$	$I_{M_{n_{DA}}}$	
x_{10}	-1	Reference point $\vec{x}_0$
x_{30}	-3	
a_2	-2	Domain $[\vec{a}, \vec{b}]$
b_2	0	
a_4	-4	
b_4	0	
a^0	0	Order bounds I^0
b^0	0	
a^1	1	Order bounds I^1
b^1	0	
$\vdots$	$\vdots$	$\vdots$
a^n	n	Order bounds I^n
b^n	0	
a^R	$n+1$	Remainder bounds I_n^R
b^R	v	
$0.0D0$	n_{DA}	

For the efficiency and compatibility, we adopt the pre-existing data structure of related data types in COSY INFINITY. Also we try to save memory space to store the data. For this purpose, we treat the above information as follows.

1. The data structure of the polynomial part $P_{n,f}$ agrees to that of the DA data type.

2. Only nonzero elements of reference point $\vec{x}_0$ and only nonzero interval elements of domain intervals $[\vec{a}, \vec{b}]$ are stored. The corresponding index information is stored in a series of integer array elements.

3. All the order bound intervals I_f^0, I_f^1,..., I_f^n and the remainder bound interval $I_{n,f}^R$ are stored. The corresponding order number is stored in a series of integer array elements.

4. The information on the order n can be read from the above data. The number of independent variables v and n_{DA}, the length of series of necessary array elements for polynomial part $P_{n,f}$ as a DA, are stored.

5.3 Operations on Taylor Models

In this section, we discuss the algorithms to perform various arithmetic operations with Taylor models. We begin the task with addition and subtraction, then proceed to the multiplication including the square. The division by a Taylor model is treated as an intrinsic function like the exponential and so on, as discussed in the preceding chapter. We also provide the algorithm to perform an integral of a Taylor model.

Since COSY INFINITY supports all the necessary tools to perform all the possible arithmetic in DA, any arithmetic of polynomials up to the order n are performed within the framework of DA. We adopt all the available tools and algorithms of DA in COSY INFINITY to perform arithmetic of the polynomial part of Taylor models. In the following subsections, we concentrate on the order bound intervals and the remainder bound intervals of Taylor models.

5.3.1 Addition and Subtraction, etc.

As discussed in subsection 4.3.1, a Taylor model as a sum of two Taylor models is obtained in a straightforward way. While the addition of the polynomial part is done in the framework of DA, our task here is to perform the addition of the order bound intervals and the remainder bound intervals. We restrict the operations between two Taylor models to the case in which all the conditions like the reference point $\vec{x}_0$ and the domain $[\vec{a}, \vec{b}]$, agree between these two Taylor models. Suppose we have two Taylor models, an mth order Taylor model $T_{m,f} = (P_{m,f}, I^R_{m,f})$ with its order bound intervals I^0_f, I^1_f,..., I^m_f and an lth order Taylor model $T_{l,g} = (P_{l,g}, I^R_{l,g})$ with its order bound intervals I^0_g, I^1_g,..., I^l_g. In case the orders of two Taylor models do not agree, the higher one has to be truncated to the lower order. We take the target order n as the minimum of m and l.

Let us start the algorithm to truncate a Taylor model of order m, $T_{m,f}$=($P_{m,f}$,$I^R_{m,f}$) with its order bound intervals $I^0_f, I^1_f, ..., I^m_f$, to the lower order n. After truncating the polynomial $P_{m,f}$ to the lower order polynomial $P_{n,f}$, all the order bound intervals up to nth order are kept as they were, and all the order bound intervals with the order higher than n are added to the remainder bound interval;

$$I^R_{n,f} \subseteq I^R_{m,f} + \sum_{k=n+1}^{m} I_{k,f}.$$

After truncating the order of two Taylor models to $n = \min(m, l)$, we have two Taylor models $T_{n,f} = (P_{n,f}, I^R_{n,f})$ with order bound intervals I^0_f, I^1_f,..., I^n_f and $T_{n,g} = (P_{n,g}, I^R_{n,g})$ with order bound intervals I^0_g, I^1_g,..., I^n_g. The addition of the polynomial part and the remainder bound interval part is obtained as described in subsection 4.3.1. The order bound intervals of the resulting Taylor model as the sum are obtained as

$$I^k_{f+g} \subseteq I^k_f + I^k_g.$$

The subtraction of two Taylor models is performed in a similar way.

Corresponding to the list of all the binary operations in section 5.1, let us also discuss how to treat a RE data type number compatible to a RD data type. Suppose we have a RE data type number with value t. Then we can have a Taylor model to represent that number as $T_{n,t} = (P_{n,t}, I^R_{n,t}) = (t, [0,0])$ with its order bound intervals $I^0_t = [t,t]$, $I^1_t = [0,0]$,..., $I^n_t = [0,0]$.

5.3.2 Multiplication

The question of how to obtain a Taylor model for the product of two functions that are themselves described by Taylor models needs more consideration. Again, we concentrate on the order bound intervals and the remainder bound intervals. As above, we restrict the operations between two Taylor models to the case where all the conditions like the reference point $\vec{x}_0$ and the domain $[\vec{a}, \vec{b}]$, agree between these two Taylor models. Suppose we have two Taylor models, an mth order Taylor model and an lth order Taylor model. First we make the orders of two Taylor models in agreement to $n = \min(m, l)$ by truncating the higher order one if necessary. Now we have two Taylor models $T_{n,f} = (P_{n,f}, I^R_{n,f})$ with its order bound intervals I^0_f, I^1_f,..., I^n_f and $T_{n,g} = (P_{n,g}, I^R_{n,g})$ with its order bound intervals I^0_g, I^1_g,..., I^n_g.

The multiplication of the polynomial part is performed in the framework of DA. To obtain the remainder bound interval of the product, we have to have $B(P_e)$, $B(P_{n,f})$ and $B(P_{n,g})$, as described in subsection 4.3.1, bounds of polynomials P_e, $P_{n,f}$ and $P_{n,f}$, where P_e is the extra polynomial of $P_{n,f} \cdot P_{n,g}$ with order higher than n. Since the width of those polynomial bounds directly influences the width of the remainder bounds of the resulting Taylor model as seen in (4.7), it is useful to keep the polynomial bounds as tight as possible. This aspect will be revisited in the next section. Here, we have to note that carrying the order bound intervals as a part of RD data type can help to simplify the task to have Taylor model multiplication significantly, because in practice the detailed information on the extra higher order polynomial P_e is lost in the process of a series of arithmetic. Considering the importance of the role of multiplication in Taylor model arithmetic, to carry the order bound intervals is nearly unavoidable.

The order bound intervals of the resulting Taylor model of the product are obtained as

$$I^k_{f \cdot g} \subseteq \sum_{j=0}^{k} I^j_f \cdot I^{k-j}_g .$$

The remainder bound interval of the resulting Taylor model of the product is obtained as

$$I^R_{n,f\cdot g} \subseteq \sum_{\substack{j,\,k\,=\,0 \\ j\,+\,k\,>\,n}}^{n} I^j_f \cdot I^k_g + \left(\sum_{j=0}^{n} I^j_f \right) \cdot I^R_{n,g} + \left(\sum_{j=0}^{n} I^j_g \right) \cdot I^R_{n,f} + I^R_{n,f} \cdot I^R_{n,g}. \tag{5.1}$$

By denoting the remainder bound intervals $I^R_{n,f}$ and $I^R_{n,g}$ by I^{n+1}_f and I^{n+1}_g, we can simplify the above expression (5.1) as

$$I^R_{n,f\cdot g} \subseteq \sum_{\substack{j,\,k\,=\,0 \\ j\,+\,k\,>\,n}}^{n+1} I^j_f \cdot I^k_g . \tag{5.2}$$

Practically, considering the semi-distributibity of interval arithmetic (4.2), we use the following estimate

$$I^R_{n,f\cdot g} \subseteq \left(\sum_{i=0}^{n+1} I^i_f \cdot \left(\sum_{\substack{j\,=\,0 \\ i\,+\,j\,>\,n}}^{n+1} I^j_g \right) \right) \cap \left(\sum_{i=0}^{n+1} I^i_g \cdot \left(\sum_{\substack{j\,=\,0 \\ i\,+\,j\,>\,n}}^{n+1} I^j_f \right) \right) . \tag{5.3}$$

The square of a Taylor model $T_{n,f} = (P_{n,f}, I^R_{n,f})$ with order bound intervals I^0_f, I^1_f,..., I^n_f can be computed as follows. Again, we denote the remainder bound interval $I^R_{n,f}$ by I^{n+1}_f for the notational convenience. The order bound intervals of the resulting squared Taylor model T_{n,f^2} is

$$I^k_{f^2} \subseteq 2 \cdot \left(\sum_{\substack{i,j=0 \\ i+j=k,\, i<j}}^{k} I^i_f \cdot I^j_f \right) + \begin{cases} \left(I^{k/2}_f\right)^2 & \text{if } k \text{ is even} \\ 0 & \text{else} \end{cases} ,$$

and the resulting remainder bound interval is

$$I^R_{n,f^2} \subseteq 2 \cdot \left(\sum_{i=0}^{n+1} I^i_f \cdot \left(\sum_{\substack{j=0 \\ i+j>n}}^{i-1} I^j_f \right) \right) + \sum_{\substack{i=0 \\ 2i>n}}^{n+1} \left(I^i_f\right)^2 .$$

Here, we can take advantage of explicit expressions for the square in interval arithmetic; for an interval $I = [a, b]$

$$I^2 = \begin{cases} [\ 0\ ,\ \max(a^2, b^2)\ ] & \text{if } ab \leq 0 \\ [\ \min(a^2, b^2)\ ,\ \max(a^2, b^2)\ ] & \text{else} \end{cases} . \tag{5.4}$$

5.3.3 Intrinsic Functions

The actual computation of intrinsic functions of Taylor models is performed in two steps. Let us take the exponential function as an example. Firstly, we prepare a Taylor model of $\bar{f}(\vec{x}) \equiv f(\vec{x}) - c_{n,f}$ from the Taylor model of $f(\vec{x})$. This can be done trivially according to subsection 5.3.1, and thus we have a Taylor model of $\bar{f}(\vec{x})$, $T_{n,\bar{f}} = (P_{n,\bar{f}}, I^R_{n,\bar{f}})$ with order bound intervals $I^0_{\bar{f}}$, $I^1_{\bar{f}}$,..., $I^n_{\bar{f}}$. As described in subsection 4.3.2, the exponential of a function $f(\vec{x})$ can be expressed as

$$\begin{aligned} \exp(f(\vec{x})) &= \exp(c_{n,f}) \cdot \left\{ 1 + \bar{f}(\vec{x}) + \frac{1}{2!}(\bar{f}(\vec{x}))^2 + \cdots + \frac{1}{n!}(\bar{f}(\vec{x}))^n \right\} \\ &\quad + \exp(c_{n,f}) \cdot \frac{1}{(n+1)!}(\bar{f}(\vec{x}))^{n+1} \exp\left(\theta \cdot \bar{f}(\vec{x})\right) . \end{aligned}$$

Then, we compute the polynomial of $\bar{f}(\vec{x})$

$$\exp(c_{n,f}) \cdot \left\{ 1 + \bar{f}(\vec{x}) + \frac{1}{2!}(\bar{f}(\vec{x}))^2 + \cdots + \frac{1}{n!}(\bar{f}(\vec{x}))^n \right\} \tag{5.5}$$

according to the recipe for addition and multiplication of Taylor models. Practically, we use Horner's scheme (4.8) to make the computation efficient. The polynomial part of the resulting Taylor model, $P_{n,\exp f}$, is computed in this process, and we denote the remainder bound interval of the resulting Taylor model of the polynomial (5.5) by $I^R_{n,poly}$.

As the second process, we compute the remainder bound interval of the resulting final Taylor model, $I^R_{n,\exp f}$, using (4.10), as

$$I^R_{n,\exp f} \subseteq I^R_{n,poly} + \exp(c_{n,f}) \cdot \frac{1}{(n+1)!} \left(B(P_{n,\bar{f}}) + I^R_{n,\bar{f}} \right)^{n+1} \exp\left([0,1] \cdot (B(P_{n,\bar{f}}) + I^R_{n,\bar{f}}) \right),$$

where $B(P_{n,\bar{f}})$ is readily obtained as $\sum_{i=0}^n I^i_{\bar{f}}$. The other intrinsic functions of a Taylor model including the multiplicative reciprocal are estimated in a similar way.

5.3.4 Integral

Assume we have a Taylor model of a function $f(\vec{x})$, $T_{n,f} = (P_{n,f}, I^R_{n,f})$ with order bound intervals I^0_f, I^1_f,..., I^n_f. To obtain a Taylor model for the indefinite integral $\partial_i^{-1} f = \int f \, dx_i$ with respect to variable x_i on the Taylor model $T_{n,f,}$ we just follow the description in subsection 4.3.3. The polynomial part of the resulting Taylor model $P_{n,\partial^{-1}f}$ is obtained in the framework of DA. The remainder bound interval of the resulting Taylor model $I^R_{n,\partial^{-1}f}$ is estimated as

$$I^R_{n,\partial^{-1}f} \subseteq (B(P_{n,f} - P_{n-1,f}) + I^R_{n,f}) \cdot B(x_i) = (I^n_f + I^R_{n,f}) \cdot (b_i - a_i).$$

In the process of performing the indefinite integral of polynomial part, the information on the order bound intervals are lost. To reconstruct the order bound intervals of the resulting Taylor model, $I^0_{\partial^{-1}f}$, $I^1_{\partial^{-1}f}$,..., $I^n_{\partial^{-1}f}$, we use a method to estimate order bound intervals discussed in the next section.

5.4 Methods to Tighten Order Bound Intervals

The preceding section discussed the algorithms to compute Taylor models by combining DA techniques and interval arithmetic. While the polynomial part of the Taylor model computation is performed precisely using DA, the width of the remainder bound intervals depends on the method to estimate the bounds. For example, we saw that the estimate of the remainder bound interval of a Taylor model as a product of two Taylor models discussed in subsection 5.3.2 could be done in three ways, (5.1), (5.2) and (5.3), and the last way gives the narrowest width for the resulting remainder bound interval $I^R_{n,f\cdot g}$. However, still a naive application may not result in optimally sharp bounds, and it is useful to study the question of polynomial bounding. In this section, we discuss several already implemented and working algorithms to bound higher order multidimensional polynomials as well as some future ideas to narrow the bounds further.

5.4.1 Implemented Methods

Currently, we have five ways to tighten order bound intervals of a Taylor model, beside an option to do nothing. The number of algorithm is switched freely using the intrinsic service procedure **RDITIG** in section 5.1. These algorithms are

0: No further tightening is done.

1: Default method. Each monomial of the polynomial part $P_{n,f}$ is estimated using interval arithmetic including an efficient estimate of interval powers, and summed to each order bound interval.

2: For each polynomial of the exact order as a part of the polynomial $P_{n,f}$, utilize the existing multidimensional polynomial evaluator in interval arithmetic.

3: Same as 2, but using Horner's scheme in the multidimensional polynomial evaluator.

4: Each monomial of the polynomial part $P_{n,f}$ is estimated using a real number scanning technique. Obviously this method is not rigorous and thus used for test purposes only. It furthermore yields widths that are narrower than the sharpest true bounds.

5: Same as 4, but for the purpose to compensate the underestimation, a correction factor is imposed.

The default algorithm 1 is used in all the Taylor model computations except for this section, and it is showing the practical strength of the RDA technique. The method to evaluate the powers of intervals is the same with the squaring method (5.4). It is most efficient when the expansion point $\vec{x}_0$ is positioned at the center of the domain $[\vec{a}, \vec{b}]$, since $[\vec{a} - \vec{x}_0, \vec{b} - \vec{x}_0]$ is used as an argument for interval arithmetic.

The existing evaluator of multidimensional polynomials in COSY INFINITY, the intrinsic procedure POLVAL, accepts all arithmetic data types as arguments. The current interval evaluation in POLVAL does not use optimally efficient way of computing powers of intervals, and this presently limits the practical strength of the method of the algorithms 2 and 3. The method would be more useful in the future when another kind of order bounds information is needed, for example $I^{\leq k}$, bounds of polynomial of the order k.

The interval based algorithms 1, 2 and 3 can give a mathematically rigorous estimate, but the blow-up problem, a crucial problem of interval arithmetic, again limits the effort to tighten the bounds of a polynomial. In practice this is often not so significant because the remainder is by nature much smaller than the original function. In addition, there is another mechanism to get an estimate on bounds of polynomials by a scanning in real numbers. Scanning can give a good feeling on the behaviour of a function, however the estimate is not rigorous. And it is even dangerous when the function has many local minima and maxima in the domain and when we cannot have enough sample points to scan especially in high dimensional cases. Knowing this mathematical limit, we can use this approach to check the efficiency of Taylor model arithmetic, and it can supply a good measure of the size of order bound intervals.

For the real number scanning algorithms 4 and 5, the whole domain $[\vec{a}, \vec{b}]$ is covered by the total of N_{scan} equidistant points. Let m be the number of points in each dimension. Then the total number of points N_{scan} is calculated as m^v, where v is the number of dimensions. The number m is determined to be the maximum integer which satisfies the condition $m^v \leq N_{total}$, where N_{total} can be specified using an intrinsic service procedure **RDNPNT** in section 5.1. The default value of N_{total} is 1000. To increase the performance on computers regarding CPU time, we utilize an array of VE data type in COSY INFINITY. The ith array element of the VE array is filled with N_{scan} coordinate values of the ith independent variable x_i, which are the m^{v-1} times iteration of m equidistant values in $[a_i, b_i]$ including a_i and b_i, and as a total, the VE array with v elements can represent the whole equidistant points N_{scan} in the whole domain. We use a correction factor $1 + 1/m$ in the algorithm 5. After getting the estimate of order bound intervals using the VE array described above, shift the centers of the bound intervals to 0, then multiply with the correction factor, and shift back to the original centers.

5.4.2 Examples of Performance

In this subsection, some examples are presented to compare the performance of the algorithms in the previous subsection. The first example is a three dimensional example function treated in 5th order Taylor model arithmetic. The function

$$
\begin{aligned}
f(x_1,x_2,x_3) \quad = \quad & \frac{4\tan(3x_2)}{3x_1 + x_1\sqrt{\dfrac{6x_1}{-7(x_1-8)}}} - 120 - 2x_1 - 7x_3(1+2x_2) \\
& - \sinh\left(0.5 + \frac{6x_2}{8x_2+7}\right) + \frac{(3x_2+13)^2}{3x_3} \\
& -20x_3(2x_3-5) + \frac{5x_1\tanh(0.9x_3)}{\sqrt{5x_2}} - 20x_2\sin(3x_3) \qquad (5.6)
\end{aligned}
$$

is computed around the reference point $\vec{x}_0 = (2,1,1)$ in the domain $[a_1,b_1]\times[a_2,b_2]\times[a_3,b_3] = [1.9,2.1]\times[0.9,1.1]\times[0.9,1.1]$. The following is the resulting order bound and remainder bound intervals, and the resulting total bound intervals for each tightening algorithm.

0: No tightening.

Order	Bound interval
0	[-.3928616701165386 ,-.3928616701165386]
1	[-32.85040954846235 , 32.85040954846235]
2	[-2.611562619634850 , 2.641562619634851]
3	[-.2545082014874667 ,0.2545082014874666]
4	[-.2634629627970872E-01,0.2634629627970871E-01]
5	[-.2739252867613695E-02,0.2739252867613695E-02]
Remainder	[-.3160819763199970E-03,0.5077650001011438E-03]
Total	[-36.13874367082485 , 35.38321201361556]

1: Default.

Order	Bound interval
0	[-.3928616701165386 ,-.3928616701165386]
1	[-4.036832719935209 , 4.036832719935209]
2	[-.1517612694112098 ,0.7550980336507371]
3	[-.2288204979419630 ,0.2288204979419625]
4	[-.1303457714927084E-01,0.2049495829307956E-01]
5	[-.9783281805866122E-03,0.9783281805866111E-03]
Remainder	[-.1772503727079576E-03,0.3571133959329409E-03]
Total	[-4.824466313107486 , 4.649719981280969]

2, 3: Using the polynomial evaluator POLVAL.

Order	Bound interval
0	[-.3928616701165386 ,-.3928616701165386]
1	[-4.036832719935209 , 4.036832719935209]
2	[-.7553220623800865 ,0.7553220623800874]
3	[-.2288204979419630 ,0.2288204979419629]
4	[-.2088368942326483E-01,0.2088368942326483E-01]
5	[-.9783281805866133E-03,0.9783281805866130E-03]
Remainder	[-.1932338291952928E-03,0.3848962108177413E-03]
Total	[-5.435892201806844 , 4.650360523955390]

4: Real number scanning.

Order	Bound interval
0	[-.3928616701165386 ,-.3928616701165386]
1	[-4.036832719935209 , 4.036832719935209]
2	[-.7674533702027161E-03,0.7352759591802416]
3	[-.2275570744837629 ,0.2275570744837623]
4	[-.5020266876434484E-02,0.2004305598998429E-01]
5	[-.9628726613536728E-03,0.9628726613536709E-03]
Remainder	[-.1770676431478265E-03,0.3568468875725592E-03]
Total	[-4.664179125086649 , 4.628166859021585]

5: Real number scanning with a correction factor.

Order	Bound interval
0	[-.3928616701165386 ,-.3928616701165386]
1	[-4.440515991928730 , 4.440515991928730]
2	[-.3756962399772495E-01,0.7720781298077638]
3	[-.2503127819321392 ,0.2503127819321386]
4	[-.6273433019755423E-02,0.2129622213330522E-01]
5	[-.1059159927489040E-02,0.1059159927489038E-02]
Remainder	[-.2095690927896201E-03,0.3882500914646420E-03]
Total	[-5.128802230015166 , 5.092788865704353]

The size of the remainder bound intervals are in the same range in all the algorithms even without any tightening (0). However, the total bounds estimate without tightening shows a big blow up. The function evaluation in interval arithmetic gives the resulting bounds

```
[-31.84061451917199     , 33.73372787735954     ],
```

so the necessity of any tightening mechanism is clear. Other than that, all the tightening algorithms are working more or less well. The default algorithm 1 is working well compared to the uncorrected scanning algorithm 4. To see the performance of the total bounds, the function evaluation in real number scanning using the same scanning condition with the algorithm 4, gives a good idea, which is

```
[-4.129818215892230     , 4.397287227019497     ].
```

The estimate of the total bounds in Taylor model arithmetic is comparable to the bounds estimate of the function in real number scanning.

The second example shows the order bounds estimates of a multidimensional polynomial with randomly created coefficients. The polynomial is 5th order in four dimensions, and about half of its coefficients are nonzero and range from -1 to 1;

$$\begin{aligned} P_5(x_1,x_2,x_3,x_4) &= 0.4007583772763610x_1 + 0.9651407347992063x_2 \\ &\quad -0.05361263593658805x_4 - 0.9843004139838740x_1x_2 \\ &\quad +0.2131298496387899x_3x_4 + 0.9396666898392141x_4^2 \\ &\quad -0.2948265776503831x_1^3 + \cdots \\ &\quad -0.03892374457791448x_3^2x_4^3 - 0.9986851064022630x_3x_4^4. \end{aligned}$$

The estimate on order bound intervals is made around the reference point $\vec{x}_0 = \vec{0}$ in the domain $[\vec{a}, \vec{b}] = [-\overrightarrow{0.1}, \overrightarrow{0.1}]$. The following are the resulting order bounds and total bounds for each tightening algorithm.

1: Default.

Order	Bound interval
1	[-.1419511748012155 ,0.1419511748012155]
2	[-.1197430263622664E-01,0.2137096953461878E-01]
3	[-.7059430076740683E-02,0.7059430076740683E-02]
4	[-.8311970126233067E-03,0.7847530025173913E-03]
5	[-.1136391405879840E-03,0.1136391405879840E-03]
Total	[-.1619297436673942 ,0.1712799665556804]

2: Using the polynomial evaluator POLVAL.

Order	Bound interval
1	[-.1419511748012155 ,0.1419511748012155]
2	[-.2137096953461878E-01,0.2137096953461878E-01]
3	[-.7059430076740683E-02,0.7059430076740683E-02]
4	[-.8483902763924564E-03,0.8483902763924564E-03]
5	[-.1136391405879840E-03,0.1136391405879840E-03]
Total	[-.1713436038295554 ,0.1713436038295554]

3: Using the polynomial evaluator POLVAL with Horner's scheme.

Order	Bound interval
1	[-.1419511748012155 ,0.1419511748012155]
2	[-.2137096953461878E-01,0.2137096953461878E-01]
3	[-.7059430076740683E-02,0.7059430076740683E-02]
4	[-.8483902763924563E-03,0.8483902763924563E-03]
5	[-.1136391405879840E-03,0.1136391405879840E-03]
Total	[-.1713436038295554 ,0.1713436038295554]

4: Real number scanning.

Order	Bound interval
1	[-.1419511748012155 ,0.1419511748012155]
2	[-.9843004139838742E-02,0.2137096953461878E-01]
3	[-.3913808809127659E-02,0.3913808809127659E-02]
4	[-.2922135995118879E-03,0.4383278226363474E-03]
5	[-.3743960746774976E-04,0.3743960746774976E-04]
Total	[-.1560376409571615 ,0.1677117205750661]

5: Real number scanning with a correction factor.

Order	Bound interval
1	[-.1703414097614586 ,0.1703414097614586]
2	[-.1296440150728449E-01,0.2449236690206454E-01]
3	[-.4696570570953191E-02,0.4696570570953191E-02]
4	[-.3652677417267115E-03,0.5113819648511710E-03]
5	[-.4492752896129971E-04,0.4492752896129971E-04]
Total	[-.1884125771103843 ,0.2000866567282888]

Again, the default algorithm 1 is working well and yields a result that is comparable to the uncorrected scanning algorithm 4.

5.4.3 Further Tightening Methods

As discussed above, to maximize the sharpness of the remainder bounds it is useful to employ efficient polynomial bounders, and in this subsection, some algorithms for this purpose are presented. First we observe that if d is the width of the domain interval, then very roughly we expect the widths of the I^k to scale like

$$\mathrm{width}(I^k) \approx d^k.$$

Hence the dominating part of the total bounds of a polynomial comes from the linear part and the successive lower order parts. The linear part is easily bounded, and so it is useful to derive methods to bound the second and third order parts. Let us begin the discussion with a second order bounder, which can give an exact bounds for $I^{\leq 2}$.

At first, let us bound a one dimensional function

$$F(x) = c_{11}x^2 + 2c_1 x + c_0$$

in the domain $[a, b]$; while it is trivial, it is the basics of the whole effort to bound a multidimensional quadratic function. If $c_{11} = 0$, the function is linear, so

$$B(F) = [\min(F(a), F(b)), \max(F(a), F(b))]\,. \tag{5.7}$$

If $c_{11} \neq 0$, by writing the function as

$$F(x) = c_{11}\left(x + \frac{c_1}{c_{11}}\right)^2 - \frac{c_1^2}{c_{11}} + c_0,$$

if $-c_1/c_{11} \in [a, b]$,

$$B(F) = \left[\min\left(F(a), F(b), -\frac{c_1^2}{c_{11}} + c_0\right), \max\left(F(a), F(b), -\frac{c_1^2}{c_{11}} + c_0\right)\right],$$

and if not, $B(F)$ is found by (5.7).

As the next step, let us bound a two dimensional function

$$G(x_1, x_2) = c_{11}x_1^2 + 2c_{12}x_1x_2 + c_{22}x_2^2 + 2c_1x_1 + 2c_2x_2 + c_0$$

in the domain $[a_1, b_1] \times [a_2, b_2]$. If $c_{11} = c_{12} = c_{22} = 0$,

$$\begin{aligned} B(G) \;&=\; [\min(G(a_1, a_2), G(a_1, b_2), G(b_1, a_2), G(b_1, b_2)), \\ &\quad\;\; \max(G(a_1, a_2), G(a_1, b_2), G(b_1, a_2), G(b_1, b_2))]. \end{aligned}$$

If $c_{11} \neq 0$ or $c_{12} \neq 0$ or $c_{22} \neq 0$, we have to search the maximum and the minimum at the boundaries $x_1 = a_1,\ x_1 = b_1,\ x_2 = a_2$, and $x_2 = b_2$, as well as the stationary points inside the domain. The bounds at each boundary is nothing other than the bounds of the one dimensional quadratic function at the boundary; for example, at the boundary $x_1 = a_1$, we estimate the bounds of $G(a_1, x_2)$ in the domain $[a_2, b_2]$. The stationary points, if they exist, satisfy

$$\vec{\nabla} G = \begin{pmatrix} 2c_{11}x_1 + 2c_{12}x_2 + 2c_1 \\ 2c_{12}x_1 + 2c_{22}x_2 + 2c_2 \end{pmatrix} = 2\hat{J}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + 2\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = 0, \tag{5.8}$$

where

$$\hat{J} = \begin{pmatrix} c_{11} & c_{12} \\ c_{12} & c_{22} \end{pmatrix}, \text{ and } |\hat{J}| = c_{11}c_{22} - c_{12}^2.$$

If $|\hat{J}| \neq 0$, there is only one stationary point

$$\begin{pmatrix} x_{1s} \\ x_{2s} \end{pmatrix} = -\hat{J}^{-1}\begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = -\frac{1}{|\hat{J}|}\begin{pmatrix} c_{22}c_1 - c_{12}c_2 \\ c_{11}c_2 - c_{12}c_1 \end{pmatrix}.$$

If $|\hat{J}| = 0$, then $c_{11}c_{22} = c_{12}^2$, thus $c_{11} \neq 0$ or $c_{22} \neq 0$ must hold for the function to be of second order. The equations of stationary points (5.8) have solutions if

$$c_{22}c_1 - c_{12}c_2 = 0 \text{ and } c_{11}c_2 - c_{12}c_1 = 0.$$

Now, if $c_{11} = 0$, then it follows that $c_{22} \neq 0$, $c_{12} = 0$ and $c_1 = 0$, so

$$G(x_1, x_2) = c_{22}x_2^2 + 2c_2x_2 + c_0 = c_{22}\left(x_2 + \frac{c_2}{c_{22}}\right)^2 - \frac{c_2^2}{c_{22}} + c_0;$$

hence the stationary points lay on $x_{2s} = -c_2/c_{22}$. Similarly, if $c_{22} = 0$, the stationary points lay on $x_{1s} = -c_1/c_{11}$. If $c_{11} \neq 0$ and $c_{22} \neq 0$, the two equations in (5.8) are identical, and the stationary points lay on them, namely

$$x_{2s} = -\frac{c_{12}x_{1s} + c_2}{c_{22}}, \text{ and } G(x_{1s}, x_{2s}) = -\frac{c_1^2}{c_{11}} + c_0 = -\frac{c_2^2}{c_{22}} + c_0.$$

These stationary points are inside the domain if

$$[a_2, b_2] \cap \left[\min\left(-\frac{c_{12}a_1 + c_2}{c_{22}}, -\frac{c_{12}b_1 + c_2}{c_{22}}\right), \max\left(-\frac{c_{12}a_1 + c_2}{c_{22}}, -\frac{c_{12}b_1 + c_2}{c_{22}}\right)\right] \neq \emptyset.$$

Using induction, the bounding of quadratic functions in more than two variables can be reduced to the above two variable case. Consider a v dimensional function

$$\begin{aligned} H(x_1, x_2, \cdots, x_v) &= c_{11}x_1^2 + 2c_{12}x_1x_2 + \cdots + 2c_{1v}x_1x_v + c_{22}x_2^2 + \cdots + c_{vv}x_v^2 \\ &\quad +2c_1x_1 + \cdots + 2c_vx_v + c_0 \end{aligned}$$

in the domain $[a_1, b_1] \times \cdots \times [a_v, b_v]$. If $c_{11} = \cdots = c_{vv} = 0$, the bounds of H is determined by the maximum and the minimum of the set

$$\{H(x_{1b}, \cdots, x_{vb}) | x_{1b} \in \{a_1, b_1\}, \cdots, x_{vb} \in \{a_v, b_v\}\},$$

which has 2^v elements. If at least one of c_{ij} is nonzero, the maximum and the minimum are on the $2v$ boundaries, $x_1 = a_1$, $x_1 = b_1, \cdots$, $x_v = a_v$, $x_v = b_v$, or at the stationary points inside the domain. To find the extrema on each boundary reduces to the bounding of quadratic functions in $v-1$ dimensions.

On the other hand, the inner stationary points satisfy

$$\vec{\nabla}H = 2\hat{J}\begin{pmatrix} x_1 \\ \vdots \\ x_v \end{pmatrix} + 2\begin{pmatrix} c_1 \\ \vdots \\ c_v \end{pmatrix} = 0, \text{ where } \hat{J} = \begin{pmatrix} c_{11} & \cdots & c_{1v} \\ \vdots & \ddots & \vdots \\ c_{1v} & \cdots & c_{vv} \end{pmatrix},$$

so the stationary points $\vec{x}_s$ are found as follows. If $|\hat{J}| \neq 0$, then $x_{is} = -|\hat{A}_i|/|\hat{J}|$ where $\hat{A}_i$ is obtained by replacing the ith column of $\hat{J}$ by $(c_1, \cdots, c_v)$. If it happens that $|\hat{J}| = 0$, the task to find $\vec{x}_s$ is quite complicated for $v \geq 3$. and usually not possible in closed form.

The implementation of the multidimensional quadratic function bounder requires some careful consideration because of the increase of the effort with dimension. Assume we have an explicit k-dimensional bounder. Then the generation of a $k+1$ dimensional bounder requires $2k$ applications of the k dimensional bounder for the boundaries in addition to the study of internal stationary points. To obtain a v dimensional bounder from a k dimensional one thus requires $v-k$ internal searches, and $2k \cdot 2(k+1) \cdot ... \cdot 2v$ applications of the k-dimensional bounder.

Table 5.5 shows the number of applications of a two, three or four dimensional bounder at the boundaries to bound a higher dimensional quadratic function. The implementation of the three dimensional bounder looks most useful.

Table 5.5: Number of applications of the lower dimensional quadratic function bounders to bound a higher dimensional quadratic function at the boundaries.

Target dimension v	by two dimensional bounder	by three dimensional bounder	by four dimensional bounder
3	6	1	
4	48	8	1
5	480	80	10
6	5,760	960	120
7	80,640	13,440	1,680
8	1,290,240	215,040	26,880
Formula	$2^{v-3}v!$	$2^{v-4}v!/3$	$2^{v-7}v!/3$

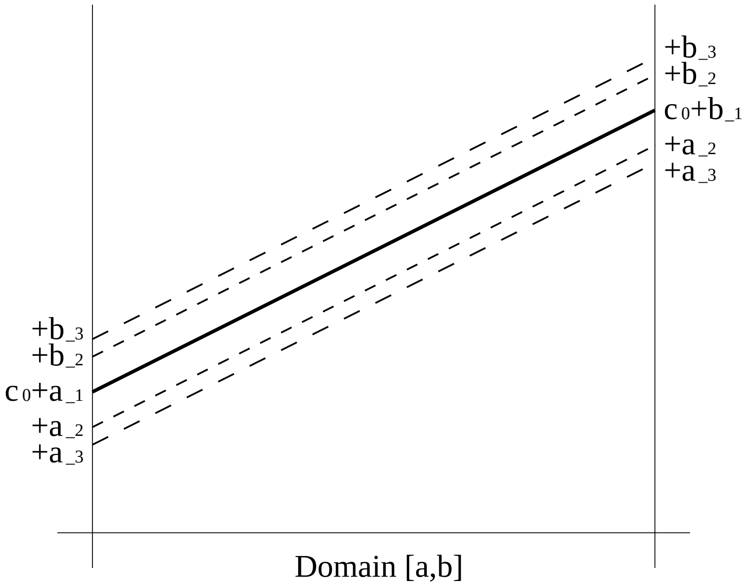

Figure 5.1: The order bounds $I^2 = [a_{\_2}, b_{\_2}]$ and $I^3 = [a_{\_3}, b_{\_3}]$ around the linear line $c_0 + c_{\_1}x \in [c_0 + a_{\_1}, c_0 + b_{\_1}]$.

We can pursue the effort to tighten a polynomial bounds further. Another method is to refine bounds of a third order polynomial $I^{\leq 3}$ iteratively using the order bounds I^0, I^1, I^2, and I^3. The method uses the fact that the linear part dominates the behavior of the function. Let us bound a one dimensional third order polynomial

$$P(x) = c_0 + c_{\_1}x + c_{\_2}x^2 + c_{\_3}x^3 \tag{5.9}$$

in the domain $[a, b]$. Assume that we have the order bounds

$$I^0 = [c_0, c_0],\ I^1 = [a_{\_1}, b_{\_1}],\ I^2 = [a_{\_2}, b_{\_2}] \text{ and } I^3 = [a_{\_3}, b_{\_3}].$$

From these, we can estimate $I^{\leq 3}$ simply

$$I^{\leq 3} \subseteq [c_0 + a_{\_1} + a_{\_2} + a_{\_3},\ c_0 + b_{\_1} + b_{\_2} + b_{\_3}],$$

but a careful look at Figure 5.1, which shows how the order bounds I^2 and I^3 add the width of the band around the linear line $c_0 + c_{\_1}x \in [c_0 + a_{\_1}, c_0 + b_{\_1}]$, gives

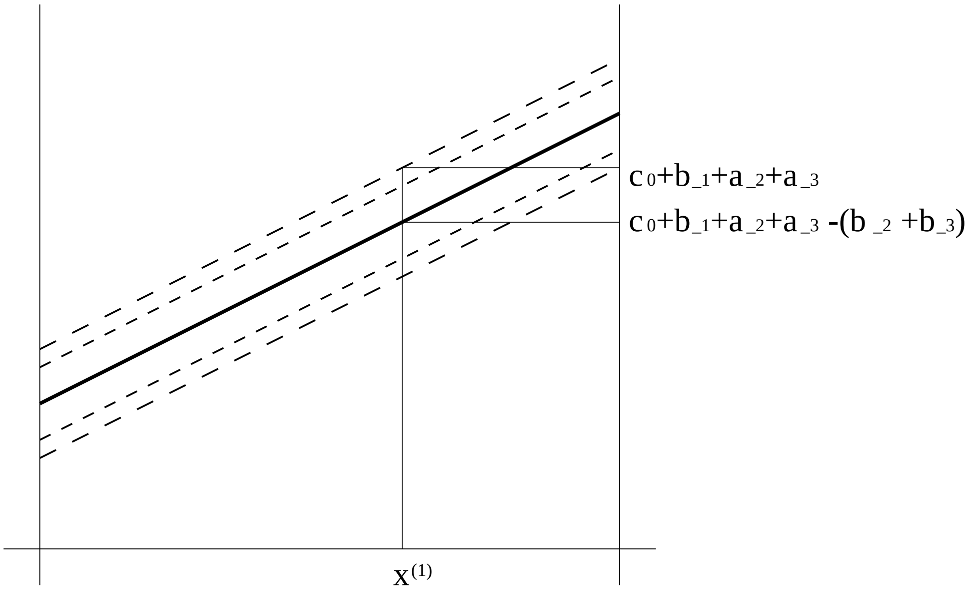

Figure 5.2: An upper bound of $I^{\leq 3}$ is above $c_0 + b_{\_1} + a_{\_2} + a_{\_3}$. Find a point $x^{(1)}$ to specify a new domain.

an insight to tighten $I^{\leq 3}$. Namely, an upper bound of $I^{\leq 3}$ certainly must lie above $c_0 + b_{\_1} + a_{\_2} + a_{\_3}$. Now, find a point $x^{(1)}$ in the domain such that

$$c_0 + c_{\_1} x^{(1)} = c_0 + b_{\_1} + a_{\_2} + a_{\_3} - (b_{\_2} + b_{\_3}). \tag{5.10}$$

For the sake of simplicity of the argument, assume $c_{\_1} > 0$ in the following. The new domain $[x^{(1)}, b]$ contains the point which gives the maximum of the polynomial (5.9). Now, compute the order bounds $I^2_{(1)} = [a^{(1)}_{\_2}, b^{(1)}_{\_2}]$ and $I^3_{(1)} = [a^{(1)}_{\_3}, b^{(1)}_{\_3}]$ in the new domain. Since the new domain is smaller than the original domain, the new order bounds $I^2_{(1)}$ and $I^3_{(1)}$ are narrower than the original order bounds I^2 and I^3 respectively. Thus, new bounds $I^{\leq 3}_{(1)}$ computed from $I^2_{(1)}$ and $I^3_{(1)}$, has a smaller upper bound $c_0 + b_{\_1} + b^{(1)}_{\_2} + b^{(1)}_{\_3}$. This procedure can be applied iteratively until the desired sharpness of the upper bound of $I^{\leq 3}_{(k)}$ is achieved. The lower bound of $I^{\leq 3}$ is refined in the same way.

The key in multidimensional case is how to define a new domain. Pick up the boundary corner point A which gives the upper bound of I^1, search the points which satisfy a condition similar to (5.10) at each neighbouring boundary of A, and these points and the point A define a new domain.

As an example, let us bound $P(x) = x - x^2 - x^3$ in the domain $[-0.1, 0.1]$. The order bounds are $I^0 = [0, 0]$, $I^1 = [-0.1, 0.1]$, $I^2 = [-0.01, 0]$ and $I^3 = [-0.001, 0.001]$. From these, $I^{\leq 3} \subseteq [-0.111, 0.101]$. Find $x^{(1)}$ from the condition (5.10) to be 0.088, so the first new domain is $[0.088, 0.1]$. In this new domain, compute the order bounds $I^2_{(1)} = [-0.01, -7.744 \times 10^{-3}]$ and $I^3_{(1)} = [-0.001, -6.81472 \times 10^{-4}]$ to get $I^{\leq 3}_{(1)} \subseteq [0.077, 0.091574528]$. Finding $x^{(2)}$ gives the second new domain $[0.0974, 0.1]$. The new estimate of the order bounds $I^2_{(2)}$ and $I^3_{(2)}$ gives $I^{\leq 3}_{(2)}$. Altogether, after two iterations, the upper bound of $I^{\leq 3}$ is found to be 0.089589229, which is much smaller than 0.101.

The methods discussed in this subsection will help to get exact bounds of a second order polynomial $I^{\leq 2}$ and sharper bounds of third and also fourth order polynomials $I^{\leq 3}$ and $I^{\leq 4}$, and it will tighten the total bounds of a polynomial $B(P)$ significantly. However, as discussed in subsection 5.3.2, the carriage of the order intervals I^0, I^1, $I^2, \cdots, I^n$ as a part of Taylor model data on computers is indispensable, unless we have a direct mechanism to compute the remainder bound interval of a product of two Taylor models. One also should try a path in this direction, where the problem is cast into the following form; find bounds of a function

$$F(\vec{x}, s, t) = P_{e,f\cdot g}(\vec{x} - \vec{x}_0) + P_{n,f}(\vec{x} - \vec{x}_0) \cdot t + P_{n,g}(\vec{x} - \vec{x}_0) \cdot s + s \cdot t$$

$\forall \vec{x} \in [\vec{a}, \vec{b}]$, $\forall s \in I^R_{n,f}$ and $\forall t \in I^R_{n,g}$.

5.5 Examples of Computation

In this section, we present some examples and applications to show its practical power. First, we make an inspection of the method using the multidimensional function (5.6) on page 119. Second, we apply the method to compute multidimensional definite integrals. The last example in this section bounds a normal form deviation function discussed in chapter 1, which is a six dimensional polynomial of roughly 200th order, and hence has many local extrema.

5.5.1 A Small Multidimensional Function

A three dimensional function (5.6) on page 119 is computed in Taylor model arithmetic around the same reference point as in subsection 5.4.2, $\vec{x}_0 = (2, 1, 1)$. The domain is chosen centered at $\vec{x}_0$ such that the width is 0.1 in all the dimensions, namely $[a_1, b_1] = [1.95, 2.05]$, $[a_2, b_2] = [0.95, 1.05]$, $[a_3, b_3] = [0.95, 1.05]$. The bounds estimate of the function by a scanning in real numbers with the same scanning condition with subsection 5.4.2 is

```
[ -2.31165715,  1.78168226].
```

When the function is evaluated with the domain intervals in interval arithmetic, it gives the bounds

```
[-16.36393303, 16.09747985],
```

which is almost ten times wider than the bounds evaluated by the scanning. Table 5.6 is the summary of the remainder bound intervals of the function via Taylor model computation in various orders. The table also lists the maximum number of terms of polynomial in each order.

While the number of terms of polynomial increases moderately in order, the width of remainder bound interval drops down as expected from the study in the previous chapter 4.

Table 5.6: Remainder bound intervals of a small multidimensional function for various orders.

Order	terms	Remainder bound interval
1	4	[-.3914054034075695 ,0.7252479186770013]
2	10	[-.3395018823172723E-01,0.3394051630619647E-01]
3	20	[-.1020280049382976E-02,0.1609662094279157E-02]
4	35	[-.8413202994873543E-04,0.8402878906072845E-04]
5	56	[-.2410738165297321E-05,0.4383393685666119E-05]
6	84	[-.3355536894608123E-06,0.3343170624752108E-06]
7	120	[-.1631941909229007E-07,0.2051811636562382E-07]
8	165	[-.2424624748948692E-08,0.2410781342028141E-08]
9	220	[-.1721943529722954E-09,0.1736796027250248E-09]
10	286	[-.2313819358227192E-10,0.2298654603065415E-10]
11	364	[-.1928098381883883E-11,0.1821047609084028E-11]
12	455	[-.2424312972351917E-12,0.2407755156049230E-12]
13	560	[-.2163485755166328E-13,0.2012630346355465E-13]
14	680	[-.2614793046930413E-14,0.2596691498167008E-14]
15	816	[-.2417223411345871E-15,0.2242821131576220E-15]

5.5.2 Multidimensional Integrals

We can use antiderivations of Taylor models to compute bounds for multidimensional definite integrals that cannot be treated analytically. The interval method allows us to get a verified estimate of a definite integral. Figure 5.3 shows schematically how to obtain a definite integral of a one dimensional function f in the domain $[a, b]$. Interval arithmetic can give a bounds estimate of the function in the domain as $[f_L, f_U]$, so the bounds of the definite integral is estimated as $[f_L, f_U] \cdot B(x) = [f_L, f_U] \cdot (b - a)$. On the other hand, Taylor model arithmetic would give a Taylor model of the the function (P_f, I_f^R) in the domain, from which we can get a Taylor model for the indefinite integral $(P_{\partial^{-1}f}, I_{\partial^{-1}f}^R)$ as discussed in subsection 4.3.3. And the bounds of the definite integral can be computed using (4.13) on page 93 as $P_{\partial^{-1}f}(b) - P_{\partial^{-1}f}(a) + I_{\partial^{-1}f}^R$.

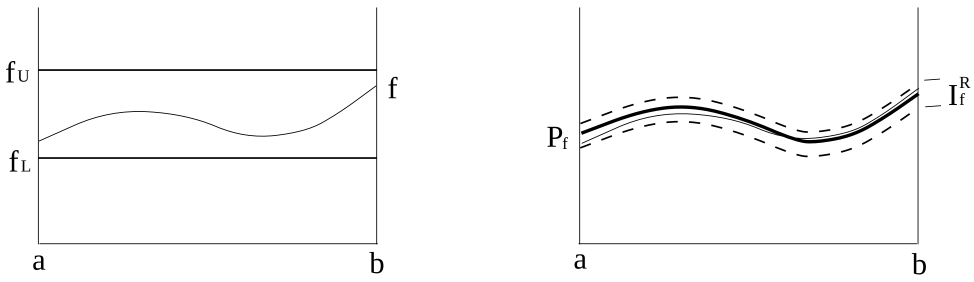

Figure 5.3: Bounds estimates of a definite integral of a function f by the interval method (left) and the RDA method (right).

In the following example calculation based on the RDA method, we study the following analytical double definite integral, which is found in section 4.621 of Gradshteyn and Ryzhik [30].

$$\int_0^{\frac{\pi}{2}}\int_0^{\frac{\pi}{2}} \frac{\sin y\sqrt{1 - k^2 \sin^2 x \sin^2 y}}{1 - k^2 \sin^2 y} \, dx \; dy = \frac{\pi}{2\sqrt{1 - k^2}}. \tag{5.11}$$

The definite integral for $k^2 = 0.1$ is 1.655764710966017.

The interval method gives an estimate of the bounds covering the domain with one interval box $[0, \pi/2] \times [0, \pi/2]$ as

```
[ 0.1433315723E-15,  2.741556778    ],
```

which is too wide to be useful. To increase the sharpness of the bounds, we divided the whole domain into many smaller interval boxes and we obtained the following result shown in 5.7, where the case of 100×100 smaller domain interval boxes gives a reasonable estimate.

Table 5.7: Bounds estimates of a two dimensional integral with the interval method.

Interval Method	
domain interval boxes	bounds estimate
1	[0.1433315723E-15, 2.741556778]
10×10	[1.511886201 , 1.793726593]
100×100	[1.641637834 , 1.669832424]

The bounds estimates with Taylor model arithmetic are much sharper, even without any division of the domain. Table 5.8 shows a clear superiority of the Taylor model approach.

Table 5.8: Bounds estimates of a two dimensional integral with the RDA method.

RDA Method		
order	domain interval boxes	bounds estimate
5	1	[1.445463096 , 1.865215766]
	2×2	[1.651327785 , 1.660154065]
	4×4	[1.655670149 , 1.655858472]
10	1	[1.649379060 , 1.662771976]
	2×2	[1.655760180 , 1.655769308]
	4×4	[1.655764708 , 1.655764714]

The situation becomes more dramatic in higher dimensions. Since it is very difficult to find any suitable formula for such an example in books, we extended the definite integral (5.11) to the four dimensional case.

$$\int_0^{\frac{\pi}{2}}\int_0^{\frac{\pi}{2}}\int_0^{\frac{\pi}{2}}\int_0^{\frac{\pi}{2}} \left(\frac{\sin y\sqrt{1-k^2\sin^2 x\sin^2 y}}{1-k^2\sin^2 y} + \frac{\sin w\sqrt{1-k^2\sin^2 z\sin^2 w}}{1-k^2\sin^2 w} \right) dx\,dy\,dz\,dw$$

$$= \frac{\pi^3}{4\sqrt{1-k^2}}.$$

The definite integral for $k^2 = 0.1$ is 8.170871339259325. Similar to before, computations are made to obtain the bounds with the interval method and the RDA method as shown in Table 5.9 and 5.10.

Table 5.9: Bounds estimates of a four dimensional integral with the interval method.

Interval Method	
domain interval boxes	bounds estimate
1	[0.7073129582E-15, 13.52904042]
4^4	[6.344461569 , 9.814754327]
16^4	[7.730435550 , 8.599902240]

Table 5.10: Bounds estimates of a four dimensional integral with the RDA method.

RDA Method		
order	domain interval boxes	bounds estimate
5	1	[7.133074468 , 9.204470869]
	2^4	[8.148975985 , 8.192531932]
	4^4	[8.170404693 , 8.171334032]
10	1	[8.139359412 , 8.205450807]
	2^4	[8.170848978 , 8.170894025]
	4^4	[8.170871325 , 8.170871354]

5.5.3 Bounds of Normal Form Deviation Function

The last example in this section is a bounds estimate of a deviation function from a normal form invariance in six dimensions discussed in chapter 1. The function is a six dimensional polynomial up to roughly 200th order which involves a large number of local minima and maxima and has many cancellations, thus representing a substantial challenge for interval methods. We made the estimate in the domain around the reference point $\vec{x}_0 = \overrightarrow{0.05}$ with the width in each dimension 0.02

$$[0.04, 0.06] \times [0.04, 0.06] \times [0.04, 0.06] \times [0.04, 0.06] \times [0.04, 0.06] \times [0.04, 0.06].$$

The value of the function at the reference point is

$$f(\vec{x}_0) = 0.6976700784514303 \times 10^{-5},$$

and a bounds estimate is obtained by scanning in real numbers at the total of 1729 points in the whole domain, which consist of the 3^6 equidistant points including boundary points, and 1000 random points, as

```
[-0.31211856E-05, 0.42124293E-04].
```

While this bounds estimate can give some idea, it is not very reliable considering the small number of sampling points and the specific feature of the function. The interval arithmetic covering the whole domain by one interval box gives a mathematically rigorous bounds

```
[ -4.47134    ,  4.80774    ],
```

which dramatically shows the typical blow-up problem in interval arithmetic. By dividing the domain in question into smaller interval boxes, we expect to have a narrower bounds estimate. Table 5.11 shows bounds estimates in successively smaller domain interval boxes at various locations. Only the smallest boxes yield bounds of a size comparable to those obtained by the scanning estimate. However, to cover the entire domain in this fashion requires 10^{24} small interval boxes, showing the practical limitations of the interval approach for this problem.

Table 5.11: Bounds estimates of a normal form deviation function with the interval method.

Interval Method		
domain interval box	bounds estimate	width
$[0.040000, 0.060000]^6$	[-4.47134 , 4.80774]	9.27908
$[0.040000, 0.042000]^6$	[-0.281964E-02, 0.424588E-02]	0.70655E-02
$[0.040000, 0.040200]^6$	[-0.311303E-03, 0.327498E-03]	0.63880E-03
$[0.040000, 0.040020]^6$	[-0.304618E-04, 0.329529E-04]	0.63414E-04
$[0.040000, 0.040002]^6$	[-0.199253E-05, 0.434435E-05]	0.63368E-05
$[0.049000, 0.051000]^6$	[-0.544365E-02, 0.100332E-01]	0.15476E-01
$[0.049900, 0.050100]^6$	[-0.697752E-03, 0.762484E-03]	0.14602E-02
$[0.049990, 0.050010]^6$	[-0.657704E-04, 0.802314E-04]	0.14600E-03
$[0.049999, 0.050001]^6$	[-0.320844E-06, 0.142793E-04]	0.14600E-04
$[0.058000, 0.060000]^6$	[-0.133070E-01, 0.265293E-01]	0.39836E-01
$[0.059800, 0.060000]^6$	[-0.160977E-02, 0.188511E-02]	0.34948E-02
$[0.059980, 0.060000]^6$	[-0.144312E-03, 0.207897E-03]	0.35220E-03
$[0.059998, 0.060000]^6$	[0.131290E-04, 0.483782E-04]	0.35249E-04

On the other hand, the Taylor model computation gives very small remainder bounds as shown in Table 5.12 computed in various orders, and the total bounds estimates are comparable to the bounds by the scanning. Table 5.12 also shows the number of terms of polynomials, which is very moderate compared to the division number necessary for a comparable interval evaluation.

Table 5.12: Bounds estimates of a normal form deviation function with the RDA method.

RDA Method			
order	terms	remainder bounds interval	total bounds
6	924	[-0.53585E-05,0.53588E-05]	[-0.3466E-04,0.5358E-04]
7	1716	[-0.83873E-06,0.83884E-06]	[-0.3016E-04,0.4902E-04]
8	3003	[-0.12321E-06,0.12321E-06]	[-0.2945E-04,0.4831E-04]

Chapter 6

Verified Integration of ODEs and Flows

In this chapter, Remainder-enhanced Differential Algebraic (RDA) methods are applied for the development of verified integration algorithms for ODEs and flows of ODEs. We will use anti-derivations of Taylor models for the solution of the initial value problem

$$\frac{d}{dt}\vec{r}(t) = \vec{F}(\vec{r}(t), t), \tag{6.1}$$

where $\vec{r}(t_0) = \vec{r}_0$ and $\vec{F}$ is continuous and bounded. We are interested in both the case of a specific initial condition $\vec{r}_0$, as well as the case in which the initial condition $\vec{r}_0$ is a variable and our interest is in the flow of the differential equation

$$\vec{r}(t) = \mathcal{M}(\vec{r}_0, t).$$

6.1 Verified Integration with Taylor Models

The goal is to establish a Taylor model for $\mathcal{M}(\vec{r}_0, t)$, and in particular rigorous bounds for the remainder term of the flow of the differential equation over a domain $[\vec{r}_{01}, \vec{r}_{02}] \times [t_0, t_1]$. In particular, $\vec{r}_0$ itself may be a Taylor model, as long as its range is known to lie in $[\vec{r}_{01}, \vec{r}_{02}]$. We have to deviate from the direct use of conventional numerical integrators, because they don't provide rigorous estimates for the integration error, but only approximate estimates. Rather, we have to start from scratch from the foundations of the theory of differential equations [18].

6.1.1 Schauder's Fixed Point Theorem

As is common for the application of functional analysis tools to the study of differential equations, we re-write the differential equation (6.1) as an integral equation

$$\vec{r}(t) = \vec{r}_0 + \int_{t_0}^{t} \vec{F}(\vec{r}(t'), t')\, dt',$$

noting that the initial value problem has a (unique) solution if and only if the corresponding integral equation has a (unique) solution. Now we introduce the operator

$$A : \vec{C}^0[t_0, t_1] \to \vec{C}^0[t_0, t_1]$$

on the space of continuous functions from $[t_0, t_1]$ to R^n via

$$A\left(\vec{f}\right)(t) = \vec{r}_0 + \int_{t_0}^{t} \vec{F}(\vec{f}(t'), t')\, dt'; \tag{6.2}$$

so a general function $\vec{f}$ in $\vec{C}^0[t_0, t_1]$ is transformed into a new function in $\vec{C}^0[t_0, t_1]$ via the insertion into $\vec{F}$ and subsequent integration. Having introduced the operator A, the problem of finding a solution to the differential equation is reduced to a fixed-point problem

$$\vec{r} = A(\vec{r}).$$

It is common fare in the theory of differential equations to establish that Schauder's fixed point theorem asserts the existence of a solution of an ODE over $[t_0, t_1]$ in case $\vec{F}$ is continuous on $[t_0, t_1] \times R^n$ and bounded there. If $\vec{F}$ is even Lipschitz with respect to the first argument $\vec{f}$, then Banach's fixed point theorem even asserts a locally unique solution.

We will now apply Schauder's fixed point theorem in a different way to rigorously obtain a Taylor Model for the flow.

Theorem (Schauder): *Let A be a continuous operator on the Banach Space X. Let $M \subset X$ be compact and convex, and let $A(M) \subset M$. Then A has a fixed point in M, i.e. there is an $\vec{r} \in M$ such that $A(\vec{r}) = \vec{r}$.*

One should be reminded that the fixed point is not necessarily unique, for example, the identity map on M has every element of M as fixed points; furthermore compactness and convexity of M are essential, as simple counter-examples show.

6.1.2 Strategy to Satisfy the Requirements

In our specific case, $X = \vec{C}^0[t_0, t_1]$, the space of continuous vector functions on the interval, equipped with the usual maximum norm, and A is the integral operator in (6.2). From continuity of $\vec{F}$, it follows easily that A is continuous on X. The process of our application of Schauder's theorem now has three major steps:

1. Determine a sufficiently large family Y of subsets of X from which to draw candidates for the set M. To satisfy the requirements of Schauder's theorem, the sets in Y have to be **compact** and **convex**; and to fit within our computational framework, it should be possible to contain them in suitable **Taylor models**.

2. Using the Differential Algebraic structure on Taylor models, construct an initial set $M_0 \in Y$ that satisfies the **inclusion** property $A(M_0) \subset M_0$. Once this set has been determined, all requirements of the fixed point theorem are satisfied, and the existence of a solution in M_0 has been established. Since the sets in Y were chosen in such a way that they can be contained in Taylor models, a Taylor model inclusion of a solution of the ODE has been found.

3. Finally, the set M_0 is iteratively reduced in size in order to obtain bounds that are as sharp as possible. For this purpose, for $i = 1, 2, 3, ...$ we construct the

sequence $M_i = A(M_{i-1})$. We have the chain $M_1 \supset M_2 \supset ...$, and we continue to **iterate** until no significant further reduction in size is possible.

6.1.3 Schauder Candidate Sets

For the first step, it is necessary to establish a family of sets Y from which to draw candidates for M_0. We define Y in the following way. Let $(\vec{P}+\vec{I})$ be a Taylor model depending on time t as well as the initial condition $\vec{r}_0$. Then we define the associated set $M_{\vec{P}+\vec{I}}$ as follows:

$$
\begin{aligned}
M_{\vec{P}+\vec{I}} &\subset \vec{C}^0[t_0,t_1]; \quad \text{and for } \vec{r} \in M_{\vec{P}+\vec{I}}: \\
\vec{r}(t_0) &= \vec{r}_0 \\
\vec{r}(t) &\in \vec{P}+\vec{I} \quad \forall t \in [t_0,t_1] \quad \forall \vec{r}_0 \\
|\vec{r}(t') - \vec{r}(t'')| &\leq k|t'-t''| \quad \forall t', t'' \in [t_0,t_1] \quad \forall \vec{r}_0.
\end{aligned}
$$

In the last condition, k is bounds for $\vec{F}$, which exists because $\vec{F}$ is continuous and the solutions can cover only finite range over interval $[t_0, t_1]$. The last condition means that all $\vec{r} \in M_{\vec{P}+\vec{I}}$ are uniformly Lipschitz with constant k. Define the family of candidate sets Y as

$$Y = \bigcup_{\vec{P}+\vec{I}} M_{\vec{P}+\vec{I}}.$$

6.1.4 Convexity, Compactness, and Invariance

First let us check the convexity of the Schauder candidate sets defined above.

Definition (Convexity): *Let X be a real vector space. $M \subset X$ is called convex if $\forall x_1, x_2 \in M,\ \forall \alpha \in [0,1],\ \alpha x_1 + (1-\alpha)x_2 \in M$.*

Let $M \subset Y$ be a Schauder candidate set. Then M is convex, because

$$\vec{x}_1, \vec{x}_2 \in M \Rightarrow$$

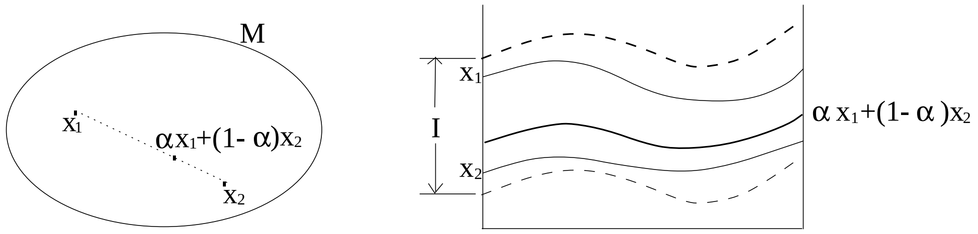

Figure 6.1: Convexity; a set of numbers M (left) and a set of functions inside a Taylor model (right).

$$\alpha\vec{x}_1 + (1-\alpha)\vec{x}_2 \quad \in \quad M \; \forall\, \alpha \in [0,1],$$

as any such linear combination of two k-Lipschitz functions is k-Lipschitz, is in the same Taylor models as $\vec{x}_1$ and $\vec{x}_2$, and assumes the value $\vec{r}_0$ at t_0.

It is a little more involved to show that M is compact, and it needs a help of the Ascoli-Arzela Theorem.

Definition (Compactness): *Let (X, d) be a metric space. $M \subset X$ is called compact if every sequence in M has at least one cluster point in M.*

Theorem (Ascoli-Arzela): *Let (x_n) be a uniformly equicontinuous sequence of functions on $[t_0, t_1]$, and let (x_n) be uniformly bounded, i.e.*

$$\exists K \in R: \; |x_n(t)| < K \;\; \forall t \in [t_0, t_1], \; \forall n \in N.$$

Then, there exists a subsequence of (x_n) that converges uniformly on $[t_0, t_1]$.

To see that M is compact, let $(\vec{x}_n)$ be a sequence of functions in M. Then all $\vec{x}_n$ are k-Lipschitz and hence uniformly equicontinuous; since they are in the same Taylor model, they are uniformly bounded. Thus according to the Ascoli-Arzela theorem, $(\vec{x}_n)$ has a uniformly convergent subsequence. Let $\vec{x}^*$ be the limit of this subsequence. Since the $\vec{x}_n$ are continuous, so is $\vec{x}^*$, and we obviously have $\vec{x}^*(t_0) = \vec{r}_0$. Since the elements of the subsequence converging to $\vec{x}^*$ are k-uniformly Lipschitz, so is $\vec{x}^*$ itself,

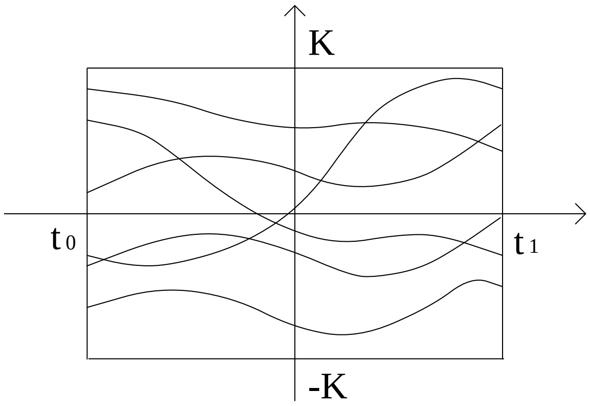

Figure 6.2: Functions on $[t_0, t_1]$ are uniformly equicontinuous and uniformly bounded. (The Ascoli-Arzela Theorem)

as a simple indirect proof reveals. Similarly, since the subsequence converging to $\vec{x}^*$ is in $\vec{P}+\vec{I}$, so is $\vec{x}^*$. The interesting to notice is that the schematic explanation of the Ascoli-Arzela theorem in Figure 6.2 reminds us of Taylor models.

Finally, the operator A maps any set in Y into another set in Y. Indeed, the image functions of A go through $\vec{r}_0$ and are continuous because they are integrals, and they are k-Lipschitz because $\vec{F}$ is bounded by k. Finally, since A is continuous, all images of functions inside a Taylor model are bounded and hence themselves in a Taylor model.

Hence the entire problem is reduced to finding a Taylor model $\vec{P}+\vec{I}$ such that

$$A(\vec{P}+\vec{I}) \subset \vec{P}+\vec{I},$$

a condition which can be checked computationally using the differential algebraic operations on the set of Taylor models.

6.1.5 Satisfying the Inclusion Requirement with Differential Algebraic Methods

For practical purposes it is of course in addition desirable to have $\vec{I}$ small. For this purpose it turns out to be important to determine a starting candidate that is on the one hand sufficiently small in width, but on the other hand shaped in such a way as to contain the true solution. This thought leads to attempt sets M^* of the form

$$M^* = M_{\mathcal{M}_n(\vec{r},t)+\vec{I}^*},$$

where $\mathcal{M}_n(\vec{r},t)$ is n-th order Taylor expansion of the solution. If n is high enough, we may expect that the true solution of the ODE and hence the fixed point problem is sufficiently close to the $n-$th order expansion, and hence that it may be possible to choose $\vec{I}^*$ rather small.

This approach requires the knowledge of the solution $\mathcal{M}_n(\vec{r},t)$, and contrary to the usual situation in which we are only interested in $\mathcal{M}_n(\vec{r},t)$ at the final value of t, here the explicit dependence on t is required. This quantity can be obtained by iterating (6.2) within the framework of DA. To begin with, one chooses an initial function

$$\mathcal{M}_n^{(0)}(\vec{r},t) = \mathcal{I},$$

where $\mathcal{I}$ is the identity function, and then iteratively sets

$$\mathcal{M}_n^{(k+1)} =_n A(\mathcal{M}_n^{(k)}).$$

This process converges to the exact result $\mathcal{M}_n$ as a Truncated Taylor Series in $n+1$ steps.

As the next step, we try to find $\vec{I}^*$ such that in fact the inclusion property necessary for Schauder's theorem is satisfied, namely

$$A(\mathcal{M}_n(\vec{r},t) + \vec{I}^*) \subset \mathcal{M}_n(\vec{r},t) + \vec{I}^*.$$

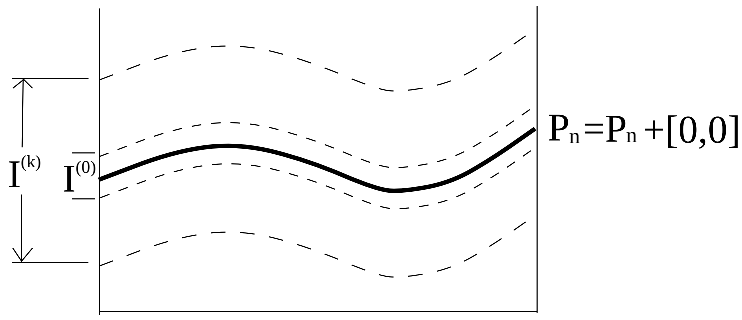

Figure 6.3: Finding an inclusion set as a Taylor model satisfying Schauder requirement.

The suitable choice of $\vec{I}$ requires a little experimenting, it is however greatly simplified by the observation that it is necessary that computationally,

$$\vec{I}^* \supset \vec{I}^{(0)},$$

where $\vec{I}^{(0)}$ is defined as

$$\mathcal{M}_n(\vec{r}, t) + \vec{I}^{(0)} = A(\mathcal{M}_n(\vec{r}, t) + [\vec{0}, \vec{0}]). \tag{6.3}$$

$\vec{I}^{(0)}$ is a good benchmark for the size of intervals that is to be expected; and so we iteratively try the sequence

$$\vec{I}^{(k)} = 2^k \cdot \vec{I}^{(0)}, \tag{6.4}$$

until a computational inclusion can be found, which means that we have established

$$A(\mathcal{M}_n(\vec{r}, t) + \vec{I}^{(k)}) \subset \mathcal{M}_n(\vec{r}, t) + \vec{I}^{(k)}. \tag{6.5}$$

Once this computational inclusion has been determined, a solution of the ODE is certainly contained in the Taylor model $\mathcal{M}_n(\vec{r}, t) + \vec{I}^{(k)}$, which satisfies our demand.

6.1.6 Iterative Refinement of the Inclusion

For practical purposes it is useful that the sharpness of this solution can be improved. Denoting $\vec{I}_1 = \vec{I}^{(k)}$, the first obtained interval satisfying our requirement, we itera-

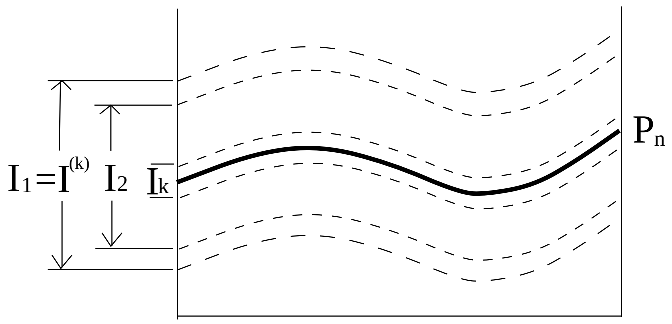

Figure 6.4: Iterative refinement of the inclusions as Taylor models.

tively define a sequence of Taylor models

$$\mathcal{M}_n(\vec{r},t) + \vec{I}_k = A(\mathcal{M}_n(\vec{r},t) + \vec{I}_{k-1}).$$

We then must have $\vec{I}_k \subset \vec{I}_{k-1}$ to get refinement for all $k = 2, 3,...$ To see this, we observe that this is the case for $k = 2$ by definition of $\vec{I}_1$, and then we infer inductively

$$\begin{aligned} \mathcal{M}_n(\vec{r},t) + \vec{I}_k &\subset \mathcal{M}_n(\vec{r},t) + \vec{I}_{k-1} \Rightarrow \\ A(\mathcal{M}_n(\vec{r},t) + \vec{I}_k) &\subset A(\mathcal{M}_n(\vec{r},t) + \vec{I}_{k-1}) \Rightarrow \\ \mathcal{M}_n(\vec{r},t) + \vec{I}_{k+1} &\subset \mathcal{M}_n(\vec{r},t) + \vec{I}_k. \end{aligned}$$

Furthermore, the fixed point function $\vec{r}$ must actually be contained in each of the elements of the sequence of Taylor models $\mathcal{M}_n(\vec{r},t) + \vec{I}_k$. Again by definition of $\vec{I}_1$, the fixed point function is contained in $\mathcal{M}_n(\vec{r},t) + \vec{I}_1$, as mentioned in the previous subsection. And by induction, we see

$$\begin{aligned} \vec{r} &\in \mathcal{M}_n(\vec{r},t) + \vec{I}_k \Rightarrow \\ A(\vec{r}) &\in A(\mathcal{M}_n(\vec{r},t) + \vec{I}_k) \Rightarrow \\ \vec{r} &\in \mathcal{M}_n(\vec{r},t) + \vec{I}_{k+1}. \end{aligned}$$

So this provides a mechanism to iteratively refine the inclusion until no further worthwhile decrease in size can be obtained.

6.2 Example: Remainder Bounds for a Dipole of the S800 Spectrograph

In this section, we will provide an example for the practical use and performance of the method discussed above. In this example, we analyze the motion of a charged particle in a dipole of the S800 spectrograph [60] with a deflection radius of 2.67m and a deflection angle of 75 degrees (Refer to Table 3.8). The Taylor transfer map of the system of differential equations with remainder bounds for a region of initial conditions is determined. The motion is described by the coupled differential equations (2.68), (2.69), (2.70) and (2.71) derived in chapter 2, where $h = 1/R$ with R denoting the deflection radius and $B_y = p_0/eR$, and $p = p_0$.

The integration was carried out via many steps through the dipole, and the Taylor polynomials describing the dependence of the four final coordinate values x, a, y, b on the initial coordinate values and their error bounds were determined. At each step, Taylor models as solutions of the fixed point problem were sought to satisfy the computational inclusion requirement. The computation was performed with various conditions for the comparison. The initial conditions of four phase space variables were chosen to be within the domain interval box $[-\overrightarrow{.01}, \overrightarrow{.01}]$, or $[-\overrightarrow{.02}, \overrightarrow{.02}]$. The order in time and initial conditions was chosen to be 5, 10 or 12, and the fixed step size of 1 degree or 0.5 degree was used. The error bounds estimates under those computational conditions are listed in Table 6.1, 6.2 and 6.3. The overall accuracy goes below 10^{-8} using the twelfth order Taylor models with a fixed step size of 0.5 degree with the initial condition within the domain interval box $[-\overrightarrow{.01}, \overrightarrow{.01}]$. Since no automatic step size control was utilized, the estimates proved conservative and the actual resulting errors were somewhat lower.

Table 6.1: Error bounds of the Taylor transfer map of a S800-like dipole with initial condition within $[-0.02, 0.02]^4$ and step size $1°$.

Error Bounds of the Map with Initial Condition within $[-0.02, 0.02]^4$			
step size	order	error bounds estimate	
$1°$	5	x	[-.1149558375574990E-03,0.9296078153993057E-04]
		a	[-.4666650061895221E-04,0.4946294642968968E-04]
		y	[-.4526632143133069E-05,0.4526632143133069E-05]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	10	x	[-.4736194115913304E-06,0.7045006167034195E-06]
		a	[-.4457265344693460E-07,0.4011287904726302E-07]
		y	[-.1399063567696992E-07,0.1399063567696992E-07]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	12	x	[-.6401828174754347E-07,0.4534782588950649E-07]
		a	[-.9293828606969869E-08,0.1002164821121110E-07]
		y	[-.2715296293606005E-08,0.2715296293606005E-08]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]

Table 6.2: Error bounds of the Taylor transfer map of a S800-like dipole with initial condition within $[-0.01, 0.01]^4$ and step size $1°$.

Error Bounds of the Map with Initial Condition within $[-0.01, 0.01]^4$			
step size	order	error bounds estimate	
$1°$	5	x	[-.4774670173984800E-04,0.4224639157994040E-04]
		a	[-.2325461997662418E-04,0.2395429647130834E-04]
		y	[-.1118914478174159E-05,0.1118914478174159E-05]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	10	x	[-.1148378507775330E-06,0.1721148653001733E-06]
		a	[-.7449115285435551E-08,0.6534666433615125E-08]
		y	[-.1680086507721105E-08,0.1680086507721105E-08]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	12	x	[-.1295756988293815E-07,0.8486627448846389E-08]
		a	[-.6603760216278913E-09,0.7494712313226484E-09]
		y	[-.1777086692946516E-09,0.1777086692946516E-09]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]

Table 6.3: Error bounds of the Taylor transfer map of a S800-like dipole with initial condition within $[-0.01, 0.01]^4$ and step size $0.5°$.

Error Bounds of the Map with Initial Condition within $[-0.01, 0.01]^4$			
step size	order	error bounds estimate	
$0.5°$	5	x	[-.2309808273859498E-04,0.2031288462862392E-04]
		a	[-.1148663798183059E-04,0.1181920378129873E-04]
		y	[-.5527454999463349E-06,0.5527454999463349E-06]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	10	x	[-.5756202685083773E-07,0.8538622654758211E-07]
		a	[-.3131363953715242E-08,0.2868160910483114E-08]
		y	[-.8407415598669751E-09,0.8407415598669751E-09]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]
	12	x	[-.6429678314475644E-08,0.4256224895997219E-08]
		a	[-.3003741054797643E-09,0.3298718801248672E-09]
		y	[-.8913751143938564E-10,0.8913751143938564E-10]
		b	[0.0000000000000000E+00,0.0000000000000000E+00]

Table 6.4 shows an example of the typical computational behavior in the process to find Taylor models as solutions of the fixed point problem in a step of the verified integration of the S800-like dipole. We are considering the first s step of the integration using fifth order Taylor model computation with a step size of 1 degree with the initial condition within the domain interval box $[-\overrightarrow{.01}, \overrightarrow{.01}]$. The benchmark intervals $\vec{I}^{(0)}$ and $\vec{I}^{(k)}$ in (6.3), (6.4) and their mapped intervals via the operation A are estimated as follows until the inclusion requirement of (6.5) is met, which here was achieved in two steps. The resulting mapped Taylor models are the first solution of the fixed point problem to satisfy the inclusion requirement in this s step. The list shows only the error bound intervals of Taylor models.

Table 6.4: Computational behavior of the iteration to find the first solution of the inclusion requirement.

Iteration to Find the First Solution of Inclusion Requirement			
iteration step		error bounds	
	candidate set	x	[-.851609168266E-10,0.117999365650E-09]
		a	[-.107333446719E-11,0.110180867568E-11]
		y	[-.549821641117E-10,0.549821641117E-10]
$k=0$	$\vec{I}^{(0)}$	b	[0.000000000000E+00,0.000000000000E+00]
	mapped set	x	[-.426306772250E-10,0.590512338784E-10]
		a	[-.130800710095E-11,0.110758538377E-11]
		y	[-.274910820558E-10,0.274910820558E-10]
		b	[0.000000000000E+00,0.000000000000E+00]
	candidate set	x	[-.170321833653E-09,0.235998731300E-09]
		a	[-.214666893438E-11,0.220361735136E-11]
		y	[-.109964328223E-09,0.109964328223E-09]
$k=1$	$\vec{I}^{(1)}$	b	[0.000000000000E+00,0.000000000000E+00]
	mapped set	x	[-.426808960367E-10,0.591027849318E-10]
		a	[-.207934696832E-11,0.166426642970E-11]
		y	[-.274910820558E-10,0.274910820558E-10]
		b	[0.000000000000E+00,0.000000000000E+00]

The trial to refine the solution discussed in subsection 6.1.6 was made, but because the intervals are already so small, in this case no significant refinement resulted.

After the integration through the whole dipole via many s steps was done, the resulting Taylor polynomials describing the dependence of final on initial coordinates were compared with those obtained by DA computation, and agreement was found.

A check about the validity of the remainder bound intervals was done by launching a large collection of rays through the dipole. Because of the homogeneity of the field, orbits can be calculated from purely geometric arguments, and these results were compared by the prediction of the twelfth order map obtained from the verified integrator. For all the 3^4 rays studied, originating from the interval box of the initial conditions, the difference between the final coordinates determined geometrically and those predicted by the Taylor polynomials via the verified integrator were within the calculated remainder bounds.

BIBLIOGRAPHY

Bibliography

[1] G. Alefeld and J. Herzberger. *Introduction to Interval Computations.* Academic Press, 1983.

[2] A. O. Barut. *Electrodynamics and Classical Theory of Fields and Particles.* Dover, 1964, 1980.

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

[4] M. Berz. Arbitrary order description of arbitrary particle optical systems. *Nuclear Instruments and Methods*, A298:426, 1990.

[5] M. Berz. *High-Order Computation and Normal Form Analysis of Repetitive Systems, in: M. Month (Ed), Physics of Particle Accelerators*, volume AIP 249, page 456. American Institute of Physics, 1991.

[6] M. Berz. COSY INFINITY Version 6. In *M. Berz, S. Martin and K. Ziegler (Eds.), Proc. Nonlinear Effects in Accelerators*, page 125. IOP Publishing, 1992.

[7] M. Berz. New features in COSY INFINITY. In *Third Computational Accelerator Physics Conference*, page 267. AIP Conference Proceedings 297, 1993.

[8] M. Berz. Modern map methods for charged particle optics. *Nuclear Instruments and Methods*, 363:100, 1995.

[9] M. Berz. Differential algebras with remainder and rigorous proofs of long-term stability. In *Fourth Computational Accelerator Physics Conference*, volume 391, page 221. AIP Conference Proceedings, 1996.

[10] M. Berz. COSY INFINITY version 7. In *1997 Particle Accelerator Conference.* APS, 1997.

[11] M. Berz. COSY INFINITY Version 8 reference manual. Technical Report MSUCL-1088, National Superconducting Cyclotron Laboratory, Michigan State University, East Lansing, MI 48824, 1997.

[12] M. Berz. From Taylor series to Taylor models. *AIP*, 405:1–25, 1997.

[13] M. Berz, C. Bischof, A. Griewank, G. Corliss, and Eds. *Computational Differentiation: Techniques, Applications, and Tools.* SIAM, Philadelphia, 1996.

[14] M. Berz and G. Hoffstätter. Exact bounds of the long term stability of weakly nonlinear systems applied to the design of large storage rings. *Interval Computations*, 2:68–89, 1994.

[15] M. Berz, G. Hoffstätter, W. Wan, K. Shamseddine, and K. Makino. COSY INFINITY and its applications to nonlinear dynamics. *in: Computational Differentiation: Techniques, Applications, and Tools, M. Berz, C. Bischof, G. Corliss, A. Griewank (Eds.)*, SIAM, 1996.

[16] M. Berz, K. Joh, J. A. Nolen, B. M. Sherrill, and A. F. Zeller. Reconstructive correction of aberrations in nuclear particle spectrographs. *Physical Review C*, 47,2:537, 1993.

[17] M. Berz and K. Makino. Arbitrary order maps, remainder terms, and long term stability in particle accelerators. In *Optical Science, Engineering and Instrumentation '97*. SPIE, 1997.

[18] M. Berz and K. Makino. Verified integration of ODEs and flows with differential algebraic methods on Taylor models. *Reliable Computing*, in print.

[19] M. Berz and K. Makino. Perturbative determination of first integrals and Lyapunov functions for dynamical systems near fixed points. *Journal of Symbolic Computation*, submitted.

[20] M. Berz, K. Makino, K. Shamseddine, and W. Wan. *Applications of Modern Map Methods in Particle Beam Physics*. Academic Press, Orlando, Florida, in print, 1998.

[21] M. Berz, S. Martin, and K. Ziegler (Eds.). *Proceedings International Workshop on Nonlinear Problems in Accelerators Berlin*. Institute of Physics Publishing, Bristol, 1993.

[22] I. M. Bomze, T. Csendes, R. Horst, and P. M. Pardalos, editors. *Developments in Global Optimization*. KLUWER, 1997.

[23] J. A. Caggiano, D. Bazin, B. S. Davids, R. Foutus, D. Karnes, P. Johnson, B. Sherrill, and A. Zeller. S800 spectrograph dipole mapping. Annual Report of the Michigan State University National Superconducting Cyclotron Laboratory, 1996.

[24] J. A. Caggiano and B. M. Sherrill. Data analysis techniques for S800 dipole magnetic field maps. Annual Report of the Michigan State University National Superconducting Cyclotron Laboratory, 1995.

[25] D. C. Carey. *The Optics of Charged Particle Beams*. Harwood, 1987.

[26] A. W. Chao. *Physics of Collective Beam Instabilities in High Energy Accelerators*. Wiley, 1993.

[27] Ingrid Daubechies. *Ten Lectures on Wavelets*. SIAM, Philadelphia, 1992.

[28] D. A. Edwards and M. J. Syphers. *An Introduction to the Physics of High Energy Accelerators*. Wiley, 1993.

[29] H. Goldstein. *Classical Mechanics*. Addison-Wesley, Reading, MA, 1980.

[30] I. S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, New York, 1980.

[31] A. Griewank, G. F. Corliss, and Eds. *Automatic Differentiation of Algorithms*. SIAM, Philadelphia, 1991.

[32] E. Hansen. *Global Optimization using Interval Analysis*. Marcel Dekker, 1992.

[33] E. R. Hansen. Global optimization using interval analysis – the one–dimensional case. *J. Optim. Theor. and Appl.*, 29:331–334, 1979.

[34] E. R. Hansen. *An Overview of Global Optimization Using Interval Analysis*, pages 289–307. Academic Press, New York, 1988.

[35] P. W. Hawkes and E. Kasper. *Principles of Electron Optics*. Academic Press, London, 1989.

[36] G. Hoffstätter and M. Berz. Efficient computation of fringe fields using symplectic scaling. In *Third Computational Accelerator Physics Conference*, page 467. AIP Conference Proceedings 297, 1993.

[37] G. Hoffstätter and M. Berz. Accurate and fast computaton of high–order fringe field maps via symplectic scaling. *Nuclear Instruments and Methods*, 363:124, 1995.

[38] G. Hoffstätter and M. Berz. Symplectic scaling of transfer maps including fringe fields. *Physical Review E*, 54,4, 1996.

[39] G. H. Hoffstätter. *Rigorous bounds on survival times in circular accelerators and efficient computation of fringe–field transfer maps*. PhD thesis, Michigan State University, East Lansing, Michigan, USA, 1994. also DESY 94-242.

[40] S. Humphries. *Principles of Charged Particle Acceleration*. Wiley, New York, 1986.

[41] K. Ichida and Y. Fujii. An interval arithmetic method for global optimization. *Computing*, 23:85–97, 1979.

[42] J. D. Jackson. *Classical Electrodynamics*. Wiley, New York, 1975.

[43] C. Jansson. A global optimization method using interval arithmetic. *IMACS Annals of Computing and Applied Mathematics*, 1992.

[44] X. Jiye. *Aberration Theory in Electron and Ion Optics*. Advances in Electronics and Electron Physics, Supplement 17. Academic Press, Orlando, Florida, 1986.

[45] R. Baker Kearfott. *Rigorous Global Search: Continuous Problems*. KLUWER, 1996.

[46] R. Baker Kearfott and Vladik Kreinovich, editors. *Applications of Interval Computations*, volume 3. Kluwer, 1996.

[47] S. Kowalski and H. Enge. RAYTRACE. Technical report, MIT, Cambridge, Massachussetts, 1985.

[48] U. W. Kulisch and W. F. Miranker. *Computer Arithmetic in Theory and Practice.* Academic Press, New York, 1981.

[49] R. S. MacKay and J. D. Meiss. *Hamiltonian Dynamical Systems.* Adam Hilger, 1987.

[50] K. Makino. Rigorous integration of maps and long-term stability. In *1997 Particle Accelerator Conference.* APS, 1997.

[51] K. Makino and M. Berz. COSY INFINITY Version 7. In *Fourth Computational Accelerator Physics Conference*, volume 391, page 253. AIP Conference Proceedings, 1996.

[52] K. Makino and M. Berz. Remainder differential algebras and their applications. *in: Computational Differentiation: Techniques, Applications, and Tools, M. Berz, C. Bischof, G. Corliss, A. Griewank (Eds.)*, SIAM, 1996.

[53] K. Makino and M. Berz. Arbitrary order aberrations for elements characterized by measured fields. In *Optical Science, Engineering and Instrumentation '97.* SPIE, 1997.

[54] K. Makino and M. Berz. Implementation and applications of Taylor model methods. *Reliable Computing*, submitted, 1997.

[55] K. Makino and M. Berz. Verified quadrature and ODE integration involving elementary functions of high complexity. *Journal of Symbolic Computation*, submitted.

[56] L. Michelotti. *Intermediate Classical Dynamics with Applications to Beam Physics.* Wiley, 1995.

[57] Ramon E. Moore. *Methods and Applications of Interval Analysis.* SIAM, 1979.

[58] S. Moriguchi, K. Udagawa, and N. Ichimatsu. *Mathematics Formulas I.* Iwanami Zensho. Iwanami Shoten, Tokyo, 1956.

[59] A. Neumaier. *Interval Methods for Systems of Equations.* Cambridge, 1990.

[60] J. Nolen, A.F. Zeller, B. Sherrill, J. C. DeKamp, and J. Yurkon. A proposal for construction of the S800 spectrograph. Technical Report MSUCL-694, National Superconducting Cyclotron Laboratory, 1989.

[61] Annual Report of the Michigan State University National Superconducting Cyclotron Laboratory, 1994.

[62] Annual Report of the Michigan State University National Superconducting Cyclotron Laboratory, 1995.

[63] Annual Report of the Michigan State University National Superconducting Cyclotron Laboratory, 1996.

[64] M. Reiser. *Theory and Design of Charged Particle Beams*. Wiley, 1994.

[65] W. Wan. Private communication, 1997.

[66] R. L. Warnock and R. D. Ruth. Long–term bounds on nonlinear Hamiltonian motion. *Physica D*, 56(14):188–215, 1992. also SLAC–PUB–5267.

[67] H. Wiedemann. *Particle Accelerators Physics*. Springer-Verlag, 1994.

[68] H. Wiedemann. *Particle Accelerators Physics II*. Springer-Verlag, 1995.

[69] H. Wollnik. *Charged Particle Optics*. Academic Press, Orlando, Florida, 1987.