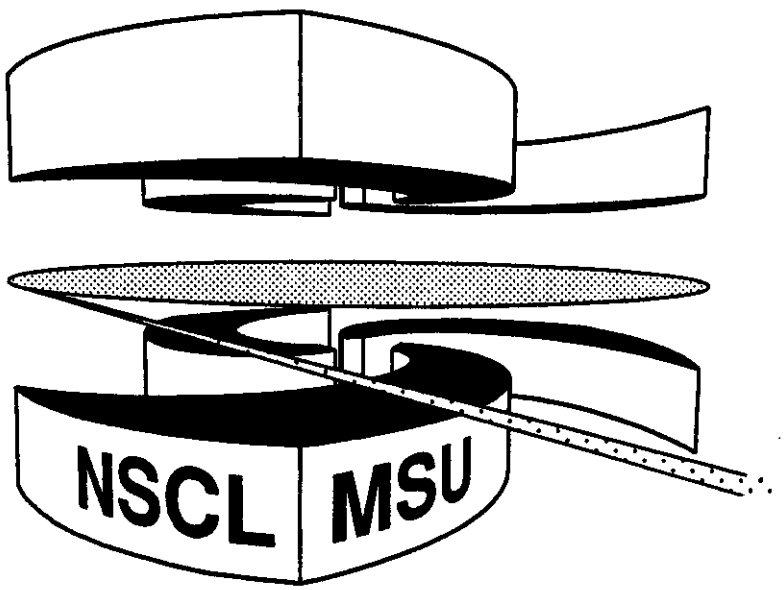
Michigan State University

National Superconducting Cyclotron Laboratory

DIRECT COMPUTATION AND CORRECTION OF CHROMATICITIES AND PARAMETER TUNE SHIFTS IN CIRCULAR ACCELERATORS

MARTIN BERZ



MSUCL-807

DECEMBER 1991

MSUCL-807
December, 1991

Direct Computation and Correction of Chromaticities and Parameter Tune Shifts in Circular Accelerators

Martin Berz

Department of Physics and Astronomy
and National Superconducting Cyclotron Laboratory
Michigan State University
East Lansing, Mi 48824

Abstract

A new efficient method for the direct and accurate computation and correction of chromaticities and parameter dependent tune shifts in accelerators is presented. The method is based on the differential algebraic treatment of beam dynamics and is applicable to chromaticities and tune shifts to arbitrary order. Contrary to the previous approach to obtain the same result, no Lie algebraic methods and no normal form theory is needed, resulting in a significantly more efficient algorithm. The method can be used for a direct correction of chromaticities. The method has been implemented in the design and simulation code COSY INFINITY and tested against the normal form approach as well as conventional numerical techniques.

1 Introduction

The dependence of the tune of a repetitive system on energy and amplitude as well as on system parameters is one of the most important characteristics of the system and usually has to be adjusted carefully. Thus the computation of these dependences is of prime importance. Traditionally, the linear tune of the system is readily calculated from the linear matrix following the Courant-Snyder theory [1].

The calculation of the tune dependence on energy, the chromaticity, and the dependence on the amplitude or emittance is more involved and usually requires extensive tracking and subsequent Fourier analysis. This task is particularly involved if the dependence on energy deviation or amplitude is rather nonlinear and if the dependence on a larger number of system parameters is required. In this case, often a rather large number of particles has to be analyzed, which significantly increases the effort.

Parallel to the numerical techniques there were efforts to compute the nonlinear tune dependences analytically. In this context the main idea was to perform a nonlinear change of variables to coordinates in which the motion performs a simple rotation with a frequency that does not change from turn to turn. The dependence of this frequency on energy, amplitude and perhaps parameters then directly gives the desired chromaticities and amplitude- and parameter tune shifts.

The pioneering idea in this direction was formulated under the name normal form theory in [2] in the Lie algebraic picture, and in a conceptually similar form in [3]. Outside the field of accelerator physics, similar ideas have been discussed in [4, 5]. It took several years before the technicalities were refined to allow a first practical use. In 1986 Neri [6], among other things benefiting from

an idea of Forest [7] that simplifies the treatment of the linear case, provided the first implementation of the Lie algebraic Dragt-Finn normal form theory to fifth order. The resulting program allowed the computation amplitude tune shifts, but chromaticities and parameter tune shifts were not directly accessible.

The more recent differential algebraic method [8, 9, 10, 11] allows an elegant arbitrary order extension of beam dynamics; in particular it allows the computation of the partial derivatives or power series expansion of the map relating the $2v$ phase space coordinates to their initial values

$$\vec{z}_f = \mathcal{M}(\vec{z}_i, \vec{\delta}) \tag{1}$$

The transfer map $\mathcal{M}$ is the unique flow of the differential equations of motion

$$\frac{d}{dt}\vec{z} = \vec{F}(\vec{z}, \vec{\delta}) \tag{2}$$

We denote the partial derivatives of $\mathcal{M}$ to order n by $]\mathcal{M}]_n$; formally, $[\ ]_n$ describes an equivalence class. In the underlying relation, functions are said to be equivalent if their value and derivatives at a point x agree up to order n. For details, we refer to [9]. We note that the determinant of the linear part of the transfer map is positive; for symplectic systems, it is even unity.

Very quickly it was realized [12] that it is possible to develop a general normal form algorithm by casting the Lie algebraic algorithm in a differential algebraic form. Albeit being a tour de force and somewhat unelegant, the resulting DA-based program was the first tool to compute the energy, parameter, and amplitude tune shifts for arbitrary systems to arbitrary order.

In [13] we showed that it is possible to develop the whole normal form algorithm in a much more direct and efficient way using differential algebraic techniques alone. Besides the gain in efficiency, which also entails a noticeable gain in computation speed, the method also allows the treatment of non-symplectic systems like electron rings with damping and gives new insight in their behaviour. Since any system has residual damping, the theory explains why resonances of very high order eventually become irrelevant. This resolves the classic paradox that it is necessary to stay away from any resonance while the forbidden lines lie dense in tune space.

This differential algebraic normal form algorithm presents a complete solution for the computation of all tune shifts and, like normal form theory in general, provides interesting insight. However, as we will show here, it is possible to present an even much more direct algorithm for the computation of some important tune shifts, namely the chromaticities and parameter tune shifts. The

striking simplicity of the algorithm once more seems to stress the power of the differential algebraic approach.

In sections 2 and 3, we will show how to perform the transformation to the energy and parameter dependent fixed point in the differential algebraic picture. This method is also the first step of both the differential algebra-Lie algebra algorithm [12] and the new purely differential algebraic method [13]. Section 4 describes the necessary decoupling of the motion into separate planes; section 5 provides a non-symplectic generalization of the Courant-Snyder theory of linear motion. Section 6, almost anticlimactically, shows how the ideas of the previous sections can be used for a direct differential algebraic computation of chromaticities and parameter tune shifts. Section 7 finally provides methods for chromaticity correction using systems parameters.

2 Inversion of DA Maps

In this section, we will give an important tool for the following theory. Given a map $\mathcal{A}$ to a certain order n, i.e. its equivalence class $[\mathcal{A}]_n$, we will give an algorithm for the computation of the equivalence class of the inverse of the map $[\mathcal{A}^{-1}]_n$ in an iterative manner in finitely many steps.

We begin by splitting the map $[\mathcal{A}]_n$ given to order n into its linear and nonlinear parts:

$$[\mathcal{A}]_n = [\mathcal{A}_1]_n + [\mathcal{A}_2]_n. \tag{3}$$

Composing $[\mathcal{A}]_n$ and its inverse and letting $\mathcal{E}$ denote the identity map, we obtain

$$\begin{aligned} ([\mathcal{A}_1]_n + [\mathcal{A}_2]_n) \circ [\mathcal{A}^{-1}]_n &= [\mathcal{E}]_n \Rightarrow \\ [\mathcal{A}_1] \circ [\mathcal{A}^{-1}]_n &= [\mathcal{E}]_n - [\mathcal{A}_2]_n \circ [\mathcal{A}^{-1}]_n \Rightarrow \\ [\mathcal{A}^{-1}]_n &= [\mathcal{A}_1^{-1}] \circ ([\mathcal{E}]_n - [\mathcal{A}_2]_n \circ [\mathcal{A}^{-1}]_{n-1}). \end{aligned} \tag{4}$$

In the last step use has been made of the fact that knowing $[\mathcal{A}]_{n-1}$ allows us to determine $[\mathcal{A}]_{2n} \circ [\mathcal{A}^{-1}]_n$ to order n since the first nonzero derivatives of $\mathcal{A}$ are of order 2. The necessary computation of $\mathcal{A}_1^{-1}$ is a linear matrix inversion. Apparently it is necessary that the linear part of the map $[\mathcal{A}_1]$ is invertible; the above algorithm shows that this is also sufficient.

Equation (4) can now be used in a recursive manner to compute the $[\mathcal{A}^{-1}]_i$ order by order. For the sake of completeness we note that in case $[\mathcal{A}]_n = [\mathcal{E}]_n +$

$[\mathcal{A}_m]_n$, where $[\mathcal{A}_m]_n$ has only derivatives of order m and higher, the algorithm converges much more quickly. This entails a noticeable gain in efficiency in [13].

3 Parameter Dependent Fixed Points

To study the behaviour of iterative nonlinear systems, it is advantageous (and for the tune shift computation also necessary) to first find a parameter dependent fixed point of the system. This parameter dependent fixed point satisfies

$$(\vec{z}_F, \vec{\delta}) = \mathcal{M}(\vec{z}_F, \vec{\delta}) \tag{5}$$

In general, a different set of parameters entails a different fixed point. The fixed point equation can be rewritten in the following way:

$$(\mathcal{M} - \mathcal{E}_H)(\vec{z}_F, \vec{\delta}) = (\vec{0}, \vec{\delta}) \tag{6}$$

where the map $\mathcal{E}_H$ contains a unity map in the upper block describing the variables and zeros everywhere else. This form of the fixed point equation clearly shows how the parameter dependent fixed point $\vec{z}_F$ can be obtained: it necessary to invert the map $\mathcal{M} - \mathcal{E}_H$. Since we are interested only in the properties of the the inverse of up to order n, we pass to the equivalence class of $[\mathcal{M} - \mathcal{E}]_n$ of the map, apply the results of the last section, and obtain the equivalence class $[\vec{z}_F]_n$. According to the previous section, an equivalence class inverse exists if and only if the linear part of the map is invertible.

While for transfer maps, this is always the case, here the situation is more subtle; clearly if the map $\mathcal{M}$ is the identity and $\vec{\delta} \neq 0$ no solution exists. A closer inspection reveals that the map is invertible if and only if the phase space part of $\mathcal{M}$ does not have 1 as an eigenvalue. But since this case corresponds to a fundamental resonance, it is always avoided in accelerator design. In this case, up to order n, the fixed point is given as the upper part of

$$(\vec{z}_F, \vec{\delta}) = (\mathcal{M} - \mathcal{E}_H)^{-1}(\vec{0}, \vec{\delta}) \tag{7}$$

In passing we note that if there is no constant part in the transfer map, this is the exact solution. If this is not the case, it may be necessary to iterate the above equation a few times.

As the first step in the computation of tune shifts, we perform a transformation to coordinates around the fixed point. In these coordinates, the map is

origin preserving, i.e. $\mathcal{M}(\vec{0}, \vec{\delta}) = \vec{0}$. This also implies that all partial derivatives of the final coordinates with respect to parameters vanish.

The key consequence of this is that we can now view the map such that the partial derivatives of the final phase space variables with respect to the initial phase space variables (the aberrations) depend on the system parameters, but the system parameters do not influence the map otherwise. So altogether, our map now relates initial phase space coordinates to final phase space coordinates, and the expansion coefficients depend on the parameters.

4 The Decoupling of Planes

In this section we will discuss the decoupling of the linear map around the fixed point into separate 2 by 2 blocks. This decoupling provides a stepping stone to the computation of the relevant accelerator quantities.

In the following we assume that the linear part of the phase space map has distinct Eigenvalues. Similar to the condition of 1 not being an eigenvalue imposed in the last section, this is commonly the case in repetitive systems, since otherwise the system is on a linear resonance. If there are $2v$ distinct Eigenvalues, the map can be diagonalized. Since the underlying matrix is real, complex Eigenvalues will always have accompanying conjugates as Eigenvalues. We now group the Eigenvalues such that all complex conjugate pairs form one pair; any remaining real Eigenvalues we group into pairs by just demanding that the elements of a pair have the same sign. This is possible since the determinant is positive and thus there is an even number of negative eigenvalues.

Each pair we write as $r_j \cdot e^{\pm i\mu_j}$. In the case of a complex pair, this is readily accomplished by choosing r_j and μ_j as the modulus and phase. In the real case, we choose $r_j = \pm\sqrt{R_{1j}R_{2j}}$, where the sign is determined to be the same as the one of $R_1 j$ and $R_2 j$. μ_j is chosen as $i \cdot \log(R_{1j}/R_{2j})$. Since the determinant is nonzero and R_1, R_2 are of the same sign, r_j and μ_j are always well defined.

Denoting the eigenvectors corresponding to $r_j e^{\pm i\mu}$ with $s_j^{\pm}$, we obtain that in the eigenvalue basis, the linear part of the map has the form

$$\begin{pmatrix} r_1 e^{+i\mu_1} & & & & & \\ & r_1 e^{-i\mu_1} & & & & \\ & & \ddots & & 0 & \\ & & & r_v e^{+i\mu_v} & & \\ & & 0 & & \ddots & \\ & & & & & r_v e^{-i\mu_v} \end{pmatrix} \tag{8}$$

We note that if the jth eigenvalue pair is complex conjugate, so are the associated eigenvectors, and if the jth eigenvalue pair is real, so are the eigenvectors.

We now perform another change of basis after which the matrix is real. For each conjugate pair of eigenvalues, we choose the real part and the imaginary parts as two basis vectors. For the pairs of real eigenvalues, we choose the two real eigenvectors directly.

The result of this basis change is a matrix that has two-by-two sub-blocks along the diagonal. A sub-block originating from a complex Eigenvalue pair will have four nonzero entries, and a sub-block originating from a real Eigenvalue pair will be diagonal. So altogether, the matrix has the form

$$\begin{pmatrix} a_1 & b_1 & & & & \\ c_1 & d_1 & & & & \\ & & \cdot & & 0 & \\ & 0 & & \cdot & & \\ & & & & a_v & b_v \\ & & & & c_v & d_v \end{pmatrix} \tag{9}$$

We note that if the underlying matrix is symplectic, it is possible to scale the transformation matrix such that it is also symplectic. Since products of symplectic matrices are symplectic, so is the transformed matrix.

5 The Computation of Dampings, Tunes and Tuneshifts

As a result of the last section, the linear $2v$ by $2v$ matrix consists of v two-by-two blocks along the diagonal. In this section we will review how to compute the linear tunes from this matrix. The reader will notice that most of the steps of this section could also be obtained as a byproduct of the eigenvector manipulations outlined in the last section. The reason why we perform the operations in this section separately is twofold.

Firstly, the eigenvector decomposition outlined in the last section is computationally expensive compared to the algebra of this section and can be avoided if the matrix is already in two-by-two block form. This however is always the case if the system has midplane symmetry, as many systems do. Secondly, the algorithms presented here are rather illuminating and extend Courant-Snyder arguments to the non-symplectic case.

We begin by reviewing some analytic formulas about eigenvectors and Eigenvalues of 2 by 2 matrices. Let

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \tag{10}$$

The characteristic polynomial of M has the form $\lambda^2 - T \cdot \lambda + D = 0$, where T and D are the trace and determinant of the matrix, respectively. Since trace and determinant are invariant under similarity transformations, so is the characteristic polynomial and hence the eigenvalues. Thus the eigenvalues assume the form:

$$\lambda_{1,2} = \frac{T}{2} \pm \sqrt{(\frac{T}{2})^2 - D} = \sqrt{D}\left(\frac{T}{2\sqrt{D}} \pm i\sqrt{1 - (\frac{T}{2\sqrt{D}})^2}\right) \tag{11}$$

where the last step requires $D \neq 0$. Introducing

$$\begin{aligned} r &= \sqrt{D} \\ \mu &= \text{sign}(b) \cdot \text{acos}\left(\frac{T}{2\sqrt{D}}\right), \end{aligned} \tag{12}$$

where $\text{sign}(b) = +1$ if $b \geq 0$, -1 else. The solutions can be written as $\lambda_{1,2} = r \cdot e^{\pm i\mu}$

Note that μ is always nonzero because otherwise there are two identical eigenvalues which we have already excluded in the previous section. It is purely real for $|T| \leq 2\sqrt{D}$ and purely imaginary otherwise. In the latter case, $\mu = i \cdot \text{acosh}(T/2\sqrt{D})$, and $\lambda_{1,2} = r \cdot e^{\mp|\mu|}$. For computational purposes it is useful to utilize that for $x > 1$, $\text{acosh}(x) = \ln[x + \sqrt{x^2 - 1}]$.

Now it is useful customary [1] to introduce new quantities, the so-called Twiss parameters:

$$\begin{aligned} \alpha &= \frac{a - d}{2r \sin \mu} \\ \beta &= \frac{b}{r \sin \mu} \\ \gamma &= \frac{-c}{r \sin \mu}. \end{aligned} \tag{13}$$

The Twiss parameters satisfy $\beta\gamma - \alpha^2 = 1$, so they are not independent. Two of them, together with r and μ, determine the matrix M. We note that β is never negative, and it vanishes only for $b = 0$ which implies real eigenvalues and hence imaginary μ.

Using the Twiss parameters, the matrix can be written as

$$M = r \cdot \begin{pmatrix} \cos(\mu) + \alpha \sin(\mu) & \beta \sin(\mu) \\ -\gamma \sin(\mu) & \cos(\mu) - \alpha \sin(\mu) \end{pmatrix} \tag{14}$$

For the sake of completeness we note that in the important case of $\beta \neq 0$, the eigenvectors assume the rather compact form

$$v_{1,2} = (i\beta, -i\alpha \mp 1). \tag{15}$$

They define the similarity transformation in which the map is diagonal, and altogether we obtain

$$S\ M\ S = \begin{pmatrix} r \cdot e^{+i\mu} & 0 \\ 0 & r \cdot e^{-i\mu} \end{pmatrix} \tag{16}$$

where

$$S^{-1} = \begin{pmatrix} i\beta & i\beta \\ -1 - i\alpha & 1 - i\alpha \end{pmatrix} \quad S = \begin{pmatrix} (1 - i\alpha)/2i\beta & -1/2 \\ (1 + i\alpha)/2i\beta & +1/2 \end{pmatrix} \tag{17}$$

Note that in the case of $T \geq 2\sqrt{D}$, the transformation matrices are purely real since all Twiss parameters are purely imaginary. In the case $\beta = 0$, we have $T/s\sqrt{D} = (a + d)/2\sqrt{ad} > 1$ because geometric means never exceed arithmetic means. This entails that one the eigenvectors is greater than 1 and hence the motion is unstable and thus not of primary interest for accelerator physics.

Similar to the previous section, the total matrix after the similarity transformation will consist of the pairs of the form $r \cdot e^{\pm i\mu}$ on the diagonal.

6 Chromaticities and Parameter Tune Shifts

After having outlined the algebra to compute the tune and dampings in detail, we now show how to compute the dependence of these quantities on system

parameters. We begin by restating that after the fixed point computation, the map does not depend on the parameters alone anymore. Alternatively, this can be interpreted as the matrix elements being dependent on parameters.

In this view, the matrix elements themselves now become equivalence classes, each containing the value of the element and its derivatives with respect to the parameters. In particular, the two-by-two sub-blocks of the last section now become matrices of equivalence classes

$$M = \begin{pmatrix} [a]_{n-1} & [b]_{n-1} \\ [c]_{n-1} & [d]_{n-1} \end{pmatrix} \tag{18}$$

Note that one order is lost in the process since a was a first derivative and so its $n-1$st derivatives are certain nth derivatives of the original map $\mathcal{M}$.

As advertised in the introduction, the computation of the parameter dependence of the tunes is now very straightforward and almost anti-climactic: following standard DA practice, we just replace all real operations for the computation of the tunes and dampings by the corresponding DA operations.

In case the motion is decoupled, i.e. the map is already in two-by-two block form, this merely involves the computation of the class of μ from the determinant and trace; in particular, we obtain

$$[\mu]_{m-1} = \text{sign}(b)\ \text{acos}\left(\frac{[a]_{m-1} + [b]_{m-1}}{2([a]_{m-1} \cdot [d]_{m-1} - [b]_{m-1} \cdot [c]_{m-1})}\right) \tag{19}$$

So like in most DA operations, standard formulas get simply replaced by their corresponding ones in the proper equivalence classes. The differential algebra software to manipulate the classes in COSY INFINITY [14, 15, 16] can readily perform all these operations, and only the inverse cosine required a little care.

Altogether, the algorithm is very direct and orders of magnitude more efficient than the normal form algorithm, even in its more efficient DA representation [13].

In the case of coupled motion, conceptually the strategy is the same. In addition one now blindly replaces all operations in the whole eigenvalue package by the corresponding ones in DA. Since for low dimensions, good eigenvalue and eigenvector algorithms are very efficient, this again results in an algorithm that outperforms the normal form approach significantly. In this context it is worthwhile to note that for symplectic systems, it is possible to compute the Eigenvalues directly as solutions of third order equations for up to three phase

space pairs. This was discovered by Neri [6] and would allow a more direct DA computation of the parameter dependences of the tunes for the symplectic case.

The direct computation of tune parameter dependences on tune shifts have been implemented in COSY INFINITY [14, 15, 16]. Because of the direct availability of the differential algebraic data type in the object oriented language of COSY INFINITY, the implementation is very straightforward. Clearly it could also be done in a FORTRAN environment by using the FORTRAN precompiler [17] or even direct calls to DA libraries, however at considerable loss of the ease of implementation.

The computation of parameter dependences in COSY INFINITY have been checked in two ways. Firstly, the values were compared with the ones obtained using the DA normal form algorithm [13]. Not surprisingly, agreement to machine precision was obtained.

As an independent test, we compared the results with numerically computed tune shifts using the code DIMAD [18]. To this end, a version of DIMAD that allows DA-based map extraction [18] was used to compute the map of the Saskatoon EROS ring; the map was read by COSY INFINITY and analyzed. Remarkably good agreement was obtained even for some higher order tune dependences, owing to the careful implementation of the numerical methods in DIMAD.

7 The Correction of Chromaticities

In this section we would like to illustrate an immediate and useful application of the algorithm outlined in the previous sections, namely the correction of chromaticities using system parameters. To this end, we write the v tunes in terms of the system parameters:

$$\vec{\mu} =_{n-1} \mathcal{T}(\vec{\delta}) \tag{20}$$

The map $\mathcal{T}$ contains a constant part, the linear tunes, as well as nonlinear parts, and the algorithm of the last section allowed us to compute the class $[\mathcal{T}]_{m-1}$ of $\mathcal{T}$.

We now split the parameters into the energy deviation δ_k and the true system parameters. For the further discussion, we are only interested in the case of v true system parameters, i.e. one for each phase space pair. Furthermore, we choose the parameters such that they do not produce tune shifts by themselves, but only in connection with energy deviations. This can for example be

achieved by using the strengths of v suitably chosen hexapoles as parameters. Quadrupoles strengths are not useful because they produce tune shifts even without δ_k since they obviously affect the linear tune. In this case, the tune equations reduce to

$$\vec{\mu} = \vec{\mu}_0 + \delta_k \cdot \vec{c} + \delta_k \cdot \mathcal{S}(\vec{\delta}) \tag{21}$$

where $\mathcal{S}$ is a nonlinear map. To correct the chromaticities, i.e. make the tune independent of δ_k, now requires to satisfy

$$\vec{c} + \mathcal{S}(\vec{\delta}) = 0 \tag{22}$$

which can be obtained by choosing

$$\vec{\delta} = \mathcal{S}^{-1}(-\vec{c}) \tag{23}$$

if the inverse exists. From $\mathcal{S}^{-1}$ we now pass to its equivalence classes and use the inversion algorithm of section 2. This yields the classes $[\vec{delta}]_{n-1}$ and hence the Taylor expansion of the strengths of v suitably chosen elements to correct the chromaticities. Using these Taylor expansion, an approximate value for $\vec{\delta}$ can be computed. Obviously the missing terms scale with the nth power of $\vec{\delta}$, so iterating the procedure yields very fast convergence, requiring only very few steps in practice.

8 Conclusion

In this paper, we have presented an algorithm that allows the direct computation of parameter dependent tune shifts including chromaticities to arbitrary order. The algorithm is very general, requiring only certain properties of linear maps. They are that the determinant has to be positive and that the eigenvalues of the linear map are mutually distinct and distinct from 1, which conditions are of course readily met by all accelerator lattices. It has the flavor of all DA algorithms in the sense that it boils down to replacing suitable real number operations by DA operations and automatically works to arbitrary order.

The algorithm has been implemented in the code COSY INFINITY [15, 16, 14]. It has been compared against the results of the more involved DA normal form algorithm [13] which can also compute amplitude dependent tune shifts as well as against conventional numerical techniques.

References

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

[2] A. J. Dragt and J. M. Finn. Normal form for mirror machine Hamiltonians. *Journal of Mathematical Physics*, 20(12):2649, 1979.

[3] A. Bazzani, P. Mazzanti, G. Servizi, and G. Turchetti. Normal forms for Hamiltonian maps and nonlinear effects in a particle accelerator. *Il Nuovo Cimento*, 102 B, N.1:51, 1988.

[4] G.Hori. *Publications of the Astronomical Society of Japan*, 18:287, 1966.

[5] A. Deprit. *Celestial Mechanics*, 1:12, 1969.

[6] Filippo Neri. Private communication.

[7] Etienne Forest. Private communication.

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

[9] M. Berz. Arbitrary order description of arbitrary particle optical systems. *Nuclear Instruments and Methods*, A298:426, 1990.

[10] M. Berz. Differential algebraic description and analysis of trajectories in vacuum electronic devices including spacecharge effects. *IEEE Transactions on Electron Devices*, 35-11:2002, 1988.

[11] M. Berz. Differential algebra - a new tool. In *Proceedings of the 1989 IEEE Particle Accelerator Conference*, page 1419. IEEE, 1989.

[12] E. Forest, M. Berz, and J. Irwin. Normal form methods for complicated periodic systems: A complete solution using Differential algebra and Lie operators. *Particle Accelerators*, 24:91, 1989.

[13] M. Berz. Differential algebraic formulation of normal form theory. 1991.

[14] M. Berz. COSY INFINITY Version 3 reference manual. Technical Report MSUCL-751, National Superconducting Cyclotron Laboratory, Michigan State University, East Lansing, MI 48824, 1990.

[15] M. Berz. Computational aspects of design and simulation: COSY INFINITY. *Nuclear Instruments and Methods*, A298:473, 1990.

[16] M. Berz. COSY INFINITY, an arbitrary order general purpose optics code. *Computer Codes and the Linear Accelerator Community*, Los Alamos LA-11857-C:137, 1990.

[17] M. Berz. Differential algebra precompiler version 3 reference manual. Technical Report MSUCL-755, Michigan State University, East Lansing, MI 48824, 1990.

[18] Roger Servranckx. Private communication.