High-Order Beam Optics*
(An Overview)

Edward A. Heighway
Los Alamos National Laboratory, Los Alamos, NM 87545

Introduction

Beam-transport codes have been around for as long as thirty years and high-order codes, second-order at least, for close to twenty years. Before this period of design-code development, there was considerable high-order treatment, but it was almost entirely analytical. History has a way of repeating itself, and the current excitement in the field of high-order optics is based on the application of Lie algebra and the so-called differential algebra to beam-transport codes, both of which are highly analytical in foundation. I will describe some of the main design tools available today, giving a little of their history, and will conclude by trying to convey some of the excitement in the field through a brief description of Lie and differential algebra.

Ancient and Not So Ancient History

High-order optics has been around a long time and Newton in his 1687 masterpiece *Principia* described a fairly advanced light optical spectrograph (Fig. 1). If Newton was with us today, no doubt he would be reveling in differential algebra.

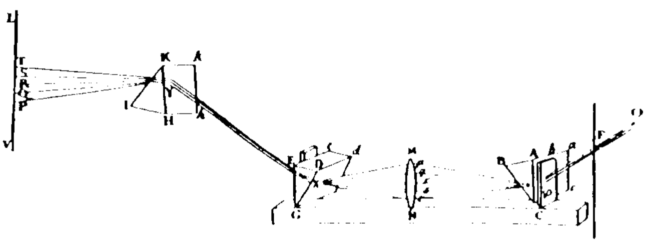

Fig. 1. Early optical spectograph designed by Newton.

To most of us in the community, high order is synonymous with aberrations, that is, deviations from the linear or paraxial approximation. These have been known and indeed understood for a long time. Electron microscopy was an early and continuing application for high-order theory, and as early as 1936, Scherzer[1] had completed an analytical description of third-order optics for such devices and later developed the theory to correct them using octupole elements.[2] This theory has stood the test of time and only recently have the needs of high-resolution electron lithography pressed for a more advanced understanding. Even matrix theory as applied to optics is not new; Cotte, in 1938,** had developed a first-order matrix theory. There was much high-order work on early synchrotron and cyclotron theory to describe their nonlinearities,[3,4] and special devices were often calculated to very high order. A typical example of this is the work of Lee-Whiting[5] on the Pi-Root-2 Beta Ray Spectrometer where the dynamics in the horizontal plane was calculated to sixth order.

Why Do We Need High Order?

The electron microscopists' problems carried over directly from light optics in the form of image aberrations. Figure 2 shows the rays typical of spherical or aperture aberrations in which the outer rays are overfocused in comparison with the paraxial rays. The point to note is that the deviation or aberration of these outer rays from the design focal spot in the image plane is proportional to the cube of the ray distance from the axis in the lens and so the aberrations are third order. Third-order theory has served the community well and only recently has the need for fifth-order theory become necessary as technology such as electron lithography advances into the micron resolution regime.

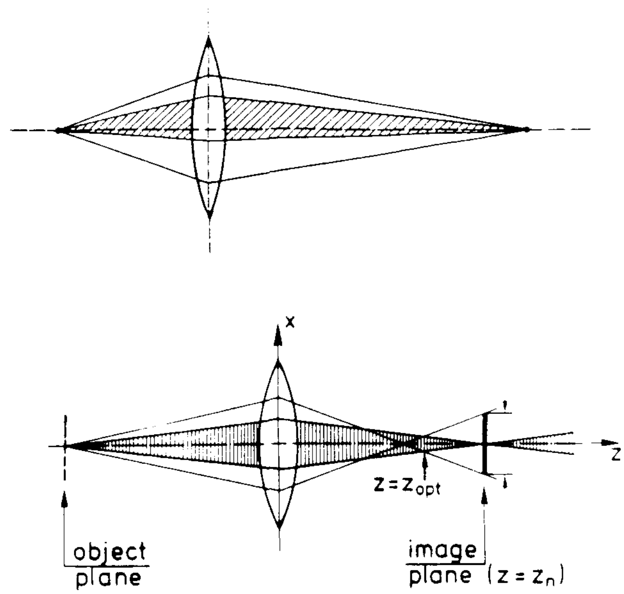

Fig. 2. Typical spherical aberrations in a simple lens system showing how off-axis rays are overfocused.

Beam-transport design to first order has been adequate for many years. If we look at a simple achromatic bend in wide use (Fig. 3) and just consider second-order theory, we see that the dispersed rays no longer come back on axis. The correcting quadrupole overcorrects the low-energy ray and undercorrects the high-energy ray so that both rays exit the magnet to the same side of the central ray. This device is, in fact, the analog of Newton's spectrograph, and one suspects he understood the effect. Karl Brown has solved this problem by developing second-order theory and, in particular, inventing the second-order achromat.[6]

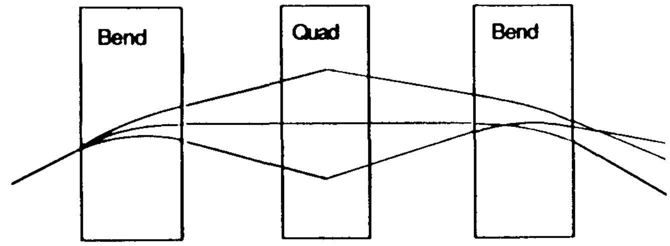

Fig. 3. Simple achromatic bend system of two dipoles and a central quadrupole showing that to second order achromaticity is lost.

*Work supported by Los Alamos National Laboratory Institutional Supporting Research, under the auspices of the United States Department of Energy.

**Private communication with Hermann Wollnik.

I have mentioned the need for extremely high quality electron-beam images for lithography, and Fig. 4 shows an image calculated to fifth order showing the aberration lobes in the image. Another application for such advanced theory is the production of extremely parallel particle beams needed in the Strategic Defense Initiative Program for beam propagation in space.

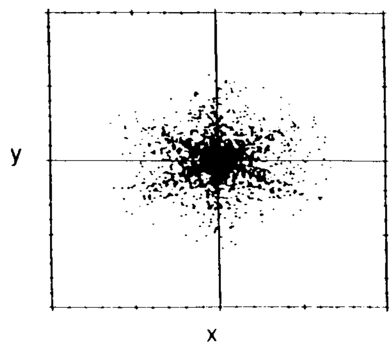

Fig. 4. Fifth-order aberrated image showing typical lobes in the image.

TRANSPORT to MAD

North American high-order design-code development (as opposed to analysis) was pioneered by Karl Brown in the code TRANSPORT. In the fall of 1958, while on a visit to Orsay, Brown, innocent of the knowledge that it couldn't be done, invented second-order matrix theory. Brown had been influenced by S. Penner of the National Bureau of Standards and the early use of matrix methods to model the beam dynamics in the alternating gradient synchrotron (AGS); he also made heavy use of J. Streib's work on field integrals. The formal theory was developed at the Stanford Linear Accelerator Center in the summer of 1959 and resulted in the publication of SLAC-75.[7] H. Butler, S. Howry, and C. Moore developed the code in ALGOL for a Burrows machine in the early 1960s; this code was translated to FORTRAN by S. Kowalski in 1964-65. The manual for TRANSPORT was published as SLAC-91,[8] and rapidly became the beamline designer's bible.

David Carey of Fermi Lab became a key collaborator on TRANSPORT in late 1969, early 1970. Carey had written the code RABBIT to analyze the useful aperture of the Fermi Lab quadrupoles. To make ray tracing more accessible, he then wrote an independent and more general code (hare, tortoise) TURTLE,[9] the successful ray-tracing partner to TRANSPORT. Carey is now the primary focus for maintenance and development of TRANSPORT and TURTLE and successfully made the extension of TRANSPORT to third order in 1983-84. This was extended in 1986-88 to include a combined function dipole and fringe fields.[10] Current features of TRANSPORT are summarized later in Table I.

F. Christoph Iselin at CERN became a major contributor to beam-transport codes starting in 1972 when he collaborated on developing a standard portable version of TRANSPORT, which led to the joint publication of Revision 2 of SLAC-91 by CERN, FNAL, and SLAC in 1977.[11] Iselin had been working on magnet design and began working on beam-transport codes when Brown visited CERN in 1972. He and Brown developed a special version of the code DECAY TURTLE at that time. Based on an earlier code, HALO, Iselin then developed his own code, MAD (Methodical Accelerator Design), to improve the user interface for designers. This interface was a great enough improvement over previous notations that the so-called MAD notation was suggested at a SLAC workshop in 1984 and later adopted at the 1984 Snowmass Conference as the standard for beam-transport design codes. This notation has been adopted and is available in the code MAD, itself, as well as in TRANSPORT, GIOS, and MARYLIE. However, MAD is a powerful design tool and is the primary design tool for LEP. It is almost machine independent and will operate in a UNIX environment on IBM, CDC, or CRAY. It is network oriented and will run on Apollo, VAX, or SUN workstations and IBM PC's. It has a distinctive circular machine flavor with the ability to calculate tunes, closed orbits and corrections, chromaticities, resonances, and radiation loss. J. Niederer at Brookhaven National Laboratory has become a major collaborator on MAD development[12] with emphasis on operation in the modern computer environment and interface to other codes in the areas of controls, graphics, and drafting. An important development on the books, but taking second priority to LEP applications, is an extension of MAD to third order. An early version of the Lie algebra formalism already has been included as a subset of the MAD library; this extension should benefit from the recent developments by Dragt on MARYLIE.

RAYTRACE

Spectrometer designers, such as H. Enge at MIT, worried about high order long before design codes were fashionable. Enge's early technique was to actually lay out the system, often full scale, on a drawing and trace out the system optics by plotting each ray geometrically on the drawing. This procedure worked well and led to many extremely successful spectrometer designs. The design code RAYTRACE,[13-15] a computer extension of this technique, was originally written by J. Spencer as an undergraduate thesis project at MIT under Enge and later actively developed by Enge and Kowalski at MIT. This method should not be underrated for high-order design; RAYTRACE has been extended by Spencer, while he was working on the high-resolution spectrometer at Los Alamos, to include fifth-order integration. One hundred rays can be traced through up to two hundred elements. Many new elements were added by Kowalski; the elements treated include a homogeneous and combined function dipole (with transverse derivatives in the field up to fourth order allowed), cylindrical electrostatic deflector, any multipole magnet up to dodecapole including mixed multipoles, magnetic solenoid, velocity selector, thin quadrupole lens, and drift length. Coordinate translations and rotations between elements are possible. Two special elements, a collimator and spectrometer multipole corrector element, are available. During the 1970s, RAYTRACE was adapted at LANL, primarily by M. Klein, R. Christian, and A. Thiessen, to include optimization. This version, called MOTER,[16] is still in use with developments still continuing for spectrometer design at Michigan State University and the Continuous Electron Beam Accelerator Facility (CEBAF).

PATH

The code PATH,[17] which is a PARMILA[18] and TURTLE Hybrid, is not widely known or distributed but is nonetheless useful in that it combines some third-order elements with 3-D space charge. The code was originally developed at Los Alamos by J. Farrell and D. Rosthoi and the most recent work has been at McDonnell Douglas Corporation, St. Louis, by R. Kashuba and R. Schmitt. The code traces particle distributions through discrete elements that are modelled by transfer matrices. A very useful feature of the code is the BEAMGEN routine, which generates the particle distributions used in the tracking. The distribution can be selected from 2-D, 4-D, 6-D, Gaussian, Lorentian, or Binomial.

Kashuba and Schmitt[19] have added an improved nonlinear model of a quadrupole with fringe fields, incorporated the RAYTRACE ray generation and matrix-element calculation routines, and added an optimization code ORACLE, developed at Los Alamos by T. Mottershead

Table I. Current features of TRANSPORT, GIOS, MARYLIE, and COSY

	TRANSPORT	GIOS	MARYLIE	COSY
Coordinates	x, x', y, y', l ,delta	x, x', y, y', l ,delta delta(mass)	x, x', y, y', l ,delta	x, x', y, y', l ,delta delta(mass)
Space Charge	1st order, 3-D homogeneous	3rd order, 3-D inhomogeneous	3rd order, 2-D, CHARLIE self-consistent map	2-D gaussian beamlets
Fitting	Newton Marquand-Levinberg	Simplex Conjugate gradient	Inverse Interpolation Quadratic Form	Quadratic Form Conjugate gradient
Elements	Magnetic	Magnetic, Electric	Magnetic	Magnetic, Electric
Dipoles Fringe Field	3rd Order n1, n2,n3 variable pole face rotation, curvature	3rd Order n1, n2,n3 variable pole face rotation, curvature	3rd Order n1, n2,n3 variable pole face rotation	5th Order n1, n2,n3 variable pole face rotation, curvature
4-pole Fringe Field	3rd order none	3rd order 3rd order	5th order 5th order	5th order 5th order
6-Pole Fringe Field	3rd order none	3rd order x 3rd, y 2nd order	3rd order, point 5th order 3rd order	5th order 5th order
8-Pole Fringe Field	3rd order none	3rd order x 3rd, y 2nd order	3rd order none	5th order 5th order
10-Pole Fringe Field				5th order 5th order
12-Pole Fringe Field				5th order 5th order
Multipole Fringe Field		4+6+8 3rd order		
Input	Type Codes and MAD Blocks and Loops	Memorizable Algebraic relations Blocks and Loops	Memorizable Algebraic relations Block and Loops	Memorizable Algebraic relations Blocks and Loops
Output	System Description Printer Plots	System Description Printer Plots System Layout	System Description	System Description Printer Plots System Layout
Matrices	Standard Aberrations Transfer Matrices	Normalized Aberrations Transfer Matrices	Lie Polynomials Transfer Matrices	Normalized Aberrations Transfer Matrices
Particle Distributions	TURTLE Printer plots	3rd order Printer plots Graphics	Scatter plots	3rd order Printer plots Graphics

and K. Overley.[20] I include Fig. 5 to show the importance of intercommunication among workers in the field to the development of PATH through the contributions made from PARMILA, TRANSPORT, TURTLE, RAYTRACE, and ORACLE.

Elements supported are the usual transport elements — dipoles, multipoles, octupoles; but because of the code origins in PARMILA, there are several accelerator elements — buncher, accelerator column, and DC and rf accelerating gaps. At present only quadrupoles and octupoles are modeled to third order, but, in principle, all the third-order upgrades made by Carey to TRANSPORT could be included.

MARYLIE

The application of Lie algebra to beam-transport theory began in 1979 when A. Dragt, University of Maryland, came to work with R. Cooper on the Proton Storage Ring at Los Alamos. The basic concepts were taken from Dragt's experience in group theoretical analysis and rapidly developed for application to charged particle beam

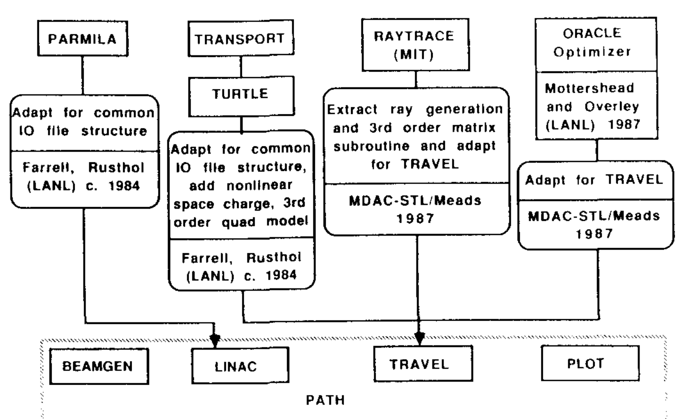

Fig. 5. PATH is a good example of how many diverse developments in several codes have been combined into a single design code.

transport. At the 1981 Fermi Lab Summer School, Dragt had much of the theory on a solid base. By 1983, the first workable Lie algebraic code was available and was already being extended to include misalignments. By 1985, Dragt had developed his code (now named MARYLIE) to rival TRANSPORT in reliable operation and treating all elements to third order. MARYLIE was extended in 1987 to add a fitting routine that changed the code from primarily a powerful simulation tool into a more versatile design tool; this year the optimizer ORACLE[20] is being added to include more complex optimizations. Two other important developments are the addition of 2-D nonlinear space charge in a special version of the code CHARLIE[21] as well as treatment of several elements to fifth order. CHARLIE is the first design code to treat nonlinear space charge in an analytic way.

Lie Algebra

I will digress a little to provide some insight into Lie algebra because, for most of us, it is a new and unfamiliar field but an extremely important development. Dragt has published many treatises[22, 23] and to convert these to a few lines in this paper is presumptuous but nonetheless worthwhile.

Just to refresh our memories, we should remember the matrix representation. In two dimensions we have

$$z_f = M \cdot z_0$$

or

$$\begin{pmatrix} x_f \\ p_f \end{pmatrix} = \begin{pmatrix} x/x & x/p \\ p/x & p/p \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ p_0 \end{pmatrix} .$$

Adding second-order terms, we have x_f depending on x^2, xp, and p^2; and to third order, x could be expressed as a third-order polynomial

$$x_f = ax_0^3 + bx_0^2p_0 + cx_0p_0^2 + dp_0^3 \quad . \tag{1}$$

To start on the Lie algebra, if z_0 represents the phase-space coordinates of a particle, the operator M maps these to the new coordinates z_f by $z_f = \mathrm{M}\, z_0$ similar in form to the matrix transformation. If $g(z)$ is a function of the phase-space coordinates and $:f:$ is defined as a Lie operator, then the operation $:f:g(z)$ is defined by the rule $:f:g(z) = \{f,g\}$, which is the Poisson bracket

$$\{f,g\} \equiv \frac{\partial f}{\partial x}\frac{\partial g}{\partial p} - \frac{\partial f}{\partial p}\frac{\partial g}{\partial x} \quad .$$

For each Lie operator there is a corresponding Lie transform defined by the exponential $e^{:f:}$, which can be expanded as the series

$$e^{:f:} \equiv \sum_{n=0}^{\infty} \frac{:f:^n}{n!} = 1 + :f: + \frac{1}{2}:f:^2 + \ldots \quad .$$

Thus, the application of the transform to $g(z)$ has the form

$$e^{:f:}g = g + \{f,g\} + \frac{1}{2}\{f,\{f,g\}\} + \ldots \quad .$$

This equation will still seem obscure to most of us, but the application to a one-dimensional canonical pair $z = (x,p)$ makes things more familiar.

$$e^{:f:}x = x - \frac{\partial f}{\partial p} + \frac{1}{2}\frac{\partial f}{\partial p}\frac{\partial^2 f}{\partial p\,\partial x} - \frac{1}{2}\frac{\partial f}{\partial x}\frac{\partial^2 f}{\partial p^2} + \ldots$$

$$e^{:f:}p = p + \frac{\partial f}{\partial p} + \frac{1}{2}\frac{\partial f}{\partial p}\frac{\partial^2 f}{\partial p\,\partial x} - \frac{1}{2}\frac{\partial f}{\partial p}\frac{\partial^2 f}{\partial x^2} + \ldots$$

The most general fourth-order homogeneous polynomial function f_4 is then

$$f_4(x,p) = kx^4 + lx^3p + mx^2p^2 + nxp^3 + op^4 \ .$$

The corresponding map to third order is then

$$e^{:f_4:}x = x + \{f_4,x\} = x - \frac{\partial f_4}{\partial p} \quad ,$$

and

$$e^{:f_4:}p = p + \{f_4,p\} = p + \frac{\partial f_4}{\partial p} \quad .$$

By expanding, we will obtain a polynomial equation

$$x_f = 4kx^3 + 3lx^2p + 2mxp^2 + np^3 \quad ,$$

which is very similar to the form of Eq. (1) that we are familiar with from matrix theory.

In both the matrix approach and the Lie algebra approach, we have a polynomial expression linking the initial and final coordinates. The difference is that in the Lie algebra approach, the polynomials are formed by differentiating the generating functions f. It is clear that the number of coefficients required to totally define the mapping, especially for a multidimensional space, will be much less in the Lie algebra than in the matrix approach. The key is that Lie algebra uses the symmetry of group theory to greatly simplify the representation of the map. There is another advantage in that when a designer using MARYLIE requests to zero some coefficient of a generating function, he is making a request at a much higher level and can, as a result, zero an entire family of aberrations (specific high-order terms). The specific features of MARYLIE are described in Table I.

GIOS, COSY

In Europe, there was a parallel and independent development of matrix theory for spectrometer and spectrograph design. From 1959 to 1964, Hermann Wollnik (University of Giessen) developed a second-order code SPECTRO paralleling the work of Brown. The initial hard-edge, second-order code was extended later to include proper fringe fields. Indeed, a consistent feature of Wollinik's work is the proper and careful treatment of fringe fields. SPECTRO evolved into a third-order code GIOS,[24] which was operational with two-dimensional space charge in 1976. Although initially operating with a limited spectrum of elements, this code was the first solid third-order code available to the community. Wollnik's specialty is mass spectrometer design and in the early 1980s, he developed the separate code BEAMTRACE,[25, 26] which has much in common with Kowalski's RAYTRACE. The code GIOS is described in Table I. Martin Berz has continued work on the GIOS code with an extension to fifth order in the new code COSY, which is now operational but still at the testing and verification stage. COSY was implemented using the tools developed by Berz in his code HAMILTON.[27] The features of COSY are also described in Table I.

Differential Algebra

Although the concept has been around in mathematical circles for some time, the recent burst of activity, and the name differential algebra, are products of work by Berz.[28, 29] This work began when Berz was at Giessen with Wollnik and was inspired primarily by the need to calculate functional derivatives for high-order matrix elements in GIOS and COSY. Berz had already developed a compiler to translate the input data files for GIOS as well as the code HAMILTON to calculate the matrix coefficients. Therefore, the extension into differential algebra was a natural one. Much of the basic work was done by Berz at Giessen and then at Los Alamos, where he wrote his differential algebra compiler, which he later applied to a specific design code during his period with the Berkeley Superconducting Super Collider Central Design Group and where the association with Etienne Forest was particularly fruitful.

As was the case for Lie algebra, it is impossible to give an adequate description of this concept in a short overview paper, but I think it is useful to try to explain the main thrust.

In the most elementary case of a one-dimensional space and considering only first derivatives, a point in space can be described by the coordinate pair $(y, y') = (y, dy/dx)$ where y is a function of x. Differential algebra writes down rules for combining such pairs so that

$$(a, b) + (c, d) = (a+c, b+d) \quad ,$$
$$t \cdot (a, b) = (t \cdot a, t \cdot b) \quad ,$$
$$(a, b) \cdot (c, d) = (a \cdot c, a \cdot c, a \cdot d + b \cdot c) \quad ,$$

and

$$(a, b)^{-1} = (1/a, -b/a^2) \quad .$$

These rules can be derived rigorously and are similar to the simple rules developed for writing down the derivatives of polynomials, which we just accept.

Let us look at the function $f = x^2 + 1/x$ evaluated at $x = 2$. The function x itself (evaluated at $x = 2$) can be written as (2, 1). The function $1/x$, we can either write as (1/2, −1/4) by inserting the function value and its derivative value at $x = 2$, or we can write it as $(2, 1)^{-1}$ and, using the derivative rule, get the same answer.

To find $f = x^2 + 1/x$, we get (4, 4) + (1/2, −1/4) = (9/2, 15/4), which is the value and derivative value of f at $x = 2$, as required.

Thus, the kernel of differential algebra is that knowing the values and values of the derivatives of any functions at specific points in space, we are able to write down the values and values of the derivatives of any combination of those functions using simple arithmetic rules. This is clearly very attractive and ideal for computer application.

This example is, of course, trivial, but a second example is worth examining to see the extension to more complex spaces, in this case to two dimensions and second-order derivatives. The simple coordinate pair is now replaced by a six-element scalar comprising the values and all derivative values of the function up to second order, namely

$$(f, df/dx, df/dy, d^2f/dx^2, d^2f/dxdy, d^2f/dy^2) \quad .$$

Let us look at the function $g = y - 2x$ at $x = 1, y = 2$. The term y can be written as (2, 0, 1, 0, 0, 0); x can be written as (1, 1, 0, 0, 0, 0), so $2x$ is (2, 2, 0, 0, 0, 0), and $y - 2x$ is (2, 0, 1, 0, 0, 0) − (2, 2, 0, 0, 0, 0), which, using our differential algebra rules, becomes (0, −2, 1, 0, 0, 0), which is the six-element scalar defining the value and derivatives of $y - 2x$ at $x = 1$, $y = 2$, which is what we wanted.

Clearly, the rules required to extend this algebra to more dimensions and higher orders get more complex, but they nonetheless can be written down. Berz has fully developed this algebra and indeed developed the rules for the basic functional types common to beam dynamics, i.e., trigonometric, logarithmic, exponential, and hyperbolic functions. Even beyond this, Berz has written a FORTRAN library for these functions and a FORTRAN compiler DAFOR[30] that will convert existing FORTRAN code to use his differential algebra library.

In summary, what has differential algebra made available to us? The answer is that without clumsy numerical differentiation, derivatives of multidimensional functions (for example the typical six-dimensional functions we associate with particle-beam phase space) can be evaluated to any order to machine precision.

To illustrate what the method has achieved in application to a nonlinear problem, Fig. 6 shows the x, px phase-space diagram of modified Henon map calculated by Etienne Forest and Berz using a fifteenth-order expansion. The high-order, nonlinear, stable and unstable fixed points are clearly shown. This figure is identical to that generated by an exact ray-tracing calculation for the same map.

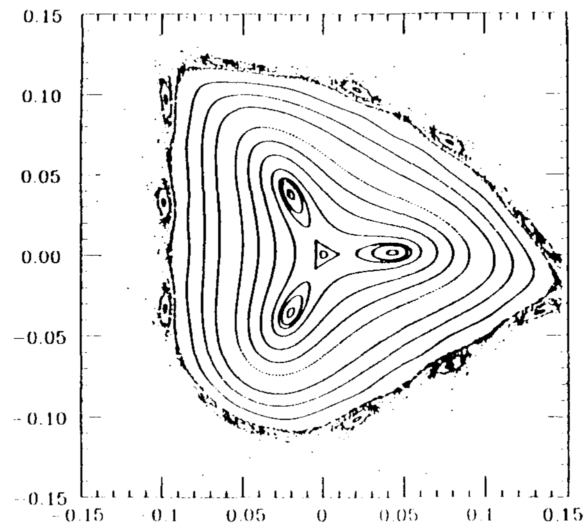

Fig. 6. A modified Henon nonlinear map calculated using 15th order differential algebra.

Differential algebra is already a serious contributor to beam dynamics and is the key to extension of design to very high order. The development from a laboratory curiosity to an operational code has been extremely rapid and is the most exciting development in beam transport and in dynamics for many years.

Summary

Figure 7 shows pictorially the history of beam-transport development since the 1930s. As one can see, the growth into high order has been exponential and, indeed, infinite order is just around the corner.

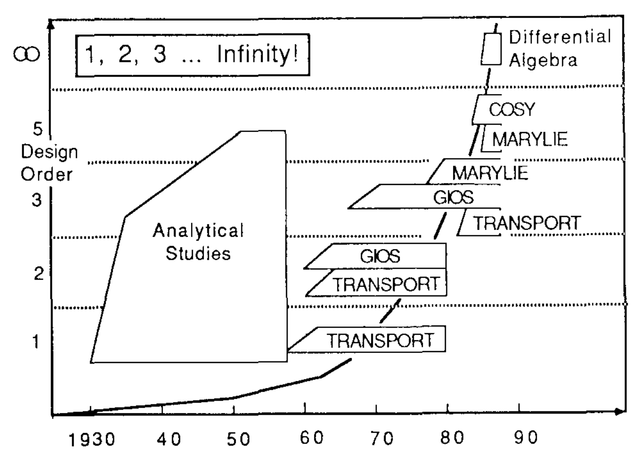

Fig. 7. An historical view of nonlinear code development showing the exponential growth in the field.

Although high order has been around for a long time, design codes of high order are a fairly recent development. This development is timely because the more advanced concepts now on the drawing boards require that high-order design capability. Lie algebra, with its savings in representation, and differential algebra with its ability to accurately evaluate high-order functions, are indeed a forbidding but extremely exciting combination.

Acknowledgments

I am indebted to Martin Berz, Karl Brown, Harold Butler, David Carey, Alex Dragt, Thomas Mottershead, James Niederer, James Spencer, and Hermann Wollnik for their invaluable contributions to this paper and take this opportunity to express my thanks to them.

References

1. O. Scherzer, Z. Phys. 101, 593 (1936).
2. O. Scherzer, Optik 10, 114 (1947).
3. L. Smith and A. Garren, "Orbit Dynamics in a Spiral Ridged Cyclotron," University of California Lawrence Radiation Laboratory report UCRL-8598 (1959).
4. H. L. Hagedoorn and N. F. Verster, "Orbits in an AVF Cyclotron," Nucl. Instr. and Meth. 18-19, 201 (1962).
5. G. E. Lee-Whiting and E. A. Taylor, "High Order Focusing in the Pi-Root-2 Beta Ray Spectrometer," Can. J. Phys. 35, 1-15 (1957).
6. K. L. Brown, "A Second Order Magnetic Optical Achromat," IEEE Trans. Nucl. Sci. 26 3490 (1979).
7. K. L. Brown, "First and Second Order Matrix Theory for Design of Beam Transport Systems," Stanford Linear Accelerator Center report SLAC-75 (1967).
8. K. L. Brown and S. Howry, "TRANSPORT/360," Stanford Linear Accelerator Center report SLAC-91 (1970).
9. D. C. Carey, "TURTLE, A Computer Program for Simulating Charged Particle Beam Transport Systems," Fermi National Accelerator Laboratory report NAL-64 (1978).
10. D. C. Carey, "New Features in TRANSPORT," Fermi National Accelerator Laboratory report FNAL TM-1064 (1981), "Third Order TRANSPORT with MAD Input," Fermi National Accelerator Laboratory report, 1988 (to be published); and "Optics of Charged Particle Beams," (Harwood Academic Publishers, NY, 1987).
11. K. L. Brown, F. Rothacker (SLAC), D.C. Carey (FNAL), and C. Iselin (CERN); Stanford Linear Accelerator Center reports SLAC-91; Fermi National Accelerator Laboratory report FNAL-91; and European Center for Nuclear Research report CERN 73-16 (1977).
12. F. C. Iselin, E. Keil, J. Niederer, and J-M. Veuillin, "The MAD Program, Version 6, Programmers Reference Manual," European Center for Nuclear Research report CERN/LEP-TH/87-60 (1987).
13. J. Spencer, "Design of a Magnetic Spectrograph," MIT Bachelor's Thesis (1964).
14. S. B. Kowalski and H. A. Enge, "The Ion-Optical Program RAYTRACE," Nucl. Instr. and Meth, A258, 407 (1987).
15. S. Kowalski and H.A. Enge, unpublished "RAYTRACE Users Manual," Dept. of Phys., MIT, (private communication, S. Kowalski, MIT) (1986).
16. W. Lysenko, unpublished Technical Note "MOTER Users Manual," Los Alamos National Laboratory, (private communication, W. Lysenko, LANL) (1987).
17. D. Rusthoi and J. Farrell, unpublished "PATH Users Manual," Los Alamos National Laboratory (private communication, D. Rusthoi, LANL).
18. G. P. Boicourt, "PARMILA, an Introduction," Linear Accelerator and Beam Optics Codes Workshop, San Diego, AIP Proc. 177, 22 (1988).
19. R. J. Kashuba, R. J. Schmitt, and P. F. Meads, Jr., "Status of the PATH Magnetic Optics Code," Linear Accelerator and Beam Optics Codes Workshop, San Diego, AIP Proc. 177, 61-76 (1988).
20. K. Overley, unpublished Technical Note "ORACLE: A Quadratic Form Fitting Algorithm for Unconstrained Minimization," Los Alamos National Laboratory, (private communication, T. Mottershead, LANL, 1987).
21. R. D. Ryne, "Lie Algebraic Treatment of Space Charge," Ph.D. thesis, Dept. of Phys., University of Maryland (1987).
22. A. J. Dragt, "Lie Algebraic Methods for Charged Particle Optics," Linear Accelerator and Beam Optics Codes Workshop, San Diego, AIP Proc. 177, 267-270 (1988).
23. A. J. Dragt, R. D. Ryne, L. M. Healy, F. Neri, D. Douglas, and E. Forest, "MARYLIE 3.0 User's Manual," University of Maryland report (to be published).
24. H. Wollnik, J. Brezina, M. Berz, and W. Wendel, Proc. Seventh Conf. Atomic Mass and Fund. Const., GSI report THD–26, 679 (September 1984).
25. H. Wollnik, J. Brezina, and M. Berz, "GIOS-BEAMTRACE — A Program Package to Determine Optical Properties of Intense Ion Beams," Nucl. Instr. and Meth. A258, 408 (1987).
26. H. Wollnik, "Optics of Charged Particles," (Academic Press, Orlando, Florida 1987).
27. M. Berz and H. Wollnik, "The Program HAMILTON for the Analytic Solution of the Equations of Motion through Fifth Order," Nucl. Instr. and Meth., A258, 364 (1987).
28. M. Berz, "A Survey of Differential Algebra and its use for Extraction of Maps to Arbitrary Order," SSC Central design Group, LBL (1988).
29. M. Berz, "Treatment of Beam Dynamics to Very High Orders including Space Charge," Linear Accelerator and Beam Optics Codes Workshop, San Diego, AIP Proc. 177, 281–306 (1988).
30. M. Berz, unpublished Technical Note "The Differential Compiler DAFOR," Los Alamos National Laboratory, (private communication, W. Lysenko, LANL) (1987)