CYBERNETICS AND PHYSICS, VOL. 3, NO. 3, 2014 , 93–98

PARAMETRIC AND NUMERICAL METHODS FOR PARTICLE TRANSPORT SYSTEMS OPTIMIZATION

Serge Andrianov
Saint Petersburg State University
Russia
sandrianov@yandex.ru

Nikolai Edamenko
Saint Petersburg State University
Russia
nse47@yandex.ru

Abstract
In this paper we discuss some methods of parametric control of the correctness for numerical optimization of dynamics of a beam of charged particles.

Key words
Particle beam accelerator, control, parametric and numerical optimization, exact and approximate integration.

1 Introduction

Over the last years, different effective algorithms have used for solving of accelerator optimization problems. In general the most of approaches and methods are based on numerical methods for simulation of the dynamics as well as the implementation of the optimization process (see, for example, [Andrianov, 2004a; Andrianov, 2004b; Makino, Berz, 2014; Proc. BDO'2008, 2008; Streichert *et al.*, 2012; Gong and Chao, 2012]).

Any numerical calculations (both for dynamics and for implementing the optimization process) can lead to additional errors, which may substantially distort of the results. Therefore, it is necessary to ensure the implementation of additional check as the accuracy of calculations of the dynamics as well as for the optimization process itself.

First of all, it should be noted the problem of simulating long-term evolution of the particle beams, as well as the study of the influence of various effects (fringe fields, distortion parameters controls, etc.) on the features of the behavior of the beam. In particular, the problems of modeling the dynamics of polarized beams (for example, to measure the electric dipole element (EDM) [Lehrach, 2012]). Particular attention should be paid to non-linear parasitic effects, which can significantly distort the quality of the control system.

In this paper, we consider some methods of quality control for the simulation process. The proposed methods are based on the matrix formalism for Lie algebraic methods [Andrianov, 2004a; Andrianov, 1997] that allows efficient use both numerical as well as symbolic methods (for example, using the package Mathematica).

2 Formulation of the Problem

In this paper we consider some problems of modeling of multicomponent magnetic and/or electrostatic accelerators. It is known that such objects are complex and consist of a large number (up to tens of thousands) of different controls. In general case the corresponding devices generate controlling electromagnetic fields, which ensure the desired behavior of the beam. Here we mean different kinds of requirements for the particle beam. Since each control element is responsible for different effects, then their combination may result to complex effects, including to undesirable behavior of the beam. It should be noted that the various components of the control field can lead to a variety of effects in the behavior of the beam, and it is often difficult to understand exactly what kind of characteristics of the control field create certain effects.

It is these problems make it necessary to use not only the correct models for the control fields, but and to identify possible undesired effects with the purpose to their further correction or (full/partial) compensation. It should be noted that usage of only the numerical methods for solving the corresponding evolution equations cannot effectively allow such a study. It is therefore, in this paper we focuse not only to the symbolic construction of the corresponding the solutions of evolution equations, but also on construction of exact or approximate invariants. In general case the corresponding control devices generate control electromagnetic fields, which ensure the desired behavior of the beam. Here we mean different kinds of requirements to the beam which can be described using different conditions in the form of so called invariants. Indeed, if in the process of modeling we are will observe a significant distortion of the values of these invariants, then this indicates that either our computational methods do work incorrectly or we are not taking into account some other real effects.

In this paper, we propose testing the computational procedures using the exact and/or approximate (up to the considered order of nonlinearity) invariants [Andrianov, 2001a; Andrianov, 2001b]. It should also be noted that the special visualization instruments can play the importance role in similar investigations, in particular when using appropriate tools for 2D- and 3D-animation of computed data.

3 A Short Introduction to the Matrix Formalism for Lie Algebraic Tools

In this section, we discuss some methods for solving nonlinear dynamical systems that allow a natural application of the methods of perturbation theory.

3.1 The Evolution Operator as Instrument of Description of Dynamical Systems

It is known that for any dynamical system we can write

$$\mathbf{X}(t) = \mathcal{M}(\mathbf{U}; t|t_0) \circ \mathbf{X}(t_0), \tag{1}$$

where $\mathbf{X}(t)$ – phase vector describing the state of the dynamical system (in our case of the beam) at the time of s, $\mathcal{M}(\mathbf{U}; s|s_0)$ – the evolution operator of the dynamical system the system under study (in general, nonlinear), $\mathbf{U}$ – vector of control "parameters", s, s_0 – the current and initial values of the independent variable (e.g. distance, measured along the reference trajectory).

In general, the dimension of $\mathbf{U}$ may be infinite (for example, in the case of controlling using the electromagnetic field). In the case of the different problems (electrodynamics, biodynamics, and others) it is possible also to enter special objects, such as state matrices $\mathbb{X}^N = \{\mathbf{X}^1, \ldots, \mathbf{X}^N\}$, $N >> 1$, the distribution function $f(\mathbf{X}, t)$, etc. It is known that in many number of problems we need consider the interparticle interaction. In this case, the evolution operator in equation (1) will depend on the phase vector $\mathbf{X}(s)$ or $\mathbb{X}^N(s)$.

All of this leads to the need for adequate methods for describing the corresponding dynamical systems. Considering that the dynamic system control is defined by the equation (1), its solution can be written using the analogue of Dyson time-ordering operator (see., for example, [Dyson, 1949]).

$$\mathcal{M}(\mathbf{X}; t|t_0) = \mathrm{T}\exp\left\{\int\limits_{t_0}^{t} [\mathcal{H}(\tau), \circ]\, d\tau\right\}. \tag{2}$$

It should be noted, however, that for practical calculations this presentation is not convinient, and that is why for practical investigations is commonly used so-called Magnus representation [Andrianov, 2004a; Magnus, 1954]. In accordance with this approach, we can turn to the operator of the evolution of the dynamical system in the form of an exponential operator with a new generating operator $\mathcal{W}(t|t_0)$

$$\mathcal{M}(t|t_0) = \exp\left(\mathcal{W}(t|t_0)\right). \tag{3}$$

The equality (3) allows us to represent the solution in the form of a usual (non time ordered) operator exponential from a new operator ($\mathcal{W}(t|t_0)$, which can be calculated up to the desired order of nonlinearity according to the well known formula

$$\begin{aligned}\mathcal{W}(t|_0) = &\int\limits_{t_0}^{t} \mathcal{V}(\tau)d\tau + \alpha_1 \int\limits_{t_0}^{t} \left\{\mathcal{V}(\tau), \int\limits_{t_0}^{\tau} \mathcal{V}(\tau')\right\} d\tau + \\ &+ \alpha_1^2 \int\limits_{t_0}^{t} \left\{\mathcal{V}(\tau), \int\limits_{t_0}^{\tau} \left\{\mathcal{V}(\tau'), \int\limits_{t_0}^{\tau'} \mathcal{V}(\tau'')d\tau''\right\} d\tau'\right\} d\tau + \\ &+ \alpha_1\alpha_2 \int\limits_{t_0}^{t} \left\{\left\{\mathcal{V}(\tau), \int\limits_{t_0}^{\tau} \mathcal{V}(\tau')d\tau'\right\}, \int\limits_{t_0}^{\tau} \mathcal{V}(\tau')d\tau'\right\} d\tau + \ldots\end{aligned} \tag{4}$$

Using the well-known formula Campbell-Baker-Hausdorff and calculating the corresponding integrals, we can get an idea about the evolution operator up to the desired order of nonlinearity. It should be noted that the respective computations may be implemented both in numerical and symbolic presentations (see, e. g., [Andrianov, 2010a]).

3.2 The Matrix Representation of the Evolution Operator

According to [Andrianov, 2004a; Andrianov, 1997] we can write the presentation of eq. (1) in the Poincare–Witt basis $1, \mathbf{X}, \mathbf{X}^{[2]}, \ldots, \mathbf{X}^{[n]}, \ldots$:

$$\mathbf{X}(t) = \sum_{k=0}^{\infty} \mathbb{M}^{1k}(t|t_0)\mathbf{X}_0^{[k]}, \tag{5}$$

where $\mathbb{M}^{1k}$ are matrices $n \times \binom{n+k-1}{k}$ in general case may depend on the phase vector $\mathbf{X}(t)$ (for example, in the case of high-intensity beams). Namely these matrices can be evaluated up to necessary order both in symbolic and numerical forms [Andrianov, 2004a].

4 Some Necessary Remarks

In the papers [Andrianov, 2001a; Andrianov, 2001b] examined methods of calculating both exact and approximate invariants generated by operators of evolution. It should be noted that an important role is played by the symplecticity property of Hamiltonian systems, which is inherent in any object that obeys the Hamiltonian equations of motion, regardless of their particular properties. In the case of distributed dynamic systems (e.g., charged particle beams in accelerators and in plasma) in the modeling process dynamics we must ensure this property for each object, which significantly complicates both the modeling process itself and the search for optimal solutions.

With this in mind, we can formulate the following stages of the optimization process:

1. creation of the mathematical model of the investigated dynamic process;
2. formulation of exact and approximate invariants as properties for ensemble of particles (the property of symplecticity) and single-particle (the law of conservation of energy) and so on;
3. the choice of numerical and symbolic solutions in accordance with the used formalization;
4. debugging methods using test problems;
5. carrying out computational procedures for the modeling of real processes and objects.

All these stages are closely related with the mathematical tools used in the particular investigation. Indeed, in many practical problems need find nonlinear evolution of some systems (in space and/or in time) of complex dynamic systems, while ensuring the preservation of certain characteristics inherent in these dynamic systems. Therefore, we should be given special attention to the choice of mathematical and computational tools which are used for simulation. In this paper, we describe a unified approach, based on the matrix formalism that allows a natural way to describe the dynamics of these processes, and the management and control of the correctness of the research process.

The concept of evolution operators in terms of operator Dyson is primarily a qualitative method. But at present there exists a family of methods for constructive presentation of the corresponding operators. These techniques include three approaches. The first (developed in the works K.L.Brown [Brown, 1982]) not only was used in many papers, but and found its realization in a variety of software packages, among which we should highlight first of all package MAD [Iselin, 1994] and package COSY Infinity [Berz, 1990]. It should be noted that COSY Infinity is also based on the ideology of the tensor representation of nonlinear effects (model Taylor), but its computing core is based on differential algebra, which significantly increases computational performance.

The third approach suggested in papers of Alex Dragt (See, eg, [Dragt, 1994]), is based on the use of Lie groups and algebras and is realized in the package MARYLIE 3.0. There are also a number of packages, but they are also mainly based on the Taylor series. Besides there are rather many packages which can considered as superstructures over existing mathematical approaches. In this paper we base on the method proposed in the article S.N.Andrianov [Andrianov, 1997].

Formally, it also uses a representation of the equations of motion in the form of a Taylor series, but the construction of the solution is based on the matrix formalism, according to which the non-linear evolution equation can be written in the form of an infinite-dimensional linear ordinary differential equation. This view allows us to use well developed numerical and analytical methods for solving of similar equations. Besides this we obtain a possibility to use of methods and tools of symbolic mathematics for construction of computational algorithms, which not only greatly improves the computational efficiency, but gives possible naturally way to construct invariants (exact and approximate), but also to incorporate them into the computational process. We should also note that the usage of the matrix representation of the necessary facilities allows significantly easier keep track of these or other effects that influence the dynamics of the beam.

5 Control and Correctness of Numerical Calculations

In this section we briefly describe the methods for quality control of the correctness of the calculations. Any numerical algorithm must be verified on the relevant problems with the exact solution. As an example, we demonstrate that the proposed matrix formalism really allows us to build not only the correct solutions (with clearly fixed estimates for accuracy of calculations), but also to obtain exact solutions of nonlinear equations (see, for example [Andrianov, 2004a]).

5.1 The Exact Solutions as a Control of Computational Processes

As a simple example, consider the problem of solving a scalar nonlinear differential equation with a second-order nonlinearity

$$\frac{dx}{dt} = K_2\, x^2, \quad x \in R^1. \tag{6}$$

In particular, this equation describes dimensional motion of particles in a thin sextupole where K_2 – a strength of the sextupole field. As is known, the solution of equation (1) with $x(t_0) = x_0$ has the form

$$x = \frac{x_0}{1 - K_2(t - t_0)}, \quad t \geq t_0, \tag{7}$$

whose graph is shown in Fig. 1. After applying to the equation (1) ideology the matrix formalism for Lie transformations the corresponding propagator Lie for the equation (1) takes the form

$$\mathcal{M}(t|t_0) = \exp\left\{(t - t_0)K_2\, x_0^2 \frac{\partial}{\partial x_0}\right\}.$$

After simple algebraic manipulations (see., for exam-

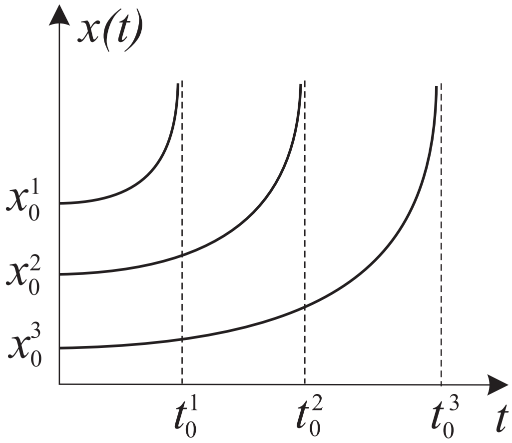

Figure 1. An approximate form the graph of (1)

ple [Andrianov, 2004a]), we obtain

$$\begin{aligned}\mathcal{M}\circ x_0 &= \sum_{k=0}^{\infty}\frac{(t-t_0)^k}{k!}\mathbb{P}_2^{k1}x_0^{k+1} = \\ &= \sum_{k=0}^{\infty}\frac{(K_2(t-t_0))^k}{k!}k!x_0^{k+1} = \\ &= \sum_{k=0}^{\infty}(K_2(t-t_0))^k\, x_0^{k+1}. \quad (8)\end{aligned}$$

After summation we find

$$\mathcal{M}\circ x_0 = \frac{x_0}{1-K_2(t-t_0)x_0}.$$

As expected, the solution coincides with the expression (2).

As a second example, consider the system of Hamiltonian equations of the following form

$$\begin{cases} \frac{dx}{dt} = a\,x^2, \\ \frac{dP_x}{dt} = b\,x^2 - 2\,a\,x\,P_x, \end{cases}$$

Its solution $\mathbf{X} = \mathbf{X}(\mathbf{X}_0; t\,|\,t_0) = (x, P_x)^*$:

$$\mathbf{X} = \frac{\mathbf{X}_0 + \mathbb{P}_2^2\mathbf{X}_0^{[2]} + \mathbb{P}_2^3\mathbf{X}_0^{[3]} + \mathbb{P}_2^4\mathbf{X}_0^{[4]}}{1+\mathbf{Q}_1^*\mathbf{X}_0}.$$

where the matrices and the vector can be computed using the above described method

$$\mathbf{Q}_1 = -(t-t_0)\begin{pmatrix}0\\a\end{pmatrix},\ \mathbb{P}_2^2 = a(t-t_0)\begin{pmatrix}0 & 3a & -b\\ 0 & 0 & 0\end{pmatrix},$$
$$\mathbb{P}_2^3 = a(t-t_0)^2\begin{pmatrix}0 & 0 & 3a & -b\\ 0 & 0 & 0 & 0\end{pmatrix},$$
$$\mathbb{P}_2^4 = \frac{a^2(t-t_0)^3}{3}\begin{pmatrix}0 & 0 & 0 & 3a & -b\\ 0 & 0 & 0 & 0 & 0\end{pmatrix}.$$

The similar solutions allow us to control the calculations carried out both in the analytical and in numerical form. In particular, at the stage of debugging software developed as a model of dynamic systems is necessary to use complex (primarily non-linear) dynamical systems, which have the exact solution. This allows to properly choose the details to the used numerical methods, for example, the step size of discretization of the independent variable ("time"), verify the energy conservation for conservative systems, etc.

5.2 The Exact and Approximate Invariants

In the previous subsection we briefly described the method of controlling of correctness of used approximate calculation procedures. Besides this we should have some instruments for additional control of the conservation of exact and approximate invariants inherent to the dynamical process under study. It should be noted that these invariants can divided into two classes: kinematic and dynamical invariants [Andrianov, 2004a].

Definition 1. *The own kinematic invariant of the dynamical system is named a function $I(\mathbf{X}, t)$ that does not change its value at any transformations generated by the dynamical system.*

Let $I(\mathbf{X}, t)$ is a kinematic invariant and $\{\mathcal{M}\}$ is a map which are generated by some class of dynamical systems (e.g. Hamiltonian systems) then formally one can write $\{\mathcal{M}\}\circ I(\mathbf{X}, t) = I(\mathbf{X}, t), \forall\,\mathbf{X},\, t$.

As an example, we can specify the symplectic property, according to which for Hamiltonian systems there exist the well known identity – the condition of symplecticity. Let us introduce the following designation

$$\mathbb{M}(\mathbf{X}, t\,|\,t_0; \mathcal{M}) = \mathbb{M}(\mathbf{X}; t\,|\,t_0) = \frac{\partial\mathcal{M}(t\,|\,t_0; \mathcal{H})\circ\mathbf{X}}{\partial\mathbf{X}^*},$$

then the matrix $\mathbb{M}$ satisfies to the well known symplecticity identity

$$\mathbb{M}^*(\mathbf{X}; t\,|\,t_0)\mathbb{J}(\mathbf{X})\mathbb{M}(\mathbf{X}; t\,|\,t_0) = \mathbb{J}(\mathbf{X}).$$

Failure to comply with this identity leads to a substantial violation of the qualitative and quantitative properties, and as a result, to incorrect results of mathematical (theoretical) and computational modeling.

Using methods published in [Andrianov, 2004a; Andrianov, 2001c] one can construct a chain (in general case an infinite chain) of linear algebraic equations for elements of matrices which represent the step-by-step matrix presentation of the evolution operator M which generated by the our dynamical system. This allows us guarantee that the truncated series (containing a finite number of terms in accordance with the order of nonlinearity) ensures strict compliance with the conditions of the symplecticity. It should also be noted that the degree of closeness of the solutions can be estimated in any order of nonlinearity (see., for example, [Andrianov, 2012]). Here it should be noted that the corresponding estimates are universal, in other words do not depend on the trajectory of the object of control (in this case the particles). Thus we can obtain an estimate of the approximate evolution operator itself for an arbitrary initial phase coordinates of the beam particles (see, for example, [Andrianov and Kulabukhova, 2013].

In other words, for each initial state of a conservative dynamical system conserved a value of energy which corresponds to the selected particle trajectory (see,i. e. [Zhong and Marsden, 1988]). The energy conservation law is an example the second type of invariants – the own dynamic invariant.

Definition 2. *The function $I(\mathbf{X}, t)$, which is conserved under the transformations generated by a particular dynamical system is named an own dynamic invariant of the dynamical system.*

In the case of the law of conservation of energy, we obtain a map that displays a given initial state of the particle in a current one. Thus the corresponding evolution operator depends on the initial energy of a particular particle.

6 Some Remarks on the Optimization Procedure

Described in the previous sections matrix formalism allows not only to efficiently carry out calculations on the dynamics of particle beams, but also to search for the optimal values of the control parameters, providing not only a given behavior of the beam as a whole, but also some specific beam characteristics which defined in the structural control elements. It should be noted that in general such problems are multi-purpose tasks of multiparameter optimization. It is this fact leads to the need for choosing the technologies that can effectively address such problems, especially with the use of parallel and distributed computing resources. In particular, as an example of such technologies, we can consider the so-called multi-agent systems (see., for example, [Camelia Chira, 2007],[Andrianov, 2015]). It should be noted that it is multi-agent technology can be adequately implemented it using the matrix formalism described in the previous sections. The use of exact and approximate invariants allows you to enter additional criteria for optimization problems, which improves the efficiency of the search for optimal solutions as the optimization process will guarantee that the conditions of the correctness performed during computational procedures. It should also be noted that the methods of global optimization [Gong and Chao, 2012], [Andrianov *et al.*, 2011] also require specialized software which is adequate to mathematical methods used in modeling the dynamics of the particles in the accelerator systems.

7 Conclusion

In the paper we considered some problems of mathematical and computer methods of modeling of long time evolution of the beam in cyclic accelerators. The described approach is based both on using of some exact and approximate solution methods for beam motion equations and also on a paradigm of parallel and distributed computational experiments. All the proposed tools are realized in an analytic form (in the form of appropriate formulas and algorithms, derived using the such packages of computer algebra packages as Maple and Mathematica, see, for example, [Andrianov, 2010b; Andrianov and Kirsanov, 2014]), and also numerical methods that implement the proposed approaches. Testing of the respective software products demonstrated sufficiently effectiveness in solving some practical problems.

Acknowledgements

The work is supported by Saint Petersburg State Universit (project 9.38.673.2013).

References

Andrianov, S.N. (1997) A matrix representation of Lie algebraic methods for design of nonlinear beam lines. In: *AIP Conference Proceedings*, No. 391, pp. 355–360.

Andrianov, S.N. (2001a) Symplectification of truncated maps for Hamiltonian systems. *Mathematics and Computers in Simulation*, Vol. 57, No. 3–5, pp. 139–145.

Andrianov, S.N. (2001b) Symbolic computation of approximate symmetries for ordinary differential equations. *Mathematics and Computers in Simulation*, Vol. 57, No. 3–5, pp. 139–146.

Andrianov, S. (2001c) Order-by-order symplectification of truncated Lie maps. In: *Proc. of the 2001 Particle Accelerator Conf.* Chicago, USA, pp. 1787-1789. http://accelconf.web.cern.ch/AccelConf/p01/papers/tpph053.pdf

Andrianov, S.N. (2004a) *Dynamical Modeling of Particle Beam Control Systems*. SPbSU. SPb. (in Russian).

Andrianov, S.N. (2004b) Step-by-step optimization for beam lines. *Nuclear Instruments and Methods in Physics Research, Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol. 519, No. 1-2, pp. 28–36.

Andrianov, S.N. (2010a) A role of symbolic computations in beam physics. *Lecture Notes in Computer Science* Vol. 6244/2010, pp. 19–30.

Andrianov, S.N. (2010b) A role of symbolic computations in beam physics. *Lecture Notes in Computer Science*, Vol. 6244/2010, pp. 19–30.

Andrianov, S., Ivanov, A., Kosovtsov, M., and Podzyvalov, E. (2011) A global optimization approach based on symbolic presentation of a beam propagator. In: *Proc. of IPAC2011*, San Sebastián, Spain. pp. 2280–2282.

http://accelconf.web.cern.ch/AccelConf/IPAC2011/papers/wepc115.pdf

Andrianov, S. (2012) The Convergence and accuracy of the matrix formalism approximation. In *Proc. of ICAP2012*, Rostock-Warnemünde. Germany, pp. 93–95. http://accelconf.web.cern.ch/AccelConf/ICAP2012/papers/tuscc2.pdf

Andrianov, S., and Kulabukhova, N.(2013) Two and three Dimensional Models for Analytical and Numerical Space Charge. Simulation. In: *Proc. of IPAC2013, Shanghai*, China, pp. 2585–2587. http://accelconf.web.cern.ch/AccelConf/IPAC2013/papers/ wepea038.pdf

Andrianov, S.N., and Kirsanov, A.V. (2014) Automation of the matrix formalism the field equations and equations of motion. In: *Proc. of BDO'2014*, St. Petersburg, Russia, pp. 2.

Andrianov, S.N. (2015) Lie algebraic methods as mathematical models for high performance computing using the multi-agent approach. In: *Trans. on Comp. Sci.* pp. 15.

Berz, M. (1990) *COSY INFINITY Reference Manual*. Lawrence Berkeley Laboratory, University of California, Accelerator and Fusion Research Division.

Brown, K.L. (1982) A First- and Second-Order Matrix Theory for the Design of Beam Transport Systems and Charged Particle Spectrometers. *SLAC Report-75*.

Camelia Chira, A. (2007) Multi-agent approach to distributed computing. *Computational Intelligence Reports*, No. 042007, pp. 1–8. http://www.cir.cs.ubbcluj.ro/

Dragt, A.J. et al (1994) *MARYLIE 3.0 User's Manual*. Univ. of Maryland Physics Departmen.

Dyson, F.J. (1949) The S-Matrix in quantum electrodynamics. *Physical Review*, 75 , 486, pp. 1736–1755.

Gong, C. and Chao, Y.C. (2012) A new platform for global optimization. In: *Proc. of IPAC2012*, New Orleans, Louisiana, USA, pp. 256-258. http://accelconf.web.cern.ch/AccelConf/IPAC2012/papers/moppc055.pdf

Iselin, F.Ch. (1994) *The MAD Program (Methodical Accelerator Design) Version 8.13 Physical Methods Manual*. Geneva, Switzerland.

Lehrach, A. (2012) Project overview and computational needs to measure electric dipole moments at storage rings. In: *Proc. of ICAP2012*, Rostock-Warnemünde, Germany, pp. 7–11. http://accelconf.web.cern.ch/AccelConf/ICAP2012/papers/moaai1.pdf

Magnus, W. (1954) On the exponential solution of differential equations for a linear operator. *Comm. Pure and Appl. Math.* VII: pp. 649–673.

Makino, K., Berz, M. (2014) Rigorous Global Optimization of System Parameters. Vestnik Mathematica 10, 2, pp. 62–72.

Proceedings of Fifteenth Int. Workshop on Beam Dynamics and Optimization (2008). Ovsyannikov, D., Berz, M., Snopok, P. (Eds.) St. Petersburg, Florida, World Scientific, ISSN 0217-751X.

Streichert, M., Hiller, N., Huttel, E. et al. (2012) Global optimization of the anka lattice using multiobjective genetic algorithms (MOGA). In: *Proc. of ICAP2012*, Rostock-Warnemunde, Germany, pp. 72–74. http://accelconf.web.cern.ch/AccelConf/ICAP2012/papers/tuabc2.pdf

Zhong, Ge, and Marsden, J.E. (1988) Lie-Poison Hamiltonian-Jacobi theory and Lie-Poisson integrators. In *Physics Letters A*, Vol. 133, No. 3.