LEGO-TECHNOLOGY APPROACH FOR BEAM LINE DESIGN

S.N. Andrianov, SPbSU, Saint-Petersburg, Russia

Abstract

In this paper we present an approach to study some problems of beam line modeling and optimization. We use a modular principle for all levels of the modeling and optimization procedures. The design code proposed by J.Irwin with colleagues is based on modular presentation only for beam line description. For every module (LEGO-module) they used numerical methods for equation motion integration and additional operations. In our approach we introduce LEGO-objects on all levels of the modeling and optimization processes. The beam line components each have two main sets of LEGO-objects: the first contains all necessary objects for beam line component description and the second - all objects which correspond to a transfer map (a beam propagator) as an aggregate of two-dimensional matrices. These matrices are computed in symbolic forms up to some approximation order of the nonlinear aberrations. Similar approach we developed for the space charge forces description too. In this case a set of model distribution functions is used. Some examples of practical application are discussed.

1 INTRODUCTION

In the present time the computer experiment is very complex process lodged hard restrictions on the used software. To satisfy the restrictions, it is necessary to generate computer codes which will be comfortable, adaptive, updatable and extended. These properties can be realized using object-oriented paradigm. But usually the programmers are limited themselves only object-oriented programming without modification of physics-mathematics presentation for the object under study. The modular principle is based on the deep decomposition all levels of the modeling and optimization procedures. For the beam line design process such approach was proposed by J.Irwin with his colleagues. For this aim they used the LEGO-concept for description of numerical methods for equation motion integration and some auxillary operations. In the present paper we introduce more wide and deep structure for the set of LEGO-objects: from objects for beam line component description up to objects which correspond to a transfer map (a beam propagator) as an aggregate of two-dimensional matrices. These matrices are computed in symbolic forms up to some approximation order of the nonlinear aberrations or can be evaluated numerically. Similar approach we developed for the space charge forces description too. In the case of space charge dominated beams we use two sets of auxiliary functions: model distribution functions and functions describing boundaries of manifold occupied by phase coordinates of beam particles.

2 THE BASIC CONCEPTS

Complexity of the beam physics problems leads to necessity to give users comfortable and powerful computer tools for their investigations. Obviously that effectiveness of such programming products depends from one hand on mathematical methods and conceptual approaches which are put in the corresponding software. It is not enough to offer only human graphical user interface (GUI). It is necessary to present flexible, extendable and rebuilt product with (of course) GUI ensuring comfortable interaction with computer or with computer systems (for example, distributed computer system). This form of the GUI depends on concrete beam physics problem. It should be noted that all beam physics problems can be separated in to two wide classes: the problems of *beam line design* with some given characteristics (focusing systems, separators, periodical structures in accelerators and so on) and the problems, connected with studying of beam evolution including effects of different nature (first of all, nonlinear effects, influence of space-charge forces and so on). The choice of mathematical methods and as a corollary a type of GUI depends on to what class this or that problem belongs. The type of problem dictates the types of LEGO-objects. A part of them can first of all has ”physical” sense (quadrupole, dipole,...) and only further has corresponding mathematical filling (motion equations, maps and so on). The other part deals with mathematical description of some physical problem and only after careful investigation a researcher go to corresponding physical description. We note that it is often useful to introduce a special cls=ass of LEGO-objects: the class of virtual objects which have a sense only as abstract (computer) object [8]. In this paper we use the unified mathematical tools — the matrix formalism for Lee algebraic methods [1].

3 THE MATHEMATICAL BACKGROUND

The use of Lee algebraic methods have been developed for beam physics problems for last twenty years. The basic theoretical contribution was made by Alex J.Dragt (see, for example, [2]). But the polynomial presentation developed in traditional Lee tools (for Hamiltonian formalism) limits possibilities to create software satisfied to the modern technologies requirements. In the present paper we describe the matrix formalism satisfied necessary requirements and serving all virtue of Lie algebraic methods. In the previous authors works this formalism was developed for wide classes problems (see, for example, [3]–[5]). It is important to note that the matrix formalism are accepted with parallel and distributed computing using high performance computers.

3.1 The Lie Transformations

The time evolution of dynamic systems may be represented by one-parameter groups of maps $\mathcal{M}(s|s_0)$ acting on the initial values of phase space variables $\mathcal{M} : \mathbf{X}_0 \to \mathbf{X} = \mathcal{M} \circ \mathbf{X}_0$. In the case of Hamiltonian systems such maps form a symplectic group of symplectic maps, so called Lie maps. In this way one have to compute the action of this group for given dynamic systems.

3.2 The Matrix Formalism

Let

$$\frac{d\mathbf{X}}{ds} = \mathbf{F}(\mathbf{X}, \mathbf{U}, s)$$

be a motion equation for charged particles in a beam line and there is an expansion $\mathbf{F}(\mathbf{X}, s) = \sum_{k=0}^{\infty} \mathbb{P}_{1k}(s)\mathbf{X}^{[k]}$. Here $\mathbf{X}$ is a phase vector in a local coordinate system, $\mathbf{U}$ is a control vector describing external control fields (generated, for example, by dipoles, quadrupoles and so on) and corresponding geometrical parameters. $\mathbf{X}^{[k]}$ is the Kronecker power of a phase vector X of k-th order, $\mathbb{P}^{1k}(\mathbf{U}; s)$ is a $(n \times \binom{n+k-1}{k})$–dimensional matrix. For non-autonomous systems one can use the so called Magnus's representation [1]. This approach allows to pass from the time-ordered exponent operator to a routine exponential operator. The Dragt-Finn factorization for the Lie transformations allows to rewrite the exponential operator as an infinite product of exponential operators of Lie operators

$$\mathcal{M} = \ldots \cdot \exp\{\mathcal{L}_{H_2}\} \cdot \exp\{\mathcal{L}_{H_1}\} =$$

$$= \exp\{\mathcal{L}_{V_1}\} \cdot \exp\{\mathcal{L}_{V_2}\} \cdot \ldots,$$

where $H_k = \mathbf{H}_k\mathbf{X}^{[k]}$, $V_k = \mathbf{V}_k\mathbf{X}^{[k]}$ are homogeneous polynomials of k-th order. The vectors $\mathbf{H}_k$ or $\mathbf{V}_k$ can be calculated with the help of the continuous analogue of the CBH-and Zassenhauss formulae and by using the Kronecker product and Kronecker sum technique for matrices [1]. Moreover, using the matrix representation for the Lie operators one can write a matrix representation for the Lie map generated by these Lie operators

$$\mathcal{M} \cdot \mathbf{X} = \sum_{k=0}^{\infty} \mathbb{M}^{1k}\, \mathbf{X}^{[k]},$$

where the matrices $\mathbb{M}^{1k}$ (*solution matrices*) can be calculated according to the recurrent sequence of formulae of the following types:

$$\mathcal{M}_k \cdot \mathbf{X}^{[l]} = \exp\{\mathcal{L}_{G_k}\} \cdot \mathbf{X}^{[l]} =$$

$$= \mathbf{X}^{[l]} + \sum_{m=1}^{\infty} \frac{1}{m!} \prod_{j=1}^{m} \mathbb{G}_m^{\oplus((j-1)(k-1)+l)}\ X^{[m(k-1)+l]},$$

where $\mathbb{G}^{\oplus l} = \mathbb{G}^{\oplus(l-1)} \otimes \mathbb{E} + \mathbb{E}^{[l-1]} \otimes \mathbb{G}$ denotes the Kronecker sum of l-th order. For the inverse map $\mathcal{M}^{-1}$: $\mathbf{X} \to \mathbf{X}_0 = \mathcal{M}^{-1} \cdot \mathbf{X}$ one can compute the corresponding block-matrices using the generalized Gauss's algorithm.

The desired solution is created in the form of power series. It is clear that this way can be realized only with truncated procedures for some chosen order of expansions. The corresponding matrices and vectors $\mathbb{P}^{1k}$, $\mathbb{G}^{1k}$, $\mathbf{G}_k$, $\mathbf{V}_k$ and $\mathbb{M}^{1k}$ can be calculated up in symbolic forms using the computer algebra codes (in our case we use Maple codes). It is necessary to note that for this approach there appear two problems. The first of them is connected with the support of the accuracy of truncated expansions and the second — with the support of intrinsic properties (for example, symplecticity for Hamiltonian systems, [5], [6]).

So the description for the beam physics problem can be presented in the terms of different sets of vectors, two-dimensional matrices and operations corresponding operations. It should be noted that all our manipulations lay in linear algebra field. This is very important from computational point of view. Indeed in this case we obtain the mathematical description admitted parallel and distributed processing naturally.

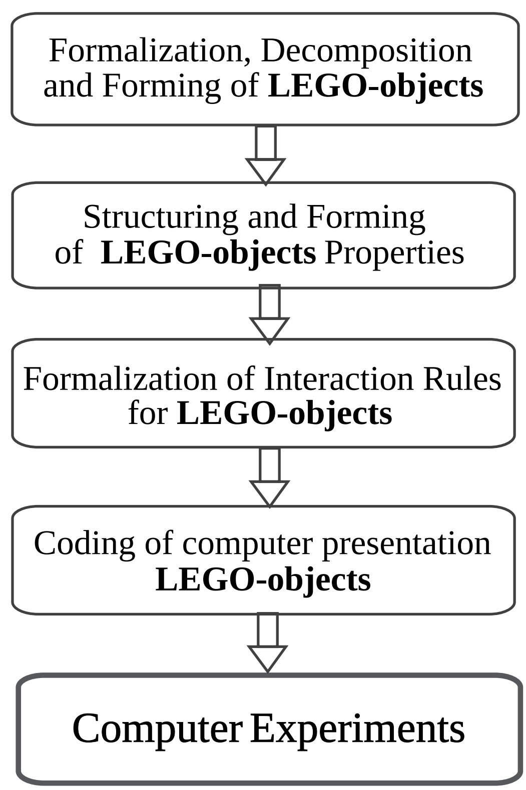

Figure 1: The main steps of construction of a computer model using the LEGO-objects technology.

4 THE BASIC PROPERTIES OF LEGO-OBJECTS

Let us formulate the basic properties of imposed LEGO-objects.

- *The LEGO-object has property of minimality*. The further decomposition is impossible. The properties of a universality and abstracting are proper in LEGO-object.
- *The LEGO-objects are constructed for all levels of the computer model*. In the offered approach of the solution — object are basic, and the main emphasis is made on their calculation, performance and storage.
- *The LEGO-objects are compared to all methods, operating during simulation, objects etc*.
- *The majority of LEGO-objects represents matrices of small dimension*, from which with the help of operations of a matrix algebra the matrix objects used during simulation and optimization "gather". Such form (shape) of performance of LEGO-objects is most adequate to modern computing systems, including distributed and parallel computer systems.

The scheme of necessary levels of LEGO-objects creation is presented on the Fig. 1.

5 THE BASIC LEGO-OBJECTS DESCRIPTION

Let us formulate the basic types of LEGO-objects which can be imposed in the frame of the matrix formalism.

- Abstract matrices and built-in resulting expanded matrix operations.
- A system of the geometrical characteristics describing technological parameters of control elements.
- A set of base model functions $g(\mathbf{A}, s)$ describing fringe fields for the control elements as functions of the independent variable s measured along the electrical axis of a control element. These model functions are presented parametrically using the parameter vector $\mathbf{A}$. This set universal. In another words they are the same for all types of the control elements: from dipoles up to multipole lenses or other control elements.
- Matrices $\mathbb{P}^{ik}(g(\mathbf{A}, s); s)$ entering in expansion of right hand site for the motion equations which include external electromagnetic fields, generated by control elements (beam line elements).
- A set of model distribution functions $f(\mathbf{B})$ for space-charge distribution models. Here a researcher realized a preliminary investigation for generation such model functions using experimental data.
- Matrices $\mathbb{P}^{ik}(f(\mathbf{B}); s)$, entering in expansion of right hand site for the motion equations, generated by space-charge forces and correspond to different model distribution functions which are describing in the term of parameter vectors $\mathbf{B}$.
- A system of the *matrices-solutions* $\mathbb{M}^{ik}$ for some base set of command elements, down to some order of non-linearity N, $i < k$, $k \leq N$.
- A system of the matrixes - solutions $\mathbb{M}^{ik}$ for some base set of model distribution functions, up to some order of nonlinearity N, $i < k$, $k \leq N$.
- A system of basic LEGO-objects describing beam state. The form of presentation is also matrix.

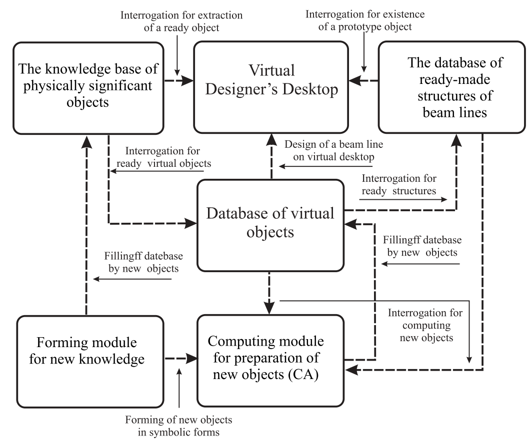

Figure 2: The scheme of interaction between basic modules of computer constructor for LEGO-objects technology.

6 REFERENCES

[1] S.N. Andrianov, in: AIP Conf. Proc. No 391, NY, 1997, P. 355–360.

[2] A.J. Dragt, in: Physics of High Energy Particle Accelerators. AIP Conf. Proc., N 87. N.-Y. (1982). P. 147–313.

[3] S.N. Andrianov, in: Proc. of the 1999 Particle Accelerator Conference, 1999, NY (1999). P. 2701–2703.

[4] S.N. Andrianov, in: Proc. of the Sixth European Particle Accelerator Conference, Stockholm, Sweden, 1998. Bristol (UK) (1998). P. 1091–1093.

[5] S.N. Andrianov, in: Mathematics and Computers in Simulation. Vol. 57. Issue 3–5, (2001), P. 139–145.

[6] S.N. Andrianov, in: Mathematics and Computers in Simulation. Vol. 57. Issue 3–5, (2001), P. 146–154.

[7] Andrianov S.N., Taryanik V. M., in: Proc. Seventh European Particle Accelerator Conference — EPAC'2000. Vienna (Austria), 2000. P. 1351–1353.

[8] Andrianov S.N., in: Proc. of the 11-th Int. IFAC Workshop Control Application of Optimization. Vol. 1. Pergamon Press. Oxford. UK. 2001. P. 11-16.