

Lawrence Berkeley National Laboratory

Lawrence Berkeley National Laboratory

Title

GOEMON, a C++ library for accelerator modeling and analysis

Permalink

<https://escholarship.org/uc/item/7874z40h>

Author

Nishimura, Hiroshi

Publication Date

2001-06-22

GOEMON, A C++ LIBRARY FOR ACCELERATOR MODELING AND ANALYSIS*

Hiroshi Nishimura, LBNL, Berkeley, CA 94720 , USA

Abstract

Goemon is a C++ library that supersedes Tracy2 written in Pascal. It has been continuously developed at ALS since 1993. This paper describes its design and implementation issues, and applications.

1 INTRODUCTION

During the design and construction phase of the ALS project[1], we have developed three software programs, Tracy[2], Gemini and Tracy2[3], for flexible accelerator modeling and simulation studies. They are modified versions of the Pascal S compiler[4] that are linked to accelerator physics routines. Goemon is a superset of the Tracy2 physics routines ported to C++ and restructured as a library for a better portability and flexibility. By using operator overloading, it supports differential algebra (DA)[5] and lattice definitions effectively. It has been used to create interactive-graphical applications and to carry out automated simulation studies to optimize parameters. It also cooperates with tools for graphical display programs and interactive language processors.

2 PHYSICS

Goemon inherits the models of Tracy2 and supports the 4x5 standard matrix model and a full 6x6 drift-and-kick model for the following Hamiltonian:

$$H = \frac{p_x^2 + p_y^2}{2(1+\mathbf{d})} - \frac{x\mathbf{d}}{\mathbf{r}} + \frac{x^2}{2\mathbf{r}^2} - \Delta\mathbf{d} + V(x, y)$$
$$\mathbf{d} = \frac{p_\tau - p_0}{p_0}, \Delta\mathbf{d} = \mathbf{d} - 1 + \frac{p_\tau}{\mathbf{b}}$$

Here (x, p_x, y, p_y, τ, d) are the 6dim coordinates of a particle, p_τ energy, p₀ nominal beam energy, ? a bending radius and β is the beam velocity/c. V(x, y) is a potential that depends on a position. The Kpot Hamiltonian[6] is also supported to model small rings:

$$H = -\left(1 - \frac{x}{\mathbf{r}}\right) \sqrt{(1+\mathbf{d})^2 - p_x^2 - p_y^2} + \frac{x}{\mathbf{r}} + \frac{x^2}{2\mathbf{r}^2} + V(x, y)$$

For insertion devices, a newly developed symplectic integrator[7] has been just implemented. These models are available in a fully coupled mode with various kinds of error settings.

* This work was supported by the Director, Office of Energy Research, Office of Basic Energy Sciences, Material Sciences Division, U. S. Department of Energy, under Contract No. DE-AC03-76SF00098

3 DESIGN ISSUE

Goemon is not an integrated application. It is a library to be a part of various application programs. The following items are to attain a better modularity as a library.

3.1 Input Parsing

Goemon does not include any input parser for user's algorithm. An advancement of computer technologies has enabled to recompile users program and link the library quite promptly. It usually takes only a few seconds in Microsoft Visual Studio 6.0 on Windows NT/2000. It can be also used to extend the commands of interactive language processors like Perl or Matlab[8]. We present two examples of using Matlab and Omatrix[9].

3.2 Lattice Parsing

Goemon supports lattice definitions by using a beam line class. It does not assume any lattice input file. However, it provides a set of C++ macro definitions to define a lattice structure by using operator overloading. The next example shows how we can construct a beam line BL from 4 elements.

```
MARKER(SYM);  
DRIFT(D1, 0.5);  
QUAD(Q1, 0.2, 2.34);  
DRIFT(D2, 0.2);  
Eline BL=SYM+D1+2*Q2+D2;
```

Elements including drift spaces and magnets are treated like characters and beam lines are like strings.

3.3 Portability

Goemon is developed in Microsoft Visual C++ and then recompiled in Borland C++ Builder and gcc 2.95.2 on various Unix platforms. The only difference among these three compilers is the way of using the 'const' attribute when overloading is involved.

3.4 Compatibility

Goemon includes only mathematical and accelerator physics routines. Therefore, it is compatible with any C/C++ library in other field of graphics, database or networking.

3.5 Class Structure

Tracy 2 has treated magnets in a very generic way by introducing a series of multi-pole kicks. On the other hand, Goemon has a simple but redundant class structure. For example, a quadrupole can be either an instance of the Quad class or the Mpole class depending on the context of the calculations. This is primarily for the faster execution speed for particle trackings. The switching between the integrators is done by setting a flag.

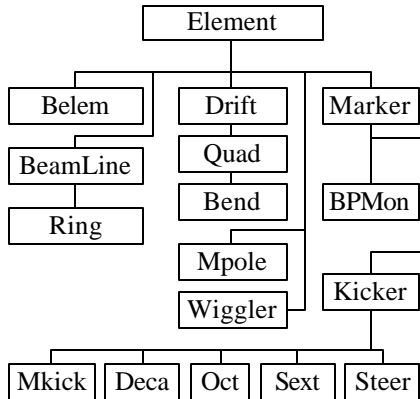


Fig.1. Goemon Class Structure

4 IMPLEMENTATION ISSUE

4.1 Operator Overloading

Operator overloading plays three critical roles: (1) ordinary vector/matrix arithmetic, (2) DA and (3) lattice definitions. Goemon has a fast linear DA to calculate transfer matrices and radiation-related quantities such as damping times and emittance. Non-linear DA for a map generation is also designed for faster execution speed by limiting itself to accept up to 8 independent variables to 15-th order so that it can be allocated on a stack, not on a heap.

4.2 Template

C++ templates make program development very efficient. However, Goemon does not use them for both portability and performance issues. It is crucial to guarantee rapid rebuild cycle by keeping the overhead of the compiling low. Structures like linked list are individually implemented.

5 ASSOCIATED TOOLS

5.1 Graphical Tools

When Goemon is used to create console programs, users can use several graphics utilities to display Twiss functions, dispersion functions, orbits and dynamic apertures.

5.2 MAD Lattice Parser and Generator

We have developed a utility to import a lattice defined in the MAD8[10] format. It parses a subset of a MAD8 lattice file and generates a lattice objects in Goemon. A MAD8 lattice generation is supported by Goemon library itself as a method.

6 APPLICATIONS

2.1 Automated Simulation Studies

One of the most important and time-consuming tasks in the lattice optimization is a parameter search under realistic assumptions. By changing lattice parameters in steps, it refits the lattice parameters, distributes various kinds of errors and applies corrections and, for example, calculates the residual COD and dynamic apertures to find better working conditions. Our recent experiences are as follows:

(1) Vertical Dispersion Bump.

A femto-second electron beam slicing designed at ALS[11] uses vertical dispersion generated at the center of a long straight section. It requires skew quadrupole magnets to kick horizontal dispersion onto the vertical plane. Goemon was used to simulate a vertical emittance growth in presence of realistic orbit errors.

(2) NLC Damping Ring

The NLC damping ring for 120 Hz operation[12] was designed by using Goemon. The main issue was the selection of the working point to attain sufficient dynamic aperture. An intensive dynamic aperture scan was carried out over wide range of tune space by using a fully automated simulation program that uses Goemon.

(3) VLHC Main Ring.

The orbit correction scheme was tested for the VLHC lattice with low-field bending magnets for various random seeds. The number of elements in the lattice is over 66000 including 9720 bending magnets, 824 correctors and 870 BPMs in both planes. By utilizing customized doubly-linked list for beam transfer lines, Goemon can handle such a large ring seamlessly including the COD correction with SVD and dynamic aperture calculations[13].

2.2 Small Ring Calculations

By using the Kpot Hamiltonian, Goemon enables optics calculations and particle tracking in a consistent manner for small ring such as the IR Ring[14] being designed at ALS that has 30-degree gradient bending magnets.

2.3 Interactive GUI Applications

TracyV[15] has been a name for interactive graphical applications with Tracy physics library in it. They were originally written in Borland Delphi on Windows. Newer versions are in Borland C++ Builder with the Goemon library. The example is the one for ALS with superbend.

2.4 Model-based Control System

Goemon supplies routines to control orbits and other physics quantities to the online programs at ALS. However, it is not an integrated part of the control system as it was developed after the commissioning. On the other hand, BESSY II has adopted it as a part of its control system for model-based controls[15].

2.5 Cooperation with Matlab and Omatrix

Goemon can be used to extend the functions of interactive language environment like Matlab[16]. A lattice definition can be directly supported by the list structure of an environment itself. Graphics and various routines optimized for matrix operations are ready to use for accelerator simulations. We have two examples: TracyM[17,18] and TracyML[17,19]. TracyM is an application in Omatrix[20] that is a Matlab-like environment. Goemon is wrapped with a set of C API in a DLL and exported to it. TracyML: This is a subset of Goemon rewritten in Matlab 5 by using its user-definable data structures, which has been superseded by AT[21] for the use at SSRL, SLAC.

6 REFERENCES

- [1] "1-2 GeV Synchrotron Radiation Source, Conceptual Design Report," LBL PUB-5172 Rev. LBL,1986.
A. Jackson, IEEE 93PAC, 93CH3279-7,1432, 1993.
- [2] H. Nishimura, "TRACY, A Tool for Accelerator Design and Analysis", EPAC 88,803,1989.
- [3] J.Bengtsson, E.Forest and H.Nishimura, "Tracy Users Manual",unpublished.
- [4] R.E.Berry, "Programming Language Translation", Ellis Horwood Ltd., England, 1981.
- [5] M.Berz, SSC-152, 1988
Leo Michelotti, IEEE PAC89, CH2669-0(1989)839
N. Malitskey, A. Reshetov and Y. Yan,SSCL-659, 1994.
- [6] E. Forest, "Beam Dynamics, A New Attitude and Framework", Chap.12, 1998, Harwood Academic Publications.
- [7] Y. Wu, E. Forest, D. S. Robin, H. Nishimura, A. Wolski, and V. N. Litvinenko, "Symplectic Models for General Insertion Devices", in this proceedings.
- [8] <http://www.mathworks.com/>
- [9] <http://www.omatrix.com/>
- [10] <http://fci.home.cern.ch/fci/mad/mad8/user/mad.html>
- [11] Y. Wu, H.Nishimura, D.S.Robin, A.A.Zholents and E. Forest, "Mini-Beta Lattice for the Femto-Second X-Ray Source at the Advanced Light Source", to be appeared on the proceeding of EPAC 2000.
- [12] http://www-project.slac.stanford.edu/lc/local/notes/dr/NLC_2000_04_03.pdf
- [13] H. Nishimura and A. Jackson "Correctors for the Low-Field VLHC", February 6, 2001, unpublished.
- [14] J. Byrd, et.al., "On a Dedicated Infrared Synchrotron Radiation Source at the ALS", in this proceedings.
- [15] R. Bakker, D. Kramer, B. Kuske, P.Kuske, R.Muller, "Establishment Of A Model For Interpretation and Correction Tools for BESSY II", PAC'99, New York(1999), p2379.
- [16] <http://www.mathworks.com/>
- [17] H. Nishimura " Matlab-like Environment for Accelerator Modeling and Simulation", to be published in the proceedings of ICAP 98.
- [18] H. Nishimura, "Introduction to TracyML", LBNL ALS/LSAP-246,1998.
- [19] H. Nishimura, "Introduction to TracyM, Part 1", LBNL ALS/LSAP-258,1998.
- [20] <http://www.omatrix.com/>
- [21] A.Terebilo, "Accelerator Toolbox for Matlab", SLAC-PUB 8732, May 2001. <http://www-ssrl.slac.stanford.edu>