INTERACTIVE AND PROGRAMMABLE ENVIRONMENT FOR ACCELERATOR MODELING AND SIMULATION*

H. Nishimura and W. Decking, LBNL, California, USA

Abstract

An interactive and programmable environment for accelerator design, simulation and modeling studies has been created by using O-matrix. O-Matrix is a MATLAB-like environment on Windows. TracyM is an external library that wraps the C++ class library Goemon for the use in the O-Matrix environment.

1 INTRODUCTION

It has been well recognised that object-oriented programming (OOP) enables a very efficient software development. This is especially true in the field of accelerator design, simulation and modeling where physical entities can be naturally mapped onto software objects. OOP was adopted at an early stage of the ALS project at LBNL and one of the C++ class libraries developed is Goemon[1], which models and simulates accelerators. It has been used as a building block of various application programs. TracyM is a recent effort to provide an interactive and programmable environment to the library at run-time.

2 BACKGROUND

It is not a trivial task to make a simulation and modeling program that has flexible programmability in a highly interactive and graphical environment. A kind of a language interpreter and a complex graphical user interface (GUI) programming must be combined. When we started looking for a possible solution after creating programs like Tracy[2] and TracyV[3], MATLAB[4] was successfully introduced for the ALS machine studies and controls[5]. It drove us to use MATLAB or some other similar environment on top of Goemon. As Goemon is written in C++, we can expect a better connectivity to the environment than Tracy, which is written in Pascal.

Our approach is to create a thin layer of external functions through which a MATLAB-like environment accesses the Goemon class library on Windows NT. It is a dynamic link library (DLL) that implements external functions for the use from a MATLAB-like environment.

3 O-MATRIX

O-MATRIX[6] is one of the MATLAB-like environments that are available on Windows. It was chosen for its better connectivity to external function calls. It allows a single DLL to support multiple external functions instead of requiring a separate DLL for each individual external function. As external functions must share Goemon objects created in the DLL memory space, it is crucial to have one DLL to support multiple external functions.

4 TRACYM

TracyM is a DLL written in Visual C++ on Windows NT and wraps the Goemon class library for the use through O-matrix. It also serves as a layer to hold C++ objects of Goemon that are to be manipulated by O-Matrix programs. As it is a thin layer to a compiled library where all the CPU-intensive numerical calculations are done, there is no visible reduction in the execution speed. Fig.1 is an example of TracyM running in the O-matrix environment. A user's logic including the lattice configuration is written in the O-Matrix language that is similar to that of MATLAB.

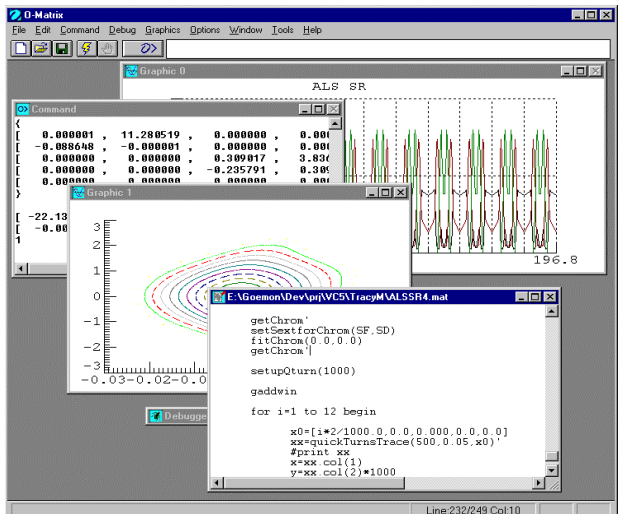


Fig.1. TracyM in O-matrix

* 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.

Here are the major feature of TracyM from a user's point of view:

2.1 ELEMENT DEFINITION

Elements are the building blocks of beam transport lines, such as drift spaces, magnets and beam position monitors. They are defined in the O-Matrix language as follows:

```
SYM =marker("SYM")
L1  =drift("L1", 3.378695)
QF =quad ("QF", 0.17200*2, 2.250)
BU =bend ("BU", 0.43257,5.0,3.0,0.00,-0.81)
```

Here the functions like "marker" are all external calls to the Goemon routines in the DLL that return references to the element objects created and stored inside the DLL memory space.

2.3 LATTICE DEFINITION

Lattice structures are described as O-Matrix lists of elements like:

```
M1 = [D1,F1,F1,D2]
M2 = [M1, D3,MON,D4,B1]
```

Therefore, TracyM does not have to support the syntax of lattice definitions.

2.4 BEAM LINES AND A RING DEFINITION

TracyM holds beam line objects and a ring object of Goemon. A beam line is defined as:

```
BL1=beamline(CELL)
```

where CELL is a lattice definition. For a ring, symmetry is added as the second parameter.

```
ring(CELL,12)
```

As only one ring is supported, it is not named. Lattice definitions are the lists of elements where physical properties are not attached. Beam lines and a ring are the objects of Goemon.

2.5 GENERAL FUNCTION CALLS

As various kinds of mathematics routines including fitting and graphics routines are supported by O-Matrix itself, TracyM can focus on the physics part. It wraps most of the Goemon routines so that the virtual machine can be manipulated by O-Matrix programs freely at run time. A MATLAB-like convention that treats vectors and matrices like scalars simplifies the expression effectively.

5 EXAMPLES

Here is a simple example of a TracyM program written in the O-Matrix language as shown in Fig.2. This program calculates and displays the basic parameters of the ALS booster ring.

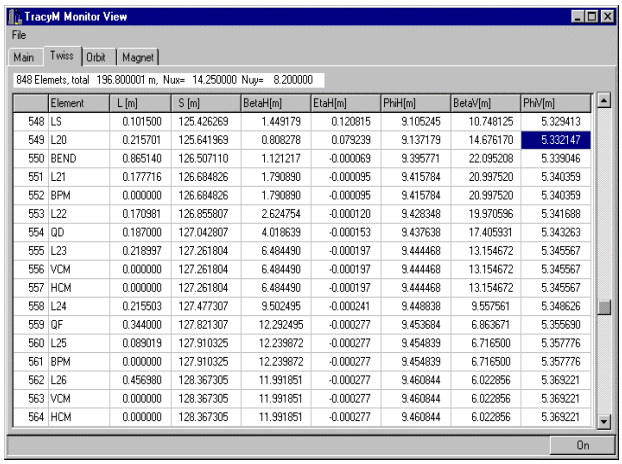
```
clear
include TracyM.mat # load DLL
initTracy

SYM=marker("")
LBF=drift("LBF",0.496875)
LBD=drift("LBD",0.546875)
LB =drift("LB ",1.05)
## Bending
BU =bend("BU",0.525,7.5,7.5,0.0, 0.0)
BD =bend("BD",0.525,7.5,0.0,7.5, 0.0)
BB =[BU,BD] # a beam line defined as a list
## Quadrupole
F2 =quad("F2",0.15, 2.7682214)
D2 =quad("D2",0.10,-2.5401249)
## Construct One Superperiod
DBF =[D2, LBD , BB , LBF , F2]
FBD =[F2, LBF , BB , LBD , D2]
FLD =[F2, LBF , LB,  LBD , D2]
DLF =[D2, LBD , LB,  LBF , F2]
## Ring
BL1 =[SYM,DBF,FBD,DBF,FLD,DLF,FBD,DBF,FBD]
ring(BL1,4) # define a ring / symmetry = 4

setQforTune(F2,D2)
getTwiss(1)

PrintTwiss

ginit
x=getS
y=getBetaX
gplot(x,y)  # plot BetaX
y=getBetaY
gplot(x,y)  # plot BetaY

format int "5"
format double "e10.2"
calcIntegral # get synchrotron integrals

for n=1 to 30 begin
  E = n*50*1.0e6
  emt=calcEmittance(E)
  taux=DampingTimeH(emt)
  Emit0=Emittance(emt)
  print n*50, taux, Emit0
end

terminateTracy
```

Fig.2 TracyM Example Program

6 ADVANCED FEATURES

O-Matrix supports graphics programming including the support for GUI. Therefore, it is possible to program a customised GUI application entirely in O-Matrix. On the other hand, as the kinds of graphics or GUI for most of the accelerator modeling and simulation have been established, it will be more convenient to prepare a standard set of GUI dedicated to the use with TracyM.

A program MONITOR VIEW was developed for this purpose. It is an external program that communicates with TracyM and dynamically displays the status of the virtual machine created inside TracyM. It runs outside of the O-Matrix environment and reads the machine parameters from a shared memory implemented by using WIN32 file mapping. Fig.3. shows the page displaying the Twiss parameter table.



	Element	L [m]	S [m]	BetaH[m]	EtaH[m]	PhiH[m]	BetaV[m]	PhiV[m]
548	LS	0.101500	125.426269	1.449179	0.120815	9.105245	10.748125	5.329413
549	L20	0.215701	125.641969	0.808278	0.079239	9.137179	14.676170	5.332147
550	BEND	0.865140	126.507110	1.121217	-0.000069	9.395771	22.095208	5.339046
551	L21	0.177716	126.684826	1.790890	-0.000095	9.415784	20.997520	5.340359
552	BPM	0.000000	126.684826	1.790890	-0.000095	9.415784	20.997520	5.340359
553	L22	0.170981	126.855807	2.624754	-0.000120	9.428348	19.970596	5.341688
554	QD	0.187000	127.042807	4.018639	-0.000153	9.437638	17.405931	5.343263
555	L23	0.218997	127.261804	6.484490	-0.000197	9.444468	13.154672	5.345567
556	VCM	0.000000	127.261804	6.484490	-0.000197	9.444468	13.154672	5.345567
557	HCM	0.000000	127.261804	6.484490	-0.000197	9.444468	13.154672	5.345567
558	L24	0.215503	127.477307	9.502495	-0.000241	9.448838	9.557561	5.348626
559	QF	0.344000	127.821307	12.292495	-0.000277	9.453684	6.863671	5.355690
560	L25	0.089019	127.910325	12.239872	-0.000277	9.454839	6.716500	5.357776
561	BPM	0.000000	127.910325	12.239872	-0.000277	9.454839	6.716500	5.357776
562	L26	0.456980	128.367305	11.991851	-0.000277	9.460844	6.022856	5.369221
563	VCM	0.000000	128.367305	11.991851	-0.000277	9.460844	6.022856	5.369221
564	HCM	0.000000	128.367305	11.991851	-0.000277	9.460844	6.022856	5.369221

Fig 3. MONITOR VIEW

It can also send commands to TracyM to manipulate the virtual machine by using a WIN32 named pipe. In this case, TracyM runs in a server mode and takes commands from this program.

7 CONCLUSION

By using O-Matrix as a front-end for the C++ class library Goemon, TracyM provides an interactive and programmable environment for accelerator simulation and modeling studies without sacrificing the execution speed. A rich set of mathematical and graphical functions of O-Matrix has made various kinds of parameter studies much easier and smoother. An option is a separate GUI program that runs in parallel with TracyM and enables a highly interactive operation.

REFERENCES

[1] H. Nishimura, " Taking an Object-Oriented View of Accelerators“Paper”, PAC95, 2162.199.

[2] H. Nishimura, "TRACY, A Tool for Accelerator Design and Analysis", EPAC 88,803,1989.
J.Bengtsson, E.Forest and H.Nishimura, "Tracy Users Manual",unpublished.

[3] H. Nishimura, " Accelerator Modeling and Control Using Delphi on Windows NT", IWCSMSA96, KEK Proceedings 97-19, 174.

[4] The Mathworks, Inc. Natic, MA., U.S.

[5] G. J. Portmann,"ALS Storage Ring Setup & Control Using MATLAB", LBNL ALS/LSAP-248,1998.
G. J. Portmann,"Recipe for ALS Storage Ring Operation", LBNL ALS/LSAP-249,1998.

[6] Harmonic Software, Inc. Seattle, WA., U.S.