Implementation of One-Turn Maps in SSCTRK using ZLIB

S.K. Kauffmann, D.M. Ritson, Y.T. Yan
Accelerator Division
Superconducting Super Collider Laboratory*
2550 Beckleymeade Avenue
Dallas, Texas 75237

Abstract

The particle tracking code SSCTRK is currently being adapted to operational simulation and beam-beam effect studies for the Collider rings of the Superconducting Super Collider (SSC). During beam-beam effect studies, the lattice content of the bending arcs is normally not varied, making fast truncated Taylor map tracking through the arcs an attractive option. The implementation of SSCTRK as a truncated Taylor map tracking program has been carried out using the differential algebra library ZLIB, which simplified the task to that of straightforward translation of SSCTRK kick and drift arithmetic operations to calls to the corresponding polynomial operation subroutines of ZLIB. The accuracy and speed (relative to normal SSCTRK tracking) of truncated Taylor map tracking at 2 mm betatron oscillation amplitude was studied in various orders of the map. The seventh order map was found to be in agreement with the normal SSCTRK to about eight significant figures on the first turn, and to a fraction of 1% on the 100,000th turn, for a typical 5 cm magnet aperture lattice, and could be made to track at ten times the speed of the normal SSCTRK kick-drift tracking on a scalar architecture (Sun) workstation. (The map tracking subroutines of ZLIB are optimized for vector and parallel architecture supercomputers, and typically achieve even faster relative performance on these, but operational simulation studies will be more conveniently carried out on dedicated workstations which have the incoming generation of "superscalar" CPUs.)

I. Introduction

The differential algebra library ZLIB [1] offers analytic multivariable polynomial manipulation and truncated Taylor map tracking which permit accelerator Taylor map construction, tracking, and analysis to be carried out with ease and efficiency under the framework of an IMSL-style user library. ZLIB is divided into two sublibraries, ZPLIB and TPALIB. ZPLIB is oriented toward speed and efficiency on vectorizing or parallelizing (multitasking) supercomputers, while the more straightforward subroutines of TPALIB generally run a little faster on scalar architecture computers. The fast, thin-concatenated-element, kick-drift tracking code SSCTRK (written for the purpose of tracking proton beams through the Collider rings of the SSC) is in the process of being reconfigured for operational simulation and beam-beam effect studies in the SSC, to be carried out on very fast, dedicated workstations which have the incoming generation of "superscalar" CPUs. As it is usual to hold the lattice content of the bending arcs fixed during beam-beam effect simulation studies, fast tracking with a truncated Taylor map through those arcs at a maximum betatron oscillation amplitude of about 2 mm is considered to be an attractive option. Below we discuss the translation of the kick-drift tracking section of SSCTRK into one-turn truncated Taylor map construction code through use of the ZLIB sublibrary TPALIB. The accuracy of the various map truncation orders is evaluated vis-a-vis the corresponding SSCTRK kick-drift results at 2 mm betatron oscillation amplitude over 100,000 turns for a typical SSC 5 cm magnet aperture lattice, and the speed advantage of the acceptably accurate seventh order map on a scalar CPU (Sun) workstation over the original SSCTRK kick-drift code is presented.

II. Translation Of SSCTRK Kick-Drift Tracking To Taylor Maps Using ZLIB

Unlike the Teapot [3] tracking program as well as the post-Teapot one-turn map extraction program Zmap [4], which contains hundreds of statements and involves square roots and trigonometric functions, the kick-drift tracking section of SSCTRK—consisting of just a few dozen lines of highly optimized, simple code which involve essentially only additions and multiplications—allows economic implementation of one-turn maps. With the help of the ZLIB User's Guide, [1] these arithmetic operations are straightforwardly translated into calls to ZLIB subroutines which perform the corresponding polynomial operations. For example, the basic (FORTRAN coded) vertical kick in SSCTRK,

```
PH = PH - RIGITY * IMPOL,
```

translates to one polynomial multiplication subroutine call to the TPALIB sublibrary of ZLIB, followed by one polynomial subtraction subroutine call to that sublibrary,

```
CALL TPAMUL(FRIG,FIMPOL,FTMPO,5) followed by
CALL TPASUB(FPH,FTMPO,FPH,NM).
```

Notice that the polynomial FRIG corresponds to the quan-

*Operated by the Universities Research Association, Inc., for the U.S. Department of Energy under Contract No. DE-AC02-89ER40486.

tity RIGITY, the polynomial FIMPOL corresponds to the quantity IMPOL, and the polynomial FPH corresponds to the vertical betatron oscillation angle PH. In addition, we store the intermediate result corresponding to the "kick" product RIGITY*IMPOL in the temporary polynomial FTMPO, which, in the second call, is subtracted from the polynomial FPH to "kick update" it. The last argument "5" of the call to subroutine TPAMUL above is the number of variables of which all of our polynomials are functions, while the last argument NM of the call to subroutine TPASUB above is the number of linearly independent terms in each of our polynomials (this is determined by their number of variables, i.e., 5, and their common order—note that TPAMUL above will truncate its polynomial multiplication result FTMPO to that common order NO, which needs to have been previously specified in an initial call to the TPALIB "preparation" subroutine TPAPRP(5,NO,NM)).

Not every quantity occurring in the kick-drift tracking section of SSCTRK is translated into a polynomial. Any quantity which remains fixed turn-by-turn around the Collider is treated as a constant in the polynomial algebra. Thus the basic SSCTRK vertical drift,

```
Y = Y + LO * PH,
```

contains the fixed drift length L0, which is treated as a constant rather than a polynomial. Of course, the vertical betatron oscillation displacement Y and angle PH are treated as polynomials FY and FPH (all such polynomials are represented in the FORTRAN code as single-index, double-precision arrays of length NM, while the constants are represented as just double-precision unindexed variables). This basic vertical drift is taken care of by a single TPALIB subroutine call which simply updates the polynomial FY as the appropriate linear combination of FY and FPH,

```
CALL TPALIN(FY,LO,FPH,FY,NM).
```

We could manage to do with polynomials of five variables rather than the general accelerator physics set of six variables in this instance because we did not include ripple or noise effects and needed only to update the final longitudinal displacement after a full turn through the arcs and RF cavity by an additive term which does not depend at all on the initial longitudinal displacement (we did not actually include the RF cavity in our polynomial mapping of the arcs, as it would have cost accuracy for a negligible saving of tracking time). Since the variables of which our polynomials are functions correspond to the irreducible set of turn-by-turn varying initial values at the beginning of each turn, we can effectively drop the longitudinal displacement as such a variable. The change in the longitudinal displacement is, however, calculated as a polynomial depending on the other variables because its value after one turn through the arcs is needed by the RF cavity section of the code. The momentum displacement, on the other hand, must be counted as a variable even though it does not change at all in the arcs—it undergoes a complicated change on every turn in the RF cavity. The remaining four variables are, of course, the vertical and horizontal betatron oscillation displacements and angles. The TPALIB subroutine TPAPOK1 (polyn, const, var no, NM) initializes polyn as const times the variable associated with var no, which indexes the initial value array that is to be used for map tracking. Thus, after initializing TPALIB with CALL TPAPRP(5,NO,NM), we initialize those polynomials which are to begin the map tracking turn as one of the five pure variables having values equal to those of the appropriately indexed members of the initial value array XINPT, e.g.,

```
CALL TPAPOK1(FX,1.0D+0,1,NM),
CALL TPAPOK1(FTH,1.0D+0,2,NM),
```

and also

```
CALL TPAPOK1(FY,1.0D+0,3,NM),
CALL TPAPOK1(FPH,1.0D+0,4,NM), etc.
```

The initial value array for the beginning of the turn is FORTRAN dimensioned as XINPT(5), and the above statements will initialize the polynomial FX to a particular unit coefficient first-order monomial which will have the value XINPT(1) at the beginning of the turn, the polynomial FTH to another unit coefficient first-order monomial which will have the value XINPT(2) at the beginning of the turn, the polynomial FY to yet another unit coefficient first-order monomial which will have the value XINPT(3) at the beginning of the turn, the polynomial FPH to yet a fourth unit coefficient first-order monomial which will have the value XINPT(4) at the beginning of the turn, etc.

From the above examples of code we see that the translation of the tracking section of SSCTRK into a polynomial mapping through the use of the TPALIB sublibrary of ZLIB was relatively straightforward and intuitive (the full code produced is given in Appendix A of Reference [2]). In the map construction subroutine OPSTPA, the original kick-drift code statements from SSCTRK subroutines OPSTRK and PTRACK are given in commented lines just before their translation into TPALIB subroutine calls, which it may be instructive to study in conjunction with the reading of the ZLIB User's Guide. Two "unofficial" modified TPALIB routines were incorporated into the code, the map writeout (to a file) routine WTPKMAP and the specialized map tracking routine TPKMTRK5, which sacrifices generality in the number of variables and the number of output polynomials (both frozen at five) for maximum map tracking speed.

III. Comparison Of The ZLIB Translated Map Tracking With SSCTRK Itself

The one-turn truncated Taylor map construction code for the SSCTRK lent itself to a precise check for the correctness of the map in the special case of a purely linear lattice (the number of multipoles PPOLES=1, the relative momentum offset DELP=0, and the RF voltage PRFV0=0, in the language used in subroutines OPSTPA and PRFACC). In this instance the results of the first-order one-turn map agreed precisely (essentially to all the digits of the double-precision accuracy used) with the results of the original SSCTRK kick-drift code when tracked through a single turn. This perfect correlation of the purely linear lattice with the corresponding first-order map for one turn is a useful consistency check for the correctness of mapping translations carried out with the aid of ZLIB.

The remaining evaluation of the TPALIB driven mapping translation found in subroutine OPSTPA was done with a typical SSC lattice for 5 cm magnet aperture (e.g., the number of multipoles PPOLES=6) with a realistic value for the relative momentum offset DELP (5×10^{-4}), and the RF voltage turned on. All tracking was done with the invariant horizontal and vertical amplitudes each initialized at 2 mm, and various truncation orders of the map were compared with the original SSCTRK kick-drift code result over 100,000 turns. Typically, the combined horizontal-vertical invariant (which is not sensitive to horizontal-vertical coupling) averaged over 5000 turns was monitored. Under these circumstances the second order map behaved unacceptably, with its 5000-turn-averaged combined horizontal-vertical invariant increasing at an accelerating pace with turn number (this invariant was essentially constant for the original SSCTRK kick-drift code), and the particle leaving the beam tube aperture entirely before it had gone 20,000 turns. The third-order map behaved much more reasonably, though it displayed rather the opposite behavior from the second order map—the horizontal-vertical invariant tended to become smaller with turn number rather than remain constant as it ought to have (in accord with the original SSCTRK kick-drift code behavior), i.e., the particle spiraled inward (see Figure 1). Figure 1 indicates the fourth-order map to have substantially the same behavior as the third-order map, albeit marginally worse. The fifth-order map was a vast improvement (again see Figure 1) and the sixth-order map again showed marginal deterioration relative to the fifth-order (both are sufficiently good that this is hard to follow on Figure 1 without a magnifying glass). This odd-even order "great improvement—marginal deterioration" was a consistent feature of the lattice we were testing at 2 mm betatron oscillation amplitudes, but likely was a peculiarity of that lattice rather than a general property of truncated Taylor maps. The "great improvement" of the seventh order map was sufficient to bring the combined horizontal-vertical invariant within one part in 20,000 of that of the original SSCTRK kick-drift code, which difference cannot be resolved within the thickness of the line in Figure 1. Further, on the 100,000th turn itself the horizontal betatron oscillation was still faithful within a fraction of one percent to that of the original SSCTRK kick-drift code, from which the map had, of course, been translated. (After the first turn, the agreement in horizontal betatron oscillation is to about eight significant figures.)

This very acceptably accurate (for use with the bending arcs in beam-beam effect simulation studies) seventh-order map executed a factor of ten faster on the scalar architecture Sun workstation than did the original SSCTRK kick-drift code when the special purpose map tracking subroutine TPKMTRK5 was used. Subroutine TPKMTRK5 is related to the subroutine TPAMTRK of TPALIB, but sacrifices generality in the number of inputs and outputs (with both frozen at five) to speed. On supercomputers the corresponding subroutines from ZPLIB, which are optimized for vectorization and multitasking (parallelization), could be expected to have an even greater relative speed advantage.

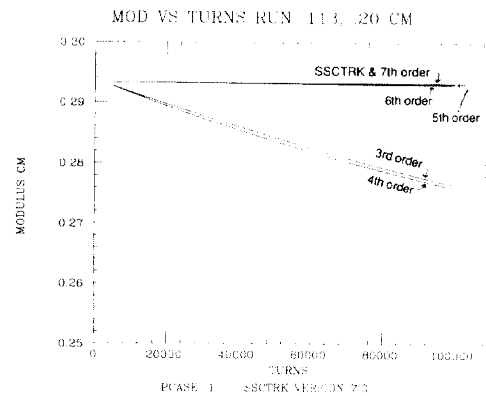

Figure 1: Comparison of the combined horizontal-vertical invariant for the truncated Taylor maps of order three through seven with that of the original SSCTRK kick-drift code for initial 2 mm amplitude over 100,000 turns.

IV. Conclusion

This use of ZLIB to effect the translation of the tracking section of the SSCTRK kick-drift code to mapping construction and map tracking code shows that ZLIB is indeed a user-friendly, effective, and efficient way to build and track truncated Taylor maps. The unique and powerful vectorization and parallelization features of ZLIB (in the sublibrary ZPLIB) were not used in this particular application, nor were ZLIB's map analysis features, but the easy-to-use, familiar (from its IMSL-like framework), and efficient features of ZLIB made this mapping application one that was easily and quickly carried out.

V. References

[1] Y. Yan and C. Yan, "ZLIB: A Numerical Library for Differential Algebra (User's Guide)," SSC Laboratory Report SSCL-300, July 1990.

[2] S.K. Kauffmann, Y.T. Yan, and D.M. Ritson, SSC Laboratory Report SSCL-321 (1990)

[3] L. Schachinger and R. Talman, "Teapot: A Thin-Element Accelerator Program for Optics and Tracking," Particle Accelerators, Vol. 22, (1987)

[4] Y. Yan, "Zmap: A Differential Algebraic High-Order Map Extraction Program for Teapot using ZLIB," SSC Laboratory Report SSCL-299, 1990.