COMPUTATION OF LATTICE MAPS USING MODULAR BCH AND SIMILARITY COMPOSITION RULES

J. Irwin, Stanford Linear Accelerator Center, Stanford University, Stanford, CA 94309 USA

A beam line map construction method for linear and circular high-energy colliders is described which avoids truncated power-series maps through systematic use of a two-term Baker-Campbell-Hausdorff (BCH) formula in combination with similarity transformations. The beam line map ultimately assumes the form of a product of a linear map and a single-exponential Lie-operator map. The method i) provides insight into map generator sources, ii) is accurate, iii) is complete in that all effects, such as edges and soft fringes, mis-alignments and mis-powerings, multipole errors, and input beam errors can be simply included, iv)permits faster map computation times, and v) bypasses truncated power-series map methods allowing for higher order, and even non-polynomial generators.

I. INTRODUCTION

There are many possible motives for creating and using maps in conjunction with accelerator lattice design and analysis, and the method chosen for creating the map will depend on this motive. We describe four distinct primary motives.

1. **Structure.** It is highly desirable to have maps that correspond exactly with the algorithm used for single particle tracking. This allows the user to establish the validity of these maps by comparing map tracking with element-by-element tracking. The map creation algorithms developed by E. Forest and collaborators, represented in the code DESPOT [1], have focused on this objective. A symplectic integrator must be found for each lattice element or sub-element to represent it in element-by-element tracking. The map creation can then be carried out by "tracking" a power series through the lattice. Reliability is also enhanced by the methodical nature in which elements are pieced together, referred to by Forest as “lego like”. The resultant power series map for the lattice is guaranteed to be symplectic and may be exponentiated, if desired, by using the Dragt-Finn factorization algorithm [2]. The major disadvantage of this method is that it becomes very computer intensive for large lattices, high order or many additional variables. However tools in the LIELIB package which accompanies DESPOT do allow the user to implement other map composition methods.
2. **Accuracy.** A second motive places primary emphasis on machine precision accuracy . The programs developed by A. Dragt and represented by MARYLIE [3] and M. Berz in COSY-Infinity [4] are examples. MARYLIE uses a generator based map-concatenation in which each element or element part is represented by an ordered generator-based factorization, and map concatenation is based on an algorithm that can take the product of two such factorizations and produce a single factorization of the same form. The program "Genmap" accomplishes this task for continuously varying Hamiltonians, such as those occurring in fringe-field regions. To do a corresponding element-by-element tracking requires use of mixed variable generators for each element. Coresspondence with maping is not exact. The COSY-Infinity uses power series maps by direct expansion of the original Lie generators, operating on coordinates or polynomials.
3. **Insight.** A third motive, is the creation of maps in a transparent way, that yield analytical results and provide insight into lattice function. In principle analytical results can be obtained by the previous methods, but a BCH based map concatenation provides better insight into map composition. BCH composition and has been carried out by N. Walker in the code LAMA [5] using a symbolic manipulation program. This method is especially useful for lattice modules such as linear collider final focus or collimation systems. One can allow all lattice and error parameters as well as incoming beam conditions to be variables. This method is far too slow for use with large lattices.
4. **Speed.** Speed becomes an issue in a large lattice design project because the number of lattice variations encountered is so very large: variations of a multitude of error types and strengths; inclusion of fringes, kinetic nonlinearities, and/or parasitic crossings; changes in section phase relations, changes in chromatic correction techniques, tune-shift-with-amplitude control, insertion devices, solenoid with skew compensation schemes, beam line geometry changes, and so on. For each of these lattice changes one would like to have a map to assess performance with a tune-space scan [6] . Another advantage of a fast map algorithm is the ability to introduce many lattice parameters as variables, and fit or optimize these variables to achieve desired aberration coefficients or performance. A map composition

process whose primary motive is speed is described below. It is based on experience with method 3 above, and hence can also provide insight into lattice dynamics.

II. MAP COMPOSITION METHOD

We summarize the steps of a composition method based on simlilarity transformatins and a low order BCH composition formula. The first six steps describe the conceptual setup. Compustation begins with step 7.

1) Represent each element as an infinite Lie product corresponding to cutting the elements into thin slices. The generators will be s-dependent when necessary, as for example in fringe regions. Lie maps may be inserted at the ends of the magnets to provide for translations and rotations of the magnet [7].
2) In each slice, separate the linear design part of the generator from the remainder and write the map of each slice as a product of two maps. This is possible because the generator strengths are infinitesimal.
3) Use similarity transformations to move all design linear maps (2nd order in transverse variables) to the front of the beam line product. This can be accounted for by replacing x (or y, or p_x or p_y) in each slice by x_i where it is understood that x_i is x at the ith element written as a linear function of the particle position and momentum at the end of the beam line.
4) Use a 2nd order BCH algorithm to integrate the map factors for each element into a single generator [8]. If the element is particularly long or strong, this can be done for sections of the element rather than the whole element. If fringes effects are to be considered, the integral for the fringe region is done by first expressing the x_i as function of position and slope at a place within the fringe [9], and later expressing the coorcinates there as functions of coordinates at the end of the line. To do the body integral, x_i is written as a function of the momentum and position at the center of the element. The results for the body integrals will be a function of x_k and y_k the transverse coordinates at the center of the kth element, plus a function which depends also on $p_{x,k}$ and $p_{y,k}$.
5) Factor the map for the element body integrals found above into a central term surrounded by two side factors so that the central term has a generator containing terms depending on x_k and y_k alone.

Now “big” terms are either in a central factor, which depends only on transverse coordinates (hence are kicks), or in the translation and rotation generators surrounding the element. The central factors can be totally factored because all terms in the generator “commute” (have zero Poisson brackets with one another). The translations and rotations are steering elements: For translations the generator is $\Delta y\, p_y$, and for small vertical rotations the generator is $\Delta\theta(y+L/2\, p_y)$ where L is the length of the element. Next we remove the main dispersion generators.

6) Perhaps the largest of the body terms are the sources of design dispersion coming from the main dipole magnets. These terms can be removed using similarity transformations, very much like the design linear terms. The net effect of removing these terms will be that x_k and $p_{x,k}$ are replaced by $x_k+\eta_{x,k}\delta$ and $x_k+\eta'_{x,k}\delta$ where $\eta_{x,k}$ is the horizontal design dispersion.

We now have the basic factored representation of the lattice with the linear design, including linear dispersion, removed. We proceed to consolidate this representation into a few factors after taking care of the steering terms, which can be quite large.

7) Starting at the end of the beam line, use similarity transformations to move all the first-order (steering) generators to the front of the beam line. Itermediate transformed generators will have feed down terms some of which will be steering terms. For the large central factors these are easily factored out and added to the steering generator. This is not as simple for the side-factors. If the feed down steering in the side factors is thought to be significant, these factors can be further factored, pulling the steering terms into side factors. One then brings the two steering terms to the front side by a subsequent similarity transformation. Further feed down terms are formed, but these are now due to steering from the side factor itself, and can be assumed to be small. (Solenoid fringes can contain important steering terms, but these are already present in the linear model.) Since the exact values of steering correctors throughout the ring are not exactly known, there is a limit to the precision one can or should attempt to achieve. It should also be noted that one must implement a steering correction algorithm to determine corrector strengths before embarking on the map generation process, or algorithms can be used which calculate the steering corrector strengths during this step.

We are ready to begin the map consolidation process. The exact procedure must be chosen to suit the particular situation. If there are especially large sextupole and/or chromatic correction terms, these should be flagged for a similarity composition process and will define ends of a module. In the next step the maps for the modules are assembled.

8) The lattice will consist of several modules which may either be what is normally understood as a module, like an arc, a straight section, a chromatic correction section, a beta-match section, and so on, or just the lattice between two large higher order elements that one intends to collapse using the similarity composition rule.

   In this step the map for each module is assembled either as a single exponential map, or as a factored map consisting of a product of three maps: the first with a generator that is of first order in transverse variables (dispersion terms), the second with a generator of 2nd order in transverse variables (linear terms), and the third whose generator contains higher order terms. For a factored map the dispersion terms are found for the module using a similarity process to move these terms to the front of the module as was described for steering (7) or dispersion (6) above. This is followed by using a similarity process to move the linear terms to the front.

   The final step uses the BCH formula to assemble all remaining generators into a single generator. Note that all of the large terms have been (or will be) handled with similarity transforms. Thus the BCH process is expected to converge quite rapidly, and in most cases a second order BCH formula is sufficient. A third order BCH formula may be used to check the sufficiency of the second order formula. It is important to distinguish BCH order from map order. Map order can be, an is, much higher than BCH order.

9) Strong sextupole terms in the lattice are "collapsed" using similarity transformations. The details of this will depend on the design phases between the sextupoles. Use of the similarity transformation is especially important in systems such as final focus systems, or in local chromaticity correction modules of low beta insertions in storage rings. If strong sextupoles are interleaved, special attention is required. One collapses the paired sextupoles, transforming the enclosed sextupole. The transformed sextupole generator is now factored with the central factor being the sextupole and the remainder placed in the side factors. If the interleaving is too strong to do this in one step, the original enclosed sextupole is factored into two halves, and each half is factored as described above. This process converges quite rapidly: the approximation represented by the three factors improving by a factor of 8 with each halving of the sextupole.

10) Strong chromaticity terms are "collapsed" using similarity transformations. This is very similar to the process. of step 9). Many of the strong chromatic generators will be at the sextupole locations. Additional generators of this type will be in final doublets. The doublets will have been split into several pieces in the integration of those elements.

11) Use the BCH rule to assemble the generator for the various beam line modules.

12) Use the BCH rule to assemble the generator for the total beam line.

IV. REFERENCES

[1] E. Forest, The correct local description for tracking in rings, Part. Accel. 45, p65 (94). There is no published manual for the program DESPOT which is being used at SLAC for PEP-II lattice design.

[2] A. Dragt and J. Finn, J. Math. Phys. 20, 22649 (1979).

[3] D. Douglas and A. Dragt, MARYLIE, the Maryland Lie Algebraic Beam Transport and Particle Tracking Program, IEEE Trans. Nucl. Sci. NS-30, 2442 (1983).

[4] M. Berz, New features in COSY Infinity, In Los Alamos 1993, Proceedings, Computational accelerator physics, p267 (94).

[5] N. Walker, J. Irwin, M. Woodley, Analysis of higher order optical aberrations in the SLC final focus using Lie algebra techniques, SLAC-PUB-6205 (93) and Proceedings of the 1993 Part. Accel. Conf (PAC93.)

[6] J. Irwin, T. Chen, and Y. Yan, A fast tracking method using resonance basis Hamiltonians, SLAC-PUB-95-6727, and Conference on nonlinear dynamics in particle accelerators proceedings, Arcidosso, Italy (94). Also Y. T. Yan et. al., Swamp plots for PEP-II lattices, RAA26, these proceedings.

[7] For a further exposition of the notation we use as well as Hamiltonian and generator details see J. Irwin, to be published in Nucl. Instr. and Methods, and SLAC-PUB-___.

[8] J. Irwin, The application of Lie algebra techniques to beam transport design, Nucl. Inst. and Methods, A298, 460 (90).

[9] C-X Wang and J. Irwin, Explicit maps for soft-edge fringe fields, MPC31, these proceedings