HIGH-ORDER REPRESENTATION OF POINCARE MAPS

JOHANNES GROTE, MARTIN BERZ, AND KYOKO MAKINO

Extended abstract

1. Introduction

For many questions in dynamical systems theory it is desirable to have Poincare maps available as a tool to study properties like stability or long-term behavior of the dynamical system under consideration, e.g. the flow generated by an ordinary differential equation. A Poincare map essentially describes how points on a plane (the Poincare section) in phase space which is transversed by a periodic or almost periodic orbit (the reference orbit) of the flow get mapped back onto the plane by the flow, provided they are sufficiently close to the reference orbit. Questions e.g. about the asymptotic stability of the reference orbit can equivalently be answered by analyzing the action of the Poincare map on these points on the plane. One of the key benefits is that the problem which we need to analyze has a dimensionality which has been reduced by one, since the trivial direction of motion along the reference orbit is neglected by restricting the analysis to the Poincare section.

In the numerical treatment of these problems one is faced with the question which numerical representations of a flow are particularly favorable in that they easily allow the computation of corresponding Poincare maps for a given reference orbit and Poincare section. In this paper we will show that high-order polynomial approximations of the flow, which have been obtained using differential algebraic (DA) tools as in [1][2], allow a direct deduction of polynomial approximations of Poincare maps of a certain type. We want to put particular focus on the case where the flow under consideration has been generated by an ODE.

We remark that the proposed algorithm is a part of an extended method which allows the computation of rigorous interval enclosures of the polynomial approximation of the Poincare map discussed here.

2. The algorithm

The DA tools which are necessary to appreciate the method are described in detail in[1]. There it is demonstrated how one can use these tools to obtain high-order polynomial representations of the flows of a given ODE $\frac{d}{dt}x(t) = f(x(t))$, where $f : \mathbb{R}^\nu \supset D \longrightarrow \mathbb{R}^\nu$ is a function which can be represented as a composition of intrinsic functions which are defined in the DA context, which means that f has sufficient smoothness. The polynomial approximation $\varphi(x_0, t)$ is then given in terms of a so-called DA vector which depends on the chosen initial conditions $x(t_0) = x_0$ as well as on the independent time coordinate t. We will assume that

Date: April 9, 2004.

Key words and phrases. Poincare map, Differential Algebra .

such a polynomial representation $\varphi(x_0, t)$ of the solution of our ODE at hand has been obtained.

In the method we will concentrate on the special case that the Poincare section S under consideration is of the form $S := \{y \in R^v : y_1 = 0\}$. We expect that in the near future a generalization of this idea can be published where the class of surfaces which can be analyzed as a Poincare section is greatly increased. Whereas it is a significant restriction that S has the form of a plane where exactly one component of its elements vanishes, the fact that this is the first component does not mean any loss of generality, since any plane S'through the origin can be reduced to the case where $y_1 = 0 \ \forall y \in S'$ by the application of a suitable rotation.

The general goal of the method is to define a Poincare map which is acting on the section S using information available from the DA vector representation of the flow. For every possible initial condition, we wish to derive an expression of the crossing time $t_c(x_0)$ at which the trajectory originating at the said initial value traverses the section S, and then reinsert this time $t_c(x_0)$ back into the DA vector $\varphi(x_0, t)$ describing the flow. This yields a polynomial $\varphi(x_0, t_c(x_0))$ only depending on the initial conditions x_0 which projects these values almost exactly onto the Poincare section, up to accuracy restrictions depending on the approximation order. The information about this crossing time is contained in the flow in an implicit way, hence we need to use suitable algorithms of functional inversion in the DA context, which will be reviewed in detail in this paper. The function $\varphi(x_0, t)$ as such cannot be invertible, since the dimensionality of its domain and range do not even agree. Instead, we will introduce an auxiliary function $\psi(x, t)$ which is substantially easier to handle and yields all relevant results. For this we remark that the "interesting" direction as far as the Poincare map goes is the one along the first coordinate axis e_1, and this will be the only one we consider. Furthermore, for $\psi(x, t)$ to be invertible in the first place we need ψ to map into $\mathbb{R}^{\nu+1}$. This motivates the following

Definition 1. *(Auxiliary function for φ) Let φ be a polynomial representation for the flow under consideration. We then define the* ***auxiliary function*** ψ : $\mathbb{R}^{\nu+1} \supset D + I \longrightarrow \mathbb{R}^{\nu+1}$ *by* $\psi_1(x, t) := \varphi_1(x, t)$, $\psi_k(x, t) := x_k \forall k \in \{2, ..., \nu\}$ *and* $\psi_{\nu+1}(x, t) := t$, *where* $\varphi_1(x, t)$ *is the polynomial described by the first component of* $\varphi(x, t)$.

Essentially ψ contains the crucial part of the flow and is "filled up" with trivial identities in order to achieve invertibility. We remark that the setting up of this function actually represents the crucial point in the whole algorithm. Also, as one can see immediately from the Jacobian of ψ, ψ is invertible if and only if $\frac{\delta\varphi_1(y,t)}{\delta x_1}$ $\neq 0$ for y with $y_1 = 0$. This, however, is satisfied if indeed the orbits originating from points in D are transverse to the section S. We can now employ DA inversion tools to manipulate ψ and obtain the inverse $\psi^{-1}(x, t)$. Naturally, because of the identities in ψ,also ψ^{-1} will preserve these identities and hence only the first component $\psi_1^{-1}(x, t)$ is nontrivial. Once we have established $\psi_1^{-1}(x, t)$ we evaluate it at the point $y := (0, x_{0,2}, ..., x_{0,\nu})$ to solve for t in dependence on x_0 and call this result $t_c(x_0)$. Once this is achieved, a DA vector describing the Poincare map can be obtained through simple DA arithmetic by evaluating $\varphi(x_0, t_c(x_0))$.

3. An example: The Volterra-Lotka equations

In the following we give the results for a nontrivial two-dimensional example, the Volterra-Lotka equations, which model the time-evolution of two conflicting populations (predator-prey relation). They are given by the two-dimensional autonomous system $\frac{dx}{dt} = 2x(1-y)$, $\frac{dy}{dt} = -y(1-x)$, and the solution trajectories have to obey the constraint $C(x,y) = xy^2 e^{-x-2y} =$ const. In the quadrant characterized by $x_{1,2} > 0$, the contour lines are bounded and follow closed curves.

We consider the reference orbit with the initial conditions $(x_{ref}, y_{ref}) = (1,3)$, which results in a closed periodic orbit with a minimal period $T \simeq 5.488138468035$. The Poincare section S on which we want to project is given by $S := \{(x,y) \in \mathbb{R}^2 : x = 1, y \geq 0\}$, a vertical line just going through the initial conditions. We are considering initial values on S with a y_0-component in a $[-0.1, 0.1]$-interval around 3, and the tables show the result for the difference of the crossing time and the period T, and the final Poincare map for the y- and x components of the flow. The computation was executed in twentieth order, but for the sake of readability the coefficient tables have been truncated to show only coefficients up to fifth order in y_0.

First the result for the deviation of the crossing time from the period T is shown in dependence on the dislocation y_0 from the initial condition 3. The first entry should theoretically be zero and probably has a small error because the value for T used in the computation deviates slightly from its true value. However, the important part is the second term in the table, as it describes the linear change of the crossing time with the deviation y_0. Since $y_0 \in [-0.1, 0.1]$, the total value of the difference between the crossing time and the period T has the order of 0.01 over the box of initial conditions.

```
I  COEFFICIENT             ORDER EXPONENTS
1  0.7124867362762188E-10    0    0 0   0
2  0.1054540118235040        1    1 0   0
3  0.1029401891599718E-03    2    2 0   0
4  -.2242556468078237E-06    3    3 0   0
5  0.3641665396373908E-09    4    4 0   0
```

The next table shows the dependence of final values y_f on initial vertical deviation y. One can see that the reference orbit, corresponding to $y_f = 3$, is conserved, and that the deviation in final y-values depends mostly linearly on y_0:

```
I  COEFFICIENT             ORDER EXPONENTS
1   3.000000000000035        0    0 0   0
2  0.1000000000003481E-01    1    1 0   0
3  -.5540934121546351E-13    2    2 0   0
4  0.3063051404542127E-13    3    3 0   0
5  -.1007089366346760E-13    4    4 0   0
```

This linear dependence is to be expected, since the motion has an invariant; hence every launch point on the Poincare section is necessarily re-visited after one revolution.

The last table can serve as a cross check; it shows the dependence of final x-variable x_f on initial deviation y_0. The Poincare map is supposed to project initial values almost exactly back onto S, and this can easily be seen here. Up to minor inaccuracies, for all $y_0 \in [-0.1, 0.1]$ we have that $x_f(y_0) = 1$.

I	COEFFICIENT	ORDER	EXPONENTS		
1	1.000000000393896	0	0	0	0
2	-.7852022329538491E-13	1	1	0	0
3	-.2798715770022655E-15	2	2	0	0
4	-.6960822147332478E-17	3	3	0	0
5	-.3005105238605367E-19	4	4	0	0

References

[1] M. Berz. *Modern Map Methods in Particle Beam Physics.* Academic Press, San Diego, 1999. Also available at http://bt.pa.msu.edu/pub.

[2] M. Berz and K. Makino. Verified integration of ODEs and flows using differential algebraic methods on high-order Taylor models. *Reliable Computing*, 4(4):361–369, 1998.

Current address: Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48823, USA