JSIAM Letters Vol.5 (2013) pp.37–40 ©2013 Japan Society for Industrial and Applied Mathematics

A new geometric integration approach based on local invariants

Takeru Matsuda[1] and Takayasu Matsuo[1]

[1] Department of Mathematical Informatics, Graduate School of Information Science and Technology, The University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo 113-0033, Japan

E-mail *Takeru_Matsuda@mist.i.u-tokyo.ac.jp*

Received October 2, 2012, Accepted December 28, 2012

Abstract

In this report we propose a simple new geometric integration approach for solving ordinary differential equations based on the concept of "local" invariants. The approach basically belongs to the class of invariants-preserving integrations, but it differs from any existing methods in that it can automatically detect enough number of invariants, and work even for non-conservative systems. Numerical examples show that the approach can in fact work.

Keywords conservative method, local invariant, discrete gradient method

Research Activity Group Scientific Computation and Numerical Analysis

1. Introduction

In this paper, we show a simple new geometric integration approach for solving ordinary differential equations of the form:

$$\frac{\mathrm{d}z}{\mathrm{d}t} = f(z) \tag{1}$$

where $z(t) \in \mathbb{R}^N$ and $f : \mathbb{R}^N \to \mathbb{R}^N$.

ODE (1) often happens to have invariants such as energy and momentum, and for both of numerical stability and qualitatively better behaviors, it is generally encouraged to employ some special integrators that preserve the invariants. The most typical integrator of this class is the discrete gradient method (see, for example, [1,2]). The method (or any similar ideas) can be, however, only applied to the systems where invariants are explicitly known, and otherwise it is even impossible to write down the scheme itself. Although this is not a crucial restriction from a practical point of view, one may still ask a mathematically challenging question: *is it possible to preserve even unknown invariants?*

It is interesting to rephrase the question in the following slightly different and stronger way: intuitively, if we know one invariant we can essentially decrease the degree of freedom of the ODE by one. Similarly, if we know the $N-1$ (independent) invariants of ODE (1), we should be able to exactly trace the trajectory of the solution. This is extensively discussed in the field of integrable systems, but we would like to remind the readers that even for non-integrable systems it is still possible to consider enough number of, i.e., $N-1$ invariants in some weak sense that determine the trajectory. The invariants are called "local invariants" (see, for example, [3]). It seems that this concept has not come into play in the area of structure-preserving methods so far, to the best of the authors' knowledge, and that motivated our project below.

Our main idea in this report is to more actively utilize the local invariants to solve ODEs. Since they are not explicitly known in general, we commence by numerically detecting them. Then we will construct a numerical integrator that preserves the detected invariants by the discrete gradient method. The numerical examples below demonstrates that the idea can in fact work. Note that this report is to raise the mathematically simple idea, and not to discuss the actual efficiency of computation.

This paper is organized as follows. In Section 2 the concept of local invariants is introduced and its convenient features are briefly reviewed. In Section 3 the principle of the proposed method is explained. In Section 4 numerical examples are presented. Concluding remarks are given in Section 5.

2. Local Invariants

In this section the concept of local invariants is briefly reviewed. The description is based on Chapter 2 of [3]. First we review the conventional invariants—what we should call "global invariants"—of ODE.

Definition 1 (Global invariants) *A function $F : \mathbb{R}^N \to \mathbb{R}$ is called a global invariant of ODE* (1) *if for all $z \in \mathbb{R}^N$ and all $t \in \mathbb{R}$,*

$$F(\phi_t(z)) = F(z), \tag{2}$$

where ϕ_t is the flow of ODE (1)*.*

A typical example of the global invariants is the total energy when the ODE is a Hamiltonian equation. The concept of local invariants is a weaker version of the above definition.

Definition 2 (Local invariants) *Let U be a domain in $\mathbb{R}^N$. A function $F : U \to \mathbb{R}$ is called a local invariant of ODE* (1) *if for all $z \in U$ and all $t \in \mathbb{R}$ such that $\phi_t(z) \in U$,*

$$F(\phi_t(z)) = F(z). \tag{3}$$

In order to explain the existence result of the local invariants, let us recall the definition of the independence of the local invariants.

Definition 3 (Independence of local invariants) *Let $F_1,\dots,F_k$ be smooth real-valued functions defined on a domain U in $\mathbb{R}^N$. $F_1,\dots,F_k$ are called functionally dependent if for each $z \in U$ there is a neighborhood V of z and a smooth real-valued function $G : \mathbb{R}^k \to \mathbb{R}$ not identically zero on any open subset of $\mathbb{R}^k$, such that*

$$G(F_1(x),\dots,F_k(x)) = 0 \tag{4}$$

for all $x \in V$. $F_1,\dots,F_k$ are called functionally independent if they are not functionally dependent when restricted to any open subset $V \subset U$.

The flow ϕ_t of ODE (1) can be seen as a one–dimensional Lie group acting on $\mathbb{R}^N$. Then, from the general theory of Lie groups, we have the following theorem.

Theorem 1 (Maximal number of independent local invariants) *If ODE (1) has a unique solution, then for all $z \in \mathbb{R}^N$, there exist precisely $N-1$ functionally independent local invariants $F_1,\dots,F_{N-1}$ of ODE (1) defined in a neighborhood of z. Moreover, any other local invariant of the ODE (1) defined near z is of the form $G(F_1(z),\dots,F_{N-1}(z))$ for some smooth function G.*

For example, when $N=3$, there are two independent local invariants around any point in $\mathbb{R}^3$. The intersection of the two surfaces spanned by the local invariants corresponds to a piece of the exact trajectory (see Fig. 1).

We note that a function I is a (local) invariant of ODE (1) if and only if it satisfies

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \nabla I \cdot z_t = \nabla I \cdot f = 0. \tag{5}$$

For a given vector field f, this can be regarded as a linear PDE regarding the unknown function $I(z)$. In this respect, the above statement can be rephrased that if the PDE (5) has a solution (we do not get into the discussion of regularity, but simply assume it is smooth enough so that the following argument makes sense), then it should be an invariant. Moreover, if the solution is defined on entire $\mathbb{R}^N$, it is a global invariant. But of course this cannot happen in general (consider dissipative systems), and instead Theorem 1 guarantees the existence of $N-1$ functionally independent local solutions around any point in $\mathbb{R}^N$.

As an illustration, let us consider the damped harmonic oscillator:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & -0.1 \end{pmatrix}\begin{pmatrix} z_1 \\ z_2 \end{pmatrix}. \tag{6}$$

The blue curve in Fig. 2 shows the exact solution. This is a two-dimensional dissipative equation, and thus obviously does not have any global (non-trivial) invariant. However, from Theorem 1, ODE (6) still should have a local invariant; this can be understood in the following way. For this ODE, PDE (5) becomes

$$z_2\frac{\partial I}{\partial z_1} - (z_1 + 0.1 z_2)\frac{\partial I}{\partial z_2} = 0. \tag{7}$$

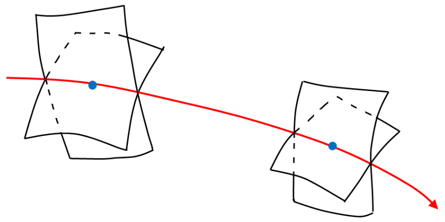

Fig. 1. Local invariants and the trajectory in $\mathbb{R}^3$.

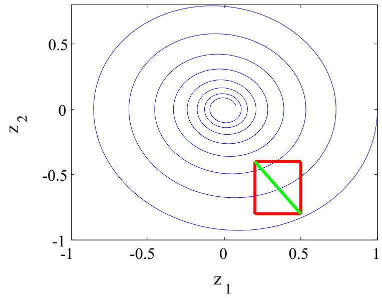

Fig. 2. Construction of a local invariant: the damped oscillator case.

From the general theory of advection equations, the solution of PDE (7) is uniquely determined on a neighborhood of a given point by giving initial values on initial surfaces which is not parallel to the trajectory everywhere [4] (note that the trajectory of ODE (6) corresponds to the characteristics of PDE (7)). For example, let us consider the small red rectangle in Fig. 2. Once the value of I on some initial surface, say the green diagonal line, is given, then the value of I in the whole rectangle can be uniquely determined. This implies the existence of a local invariant. If this procedure fails, then we can just take a smaller rectangle to retry the procedure. Our idea in this report is to do the procedure numerically to find enough number of local invariants by varying initial data.

3. Proposed Method

In this section we explain the principle of our new method. The method consists of the following steps:

- Step 1: Set a local computational domain to find local invariants,
- Step 2: Time-step while preserving all the local invariants,
- Step 3: In Step 2, when the time evolution approaches the boundary, reject the local computational domain and go back to Step 1.

We explain these steps in detail below.

3.1 Setting a local computational domain and finding local invariants

Before time-stepping we compute the $N-1$ local invariants around the current point, say z_n (n is the time step number; $z_n \simeq z(n\Delta t)$). We solve PDE (5) numerically to compute the local invariants. PDE (5) is of the form of advection equations, for which various efficient finite difference methods such as upwind scheme and CIP method are already known. If in the procedure below the

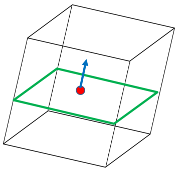

Fig. 3. Example of the computational domain and the initial surface.

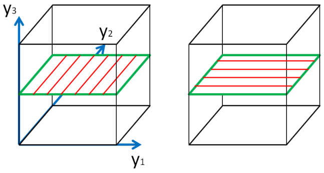

Fig. 4. Coordinate system and two initial data ($N = 3$).

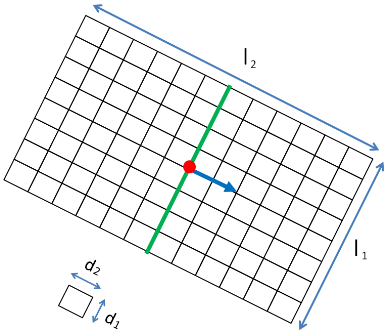

Fig. 5. Notation for the computational domain ($N = 2$).

value of I on points other than the mesh points becomes necessary, it is simply computed by linear interpolation.

In order to carry out the procedure, we first have to decide the computational domain and the initial condition for (5). As noted before, if the vector field f becomes parallel to the initial surface (the surface on which the initial value is given) in the computational domain, the computation should fail. So here we set the computational domain to a rectangular region which has z_n as the center point, and by carefully observing the direction of $f(z_n)$, we set an initial surface so that it includes z_n and is perpendicular to the vector field $f(z_n)$. An example in the case of $N = 3$ is shown in Fig. 3; z_n is denoted by the red point, the initial surface by the green rectangle, and $f(z_n)$ by the blue arrow. The black box shows the local computational domain.

Next we give $N-1$ initial values for PDE (5) on the initial surface. This should be done so that the resulting $N-1$ local invariants to be functionally independent. A simple choice is to adopt $N-1$ linear function on the surface: a 3-dimensional example is shown in Fig. 4. Consider a local coordinate system (y_1, y_2, y_3) of the local rectangular domain, and suppose the initial surface is perpendicular to y_3. Then we can simply choose initial data $I = y_1$ (the left panel; the red lines show the level sets) and $I = y_2$ (right).

Then we solve the PDE (5). Obviously the choice of the computational domain and the mesh–size should be appropriately determined. We discuss this point in the subsequent section. For the discussion, we use the notation shown in Fig. 5: l_i's are the length of the domain, and d_i's are the grid sizes.

3.2 Time-stepping with all the local invariants preserved

After the $N-1$ local invariants are obtained numerically, we solve ODE (1) by the discrete gradient method which is briefly reviewed below. Discrete gradient $\nabla_{\mathrm{d}} I$ is a function $\mathbb{R}^N \times \mathbb{R}^N \to \mathbb{R}^N$ which satisfies

$$I(y) - I(x) = \nabla_{\mathrm{d}} I(x, y) \cdot (y - x). \tag{8}$$

Discrete gradient is not unique, and several formulas are proposed [1,5]. With a discrete gradient fixed, a scheme can be written as

$$\frac{z_{n+1} - z_n}{\Delta t} = S(z_n) \nabla_{\mathrm{d}} I(z_n, z_{n+1}) \tag{9}$$

where $S(z)$ is a skew–symmetric matrix which satisfies $\frac{\mathrm{d}z}{\mathrm{d}t} = S(z)\nabla I(z)$. This scheme preserves I. The existence of $S(z)$ is mathematically proved [6]. While the scheme (9) can preserve only one invariant I, it can be further extended to preserve several invariants [7]:

$$\begin{aligned} &\frac{z_{n+1} - z_n}{\Delta t} \\ &\quad = S(z_n) \nabla_{\mathrm{d}} I_1(z_n, z_{n+1}) \cdots \nabla_{\mathrm{d}} I_k(z_n, z_{n+1}) \end{aligned} \tag{10}$$

where $S(z)$ is a completely anti-symmetric tensor satisfying $\frac{\mathrm{d}z}{\mathrm{d}t} = S(z)\nabla I_1(z)\nabla I_2(z) \cdots \nabla I_k(z)$. This scheme preserves $I_1, \cdots, I_k$ simultaneously. Below we employ this to preserve the $N-1$ local invariants.

3.3 Rejecting the local computational domain

The time-stepping above can be done only within the local computational domain, and if the solution approaches to one of the boundaries, we have to reject the domain and restart the whole procedure. This can be simply detected by the rule that when the value of the local invariants out of the computational domain becomes necessary, that is the timing of the rejection.

4. Numerical Examples

In this section we present some numerical examples that illustrate that the proposed method can in fact work. We used the upwind scheme for solving (5), and the 2nd order discrete gradient scheme (10) with Gonzalez's discrete gradient [1] for solving (1).

4.1 Damped oscillation

Fig. 7 shows the numerical solution of ODE (6) by the proposed method. Initial point is $(1.0, 0.0)$ and parameters are set to $\Delta t = 0.05, l_1 = 0.1, l_2 = 0.3, d_1 = 10^{-2}, d_2 = 10^{-4}$. The error of the numerical solution at $t = 50$ was 0.0023 in 2-norm, which is fair for the discretization widths. Note that this example clearly shows our point that even non-conservative systems can be solved by *conservative* integrators.

4.2 Kepler problem

Next let us consider the two-dimensional Kepler problem:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{pmatrix} = \begin{pmatrix} z_3 \\ z_4 \\ -\frac{z_1}{\left(z_1^2+z_2^2\right)^{\frac{3}{2}}} \\ -\frac{z_2}{\left(z_1^2+z_2^2\right)^{\frac{3}{2}}} \end{pmatrix} \tag{11}$$

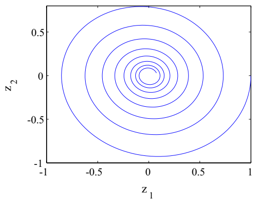

Fig. 6. Exact solution.

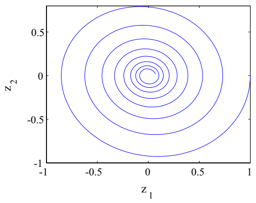

Fig. 7. Proposed method.

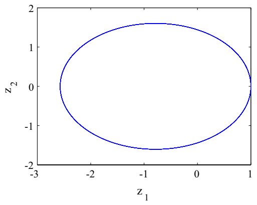

Fig. 8. Exact solution.

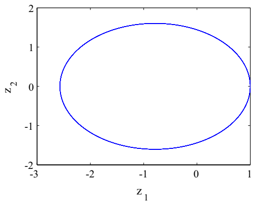

Fig. 9. Proposed method.

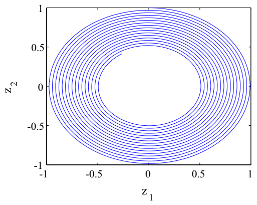

Fig. 10. $d_2 = 10^{-2}$.

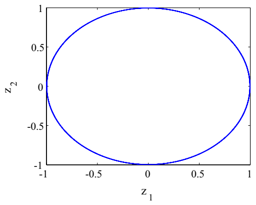

Fig. 11. $d_2 = 10^{-4}$.

Fig. 9 shows the numerical solution (z_1, z_2) by the proposed method. Initial point is $(1.0, 0.0, 0.0, 1.2)$ and parameters are set to $\Delta t = 0.005, l_1 = l_2 = l_3 = 0.01, l_4 = 0.03, d_1 = d_2 = d_3 = 0.002, d_4 = 0.0002$. The problem was solved for $0 \leq t \leq 150$ (the solution goes around approximately 10 times). We observed that the energy, angular momentum, and Runge–Lenz vector, which are the global invariants of the Kepler problem, were in fact well preserved up to the scheme accuracy, which describes the beautiful elliptic trajectory of the numerical solution.

4.3 Harmonic oscillation: parameter tuning

Here we examine the effect of the parameter setting to the numerical result, taking the harmonic oscillator:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}\begin{pmatrix} z_1 \\ z_2 \end{pmatrix}. \tag{12}$$

Figs. 10 and 11 show the numerical solutions by the proposed method when parameters are set to $\Delta t = 0.05, l_1 = 0.1, l_2 = 0.3, d_1 = 10^{-2}, d_2 = 10^{-2}$ and $\Delta t = 0.05, l_1 = 0.1, l_2 = 0.3, d_1 = 10^{-2}, d_2 = 10^{-4}$, respectively. The exact solution is the unit circle, so Fig. 11 is qualitatively better. We compared the numerical solution with various parameter settings, and found that the mesh–size d_2 in the direction of vector field is critical for the final accuracy.

5. Concluding Remarks

In this paper, we proposed a new simple geometric integration approach based on local invariants, and showed it actually works by several numerical examples. The new approach automatically detects enough number of invariants, and it can work also for non-conservative systems.

It should be noted, however, that the aim of this report is just to point out the principle itself, and the practical efficiency of the approach is left aside here. More careful consideration is required to discuss the important point, in connection with, for example, the choice of the discrete gradients, how to efficiently and precisely compute the local invariants, the tuning of the parameters, and so on. It should be also noted that in this approach the computational time obviously scales exponentially as the dimension N increases, and some cost saving technique should be incorporated.

References

[1] O. Gonzalez, Time integration and discrete Hamiltonian systems, J. Nonlinear Sci., **6** (1996), 449–467.

[2] G. R. W. Quispel and G. S. Turner, Discrete gradient methods for solving ODE's numerically while preserving a first integral, J. Phys. A, **29** (1996), 341–349.

[3] P. J. Olver, Applications of Lie groups to Differential Equations, Springer-Verlag, New York, 2000.

[4] F. John, Partial Differential Equations, Springer-Verlag, New York, 1981.

[5] T. Itoh and K. Abe, Hamiltonain-conserving discrete canonical equations based on variational differential quotients, J. Comput. Phys., **77** (1988), 85–102.

[6] G. R. W. Quispel and H. W. Capel, Solving ODE's numerically while preserving a first integral, Phys. Lett. A, **218** (1996), 223–228.

[7] G. R. W. Quispel and H. W. Capel, Solving ODE's numerically while preserving all first integrals, preprint.