LIE–POISSON HAMILTON–JACOBI THEORY AND LIE–POISSON INTEGRATORS

Ge ZHONG [1]
Computing Center, Academica Sinica, Beijing, PR China

and

Jerrold E. MARSDEN [2]
Department of Mathematics, Cornell University, Ithaca, NY 14853-7901, USA
and University of California, Berkeley, CA 94720, USA

Received 5 May 1988; accepted for publication 27 June 1988
Communicated by D.D. Holm

We present results on numerical integrators that *exactly preserve* momentum maps and Poisson brackets, thereby inducing integrators that preserve the natural Lie–Poisson structure on the duals of Lie algebras. The techniques are based on time-stepping with the generating function obtained as an approximate solution to the Hamilton–Jacobi equation, following ideas of deVogelaére, Channell, and Feng. To accomplish this, the Hamilton–Jacobi theory is reduced from T^*G to $\mathfrak{g}^*$, where $\mathfrak{g}$ is the Lie algebra of a Lie group G. The algorithms exactly preserve any additional conserved quantities in the problem. An explicit algorithm is given for any semi-simple group and in particular for the Euler equation of rigid body dynamics.

1. Introduction

This note is motivated by symplectic integrators as developed in refs. [1–4] and other references therein. This algorithm is based on the use of generating functions together with Hamilton–Jacobi theory for canonical hamiltonian systems. One version of this algorithm proceeds as follows: Start with a given configuration manifold Q and construct the corresponding phase space T*Q with canonical coordinates (q^i, p_i). (We use finite dimensional coordinates; however, the constructions are also valid in the infinite dimensional case.) Next, find an approximate solution $S(q^i, q_0^i, t)$ of the time dependent Hamilton–Jacobi equation

$$\frac{\partial S}{\partial t} + H\left(q^i, \frac{\partial S}{\partial q^i}\right) = 0 . \quad (1)$$

[1] Current address: MSRI, 1000 Centennial Drive, University of California, Berkeley, CA 94720, USA.

[2] Research partially supported by DOE contract DE AT03-85ER 12097 at UC Berkeley and MSI at Cornell University.

Now generate a canonical transformation φ_S: $(q_0, p_0) \mapsto (q, p)$ for a time step, which we shall denote Δt, using the standard formulae

$$p_{0i} = -\frac{\partial S}{\partial q_0^i}, \quad p_i = \frac{\partial S}{\partial q^i} . \quad (2)$$

Choose an initial condition so that S generates the identity transformation at $t=0$, such as

$$S = \frac{1}{2t}(q^i - q_0^i)^2 .$$

The resulting time dependent map is iterated N times to get the approximate dynamical solution for longer times $t = N\,\Delta t$. Here it is assumed that the first equation in (2) can indeed be solved for q^i in terms of q_0^i and p_{0i} which is then substituted into the second equation of (2). This procedure has several interesting features. The first, as explained in the papers of Channell and Scovel, and Feng, is its apparently better accuracy in representing the hamiltonian dynamics, especially the long term dynamics. Not only is the transformation generated this way exactly symplectic, but it appears to typically conserve energy more accurately than other methods. (We thank Swan Kim for demonstrating properties of this algorithm for use at Cornell.)

There are several other versions of the algorithm that one can also treat. For example, if specific coordinates are chosen on the phase space, one can use a generating function of the form $S(q', p_0, t)$. In this case one can get a simple formula for a first order algorithm simply by using $S = p_0 q' - \Delta t\, H(q', p_0)$, which is easy to implement, and for hamiltonians of the form kinetic plus potential, leads to an *explicit* symplectic algorithm.

In this paper we are interested in algorithms of this sort for systems with symmetry and in reducing them. For this purpose, it seems that only generating functions of the form $S(q', q'_0, t)$ are capable of a global treatment, which is important for systems with symmetry and their reductions.

2. Conservation laws

The first thing we point out is that if S is invariant under a group action, then the map φ_S defined by eq. (2) exactly preserves the corresponding conserved quantity. To precisely state this, we recall a bit of terminology. Suppose G is a Lie group that acts on Q and hence on T*Q by point transformations (cotangent lifts) with the corresponding equivariant momentum mapping $\boldsymbol{J}: T^*Q \to \mathfrak{g}^*$ with the associated map $J: \mathfrak{g} \to \mathcal{F}(T^*Q)$ defined by

$$J(\xi)(\alpha_q) = \langle \xi_Q, \alpha_q \rangle \,. \tag{3}$$

where $\mathfrak{g}$ is the Lie algebra of G, $\mathcal{F}(T^*Q)$ is the space of smooth functions on T*Q, $\alpha_q \in T^*_q Q$ is a covector at $q \in Q$, and ξ_Q is the infinitesimal generator of the action of G on Q. (See ref. [5] for further explanation of the notation.) We give the result for generating functions of the form $S(q', q'_0, t)$, but of course, there are analogous results for the other forms, and these can be easily checked numerically (for example, the choice $S = p_0 q' - \Delta t\, H(q', p_0)$ satisfies the requisite invariance properties).

Proposition 1. Suppose that $S: Q \times Q \to \mathbb{R}$ is invariant under the diagonal action of G, i.e., $S(gq, gq_0) = S(q, q_0)$. Then the momentum map $\boldsymbol{J}$ is invariant under the canonical transformation φ_S generated by S, i.e., $\boldsymbol{J} \circ \varphi_S = \boldsymbol{J}$.

This follows by differentiating the invariance condition assumed on S with respect to $g \in G$ in the direction of $\xi \in \mathfrak{g}$ and utilizing the definitions of φ_S, $\boldsymbol{J}$ and ξ_Q. The following is also true: If G acts on Q freely, and a given canonical transformation φ conserves J, then its generating function S can be defined on an open set of Q×Q which is invariant under the action of G, and S is invariant under the action of G. This is proved in ref. [6].

Note that if H is invariant under the action of G, then the corresponding solution of the Hamilton–Jacobi equation is G invariant as well. This follows from the short time uniqueness of the generating function of the type assumed for the flow of the hamiltonian vector field X_H determined by H. It also follows from proposition 1 that if the approximate solution of the Hamilton–Jacobi equation is chosen to be G invariant, then the corresponding algorithm will exactly conserve the momentum map.

3. Conserving energy

We recall that there are other algorithms which exactly preserve energy, some of which also preserve other conserved quantities; see refs. [7–10] and further references therein. However, *these algorithms cannot be symplectic*, according to the following result of Ge [11]:

Let H be a hamiltonian which has no other conserved quantities (in a given class $\mathcal{K}$, for example analytic functions) other than functions of H. That is, if $\{F, H\} = 0$, then $F(z) = F_0(H(z))$ for a function F_0. Let $\Phi_{\Delta t}$ be an algorithm which is defined for small Δt and is smooth. *If this algorithm is symplectic, and conserved H exactly, then it is the time advance map for the exact hamiltonian system up to a reparametrization of time.* In other words, approximate symplectic algorithms cannot preserve energy for nonintegrable systems.

This result is in fact easy to prove. The algorithm, being symplectic, is generated by a time dependent function $F(z, t)$, which we assume belongs to $\mathcal{K}$.

Since $\Phi_{\Delta t}$ preserves H, and H is assumed to be time independent, F commutes with H, and so $F(z)=F_0(H(z))$. It follows that the hamiltonian vector fields of F and H are parallel, so their integral curves are related by a time reparametrization.

For systems that have integrals, the above theorem can be applied to the induced algorithm on the symplectic or Poisson reduced spaces, as described below. On these reduced spaces, the assumption of nonintegrability is reasonable.

The above result suggests that conserving energy is a good criterion for a test of the accuracy of the algorithm. Numerical evidence (see, for example, ref. [4]) suggests that the constraint of conserving the symplectic structure is sufficiently strong that the method has good energy behavior, despite the fact that it cannot conserve energy exactly. We have seen in proposition 1, however, that algorithms can easily be symplectic and also preserve other conserved quantities, such as angular momentum. The reason for the good long time energy behavior of symplectic schemes in some cases is not well understood at the present time.

4. The Lie–Poisson Hamilton–Jacobi equation

Now assume we are dealing with the above situation and produce a G invariant generating function S. Since it is G invariant, it can be reduced, either by symplectic or Poisson reduction to produce an algorithm on the reduced space. It also gives rise to a reduced version of Hamilton–Jacobi theory. This can be applied to, for example, the rigid body in body representation or, in principle, to fluids and plasmas in the spatial representation. (See ref. [12] for an account of this theory.) Instead of giving the generalities of the theory, we shall illustrate it in an important case, namely, with the case of Lie–Poisson reduction, whereby we take Q=G, so the reduced space T*Q/G is isomorphic with the dual space $\mathfrak{g}^*$ with the Lie–Poisson bracket (with a plus sign for right reduction and a minus sign for left reduction). We shall give the special case of the rigid body for illustration, taking G=SO(3). Since the momentum map is preserved, one also gets an induced algorithm on the coadjoint orbits, or in the more general cases, on the symplectic reduced spaces. The proofs are routinely provided by tracing through the definitions, so we will just state the results. We begin with the reduced Hamilton–Jacobi equation itself. Thus, let H be a G invariant function on T*G and let H_L be the corresponding left reduced hamiltonian on $\mathfrak{g}^*$. (To be specific, we deal with left actions – of course there are similar statements for right reduced hamiltonians.) If S is invariant, there is a unique function S_L such that $S(g,g_0)=S_L(g^{-1}g_0)$. (One gets a slightly different representation for S by writing $g_0^{-1}g$ in place of $g^{-1}g_0$.)

Proposition 2. The *left* reduced Hamilton–Jacobi equation is the following equation for a function $S_L: G\to\mathbb{R}$:

$$\frac{\partial S_L}{\partial t}+H_L(-\mathrm{TR}_g^*\cdot \mathrm{d}S_L(g))=0\,, \tag{4}$$

which we call the *Lie–Poisson Hamilton–Jacobi equation*. The Lie–Poisson flow of the hamiltonian H_L is generated by the solution S_L of (4) in the sense that the flow is given by the Poisson transformation of $\mathfrak{g}^*: \Pi_0\mapsto\Pi$ defined as follows: Define $g\in G$ by solving the equation

$$\Pi_0=-\mathrm{TL}_g^*\cdot \mathrm{d}_g S_L \tag{5}$$

for $g\in G$ and then setting

$$\Pi=\mathrm{Ad}_{g^{-1}}^*\,\Pi_0\,. \tag{6}$$

Here Ad denotes the adjoint action and so the action in (6) is the coadjont action. Note that (6) and (5) give $\Pi=-\mathrm{TR}_g^*\cdot \mathrm{d}S_L(g)$. Note also that (5) and (6) are the analogues of eq. (2) and that (4) is the analogue of (1). Thus, one can obtain a Lie–Poisson integrator by approximately solving (4) and then using (5) and (6) to generate the algorithm. This algorithm (6) manifestly preserves the coadjoint orbits (the symplectic leaves in this case). As in the canonical case, one can generate algorithms of arbitrary accuracy this way. See refs. [6,11] for additional related results.

There may be conditions necessary on Π_0 for the solvability of eq. (5). This is noted in the example of the rigid body below.

For the case of the rigid body, these equations read as follows. First, eq. (4) reads

$$\frac{\partial S_L}{\partial t}+H_L(-\nabla S_L(A)\cdot A^T)=0\,, \tag{7}$$

i.e.

$$\frac{\partial S_L}{\partial t}+H_L\left(-\frac{\partial S_L}{\partial A^j_i}A^k_j\right) \tag{8}$$

(sum over j) where the action function S_L is a function of an orthogonal matrix A and where we have identified tangent and cotangent spaces using the bi-invariant metric on the rotation group. This metric corresponds to the standard euclidean metric on the Lie algebra, when identified with euclidean three-space. This identification maps the Lie algebra bracket to the cross product. The expression $\nabla S_L(A)\cdot A^T$ is a skew symmetric matrix, i.e., it lies in the Lie algebra so(3), so it makes sense for H_L to be evaluated on it. As usual, one has to be careful how the gradient (derivative) ∇S_L is computed, since there is a constraint $AA^T=I$ involved. If it is computed naively in the coordinates of the ambient space of 3×3 matrices, then one interprets the expression $\nabla S_L(A)\cdot A^T$ using naive partial derivatives and skew symmetrizing the result; this projects the gradient to the constraint space, so produces the gradient of the constrained function.

Eq. (7) thus is the Hamilton–Jacobi equation for the dynamics of a rigid body written directly in body representation. The flow of the hamiltonian is generated by S_L in the following way: it is the transformation of initial conditions at time $t=0$ to a general t determined by first solving the equation

$$\hat{\Pi}_0=-A^T\cdot\nabla S_L(A) \tag{9}$$

for the matrix A and then setting $\Pi=A\Pi_0$, where $\hat{\Pi}=[\Pi^i_j]$ is the skew *matrix* associated to the *vector* Π in the usual way: $\hat{\Pi}\cdot v=\Pi\times v$. (Again, the right hand side of (9) is to be skew symmetrized if the derivative was taken in the naive way with the constraint ignored.) We have written the results in terms of the body angular momentum vector Π; one can rewrite it in terms of the body angular velocity vector by using the relation $\Pi=I\omega$, where I is the moment of inertia tensor. In coordinates, eq. (9) reads as follows:

$$(\Pi_0)^k_i=-A^j_i\frac{\partial S_L}{\partial A^j_k}\,. \tag{10}$$

Finally, we note that similar equations also apply for fluids and plasmas, since they are also Lie–Poisson systems (but with right reduction). Also, the methods here clearly will generalize to the situation for reduction of any cotangent bundle; this generality is needed for example, for the case of free boundary fluids, ref. [13].

5. First order Lie–Poisson algorithms

A general way to construct first order algorithms valid in the Lie–Poisson setting (as well as its analogues in the symplectic and Poisson context) is as follows. Let $H:\mathfrak{g}^*\to\mathbb{R}$ be a given hamiltonian function and let S_0 be a function that generates a Poisson transformation $\varphi_0:\mathfrak{g}^*\to\mathfrak{g}^*$ and let

$$S_{\Delta t}=S_0+\Delta t\,H(L_g^*\,\mathrm{d}S_0)\,. \tag{11}$$

For small Δt, (11) generates a Poisson transformation, say $\varphi_{\Delta t}:\mathfrak{g}^*\to\mathfrak{g}^*$. Then we have:

Proposition 3. With the assumptions above, the algorithm

$$\Pi^k\mapsto\Pi^{k+1}=\varphi_0^{-1}\circ\varphi_{\Delta t}(\Pi^k) \tag{12}$$

is a Poisson difference scheme that is a first order difference scheme for the hamiltonian system with hamiltonian H.

In particular, if one can generate the identity transformation with a function S_0, then one can get a specific first order scheme. On G, one can introduce singularities in the time variable to do this, as we have already remarked. Interestingly, for $\mathfrak{g}$ semi-simple, one can do this in a *non-singular* way on $\mathfrak{g}^*$. In fact, in ref. [6] it is shown that in this case, the function

$$S_0(g)=\mathrm{trace}(\mathrm{Ad}_g^*) \tag{13}$$

generates the identity in a G-invariant neighborhood of the zero of $\mathfrak{g}^*$. One can also check this with a direct calculation using (5) and (6). The neighborhood condition is necessary since there may be some conditions on Π_0 required for the solvability of (5). For example, for the rigid body the condition is checked to be $\|\Pi_0\|<1$. This condition can be dealt with using a scaling argument. We note that when one solves (5) for g, it need not be the identity, and consistent with (6) we observe that g lies in the coadjoint isotropy of the element Π_0.

Formula (13) can be generalized to regular quadratic Lie algebras; i.e. Lie algebras with a nondegenerate, symmetric Ad-invariant bilinear form, say B. Indeed, a function u in a neighborhood of $e \in G$ exists which generates the identity in $\mathfrak{g}^*$ iff $\mathfrak{g}$ is regular quadratic. In fact, one takes $u(g)=B(\ln g, \ln g)$ where ln is a local inverse for exp, and conversely

$$B(a,b)=\frac{d}{dt}\bigg|_{t=0}\frac{d}{ds}\bigg|_{s=0} u(\exp(sa)\exp(tb)) .$$

Combining (11) and (13) we get the following proposition.

Proposition 4. The generating function

$$S_{\Delta t}(g)=\operatorname{trace}(\mathrm{Ad}_g^*)+\Delta t\, H(\mathrm{L}_g^*\,\mathrm{d}\operatorname{trace}(\mathrm{Ad}_g^*)) \tag{14}$$

defines, via (5) and (6), a Poisson map which is a first order Poisson integrator for the hamiltonian H.

We remark that this scheme will automatically preserve additional conserved quantities on $\mathfrak{g}^*$ that, for example, arise from invariance of the hamiltonian under a subgroup of G acting on the *right*. This is the situation for a rigid body with symmetry and fluid flow in a symmetric container (with left and right swapped) for instance.

6. Example: the free rigid body

For the case of the free rigid body, we let so(3), the Lie algebra of SO(3), be the space of skew symmetric 3×3 matrices. An isomorphism between so(3) and $\mathbb{R}^3$ is given by mapping the skew vector v to the matrix $\hat{v}$ defined previously. Using the Killing form $\langle A, B\rangle=\frac{1}{2}\operatorname{trace}A^{\mathrm{T}}B$, which corresponds to the standard inner product on $\mathbb{R}^3$, i.e. $\langle\hat{v},\hat{w}\rangle=v\cdot w$, we identify so(3) with so(3)*. We write the hamiltonian H: so(3)→$\mathbb{R}$ as $H(\hat{v})=\frac{1}{2}v\cdot Iv$, where I is the moment of inertia tensor. Let $\hat{I}$: so(3)→so(3) be defined by $\hat{I}(\hat{v})=(Iv)\hat{}$. Thus, $H(\hat{v})=\frac{1}{4}\langle\hat{v},\hat{I}(\hat{v})\rangle$. Eq. (13) becomes $S_0(A)=\operatorname{trace}(A)$ and so $\mathrm{TL}_A^*\,\mathrm{d}S_0=\frac{1}{2}(A-A^{\mathrm{T}})$. Therefore, (14) becomes

$$S_{\Delta t}=\operatorname{trace}(A)+\Delta t\, H(\tfrac{1}{2}(A-A^{\mathrm{T}})) , \tag{15}$$

and so using (15) in eqs. (5) and (6) gives the following specific Lie–Poisson algorithm for rigid body dynamics: it is the scheme $\Pi^k\mapsto\Pi^{k+1}$ defined by

$$\hat{\Pi}^k=\tfrac{1}{2}\{\tfrac{1}{4}[A\hat{I}(A-A^{\mathrm{T}})+\hat{I}(A-A^{\mathrm{T}})A^{\mathrm{T}}]\,\Delta t+(A-A^{\mathrm{T}})\} , \tag{16a}$$

$$\hat{\Pi}^{k+1}=\tfrac{1}{2}\{\tfrac{1}{4}[\hat{I}(A-A^{\mathrm{T}})A+A^{\mathrm{T}}\hat{I}(A-A^{\mathrm{T}})]\,\Delta t+(A-A^{\mathrm{T}})\} , \tag{16b}$$

where, as before, eq. (15a) is to be solved for the rotation matrix A and the result substituted into (15b). Letting $A^{\mathrm{S}}=\frac{1}{2}[A-A^{\mathrm{T}}]$ denote the skew part of the matrix A, we can rewrite (16) as

$$\hat{\Pi}^k=A^{\mathrm{S}}+(A\hat{I}A^{\mathrm{S}})^{\mathrm{S}}\Delta t , \tag{17a}$$

$$\hat{\Pi}^{k+1}=A^{\mathrm{S}}+(A^{\mathrm{T}}\hat{I}A^{\mathrm{S}})^{\mathrm{S}}\Delta t \tag{17b}$$

Of course, one can write $A=\exp(\xi)$ and solve (17a) for ξ and express the whole algorithm in terms of $\mathfrak{g}$ and $\mathfrak{g}^*$ alone, which may be important for computational purposes.

We known from the general theory that this scheme will automatically be Poisson and will, in particular, preserve the coadjoint orbits, i.e., the total angular momentum surfaces $\|\Pi\|^2=$constant. Of course, using (12) and other choices of S_0, it is possible to generate other algorithms for the rigid body, but the choice $S_0(A)=\operatorname{trace}(A)$ is particularly simple. For the regular quadratic case, the more general algorithm (12), with a particular choice of S_0 will still give a Lie–Poisson algorithm. We point out the interesting feature that the function (13) for the case of ideal Euler fluid flow is the function that assigns to a fluid placement field φ (an element of the diffeomorphism group of the containing region) the trace of the linear operator $\omega\mapsto\varphi^*\omega$, on vorticity fields ω, which measures the vortex distortion due to the nonrigidity of the flow. (See ref. [14] for further information.)

Acknowledgement

This research was partially supported by DOE contract DEAT03-85ER 12097 and the Mathematical Sciences Institute at Cornell. It is a pleasure to thank Rene deVogelaére, Bob Grossman, Darryl Holm, Feng Kang, Swan Kim, P.S. Krishnaprasad, Debbie Lewis, Clint Scovel, Juan Simo, and Alan Weinstein for useful discussions. We would especially like to thank Paul Channell for a careful reading of the manuscript and for informing us of the implementation of algorithms of this sort.

References

[1] R. DeVogelaére, Methods of integration which preserve the contact transformation property of the hamiltonian equations, Department of Mathematics, University of Notre Dame Report 4 (1956).

[2] P. Channell, Symplectic integration algorithms, Los Alamos National Laboratory Report AT-6: ATN-83-9 (1983).

[3] Feng Kang, J. Comput. Math. 4 (1986) 279.

[4] P. Channell and C. Scovel, Symplectic integration of hamiltonian systems, preprint (1988).

[5] R. Abraham and J.E. Marsden, Foundations of mechanics (Addison-Wesley, Reading, 1978).

[6] Ge Zhong, The generating function for the Poisson map, preprint (1986).

[7] A. Chorin, T.J.R. Hughes, J.E. Marsden and M. McCracken, Commun. Pure Appl. Math. 31 (1978) 205.

[8] D.M. Stofer, Some geometric and numerical methods for perturbed integrable systems, Thesis, Zurich (1987).

[9] D. Greenspan, Discrete numerical methods in physics and engineering (Academic Press, New York, 1974); J. Comput. Phys. 56 (1984) 21.

[10] Xie Zhi-Yun, Conservative numerical schemes for hamiltonian systems, J. Comput. Phys., to be published.

[11] Ge Zhong, Geometry in symplectic difference schemes and generating functions, preprint (1988).

[12] J.E. Marsden, A. Weinstein, T. Ratiu, R. Schmid and R.G. Spencer, Atti Accad. Sci. Torino 117 (1983) 289.

[13] D. Lewis, J. Marsden, R. Montgomery and T. Ratiu, Physica D 18 (1986) 391.

[14] J.E. Marsden and A. Weinstein, Physica D 7 (1983) 305.