Advances in Applied Mathematics and Mechanics
Adv. Appl. Math. Mech., Vol. **6**, No. 1, pp. 87-106

DOI: 10.4208/aamm.12-m12112
February 2014

Generating Function Methods for Coefficient-Varying Generalized Hamiltonian Systems

Xueyang Li*, Aiguo Xiao and Dongling Wang

Hunan Key Laboratory for Computation and Simulation in Science and Engineering, Key Laboratory of Intelligent Computing and Information Processing of Ministry of Education, Xiangtan University, Xiangtan 411105, Hunan, China

Received 2 September 2012; Accepted (in revised version) 21 May 2013

Available online 13 December 2013

Abstract. The generating function methods have been applied successfully to generalized Hamiltonian systems with constant or invertible Poisson-structure matrices. In this paper, we extend these results and present the generating function methods preserving the Poisson structures for generalized Hamiltonian systems with general variable Poisson-structure matrices. In particular, some obtained Poisson schemes are applied efficiently to some dynamical systems which can be written into generalized Hamiltonian systems (such as generalized Lotka-Volterra systems, Robbins equations and so on).

AMS subject classifications: 65P10, 65L05, 37M15, 70H05

Key words: Generalized Hamiltonian systems, Poisson manifolds, generating functions, structure-preserving algorithms, generalized Lotka-Volterra systems.

1 Introduction

We consider the generalized Hamiltonian systems (cf. [3,5,9])

$$y'(t)=B(y)\nabla H(y), \qquad y=(y_1,y_2,\cdots,y_n)^T\in\mathcal{M}, \tag{1.1}$$

where $\mathcal{M}$ is a differential manifold in $\mathbb{R}^n$, ∇ is the gradient operator, $H\in\mathbf{C}^\infty(\mathcal{M})$ is a Hamiltonian function, $B(y)=(b_{ij}(y))_{i,j=1}^n$ is a skew-symmetric Poisson-structure matrix and satisfies the Jacobi identity

$$b_{ij}(y)\frac{\partial b_{lk}(y)}{\partial y_i}+b_{ik}\frac{\partial b_{jl}(y)}{\partial y_i}+b_{il}(y)\frac{\partial b_{kj}(y)}{\partial y_i}=0, \qquad i,j,k,l=1,2,\cdots,n.$$

*Corresponding author.
Email: lixy1217@163.com (X. Y. Li)

The corresponding Poisson bracket (cf. [16,17]) is defined as

$$\{F,H\}(y)=(\nabla F(y))^T B(y)\nabla H(y), \quad \forall F,H\in \mathbf{C}^\infty(\mathcal{M}).$$

Definition 1.1 (cf. [5], Chapter 12)**.** A map $y\rightarrow\hat{y}=g(y)$: $\mathcal{M}\rightarrow\mathcal{M}$ is called a Poisson map, if it is a (local) diffeomorphism and preserves the Poisson bracket, i.e.,

$$\{F\circ g,H\circ g\}=\{F,H\}\circ g, \quad \forall F,H\in \mathbf{C}^\infty(\mathcal{M}).$$

An n-order square matrix $M(y,\hat{y})$ is called a Poisson matrix if

$$M(y,\hat{y})B(y)M(y,\hat{y})^T=B(\hat{y}).$$

A function $C(y)\in \mathbf{C}^\infty(\mathcal{M})$ is called a Casimir function if

$$\{C,F\}(y)=0, \quad \forall F\in \mathbf{C}^\infty(\mathcal{M}).$$

It is easy to prove that $g(y)$ is a Poisson map if and only if

$$g_y(y)B(y)(g_y(y))^T=B(\hat{y}).$$

For a numerical algorithm applied to the systems (1.1), we hope that it can preserve more structure characterizations of the original systems. If the discrete flow obtained by an algorithm for the systems (1.1) is a Poisson map, then we say this algorithm is a Poisson-structure-preserving algorithm, referred to a Poisson scheme. And the Poisson scheme is also an extension of the symplectic algorithm (cf. [2,5,14,18]).

Generating function methods (cf. [5,8,11,15]) are also important approaches to construct the symplectic scheme for canonical Hamiltonian systems and the Poisson schemes for generalized Hamiltonian systems. So far, some generating function methods preserving the Poisson structures for linear generalized Hamiltonian systems (i.e., Lie-Poisson systems) and the generalized Hamiltonian systems with constant or invertible Poisson-structure matrices have been presented respectively (cf. [6,7,19]). Moreover, in this paper, we extend these results and present the generating function methods for generalized Hamiltonian systems with general variable Poisson-structure matrices which can be singular. In particular, the obtained Poisson schemes are applied efficiently to some dynamical systems which can be written into the forms of generalized Hamiltonian systems (such as generalized Lotka-Volterra systems, Robbins equations and so on).

In Section 2, we extend the Hamilton-Jacobian theorem to the coefficient-varying generalized Hamiltonian systems (1.1), and get their generating functions. In Section 3, based on the obtained results, we construct some Poisson schemes for the systems (1.1). In Section 4, we use these obtained schemes to solve several specific systems and give the corresponding numerical results.

2 Generating functions of generalized Hamiltonian systems

We first make some similar preparations to those in [6,19]. For a definite Poisson manifold $\mathcal{M}$,

$$\alpha(y,\hat{y})=\begin{pmatrix} A_\alpha(y,\hat{y}) & B_\alpha(y,\hat{y}) \\ C_\alpha(y,\hat{y}) & D_\alpha(y,\hat{y}) \end{pmatrix}\in\mathbf{GL}(2n)$$

is a matrix function defined on $\mathcal{M}\times\mathcal{M}$, where $\mathbf{GL}(2n)$ denotes the group of all the $2n$-order reversible matrices, and there exists an n-order skew-symmetric constant matrix K such that

$$\alpha\begin{pmatrix} B(\hat{y}) & O \\ O & -B(y) \end{pmatrix}\alpha^T=\begin{pmatrix} O & K \\ K & O \end{pmatrix}. \tag{2.1}$$

Expanding (2.1) leads to

$$\begin{aligned} &A_\alpha B(\hat{y})A_\alpha^T-B_\alpha B(y)B_\alpha^T=O, && A_\alpha B(\hat{y})C_\alpha^T-B_\alpha B(y)D_\alpha^T=K,\\ &C_\alpha B(\hat{y})A_\alpha^T-D_\alpha B(y)B_\alpha^T=K, && C_\alpha B(\hat{y})C_\alpha^T-D_\alpha B(y)D_\alpha^T=O. \end{aligned}$$

The inverse of $\alpha(y,\hat{y})$ is denoted by

$$\alpha^{-1}(y,\hat{y})=\begin{pmatrix} A^\alpha(y,\hat{y}) & B^\alpha(y,\hat{y}) \\ C^\alpha(y,\hat{y}) & D^\alpha(y,\hat{y}) \end{pmatrix}.$$

For a given $\alpha\in\mathbf{GL}(2n)$, we can define the linear fractional transformation σ_α

$$N(y,\hat{y}):=\sigma_\alpha(M(y,\hat{y}))=(A_\alpha M+B_\alpha)(C_\alpha M+D_\alpha)^{-1} \tag{2.2}$$

under the transversal condition

$$|C_\alpha M+D_\alpha|\neq 0. \tag{2.3}$$

Its inverse transformation $\sigma_\alpha^{-1}=\sigma_{\alpha^{-1}}$ is

$$M(y,\hat{y})=\sigma_{\alpha^{-1}}(N(y,\hat{y}))=(A^\alpha N+B^\alpha)(C^\alpha N+D^\alpha)^{-1}, \tag{2.4}$$

where

$$|C^\alpha N+D^\alpha|\neq 0. \tag{2.5}$$

Under the direct proof, we can easily obtain the lemma as follows.

Lemma 2.1. *Suppose that $\alpha\in\mathbf{GL}(2n)$ satisfies* (2.1). *Then $M(y,\hat{y})$ is a Poisson matrix satisfying* (2.2) *if and only if $N(y,\hat{y})=\sigma_\alpha(M)$ satisfies* (2.5) *and $NK\in\mathbf{Sm}(n)$, where $\mathbf{Sm}(n)$ denotes the group of n-order symmetrical matrices.*

Theorem 2.1. *Suppose that $y\to\hat{y}=g(y)$ is a Poisson map with Jacobian matrix $M(y)=g_y(y)$, and $\hat{w}$,w are differentiable maps defined on $\mathcal{M}\times\mathcal{M}$ satisfying*

$$\begin{aligned}&\hat{w}=\hat{w}(y,\hat{y}),\quad w=w(y,\hat{y}),\\&\frac{\partial\hat{w}}{\partial\hat{y}}=A_\alpha,\quad \frac{\partial\hat{w}}{\partial y}=B_\alpha,\quad \frac{\partial w}{\partial\hat{y}}=C_\alpha,\quad \frac{\partial w}{\partial y}=D_\alpha,\end{aligned}$$

where α and $M(y)$ satisfy (2.1) *and* (2.2)*. Then there exists a map $w\to\hat{w}=f(w)$ such that its Jacobian matrix $N(w)=f_w(w)=\sigma_\alpha(M(y))$ satisfies*

$$N(w)K\in\mathbf{Sm}(n).$$

Proof. According to the implicit function theorem and $|C_\alpha M(y)+D_\alpha|\neq 0$, we know that $w=w(y,g(y))$ is invertible in a neighborhood of y. Thus,

$$\frac{\delta\hat{w}}{\delta w}=\frac{\delta\hat{w}}{\delta y}\Big(\frac{\delta w}{\delta y}\Big)^{-1}=\Big(\frac{\partial\hat{w}}{\partial\hat{y}}M(y)+\frac{\partial\hat{w}}{\partial y}\Big)\Big(\frac{\partial w}{\partial\hat{y}}M(y)+\frac{\partial w}{\partial y}\Big)^{-1}=\sigma_\alpha(M(y))=N(w),$$

and there exists a map f satisfying $\hat{w}=f(w)$.

Moreover, by Lemma 2.1, we have $f_w(w)K=N(w)K\in\mathbf{Sm}(n)$. $\square$

According to the Darboux Theorem (cf. [9]), there exists a diffeomorphic map h: $\mathcal{M}\to h(\mathcal{M})$ defined in a neighborhood of $y\in\mathcal{M}$ such that $h_y(y)B(y)h_y(y)^T$ is a constant matrix. Let $K=h_y(y)B(y)h_y(y)^T$. Then the generalized Hamiltonian systems (1.1) become

$$y'(t)=(h_y(y))^{-1}K\big(h_y(y)\big)^{-T}\nabla H(y). \tag{2.6}$$

Suppose that $y\to\hat{y}=g(y):\mathcal{M}\to\mathcal{M}$ is a Poisson map, and let

$$A_\alpha=h_{\hat{y}}(\hat{y}),\quad B_\alpha=-h_y(y),\quad C_\alpha=\frac{1}{2}h_{\hat{y}}(\hat{y}),\quad D_\alpha=\frac{1}{2}h_y(y). \tag{2.7}$$

Then α satisfies (2.1) obviously, and

$$\alpha^{-1}=\begin{pmatrix}A^\alpha & B^\alpha\\ C^\alpha & D^\alpha\end{pmatrix}=\begin{pmatrix}\frac{1}{2}(h_{\hat{y}}(\hat{y}))^{-1} & (h_{\hat{y}}(\hat{y}))^{-1}\\ -\frac{1}{2}(h_y(y))^{-1} & (h_y(y))^{-1}\end{pmatrix}. \tag{2.8}$$

Let

$$\hat{w}(y,\hat{y})=h(\hat{y})-h(y),\quad w(y,\hat{y})=\frac{1}{2}h(\hat{y})+\frac{1}{2}h(y). \tag{2.9}$$

Then its inverse mapping is

$$\hat{y}(w,\hat{w})=h^{-1}\Big(\frac{1}{2}\hat{w}+w\Big),\quad y(w,\hat{w})=h^{-1}\Big(\frac{1}{2}\hat{w}-w\Big).$$

Theorem 2.2. *Suppose that $\phi(w,\tau)$ is the solution of the partial differential equation*

$$\frac{\partial\phi(w,\tau)}{\partial\tau}=-H\Big(h^{-1}\Big(-\frac{1}{2}K\nabla\phi(w,\tau)+w\Big)\Big),\qquad \phi(w,0)=0, \tag{2.10}$$

$g^\tau y$ is the phase flow of the generalized Hamiltonian systems (2.6)*. Then, in a neighborhood of $y\in\mathcal{M}$ for $\tau>0$ small enough, $\hat{y}=g^\tau y$ if and only if*

$$h(\hat{y})-h(y)=-K\nabla\phi\Big(\frac{1}{2}h(\hat{y})+\frac{1}{2}h(y),\tau\Big). \tag{2.11}$$

Proof. Since the analytical solution of the generalized Hamiltonian systems (2.6) is existent and unique, we just prove that the Eq. (2.11) can yield

$$\frac{d\hat{y}}{d\tau}=B(\hat{y})\nabla H(\hat{y}).$$

From (2.9) and (2.11), we have

$$\hat{w}=-K\nabla\phi(w,\tau),\qquad \hat{y}=-\frac{1}{2}K\nabla\phi(w,\tau)+w.$$

Differentiating the both sides of the Eq. (2.11) with respect to τ yields

$$(h_{\hat{y}}(\hat{y}))\frac{d\hat{y}}{d\tau}=-\frac{1}{2}K\phi_{ww}(w,\tau)h_{\hat{y}}(\hat{y})\frac{d\hat{y}}{d\tau}-K\frac{\partial}{\partial\tau}\frac{\partial}{\partial w}\phi(w,\tau) \tag{2.12}$$

and

$$\begin{aligned}\frac{\partial}{\partial\tau}\frac{\partial}{\partial w}\phi(w,\tau)&=-\Big((h_{\hat{y}}(\hat{y}))^{-1}\Big(-\frac{1}{2}K\phi_{ww}(w,\tau)+I\Big)\Big)^T\nabla H(\hat{y})\\&=-\Big(I+\frac{1}{2}\phi_{ww}(w,\tau)K\Big)(h_{\hat{y}}(\hat{y}))^{-T}\nabla H(\hat{y}).\end{aligned} \tag{2.13}$$

Moreover, we have

$$\begin{aligned}\Big(I+\frac{1}{2}K\phi_{ww}(w,\tau)\Big)h_{\hat{y}}(\hat{y})\frac{d\hat{y}}{d\tau}&=K\Big(I+\frac{1}{2}\phi_{ww}(w,\tau)K\Big)(h_{\hat{y}}(\hat{y}))^{-T}\nabla H(\hat{y})\\&=\Big(I+\frac{1}{2}K\phi_{ww}(w,\tau)\Big)K(h_{\hat{y}}(\hat{y}))^{-T}\nabla H(\hat{y}).\end{aligned}$$

Since $\phi_{ww}(w,0)=0$ and $I+K\phi_{ww}(w,\tau)/2$ is reversible when τ is small enough, we have

$$\frac{d\hat{y}}{d\tau}=(h_{\hat{y}}(\hat{y}))^{-1}K(h_{\hat{y}}(\hat{y}))^{-T}\nabla H(\hat{y}).$$

By $B(\hat{y})=(h_{\hat{y}}(\hat{y}))^{-1}K(h_{\hat{y}}(\hat{y}))^{-T}$, we have $d\hat{y}/d\tau=B(\hat{y})\nabla H(\hat{y})$. Therefore, the original proposition is established. $\square$

Suppose that $\hat{y}=g^{\tau}y$ is the phase flow of the generalized Hamiltonian systems (2.6), and $(\hat{w},w)$ is determined by (2.9). The scalar function $\phi(w,\tau)$ in Theorem 2.2 is called a generating function of the generalized Hamiltonian systems (2.6), and the Eq. (2.10) is called a Hamilton-Jacobian equation.

If $H(y)$ depends analytically on y, then for $\tau>0$ small enough, the generating function $\phi(w,\tau)$ can be expanded into a convergent power series

$$\phi(w,\tau)=\sum_{j=0}^{\infty}\phi^{(j)}(w)\tau^{j}.$$

We have

$$\nabla\phi(w,\tau)=\phi_w(w,\tau)=\sum_{j=0}^{\infty}\nabla\phi^{(j)}(w)\tau^{j},\quad \frac{\partial\phi}{\partial\tau}=\phi_\tau(w,\tau)=\sum_{j=0}^{\infty}(j+1)\tau^{j}\phi^{(j+1)}(w),$$

where $\phi^{(0)}(w)=\nabla\phi^{(0)}(w)=0$ as $\phi(w,0)=0$. From the above formulas and (2.10), we have

$$\begin{aligned}\frac{\partial\phi}{\partial\tau}&=-H\Big(h^{-1}\Big(-\frac{1}{2}K\nabla\phi(w,\tau)+w\Big)\Big)\\&=-\sum_{m=0}^{\infty}\frac{1}{m!}d_w^m(H\circ h^{-1})\cdot\Big(-\frac{1}{2}\sum_{j_1=1}^{\infty}K\nabla\phi^{(j_1)},\cdots,-\frac{1}{2}\sum_{j_m=1}^{\infty}K\nabla\phi^{(j_m)}\Big)\\&=-\sum_{m=0}^{\infty}\frac{1}{m!}\sum_{j=1}^{\infty}\tau^{j}\sum_{j_1+\cdots+j_m=j}d_w^m(H\circ h^{-1})\cdot\Big(-\frac{1}{2}K\nabla\phi^{(j_1)},\cdots,-\frac{1}{2}K\nabla\phi^{(j_m)}\Big),\end{aligned}$$

where the operator d_w^m is defined as

$$d_w^m f(w)\cdot(p^{(1)},\cdots,p^{(m)})=\sum_{j_1,\cdots,j_m=1}^{n}\frac{\partial^n f(w)}{\partial w_{j_1}\cdots\partial w_{j_n}}p_{j_1}^{(1)}\cdots p_{j_m}^{(m)},$$
$$\text{for }\ \forall f\in\mathbf{C}^{\infty}(\mathbb{R}^n),\quad p^{(1)},\cdots,p^{(m)}\in\mathbb{R}^n.$$

Thus, $\phi^{(j)}(w)$, $j=1,2,\cdots$, can be obtained by the recursive formula

$$\phi^{(1)}(w)=-H\circ h^{-1}(w), \tag{2.14a}$$

$$\begin{aligned}\phi^{(j+1)}(w)=&\frac{-1}{j+1}\sum_{m=1}^{j}\frac{1}{m!}\sum_{j_1+\cdots+j_m=j}d_w^m(H\circ h^{-1})(w)\\&\cdot\Big(-\frac{1}{2}K\nabla\phi^{(j_1)}(w),\cdots,-\frac{1}{2}K\nabla\phi^{(j_m)}(w)\Big).\end{aligned} \tag{2.14b}$$

3 Poisson schemes of coefficient-varying generalized Hamiltonian systems

In this section, we will construct the Poisson-Structure-Preserving schemes for solving the generalized Hamiltonian systems (2.6).

Theorem 3.1. *Suppose that $\phi(w,\tau)$ is the analytical solution of the Hamilton-Jacobian equation* (2.10)*,*

$$\psi^{(m)}(w,\tau)=\sum_{j=1}^{m}\phi^{(j)}(w)\tau^j, \qquad m=1,2,\cdots,$$

is the m-order approximation to $\phi(w,\tau)$ with $\tau>0$ small enough and $\phi^{(j)}(w)$ is determined by (2.14)*. Then the m-order Poisson scheme for the generalized Hamiltonian systems* (1.1) *is given by*

$$h(y^{(k+1)})-h(y^{(k)})=-K\sum_{j=1}^{m}\nabla\phi^{(j)}\Big(\frac{1}{2}h(y^{(k+1)})+\frac{1}{2}h(y^{(k)})\Big)\tau^j. \tag{3.1}$$

Proof. Consider the Jacobian matrix of the scheme (3.1) with respect to $y^{(k)}$. We have

$$h_y(y^{(k+1)})\frac{\partial y^{(k+1)}}{\partial y^{(k)}}-h_y(y^{(k)})=-K\sum_{j=1}^{m}\phi_{ww}^{(j)}(w)\tau^j\Big(\frac{1}{2}h_y(y^{(k+1)})\frac{\partial y^{(k+1)}}{\partial y^{(k)}}+\frac{1}{2}h_y(y^{(k)})\Big),$$

where $w=h(y^{(k+1)})/2+h(y^{(k)})/2$. Let

$$y=y^{(k)}, \quad \hat{y}=y^{(k+1)}, \quad M=\frac{\partial y^{(k+1)}}{\partial y^{(k)}}, \quad N=-K\sum_{j=1}^{m}\phi_{ww}^{(j)}(w)\tau^j, \tag{3.2}$$

where α and α^{-1} are defined by (2.7) and (2.8), and we obtain

$$A_\alpha M+B_\alpha=N(C_\alpha M+D_\alpha).$$

It follows from the equality $\psi^{(m)}(w,0)=0$ and Lemma 2.1 that the matrix

$$C^\alpha N+D^\alpha=\frac{1}{2}(h_y(y))^{-1}K\psi_{ww}^{(m)}(w,\tau)+(h_y(y))^{-1}$$

is reversible for $\tau>0$ small enough, $NK=K\psi_{ww}^{(m)}(w,\tau)K$ is a real symmetric matrix, α satisfies obviously (2.1), M is a Poisson matrix, and $C_\alpha M+D_\alpha$ is reversible. Thus, (3.1) is a Poisson scheme. Moreover, we have

$$\begin{aligned}&h(g^\tau y)-h(y)+K\sum_{j=1}^{m}\nabla\phi^{(j)}\Big(\frac{1}{2}h(g^\tau y)+\frac{1}{2}h(y)\Big)\tau^j\\ =&-K\sum_{j=m+1}^{\infty}\nabla\phi^{(j)}\Big(\frac{1}{2}h(g^\tau y)+\frac{1}{2}h(y)\Big)\tau^j\\ =&\mathcal{O}(\tau^{m+1}).\end{aligned}$$

Therefore, (3.1) is of order m. $\square$

Based on Theorem 3.1 and (2.14), we construct the second-order and fourth-order Poisson schemes for the systems (2.6). In fact,

$$\begin{aligned}
&\phi^{(1)}(w)=-H\circ h^{-1}(w),\\
&\phi^{(2)}(w)=\phi^{(4)}(w)=0,\\
&\phi^{(3)}(w)=\frac{1}{24}\Big((\nabla H\circ h^{-1})^T K d^2(H\circ h^{-1})K\nabla H\circ h^{-1}\Big)(w),
\end{aligned}$$

where $d^2(H\circ h^{-1})$ is the Hessian matrix of $H\circ h^{-1}$. Therefore, we can get the second-order Poisson scheme

$$h(y^{(k+1)})=h(y^{(k)})+\tau K\nabla(H\circ h^{-1})\Big(\frac{1}{2}h(y^{(k+1)})+\frac{1}{2}h(y^{(k)})\Big), \tag{3.3}$$

and the fourth-order Poisson scheme

$$\begin{aligned}
h(y^{(k+1)})=&h(y^{(k)})+\tau K\nabla(H\circ h^{-1})\Big(\frac{1}{2}h(y^{(k+1)})+\frac{1}{2}h(y^{(k)})\Big)\\
&-\frac{\tau^3}{24}K\nabla\Big(\sum_{i,j=1}^{n}(Kd^2(H\circ h^{-1})K)_{ij}(H\circ h^{-1})_{z_i}(H\circ h^{-1})_{z_j}\Big)\Big(\frac{1}{2}h(y^{(k+1)})\\
&+\frac{1}{2}h(y^{(k)})\Big),
\end{aligned} \tag{3.4}$$

respectively.

The Poisson schemes obtained by generating function methods include n-dimensional nonlinear algebraic equations, and have less computation cost than implicit Runge-Kutta methods. These schemes are of even order since the terms with the even powers of τ in the expansion of $\phi(w,\tau)$ given in (2.10) don't appear.

The key of constructing Poisson schemes is to obtain the diffeomorphism h satisfying (2.6), and from the previous paper, we know that the differentiable functions is existed. However, we can not obtain the diffeomorphism directly for the general case. But for some special and useful generalized Hamiltonian systems, it is easy to find such differentiable functions. And that will be shown in the next section.

4 Numerical examples

In this section, we will use the Poisson schemes given in Section 3 to solve some special generalized Hamiltonian systems.

Example 4.1 (cf. [10])**.** A generalized Lotka-Volterra (GLV) system consists of ordinary differential equations which can be written as

$$y_i'(t)=y_i\Big(\lambda_i+\sum_{j=1}^{m}A_{ij}\prod_{k=1}^{n}y_k^{R_{jk}}\Big), \qquad i=1,2,\cdots,n, \tag{4.1}$$

where n and m are positive integers, $m\geq n$, A, R, λ are $n\times m$, $m\times n$, $n\times 1$ real matrices respectively.

It is easy to prove that for the GLV system (4.1), if there exist t_1 and y_i such that $y_i(t_1)=0$, then $y_i(t)\equiv 0$ and the system (3.2) degenerates to an $n-1$ dimensional system. Therefore, we may assume that if $y_i(t_0)\neq 0$ $(i=1,2,\cdots,n)$ in the initial time t_0, then we have $y_i(t)>0$ or $y_i(t)<0$ for the whole time.

If the coefficients A, R and λ of the GLV system (4.1) can be written as the format

$$\lambda=\overline{K}L, \qquad A=\overline{K}R^TD,$$

where $\overline{K}$ is an $n\times n$ antisymmetric real matrix, L is an n dimensional vector, D is an $m\times m$ diagonal matrix with full rank. Then the system (4.1) can be written as the form of generalized Hamiltonian systems (1.1), where

$$B(y)=Y\overline{K}Y, \qquad Y=\mathrm{diag}\{y_1,y_2,\cdots,y_n\}, \tag{4.2a}$$

$$H(y)=\sum_{j=1}^{m}D_{jj}\prod_{k=1}^{n}y_k^{R_{jk}}+\sum_{j=1}^{n}L_j\ln|y_j|, \qquad \nabla_iH=\frac{\partial H(y)}{\partial y_i}=\sum_{j=1}^{m}D_{jj}\prod_{k=1}^{n}y_k^{R_{jk}-\delta_{ik}}+L_i\frac{1}{y_i}, \tag{4.2b}$$

here "ln" denotes logarithmic function. So this system is also called a generalized Lotka-Volterra-Poisson system, referred to a GLVP system [20]. For the theory studies of GLVP systems, we can refer to [20] and other literature. Let

$$z_i=h_i(y)=\ln|y_i|, \qquad i=1,2,\cdots,n,$$

here $h_i(y)$ denotes the ith component of $h(y)$. Apparently we have $h(y)$ is invertible when $y_i\neq 0$, $i=1,2,\cdots,n$ and $B(y)=(h_y(y))^{-1}\overline{K}(h_y(y))^{-T}$. Then the system can be written as the form of (2.6), and $\overline{K}$ is equivalent to K in (2.6), so we use K instead of $\overline{K}$. Moreover, for α and α^{-1} defined in (2.7) and (2.8), we can obtain $(\hat{w},w)$ as follows:

$$\begin{aligned}
\hat{w}(y,\hat{y})&=\big(\ln|\hat{y}_1|,\cdots,\ln|\hat{y}_n|\big)^T-\big(\ln|y_1|,\cdots,\ln|y_n|\big)^T,\\
w(y,\hat{y})&=\frac{1}{2}\big(\ln|\hat{y}_1|,\cdots,\ln|\hat{y}_n|\big)^T+\frac{1}{2}\big(\ln|y_1|,\cdots,\ln|y_n|\big)^T,\\
\hat{y}(w,\hat{w})&=\Big(\exp\Big(\frac{1}{2}\hat{w}_1+w_1\Big),\cdots,\exp\Big(\frac{1}{2}\hat{w}_n+w_n\Big)\Big)^T,\\
y(w,\hat{w})&=\Big(\exp\Big(\frac{1}{2}\hat{w}_1-w_1\Big),\cdots,\exp\Big(\frac{1}{2}\hat{w}_n-w_n\Big)\Big)^T.
\end{aligned}$$

Thus

$$h^{-1}(w)=(\theta_1\exp(w_1),\cdots,\theta_n\exp(w_n))^T=(\theta_1\sqrt{y_1\hat{y}_1},\cdots,\theta_n\sqrt{y_n\hat{y}_n})^T, \tag{4.3}$$

where $\theta_i=\mathrm{sgn}(y_i^{(0)})$, $i=1,2,\cdots,n$, "sgn" denotes sign function. Inserting (4.3) into (3.3) and (3.4) yields the second-order and fourth-order Poisson schemes for solving the system

(4.1). The second-order scheme is

$$
\begin{aligned}
&\begin{pmatrix} \ln|y_1^{(k+1)}| \\ \vdots \\ \ln|y_n^{(k+1)}| \end{pmatrix} - \begin{pmatrix} \ln|y_1^{(k)}| \\ \vdots \\ \ln|y_n^{(k)}| \end{pmatrix} \\
=&\tau K \begin{pmatrix} \theta_1\sqrt{y_1^{(k+1)}y_1^{(k)}} & & \\ & \ddots & \\ & & \theta_n\sqrt{y_n^{(k+1)}y_n^{(k)}} \end{pmatrix} \\
&\cdot \nabla H\left(\theta_1\sqrt{y_1^{(k+1)}y_1^{(k)}},\cdots,\theta_n\sqrt{y_n^{(k+1)}y_n^{(k)}}\right), \quad k=0,1,2,\cdots.
\end{aligned}
\tag{4.4}
$$

Example 4.2 (cf. [9])**.** Consider the two-dimensional Lotka-Volterra system

$$y_1'(t)=y_1(y_2-1), \qquad y_2'(t)=y_2(2-y_1), \tag{4.5}$$

which is equivalent to the following form

$$y'(t)=YKY\nabla H,$$

where

$$y'(t)=\begin{pmatrix} y_1'(t) \\ y_2'(t) \end{pmatrix}, \qquad Y=\begin{pmatrix} y_1 & 0 \\ 0 & y_2 \end{pmatrix}, \qquad K=\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

$H=H(y)$ is a Hamiltonian function, and

$$H(y)=y_1-2\ln|y_1|+y_2-\ln|y_2|, \quad \nabla H=\begin{pmatrix} 1-\dfrac{2}{y_1} \\ 1-\dfrac{1}{y_2} \end{pmatrix}.$$

Using the second-order Poisson scheme (4.4), we obtain the corresponding algorithm

$$
\begin{aligned}
&\begin{pmatrix} \ln|y_1^{(k+1)}| \\ \ln|y_2^{(k+1)}| \end{pmatrix} - \begin{pmatrix} \ln|y_1^{(k)}| \\ \ln|y_2^{(k)}| \end{pmatrix} = \tau \begin{pmatrix} \theta_2\sqrt{y_2^{(k)}y_2^{(k+1)}}-1 \\ -\theta_1\sqrt{y_1^{(k)}y_1^{(k+1)}}+2 \end{pmatrix}, \\
&\theta_1=\mathrm{sgn}(y_1^{(0)}), \qquad \theta_2=\mathrm{sgn}(y_2^{(0)}).
\end{aligned}
$$

The numerical solution of the Eq. (4.5) is shown in the Fig. 1 with the time interval $[0,100]$, and the time stepsize $\tau=0.1$. Three curves from the outside to inside are the images of numerical solutions while the initial values are $(0.5,1)$, $(1,1)$ and $(1.5,1)$, respectively. Taking the initial value $(0.5,1)$, the errors in the Hamiltonian function $H(y)$

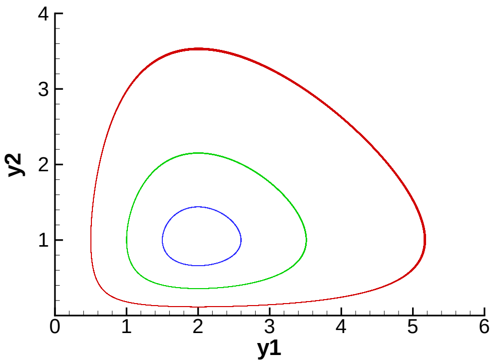

Figure 1: Numerical solution images of the system (4.5).

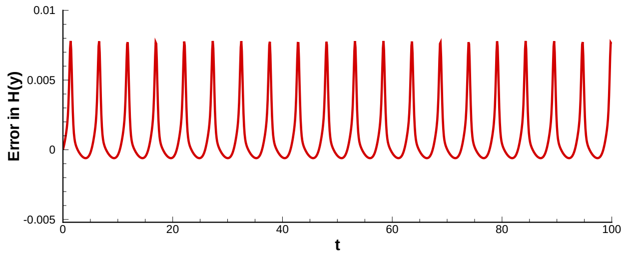

Figure 2: Errors in $H(y)$ from the generating function method with stepsize $\tau=0.1$.

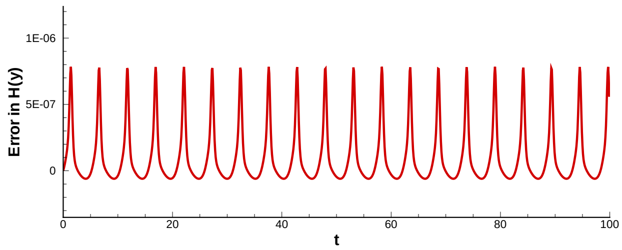

Figure 3: Errors in $H(y)$ from the generating function method with stepsize $\tau=0.001$.

from the second-order generating function method and the Euler midpoint rule are given in Figs. 2-4 and Figs. 5-6, respectively. In Fig. 2 and Fig. 5, we can see the Hamiltonian deviation from the two methods are the same order. But for the time interval $[0,5000]$ as Fig. 7, it is obvious that the error bounds of $H(y)$ from the Euler midpoint rule are increasing by the growth of time, but the error bounds from the generating function method are not apparently changed by the growth of time. Therefore, an important advantage of generating function method is to control the error bounds of the Hamiltonian function over the long time. Fig. 4 shows the error bounds of $H(y)$ from the generating function method with different time intervals and time stepsizes, and we can see the convergence of the numerical method.

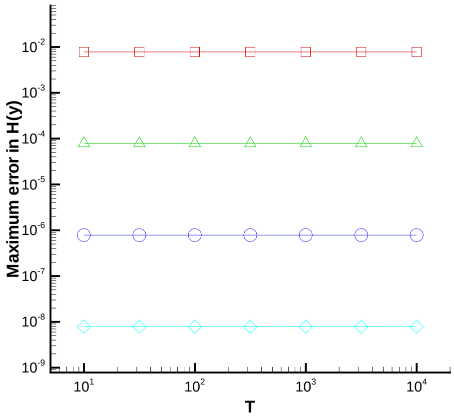

Figure 4: Maximum errors in $H(y)$ from the generating function method with different time intervals $[0,T]$ and stepsizes. $\square$: $\tau=0.1$, $\triangle$: $\tau=0.01$, $\bigcirc$: $\tau=0.001$, $\diamond$: $\tau=0.0001$.

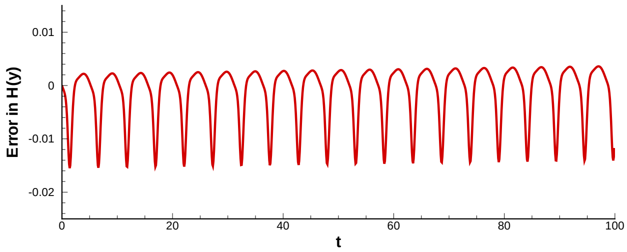

Figure 5: Errors in $H(y)$ from the Euler midpoint rule with stepsize $\tau=0.1$.

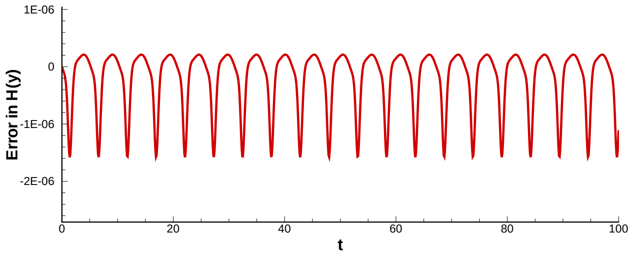

Figure 6: Errors in $H(y)$ from the Euler midpoint rule with stepsize $\tau=0.001$.

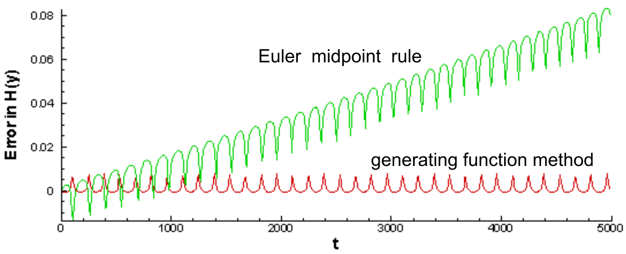

Figure 7: Comparisons of the errors in $H(y)$ from the two methods within long time.

Example 4.3 (cf. [4])**.** Consider the Volterra lattice system

$$y_i'(t)=y_i(y_{i+1}-y_{i-1}), \qquad i=1,2,\cdots,n, \tag{4.6}$$

with the periodic boundary conditions $y_{n+i}=y_i$, where $y_i>0$, $i=0,1,2,\cdots$, and n is even.

The system (4.6) can be written as

$$y'(t)=B(y)\nabla H(y)$$

with the Hamiltonian function

$$H(y)=\sum_{i=1}^{n} y_i,$$

and the Poisson structure matrix

$$B(y)=\begin{pmatrix} 0 & y_1y_2 & 0 & \cdots & 0 & -y_1y_n \\ -y_1y_2 & \ddots & \ddots & \ddots & & 0 \\ 0 & \ddots & 0 & y_iy_{i+1} & \ddots & \vdots \\ \vdots & \ddots & -y_iy_{i+1} & \ddots & \ddots & 0 \\ 0 & & \ddots & \ddots & 0 & y_{n-1}y_n \\ y_1y_n & 0 & \cdots & 0 & -y_{n-1}y_n & 0 \end{pmatrix}.$$

We have

$$B(y)=\begin{pmatrix} y_1 & & \\ & \ddots & \\ & & y_n \end{pmatrix} K \begin{pmatrix} y_1 & & \\ & \ddots & \\ & & y_n \end{pmatrix},$$

here

$$K=\begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 0 & -1 \\ -1 & 0 & 1 & \ddots & & 0 & 0 \\ 0 & -1 & 0 & \ddots & \ddots & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & & \ddots & \ddots & \ddots & 1 & 0 \\ 0 & 0 & & \ddots & -1 & 0 & 1 \\ 1 & 0 & 0 & \cdots & 0 & -1 & 0 \end{pmatrix}.$$

It is obvious that K is a singular skew-symmetric matrix.

Now we take the initial conditions

$$y_i(0)=1+\frac{1}{2n^2}\frac{1}{\cosh^2(x_i)}, \quad x_i=-1+(i-1)\frac{1}{2n}, \quad i=1,2,\cdots,n,$$

with the dimensions $n=20,40,80$, and the time stepsizes $\tau=0.2,0.1,0.05$ over the time interval $t\in[0,2000]$. Consider the four first integrals of (4.6)

$$H_1(y)=\sum_{i=1}^{n}y_i,\qquad\qquad H_0(y)=\frac{1}{2}\sum_{i=1}^{n}\ln y_i,$$
$$I_q(y)=\sum_{i=1}^{n}\Big(\frac{1}{2}y_i^2+y_iy_{i+1}\Big),\qquad I_c(y)=\sum_{i=1}^{n}\Big(\frac{1}{3}y_i^3+y_iy_{i+1}(y_i+y_{i+1}+y_{i+2})\Big),$$

where H_0 is the Casimir function of (4.6). Using the second-order generating function method (3.3), we obtain the following scheme

$$\begin{pmatrix}\ln|y_1^{(k+1)}|\\ \vdots\\ \ln|y_n^{(k+1)}|\end{pmatrix}-\begin{pmatrix}\ln|y_1^{(k)}|\\ \vdots\\ \ln|y_n^{(k)}|\end{pmatrix}=\tau K\begin{pmatrix}\theta_1\sqrt{y_1^{(k)}y_1^{(k+1)}}\\ \vdots\\ \theta_n\sqrt{y_n^{(k)}y_n^{(k+1)}}\end{pmatrix}. \tag{4.7}$$

Consider the preservation of numerical solution for the first integrals, and the errors are given in Table 1. Here we use the average errors as $\sqrt{\sum_{i=1}^{N}(I^i-I_0)^2/N}$, where N is the number of time steps, I_0 is a first integral of the analytical solution, and I_i denotes the value of first integral function at time t_i with the variable given by generating function method.

In Table 1, we can see that the numerical solutions have a nice approximation accuracy, especially in Casimir function $H_0(y)$. In fact, the scheme (4.7) preserves $H_0(y)$ accurately, and the factors affecting errors in $H_0(y)$ is only caused by the round-off errors.

Table 1: Average errors of first integrals.

n	τ	H_1	H_0	I_q	I_c
20	0.5	1.9167e-14	2.1380e-17	2.0321e-13	1.4091e-12
20	0.2	2.4958e-15	4.5900e-17	2.7520e-14	1.9199e-13
20	0.1	4.6343e-16	4.8809e-17	5.0923e-15	3.5587e-14
40	0.5	6.0083e-16	8.7867e-17	7.1169e-15	5.1242e-14
40	0.2	1.6057e-16	6.0697e-17	1.2380e-15	7.5692e-15
40	0.1	1.0882e-16	5.7388e-17	5.8318e-16	2.0410e-15
80	0.5	6.0261e-16	2.7149e-16	2.0246e-15	5.7740e-15
80	0.2	6.5632e-16	3.3096e-16	2.1203e-15	5.8995e-15
80	0.1	5.5292e-16	2.5232e-16	1.4605e-15	5.6903e-15

Then, we apply the generating function methods to solve other generalized Hamiltonian systems.

Example 4.4 (cf. [13], §8.2)**.** Consider the Robbins model system

$$y_1'(t)=-y_2y_3+\varepsilon(1-y_1),\quad y_2'(t)=y_1y_3-\varepsilon y_2,\quad y_3'(t)=y_2-\varepsilon\sigma y_3, \tag{4.8}$$

where σ, ε are constants, and ε is a perturbation parameter.

Here we just consider the system (4.8) without disturbance, i.e., $\varepsilon=0$, (4.8) can be written as the three dimensional generalized Hamiltonian system

$$\begin{pmatrix} y_1'(t) \\ y_2'(t) \\ y_3'(t) \end{pmatrix} = \begin{pmatrix} 0 & 0 & -y_2 \\ 0 & 0 & y_1 \\ y_2 & -y_1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ y_3 \end{pmatrix} \tag{4.9}$$

with the Hamiltonian function

$$H(y)=y_1+\frac{1}{2}y_3^2$$

and the Casimir function

$$C(y)=y_1^2+y_2^2.$$

Assume that $C(y)\neq 0$. Let

$$z_1=h_1(y)=y_3,\quad z_2=h_2(y)=\begin{cases} \arctan\dfrac{y_2}{y_1}, & \text{if } y_1\neq 0, \\ \dfrac{1}{2}\pi\cdot\mathrm{sgn}(y_2), & \text{if } y_1=0, \end{cases} \quad z_3=h_3(y)=y_1^2+y_2^2.$$

We have

$$y_1=\lambda\sqrt{z_3}\cos z_2,\quad y_2=\lambda\sqrt{z_3}\sin z_2,\quad y_3=z_1,$$

where $\lambda=1$ when $y_1(0)\geq 0$ and $\lambda=-1$ when $y_1(0)<0$. Using the second-order generating function method (3.3), we obtain the algorithm

$$\begin{pmatrix} h_1(y^{(k+1)}) \\ h_2(y^{(k+1)}) \\ h_3(y^{(k+1)}) \end{pmatrix} - \begin{pmatrix} h_1(y^{(k)}) \\ h_2(y^{(k)}) \\ h_3(y^{(k)}) \end{pmatrix} = \tau \begin{pmatrix} 2\sin\dfrac{h_2(y^{(k+1)})+h_2(y^{(k)})}{2} \\ \dfrac{1}{2}(h_1(y^{(k+1)})+h_1(y^{(k)})) \\ 0 \end{pmatrix}.$$

Taking the initial value (1,1,1) and the time stepsize $\tau=0.1$ over the time interval [0,100], we obtain the numerical solution in Fig. 8 and the error analysis in Figs. 9-11.

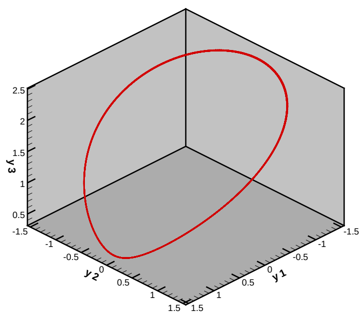

Figure 8: Numerical results of (4.8) without disturbance.

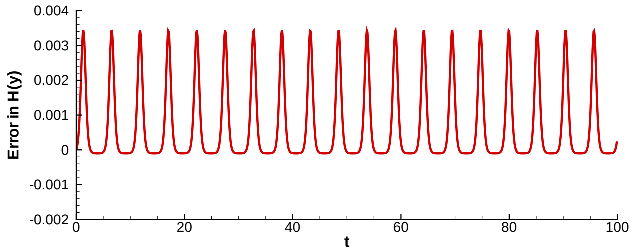

Figure 9: Errors in $H(y)$ of Numerical solutions of (4.8) with stepsize $\tau=0.1$.

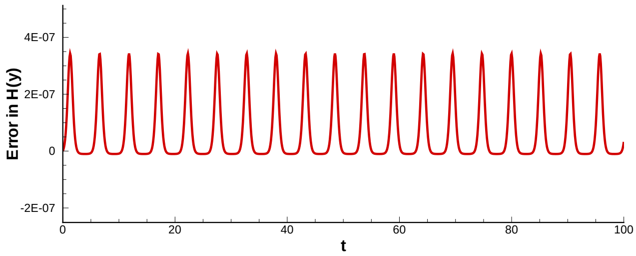

Figure 10: Errors in $H(y)$ of Numerical solutions of (4.8) with stepsize $\tau=0.001$.

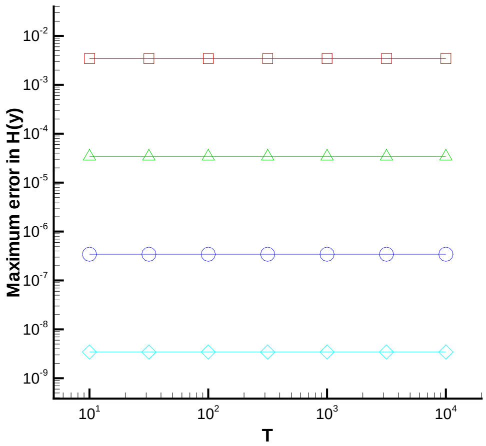

Figure 11: Maximum errors in $H(y)$ from the generating function method with different time intervals $[0,T]$ and stepsizes. $\square$: $\tau=0.1$, $\triangle$: $\tau=0.01$, $\bigcirc$: $\tau=0.001$, $\diamond$: $\tau=0.0001$.

Example 4.5 (cf. [1,12]). Consider the Euler equation which describes the motion of rigid body

$$y_1'(t)=\frac{I_2-I_3}{I_2I_3}y_2y_3, \quad y_2'(t)=\frac{I_3-I_1}{I_3I_1}y_3y_1, \quad y_3'(t)=\frac{I_1-I_2}{I_1I_2}y_1y_2, \tag{4.10}$$

where (y_1,y_2,y_3) denotes the angular momentum vector of the rigid body's rotation around a fixed point, and the constants I_1, I_2, I_3 denote the eigenvalues of the Rigid body's inertia tensor matrix.

Then, the Euler equation (4.10) can be written as the generalized Hamiltonian system

$$\begin{pmatrix} y_1'(t) \\ y_2'(t) \\ y_3'(t) \end{pmatrix} = \begin{pmatrix} 0 & -y_3 & y_2 \\ y_3 & 0 & -y_1 \\ -y_2 & y_1 & 0 \end{pmatrix} \begin{pmatrix} y_1/I_1 \\ y_2/I_2 \\ y_3/I_3 \end{pmatrix} \tag{4.11}$$

with the Hamiltonian function

$$H(y) = \frac{1}{2}\sum_{i=1}^{3} y_i^2/I_i$$

and the Casimir function

$$C(y) = \frac{1}{2}(y_1^2+y_2^2+y_3^2).$$

Assume that $y_1^2+y_2^2 \neq 0$. Let

$$z_1 = h_1(y) = y_1, \quad z_2 = h_2(y) = \begin{cases} \arctan\dfrac{y_3}{y_2}, & \text{if } y_2 \neq 0, \\ \dfrac{1}{2}\pi\cdot\mathrm{sgn}(y_3), & \text{if } y_2 = 0, \end{cases} \quad z_3 = h_3(y) = \frac{1}{2}(y_1^2+y_2^2+y_3^2).$$

We have

$$y_1 = z_1, \quad y_2 = \lambda\sqrt{2z_3 - z_1^2}\cos z_2, \quad y_3 = \lambda\sqrt{2z_3 - z_1^2}\sin z_2,$$

where $\lambda = 1$ when $y_2(0) \geq 0$ and $\lambda = -1$ when $y_2(0) < 0$. Using the second-order generating function method, we obtain the following Poisson algorithm

$$\begin{pmatrix} h_1(y^{(k+1)}) \\ h_2(y^{(k+1)}) \\ h_3(y^{(k+1)}) \end{pmatrix} - \begin{pmatrix} h_1(y^{(k)}) \\ h_2(y^{(k)}) \\ h_3(y^{(k)}) \end{pmatrix} = \tau \begin{pmatrix} 2\sin\dfrac{h_2(y^{(k+1)})+h_2(y^{(k)})}{2} \\ \dfrac{1}{2}(h_1(y^{(k+1)})+h_1(y^{(k)})) \\ 0 \end{pmatrix}.$$

We take $I_1 = 3$, $I_2 = 2$, $I_3 = 1$, the initial values $(1/\sqrt{2},1/\sqrt{2},0)$, $(-1/\sqrt{2},1/\sqrt{2},0)$, $(0,1/\sqrt{2},-1/\sqrt{2})$, and the time stepsize $\tau = 0.1$ over the time interval $[0,100]$. The numerical solutions are given in Fig. 12, and the error analysis are given in Figs. 13-15 with the initial value $(1/\sqrt{2},1/\sqrt{2},0)$.

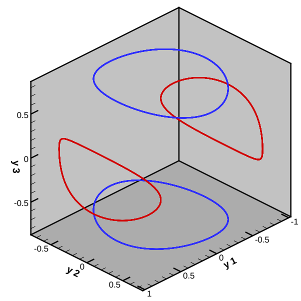

Figure 12: Numerical results of Euler equations (4.10) with different initial values.

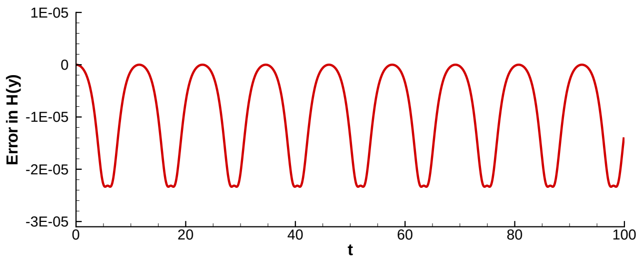

Figure 13: Errors in $H(y)$ of numerical solution for Euler equations (4.10) with stepsize $\tau=0.1$.

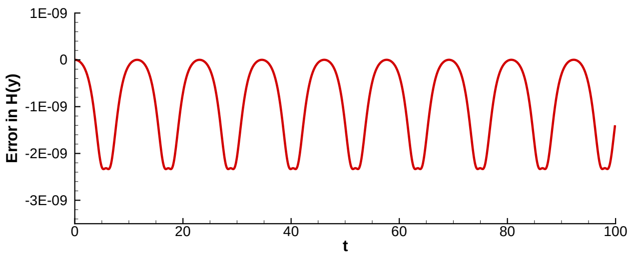

Figure 14: Errors in $H(y)$ of Numerical solutions of Euler equations (4.10) with stepsize $\tau=0.001$.

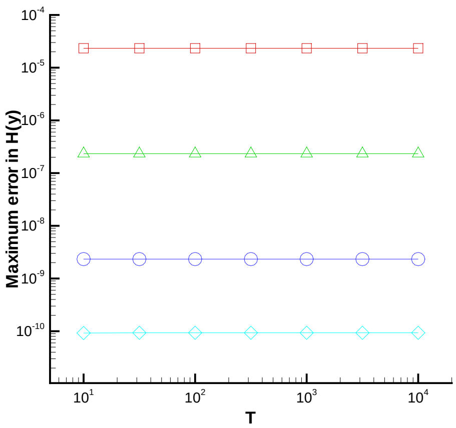

Figure 15: Maximum errors in $H(y)$ from the generating function method with different time intervals $[0,T]$ and stepsizes. $\square:\tau=0.1$, $\triangle:\tau=0.01$, $\bigcirc:\tau=0.001$, $\diamond:\tau=0.0001$.

5 Conclusions

In this paper, we use the generating function methods to construct the Poisson scheme for solving coefficient-varying generalized Hamiltonian systems, and some numerical examples are given. These methods have the nice approximation to the analytical solution, and they can solve a large class of generalized Hamiltonian systems. Using the generating function methods of any order, we need to solve a nonlinear algebraic equation of dimension n, and the computational costs are less than the high-order symplectic Runge-

Kutta methods. However, the high-order generating function methods need to give the high-order derivation of Hamiltonian function, this brings much trouble. [15] gives a easy way to solve Hamilton-Jacobian equations, and it can be applied to the methods in this paper.

Acknowledgments

This work is supported by projects NSF of China (11271311), Program for Changjiang Scholars and Innovative Research Team in University of China (IRT1179), the Aid Program for Science and Technology, Innovative Research Team in Higher Educational Institutions of Hunan Province of China, and Hunan Province Innovation Foundation for Postgraduate (CX2011B245).

References

[1] M. AUSTIN, P. S. KRISHNAPRASAD AND L. S. WANG, *Almost poisson integration of rigid body systems*, J. Comput. Phys., 107 (1993), pp. 105–117.
[2] C. M. CHEN, Q. TANG AND S. F. HU, *Finite element method with superconvergence for nonlinear Hamiltonian systems*, J. Comput. Math., 29 (2011), pp. 167–184.
[3] D. COHEN AND E. HAIRER, *Linear energy-preserving integrators for Poisson systems*, BIT Numer. Math., 51 (2011), pp. 91–101.
[4] T. ERGENÇ AND B. KARASÖZEN, *Poisson integrators for Volterra lattice equations*, Appl. Numer. Math., 56 (2005), pp. 870–887.
[5] K. FENG AND M. Z. QIN, Symplectic Geometric Algorithms for Hamiltonian Systems, Springer-Verlag, 2010.
[6] K. FENG AND D. L. WANG, *Symplectic difference schemes for Hamiltonian systems in general symplectic structures*, J. Comput. Math., 9 (1991), pp. 86–96.
[7] K. FENG, H. M. WU AND M. Z. QIN, *Symplectic difference schemes for the linear Hamiltonian canonical systems*, J. Comput. Math., 8 (1990), pp. 371–380.
[8] K. FENG, H. M. WU, M. Z. QIN AND D. L. WANG, *Construction of canonical difference schemes for Hamiltonian formalism via generating functions*, J. Comput. Math., 7 (1989), pp. 71–96.
[9] E. HAIRER, C. LUBICH AND G. WANNER, Geometric Numerical Integration, Structure-Preserving Algorithms for Ordinary Differential Equations, Springer-Verlag, 2002.
[10] B. HERNÁNDEZ-BERMEJO AND V. FAIRÉN, *Hamiltonian structure and Darboux Theorem for families of generalized Lotka-Volterra systems*, J. Math. Phys., 39 (1998), pp. 6162–6174.
[11] J. L. HONG AND Y. J. SUN, *Generating functions of multi-symplectic RK methods via DW Hamilton-Jacobi equations*, Numer. Math., 110 (2008), pp. 491–519.
[12] L. O. JAY, *Preserving Poisson structure and orthogonality in numerical integration of differential equations*, Computers Math. Appl., 48 (2004), pp. 237–255.
[13] J. B. LI, X. H. ZHAO AND Z. R. LIU, Theory and Application of Generalized Hamiltonian Systems, Science Press, 1994.
[14] X. S. LIU, Y. Y. QI, J. F. HE AND P. Z. DING, *Recent progress in symplectic algorithms for use in quantum systems*, Commun. Comput. Phys., 2 (2007), pp. 1–53.

[15] J. MAKAZAGA AND A. MURUA, *A new class of symplectic integration schemes based on generating functions*, Numer. Math., 113 (2009), pp. 631–642.

[16] J. E. MARSDEN AND S. T. RATIU, Introduction to Mechanics and Symmetry, Springer-Verlag, 1994.

[17] P. J. OLVER, Applications of Lie Groups to Differential Equations, Springer-Verlag, 1993.

[18] T. TANG AND X. XU, *Accuracy enhancement using spectral postprocessing for differential equations and integral equations*, Commun. Comput. Phys., 5 (2009), pp. 779–792.

[19] D. L. WANG, *Poisson difference schemes for Hamiltonian systems on Poisson manifolds*, J. Comput. Math., 9 (1991), pp. 115–124.

[20] S. Y. ZHANG AND Z. C. DENG, Geometric Integration Theory and Application of Nonlinear Dynamical Systems, Northwestern Polytechnical University Press, 2005.